

On the efficiency of generic BE substructuring algorithms based on Krylov solvers

F. C. Araújo¹ and L. J. Gray²

Summary

This paper is concerned with the solution of boundary-element models based on substructuring. Structured matrix-vector products and the matrix-copy option are proposed to increase the efficiency of algorithms based on Krylov solvers. The former technique was designed to avoid the excessive number of conditional tests during solver iterations, and the latter one, to avoid the repeated calculation of coefficient matrices for identical subregions. Potential applications of the algorithm to composite materials, and to develop parallel codes, are noted.

Introduction

Mainly in the 90s, iterative Krylov solvers began to be widely considered in the development of computer codes to solve engineering problems, including those based on Boundary Element Methods. The major advantage of this kind of solver is their efficiency for solving large-order systems and their suitability for developing parallel codes. In Computational-Fluid-Dynamics (CFD) simulations, for instance, they have been commonly applied to develop parallel finite-element codes [1-3].

For Boundary Element Methods, Krylov solvers have also been successfully applied, in particular for substructuring algorithms [4-5]. In this connection, optimized data structures for a generic number of coupled subdomains, with complete exclusion of zero blocks, have been proposed [6-9]. This paper considers two further improvements to these algorithms: the structuring of matrix-vector products (SMVP) involved in the iterative solvers and the implementation of the matrix-copy option (MCO). The first technique was designed to exclude the many conditional tests, necessary when the matrix-vector products are left unstructured (UNSMVP). The second technique is designed to avoid calculating and assembling, repeatedly, the coefficient matrices for identical subregions.

¹Department of Civil Engineering, UFOP, 35400-000 Ouro Preto-MG, Brazil; currently postdoc at ORNL

²Computer Science and Mathematics Division, ORNL, Oak Ridge, P.O. Box 2008, USA

Generic Coupling of BE Models

As an iterative solver does not transform the coefficient matrix, formats can be devised that reduce memory requirements and solution CPU time. Here, a subregion-by-subregion data structure, described below, is used.

Generically (for n_s subregions), the corresponding subsystems of boundary element equations can be written as

$$\sum_{m=1}^{i-1} (\mathbf{H}_{im} \mathbf{u}_{mi} - \mathbf{G}_{im} \mathbf{p}_{im}) + \mathbf{A}_{ii} \mathbf{x}_i + \sum_{m=i+1}^n (\mathbf{H}_{im} \mathbf{u}_{im} + \mathbf{G}_{im} \mathbf{p}_{mi}) = \mathbf{B}_{ii} \mathbf{y}_i, \quad i = 1, n_s, \quad (1)$$

where, \mathbf{H}_{ij} and \mathbf{G}_{ij} denote the usual BE matrices obtained for source points pertaining to subregion Ω_i and associated respectively with the boundary vectors \mathbf{u}_{ij} and \mathbf{p}_{ij} at Γ_{ij} . Here Γ_{ij} with $i \neq j$ is the interface between Ω_i and Ω_j , and Γ_{ii} is the outer boundary of Ω_i . In the subregion-by-subregion data structure considered herein, these subsystems are not explicitly assembled into a global system, but separately stored. Thus, the many zero blocks, unavoidably present at these systems, are completely excluded [9]. Moreover, it is assumed that discontinuous boundary elements are employed, so that only compatibility and equilibrium conditions at the interfaces, given by

$$\begin{cases} \mathbf{u}_{ij} = \mathbf{u}_{ji} \\ \mathbf{p}_{ij} = -\mathbf{p}_{ji} \end{cases} \quad \text{at } \Gamma_{ij}, \quad (2)$$

have to be imposed for coupling the BE models. This simplify enormously the coupling algorithm since no traction-continuity condition together with undesirable additional subregions has to be used for simulating inner edges and corners [8].

A first topic addressed in this work is the structuring of the matrix-vector products embedded in the iterative solver. Here particularly, the J-BiCG solver [4, 6] is adopted, for which the matrix-vector products of the type $(\mathbf{D}^{-1} \mathbf{A}) \mathbf{p}_j$ and $(\mathbf{D}^{-1} \mathbf{A})^T \mathbf{p}_j^*$ have to be computed, where \mathbf{A} and \mathbf{D} are the global matrix of the coupled system and its corresponding diagonal matrix (Jacobi preconditioning), respectively. In previous versions of the code [9], unstructured matrix-vector products (UNSMVP) have been considered, what means that no column re-

ordering of the subregion matrices have been carried out, and so, conditional tests to identify the type of boundary condition present at each degree of freedom of each subregion (if interface or prescribed boundary value) had to be performed along all iterations of the solver.

In this paper, structured matrix-vector products (SMVP) are taken into account, meaning then that the columns of the matrix of a given subregion, say Ω_i , are actually re-ordered so as to get its coefficients grouped into different blocks. Three blocks are adopted: a first one associated with interfaces Γ_{ij} for which $i > j$, a second one associated with the outer boundary Γ_{ii} (at which boundary values are prescribed), and a last one associated with interfaces Γ_{ij} for which $i < j$ (see equation 2). As one can infer, the data structure shown in equation (1) was exactly that adopted. Thus, matrices **A** and **B** for the i -th subregion, after interchanging columns for introducing the boundary conditions at Γ_{ii} , have the following generic aspects:

$$\begin{array}{c}
 \begin{array}{ccccccc}
 & \text{block 1} & & \text{block 2} & & \text{block 3} & \\
 \leftarrow & & \rightarrow & \leftarrow & & \rightarrow & \\
 \mathbf{H}_i = [& \mathbf{H}_{i1} & \cdots & \mathbf{H}_{i,j-1} & \mathbf{A}_{ii} & \mathbf{H}_{i,j+1} & \cdots & \mathbf{H}_{im}] \\
 \mathbf{G}_i = [& \mathbf{G}_{i1} & \cdots & \mathbf{G}_{i,j-1} & \mathbf{B}_{ii} & \mathbf{G}_{i,j+1} & \cdots & \mathbf{G}_{in}]
 \end{array}
 \end{array} \quad (2)$$

In the code, each block is delimited by an initial and a final column, calculated for each subregion matrix.

A second topic included here is the implementation of matrix-copy option (MCO), which might be very useful for modeling problems which may involve several identical subregions such as identical inclusions or layers in composite materials (Figure 1). By considering the matrix-copy option, a matrix for a subdomain which is identical to another subdomain whose matrix has been already assembled has no longer to be calculated; it has to be only read from the memory space into where the matrix of an identical subdomain has been copied. In this way, CPU time and, possibly, allocating memory may be saved. Notice that for solid mechanics problems, rotation transformation of the copied matrices may be further necessary.

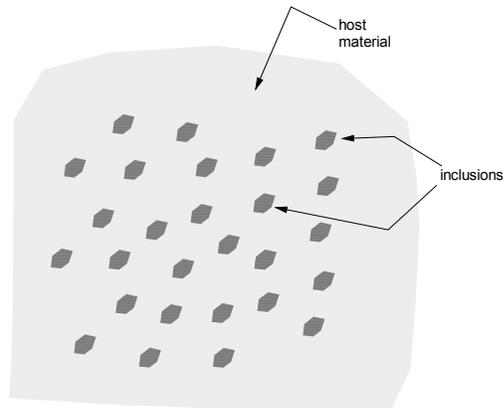


Figure 1. Composite material

Results and Discussion

To discuss further the importance of the techniques implemented, the holed rod shown in Figure 2 is analyzed. Its length is 1500 mm , and its cross section has external perimeter with 140 mm width and 225 mm height. The rod-wall thickness is 50 mm . The elasticity modulus and Poisson's ratio adopted are $E = 205 \times 10^3\text{ MPa}$ and $\nu = 0.30$ respectively. The rod is subjected to a normal pressure of 1.0 MPa . In Figure 2(a) and 2(b), the general aspect of the whole mesh and the coupled subdomains may be visualized. In Figure 2(c), details of each subregion are shown. In sum, the BE model is constituted of six identical subregions, each one with 144 boundary elements and 432 nodes, corresponding to total of 1296 degrees of freedom per subregion and of 7776 for the global system.

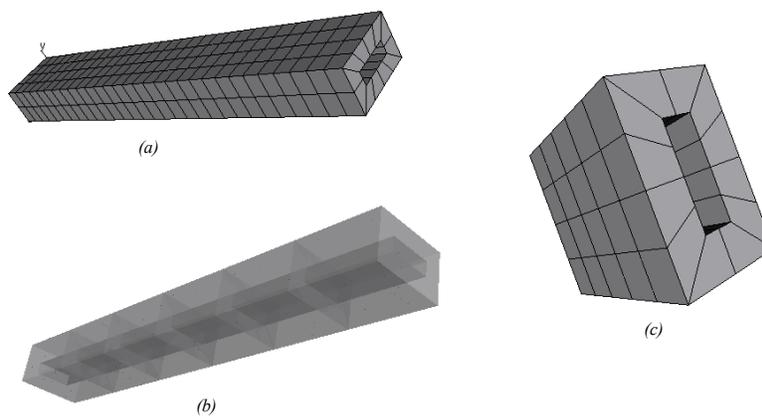


Figure 2. BE mesh adopted

For the rod dimensions chosen, the response is well described by the one-dimensional bar theory. The results agree pretty well with the analytical ones (error less than 0.4% at several nodes observed) and are not presented here. Results in terms of number of iterations and CPU times are given in Table 1, where *nit* denotes the number of iterations and *n* the system order. In Table 2, the matrix-assembly CPU times using and not using matrix-copy option (MCO) are shown. The solver has been stopped when $\|\delta\|_2 / \|\mathbf{Q}^{-1}\mathbf{b}\|_2 < 10^{-5}$.

Table 1. Efficiency parameters

	<i>nit</i>	<i>solver CPU time (s)</i>	<i>CPU time/nit</i>	<i>nit/n</i>
SMVP	2141	242	0.11	0.275
UNSMVP	2165	290	0.13	0.278

Table 2. Matrix-assembly CPU time (s)

	with no MCO	with MCO
SMPV	87	18

For this single test, a CPU time reduction per iteration of about 15% is obtained using the SMPV option (see Table 1). The small variation between the number of iterations using SMVP and UNSMVP is actually expected since the order of operations are different in these strategies, and arithmetic properties like the associativity of the addition operation are not valid in the presence of the finite representation of numbers on the computer. Therefore, to measure the CPU-time performance difference, the relation *CPU time/nit* was adopted. The relations *nit/n* measured hints the good performance of the J-BiCG solver. As seen, the CPU time for assembling the global coupled matrix not using MCO (matrix-copy option) is about five times that using MCO.

Conclusions

For the numerical test done, the efficiency of the coupling algorithm improved somewhat by adopting structured matrix-vector products (about 15%). Of course, for larger problems, this increase in efficiency will be more significant since more conditional tests have to be performed. Concerning the matrix-copy option (MCO), it considerably reduced, as expected, the assembly time of the global system, and might be very useful for modeling problems with several identical subdomains such as in some composite materials. Furthermore, if the matrix-copy option is allied with unstructured matrix-vector products, memory space may be additionally saved. As a natural consequence of the

domain-decomposition strategy considered, the algorithm is pretty suitable for developing parallelized BE codes.

Acknowledgment

This research was supported in part by the Applied Mathematical Sciences Research Program of the Office of Mathematical, Information, and Computational Sciences, U.S. Department of Energy under contract DE-AC05-00OR22725 with UT-Batelle, LLC, in part by the Brazilian Research Council, CNPq, and in part by the U.S. Department of Energy Higher Education Research Experiences (HERE) Program/ORISE.

References

1. Baum, J. D., Luo, H. and Löhner, R. (1994): "Edge-based finite element scheme for Euler equations", *AIAA J.*, Vol. 32, pp. 1183-1190.
2. Dutto, L. C., Lepage, C. Y. and Habashi, W.G. (2000): "Effect of the storage format of sparse linear systems on parallel CFD computations", *Comput. Meth. Appl. Mech. Engrg.*, Vol 188, pp. 441-453.
3. Catabriga, L., Coutinho, A. L. G. A. (2002): "Implicit SUPG solution of Euler equations using edge-based data structures", *Comput. Meth. Appl. Mech. Engrg.*, Vol. 191, pp. 3477-3490.
4. Mansur, W. J., Araújo, F. C. and Malaghini, J. E. B. (1992): "Solution of BEM systems of equations via iterative techniques", *Int. J. Numer. Methods Engrg.*, Vol. 33, pp. 1823-1841.
5. Prasad, K. G., Kane, J. H., Keyes, D. E. and Balakrishna, C. (1994): "Preconditioned Krylov solvers for BEA", *Int. J. Numer. Methods Engrg.*, Vol. 37, pp. 1651-1672.
6. Araújo, F. C., Martins, C. J. and Mansur, W. J. (2001): "An efficient BE iterative-solver-based substructuring algorithm for 3D time-harmonic problems in elastodynamics", *Engineering Analysis with Boundary Elements*, Vol. 25, pp. 795-803.
7. Araújo, F. C., Dors, C. and Mansur, W. J. (2003): "Dynamic cross-interaction between footings - 3D frequency-domain parallelized analysis via the BEM", In: *Proc. International Conference on Boundary Element Techniques IV*, Granada, Queen Mary, University of London: London, Vol. I, pp. 253-258.
8. Araújo, F. C., Dors, C., Martins, C. J. and Mansur, W.J. (2004): "New developments on BE/BE multi-zone algorithms based on Krylov solvers – Applications to 3D frequency-dependent problems", *J. Braz. Soc. Mech. Sc. & Engrg.*, Vol. 26, pp. 231-248.
9. Araújo, F. C., Silva, K. I. and Telles, J. C. F. (in press): "Generic domain decomposition and iterative solvers for 3D BEM problems", *Int. J. Numer. Methods Engrg.*