Fast Simulation of Microwave Devices via a Data-Sparse and Explicit Finite-Element Time-Domain Method

T. Wan [1,2], L. Du [3], and J. Zhu [1]

[1] Department of Communication Engineering
Nanjing University of Posts and Telecommunications, Nanjing, 210003, China
want@njupt.edu.cn

[2] State Key Laboratory of Millimeter Waves
Southeast University, Nanjing, 210096, China

[3] Institute of Electrostatic and Electromagnetic Protection
Ordnance Engineering College, Shijiazhuang, 050003, China
billwanting@163.com

Abstract ─ An unconditionally stable and explicit finite-element time-domain (FETD) method is presented for the fast simulation of microwave devices. The Crank-Nicolson (CN) scheme is implemented leading to an unconditionally stable mixed FETD method. A data-sparse approximate inverse algorithm is introduced to provide a data-sparse way to approximate the inverse of FETD system matrix which is dense originally. This approximate inverse matrix can be constructed and stored with almost linear complexity, and then the FETD method can be computed explicitly at each time step without solving a sparse linear system. An efficient recompression technique is introduced to further accelerate the explicit solution at each time step. Some microwave devices are simulated to demonstrate the efficiency and accuracy of the proposed method.

Index Terms ─ Approximate inverse, electromagnetic simulation, finite-element time-domain (FETD), microwave devices.

I. INTRODUCTION

The finite-element time-domain (FETD) method has been widely used for electromagnetic analysis due to its ability to deal with complex geometries and broadband characterizations [1-9]. However, the FETD generally requires solving a sparse linear system at each time step. This greatly reduces the solution efficiency, and has been a major bottleneck for the development of the FETD. Hence, it is highly desirable to develop explicit schemes. Some research has been studied to obtain an explicit FETD method. One way is to construct diagonal or block diagonal mass matrix to avoid solving a sparse linear system, such as mass lumping [1], orthogonal vector basis function [2,3] and spectral-element time-domain (SETD) method [4]. Another way to achieve an explicit scheme is to directly invert the mass matrix. However, the computational costs can be very high and the inverse matrix is generally full. Recently, a thresholding method has been proposed to approximate the inverse mass matrix by a sparse matrix to obtain an explicit FETD [5,6]. Besides, a recursive sparsification method has been applied to approximate the inverse matrix in the alternating-direction implicit FETD method [7].

In this article, a data-sparse approximate inverse (DSAI) algorithm is developed to construct an explicit FETD method. The Crank-Nicolson (CN) scheme is employed to generate an unconditionally stable mixed FETD method [8,9]. The DSAI algorithm provides a data-sparse way to approximate the inverse of the CN-FETD system matrix which is dense originally. The inverse matrix is not sparse but data-sparse in the sense that its certain sub-blocks can be described by a product of two low-rank matrices [10,11]. Based on the hierarchical tree and the formatted hierarchical matrix arithmetic, the DSAI algorithm can reduce the computational complexity and storage requirement of matrix inversion to be almost linear [12,13]. The resulting data-sparse inverse matrix has been highly compressed but still contains redundant data [12,14]. A recompression technique is introduced to further compress the inverse matrix to accelerate the explicit update at each time step. Some numerical examples are presented and discussed.

II. DATA-SPARSE AND EXPLICIT FETD

A. Mixed E-B FETD method

According to the Maxwell's equations:

Submitted On: September 7, 2015
Accepted On: March 18, 2016

$$\frac{1}{\mu}\nabla\times\vec{B}=\varepsilon\frac{\partial\vec{E}}{\partial t}, \quad (1)$$

$$\nabla\times\vec{E}=-\frac{\partial\vec{B}}{\partial t}, \quad (2)$$

the examined 3-D domain is discretized by tetrahedral elements. The electric field $\vec{E}$ and the magnetic flux $\vec{B}$ are expanded in terms of Whitney 1-form vector basis function $W^{(1)}$, and Whitney 2-form vector basis function $W^{(2)}$, respectively, as follows:

$$\vec{E}=\sum\nolimits_i e_i W_i^{(1)}, \quad (3)$$

$$\vec{B}=\sum\nolimits_i b_i W_i^{(2)}. \quad (4)$$

Testing Equations (1) and (2) with corresponding basis function by Galerkin technique, one can obtain elemental matrix equations as follows:

$$[T^e]\frac{\partial e}{\partial t}=[C^e]^T[K^e]^T b, \quad (5)$$

$$\frac{\partial b}{\partial t}=-[C^e]e, \quad (6)$$

where $[C^e]$ is an incidence matrix [9] and,

$$[T^e]=\varepsilon\int_\Omega\left(W_i^{(1)}\bullet W_j^{(1)}\right)dv, \quad (7)$$

$$[K^e]=\frac{1}{\mu}\int_\Omega\left(W_i^{(2)}\bullet W_j^{(2)}\right)dv. \quad (8)$$

Using the Crank-Nicolson (CN) formulation for time discretization, and assembling all the elemental equations, one can get the global electric field and magnetic flux update equations [9]:

$$\left([T]+\frac{\Delta t^2}{4}[G]\right)e^{n+1}=\left([T]-\frac{\Delta t^2}{4}[G]\right)e^n+[L]\Delta t b^n, \quad (9)$$

$$b^{n+1}=b^n-\frac{\Delta t}{2}[C]\left(e^{n+1}+e^n\right), \quad (10)$$

where $[G^e]=[C^e]^T[K^e][C^e]$ and $[L^e]=[C^e]^T[K^e]$. It can be seen that the update of magnetic flux is explicit. However, the update of electric field needs to solve a linear system at each time step, which is a major obstacle to the FETD solution. Hence, it is very necessary to develop an explicit update for the electric field Equation (9). In view of (9), the most direct way to achieve an explicit update can be denoted as:

$$e^{n+1}=[A]^{-1}\left([T]-\frac{\Delta t^2}{4}[G]\right)e^n+[A]^{-1}[L]\Delta t b^n, \quad (11)$$

where $[A]$ denotes the system matrix $\left([T]+\Delta t^2[G]/4\right)$. Although $[A]$ is sparse, $[A]^{-1}$ is dense. Direct computation of $[A]^{-1}$ is very costly and even impractical for large problems. Here, an efficient DSAI algorithm is developed for the computation of $[A]^{-1}$ to produce a fully explicit FETD method.

B. Data-sparse approximate inverse algorithm

The DSAI algorithm can be performed by five steps as follows: 1. Construct the data-sparse representation $[A]_{\mathrm{DS}}$ of the system matrix $[A]$. 2. Compute the data-sparse inverse $[A]_{\mathrm{DS}}^{-1}$ in formatted hierarchical matrix arithmetic. 3. Complete the explicit update as (11) by the matrix-vector product of $[A]_{\mathrm{DS}}^{-1}$ at each time step. The details are given below.

First, a cluster tree should be constructed. A cluster is a finite index set of basis functions in the FETD. We define an original cluster $I = \{1,2,\ldots,N\}$ to denote all the basis functions W_i. A cluster tree T_I is generated by recursive subdivision of I, as shown in Fig. 1 (a). One index set is subdivided into two subsets recursively, until the number of basis functions in the subset (denoted as "#") is smaller than a threshold n_{leaf}. The resulting cluster tree T_I is a binary tree, as shown in Fig. 1 (b).

Then, a block cluster tree $T_{I\times I}$ can be constructed by the hierarchical partitioning of $I\times I$. A block cluster tree is nothing but the interaction of two cluster trees: T_I of original basis function set and T_I of testing basis function set. The block cluster tree terminates at blocks $t\times s\in T_{I\times I}$ ($t\in T_I$ and $s\in T_J$) satisfying:

1. $\#t\le n_{\mathrm{leaf}}$ or $\#s\le n_{\mathrm{leaf}}$.
2. Clusters t and s satisfy admissibility condition of:
$$\min\{diam(\Omega_t),diam(\Omega_s)\}\le\eta dist(\Omega_t,\Omega_s), \quad (12)$$

where *diam* and *dist* denote the Euclidean diameter and distance of the supports of the basis functions in s, t, and $\eta>0$ controls the trade-off between admissible blocks. Blocks $t\times s\in T_{I\times I}$ satisfying (12) are called admissible blocks, which can be approximated by low-rank matrices in the following representation:

$$G=XY^T\ (G\in\mathbb{R}^{m\times n},X\in\mathbb{R}^{m\times k},Y\in\mathbb{R}^{n\times k},k\ll m,n). \quad (13)$$

The process of building $T_{I\times I}$ is shown in Fig. 1 (c). There are only two types of blocks in $T_{I\times I}$, i.e., admissible blocks stored as low-rank matrices and inadmissible blocks stored as full matrices.

For constructing the data-sparse representation $[A]_{\mathrm{DS}}$, all the non-zero matrix entries in $[A]$ are filled in inadmissible leaves while admissible leaves keep empty because the partial differential operator is local. Hence, the representation of $[A]_{\mathrm{DS}}$ is exact without approximation.

The obtained $[A]_{\mathrm{DS}}$ has a structure of a quad tree, which be written as:

$$A=\begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}. \quad (14)$$

Then, $[A]_{\mathrm{DS}}^{-1}$ can be computed recursively from this 2×2 partitioned block matrix as follows:

$$A^{-1}=\begin{bmatrix} A_{11}^{-1}+A_{11}^{-1}A_{12}S^{-1}A_{21}A_{11}^{-1} & -A_{11}^{-1}A_{12}S^{-1} \\ -S^{-1}A_{21}A_{11}^{-1} & S^{-1} \end{bmatrix}, \quad (15)$$

where $S=A_{22}-A_{21}A_{11}^{-1}A_{12}$. The inverses A_{11}^{-1} and S^{-1} are computed using (15) recursively. Here, the exact addition and multiplication are replaced by the formatted hierarchical matrix counterparts ($\oplus$ and $\otimes$). $[A]_{\mathrm{DS}}^{-1}$ has the same block cluster tree structure as $[A]_{\mathrm{DS}}$, whereas the admissible blocks are not empty but filled with non-zero entries during the recursive inversion. The DSAI algorithm can reduce the computational complexity and storage requirement to be O($k^2N\log^2N$) and O($kN\log N$), respectively [12,13]. Here, k is the average rank of low-rank matrices. Once $[A]_{\mathrm{DS}}^{-1}$ is obtained, the MVP can be performed with O($kN\log N$) computational complexity to complete a explicit update at each time step.

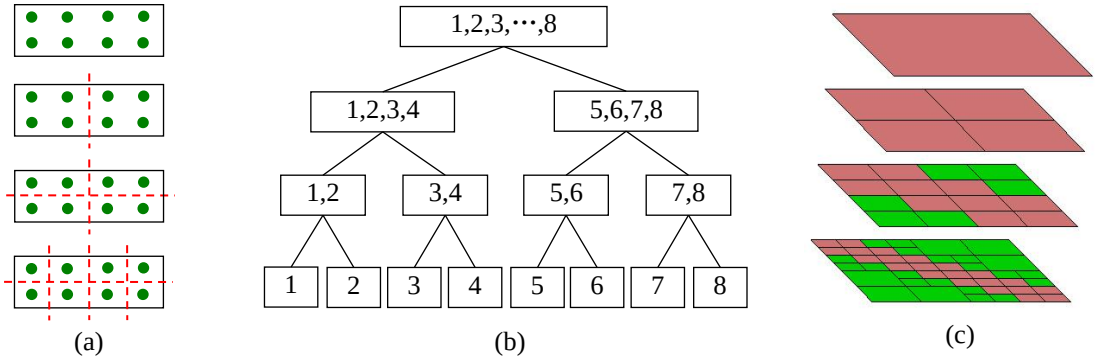


Fig. 1. (a) Recursive subdivision of the basis function set. (b) Cluster tree T_I. (c) Block cluster tree $T_{I\times I}$ based on T_I. Full matrices are red and low-rank matrices are green.

C. Recompression technique

As mentioned above, the DSAI algorithm compresses the originally dense inverse matrix to a data-sparse matrix, in which each admissible block $G_{m\times n}$ have a low-rank representation of $G_{m\times n}=X_{m\times k}Y_{n\times k}^T$ ($k\ll m,n$). However, these admissible blocks contain redundant information, and can be further compressed. In this section, a recompression technique is introduced to improve the efficiency of the explicit FETD solution.

In the recompressed inverse matrix $[A]_{\mathrm{RDS}}^{-1}$, any admissible block $G'_{m\times n}$ has a new low-rank representation of $G'_{m\times n}=U_{m\times k_1}S_{k_1\times k_2}V_{n\times k_2}^T$ ($k_1,k_2\ll m,n$), where U and V are called row cluster basis and column cluster basis, which satisfy $U^TU=I$ and $V^TV=I$, respectively. S is a coupling matrix.

Before introducing the recompression process, we define a far cluster set Far($\cdot$). For any cluster i on level l, Far(l_i) denotes the set of clusters satisfying the admissibility condition (12). If cluster $j\in$ Far(l_i), the interaction of i and j will lead to an admissible block $G_{m\times n}^{lij}=X_{m\times k}^{lij}(Y_{n\times k}^{lij})^T$, as shown in Fig. 2.

The recompression process can be divided into the following four steps:

1. For all $j\in$ Far(l_i), extract $X_{m\times k}^{lij}$ and put them together in sequence as $M^{li}=(\cdots,X_{m\times k}^{lij},\cdots)$.
2. Perform singular value decomposition (SVD) for the assembled matrix M^{li}, and truncate it with a pre-determined accuracy ε_t, we obtain $M^{li}=\tilde{U}^{li}\tilde{S}^{li}(\tilde{V}^{li})^T$, where $\tilde{U}^{li}$ and $\tilde{V}^{li}$ are orthogonal matrices, and $\tilde{S}^{li}$ is diagonal matrix. Matrix $\tilde{U}^{li}$ is the row cluster basis for cluster i on level l. U^{li} is also the column cluster basis for symmetric system matrix. The process of step 1 and 2 are described in Fig. 3.
3. For $j\in$ Far(l_i), $(\tilde{V}^{li})^T$ can be written as $(\cdots,\tilde{V}^{lij},\cdots)^T$. The coupling matrix S^{lij} of cluster i and j can be found by $S^{lij}=\tilde{S}^{li}(\tilde{V}^{lij})^T(Y^{lij})^T\tilde{U}^{lj}$.
4. For each level l ($2\le l\le L$, L is the depth of the cluster tree), repeat step 1 to 3 for all cluster to obtain all cluster basis matrices and coupling matrices.

In the resulting recompressed matrix $[A]_{\mathrm{RDS}}^{-1}$, each admissible block has a representation of $G_{m\times n}^{lij\prime}=G_{m\times n}^{lij\prime}=U_{m\times k_1}^{li}S_{k_1\times k_2}^{lij}(U_{n\times k_2}^{lj})^T$. Although $G'_{m\times n}$ has one more factor than $G_{m\times n}$, the compression degree of $[A]_{\mathrm{RDS}}^{-1}$ is higher than $[A]_{\mathrm{DS}}^{-1}$. The reason is that all the admissible blocks with the same row cluster share a row cluster basis U, and those with the same column cluster share a column cluster basis V. $[A]_{\mathrm{RDS}}^{-1}$ enters the explicit solution of FETD by the MVP at each time step.

The computational complexity of the MVP of $[A]_{\mathrm{RDS}}^{-1}$ is O($kN\log N$), which is the same as that of $[A]_{\mathrm{DS}}^{-1}$. However, the MVP of $[A]_{\mathrm{RDS}}^{-1}$ can be much faster because it reduces the constant in the complexity estimation. Hence, the recompression technique can further accelerate the efficiency of the FETD.

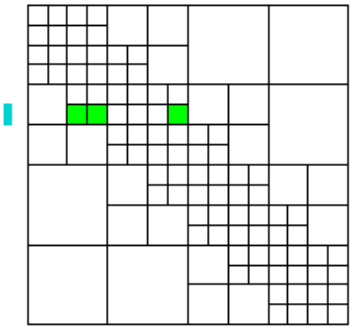


Fig. 2. The far cluster set Far($\cdot$).

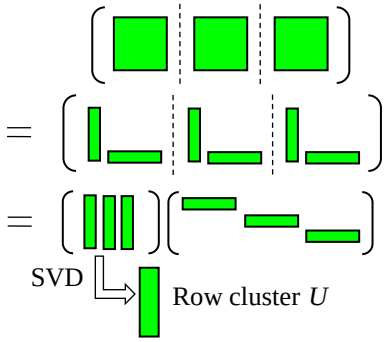


Fig. 3. Construction of the row cluster U.

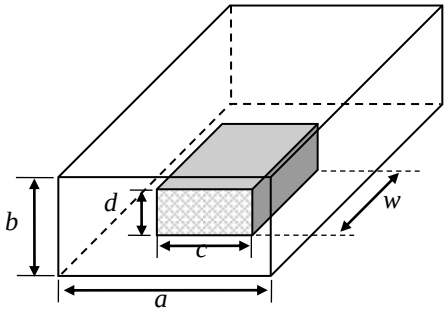
III. NUMERICAL EXAMPLES

In this section, three numerical examples are presented to demonstrate the efficiency and accuracy of the proposed method for the fast simulation of microwave devices. For testing the accuracy of $[A]_{\mathrm{DS}}^{-1}$, the relative error is defined as $\delta = \left\| I - [A]_{\mathrm{DS}}^{-1}[A] \right\| / \|I\|$, where I is identity matrix and $\|\cdot\|$ denotes 2-Norm.

A. A dielectric-filled rectangular waveguide example

The first example analyzes a rectangular waveguide filled with a half-height dielectric block. The configuration and dimensions of this waveguide are presented in Fig. 4.

In order to obtain an input reflection coefficient, perfect match layers (PMLs) are placed at the input and out ports to simulate the matched loads. A modulated Gaussian pulse is applied with a center frequency of $f_0 = 10$ GHz and a bandwidth of 4 GHz. The time step size is $\Delta t = 1.67$ ps, which is ten times larger than conventional FETD method due to the unconditional stability of CN scheme. For the data-sparse approximate inversion, η in the admissibility condition (12) is set to be $\eta = 1.0$ and the minimal block size is chosen as $n_{\mathrm{leaf}} = 32$. The truncated rank k is set to be 10 and the accuracy of approximate inverse can reach the order of 10^{-5}, which is enough for an accurate solution. As shown in Fig. 5, the S parameters computed by the proposed method agree well with the results simulated by Ansoft HFSS software.



a=20.0, b=10.0, c=8.88, d=3.99, w=8.0 (Unit: mm)

Fig. 4. Dimensions of the dielectric-filled waveguide.

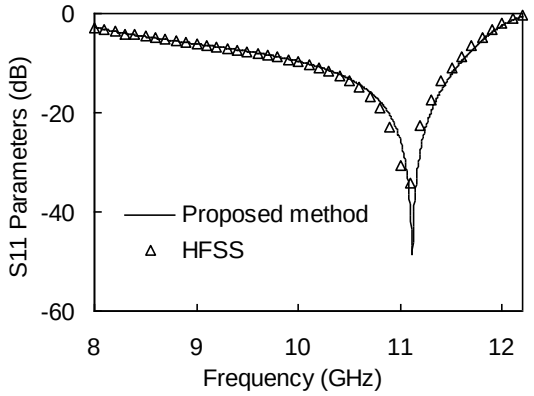


Fig. 5. S11 parameter of the dielectric-filled waveguide.

Then, the performance of data-sparse approximate inversion algorithm is tested for different mesh sizes and fixed accuracy. We increase the umber of unknowns N to test the time and memory costs of the DSAI algorithm, by increasing the mesh density and reducing Δt proportionally with the mesh density. As the theoretical analysis, the time and memory usages actually should be estimated as O($k^2N\log^2N$) and O($kN\log N$). For electrodynamic analysis, k will increase with N, but it is usually much smaller than N for the problems of moderate size. Hence, the time and memory usages can be observed to be very close to O($N\log^2N$) and O($N\log N$) for this example, as shown in Fig. 6 and Fig. 7.

In Table 1, the DSAI algorithm is compared with full matrix inversion algorithm for the explicit formulation. It can be seen that the DSAI can highly compress the original full inverse matrix with sufficient accuracy, and subsequently decrease the solution time.

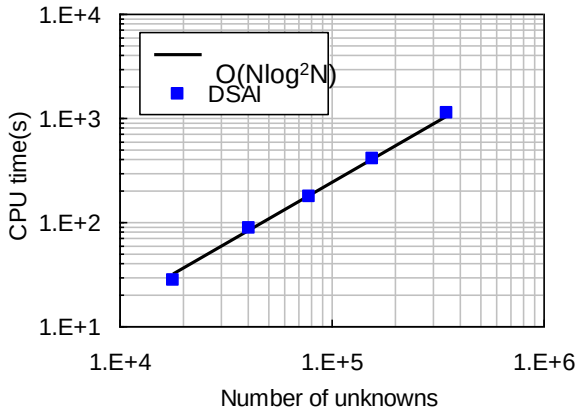


Fig. 6. Time usages for the DSAI algorithm.

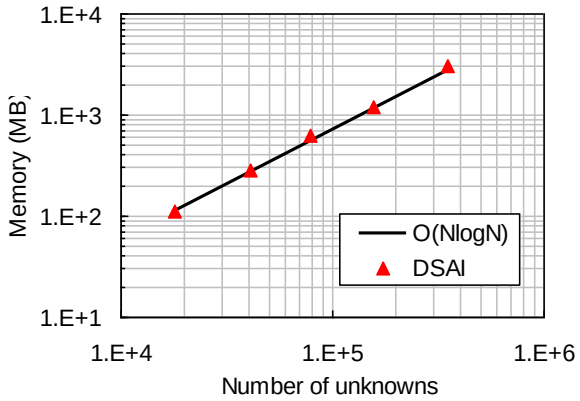


Fig. 7. Memory usages for the DSAI.

Table 1: Comparisons between the DSAI and full matrix inversion algorithm

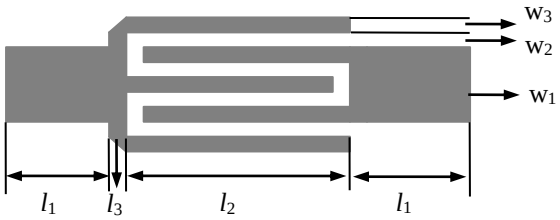
Mesh Sizes	N_{unk}	M_{full} (GB)	M_{DSAI} (GB)	R_{comp}
Case 1	9,775	0.76	0.13	17.1 %
Case 2	18,068	2.49	0.26	10.4 %
Case 3	40,672	12.62	0.67	5.3 %

N_{unk}: number of unknowns.
M_{full}: memory required for the full inverse matrix.
M_{DSAI}: memory required for the DSAI matrix.
R_{comp}: compression ratio computed by M_{DSAI}/M_{full}.

B. An interdigital microstrip example

The second example deals with an interdigital capacitor microstrip filter. Its dimensions are shown in Fig. 8. The microstrip line is on a dielectric substrate with a height of h=0.508 mm and a relative inductivity of $\varepsilon_r = 2.2$.

A modulated Gaussian pulse is applied with a center frequency of $f_0 = 6.5$ GHz and a bandwidth of 5 GHz. The number of time steps is 2700. Mesh discretization generates 85,172 unknowns. The DSAI associated parameters are set to be $\eta = 2.0$ and $n_{leaf} = 16$. Figure 9 shows the S parameters computed by the proposed method in the case of truncated rank k=5, and compares the results with those simulated by Ansoft Designer software.



l_1=8.05, l_2=8.4, l_3=0.3, w_1=1.51, w_2=w_3=0.3 (Unit: mm)

Fig. 8. Dimensions of the interdigital microstrip.

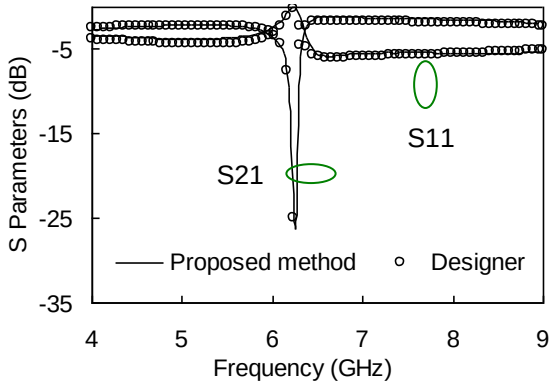


Fig. 9. S parameters of the interdigital microstrip.

Table 2 shows the performance of the DSAI algorithm. It can be seen that the inversion accuracy δ improves but the computational costs rise with the rank k increasing. The performance of the proposed recompression technique is tested for different mesh sizes and the results are reported in Table 3. For the three cases in the test, the accuracy of the solutions with and without recompression is controlled to be the same order of 10^{-5} by adjusting the rank k and the truncated SVD error ε_t. It is obvious that the total solution time can be significantly reduced by using the recompression technique. The main reason is that the recompression technique can further accelerate the DSAI-based MVP. Figure 10 demonstrates this conclusion more intuitively. As shown in Fig. 10, the recompressed MVP outperforms the unrecompressed MVP, although they both can be observed to scale nearly with O(Nlog$N$) computational costs.

Table 2: Performance of the DSAI algorithm

k	δ	T_{inv} (s)	M_{inv} (MB)	T_{MVP} (s)	T_{total} (s)
1	2.9e-3	289	364	0.64	1,758
5	1.0e-4	438	510	0.91	2,487
10	3.2e-5	763	691	1.23	3,391

T_{inv}: time used for DSAI algorithm.
M_{inv}: memory required for the data-sparse inverse matrix.
T_{MVP}: time used for the matrix-vector product of the data-sparse inverse matrix.
T_{total}: total solution time for all time steps.

Table 3: Performance of the recompression technique

Mesh Sizes	Without Recompression		With Recompression		
	T_{MVP}	T_{total}	T_{recomp}	T_{MVP}	T_{total}
Case 1	0.48	1,345	102.3	0.28	858
Case 2	1.37	3,780	323.3	0.79	2,541
Case 3	4.57	12,492	582.6	2.73	7,955

T_{MVP}: time used for the MVP of the inverse matrix.
T_{recomp}: time used for the recompression of the inverse matrix.
T_{total}: total solution time for all time steps.

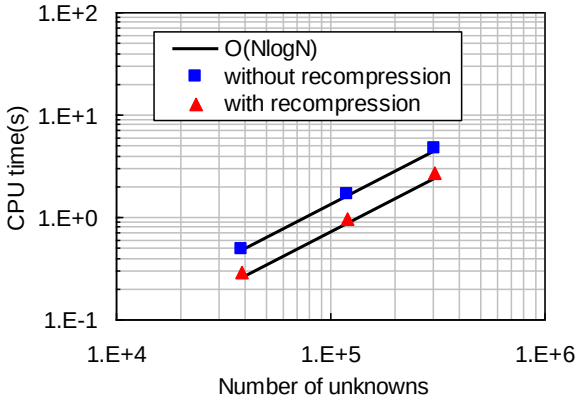
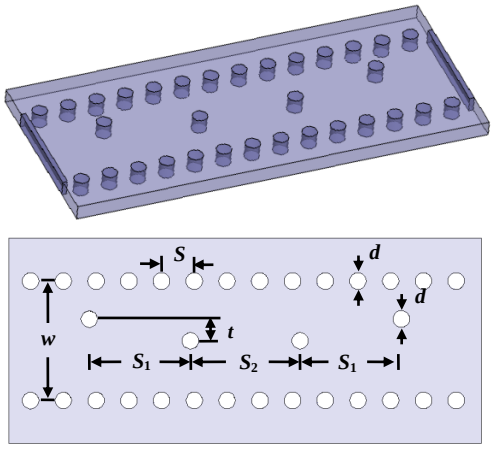


Fig. 10. Time usages for the data-sparse MVP.

C. A substrate integrated waveguide example

The last example considers a substrate integrated waveguide (SIW). SIW is a new type of microwave transmission line emerged in recent years, which holds the advantages of both conventional waveguide and microstrip line. The configuration and detailed dimensions of the SIW are presented in Fig. 11, respectively. The permittivity of the dielectric substrate is $\varepsilon_r = 2.2$ and the height of the substrate is h=0.787 mm. A modulated Gaussian pulse with a center frequency of $f_0 = 28$ GHz and a bandwidth of 8 GHz is adopted in our simulation.

In Fig. 12, S parameters computed by the proposed method are plotted together with the measurement results presented in [15], and good agreement can be observed. Then, the performance of the proposed recompression technique is tested. The numerical results are reported for three cases of different solution accuracy δ in Table 4. It can be found that the recompression technique can further reduce the total solution time of the FETD with the same order of solution accuracy.



w=5.563, S=1.525, d=0.775, t=1.01, S_1=4.71, S_2=5.11 (mm)

Fig. 11. Configuration and dimensions of the SIW.

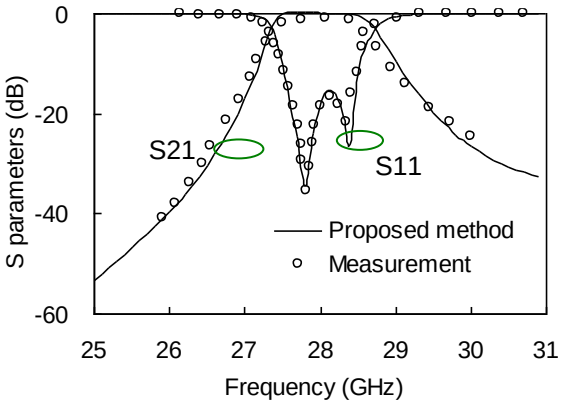


Fig. 12. S parameters of the SIW.

Table 4: Performance of the recompression technique

δ	Without Recompression		With Recompression		
	T_{MVP}	T_{total}	T_{recomp}	T_{MVP}	T_{total}
10^{-4}	1.42	3,620	281.5	0.72	2,153
10^{-6}	2.27	5,748	467.6	1.35	3,842
10^{-8}	3.91	10,109	834.1	2.23	6,558

IV. CONCLUSION

A data-sparse and fully explicit FETD method is developed for the fast electromagnetic simulation. By the data-sparse approximate inverse (DSAI) algorithm,

the inverse of the system matrix of CN-FETD electric field update equation can be computed and stored in a data-sparse way with almost linear costs, and then the solution at each time step is fully explicit. A recompression technique is proposed to further compress the data-sparse inverse matrix to improve the efficiency of explicit solution at each time step. Numerical results demonstrate the proposed method is robust for the simulation of microwave devices.

ACKNOWLEDGMENT

This work was supported by Open Research Program Foundation of State Key Laboratory of Millimeter Waves of K201411, Natural Science Foundation of Jiangsu Province of BK20140893, Natural Science Foundation of the Higher Education Institutions of Jiangsu Province of 14KJB510023, and NUPTSF of NY214040.

REFERENCES

[1] J. F. Lee, R. Lee, and A. Cangellaris, "Time-domain finite-element methods," *IEEE Trans. Antennas Propag.*, vol. 45, pp. 430-442, Mar. 1997.

[2] D. A. White, "Orthogonal vector basis functions for time domain finite element solution of the vector wave equation," *IEEE Trans. Magn.*, vol. 35, no. 3, pp. 1458-1461, May 1999.

[3] D. Jiao and J. M. Jin, "Three-dimensional orthogonal vector basis functions for time-domain finite element solution of vector wave equations," *IEEE Trans. Antennas Propag.*, vol. 51, no. 1, pp. 59-66, Jan. 2003.

[4] J. H. Lee, T. Xiao, and Q. H. Liu, "A 3-D spectral-element method using mixed-order curl conforming vector basis functions for electromagnetic fields," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 1, pp. 437-444, Jan. 2006.

[5] B. He and F. L. Teixeira, "Sparse and explicit FETD via approximate inverse Hodge (mass) matrix," *IEEE Microw. Wireless Compon. Lett.*, vol. 16, no. 6, pp. 348-350, June 2006.

[6] J. Kim and F. L. Teixeira, "Parallel and explicit finite-element time-domain method for Maxwell's equations," *IEEE Trans. Antennas Propag.*, vol. 59, no. 6, pp. 2350-2356, June 2011.

[7] A. S. Moura, E. J. Silva, R. R. Saldanha, and W. G. Facco, "Performance of the alternating direction implicit scheme with recursive sparsification for the finite element time domain method," *IEEE Trans. Magn.*, vol. 51, no. 3, Mar. 2015.

[8] M. Movahhedi, A. A. H. Ceric, A. Sheikholeslami, and S. Selberherr, "Alternation-direction implicit formulation of the finite-element time-domain method," *IEEE Trans. Microw. Theory Tech.*, vol. 55, no. 6, pp. 1322-1331, 2007.

[9] R. S. Chen, L. Du, Z. B. Ye, and Y. Yang, "An efficient algorithm for implementing Crank-Nicolson scheme in the mixed finite-element time-domain method," *IEEE Trans. Antennas Propag.*, vol. 57, no. 10, pp. 3216-3222, Oct. 2009.

[10] M. Bebendorf and W. Hackbusch, "Existence of H-matrix approximants to the inverse FE matrix of elliptic operators with L∞-coefficients," *Numer. Math.*, 95, pp. 1-28, 2003.

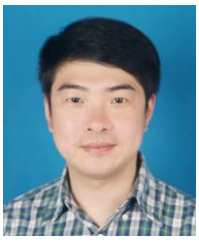
[11] L. Grasedyck and W. Hackbusch, "Construction and arithmetics of H-matrices," *Computing*, vol. 70, no. 4, pp. 295-344, Aug. 2003.

[12] S. Börm, L. Grasedyck, and W. Hackbusch, "Introduction to hierarchical matrices with applications," *Engineering Analysis with Boundary Elements*, no. 27, pp. 405-422, 2003.

[13] H. Liu and D. Jiao, "Existence of H-matrix representations of the inverse finite-element matrix of electrodynamic problems and H-based fast direct finite-element solvers," *IEEE Trans. Microw. Theory Tech.*, vol. 58, no. 12, pp. 3697-3709, 2010.

[14] S. Borm, "Data-sparse approximation by adaptive H^2-matrices," *Computing*, vol. 69, no. 1, pp. 1-35, Sep. 2002.

[15] M. Bozzi, L. Perregrini, and K. Wu, "Modeling of conductor, dielectric, and radiation losses in substrate integrated waveguide by the boundary integral-resonant mode expansion method," *IEEE Trans. Microw. Theory Tech.*, vol. 56, no. 12, pp. 3153-3161, 2008.



Ting Wan received the Ph.D. degree in Information and Communication Engineering from Nanjing University of Science and Technology.

He is currently a Lecturer of Nanjing University of Posts and Telecommunications. His research interests include computational electromagnetics, electromagnetic scattering and radiation, and electromagnetic modeling of microwave/ millimeter wave integrated circuits.

Lei Du received the Ph.D, degrees in Electromagnetic Fields and Microwave Technology from Nanjing University of Science and Technology, China, in 2010.

Her current research interest is computational electromagnetics.

Jian Zhu received the Ph.D. degrees in Electromagnetic Fields and Microwave Technology from Nanjing University of Science and Technology (NUST), China, in 2010.

She is currently a Lecturer of Nanjing University of Posts and Telecommunications. Her current research interests include CEM and antennas.