On the use of iterative solvers for generating efficient 3D multidomain boundary element algorithms

F.C. de Araújo, G.J. Belmonte, M.S.R. Freitas

Department of Civil Engineering/Universidade Federal de Ouro Preto
35400-000, Ouro Preto-MG, Brazil
EMail: fcelio@em.ufop.br

Abstract

In this work one aims to present an efficient algorithm for coupling a generic number of subregions discretized with boundary elements. The followed strategy does not base on condensing the field variables into certain variable blocks, as e.g. the interface variable ones, but instead considers directly the resulting sparse matrix of the coupled system. The main idea of the algorithm is that by using reliable iterative solvers it is then possible to develop multi-zone algorithms without condensing techniques such that, independently of the numbering scheme adopted, only the block matrices with non-zero coefficients are stored and manipulated during the analysis process and, furthermore, the CPU time for solving the system of equations is strongly decreased in comparison with necessary CPU times by using direct ones, as profit of the efficiency of iterative solvers, which become specially suitable for treatment of sparse matrix, is taken into account. Specifically, the Lanczos acceleration process, eventually preconditioned, was used in the formulation presented in this work. Details of the data organisation in the sparse system matrix and of the formulation of the adopted iterative solver, inclusive its convergence analysis, are discussed. The performance of the presented algorithm is then verified by determining the stress intensity factor (SIF) in a 3D elastostatic problem with crack and, with aid of the results presented in this paper (in terms of storage requirements, CPU times and precision) and also with aid of recently reported researches about iterative solvers, one aims to show concretely the efficiency of the proposed coupling strategy.
1 Introduction

Multi-domain boundary element algorithms, as well known, are used for treatment of nonhomogeneities in general problems or also for simulating in a simpler way domain cracks, thin barriers, etc. by means of The Boundary Element Method, being therefore a fundamental technique. In these cases, it is necessary to idealise the whole definition domain of the problem as an assembly of a series of homogeneous parts (subregions or subdomains) for which the boundary integral equation of the physical problem being considered can then be applied (Brebbia et al.[8] and Brebbia and Dominguez[9]). Following the normal BE discretizing process and with further consideration of coupling conditions on the interfaces between the subdomains a coupled system is obtained that furnishes the response for the whole domain.

As for a multidomain BE algorithm the originated coupled system matrix exhibits a great deal of blockmatrices with zero coefficients that should naturally not be processed during the analysis, some researches have been recently developed in order to get, concerning the storage and manipulation techniques of the matrix coefficients in coupling algorithms, optimised strategies (Bialecki et al.[7], Kane et al.[11], Santiago[17]). The task of developing such algorithms grounds normally on renumbering processes of the subregions, so that an optimum arrangement of the final matrix is obtained (Kane et al.[11], Rigby and Aliabadi[16]) and furthermore, direct solvers based on Gauss elimination have been employed for solving the resulting system of algebraic equations (Bialecki et al.[7], Santiago[17]). Yet it is known, that by analysing more complex coupled systems which consist of various subdomains, there is no way of avoiding completely the so-called fill-in effects, such that at least some originally zero blocks are filled with non-zero coefficients during the transformations of the direct procedure. Such coupled strategies can therefore not be termed as ideal ones.

In this article an algorithm for coupling a generic number of subregions that bases on the consideration of the global sparse matrix of the coupled system, without condensing the problem quantities into the variables on the interfaces, but also without taking into account the great deal of zero blocks appearing in the coupled system, is presented. With the proposed algorithm only the block matrices with non-zero coefficients are stored and manipulated during the analysis process and, in order still more to increase its efficiency, the already good established iterative solver based on the Lanczos acceleration process, eventually preconditioned, has been applied (see Araújo and Mansur[1], [2], Araújo[3] and Mansur et al.[13]). Details of the Lanczos process formulation applied to solve linear system of equations, inclusive its onvergence analysis, are presented, being also the way the blockmatrices are organised and stored in the coupled system for further manipulation in the solution phase commented.

With aid of the results presented in this article for determining the stress intensity
factor (SIF) in a 3D elasticity problem with crack and also with aid of results concerning iterative schemes for BE systems of equations reported in the last ten years by Araújo and Mansur[1], Araújo et al.[2], Araújo[3], Kane et al.[12], Mansur et al.[13], Barra et al.[4], Prasad et al.[15] among other researchers, it is tried to show that the multi-zone strategy proposed here is a very efficient one and should be followed in other coupling schemes for general problems. The results shown in this work are discussed in terms of CPU time, storage room requirements and accuracy.

2 The Multi-Domain Algorithm

In BEM formulations one comes in general to a system of equations of the following aspect, as well-known:

\[ Hu = Gp + f, \]  

which is in principle established for a homogeneous domain, being naturally necessary to be considered a series of such systems, if e.g. a nonhomogeneous domain should be analysed. Details on the physical problem will not be given here, then solely the coupled geometric definition of the problem domain will be highlighted, but, of course, the ideas followed in this work are equally applicable for general physical situations (stationary and transient ones). Particularly the application done in this work concerns the determination of SIF in 3D elasticity problems, as mentioned above.

Concerning the multi-domain algorithm properly proposed, just two points should be commented, namely, how should be organised the coefficient blocks of each subregion in the coupled system (whether condensed or not) and how should they be ideally stored. Defining a coefficient block as the submatrix of the coefficient matrix of a certain subregion that corresponds only to degrees of freedom of the external boundary or only to degrees of freedom of the intersection of this subregion with each other one (common surfaces) or with simultaneously each two, three or more other ones (curves and points), the questions above can then be discussed. In respect to the first question, the option adopted here was not to condense the field quantities in any variable blocks, but in allocating them in a global sparse matrix, at least implicitly, involving all the system unknowns. Thereby one avoids the necessary inversions of submatrices in order to get the condensed system. The second question was solved by storing, exclusively, the non-zero blocks in a work vector, beginning with the first column of the first non-zero block of the first subregion and so on, column by column, block by block, subregion by subregion. Some details on the storing procedure are that the coefficient blocks (in terms of global node numbers pertaining to them) are determined during the mesh generation in the pre-processing module of the computer code NAESY (program being developed by the research group of the authors) and, furthermore, two variables indicating the position of each block in the coupled sparse matrix of the whole system (in terms of its initial and final
lines and columns) and one other variable pointing the position of the first element of each block in the work vector are used.

![Subregions of body](image)

Figure 1. Subregions of body.

In order to give a more concrete idea of the multi-zone algorithm, the domain of Figure 1 with three subdomains will be considered (it could naturally be taken a domain with any number of substructures). By writing equation (1) for each subregion of the body in Figure 1 and by introducing the equilibrium and compatibility conditions at points of common variable blocks (not pertaining to the external boundary), the following noncondensed coupled system, organised in terms of the block variables,

\[
\begin{bmatrix}
H^1 & H^{1,2} & -G^{1,2} & H^{1,3} & -G^{1,3} & H^{1,2,3} & -G^{1,2,3} & 0 & 0 & 0 & 0 \\
0 & H^{2,1} & G^{2,1} & 0 & 0 & H^{2,1,3} & G^{2,1,3} & H^2 & H^{2,3} & -G^{2,3} & 0 \\
0 & 0 & 0 & H^{3,1} & G^{3,1} & H^{3,1,2} & G^{3,1,2} & 0 & H^{3,2} & G^{3,2} & H^3
\end{bmatrix}
\]

\[
\begin{bmatrix}
u^1 \\ u^{1,2} \\ p^1 \\ u^{1,3} \\ p^{1,2,3} \\ u^{2} \\ p^{2,3} \\ u^3
\end{bmatrix}
\]
is obtained, where the superscripts in equation (2) indicate the subregion or subregions to which the degrees of freedom pertain. After introducing the boundary conditions of the subdomains, it results then a system of equations of the form

\[ Ax = b, \quad (3) \]

usual in the Boundary Element Method. Though *implicitly* the matrix A is treated as the matrix on the l.h.s. of equation (2), *explicitly* this matrix is the work vector commented above, which contains only non-zero coefficients. The coupling of a generic number of subregions is therefore done, being no more discussion about complicated numbering schemes necessary.

### 3 The Iterative Solution Procedure

There are nowadays some important works emphasizing the efficiency of iterative solvers (Araújo and Mansur[1], Araújo *et al.*[2], Barra *et al.*[4], Davey and Rosidale[10], Kane *et al.*[12], Mansur *et al.*[13], Prasad *et al.*[15], Urekew and Rencis[18], Urekew and Rencis[19], Walker and Lee[20]), in which even the lack of convergence normally reported in anterior works (Bettess [5, 6]. Mullen and Rencis [14]) has not been found. In the study presented in this paper only the solution scheme based on the Lanczos tridiagonalization algorithm is considered (Wilkinson[21]).

In order to apply the Lanczos algorithm for solving algebraic systems of equations the iterative formula

\[ u^{n+1} = \rho_{n+1} \gamma_{n+1} r^n + \rho_{n+1} u^n + (1 - \rho_{n+1}) u^{n-1}, \quad (4) \]

which gives the system solution at iteration \( n+1 \), is considered, being the correspondent residue at this iteration, \( r^{n+1} \), consequently

\[ r^{n+1} = \rho_{n+1} (\gamma_{n+1} A r^n + r^n) + (1 - \rho_{n+1}) r^{n-1}. \quad (5) \]

The residue vector (5) has the aspect of the vectors generated from A-matrix by the Lanczos algorithm. This fact hints the consideration of vectors \( \bar{r}^{n+1} \) associated with \( A^T \) of the form

\[ \bar{r}^{n+1} = \bar{\rho}_{n+1} (\bar{\gamma}_{n+1} A^T \bar{r}^n + \bar{r}^n) + (1 - \bar{\rho}_{n+1}) \bar{r}^{n-1}. \quad (6) \]
By imposing now that vectors $\mathbf{r}^{n+1}$ and $\overline{\mathbf{r}}^{n+1}$ are actually Lanczos vectors, expressions for the parameters in iterative formulae (4), (5) and (6) can then be obtained. It results:

$$\gamma_{n+1} = \overline{\gamma}_{n+1} = \frac{\overline{\mathbf{r}}^{n,T} \mathbf{r}^n}{\overline{\mathbf{r}}^{n,T} \mathbf{A} \mathbf{r}^n} \quad (7a)$$

and

$$\rho_{n+1} = \overline{\rho}_{n+1} = \left[ 1 - \frac{\gamma_{n+1}}{\gamma_n} \cdot \frac{\overline{\mathbf{r}}^{n,T} \mathbf{r}^n}{\overline{\mathbf{r}}^{n-1,T} \mathbf{r}^{n-1}} \cdot \frac{1}{\rho_n} \right]^{-1} \quad (7b)$$

with $\rho_1 = 1$ and $\mathbf{r}^1 = \overline{\mathbf{r}}^1$. With these expressions two vector sets $\mathbf{r}^1, \mathbf{r}^2, \cdots$ and $\overline{\mathbf{r}}^1, \overline{\mathbf{r}}^2, \cdots$ are then generated for which the orthogonality property

$$\left( \mathbf{r}^i, \overline{\mathbf{r}}^j \right) = 0 \quad \text{if} \quad i \neq j \quad (8)$$

is verified, which is of fundamental importance for the reliability of the iterative scheme, at least, in what concerns the convergence aspect of the mathematical viewpoint, then being the generated set $\mathbf{r}^1, \mathbf{r}^2, \cdots$ linearly independent (see Wilkinson[21]) and pertaining the residue vector $\mathbf{r}^n$ to the $N$-dimensional Euclidean space $R^N$, where $N$ is the system order, so it must be found that $\mathbf{r}^{N+1} = 0$. Naturally as consequence of truncating errors introduced in the data processing in the computer, it may happen that convergence is not reached for $n < N$, mainly for ill-conditioned systems. This eventual lack of convergence has then been avoided by considering preconditioning matrices in the iterative schemes (Araújo and Mansur[1], Araújo et al.[2], Araújo[3], Barra et al.[4], Kane et al.[12], Mansur et al.[13], Prasad et al.[15]). It should be also observed that there is no difficulty in avoiding the manipulation of all zero blocks, as the most complex operation in the Lanczos algorithm (as well as in other iterative scheme in general) is a matrix-vector multiplication. In Araújo[3] it is given the complete derivation of all expressions involved in the Lanczos iterative solution scheme and correspondent algorithms.

4 Applications

The performance of the multi-zone algorithm with the Lanczos-based iterative solver (preconditioned and non-preconditioned) is observed by analysing the
plate with crack depicted in Figure 2(a), which is submitted to a static uniformly distributed tensile load of \( \sigma = 1.0 \text{ Ksi} \) at both ends of the plate in direction normal to the crack surface. The material constants are \( E=3.0 \times 10^4 \text{ Ksi} \) and \( v=0.25 \), and the adopted BE mesh is generated in such a way that the element lengths in direction normal to the crack front change obeying the relation: the biggest one is 10 times the smallest (near the crack front), being the other element lengths in that direction gradually adapted. In Figure 2(b) is shown the BE mesh for one subregion (one half of the plate), being the mesh for the other subregion similar. The whole mesh contains 2 substructures and 204 elements, what corresponds to 616 nodes and system order \( N=1848 \).

For evaluating the stress intensity factor (SIF) for 3D problems the expressions describing the displacement field in front to the crack (obtained by taking the plane strain state in the plane normal to the crack front) can be used (Sih and Liebowitz[22]). Particularly for the problem being analysed in this work, \( K_I=11.8 \text{ Ksi} \sqrt{\text{in}} \) (SIF for mode I) was obtained with aid of the vertical displacement at point P (indicated in Figure 2(b)). The correspondent SIF obtained analytically by Broek[23] for a two-dimensional analysis of the plate is \( K_I=12.3 \text{ Ksi} \sqrt{\text{in}} \), such that the numerical solution corresponds to an error of about 4%.

The results in terms of storage room, number of iterations and CPU times are shown in table I. The stop criterion considered for the iterative solver was defined by \( \| \delta \| \leq \| b \| \cdot tol = 10^{-6} \), where \( \| \delta^n \| = \| u^n - u^{n-1} \| \) and in the
analysis it was used a pentium II computer with 300 MHz processor and 64 MBytes of RAM under Microsoft-FORTRAN compiler Power Station 90.

Table I: Storage room requirements, number of iterations and CPU times

<table>
<thead>
<tr>
<th>Work vector (Mb)</th>
<th>Preconditioner</th>
<th>Iterations</th>
<th>CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.4</td>
<td>none</td>
<td>118</td>
<td>315.9</td>
</tr>
<tr>
<td>14.4</td>
<td>Jacobi</td>
<td>68</td>
<td>315.9</td>
</tr>
</tbody>
</table>

5 Conclusions

As expected, the use of the Lanczos iterative technique for treatment of the coupled system had a quite good performance, what can be concluded, though no direct comparison with a direct solver could be done, by observing the relation between the system order and the number of iterations necessary for reaching the convergence ($N/n$). For the problem analysed convergence was obtained with $n=118$ iterations (without preconditioner) and with $n=68$ iterations (with preconditioner), what corresponds to $N/n=15$ and $N/n=27$ respectively. For such proportions $N/n$ the iterative scheme has surely superior performance to the direct ones (see results given by Araújo and Mansur[1], Araújo et al[2], Kane et al[12], Mansur et al[13]), mainly if pivotal search must be conducted in order to ensure numerical stability of the direct solver. It should be furthermore mentioned that in the multi-zone strategy presented here the CPU time per iteration is smaller than in those cases for which systems of equivalent order are originated from single domain discretizations, as no manipulation of zero blocks is carried out in the resulting sparse coupled systems here. Concerning the storage saving, it should be observed that if the system matrix were fully stored, so it would be necessary 26.1 MBytes storage room instead of 14.4 MBytes (see Table I), what corresponds to storage savings of about 45%.

References


Janeiro, Brazil, 1989.


