Available online at www.prace-ri.eu

Partnership for Advanced Computing in Europe

A Parallel Fast BEM for the Helmholtz Equation as an Extension of SPECFEM3D

Dalibor Lukas, Jan Zapletal

Department of Applied Mathematics, IT4Innovations, VSB-Technical University of Ostrava, Czech Republic

---

Abstract

In this paper, a new parallel acoustic simulation package has been created, using the boundary element method (BEM). The package is built on top of SPECFEM3D, which is parallel software for doing seismic simulations, e.g. earthquake simulations of the globe. The acoustical simulation relies on a Fourier transform of the seismic elastodynamic data, resulting from SPECFEM3D_GLOBE, which are then postprocessed by a sequence of solutions to Helmholtz equations, in the exterior of the globe. For the acoustic simulations BEM has been employed, which reduces computation to the sphere; however, its naive implementation suffers from quadratic time and memory complexity, with respect to the number of unknowns. To overcome the latter, the method was accelerated by using hierarchical matrices and adaptive cross approximation techniques, which is referred to as Fast BEM. First, a hierarchical clustering of the globe surface triangulation is performed. The arising cluster pairs decompose the fully populated BEM matrices into a hierarchy of blocks, which are classified as far-field or near-field. While the near-field blocks are kept as full matrices, the far-field blocks are approximated by low-rank matrices. This reduces the quadratic complexity of the serial code to almost linear complexity, i.e. O(n*log(n)), where n denotes the number of triangles. Furthermore, a parallel implementation was done, so that the blocks are assigned to concurrent MPI processes with an optimal load balance. The processes share the triangulation data. The parallel code reduces the computational complexity to O(n*log(n)/N), where N denotes the number of processes. This is a novel implementation of BEM that overcomes computational times of traditional volume discretization methods, e.g. finite elements, by an order of magnitude.

---

1. Introduction

The boundary element method (BEM) is an efficient technique for the simulation of physical fields in homogeneous media, relying on partial differential equations. In particular, we are interested in acoustics in unbounded domains, in which case the use of BEM is theoretically proven [4] to be superior to volume discretization techniques. The developed package serves as a post-processing stage to software SPECFEM3D_GLOBE, which solves seismic simulations of the Earth. The package is a new contribution, rather than an improvement, to SPECFEM3D_GLOBE.

Though boundary integral equations were discovered long before the variational approach leading to the methods of finite elements (FEM), for a long time they were not suitable for large scale computations due to the fact that they lead to fully-populated matrices, i.e. to complexity O(n^2), where n stands for the number of boundary triangles. The reveal of BEM dates back to Greengard and Rokhlin [1], who developed the Fast Multipole Method. Their method hierarchically clusters the geometrical domain under consideration into near and far-field couples of clusters and replaces the integral kernel of the far-field by proper low-order expansions, e.g. spherical harmonics. Another important class of Fast BEM relies on hierarchical matrices that were introduced by Hackbusch and Nowak [2]. Instead of the integral kernel, far-field parts of the matrix are hierarchically approximated by matrices of low ranks, which are typically constructed by interpolation. Recently, Bebendorf [3] has proposed the adaptive cross approximation (ACA) method. ACA is a black-box method that interpolates the far-field submatrices without any knowledge of the related kernel, for which reason it has become a very frequently used variant of Fast BEM. All the mentioned methods reduce the complexity of the matrix assembling as well as solving to related linear systems to O(n*log(n)).

Yet a parallel implementation of Fast BEM has been an issue. In this paper we propose a parallel Fast BEM for shared memory systems. In Section 2 we describe BEM applied to Helmholtz Equation. In Section 3 sparsification techniques of hierarchical matrices and ACA are recalled, and in addition some details of the parallel implementation are given. Finally, in Section 4 numerical results document the theoretical parallel scalability.

2. BEM for Helmholtz Equation

We consider the following exterior Helmholtz problem with the prescribed Neumann datum:

$$\begin{aligned} -\triangle p(\mathbf{x}) - \kappa^2 p(\mathbf{x}) &= 0 && \text{in } \Omega^{\mathrm{e}} := \mathbb{R}^3 \setminus \overline{\Omega}, \\ \partial p(\mathbf{x})/\partial \mathbf{n}(\mathbf{x}) &= v_n(\mathbf{x}) := \omega^2 \rho \mathbf{u}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) && \text{on } \Gamma, \\ |\nabla p(\mathbf{x})\, \mathbf{x}/|\mathbf{x}| - \mathrm{i}\kappa p(\mathbf{x})| &= O(|\mathbf{x}|^{-2}) && \text{as } |\mathbf{x}| \to \infty. \end{aligned}$$

The resulting pressure p is a complex valued function, kappa := omega/c stands for the wave number, omega denotes the angular frequency, c denotes the speed of sound, n denotes the unit outer normal to the domain Omega, and u denotes the related component of the displacement field in the frequency domain. Note that, in case of SPECFEM3D_GLOBE we shall first translate the resulting time-dependent displacement field to the frequency domain, solve a sequence of Helmholtz problems, and finally transfer the results back to the time-domain.

We shall make use of the following fundamental solution:

$$P(\mathbf{x}, \mathbf{y}) := \frac{\mathrm{e}^{\mathrm{i}\kappa r}}{4\pi r}, \quad \mathbf{x} \in \Omega^{\mathrm{e}}$$

We consider the direct boundary integral approach, which leads to the following ansatz:

$$p(\mathbf{x}) = -\int_\Gamma v_n(\mathbf{y}) P(\mathbf{x}, \mathbf{y})\, dS(\mathbf{y}) + \int_\Gamma p(\mathbf{y}) \left(\partial P(\mathbf{x}, \mathbf{y})/\partial \mathbf{n}(\mathbf{y})\right)$$

Applying the normal derivative to the latter, and using the Galerkin approach, we arrive at

$$\langle D_\kappa p, v\rangle_\Gamma = \langle(-1/2 I + K'_\kappa) v_n, v\rangle_\Gamma \text{ on } H^{1/2}(\Gamma)$$

with the following boundary integral operators:

$$\langle D_\kappa p, v\rangle_\Gamma := \int_\Gamma \int_\Gamma P(\mathbf{x}, \mathbf{y}) \left[(\mathbf{n}(\mathbf{x}) \times \nabla \widetilde{v}(\mathbf{x})) \cdot (\mathbf{n}(\mathbf{y}) \times \nabla \widetilde{p}(\mathbf{y})) - \kappa^2 \mathbf{n}(\mathbf{x}) v(\mathbf{x}) \cdot \mathbf{n}(\mathbf{y}) p(\mathbf{y})\right]\, d$$

$$\langle v_n, v\rangle_\Gamma := \int_\Gamma v_n(\mathbf{x}) v(\mathbf{x})\, dS(\mathbf{x}), \quad \langle K'_\kappa v_n, v\rangle_\Gamma := \int_\Gamma \int_\Gamma \left(\partial P(\mathbf{x}, \mathbf{y})/\partial \mathbf{n}(\mathbf{y})\right) v_n(\mathbf{y}) v(\mathbf{x})\, dS(\mathbf{y})\, d$$

Furthermore, we triangulate the boundary into n triangles and replace the fractional Sobolev spaces by continuous piecewise linear functions for the pressure and piecewise constants for the velocity field. The inner integrals are evaluated analytically, which gets rid of singularities, while the other integration is efficiently calculated by Gauss quadrature. We refer to [4] for details. This leads to a dense system of linear equations.

3. Hierarchical matrices and ACA

The triangulation of the boundary is hierarchically decomposed into clusters as depicted in Fig. 1. The pairs of clusters are related to submatrices, which are classified as near or far-field, depending on a relation between the cluster sizes and their distance. While the near-field submatrices are kept fully-populated, the far-field submatrices are approximated by a low-rank format.

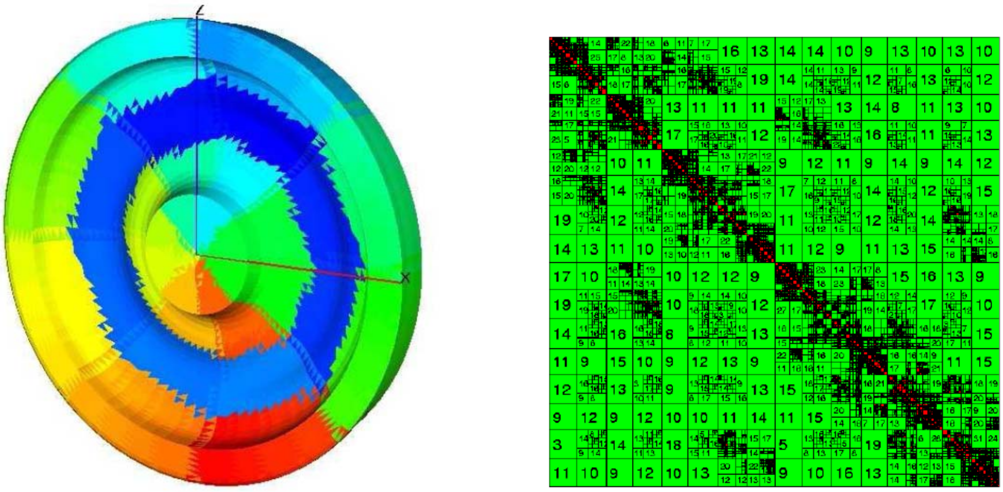

Fig. 1: Hierarchical clustering of the geometry (left), related hierarchical matrix (right)

The method ACA subsequently chooses pivots in the residuum of the actual approximation of the far-field matrix, and then updates the approximation by a rank-1 matrix. It is a product of the pivoted row and column of the residual matrix. It can be viewed as an interpolation of the original far-field matrix to the pivot rows and columns, see the equation below. The approximation error is the spectral norm of the Schur complement. ACA in a combination with hierarchical matrices reduces the quadratic complexity to almost linear, i.e. O(n*log n).

$$\mathbf{P}_{C_x}\,\mathbf{A}\,\mathbf{P}_{C_y}^T =: \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{22} \end{pmatrix} \approx \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{21}\,\widetilde{\mathbf{A}}_{11}^{-1}\,\widetilde{\mathbf{A}}_{12} \end{pmatrix} = \begin{pmatrix} \widetilde{\mathbf{A}}_{11} \\ \widetilde{\mathbf{A}}_{21} \end{pmatrix} \left[\widetilde{\mathbf{A}}_{11}^{-1} \left(\widetilde{\mathbf{A}}_{11}, \widetilde{\mathbf{A}}_{12} \right) \right]$$

$$=: \left(\mathbf{u}_1, \ldots, \mathbf{u}_r \right) \left(\mathbf{v}_1, \ldots, \mathbf{v}_r \right)^T .$$

4. Parallel implementation and numerical results

In Fig. 2 a scheme of our parallel implementation of ACA-BEM is depicted. We employ N processes, the first of which is the master scheduling jobs to itself as well as to the remaining slaves. Both near and far-field blocks are efficiently assigned to processes making use of reliable load estimates. The expensive blocks are scheduled first. The scheduling is performed at the master side, its computational time is negligible. After we have the blocks assigned, the master sends their indices to the processes. Then, each process including the master reads the geometry data from the shared memory and initiates assembling of the densely populated near-field submatrices as well as ACA approximations to the far-field submatrices. When the assembling part is done, the processes are waiting for a signal to contribute to the matrix-times-vector multiplication within GMRES iteration method for solution to a linear system of equations.

In Tab. 1 we demonstrate the efficiency of the method. A Dirichlet boundary value problem for the Helmholtz equation is considered with a known analytical solution in order to document the linear decay of the approximation error. The rows correspond to the levels of discretization. In the first column numbers of triangles are given. In the second column, the linear error decay in terms of the L2-norm of the computed Neumann data is shown. The third column shows the logarithmic compression rate of the full matrix. Finally, columns 4-8 document the overall serial (top-to-bottom) as well as parallel (left-to-right) scalability of the implementation. A comparable, i.e. producing similar error, FEM discretization of the largest problem would lead to 1e8 volume unknowns.

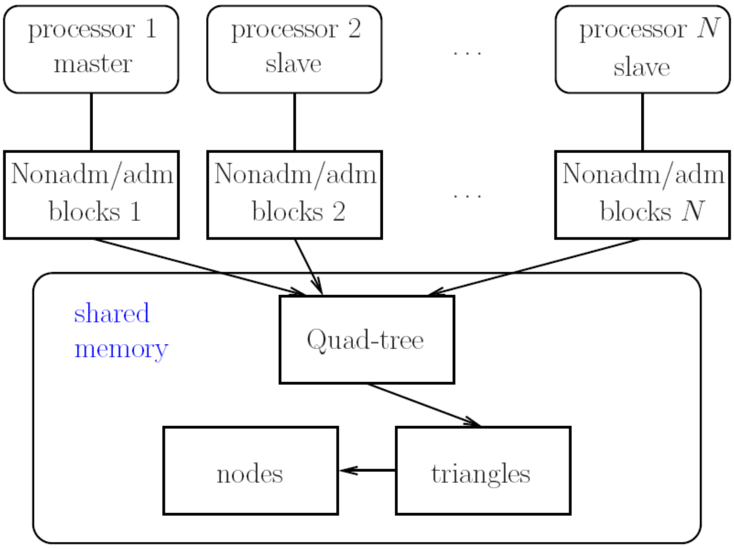

Fig. 2: Scheme of the parallel implementation

Helmholtz, Dirich. $u(\mathbf{x}) := e^{\imath\kappa|\mathbf{x}-\mathbf{x}_s|}/(4\pi|\mathbf{x}-\mathbf{x}_s|)$, $\kappa := 2.8$, $\mathbf{x}_s := (2,2,2)$ **on** $\mathcal{B}_1$

		compr.	scheduling+assembling times of $\mathbf{V}_\kappa$ [s]				
n	err.	of $\mathbf{V}_\kappa$	$N := 2$	$N := 4$	$N := 8$	$N := 16$	$N := 32$
40	3.3e-1	100%	0+0	0+0	0+0	0+0	0+0
160	1.2e-1	100%	0+1	0+1	0+1	0+1	0+1
640	3.6e-2	100%	0+10	0+4	0+3	0+2	0+2
2560	9.9e-3	100%	0+142	0+72	0+38	0+20	0+9
10240	2.8e-3	65%	66+1388	27+673	7+335	7+168	5+88
40960	9.0e-4	26%			452+3600	280+1823	233+929
163840	3.3e-4	8%					4011+19892

Tab. 1: Efficiency of the parallel Fast BEM method

Acknowledgements

This work was financially supported by the PRACE project funded in part by the EUs 7th Framework Programme (FP7/2007-2013) under grant agreement no. RI-261557

References

[1] L. Greengard and V. Rokhlin, A fast algorithm for particle simulations. J. Comput. Phys. 73(2):325-348, 1987.

[2] W. Hackbusch and Z. P. Nowak, On the fast matrix multiplication in the boundary element methods by panel clustering. Numer. Math. 54(4):463-491, 1989.

[3] M. Bebendorf, Approximation of boundary element matrices. Numer. Math. 86(4):565-589, 2000.

[4] O. Steinbach and S. Rjasanow, The Fast Solution of Boundary Integral Equations. Springer-Verlag, 2007.