Efficient algorithm for the unconditionally stable 3-D ADI-FDTD method

Tan, Eng Leong

2007

Tan, E. L. (2007). Efficient algorithm for the unconditionally stable 3-D ADI-FDTD method.
IEEE Microwave and Wireless Components Letters, 17(1), 7-9.
doi:10.1109/LMWC.2006.887239

<https://hdl.handle.net/10356/138245>

<https://doi.org/10.1109/LMWC.2006.887239>

© 2007 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at:
<https://doi.org/10.1109/LMWC.2006.887239>

Downloaded on 26 Apr 2023 18:32:34 SGT

Efficient algorithm for the unconditionally stable 3-D ADI-FDTD method

Eng Leong Tan, *Senior Member, IEEE*

Abstract—This letter presents a new efficient algorithm for the unconditionally stable three-dimensional alternating direction implicit finite-difference time-domain (3-D ADI-FDTD) method. The algorithm involves updating equations whose right-hand sides are much simpler and more concise than those in the conventional implementation. This leads to substantial reduction in the number of arithmetic operations required for their computations. To take full advantage of the efficiency gain, efforts are made as well to maintain or reduce other computation costs in for-looping and solving tridiagonal systems. Certain field array reuse is also exploited which reduces the memory space and the memory indexing overhead.

Index Terms—ADI-FDTD method, FDTD method, unconditionally stable method, computational electromagnetics.

I. INTRODUCTION

There has been considerable interest in the numerical solutions of electromagnetic wave problems based on finite-difference time-domain (FDTD) methods [1]. One of the celebrated FDTD methods is that developed by Yee [2], which is an explicit but conditionally stable scheme with time step size constrained by Courant-Friedrich-Levy (CFL) condition. To remove the CFL constraint, the unconditionally stable FDTD method based on the alternating direction implicit (ADI) technique has been developed [3], [4]. This method has been demonstrated to be useful for problems with very fine meshes relative to wavelength. However, from the implementation point of view, the ADI-FDTD algorithm described in [3], [4] has been rather complicated. This is because not only are there tridiagonal systems that need to be dealt with, the right-hand sides of their updating equations also contain numerous terms that call for considerable arithmetic operations. To improve the computation efficiency, many researchers have proposed alternative unconditionally stable methods such as those based on semi-implicit approximation, locally one-dimensional and other Crank-Nicolson schemes [5]–[7]. Still these methods differ from the ADI-FDTD method of [3], [4], which continues to receive much popularity in recent years.

In this letter, a new algorithm for implementing the ADI-FDTD method is presented. The algorithm involves updating equations whose right-hand sides are much simpler and more concise than those in the conventional implementation [3], [4]. This leads to substantial reduction in the number of arithmetic operations required for their computations. To take full advantage of the efficiency gain, other computation costs in for-looping and solving tridiagonal systems are maintained

or reduced. Certain field array reuse is also exploited which reduces the memory space and the memory indexing overhead. For generality and completeness, we shall consider the algorithm for three-dimensional (3-D) case in the following.

II. EFFICIENT 3-D ADI-FDTD ALGORITHM

Like the conventional 3-D ADI-FDTD implementation [3], [4], the new efficient algorithm involves two procedures for advancement of time step from n to $n + \frac{1}{2}$ and from $n + \frac{1}{2}$ to $n + 1$. Henceforth, we shall denote the intermediate fields at $n + \frac{1}{2}$ and the final fields at $n + 1$ using capital letters superscripted with ‘*’, E^* and H^* ($E^*|^{n+1}$ and $H^*|^{n+1}$ are directly related to the physical fields as to be clarified later.) In addition, some auxiliary field variables denoted by small letters e and h are introduced as well. Exploitation of these variables leads to the implementation as follows:

A. First procedure from n to $n + \frac{1}{2}$

(i) Auxiliary updating for $e_x|^{n+1/2}, h_z|^{n+1/2}$:

$$e_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} = E_x^*|_{i+\frac{1}{2},j,k}^n - e_x|_{i+\frac{1}{2},j,k}^{n-\frac{1}{2}} - \frac{\Delta t}{2\epsilon} J_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} \quad (1)$$

$$h_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+\frac{1}{2}} = 2H_z^*|_{i+\frac{1}{2},j+\frac{1}{2},k}^n - h_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^{n-\frac{1}{2}} \quad (2)$$

(ii) Implicit updating for $E_x^*|^{n+1/2}$:

$$\begin{aligned} & -\frac{\Delta t^2}{8\mu\epsilon\Delta y^2} E_x^*|_{i+\frac{1}{2},j-1,k}^{n+\frac{1}{2}} + \left(\frac{1}{2} + \frac{\Delta t^2}{4\mu\epsilon\Delta y^2}\right) E_x^*|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} \\ & - \frac{\Delta t^2}{8\mu\epsilon\Delta y^2} E_x^*|_{i+\frac{1}{2},j+1,k}^{n+\frac{1}{2}} \\ & = e_x|_{i+\frac{1}{2},j,k}^n + \frac{\Delta t}{2\epsilon\Delta y} (h_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+\frac{1}{2}} - h_z|_{i+\frac{1}{2},j-\frac{1}{2},k}^{n+\frac{1}{2}}). \end{aligned} \quad (3)$$

(iii) Explicit updating for $H_z^*|^{n+1/2}$:

$$\begin{aligned} H_z^*|_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+\frac{1}{2}} & = h_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^n \\ & + \frac{\Delta t}{4\mu\Delta y} (E_x^*|_{i+\frac{1}{2},j+1,k}^{n+\frac{1}{2}} - E_x^*|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}}). \end{aligned} \quad (4)$$

The updating equations for other field components can be written down simply by permuting the indices of (1)–(4).

B. Second procedure from $n + \frac{1}{2}$ to $n + 1$

(i) Auxiliary updating for $e_x|^{n+1/2}, h_y|^{n+1/2}$:

$$e_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} = E_x^*|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} - e_x|_{i+\frac{1}{2},j,k}^n - \frac{\Delta t}{2\epsilon} J_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} \quad (5)$$

The author is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: eeltan@ntu.edu.sg).

$$h_y|_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+\frac{1}{2}} = 2H_y^*|_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+\frac{1}{2}} - h_y|_{i+\frac{1}{2},j,k+\frac{1}{2}}^n. \quad (6)$$

(ii) Implicit updating for $E_x|^{n+1}$:

$$\begin{aligned} & -\frac{\Delta t^2}{8\mu\epsilon\Delta z^2}E_x^*|_{i+\frac{1}{2},j,k-1}^{n+1} + \left(\frac{1}{2} + \frac{\Delta t^2}{4\mu\epsilon\Delta z^2}\right)E_x^*|_{i+\frac{1}{2},j,k}^{n+1} \\ & - \frac{\Delta t^2}{8\mu\epsilon\Delta z^2}E_x^*|_{i+\frac{1}{2},j,k+1}^{n+1} \\ & = e_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} - \frac{\Delta t}{2\epsilon\Delta z}(h_y|_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+\frac{1}{2}} - h_y|_{i+\frac{1}{2},j,k-\frac{1}{2}}^{n+\frac{1}{2}}). \end{aligned} \quad (7)$$

(iii) Explicit updating for $H_y|^{n+1}$:

$$\begin{aligned} H_y^*|_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1} & = h_y|_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+\frac{1}{2}} \\ & - \frac{\Delta t}{4\mu\Delta z}(E_x^*|_{i+\frac{1}{2},j,k+1}^{n+1} - E_x^*|_{i+\frac{1}{2},j,k}^{n+1}). \end{aligned} \quad (8)$$

The updating equations for other field components can likewise be written down by permuting the indices of (5)-(8).

III. DISCUSSION AND COMPARISON

It is interesting to find that the algorithm above actually corresponds to the ADI-FDTD method of [3], [4] (with identical stability and numerical characteristics), even though it takes much simpler and more concise form compared to the conventional one. To clarify this point further, let us recall the updating equations of the conventional implementation. For instance, the implicit updating in the first procedure reads

$$\begin{aligned} & -\frac{\Delta t^2}{4\mu\epsilon\Delta y^2}E_x|_{i+\frac{1}{2},j-1,k}^{n+\frac{1}{2}} + \left(1 + \frac{\Delta t^2}{2\mu\epsilon\Delta y^2}\right)E_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} \\ & - \frac{\Delta t^2}{4\mu\epsilon\Delta y^2}E_x|_{i+\frac{1}{2},j+1,k}^{n+\frac{1}{2}} \\ & = E_x|_{i+\frac{1}{2},j,k}^n + \frac{\Delta t}{2\epsilon\Delta y}(H_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^n - H_z|_{i+\frac{1}{2},j-\frac{1}{2},k}^n) \\ & - \frac{\Delta t}{2\epsilon\Delta z}(H_y|_{i+\frac{1}{2},j,k+\frac{1}{2}}^n - H_y|_{i+\frac{1}{2},j,k-\frac{1}{2}}^n) \\ & - \frac{\Delta t^2}{4\mu\epsilon\Delta y\Delta x}[(E_y|_{i+1,j+\frac{1}{2},k}^n - E_y|_{i,j+\frac{1}{2},k}^n) \\ & - (E_y|_{i+1,j-\frac{1}{2},k}^n - E_y|_{i,j-\frac{1}{2},k}^n)] - \frac{\Delta t}{2\epsilon}J_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}}; \end{aligned} \quad (9)$$

while the explicit updating is

$$\begin{aligned} H_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+\frac{1}{2}} & = H_z|_{i+\frac{1}{2},j+\frac{1}{2},k}^n \\ & + \frac{\Delta t}{2\mu\Delta y}(E_x|_{i+\frac{1}{2},j+1,k}^{n+\frac{1}{2}} - E_x|_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}}) \\ & - \frac{\Delta t}{2\mu\Delta x}(E_y|_{i+1,j+\frac{1}{2},k}^n - E_y|_{i,j+\frac{1}{2},k}^n). \end{aligned} \quad (10)$$

Notice that while the coefficients in the left-hand sides of (9) and (3) are essentially the same to within a constant ($\frac{1}{2}$), the right-hand sides have been considerably simplified in the latter. Similar arguments apply to other updating equations, which may be referred to [1], [3], [4]. These equations, as well as those presented in the previous section, can be readily implemented on a computer, in which it should be easier for programming the latter. One can then validate the equivalence

Fig. 1. Time-domain electric fields recorded at the center of cavity based on Yee-FDTD and ADI-FDTD. Both conventional and present algorithms for ADI-FDTD have been found to yield the same results.

of both conventional and present algorithms by noting the relationships between their field variables as

$$E_\xi|^{n+1} = \frac{1}{2}E_\xi^*|^{n+1} \quad (11)$$

$$H_\xi|^{n+1} = H_\xi^*|^{n+1} \quad (12)$$

where $\xi = x, y, z$.

In all our numerical simulations and testings (assuming PEC walls and zero initial condition), we have been able to achieve excellent agreement of (11)-(12) for various time steps (Courant numbers), spatial discretizations, source excitations and material parameters. As a simple illustration, Fig. 1 plots the time-domain E_z recorded at the center of an air-filled cavity meshed with $50 \times 30 \times 9$ uniform grid cells of size 2 mm each. The cavity is excited by a current source J_z with Gaussian pulse waveform $[\exp(-(t-t_d)^2/\tau^2)]$, $\tau = 150$ ps, $t_d = 450$ ps] flowing along z -axis from the bottom to the top passing through the cavity center. The simulations have been carried out using Yee's FDTD method with $\Delta t = \Delta t_{\text{CFL}}$ (Courant limit time step) and ADI-FDTD method with various $\text{CFLN} = \Delta t/\Delta t_{\text{CFL}}$. For the latter, both conventional and present algorithms have been found to yield the same results. Notice that since the observation point lies inside the source region, there is (same) relatively large error that is proportional to the time step size during the strong source excitation. This finding is similar to that observed in the near field study of [8].

Having verified the new algorithm, it will be desirable to provide detailed assessment and comparison particularly with regard to its computation efficiency gain over the conventional implementation. To that end, let us acquire the floating point operations (flops) count taking into account the number of multiplications/divisions (M/D) and additions/subtractions (A/S) required for one complete time step. Table I lists the flops count for both conventional [3], [4] and present algorithms, based on the right-hand sides of their respective

TABLE I
FLOPS COUNT FOR 3-D ADI-FDTD ALGORITHMS

Algorithm		[3], [4]	This letter
Auxiliary	M/D	0	6
	A/S	0	12
Implicit	M/D	18	6
	A/S	48	12
Explicit	M/D	12	6
	A/S	24	12
Total	M/D	30	18
	A/S	72	36

updating equations such as (1)-(8) and (9)-(10) etc. For simplicity, the source terms have been excluded and the number of electric and magnetic field components in all directions have been taken to be the same. From the table, it is clear that with the aid of auxiliary field updating, the total flops count (M/D+A/S) has been reduced substantially from 102 to 54 in the present implementation. This corresponds to an efficiency gain of 1.89 in flops count reduction for the right-hand sides of updating equations. Moreover, in performing the auxiliary updating (e.g. (1)-(2)), the six multiplications are merely simple doublings while the subtractions may be carried out essentially in-place (see below). Note that the flops count in Table I assumes that all multiplicative factors have been precomputed and stored. When the multiplications/divisions are performed on the fly, the efficiency gain should be even higher using the present algorithm.

To take full advantage of the preceding efficiency gain attained for the right-hand sides of updating equations, every effort should be made as well to (at least) maintain or reduce other computation costs besides the arithmetic operations. In particular, one should also consider the for-looping overheads incurred in most programming languages [7]. To avoid introducing additional for-loops in the implementation of the algorithm in Section II, the auxiliary updating may be incorporated into the same loops of implicit and explicit updates through careful arrangement of the codes. Meanwhile, it is imperative to reduce the high cost of solving tridiagonal systems. While this is by no means an easy task, the cost may be reduced somewhat by exploiting the precomputations and storing not the tridiagonal matrix elements but their (same number of) bidiagonally factorized elements. This then leads to approximately $5N$ flops (saving some cost of factorization) for solving a tridiagonal system of order N during iterations. Taking these flops into account, the present algorithm still achieves a gain of 1.57 in flops count reduction over the conventional one.

Finally some remarks pertaining to memory space requirement are in order. At first sight, it appears that there are at least four field arrays needed for $[e_\xi; h_\xi]^{n+\frac{1}{2}}$, $[E_\xi^*; H_\xi^*]^{n+\frac{1}{2}}$, $[e_\xi; h_\xi]^{n+1}$ and $[E_\xi^*; H_\xi^*]^{n+1}$ in the algorithm of Section II; while only two (or one and a half) are needed for $[E_\xi; H_\xi]^{n+\frac{1}{2}}$ and $[E_\xi; H_\xi]^{n+1}$ in the conventional approach. A closer scrutiny of the algorithm should reveal the reusability of spaces for two of the field arrays, e.g.

$$E_{x|i+\frac{1}{2},j,k}^{*n} \leftarrow E_{x|i+\frac{1}{2},j,k}^{*n} - E_{x|i+\frac{1}{2},j,k}^{*n-\frac{1}{2}} - \frac{\Delta t}{2\epsilon} J_{x|i+\frac{1}{2},j,k}^{n+\frac{1}{2}} \quad (13)$$

$$H_{z|i+\frac{1}{2},j+\frac{1}{2},k}^{*n} \leftarrow 2H_{z|i+\frac{1}{2},j+\frac{1}{2},k}^{*n} - H_{z|i+\frac{1}{2},j+\frac{1}{2},k}^{*n-\frac{1}{2}}. \quad (14)$$

Apart from demanding less memory, exploiting such field array reuse also reduces the memory indexing overhead through in-place arithmetic operations.

IV. CONCLUSION

This letter has presented a new efficient algorithm for the unconditionally stable 3-D ADI-FDTD method. The algorithm involves updating equations whose right-hand sides are much simpler and more concise than those in the conventional implementation. This leads to substantial reduction in the number of arithmetic operations required for their computations. To take full advantage of the efficiency gain, efforts have been taken as well to maintain or reduce other computation costs in for-looping and solving tridiagonal systems. Certain field array reuse has also been exploited which reduces the memory space and the memory indexing overhead. With its simplicity, conciseness and efficiency advantages, the present algorithm should find much usefulness in subsequent developments and applications of ADI-FDTD method [9].

REFERENCES

- [1] A. Taflov and S. C. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method* (Boston, M. A.: Artech House, 2000).
- [2] K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," *IEEE Trans. Antennas Propagat.*, vol. 14, pp. 302-307, 1966.
- [3] F. Zheng, Z. Chen and J. Zhang, "Toward the Development of a Three-Dimensional Unconditionally Stable Finite-Difference Time-Domain Method," *IEEE Trans. Microwave Theory Tech.*, vol. 48, pp. 1550-1558, 2000.
- [4] T. Namiki, "3-D ADI-FDTD Method – Unconditionally Stable Time-Domain Algorithm for Solving Full Vector Maxwell's Equations," *IEEE Trans. Microwave Theory Tech.*, vol. 48, pp. 1743-1748, 2000.
- [5] A. P. Zhao, "More accurate and efficient unconditionally stable FDTD method" *Electron. Lett.*, vol. 38, pp. 862-864, 2002.
- [6] J. Shibayama, M. Muraki, J. Yamauchi and H. Nakano, "Efficient Implicit FDTD algorithm based on locally one-dimensional scheme," *Electron. Lett.*, vol. 41, no. 19, Sept. 2005.
- [7] G. Sun and C. W. Trueman, "Efficient Implementations of the Crank-Nicolson Scheme for the Finite-Difference Time-Domain Method," *IEEE Trans. Microwave Theory Tech.*, vol. 54, pp. 2275-2284, 2006.
- [8] S. Wang, "On the Current Source Implementation for the ADI-FDTD Method," *IEEE Microw. Wireless Comp. Lett.*, vol. 14, pp. 513-515, 2004.
- [9] W. Fu and E. L. Tan, "A Parameter Optimized ADI-FDTD Method Based on the (2,4) Stencil," *IEEE Trans. Antennas Propagat.*, vol. 54, pp. 1836- 1842, 2006.