An HSS-Matrix-Based Fast Direct Solver with Randomized Algorithm

Weikang Yu, Hu Yang, Shengguo Li, and Yanlin Xu

College of Electronic Science and Engineering
National University of Defense Technology, Changsha, 410073, China
769280669@qq.com, yanghu90@163.com, shengguolsg@126.com, 13298656824@163.com

Abstract **─** Discretization of the electric field integral equation (EFIE) generally leads to dense impedance matrix. The resulting matrix, however, can be compressed in sparsity based on hierarchical structure and low rank approximation. In this paper, we propose an HSS-matrix-based fast direct solver for surface integral equation (SIE) that has a compression complexity of O(rN^2) to analysis the large-scale electromagnetic problems, where r is a modest integer. The proposed solver efficiently compresses the dense matrices using a randomized algorithm and requires modest memory. Efficiency and accuracy is validated by numerical simulations. In addition, being an algebraic method, the HSS-matrix-based fast solver employs Green's kernels and hence is suitable for other integral equations in electromagnetism.

Index Terms **─** Electric field integral equation, HSS matrix, randomized algorithm.

I. INTRODUCTION

Method of Moment (MoM) is a powerful numerical method to obtain the electromagnetic characteristics based on the solution of integral equation (IE) on the surface of the object [1]. The discretization process usually generates dense impedance matrices with memory requirement of O(N^2) and operation complexity of O(N^3), where N represents the matrix size. In contrast, iterative methods often win the favor in many engineering applications with time complexity of O($N_{iter}N^2$), where N_{iter} is the number of iterations. Fast iterative algorithms, e.g., CG, GMRES and BiCGStab, generally being based on a Krylov subspace, take advantage of the rapid matrix-vector multiplication. The typical one is the multilevel fast multi-pole algorithm (MLFMA) which reduces the complexity to O($NlogN$) [2].

However, fast direct solvers have been rapidly developing in the resent years, and before proceeding any further, we would like to point out some relevant features of the direct methods. One of the most important advantages of the direct solver is that it works well with multiple right hand sides, which means once the matrix is efficiently compressed all right hand sides can be considered with low computational cost [3]. Therefore, fast direct solvers offer great advantage especially in solving monostatic scattering problems. Besides, the direct method is robust while the iterative method is highly sensitive to the condition of the matrix itself. Finding a good preconditioner is difficult and imperative to avoid ill-conditioning.

In this paper, we introduce the rank-structured matrix to deal with the dense equation arising in MoM [4]. These theories all rely on the property that the off-diagonal blocks have low-rank properties and thus can be well-approximated. Such matrices are termed hierarchical matrices, which include H-matrix [5], H2-matrx [6], Hierarchically Semi-separable (HSS) [7,8], SSS [9], and so on. In some engineering applications, the given matrices are often rank-deficient or nearly so, which can be divided into a hierarchical structure and then compressed exactly or approximately using SVD or rank-revealing QR factorization. Due to this special structure, those dense matrices can be described as data sparse. The HSS method in this paper was first introduced by Chandrasekaran, Gu, Pals and others. Compare with earlier H-matrix, it develops a recursive relation between the generators appearing at different levels of recursions [6].

From a mathematical point of view, existing low-rank compression algorithm based on IE methods, although their exact methods could be different, utilizes the low-rank properties inherently in the impedance matrix [10]. HSS matrix-based solvers for Volume Integral Equation (VIE) and Surface Integral Equation (SIE) have been proposed in [11, 12]. In this work, we develop a new HSS-matrix-based fast direct solver based on randomized sampling algorithm. To be specific, ULV factorization method and Interpolative Decomposition (ID) method are adopted in this work which have not been seen in the existing literature [7]. Major advantages of this method are that the whole impendence matrix does not need to be explicitly formed and only requires the access to selected elements. Besides, a space grouping method for basis functions is also proposed to improve the low-rank properties of impedance matrices.

Submitted On: August 8, 2017
Accepted On: June 8, 2018

II. HSS MATRIX THEORY

Consider the electric field integral equation (EFIE):

$$\boldsymbol{n}\times ik\eta\iint_S\left(\begin{array}{l}\boldsymbol{J}(\boldsymbol{r}')g(\boldsymbol{r},\boldsymbol{r}')\\ +\dfrac{1}{k^2}\nabla'\cdot\boldsymbol{J}(\boldsymbol{r}')\nabla g(\boldsymbol{r},\boldsymbol{r}')\end{array}\right)dS'=\boldsymbol{n}\times\boldsymbol{E}^{inc},\quad(1)$$

in which $g(\boldsymbol{r},\boldsymbol{r}')=\frac{e^{-jk|\boldsymbol{r}-\boldsymbol{r}'|}}{|\boldsymbol{r}-\boldsymbol{r}'|}$ represents Green's function for free space. The Equation (1) can be compactly written as:

$$n\times L(J)=n\times E^{inc}(\boldsymbol{r}),\qquad r\in S,\quad(2)$$

where L represents a linear operator, J is the induced surface current density, and E^{inc} is the imposed electric field. Discretizing J with a series of RWG (Rao-Wilton-Gilsson) basis functions results in the following linear system of equations:

$$ZI=V.\quad(3)$$

Here, we introduce the HSS matrix to show how to represent the matrix Z in data-sparse-form and underline the numbering rules of the RWG functions.

An HSS representation of matrix Z relies on a recursive clustering of the index set I={1,…,n}. This partitioning is represented by a tree T, and there is no restriction on the shape of the tree, and usually a binary tree is used. Each node v of the tree is associated with a subset I_v of I. The subset associated with the root node is I={1,…,n}, and for non-leaf node v with left child υ1 and right child υ2:

$$I_\tau=I_{\upsilon1}\cup I_{\upsilon2},\cdots I_{\upsilon1}\cap I_{\upsilon2}=\phi,\quad(4)$$

and $\bigcup_{v\in LN} I_\upsilon=I$ where LN denotes the set of all leaf nodes. We further assume T is postordered, i.e., the nodes are ordered so that a non-leaf node v satisfies $v_1<v_2<v$, as done in [13, 14]. Furthermore, for each node v, let I_v^L be the set of all indices less than those in I_v and I_v^R be the set of all indices greater than those in I_v. Thus, $I=I_v^L\cup I_v\cup I_v^R$ for each node v. Figure 1 illustrates these sets.

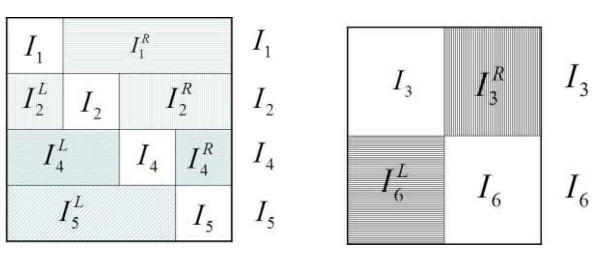

Fig. 1. Matrix partition and the corresponding index sets.

For each node v in T, there are matrices $D_v^{big},U_v^{big},V_v^{big}$ and Σ_v associated with it, called *generators*, such that:

$$D_v^{big}=Z(I_v,I_v)=\begin{bmatrix}D_{v1}^{big} & U_{v1}^{big}\Sigma_{v1}(V_{v2}^{big})^T\\ U_{v2}^{big}\Sigma_{v2}(V_{v1}^{big})^T & D_{v2}^{big}\end{bmatrix},\quad(5)$$

$$U_v^{big}=\begin{bmatrix}U_{v1}^{big} & \\ & U_{v2}^{big}\end{bmatrix}U_v,V_v^{big}=\begin{bmatrix}V_{v1}^{big} & \\ & V_{v2}^{big}\end{bmatrix}V_v.\quad(6)$$

For a leaf node v, $D_v^{big}=Z_v,U_v^{big}=U_v,V_v^{big}=V_v$. Only the generators D_v,U_v,V_v and Σ_v are stored and the $(\bullet)^{big}$ matrices can be constructed recursively when needed. Z a 4×4 HSS matrix can be written as:

$$Z=\begin{bmatrix}\begin{bmatrix}Z_1 & U_1\Sigma_1V_2^T\\ U_2\Sigma_2V_1^T & Z_2\end{bmatrix} & U_3^{big}\Sigma_3(V_6^{big})^T\\ U_6^{big}\Sigma_6(V_3^{big})^T & \begin{bmatrix}Z_4 & U_4\Sigma_4V_5^T\\ U_5\Sigma_5V_4^T & Z_5\end{bmatrix}\end{bmatrix}.\quad(7)$$

There is a recursive relation between the generators appearing at different levels of recursion which is the essential difference between HSS and H^2-matrix with other classes of H-matrix. The Equation (6) shows how the generators at different levels are nested.

In practice, the low-rank off-diagonal blocks arise because the associated Green's function is smooth with source point and field point fast decay away from the diagonal [14]. Thus, the order of the rows and columns of matrix Z matters. If Z is shuffled randomly [15], the low-rank property could be lost or not exactly. Therefore, it is necessary to renumber the RWG functions before the generation of the impedance matrix based on SIE. Referring to the idea in MLFMA, the RWG functions on the surface of the PEC model are divided into several groups as is shown in Fig. 2. The lowest cube unit contains the information of the number of RWGs, and it can find father cube according to tree structure. Once the new order of the RWGs is determined, we can use the traditional MoM method to generate the linear system.

III. GENERATION OF THE HSS MATRIX

In this section, we introduce the main step in the construction of HSS matrix and give the analysis of time complexity for each step. We are not going to introduce it in detail since all the mathematic details of theory can be found in [16].

Compression: the HSS compression method is based on the randomized sampling algorithm introduced by Martinsson and the major component is the Interpolative Decomposition (ID) [17]. The complexity of the compression operation is O(rN^2) based on classical matrix-vector product, and r is the maximum rank found during the compression. The consequence of the algorithm is to construct the hierarchy form and special structure for generators U and V. One notable feature is that the original matrix could not be exact and only requires some selected elements and fast matrix-vector product routine. Therefore, if fast multipole method (FMM) is used, the complexity can drop to O(r^2N). We introduce the threshold ε which controls the size of generators and thus influence the memory-consuming in HSS representation, at the price of accuracy.

ULV factorization: in this factorization, orthogonal transformations are used to transform the problem of eliminating O(r) unknowns [7]. These remaining unknowns are eliminated using a standard LU factorization. The complexity is O(r^2N).

Solution: after ULV factorization, a linear system ZI = V is solved using triangular solution, which involves two processes: forward elimination and backward substitution. The complexity is O(rN). Furthermore, the accuracy of the solution can be improved by efficient iterative refinement (IR) method [16], which takes advantage of the fast matrix-vector product based on the HSS-matrix representation.

IV. NUMERICAL RESULTS

In this work, we illustrate the performance of the HSS-matrix-based solver in accelerating the direct solution of the EIEF-based analysis of electromagnetic problems. We validated the proposed solver on a conducting sphere (r=1m) grouped in three levels.

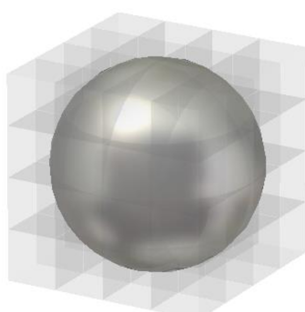

Fig. 2. Conducting sphere using space grouping.

Then the spherical surface was discretized into different number of RWGs by changing the frequency of incident wave. The matrix is built on complete binary trees with compression threshold $\varepsilon = 10^{-4}$ and iterative refinement in 5 times. We report time complexity for the HSS compression, the ULV factorization and the triangular solution. We tested the accuracy of the HSS-matrix representation by $err = \frac{\|Z*I-V\|_F}{\|V\|_F}$, where "$\|\bullet\|_F$" denotes the Frobenius norm and compare with the run time for solving a system with direct solver in Armadillo. The results are reported in Table 1. We observe from the table that the efficiency improves significantly based on HSS compared with the traditional solver in Armadillo. The compression and ULV factorization occupy the major time and iterative refinement time increases as complexity in matrix product increases. Thereby, balance must be found between the accuracy and the number of iterations.

Table 1: Time consumption in HSS-matrix based solver and in Armadillo

Unknowns	Compression (s)	ULV (s)	Solution +IR(s)	Total (s)	Armadillo (s)
2421	2.1	0.9	0.6	3.6	5.44
3624	9.3	1.8	0.7	11.8	20.3
4212	15.6	2.9	1.6	20.1	29.6
5808	23.8	5.7	2.8	32.3	78.8
6072	27.8	5.7	3.2	37.7	97.3
6612	29.9	5.9	3.7	39.6	124.5
7272	35.4	8.8	4.2	48.4	168.0

Next, we test the accuracy of HSS-based solver in each iterative refinement step as is shown in Fig. 3. The example also considers a conducting sphere illuminated by a normally incident plane wave. The sphere is discretized into 5808 unknowns. The compression threshold ε equals 10^{-4} and the max iteration is 9 times. Clearly good accuracy is obtained at the expense of matrix-vector product time. More importantly, the matrix-vector product process based on HSS matrix construction is more efficient than direct matrix-vector product.

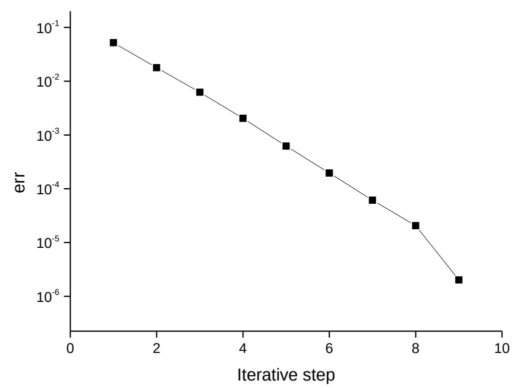

Fig. 3. HSS solution error in iterative refinement step.

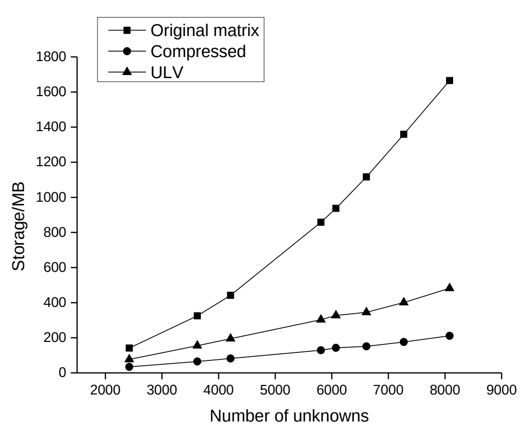

Fig. 4. Memory requirement for original matrix, compressed matrix and ULV factors.

In terms of memory, ignoring small temporary storage and communication buffers, the memory footprint for matrix Z in Armadillo is simply the storage of the matrix Z. In our new code, the matrix Z can be compressed in form of HSS binary tree together with ULV factors. In Fig. 4, the storage requirement in MB of the HSS-matrix-based solver was plotted with respect to the number of unknowns, from which the memory advantage of HSS structure can be clearly observed.

V. CONCLUSION

Integral-equation-based methods generally lead to dense systems of linear equation. In this paper, we introduce HSS matrix as a general algebraic framework to accelerate the solution of electromagnetic problem based on SIE and point out the importance of grouping the basic functions. Randomized sampling algorithm is introduced in compression process and the matrix in HSS form is factorized using classical ULV method. Numerical simulations have demonstrated the efficiency and accuracy of the HSS-matrix-based solver. We illustrate the memory cost before and after HSS compression and show the advantage in memory consumption. The HSS-matrix-based solver is also helpful in analyzing the field in actual radar systems because the characteristic of the impedance matrix will not change in radiation problem [18]. The randomized sampling algorithm in HSS is free of exact original matrix and high efficiency could be achieved with fast matrix-vector product, which is our future research topic.

ACKNOWLEDGMENTS

This work was supported by the Chinese National Science Foundation through Grant No. 11401580.

REFERENCES

[1] R. F. Harrington, *Field Computation by Moment Methods*. New York: Macmillan, 1968.

[2] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," *IEEE Antennas Propag.* Mag., vol. 35, no. 3, pp. 7-12, June 1993.

[3] S. Ambikasaran and E. Darve, "An O(*NlogN*) fast direct solver for partial hierarchically semi-separable matrices," *Journal of Scientific Computing*, vol. 57, no. 3, pp. 477-501, 2013.

[4] J. Xia, S. Chandrasekaran, M. Gu, et al., "Fast algorithms for hierarchically semiseparable matrices," *Numerical Linear Algebra with Applications*, vol. 17, no. 6, pp. 953-976, 2010.

[5] W. Hackbusch, "A sparse matrix arithmetic based on H-matrix," *Computing*, vol. 62, no. 2, pp. 89-108, 1999.

[6] W. Hackbusch, "Data-sparse approximation by adaptive H2-matrices," *Computing*, vol. 69, pp. 1-35, 2002.

[7] S. Chandrasekaran, M. Gu, and T. Pals, "A fast ULV decomposition solver for hierarchically semi-separable representations," *Siam Journal on Matrix Analysis & Applications*, vol. 28, no. 3, pp. 603-622, 2006.

[8] S. Chandrasekaran, M. Gu, and T. Pals, "Fast and stable algorithms for hierarchically semi-separable representations," *Technical Report, University of California: Berkeley*, CA, 2004.

[9] S. Chandrasekaran, P. Dewilde, M. Gu, T. Pals, X. Sun, AJ van der Veen, and D. White, "Some fast algorithms for sequentially semiseparable representation," *SIAM Journal on Matrix Analysis and Applications*, vol. 27, pp. 341-364, 2005.

[10] W. Chai and D. Jiao, "Fast-matrix-based direct integral equation solver with reduced computational cost for large-scale interconnect extraction," *IEEE Transactions on Components Packaging & Manufacturing Technology*, vol. 3, no. 2, pp. 289-298, 2013.

[11] M. Ma and D. Jiao, "Exact-arithmetic HSS-inversion algorithm for fast direct solution of electrically large volume integral equations," *International Conference on Electromagnetics in Advanced Applications, IEEE*, pp. 597-599, 2016.

[12] A. Manic, A. Smull, F. H. Rouet, et al., "Efficient scalable parallel higher order direct MoM-SIE method with hierarchically semiseparable structures for 3D scattering," *IEEE Transactions on Antennas & Propagation*, pp. 99:1-1, 2017.

[13] J. Xia and M. Gu, "Robust approximate Choleksy factorization of rank-structured symmetric positive definite matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, pp. 2899-2920, 2010.

[14] S. Li, M. Gu, L. Cheng, X. Chi, and M. Sun, "An accelerated divide-and-conquer algorithm for the bidiagonal SVD problem," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, pp. 1038-1057, 2014.

[15] A. Napov and X. S. Li, "An algebraic multifrontal preconditioner that exploits the low-rank property," *Numerical Linear Algebra with Applications*, vo. 23, no. 1, pp. 61-82, 2016.

[16] F. H. Rouet and X. S. Li, et al., "A distributed-memory package for dense hierarchically semi-separable matrix computations using Randomization," vol. 42, no. 4, p. 27, 2015.

[17] P. G. Martinsson, "A fast randomized algorithm for computing a hierarchically semiseparable representation of a matrix," *Siam Journal on Matrix Analysis & Applications*, vol. 32, no. 4, pp. 1251-1274, 2011.

[18] P. Rocca and A. F. Morabito, "Optimal synthesis of reconfigurable planar arrays with simplified architectures for monopulse radar applications," *IEEE Transactions on Antennas & Propagation*, vol. 63, no. 3, pp. 1048-1058, 2015.