Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 205 (2005) 343–356
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Theoretical basis for numerically exact three-dimensional time-domain algorithms

Christopher L. Wagner

School of Electrical Engineering and Computer Science, Washington State University, P.O. Box 642752, Pullman, WA 99164-2752, USA

Received 26 March 2004; received in revised form 11 November 2004; accepted 14 November 2004
Available online 30 December 2004

Abstract

In a one-dimensional (1D) homogeneous space, the classic Yee finite-difference timedomain (FDTD) algorithm is numerically exact when operated at the Courant stability limit. Numerically exact is taken to mean that, to within the sampling limit imposed by the discretization in space and time, the only errors are due to the finite precision of digital computer arithmetic. Unfortunately, the Yee algorithm is not numerically exact in two or more dimensions. However, using the design shown here, three-dimensional (3D) spatial differential operators can have 1D dispersion properties. Just as the space and time errors can be made to cancel in the 1D Yee algorithm, 3D algorithms (for hyperbolic systems of coupled first order equations) in an unbounded homogeneous space can be constructed which are, in theory, numerically exact. The differential operators presented here extend over a localized non-zero volume, unlike the usual nabla (or Del) operator which acts at a point. Our computer implementations are based on reconstruction methods, producing global range operators, thus our implementations of these operators are computationally expensive. A sample implementation of an approximate electromagnetic algorithms is described and is shown to produce results that are superior to the classic Yee algorithm for the cubic resonator problem.

Keywords: Finite difference time domain; FDTD; Exact FDTD; Maxwell's equations; Spherical derivative; Volume derivative; Electromagnetics

1. Introduction

The "magic time step" 1D Yee algorithm is a numerically exact differential equation solver [1,2]. To obtain theoretically similar performance in 3D, we express the divergence, gradient, and curl operators as derivatives with respect to a single variable. The spatial derivative operators presented here provide

E-mail address: clwagner@eecs.wsu.edu.

0021-9991/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2004.11.009

(theoretically) numerically exact solutions when used with a standard Yee "leap-frog" central difference time derivative. Once the differential operators are defined a dispersion relation is derived for hyperbolic systems of two coupled first-order equations and a specific dispersion relations for 3D electromagnetics algorithms is then derived. With the specified choice of algorithm parameters, the dispersion analysis shows theoretically exact propagation. An exact theory can be used as the basis for designing practical approximations. As a test of these volumetric differential operators, acoustic and electromagnetic algorithms have been coded, however, only electromagnetic results are presented here. The performance of this algorithm and the classic Yee algorithm are shown for the resonator test case.

2. Spherical volume differential operators

The spherical differential operators needed for exact algorithms can be defined as follows. Start with the gradient identity, and the closely related Gauss's divergence and vector Stokes' theorems:

$$\int\int\int_{\mathrm{B}} \nabla\psi \,\mathrm{d}v = \int\int_{\partial\mathrm{B}} \hat{\mathrm{n}}\psi \,\mathrm{d}s, \tag{1}$$

$$\int\int\int_{\mathrm{B}} \nabla\cdot\vec{W} \,\mathrm{d}v = \int\int_{\partial\mathrm{B}} \hat{\mathrm{n}}\cdot\vec{W} \,\mathrm{d}s, \tag{2}$$

$$\int\int\int_{\mathrm{B}} \nabla\times\vec{W} \,\mathrm{d}v = \int\int_{\partial\mathrm{B}} \hat{\mathrm{n}}\times\vec{W} \,\mathrm{d}s. \tag{3}$$

Assume the fields, surfaces, and volumes satisfy the conditions necessary for the theorems (1)–(3) to hold. Choose a sphere of radius $\rho \geqslant 0$ as the shape of the volume B. Now define the spherical gradient, divergence and curl as the scalar derivative of the corresponding integral identity:

$$\nabla_{\mathrm{s}}\psi(\vec{r}) = \frac{\partial}{\partial V}\int\int_{\partial\mathrm{B}} \hat{\mathrm{n}}\psi(\vec{r}+\vec{r}')\,\mathrm{d}s, \tag{4}$$

$$\nabla_{\mathrm{s}}\cdot\vec{W}(\vec{r}) = \frac{\partial}{\partial V}\int\int_{\partial\mathrm{B}} \hat{\mathrm{n}}\cdot\vec{W}(\vec{r}+\vec{r}')\,\mathrm{d}s, \tag{5}$$

$$\nabla_{\mathrm{s}}\times\vec{W}(\vec{r}) = \frac{\partial}{\partial V}\int\int_{\partial\mathrm{B}} \hat{\mathrm{n}}\times\vec{W}(\vec{r}+\vec{r}')\,\mathrm{d}s, \tag{6}$$

where the derivative is with respect to the volume V of sphere B, $V = \frac{4}{3}\pi\rho^3$ and where primes indicate variables of integration when necessary. The symbol ∇_{s} is used to denote a nabla operator that measures over a finite spherical volume (not the usual point operator). With the choice of spherical volumes, we can for illustrative purposes write out details of the right-hand side of (e.g.) (4)

$$\begin{aligned}\nabla_{\mathrm{s}}\psi(\vec{r}) = \frac{1}{4\pi\rho^2}\frac{\partial}{\partial\rho}\int\int_{\partial\mathrm{B}} &(\sin\theta\cos\phi\hat{\mathrm{x}} + \sin\theta\sin\theta\hat{\mathrm{y}} + \cos\theta\hat{\mathrm{z}})\psi((x+\rho\sin\theta\cos\phi)\hat{\mathrm{x}} + (y \\ &+ \rho\sin\theta\cos\phi)\hat{\mathrm{y}} + (z+\rho\cos\phi)\hat{\mathrm{z}})\rho^2\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi,\end{aligned} \tag{7}$$

where

$$\begin{aligned}\vec{r} &= r\hat{\mathrm{r}} = x\hat{\mathrm{x}} + y\hat{\mathrm{y}} + z\hat{\mathrm{z}}, \\ \vec{r}' &= \rho\hat{\mathrm{n}} = \rho(\sin\theta\cos\phi\hat{\mathrm{x}} + \sin\theta\sin\phi\hat{\mathrm{y}} + \cos\theta\hat{\mathrm{z}}),\end{aligned} \tag{8}$$

and where the variables of integration are the spherical polar angle θ and the equatorial angle ϕ. To see that the spherical deivatives correspond to the usual point derivatives when the sphere size is zero, write the volume partial derivative as a limit and let $F(V,\vec{r}) = \int\int \hat{\mathrm{n}} \cdot \vec{W}(\vec{r}+\vec{r}')\,\mathrm{d}s$, in (5)

$$\nabla_{\mathrm{s}} \cdot \vec{W}(\vec{r}) = \lim_{v\to 0} \frac{F(V+v,\vec{r}) - F(V,\vec{r})}{v} \quad V,\ v \geqslant 0, \tag{9}$$

then set the volume V to zero

$$\begin{aligned} \nabla_{\mathrm{s}} \cdot \vec{W}(\vec{r})|_{V=0} &= \lim_{v\to 0} \frac{F(v,\vec{r}) - F(0,\vec{r})}{v} = \lim_{v\to 0} \frac{F(v,\vec{r})}{v} \\ &= \lim_{v\to 0} \frac{1}{v} \int\int_{\partial \mathrm{B}} \hat{\mathrm{n}} \cdot \vec{W}(\vec{r}+\vec{r}')\,\mathrm{d}s = \nabla \cdot \vec{W}(\vec{r}). \end{aligned} \tag{10}$$

The final line of (10) is a typical definition of the usual point divergence [3], "typical" because the usual point divergence definition does not need to specify the shape. The volume operator ∇_{s} is useful because it is the usual nabla operator when computed with zero radius, while for non-zero radius the operator can produce exact time-domain algorithms. Eqs. (4)–(6) are the definitions of the space derivative operators for the present class of exact algorithms. As written, (4)–(6) are functions of the size of the sphere. For some particular algorithm, the as yet undetermined radius of the sphere is fixed.

3. Stability and dispersion analysis

A simple technique for analyzing the stability of finite difference methods was given by von Neumann and Richtmyer [4]. In summary, substituting a complete basis set of complex-exponential plane-wave solutions into the finite difference equations of the system converts time and space derivatives into multiplicative factors. Then the fields are eliminated from the resulting algebraic system to obtain the dispersion relation. The dispersion relation can be analyzed to obtain stability requirements. The stability conditions are obtained by examining the dispersion equation for the relationship between the wave vector $\vec{k}$ and angular frequency ω.

When the relationship is complex, the result is an unstable algorithm [4] or superluminal propagation [5]. The material parameters are assumed constant in the region of analysis. For a 3D space, plane waves of arbitrary wave-vector and frequency form a complete basis. While von Neumann did obtain a dispersion relation (without calling it such), he only considered stability properties. Taflove and Brodwin [6] demonstrated the utility of analyzing the dispersion relation for understanding propagation behavior, i.e., the accuracy of the simulation. Thus, the dispersion relation for a time-domain algorithm provides a powerful tool for analyzing the algorithm for both stability and accuracy.

3.1. Effect on plane waves

In order to derive the algorithm stability properties and dispersion relation the effect of the spherical volume differential operators on plane waves is required. For simplicity a cubic cell grid of step size Δ_g will be used throughout. For a specific implementation of an algorithm a spatial grid will need to be chosen (e.g., staggered or collocated), at this point we only require a staggered-in-time grid. The staggering or collocation of the spatial grid need not be specified yet for the volume operators. Given a complex scalar plane wave with wave vector $\vec{k}$ and frequency ω

$$\psi(\vec{r},t) = \mathrm{e}^{\mathrm{i}(\vec{k}\cdot\vec{r}-\omega t)}, \tag{11}$$

where

$$\vec{k} = k\hat{\mathrm{k}} = k(\sin\alpha\cos\beta\hat{\mathrm{x}} + \sin\alpha\sin\beta\hat{\mathrm{y}} + \cos\alpha\hat{\mathrm{z}}) = k_x\hat{\mathrm{x}} + k_y\hat{\mathrm{y}} + k_z\hat{\mathrm{z}}, \tag{12}$$

the volume derivatives and the usual Yee time derivative are computed. The angles α and β are the spherical polar and equator angles of the wave vector. The classic Yee algorithm uses central differences to approximate the temporal derivatives. Thus, in the Yee algorithm the temporal derivative of (11) is given by

$$\partial_t^{\mathrm{Yee}}\psi(\vec{r},t) = \frac{\psi(\vec{r},t+\Delta_t/2) - \psi(\vec{r},t-\Delta_t/2)}{\Delta_t} = -\frac{2\mathrm{i}}{\Delta_t}\sin\left(\frac{\omega\Delta_t}{2}\right)\psi(\vec{r},t), \tag{13}$$

where Δ_t is the temporal step size, and $\partial_t^{\mathrm{Yee}}$ denotes the Yee finite-difference time derivative. The space derivative operators for an exact algorithm must have the same functional effect on the 3D plane wave as (13) in order to obtain a functionally 1D dispersion relation for the 3D space. Then it will be possible to choose values of the algorithm parameters to obtain matched space and time derivatives, giving theoretically exact homogeneous-space propagation.

When acting on a plane wave (11), the spherical volume gradient (4) becomes

$$\begin{aligned}\nabla_{\mathrm{s}}\psi(\vec{r},t) &= \frac{1}{4\pi\rho^2}\frac{\partial}{\partial\rho}\left(\int_0^{2\pi}\int_0^{\pi}\hat{\mathrm{n}}\psi(\vec{r}+\vec{r}',t)\rho^2\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi\right)\\ &= \frac{1}{4\pi\rho^2}\frac{\partial}{\partial\rho}\left(\rho^2\int_0^{2\pi}\int_0^{\pi}\hat{\mathrm{n}}\,\mathrm{e}^{\mathrm{i}\vec{k}\cdot\vec{r}'}\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi\right)\psi(\vec{r},t).\end{aligned} \tag{14}$$

The integrals in (14) express the standard relation between spherical Bessel functions and surface integrals of spherical harmonics. See for example, Stratton [7, Section 7.7, Eq. (60)]. The spherical Bessel function integral is

$$i^n 4\pi j_n(k_\rho)P_n^m(\cos\alpha)\begin{matrix}\sin(m\beta)\\ \cos(m\beta)\end{matrix} = \int_0^{2\pi}\int_0^{\pi}\mathrm{e}^{\mathrm{i}\vec{k}\cdot\vec{r}'}P_n^m(\cos\theta)\begin{matrix}\sin(m\phi)\\ \cos(m\phi)\end{matrix}\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi, \tag{15}$$

where j_n is the spherical Bessel function of order n, and P_n^m is the associated Legendre polynomial. Using (15) and (12), the integrals in (14) become

$$\nabla_{\mathrm{s}}\psi(\vec{r},t) = \frac{1}{4\pi\rho^2}\frac{\partial}{\partial\rho^2}\left(\mathrm{i}4\pi\rho^2 j_1(k\rho)\right)\hat{\mathrm{k}}\psi(\vec{r},t). \tag{16}$$

The derivative is best handled by using the spherical Bessel differentiation formula [8, Eq. (10.1.23)]

$$\left(\frac{\mathrm{d}}{z\,\mathrm{d}z}\right)^m[z^{n+1}j_n(z)] = z^{n-m+1}j_{n-m}(z). \tag{17}$$

By application of (17) to (16), one obtains the effect on the plane wave of the volume gradient operator

$$\nabla_{\mathrm{s}}\psi(\vec{r},t) = \mathrm{i}j_0(k\rho)\vec{k}\psi(\vec{r},t). \tag{18}$$

Now set the radius $\rho = a\Delta_g$, where a is the radius of the operator sphere in grid steps. With all the terms of the volume gradient found, the various integrals in the volume divergence and volume curl have been solved for vector plane waves. The volume gradient, divergence, and curl of some scalar ψ or vector $\vec{A}$ plane wave become:

$$\nabla_{\mathrm{s}}\psi(\vec{r},t) = \mathrm{i}j_0(ka\Delta_g)\vec{k}\psi(\vec{r},t), \tag{19}$$

$$\nabla_{\mathrm{s}}\cdot\vec{A}(\vec{r},t) = \mathrm{i}j_0(ka\Delta_g)\vec{k}\cdot\vec{A}(\vec{r},t), \tag{20}$$

$$\nabla_{\mathrm{s}}\times\vec{A}(\vec{r},t) = \mathrm{i}j_0(ka\Delta_g)\vec{k}\times\vec{A}(\vec{r},t). \tag{21}$$

The key feature of the results (19)–(21) is the $j_0(ka\Delta_g)$ factor. As will be shown, this factor allows time-domain algorithms constructed with these volume operators to be exact. Note that if the algorithm radius

a is zero, then (19)–(21) are the usual gradient, divergence, and curl of a plane wave, as expected from the previously established correspondence of zero-sized spherical derivatives and point derivatives.

3.2. Dispersion relation for exact algorithms

The operators presented here can be used to construct numerically exact algorithms for hyperbolic systems of coupled first-order equations, for example Maxwell's equations. The time derivatives will be approximated with the usual "leap-frog" central difference as used in the classical Yee FDTD algorithm [1], with the volume divergence, volume gradient, or volume curl given by (4)–(6). Given a self-consistent hyperbolic system of two coupled first-order equations

$$\frac{\partial F_1}{\partial t} = h_1 \Delta_1 F_2, \quad \frac{\partial F_2}{\partial t} = h_2 \Delta_2 F_1, \tag{22}$$

where the Δ_is are one of div, grad, or curl and the h_is are the given scale constants for each equation. The system must have propagating solutions. A plane wave solution $e^{i(\vec{k}\cdot\vec{r}-\omega t)}$ is assumed (vector or scalar, as required), and the numerical differentiation on the plane wave as given by (19)–(21). The usual finite-difference time derivatives (13) are substituted into (22) and the resulting system is reduced and simplified. There are three possible forms for each of the system's equations

$$\frac{\partial F_i}{\partial t} = h_i \nabla \cdot \vec{F}_j, \quad \frac{\partial \vec{F}_m}{\partial t} = h_m \nabla F_n, \quad \frac{\partial \vec{F}_p}{\partial t} = h_p \nabla \times \vec{F}_q, \tag{23}$$

where the Fs, represent the fields of the system. The finite-difference time derivative (13) and the spherical volume derivatives (19)–(21) acting on (23) with assumed plane wave solutions will produce

$$\begin{aligned} -\frac{2}{\Delta_t}\sin\left(\frac{\omega\Delta_t}{2}\right) F_i &= h_i j_0(ka\Delta_g)\vec{k}\cdot\vec{F}_j, \\ -\frac{2}{\Delta_t}\sin\left(\frac{\omega\Delta_t}{2}\right) \vec{F}_m &= h_m j_0(ka\Delta_g)\vec{k}F_n, \\ -\frac{2}{\Delta_t}\sin\left(\frac{\omega\Delta_t}{2}\right) \vec{F}_p &= h_p j_0(ka\Delta_g)\vec{k}\times\vec{F}_q. \end{aligned} \tag{24}$$

Since the system is hyperbolic with propagating solutions, the dispersion relation results from simultaneously solving the algebraic system of equations produced by substituting (24) into (22) and eliminating common factors. This yields

$$\left(\frac{2}{\Delta_t}\sin\left(\frac{\omega\Delta_t}{2}\right)\right)^2 = (h_1 h_2 k j_0(ka\Delta_g))^2. \tag{25}$$

Substituting the trigonometric form for the spherical Bessel function into (25) and simplifying yields

$$\left(\frac{2}{\Delta_t}\sin\left(\frac{\omega\Delta_t}{2}\right)\right)^2 = \left(h_1 h_2 \frac{\sin(ka\Delta_g)}{a\Delta_g}\right)^2. \tag{26}$$

Note that, by design, this is functionally equivalent to the Yee 1D dispersion relation [2]. The algorithm parameters (Δ_t, Δ_g, a) are chosen such that

$$\left(\frac{h_1 h_2 \Delta_t}{2a\Delta_g}\right)^2 = 1, \tag{27}$$

or equivalently, so that the Yee algorithm's Courant number, s, is

$$s = \frac{c\Delta_t}{\Delta_g} = \frac{|h_1 h_2|\Delta_t}{\Delta_g} = 2a, \tag{28}$$

where $c = |h_1 h_2|$ is the wave speed. With this choice of parameters the coefficients of the sine functions in (26) are unity. Taking the square-root and the arcsine of both sides of (26) and simplifying produces

$$\omega = |h_1 h_2| k = ck, \tag{29}$$

which is the ideal homogeneous space continuum result. In order for (29) to be true we must also have spatially and temporally band-limited waves because the computation is sampled in space and time

$$\omega \leqslant \frac{\pi}{\Delta_t} \quad \text{and} \quad |k_x|, |k_y|, |k_z| \leqslant \frac{\pi}{\Delta_g}. \tag{30}$$

A surprising aspect of the stability and exactness condition (27) is that it can be imposed for any Δ_t because the radius a is a free parameter of the algorithm. It is unknown if practical versions of this algorithm work for large time steps.

The stability limit S_L can be obtained by solving the dispersion relation (26) for the angular frequency using the substitution $S_L = c\Delta_t/\Delta_g$

$$2\pi f = \omega = \frac{2}{\Delta_t} \arcsin\left(\frac{S_L}{2a}\left|\sin(ka\Delta_g)\right|\right). \tag{31}$$

Stability requires that ω be real-valued for all grid-sample-permitted $\vec{k}$ vectors. For this algorithm the argument of the arcsin function must be between negative and positive unity, inclusive. Thus, the stability limit S_L as a function of the radius a is given by

$$\frac{1}{S_L(a)} = \max_{\substack{\vec{k} \\ |k_x|, |k_y|, |k_z| \leqslant \frac{\pi}{\Delta_g}}} \left(\frac{1}{2a}\left|\sin(ka\Delta_g)\right|\right). \tag{32}$$

The solution for S_L can be explicitly written with two cases

$$S_L(a) = \begin{cases} \frac{2a}{\sin(\sqrt{3}\pi a)}, & a < \frac{1}{2\sqrt{3}}, \\ 2a, & a \geqslant \frac{1}{2\sqrt{3}}. \end{cases} \tag{33}$$

Thus, for small spheres (first case) the stability limit is greater then the exactness condition, and for large spheres (second case) the stability limit and the exactness condition are equal.

3.3. Electromagnetics dispersion relation

A specific example of obtaining a dispersion relation for electromagnetics starts with Maxwell's curl equations in homogeneous source-free space

$$\epsilon \frac{\partial \vec{E}}{\partial t} = \nabla \times \vec{H}, \quad \mu \frac{\partial \vec{H}}{\partial t} = -\nabla \times \vec{E}, \tag{34}$$

where $\vec{E}, \vec{H}$ are the electric and magnetic fields, and ϵ, μ are the permittivity and permeability. Applying (13) and (21) to each term of (34) for an assumed vector plane wave solution yields

$$\begin{aligned} -\frac{2i\epsilon}{\Delta_t} \sin\left(\frac{\omega\Delta_t}{2}\right)\vec{E} &= i j_0(ka\Delta_g)\vec{k} \times \vec{H}, \\ -\frac{2i\mu}{\Delta_t} \sin\left(\frac{\omega\Delta_t}{2}\right)\vec{H} &= i j_0(ka\Delta_g)\vec{k} \times \vec{E}. \end{aligned} \tag{35}$$

Eliminating one of the fields, chosen here to be $\vec{H}$, produces a wave equation in the remaining field

$$\frac{4}{c^2\Delta_t^2}\sin^2\left(\frac{\omega\Delta_t}{2}\right)\vec{E} = -j_0^2(ka\Delta_g)\vec{k}\times(\vec{k}\times\vec{E}) = j_0^2(ka\Delta_g)k^2\vec{E}, \tag{36}$$

where $c = 1/\sqrt{\epsilon\mu}$ is the wave speed. Eliminating the remaining common factors gives the dispersion relation

$$\frac{4}{c^2\Delta_t^2}\sin^2\left(\frac{\omega\Delta_t}{2}\right) = k^2 j_0^2(ka\Delta_g) = \frac{1}{(a\Delta_g)^2}\sin^2(ka\Delta_g). \tag{37}$$

This is the same as the Yee 1D dispersion equation, so the previous stability and exactness conditions (27) and sampling limits (30) apply. Then using these conditions the dispersion relation becomes

$$\omega = ck, \tag{38}$$

which is the ideal result up to the sampling limits.

3.4. Theory summary and comments

Von-Neumann stability analysis is strictly applicable to an unbounded, homogeneous region. In the Yee algorithm inhomogeneous spaces are stabilized in the region with the greatest wave-speed, then the slower wave-speed regions will be stable (though not necessarily sufficiently sampled). This piecewise homogeneity method typically produces stable algorithms. Analysis more complicated than the Von-Neumann method is required for general stability analysis of inhomogeneous problems, and is not considered here.

The same piecewise concept can be applied to a spherical derivative algorithm, stabilize exactly in the region with the greatest wave-speed, then the slower wave-speed regions will be stable. However, because the radius of the operator is a free parameter, spherical derivatives can be defined to be exact in all (piecewise homogeneous) material regions. One simply needs a spherical operator with radii matched to the materials in each region, that is, a different sized sphere in each material region as given by (27), keeping a constant time step throughout the space. Sampling conditions must also be satisfied in all regions with either approach. It is unknown if practical algorithms are stable with these approaches.

Given proper sampling of the variables, we have defined spherical derivative operators which combined with the Yee central-difference time-derivative theoretically produces exact propagation in a homogeneous region. The exactness condition is stable for any time-step. It is important to distinguish between the theoretical spherical derivatives, which are spheres of some chosen radius, and the following Shannon sampling theory computer program implementations. Sampling theory calculations involve sums over the entire domain. The result is a global range computation of the finite sized operator. This global range results from the sampling theory calculations, it is an implementation detail, the mathematical operator is local to a finite spherical region.

4. Proof-of-principle algorithms

In order to test the concepts of the volume operators presented here, a canonical problem is solved analytically and numerically with the classical Yee and volume algorithms. The test case is a cubic resonator, with a volume of L^3 cubic meters. This is discretized with $M = 10$ cells in each axis, so $L = M\Delta_g$. The source and sample point is centered in the resonator. The resonator walls are aligned with the grid. The electromagnetics resonator boundary is a perfect electric conductor (PEC).

The electromagnetic source and sample is the two center-most E_z nodes, two nodes are required because of the symmetries of the electromagnetics Yee grid. The E_z grid will have $(M+1)^2M$ nodes. The E_z nodes

are zero on the x constant and y constant boundary surfaces, there are no E_z. nodes in the z constant boundary surfaces. Centering the source and sample requires M even. The other $\vec{E}$ components, and the $\vec{H}$ components are staggered around E_z in the usual Yee grid.

The test resonators are sampled at the center of the cubic domain, the resulting time series is Fourier transformed with a raised-cosine window ($1 - \cos(2\pi n\Delta_t/T)$, where T is the total simulation time) to obtain amplitude spectra. The material parameters, wave speed, and space delta are all set to unity. All calculations are done with single precision (32 bit) floating point numbers. The computations are run for 65,536 time steps which is sufficient to see the approximately 10^7 dynamic range of the 32 bit computations. The spectra are plotted from zero frequency to the Nyquist limit. To facilitate meaningful comparisons, all algorithms are implemented on the usual Yee staggered cubic-cell grid with $c\Delta_t/\Delta_g = 1/\sqrt{3}$.

4.1. Reference theory

The reference theory is an analytic resonator with band-limited $\vec{k}$ and ω where the band-limiting accounts for the sampled nature of the simulations. The eigenfunction sums are truncated to accomplish the band-limiting. The coordinate systems for analysis are chosen so that the origin is at the center of the cubic resonators, with walls aligned with coordinate surfaces. The interior region of the resonators will be denoted as Ω, the boundary by $\partial\Omega$ and the closure by $\bar{\Omega}$.

As usual, the electromagnetics simulations (the computer programs) update the fields using the coupled first order system of Maxwell's curl equations, constitutive relations, boundary conditions, and initial conditions:

$$\frac{\partial\vec{D}}{\partial t} = \nabla\times\vec{H} - \vec{J}, \quad \frac{\partial\vec{B}}{\partial t} = -\nabla\times\vec{E}, \quad \vec{r}\in\Omega,\ t\geqslant 0, \tag{39}$$

$$\vec{B} = \mu\vec{H}, \quad \vec{D} = \epsilon\vec{E}, \quad \vec{J}_c = \sigma\vec{E}, \quad \vec{r}\in\Omega,\ t\geqslant 0, \tag{40}$$

$$\hat{\mathrm{n}}\times\vec{E} = 0, \quad \vec{r}\in\partial\Omega,\ t\geqslant 0, \tag{41}$$

$$\vec{E}\equiv 0, \quad \vec{D}\equiv 0, \quad \vec{B}\equiv 0, \quad \vec{H}\equiv 0, \quad \vec{J}\equiv 0, \quad \vec{r}\in\bar{\Omega},\ t<0. \tag{42}$$

The difference between various algorithms is how they compute derivatives in (39). For analysis the electromagnetics resonator is modeled as a boundary-value problem with a forcing function. The model for $\vec{E}$ is obtained by eliminating $\vec{H}$,$\vec{B}$ and $\vec{D}$ from (39)–(42), using the assumptions that the permittivity ϵ and permeability μ are invariant scalars, and the conductivity σ is zero. The validation model is:

$$\nabla\times\nabla\times\vec{E} + \frac{1}{c^2}\frac{\partial^2\vec{E}}{\partial t^2} = -\mu\frac{\partial\vec{J}}{\partial t}, \quad \vec{r}\in\Omega,\ t\geqslant 0, \tag{43}$$

$$\hat{\mathrm{n}}\times\vec{E} = 0, \quad \vec{r}\in\partial\Omega,\ t\geqslant 0, \tag{44}$$

$$\vec{E}\equiv 0, \quad \vec{J}\equiv 0, \quad \vec{r}\in\bar{\Omega},\ t<0. \tag{45}$$

Eq. (43) is the governing partial differential equation, (44) gives the boundary conditions, and (45) are causality conditions. The current density $\vec{J}$ is a centered, two cell long, z-directed filament

$$\vec{J}(\vec{r},t) = \delta(x)\delta(y)(H(z+\Delta_g) - H(z-\Delta_g))\delta'(t)\hat{z}, \tag{46}$$

where $\delta(\,)$ is a delta function, $H(\,)$ is a step function, and $\delta'(\,)$ is a doublet. The formal solution for the $\vec{E}$ field is

$$\vec{E}(\vec{r},t) = \sum_{\substack{m,p=1 \\ \text{ODD}}}^{\infty} \sum_{\substack{q=0 \\ \text{EVEN}}}^{\infty} \left[A^X_{mpq}(t)\Phi^X_{mpq}(\vec{r})\hat{\mathrm{x}} + A^Y_{mpq}(t)\Phi^Y_{mpq}(\vec{r})\hat{\mathrm{y}} + A^Z_{mpq}(t)\Phi^Z_{mpq}(\vec{r})\hat{\mathrm{z}} \right], \tag{47}$$

where $A^X_{mpq}(t)$ is the x axis mode time dependence, and where $\Phi^X_{mpq}(\vec{r})$ is the x axis eigenfunction, and similarly for the y and z axes. The symmetry and boundary conditions require that the m, p indices are odd, and the q index is even. For notational simplicity let $k_m = m\pi/L$, $k_p = p\pi/L$ and $k_q = q\pi/L$. Only the z axis eigenfunction and time dependence are needed for the validation tests. The z axis eigenfunction is

$$\Phi^Z_{mpq}(\vec{r}) = \cos(k_m x)\cos(k_p y)\cos(k_q z). \tag{48}$$

The z axis time-dependence is

$$A^Z_{mpq}(t) = \begin{cases} \frac{-16c^2}{L^3}\sin\left(\Delta_g k_g\right)\left(\frac{\mu\delta(t)}{k_q} + \left(\frac{k_q}{\epsilon\omega_{mpq}} - \frac{\mu\omega_{mpq}}{k_q}\right)\sin\left(\omega_{mpq}t\right)H(t)\right), & q > 0, \\ \frac{-8c^2\Delta_g}{L^3}\left(\mu\delta(t) - \mu\omega_{mpq}\sin\left(\omega_{mpq}t\right)H(t)\right), & q = 0, \end{cases} \tag{49}$$

where the eigenvalues or resonant mode frequencies are

$$k^2_{mpq} = \left(\frac{\pi}{L}\right)^2(m^2 + p^2 + q^2) = \left(\frac{\omega_{mpq}}{c}\right)^2. \tag{50}$$

Because of the sampling considerations, the indices must satisfy m, $p < M$ and $q < M - 1$. The resonator is discretized with $M = 10$ cells in each axis. Analytically, the potential between the sample nodes is given by a line integral of the $\vec{E}$ field, along the z axis sample line. The value of "potential" extracted from the simulations is the sum of the two central E_z field samples times the grid step size. The predicted potential is then the truncated (band-limited) sum

$$U(t) = -C_1 \sum_{\substack{m,p=1 \\ \text{ODD}}}^{M-1} \sum_{\substack{q=0 \\ \text{EVEN}}}^{M-2} A^Z_{mpq}(t) 2\Delta_g \cos\left(\frac{q\pi}{2M}\right), \tag{51}$$

where C_1 is a constant described in the next section.

4.2. Analytic and sampled sources

In the reference theory the source current density (46) consists of delta, step, and doublet functions. In the simulations these become discrete functions, acting on space-time cells. The theoretical filament $\vec{J}$ passes through a point, in the simulation $\vec{J}$ acts on the entire area of a cell. Thus, we need to divide the $\vec{J}$ in the simulation by Δ_g^2.

A familiar sampling theory consideration is that two samples per wavelength are required, a frequency domain condition. The time domain version of this is that two samples per pulse are required. Using the two-samples per pulse condition the correct time series for a sampled delta in the simulation is $(\ldots, 0, 1/2, 1/2, 0, \ldots)/\Delta_t$. This series is the properly sampled unit-amplitude delta, which has a spectral null at the temporal Nyquist frequency. The theoretical source time-dependence is a doublet $\delta'(t)$. Taking a Yee time derivative of the sampled unit-amplitude delta gives the series $(\ldots, 0, 1/2, 0, -1/2, 0, \ldots)/\Delta_t^2$. Now, including the current density area factor from above we get the series $(\ldots, 0, 1/2, 0, -1/2, 0, \ldots)/(\Delta_t^2\Delta_g^2)$. The current-density time series used in the simulation is $(\ldots, 0, 1/2, 0, -1/2, 0, \ldots)$, so the scale factor is then $C_1 = \Delta_t^2\Delta_g^2$. This C_1 factor is used in (51) for the results presented.

As shown in [9], given the complete analytical solution of the sampled resonator, the frequencies that will exist in the numerical grid can be precisely predicted by using the dispersion relation. After dispersion shifting there will be a new list of frequencies and amplitudes. This new list is the set of basis functions for generating a dispersion-shifted theoretical prediction of the algorithm behavior. A plot of the last few time steps of the potential sample and prediction is shown in Fig. 1. The dispersion-shifted theory uses (51), with

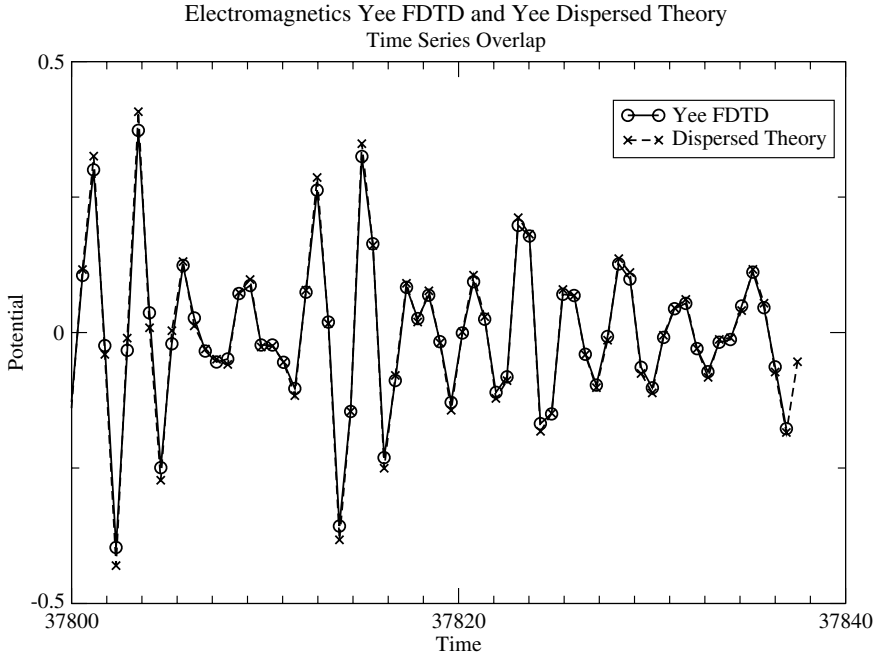

Fig. 1. Plot of the last few time steps of the potential sample in an electromagnetic Yee FDTD resonator simulation and Yee dispersion-shifted theoretical time evolution of the resonator. The overlap between theory and simulation is fair, though not perfect.

all instances of ω_{mpq} replaced by the Yee dispersed frequencies. This plot shows that there is fair agreement between the time-series data of the Yee-dispersed theoretical prediction and the simulation.

4.3. Sampling theory algorithms

In order to implement the algorithms presented here, some method of computing derivatives (4)–(6) is required. Shannon's sampling and reconstruction methods are used here. Such a computation is inefficient, but simple to implement. A review of one-dimensional sampling theory calculation is presented first.

Given basis functions ϕ, some band-limited function $f(x)$ is exactly reconstructed at an continuously varying arbitrary point from the discrete samples $f(i\Delta_g)$ with the infinite sum

$$f(x) = \sum_{i=-\infty}^{\infty} \phi\left(\frac{x}{\Delta_g} - i\right) f(i\Delta_g), \tag{52}$$

where for Shannon reconstruction the basis function ϕ is the $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$ function. Now suppose we wish to compute the derivative of f. From (52) this becomes

$$f'_x = \frac{\partial}{\partial x}\left(\sum_i \phi\left(\frac{x}{\Delta_g} - i\right) f(i\Delta_g)\right) = \frac{1}{\Delta_g}\sum_i f(i\Delta_g)\phi'\left(\frac{x}{\Delta_g} - i\right), \tag{53}$$

where $\phi'(x) = \partial\phi(x)/\partial x$. The important observation is that algorithms only require the value of $f'(x)$ on a discrete lattice, so the function ϕ' can be pre-computed and called derivative coefficients C. In terms of derivative coefficients, the derivative of f on the discrete lattice is just a weighted sum of the samples

$$f'(x) = \frac{1}{\Delta_g} \sum_i f(i\Delta_g) C\left(\frac{x}{\Delta_g} - i\right). \tag{54}$$

To find the derivative at one point we need to sum over all the samples. Eq. (54) is the direct-domain equivalent of the 1D pseudo-spectral time-domain (PSTD) [10] algorithm spatial derivative. The algorithm implementations presented subsequently to test spherical derivatives are 3D generalizations of (54), with very complicated calculations required to obtain the coefficients. Nonetheless, the algorithm employs a simple weighted sum over the volume to compute a spatial derivative at one point.

The 3D function reconstruction sum is

$$f(x,y,z) = \sum_i \sum_j \sum_k \phi\left(\frac{x}{\Delta_g} - i\right) \phi\left(\frac{y}{\Delta_g} - j\right) \phi\left(\frac{z}{\Delta_g} - k\right) f(i\Delta_g, j\Delta_g, k\Delta_g). \tag{55}$$

For notational simplicity let $g(\vec{r}) = \phi(x)\phi(y)\phi(z)$, and use vector notation for sum indices, $\vec{I} = (i,j,k)$, so (55) becomes

$$f(\vec{r}) = \sum_{\vec{I}} g\left(\frac{\vec{r}}{\Delta_g} - \vec{I}\right) f(\Delta_g \vec{I}). \tag{56}$$

Now compute a spherical gradient of f

$$\nabla_s f(\vec{r}) = \nabla_s \sum_{\vec{I}} g\left(\frac{\vec{r}}{\Delta_g} - \vec{I}\right) f(\Delta_g \vec{I}) = \sum_{\vec{I}} f(\Delta_g \vec{I}) \nabla_s g\left(\frac{\vec{r}}{\Delta_g} - \vec{I}\right), \tag{57}$$

where the exchange of the summation and ∇_s is permitted because the ∇_s only operates on $\vec{r}$, and not on $\vec{I}$. Additionally in an algorithm the sums are finite, so convergence issues do not apply. Just as in the 1D derivative example (54), the function $\nabla_s g(\vec{r})$ can be precomputed. Define the gradient update coefficients

$$\vec{C}_G(\vec{r}) = \nabla_s g(\vec{r}) = \frac{\partial}{\partial V} \int\int_{\partial B} \hat{n} g(\vec{r} + \vec{r}')\, ds = \frac{1}{4\pi\rho^2} \frac{\partial}{\partial \rho} \int\int_{\partial B} \hat{n} g(\vec{r} + \vec{r}')\, ds, \tag{58}$$

where the surface integration is over the spherical surface of the operator. Eq. (58) can be expanded for calculation as shown in (7). This is a very complicated calculation which is performed numerically for a fixed radius. In terms of the coefficients, the spherical gradient, divergence, and curl are:

$$\nabla_s \psi(\vec{r}) = \sum_{\vec{I}} \vec{C}_G\left(\frac{\vec{r}}{\Delta_g} - \vec{I}\right) \psi(\Delta_g \vec{I}), \tag{59}$$

$$\nabla_s \cdot \vec{W}(\vec{r}) = \sum_{\vec{I}} \vec{C}_G\left(\frac{\vec{r}}{\Delta_g} - \vec{I}\right) \cdot \vec{W}(\Delta_g \vec{I}), \tag{60}$$

$$\nabla_s \times \vec{W}(\vec{r}) = \sum_{\vec{I}} \vec{C}_G\left(\frac{\vec{r}}{\Delta_g} - \vec{I}\right) \times \vec{W}(\Delta_g \vec{I}), \tag{61}$$

where $\vec{C}_G$, $\vec{C}_D$ and $\vec{C}_C$ are the gradient, divergence and curl update coefficients. Eqs. (59)–(61) are weighted sums over the volume. Depending on the contents of the coefficient arrays, they could provide classic Yee (with almost all coefficients zero), higher-order Yee, direct-domain equivalent of PSTD, or other algorithms. The spherical-derivative electromagnetics algorithm uses two staggered curls (61). Only one octant of one component of $\vec{C}_G$ needs to be computed, the other octants and components can be obtained from simple symmetry considerations. Similar considerations allow finding the $\vec{C}_D$ and $\vec{C}_C$ coefficients from the $\vec{C}_G$ if staggering is properly considered. For a collocated grid $\vec{C}_G = \vec{C}_D = \vec{C}_C$. In the Yee staggered grid

used here there are simple fixed origin shifts, or staggers, between the $\vec{E}$ and $\vec{H}$ fields and the respective curl coefficients.

Eqs. (59)–(61) directly translate to computer code. In the analysis $\psi(\Delta_g\vec{I})$ refers to field samples with physical units for the arguments. In the computer code these samples are referenced with integer indexes independent of the physical scales. A field located at integer physical coordinates $f(\Delta_g\vec{I})$ is indexed with the samples at integer indexes $f(\vec{I})$. Similarly, only integers index the coefficient arrays. Let the vector $\vec{R}$ be the staggering vector in units of Δ_g, so $\vec{R} = 0$ for a collocated derivative. Let $\vec{L} = (l,m,n)$ be a second index, referring to a possibly staggered coordinate. Then the spherical gradient, divergence, curl, and coefficients at integer index points are given by

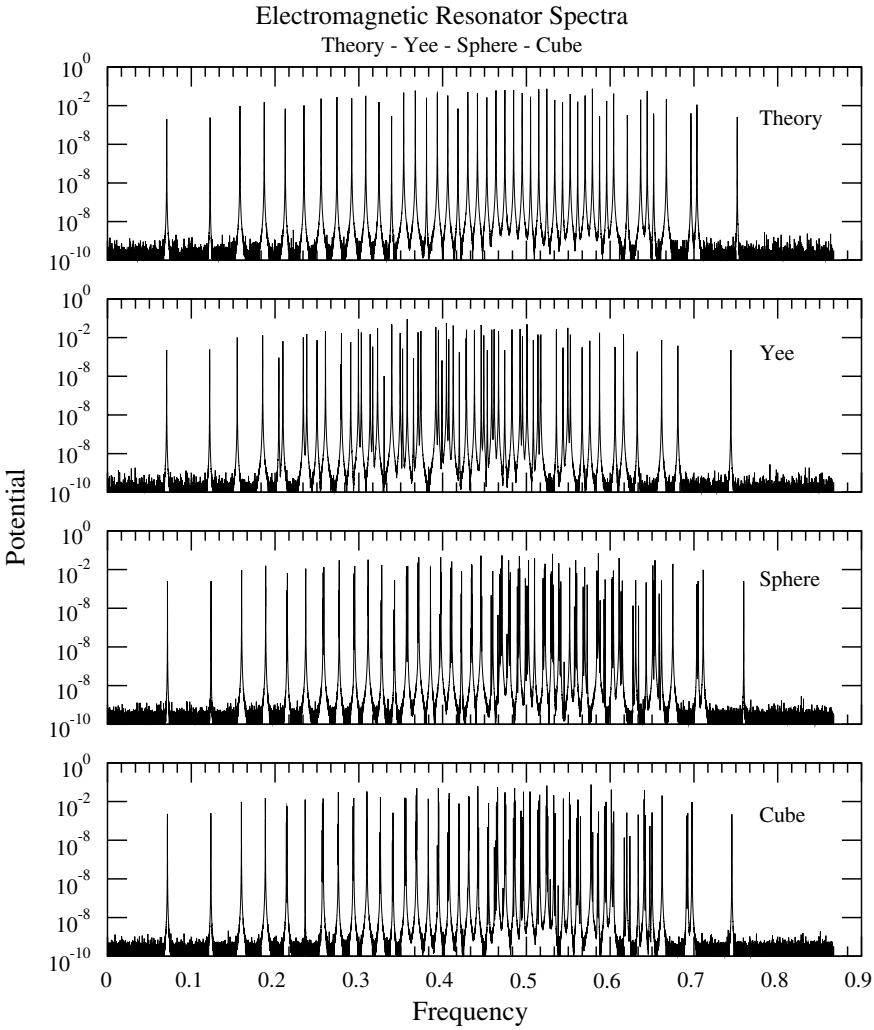

Fig. 2. Spectra of solutions to the electromagnetics resonator problem. Top plot is the theory (51), middle plot is Yee algorithm and bottom plot is spherical derivative algorithm. Note the mode structure is scrambled in the Yee algorithm. The spherical algorithm shows much less mode splitting, the mode structure is closer to the ideal.

$$\nabla_s \psi(\vec{L}) = \sum_{\vec{I}} \vec{C}_G(\vec{L} - \vec{I}) \psi(\vec{I}), \tag{62}$$

$$\nabla_s \cdot \vec{W}(\vec{L}) = \sum_{\vec{I}} \vec{C}_D(\vec{L} - \vec{I}) \cdot \vec{W}(\vec{I}), \tag{63}$$

$$\nabla_s \times \vec{W}(\vec{L}) = \sum_{\vec{I}} \vec{C}_C(\vec{L} - \vec{I}) \times \vec{W}(\vec{I}), \tag{64}$$

$$\vec{C}_G(\vec{L}) = \frac{\partial}{\partial V} \int\int_{\partial B} \hat{n} g(\vec{L} + \vec{r}'/\Delta_g + \vec{R}) \, ds. \tag{65}$$

Eqs. (62)–(64) are weighted sums over the volume. To find the coefficients, expand (65) as in (7), take the radial derivative, set the radius as per (27) with the condition $c\Delta_t/\Delta_g = 1/\sqrt{3}$ specified above, and numerically perform the surface integration. As was the case in Section 3.1, once one has the gradient coefficients, the coefficients are known for the equivalently staggered divergence and curl.

In practice the summations in (59)–(61) or (62)–(64) need to be carried out over the finite-sized model space, hence the reconstruction-based computation is an approximation of the calculation. The algorithm will not be exact, since the spherical derivatives are not exactly calculated. This also means that the dispersion relation (26) only approximately characterizes a reconstruction implementation of the spherical derivative algorithm.

4.4. Numerical results

Fig. 2 shows the results of our calculations for the electromagnetic resonator. The plots show the magnitude of the FFT vs. frequency of the sample data. The top plot shows the theoretical prediction given by (51). The middle plot is the result of a Yee algorithm simulation and the bottom plot is the spherical derivatives algorithm. The Yee algorithm exhibits mode-splitting, combining, and shuffling [11,9], due to the anisotropy of the Yee algorithm. The mode structure in the spherical algorithm is much better than the Yee algorithm, but at higher frequencies splitting is clearly visible.

5. Conclusions and observations

The spherical operators presented here provide for theoretically exact 3D time-domain differential equation solvers. The theory of exact 3D time-domain solvers can be used to guide the construction of practical algorithms. We have shown the theoretical existence of 3D time-domain solvers which can, in principle, be exact for any time-step size. While the volume algorithms provides very good performance to the sampling limit, the $O[n^2]$ per axis cost is far too computationally expensive to be practical for much larger model spaces than the small resonator used here. Nonetheless, we have a theoretical framework for exact algorithms with a "magic" time step in 3D, which can be used to guide the construction of practical algorithms.

A well known method to improve performance is to use fast transform techniques. Such methods would still use the full space, i.e., the operators would remain global. Alternatively, preliminary testing suggests, that practical algorithms can be made with some set of only local neighbor nodes contributing to the operators. Such local operators are much less computationally expensive than global reconstruction. The range of the operator could be increased or decreased to improve accuracy or speed. A method to reduce the needed computational range of the volume operators is to use reconstructors with faster spatial falloff than the Shannon reconstructors. This can be accomplished by modifying the rectangular spectrum of the

Shannon reconstructor to have a slower falloff in the spectrum of the reconstructor – since a slower spectral falloff will produce a faster falloff in the reconstructor. The purpose of a localized reconstructor is to retain the same algorithmic cost as the Yee algorithm, albeit with a much larger constant cost factor.

Our test implementations show potential of the method, but an efficient implementation that exploits this theory has not yet been obtained. Our tests demonstrate solvers for both acoustics (not included here) and Maxwell's equations, with good results.

Acknowledgments

This work was supported primarily by the Harold and Diana Frank EECS graduate fellowship at Washington State University, by private funds, and by the Office of Naval Research code 321OA.

References

[1] K.S. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, IEEE Transactions on Antennas and Propagation 14 (3) (1966) 302–307.
[2] A. Taflove, S. Hagness, Computational Electrodynamics: The Finite-Difference Time-Domain Method, second ed., Artech House, Boston, MA, 2000.
[3] A.I. Boresenko, I.E. Tarapov, Vector and Tensor Analysis with Applications, Revised English ed., Prentice Hall, Englewocd Cliffs, NJ, 1968.
[4] J. von Neumann, R.D. Richtmyer, On the numerical solutions of partial differential equations of parabolic type, in: A.H. Taub (Ed.), John von Neumann Collected Works, vol. V, Pergamon Press, Oxford, 1963, pp. 652–663, los Alamos Report LA-657, December 25, 1947.
[5] J.B. Schneider, C.L. Wagner, FDTD dispersion revisited: faster-than-light propagation, IEEE Microwave Guided Wive Letters 9 (2) (1999) 54–56.
[6] A. Taflove, M.E. Brodwin, Numerical solution of steady-state electromagnetic scattering problems using the time-dependent Maxwell's equations, IEEE Transactions on Microwave Theory and Techniques MTT-23 (8) (1975) 623–630.
[7] J.A. Stratton, Electromagnetic Theory, McGraw-Hill, New York, 1941.
[8] M. Abramowitz, I.E. Stegun, Handbook of Mathematical Functions, Nat. Bur. Standards, 1985.
[9] C.L. Wagner, J.B. Schneider, On the analysis of resonators using finite-difference time-domain techniques, IEEE Transactions on Antennas a Propagation 51 (10) (2003) 2885–2890.
[10] Q.H. Liu, The pseudo-spectral time-domain (PSTD) method: a new algorithm for solutions of Maxwell's equations, in: IEEE Antennas and Propagation Society International Symposium, vol. 1, Montreal, Canada, 1997, pp. 122–125.
[11] C.L. Wagner, J.B. Schneider, Using the dispersion relation to understand finite-difference time-domain worlds, in: International Conference on Electromagnetics in Advanced Applications (ICEAA 01), Torino, Italy, 2001, pp. 375–378.