European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS 2000
Barcelona, 11-14 September 2000
© ECCOMAS

IMPROVING DOMAIN DECOMPOSITION FOR MULTIFRONTAL RESOLUTION OF MECHANICAL PROBLEMS: FINITE ELEMENT REORDERING AND LOAD BALANCING ISSUES

J.P. Boufflet[*], P. Breitkopf[†], S. Negre[‡], A. Rassineux[†], M. Vayssade[†]

[*]Centre de Recherches
UMR CNRS 6599 HEUDIASYC
BP 529, 60205 Compiègne, France
Email: jean-paul. boufflet@utc.fr

[‡]INSSET
48 rue Raspail, 02100 Saint-Quentin, France
Email: stephane.negre@u-picardie.fr

[†] Centre de Recherches
UMR CNRS 6066 ROBERVAL
BP 529, 60205 Compiègne, France
Email: piotr.breitkopf@utc.fr

Key words: Load balancing, graph partitioning, multifrontal solver.

Abstract. *This discusses for the load balancing issues in the scope of the resolution of systems of linear equations encountered in large scale finite element problems. We first introduce the principle of a multifrontal solver. Then, some load balancing heuristics used by the numerical mechanics community are presented. We use an estimator of the computing time of the tasks of the multifrontal solver to measure the efficiency of our heuristics. Our tests on a collection of finite element meshes show the efficiency of the proposed method.*

1 INTRODUCTION

Finding an optimal order of the columns of a matrix in order to minimize the fill-in is NP-hard[29]. Minimizing the maximal bandwidth of an upper triangular matrix and minimizing its maximum frontwidth are also NP-hard[9,22,1]. However these discrete optimization problems are frequently encountered and occur particularly in solving linear systems issued from finite element models. The challenge is to design methods which provide good solutions for these kinds of reordering problems in a reasonable amount of time.

The parallel resolution of such systems consists in decomposition of graphs of finite elements into an equilibrated set of tasks. The tasks are then affected to the individual processors. In general, this problems are also N-P hard[9]. On the other hand, the global computational cost of the optimization method has to be considered when attempting to solve practical engineering problems.

The work we present concerns the reordering of finite elements for the frontal method[13]. The computing time of the frontal solver depends on the treatment order of the finite elements. We estimate it accurately with a new objective function. The improvements we have made on classical reordering heuristics[4,10,26] concern, on average, the quality of the solutions with a similar cost[19,20]. The Everstine collection of finite element meshes[5] is used as a benchmark. The Everstine examples are given in the matrix form. In order to use them for the frontal method, we had to reconstruct equivalent finite element problems resulting in the same topology of matrices. The criteria we consider are the profile and the estimator we have introduced for the frontal solver. The solutions thus obtained are close to the optimal ones. Our improvements of heuristics can be usefully applied to practical problems.

The second aspect of our work concerns the load balancing issues[2,8,16,23,28]. We propose a new heuristic that combines reordering techniques and load balancing iterative method that uses an accurate measure of the computational time for the multifrontal solver[20,21]. We refere our work to that of Malone[17] and Fahrat[6,7,12]. Three criteria are used. The first is the number of unknowns at the interfaces. The second one measures the unbalance of subdomains using our estimators of the computing time. The third one is the estimation of the overall solution time under a set of hypothesis (completion time criterion).

2 RESULTS

We present here the results obtained using the approach coupling both the subdomain decomposition heuristics and the renumbering schemes.

The Figure 1 presents the size of subdomain boundaries (total number of interface unknowns) for four methods of decomposition as applied to eight meshes when split into 4 subdomains. The meshes are identified by the respective numbers of nodes. The overall lengths of all subdomain boundaries are given for the two proposed methods: MB+ (improved Minimum Bandwidth) and ECMF - equilibrated load balancing for multifrontal method. The results for FLH - Fahrat, Lesoinne, Hsieh and MB - Minimum Bandwidth are given for comparison. We see, that MB+ and ECMF perform uniformly well.

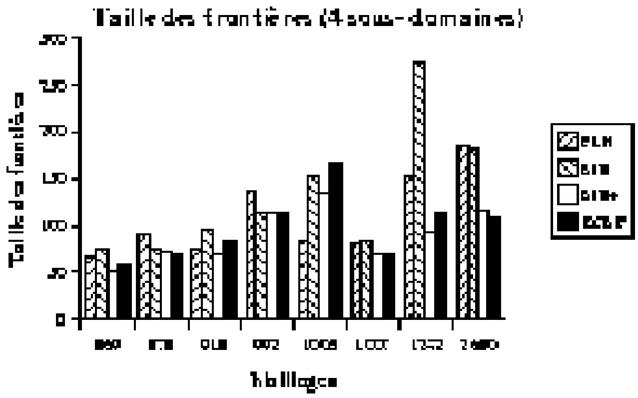

Figure 1: Total number of interface unknowns for four methods of decomposition as applied to eight meshes; FLH - Fahrat, Lesoinne, Hsieh; MB - Minimum Bandwidth; MB+ - improved Minimum Bandwidth; ECMF - equilibrated load balancing for multifrontal method

The Figure 2 illustrates the load balancing obtained with the same set of examples as in Figure 1. The ECMF method gives the best results which may be explained by the use of the criteria of the condensation time rather then equilibrating the number of nodes or elements in the subdomains. The cost of the criteria involves that of the virtual condensation time of each subdomain and varies linearly with the number of elements.

The coupled effect of the two steps illustrated in the figures 1 and 2 results in a significant improvent of the total resolution time estimation consisting of the condensation phase and that of the interface problem as given in the Figure 3.

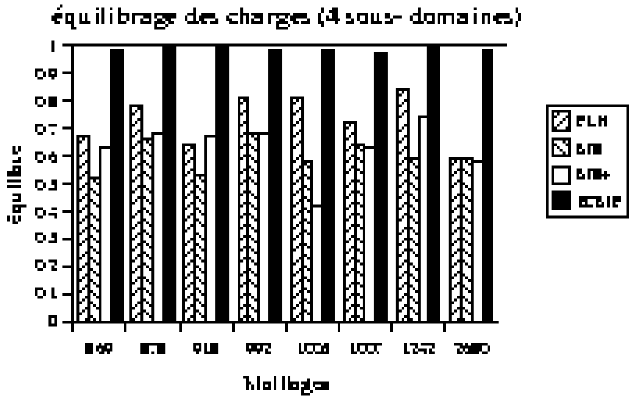

Figure 2: load balancing obtained with the same subset of examples as Figure 1

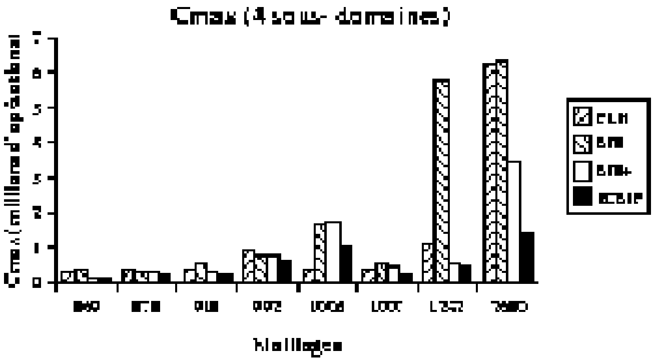

Figure 3: Improvent of the total resolution time estimation for the four methods

The Figure 4 shows the evolution of the overall estimation versus the number of subdomains for the 2680 node problem. One can easily see that there exists a limit numer of subdomains over which further decomposition does not diminish the solution time. In fact, the maximum number of subdomains for a frontal solver is equivalent to the number of finite elements and the whole solution task is reported in this case on the interface problem.

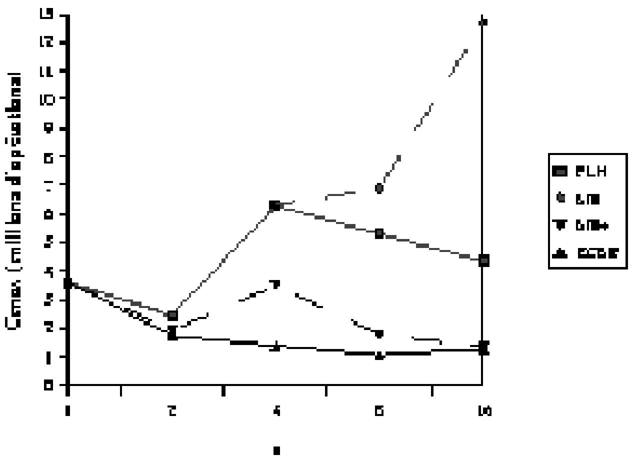

Figure 4: Evolution of the overall estimation versus the number of subdomains for the 2680 node problem

The following three tables give the mean results obtained at the 30 Everstine cases, for the analyzed methods when applied to a series of decompositions into 2, 4, 8 and 16 subdomains. We see again, that the size of the interface problem alone is either equivalent or slightly better for the proposed methods, excepted for the FLH algorithm with high number of subdomains. The mean values are penalized then by the results on the small examples where MB+ and ECMF are outperformed by the other methods.

s	MB+/FLH	MB+/MB	ECMF/FLH	ECMF/MB
2	+14,5%	+9,2%	+6,2%	+0,7%
4	+13,9%	+20,3%	+11,3%	+18%
8	+1,1%	+24%	-0,3%	+22,8%
16	-11,9%	+23,2%	-12,3%	+22,7%

Table 1 : Everage size of the boundaries for the whole set of examples - positive values indicate the gain in the interface size

The comparison MB+/FLH in the Table 2 is due to the criteria used for the load balancing when decomposing the domain by the different methods. The FLH gives the subdomains of minimal interface for maximum number of internal nodes. In ECMF case, the condensation time estimation is used as the unbalance measure. This explains the performance of the method.

s	MB+/FLH	MB+/MB	ECMF/FLH	ECMF/MB
2	+2,4%	+2%	+11,9%	+11,5%
4	-4,8%	+6,1%	+27,7%	+36,1%
8	-0,1%	+4%	+29,1%	+33,1%
16	+3,4%	+13,3%	+25,4%	+35,1%

Table 2 : Load balancing criterion.

The Table 3 gives the mean global resolution times taking into account the condensation of the subdomains and the resolution of the interface problem. The best results are obtained in all cases with ECMF and are due to an equilibrated decomposition while preserving reasonable interface problem size.

s	MB+/FLH	MB+/MB	ECMF/FLH	ECMF/MB
2	+14,6%	+12,6%	+24,2%	22,3%
4	+14,7%	+30,2%	+40,3%	52,5%
8	+20,1%	+44,4%	+31,2%	52,2%
16	+20,8%	+47,9%	+24,5%	51,1%

Table 3 : Evaluation of the completion time C_{max} (global resolution time).

3 CONCLUSIONS

The results presented in this paper have to be handled with care as they are based on evaluation of the performance for the benchmark tests.

The use of the Everstine collection in both aspects of our research is guided by the need of comparison with the previous work in the domain. The relatively small size of examples makes experimenting easier while the further work on validation of the results on large scale problems is still needed.

REFERENCES

[1] S. Arnborg, "Efficient Algorithms For Combinatorial Problems on Graphs With Bounded Decomposability: a Survey", *BIT nordisk tidskrift for informationsbehandling*, 25, 2-23, (1985).

[2] S.T. Barnard and H.D. Simon, "A Fast Multilevel Implementation of Recursive Spectral Bisection for Partitioning Unstructured Problems", *Concurency: Practice and Experience*, 6, 101-107, (1994).

[3] E. Cuthill and J. McKee, "Reducing the bandwidth of sparse symmetric matrices", *Proc. 24th National Conference of the Association for Computing Machinery*, 157-172, (1969).

[4] I. S. Duff and J. K. Reid and J. A. Scott, "The Use of Profile Reduction Algorithms with a Frontal Code", *Int. J. for Num. Meth. in Eng.* , 28, 2555-2568, (1989).
[5] G. C. Everstine, "A Comparison of Three Resequencing Algorithms for the Reduction of Matrix Profile and Wavefront", *Int. J. for Num. Meth. in Eng.*, 14, 837-853, (1979).
[6] C. Farhat, "A Simple and Efficient Automatic FEM Domain Decomposer", *Computers and Structures*, 28, 579-602, (1988).
[7] C. Farhat and M. Lesoinne, "Automatic Partioning of Unstructured Meshes for the Parallel Solution of Problems in Computational Mechanics", *Int. J. for Num. Meth. in Eng.*, 36, 745-764, (1993).
[8] C.M. Fiduccia and R. M. Mattheyses, "A Linear-time Heuristic for Improving Network Partitions", *Proceedings of the 19th Design Automation Conference*, 175-181, (1982).
[9] M. R. Garey and D. S. Johnson, "Computers and Intractability : a Guide to the Theory of NP-Completeness", *W. H. Freeman*, (1979).
[10] N. E. Gibbs and W.G. Poole and P. K. Stockmeyer, "An Algorithm for Reducing the Bandwidth and the Profile of a Sparse Matrix", *SIAM J. Numer. Anal.*, 13, 236-250, (1976).
[11] B. Hendrickson and R. Leland, "An Improved Spectral Graph Partitioning for Mapping Parallel Computations", *SIAM Journal on Scientific Computing*, 16, 452-469, (1995).
[12] S.H. Hsieh, "A Mesh Partitioning Tool to Parallel Finite Element Processing", *ICPADS'94 IEEE*, (1994).
[13] B. M. Irons, "A Frontal Solution Program for Finite Element Analysis", *Int. J. for Num. Meth. in Eng.*, 2, 5-32, (1970).
[14] G. Kapyris and V. Kumar, "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs", *SIAM Journal on Scientific Computing*, 20, 359-392, (1998).
[15] G. Kapyris and V. Kumar, "A Parallel Algorithm for Multilevel Graph Partitioning and Sparse Matrix Ordering", *Journal of Parallel and Distributed Computing*, 48, 64-70, (1998).
[16] B.W. Kernighan and S. Lin G, "An Efficient Heuristic Procedure for Partitioning Graphs", *BELL System Technical Journal*, February, (1970).
[17] J.G. Malone, "Automated Mesh Decomposition and Concurrent Finite Element Analysis For Hypercube Multiprocessors Computers", *Comp. Meth. Appl. Mech. Eng.*, 70, 27-58, (1988).
[18] S. Negre and J.P. Boufflet, "Toward an integrated design for large systems: New finite element reordering methods for the frontal solver", *2nd International Conference on Integrated Design and Manufacturing in Mechanical Engineering*, 1, 315-323, (1998).
[19] S. Negre and J.P. Boufflet, "Two New Approaches for Ordering Finite Element in the Frontal Method Context", *Submitted to Revue Européenne des Eléments finis*, (1999).
[20] S. Negre, "Optimisation de la méthode multifrontale en vue de sa parallélisation", P*hD Thesis*, Université de Technologie de Compiègne, (1997).

[21] S. Negre and J.P. Boufflet, “Partitionnement de maillage et équilibrage de charges pour un solveur parallèle multifrontal”, *RenPar'10, Rencontre Francophone du Parallélisme des Architectures et des Systèmes*, 187-190, (1998).
[22] C. H. Papadimitriou, “The NP-Completeness of the Bandwidth Minimization Problem”, *Computing*, 16, 263-270, (1976).
[23] F. Pellegrini, “Placement Statique par Bipartitionnement Récursif Conjoint des Graphes de Processus et d'Architecture”, *RenPar'6, Rencontre Francophone du Parallélisme*, 41-44, (1994).
[24] A. Razzaque, “Automatic Reduction of Frontwidth for Finite Element Analysis”, *Int. J. for Num. Meth. in Eng.*, 15, 1315-1324, (1980).
[25] J.A. Scott, “On Ordering Elements for a Frontal Solver”, *Commun. Numer. Meth. Eng.*, 15, 309-323, (1999).
[26] S. W. Sloan, “FORTRAN Program for Profile and Wavefront Reduction”, *Int. J. for Num. Meth. in Eng.*, 28, 2651-2679, (1989).
[27] D. Vanderstraeten and R. Keunings, “Optimized partitioning of unstructured finite element meshes”, *Int. J. for Num. Meth. in Eng.*, 38, 989-1000, (1995).
[28] C. Walshaw and M. Cross and M.G. Everett and S. Johnson and K. McManus, “Partitioning & Mapping of Unstructured Meshes to Parallel Machine Topologies, Irregular'95: Parallel Algorithms for Irregularly Structured Problems”, *LCNS series*, 980, 121-126, (1995).
[29] M. Yannakakis, “Computing the Minimum Fill-in is NP-Complete”, *SIAM J. Algebraic Discr. Meth.*, 2, 77-79, (1981).