Multi-level Power Series Solution for Large Surface and Volume Electric Field Integral Equation

Y. K. Negi[1], N. Balakrishnan[1], and S. M. Rao[2]

[1]Supercomputer Education Research Centre
Indian Institute of Science, Bangalore, Karnataka, 560012 India
yknegi@gmail.com, balki@iisc.ac.in

[2]Naval Research Laboratory
Washington DC, 20375, USA
sadasiva.rao@nrl.navy.mil

Abstract – In this paper, we propose a new multi-level power series solution method for solving a large surface and volume electric field integral equation-based H-Matrix. The proposed solution method converges in a fixed number of iterations and is solved at each level of the H-Matrix computation. The solution method avoids the computation of a full matrix, as it can be solved independently at each level, starting from the leaf level. Solution at each level can be used as the final solution, thus saving the matrix computation time for full H-Matrix. The paper shows that the leaf level matrix computation and solution with power series gives as accurate results as the full H-Matrix iterative solver method. The method results in considerable savings time and memory savings compared to the H-Matrix iterative solver. Further, the proposed method retains the $O(NlogN)$ solution complexity.

Index Terms – H-Matrix, Method of Moments (MoM), power series solution, surface electric field integral equation, volume electric field integral equation.

I. INTRODUCTION

With the use of ever increasing higher frequencies for various defence and civilian applications in the current world, the electrical size of electromagnetic scattering/radiation problem has grown drastically[1, 2]. Solving the electrically large problems numerically to obtain fast and accurate results is the biggest challenge in the Computational Electromagnetics (CEM) community. Also, with the increase in computing power and memory, the need for large-scale solution algorithms has grown even more. Out of the various numerical methods in CEM, the most popular methods are: a) the Finite Difference Time Domain (FDTD) [3] method in the time domain and b) the Method of Moments (MoM) [4] and Finite Element Method (FEM) [5] in the frequency domain. Traditionally, the frequency domain methods have been more popular than the time domain methods as most of the early experimental results were available in the frequency domain and validating the computational results was convenient and easy. Out of the various frequency domain methods, MoM based methods are highly accurate and flexible for modeling irregular structures, the MoM matrix can be computed with the Surface Electric Field Integral Equation (S-EFIE) for solving Perfect Electrical Conductor (PEC) problems with surface mesh, and the Volume Electric Field Integral Equation (V-EFIE) [6] for solving inhomogeneous dielectric problems with volume mesh. Further, the MoM leads to a smaller number of unknowns compared to FEM and is free from grid dispersion error. However, the MoM matrix is a full matrix compared to a sparse matrix for the FEM method. Hence, the solution to large size problems with MoM in electromagnetics requires high matrix memory and computation time due to the dense matrix. Note that MoM dense matrix computation, matrix vector product and storage cost scales to $O(N^2)$ for N number of unknowns. Solving the dense matrix with an iterative solver leads to $N_{itr}O(N^2)$ calculations for N_{itr} iteration with $O(N^2)$ for matrix-vector multiplication cost. With a direct solver, the complexity grows as $O(N^3)$. Various fast solver algorithms like Multi-Level Fast Multipole Algorithm (MLFMA) [6], Adaptive Integral Method (AIM) [7], FFT [8], IE-QR [9], and Hierarchical Matrix (H-Matrix) [10–12] have been proposed to overcome the MoM limitations of high memory and computation cost. Fast solver reduces the matrix memory, matrix fill time, and matrix-vector product time to $O(NlogN)$. The reduced matrix-vector product time improves the solution time to $N_{itr}O(NlogN)$ for N_{itr} iterations with various iterative solution methods like Bi-Conjugate Gradient (BiCG) or Generalized Minimum Residual (GMRES).

Fast solvers are built on the compressibility property of the far-field interaction matrices. The compression of the far-field matrices can be done using analytical matrix compression methods like MLFMA or AIM, and also with numerical matrix compression methods like H-Matrix. Compared to analytical compression methods, numerical compression methods are easy to implement and are kernel independent. All the fast solvers depend on the iteration count of the iterative solution methods. The convergence of the iterations depends on the condition number of the computed MoM matrix, and further, for a large number of unknowns, the convergence iteration count also increases. The high iteration count can be mitigated by using various preconditions like ILUT, Null-Field, and Schur's complement method based preconditioners [13–15]. The matrix preconditioner improves the condition number of the matrices and reduces the iteration count of the overall matrix solution. Despite the improvement in solution time, the use of preconditioners comes with the overhead of preconditioner computation time and extra preconditioner solution time for each iteration. Also, for the solving of a large number of unknowns, the iteration count may still be high.

Recently there has been a trend in the CEM community for the development of an iteration-free fast solver method for solving problems with a large number of unknowns. Various fast direct solvers [16, 17] have been proposed to overcome the iteration dependency of the solution process. These direct solvers are based on LU decomposition and compression methods. The methods are complex to implement and give quadratic scaling for complex real-world problems.

In this work, we propose a Multi-Level (ML) fast matrix solution method based on the power series [18, 19]. The proposed method exploits the property of ML matrix compression of the H-Matrix. The matrix is solved for each level using the matrix computation of the leaf level only, and the matrix solution can be terminated at the desired level as per the required accuracy. Our experimental results show that we get good accuracy even for the lowest level solution. The method relies on matrix-vector multiplication at each level and using the solution of the lowest level saves matrix computation time and memory requirement for the overall matrix solution.

The rest of the paper is organized as follows. Section II gives a summary of MoM computation for S-EFIE and V-EFIE, section III covers H-Matrix computation for S-EFIE and V-EFIE. The derivation of the proposed ML power series solver is given in section IV. The numerical results of the proposed method, and conclusion are discussed in sections V, and VI.

II. METHOD OF MOMENTS

MoM is a popular and efficient integral equation based method for solving various electromagnetic radiation/scattering problems. MoM can be computed using Electric Field Integral Equation (EFIE) for both surface and volume modeling. Surface modeling can be done using Rao Wilton Glisson (RWG) [20] triangle basis function, whereas volume modeling can be done using Schaubert Wilton Glisson (SWG) [21] tetrahedral basis function. In the case of dielectric modeling compared to S-EFIE, V-EFIE is an integral equation of the second kind and is more well-conditioned and stable. V-EFIE can model inhomogeneous bodies more efficiently than surface EFIE. In this work, we use RWG basis function for PEC surface S-EFIE modeling and SWG basis function for volume V-EFIE modeling. The surface/volume EFIE governing equation for the conductor/dielectric scattering body illuminated with the incident plane wave is given as the total electric field ($\boldsymbol{E}^{total}$) from a scattering surface/volume and is the sum of incident electric field ($\boldsymbol{E}^{inc}$) and scattered electric fields ($\boldsymbol{E}^{scatt}$).

$$\boldsymbol{E}^{total} = \boldsymbol{E}^{inc} + \boldsymbol{E}^{scatt}. \tag{1}$$

The scatted electric field is due to the surface current in PEC surface or volume polarization current in the dielectric media and is given as:

$$\boldsymbol{E}^{scatt} = -j\omega\boldsymbol{A}(\boldsymbol{r}) - \nabla\phi(\boldsymbol{r}). \tag{2}$$

In the above equation $\boldsymbol{A}(\boldsymbol{r})$ is the magnetic vector potential and describes radiation of current, $\phi(\boldsymbol{r})$ is electric potential and describes associate bound charge. Applying the boundary condition for PEC structure the S-EFIE can be written as:

$$\boldsymbol{E}^{inc} = j\omega\boldsymbol{A}(\boldsymbol{r}) + \nabla\phi(\boldsymbol{r}). \tag{3}$$

Similarly, the V-EFIE can be written for a dielectric inhomogeneous body as:

$$\boldsymbol{E}^{inc} = \frac{\boldsymbol{D}(\boldsymbol{r})}{\varepsilon(\boldsymbol{r})} + j\omega\boldsymbol{A}(\boldsymbol{r}) + \nabla\phi(\boldsymbol{r}). \tag{4}$$

In the above, equation $\boldsymbol{D}(\boldsymbol{r})$ is the electric flux density and $\varepsilon(\boldsymbol{r})$ is the dielectric constant of the scattering volume media. The surface current in equation (3) for PEC structure is expanded with RWG function, and similarly in equation (4) for dielectric volume structure polarization current and charge is modeled with SWG basis function. Performing Galarkin testing over each term with integrating over the surface/volume, the final system of equation boils down to the linear system of the equation as below:

$$[\boldsymbol{Z}]\boldsymbol{x} = \boldsymbol{b}. \tag{5}$$

In the above equation, $\boldsymbol{Z}$ is a dense MoM matrix, $\boldsymbol{b}$ is a known incident plane wave, and $\boldsymbol{x}$ is an unknown coefficient to be computed. The dense matrix leads to high cost matrix computation and memory requirement as well as solution time complexity. In the next section, we discuss the implementation of the H-Matrix for the mitigation of high cost of the conventional MoM matrix

III. H-MATRIX

The high cost of MoM limits its application to a few λ problem sizes. This limitation of MoM can be overcome by incorporating fast solvers. Most of the fast solvers work on the principle of compressibility of the far-field matrices. For the implementation of a fast solver, the mesh of geometry is divided into blocks using an oct-tree or binary-tree division process and terminated at the desired level with a limiting edge or face count in each block. The non-far-field interaction blocks at the lowest level are considered near-field blocks and are in the dense matrix form. The compression of the far-field block matrix at each level can be done analytically or numerically. The system of equations in equation (5) can now be written as the sum of near-field and far-field matrix form as:

$$[\boldsymbol{Z}_N + \boldsymbol{Z}_F]\boldsymbol{x} = \boldsymbol{b}. \tag{6}$$

In the above equation $\boldsymbol{Z}_N$ is a near-field block matrix and $\boldsymbol{Z}_F$ is far-field compressed block matrices for the MoM fast solver matrix. Numerical compression of far-field matrices is easy to implement and is kernel-independent. A few of the popular fast solvers using numerical compression methods are IE-QR, H-Matrix. In this work, we have implemented H-Matrix for ML matrix compression. For the ML compression computation, the mesh is divided into ML binary tree division-based subgroups. H-Matrix works on the computation of a far-field matrix for the interaction blocks satisfying the admissibility condition given in equation (7). The admissibility condition states that η times the distance between the observation cluster (Ω_t) and source cluster (Ω_s) should be greater or equal to the minimum diameter of the observation cluster or source cluster for far-field computation, where η is the admissibility control parameter, and its value is taken as 1.0.

$$\eta \ dist(\Omega_t, \Omega_s) \geq min(diam(\Omega_t), diam(\Omega_s)). \tag{7}$$

The far-field matrix block compression is done in such a way that its parent interaction matrix should not be computed at the top level. Matrix compression at each level is carried out using Adaptive Cross Approximation (ACA) [22] [23] method. The method exploits the rank deficiency property of the far-field matrix blocks. The low-rank sub-block of the far-field $\boldsymbol{Z}_{sub}$ with m rows and
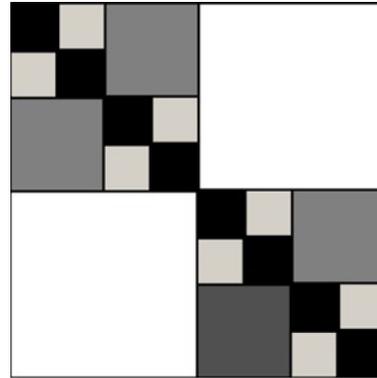

Fig. 1. Compressed far-field and dense near-field matrix blocks layout.

n columns is decomposed into approximate $\boldsymbol{U}_{(m \times k)}$ and $\boldsymbol{V}_{(k \times n)}$ matrices where k is the numerical rank of the low-rank sub-block far-field matrix such that $k << min(m,n)$. In this work, for memory savings, we only compute half of the H-Matrix [12] by making the computation process symmetric, and to maintain the accuracy of the H-Matrix, we use re-compressed ACA [24] for far-field block compression. The solution of the iterative solver is iteration count dependent, and further, the convergence iteration count depends on the condition number of the matrix. Also, as the number of unknowns increases, the iterating count for the convergence increases. In the next section, we discuss our proposed method, which is an iteration count and far-field level block independent solution process.

IV. MULTI-LEVEL POWER SERIES SOLUTION

The full H-Matrix is a combination of near-field and far-field block matrices. The far-field compressed block matrices are computed for various levels, and in equation (6), the far-field matrix ($\boldsymbol{Z}_F$) can be further decomposed into the different matrix levels as below:

$$[\boldsymbol{Z}_F] = [\boldsymbol{Z}_{F1}] + [\boldsymbol{Z}_{F2}] + [\boldsymbol{Z}_{F3}]. \tag{8}$$

In the above equation far-field matrix $\boldsymbol{Z}_{F1}$ is for level 1, $\boldsymbol{Z}_{F2}$ is for level 2 and, $\boldsymbol{Z}_{F3}$ is for level 3. Level 3 forms the leaf level of the binary tree and level 1 as the top level of the tree. Figure 1 shows the H-Matrix layout for a two-dimension strip. In Fig. 1, light gray boxes represent $\boldsymbol{Z}_{F1}$ far-field matrix at level 1, dark gray boxes as $\boldsymbol{Z}_{F2}$ is for level 2 and large white boxes as $\boldsymbol{Z}_{F3}$ for level 3, the black boxes are the near-field dense matrices. For illustrative purposes, the near-field matrix is a diagonal block form for a two-dimension strip. The real-world problems are three-dimension in structure, giving a non-diagonal block near-field matrix. To implement our ML power series solution method, we must diagonalize the

near-field block matrix. The near-field matrix in equation (6) is diagonalized using diagonal scaling coefficient $[\boldsymbol{\alpha}]$, as computed in [15] such that the scaled diagonal block near-field matrix can be given as:

$$[\tilde{\boldsymbol{Z}}_N] = [\boldsymbol{\alpha}][\boldsymbol{Z}_N]. \qquad (9)$$

Expanding equation (8) and scaling it with the scaling coefficients $[\boldsymbol{\alpha}]$ gives:

$$[\boldsymbol{\alpha}][\boldsymbol{Z}_N + \boldsymbol{Z}_{F1} + \boldsymbol{Z}_{F2} + \boldsymbol{Z}_{F3}]\boldsymbol{x} = [\boldsymbol{\alpha}]\boldsymbol{b}, \qquad (10)$$

$$[\tilde{\boldsymbol{Z}}_N]\boldsymbol{x} + [\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}]\boldsymbol{x} + [\boldsymbol{\alpha}][\boldsymbol{Z}_{F2}]\boldsymbol{x} + [\boldsymbol{\alpha}][\boldsymbol{Z}_{F3}]\boldsymbol{x} = \tilde{\boldsymbol{b}}. \quad (11)$$

In the above equation $\tilde{\boldsymbol{b}}$ is a $[\boldsymbol{\alpha}]$ scaled vector $\boldsymbol{b}$ and can be further simplified as:

$$\begin{aligned}\boldsymbol{x} + [\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}]\boldsymbol{x} + [\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F2}]\boldsymbol{x} \\ + [\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F3}]\boldsymbol{x} = [\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}.\end{aligned} \qquad (12)$$

Let $[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}] = [\boldsymbol{U}_1]$, $[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F2}] = [\boldsymbol{U}_2]$ and $[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F3}] = [\boldsymbol{U}_3]$ equation (12) can further be simplified as:

$$\boldsymbol{x} + [\boldsymbol{U}_1]\boldsymbol{x} + [\boldsymbol{U}_2]\boldsymbol{x} + [\boldsymbol{U}_3]\boldsymbol{x} = [\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}, \qquad (13)$$

$$[\boldsymbol{I} + \boldsymbol{U}_1]\boldsymbol{x} + [\boldsymbol{U}_2]\boldsymbol{x} + [\boldsymbol{U}_3]\boldsymbol{x} = [\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}, \qquad (14)$$

$$\begin{aligned}\boldsymbol{x} + [\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_2]\boldsymbol{x} + [\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_3]\boldsymbol{x} \\ = [\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}.\end{aligned} \qquad (15)$$

Let $[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_2] = [\boldsymbol{V}_2]$ and $[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_3] = [\boldsymbol{V}_3]$ equation (15) can further be simplified as:

$$\boldsymbol{x} + [\boldsymbol{V}_2]\boldsymbol{x} + [\boldsymbol{V}_3]\boldsymbol{x} = [\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}, \qquad (16)$$

$$\boldsymbol{x} + [\boldsymbol{I} + \boldsymbol{V}_2]^{-1}[\boldsymbol{V}_3]\boldsymbol{x} = [\boldsymbol{I} + \boldsymbol{V}_2]^{-1}[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}. \qquad (17)$$

Let $[\boldsymbol{I} + \boldsymbol{V}_2]^{-1}[\boldsymbol{V}_3] = [\boldsymbol{W}_3]$ and equation (17) can be written as:

$$\boldsymbol{x} + [\boldsymbol{W}_3]\boldsymbol{x} = [\boldsymbol{I} + \boldsymbol{V}_2]^{-1}[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}, \qquad (18)$$

$$\boldsymbol{x} = [\boldsymbol{I} + \boldsymbol{W}_3]^{-1}[\boldsymbol{I} + \boldsymbol{V}_2]^{-1}[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}\tilde{\boldsymbol{b}}. \qquad (19)$$

In the above equations $[\boldsymbol{I} + \boldsymbol{W}_3]^{-1}, [\boldsymbol{I} + \boldsymbol{V}_2]^{-1}$ and $[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}$ can be solved independently at each level using a power series solution method with the expansion as below:

$$[\boldsymbol{I} + \boldsymbol{U}_1]^{-1} = [\boldsymbol{I} + [\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}]]^{-1}, \qquad (20)$$

$$[\boldsymbol{I} + \boldsymbol{V}_2]^{-1} = [\boldsymbol{I} + [\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_2]]^{-1}$$

$$= [\boldsymbol{I} + [\boldsymbol{I} + [\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}]]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F2}]]^{-1}, \qquad (21)$$

$$[\boldsymbol{I} + \boldsymbol{W}_3]^{-1} = [\boldsymbol{I} + [\boldsymbol{I} + \boldsymbol{V}_2]^{-1}[\boldsymbol{V}_3]]^{-1}$$

$$= [\boldsymbol{I} + [\boldsymbol{I} + [\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_2]]^{-1}[\boldsymbol{I} + \boldsymbol{U}_1]^{-1}[\boldsymbol{U}_3]]^{-1}$$

$$= [\boldsymbol{I} + [\boldsymbol{I} + [\boldsymbol{I} + [\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}]]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F2}]]^{-1}$$

$$[\boldsymbol{I} + [[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F1}]]^{-1}[\tilde{\boldsymbol{Z}}_N]^{-1}[\boldsymbol{\alpha}][\boldsymbol{Z}_{F3}]]^{-1}. \qquad (22)$$

From equations (20), (21), and (22), it can be observed that the solution of these equations is dependent on that level and the lower levels of the binary tree block interaction matrix. At each level, the inverse of the matrix system equation can be efficiently computed by using a fast power series solution [18]. The fast power series iterative solution converges in two fixed iterations. The solution process only depends on the matrix-vector product of the H-Matrix, thus retaining the complexity of $O(NlogN)$[18]. The ML solution can be computed at the desired level per the required accuracy. Our results show that the solution at the leaf level gives an accurate result leading to time and memory savings.

V. NUMERICAL RESULTS

In this section, we show the accuracy and efficiency of the proposed method. The simulations are carried out on 128 GB memory and an Intel (Xeon E5-2670) processor system for the double-precision data type. The H-Matrix computation is done with the ACA matrix compression error tolerance of 1e-3 [22] and solved with GMRES iterative solver with convergence tolerance of 1e-6 [12]. For a compressed or dense matrix $[\boldsymbol{Z}]$ if we want to expand $[1 + \boldsymbol{Z}]^{-1}$ in power series, the necessary and sufficient condition for convergence is $|\boldsymbol{Z}| < 1$ and we choose 0.1 for our simulations [25]. The conductor and dielectric geometry with dielectric constant ε_r is meshed with an element size less than $\lambda/10$ and $\lambda/(10\sqrt{\varepsilon_r})$ respectively. To show the accuracy of the proposed method, the RCS results are compared with full H-Matrix iterative solver [12]. In the further subsections, we demonstrate the far-field memory and computation time savings in along with in solution time saving with our proposed ML power series solution with different examples.

A. PEC square plate

To show the accuracy and efficiency on a PEC object in this subsection, we consider a square plate of size 15.0 λ along x and y axis meshed with 67,200 unknown edges. The square plate mesh is divided with binary tree division till level 6. The PEC S-EFIE H-Matrix is solved with ML the power series solution method and H-Matrix

iterative solver. The ML power series converges in 2 iterations, and the iterative solver solution converges in 686. Only the far-field matrix at leaf level 6 is computed for the ML power series solution, ignoring far-field computation from levels 1 to 5 of the binary tree.
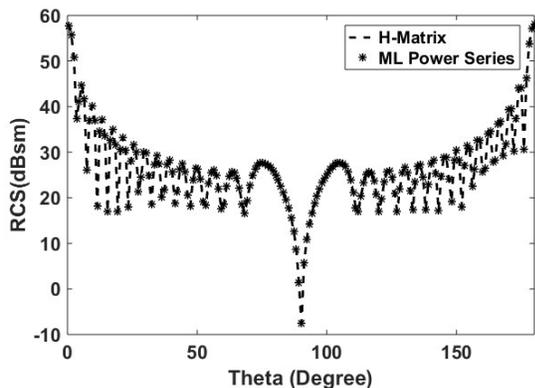

Fig. 2. Bi-static RCS of the PEC square plate with VV polarized plane wave incident at $\theta = 0°$, $\phi = 0°$, and observation angles $\theta = 0°$ to $180°$, $\phi = 0°$.

Figure 2 shows the Bi-static RCS of a PEC square plate, and from the figure it can be observed that the solution with the ML power series solver matches with the H-Matrix iterative solver. Table 1 shows the savings in memory, computation, and solution time of the ML power series solution method as compared with a conventional H-Matrix-based iterative solver.

Table 1: Matrix memory, fill and solution time for a PEC square plate

	Memory (GB)	Matrix Fill Time (H)	Solution Time (sec)
H-Matrix	5.04	1.24	500.85
ML Power Series	4.71	1.08	3.95

B. Dielectric slab

To show the accuracy and efficiency for a considerable size dielectric problem in this subsection, we consider a dielectric slab elongated along the y-axis with a height of $10.0\,\lambda$ length, $1.0\,\lambda$ width, and $0.1\,\lambda$ thickness and dielectric constant ($\varepsilon_r = 2.0$) meshed with 120,080 tetrahedral faces. The ML power series converges in 2 iterations, and the regular H-Matrix iterative solver converges in 33 iterations.

The dielectric slab mesh is divided with binary tree division till level 10. Only the far-field matrix at leaf level 10 is computed for the ML power series solution. The accuracy of the method for a Bi-static RCS is shown
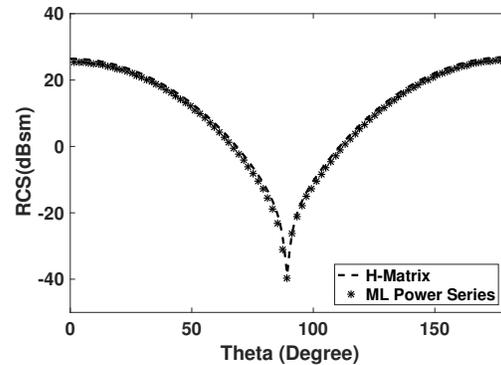

Fig. 3. Bi-static RCS of the dielectric slab with VV polarized plane wave incident at $\theta = 0°$, $\phi = 0°$, and observation angles $\theta = 0°$ to $180°$, $\phi = 0°$.

Table 2: Matrix memory, fill and solution time for a dielectric slab

	Memory (GB)	Matrix Fill Time (H)	Solution Time (sec)
H-Matrix	2.09	6.12	24.52
ML Power Series	0.50	1.46	7.50

in Fig. 3. Table 2 shows the significant matrix memory, matrix fill and solution time savings of the ML power series solution compared to the conventional H-Matrix-based iterative solver.
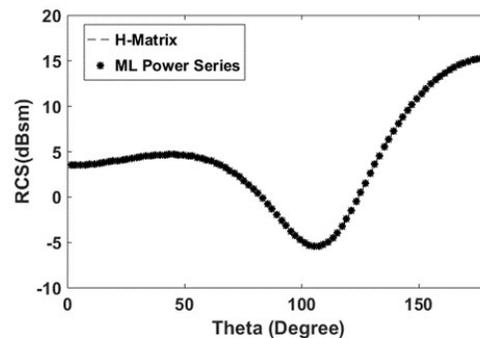

Fig. 4. Bi-static RCS of a dielectric hollow cylinder with VV polarized plane wave incident at $\theta = 0°$, $\phi = 0°$, and observation angles $\theta = 0°$ to $180°$, $\phi = 0°$.

C. Dielectric hollow cylinder

In this subsection, we consider a dielectric hollow cylinder elongated along the y-axis with a size of 6.0λ length, 0.4λ outer radii, and 0.05λ thickness with a dielectric constant ($\varepsilon_r = 2.0$), meshed with 158,830 tetrahedral faces. The ML power series converges in 2 iterations, and the H-Matrix iterative solver converges in 24 iterations.

The hollow cylinder mesh is partitioned with a binary tree division till level 8, and for the ML power series solution only the far-field matrix at leaf level 8 is computed. Figure 4 shows the close match in the bi-static RCS computed using the ML power series method and that with regular H-Matrix iterative solver. Table 3 shows the memory and time saving of the ML power series solution compared to the conventional H-Matrix iterative solver.

Table 3: Matrix memory, fill and solution time for a dielectric hollow cylinder

	Memory (GB)	Matrix Fill Time (H)	Solution Time (sec)
H-Matrix	3 .38	10.00	54.52
ML Power Series	0.44	1.26	16.16

VI. CONCLUSION

It can be observed from the illustrative examples in the previous sections that our proposed ML power series solution method gives considerable matrix memory, fill and solve time saving for significant size problems. The solution method is as accurate as the H-Matrix iterative solver. The savings may not be substantial for small-size mesh structures. Still, the method will give significant savings for large-size problems taken up for illustration and for complex and sizeable electrical problems like antenna arrays and complex composite structures. Also, the technique is entirely algebraic in nature and can apply to fast analytical solver-based methods like AIM and MLFMA. The matrix block in each level can be computed independently, and the solution of the method only depends on the matrix-vector product of the system matrix. Hence, the proposed method is amenable to efficient parallelization.

REFERENCES

[1] V. P. Padhy, Y. K. Negi, and N. Balakrishnan, "RCS enhancement due to Bragg scattering," *2012 International Conference on Mathematical Methods in Electromagnetic Theory, Kharkiv Ukraine*, pp. 443-446, IEEE, 28-30 Aug. 2012.

[2] Y. K. Negi, V. P. Padhy and N. Balakrishnan, "RCS enhancement of concealed/hidden objects at Terahertz (THz)," *27th Annual Review of Progress in Applied Computational Electromagnetics Williamsburg, Virginia, USA*, pp. 347-350, 27-31 March 2011.

[3] A. Taflove, *The Finite Difference Time Domain Method.*, Boston, Artech House, 1995.

[4] R. Harrington, *Field Computation by Moment Methods*, Malabar, Florida, RE Krieger Publishing Company, 1982.

[5] J.-M. Jin, *The Finite Element Method in Electromagnetics*, Hoboken New Jersey, John Wiley & Sons, 2015.

[6] W. C. Chew, E. Michielssen, J. Song, and J.-M. Jin, *Fast and Efficient Algorithms in Computational Electromagnetics*, Norwood, MA, USA: Artech House, Inc., 2001.

[7] E. Bleszynski, M. Bleszynski, and T. Jaroszewicz, "AIM: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems," *Radio Science*, vol. 31, no. 5, pp. 1225-1251, Sept.-Oct. 1996.

[8] J. R. Phillips and J. K. White, "A precorrected-FFT method for electrostatic analysis of complicated 3-D structures," *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, vol. 16, no. 10, pp. 1059-1072, Oct. 1997.

[9] S. Kapur and D. E. Long, "N-body problems: IES 3: Efficient electrostatic and electromagnetic simulation," *IEEE Computational Science and Engineering*, vol. 5, no. 4, pp. 60-67, Oct.-Dec. 1998.

[10] W. Hackbusch, "A sparse matrix arithmetic based on-matrices. Part I: Introduction to-matrices," *Computing*, vol. 62, no. 2, pp. 89-108, Apr. 1999.

[11] W. Hackbusch and B. N. Khoromskij, "A sparse H-matrix arithmetic, part II: Application to multi-dimensional problems," *Computing*, vol. 64, no. 1, pp. 21-47, Feb. 2000.

[12] Y. K. Negi, "Memory reduced half hierarchal matrix (H-matrix) for electrodynamic electric field integral equation," *Progress in Electromagnetics Research Letters*, vol. 96, pp. 91-96, Feb. 2021.

[13] Y. Saad, "ILUT: A dual threshold incomplete LU factorization," *Numerical Linear Algebra with Applications*, vol. 1, no. 4, pp. 387-402, Jul.-Aug. 1994.

[14] Y. K. Negi, N. Balakrishnan, S. M. Rao, and D. Gope, "Null-field preconditioner with selected far-field contribution for 3-D full-wave EFIE," *IEEE Transactions on Antennas and Propagation*, vol. 64, no. 11, pp. 4923-4928, Nov. 2016.

[15] Y. K. Negi, N. Balakrishnan, and S. M. Rao, "Symmetric near-field Schur's complement preconditioner for hierarchal electric field integral equation solver," *IET Microwaves, Antennas & Propagation*, vol. 14, no. 14, pp. 1846-1856, Nov. 2020.

[16] L. Greengard, D. Gueyffier, P.-G. Martinsson, and V. Rokhlin, "Fast direct solvers for integral equations in complex three-dimensional domains," *Acta Numerica*, vol. 18, pp. 243-275, May 2009.

[17] J. Shaeffer, "Direct solve of electrically large integral equations for problem sizes to 1 M unknowns," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2306-2313, Aug. 2008.

[18] Y. K. Negi, N. Balakrishnan, and S. M. Rao, "Fast power series solution of large 3-D electrodynamic integral equation for PEC scatterers," *Applied Computational Electromagnetics Society (ACES) Journal*, pp. 1301-1311, Nov. 2021.

[19] S. M. Rao, "A new multi-level power series solution algorithm to solve electrically large electromagnetic scattering problems applicable to conducting bodies," *2021 International Conference on Electromagnetics in Advanced Applications (ICEAA)*, Honolulu, HI, USA, pp. 152-152, 09-13 Aug. 2021.

[20] S. Rao, D. Wilton, and A. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 3, pp. 409-418, May 1982.

[21] D. Schaubert, D. Wilton, and A. Glisson, "A tetrahedral modeling method for electromagnetic scattering by arbitrarily shaped inhomogeneous dielectric bodies," *IEEE Transactions on Antennas and Propagation*, vol. 32, no. 1, pp. 77-85, Jan. 1984.

[22] M. Bebendorf, "Approximation of boundary element matrices," *Numerische Mathematik*, vol. 86, pp. 565-589, Oct. 2000.

[23] S. Kurz, O. Rain, and S. Rjasanow, "The adaptive cross-approximation technique for the 3D boundary-element method," *IEEE Transactions on Magnetics*, vol. 38, no. 2, pp. 421-424, Aug. 2002.

[24] Y. K. Negi, V. P. Padhy, and N. Balakrishnan, "Re-compressed H-matrices for fast electric field integral equation," *2020 IEEE International Conference on Computational Electromagnetics (ICCEM)*, pp. 176-177, Singapore, 2020.

[25] S. M. Rao and M. S. Kluskens, "A new power series solution approach to solving electrically large complex electromagnetic scattering problems," *Applied Computational Electromagnetics Society (ACES) Journal*, pp. 1009-1019, Aug. 2021.

Yoginder Kumar Negi obtained the B.Tech degree in Electronics and Communic-ation Engineering from Guru Gobind Singh Indraprastha University, New Delhi, India, in 2005, M.Tech degree in Microwave Electronics from Delhi University, New Delhi, India, in 2007 and the PhD degree in engineering from Indian Institute of Science (IISc), Bangalore, India, in 2018.

Dr Negi joined Supercomputer Education Research Center (SERC), IISc Bangalore in 2008 as a Scientific Officer. He is currently working as a Senior Scientific Officer in SERC IISc Bangalore. His current research interests include numerical electromagnetics, fast techniques for electromagnetic application, bio-electromagnetics, high-performance computing, and antenna design and analysis.

B. Narayanaswamy received the B.E. degree (Hons.) in Electronics and Communi-cation from the University of Madras, Chennai, India, in 1972, and the Ph.D. degree from the Indian Institute of Science, Bengaluru, India, in 1979.

He joined the Department of Aerospace Engineering, Indian Institute of Science, as an Assistant Professor, in 1981, where he became a Full Professor in 1991, served as the Associate Director, from 2005 to 2014, and is currently an INSA Senior Scientist at the Supercomputer Education and Research Centre. He has authored over 200 publications in the international journals and international conferences. His current research interests include numerical electromagnetics, high-performance computing and networks, polarimetric radars and aerospace electronic systems, information security, and digital library.

Dr. Narayanaswamy is a fellow of the World Academy of Sciences (TWAS), the National Academy of Science, the Indian Academy of Sciences, the Indian National Academy of Engineering, the National Academy of Sciences, and the Institution of Electronics and Telecommunication Engineers.

Sadasiva M. Rao obtained his Bachelors, Masters, and Doctoral degrees in electrical engineering from Osmania University, Hyderabad, India, Indian Institute of Science, Bangalore, India, and University of Mississippi, USA, in 1974, 1976, and 1980, respectively. He is well known in the electromagnetic engineering community and included in the Thomson Scientifics' *Highly Cited Researchers List*.

Dr. Rao has been teaching electromagnetic theory, communication systems, electrical circuits, and other related courses at the undergraduate and graduate level for the past 30 years at various institutions. At present, he is working at Naval Research Laboratories, USA. He published/presented over 200 papers in various journals/conferences. He is an elected Fellow of IEEE.