The Stability and Numerical Dispersion Study of the ADI-SFDTD Algorithm

Zongxin Liu(Corresponding author)
Engineering Institute of Engineering corps, PLA University of Science and Technology
Nanjing 21007, China
&
The XXX PLAAF Rear of Testing and Training
Jiuquan Gansu province 735018,China
E-mail: liuzongxin19840717@163.com

Yiwang Chen
Ministry of Science Research, PLA University of Science and Technology
Nanjing 21007, China
Tel:86-25-8082-0501 E-mail: chenyiwang2010@163.com

Xin Xu
Engineering Institute of Engineering corps, PLA University of Science and Technology
Nanjing 21007, China
E-mail: xxinn_815@sina.com

Xuegang Sun
The XXX PLAAF Rear of Testing and Training
Jiuquan Gansu province 735018,China
E-mail: sunxuegang@163.com

Received: August 4, 2011 Accepted: October 23, 2011 Published: December 1, 2011
doi:10.5539/mas.v5n6p157 URL: http://dx.doi.org/10.5539/mas.v5n6p157

Abstract

In this letter, the alternating-direction-implicit (ADI) technique is applied to Symplectic finite-difference time-domain (SFDTD) method, the curl operator is endued with two different styles when doing computation from the $(s-1)$ th progression to s th progression. It holds the advantages of both ADI-FDTD and SFDTD, not only eliminating the restriction of the Courant-Friedrich-Levy (CFL), but also holding the inner characteristics of Maxwell's equations. The analytical accuracy and efficiency of the proposed method is verified good.

Keywords: Alternating-direction-implicit (ADI), Symplectic finite-difference time-domain (SFDTD), Growth matrix, Numerical dispersion

1. Introduction

FDTD method is a very useful numerical simulation technique for solving electromagnetic questions. As we know, the traditional FDTD method is based on the explicit finite-difference algorithm, hence, it is limited by Courant-Friedrich-Levy (CFL) stability condition. In order to eliminate the Courant–Friedrich–Levy (CFL) condition restraint, Unconditionally sTable algorithm ADL-FDTD(the alternating-direction-implicit technique finite difference time domain) has been proposed. But in the common ADI-FDTD method, the choice of large time intervals leads to substantial dispersion errors that degrade its performance (A. P. Zhao, 2000; F. Zheng et al., 2000; Huang Z X et al., 2007).

Effective studies (M Kusaf et al., 2005; Ruth F D, 1983) revealed that Maxwell's equations can be viewed as an infinite dimension Hamilton system. FDTD and ADI-FDTD destroy Maxwell's equations' Symplectic structure, so they are not good algorithms for Maxwell's equations' numerical simulation. A good algorithm must hold Maxwell's equations' Symplectic structure.

In this paper a novel algorithm that bases on SFDTD (symplectic finite difference time domain) and ADI has been proposed. We transform Maxwell's equations to Hamilton's equations, and use symplectic propagation technique disperse Hamilton's equations in time domain, and use the ADI technique to discredited Hamilton's equations' curl operator R in spatial domain, then, we discuss the ADI-SFDTD algorithm's stability and numerical dispersion systemically, finally we validate the proposed ADI-SFDTD formulation by a numerical example.

2. ADI-SFDTD method

2.1 Hamilton transform of Maxwell's equations

In a linear, homogeneous, and isotropic medium, Maxwell's equations can be written as (J. W. Thomas, 1995):

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E} \tag{1a}$$

$$\frac{\partial \mathbf{D}}{\partial t} = \nabla \times \mathbf{H} - \mathbf{J} \tag{1b}$$

$$\mathbf{D} = \varepsilon \mathbf{E}, \mathbf{B} = \mu \mathbf{H} \tag{1c}$$

Where, ε is medium's permittivity and μ is medium's permeability. In the Hamilton system, Maxwell's equations can be written as

$$\begin{pmatrix} \dfrac{\partial \mathbf{B}}{\partial t} \\ \dfrac{\partial \mathbf{D}}{\partial t} \end{pmatrix} = \begin{pmatrix} \mathbf{0}_3 & -\mathbf{I}_3 \\ \mathbf{I}_3 & \mathbf{0}_3 \end{pmatrix} \begin{pmatrix} \dfrac{\delta H(\mathbf{B}, \mathbf{D})}{\delta \mathbf{B}} \\ \dfrac{\delta H(\mathbf{B}, \mathbf{D})}{\delta \mathbf{D}} \end{pmatrix} \tag{2}$$

Where, Hamilton function $H(\mathbf{B}, \mathbf{D})$ is defined as

$$H(\mathbf{B}, \mathbf{D}) = \frac{1}{2}\left(\frac{1}{\mu}\mathbf{B} \cdot \nabla \times \mathbf{B} + \frac{1}{\varepsilon}\mathbf{D} \cdot \nabla \times \mathbf{D}\right) - \mathbf{J} \cdot \mathbf{B} \tag{3}$$

The $\dfrac{\delta}{\delta \mathbf{B}}$ is defined as

$$\frac{\delta H(\mathbf{B}, \mathbf{D})}{\delta \mathbf{B}} = \left(\frac{\delta H}{\delta B_x}, \frac{\delta H}{\delta B_y}, \frac{\delta H}{\delta B_z} \right)^{\mathrm{T}} \tag{4}$$

Where $\dfrac{\delta H}{\delta B_\theta} (\theta = x, y, z)$ is defined as

$$\left[\frac{d}{d\rho} H(B_\theta + \rho\varphi) \right] \Big|_{\rho=0} = \int \frac{\delta H}{\delta B_\theta} \varphi dV \tag{5}$$

In the equation (5), φ is the inspection function.

2.2 ADI-SFDTD method

Studies revealed that Hamilton's equations can be transformed into (6), written as

$$\frac{\partial}{\partial t}\begin{pmatrix} \mathbf{B} \\ \mathbf{D} \end{pmatrix} = \begin{pmatrix} \mathbf{0}_3 & -\varepsilon^{-1}\mathbf{R} \\ \mu^{-1}\mathbf{R} & \mathbf{0}_3 \end{pmatrix} \begin{pmatrix} \mathbf{B} \\ \mathbf{D} \end{pmatrix} \equiv (\mathbf{L}_A + \mathbf{L}_B) \begin{pmatrix} \mathbf{B} \\ \mathbf{D} \end{pmatrix} \tag{6}$$

Where, $L_A = \begin{pmatrix} \mathbf{0}_3 & -\varepsilon^{-1}\mathbf{R} \\ \mathbf{0}_3 & \mathbf{0}_3 \end{pmatrix}, L_B = \begin{pmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mu^{-1}\mathbf{R} & \mathbf{0}_3 \end{pmatrix}, \mathbf{R} = \begin{pmatrix} 0 & -\frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial x} \\ -\frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{pmatrix}$

In time domain, from time $t = 0$ to time $t = \tau$, the results of (6) can be written as

$$\begin{pmatrix} \mathbf{B} \\ \mathbf{D} \end{pmatrix}(\tau) = \exp(\tau(\mathbf{L}_A + \mathbf{L}_B))\begin{pmatrix} \mathbf{B} \\ \mathbf{D} \end{pmatrix}(0) \tag{7}$$

For exponential operator can't be used to compute, the exponential operator is approximate to (8) by using symplectic propagation technique.

$$\exp[\tau(\mathbf{L}_A + \mathbf{L}_B)] = \prod_{s=1}^{m} \exp(d_s \tau \mathbf{L}_B)\exp(c_s \tau \mathbf{L}_A) + O(\tau^{p+1}) \tag{8}$$

Where, $m, p(m \geq p)$ are symplectic propagation's progression and order. According to (J. W. Thomas, 1995), choose the suiTable propagation sub coefficients $\{c_s\}$ and $\{d_s\}$, it can preserve Maxwell's equations' inner characters. In this paper, we use the optimized 5 progression and 4 order propagation sub coefficients.

For $(\mathbf{L}_u)^{\ell} = 0, (u = A, B, \ell = 2, 3, \cdots)$, the exponential operators $\exp(\tau\mathbf{L}_A)$ and $\exp(\tau\mathbf{L}_B)$ can be explicitly expressed as

$$\exp(\tau\mathbf{L}_A) = \mathbf{I}_6 + \tau\mathbf{L}_A \tag{9a}$$

$$\exp(\tau\mathbf{L}_B) = \mathbf{I}_6 + \tau\mathbf{L}_B \tag{9b}$$

There is curl operator $\mathbf{R}$ in Factors $\mathbf{L}_A$ and $\mathbf{L}_B$, so in order to get the numerical results of Maxwell's equations, we must discredited the equations in spatial domain again.

Introducing the plane wave' propagation equation, written as:

$$f(x, y, z, t) = f_0 \exp(-\tilde{j}(i\Delta x k_x + j\Delta y k_y + z\Delta z k_z - wn\Delta t)) \tag{10}$$

In the spherical coordinate system: $k_x = k_0 \sin\theta\cos\phi$, $k_y = k_0 \sin\theta\sin\phi$, $k_z = k_0 \cos\theta$.

The positive direction of ϕ is that of the right-handed rotation from x to y about z axis, the positive direction of θ is from the positive z axis towards the negative z axis, $\tilde{j} = \sqrt{-1}$ and k_0 is the numerical wave number.

$f_{(i,j,k)}^{n+\frac{s}{p}} = f(i\Delta x, j\Delta y, k\Delta z; (n + \tau_s)\Delta t)$ indicates the s th progression approximate solution of function $f's$ closed-form solution at discretization point $(i\Delta x, j\Delta y, k\Delta z)$ in the n th time step. There, Every time step need m progression to simulate and the time increment of the s th progression to $s+1$ th progression is $\tau_s \Delta t$.

Applying the ADI principle into $\mathbf{R}$, we define two different curl operators about $\mathbf{R}$, marked as $\mathbf{R}_1$ and $\mathbf{R}_2$ in following. In $\mathbf{R}_\nu (\nu = 1, 2)$, the I indicates implicit form and E indicates explicit form.

$$\mathbf{R}_1 = \begin{pmatrix} 0|_E & -\frac{\partial}{\partial z}|_I & \frac{\partial}{\partial y}|_E \\ \frac{\partial}{\partial z}|_I & 0|_E & -\frac{\partial}{\partial x}|_I \\ -\frac{\partial}{\partial y}|_E & \frac{\partial}{\partial x}|_I & 0|_E \end{pmatrix} \tag{11a}$$

$$\mathbf{R}_2 = \begin{pmatrix} 0|_I & -\frac{\partial}{\partial z}|_E & \frac{\partial}{\partial y}|_I \\ \frac{\partial}{\partial z}|_E & 0|_I & -\frac{\partial}{\partial x}|_E \\ -\frac{\partial}{\partial y}|_I & \frac{\partial}{\partial x}|_E & 0|_I \end{pmatrix} \tag{11b}$$

At s th progression of n th time step, in x-direction, the implicit form is defined in equation (12a) and the explicit form is defined in (12b).

$$R_{I,i} \cdot f_{i,j,k}^{n+\frac{s}{p}} \equiv \frac{f_{i+\frac{1}{2},j,k}^{n+(s+\frac{1}{2})/p} - f_{i-\frac{1}{2},j,k}^{n+(s+\frac{1}{2})/p}}{\Delta x} \approx \partial_x f_{i,j,k}^{n+\frac{s}{p}} + O_I(\Delta x^2) \tag{12a}$$

$$R_{E,i} \cdot f_{i,j,k}^{n+\frac{s}{p}} \equiv \frac{f_{i+\frac{1}{2},j,k}^{n+s/p} - f_{i-\frac{1}{2},j,k}^{n+s/p}}{\Delta x} \approx \partial_x f_{i,j,k}^{n+\frac{s}{p}} + O_E(\Delta x^2) \tag{12b}$$

In the same way, y-direction's implicit form is $R_{I,j} \cdot f_{i,j,k}^{n+\frac{s}{p}}$ and explicit form is $R_{E,j} \cdot f_{i,j,k}^{n+\frac{s}{p}}$, z-direction's implicit form is $R_{I,k} \cdot f_{i,j,k}^{n+\frac{s}{p}}$ and explicit form is $R_{E,k} \cdot f_{i,j,k}^{n+\frac{s}{p}}$. Substituting (10) into (12a) and (12b), we can get

$$R_{E,i} \cdot f_{i,j,k}^{n+\frac{s}{p}} \equiv \frac{f_0 \exp\{-\tilde{j}[(i+\frac{1}{2})\Delta x k_x + j\Delta y k_y + k\Delta z k_z - w(n+\frac{s}{p})\tau_s]\} - f_0 \exp\{-\tilde{j}[(i-\frac{1}{2})\Delta x k_x + j\Delta y k_y + k\Delta z k_z - w(n+\frac{s}{p})\tau_s]\}}{\Delta x} = \eta_x f \tag{13a}$$

$$R_{I,i} \cdot f_{i,j,k}^{n+\frac{s}{p}} \equiv \frac{f_{i+\frac{1}{2},j,k}^{n+(s+\frac{1}{2})/p} - f_{i-\frac{1}{2},j,k}^{n+(s+\frac{1}{2})/p}}{\Delta x} = \exp(\tilde{j}\frac{w}{2p})\frac{f_{i+\frac{1}{2},j,k}^{n+s/p} - f_{i-\frac{1}{2},j,k}^{n+s/p}}{\Delta x} = \exp(\tilde{j}\frac{w}{2p})R_{E,i} \cdot f_{i,j,k}^{n+\frac{s}{p}} = hR_{E,i} \cdot f_{i,j,k}^{n+\frac{s}{p}} \tag{13b}$$

Where, $h = \exp(\tilde{j}\frac{w}{2p})$,

$$\eta_x = \frac{\exp(\tilde{j}\frac{\Delta x k_x}{2}) - \exp(-\tilde{j}\frac{\Delta x k_x}{2})}{\Delta x} = \tilde{j}\frac{\sin(k_x \Delta x/2)}{\Delta x/2} \tag{14a}$$

In the same way we can get

$$\eta_y = \frac{\exp(\tilde{j}\frac{\Delta y k_y}{2}) - \exp(-\tilde{j}\frac{\Delta y k_y}{2})}{\Delta y} = \tilde{j}\frac{\sin(k_y \Delta y/2)}{\Delta y/2} \tag{14b}$$

$$\eta_z = \frac{\exp(\tilde{j}\frac{\Delta z k_z}{2}) - \exp(-\tilde{j}\frac{\Delta z k_z}{2})}{\Delta z} = \tilde{j}\frac{\sin(k_z \Delta z/2)}{\Delta z/2} \tag{14c}$$

Substituting (13a) and (13b) into (11a) and (11b), we get

$$\mathbf{R}_1 + \mathbf{R}_2 = (h+1)\mathbf{R} \tag{15}$$

Equation (7) can be written as

$$
\begin{aligned}
&\begin{pmatrix}\mathbf{B}\\ \mathbf{D}\end{pmatrix}(\tau)\\
&=\exp[\tau(\mathbf{L}_A+\mathbf{L}_B)]\begin{pmatrix}\mathbf{B}\\ \mathbf{D}\end{pmatrix}(0)\\
&=\exp\{\frac{\tau}{(h+1)}[(\mathbf{L}_{A1}+\mathbf{L}_{A2})+(\mathbf{L}_{B1}+\mathbf{L}_{B2})]\}\begin{pmatrix}\mathbf{B}\\ \mathbf{D}\end{pmatrix}(0)\\
&=\exp[\frac{\tau}{(h+1)}(\mathbf{L}_{A2}+\mathbf{L}_{B2})]\exp[\frac{\tau}{(h+1)}(\mathbf{L}_{A1}+\mathbf{L}_{B1})]\begin{pmatrix}\mathbf{B}\\ \mathbf{D}\end{pmatrix}(0)\\
&=\exp[\frac{2}{(h+1)}\frac{\tau}{2}(\mathbf{L}_{A2}+\mathbf{L}_{B2})]\exp[\frac{2}{(h+1)}\frac{\tau}{2}(\mathbf{L}_{A1}+\mathbf{L}_{B1})]\begin{pmatrix}\mathbf{B}\\ \mathbf{D}\end{pmatrix}(0)\\
&=\exp[\frac{2}{(h+1)}\frac{\tau}{2}(\mathbf{L}_{A2}+\mathbf{L}_{B2})]\begin{pmatrix}\mathbf{B}\\ \mathbf{D}\end{pmatrix}(\frac{\tau}{2})
\end{aligned}
\tag{16}
$$

So the computation from the $(s-1)$ th progression to s th progression is broken up into two sub-steps: the first step $s-1\rightarrow s-\dfrac{1}{2}$ and the second step $s-\dfrac{1}{2}\rightarrow s$.

For D_x component, we take

$$D^{n+(s-1/2)/p}_{x,(i+\frac{1}{2},j,k)}=D^{n+(s-1)/p}_{x,(i+\frac{1}{2},j,k)}+\frac{\tau_s d_s}{4\mu(h+1)}\times\left[R_{I,j}\cdot B^{n+(s-1)/p}_{z,(i+\frac{1}{2},j,k)}-R_{E,k}\cdot B^{n+(s-1)/p}_{y,(i+\frac{1}{2},j,k)}\right] \tag{17a}$$

as the fist step and

$$D^{n+s/p}_{x,(i+\frac{1}{2},j,k)}=D^{n+(s-\frac{1}{2})/p}_{x,(i+\frac{1}{2},j,k)}+\frac{\tau_s d_s}{4\mu(h+1)}\times\left[R_{E,j}\cdot B^{n+(s-\frac{1}{2})/p}_{z,(i+\frac{1}{2},j,k)}-R_{I,k}\cdot B^{n+(s-\frac{1}{2})/p}_{y,(i+\frac{1}{2},j,k)}\right] \tag{17b}$$

as the second step.

For B_x we can get

$$B^{n+(s-\frac{1}{2})/p}_{x,(i,j+\frac{1}{2},k+\frac{1}{2})}=B^{n+(s-1)/p}_{x,(i,j+\frac{1}{2},k+\frac{1}{2})}+\frac{\tau_s c_s}{4\varepsilon(h+1)}\times\left[R_{I,k}\cdot D^{n+(s-1)/p}_{y,(i,j+\frac{1}{2},k+\frac{1}{2})}-R_{E,j}\cdot D^{n+(s-1)/p}_{z,(i,j+\frac{1}{2},k+\frac{1}{2})}\right] \tag{18a}$$

as the fist step and

$$B^{n+s/p}_{x,(i,j+\frac{1}{2},k+\frac{1}{2})}=B^{n+(s-\frac{1}{2})/p}_{x,(i,j+\frac{1}{2},k+\frac{1}{2})}+\frac{\tau_s c_s}{4\varepsilon(h+1)}\times\left[R_{E,k}\cdot D^{n+(s-\frac{1}{2})/p}_{y,(i,j+\frac{1}{2},k+\frac{1}{2})}-R_{I,j}\cdot D^{n+(s-\frac{1}{2})/p}_{z,(i,j+\frac{1}{2},k+\frac{1}{2})}\right] \tag{18b}$$

As the second step, where $T^d_\theta=\dfrac{\tau_s d_s}{4\mu(h+1)}$, $T^b_\theta=\dfrac{\tau_s c_s}{4\varepsilon(h+1)}$ $(\theta=(x,y,z))$ .For other components, the equations can be obtained in the same way.

In (17a) and (17b), both sides contain the unknown field components on the right hand side, so the iterative calculation can't be done directly. Substituting (13b) into (17a) and (17b), we can get (19a) and (19b).

$$D^{n+(s-1/2)/p}_{x,(i+\frac{1}{2},j,k)}=D^{n+(s-1)/p}_{x,(i+\frac{1}{2},j,k)}+T^d_x\left[R_{I,j}\cdot B^{n+(s-1)/p}_{z,(i+\frac{1}{2},j,k)}-R_{E,k}\cdot B^{n+(s-1)/p}_{y,(i+\frac{1}{2},j,k)}\right]=D^{n+(s-1)/p}_{x,(i+\frac{1}{2},j,k)}+T^d_x\left[hR_{E,j}\cdot B^{n+(s-1)/p}_{z,(i+\frac{1}{2},j,k)}-R_{E,k}\cdot B^{n+(s-1)/p}_{y,(i+\frac{1}{2},j,k)}\right] \tag{19a}$$

$$D^{n+s/p}_{x,(i+\frac{1}{2},j,k)}=D^{n+(s-\frac{1}{2})/p}_{x,(i+\frac{1}{2},j,k)}+T^d_x\left[R_{E,j}\cdot B^{n+(s-\frac{1}{2})/p}_{z,(i+\frac{1}{2},j,k)}-R_{I,k}\cdot B^{n+(s-\frac{1}{2})/p}_{y,(i+\frac{1}{2},j,k)}\right]=D^{n+(s-\frac{1}{2})/p}_{x,(i+\frac{1}{2},j,k)}+T^d_x\left[R_{E,j}\cdot B^{n+(s-\frac{1}{2})/p}_{z,(i+\frac{1}{2},j,k)}-hR_{E,k}\cdot B^{n+(s-\frac{1}{2})/p}_{y,(i+\frac{1}{2},j,k)}\right] \tag{19b}$$

3. Analysis of stability and numerical dispersion

3.1 Stability analysis

According to (16), growth matrix $\mathbf{G}$ can be presented as the product of the first procedure growth matrix $\mathbf{G}_1$ and the second procedure $\mathbf{G}_2$, written as:

$$\mathbf{G}=\mathbf{G}_2\mathbf{G}_1 \tag{20a}$$

$$\mathbf{G}_1=\exp[\frac{\tau}{h+1}(\mathbf{L}_{A1}+\mathbf{L}_{B1})]=\prod_{s=1}^{m}\exp(\frac{d_{1s}\tau}{h+1}\mathbf{L}_{B1})\exp(\frac{c_{1s}\tau}{h+1}\mathbf{L}_{A1}) \tag{20b}$$

$$\mathbf{G}_2 = \exp[\frac{\tau}{h+1}(\mathbf{L}_{A2}+\mathbf{L}_{B2})] = \prod_{s=1}^{m}\exp(\frac{d_{2s}\tau}{h+1}\mathbf{L}_{B2})\exp(\frac{c_{2s}\tau}{h+1}\mathbf{L}_{A2}) \tag{20c}$$

ξ indicates the growth factor of the total procedure, ξ_1 indicates growth factor of the first procedure and ξ_2 indicates growth factor of the second procedure. According to the principle of the matrix growth, we obtain that ξ_1 satisfies equation (21a) and ξ_2 satisfies equation (21b):

$$\xi_1^2 - tr(\mathbf{G}_1)\xi_1 + 1 = 0 \tag{21a}$$

$$\xi_2^2 - tr(\mathbf{G}_2)\xi_2 + 1 = 0 \tag{21b}$$

Where,

$$tr(\mathbf{G}_1) = 2 + \sum_{s=1}^{m} g_s[\frac{\tau^2(h^2\eta_x^2 + \eta_y^2 + h^2\eta_z^2)}{\varepsilon\mu(h+1)^2}]^s \tag{22a}$$

$$tr(\mathbf{G}_2) = 2 + \sum_{s=1}^{m} g_s[\frac{\tau^2(\eta_x^2 + h^2\eta_y^2 + \eta_z^2)}{\varepsilon\mu(h+1)^2}]^s \tag{22b}$$

$$g_s = \sum_{1\le i_1\le j_1\le\cdots\le i_s\le j_s\le m} c_{i_1}d_{j_1}\cdots c_{i_s}d_{j_s} + \sum_{1\le i_1\le j_1\le\cdots\le i_s\le j_s\le m} d_{i_1}c_{j_1}\cdots d_{i_s}c_{j_s} \tag{22c}$$

By solving (21a) and (21b), the growth factor of the first procedure ξ_1 is

$$\xi_{1(1,2)} = \frac{tr(\mathbf{G}_1) \pm \tilde{j}\sqrt{4 - tr(\mathbf{G}_1)^2}}{2} \tag{23a}$$

and the second procedure ξ_2 is

$$\xi_{2(1,2)} = \frac{tr(\mathbf{G}_2) \pm \tilde{j}\sqrt{tr(\mathbf{G}_2)^2 - 4}}{2} \tag{23b}$$

Finally, ξ_1 and ξ_2 yields ξ, which indicates the growth factor of the total procedure as follows:

$$|\xi| = |\xi_1\xi_2| \le 1 \tag{24}$$

Equation (24) is always satisfied, so that the ADI-STDTD algorithm is unconditionally sTable in any case.

3.2 Numerical dispersion

Now we assume $\xi_1 = \xi_2 = \xi_s$ of (21a) and (21b)[9]. By adding (21a) and (21b) we then get

$$2\xi_s^2 - (tr(\mathbf{G}_1) + tr(\mathbf{G}_2))\xi_s + 2 = 0 \tag{25}$$

Equation (25) can be written as follows:

$$2(\xi_s + \xi_s^{-1}) - [tr(\mathbf{G}_1) + tr(\mathbf{G}_2)] = 0 \tag{26}$$

Making use of (10), (22a) and (22b), we get

$$\xi_s + \xi_s^{-1} = (\xi_s^{1/2} - \xi_s^{-1/2})^2 + 2 = -\sum_{s=1}^{m}[4\sin^2(\omega\tau/2)]^s + 2 \tag{27}$$

$$tr(\mathbf{G}_1) + tr(\mathbf{G}_2) = 4 + \sum_{s=1}^{m} g_s[\frac{\tau^2(h^2+1)(\eta_x^2 + \eta_y^2 + \eta_z^2)}{\varepsilon\mu(h+1)^2}]^s \tag{28}$$

Putting them into (26), we obtain

$$\sum_{s=1}^{m} g_s\{\frac{(h^2+1)}{(h+1)^2}[\frac{\sin^2(k_x\Delta x/2)}{(\Delta x/2)^2} + \frac{\sin^2(k_y\Delta y/2)}{(\Delta y/2)^2} + \frac{\sin^2(k_z\Delta z/2)}{(\Delta z/2)^2}]\}^s - [\frac{1}{\varepsilon\mu}\frac{\sin^2(w\tau/2)}{(\tau/2)^2}]^s = 0 \tag{29}$$

This suggests that numerical dispersion of the ADI-SFDTD method can be reduced to any degree if appropriate cells are used. Figure 1 is the normalized phase velocity of different ϕ for different FDTD schemes, we see the proposed ADI-FDTD has good performance. Figure 2 shows the normalized error contrast between ADI-FDTD and ADI-SFDTD, it clearly shows that ADI-SFDTD is more efficiency.

4. Conclusion

In this paper , a novel algorithm that based on SFDTD and ADI technique has be proposed, the spatial discretization scheme curl operator R is endued with two different styles when doing computation from the

$(s-1)$ th progression to s th progression. Then, Its stability and numerical dispersion has been analyzed. The results show that the proposed method has good efficiency and accuracy.

References

Huang, Z. X., Sha, W, Wu, X. L., et al. (2007) Optimal symplectic integrators for numerical solution of time-domain Maxwell's equations *Microwave and Optical Technology Letters computational Electrodynamics*, 49(3), 545-547.

Hirono, T., Lui, W., Seki, S., & Yoshikuni, Y. (2001). A three-dimensional fourth-order finite-difference time-domain scheme using symplectic integrator propagator. *IEEE Trans actions on Microwave theory and Techniques*, 49, 1640-1648. http://dx.doi.org/10.1109/22.942578

Iftikhar, A., &, Zhang, C. Z.(David) (2006). Dispersion-Error Optimized ADI FDTD. *IEEE Microwave Symposium Digest MTT-S International*. 173-176.

Kusaf, M. A., Oztoprak, Y., & Daoud, D. S. (2005). Optimized exponential operator coefficients for symplectic FDTD Method. *IEEE Microwave and Wireless Components Letters,* 15, 86-88. http://dx.doi.org/10.1109/LMWC.2004.842827

Namiki, T. (1999). A new FDTD algorithm based on alternating direction implicit method, *IEEE Trans. Microw. Theory Tech*, 47(10), 2003-2007. http://dx.doi.org/10.1109/22.795075

Ruth, F. D. (1983). A canonical integration technique. *IEEE Trans.Nucl Sci*, 30, 2669-2671. http://dx.doi.org/10.1109/TNS.1983.4332919

Thomas, J. W. (1995). *Numerical Partial Differential Equations: Finite Difference Methods*. Berlin, Germany: Springer Verlag.

Zhao, A. P. (2002) Two special notes on the implementation of the unconditionally sTable ADI-FDTD method, *Microw.Opt.Technol Lett,* 33(4), 273–277.http://dx.doi.org/10.1002/mop.10295

Zheng, F., Chen, Z., & Zhang, J. (2000). Toward the development of a three-dimensional unconditionally sTable finite-difference time-domain method. *IEEE Trans. Microwave. Theory Tech*, 48(9), 1550–1558. http://dx.doi.org/10.1109/22.869007

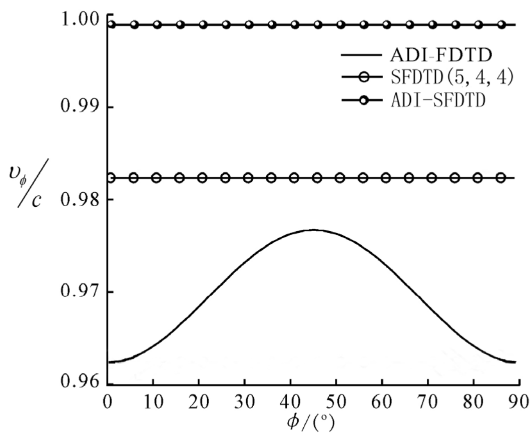

Figure 1. Normalized phase velocity of different ϕ for different FDTD schemes

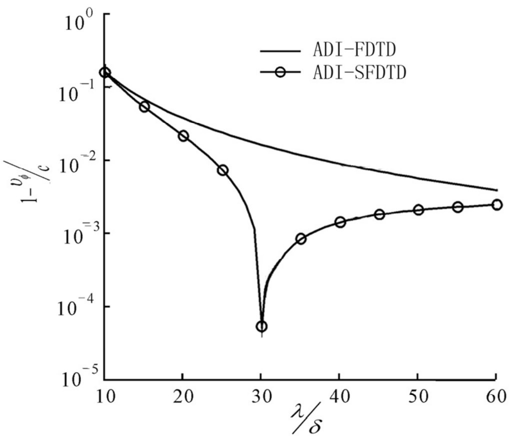

Figure 2. The normalized numerical dispersion error contrast between ADI-FDTD and ADI-SFDTD