A Linear Complexity Direct Finite Element Solver for Large-Scale 3-D Electromagnetic Analysis

Bangda Zhou* and Dan Jiao
School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA

***Abstract*—A direct finite-element solver of linear complexity is developed to analyze general 3-D electromagnetic problems where conductors and dielectrics can be arbitrarily shaped, inhomogeneous, and lossy. Numerical experiments have demonstrated its superior performance over state-of-the-art direct finite-element solvers as well as an iterative finite-element solver. A linear complexity in both CPU time and memory consumption has been theoretically proved and numerically verified with controlled accuracy. The finite-element matrix of an electrodynamic problem having over 5.64 million unknowns is factorized in less than 2.5 hours on a single core running at 2.8 GHz.**

I. Introduction

State-of-the-art fast finite element method (FEM)-based solvers rely on iterative approaches to solve large-scale electromagnetic problems. The computational complexity of an iterative solver is at best $O(N_{it}N_{rhs}N)$, where N_{it} is the number of iterations, and N_{rhs} the number of right hand sides. When N_{it} and N_{rhs} are large, an iterative solution becomes inefficient. A traditional direct solution is computationally expensive. It is shown in [1] that the optimal operation count of the direct solution of an FEM matrix in *exact* arithmetic is $O(N^{1.5})$ for 2-D problems, and $O(N^2)$ for 3-D problems. Although there have been successes in speeding up the direct finite element solution [2], [3] for electromagnetic analysis, as yet, no $O(N)$ complexity, i.e. optimal complexity, has been accomplished for FEM-based direct solutions of general 3-D electromagnetic problems. The contribution of this paper is a direct FEM solver of linear complexity for general 3-D electromagnetic analysis. Both theoretical analysis and numerical experiments have demonstrated its linear complexity and superior performance.

II. Proposed Linear-Complexity Direct Solver

In the proposed solver, we fully take advantage of the zeros in the original FEM matrix, and also maximize the zeros in $\mathbf{L}$ and $\mathbf{U}$ by nested dissection ordering [1]. Unlike [3] that represents the $\mathbf{L}$ and $\mathbf{U}$ as a whole $\mathcal{H}$-matrix [4], we only store the nonzero blocks in $\mathbf{L}$ and $\mathbf{U}$ with a compact error-controlled $\mathcal{H}$-matrix representation, compute these nonzero blocks efficiently by developing fast $\mathcal{H}$-matrix based algorithms, while removing all the zeros in $\mathbf{L}$ and $\mathbf{U}$ from storage and computation. Moreover, we organize the factorization of the original 3-D finite element matrix into a sequence of factorizations of 2-D dense matrices, and thereby control the rank to follow a 2-D based growth rate, which is much slower than a 3-D based growth rate [5]. The overall algorithm has six major steps:

1. Build cluster tree $T_{\mathcal{I}}$ based on nested dissection
2. Build elimination tree $E_{\mathcal{I}}$ from $T_{\mathcal{I}}$
3. Obtain the boundary for each node in $E_{\mathcal{I}}$
4. Generate the $\mathcal{H}$-matrix structure for each node
5. Do numerical factorization guided by $E_{\mathcal{I}}$ by $\mathcal{H}$-matrix-based fast algorithms
6. Solve for a right hand side based on $E_{\mathcal{I}}$

The steps of building the cluster tree $T_{\mathcal{I}}$ [4] and the elimination tree $E_{\mathcal{I}}$ [2] by nested dissection restructure the original matrix and minimize the number of fill-ins introduced during factorization. We recursively partition a 3-D computational domain into 8 subdomains using three surface separators until the number of unknowns in each subdomain is no greater than $leafsize$ (a pre-determined constant). We then build the elimination tree level by level with the bottom-level nodes being the subdomains having $leafsize$, and the nodes at other levels comprising the surface separators at different levels. The root node is the largest separator. The LU factorization of the FEM matrix is a bottom-up traversal of the elimination tree. Let each node in the elimination tree be N^i. We identify a tight boundary for it, denoted by $bnd(N^i)$, which is the minimal set of unknowns that would be modified in factorizing the matrix associated with N^i, $\mathbf{Y}_{N^i\times N^i}$. Here, $\mathbf{Y}$ is the matrix assembled during the direct solution procedure. We then form the frontal matrix $\mathbf{F}^i$ for each node N^i,

$$\mathbf{F}^i = \begin{pmatrix} \mathbf{Y}_{N^i\times N^i} & \mathbf{Y}_{N^i\times bnd(N^i)} \\ \mathbf{Y}_{bnd(N^i)\times N^i} & \mathbf{Y}_{bnd(N^i)\times bnd(N^i)} \end{pmatrix} \tag{1}$$

which is a dense matrix of size $\#node + \#boundary$. Since each non-leaf node in the elimination tree is a 2-D surface separator, the boundary of each node is essentially the union of the unknowns residing on the bounding box of the 2-D separator. Thus, the boundary size is proportional to the node size, which is 2-D. We hence reduce the factorization of the original large 3-D FEM matrix to a sequence of factorizations of 2-D dense matrices shown in (1). We then develop efficient $\mathcal{H}$-matrix based algorithms to accelerate the dense matrix computation. Each intermediate dense matrix (1) can be viewed as the Schur complement of the original sparse FEM matrix in a 2-D domain (a surface separator), which is nothing but the sum of the original sparse FEM matrix in the 2-D domain and the contribution from its children domains

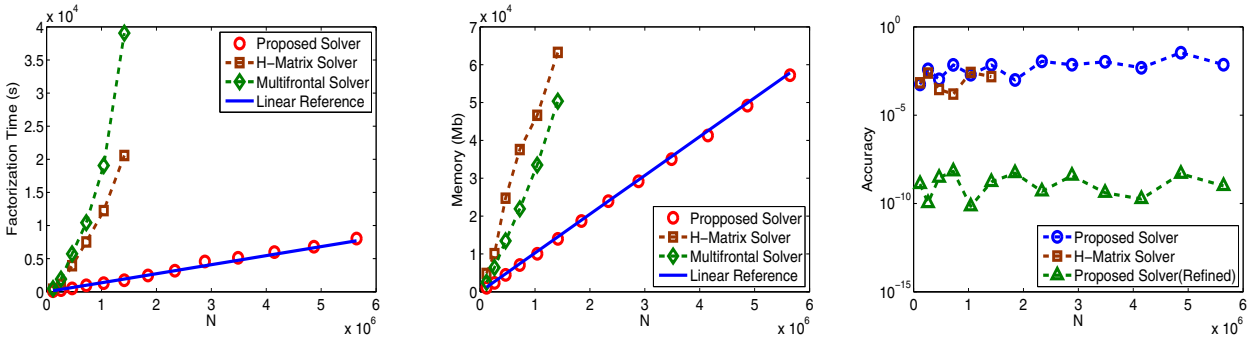

Fig. 1. Simulation of a large-scale 3-D array from 117,287 to 5,643,240 unknowns at 100 GHz. (a) LU factorization time. (b) Memory. (c) Accuracy.

to this 2-D domain. Since the rank of the inverse of a matrix sum is bounded by the rank of the inverse of any one of the matrices, the rank of each intermediate dense matrix is bounded by the rank of the inverse of the original sparse FEM matrix in a 2-D domain. Hence, the rank's growth rate follows a 2-D based growth rate rather than a 3-D based growth rate, which has also been numerically verified.

Complexity and Accuracy Analysis: Considering a 3-D computational domain, the unknown number of which along each direction is n. The total unknown number is thus $N = n^3$. Denoting the depth of the elimination tree by L, we have $L = log_2 n$ and hence, $N = 8^L$. Let the root level be $l = L$. At level l of the elimination tree, there are 8^{L-l} nodes, and the matrix size of each node is $m = O(2^l \times 2^l)$ because a node at level l is either a 2-D surface separator or a leaf domain. Since the operations associated with each node are performed in an $\mathcal{H}$-based fast algorithm, the computational cost of each node of size m is significantly reduced from m^3 to $r_l^2 m log^2 m$ [3], where r_l denotes the rank at the l-th level. As a result, we obtain

$$\text{Time Complexity} = \sum_{l=1}^{L} 8^{(L-l)} r_l^2 (2^l \times 2^l) log_2^2 (2^l \times 2^l). \quad (2)$$

Since each node is a 2-D surface separator, the rank r_l follows the 2-D based growth rate, thus it is proportional to the square root of the logarithm of the electrical size of the 2-D surface [5], and hence $r_l = rank_{2D} = O(\sqrt{log 2^l}) = O(\sqrt{l})$. Substituting it into (2), we obtain

$$\text{Time Complexity} = 8^L \sum_{l=1}^{L} l(2l)^2 / 2^l = O(N), \quad (3)$$

which is linear. The last equality holds true because the denominator grows with l much faster than the numerator. Similarly, the storage complexity of the proposed solver can be proved to be $O(N)$. In the proposed solver, we represent the intermediate dense matrices associated with each node (a leaf domain or a separator) in the elimination tree by an $\mathcal{H}$-matrix. The accuracy of such an $\mathcal{H}$-matrix representation can be proved from the fact that the original FEM matrix has an exact $\mathcal{H}$-representation and its inverse has an error-bounded $\mathcal{H}$-representation [3].

III. Numerical Results

A large-scale 3-D array structure [3] at *100* GHz is simulated on a unix server with a 2.8 GHz AMD CPU. The mesh density is fixed, while the array-element number is increased from 2×2 to 14×14, resulting in an unknown number ranging from 117,287 to 5,643,240. The electrical size of the largest array is around 15.6 wavelengths. The conductors are also discretized because of finite conductivity. The simulation parameters are $leafsize$=8 and η=2.5. In Fig. 1, we plot the factorization time, the memory cost, and the accuracy of the proposed solver versus N, in comparison with a state-of-the-art multifrontal solver and an $\mathcal{H}$-matrix based direct solver. The accuracy is measured by relative residual. The refined accuracy plotted in triangular marks is obtained by adding a few (less than 10) steps of iterative refinement (http://en.wikipedia.org/wiki/Iterative_refinement) after the solution is obtained, the cost of which is negligible since $\mathbf{L}$ and $\mathbf{U}$ have been computed. Neither the multifrontal solver nor the $\mathcal{H}$-matrix solver is capable of simulating larger than 7×7 arrays due to their large memory requirements. In contrast, the proposed direct solver not only greatly outperforms the two solvers but also demonstrates a clear linear complexity with excellent accuracy achieved in the entire unknown range. The proposed direct solver was also compared with a commercial-grade iterative FEM solver. For the 7×7 array having 98 right hand sides, it took the iterative FEM solver 8102 s to solve the problem with a matrix of size N=661,682 without discretizing into the conductors, whereas in a similar amount of time, the proposed direct solver solved the 14×14 array having over 5.64 million unknowns.

References

[1] A. George, "Nested Dissection of a Regular Finite Element Mesh," *SIAM J. on Numerical Analysis*, vol. 10, no. 2, pp. 345-363, Apirl 1973.

[2] J. W and H. Liu, "The Multifrontal Method for Sparse Matrix Solution: Theory and Practice," *SIAM Review*, vol. 34, pp. 82-109, March 1992.

[3] H. Liu and D. Jiao, "Existence of $\mathcal{H}$-matrix Representations of the Inverse Finite-Element Matrix of Electrodynamic Problems and $\mathcal{H}$-Based Fast Direct Finite-Element Solvers," *IEEE Trans. MTT*, vol. 58, no. 12, pp. 3697-3709, Dec. 2010.

[4] S. Borm, L. Grasedyck, and W. Hackbusch, *Hierarchical matrices*, Lecture note 21 of the Max Planck Institute for Mathematics in the Sciences, 2003.

[5] H. Liu et. al., "A Theoretical Study on the Rank's Dependence with Electric Size of the Inverse Finite Element Matrix for Large-Scale Electrodynamic Analysis," *IEEE Int. Symp. Ant. Propagat.*, July 2012.