Higher-order unconditionally stable algorithms to solve the time-dependent Maxwell equations

J. S. Kole,* M. T. Figge,† and H. De Raedt‡
Centre for Theoretical Physics and Materials Science Centre, University of Groningen, Nijenborgh 4, NL-9747 AG Groningen, The Netherlands
(Received 12 December 2001; published 26 June 2002)

For the recently introduced algorithms to solve the time-dependent Maxwell equations [J. S. Kole, M. T. Figge, and H. De Raedt, Phys. Rev. E **64**, 066705 (2001)], we construct a variable grid implementation and an improved spatial discretization implementation that preserve the exceptional property of the algorithms to be unconditionally stable by construction. We find that the performance and accuracy of the corresponding algorithms are significant and illustrate their practical relevance by simulating various physical model systems.

DOI: 10.1103/PhysRevE.65.066705 PACS number(s): 02.60.Cb, 03.50.De, 41.20.Jb

I. INTRODUCTION

In a recent paper, we introduced a family of algorithms to solve the time-dependent Maxwell equations [1]. Salient features of these algorithms include the rigorously provable unconditional stability for d-dimensional systems ($d=1,2,3$) with spatially varying permittivity and permeability, as well as the exact conservation of the energy density of the electromagnetic (EM) fields. Furthermore, we have demonstrated that without affecting the unconditional stability of the algorithms the order of accuracy in the time integration can be systematically increased. An important aspect that has not been considered in our earlier work [1] concerns the effect of the discretization of space on the accuracy of the algorithms. Previously, we employed only the most simple spatial discretization, namely, the central-difference scheme on a Cartesian grid with a constant mesh size [1]. We refer to this spatial discretization scheme as the *simple spatial implementation*. Many numerical simulations of realistic physical systems require algorithms with a more accurate spatial discretization and a more flexible spatial grid for an optimal use of computer resources (CPU time and computer memory). In the present paper we show that implementing a fourth-order accurate approximation of the spatial derivatives and a spatial grid of variable mesh sizes preserve the unconditional stability of the algorithms. We simulate various physical model systems using these proposed implementations to demonstrate the significant improvement with respect to the required computer resources in the computation of eigenmode spectra and to study systematically the temporal and spatial accuracy of the algorithms.

Our presentation is organized as follows: We recapitulate the theory of constructing unconditionally stable algorithms to solve the time-dependent Maxwell equations in Sec. II and describe the basic properties of the simple spatial implementation in Sec. III. Then, in Secs. IV and V, we present the implementation of, respectively, the variable grid and the improved spatial discretization. Our conclusions are given in Sec. VI.

II. UNCONDITIONALLY STABLE ALGORITHMS TO SOLVE MAXWELL EQUATIONS

We consider a d-dimensional model system of EM fields in a medium with spatially varying permittivity and/or permeability, surrounded by a perfectly conducting box. In the absence of free charges and currents, the EM fields in such a system satisfy the Maxwell equations [2]

$$\frac{\partial}{\partial t}\mathbf{H}=-\frac{1}{\mu}\boldsymbol{\nabla}\times\mathbf{E} \quad \text{and} \quad \frac{\partial}{\partial t}\mathbf{E}=\frac{1}{\varepsilon}\boldsymbol{\nabla}\times\mathbf{H}, \tag{1}$$

$$\operatorname{div}\varepsilon\mathbf{E}=0 \quad \text{and} \quad \operatorname{div}\mu\mathbf{H}=0, \tag{2}$$

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, respectively, the magnetic field and the electric field vectors. 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 spatial dependence on $\mathbf{r}=(x,y,z)^T$ unless this leads to ambiguities. On the surface of the perfectly conducting box the EM fields satisfy the boundary conditions [2]

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

with $\mathbf{n}$ denoting the vector normal to a boundary of the surface. The conditions, Eqs. (3), assure that the normal component of the magnetic field and the tangential components of the electric field vanish at the boundary [2]. Some important symmetries of the Maxwell Eqs. (1) and (2) can be made explicit by introducing the fields

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

In terms of the fields $\mathbf{X}(t)$ and $\mathbf{Y}(t)$, the time-dependent Maxwell equations (1) read

$$\frac{\partial}{\partial t}\begin{pmatrix}\mathbf{X}(t)\\ \mathbf{Y}(t)\end{pmatrix}=\begin{pmatrix}-\frac{1}{\sqrt{\mu}}\boldsymbol{\nabla}\times\frac{1}{\sqrt{\varepsilon}}\mathbf{Y}(t)\\ \frac{1}{\sqrt{\varepsilon}}\boldsymbol{\nabla}\times\frac{1}{\sqrt{\mu}}\mathbf{X}(t)\end{pmatrix}\equiv\mathcal{H}\begin{pmatrix}\mathbf{X}(t)\\ \mathbf{Y}(t)\end{pmatrix}, \tag{5}$$

*Email address: j.s.kole@phys.rug.nl
†Email address: m.t.figge@phys.rug.nl
‡Email address: h.a.deraedt@phys.rug.nl; http://rugth30.phys.rug.nl/compphys

where the operator $\mathcal{H}$ is given by

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

Writing $\Psi(t)=(\mathbf{X}(t),\mathbf{Y}(t))^T$, Eq. (5) becomes

$$\frac{\partial}{\partial t}\Psi(t)=\mathcal{H}\Psi(t). \tag{7}$$

It is easy to show that $\mathcal{H}$ is skew symmetric, i.e., $\mathcal{H}^T=-\mathcal{H}$, with respect to the inner product $\langle\Psi|\Psi'\rangle \equiv\int_V\Psi^T\cdot\Psi' d\mathbf{r}$, where V denotes the volume of the enclosing box. The formal solution of Eq. (7) is given by

$$\Psi(t)=\mathcal{U}(t)\Psi(0)=e^{t\mathcal{H}}\Psi(0), \tag{8}$$

where $\Psi(0)$ represents the initial state of the EM fields. The operator $\mathcal{U}(t)=e^{t\mathcal{H}}$ determines the time evolution. By construction $\|\Psi(t)\|^2=\langle\Psi(t)|\Psi(t)\rangle=\int_V[\varepsilon\mathbf{E}^2(t)+\mu\mathbf{H}^2(t)]d\mathbf{r}$, relating the length of $\Psi(t)$ to the energy density $w(t)\equiv\varepsilon\mathbf{E}^2(t)+\mu\mathbf{H}^2(t)$ of the EM fields [2]. As $\mathcal{U}(t)^T=\mathcal{U}(-t)=\mathcal{U}^{-1}(t)=e^{-t\mathcal{H}}$ it follows that $\langle\mathcal{U}(t)\Psi(0)|\mathcal{U}(t)\Psi(0)\rangle=\langle\Psi(t)|\Psi(t)\rangle=\langle\Psi(0)|\Psi(0)\rangle$. Hence the time-evolution operator $\mathcal{U}(t)$ is an orthogonal transformation, rotating the vector $\Psi(t)$ without changing its length $\|\Psi\|$. In physical terms this means that the energy density of the EM fields does not change with time, as expected on physical grounds [2].

A numerical procedure that solves the time-dependent Maxwell equations necessarily starts by discretizing the spatial derivatives. This maps the continuum problem described by $\mathcal{H}$ onto a lattice problem defined by a matrix H. The corresponding time-evolution operator is given by $U(t)=e^{tH}$. Ideally, this mapping should not change the basic symmetries of the original problem. The underlying symmetry of the Maxwell equations suggests to use matrices H that are real and skew symmetric. Since formally the time evolution of the EM fields on the lattice is given by $\Psi(t+\tau)=U(\tau)\Psi(t)=e^{\tau H}\Psi(t)$, the second ingredient of the numerical procedure is to choose an approximation of the time-evolution operator $U(\tau)$. The fact that $U(t)$ is an orthogonal transformation is essential for the development of an unconditionally stable algorithm to solve the Maxwell equations [1]. A systematic approach to construct orthogonal approximations to matrix exponentials is to make use of the Lie-Trotter-Suzuki formula [3,4]

$$e^{t(H_1+\cdots+H_p)}=\lim_{m\to\infty}\left(\prod_{i=1}^{p}e^{tH_i/m}\right)^m, \tag{9}$$

and generalizations thereof [5,6]. Applied to the case of interest here, the success of this approach relies on the basic but rather trivial premise that the matrix H can be written as $H=\Sigma_{i=1}^{p}H_i$, where each of the matrices H_i is real and skew symmetric. Expression, Eq. (9), suggests that

$$U_1(\tau)=e^{\tau H_1}\cdots e^{\tau H_p} \tag{10}$$

might be a good approximation to $U(\tau)$ if τ is sufficiently small. In fact, it can be shown that $U(\tau)$ and $U_1(\tau)$ are identical up to first order in τ [7]. Most importantly, if all the H_i are real and skew symmetric, $U_1(\tau)$ is orthogonal by construction. Therefore, by construction, a numerical scheme based on Eq. (10) will be unconditionally stable. The product-formula approach provides simple, systematic procedures to improve the accuracy of the approximation to $U(\tau)$ without changing its fundamental symmetries. For example, the orthogonal matrix

$$\begin{aligned} U_2(\tau)&=U_1^T(-\tau/2)U_1(\tau/2) \\ &=e^{\tau H_p/2}\cdots e^{\tau H_2/2}e^{\tau H_1}e^{\tau H_2/2}\cdots e^{\tau H_p/2} \end{aligned} \tag{11}$$

is identical to $U(\tau)$ up to second order in τ [5,6]. Suzuki's fractal decomposition approach [5] gives a general method to construct higher-order approximations based on $U_1(\tau)$ or $U_2(\tau)$. A particularly useful approximation, which is identical to $U(\tau)$ up to fourth order in τ, is given by [5]

$$U_4(\tau)=U_2(a\tau)U_2(a\tau)U_2((1-4a)\tau)U_2(a\tau)U_2(a\tau), \tag{12}$$

where $a=1/(4-4^{1/3})$. From Eqs. (10)–(12) it follows that, in practice, an efficient implementation of a scheme based on $U_1(\tau)$ is all that is needed to construct the higher-order algorithms, Eqs. (11) and (12). The approximations $U_n(\tau)$ are identical to the exact time-evolution operator $U(\tau)$ up to nth order in τ and have proven to be very useful in many applications [4,6–15]. They turn out to be equally useful for solving the time-dependent Maxwell equations [1]. In particular, it can be shown that the difference between the exact EM field vector $\Psi(t)=U(t)\Psi(0)$ and the approximate one, $\Psi_n(t)=U_n(t)\Psi(0)$ is bounded by [7]

$$\|(U(t)-U_n(t))\Psi(0)\|=\|\Psi(t)-\Psi_n(t)\|\leqslant C_n t\tau^n, \tag{13}$$

where C_n is a constant. The rigorous upper bound on the error of the EM field vector will be used to specify unconditionally stable algorithms by the temporal and spatial accuracies of the computed EM field. We denote an algorithm by $TnSm$ if its implementation involves a time integration based on $U_n(\tau)$ and a spatial discretization scheme based on an mth-order accurate approximation of the spatial derivatives.

III. SIMPLE SPATIAL IMPLEMENTATION

In this section, we briefly recapitulate the construction of the unconditionally stable algorithm to solve Maxwell equations in a one-dimensional (1D) system. Furthermore, we discuss general properties of this implementation referring also to the two-dimensional (2D) and three-dimensional (3D) cases.

Maxwell equations for a 1D system extending along the x axis contain no partial derivatives with respect to y or z. Also ε and μ do not depend on y or z. Under these conditions, the Maxwell equations reduce to two independent sets of first-

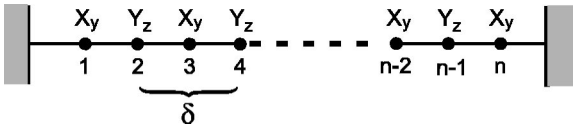

FIG. 1. Positions of the two TM-mode EM field components on the 1D grid.

order differential equations [2]. The solutions to these sets are known as the transverse electric mode and the transverse magnetic (TM) mode [2]. Restricting our considerations to the TM mode, it follows from Eq. (5) that the magnetic field $H_y(x,t)=X_y(x,t)/\sqrt{\mu(x)}$ and the electric field $E_z(x,t)=Y_z(x,t)/\sqrt{\varepsilon(x)}$ are solutions of

$$\frac{\partial}{\partial t}X_y(x,t)=\frac{1}{\sqrt{\mu(x)}}\frac{\partial}{\partial x}\left(\frac{Y_z(x,t)}{\sqrt{\varepsilon(x)}}\right), \tag{14}$$

$$\frac{\partial}{\partial t}Y_z(x,t)=\frac{1}{\sqrt{\varepsilon(x)}}\frac{\partial}{\partial x}\left(\frac{X_y(x,t)}{\sqrt{\mu(x)}}\right). \tag{15}$$

Note that in 1D the divergence of $H_y(x,t)$ and $E_z(x,t)$ is zero, hence Eqs. (2) are automatically satisfied. Using the central-difference scheme, which yields a second-order accurate approximation of the spatial derivatives, we obtain

$$\frac{\partial}{\partial t}X_y(i,t)=\frac{1}{\delta\sqrt{\mu_i}}\left(\frac{Y_z(i+1,t)}{\sqrt{\varepsilon_{i+1}}}-\frac{Y_z(i-1,t)}{\sqrt{\varepsilon_{i-1}}}\right), \tag{16}$$

$$\frac{\partial}{\partial t}Y_z(j,t)=\frac{1}{\delta\sqrt{\varepsilon_j}}\left(\frac{X_y(j+1,t)}{\sqrt{\mu_{j+1}}}-\frac{X_y(j-1,t)}{\sqrt{\mu_{j-1}}}\right), \tag{17}$$

where the spatial coordinate of an EM field component is specified through the lattice index i, e.g., $X_y(i,t)$ stands for $X_y(x=i\delta/2,t)$, and $\delta/2$ the distance between two neighboring lattice points. Following Yee [16] it is convenient to assign $X_y(i,t)$ and $Y_z(j,t)$ to the odd, respectively, even numbered lattice site, as shown in Fig. 1 for a grid of n points. The Eqs. (16) and (17) can now be combined into one equation of the form Eq. (7) by introducing the n-dimensional vector $\mathbf{\Psi}(t)$ with elements

$$\Psi(i,t)=\begin{cases} X_y(i,t)=\sqrt{\mu_i}H_y(i,t), & i \text{ odd} \\ Y_z(i,t)=\sqrt{\varepsilon_i}E_z(i,t), & i \text{ even.} \end{cases} \tag{18}$$

The vector $\mathbf{\Psi}(t)$ describes both the magnetic and the electric fields on the lattice points $i=1,\ldots,n$ and the ith element of $\mathbf{\Psi}(t)$ is given by the inner product $\Psi(i,t)=\mathbf{e}_i^T\cdot\mathbf{\Psi}(t)$, where $\mathbf{e}_i$ denotes the ith unit vector in the n-dimensional vector space. Using this notation, it is easy to show that

$$\mathbf{\Psi}(t)=U(t)\mathbf{\Psi}(0) \quad \text{with} \quad U(t)=\exp(tH), \tag{19}$$

where the matrix H is represented by two parts,

$$H=H_1+H_2, \tag{20}$$

with

$$H_1=\sum_{i=1}^{n-2}{}' \beta_{i+1,i}(\mathbf{e}_i\mathbf{e}_{i+1}^T-\mathbf{e}_{i+1}\mathbf{e}_i^T), \tag{21}$$

$$H_2=\sum_{i=1}^{n-2}{}' \beta_{i+1,i+2}(\mathbf{e}_{i+1}\mathbf{e}_{i+2}^T-\mathbf{e}_{i+2}\mathbf{e}_{i+1}^T). \tag{22}$$

Here, $\beta_{i,j}=1/(\delta\sqrt{\varepsilon_i\mu_j})$ and the prime indicates that the sum is over odd integers only. For odd n we have

$$\frac{\partial}{\partial t}\Psi(1,t)=\beta_{2,1}\Psi(2,t)$$

and

$$\frac{\partial}{\partial t}\Psi(n,t)=-\beta_{n-1,n}\Psi(n-1,t), \tag{23}$$

such that the electric field vanishes at the boundaries $[Y_z(0,t)=Y_z(n+1,t)=0]$, as required by the boundary conditions, Eqs. (3).

The representation of H as the sum of H_1 and H_2 divides the lattice into odd and even numbered cells. Most important, however, both H_1 and H_2 are skew-symmetric block-diagonal matrices, containing one 1×1 matrix and $(n-1)/2$ real 2×2 skew-symmetric matrices. Therefore, according to the general theory outlined in Sec. II, this decomposition of H is suitable to construct an orthogonal approximation

$$U_1(\tau)=e^{\tau H_1}e^{\tau H_2} \tag{24}$$

that is identical to the time-evolution operator $U(\tau)$ up to first order in τ. As the matrix exponential of a block-diagonal matrix is equal to the block-diagonal matrix of the matrix exponentials of the individual blocks, the numerical calculation of $e^{\tau H_1}$ (or $e^{\tau H_2}$) reduces to the calculation of $(n-1)/2$ matrix exponentials of 2×2 matrices. The matrix exponential of a typical 2×2 matrix appearing in $e^{\tau H_1}$ or $e^{\tau H_2}$ is simply given by

$$\exp\left[\alpha\begin{pmatrix}0 & 1\\ -1 & 0\end{pmatrix}\right]\begin{pmatrix}\Psi(i,t)\\ \Psi(j,t)\end{pmatrix}=\begin{pmatrix}\cos\alpha & \sin\alpha\\ -\sin\alpha & \cos\alpha\end{pmatrix}\begin{pmatrix}\Psi(i,t)\\ \Psi(j,t)\end{pmatrix}, \tag{25}$$

and represents the rotation of two elements of the vector $\mathbf{\Psi}(t)$ leaving all the other elements unchanged. This property of the time-evolution operator, Eq. (24), provides the intrinsic possibility to parallelize the algorithms. Furthermore, it is even possible to alter the ordering of the products in the time-evolution operator $U_n(\tau)$ in order to construct an efficient implementation for a particular system. The plane rotations, Eq. (25), are performed by simply processing an arbitrarily ordered list S of pairs of EM field vector elements using

$$U_1(\tau)=\prod_S e^{\tau\beta_{i,j}(\mathbf{e}_i\mathbf{e}_j^T-\mathbf{e}_j\mathbf{e}_i^T)}, \tag{26}$$

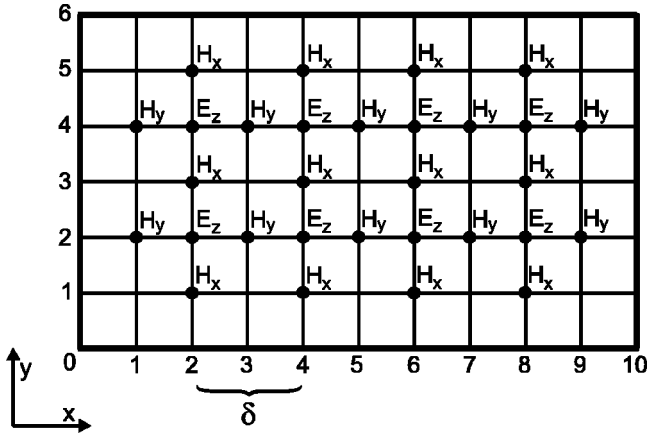

FIG. 2. Positions of the three TM-mode EM field components on the 2D grid for $n_x=9$ and $n_y=5$.

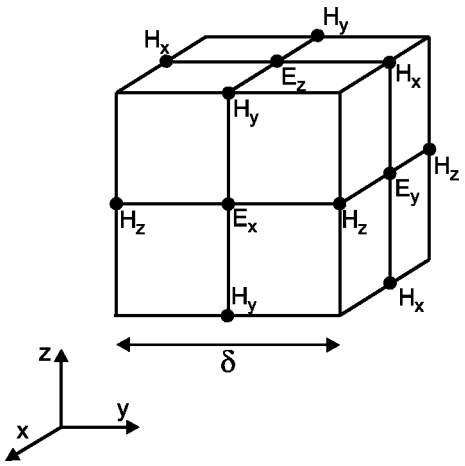

FIG. 3. Positions of the EM field components on the 3D Yee grid.

instead of the odd-even decomposition [Eq. (24)] for which $S=\{(1,2),(3,4),\ldots,(n-2,n-1),(2,3),(4,5),\ldots,(n-1,n)\}$.

The implementation for 1D can be readily extended to 2D and 3D systems, as has been illustrated in Ref. [1]. In 2D, the Maxwell equations (1) separate again into two independent sets of equations and the discretization of continuum space is done by simply reusing the 1D lattice introduced above. This is shown in Fig. 2 for the case of the 2D TM modes. The construction automatically takes care of the boundary conditions if n_x and n_y are odd and yields a real skew symmetric matrix H. Correspondingly, in 3D the spatial coordinates are discretized by adopting the standard Yee grid [16], which also automatically satisfies the boundary conditions, Eqs. (3). A unit cell of the Yee grid is shown in Fig. 3.

In general, the time step τ and the distance δ between next-nearest neighbor grid points are related due to the error that is introduced when the exact time-evolution operator $U(\tau)$ is replaced by $U_n(\tau)$. We have [5–7]

$$\|U(\tau)-U_n(\tau)\|\leqslant\gamma(d)\left(\frac{\alpha(n)\tau}{\delta}\right)^{n+1}. \tag{27}$$

Here, $\gamma(d)$ depends on the particular spatial discretization scheme used and $\alpha(n)$ represents the largest positive constant that appears as a prefactor in the exponential of the approximation $U_n(\tau)$. We find $\alpha(2)=1/2$ from Eq. (11) and inspection of Eq. (12) yields $\alpha(4)=(1/2)(4a-1)\approx 0.33$. It follows that for a required spatial resolution, which determines the smallness of δ, the time step has to be chosen such that

$$\tau\leqslant\tau^*\equiv\frac{\delta}{\alpha(n)}, \tag{28}$$

in order to keep the error, Eq. (27), small. As an example we consider a wave packet in a 2D cavity that is simulated by a $T4S2$ algorithm. For numerical purposes we use dimensionless variables throughout this paper, where the unit of length is denoted by λ and the vacuum light velocity c is taken as the unit of velocity, while the permittivity ε and permeability μ are measured in units of their corresponding values in vacuum, respectively, ε_0 and μ_0. The cavity with $\varepsilon=1$ and $\mu=1$ is of size 19×15 and contains a dielectric medium with $\varepsilon=2.25$ and $\mu=1$ that has an inclined boundary. We plot in Fig. 4 the results of simulations in which the wave packet scatters on the dielectric medium. In the four pictures we show the EM energy density distributions that are obtained after simulation time $t=12.8$ for a fixed mesh size $\delta=0.1$ and for four different time steps τ. It follows from Eq. (28) that the upper limit for the time step is given by $\tau^*=0.3$ in this case. For $\tau=0.4$ the EM energy density distribution is, in fact, seen to change dramatically such that the results become meaningless. It should be noted that the limitation, Eq. (28), on the time step is different from the Courant number that relates the time step τ to the stability of finite-difference time-domain (FDTD) algorithms [17] that are based on the Yee algorithm [16]. The algorithms presented in this paper are unconditionally stable by construction for any time step τ and produce reasonable numerical results up to $\tau=\tau^*$, a time step at which the Yee-based FDTD algorithms may have become unstable.

We conclude this section by noting that our algorithms conserve the divergence of the EM fields only in 1D systems but not in 2D and 3D systems. Although the initial state $\Psi(t=0)$ can always be chosen such that the EM fields satisfy Eqs. (2), the time integration of the Maxwell equations by an algorithm based on the approximation $U_n(\tau)$ yields EM fields whose divergence quickly acquires a finite value and then remains constant in time. This is shown in Fig. 5 where we plot the computed norm of the magnetic field divergence in a 3D system as a function of time. The 3D system is an empty cavity ($\varepsilon=1$ and $\mu=1$) of size $1.5\times 1.5\times 1.5$ and we use the $T2S2$ algorithm. The reason for this behavior of the EM field divergence is given by the fact that the divergence operation commutes with the matrix H only for a 1D system but not for 2D and 3D systems. However, we stress that the corresponding error is under control and can be reduced by using smaller time steps or algorithms with higher-order time accuracy. This can be seen in Fig. 6, where we compare the algorithms $T2S2$ and $T4S2$ as a

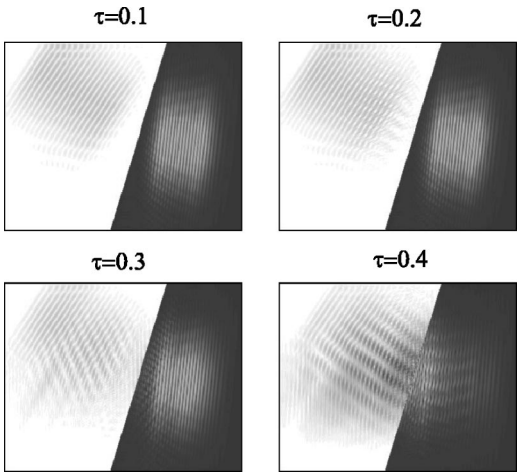

FIG. 4. Energy density distributions at simulation time $t=12.8$ for various time steps τ obtained by the $T4S2$ algorithm for a fixed mesh size $\delta=0.1$. The wave packet with initial parameters $(\sigma_x,\sigma_y)=(2,1.73)$, $(x_0,y_0)=(5,7.5)$, and $k=8$ [see for details Eq. (55) in Sec. V B] impinges on the dielectric structure from the left. The cavity measures 19×15 and contains a dielectric medium with $\varepsilon=2.25$ to the right of the line that connects the points (8.5,0) and (13,15). The origin (0,0) is located at the lower left corner. In vacuum the energy density distribution is plotted in black at locations of maximum intensity scaling and white at locations of zero intensity. Inside the dielectric medium this scheme is inverted.

function of the time step τ to show that the error in the EM field divergence vanishes for the $TnS2$ algorithm proportional to τ^n.

IV. VARIABLE GRID IMPLEMENTATION

The simple spatial implementation does not provide an optimal discretization scheme for physical systems of irregular geometrical shapes or with strongly varying permeability and/or permittivity. In a practical implementation of such

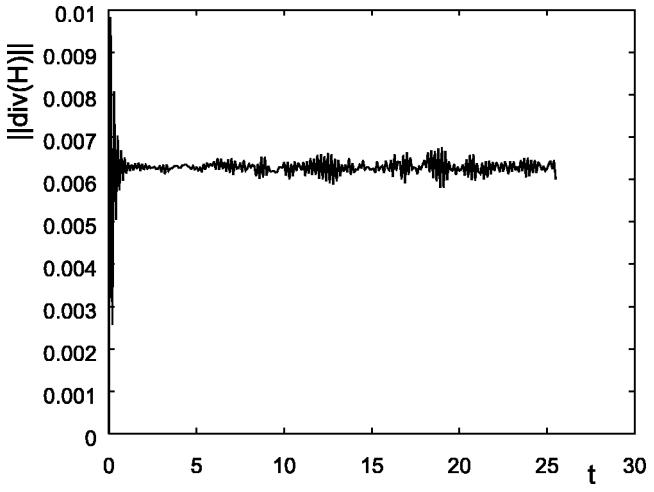

FIG. 5. The norm of the divergence of the magnetic field in a 3D empty cavity ($\varepsilon=1$ and $\mu=1$) of size $1.5\times1.5\times1.5$ as a function of time t. The computation is performed with the $T2S2$ algorithm keeping the mesh size $\delta=0.1$ fixed.

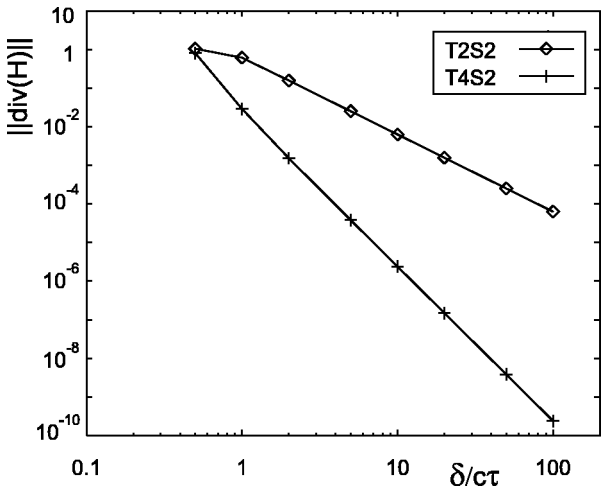

FIG. 6. The norm of the divergence of the magnetic field in a 3D empty cavity ($\varepsilon=1$ and $\mu=1$) of size $1.5\times1.5\times1.5$ as a function of $1/\tau$ for the fixed mesh size $\delta=0.1$. The computation is performed with the algorithms $T2S2$ and $T4S2$.

systems the grid has to be variable with a small mesh size in one region of the system and a large mesh size in another region of the system. In this section we show how to implement a variable grid in such a way that the algorithms to solve the time-dependent Maxwell equations remain unconditionally stable by construction.

For the sake of simplicity we consider a 1D system that is discretized using a variable grid as shown in Fig. 7. In a straightforward implementation of the variable grid we would replace the constant next-nearest neighbor distance δ in Eqs. (16) and (17) of the simple spatial implementation by the corresponding variable distance. It is convenient to write this substitution in the form

$$\delta\rightarrow\Delta_{i,i+1}\left[1+\frac{\delta_{i-1,i}-\delta_{i+1,i+2}}{2\Delta_{i,i+1}}\right], \tag{29}$$

where $\delta_{i,j}$ is the distance between grid points i and j (see Fig. 7) and

$$\Delta_{i,i+1}\equiv\frac{1}{2}(\delta_{i-1,i}+2\delta_{i,i+1}+\delta_{i+1,i+2}) \tag{30}$$

is the averaged next-nearest neighbor distance. It can be easily checked that an implementation of the variable grid that relies on the replacement Eq. (29) would destroy the skew-symmetry property of the corresponding matrix H [see Eq. (20)]. This is unphysical: The original form of the Maxwell equations do have this property. However, a variable grid implementation that does preserve the underlying symmetry

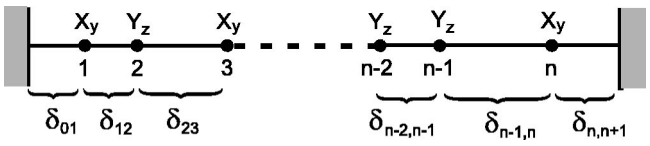

FIG. 7. Positions of the two TM-mode EM field components on the 1D variable grid.

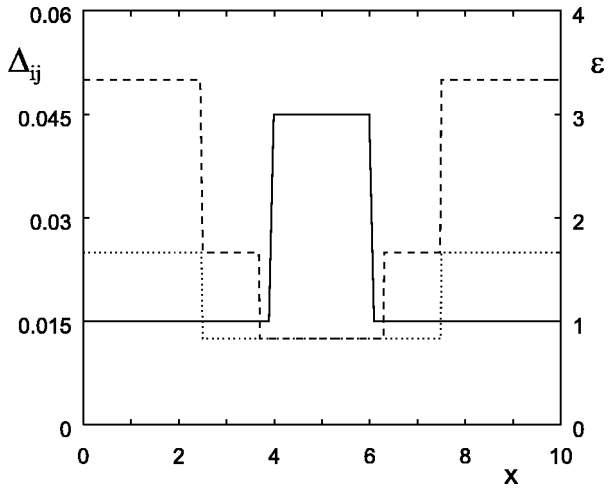

FIG. 8. The 1D cavity with the dielectric structure (solid line) and the two implemented variable grids: $\Delta_{i,i+1}=\{0.1\leftrightarrow 0.05\leftrightarrow 0.025\}$ (dashed line) and $\Delta_{i,i+1}=\{0.05\leftrightarrow 0.025\}$ (dotted line).

of Maxwell equations can be constructed for a sufficiently smooth, variable grid. In this case, the second term in the brackets of Eq. (29) may be neglected and the replacement

$$\delta \rightarrow \Delta_{i,i+1}=\Delta_{i+1,i} \tag{31}$$

may yield a resonable approximation of Eqs. (16) and (17) for the variable grid implementation:

$$\frac{\partial}{\partial t}X_y(i,t)=\frac{1}{\sqrt{\mu_i}}\left(\frac{Y_z(i+1,t)}{\Delta_{i,i+1}\sqrt{\varepsilon_{i+1}}}-\frac{Y_z(i-1,t)}{\Delta_{i,i-1}\sqrt{\varepsilon_{i-1}}}\right), \tag{32}$$

$$\frac{\partial}{\partial t}Y_z(i+1,t)=\frac{1}{\sqrt{\varepsilon_{i+1}}}\left(\frac{X_y(i+2,t)}{\Delta_{i+1,i+2}\sqrt{\mu_{i+2}}}-\frac{X_y(i,t)}{\Delta_{i+1,i}\sqrt{\mu_i}}\right). \tag{33}$$

The corresponding matrix H is seen to be skew symmetric,

$$H=\sum_{i=1}^{n}{}'\left[\frac{\mathbf{e}_i\mathbf{e}_{i+1}^T-\mathbf{e}_{i+1}\mathbf{e}_i^T}{\Delta_{i,i+1}\sqrt{\varepsilon_{i+1}\mu_i}}+\frac{\mathbf{e}_{i+1}\mathbf{e}_{i+2}^T-\mathbf{e}_{i+2}\mathbf{e}_{i+1}^T}{\Delta_{i+1,i+2}\sqrt{\varepsilon_{i+1}\mu_{i+2}}}\right], \tag{34}$$

and may again be separated into odd and even parts of which the exponents can be easily calculated following the same steps as given above in the simple spatial implementation.

It is obvious that this variable grid implementation can, in principle, be applied in any spatial dimension d. However, it is in general not possible to predict how to choose a grid that yields the best approximation to the true spectrum of eigenmodes of any nontrivial d-dimensional system. We, therefore, studied the criteria for the choice of suitable variable grids in particular systems numerically and present the results for a 1D and a 2D system in the remainder of this section.

The 1D system under consideration consists of a cavity of length $L=10$ with a constant permeability $\mu=1$ and a varying permittivity ε. The permittivity deviates from its vacuum value ($\varepsilon=1$) due to the presence of a dielectric medium with $\varepsilon=3$ that is located in the middle of the cavity and extends over a length of 2, as shown in Fig. 8. As a reference system we use a simple spatial implementation with constant next-nearest neighbor distance $\delta=0.025$ and calculate the eigenmodes ω_n of the corresponding matrix H. For two different variable grids we calculate the corresponding eigenmodes $\tilde{\omega}_n$ and the deviation $\Gamma(\omega_n,\tilde{\omega}_n)=1-\tilde{\omega}_n/\omega_n$ relative to the eigenmodes of the reference system. The two variable grids have in common that the dielectric medium and the transitions between $\varepsilon=1$ and $\varepsilon=3$ at both its sides is embedded in a grid of constant next-nearest neighbor distance that equals that of the reference system ($\Delta_{i,i+1}=\delta=0.025$). Furthermore, at the left end and at the right end of the cavity the next-nearest neighbor distance is constant over a length of 2.5 and equals, respectively, $\Delta_{i,i+1}=0.1$ and $\Delta_{i,i+1}=0.05$ in the two variable grids. The transitions in the variable grids between regions of constant next-nearest neighbor distance involve abrupt steps between

$$\Delta_{i,i+1}=0.1\leftrightarrow\Delta_{i,i+1}=0.05\leftrightarrow\Delta_{i,i+1}=0.025, \tag{35}$$

where we kept the intermediate distance $\Delta_{i,i+1}=0.05$ over eight grid points, and between

$$\Delta_{i,i+1}=0.05\leftrightarrow\Delta_{i,i+1}=0.025, \tag{36}$$

respectively.

In Fig. 9 we plot $\Gamma(\omega_n,\tilde{\omega}_n)$ for the first 50 eigenmodes of both variable grids. The relative deviation is seen to increase with the number of the frequency modes. As high mode numbers represent high frequencies this observation simply reflects the general fact that the accuracy of the eigenmodes depends on the smallness of the mesh size (numerical dispersion). Clearly, this also explains why the relative deviation $\Gamma(\omega_n,\tilde{\omega}_n)$ increases up to 2% for the variable grid with $\Delta_{i,i+1}=\{0.1\leftrightarrow 0.05\leftrightarrow 0.025\}$, while for the variable grid with $\Delta_{i,i+1}=\{0.05\leftrightarrow 0.025\}$ this deviation remains well below 0.5%. For the first few frequency modes, however, we observe an increase in $\Gamma(\omega_n,\tilde{\omega}_n)$. This behavior can be related to the error that is introduced in the variable grid implementation by applying the approximation, Eq. (31), instead of the exact replacement, Eq. (29). To check this statement we plot in Fig. 10 the deviation $\Gamma(\Omega_n,\tilde{\omega}_n)$ for the first 50 eigenmodes of the two variable grids relative to the eigenmodes Ω_n that belong to the variable grids of the exact implementation, Eq. (29). We see that the increase of the relative deviation for the first few eigenmodes is, in fact, related to the error that is made by replacing the exact substitution, Eq. (29), with the skew-symmetry conserving approximation, Eq. (31). This approximation leads to oscillations of $\Gamma(\Omega_n,\tilde{\omega}_n)$ [and also $\Gamma(\omega,\tilde{\omega}_n)$] that vanish with increasing frequency mode number. From extended numerical studies (results not shown) we find that these variations depend on several factors, such as the size in the difference between the largest and smallest distances $\Delta_{i,i+1}$ of the variable grid implementation and on how abrupt $\Delta_{i,i+1}$ changes with i. In practice, it will be necessary to check the robustness of numerical results obtained by a variable grid imple-

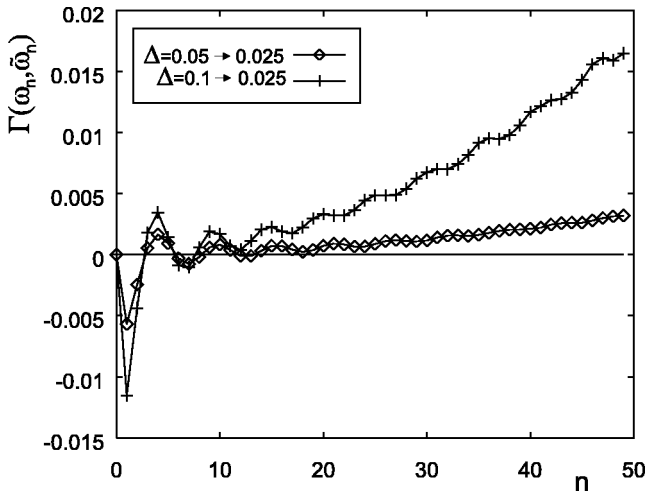

FIG. 9. Relative deviation $\Gamma(\omega_n,\tilde{\omega}_n)$ for two variable grids.

mentation against small changes in its parameters. Although this may sound as a serious disadvantage, the next example of a 2D system shows that for realistic applications it may be by far more efficient to perform several simulation runs with a variable grid implementation than to use the simple spatial implementation.

The 2D system we consider is given by the *L*-shaped cavity depicted in Fig. 11. In order to satisfy the conditions, Eq. (3), at the boundaries, the EM fields change very strongly close to the sharp edge of the cavity. Large spatial changes of the EM fields require a small mesh size. However, for the overwhelming part of the cavity a small mesh size would cause a waste of resources (computer memory and CPU time). Therefore, this system can be more efficiently simulated by a variable grid implementation with an increasing number of grid points near the edge. This is done by a uniform increase of the number of grid points along both the *x* and the *y* directions as is schematically drawn in Fig. 11. Furthermore, instead of using the odd-even decomposition of the time-evolution operator [corresponding to Eq. (24) for the 1D system] on a square grid that would contain grid

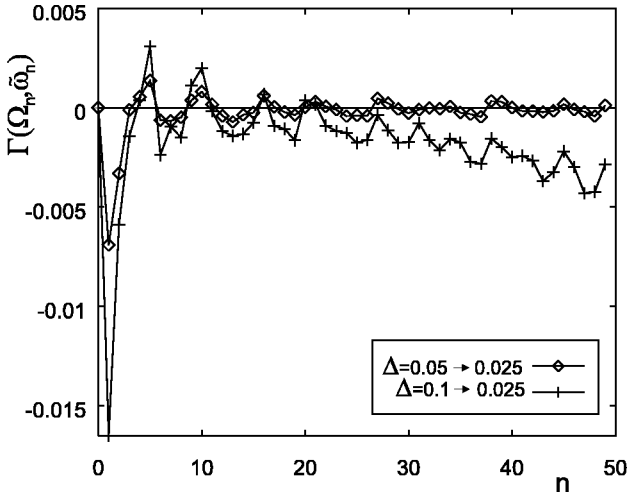

FIG. 10. Relative deviation $\Gamma(\Omega_n,\tilde{\omega}_n)$ for two variable grids.

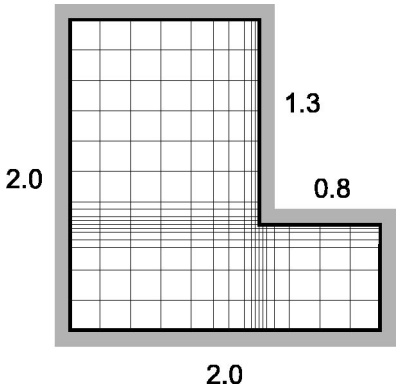

FIG. 11. The L-shaped 2D cavity with a variable grid (schematically).

points outside the L-shaped cavity, we perform the plane rotations by processing a list of pairs *S* of the EM field vector elements at the grid points that actually belong to the *L*-shaped cavity [corresponding to Eq. (26) for the 1D system].

In Table I we present the results of a numerical simulation for the eight lowest TM eigenmodes in the cavity. We used the *T*2*S*2 algorithm imposing a simple spatial implementation with $\delta=0.003\,125$ and a variable grid implementation with a mesh size ranging from $\Delta=0.05$ to $\Delta=0.003\,125$. Very similar to the procedure described above for the 1D system, the mesh size is decreased by a factor 0.5 and then kept constant for several grid points to smoothen this transition before the mesh size is decreased further. Our results are in good agreement with those obtained by the program package GDFIDL [18] for the same 2D system (see Table I). In Table II we show the location of the arbitrarily chosen third-lowest eigenmode ω_3 for several constant and variable grid implementations of the *T*2*S*2 algorithm. In all simulations we set $\delta/\tau=10$, where in the case of a variable grid δ is replaced by the smallest mesh size. The relative error Γ of the frequency ω_3 is measured with respect to the frequency $\omega_3=4.916$ of the system with constant mesh size $\delta=0.003\,125$. The numerical results obtained within the variable grid implementation are in excellent agreement with the results of the simple spatial implementation and the program

TABLE I. The eight lowest TM eigenmodes of the L-shaped cavity (see Fig. 11).

Mode *n*	*T*2*S*2		GDFIDL	
	Constant Grid ω_n	Variable Grid ω_n	Constant Grid ω_n	Variable Grid ω_n
1	2.9989	2.9913	2.9999	2.9992
2	3.9807	3.9500	3.9740	3.9720
3	4.9164	4.8857	4.9156	4.9102
4	5.4150	5.3843	5.4077	5.4004
5	5.5837	5.5453	5.5791	5.5710
6	6.0592	6.0209	6.0580	6.0494
7	6.7649	6.7265	6.7511	6.7377
8	6.8876	6.8492	6.8797	6.8674

TABLE II. Error in third-lowest eigenmode of the *L*-shaped cavity (see Fig. 11).

Constant grid δ	ω_3	Γ (in %)
0.1	4.571	7.5
0.05	4.740	3.7
0.025	4.832	1.7
0.0125	4.878	0.78
0.00625	4.901	0.31
0.003125	4.916	0
Variable grid Δ		
$0.1\rightarrow 0.05$	4.717	4.2
$0.1\rightarrow 0.025$	4.801	2.4
$0.1\rightarrow 0.0125$	4.840	1.6
$0.1\rightarrow 0.00625$	4.878	0.78
$0.05\rightarrow 0.003125$	4.886	0.61

package GDFIDL. The $T2S2$ algorithm with the simple spatial implementation and $\delta=0.003\,125$ consumes about 150 times more CPU time and 10 times more computer memory than the $T2S2$ algorithm with variable grid implementation and $\Delta=\{0.05\rightarrow 0.003\,125\}$. Clearly, these numbers justify additional simulation runs that are required to check the robustness of numerical results against small changes in the parameters of a variable grid implementation.

V. IMPROVED SPATIAL DISCRETIZATION IMPLEMENTATION

Both conditional FDTD algorithms and the unconditionally stable $TnSm$ algorithms suffer from numerical dispersion due to the discretization of continuum space on a grid with a finite mesh size [17]. Methods to reduce numerical dispersion are taking a grid with a smaller mesh size or employing more accurate finite-difference approximations to the spatial derivatives. The former obviously can be also used in the simple spatial implementation of unconditionally stable algorithms, however, for several reasons it may be more desirable to implement higher-order accurate approximations of the spatial derivatives. For example, if one is interested in global features of the distribution of a system's eigenmodes, i.e., if we want to determine *all* eigenvalues, a higher-order accurate spatial derivative implementation would be strongly preferred. The computation of a system's eigenmode spectrum is performed by calculating the Fourier transform of the inner product $F(t)=\langle \mathbf{\Psi}(0)|\mathbf{\Psi}(t)\rangle$ [1,19,20]. Using independent random numbers to initialize the elements of $\mathbf{\Psi}(0)$, the full eigenmode spectrum is obtained by averaging this Fourier transform. Taking just a smaller mesh size for the grid in the simple spatial implementation does not only reduce the numerical dispersion but also gives rise to more eigenmodes with high frequencies. In order to obtain the eigenmode spetrum with the same spectral resolution, the sampling of $F(t)$ would have to be done over smaller time intervals involving the computation of more data points. It is thus desirable to implement, instead, higher-order accurate approximations of the spatial derivatives that make a moderate use of computer resources in terms of CPU time and computer memory.

The procedure for the construction of higher-order approximations to spatial derivatives is standard [21]. In the present case, we apply this procedure keeping in mind that Maxwell equations (5) are skew symmetrical and that the electric and magnetic field components are defined at particular grid points. The grid of a *d*-dimensional system with a constant mesh size of distance $\delta/2$ between neighboring grid points is shown in Figs. 1–3. Without loss of generality we consider a 1D system, where $\Psi(i,t)=\Psi(i\,\delta/2,t)$ is the ith component of the EM field vector and denotes an electric field component for even i and a magnetic field component for odd i (see Sec. III for details). Applying the second-order accurate central-difference scheme the spatial derivative of the EM field component $\Psi(i,t)$ is given by

$$\frac{\partial}{\partial x}\Psi(i,t)=\frac{\Psi(i+1,t)-\Psi(i-1,t)}{\delta}-\frac{\delta^2}{6}\Psi^{(3)}(i,t)+O(\delta^4), \tag{37}$$

where $\Psi^{(3)}(i,t)\equiv\partial^3\Psi(i,t)/\partial x^3$. Similarly, using the third-nearest neighbor EM field points at distance $3\,\delta/2$, we have

$$\frac{\partial}{\partial x}\Psi(i,t)=\frac{\Psi(i+3,t)-\Psi(i-3,t)}{3\,\delta}-\frac{9\,\delta^2}{6}\Psi^{(3)}(i,t)+O(\delta^4). \tag{38}$$

A fourth-order accurate approximation of the spatial derivative $\partial\Psi(i,t)/\partial x$ is now constructed in terms of a linear combination of Eqs. (37) and (38), which is chosen such that the terms proportional to $\Psi^{(3)}(i,t)$ vanish. We obtain:

$$\frac{\partial}{\partial x}\Psi(i,t)=\frac{9}{8}\left(\frac{\Psi(i+1,t)-\Psi(i-1,t)}{\delta}\right)-\frac{1}{8}\left(\frac{\Psi(i+3,t)-\Psi(i-3,t)}{3\,\delta}\right)+O(\delta^4). \tag{39}$$

In practice, it is straightforward to implement the improved spatial discretization, since we can use the implementation of the central-difference scheme for the two terms separately and then combine the results according to Eq. (39). The corresponding matrix H of the 1D system [see Eq. (20)] changes from tridiagonal to pentadiagonal, but most importantly it preserves its property of being skew symmetric. It should be noted, however, that the fourth-order accurate spatial derivative introduces errors at the boundaries since the calculation of $\partial\Psi(i,t)/\partial x$ for $i=1,\ 2,\ n-1$, and n refer, respectively, to grid points $i=-2,\ -1,\ n+1$, and $n+2$ that lie outside the cavity and are implicitly assumed to be zero.

It is obvious that the fourth-order accurate approximation of the spatial derivatives can be similarly applied in systems of any spatial dimension *d*. In the remainder of this section we study the numerical dispersion and the temporal and spatial accuracies of the algorithms for various 1D and 2D systems.

A. Numerical dispersion

We illustrate the difference in the numerical dispersion between the simple spatial implementation and the improved spatial discretization implementation by a comparison of the eigenmode spectra of a 1D empty cavity ($\varepsilon=1$ and $\mu=1$) of length L. In 1D, the continuum wave equation for the EM fields [2],

$$\left[\frac{1}{c^2}\frac{\partial^2}{\partial t^2}-\frac{\partial^2}{\partial x^2}\right]\Psi(x,t)=0, \tag{40}$$

is solved by the ansatz $\Psi(x,t)\propto\cos(\omega t-kx+\phi)$ (with a phase ϕ to distinguish electrical and magnetic field components) yielding the linear dispersion relation between frequency ω and wave number k; $\omega=c|k|$. Focusing on the effect of the spatial derivatives on the numerical dispersion, we assume perfect time integration of the algorithms and impose periodic boundary conditions on the EM field components: $\Psi_p(i,t)\propto\cos(\omega_p t-k_p\delta/2+\phi)$ with wave number $k_p=2\pi p/L$ and $-L/(2\delta)<p\leqslant L/(2\delta)$. Applying the second-order accurate spatial derivative we obtain

$$\frac{\partial^2}{\partial x^2}\Psi_p(i,t)=\frac{1}{\delta^2}[\Psi_p(i+2,t)-2\Psi_p(i,t)+\Psi_p(i-2,t)]+\mathcal{O}(\delta^2), \tag{41}$$

while for the fourth-order accurate spatial derivative we find

$$\begin{aligned}\frac{\partial^2}{\partial x^2}\Psi_p(i,t)=&\left(\frac{9}{8\delta}\right)^2[\Psi_p(i+2,t)-2\Psi_p(i,t)+\Psi_p(i-2,t)]\\&+\left(\frac{1}{24\delta}\right)^2[\Psi_p(i+6,t)-2\Psi_p(i,t)\\&+\Psi_p(i-6,t)]+\left(\frac{9}{96\delta^2}\right)[\Psi_p(i+2,t)+\Psi_p(i\\&-2,t)-\Psi_p(i+4,t)-\Psi_p(i-4,t)]+O(\delta^4).\end{aligned} \tag{42}$$

For $m=2$ the analytical solution of the eigenmode spectrum for the mth-order accurate spatial derivative is given by

$$\omega_p^2=2\left(\frac{c}{\delta}\right)^2[1-\cos(k_p\delta)], \tag{43}$$

while for $m=4$ we find

$$\omega_p^2=\left(\frac{c}{\delta}\right)^2\sum_{l=0}^{3}C_l\cos(lk_p\delta), \tag{44}$$

with coefficients $C_0=365/144$, $C_1=-87/32$, $C_2=3/16$, and $C_3=-1/288$. We show in Fig. 12 that the dispersion relations that we obtained numerically by the mth-order accurate spatial derivative implementation for a 1D cavity of length $L=4$, are in excellent agreement with the corresponding analytical solutions, Eqs. (43) and (44). It is clearly visible that the dispersion relation computed by the simple spatial implementation ($T2S2$ algorithm) suffers from numerical dispersion already at frequencies above $\omega=10$, whereas for a grid with the same mesh size the fourth-order accurate spatial derivative implementation ($T2S4$ algorithm) works well up to $\omega=15$.

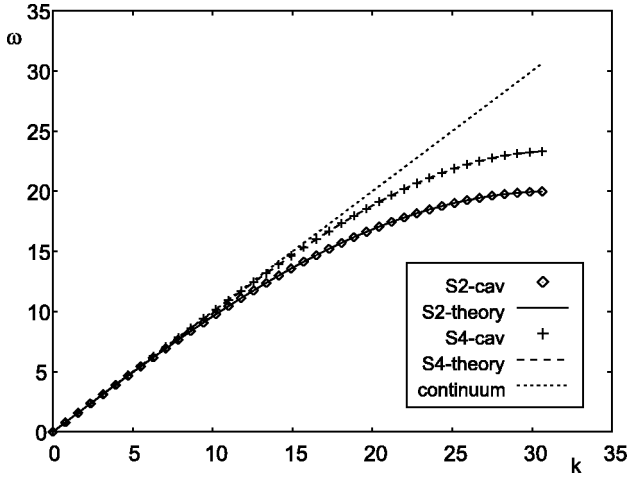

FIG. 12. Numerical and analytical dispersion relations for the 1D cavity of length $L=4$ as obtained from calculations with mth-order accurate approximations of the spatial derivatives ($m=2,4$). In both simulations we kept $\delta=0.1$ and $\tau=0.01$ fixed.

B. Temporal and spatial accuracies

To perform a systematic study of the accuracy of the algorithms as a function of the time step τ and the mesh size δ, we compute the difference between the normalized exact, $\mathbf{\Psi}(t)$, and the approximate, $\mathbf{\Psi}_{n,m}(t)$, EM field vectors as obtained by the $TnSm$ algorithm:

$$\Delta\mathbf{\Psi}_{n,m}(t)\equiv\|\mathbf{\Psi}(t)-\mathbf{\Psi}_{n,m}(t)\|. \tag{45}$$

We first consider the propagation of a Gaussian wave packet in a 1D empty cavity ($\varepsilon=1$ and $\mu=1$) of length $L=30$. At $t=0$ the Gaussian wave packet

$$E_z(x,t)=\exp[-(x-x_0-ct)^2/\sigma^2] \tag{46}$$

with standard deviation $\sigma=2$ is located at $x_0=8$. For $t>0$ the wave packet propagates with velocity c in the x direction until it hits the right boundary of the cavity, becomes reflected, and propagates in the opposite direction. To derive an analytical expression of the exact EM field vector $\mathbf{\Psi}(t)$, we expand $E_z(x,t)$ in the TM modes,

$$E_z(x,t)=-\sum_{n=1}^{\infty}a_n\sin(n\pi x/L)\sin[n\pi(x_0+ct)/L], \tag{47}$$

$$H_y(x,t)=\frac{a_0}{2}+\sum_{n=1}^{\infty}a_n\cos(n\pi x/L)\cos[n\pi(x_0+ct)/L], \tag{48}$$

with coefficients

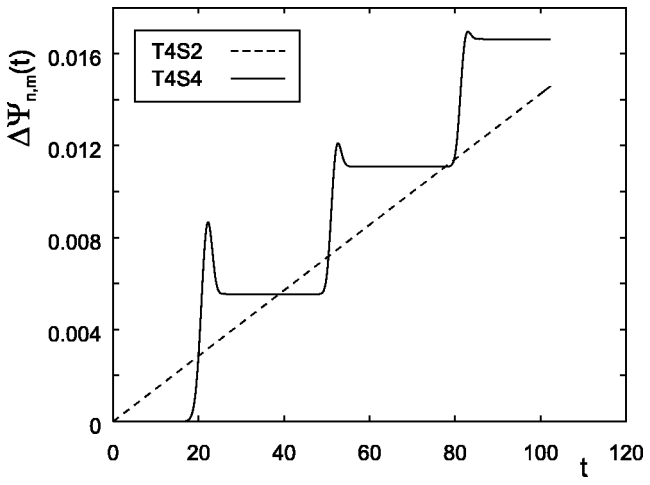

FIG. 13. The error $\Delta\mathbf{\Psi}_{n,m}(t)$ as a function of the simulation time t for fixed values of the mesh size $\delta=0.1$ and the time step $\tau=0.01$. Results are shown for the $T4S2$ and $T4S4$ algorithms.

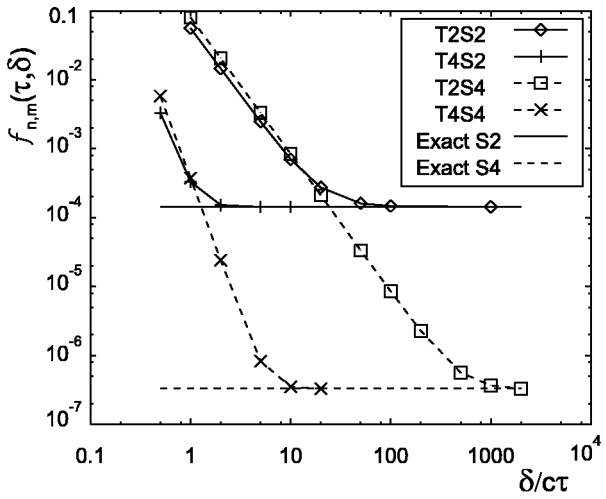

FIG. 14. $f_{n,m}(\tau,\delta)$ as a function of $1/\tau$ for the fixed mesh size $\delta=0.1$.

$$a_n=\frac{2\sigma\sqrt{\pi}}{L}\exp\left[-\frac{\sigma^2}{4}\left(\frac{n\pi}{L}\right)^2\right], \tag{49}$$

which ensure that the wave packet satisfies the boundary conditions, Eq. (3). Using Poisson's summation formula we find the following expressions for the EM field components:

$$E_z(x,t)=\sum_{n=-\infty}^{\infty}\{\exp[-(2nL+x+x_0-ct)^2/\sigma^2] - \exp[-(2nL+x-x_0+ct)^2/\sigma^2]\}, \tag{50}$$

$$H_y(x,t)=\sum_{n=-\infty}^{\infty}\{\exp[-(2nL+x+x_0-ct)^2/\sigma^2] + \exp[-(2nL+x-x_0+ct)^2/\sigma^2]\}, \tag{51}$$

from which the exact EM field vector $\mathbf{\Psi}(t)$ is constructed according to Eq. (18) on the 1D grid (see Fig. 1).

In Fig. 13 we plot $\Delta\mathbf{\Psi}_{n,m}(t)$ as a function of the simulation time t for fixed values of the mesh size δ and the time step τ using both the $T4S2$ and the $T4S4$ algorithms. We find that the error increases roughly proportional to the simulation time:

$$\Delta\mathbf{\Psi}_{n,m}(t)=2f_{n,m}(\tau,\delta)t, \tag{52}$$

where we used the prefactor 2 to ensure that $0\leqslant f_{n,m}(\tau,\delta)\leqslant 1$. The linear dependence of $\Delta\mathbf{\Psi}_{n,m}(t)$ on t is clearly visible only for the $T4S2$ algorithm but is also true for the $T4S4$ algorithm with a much smaller slope $f_{4,4}(\tau,\delta)$. Only at particular times t when the wave packet hits the boundaries of the cavity, the error $\Delta\mathbf{\Psi}_{4,4}(t)$ is seen to increase nonlinearly in the time t and takes a value that is of the same order as $\Delta\mathbf{\Psi}_{4,2}(t)$. This behavior, not described by Eq. (52), is present in fourth-order accurate spatial derivative implementations, in which the calculation of the EM field components close to system boundaries refer to several nonexisting grid points. To study the error $\Delta\mathbf{\Psi}_{n,m}(t)$ as a function of the time step τ and the mesh size δ, we compute

$$f_{n,m}(\tau,\delta)=\frac{1}{2}\frac{d}{dt}\Delta\mathbf{\Psi}_{n,m}(t). \tag{53}$$

In Fig. 14 we plot $f_{n,m}(\tau,\delta)$ as obtained for the 1D cavity by the four algorithms $T2S2$, $T4S2$, $T2S4$, and $T4S4$ as a function of $1/\tau$ for a fixed mesh size δ. For each algorithm $TnSm$ we find a linear decrease of $\log[f_{n,m}(\tau,\delta)]$ with increasing values $\log[1/\tau]$. For the algorithms $T4S2$ and $T4S4$ we find that $f_{4,m}(\tau,\delta)\propto\tau^4$, while for the $T2S2$ and $T2S4$ algorithms $f_{2,m}(\tau,\delta)\propto\tau^2$. This numerical result is in agreement with the rigorous upper bound on the error of the EM field vector, Eq. (13). For decreasing values of τ, the error in the time integration becomes negligibly small and $f_{n,m}(\tau,\delta)$ reaches minimum values that are indicated by the two lines "exact $S2$" for the algorithms $TnS2$ and "exact $S4$" for the algorithms $TnS4$. In fact, these two lines represent the numerical results that are obtained for an exact time integration and mth-order accurate approximations to the spatial derivatives.

Next, we study $f_{n,m}(\tau,\delta)$ as a function of the mesh size δ for the fixed ratio $\tau/\delta=0.1$ to ensure that the accuracy of the time integration remains constant. The numerical results are plotted in Fig. 15. We see that $\log[f_{n,m}(\tau,\delta)]$ decreases linearly with increasing $\log[1/\delta]$ until it levels off. At this point, the total number of operations has become so large that it causes the numerical loss of accuracy. Outside this regime we find for the $TnS4$ algorithms $f_{n,4}(\tau,\delta)\propto\delta^4$ and for the $TnS2$ algorithms $f_{n,2}(\tau,\delta)\propto\delta^2$. In analogy to the upper bound, Eq. (13), the upper bound for the mth-order accurate approximation of the spatial derivatives is given by

$$\|\mathbf{\Psi}(t)-\mathbf{\Psi}_{n,m}(t)\|\leqslant C_{n,m}t\delta^m, \tag{54}$$

where $C_{n,m}$ is a constant.

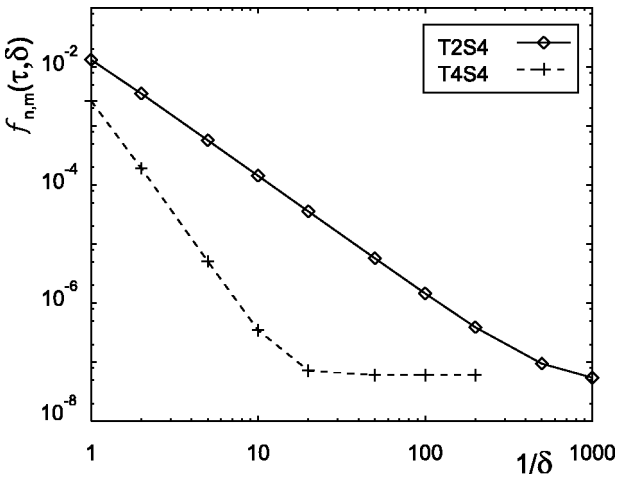

FIG. 15. $f_{n,m}(\tau,\delta)$ as a function of $1/\delta$ for the fixed ratio $\tau/\delta = 0.1$.

We consider a second example to illustrate the numerical performance of the algorithms in 2D systems. For the initial wave packet in the 2D cavity we make the ansatz

$$E_z(x,y,t) = \sin(k(x-x_0-ct))\exp[-((x-x_0-ct)/\sigma_x)^{10} - ((y-y_0)/\sigma_y)^2]. \quad (55)$$

At $t=0$ the wave packet is centered at (x_0,y_0) and moves at $t>0$ with velocity c in the x direction. The energy of the wave packet is fixed by the wave number k in the oscillating factor and its envelope is Gaussian along the y direction and has sharp edges along the x axis (due to the exponent 10). The 2D cavity of size 12×10 with $\varepsilon=1$ and $\mu=1$ contains two objects with dielectric constants $\varepsilon=5$ and $\mu=1$. The parameters of the propagating wave packet are $(\sigma_x,\sigma_y) = (1.66,1.29)$, $(x_0,y_0)=(3.5,5.5)$, and $k=5$. In Fig. 16 we show the results for the error, Eq. (45), of the $T2S2$ and $T2S4$ algorithms with different mesh sizes relative to a reference EM field vector $\mathbf{\Psi}(t)$ that is obtained from the $T2S2$ algorithm at mesh size $\delta=0.025$. In all simulations we kept $\tau=0.1\delta$ fixed to compare measurements of constant accuracy in the time integration. In Fig. 16 we show (a) the energy distribution of the initial wave packet ($t=0$) and (b) the reference energy density distribution after simulation time $t=6$ using the $T2S2$ algorithm. In Fig. 16(c)–(e), the normalized spatial distribution of the error in the energy density distribution,

$$\Delta w_{n,m}(\mathbf{r},t) = |\mathbf{\Psi}(\mathbf{r},t)^2 - \mathbf{\Psi}_{n,m}(\mathbf{r},t)^2|, \quad (56)$$

is shown for, respectively, the algorithm $T2S2$ with $\delta=0.1$, the algorithm $T2S2$ with $\delta=0.05$, and the algorithm $T2S4$ with $\delta=0.1$. We find that the improved spatial discretization implementation $T2S4$ with $\delta=0.1$ performs as well as a simple spatial implementation $T2S2$ with half the mesh size. The main advantage of using the $T2S4$ algorithm is that it used only 20% of the computer memory and 10% of the CPU time with respect to the $T2S2$ algorithm.

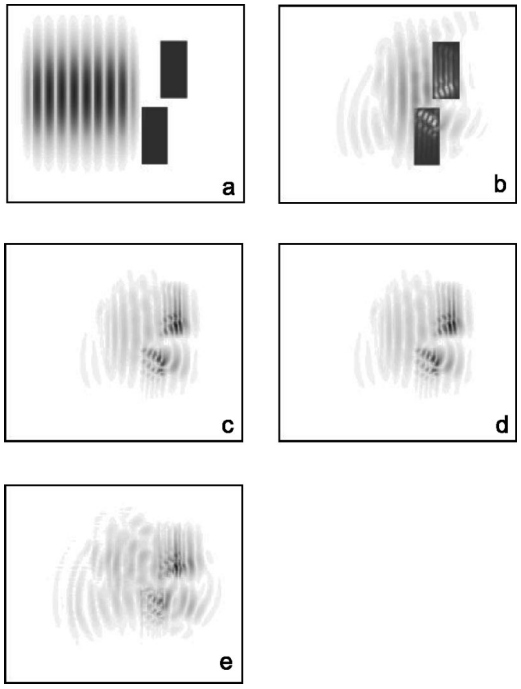

FIG. 16. The error $\Delta w_{n,m}(\mathbf{r},t)$ for various mesh sizes and algorithms in a 2D cavity of size 12×10. The origin (0,0) is located at the lower left corner. Two rectangular blocks of a dielectric medium with $\varepsilon=5$ are located at the (lower left)/(upper right) coordinates (6.8,1.8)/(8.2,4.8) and (7.8,5.2)/(9.2,8.2), respectively. In vacuum the energy density distribution is plotted in black at locations of maximum intensity scaling and white at locations of zero intensity. Inside the dielectric medium this scheme is inverted. (a) Initial energy density distribution $\mathbf{\Psi}(\mathbf{r},t)^2$. (b) Reference energy density distribution $\mathbf{\Psi}_{2,2}(\mathbf{r},t)^2$ at $t=6$ using the $T2S2$ algorithm with $\delta=0.025$. (c) The error $\Delta w_{2,2}(\mathbf{r},t)$ on the energy density distribution at $t=6$ using the $T2S2$ algorithm with $\delta=0.1$. The relative deviation is 26%. (d) The error $\Delta w_{2,2}(\mathbf{r},t)$ on the energy density distribution at $t=6$ using the $T2S2$ algorithm with $\delta=0.05$. The relative deviation is 5.9%. (e) The error $\Delta w_{2,4}(\mathbf{r},t)$ on the energy density distribution at $t=6$ using the $T2S4$ algorithm with $\delta=0.1$. The relative deviation is 5.1%.

VI. CONCLUSIONS

We have demonstrated that the previously introduced family of unconditionally stable algorithms to solve the time-dependent Maxwell equations can be implemented with a grid of variable mesh size and with a fourth-order accurate approximation to the spatial derivatives. The performance of the algorithms has been shown to increase significantly as compared to the previously applied simple spatial implementation while at the same time their property of unconditional stability by construction is preserved. Performing numerical simulations on various physical model systems, we found that a variable grid implementation can save orders of magnitude in computer memory and CPU time for a physical system of unregular geometrical shape or with strongly varying permeability and/or permittivity. Similar enhancements have been obtained for the fourth-order accurate spatial derivative implementation that does not only reduce the nu-

merical dispersion but also improves the temporal and spatial accuracies of the algorithms significantly. Clearly, in close analogy to the implementation of the fourth-order approximation of the spatial derivatives, the algorithms may be improved by constructing higher-order approximations. In general, we conclude that the family of unconditionally stable algorithms does not only preserve the fundamental symmetries of the time-dependent Maxwell equations but is also characterized by a high degree of flexibility that allows one to construct implementations that are required in different kinds of specific applications.

ACKNOWLEDGMENTS

This work was partially supported by the Dutch ''Stichting Nationale Computer Faciliteiten'' (NCF). We thank W. Bruns for providing numerical results generated by the program package GDFIDL.

[1] J. S. Kole, M. T. Figge, and H. De Raedt, Phys. Rev. E **64**, 066705 (2001).
[2] M. Born and E. Wolf, *Principles of Optics* (Pergamon, Oxford, 1964).
[3] H. F. Trotter, Proc. Am. Math. Soc. **10**, 545 (1959).
[4] M. Suzuki, S. Miyashita, and A. Kuroda, Prog. Theor. Phys. **58**, 1377 (1977).
[5] M. Suzuki, J. Math. Phys. **26**, 601 (1985); **32**, 400 (1991).
[6] H. De Raedt and B. De Raedt, Phys. Rev. A **28**, 3575 (1983).
[7] H. De Raedt, Comput. Phys. Rep. **7**, 1 (1987).
[8] H. Kobayashi, N. Hatano, and M. Suzuki, Physica A **211**, 234 (1994).
[9] H. De Raedt and K. Michielsen, Comput. Phys. **8**, 600 (1994).
[10] A. Rouhi, J. Wright, Comput. Phys. **9**, 554 (1995).
[11] B. A. Shadwick and W. F. Buell, Phys. Rev. Lett. **79**, 5189 (1997).
[12] M. Krech, A. Bunker, and D. P. Landau, Comput. Phys. Commun. **111**, 1 (1998).
[13] P. Tran, Phys. Rev. E **58**, 8049 (1998).
[14] K. Michielsen, H. De Raedt, J. Przeslawski, and N. Garcia, Phys. Rep. **304**, 89 (1998).
[15] H. De Raedt, A. H. Hams, K. Michielsen, and K. De Raedt, Comput. Phys. Commun. **132**, 1 (2000).
[16] K. S. Yee, IEEE Trans. Antennas Propag. **14**, 302 (1966).
[17] A. Taflove and S. C. Hagness, *Computational Electrodynamics—The Finite-Difference Time-Domain Method* (Artech House, Boston, 2000).
[18] For information about the program package GDFIDL, see: http://www.gdfidl.de
[19] R. Alben, M. Blume, H. Krakauer, and L. Schwartz, Phys. Rev. B **12**, 4090 (1975).
[20] A. Hams and H. De Raedt, Phys. Rev. E **62**, 4365 (2000).
[21] See F. Scheid, *Theory and Problems of Numerical Analysis* (McGraw-Hill, New York, 1968), Chap. 13.