Progress In Electromagnetics Research Letters, Vol. 96, 91–96, 2021

Memory Reduced Half Hierarchal Matrix (H-Matrix) for Electrodynamic Electric Field Integral Equation

Yoginder K. Negi*

Abstract—This letter shows 50 percent memory saving for a Hierarchal Matrix (H-matrix) by converting a regular H-matrix to symmetric H-matrix for large and complex electrodynamic problems. Only the upper diagonal near-field and compressed far-field matrix blocks of the H-matrix are stored. Far-field memory saving is achieved by computing and keeping the upper diagonal far-field blocks leading to compressed column block U and row block V at a level. Due to symmetry, the lower diagonal far-field H-matrix compressed column is transpose of V, and the compressed row block is transpose of U. Storage and computation of lower diagonal blocks are not required. Similarly, in the case of near-field, only the upper diagonal near-field blocks are computed and stored. Numerical results show that the proposed memory reduction procedure retains the accuracy and cost of regular H-matrix.

1. INTRODUCTION

Integral Equations (IE) are accurate and popular methods in Computational Electromagnetics (CEM) for solving large and complex electromagnetic problems numerically. Compared to differential equation-based methods like Finite Difference Time Domain (FDTD) and Finite Element Method (FEM), IEs are free from grid dispersion error and lead to fewer unknowns. IE-based Method of Moment (MoM) [1, 2] can be used to solve radiation/scattering problems in electromagnetics. MoM gives a dense matrix with $O(N^2)$ matrix fill time and memory requirement for $N \times N$ size matrix. Solving the MoM system of equations leads to $O(N^3)$ solution time with direct solver and $N_{itr}O(N^2)$ solution time with a conventional iterative solver for N_{itr} iterations. High matrix storage, computation, and solving cost limit the application of MoM for solving large problems in electromagnetics. Matrix compression methods like Multi-level Fast Multipole Algorithm (MLFMA) [3], Adaptive Cross Approximation (ACA) [4, 5], Hierarchal Matrices (H-matrix) [6–9] reduce high matrix computation and storage cost to $O(N \log N)$. The matrix solution time decreases to $N_{itr}O(N \log N)$ for N_{itr} iterations. Even with $O(N \log N)$ memory complexity, as the problem size grows, the memory requirement becomes significant.

MoM matrices in electrostatic and 2D problems [1, 2] are symmetric due to the Galerkin testing procedure. The matrix storage memory can be reduced by exploiting symmetric property and solving it with the iterative solver. For memory saving, only the upper diagonal block of the symmetric matrix is computed and stored. Due to symmetry, the lower diagonal block is the transpose of the upper diagonal block. The upper diagonal matrix-vector product is added to the lower diagonal matrix-vector for the complete iterative solution matrix-vector product. Following this procedure of memory reduction limits the application of electrostatic and 2D MoM to smaller unknown sized geometries as the computation cost and memory requirement grow with $O(N^2)$. In 3D electrodynamic MoM, the matrix loses its symmetric property due to the approximation made in scalar and vector potential computation [2, 10]. Hence, direct symmetric matrix memory reduction does not apply to electrodynamic MoM matrix and different fast solvers. Bodies of Revolution (BOR) [11] is a popular method for reducing memory of the

Received 8 December 2020, Accepted 4 February 2021, Scheduled 13 February 2021
* Corresponding author: Yoginder Kumar Negi (yknegi@gmail.com).
The author is with the Supercomputer Education Research Center, Indian Institute of Science Bangalore, India.

system matrix to half, and [12] shows the application of BOR for solving big problems in conjunction with compression methods. BOR memory reduction method is limited to axially rotational symmetric bodies, limiting its application to a few structurally symmetric geometries, whereas real-world electromagnetic problems are complex and asymmetric. The use of symmetric H-matrix properties to develop fast LR factorization is shown in [8, 13] but is only applicable to symmetric positive definite matrices.

The paper shows that regular electrodynamic MoM based H-matrix can be made symmetric by averaging the upper and lower diagonal values, along with 50 percent memory saving for conventional H-matrix. By exploiting the ACA compressed fast solver's algebraic nature, only the upper diagonal part is computed and stored. The upper diagonal part consists of half near-field and half far-field blocks of H-matrix, converting full H-matrix to half H-matrix. Numerical results show the accuracy and efficiency of the proposed half H-matrix. The rest of the paper is organized as follows. Section 2 gives a brief description of electrodynamic EFIE MoM. Section 3 describes the proposed memory reduced half H-matrix. Section 4 shows the efficiency and accuracy of the proposed half H-matrix. Section 5 concludes the paper.

2. INTEGRAL EQUATIONS

In this work, Electric Field Integral Equation (EFIE) is used for MoM matrix computation. The 3D full-wave MoM governing equation for EFIE is given as:

$$E_i = -j\omega A - \nabla\phi \tag{1}$$

where E_i is the incident electric field or excitation on the given geometry; A and ϕ represent the vector and scalar potentials; and ω is the angular frequency. Using the Galerkin method and testing with RWG basis function [10], the resultant MoM matrix is dense, and its elements are given by:

$$Z(i,j) = \frac{j\omega\mu}{4\pi}\int_{T_t} f_t \int_{T_s} \frac{\exp^{-jk|r-r'|}}{|r-r'|} f_s ds \cdot dt + \frac{1}{j\omega 4\epsilon}\int_{T_t} \nabla f_t \int_{T_s} \frac{\exp^{-jk|r-r'|}}{|r-r'|} \nabla f_s ds \cdot dt \tag{2}$$

where μ and ϵ represent the permeability and permittivity of the background material; k is the wave-number; f_t and f_s are the test and source bases with triangles T_t and T_s respectively; and r and r' are the global coordinates of the test and source points. Here, f_t and f_s are expanded using the RWG basis function. Following the above procedure for matrix filling gives a dense matrix with the $O(N^2)$ complexity of matrix storage and filling time. The matrix storage and fill time can be reduced by incorporating fast matrix solution methods. These methods work on the principle of matrix compressibility of far-field interaction blocks. Algebraic compression methods like ACA [4, 5], IE-QR [14], Re-compressed ACA [15], H-matrix [6–9], and Re-Compressed H-matrix [16] are kernel independent and easy to implement, which gives an advantage compared to an analytic method like MLFMA. In this work, we use conventional ACA [15] based H-matrix [6, 7] to take advantage of algebraic compression and reduce the overall matrix storage requirement. The next section describes the proposed method to reduce matrix memory.

3. HALF H-MATRIX

The complexity of $O(N^2)$ memory and solution time of MoM can be reduced to $O(N \log N)$ by adapting H-matrix decomposition. For H-matrix construction, binary-tree based 3D geometry decomposition is used with ACA compressed far-field and dense near-field blocks. The matrix compression is used for block interaction, satisfying the admissibility condition.

$$\eta dis(\Omega_t, \Omega_s) \geq \min(dia(\Omega_t), dia(\Omega_s) \tag{3}$$

The admissibility condition of Eq. (3) states that for matrix compression admissibility constant η times the distance between the test Ω_t and the source block Ω_s must be greater than or equal to the minimum of the diameter of the test or source block. The binary-tree partition is done until the number of elements in the block is less than or equal to 40 basis elements. At the leaf level, the block interaction not satisfying the admissibility condition is considered a near-field interaction. In the multi-level binary-tree case, the far-field block meeting admissibility condition interacted at a higher level does not interact at the lower level, and complete matrix structure is shown in [16].

Electrodynamic MoM matrix by nature of computation is not symmetric; subsequently, it can be made symmetric by computing upper and lower diagonal matrix elements and averaging them, as shown in Eq. (4). The averaged value forms the upper and lower diagonal elements of the MoM matrix. For averaging in Eq. (4), the first matrix element is computed for the ith test edge and jth source edge, i.e., (i, j), and the second matrix element is computed for the jth test edge and ith source edge. Both computed values are averages and substituted to upper diagonal location $Z(i, j)$ and lower diagonal location $Z(j, i)$. By averaging the matrix value and substituting at two locations, the computed MoM matrix becomes diagonally symmetric; we can save memory by storing only the upper diagonal part, i.e., $(i \leq j)$ of the matrix.

$$Z(i,j) = Z(j,i) = \frac{Z(i,j) + Z(j,i)}{2} \tag{4}$$

Further, to reduce the memory requirement for large problems, the symmetric property is extended to conventional H-matrix. H-matrix is a combination of near-field and multi-level compressed far-field matrix. Computing near-field block elements by Eq. (4), we get a symmetric near-field block matrix. Only upper diagonal near-field blocks are computed and stored. Memory and precondition computation time savings due to the symmetric property of near-field is discussed and shown in [17]. In multi-level far-field matrix compression, matrices are compressed with ACA for upper diagonal block matrices for well-separated test cluster with m edges and source cluster with n edges satisfying admissibility condition Eq. (3). The far-field upper diagonal sub-block matrix $[Z_{sub}]$ matrix compression at a particular level gives $U \times V$ block matrices. Where, U is column matrix of size $m \times r$ and V is row matrix of size $r \times n$ with m as row size, n as column size and r as the rank of $[Z_{sub}]$. Computing each element with Eq. (4) during compression gives symmetric far-field blocks. Now due to symmetric property at specific binary-tree level interaction, the lower diagonal far-field sub-block matrix is the transpose of the upper diagonal far-field sub-block, giving far-field block as $[Z'_{sub}]$. Compressed $[Z'_{sub}]$ forms a compressed column block and compressed row block. The compressed column block is the transpose of the upper diagonal compressed row block matrix, giving V' of size $n \times r$. Similarly, $[Z'_{sub}]$ compressed row block is the transpose of the upper diagonal compressed column block matrix, giving U' of size $r \times m$. Now the lower diagonal far-field compressible submatrix $[Z'_{sub}]$ can be represented as by $V' \times U'$ of size $n \times m$ and rank r, as shown in Fig. 1. matrix $[Z_{sub}]$ is used to save the storage and computation of $[Z'_{sub}]$.

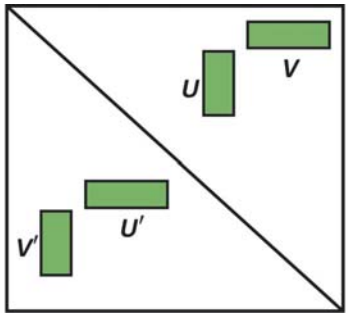

Figure 1. The figure shows the symmetric H-matrix compression for a compressible far-field block at a level.

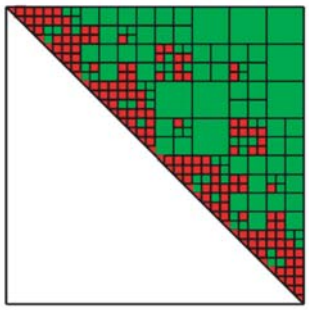

Figure 2. The figure shows the half H-matrix based on a binary-tree, Level 5, green as compressible blocks, and red non-compressible blocks.

The proposed symmetric H-matrix computation retains the $O(N \log N)$ matrix computation and matrix-vector product complexity. Storage memory savings can be achieved by only storing the upper diagonal near-field and far-field matrices, giving half H-matrix as shown in Fig. 2, where green blocks are compressible, and red blocks are non-compressible blocks. The complete matrix-vector product is carried out by adding upper and lower diagonal matrix-vector products. Matrix-vector products for lower diagonal blocks are transpose of upper diagonal blocks. Numerical results and comparisons are presented in the next section to validate accuracy and efficiency.

4. NUMERICAL RESULTS

In this section, we show the efficiency and accuracy of the proposed half H-matrix. All the computations are carried out for the double-precision data type. GMRES with convergence tolerance 1e-6 is used for matrix solution with symmetrix matrix-vector product computation discussed in Section 3. A computation system with 128 GB memory and Intel (Xeon E5-2670) processor was used.

4.1. Complexity

In this subsection, we demonstrate that the proposed half H-matrix fast solver method retains the $O(N \log N)$ solution complexity for matrix fill time, matrix-vector multiplication, and total memory. The figures below show the time and memory complexity for the sphere with increasing size and unknown.

As observed from Figs. 3, 4, and 5, the memory reduced half H-matrix retains $O(N \log N)$ complexity of matrix fill time, matrix-vector product time, and memory of a regular H-matrix.

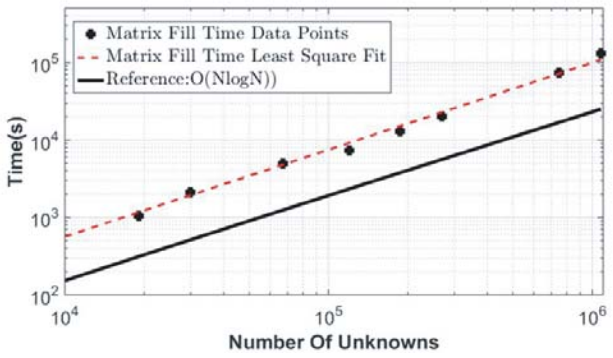

Figure 3. CPU time for a matrix fill computation of the PEC sphere with increasing unknown.

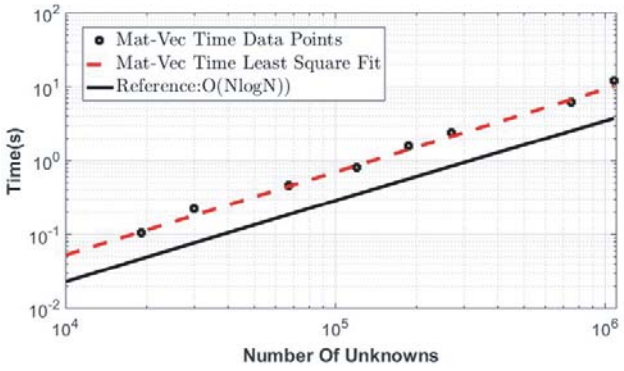

Figure 4. CPU time for a matrix-vector multiplication of PEC sphere with increasing unknown.

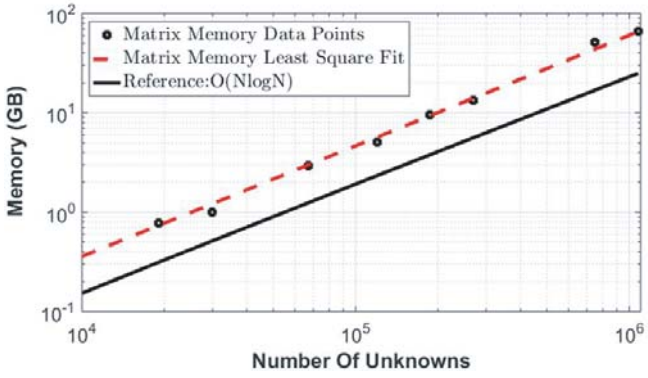

Figure 5. Total matrix storage memory requirement for half H-matrix of PEC sphere with increasing unknown.

4.2. Accuracy and Memory Savings

The subsection shows the accuracy and efficiency of the proposed method. Computed bi-static Radar Cross Section (RCS) results from the proposed half H-matrix are compared with the analytical results and H-Matrix for canonical and complex structure.

4.2.1. PEC Sphere

Here, we have compared bi-static RCS of a 10λ radius PEC sphere discretized with a $\frac{\lambda}{10}$ mesh resulting in 520,134 unknowns. The bi-static RCS from half H-matrix is compared with the Mie series analytical solution for observation angle $\theta = 0°$ to $180°$ for $\phi = 0°$ with VV polarized plane wave incident at $\theta = 0°$ and $\phi = 0°$. Fig. 6 shows the agreement of bi-static RCS computed from the proposed method with Mie series. Memory saving along with matrix fill and soluation time for 5,128 iterations are shown in Table 1.

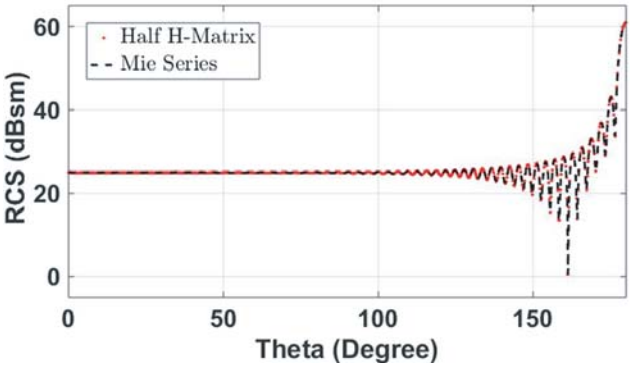

Figure 6. Bi-static RCS of 10λ sphere for observation angles $\phi = 0°$ to $180°$, $\phi = 0°$ and VV polarized plane wave incident at $\theta = 0°$, $\phi = 0°$.

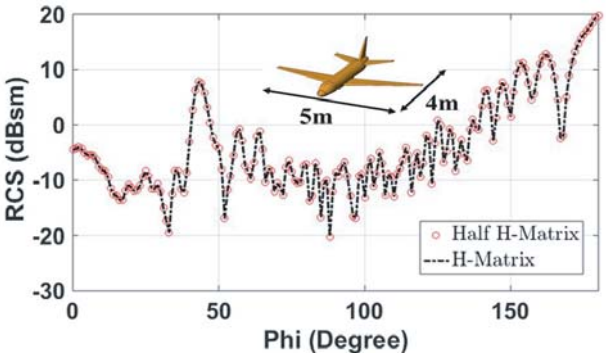

Figure 7. Bi-static RCS of model aircraft at 1.5 GHz for VV polarized plane wave nose incident at $\theta = 90°$, $\phi = 0°$ and observation angles at $\theta = 90°$, $\phi = 0°$ to $180°$.

Table 1. Memory and time for PEC sphere.

	H-matrix	Half H-matrix
Memory (GB)	45.0900	22.6768
Matrix Fill Time (H)	2.61	2.61
Solution Time (H)	10.13	10.13

4.2.2. Model Aircraft

In this example, we consider a model aircraft with a length of 4 m and wingspan 5 m. With $\frac{\lambda}{10}$ discretization of the geometry, the meshing scheme generates 155,472 unknowns. Fig. 7 shows the computed bi-static RCS at 1.5 GHz in the x-y plane with VV polarized plane wave incident at $\theta = 90°$, $\phi = 0°$, and observation angle from nose to tail $\phi = 0°$ to $\phi = 180°$ and $\theta = 90°$ from the proposed method and H-matrix. The figure shows the complete agreement of bi-static RCSs from the two methods. Table 2 shows memory saving, matrix fill time, and solution time for 11,283 iterations.

Table 2. Memory and time for model aircraft.

	H-matrix	Half H-matrix
Memory (GB)	16.8680	8.7067
Matrix Fill Time (H)	5.21	5.22
Solution Time (H)	19.31	19.31

5. CONCLUSION

The letter shows the symmetrization of regular H-matrix. Memory saving is achieved by computing and saving symmetric upper half H-matrix. The memory reduced half H-matrix is applied to solve canonically shaped and arbitrary complex geometric structures. The proposed method reduces the memory requirement by 50 percent while retaining the complexity of the regular H-matrix. The memory saving becomes significant while solving large problems. The method is amenable to efficient parallelization due to the algebraic nature of computation.

REFERENCES

1. Harrington, R. F., *Field Computation by Moment Methods*, Wiley-IEEE Press, New York, 1993.
2. Gibson, W. C. *The Method of Moments in Electromagnetics*, CRC Press, 2014.
3. Chew, W. C., J. M. Jin, E. Michielssen, and J. Song, *Fast Efficient Algorithms in Computational Electromagnetics*, Artech House, Boston, London, 2001.
4. Bebendorf, M., "Approximation of boundary element matrices," *Numerische Mathematik*, Vol. 86, No. 4, 565–589, Jun. 2000.
5. Kurz, S., O. Rain, and S. Rjasanow, "The adaptive cross-approximation technique for the 3-D boundary element method," *IEEE Transactions on Magnetics*, Vol. 38, No. 2, 421–424, Mar. 2002.
6. Hackbusch, W., "A sparse matrix arithmetic based on H-matrices. Part I: Introduction to H-matrices," *Computing*, Vol. 62, No. 2, 89–108, 1999.
7. Hackbusch, W. and B. N. Khoromskij, "A sparse H-matrix arithmetic. Part II: Application to multi-dimensional problems," *Computing*, Vol. 64, 21–47, 2000.
8. Börm, S., L. Grasedyck, and W. Hackbusch, "Hierarchical matrices," *Lecture Notes*, 21, 2003.
9. Chai, W. and D. Jiao, "H and H^2 matrix-based fast integral-equation solvers for large-scale electromagnetic analysis," *IET Microwaves, Antennas and Propagation*, No. 10, 1583–1596, 2010.
10. Rao, S. M., D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on Antennas and Propagation*, Vol. 30, No. 3, 409–418, May 1982.
11. Andreasen, M., "Scattering from bodies of revolution," *IEEE Transactions on Antennas and Propagation*, Vol. 13, No. 2, 303–310, 1965.
12. Su, T., D. Ding, Z. Fan, and R. Chen, "Efficient analysis of EM scattering from bodies of revolution via the ACA," *IEEE Transactions on Antennas and Propagation*, Vol. 62, No. 2, 983–985, 2013.
13. Benner, P. and T. Mach, "The LR Cholesky algorithm for symmetric hierarchical matrices," *Linear Algebra and its Applications*, Vol. 439, No. 4, 1150–1166, 2013.
14. Kapur, S. and D. E. Long, "IES3: Efficient electrostatic and electromagnetic solution," *IEEE Computer Science and Engineering*, Vol. 5, No. 4, 60–67, Oct.–Dec. 1998.
15. Bebendorf, M. and S. Kunis, "Recompression techniques for Adaptive Cross Approximation," *Journal of Integral Equations and Applications*, Vol. 21, No. 3, 331–357, 2009.
16. Negi, Y. K., V. P. Padhy, and N. Balakrishnan, "Re-compressed H-matrices for fast electric field integral equation," *IEEE-International Conference on Computational Electromagnetics (ICCEM 2020)*, Singapore, Aug. 24–26, 2020.
17. Negi, Y. K., N. Balakrishnan, and S. M. Rao, "Symmetric near-field Schur's complement preconditioner for hierarchal electric field integral equation solver," *IET Microwaves, Antennas and Propagation*, Vol. 14, No. 14, 1846–1856, Aug. 2020.