Tucker-FMM-FFT-Accelerated SIE Solver for Large-Scale Electromagnetic Analysis

Cheng Qian and Abdulkadir C. Yucel
School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore, 639798
cqian@ntu.edu.sg, acyucel@ntu.edu.sg

***Abstract*—A tensor decomposition methodology is combined with the fast multipole method-fast Fourier transform (FMM-FFT) technique to accelerate the surface integral equation (SIE) solvers. The proposed methodology leverages Tucker and hierarchical Tucker (H-Tucker) decompositions to compress the three-dimensional (3D) arrays storing the far-fields and five-dimensional (5D) arrays storing the translation operator samples, respectively. The compressed tensors are then used in the matrix-vector and element-wise products in aggregation/disaggregation and translation stages. By doing so, all stages of the FMM-FFT are performed via the Tucker-compressed tensors. The resulting Tucker-FMM-FFT-accelerated SIE simulator is far more memory and CPU efficient than the traditional FMM-FFT-accelerated SIE simulators. Preliminary results show that the Tucker-FMM-FFT acceleration technique requires 12x less memory and 17x less CPU time compared to the traditional FMM-FFT acceleration technique for the electromagnetic (EM) scattering analysis of a frequency-selective surface.**

Keywords—Fast Fourier transform, fast multipole method, surface integral equation, tensor decompositions, tensor train, Tucker decomposition.

I. Introduction

The fast multipole method (FMM)-accelerated surface integral equation (SIE) simulators have enabled efficient and accurate characterization of electromagnetic (EM) phenomena on electrically large and complex platforms. The variants of these solvers, such as fast Fourier transform (FFT) based ones (a.k.a. FMM-FFT) [1], exhibit $O(N^{4/3}\log^{2/3}N)$ computational complexity, where N is the number of basis functions used to discretize the surface currents. To reduce the multiplicative factor inherent in this complexity estimate, several tensor decomposition methodologies have been introduced to compress large data structures of these simulators [2-5]. Similar tensor methodologies have also been used to reduce the computational and memory requirements of FFT-accelerated integral equation solvers [6-10]. For the FMM-FFT-accelerated solvers, Tucker and hierarchical-Tucker (H-Tucker) decompositions are particularly applied to compress the far-fields and *FFT'ed* translation operator tensor, respectively [2, 3]. These decompositions are implemented in the existing codes of the FMM-FFT-accelerated SIE simulators and yield a significant reduction in their computational resource requirement with and without a negligible overhead.

In this study, a Tucker decomposition network is used to fully incorporate the Tucker-compressed tensors in the FMM-FFT technique and develop a Tucker-FMM-FFT-accelerated SIE solver. To do that, methodologies leveraging the Tucker and H-Tucker decompositions are developed for compressing 3D arrays storing the far-fields as well as the 5D arrays storing the *FFT'ed* translation operator samples. The Tucker and H-Tucker-compressed tensors are obtained for a given tolerance. [11] These tensors are then combined in such a way that all aggregation/disaggregation and translation operations of FMM-FFT are performed via the Tucker and H-Tucker-compressed tensors. The combination of the Tucker-compressed tensors [12] via a tensor network brings up significant memory and CPU time reduction, which allows increasing the applicability of FMM-FFT-accelerated SIE simulators on fixed computational resources. This judicious combination will be explained in detail during the presentation, while the general idea of methodologies and preliminary results are provided here.

II. Formulation

A brief explanation of the Tucker-FMM-FFT acceleration scheme is given here. Assume that a hypothetical box enclosing a perfect electric conducting (PEC) structure is divided into n_x, n_y, and n_z small boxes along principle axes. These boxes are centered on a uniform 3D grid points and labeled by $B_{\mathbf{u}}$ with $\mathbf{u}=(u_x,u_y,u_z)$, $u_x=1,...,n_x$, $u_y=1,...,n_y$, $u_z=1,...,n_z$. The interactions between basis functions in adjacent boxes are accounted for classically. The matrix-vector products (MVPs) related to these basis functions are directly computed during the iterative solution of SIEs. On the other hand, the interactions between basis functions in well-seperated boxes are computed via aggregation, translation, and disaggregation stages.

For the aggregation and disaggregation stages, all basis functions' far-field patterns along n_{dir} plane-wave directions are computed and stored in a 3D array. Here $n_{\text{dir}}=n_\theta n_\varphi$, n_θ and n_φ are the numbers of quadrature points selected along θ- and φ-directions on the spherical grid [2]. This 3D array $\boldsymbol{\mathcal{A}}$ with dimensions $N\times n_\theta\times n_\varphi$ is compressed via its Tucker decomposition as

$$\boldsymbol{\mathcal{A}}=\boldsymbol{\mathcal{C}}_{\text{T}}\times_1\bar{\mathbf{U}}_{\text{T}}^1\times_2\bar{\mathbf{U}}_{\text{T}}^2\times_3\bar{\mathbf{U}}_{\text{T}}^3, \tag{1}$$

where $\times_q$, $q=\{1,2,3\}$ denotes the mode $-q$ matrix product, $\boldsymbol{\mathcal{C}}_{\text{T}}$ and $\bar{\mathbf{U}}_{\text{T}}^{1,2,3}$ represent core tensor and factor matrices, respectively. These compressed tensors (indicated via blue-

This work was supported by Ministry of Education, Singapore, under grant AcRF TIER 1-2018-T1-002-077 (RG176/18), and the Nanyang Technological University under a Start-Up Grant.

colored boxes in Fig.1) are used during the aggregation stage. Complex conjugation of these tensors yields the receiving field patterns of all basis functions, used in the disaggregation stage (indicated via green-colored boxes in Fig.1). For the translation stage, the *FFT'ed* translation operator tensor for all plane-wave directions are computed and stored in a 5D array. This 5D array $\mathcal{T}$ with dimensions $2n_x \times 2n_y \times 2n_z \times n_\theta \times n_\varphi$ is compressed by its H-Tucker representation as

$$\mathcal{T} = [(\mathcal{C}_{\mathrm{HT}}^{2} \times_3 \bar{\mathbf{C}}_{\mathrm{HT}}^{12}) \times_3^3 \mathcal{C}_{\mathrm{HT}}^{1} \times_4^3 \mathcal{C}_{\mathrm{HT}}^{3}] \\ \times_1 \bar{\mathbf{U}}_{\mathrm{HT}}^{1} \times_2 \bar{\mathbf{U}}_{\mathrm{HT}}^{2} \times_3 \bar{\mathbf{U}}_{\mathrm{HT}}^{3} \times_4 \bar{\mathbf{U}}_{\mathrm{HT}}^{4} \times_5 \bar{\mathbf{U}}_{\mathrm{HT}}^{5}. \quad (2)$$

where $\bar{\mathbf{C}}_{\mathrm{HT}}^{12}$ is the transfer matrix, $\mathcal{C}_{\mathrm{HT}}^{1,2,3}$ are the transfer tensors, $\bar{\mathbf{U}}_{\mathrm{HT}}^{1,2,3,4}$ represent the factor matrices, and $\times_i^j$ stands for the tensor contraction along mode $-i$ and mode $-j$ of tensors. These compressed tensor representations are indicated via yellow-colored boxes in Fig.1. During the off-line stage (in frame with dotted lines), element-wise multiplication of the convolution is performed via one-time outer-product of core tensors and inner-product of factor matrices. During the on-line stage, multiplication between the current coefficient vector $\mathbf{I}$ and the factor matrix is performed. The resulting tensor is multiplied with the pre-computed tensors. Details of all these operations will be provided during the presentation.

III. Numerical Result

The proposed Tucker-FMM-FFT acceleration technique is implemented in an SIE solver and its performance is compared with that of the traditional FMM-FFT acceleration technique. To this end, a PEC frequency selective surface (FSS) [13] is considered. The FSS consists of square loop elements positioned on a 2D grid; the dimensions of each element are shown in the inlet of Fig.2(a). The number of elements in the FSS is varied from 10×10 to 80×80. For the analysis at 300 MHz, the FMM box size and FMM accuracy are set to 0.5λ and 5 digits, respectively, where λ denotes the wavelength. With the increase in the structure size, N is increased from 14,000 to 896,000, the size of $\mathcal{A}$ is increased from 14,000×16×31 to 896,000×16×31 while the size of $\mathcal{T}$ is increased from 45×45×5×16×31 to 325×325×5×16×31. In Fig. 2(a) and (b), we plot the memory and CPU requirements of the proposed and the traditional methodologies while N increases. Clearly, Tucker-FMM-FFT technique yields significant memory and CPU time saving compared to the traditional FMM-FFT technique. For the analysis of 80×80 elements, the proposed technique reduces the memory requirement from 17,560 MB to 1,430 MB, achieving 12x memory reduction. In parallel, implementation of MVP in Tucker-compressed format reduces the CPU time cost of one MVP from 131.7 s to 7.7 s, resulting in 17x acceleration. It is shown in Fig. 2(c), for the decomposition tolerance of 10^{-6}, the absolute difference between the MVP results obtained by the traditional FMM-FFT and Tucker-FMM-FFT is always less than 10^{-4} for the case of 10×10 elements.

IV. Conclusion

A Tucker-FMM-FFT acceleration scheme was proposed to reduce the memory requirement and computational cost of SIE solvers. Preliminary results demonstrate the memory and computational saving as well as the accuracy achieved by the proposed scheme. Currently, the developed Tucker-FMM-FFT accelerated SIE simulator is being applied to the EM scattering analysis of various canonical and realistic structures.

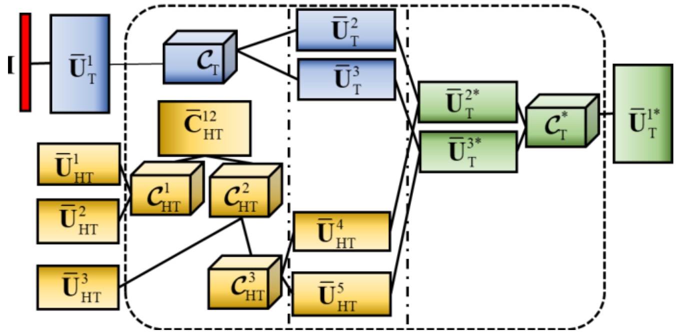

Fig. 1. The scheme for fast aggregation, translation, and disaggregation stages via Tucker decomposition tensor networks.

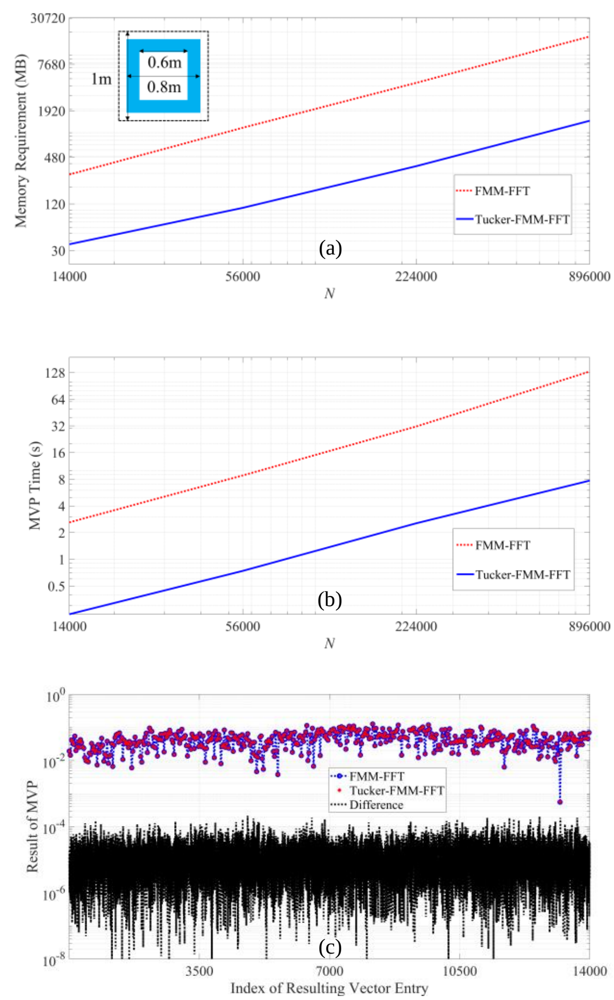

Fig. 2. (a) The memory and (b) CPU time requirements w.r.t. the number of unknowns when the size of PEC FSS is increased from 10×10 to 80×80 elements. (c) The results of one MVP and the difference between the MVP results obtained by FMM-FFT and Tucker-FMM-FFT for the 10×10 element case.

References

[1] J. M. Taboada et al., “High scalability FMM-FFT electromagnetic solver for supercomputer systems,” *IEEE Antennas Propag. Mag.*, vol. 51, no. 6, pp. 20–28, 2009.

[2] C. Qian, M. Wang, and A.C. Yucel, “Compression of far-fields in the fast multipole method via Tucker decomposition,” *IEEE Trans. Antennas Propag.*, in press, 2021.

[3] C. Qian and A. C. Yucel, “On the compression of translation operator tensors in FMM-FFT-accelerated SIE simulators via tensor decompositions," *IEEE Trans. Antennas Propag*, in press, 2020 doi: 10.1109/TAP.2020.3030981.

[4] A. C. Yucel, W. Sheng, C. Zhou, Y. Liu, H. Bagci, and E. Michielssen, “An FMM-FFT accelerated SIE simulator for analyzing EM wave propagation in mine environments loaded with conductors,” *IEEE J. Multiscale and Multiphys. Comput. Techn.*, vol. 3, pp. 3-15, 2018

[5] A. C. Yucel, L. J. Gomez, and E. Michielssen, “Compression of translation operator tensors in FMM-FFT accelerated SIE solvers via Tucker decomposition,” *IEEE Antennas Wireless Propag. Lett.*, vol. 16, pp. 2667-2670, 2017

[6] I. I. Giannakopoulos, M. S. Litsarev, and A. G. Polimeridis, “Memory footprint reduction for the FFT-based volume integral equation method via tensor decompositions,” *IEEE Trans. Antennas Propagat.* vol. 67, pp. 7476-7486, 2019

[7] I. I. Giannakopoulos et al, “Compression of volume-surface integral equation matrices via Tucker decomposition for magnetic resonance applications,” arXiv preprint arXiv:2103.06393, 2021

[8] M. Wang, C. Qian, J. K. White, and A. C. Yucel, “VoxCap: FFT-accelerated and Tucker-enhanced capacitance extraction simulator for voxelized structures,” *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 12, pp. 5154-5168, Dec. 2020

[9] Z. Chen, L. J. Gomez, S. Zheng, A. C. Yucel, Z. Zhang, and V. I. Okhmatovski, “Sparsity-aware precorrected tensor train algorithm for fast solution of 2-D scattering problems and current flow modeling on unstructured meshes,” *IEEE Trans. Microw. Theory Techn.*, vol. 67, no.12, pp. 4833-4847, 2019

[10] M. Wang, C. Qian, E. Di Lorenzo, L. J. Gomez, V. Okhmatovski, and A. C. Yucel, “SuperVoxHenry: Tucker-enhanced and FFT-accelerated inductance extraction for voxelized superconducting structures,” arvix preprint, 2021

[11] I. V. Oseledets, D. V. Savostianov, and E. E. Tyrtyshnikov, “Tucker dimensionality reduction of three-dimensional arrays in linear time,” *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 939–956, 2008.

[12] M. V. Rakhuba and I. V. Oseledets, “Fast multidimensional convolution in low-rank tensor formats via cross approximation,” *SIAM J. Sci. Comput.*, vol. 37, no. 2, pp. A565-A582, 2015.

[13] X. Fang, *et al*. “Multiscale compressed and spliced Sherman–Morrison–Woodbury algorithm with characteristic basis function method,” *IEEE Trans. on Electromagnetic Compatibility*, vol. 60, no. 3, pp. 716-724, 2017.