An Iterative ADI-FDTD With Reduced Splitting Error

Shumin Wang, Fernando L. Teixeira, and Ji Chen

***Abstract*—We present a new iterative alternating-direction-implicit finite-difference time-domain (ADI-FDTD) method. By recognizing the ADI-FDTD method as a special case of a more general iterative approach to solve the Crank–Nicolson (CN) FDTD scheme, the splitting error in ADI-FDTD can be reduced systematically. Numerical examples are used to illustrate the improved accuracy of this method.**

***Index Terms*—Alternating-direction-implicit finite-difference time-domain (ADI-FDTD) method, Crank–Nicolson (CN) FDTD scheme.**

I. Introduction

The alternating-direction-implicit finite-difference time-domain (ADI-FDTD) method is an unconditionally stable method suitable for simulation of wave propagation and scattering problems involving fine geometries [1]–[4]. Despite being an implicit method, ADI-FDTD produces only a small computational overhead due to its tridiagonal character. Although the time step size in ADI-FDTD simulations is no longer bounded by the Courant–Friedreich–Levy (CFL) criterion, the method exhibits a splitting error associated with the square of the time step size [2]. Being of second-order accuracy asymptotically, the splitting error becomes dominant in regions with larger spatial derivatives. This can be detrimental for modeling problems where strong near field coupling occurs and/or structures containing field singularities such as in tips and corners.

The splitting error associated with the Douglas–Gunn ADI scheme for parabolic equations has been tackled with in [5]. In this letter, we shall present a new approach to reduce the splitting error associated with the Peaceman–Rachford ADI scheme for Maxwell's equations (hyperbolic equation). This scheme retains the main properties of the ADI method and can also be viewed as an iterative solver of the Crank–Nicolson FDTD (CN-FDTD) scheme [2]–[4].

II. Methodology

A. ADI-FDTD and CN-FDTD

The ADI-FDTD method can be viewed as an approximate factorization of the CN-FDTD scheme [2]–[4]. We write three-dimensional (3-D) Maxwell's equations as

$$\frac{\partial \vec{u}}{\partial t} = [A]\vec{u} + [B]\vec{u} \tag{1}$$

where $\vec{u} = [E_x, E_y, E_z, H_x, H_y, H_z]^T$ and

$$[A] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \frac{\partial}{\epsilon\partial y} \\ 0 & 0 & 0 & \frac{\partial}{\epsilon\partial z} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial}{\epsilon\partial x} & 0 \\ 0 & \frac{\partial}{\mu\partial z} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial}{\mu\partial x} & 0 & 0 & 0 \\ \frac{\partial}{\mu\partial y} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[B] = \begin{bmatrix} 0 & 0 & 0 & 0 & -\frac{\partial}{\epsilon\partial z} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{\partial}{\epsilon\partial x} \\ 0 & 0 & 0 & -\frac{\partial}{\epsilon\partial y} & 0 & 0 \\ 0 & 0 & -\frac{\partial}{\mu\partial y} & 0 & 0 & 0 \\ -\frac{\partial}{\mu\partial z} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\partial}{\mu\partial x} & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Applying the CN scheme at time step $n + (1/2)$, we get

$$\left([I] - \frac{\Delta t}{2}[A] - \frac{\Delta t}{2}[B]\right)\vec{u}^{n+1} = \left([I] + \frac{\Delta t}{2}[A] + \frac{\Delta t}{2}[B]\right)\vec{u}^{n}. \tag{2}$$

The left-hand side of (2) is a sparse matrix. Solving this linear system could be time-consuming for large problems. If we instead rewrite (2) as

$$\left([I] - \frac{\Delta t}{2}[A]\right)\left([I] - \frac{\Delta t}{2}[B]\right)\vec{u}^{n+1} = \left([I] + \frac{\Delta t}{2}[A]\right) \times \left([I] + \frac{\Delta t}{2}[B]\right)\vec{u}^{n} + \frac{\Delta t^2}{4}[A][B](\vec{u}^{n+1} - \vec{u}^{n}) \tag{3}$$

and ignore the last term in (3), we have

$$\left([I] - \frac{\Delta t}{2}[A]\right)\left([I] - \frac{\Delta t}{2}[B]\right)\vec{u}^{n+1} = \left([I] + \frac{\Delta t}{2}[A]\right)\left([I] + \frac{\Delta t}{2}[B]\right)\vec{u}^{n}. \tag{4}$$

The above can be exactly solved in two steps, i.e.,

$$\left([I] - \frac{\Delta t}{2}[A]\right)\vec{u}^{tmp} = \left([I] + \frac{\Delta t}{2}[B]\right)\vec{u}^{n} \tag{5}$$

in the first step and

$$\left([I] - \frac{\Delta t}{2}[B]\right)\vec{u}^{n+1} = \left([I] + \frac{\Delta t}{2}[A]\right)\vec{u}^{tmp} \tag{6}$$

Manuscript received April 19, 2004; revised August 23, 2004. This work was supported in part by the National Science Foundation under Grants BES-0332957 and ECS-0347502. The review of this letter was arranged by Associate Editor J.-G. Ma.

S. Wang and J. Chen are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204 USA (e-mail: james.wang@ieee.org).

F. L. Teixeira is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43212 USA.

Digital Object Identifier 10.1109/LMWC.2004.842835

1531-1309/$20.00 © 2005 IEEE

in the second step, where $\vec{u}^{tmp}$ denotes an intermediate solution. In each step, a tridiagonal linear system is produced [1].

Because the ADI-FDTD solves (4) instead of (3), it introduces a splitting error, or an additional truncation error term, of the form

$$\frac{\Delta t^2}{4}[A][B](\vec{u}^{n+1} - \vec{u}^n) \quad (7)$$

to the solution. The effect of this splitting error depends on three factors: the time step size (Δt^2 factor), the spatial derivatives of the field ($[A][B]$ factor), and the temporal variation of the field ($\vec{u}^{n+1} - \vec{u}^n$ factor). When field variation and/or the time step size is large, the splitting error becomes more pronounced, as observed in [2].

B. Reduction of the Splitting Error

We write

$$[M] = \left([I] - \frac{\Delta t}{2}[A]\right)\left([I] - \frac{\Delta t}{2}[B]\right) \quad (8)$$

$$[N] = \frac{\Delta t^2}{4}[A][B] \quad (9)$$

$$\vec{b} = \left([I] + \frac{\Delta t}{2}[A] + \frac{\Delta t}{2}[B]\right)\vec{u}^n \quad (10)$$

so that (3) becomes

$$[M]\vec{u}^{n+1} = [N]\vec{u}^{n+1} + \vec{b}. \quad (11)$$

Since $[M] - [N]$ is the left-hand side of (2), (11) is exactly the same as the original CN scheme. Moreover, (11) can be recognized as a splitting scheme [6] for solving the following linear system iteratively:

$$([M] - [N])\vec{u}^{n+1} = \vec{b}. \quad (12)$$

To highlight the iterative nature, we rewrite (11) as

$$[M]\vec{u}^{n+1}_{k+1} = [N]\vec{u}^{n+1}_k + \vec{b} \quad (13)$$

where the subscript k denotes the solution of the kth iteration. If the initial guess of $\vec{u}^{n+1}$, i.e., $\vec{u}^{n+1}_0$, is set to be the previous time step value $\vec{u}^n$, we exactly recover (4) from (13). In other words, the solution of (4) is the first iterative solution of (13) by using the previous time step value as the initial guess. By choosing a different initial guess $\vec{u}^{n+1}_0$ and/or a few ADI-FDTD iterations for the solution $\vec{u}^{n+1}_{k+1}$, the splitting error can be controlled.

The implementation of this iterative scheme is straightforward. For the $(k+1)$th iteration of (13), we have

$$\left([I] - \frac{\Delta t}{2}[A]\right)\left([I] - \frac{\Delta t}{2}[B]\right)\vec{u}^{n+1}_{k+1} = \left([I] + \frac{\Delta t}{2}[A]\right) \times \left([I] + \frac{\Delta t}{2}[B]\right)\vec{u}^n + \frac{\Delta t^2}{4}[A][B]\left(\vec{u}^{n+1}_k - \vec{u}^n\right). \quad (14)$$

The above can be exactly solved in two steps using the Peaceman–Rachford approach [4], i.e.,

$$\left([I] - \frac{\Delta t}{2}[A]\right)\vec{u}^{tmp}_{k+1} = \left([I] + \frac{\Delta t}{2}[B]\right)\vec{u}^n + \frac{\Delta t^2}{8}[A][B]\left(\vec{u}^{n+1}_k - \vec{u}^n\right) \quad (15)$$

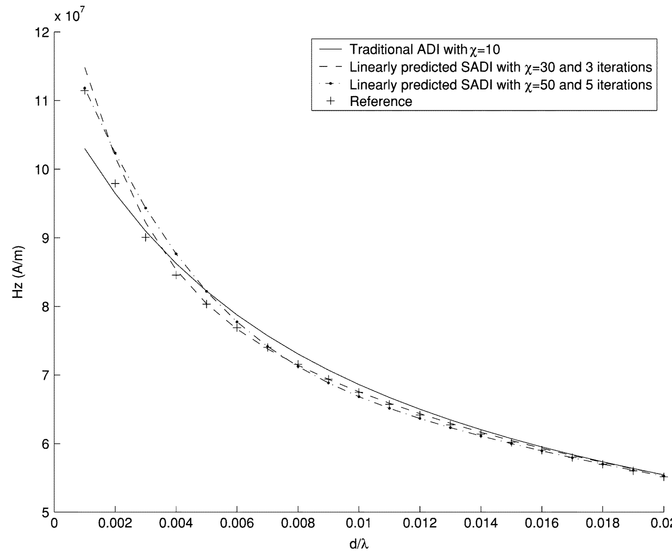

Fig. 1. Computed H_z field from various schemes with the same number of implicit updates.

in the first step and

$$\left([I] - \frac{\Delta t}{2}[B]\right)\vec{u}^{n+1}_{k+1} = \left([I] + \frac{\Delta t}{2}[A]\right)\vec{u}^{tmp} + \frac{\Delta t^2}{8}[A][B]\left(\vec{u}^{n+1}_k - \vec{u}^n\right) \quad (16)$$

in the second step, where $\vec{u}^{tmp}$ again denotes an intermediate solution. The performance of this iterative scheme can be further improved by using the Seidel method, which employs the the most recently updated values in the current iteration [6]. We denote this scheme as SADI. This will be demonstrated in the following.

III. Numerical Examples

To investigate the accuracy of the method, we simulate the field distribution of an infinitely long magnetic current source [two-dimensional (2-D) TE_z wave] in free space and compare the results against the analytical solution. The comparisons are done in the near field, where the spatial distribution exhibits larger variation and hence the splitting errors is more pronounced. A first-order differentiated Gaussian pulse is used as time-domain source excitation. The center frequency is 1 MHz and the spatial discretization size corresponds to 1000 cells per wavelength at this frequency. We perform on-the-fly discrete Fourier transforms to obtain H_z field magnitude at each observation point. A reference solution can be obtained analytically, from the second-kind zeroth-order Hankel function. In all figures, d denotes the distance from the source to the observation point. The CFL number is defined as $\chi = \sqrt{2}c\Delta t/h$, where h is the cell size of an equal-distance mesh.

Fig. 1 shows the H_z field distributions from different schemes using the same number of implicit updates. On a Pentium IV machine, it takes 397 s, 379 s, and 366 s, respectively, to execute the simulations with $\chi = 10$, $\chi = 30$, and $\chi = 50$. The slightly improved efficiency for larger CFL numbers can be mainly attributed to a more adequate initial guess for the next iteration. With a similar computational cost, the iterative scheme produces

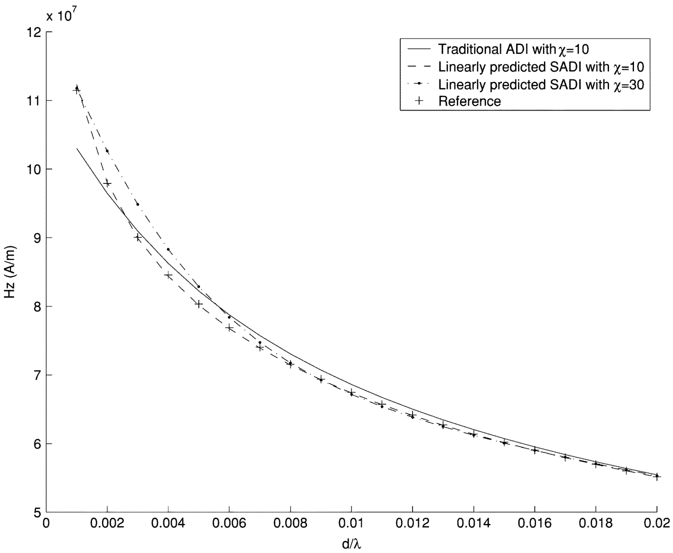

Fig. 2. Computed H_z field of ADI-FDTD and SADI with different CFL numbers.

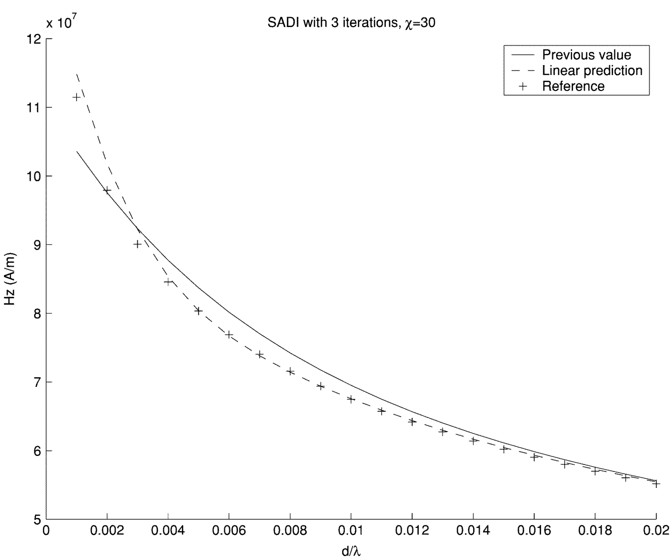

Fig. 3. Computed H_z field distribution of SADI with different initial guesses.

more accurate solutions than the standard ADI-FDTD. Fig. 2 compares the accuracy of linearly predicted SADI with one iteration against standard ADI-FDTD.

We next illustrate the effect of using different initial guesses. SADI is executed with $\chi = 30$ and three iterations per update. Fig. 3 compares the reference solution against results using the previous time step value as initial guess, and using linear extrapolation of the previous two time step values as initial guess. Linear prediction yields better results than simply using the previous time step values.

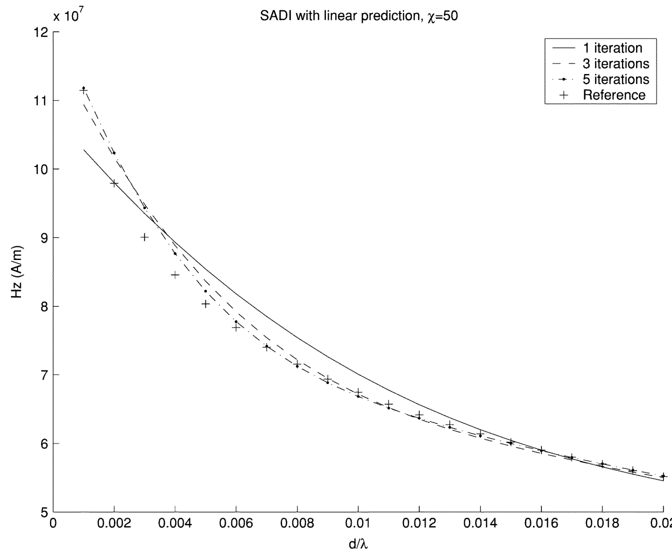

Fig. 4. Computed H_z field distribution of SADI with different number of iterations per update.

Similar to the traditional ADI-FDTD method, the effectiveness of this scheme deteriorates progressively for larger CFL numbers. On the other hand, increasing the number of iterations per update yields better results. This is confirmed in Fig. 4.

IV. Conclusion

We have presented an iterative scheme to reduce the splitting error in the ADI-FDTD method. This scheme is based on recognizing ADI-FDTD as the first iteration for solving CN-FDTD with a general splitting method. The effectiveness of this scheme was illustrated by numerical examples. Future work includes the development of acceleration approaches for the iterative scheme.

References

[1] T. Namiki, "A new FDTD algorithm based on alternating-direction implicit method," *IEEE Trans. Microwave Theory Tech.*, vol. 47, no. 10, pp. 2003–2007, Oct. 1999.

[2] S. G. García, T.-W. Lee, and S. C. Hagness, "On the accuracy of the ADI-FDTD method," *IEEE Antennas Wireless Propag. Lett.*, vol. 1, no. 1, pp. 31–34, Jan. 2002.

[3] G. Zhao and Q. H. Liu, "The unconditionally stable pseudospectral time-domain (PSTD) method," *IEEE Microw. Wireless Compon. Lett.*, vol. 13, no. 11, pp. 475–477, Nov. 2003.

[4] J. W. Thomas, *Numerical Partial Differential Equations: Finite Difference Methods*. Berlin, Germany: Springer-Verlag, 1995.

[5] J. Douglas Jr. and S. Kim, "Improved accuracy for locally one-dimensional methods for parabolic equations," *Math. Models Methods Appl. Sci.*, vol. 11, no. 9, pp. 1563–1579, 2001.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: John Hopkins Univ. Press, 1996.