01 Aug 2004

A CN-FDTD Scheme and Its Application to VLSI Interconnects/ Substrate Modeling

Rui Qiang

Dagang Wu

Ji Chen

Chen Wang

et. al. For a complete list of authors, see https://scholarsmine.mst.edu/ele_comeng_facwork/1338

Recommended Citation

R. Qiang et al., "A CN-FDTD Scheme and Its Application to VLSI Interconnects/Substrate Modeling," *Proceedings of the IEEE International Symposium on Electromagnetic Compatibility (2004, Santa Clara, CA)*, vol. 1, pp. 97-101, Institute of Electrical and Electronics Engineers (IEEE), Aug 2004.
The definitive version is available at https://doi.org/10.1109/ISEMC.2004.1350004

A CN-FDTD Scheme and its Application to VLSI Interconnects/Substrate Modeling

Rui Qiang, Dagang Wu, and Ji Chen
Department of Electrical and Computer Engineering
University of Houston
Houston, TX 77204

Chen Wang and Jim Drewniak
Department of Electrical and Computer Engineering
University of Missouri-Rolla
Rolla, MO 65409

Abstract— **In this paper, a two-dimensional (2D) Crank-Nicholason (CN) finite difference time domain (FDTD) method is proposed for VLSI interconnect/substrate characterization. Through rigorous truncation and dispersion error analyses, a guideline on using this technique is presented. Several iterative solvers are investigated to accelerate the solution of the CN-FDTD scheme. Numerical examples are given to demonstrate the accuracy and the efficiency of the proposed algorithm.**

I. INTRODUCTION

The finite-difference time-domain (FDTD) method has been successfully applied to solve many electromagnetic problems [1]. However, this technique becomes very computationally inefficient when it is used to model electrically small objects. This is due to the fact that the time step sizes in FDTD simulations are constrained by the Courant-Friedrich-Levy (CFL) stability condition, which is limited by the minimum discretization cell size. As a consequence, the FDTD method may require over millions of iterations in analyzing some integrated circuit level interconnects structures.

To improve the computational efficiency of the FDTD method, some efforts have been made towards developing time-step size constrain free FDTD schemes. For example, the alternating-direction-implicit (ADI) FDTD method [2-3], which is unconditional stable, has been proposed and applied to solve some practical problems. However, both dispersion and truncation errors of this algorithm are larger than those of the conventional FDTD method [4]. For near field applications, these errors become dominant and therefore, the algorithm cannot be applied to general electrically small structures. An alternative unconditional stable implicit scheme is the Crank-Nicholson (CN) scheme [5], in which the differential operator is not split into the x and y directions as that of the ADI-FDTD method. Using this scheme, both truncation and dispersion errors can be controlled. However, the CN-FDTD method needs to solve a block tri-diagonal matrix during each time stepping and it is generally considered computationally expensive. Before the CN-FDTD method can be applied to solve practical problems, efficient solvers must be explored.

The purpose of this paper is to analyze the truncation error and dispersion error of the CN-FDTD method and develop a guideline for its practical VLSI applications. In addition, the performance among several iterative solvers for the CN-FDTD scheme is investigated. Our results show that the algebraic multi-grid (AMG) solver is the most efficient iterative technique for our 2D CN-FDTD applications. It is observed that this algorithm can be 10 times to 100 times faster than the conventional FDTD method in terms of the CPU time.

II. CN-FDTD METHOD

We first expressed the time-dependent Maxwell's equations in the form of

$$\frac{\partial}{\partial t}\begin{pmatrix}\mathbf{E}(t)\\ \mathbf{H}(t)\end{pmatrix}=\begin{pmatrix}0 & \frac{1}{\varepsilon}\mathbf{R}\\ -\frac{1}{\mu}\mathbf{R} & 0\end{pmatrix}\begin{pmatrix}\mathbf{E}(t)\\ \mathbf{H}(t)\end{pmatrix}, \tag{1}$$

where $\mathbf{E}(t)$ and $\mathbf{H}(t)$ are time variant electric and magnetic fields, and the $\mathbf{R}$ is the curl operator. In Cartesian the coordinate system, the $\mathbf{R}$ can be decomposed into two operators that have only one differential operator in each row or column as:

$$\mathbf{R}=\mathbf{R_1}+\mathbf{R_2}=\begin{pmatrix}0 & 0 & \partial_y\\ \partial_z & 0 & 0\\ 0 & \partial_x & 0\end{pmatrix}+\begin{pmatrix}0 & -\partial_z & 0\\ 0 & 0 & -\partial_x\\ -\partial_y & 0 & 0\end{pmatrix} \tag{2}$$

Defining $\mathbf{A}=\begin{pmatrix}0 & \frac{1}{\varepsilon}\mathbf{R_1}\\ -\frac{1}{\mu}\mathbf{R_2} & 0\end{pmatrix}$ and $\mathbf{B}=\begin{pmatrix}0 & \frac{1}{\varepsilon}\mathbf{R_2}\\ -\frac{1}{\mu}\mathbf{R_1} & 0\end{pmatrix}$,

Eq. (1) can be rewritten into the form of

$$\frac{\partial}{\partial t}\begin{pmatrix}\mathbf{E}(t)\\ \mathbf{H}(t)\end{pmatrix}=(\mathbf{A}+\mathbf{B})\begin{pmatrix}\mathbf{E}(t)\\ \mathbf{H}(t)\end{pmatrix} \tag{3}$$

In the CN-FDTD method proposed here, the time-stepping is given by

$$\begin{pmatrix}\mathbf{E}\\ \mathbf{H}\end{pmatrix}^{n+1}=\frac{\mathrm{I}+\frac{1}{2}\Delta t(\mathrm{A}+\mathrm{B})}{\mathrm{I}-\frac{1}{2}\Delta t(\mathrm{A}+\mathrm{B})}\begin{pmatrix}\mathbf{E}\\ \mathbf{H}\end{pmatrix}^{n}, \tag{4}$$

which is equivalent to

$$\left(\mathrm{I}-\tfrac{1}{2}\Delta t(\mathrm{A}+\mathrm{B})\right)\begin{pmatrix}\mathbf{E}\\ \mathbf{H}\end{pmatrix}^{n+1}=\left(\mathrm{I}+\tfrac{1}{2}\Delta t(\mathrm{A}+\mathrm{B})\right)\begin{pmatrix}\mathbf{E}\\ \mathbf{H}\end{pmatrix}^{n} \tag{5}$$

0-7803-8443-1/04/$20.00 © IEEE.

As we can see from the above equation, the left-hand-side of the above equation is a large sparse matrix associated with the finite difference operator. In order to update the field components from *n*th time step to *(n+1)*th time step, one needs to use either a sparse matrix solver or an iterative solver to seek for the field update.

In the next sections, we will first perform the error analyses of this algorithm and then investigate efficient solvers for the CN-FDTD method.

III. Truncation Error Analysis

Since FDTD method belongs to the class of differential equation techniques, the errors due to the spatial and time discretization need to be considered. In this section, we first consider the truncation error of this algorithm. For simplicity, we will perform the truncation error analysis for a 2D transverse magnetic (TM) case.

For a 2D TM case, the Maxwell equations for 2D TM case can be written as:

$$\frac{\partial}{\partial t}\begin{pmatrix} Hx \\ Hy \\ Ez \end{pmatrix} = \begin{pmatrix} 0 & 0 & \partial_y \\ 0 & 0 & -\partial_x \\ -\partial_y & \partial_x & 0 \end{pmatrix}\begin{pmatrix} Hx \\ Hy \\ Ez \end{pmatrix} \tag{6}$$

Following the standard truncation error analysis and omitting the higher order terms [6], we can obtain the second-order truncation error in approximating the temporal derivative for the conventional FDTD, the ADI-FDTD and the CN-FDTD by

$$\begin{pmatrix} \Delta\partial_t \mathbf{E}_z \\ \Delta\partial_t \mathbf{H}_x \\ \Delta\partial_t \mathbf{H}_y \end{pmatrix} = \begin{pmatrix} \frac{1}{24}\Delta t^2\partial_t^3 & \frac{1}{24\varepsilon}\Delta y^2\partial_y^3 & \frac{-1}{24\varepsilon}\Delta x^2\partial_x^3 \\ \frac{1}{24\mu}\Delta y^2\partial_y^3 & \frac{1}{24}\Delta t^2\partial_t^3 & 0 \\ \frac{-1}{24\mu}\Delta x^2\partial_x^3 & 0 & \frac{1}{24}\Delta t^2\partial_t^3 \end{pmatrix}\begin{pmatrix} \mathbf{E}_z \\ \mathbf{H}_x \\ \mathbf{H}_y \end{pmatrix}, \tag{7}$$

$$\begin{pmatrix} \Delta\partial_t \mathbf{E}_z \\ \Delta\partial_t \mathbf{H}_x \\ \Delta\partial_t \mathbf{H}_y \end{pmatrix} = \begin{pmatrix} \frac{-1}{12}\Delta t^2\partial_t^3 & \frac{1}{24\varepsilon}\Delta y^2\partial_y^3 & \frac{-1}{24\varepsilon}\Delta x^2\partial_x^3 \\ \frac{1}{24\mu}\Delta y^2\partial_y^3 & \frac{-1}{12}\Delta t^2\partial_t^3 & 0 \\ \frac{-1}{24\mu}\Delta x^2\partial_x^3 & \frac{-1}{4\mu\varepsilon}\Delta t^2\partial_{xyt}^3 & \frac{-1}{12}\Delta t^2\partial_t^3 \end{pmatrix}\begin{pmatrix} \mathbf{E}_z \\ \mathbf{H}_x \\ \mathbf{H}_y \end{pmatrix}, \tag{8}$$

and

$$\begin{pmatrix} \Delta\partial_t \mathbf{E}_z \\ \Delta\partial_t \mathbf{H}_x \\ \Delta\partial_t \mathbf{H}_y \end{pmatrix} = \begin{pmatrix} \frac{-1}{12}\Delta t^2\partial_t^3 & \frac{1}{24\varepsilon}\Delta y^2\partial_y^3 & \frac{-1}{24\varepsilon}\Delta x^2\partial_x^3 \\ \frac{1}{24\mu}\Delta y^2\partial_y^3 & \frac{-1}{12}\Delta t^2\partial_t^3 & 0 \\ \frac{-1}{24\mu}\Delta x^2\partial_x^3 & 0 & \frac{-1}{12}\Delta t^2\partial_t^3 \end{pmatrix}\begin{pmatrix} \mathbf{E}_z \\ \mathbf{H}_x \\ \mathbf{H}_y \end{pmatrix}. \tag{9}$$

As we can see from the analyses above, these second-order errors can come from either temporal or spatial. The CN-FDTD method and the conventional FDTD method have similar truncation error. However, the ADI-FDTD method has an additional error term of $\frac{-1}{4\mu\varepsilon}\Delta t^2\partial_{xyt}^3$. If the Δt is on the same order of the time step size that is constrained by the CFL condition, the truncation errors of the conventional FDTD, the ADI-FDTD, and the CN-FDTD method should be on the same order. However, if a larger time step size is used, the error associated with the term, $\frac{-1}{4\mu\varepsilon}\Delta t^2\partial_{xyt}^3$ in the ADI-FDTD method will become the dominant factor. Therefore, for practical applications, the time step size of the ADI-FDTD method cannot be very large. It must be on the same order of the CFL time step size in order to maintain the truncation error of the conventional FDTD method. In Figure 1, we show the comparison of the ADI-FDTD relative truncation error due to the cross-coupling term, $\frac{-1}{4\mu\varepsilon}\Delta t^2\partial_{xyt}^3$ and the term, $\frac{-1}{12}\Delta t^2\partial_t^3$ for a simple dipole excitation. As we can clearly see from the figure, the truncation error due to the alternating procedure is significantly larger than that from the conventional FDTD central difference approximation.

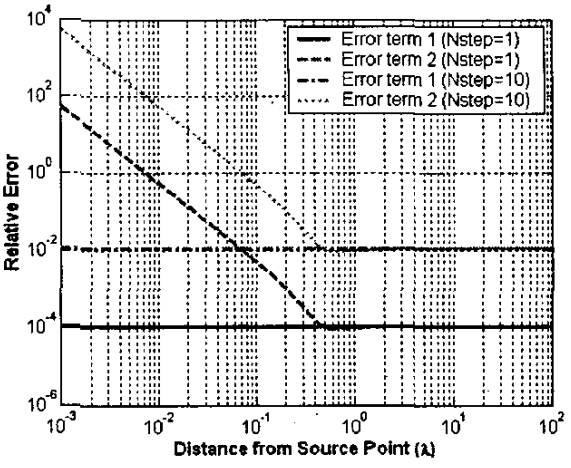

Figure 1. Truncation error due to $\frac{-1}{12}\Delta t^2\partial_t^3$ (term 1) and $\frac{-1}{4\mu\varepsilon}\Delta t^2\partial_{xyt}^3$ (term 2) in the ADI-FDTD method.

In addition to perform the theoretical analysis, we also perform a numerical experiment. In this numerical experiment, we calculate the field distribution as a function of distance from a current source. In the simulations, the sampling rate is at 100 point per wavelength. Figure 2 show the normalized electric field from both the ADI-FDTD simulation and the CN-FDTD simulation and compared with the analytical solution. As we can clearly see from the figure, the CN-FDTD method agrees very well with the analytical even in the near field region where there is a large variation in the field values. However, the ADI-FDTD method cannot provide accurate solutions when the sampling point is close to the excitation source. This observation confirms the truncation error analysis as described in the Fig. 1. Therefore, the ADI-FDTD cannot directly be applied to general electromagnetic field near field simulations.

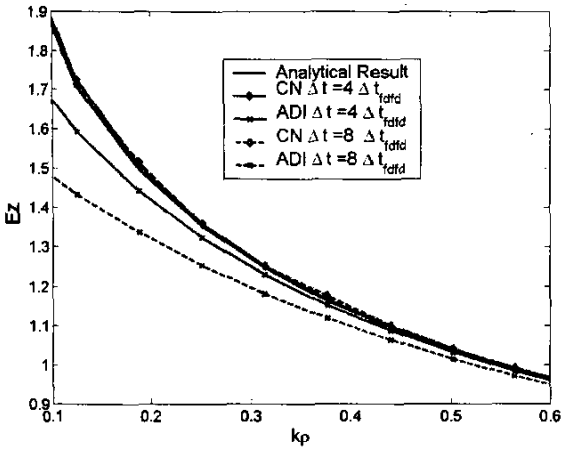

Figure 2. Comparison between the CN-FDTD method and ADI-FDTD for a near field electromagnetic simulation.

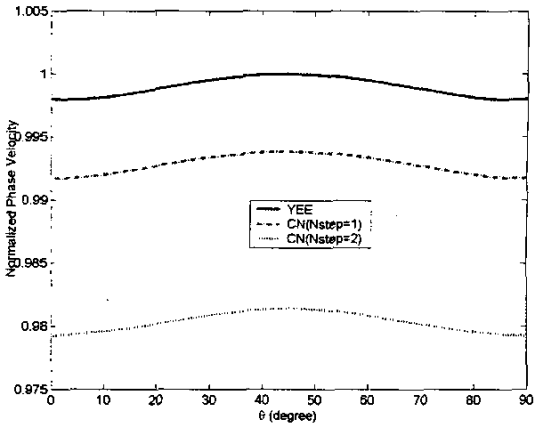

Figure 3. Dispersion error comparison between the CN-FDTD method and the conventional FDTD method.

IV. Dispersion Error Analysis

The second kind of error that needs to be addressed in the general finite difference method is the dispersion error. The dispersion error of these implicit FDTD algorithms can be performed by the Neumann analysis [6] . For the conventional FDTD method and the CN-FDTD method, the dispersion relations are given by

$$\frac{\sin^2(\omega\Delta t/2)}{(c\Delta t)^2}=\frac{\sin^2(k_x\Delta x/2)}{\Delta x^2}+\frac{\sin^2(k_y\Delta y/2)}{\Delta y^2}, \qquad (10)$$

and

$$\frac{\tan^2(\omega\Delta t/2)}{(c\Delta t)^2}=\frac{\sin^2(k_x\Delta x/2)}{\Delta x^2}+\frac{\sin^2(k_y\Delta y/2)}{\Delta y^2}. \qquad (11)$$

In Fig. 3, the dispersion error of the conventional FDTD method (Yee method) and the CN-FDTD method are shown as a function of different propagation angle at a sampling rate of 10 points per wavelength. As we can see from the figures, the conventional FDTD method has lower dispersion error than that of the CN-FDTD method. It is also noticed that as the time step increases, the dispersion error of the CN-FDTD method also increases.

For practical simulations, in order to maintain the dispersion error for CN-FDTD method within a certain limit, the time step size one cannot be very large. In Fig. 3, we show the maximum allowable time step size one can use as a function of sampling rate for different dispersion error criterion. For example, for a system with a sampling rate of 200 point per wavelength, in order to keep the dispersion error within 0.5 % level (on the same order of that of the conventional FDTD method), the maximum allowable one can use should only be 12 times larger than that of the conventional FDTD method.

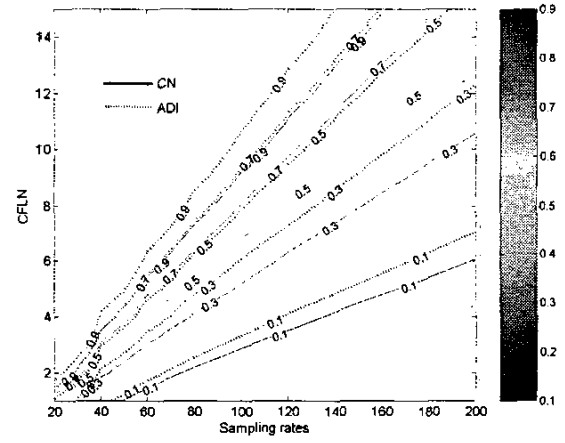

Figure 4. Maximum time step size one as a function of sampling rate.

V. Iterative Solvers

An attractive feature of the ADI-FDTD method is its low computational overhead since only simple tri-diagonal matrix needs to be solved in the time stepping. For the CN-FDTD method, the update equation can be written in the form of $AX^{n+1}=BX^n+J$, where A and B are matrices related the CN scheme and J is the external excitation. The matrix A has a broader bandwith and is often referred as 'tri-diagonal matrix with fringes' or 'block tri-diagonal matrix' [5]. To solve this matrix equation, the Conjugate Gradient Method (including BICG, BICGStab and CGS), the Generalized Minimal Residual (GMRES) method, and the Quasi_Minimal Residual (QMR) method can be used [7]. To further reduce

the overall CPU time, some pre-conditioners have also been developed here to reduce the number of iterations.

For our 2D test problem, the zero-filled in incomplete LU factorization (ILU0) preconditioner in conjuncture with the bi-conjugate gradient stabilized (BICGStab) method has been applied to solve this matrix problem. It is found that the number of iterations to solve the matrix equation is very small for lossy media applications. However, the iterative number of this method grows rapidly if larger time step sizes are used.

In addition to investigate traditional iterative solvers such as CG and BiCG methods, multigrid methods, which are generally accepted as the fastest numerical methods for the solution of elliptic partial differential equations, have also been applied to our problems [8]. It is particularly successful when this technique is used for low-frequency applications. Since the matrix structure in the 2D CN-FDTD method is similar to the discretized version of Poisson's equation, multigrid methods has great potentials to work well for this application. Here, the algebraic multigrid method (AMG), one of the robust multigrid solution methods, is incorporated into the 2D CN-FDTD for low frequency simulation. In Figure 5, we show the number of iterations for several iterative techniques when they are applied to solve a low-loss electromagnetic problem. It is found that the AMG method has the least number of iterations.

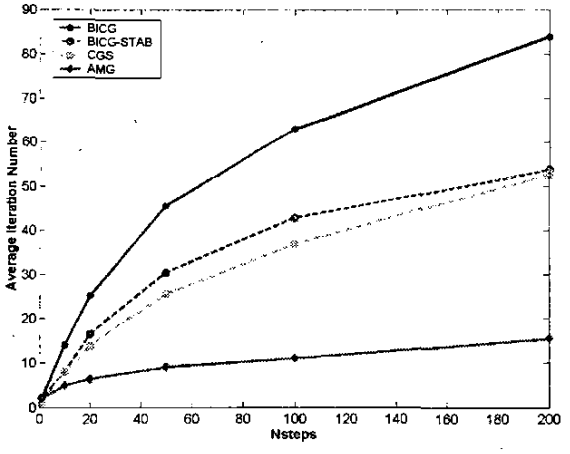

Figure 5. Iterative numbers comparison among different iterative solvers.

VI. NUMERICAL EXAMPLES

To demonstrate the efficiency of AMG method in 2D CN-FDTD scheme, two numerical experiments are given.

The first example considered here is the calculation of the skin depth for a metal strip. As shown in Fig.6, a line source is used to excite the electromagnetic fields. In order to calculate the skin effects, a spatial resolution of $dx = dy = 0.1um$ is used. The total simulation domain is $25um \times 15um$ and the $15um \times 10um$ metal strip (with conductivity of $3.0 \times 10^7 [1/\Omega]$) resides at the bottom of the simulation region. The excitation source is $5um$ above the metal. For the conventional FDTD method, the maximum time step size can be used is 2.948×10^{-12} s. The skin depths at two frequencies, 1GHz and 10GHz, are investigated. Using the CN-FDTD method, the time step size can be much larger than the conventional FDTD method. In particular, for 10GHz simulation, the time step size used in the simulation is 2500 times larger than that of the conventional FDTD method and for 1GHz simulation, the time step size is 25000 times larger than that of the conventional FDTD method. In Table 1, the results show the potential feature of the AMG method. As we can see from the table, CPU time is reduced by 91.97 % and 99.08% for 10 GHz and 1GHz cases, respectively. The calculated skin depths at three different frequencies are compared with the analytical solution. As we can see from Table 2, these results agree well with each other.

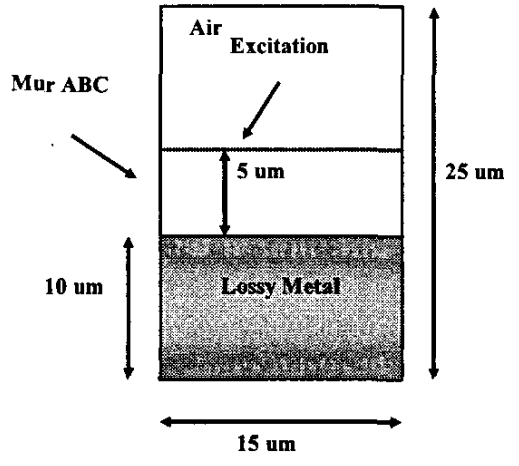

Figure 6. Simulation setup for skin depth calculation.

Table 1. CPU time comparison ($Nstep = \Delta t_{CN} / \Delta t_{FDTD}$)

	10GHz	1GHz
Nsteps (FDTD)	1	1
Nsteps (CN FDTD)	2500	25000
CPU Time (FDTD)	1148.70	11549.25
CPU Time (CN FDTD)	92.28	106.52
CPU Time ratio (CNFDTD/FDTD)	0.0803	0.0092
dx=dy=0.1um		

Table 2 Theoretical and Simulated skin depth (unit: m)

Skin Depth	1 GHz	5 GHz	10 GHz
Theoretical	2.91E-6	1.299E-6	9.188E-7
Simulated	3E-6	1.3E-6	9.0E-7

The second example considered here is the cross-talk between two traces on a substrate. The geometry of this simulation is shown in Fig. 7. Three traces, with a height of 1.9 *um* and a width of 5 *um*, reside on a silicon substrate that has a relative permittivity of 11.8 and conductivity of 10 S/m. The spacing between traces is also 5 *um*.

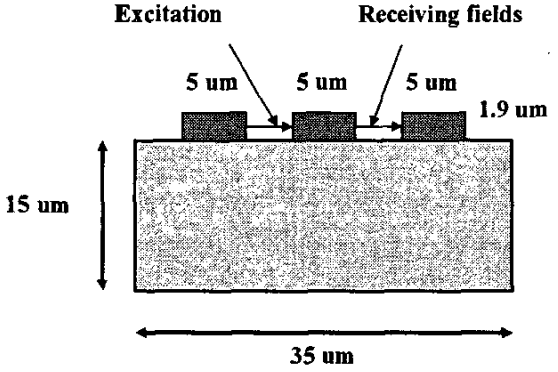

Figure 7. Geometry for the cross-talk simulation.

The excitation voltage between the two left conductors is one voltage and the source frequency is at 10 GHz. From our truncation error and dispersion error analysis, a time step size that is 8733 times larger than that of the conventional FDTD method can be used. The calculated electric field distribution (in logarithm scale) by the CN-FDTD method is shown in Fig. 8. This result agrees well with the result obtained form the conventional FDTD simulation. However, it should be noted that for such a simulation, the conventional FDTD method requires 8.5e4 seconds while the CN-FDTD method only requires 2.6e3 seconds. The CN-FDTD method has reduced the CPU time by a factor of 30. Our experience shows that if one simulates the same structure at lower frequency region, the CPU time for the CN-FDTD method will remain almost unchanged while the CPU time for the conventional FDTD method will be increased proportionally to the inverse of the excitation frequency.

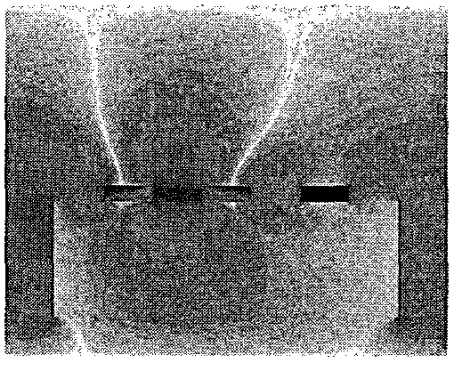

Figure 8. Field distribution obtained using the CN-FDTD simulation.

VII. Conclusions

A numerical technique is proposed for VLSI interconnects/substrate simulations. This technique is based on the application of the AMG technique to improve the computational efficiency of the unconditionally stable CN-FDTD method. Through rigorous truncation and dispersion error analysis, a guideline on using this technique is presented. Numerical simulations demonstrate the efficiency and the accuracy of this proposed algorithm.

References

[1] A. Taflove, ed., "Advances in Computational Electrodynamics: The Finite-Difference Time-Domain Method", Norwood, MA: Artech House, 1998.

[2] T. Namiki, "A new FDTD algorithm based on alternating-direction implicit method," *IEEE Trans. Microwave Theory Tech.*, vol. 47, pp. 2003–2007, Oct. 1999.

[3] F. Zhen, Z. Chen, and J. Zhang, "Toward the development of a three dimensional unconditionally stable finite-difference time-domain method', *IEEE Trans. Microw. Theory Tech.*, 2000, 48, pp 1550-1558

[4] G. Sun and C. W. Trueman, "Unconditionally stable Crank-Nicolson scheme for solving two-dimensional Maxwell's equations", *IEE Electronics Letters*, vol. 30, no. 7, pp595-597, 2003

[5] B. Fornberg, "Some numerical techniques for Maxwell's equations in different types of geometries", In *Topics in Computational Wave Propagation*, Lecture notes in *Computational Science and Engineering* 31, Springer Verlag (2003), 265-299.

[6] G. D. Smith, Numerical solution of partial differential equations: Finite Difference Method, Second Edition, Clarendon Press. Oxford, 1978

[7] *Iterative Methods for Sparse Linear Systems*, PWS Publishing, New York, 1996.

[8] U. Trottenberg, C. W. Oosterlee, A. Schuller, "Multrigrid" (ACDEMIC PRESS, San Diego, CA 2001)