One-step finite-difference time-domain algorithm to solve the Maxwell equations

H. De Raedt,* K. Michielsen,† J. S. Kole,‡ and M. T. Figge§
Applied Physics–Computational Physics, Materials Science Centre, University of Groningen, Nijenborgh 4 NL-9747 AG Groningen, The Netherlands

(Received 15 August 2002; revised manuscript received 20 December 2002; published 19 May 2003)

We present a one-step algorithm to solve the time-dependent Maxwell equations for systems with spatially varying permittivity and permeability. We compare the results of this algorithm with those obtained from the Yee algorithm and from unconditionally stable algorithms. We demonstrate that for a range of applications the one-step algorithm may be orders of magnitude more efficient than multiple time-step, finite-difference time-domain algorithms. We discuss both the virtues and limitations of this one-step approach.

DOI: 10.1103/PhysRevE.67.056706 PACS number(s): 02.60.Cb, 03.50.De, 41.20.Jb

I. INTRODUCTION

Many applications in physics and engineering require numerical methods to solve the time-dependent Maxwell equations [1–10]. A popular approach is the finite-difference time-domain (FDTD) method [2–4] based on a proposal by Yee [1]. It is flexible, fast, and easy to implement. A limitation of Yee-based FDTD techniques is that their stability is conditional, depending on the mesh size of the spatial discretization and the time step of the time integration [2].

Recently we have introduced a family of unconditionally stable algorithms to solve the time-dependent Maxwell equations [9,10]. The operator that governs the time evolution of the electromagnetic (em) fields is orthogonal and can be written as the matrix exponential of a skew-symmetric matrix [9]. Orthogonal approximations to the time-evolution operator yield unconditionally stable algorithms by construction [11]. Details of the construction of such algorithms can be found elsewhere [9,10].

A limitation of both the Yee-based [2] and our unconditionally stable algorithms [9,10] is that the amount of computational work required to propagate the em fields for long times may be prohibitive for a class of important applications, such as bioelectromagnetics and very large scale integrated design [2,12,13]. The basic reason for this is that in order to maintain a reasonable degree of accuracy during the time integration, the time step has to be relatively small.

A well-known alternative to time stepping is to use Chebyshev polynomials to construct approximations to time-evolution operators [14–20]. In this paper we make use of these rapidly converging polynomial approximations to construct a one-step algorithm that solves the time-dependent Maxwell equations. We demonstrate that the one-step algorithm can be orders of magnitude more efficient than current FDTD algorithms.

The paper is organized as follows. In Sec. II we recall some basic, essential facts about the mathematical structure of the Maxwell equations that we need in Sec. III to turn the Chebyshev polynomial expansion method into a one-step algorithm for solving the Maxwell equations. In Sec. IV we illustrate this approach by considering a one-dimensional system with a current source and present results of an error-efficiency analysis. Results of numerical experiments on a three-dimensional system as well as an error-efficiency analysis are presented in Sec. V. A summary and our conclusions are given in Sec. VI.

II. THEORY

We consider em fields in linear, isotropic, nondispersive, and lossless materials. Generalizations are discussed below. In the absence of electric charges, the time evolution of the em fields in these materials is governed by the time-dependent Maxwell equations (in mks units) [2]

$$\frac{\partial}{\partial t}\mathbf{H}=-\frac{1}{\mu}\nabla\times\mathbf{E},\quad \frac{\partial}{\partial t}\mathbf{E}=\frac{1}{\varepsilon}(\nabla\times\mathbf{H}-\mathbf{J}),\tag{1}$$

$$\nabla\cdot(\mu\mathbf{H})=0,\quad \nabla\cdot(\varepsilon\mathbf{E})=0,\tag{2}$$

where $\mathbf{H}=(H_x(\mathbf{r},t),H_y(\mathbf{r},t),H_z(\mathbf{r},t))^T$ and $\mathbf{E}=(E_x(\mathbf{r},t),E_y(\mathbf{r},t),E_z(\mathbf{r},t))^T$ denote the magnetic and the electric field vector, respectively. The source of the electric field is represented by $\mathbf{J}=(J_x(\mathbf{r},t),J_y(\mathbf{r},t),J_z(\mathbf{r},t))^T$. The permeability and the permittivity are given by $\mu=\mu(\mathbf{r})$ and $\varepsilon=\varepsilon(\mathbf{r})$. For simplicity of notation, we will omit the temporal and the spatial dependence on $\mathbf{r}=(x,y,z)^T$ unless this leads to ambiguities. For numerical purposes it is expedient to introduce dimensionless quantities. The velocity of light in vacuum is given by $c=1/\sqrt{\varepsilon_0\mu_0}$ (in mks units), where ε_0 denotes the permittivity and μ_0 the permeability in vacuum. If we measure distances in units of the wavelength λ, time and frequency are expressed in units of λ/c and c/λ, respectively. Then Eqs. (1) and (2) take a dimensionless form if we replace $\varepsilon(\mu)$ by its value relative to ε_0 (μ_0) and express $\mathbf{H}$ and $\mathbf{E}$ in units of A/m and V/m, respectively. We adopt this dimensionless form, in other words, from now on $\mathbf{H}$, $\mathbf{E}$, ε, μ, t, and $\mathbf{r}$ are dimensionless quantities.

In a simulation there is a limit to the size of the box surrounding the material of interest. On the surface of this simulation box, the EM fields are assumed to satisfy the boundary conditions [21]

*Email address: deraedt@phys.rug.nl
†Email address: kristel@phys.rug.nl
‡Email address: j.s.kole@phys.rug.nl
§Email address: m.t.figge@phys.rug.nl

$$\mathbf{n}\times\mathbf{E}=\mathbf{0}, \quad \mathbf{n}\cdot\mathbf{H}=0, \tag{3}$$

with $\mathbf{n}$ denoting the vector normal to a boundary of the surface. Conditions (3) assure that the normal component of the magnetic field and the tangential components of the electric field vanish at the boundary [21].

Some important physical symmetries of the Maxwell equations (1) and (2) can be made explicit by introducing the fields

$$\mathbf{X}(t)=\sqrt{\mu}\mathbf{H}(t), \quad \mathbf{Y}(t)=\sqrt{\varepsilon}\mathbf{E}(t). \tag{4}$$

In terms of the fields $\mathbf{X}(t)$ and $\mathbf{Y}(t)$, Maxwell's curl equations (1) read

$$\frac{\partial}{\partial t}\begin{pmatrix}\mathbf{X}(t)\\ \mathbf{Y}(t)\end{pmatrix}=\begin{pmatrix}-\frac{1}{\sqrt{\mu}}\boldsymbol{\nabla}\times\frac{\mathbf{Y}(t)}{\sqrt{\varepsilon}}\\ \frac{1}{\sqrt{\varepsilon}}\boldsymbol{\nabla}\times\frac{\mathbf{X}(t)}{\sqrt{\mu}}\end{pmatrix}-\begin{pmatrix}0\\ \frac{\mathbf{J}(t)}{\sqrt{\varepsilon}}\end{pmatrix},$$

$$=\mathcal{H}\begin{pmatrix}\mathbf{X}(t)\\ \mathbf{Y}(t)\end{pmatrix}-\begin{pmatrix}0\\ \frac{\mathbf{J}(t)}{\sqrt{\varepsilon}}\end{pmatrix}, \tag{5}$$

where

$$\mathcal{H}=\begin{pmatrix}0 & -\frac{1}{\sqrt{\mu}}\boldsymbol{\nabla}\times\frac{1}{\sqrt{\varepsilon}}\\ \frac{1}{\sqrt{\varepsilon}}\boldsymbol{\nabla}\times\frac{1}{\sqrt{\mu}} & 0\end{pmatrix}. \tag{6}$$

Writing $\mathbf{Z}(t)=(\mathbf{X}(t),\mathbf{Y}(t))^T$ it is easy to show that $\mathcal{H}$ is skew symmetric, i.e., $\mathcal{H}^T=-\mathcal{H}$, with respect to the inner product $\langle\mathbf{Z}|\mathbf{Z}'\rangle\equiv\int_V\mathbf{Z}^T\cdot\mathbf{Z}'\,d\mathbf{r}$, where V denotes the volume of the enclosing box. By construction $\|\mathbf{Z}(t)\|^2=\langle\mathbf{Z}(t)|\mathbf{Z}(t)\rangle=\int_V[\varepsilon\mathbf{E}^2(t)+\mu\mathbf{H}^2(t)]d\mathbf{r}$, relating the length of $\mathbf{Z}(t)$ to the energy density $w(t)\equiv\varepsilon\mathbf{E}^2(t)+\mu\mathbf{H}^2(t)$ of the em fields [21]. The formal solution of Eq. (5) can be written as

$$\mathbf{Z}(t)=\begin{pmatrix}\mathbf{X}(t)\\ \mathbf{Y}(t)\end{pmatrix}=e^{t\mathcal{H}}\left[\begin{pmatrix}\mathbf{X}(0)\\ \mathbf{Y}(0)\end{pmatrix}-\int_0^t e^{-u\mathcal{H}}\begin{pmatrix}0\\ \frac{\mathbf{J}(u)}{\sqrt{\varepsilon}}\end{pmatrix}du\right]. \tag{7}$$

From Eq. (7) it is clear that the operator $\mathcal{U}(t)\equiv e^{t\mathcal{H}}$ governs the time evolution of the em fields. As $\mathcal{U}(t)^T=\mathcal{U}(-t)=\mathcal{U}^{-1}(t)=e^{-t\mathcal{H}}$, the time-evolution operator $\mathcal{U}(t)$ is an orthogonal transformation that rotates the vector $\mathbf{Z}(t)$ without changing the length of $\mathbf{Z}(t)$. In physical terms this means that if $\mathbf{J}(t)=\mathbf{0}$, the energy density of the em fields does not change with time, as expected on physical grounds [21].

III. TIME-INTEGRATION ALGORITHM

A numerical algorithm that solves the time-dependent Maxwell equations necessarily involves some discretization procedure of the spatial derivatives. This procedure maps continuum space onto a lattice, i.e., it maps the differential operator $\mathcal{H}$ onto a matrix $\mathcal{L}$. The time-evolution matrix and the vector of the em fields on the lattice will be denoted by $U(t)=e^{t\mathcal{L}}$ and $\boldsymbol{\Psi}(t)$, respectively. The matrix equivalent of Eq. (5) reads

$$\frac{\partial}{\partial t}\boldsymbol{\Psi}(t)=\mathcal{L}\boldsymbol{\Psi}(t)-\boldsymbol{\Phi}(t), \tag{8}$$

where $\boldsymbol{\Phi}(t)$ is the vector that represents the current source. The formal solution of Eq. (8) is given by

$$\boldsymbol{\Psi}(t)=e^{t\mathcal{L}}\boldsymbol{\Psi}(0)-\int_0^t e^{(t-u)\mathcal{L}}\boldsymbol{\Phi}(u)du. \tag{9}$$

Ideally, the mapping from a continuum to a lattice problem should not change the basic symmetries of the Maxwell equations. The underlying symmetry of the time-dependent Maxwell equations suggests using matrices $\mathcal{L}$ that are real and skew symmetric. The discretization procedure itself is not essential for what follows as long as $\mathcal{L}$ is skew symmetric (generalizations are being discussed below). Therefore, in this paper, we do not discuss the (important) technicalities related to the spatial discretization and refer the reader to Ref. [9].

The next step is to choose an algorithm to perform the time integration for the time-dependent Maxwell equations defined on the grid. In general, this amounts to approximating the matrix exponential $U(t)=e^{t\mathcal{L}}$ by a time-evolution matrix $\tilde{U}(t)$. The corresponding approximate solution will be denoted by $\tilde{\boldsymbol{\Psi}}(t)$. If the approximation $\tilde{U}(t)$ is itself an orthogonal transformation, then $\|\tilde{U}(t)\|=1$, where $\|X\|$ denotes the two-norm of a vector or matrix X [22]. The fact that $\tilde{U}(t)$ is an orthogonal transformation is essential for the development of an unconditionally stable algorithm to solve the Maxwell equations [9]. In the absence of source terms [i.e., $\boldsymbol{\Phi}(t)=\mathbf{0}$], this implies that $\|\tilde{\boldsymbol{\Psi}}(t)\|=\|\tilde{U}(t)\boldsymbol{\Psi}(0)\|=\|\boldsymbol{\Psi}(0)\|$, for an arbitrary initial condition $\boldsymbol{\Psi}(0)$ and for all times t and hence the time-integration algorithm defined by $\tilde{U}(t)$ is unconditionally stable by construction [23,11]. In the presence of current sources, for general $\tilde{U}(t)$, it follows immediately from Eq. (9) that

$$\|\tilde{\boldsymbol{\Psi}}(t)\|\leq\|\boldsymbol{\Psi}(t)\|+\tilde{\epsilon}\left(1+\int_0^t\|\boldsymbol{\Phi}(u)\|du\right), \tag{10}$$

where $\|\tilde{U}(u)-U(u)\|\leq\tilde{\epsilon}$ for $0\leq u\leq t$ and $\tilde{\epsilon}$ is a measure for the accuracy of the approximation $\tilde{U}(t)$. In a strict sense, the one-step method we describe below does not correspond to an orthogonal approximation. However, for practical purposes it can be viewed as an extremely stable time-integration algorithm because it yields an approximation to the exact time-evolution operator $U(t)=e^{t\mathcal{L}}$ that is exact to nearly machine precision, i.e., in practice the value of $\tilde{\epsilon}$ in Eq. (10) is very small. This also implies that within the same

precision $\nabla\cdot(\mu\mathbf{H}(t))=\nabla\cdot(\mu\mathbf{H}(t=0))$ and $\nabla\cdot(\varepsilon\mathbf{E}(t))=\nabla\cdot(\varepsilon\mathbf{E}(t=0))$, i.e., Eq. (2) holds for all times.

A. Case: $\mathbf{J}(t)=0$

We first recall how the Chebyshev polynomial approach is used to approximate $U(t)=e^{t\mathcal{L}}$ and then show how to treat the source term. We begin by ''normalizing'' the matrix $\mathcal{L}$. The eigenvalues of the skew-symmetric matrix $\mathcal{L}$ are pure imaginary numbers. Hence the eigenvalues of the Hermitian matrix $A=-i\mathcal{L}$ are real and if a is one of these eigenvalues so is $-a$. The eigenvalues of A lie in the interval $[-\|\mathcal{L}\|_2,\|\mathcal{L}\|_2]$, where $\|\mathcal{L}\|_2$ is the largest (in absolute value) eigenvalue of $\mathcal{L}$ [22]. Obviously $\|\mathcal{L}\|_2$ is hard to find. However, for our purposes we only need an upper bound to $\|\mathcal{L}\|_2$. Since $\mathcal{L}$ is sparse it is easy to compute $\|\mathcal{L}\|_1\equiv\max_j\Sigma_i|\mathcal{L}_{i,j}|$ and the upper bound follows from $\|\mathcal{L}\|_2\leqslant\|\mathcal{L}\|_1$ [22]. By construction the eigenvalues of $B\equiv-i\mathcal{L}/\|\mathcal{L}\|_1$ all lie in the interval $[-1,1]$. The time-evolution operator then reads $U(t)=e^{t\mathcal{L}}=e^{izB}$, where $z=t\|\mathcal{L}\|_1$. In practice, it is easy to determine $\|\mathcal{L}\|_1$ by hand. If $\epsilon=\mu=1$, in the case of a three-point central-difference approximation to the spatial dervatives, $\|\mathcal{L}\|_1=2/\delta$ for a one-dimensional grid and $\|\mathcal{L}\|_1=4/\delta$ for a two- and three-dimensional grid, δ being the mesh size.

Expanding the initial value $\mathbf{\Psi}(0)$ in the (unknown) eigenvectors $\mathbf{b}_j$ of B we have

$$U(t)\mathbf{\Psi}(0)=e^{t\mathcal{L}}\mathbf{\Psi}(0)=e^{it\|\mathcal{L}\|_1(-i\mathcal{L}/\|\mathcal{L}\|_1)}\mathbf{\Psi}(0)=e^{izB}\mathbf{\Psi}(0)$$
$$=\sum_j e^{izb_j}\mathbf{b}_j\langle\mathbf{b}_j|\mathbf{\Psi}(0)\rangle, \tag{11}$$

where the b_j denote the (unknown) eigenvalues of B. Actually, we will only make use of the fact that $-1\leqslant b_j\leqslant 1$. We find the Chebyshev polynomial expansion of $U(t)$ by computing the expansion coefficients of each of the functions e^{izb_j} that appear in Eq. (11). In general, for $-1\leqslant x\leqslant 1$ we can represent a function $f(x)$ as

$$f(x)=\tfrac{1}{2}a_0T_0(x)+a_1T_1(x)+a_2T_2(x)+\cdots, \tag{12}$$

where $T_k(x)=\cos(k\arccos x)$ is the Chebyshev polynomial of the first kind of order k [24]. The expansion coefficients a_k can be found by computing

$$a_k=\frac{2}{\pi}\int_{-1}^{+1}\frac{f(x)T_k(x)}{\sqrt{1-x^2}}dx=\frac{2}{\pi}\int_0^{\pi}f(\cos\theta)\cos k\theta d\theta. \tag{13}$$

From Eq. (13) it is clear that, in practice, the coefficients a_k can be calculated by Fourier transformation of $f(\cos\theta)$.

Using representation [24]

$$e^{izx}=J_0(z)+2\sum_{k=1}^{\infty}i^kJ_k(z)T_k(x), \tag{14}$$

where $J_k(z)$ is the Bessel function of integer order k, we obtain

$$U(t)\mathbf{\Psi}(0)=e^{t\mathcal{L}}\mathbf{\Psi}(0)=\Big[J_0(t\|\mathcal{L}\|_1)I+2\sum_{k=1}^{\infty}J_k(t\|\mathcal{L}\|_1)\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\Big]\mathbf{\Psi}(0). \tag{15}$$

Here I is the identity matrix and $\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)$ is a matrix-valued modified Chebyshev polynomial that is defined by the recursion relation

$$\tilde{T}_0(\mathcal{L}/\|\mathcal{L}\|_1)\mathbf{\Psi}(0)=\mathbf{\Psi}(0),\quad \tilde{T}_1(\mathcal{L}/\|\mathcal{L}\|_1)\mathbf{\Psi}(0)=\mathcal{L}\|\mathcal{L}\|_1^{-1}\mathbf{\Psi}(0), \tag{16}$$

$$\tilde{T}_{k+1}(\mathcal{L}/\|\mathcal{L}\|_1)\mathbf{\Psi}(0)=2\mathcal{L}\|\mathcal{L}\|_1^{-1}\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\mathbf{\Psi}(0)+\tilde{T}_{k-1}(\mathcal{L}/\|\mathcal{L}\|_1)\mathbf{\Psi}(0)\quad\text{for}\quad k\geqslant 1. \tag{17}$$

From Eqs. (16) and (17) it is clear that $\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\mathbf{\Psi}(0)$ is real valued, as it should be in the case of the Maxwell equations. Thus, in an actual implementation of the algorithm there is no need to perform complex arithmetic.

In practice, we will have to truncate the sum in Eq. (15), i.e., we will use only the first $K+1$ contributions to approximate $U(t)\mathbf{\Psi}(0)$:

$$e^{t\mathcal{L}}\mathbf{\Psi}(0)\approx\Big[J_0(t\|\mathcal{L}\|_1)I+2\sum_{k=1}^{K}J_k(t\|\mathcal{L}\|_1)\times\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\Big]\mathbf{\Psi}(0). \tag{18}$$

To determine the value of K it is instructive to study $J_k(z)$ as a function of k. Using the downward recursion relation of the Bessel functions, we can compute K Bessel functions to machine precision using only of the order of K arithmetic operations [24,25]. In practice, a calculation of the first 20 000 Bessel functions takes less than 1 s on a Pentium III 600-MHz mobile processor, using 14-15 digit arithmetic. Hence this part of a calculation is a negligible fraction of the total computational work for solving the Maxwell equations.

From Fig. 1 it is clear that $|J_k(z)|$ vanishes rapidly if k becomes larger than z. For instance, $|J_k(z=2000)|<10^{-10}$ for all $k>z+100$. Thus we may fix the number K by requiring that $|J_k(t\|\mathcal{L}\|_1)|>\kappa$ for all $k\leqslant K$. Here κ is a small number that determines the accuracy of the approximation. In our numerical experiments we use conventional 14-15 digit floating-point arithmetic and we have taken $\kappa=10^{-10}$. Once

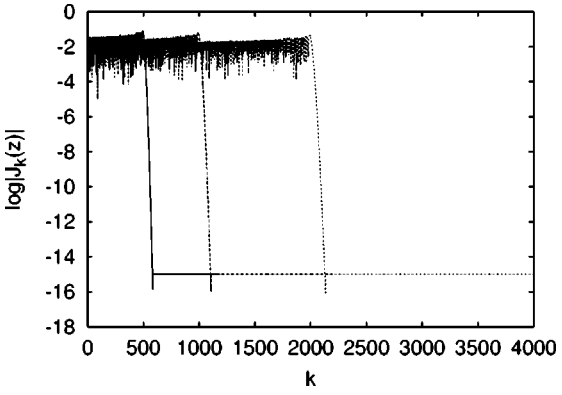

FIG. 1. Dependence of the Bessel function $J_k(z)$ on the order k. Solid line, $z=500$; dashed line, $z=1000$; dotted line, $z=2000$.

we have found the smallest K such that $|J_k(t\|\mathcal{L}\|_1)|>\kappa$ for all $k\leqslant K$, there is no point in taking more than K terms in the expansion. Indeed, since $\|\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\|\leqslant 1$ by construction of the modified Chebyshev polynomials, it follows from Fig. 1 that such contributions would only add to the noise. However, taking less than z terms has considerable negative impact on the accuracy of the results. Hence in practice the choice of K is rather limited (e.g., $K\in[z,z+100]$ if $z=2000$). In any case, for fixed κ, K increases linearly with $t\|\mathcal{L}\|_1$.

At this point one may wonder why it would not be simpler to use the Taylor series instead of the Chebyshev polynomial expansion. There are two reasons for not doing this. The first is the accuracy of the polynomial expansion. From the derivation of Eq. (18), it is clear that we use the Chebyshev polynomial expansion to approximate $\exp(izb_j)$ for *each* of the (unknown) eigenvalues b_j. After rescaling, the values of b_j enter through the variable $x\in[-1,1]$. As shown in Fig. 2, the Chebyshev polynomial expansion with $K=50$ is a very good approximation to $\exp(izx)$, for all relevant values of x. The corresponding Taylor series approximation, using the same number of terms $K=50$, does very well for $-0.25<x<0.25$ but performs much worse than the Chebyshev polynomial expansion for other values of x. Doubling the number of Taylor expansion terms helps to reduce the error but not to the same level as for the Chebyshev polynomial expansion.

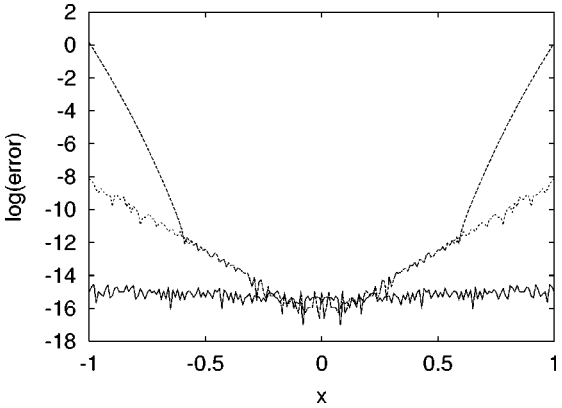

FIG. 2. Error between e^{izx} and the Chebyshev approximation to e^{izx} and two Taylor series approximations to e^{izx}, as a function of x for $z=20$. Solid line, $\log_{10}|e^{izx}-[J_0(z)+2\Sigma_{k=1}^{K}J_k(z)\tilde{T}_k(x)]|$ for $K=50$; dashed line, $\log_{10}|e^{izx}-\Sigma_{k=0}^{K}(izx)^k/k!|$ for $K=50$; dotted line, $\log_{10}|e^{izx}-\Sigma_{k=0}^{K}(izx)^k/k!|$ for $K=100$.

The fact that the truncated Taylor polynomial is less accurate for eigenvalues of large modulus also contributes to the second and main reason for not using the Taylor series to approximate $\exp(t\mathcal{L})$: Numerical instability [23]. A clear signal of this phenomenon can already be seen in Fig. 2. For $-1\leqslant x\leqslant 1$, $z=20$ and $K=50(100)$ we have $|e^{izx}-\Sigma_{k=0}^{K}(izx)^k/k!|\leqslant|z|^{K+1}/(K+1)!\approx 10^{0.16}(10^{-29})$. The upper bound for $K=100$ tells us that we should have no confidence in the numerical results for $K=100$ (dotted line) shown in Fig. 2. The reason that the numerical error for $|x|=1$ is much larger than the theoretical upper bound stems from the fact that we have performed these numerical calculations using 14-15 digit arithmetic and, most importantly, that summing the Taylor series for $xz=20$ is a numerically unstable procedure. This is most easily verified by repeating the same calculation with 32-33 digit arithmetic. Then for $xz=20$, the numerical error of the $K=100$ series is 10^{-29} (results not shown), in agreement with the theoretical upper bound.

In the Chebyshev approach we have $|J_k(z)|<1$ (recall z is real) and $\|\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\|\leqslant 1$ and therefore all contributions in Eq. (18) are roughly of the same magnitude. In contrast, to sum the Taylor series we must compute $\Sigma_{k=1}^{K}(t\mathcal{L})^k/k!$, which involves adding many small and large (real) numbers, a numerical task that can be very difficult and often results in numerical instabilities. Examples of this phenomenon for the case of the Maxwell equations are given in the Appendix.

According to Eq. (18), performing one time step amounts to repeatedly using recursion relation (17) to obtain $\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\boldsymbol{\Psi}(0)$ for $k=2,\ldots,K$, then multiplying the elements of this vector by $J_k(t\|\mathcal{L}\|_1)$ and adding all contributions. This procedure requires storage for two vectors of the same length as $\boldsymbol{\Psi}(0)$ and some code to multiply such a vector by the sparse matrix $\mathcal{L}$. For $\boldsymbol{\Phi}(t)=\mathbf{0}$, bound (10) gives

$$\|\tilde{\boldsymbol{\Psi}}(t)\|\leqslant\|\boldsymbol{\Psi}(t)\|+\epsilon(K), \tag{19}$$

where $\epsilon(K)$ denotes the error bound on the truncated Chebyshev polynomial expansions of $U(t)=e^{t\mathcal{L}}$. As $\epsilon(K)$ can be made (exponentially) small by increasing K, bound (19) suggests that in practice, the one-step algorithm may safely be used repeatedly to perform multiple time steps with a (very) large fixed time step. At most the error will simply be $\epsilon(K)$ times the number of time steps.

B. Case: $\mathbf{J}(t)\neq\mathbf{0}$

We now turn to the treatment of the source term and focus on the case where the time dependence of the source term is known explicitly. One approach might be to simply use the Chebyshev expansion for $e^{(t-u)\mathcal{L}}$ and perform the integral in Eq. (9) numerically. However, this approach is not efficient, as for each value of $t-u$ we would have to perform a recur-

sion of the kind of Eq. (17). Thus we take another route that we illustrate by considering a sinusoidal source [26]

$$\mathbf{J}(\mathbf{r},t)=\Theta(T-t)\mathbf{s}(\mathbf{r})\sin(\Omega t), \tag{20}$$

where $\mathbf{s}(\mathbf{r})$ specifies the spatial distribution of the source and Ω is the angular frequency of the current source. The source is turned on at $t=0$ and is switched off at $t=T$, as indicated by the presence of the step function $\Theta(T-t)$ in Eq. (20). Artifacts that result from the discontinuity at $t=T$ can be minimized by choosing T such that $\Omega T/2\pi$ is an integer number.

The contribution of the source term to the em field at time t is given by the last term of Eq. (9). For the sinusoidal source described by Eq. (20) the formal expression of this contribution reads

$$\begin{aligned}\int_0^t e^{(t-u)\mathcal{L}}\mathbf{\Phi}(u)du &= (\Omega^2+\mathcal{L}^2)^{-1}e^{(t-T')\mathcal{L}}\\ &\times(\Omega e^{T'\mathcal{L}}-\Omega\cos\Omega T'-\mathcal{L}\sin\Omega T')\Xi\\ &\equiv f(\mathcal{L},t,T',\Omega)\Xi,\end{aligned} \tag{21}$$

where Ξ denotes the vector representing the spatial distribution $\mathbf{s}(\mathbf{r})$ of the source and $T'=\min(t,T)$. The expansion coefficients of the Chebyshev polynomial approximation of the time-evolution operator in Eq. (21) may be calculated as follows. First we repeat the scaling procedure described above. Then we substitute in Eq. (21) $\mathcal{L}=ix\|\mathcal{L}\|_1$, $t=z/\|\mathcal{L}\|_1$, $T'=Z'/\|\mathcal{L}\|_1$, and $\Omega=\omega\|\mathcal{L}\|_1$ and compute the Fourier expansion coefficients (i.e., the coefficients of the Chebyshev polynomial expansion) of the function

$$\begin{aligned}f(x,z,Z',\omega)=\frac{Z'}{2\|\mathcal{L}\|_1}e^{ix(z-Z')}\Bigg\{&\omega Z'\frac{\sin[(\omega+x)Z'/2]}{(\omega+x)Z'/2}\\ &\times\frac{\sin[(\omega-x)Z'/2]}{(\omega-x)Z'/2}+i\Bigg[\frac{\sin[(\omega+x)Z'/2]}{(\omega+x)Z'/2}\\ &\times\cos[(\omega-x)Z'/2]-\frac{\sin[(\omega-x)Z'/2]}{(\omega-x)Z'/2}\\ &\times\cos[(\omega+x)Z'/2]\Bigg]\Bigg\}.\end{aligned} \tag{22}$$

We have written Eq. (22) in a form that emphasizes that $f(x,z,Z',\omega)$ has no singularities as a function of $-1\leqslant x\leqslant 1$.

According to the general theory, the coefficients of the formal expansion of $f(\mathcal{L},t,T',\Omega)$ in terms of Chebyshev polynomials are given by Eq. (13), i.e., by the Fourier cosine transform of $f(\cos\theta,z,Z',\omega)$ with respect to θ. In practice the expansion coefficients $S_k(t\|\mathcal{L}\|_1)$ are most easily calculated by performing a fast Fourier transform (FFT) [20]. For the sinusoidal source, Eq. (20), we have

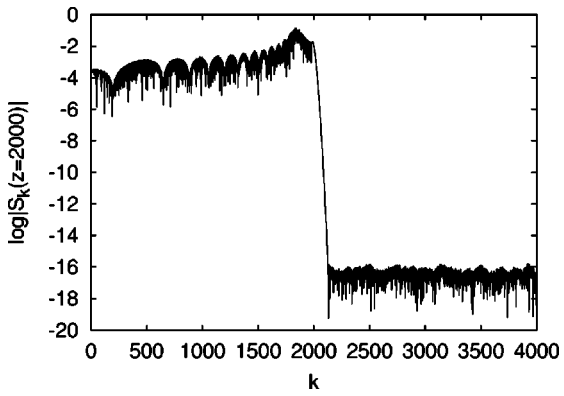

FIG. 3. Dependence of the expansion coefficients $S_k(z=2000)$ on the order k.

$$S_k(t\|\mathcal{L}\|_1)=i^{-k}\sum_{n=0}^{N-1}e^{2\pi ink/N}f\left(\cos\frac{2\pi n}{N},z,Z',\omega\right), \tag{23}$$

where N is the number of points in the FFT. The symmetry of $f(\cos(2\pi n/N),z,Z',\omega)$ guarantees that the coefficients $S_k(t\|\mathcal{L}\|_1)$ are strictly real.

The number N serves two purposes. First N has to be chosen such that the sum over n in Eq. (23) yields an accurate (i.e., better than κ) approximation to the integral in Eq. (13), for all relevant k. Second, since we have to truncate the series at K', N has to be sufficiently large so that we can find K' for which $|S_k|<\kappa$ for all $k>K'$. In our numerical experiments K' is in the range 1000–10 000 and it is the first condition that determines the value of N. In Fig. 3, we show $\ln|S_k(z=2000)|$ as a function of k obtained by using $N=2^{22}$ points in the FFT (this calculation takes less than 32 s on a Pentium III 600-MHz mobile processor, using 14-15 digit arithmetic). From Fig. 3 it is clear that for $z=2000$ we can truncate the series if K' is a little larger than z. Thus, as in the case where the expansion coefficients correspond to the Bessel functions, we fix the number K' by requiring that $|S_k(t\|\mathcal{L}\|_1)|>\kappa$ for all $k\leqslant K'$. The results for $N=2^{20},2^{21},2^{22}$ differ in the noisy, irrelevant part ($n>z+100$) only. The noise level can be reduced further by increasing N.

Putting all pieces together, the one-step algorithm to compute the em field at time t is given by

$$\begin{aligned}\mathbf{\Psi}(t)\approx&\left[J_0(t\|\mathcal{L}\|_1)I+2\sum_{k=1}^{K}J_k(t\|\mathcal{L}\|_1)\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\right]\mathbf{\Psi}(0)\\ &-\left[S_0(t\|\mathcal{L}\|_1)I+2\sum_{k=1}^{K'}S_k(t\|\mathcal{L}\|_1)\tilde{T}_k(\mathcal{L}/\|\mathcal{L}\|_1)\right]\Xi.\end{aligned} \tag{24}$$

The numerical procedure to compute the contribution of both terms in Eq. (24) is the same and involves real numbers only.

In principle, sources with a more complicated time dependence $G(t)$ can be synthesized by computing $\int_{-\infty}^{\infty}g(\omega)f(x,z,Z',\omega)d\omega$, where $g(\omega)$ is the Fourier sine transform of $G(t)$ but this requires two nested Fourier trans-

forms and may result in a substantial computational cost. Another option is to choose the form of the current pulse such that the integral in Eq. (21) can be worked out analytically. For instance, for the Gaussian pulsed source defined by [2,3]

$$\mathbf{J}(\mathbf{r},t)=\mathbf{s}(\mathbf{r})e^{-\alpha(t-t_0)^2}, \tag{25}$$

the formal expression for the contribution of the source reads

$$\begin{aligned}\int_0^t e^{(t-u)\mathcal{L}}\mathbf{\Phi}(u)du &= \sqrt{\frac{\pi}{4\alpha}}e^{(t-T')\mathcal{L}+\mathcal{L}^2/4\alpha}[\mathrm{erf}(\sqrt{\alpha}(t-t_0)\\ &\quad +\mathcal{L}/2\sqrt{\alpha})+\mathrm{erf}(\sqrt{\alpha}t_0-\mathcal{L}/2\sqrt{\alpha})]\mathbf{\Xi}\\ &\equiv p(\mathcal{L},t,t_0,\alpha)\mathbf{\Xi}.\end{aligned} \tag{26}$$

As in the case of the sinusoidal source, also for the Gaussian pulsed source the coefficients in the Chebyshev polynomial expansion can be calculated by the FFT of $p(x,z,z_0,\alpha)$.

We end this section by making some general comments on the one-step approach. First, it is important to note that in this approach the time dependence of the source is taken into account without actually sampling it as a function of time (see the example in Sec. IV). Furthermore, the treatment of the source term presented above trivially applies to the scattered-field formulation [2,3]. In the more general case where there are electrical or magnetic losses in the system, the matrix $\mathcal{L}$ is no longer skew symmetric but still normal. Then $\mathcal{L}$ still has a complete set of eigenvectors but (some of) the eigenvalues of $\mathcal{L}$ may have a nonzero real part. We leave for future research the problem of determining the conditions under which the series expansion (15) converges sufficiently fast for practical applications.

IV. ILLUSTRATIVE EXAMPLE

We consider a system, infinitely large in the y and z directions, for which $\varepsilon=1$ and $\mu=1$. Under these conditions, the Maxwell equations reduce to two independent sets of first-order differential equations [21]. The solutions to these sets are known as the transverse electric (TE) mode and the transverse magnetic (TM) mode [21]. As the equations of the TE and the TM mode only differ by a sign, we can restrict our considerations to the TM mode and obtain the result for the TE mode by reversing the time.

From Eq. (1) it follows that the magnetic field $H_y(x,t)$ and the electric field $E_z(x,t)$ of the TM mode in the one-dimensional (1D) cavity of length L are solutions of

$$\frac{\partial}{\partial t}H_y(x,t)=\frac{\partial}{\partial x}E_z(x,t), \tag{27}$$

$$\frac{\partial}{\partial t}E_z(x,t)=\frac{\partial}{\partial x}H_y(x,t)-J_z(x,t), \tag{28}$$

subject to the boundary condition $E_z(0,t)=E_z(L,t)=0$ [21]. Note that constraints (2) are automatically satisfied.

Following Yee [1], to discretize Eqs. (27) and (28), it is convenient to assign H_y to odd and E_z to even numbered lattice sites, as shown in Fig. 4. Using the second-order central-difference approximation to the first derivative with respect to x, we obtain

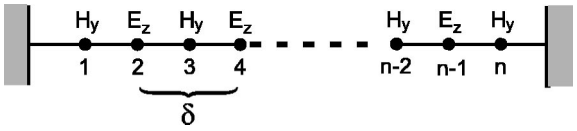

FIG. 4. Positions of the two TM-mode field components on the one-dimensional grid. The distance between two next-nearest neighbors is denoted by δ.

$$\frac{\partial}{\partial t}H_y(2i+1,t)=\delta^{-1}[E_z(2i+2,t)-E_z(2i,t)], \tag{29}$$

$$\frac{\partial}{\partial t}E_z(2i,t)=\delta^{-1}[H_y(2i+1,t)-H_y(2i-1,t)]-J_z(2i,t), \tag{30}$$

where we have introduced the notation $A(i,t)=A(x=i\delta/2,t)$. The integer i labels the grid points and δ denotes the distance between two next-nearest neighbors on the lattice (hence the absence of a factor 2 in the denominator). We define the n-dimensional vector $\mathbf{\Psi}(t)$ by

$$\mathbf{\Psi}(i,t)=\begin{cases}H_y(i,t), & i \text{ odd}\\ E_z(i,t), & i \text{ even.}\end{cases} \tag{31}$$

The vector $\mathbf{\Psi}(t)$ contains both the magnetic and the electric field on the lattice points $i=1,\ldots,n$. As usual, the ith element of $\mathbf{\Psi}(t)$ is given by the inner product $\mathbf{\Psi}(i,t)=\mathbf{e}_i^T\cdot\mathbf{\Psi}(t)$, where $\mathbf{e}_i$ denotes the ith unit vector in the n-dimensional vector space. Using this notation (which proves most useful for the case of two dimensions and three dimensions for which it is rather cumbersome to write down explicit matrix representations), it is easy to show that Eqs. (29) and (30) can be written in form (8) where the matrix $\mathcal{L}$ is given by

$$\mathcal{L}=\delta^{-1}\sum_{i=1}^{n-1}(\mathbf{e}_i\mathbf{e}_{i+1}^T-\mathbf{e}_{i+1}\mathbf{e}_i^T), \tag{32}$$

and we immediately see that $\mathcal{L}$ is skew symmetric by construction. Thus, we are in the position to apply the one-step algorithm to this problem.

First we briefly discuss the aspects that are relevant for the comparison among the Yee algorithm, the Suzuki-product-formula-based unconditionally stable algorithms [9,10], and the one-step approach. From Eq. (9) it follows that the em fields $\mathbf{\Psi}(t)$ change according to

$$\mathbf{\Psi}(t+\tau)=e^{\tau\mathcal{L}}\mathbf{\Psi}(t)-\int_t^{t+\tau}e^{(t+\tau-u)\mathcal{L}}\mathbf{\Phi}(u)du, \tag{33}$$

where $\mathbf{\Phi}(i,t)=J_z(i,t)$ if i is even and $\mathbf{\Phi}(i,t)=0$ otherwise. In practice we approximate the source term in Eq. (33) by a standard fourth-order quadrature formula [24] and obtain

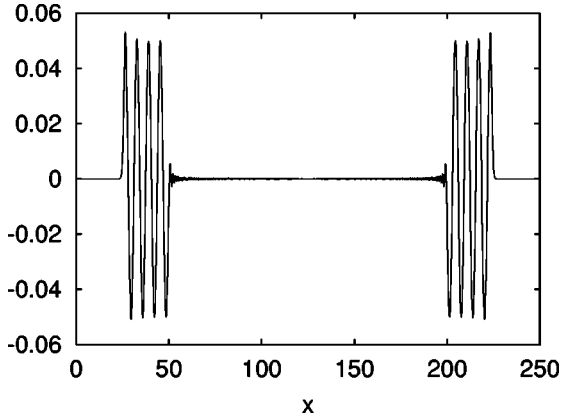

FIG. 5. The field $E_z(x,t=100)$ generated by a current source at $x=125$ that oscillates at frequency $\Omega=1$ during the interval $0\leqslant t\leqslant 4$, as obtained by the one-step algorithm (24) with $K'=2090$ ($K=0$ in this case).

$$\mathbf{\Psi}(t+\tau)=e^{\tau\mathcal{L}}\mathbf{\Psi}(t)+\frac{\tau}{6}[e^{\tau\mathcal{L}}\mathbf{\Phi}(t)+4e^{\tau\mathcal{L}/2}\mathbf{\Phi}(t+\tau/2)+\mathbf{\Phi}(t+\tau)]. \tag{34}$$

We replace $e^{\tau\mathcal{L}/2}$ and $e^{\tau\mathcal{L}}$ in Eq. (34) by an approximation $\tilde{U}(\tau/2)$ and $\tilde{U}(\tau)$, respectively. For this purpose we will use the unconditionally stable algorithms $T2S2$ and $T4S2$ (see Ref. [10] for more details). $T2S2$ ($T4S2$) is second- (fourth-) order accurate in the time step τ. Both are second-order accurate with respect to the mesh size δ.

In the presence of the current source, the application of the Yee algorithm requires considerable additional work. The Yee algorithm is second-order accurate in both the mesh size and the time step. This is due to the use of a staggered grid, both in space and time. From the point of view of time integration, the latter presents some problems that are absent in all other time-integration methods discussed in this paper. Indeed, to complete one time step with the Yee algorithm we need to know the values of say $E_z(t)$ and $H_y(t-\tau/2)$, not $H_y(t)$, and the proper, time-shifted, values of the current contributions in Eq. (34). If a current source is present it is reasonable to start with $\mathbf{\Psi}(0)=\mathbf{0}$. Then we use the one-step algorithm to compute time-shifted values of the current contributions in Eq. (34). Note that because the time dependence of the source, Eq. (20), is known explicitly, these calculations need to be carried out only once. The Yee algorithm can now be used for time stepping. To compare the final result of the Yee algorithm with those of the one-step method we have to know the numerically exact values of both $E_z(t)$ and $H_y(t-\tau/2)$. The latter can be obtained by another application of the one-step algorithm. We also used $T4S2$ with a very small time step to perform these time shifts and obtained the same results.

In Fig. 5, we show a typical result of a one-step calculation on a grid of $n=5001$ sites with $\delta=0.1$ (corresponding to a physical length of 250.1), and a current source placed at $i=2500$ to eliminate possible artifacts of the boundaries. The frequency of the source is set to 1 ($\Omega=1$) and the number of periods the source radiates is set to 4 (i.e., $T=4$). If $\mathbf{\Psi}(0)=\mathbf{0}$, which is the usual case if a current source is present, the one-step algorithm requires K' matrix-vector operations (i.e., $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$) to compute $\mathbf{\Psi}(t)$. The standard Yee, $T2S2$, and $T4S2$ algorithms require, respectively, 1, 1.5, and 6 $\mathcal{L}\mathbf{\Psi}$ operations per time step.

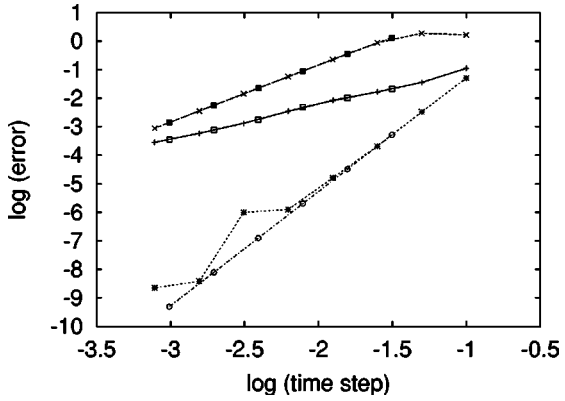

FIG. 6. The error $\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|/\|\hat{\mathbf{\Psi}}(t)\|$ at time $t=100$ as a function of the time step τ for three different FDTD algorithms. The current source is positioned at the center of the system, and oscillates at frequency $\Omega=1$ during the interval $0\leqslant t\leqslant 4$ (see Fig. 5). $\hat{\mathbf{\Psi}}(t)$ is the vector obtained by the one-step algorithm, using $K'=2090$ matrix-vector operations $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$. $\tilde{\mathbf{\Psi}}(t)$ is obtained by one of the FDTD algorithms. Plus signs, Yee algorithm [1,2] using $\tau=0.1/2^n$ for $n=0,1,\ldots,7$; crosses, second-order unconditionally stable algorithm $T2S2$ [9,10] using $\tau=0.1/2^n$ for $n=0,1,\ldots,7$; stars, fourth-order unconditionally stable algorithm $T4S2$ [9,10] using $\tau=0.1/2^n$ for $n=0,1,\ldots,7$; open squares, Yee algorithm using $\tau=0.01\pi/2^n$ for $n=0,1,\ldots,6$; solid squares, $T2S2$ using $\tau=0.01\pi/2^n$ for $n=0,1,\ldots,6$; open circles, $T4S2$ using $\tau=0.01\pi/2^n$ for $n=0,1,\ldots,6$. Lines are a guide to the eye only.

We define the error of the solution $\tilde{\mathbf{\Psi}}(t)$ for the wave form by $\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|/\|\hat{\mathbf{\Psi}}(t)\|$, where $\hat{\mathbf{\Psi}}(t)$ is the vector of the em fields obtained by the one-step algorithm. In Fig. 6, we present results of numerical experiments with the four different time-integration algorithms, for the same system used to compute the results shown in Fig. 5. We compare the solutions of the Maxwell equations at $t=100$, i.e., well before the wave fronts reach the boundaries (see Fig. 5). In these calculations, we have used two different sets of time steps, namely, $\tau=0.1/2^n$ for $n=0,1,\ldots,7$ and $\tau=0.01\pi/2^n$ for $n=0,1,\ldots,6$. On purpose, the former has been chosen such that the time at which the source is turned off ($T=4\pi$ in this example), divided by τ, is not an integer. Then it is conceivable that the discrete sampling of the source term may introduce artifacts because approximation (34) does not correctly sample the source term near the end point T. In the latter case, these artifacts should not be present. According to the rigorous bounds on the error of the $T4S2$ algorithm the error should vanish with τ^4 [9–11]. The erratic behavior of the first set of $T4S2$ data (see Fig. 6) and deviation from the τ^4 dependence are manifestations of the ''inappropriate'' choice of the time step in relation to the pulse duration T. As we use a fourth-order accurate approximation to compute the

TABLE I. The error $\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|/\|\hat{\mathbf{\Psi}}(t)\|$ at time $t=100$ as a function of the time step τ for three different FDTD algorithms. The initial values of the em fields are random, distributed uniformly over the interval [-1,1]. $\hat{\mathbf{\Psi}}(t)$ is the vector obtained by the one-step algorithm, using $K=2110$ matrix-vector operations $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$. $\tilde{\mathbf{\Psi}}(t)$ is obtained by one of the FDTD algorithms. Yee, Yee algorithm [1,2]; $T2S2$, second-order unconditionally stable algorithm [9,10]; $T4S2$, fourth-order unconditionally stable algorithm [9,10].

τ	Yee	$T2S2$	$T4S2$
0.100	$0.99\times10^{+1}$	$0.15\times10^{+1}$	$0.13\times10^{+1}$
0.010	$0.13\times10^{+1}$	$0.79\times10^{+0}$	0.29×10^{-3}
0.001	0.19×10^{-1}	0.83×10^{-2}	0.29×10^{-7}

contribution of the source term, this effect is too small to affect the results of the second-order integrators.

The rigorous bound on the error between the exact and $T4S2$ results tells us that this error should vanish with τ^4 [9–11]. This knowledge can be exploited to test if the one-step algorithm yields the exact numerical answer. Using the triangle inequality we can write

$$\|\mathbf{\Psi}(t)-\hat{\mathbf{\Psi}}(t)\|\leqslant\|\mathbf{\Psi}(t)-\tilde{\mathbf{\Psi}}(t)\|+\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|, \quad (35)$$

$$\leqslant\tilde{c}\tau^4 t\left(1+\int_0^t\|\mathbf{J}(u)\|du\right)+\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|, \quad (36)$$

where $\tilde{\mathbf{\Psi}}(t)$ and $\hat{\mathbf{\Psi}}(t)$ are the results of the $T4S2$ and the one-step algorithm, respectively, and $\tilde{c}$ is a positive constant [11]. As the numerical data in Fig. 6 show that $\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|\to0$ as τ^4, we can be confident that the one-step algorithm yields the correct answer within rounding errors.

From the data in Fig. 6 it is clear that $T2S2$ is the least efficient of the three FDTD methods: It uses about a factor of 1.5 more arithmetic operations and yields errors that are larger than those of the Yee algorithm. However, this conclusion does not generalize as the Yee algorithm yields the largest errors of the three methods if the initial em field distribution is random, as illustrated in Table I [27]. The error on the Yee-algorithm result is expected to vanish as τ^2 for sufficiently small τ and, as shown in Fig. 6, it does. However, as Fig. 6 also shows, unless τ is made sufficiently small ($\tau\leqslant0.0125$ in this example), the presence of the source term changes the quadratic behavior to almost linear.

To obtain the data of Fig. 6, the one-step algorithm requires $K'=2090$ matrix-vector operations $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$. This implies that for all $\tau<t/K'$, the FDTD algorithms will perform more $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$ operations than the one-step algorithm. This is the case if $\tau<0.05$ for the Yee algorithm and is always the case for $T4S2$ because the latter uses a factor of 6 more $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$ operations than the Yee algorithm.

The answer to the question that which of the algorithms is the most efficient crucially depends on the error level that one finds acceptable. Taking the data of Fig. 6 as an example we see that if one is satisfied with an error of more than 2.5%, one could use the Yee algorithm, though we recommend to use the one-step algorithm because then the time-integration error is negligible. The Yee algorithm is no competition for $T4S2$ if one requires an error of less than 1% but $T4S2$ is not nearly as efficient as the one-step algorithm. These conclusions seem to be quite general, i.e., we have not been able to construct counterexamples. Moreover, as we will see below, the one-dimensional case is rather favorable with respect to the efficiency of the FDTD algorithms.

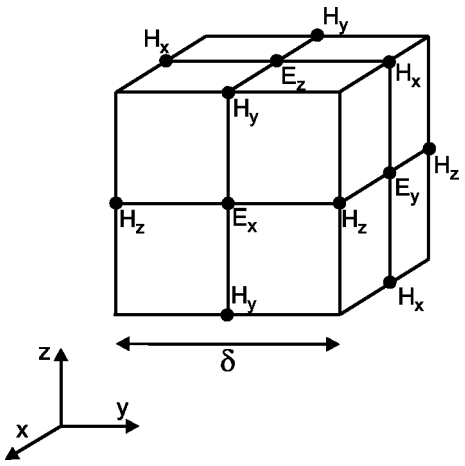

FIG. 7. Unit cell of the Yee grid.

V. SCATTERING FROM A THREE-DIMENSIONAL OBJECT

We now consider a more complicated but realistic problem of em scattering in three dimensions. In the preceding section, we already showed that the one-step algorithm is very efficient if a current source is present. In the sequel we put $\mathbf{J}(t)=\mathbf{0}$ and demonstrate that the same conclusion holds if the source is absent. First we write the Maxwell equations (1) as

$$\frac{\partial}{\partial t}\mathbf{Z}(t)=\mathcal{H}\mathbf{Z}(t)=\begin{pmatrix}0 & h\\ -h^T & 0\end{pmatrix}\mathbf{Z}(t), \quad (37)$$

where h is given by

$$h=\begin{pmatrix}0 & \frac{1}{\sqrt{\mu}}\frac{\partial}{\partial z}\frac{1}{\sqrt{\varepsilon}} & -\frac{1}{\sqrt{\mu}}\frac{\partial}{\partial y}\frac{1}{\sqrt{\varepsilon}}\\ -\frac{1}{\sqrt{\mu}}\frac{\partial}{\partial z}\frac{1}{\sqrt{\varepsilon}} & 0 & \frac{1}{\sqrt{\mu}}\frac{\partial}{\partial x}\frac{1}{\sqrt{\varepsilon}}\\ \frac{1}{\sqrt{\mu}}\frac{\partial}{\partial y}\frac{1}{\sqrt{\varepsilon}} & -\frac{1}{\sqrt{\mu}}\frac{\partial}{\partial x}\frac{1}{\sqrt{\varepsilon}} & 0\end{pmatrix}. \quad (38)$$

We discretize Eq. (37) by placing the em fields on the vertices of the Yee lattice, as indicated in Fig. 7. On this lattice, the elements of the vector $\mathbf{\Psi}(t)$ are given by

$$\mathbf{\Psi}(i,j,k,t)=\begin{cases} X_x(i,j,k,t)=\sqrt{\mu(i,j,k)}H_x(i,j,k,t), & i \text{ even, } j \text{ odd, } k \text{ odd} \\ X_y(i,j,k,t)=\sqrt{\mu(i,j,k)}H_y(i,j,k,t), & i \text{ odd, } j \text{ even, } k \text{ odd} \\ X_z(i,j,k,t)=\sqrt{\mu(i,j,k)}H_z(i,j,k,t), & i \text{ odd, } j \text{ odd, } k \text{ even} \\ Y_x(i,j,k,t)=\sqrt{\varepsilon(i,j,k)}E_x(i,j,k,t), & i \text{ odd, } j \text{ even, } k \text{ even} \\ Y_y(i,j,k,t)=\sqrt{\varepsilon(i,j,k)}E_y(i,j,k,t), & i \text{ even, } j \text{ odd, } k \text{ even} \\ Y_z(i,j,k,t)=\sqrt{\varepsilon(i,j,k)}E_z(i,j,k,t), & i \text{ even, } j \text{ even, } k \text{ odd,} \end{cases} \tag{39}$$

where we introduced the notation $A(i,j,k,t)=A(x=i\delta/2,y=j\delta/2,z=k\delta/2,t)$ and the origin of the coordinate system is chosen such that its coordinates (i,j,k) are all even. Through this arrangement, the EM fields automatically satisfy the boundary conditions (3) if the number of lattice points in the x, y, and z directions, to be denoted by L_x, L_y, and L_z respectively, is odd.

Approximating the differential operators that appear in Eq. (38) by the standard three-point difference formula, we can write [9]

$$\mathcal{L}=\sum_{i=1}^{L_x-2}{}' \sum_{j=1}^{L_y-2}{}' \sum_{k=1}^{L_z-2}{}' [\mathcal{L}^{(x)}(i,j,k)+\mathcal{L}^{(y)}(i,j,k)+\mathcal{L}^{(z)}(i,j,k)], \tag{40}$$

where the prime indicates that the stride of each summation index is 2, and the superscripts (x), (y), and (z) refer to the derivative with respect to x, y, and z, respectively. More explicitly, we have

$$\begin{aligned}\mathcal{L}^{(x)}(i,j,k)=&+\frac{\mathbf{e}_{i,j+1,k}\mathbf{e}_{i+1,j+1,k}^T-\mathbf{e}_{i+1,j+1,k}\mathbf{e}_{i,j+1,k}^T}{\delta\sqrt{\varepsilon(i+1,j+1,k)\mu(i,j+1,k)}}\\ &-\frac{\mathbf{e}_{i,j,k+1}\mathbf{e}_{i+1,j,k+1}^T-\mathbf{e}_{i+1,j,k+1}\mathbf{e}_{i,j,k+1}^T}{\delta\sqrt{\varepsilon(i+1,j,k+1)\mu(i,j,k+1)}}\\ &+\frac{\mathbf{e}_{i+1,j+1,k}\mathbf{e}_{i+2,j+1,k}^T-\mathbf{e}_{i+2,j+1,k}\mathbf{e}_{i+1,j+1,k}^T}{\delta\sqrt{\varepsilon(i+1,j+1,k)\mu(i+2,j+1,k)}}\\ &-\frac{\mathbf{e}_{i+1,j,k+1}\mathbf{e}_{i+2,j,k+1}^T-\mathbf{e}_{i+2,j,k+1}\mathbf{e}_{i+1,j,k+1}^T}{\delta\sqrt{\varepsilon(i+1,j,k+1)\mu(i+2,j,k+1)}},\end{aligned} \tag{41}$$

and the expressions for $\mathcal{L}^{(y)}(i,j,k)$ and $\mathcal{L}^{(z)}(i,j,k)$ follow from Eq. (41) by symmetry. Note that we use the triple (i,j,k) to label the $L_xL_yL_z$ unit vectors $\mathbf{e}_{i,j,k}$. The (i,j,k)th element of $\mathbf{\Psi}(t)$ is given by the inner product $\mathbf{\Psi}(i,j,k,t)=\mathbf{e}_{i,j,k}^T\cdot\mathbf{\Psi}(t)$. It is easy to check that by construction, the matrix $\mathcal{L}$ given by Eq. (40) is skew symmetric.

As an example we show results of scattering of an em wave packet by a three-dimensional scaffold [28] of dielectric material (see Fig. 8). As the results of the one-step method and FDTD calculations are visually indistinguishable on the scale used to prepare the snapshots, we only show the results of the former. In Fig. 9, we show 2D projections of

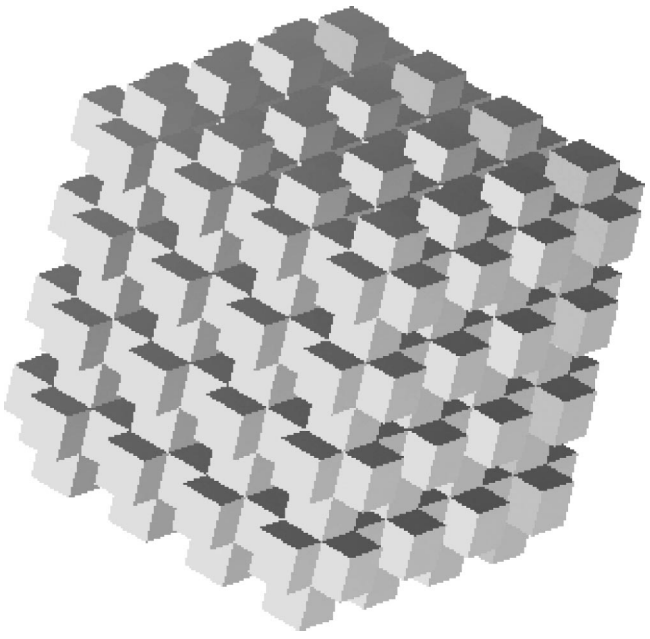

FIG. 8. Cut of the scaffold structure used in the simulation. The scaffold structure is built from square 0.5×0.5 rods with a spacing of 0.5. Dielectric constant of the rods: $\varepsilon=4$.

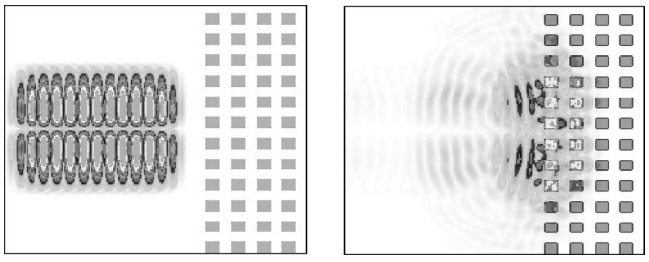

FIG. 9. Energy density distribution of the em field as a function of $(x,y=6,z)$. The simulation box measures $12\times12\times12$. The coordinates of the left-bottom corner are (0,6,0). The scaffold structure (see Fig. 8) is located in the domain $(8\leqslant x\leqslant12,0\leqslant y\leqslant12,0\leqslant z\leqslant12)$. The form of the initial wave packet is given by Eq. (42), with the center located at $\mathbf{r}_0=(3.5,6,6)^T$, the parameters that determine the width given by $\boldsymbol{\sigma}=(3,0.75,0.75)^T$, and wave vector $\mathbf{k}=(6,0,0)$. Left, $t=0$; right, $t=6.4$.

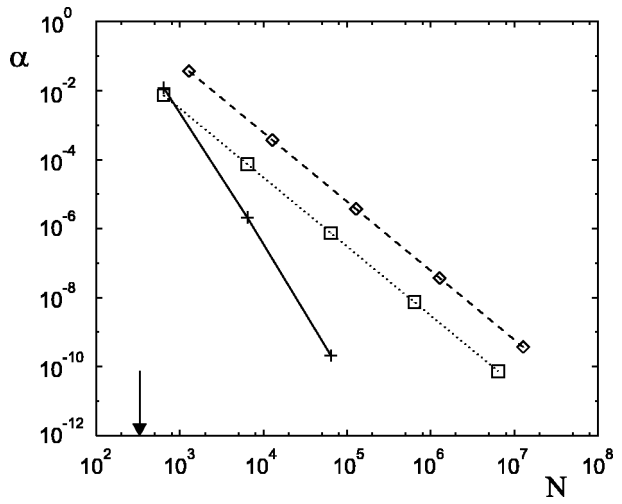

FIG. 10. Error analysis of the one-step algorithm, the Yee [1,2] algorithm, and two unconditionally stable algorithms [9,10]. The data have been obtained from the simulation of the scattering of the em field on the scaffold structure (see Figs. 8 and 9). Shown is the error $\alpha=\|\tilde{\mathbf{\Psi}}(t)-\hat{\mathbf{\Psi}}(t)\|$ at $t=6.4$ as a function of the number of times N the operation $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$ is carried out. The computational effort of each of the four methods is proportional to N. The one-step algorithm with $K=320$ (indicated by the arrow) was used to generate the reference solution $\hat{\mathbf{\Psi}}(t)$. Crosses, $\tilde{\mathbf{\Psi}}(t)$ obtained by the fourth-order unconditionally stable algorithm $T4S2$ [9,10] for $\tau=10^{-1}$, $\tau=10^{-2}$, and $\tau=10^{-3}$; open squares, $\tilde{\mathbf{\Psi}}(t)$ obtained by the standard Yee algorithm for $\tau=10^{-2}$, $\tau=10^{-3}$, $\tau=10^{-4}$, $\tau=10^{-5}$, and $\tau=10^{-6}$; open diamonds, $\tilde{\mathbf{\Psi}}(t)$ obtained by the second-order unconditionally stable algorithm $T2S2$ [9,10] for $\tau=10^{-2}$, $\tau=10^{-3}$, $\tau=10^{-4}$, $\tau=10^{-5}$, and $\tau=10^{-6}$. Lines are a guide to the eyes.

the energy density distributions of the em field at $t=0$ (left hand side) and $t=6.4$ (right hand side), respectively. The initial wave packet is defined by

$$H_x(t)=\sin[k(x-x_0-t)]e^{-(x-x_0-t)^{10}/\sigma_x^{10}} \times e^{-(y-y_0)^2/\sigma_y^2}e^{-(z-z_0)^2/\sigma_z^2}, \tag{42}$$

i.e., a product of two Gaussians in the y and z directions and a function with a much sharper cutoff in the x direction. The other components of the em field have been determined numerically, taking into account that, in vacuum, the Fourier components of the em fields are related to each other through the Maxwell equations with $\varepsilon=\mu=1$ and $\mathbf{J}(t)=\mathbf{0}$.

These and other calculations (results not shown) for different systems have been used to compare the computational efficiency of different FDTD algorithms with the one-step method described in this paper. In Fig. 10, we show the results of such a comparison for the case shown in Figs. 8 and 9. These results are representative, i.e., not intentionally selected to favor one particular method.

A quantitative analysis of the efficiency can be made as follows. We assume that the time step τ is sufficiently small such that the error of the nth-order algorithm is proportional to $\tau^n t$ (see Fig. 10). We perform a calculation for a particular time $\tilde{t}$ and compute the error $\tilde{\alpha}$ by comparing the result with the one of the one-step algorithms (with K chosen properly, see above). As we have $\tilde{\alpha}=a\tilde{\tau}^n\tilde{t}$ we can determine the constant a. Let one time step take W matrix-vector operations of the type $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$. For a 3D calculation we have $W=1$, $W=2$, and $W=10$ for the Yee, $T2S2$, and $T4S2$ algorithm, respectively (the actual number of floating-point operations carried out by our algorithms agrees with these estimates). Let $\tilde{N}$ denote the number of $\mathbf{\Psi}'\leftarrow M\mathbf{\Psi}$ operations it takes to obtain the solution at time $\tilde{t}$ with error $\tilde{\alpha}$. We have $\tilde{\tau}=W\tilde{t}/\tilde{N}$ and $\tilde{\alpha}=aW^n\tilde{t}^{n+1}/\tilde{N}^n$. We can now calculate the number of operations N it will take to compute the solution at time t with accuracy α. Using the scaling properties of the error we have $\alpha=aW^nt^{n+1}/N^n$ and eliminating the constant a yields

$$N=\tilde{N}\left(\frac{\tilde{\alpha}}{\alpha}\right)^{1/n}\left(\frac{t}{\tilde{t}}\right)^{(n+1)/n}. \tag{43}$$

In practice, for each algorithm we first have determined the numbers n, $\tilde{N}$, $\tilde{\alpha}$, and $\tilde{t}$ in Eq. (43) before we can use Eq. (43). This can be done by making one numerical experiment per algorithm for a reasonable choice of the time step. The one-step algorithm computes the solution at time t using K operations and, as we have seen above, K scales linearly with t. On the other hand, as Eq. (43) shows, an nth-order algorithm requires of the order of $t^{(n+1)/n}$ operations. Thus, for large t, the one-step algorithm will be (much) more efficient than the FDTD algorithms.

As an illustration we use the data of Fig. 10. For the one-step algorithm we have $K\approx 50+42t$. If we require an error of 1% (i.e., $\alpha=10^{-2}$) we find $N=34t^{3/2}$, $N=86t^{3/2}$, $N=76t^{5/4}$, for the Yee, $T2S2$, and $T4S2$ algorithm, respectively. Plotting these expressions for K and N as a function of t (results not shown), we find that the one-step algorithm outperforms the FDTD algorithms if $t>3$. The latter statement is rather sensitive to the accuracy of the time-integration algorithm that one finds acceptable. For instance, if one would like to have an error of at most $\alpha=10^{-3}$, we find $N=107t^{3/2}$, $N=272t^{3/2}$, $N=135t^{5/4}$, for the Yee, $T2S2$, and $T4S2$ algorithm, respectively, and the one-step algorithm is more efficient than the FDTD algorithms if $t>1$. Furthermore, as shown in Fig. 11, we conclude that for longer times none of the FDTD algorithms can compete with the one-step algorithm in terms of efficiency. The $T2S2$ algorithm is always (much) less efficient than the other two FDTD algorithms and has therefore been omitted in Fig. 11. For $t=20$, the one-step algorithm is a factor of 10 faster than the Yee algorithm. Thereby, we have disregarded the fact that the Yee algorithm yields results within an error of 0.1% while the one-step algorithm gives the numerically exact solution. It is evident that the $\tilde{N}$, $\tilde{\alpha}$, and $\tilde{t}$ in Eq. (43) may vary from problem to problem but the scaling of N with t will not. Therefore we may conclude that, for long times, the one-step algorithm can be orders of magnitude more efficient than current FDTD methods, even if we are content with a poor accuracy of the latter.

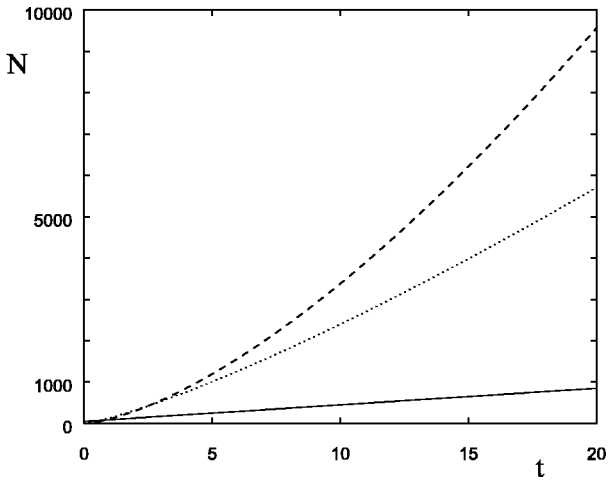

FIG. 11. The number of $\mathbf{\Psi}' \leftarrow M\mathbf{\Psi}$ operations N needed to compute the solution of the 3D Maxwell equation at time t for the system shown in Figs. 8 and 9. Solid line, one-step algorithm; dashed line, Yee algorithm [1–3] yielding a solution within 0.1% error; dotted line, $T4S2$ algorithm [9,10] yielding a solution within 0.1% error.

VI. CONCLUSION

We have described a one-step algorithm to solve the time-dependent Maxwell equations. We have presented numerical results for a 1D system with and a 3D system without a current source and compared these results with the ones obtained by using the conventional Yee algorithm and two unconditionally stable algorithms. The latter offer rigorous control over the errors and have been used to produce the reference data. In all cases the one-step algorithm reproduces these results. Typically, the one-step algorithm is more than an order of magnitude faster than the FDTD algorithms. This roughly matches our expectations based on a count of the number of arithmetic operations for the different methods. Overall our conclusions are in concert with those drawn on the basis of numerical experiments with the Schrödinger equation [17].

For some applications it will be necessary to use a better spatial discretization than the most simple one employed in this paper. It is straigthforward to adopt the approach used in the case of the unconditionally stable algorithms [10]. This will not affect our general conclusions regarding the efficiency of the different methods. Currently the mathematical justification of the one-step method requires that the matrix $\mathcal{L}$ is skew symmetric. This is of particular importance if we want to treat other (e.g., absorbing) boundary conditions by the same approach. Many practical applications use other types of boundary conditions [2] than those adopted in this paper. The problem of incorporating these boundary conditions is left for future research.

In summary, our results indicate that the main features of the one-step algorithm for solving the time-dependent Maxwell equations are the following (1) applicable to systems with spatially varying permittivity and permeability and current source, (2) no need to sample the time dependence of the current source, (3) very accurate time integration, (4) efficient method to compute the em fields at particular times, (5) may be used for time stepping with (very) large time steps (6) implementation is as easy as for the FDTD algorithms.

ACKNOWLEDGMENTS

H.D.R. and K.M. are grateful to T. Iitaka for drawing our attention to the potential of the Chebyshev method and for illuminating discussions. This work was partially supported by the Dutch ''Stichting Nationale Computer Faciliteiten'' (NCF) and the EC IST project CANVAD.

APPENDIX: TECHNICAL ISSUES

Truncation of the series expansion of $e^{t\mathcal{L}}$ in terms of the Chebyshev polynomials at the Kth term [i.e., Eq. (18)] yields a Kth-order polynomial approximation in $\mathcal{L}^0, \ldots, \mathcal{L}^K$. One might wonder why not use the corresponding Taylor polynomial instead of the more complicated Chebyshev recursion. Numerical instability is the main reason for not using the Taylor expansion [23], a point that we illustrate by computing the time evolution of a 1D Gaussian wave packet. The

TABLE II. The error $E_{Taylor} = \|U_{Taylor}\mathbf{\Psi}(t=0) - \hat{\mathbf{\Psi}}(t)\| / \|\hat{\mathbf{\Psi}}(t)\|$ for different times t and different numbers of terms in the polynomial expansion. The initial value of the em fields is a Gaussian wave packet ($\mathbf{\Psi}(t=0) \propto \exp[-(x-x_0)^2/4]$, $x_0 = 125$), centered around the middle of the system (see Fig. 5). Also shown is the error $E_{Chebyshev}$ with respect to the reference solution $\hat{\mathbf{\Psi}}(t)$ obtained by the one-step algorithm using $K = 300$ expansion coefficients. An entry with value - MP - indicates that the result is exact to machine precision.

t	K	E_{Taylor}	$E_{Chebyshev}$
2.0	50	0.48×10^{-1}	0.54×10^{-3}
2.0	55	0.11×10^{-1}	0.11×10^{-4}
2.0	60	0.15×10^{-2}	0.73×10^{-7}
2.0	70	0.94×10^{-5}	0.13×10^{-11}
2.0	140	0.61×10^{-15}	- MP -
2.0	200	0.61×10^{-15}	- MP -
2.0	205	- Overflow -	- MP -
3.0	80	$0.23\times10^{+7}$	0.96×10^{-6}
3.0	85	$0.43\times10^{+6}$	0.13×10^{-7}
3.0	90	$0.60\times10^{+5}$	0.10×10^{-9}
3.0	95	$0.64\times10^{+4}$	0.62×10^{-12}
3.0	100	$0.52\times10^{+3}$	0.25×10^{-14}
3.0	120	0.21×10^{-2}	- MP -
3.0	140	0.33×10^{-7}	- MP -
3.0	160	0.33×10^{-7}	- MP -
3.0	180	0.33×10^{-7}	- MP -
3.0	190	- Overflow -	- MP -
4.0	100	$0.20\times10^{+16}$	0.45×10^{-5}
4.0	120	$0.26\times10^{+13}$	0.13×10^{-12}
4.0	140	$0.13\times10^{+9}$	- MP -
4.0	160	$0.35\times10^{+3}$	- MP -
4.0	170	$0.15\times10^{+2}$	- MP -
4.0	172	- Overflow -	- MP -
5.0	140	$0.54\times10^{+22}$	0.21×10^{-11}
5.0	160	$0.13\times10^{+19}$	- MP -
5.0	165	- Overflow -	- MP -

system we consider is identical to the one used in Sec. IV. For convenience the vector of initial values is normalized to 1. The algorithm based on the Taylor expansion is defined by

$$U_{Taylor}=\sum_{n=1}^{K}\frac{t^n}{n!}\mathcal{L}^n. \tag{A1}$$

In Table II we compare the Taylor-expansion algorithm with the Chebyshev algorithm. It is obvious that the instability of the Taylor algorithm makes it unsuitable for integrating the Maxwell equations over extended time intervals. Note that the values of K that yield high precision are very close to the values of K for which the numerical instabilities occur.

Although the Taylor series is accurate for eigenvalues of $\mathcal{L}$ close to zero, this is clearly not sufficient to accurately approximate the matrix exponential $e^{t\mathcal{L}}$. The Chebyshev expansion on the other hand, guarantees equal maximum error over the whole interval of eigenvalues of $\mathcal{L}$. Therefore, for the same number of terms in the expansion, it is a much better approximation to $e^{t\mathcal{L}}$ than the truncated Taylor series. As Table II demonstrates, numerical instability renders the Taylor expansion useless if $t>4$. The Chebyshev expansion does not suffer from this limitation.

[1] K.S. Yee, IEEE Trans. Antennas Propag. **14**, 302 (1966).

[2] A. Taflove and S.C. Hagness, *Computational Electrodynamics-The Finite-Difference Time-Domain Method* (Artech House, Boston, 2000).

[3] K.S. Kunz and R.J. Luebbers, *Finite-Difference Time-Domain Method for Electromagnetics* (CRC Press, Boca Raton, FL, 1993).

[4] See http://www.fdtd.org

[5] F. Zheng, Z. Chen, and J. Zhang, IEEE Trans. Microwave Theory Tech. **48**, 1550 (2000).

[6] T. Namiki, IEEE Trans. Microwave Theory Tech. **48**, 1743 (2001).

[7] F. Zheng and Z. Chen, IEEE Trans. Microwave Theory Tech. **49**, 1006 (2001).

[8] W. Harshawardhan, Q. Su, and R. Grobe, Phys. Rev. E **62**, 8705 (2000).

[9] J.S. Kole, M.T. Figge, and H. De Raedt, Phys. Rev. E **64**, 066705 (2001).

[10] J.S. Kole, M.T. Figge, and H. De Raedt, Phys. Rev. E **65**, 066705 (2002).

[11] H. De Raedt, Comput. Phys. Rep. **7**, 1 (1987).

[12] O.P. Gandi, in *Advances in Computational Electrodynamics-The Finite-Difference Time-Domain Method*, edited by A. Taflove (Artech House, Boston, 1998).

[13] B. Houshmand, T. Itoh, and M. Piket-May, in *Advances in Computational Electrodynamics—The Finite-Difference Time-Domain Method*, edited by A. Taflove (Artech House, Boston, 1998).

[14] H. Tal-Ezer, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **23**, 11 (1986).

[15] H. Tal-Ezer, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **26**, 1 (1989).

[16] H. Tal-Ezer and R. Kosloff, J. Chem. Phys. **81**, 3967 (1984).

[17] C. Leforestier, R.H. Bisseling, C. Cerjan, M.D. Feit, R. Friesner, A. Guldberg, A. Hammerich, G. Jolicard, W. Karrlein, H.-D. Meyer, N. Lipkin, O. Roncero, and R. Kosloff, J. Comput. Phys. **94**, 59 (1991).

[18] T. Iitaka, S. Nomura, H. Hirayama, X. Zhao, Y. Aoyagi, and T. Sugano, Phys. Rev. E **56**, 1222 (1997).

[19] R.N. Silver and H. Röder, Phys. Rev. E **56**, 4822 (1997).

[20] Y.L. Loh, S.N. Taraskin, and S.R. Elliott, Phys. Rev. Lett. **84**, 2290 (2000); **84**, 5028(E) (2000).

[21] M. Born and E. Wolf, *Principles of Optics* (Pergamon, Oxford, 1964).

[22] J.H. Wilkinson, *The Algebraic Eigenvalue Problem* (Clarendon Press, Oxford, 1965).

[23] G.D. Smith, *Numerical Solution of Partial Differential Equations* (Clarendon Press, Oxford, 1985).

[24] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover, New York, 1964).

[25] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical Recipes*, (Cambridge University Press, New York, 1986).

[26] J.D. Jackson, *Classical Electrodynamics* (Wiley, New York, 1962).

[27] This also explains why the unconditionally stable algorithms $T2S2$ and $T4S2$ yield more accurate eigenvalue distributions than the Yee algorithm [9].

[28] H. Sami Sözüer and J.W. Haus, J. Opt. Soc. Am. B **10**, 296 (1993).