Experiences with the 3D-ACA in a CQM based Time Domain Boundary Element Method

Anita M. Haider[1], **Martin Schanz**[2,*]
[1]Institute of Applied Mechanics, Graz University of Technology, Graz, Austria
[2]Institute of Applied Mechanics, Graz University of Technology, Graz, Austria
*Email: m.schanz@tugraz.at

Abstract

The proposed presentation is a continuation of the same topic presented at Waves 2019. The acoustic wave equation is solved in time domain with a boundary element formulation. The time discretisation is performed with the generalized convolution quadrature method and for the spatial approximation standard elements and a collocation scheme is applied. To increase the efficiency of the boundary element method the so-called 3D-ACA (three-dimensional Adaptive Cross Approximation) is applied. After introducing the model problem, a Dirichlet problem in acoustics, the 3D-ACA is discussed. The implementation is presented by several examples which serve, essentially, to show the properties of such an approach by numerical tests.

Keywords: BEM, 3D-ACA, CQM

1 Problem setting

As model problem, we consider the time domain boundary element method for the homogeneous wave equation with vanishing initial conditions and given Dirichlet boundary conditions. The generalized convolution quadrature method (gCQ) is used for the temporal discretization (see, e.g. [2]) and collocation for the spatial discretization. Essentially, the gCQ requires to establish boundary element matrices of the corresponding elliptic problem in Laplace domain at several complex frequencies. Consequently, we get an array of system matrices. This array of system matrices can be interpreted as a three-dimensional array of data which we want to approximate by a data-sparse representation.

2 Generalization of the adaptive cross approximation

The idea of a generalization of adaptive cross approximation has been proposed in [1] and is sketched in Algorithm 1. As discussed above, we generate a three-dimensional array of data $\mathcal{C}_{ij,k}$.

Algorithm 1 idea generalized ACA

For $\ell = 1, 2, 3 \ldots$

1. compute face $\mathcal{H}^\ell$ via low-rank approximation $h^\ell_{ij} = \mathcal{C}_{ij,k_\ell}$
2. define pivot position $$(i_\ell, j_\ell) \coloneqq \arg\max_{i,j} |h^\ell_{ij}|$$
3. compute fiber $\mathcal{F}^\ell$ $f^\ell_k = \mathcal{C}_{i_\ell\, j_\ell,k}$

Stop if $\|\widetilde{\mathcal{H}^\ell}\|_F \|\widetilde{\mathcal{F}^\ell}\|_F \leq \varepsilon \|\mathcal{S}^\ell\|_F$

The first two indices corresponds to the spatial discretization. One is related to the collocation point and the other to the basis function. The third index of the 3D array corresponds to the complex frequencies from the gCQ. The algorithm starts by assembling the system matrix at an arbitrary chosen but fixed frequency. Let the corresponding index be k_ℓ. This system matrix, which we call face $\mathcal{H}$, is compressed by the standard adaptive cross approximation. Therefore, we have to decompose the system matrix into a hierarchical scheme first. This hierarchical structure is based on the usual geometrical considerations and used for all further assembled faces. After the low-rank approximated face is determined, we have to define the position of the pivot element. In principle, the maximum entry of the matrix determines the pivot position, where h^ℓ_{ij} denotes the entries of the current face. Regarding to this position we compute the fiber $\mathcal{F}$, an array with all frequencies where the indices related to the spatial discretization are fixed, i.e. one matrix entry at all frequencies for a distinct collocation point and shape function is assembled, f^ℓ_k. In this way the first cross with an approximated face and a fiber is generated. For the next cross, the face or the fiber has to be updated. At further iterations the residual of the face and of the fiber has to be determined and based on this residual

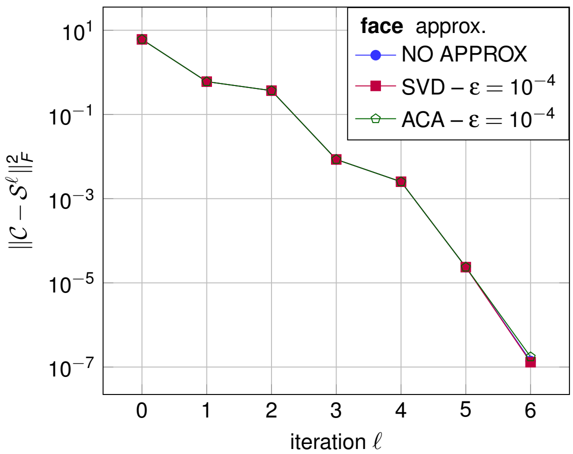

Figure 1: Approximation error in the Frobenius norm versus frequency

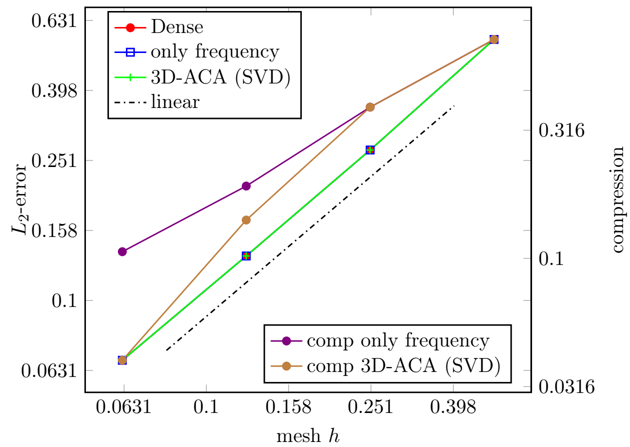

Figure 2: Convergence and compression of a Dirichlet problem computed at a cube

the position of the pivot element is computed. The algorithm terminates successfully if a suitable stopping criterion with a given accuracy ε is satisfied. The current approximation is defined as $\mathcal{S}^{\ell} = \sum_{d=1}^{\ell} \widetilde{\mathcal{H}}^{d} \otimes \widetilde{\mathcal{F}}^{d}$, where ℓ represents the used frequencies to obtain the preset accuracy.

3 Numerical results

How the introduced algorithm performs is shown by numerical experiments. In the first example, the three-dimensional array of data $\mathcal{C}$ is computed by assembling the single layer operator at 11 different frequencies given by the gCQ. The squared error of the approximation in the Frobenius norm is plotted in Fig. 1 against the iteration counter of the algorithm. Note, the iteration number corresponds to the number of necessary complex frequencies. The accuracy of the generalized ACA is chosen as $\varepsilon = 10^{-4}$. We perform the numerical experiment first without any low-rank approximation of the face, then with an approximation by a singular value decomposition and, last, by adaptive cross approximation. The accuracy of the low-rank approximated face is set as well to $\varepsilon = 10^{-4}$ for both approximation methods. For all three alternatives, the stopping criterion is satisfied after five iterations. At the fifth iteration, the ACA approximated face exhibits a slightly different approximation error. This indicates that the low-rank approximation error of the faces has already an effect. Nevertheless, it may be concluded that it is sufficient to evaluate the single layer operator only at a few instead of all frequencies and still a sufficiently quality of the data is maintained, resulting in a reduction of the computation time and the memory consumption.

The second example is the solution of a Dirichlet problem, where the given data represent an analytical solution of the wave equation. Again a cube is used as geometry and in Fig. 2 the L_2-error in space and time is plotted versus the mesh. It must be remarked that not only the spatial discretisation is uniformly refined but as well the time step size such that the relation between both is kept constant. In the same figure, the compression (see right axis) is plotted versus the refinement. The compression is the ratio of storage used by the dense BEM, i.e., all frequencies and dense matrices at each frequency, to the storage necessary for the 3D-ACA. Obviously, large savings are possible by keeping the convergence rate of the gCQ-BEM. The underlying time stepping was a 2-stage Radau IIA Runge-Kutta method. Above the SVD is used to compress the data within a face. In the presentation, as well results with ACA in the face will be shown and different geometries be used.

References

[1] M. Bebendorf, A. Kühnemund, and S. Rjasanow, An equi-directional generalization of adaptive cross approximation for higher-order tensors, *Appl. Numer. Math.* **74**, (2013), pp. 1–16 .

[2] M. López-Fernández and S. Sauter, Generalized convolution quadrature with variable time stepping, *IMA J. of Numer. Anal.* **33** (2013), pp. 1156–1175.