Sparsification of the Reduced Matrix of the CBFM for a Memory Efficient Solution of Electrically Large EM Scattering Problems

Ines Fenni, Ziad S. Haddad
Jet Propulsion Lab,
California Institute of Technology, USA
Ines.Fenni@jpl.nasa.gov

Hélène Roussel
Sorbonne Université
L2E-UR2
75005 Paris, France

Raj Mittra
University of Central Florida,
USA and King Abdulaziz
University, Saudi Arabia

***Abstract*—In this paper a sparsification approach is applied to the compressed matrix resulting from the Characteristic Basis Function Method (CBFM) process in order to significantly reduce the memory cost of this direct solver-based largely used numerical technique. Many efforts have been made in recent years to efficiently calculate this matrix but, all of them have focused on the time cost and have not dealt with the memory resources needed to store it. With the proposed sparsification approach, the present work aims to reduce the computational cost associated to the compressed matrix both in terms of CPU time and memory consumption.**

I. Introduction

The Characteristic Basis Function Method (CBFM) is a powerful recent direct solver-based numerical method [1] designed to solve large-scale electromagnetic (EM) problems, even with limited computing resources. The CBFM is based on generating a reduced set of basis functions (CBFs) adapted to the geometry of the problem of interest, so that the reduced matrix equation can be handled via a direct method at a much lower computational cost. So far, The CBFM has shown good compression performances when applied to a wide range of large electromagnetic problems (RFICs, Antennas, remote sensing scattering problems) and achieved a satisfactory level of accuracy even when applied to complex geometries and highly heterogeneous simulation scenes [1], [2], [3]. Nevertheless, the CBFM is still intensive particularly in terms of memory when applied to numerically very large EM problems. Since direct solver-based, despite the high compression rate the CBFM achieves, the reduced matrix Z^c becomes untenable both in terms of memory resources and cpu time when considering numerically very large problems (millions of unknowns). Several efficient algorithms have been proposed in recent years to efficiently generate and solve the reduced matrix Z^c, such as the adaptive cross approximation (ACA) algorithm, the fast multipole method (FMM) and the Sherman-Morrison Woodbury Formula [2], [4]. However, all of these approaches were employed only to speed up the calculation of Z^c and have not dealt with the memory resources needed to store it. To overcome this deficiency, a matrix sparsification is applied in this paper while generating Z^c to reduce the associated computational cost both in terms of CPU time and memory consumption.

II. Application of the sparsification algorithm while generating the reduced matrix Z^c

First, we recall that the CBFM procedure begins by dividing the 3D geometry of the scatterer into M blocks, such that the MoM matrix for each block is manageable in size and, therefore, could be handled by using a direct solver. Then, a set of Macro-domain Basis Functions (MBFs) is defined on each block after illuminating it by a sufficient number of plane waves incident at different angles. Next, a Singular Value Decomposition (SVD) is applied and a threshold is used to down-select only a small set of dominant macro basis functions to represent the unknown field, and be referred to as the Characteristic Basis Functions (CBFs) for the individual blocks. These new CBFs are used to construct a final reduced linear set of equations, as compared to the initial one generated by using traditional low-level basis functions.

The reduced matrix $\mathbf{Z}^c$ is generated by applying the Galerkin's method to the original matrix $\mathbf{Z}^{MoM}$ using as basis and testing functions the CBFs. The submatrix $\mathbf{Z}^c_{i,j} = \mathbf{C}^{(i)t}\mathbf{Z}^{ij}\mathbf{C}^{(j)}$ is then an $S_i \times S_j$ matrix representing the compressed interactions between blocks i and j [1], [2].

At the end of the process, a direct solver is used to calculate in total K complex coefficient vector $\boldsymbol{\alpha}$ ($K = \sum_{i=1}^{M} S_i$). The total electric field inside the scatterer is then expressed as a linear combination of the K CBFs weighted by the K thus obtained complex coefficients α.

Now, even when the CBFM achieves a high compression rate CR, defined as the ratio between the number of the original basis functions and the number of post-CBFM unknowns, we still need to store and solve a full dense matrix of size $K \times K$. This can be costly both in terms of CPU time and memory, particularly if the initial pre-CBFM EM problem involves several millions of unknowns.

To overcome this limitation, we propose to apply a simple matrix sparsification on the fly to the submatrices $\mathbf{Z}^c_{i,1\leqslant j\leqslant M}$ in order to reduce the total number of non-zero elements of Z^c.

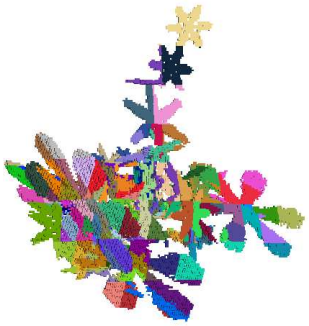


Fig. 1: Snow aggregate divided into 120 blocks

Once Z^c is appropriately transformed from a full to a sparse matrix, a high-performance, robust and memory efficient direct sparse solver such as the Intel MKL PARDISO is used to efficiently calculate the final complex coefficients α.

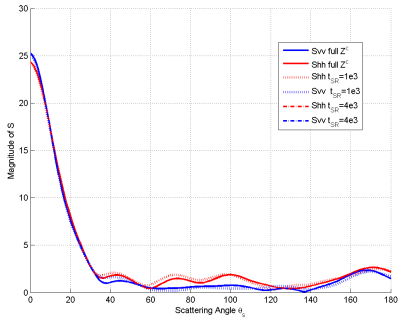
Let us consider $\mathbf{Z}_i^c = \mathbf{Z}_{i,1\leqslant j\leqslant M}^c$ the submatrix of Z^c of size $S_i \times K$ representing the interactions between block i and all the other blocks $1 \leqslant j \leqslant M$. We first determine the maximum element in magnitude of $\mathbf{Z}_i^c$, then we sparsify it by simply zeroing out all its sufficiently small elements as follow

$$\widetilde{\mathbf{Z}_i^c}(s,k) = 0 \text{ if } |\mathbf{Z}_i^c(s,k)| \leqslant \frac{max|\mathbf{Z}_i^c|}{t_{SR}} \quad 1 \leqslant s \leqslant S_i \; ; \; 1 \leqslant k \leqslant K \tag{1}$$

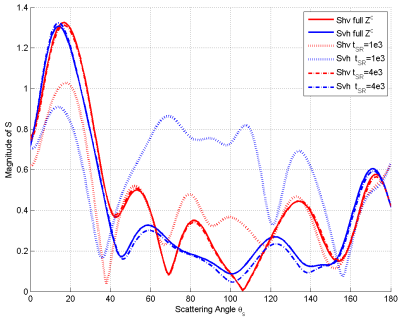
where t_{SR} is a threshold used to down-select the elements of $\mathbf{Z}_i^c$ whose magnitudes are significant compared to $max|\mathbf{Z}_i^c|$. To ensure the accuracy of the approximation, t_{SR} could be defined through an iterative process until achieving an ϵ approximation of $\mathbf{Z}_i^c$, such that $||\mathbf{Z}_i^c - \widetilde{\mathbf{Z}_i^c}(t_{SR})||_F \leqslant \epsilon$ where $||.||_F$ is the Frobenius norm and ϵ is the approximation error selected by the user. If n_i significant elements are retained for each $\mathbf{Z}_i^c$, we will need only to store $n = \sum_{i=1}^{M} n_i$ out of the total K^2 elements of the reduced matrix Z^c. Therefore. the gain in memory achieved by the sparsification is calculated as $g(\%) = 100 \times (max[S_iK, n]/K^2)$, S_iK being the size of each entirely calculated and stored $\mathbf{Z}_i^c$.

III. Numerical results

To check the validity of the proposed sparsification approach, we calculate the scattering matrix elements for the snow aggregate in Fig. 1 of maximum dimension $d_z = 8.85$ mm; effective radius $a_p = 1.16$ mm and refractive index $m = 1.8{+}\mathrm{i}{\times}13e^{-4}$ at $f = 150$ GHz ($d_z = 7.9 \times \lambda_s$ where λ_s is the wavelength inside the scatterer). The snow particle is discretized into 52364 cells and divided into 120 CBFM blocks. Fig. 2 plots the magnitude of S_{VV}, S_{HH}, S_{VH} and S_{HV} obtained with a classical full-matrix CBFM, and after sparsification with t_{SR} = [$1e3$; $4e3$]. With $t_{SR} = 1e3$, we keep 3.94% of the K^2 elements of the reduced matrix, which is not, as can be seen in Fig. 2, sufficient to ensure the accuracy of the cross-polarization S elements. On the other hand, we achieve a good level of accuracy with $t_{SR} = 4e3$ while storing only 11.71% of the total Z^c of size 7133. Furthermore, the



(a) S_{VV} and S_{HH}



(b) S_{VH} and S_{HV}

Fig. 2: Variations of the scattering matrix elements with a full Z^c CBFM and after sparsification with t_{SR} = [$1e3$; $4e3$].

use of the sparse direct solver PARDISO enables us to reduce the solving time from 51 to 21 seconds. We expect the gain in CPU time brought by the use of the sparse direct solver to increase with the numerical size of the EM problem.

IV. Conclusion

The approach proposed in this paper enables to significantly reduce the computational cost of the storage and resolution of the compressed matrix Z^c while maintaining a satisfactory level of accuracy, provided that the threshold t_{SR} is adequately selected. The achieved gain in memory will make it possible to apply the CBFM to electrically larger EM problems.

References

[1] R. Mittra and K. Du, "Characteristic basis function method for iteration-free solution of large method of moments problems," *Progress In Electromagnetics Research B*, vol. 6, pp. 307–336, 2008.

[2] R. Maaskant, R. Mittra, and A. Tijhuis, "Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm," *Antennas and Propagation, IEEE Transactions on*, vol. 56, no. 11, pp. 3440–3451, Nov 2008.

[3] I. Fenni, Z. S. Haddad, H. Roussel, K. S. Kuo, and R. Mittra, "A computationally efficient 3-d full-wave model for coherent em scattering from complex- geometry hydrometeors based on mom/ cbfm-enhanced algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. PP, no. 99, pp. 1–15, 2017.

[4] X. Chen, C. Gu, Z. Li, and Z. Niu, "Accelerated direct solution of electromagnetic scattering via characteristic basis function method with sherman-morrison-woodbury formula-based algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 64, no. 10, pp. 4482–4486, Oct 2016.