

Rapid Evaluation of Macromodel Response with the FDTD Method

Dmitry A. Gorodetsky and Philip A. Wilsey

Department of ECECS, University of Cincinnati, P.O. Box 210030, Cincinnati, OH 45221-0030 USA

Abstract-Full wave electromagnetic simulation requires numerically expensive methods such as FDTD. The computation time depends superlinearly on the number of unknowns in the simulation region. In some situations, especially when the results are not needed at every point of the grid, simulation time can be reduced. This reduction can be accomplished by partitioning the grid into macromodels and determining the macromodel impulse response. Then the impulse response can be decomposed into its eigenmodes, some of which can be eliminated because they are non-essential. In this paper we extend our earlier work and describe several strategies with which macromodels can be interconnected, that result in further savings of computation time.

Keywords: FDTD methods, Design automation, Reduced order systems, Macromodeling.

I. INTRODUCTION

One of the most efficient and easy to program methods to solve Maxwell's equations in the time domain is FDTD [1,10]. FDTD works by recursively solving the Maxwell's equations in the entire domain during every iteration. The recursive procedure continues until some pre-determined condition is achieved. Schemes of this type are often used when the analytical solution to an electromagnetic problem is prohibitive. Problems to be solved with FDTD are abundant in simulations of aircraft radar cross section at high frequency, microwave ICs, optical pulse propagation, antennas, bioelectromagnetic systems, bodies of revolution, etc. [1]. Real-life problems often require grids with very large numbers of points, due to fine features of the simulated objects and short excitation wavelengths. The simulation time is exacerbated by modeling complex and composite materials, as well as increasing the order of the simulation in space and time to reduce error [2,3]. In some cases the simulation time can become unreasonable which has lead researchers to look for alternatives to conventional approaches. One way with which it may be possible to reduce this simulation time to a more acceptable level is presented in this paper.

The generation of macromodels for simulations has attracted some attention recently [4,5,6]. A

macromodel (subcell model) is basically an encapsulation of some simulation region. A macromodel establishes an interface between itself and the surrounding region. The authors have described how to create the macromodel impulse response matrix from the state transition matrix that is easily obtained by inspection of the FDTD method [8]. The advantage of such macromodels is that the impulse response history is computed only once and permanently saved as part of the model. In the future, this history can be reused as the surrounding region changes. If the solution within the macromodel is not important, then further savings can be made by computing the solution at a limited number of interface points. Conversion of macromodels to equivalent electrical circuits to model the entire system with circuit simulators when EM simulators are not suitable is straightforward [5]. Conversion to system level is also possible in order to allow for behavioral modeling [7]. The use of a similar method to construct primitive boundary conditions has also been studied [14].

The structure of the paper is as follows. The theoretical derivations of the eigenmodal method are discussed in Section II. In [9] we discussed the operation of a single module. Here we remove this assumption and we extend our earlier work to a system that consists of multiple modules. In Section III we discuss how the proposed system can handle various types of excitation scenarios such as steady state and transient excitations, as well as initial conditions. We consider several alternative methods to evaluate the response of a cascade of multiple modules in Section IV. Our experimental results from waveguide simulation are given and discussed in Section V. Finally section VI concludes the paper with a summary and suggestions for future research.

II. THEORY

The results of our earlier work describe the advantages and disadvantages of computing the impulse response of macromodels [8]. We have overcome some of the disadvantages with the introduction of the eigenmodal decomposition approach [9]. This involves expressing the inputs to a macromodel (module) as a superposition of the eigenvalues of the state transition matrix. The

outcome is that constant time for the convolution can be achieved. Further improvement can be made if some eigenmodes are discarded [9,16].

The method proposed in [9] discusses the interface of a modular region and an FDTD region. The output of the modular region is obtained by performing convolution of the inputs with the impulse response by keeping track of the coefficients for each eigenmode as follows:

$$\begin{aligned}
\mathbf{Y}(1) &= \mathbf{h}(1) \mathbf{X}(0) \\
&= \lambda_0 a_{00} \mathbf{v}_0 + \lambda_1 a_{01} \mathbf{v}_1 + \dots + \lambda_N a_{0N} \mathbf{v}_N \\
&= s_{10} \mathbf{v}_0 + s_{11} \mathbf{v}_1 + \dots + s_{1N} \mathbf{v}_N \\
\mathbf{Y}(2) &= \mathbf{h}(2) \mathbf{X}(0) + \mathbf{h}(1) \mathbf{X}(1) \\
&= \lambda_0 s_{10} \mathbf{v}_0 + \lambda_1 s_{11} \mathbf{v}_1 + \dots + \lambda_N s_{1N} \mathbf{v}_N \\
&+ \lambda_0 a_{10} \mathbf{v}_0 + \lambda_1 a_{11} \mathbf{v}_1 + \dots + \lambda_N a_{1N} \mathbf{v}_N \\
&= \frac{\lambda_0 s_{10} \mathbf{v}_0 + \lambda_1 s_{11} \mathbf{v}_1 + \dots + \lambda_N s_{1N} \mathbf{v}_N}{s_{20} \mathbf{v}_0 + s_{21} \mathbf{v}_1 + \dots + s_{2N} \mathbf{v}_N} \\
&\quad \dots \\
&\quad \dots \\
\mathbf{Y}(n) &= \mathbf{h}(n) \otimes \mathbf{X}(n), \tag{1}
\end{aligned}$$

where \otimes represents convolution, $\mathbf{h}(n)$ is the impulse response, \mathbf{v}_k are the eigenvectors, \mathbf{X} and \mathbf{Y} are the inputs and the outputs of the modular region, respectively, and λ_k are the eigenvalues of \mathbf{A} , the state transition matrix, all defined in [8]. In the above, each input is decomposed into a sum of eigenvectors (\mathbf{v}_k) of the state transition matrix, modulated by the coefficients a_k . In order to evolve the state of the module by just one time step, each eigenvector must be multiplied by its eigenvalue λ_k [16].

In order for the module to express the inputs as a superposition of eigenmodes as shown in (1) it must solve N equations in N unknowns. The matrix equation to be solved is [8]:

$$\begin{aligned}
\mathbf{V} \mathbf{a} &= \mathbf{B}_1 \mathbf{X} \\
\Rightarrow \mathbf{a} &= \mathbf{V}^{-1} \mathbf{B}_1 \mathbf{X}, \tag{2}
\end{aligned}$$

where \mathbf{a} is the vector containing the coefficients a_k used in (1) and \mathbf{B}_1 is a selector matrix for one interface input.

The matrix $\mathbf{V}=[\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N]$ does not change and therefore its inverse must be computed only once. The stepwise computation in (2) requires NI multiplications, where N is the total number of points in the module and I is the number of inputs. The computation time of (1) and (2) can be significantly reduced in some situations by discarding non-essential eigenmodes [9]. We have shown that the time taken to compute (1) and (2) with our method can be up to *four times less* than the time to compute the same results by FDTD [18].

The more modules are used, the greater the improvement in the computation time. The advantage to having multiple modules is that they can be made to

communicate in terms of their coefficients, with a linear transformation [17] replacing the stepwise calculation of (2). The number of multiplications with the linear transformation is decreased to P^2 , where P is the number of modes remaining after the reduction.

In our earlier work we assumed that the excitation plane is located in the FDTD region. If the excitation region is partially or fully located within a module then some simplifications can be made, as described below.

III. EXCITATION

A. Transient Excitation

In the first mention of the FDTD method, only the propagation of a single-pulse in a lossless medium is described [10]. To initialize the system, the state of the electromagnetic field is specified at time step 1. For the following time steps the FDTD algorithm takes over the propagation and scattering of the pulse in time and in space. If a portion of the pulse is initially located somewhere *within* the module, then we have a *non-zero* state system that must be converted to one with *zero* state. This can be done if we assume that the system (module) is *completely controllable*. In this case the equation to solve is given by:

$$\mathbf{V} \mathbf{a} = \mathbf{Q}, \tag{3}$$

where \mathbf{Q} denotes the system state as described in [7]. We end up with a set of non-zero coefficients a_{0k} .

B. Steady State Excitation

For long duration sources Taflove argues that Yee's excitation method can lead to an excessive simulation time. Taflove's suggestion is to perform stepwise computation of the desired time function (sinusoidal, Gaussian, etc) for the Electric field components in the FDTD domain [11,12]. Usually these field components are the same across an entire plane. We consider sinusoidal excitation below.

To use this approach with our method, assume the excitation is in the form:

$$E_z(i=i_s, j, k, n\Delta t) = E_0 e^{j\omega n\Delta t}. \tag{4}$$

The first step is to solve (2) with every component of \mathbf{X} equal to E_0 . This is the situation for $n=0$. We will end up with a set of non-zero coefficients b_{0k} which should be kept separate from a_{0k} . Taking the real part of (4), every successive input will be equal to the multiplication of $\sin(\omega n\Delta t)$ by E_0 . Since E_0 is written as a superposition of eigenvectors, each of the b_{0k} must be multiplied by the sinusoid as follows:

$$\begin{aligned}
\mathbf{Y}(1) &= \mathbf{h}(1) \mathbf{X}(0) \\
&= \lambda_0 b_{00} \mathbf{v}_0 + \lambda_1 b_{01} \mathbf{v}_1 + \dots + \lambda_N b_{0N} \mathbf{v}_N \\
&= s_{10} \mathbf{v}_0 + s_{11} \mathbf{v}_1 + \dots + s_{1N} \mathbf{v}_N \\
\mathbf{Y}(2) &= \mathbf{h}(2) \mathbf{X}(0) + \mathbf{h}(1) \mathbf{X}(1) \\
&= \lambda_0 s_{10} \mathbf{v}_0 + \lambda_1 s_{11} \mathbf{v}_1 + \dots + \lambda_N s_{1N} \mathbf{v}_N \\
&+ \sin(x) [\lambda_0 b_{00} \mathbf{v}_0 + \lambda_1 b_{01} \mathbf{v}_1 + \dots + \lambda_N b_{0N} \mathbf{v}_N] \\
&= s_{20} \mathbf{v}_0 + s_{21} \mathbf{v}_1 + \dots + s_{2N} \mathbf{v}_N, \quad (5)
\end{aligned}$$

where $x = \text{Re}(j\omega n \Delta t) = \omega n \Delta t$. With this approach, the number of multiplications is reduced from NI to N because (2) does not have to be solved for every time step.

IV. MULTIPLE MODULES

A. Impulse Response

The number of interfaces that a module will have to the outside world will usually depend on whether partitioning is done in one, two, or three dimensions. Several alternatives exist for the simulation of a cascade of modules. One way, discussed in detail in [8] involves writing the impulse responses due to each interface of a two interface module as follows:

$$\begin{bmatrix} \mathbf{Y}_1(n) \\ \mathbf{Y}_2(n) \end{bmatrix} = \begin{bmatrix} \mathbf{h}_{11}(n) & \mathbf{h}_{12}(n) \\ \mathbf{h}_{21}(n) & \mathbf{h}_{22}(n) \end{bmatrix} \otimes \begin{bmatrix} \mathbf{X}_1(n) \\ \mathbf{X}_2(n) \end{bmatrix}. \quad (6)$$

The above equation can be simplified to use the method shown in (1). However, in order to do that, we must first solve (2) with the modification shown below:

$$\mathbf{V}\mathbf{a} = \mathbf{B}_{1,2} [\mathbf{X}_1 \mathbf{X}_2]^{\text{TR}}. \quad (7)$$

An alternative method for cascade modules is to combine every two modules into one such that we end up with only one module. As shown in Fig.1, the combination process begins with the two modules that are closest to the boundary. Their impulse responses can be combined as follows [15]:

$$\begin{aligned}
\mathbf{h} &= \mathbf{h}_{A11} + \mathbf{h}_{A12} \otimes (\mathbf{h}_B \otimes \mathbf{h}_{A21}) + \mathbf{h}_{A12} \otimes \\
&(\mathbf{h}_B \otimes (\mathbf{h}_{A22} \otimes (\mathbf{h}_B \otimes \mathbf{h}_{A12}))) + \dots \quad (8)
\end{aligned}$$

B. Linear Transformation

Linear algebra tells us that if the eigenvalues are distinct, then the corresponding eigenvectors are linearly independent. If there are N linearly independent eigenvectors, then they constitute a basis for the N -dimensional vector space. Any vector in an N -dimensional vector space can be written as a sum of the N eigenvectors in the basis [17]. Using these theorems allows us to express the input as a superposition of N eigenvectors without any loss of data. When the coefficients are exchanged between

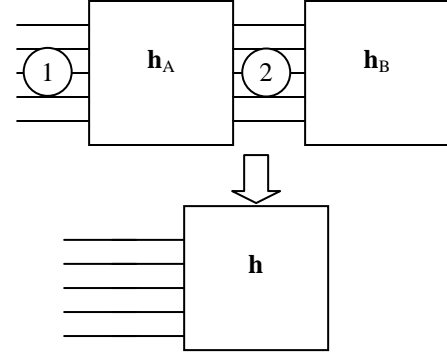


Fig. 1. Cascade connection of two modules.

adjacent modules, they will need to go through a rotation transformation in order to relocate the field from the rightmost position of the output module to the leftmost position of the input module and vice versa. Additionally, if the medium is inhomogeneous, the coefficients will have to be converted from one basis to another. Below we give details of the linear transformation whereas the details of the change of basis can be found elsewhere [17].

In order to construct a conversion matrix we need to understand what the linear transformation will do to the basis vectors \mathbf{v}_k . An example of the left to right transformation for a single vector is shown below:

$${}^{\text{T}} \begin{bmatrix} \mathbf{E}(1) \\ \mathbf{E}(2) \\ \mathbf{H}(1) \\ \mathbf{H}(2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mathbf{H}(2) \\ 0 \end{bmatrix}$$

Fig. 2. Example of linear transformation from left to right.

By performing the linear transformation for every vector \mathbf{v}_k we will end up with a set of vectors \mathbf{e}_k that can be saved in the matrix \mathbf{E} . After the one time computation $\mathbf{F} = \mathbf{V}^{-1} \mathbf{E}$, we end up with \mathbf{F} which is the matrix representation of the linear transformation from left to right. The linear transformation from right to left is obtained in an analogous manner. Now, in order to obtain the input coefficients for the right module, the following computation will be required:

$$\mathbf{a}_r = \mathbf{F}\mathbf{a}_l, \quad (9)$$

where the \mathbf{a} 's are the coefficient vectors discussed in (2).

V. RESULTS

A parallel plate waveguide structure was simulated with the conventional FDTD and with the algorithm presented in this paper. The eigenvectors were discarded according to the procedure outlined in [9]. The waveguide was excited with a constant plane wave source of 10 GHz at $x=1$ and shorted at the other end. A thousand time steps were required to reach a stable solution. The x,y,z mesh dimensions were 133,20,9 cells, respectively. The modules were adjacent to each other in cells at locations $x=58$ and $x=59$. In Fig. 3, the electric field values in location $x=58$ obtained with both methods are compared. As can be seen, the approximate results are very close to the exact ones calculated with the FDTD algorithm, and degrade only slightly near the end of the simulation.

A combined impulse response using (8) was obtained for all the modules of an inhomogeneously filled waveguide and keeping the first ten terms. Special care had to be taken to keep the round-off error to a minimum. The results are plotted in Fig. 4. As can be seen, the primary consequence of dropping the terms is that the electric field values do not reach the peak. This is due to the infinite impulse response (IIR) nature of the FDTD modules.

VI. CONCLUSION

In this paper we discussed a modular alternative to the brute-force FDTD approach to perform full-wave simulation of the propagation of the electromagnetic field. Simulation time for each module is reduced by nearly a factor of four as was shown by our earlier work [18]. We showed that our recursive algorithm can provide good results even with multiple modules while reducing the simulation time. We considered different types of excitations possible with the simulation and introduced new methods to work with them. We discussed how a system with multiple modules can lead to even further reduction in simulation time by using linear transformations. Techniques to obtain the impulse response of multiple modules were given and their effects on the simulation results were shown.

An advantage of using linear transformations is to be able to communicate only with coefficients. The number of coefficients is equal to the number of non-discarded modes. This becomes useful as the modules are distributed among multiple processors because communication time can be decreased in comparison to the conventional FDTD which must communicate the data for every single point on the boundary. Another advantage of using linear transformations is that the output of the inner modules does not have to

be computed during every time step, which leads to further savings of computation time.

Future work will involve the investigation into the techniques to utilize the modular algorithm in parallel computation. In particular, the use of linear transformations as a means to further reduce communication and computation will be studied. Also, parallelizable methods to solve systems of simultaneous equations that do not require matrix inversion will be considered.

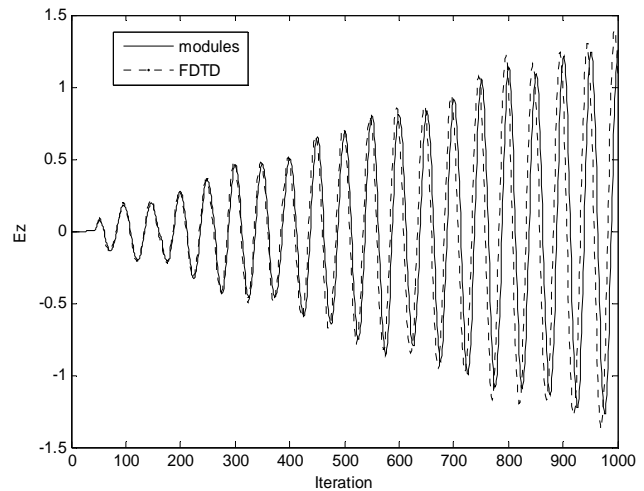


Fig. 3. Comparison between the output of the modules and the output of conventional FDTD for one period. Results indicate a very close match in the waveforms.

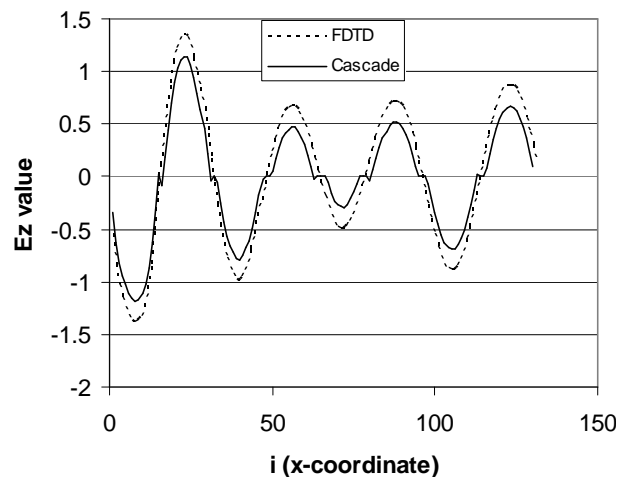


Fig. 4. Spatial Variation of The Electric Field. Results obtained using (8) are compared with conventional FDTD. The y,z location is the same as in Fig. 2, $n=1000$.

VII. REFERENCES

- [1] A. Taflove and S. C. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, 2nd Ed., Boston: Artech House, 2000.
- [2] A. Taflove, *Advances in Computational Electrodynamics: the Finite-Difference Time-Domain Method*, Boston: Artech House, 1998.
- [3] M. N. O. Sadiku, *Numerical Techniques in Electromagnetics*, 2nd ed., Boca Raton: CRC Press, 2001.
- [4] S. Grivet-Talocia, "Generation of Passive Macromodels from Transient Port Responses," *Proc. IEEE 12th Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 287-290, 2003.
- [5] J. E. Bracken, D-K.Sun, Z. J. Cendes, "S-Domain Methods for Simultaneous Time and Frequency Characterization of Electromagnetic Devices," *IEEE Trans. Microwave Theory Tech.*, vol. 46, no. 9, pp. 1277-1290, 1998.
- [6] B. Denecker, F. Olyslager, L. Knockaert, and D. De Zutter, "Generation of FDTD subcell equations by means of reduced order modeling," *IEEE Transactions on Antennas and Propagation*, vol. 51, no.8, pp.1806-1817, 2003.
- [7] P. J. Ashenden, G. D. Peterson, and D. A. Teergarten, *The System Designer's Guide to VHDL-AMS*, San Francisco: Morgan Kaufmann Publishers, 2003.
- [8] D. A. Gorodetsky and P. A. Wilsey, "A signal processing approach to finite-difference time-domain computations," *IEEE International Midwest Symposium on Circuits and Systems*, 2005.
- [9] D. A. Gorodetsky and P. A. Wilsey, "Reduction of FDTD simulation time with modal methods," *Progress in Electromagnetics Research Symposium*, 2006.
- [10] K. S. Yee, "Numerical Solution of Initial Boundary Value Problems Involving Maxwell's Equations in Isotropic Media," *IEEE Trans. Ant. Propagat.*, vol. 14, no. 3, pp. 302-307, 1966.
- [11] A. Taflove, *Computation of the Electromagnetic Fields and Induced Temperatures within a Model of the Microwave-Irradiated Human Eye*, Ph.D. Dissertation, Department of Electrical Engineering, Northwestern University, Evanston, IL, 1975.
- [12] A. Taflove, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, Boston: Artech House, 1995.
- [13] J. P. Berenger, "A perfectly matched layer for the absorption of electromagnetics wave," *J. Computational Physics*, vol. 114, no. 1, pp. 185-200, 1994.
- [14] T. -W. Huang, B. Houshmand, and T. Itoh, "The FDTD Wide-Band Absorbing Boundary Conditions for the Excitation Plane," *IEEE Antennas and Propagation Society International Symposium*, pp. 2-5, 1993.
- [15] W. J. Hoefler, "The discrete time domain green's function or john's matrix – a new powerful concept in transmission line modeling (TLM)," *Int. J. Num. Modeling*, vol. 2, pp. 215-225, 1989.
- [16] Z. Chen and P. P. Silvester, "Analytic solutions for the finite-difference time-domain and transmission-line-matrix methods," *Microwave Opt. Technol. Lett.*, vol. 7, no.1, pp. 5-8, 1994.
- [17] S. I. Grossman, *Elementary Linear Algebra*, Fort Worth: Saunders College Publishing, 1994.
- [18] D. A. Gorodetsky and P. A. Wilsey, "Generation of 3-D finite-difference time-domain macromodels for faster simulation," Electrical and Computer Engineering and Computer Science Department Technical Report, ECECS-TR-2006-1, University of Cincinnati, Cincinnati, Ohio, 2006.