Tucker Decomposition for Compressing Translation Operator Tensors in FMM-FFT Accelerated SIE Solvers

Abdulkadir C. Yücel*, Luis J. Gomez, and Eric Michielssen
Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA

Fast multipole method – Fast Fourier transform (FMM-FFT) accelerated surface integral equation (SIE) solvers allow for accurate and efficient analysis of electromagnetic (EM) scattering from and radiation by complex and large scale structures (R. L. Wagner et. al., *IEEE Trans. Antennas Propagat.*, 45(2), 235-245, 1997). These solvers (and their multilevel extensions) provide an increasingly appealing avenue for solving EM scattering problems involving hundreds of millions (and billions) of unknowns (Taboada et. al., *IEEE Antennas Propagat. Mag.*, 51(6), 20-28, 2009; Taboada et. al., *Progress in Electromagnetics Research*, 105, 15-30, 2010). When used on present high-performance computers to solve practical problems of current interest, these solvers tend to be memory as opposed to CPU-limited. The solvers' memory requirements directly depends on the storage requirements for (i) near-field interaction matrices, (ii) matrices that hold the far-field signatures of basis functions, and (iii) tensors that hold *FFT'ed* translation operator values on a structured grid. In past, the memory requirements of the first two data structures were successfully reduced by singular value decomposition (SVD) (Kapur and Long, *IEEE Comp. Sci. Eng.*, 5(4), 60-67, 1998; Rodriguez et. al., *IEEE Trans. Antennas Propagat.*, 56(8), 2325-2334, 2008). To date, no compression scheme has been reported to reduce the memory requirements of translation operator tensors.

In this study, Tucker decompositions (a.k.a. higher-order SVDs) are leveraged to compress translation operator tensors that arise in FMM-FFT accelerated SIE solvers. The proposed decomposition enables to represent translation operator tensors in terms of low-rank core tensors and factor matrices, which are constructed for each plane-wave direction during the solvers' setup stage and then used to restore the original tensor during the solvers' matrix-vector multiplication stage. For many practical examples, the proposed code enhancement achieves over 90% reduction in the memory required for storing translation operator tensors (to high accuracy, say 10^{-6}). These memory savings are realized using minimal CPU resources since the computational time for restoration/decompression operation is only a small fraction of the overall matrix-vector multiplication time. The accuracy of, memory savings achieved with, and (negligible) computational overhead associated with the proposed drop-in code enhancement for FMM-FFT accelerated SIE solvers will be demonstrated through various numerical examples involving complex and large scale structures.