



Contents lists available at ScienceDirect

# Journal of Computational and Applied Mathematics

journal homepage: [www.elsevier.com/locate/cam](http://www.elsevier.com/locate/cam)

## A new high order dispersive FDTD method for Drude material with complex interfaces



Duc Duy Nguyen, Shan Zhao\*

Department of Mathematics, University of Alabama, Tuscaloosa, AL 35487, USA

### ARTICLE INFO

#### Article history:

Received 27 August 2014

Received in revised form 3 November 2014

#### MSC:

65M06

78A48

78M20

#### Keywords:

Finite-difference time-domain (FDTD)

Maxwell's equations

Drude dispersive material

Transverse magnetic (TM) modes

High order interface treatments

Matched interface and boundary (MIB)

### ABSTRACT

In this paper, motivated by the needs of tracking the transient change in the regularity of the electromagnetic fields across a Drude interface, we propose a new Maxwell–Drude formulation for transverse magnetic problems with inhomogeneous Drude dispersive materials. Based on the auxiliary differential equation approach, the proposed formulation couples the wave equation for the electric component with Maxwell's equations for the magnetic components. A new finite-difference time-domain (FDTD) algorithm is introduced for solving the proposed Maxwell–Drude system, in which the time dependent jump conditions across the Drude interface are enforced through the matched interface and boundary (MIB) method. The proposed FDTD method achieves a second order of accuracy in solving Drude interfaces with fluctuating curvatures and non-smooth corners based on a simple Yee lattice, while it can be generalized to up to sixth order in dealing with a straight Drude interface. Therefore, the proposed FDTD method is more accurate and cost-efficient than the classical FDTD methods for Drude material with complex interfaces.

© 2015 Elsevier B.V. All rights reserved.

### 1. Introduction

In the electromagnetics, a medium is called dispersive if its permittivity or permeability depends on the wave frequency. When an electromagnetic wave is propagating in a dispersive medium, different frequencies will travel at different speeds. The time domain dispersion simulation is indispensable for many practical applications involving dispersive materials. For example, in the ground penetrating radar [1] that uses the transmittance of pulses to image the subsurface in order to detect buried objects, the geological media are dispersive. Similarly, the microwave imaging has been used for the early detection of breast cancer [2], in which the biological tissues are dispersive. In the near-field optics, the scattering of light by metallic nano-structures is of great interests for the sub-wavelength imaging and sensing. At optical frequencies, metals shall be described as dispersive [3].

Many novel dispersive approaches have been developed in the literature to simulate the wave propagation in dispersive media, through solving Maxwell's equations in the time domain, such as finite-difference time-domain (FDTD) methods [4–8], finite element time domain methods [9–13], discontinuous Galerkin time domain methods [14–18], and so on. A major focus in these developments is on how to incorporate the frequency dependent material constitutive relations into the time domain Maxwell formulation. Nevertheless, less attention has been paid on how to treat dispersive interfaces of complex shapes in the time domain Maxwell solvers. Because the wave solutions are non-smooth across a dispersive interface, the absence of a proper interface treatment often degrades the accuracy of a Maxwell solver to just first order.

\* Corresponding author. Tel.: +1 205 3485303; fax: +1 205 3487067.

E-mail address: [szhao@ua.edu](mailto:szhao@ua.edu) (S. Zhao).

It is well known that the FDTD method has been a main workhorse of the computational electromagnetics (CEM) in the time domain [19], owing to its simplicity, lack of dissipative error, and having very low cost per grid node. However, comparing with other dispersive approaches, the dispersive FDTD algorithms [4–8] suffer from a larger accuracy reduction at a dispersive interface, due to a staircase representation for complex geometry. To reduce the staircase error of the dispersive FDTD simulation, several smoothing techniques [20–22] have been constructed to approximate the permittivities in the vicinity of the interface by an averaged smooth effective permittivity. In [23], a domain decomposition method based on overlapping grids has been introduced to effectively treat curved dispersive interfaces, so that the staircase error could be eliminated in the dispersive FDTD computation.

To fully restore the accuracy near a dielectric interface, besides the removal of the staircase error, the physical jump conditions defined on the material interface must also be satisfied in the Maxwell discretization [24–26]. For nondispersive interfaces, many high order embedded FDTD methods [24] have been successfully developed to enforce the jump conditions in the FDTD discretization. However, these high order embedded FDTD methods [24] cannot be directly generalized to treat a dispersive interface, because the jump conditions become time variant for the dispersive interface [27–29].

Recently, we have introduced a high order FDTD method for dealing with Debye type dispersive interfaces, based on the matched interface and boundary (MIB) scheme [27–29]. This MIB time domain (MIBTD) approach is not only free of staircase error in accommodating curved interfaces, but also can deliver a second order of accuracy in treating inhomogeneous Debye media. The success of the MIBTD approach lies in a new modeling of the time dependent jump conditions and a new Maxwell–Debye system [27–29]. However, the previous MIBTD development only focused on Debye materials. Additional issues need to be addressed in extending it to the Drude materials. In particular, the auxiliary differential equation (ADE) of the Drude material is a second order differential equation while that of the Debye medium is just first order. We note that the Drude model may be decomposed into a linear combination of a Debye model and a constant-conductivity model, so that only first order ADEs need to be solved numerically. However, such an approach involves not only electromagnetic fields, but also the current density. This introduces grand difficulty to the MIBTD modeling, because the jump conditions of the current density are generally unknown. A second order ADE model for the Drude material is thus more tractable to the MIBTD method, in which jump conditions are only enforced for electric and magnetic fields. To the authors' knowledge, no high order interface treatment has ever been developed to deal with the second order Drude materials in the FDTD literature.

The objective of this paper is twofold. First, a novel hybrid Maxwell–Drude transverse magnetic (TM) formulation will be constructed, which couples the wave equation for the electric component with Maxwell's equations for the magnetic components. This hybrid formulation enables an accurate tracking of the transient change in the regularity of the electromagnetic fields across the Drude interface. Second, a high order FDTD method based on the MIB scheme will be developed for solving two-dimensional (2D) TM Maxwell's equations with irregularly shaped Drude interfaces based on a simple Cartesian grid. The enforcement of the time dependent jump conditions essentially fits the finite difference weights to the dispersive interfaces so that the staircase error is eliminated.

This paper is organized as follows. In Section 2, a brief introduction to Maxwell theory and Drude dispersive model is first given. A new Maxwell–Drude formulation will then be proposed. In Section 3, to facilitate the interface treatments, we derive the jump conditions across a Drude interface in both local coordinates and Cartesian coordinates. The development of the MIBTD algorithms for smooth and non-smooth interfaces will be presented in Section 4. The performance of the proposed MIBTD method for Drude interface problems is verified in Section 5. Finally, this paper ends with a conclusion.

## 2. Maxwell–Drude formulation

Consider a linear isotropic electromagnetic structure involving dispersive materials. Assume that both the structure and incident wave are invariant in the  $z$  direction, so that all  $z$  derivatives of electromagnetic fields can be removed in Maxwell's equations. Then, the resulting 2D equations can be decomposed into two independent sets, i.e., the TM mode and the transverse electric (TE) mode. In this work, we will focus only on the TM system in some inhomogeneous media

$$\frac{\partial D_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}, \quad \frac{\partial H_y}{\partial t} = \frac{1}{\mu} \frac{\partial E_z}{\partial x}, \quad \frac{\partial H_x}{\partial t} = -\frac{1}{\mu} \frac{\partial E_z}{\partial y}, \quad (1)$$

where  $E_z$  is the electric field component,  $H_x$  and  $H_y$  are, respectively, the magnetic field  $x$  and  $y$  components, and  $D_z$  is the electric displacement component.

In the present study, we assume the materials to be non-magnetic, i.e., the magnetic permeability  $\mu = \mu_0$ . In a dispersive medium, the electric constitutive relation is expressed in the frequency domain

$$\hat{D}_z = \hat{\epsilon}(\omega) \hat{E}_z, \quad (2)$$

where  $\epsilon$  is the electric permittivity, and the time-harmonic components are obtained via the Fourier transforms of the time-varying components. The single-pole Drude dispersion model is considered in this paper [3]

$$\hat{\epsilon}(\omega) = \epsilon_0 \left[ \epsilon_\infty - \frac{\omega_1^2}{\omega^2 - i\omega\gamma} \right], \quad (3)$$

where  $i$  is the imaginary unit and  $\omega$  is the angular frequency. Here,  $\omega_1$  is the Drude pole frequency,  $\gamma$  is the inverse of the pole relaxation time constant, and  $\epsilon_0$  and  $\epsilon_\infty$  are, respectively, the permittivities of free space and at high frequency limit.

By using the auxiliary differential equation (ADE) approach [4], the Drude constitutive equation can be represented in the time domain

$$\frac{\partial^2 D_z}{\partial t^2} + \gamma \frac{\partial D_z}{\partial t} = \epsilon_0 \epsilon_\infty \frac{\partial^2 E_z}{\partial t^2} + \epsilon_0 \epsilon_\infty \gamma \frac{\partial E_z}{\partial t} + \epsilon_0 \omega_1^2 E_z. \tag{4}$$

The classical TM system formed by Eqs. (1) and (4) is a closed system of four components  $D_z$ ,  $E_z$ ,  $H_x$ , and  $H_y$ , and its finite difference discretization yields a dispersive FDTD algorithm for Drude materials.

In this paper, motivated by interface considerations, we propose a new Maxwell–Drude system

$$\begin{aligned} \frac{\partial E_z}{\partial t} &= \dot{E}_z, & \frac{\partial H_y}{\partial t} &= \frac{1}{\mu_0} \frac{\partial E_z}{\partial x}, & \frac{\partial H_x}{\partial t} &= -\frac{1}{\mu_0} \frac{\partial E_z}{\partial y}, \\ \frac{\partial \dot{E}_z}{\partial t} &= -\gamma \dot{E}_z - \frac{\omega_1^2}{\epsilon_\infty} E_z + \frac{1}{\mu_0 \epsilon_0 \epsilon_\infty} \left( \frac{\partial^2 E_z}{\partial x^2} + \frac{\partial^2 E_z}{\partial y^2} \right) + \frac{\gamma}{\epsilon_0 \epsilon_\infty} \left( \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \right). \end{aligned} \tag{5}$$

Obviously, this new system is a hybrid one coupling the first order equations for  $H_x$  and  $H_y$  with the wave equation for  $E_z$ . Its equivalence to the classical Maxwell–Drude system can be easily established within a homogeneous Drude material. For an inhomogeneous problem involving a dispersive interface, physical interface conditions must be satisfied in the numerical discretization. We note that three Drude parameters  $\epsilon_\infty$ ,  $\omega_1$ , and  $\gamma$  could be discontinuous across the dispersive interface.

### 3. Interface jump conditions

A typical Drude interface problem will be studied in this paper. Consider a 2D domain of a rectangle shape  $\Omega = [a, b] \times [c, d]$ . Assume that an interface  $\Gamma$  separates the domain  $\Omega$  into two parts  $\Omega^+$  and  $\Omega^-$ , where  $\Gamma$  could be either a straight line or a curve. For two media on both hand sides of  $\Gamma$ , if one of them is a Drude medium, we will refer to  $\Gamma$  as a Drude interface. For simplicity, we will assume the other medium, say  $\Omega^+$ , being the air or vacuum in the present study. Define the jump of a function  $u(x, y)$  at  $\Gamma$  to be  $[u] := u^+ - u^-$ , where the superscript,  $-$  or  $+$ , denotes the limiting value of function from one side or from the other side of the interface. For the present Drude–air dispersive interface, we assume the material equation (3) in the vacuum  $\Omega^+$  as well with  $\epsilon_\infty^+ = 1$  and  $\omega_1^+ = 0$ . Then, the high frequency limit  $\epsilon_\infty$  and the Drude pole frequency  $\omega_1$  in (4) are both discontinuous across the interface  $\Gamma$ . Here,  $\gamma^+$  is a free parameter so that we can assume  $\gamma = \gamma^- = \gamma^+$  being a constant throughout  $\Omega$ .

Across the interface  $\Gamma$ , field solutions in both media are related analytically via the jump conditions [19]

$$\begin{aligned} \vec{n} \times (\mathbf{E}^+ - \mathbf{E}^-) &= \mathbf{0}, & \vec{n} \cdot (\mathbf{D}^+ - \mathbf{D}^-) &= 0, \\ \vec{n} \times (\mathbf{H}^+ - \mathbf{H}^-) &= \mathbf{0}, & \vec{n} \cdot (\mu \mathbf{H}^+ - \mu \mathbf{H}^-) &= 0, \end{aligned} \tag{6}$$

where  $\vec{n}$  is the unit vector normal to the interface, pointing from  $\Omega^-$  into  $\Omega^+$ . Denote  $\vec{\tau}$  as the tangential direction at an interface point. A local Cartesian coordinate  $(\vec{n}, \vec{\tau}, \vec{z})$  can then be assumed at this interface point. Consequently, six zeroth order jump conditions can be derived from (6)

$$[E_z] = 0, \quad [E_\tau] = 0, \quad [D_n] = 0, \quad [H_z] = 0, \quad [H_\tau] = 0, \quad [H_n] = 0. \tag{7}$$

As in [25,26], these zeroth order jump conditions will be combined with Maxwell’s equations to derive the first order jump conditions for both electric and magnetic components

$$\left[ \frac{\partial E_z}{\partial n} \right] = 0, \quad \left[ \frac{\partial E_z}{\partial \tau} \right] = 0, \quad \left[ \frac{\partial H_\tau}{\partial \tau} \right] = 0, \quad \left[ \frac{\partial H_n}{\partial \tau} \right] = 0, \quad \left[ \frac{\partial H_n}{\partial n} \right] = 0. \tag{8}$$

We still need one more first order jump condition. Following [27,28], let us suppose that  $[\partial D_z / \partial t] = \psi(t, x, y)$  for some unknown function  $\psi(t, x, y)$ . We then take the jump operation to the first equation of (1) to deduce the last jump condition

$$\psi(t, x, y) = \left[ \frac{\partial D_z}{\partial t} \right] = \left[ \frac{\partial H_y}{\partial x} \right] - \left[ \frac{\partial H_x}{\partial y} \right]. \tag{9}$$

Since the field components are Cartesian ones in (1), we need to transfer the jump conditions (7) and (8) into the Cartesian coordinate [29]. Denote the angle between  $\vec{n}$  and the  $x$ -axis as  $\theta$ . Then the field components in the local coordinate and the Cartesian coordinate are related via simple coordinate transformations, i.e.,  $H_n = \cos \theta H_x + \sin \theta H_y$  and  $H_\tau = -\sin \theta H_x + \cos \theta H_y$ . It is easy to derive the following zeroth order jump conditions

$$[H_x] = 0, \quad [H_y] = 0, \quad [E_z] = 0. \tag{10}$$

The first order jump conditions can be found by using the coordinate transformation for derivative operator

$$\frac{\partial}{\partial n} = \cos \theta \frac{\partial}{\partial x} + \sin \theta \frac{\partial}{\partial y}, \quad \frac{\partial}{\partial \tau} = -\sin \theta \frac{\partial}{\partial x} + \cos \theta \frac{\partial}{\partial y}. \tag{11}$$

For two conditions of  $E_z$  given in (8), the coordinate rotation gives

$$\left[ \frac{\partial E_z}{\partial x} \right] = 0, \quad \left[ \frac{\partial E_z}{\partial y} \right] = 0. \quad (12)$$

In combining these two conditions with the continuity condition of  $E_z$  given in (10), we thus have three Cartesian conditions for  $E_z$ , which can be handled by the previous MIB method [29]. Similarly, we apply the coordinate transformation to other three conditions in (8). This gives rise to

$$\begin{aligned} 0 &= \left[ \frac{\partial H_n}{\partial n} \right] = \cos^2 \theta \left[ \frac{\partial H_x}{\partial x} \right] + \cos \theta \sin \theta \left[ \frac{\partial H_x}{\partial y} \right] + \sin \theta \cos \theta \left[ \frac{\partial H_y}{\partial x} \right] + \sin^2 \theta \left[ \frac{\partial H_y}{\partial y} \right], \\ 0 &= \left[ \frac{\partial H_\tau}{\partial \tau} \right] = \sin^2 \theta \left[ \frac{\partial H_x}{\partial x} \right] - \cos \theta \sin \theta \left[ \frac{\partial H_x}{\partial y} \right] - \sin \theta \cos \theta \left[ \frac{\partial H_y}{\partial x} \right] + \cos^2 \theta \left[ \frac{\partial H_y}{\partial y} \right], \\ 0 &= \left[ \frac{\partial H_n}{\partial \tau} \right] = -\sin \theta \cos \theta \left[ \frac{\partial H_x}{\partial x} \right] + \cos^2 \theta \left[ \frac{\partial H_x}{\partial y} \right] - \sin^2 \theta \left[ \frac{\partial H_y}{\partial x} \right] + \cos \theta \sin \theta \left[ \frac{\partial H_y}{\partial y} \right]. \end{aligned}$$

By solving these three equations together with (9), we attain the desired Cartesian jump conditions for  $H_x$  and  $H_y$

$$\left[ \frac{\partial H_y}{\partial x} \right] = \psi(t, x, y) \cos^2 \theta, \quad \left[ \frac{\partial H_x}{\partial y} \right] = -\psi(t, x, y) \sin^2 \theta. \quad (13)$$

We note that for a straight interface case with  $\theta = 0$ , (13) can be further simplified.

The nonhomogeneous terms in (13) are unknown and time dependent. Thus, in the computation, one needs to estimate  $\psi(t, x, y)$  at interface points at each time step. In order to calculate such values, we apply jump operators to (4) to derive the following interface auxiliary differential equation (IADE)

$$\frac{\partial \psi(t, x, y)}{\partial t} + \gamma \psi(t, x, y) = \epsilon_0 \left[ \epsilon_\infty \frac{\partial \dot{E}_z}{\partial t} \right] + \epsilon_0 \gamma [\epsilon_\infty \dot{E}_z] + \epsilon_0 [\omega_1^2 E_z]. \quad (14)$$

We note that the proposed IADE (14) for the Drude material is a second order differential equation, while that of the Debye material is a first order one [27–29]. A new difficulty here is how to approximate the highest order term, i.e.,  $\frac{\partial \dot{E}_z}{\partial t}$ , in the MIBTD algorithm.

We first simplify the right hand side term of (14). Thanks to the continuity of  $E_z$  and its time derivatives across  $\Gamma$ , this term can be rewritten as

$$\begin{aligned} g(t, x, y) &:= \epsilon_0 \left[ \epsilon_\infty \frac{\partial \dot{E}_z}{\partial t} \right] + \epsilon_0 \gamma [\epsilon_\infty \dot{E}_z] + \epsilon_0 [\omega_1^2 E_z] \\ &= \epsilon_0 (\epsilon_\infty^+ - \epsilon_\infty^-) \left( \frac{\partial \dot{E}_z}{\partial t} \right)^+ + \epsilon_0 \gamma (\epsilon_\infty^+ - \epsilon_\infty^-) \dot{E}_z^+ + \epsilon_0 ((\omega_1^+)^2 - (\omega_1^-)^2) E_z^+. \end{aligned} \quad (15)$$

Here we have introduced a new notation for  $g(t, x, y)$ , so that (14) can be denoted as

$$\frac{\partial \psi(t, x, y)}{\partial t} + \gamma \psi(t, x, y) = g(t, x, y). \quad (16)$$

Eq. (15) suggests that  $g(t, x, y)$  can be evaluated based on  $E_z$ ,  $\dot{E}_z$  and  $\frac{\partial \dot{E}_z}{\partial t}$  values from the positive side. The details of such an evaluation procedure will be offered in next section. Here, the positive side is chosen in the present study, because the incident wave propagates from the positive domain  $\Omega^+$ . Alternatively, we can also use  $\dot{E}_z^-$  and  $E_z^-$  to estimate  $g(t, x, y)$ , if necessary. With the approximated  $g(t, x, y)$  value, (16) enables us to calculate  $\psi(t, x, y)$  at various interface points at each time step. We thus are able to track the unsteady nonhomogeneous values of (13), so that all Cartesian jump conditions become deterministic. Finally, we note that it is because we need to know values of  $\dot{E}_z$  in (15) that we propose the new Maxwell–Drude system (5), instead of employing the classical TM equations (1) and (4).

#### 4. MIB time domain algorithm

The present Drude interface problem consists of the proposed Maxwell–Drude system (5) and the IADE (16). The boundary conditions on the interface  $\Gamma$  are given by (10), (12) and (13), while some proper boundary conditions will be assumed on the  $\partial\Omega$ .

In the present study, the classical fourth order Runge–Kutta method [30] is employed to integrate both the Maxwell–Drude system (5) and the IADE (16) with a fixed time increment  $\Delta t$ . Because each fractional step of the four stage Runge–Kutta time integration can be regarded as an Euler integration [30], we will illustrate the proposed MIBTD algorithm by considering a simple forward Euler time-stepping from  $t_k$  to  $t_{k+1}$  for simplicity.

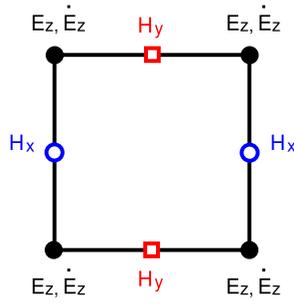


Fig. 1. The staggered grid system.

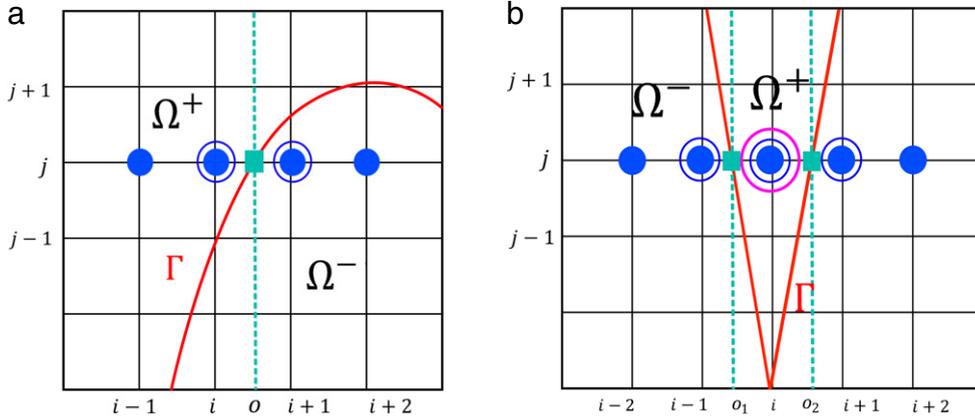


Fig. 2. Illustration of the MIB grid partition. (a) A smooth interface; (b) A sharp corner. Filled circles: grid nodes; Open circles: fictitious nodes; Squares: interface nodes.

A uniform staggered grid shown in Fig. 1 is used in our computations. Without the loss of generality, we assume that the interface  $\Gamma$  will not pass any grid point in the present work. Since the geometry is time invariant, we can determine all necessary interface points  $(x_o, y_o)$ , which are the intersection points between  $\Gamma$  and  $x$  grid lines or  $y$  grid lines, at the beginning of computation. In the following discussion, without the loss of generality, we assume that at the time  $t_k$ , we know field values of  $E_z$ ,  $H_x$ , and  $H_y$  on staggered nodes and interface values of  $\psi(t_k, x_o, y_o)$  on interface points.

In the MIBTD algorithm, away from the interface, the standard finite difference discretization of (5) is conducted, while for staggered nodes near  $\Gamma$ , the MIB interface treatment [27–29] will be carried out to correct finite difference approximations via imposing the jump conditions. In particular, the MIB interface treatment consists of two stages [25,26]. First, one determines the necessary fictitious values of  $E_z$ ,  $H_x$ , and  $H_y$  satisfying the jump conditions (10), (12) and (13). Second, using fictitious values, the standard finite difference approximations can be applied across the interface  $\Gamma$ . We illustrate the details for the MIB treatment of  $E_z$  along  $x$  direction. Other MIB treatments including  $E_z$  along  $y$  direction,  $H_y$  along  $x$  direction, and  $H_x$  along  $y$  direction, can be similarly conducted, because the jump conditions involved in these treatments are decoupled.

We first discuss the MIB scheme for a smooth interface which is a  $C^1$  continuous curve. Consider a case shown in Fig. 2 (a), where for the interface point  $(x_o, y_o)$ , we have  $y_o = y_j$  and  $x_i < x_o < x_{i+1}$ . We denote  $E_{i,j} := E_z(t_k, x_i, y_j)$ . The  $x$  derivative approximations at  $(x_i, y_j)$  and  $(x_{i+1}, y_j)$  will be corrected to be

$$\frac{\partial^2 E_z}{\partial x^2} \Big|_{i,j} \approx \frac{E_{i-1,j} - 2E_{i,j} + \tilde{E}_{i+1,j}}{\Delta x^2}, \quad \frac{\partial^2 E_z}{\partial x^2} \Big|_{i+1,j} \approx \frac{\tilde{E}_{i,j} - 2E_{i+1,j} + E_{i+2,j}}{\Delta x^2}, \tag{17}$$

where  $\tilde{E}_{i,j}$  and  $\tilde{E}_{i+1,j}$  are fictitious values. We note that in the MIB treatment [25,26], in order to approximate function or its derivative on one side of interface, we never use directly the function values of the other side. As illustrated in (17), the fictitious values on the opposite side will be employed.

Two unknown fictitious values  $\tilde{E}_{i,j}$  and  $\tilde{E}_{i+1,j}$  can be determined by discretizing two jump conditions (10) and (12) in the same manner as (17), i.e., never referring to function values across the interface

$$\begin{aligned} w_{0,1}^+ E_{i-1,j} + w_{0,2}^+ E_{i,j} + w_{0,3}^+ \tilde{E}_{i+1,j} &= w_{0,1}^- \tilde{E}_{i,j} + w_{0,2}^- E_{i+1,j} + w_{0,3}^- E_{i+2,j} + [E], \\ w_{1,1}^+ E_{i-1,j} + w_{1,2}^+ E_{i,j} + w_{1,3}^+ \tilde{E}_{i+1,j} &= w_{1,1}^- \tilde{E}_{i,j} + w_{1,2}^- E_{i+1,j} + w_{1,3}^- E_{i+2,j} + \left[ \frac{\partial E}{\partial x} \right], \end{aligned} \tag{18}$$

where  $w_{i,j}^-$  and  $w_{i,j}^+$  for  $l = 0, 1$  and  $J = 1, 2, 3$  are one-sided finite difference weights, respectively, for negative and positive subdomains. Here the subscript  $l$  represents interpolation ( $l = 0$ ) and the first derivative approximation ( $l = 1$ ), and  $J$  is for grid index. The zeroth and first order jumps of  $E_z$  at the interface node  $(x_0, y_j)$ , i.e.,  $[E]$  and  $[\frac{\partial E}{\partial x}]$ , are vanishing for the present case. However, we choose to present the MIB scheme in such a general way so that it can be readily applied to more complicated occasion with nonzero jump values. After the discretization, (18) actually represents two algebraic equations. By solving (18), one can determine  $\tilde{E}_{i,j}$  and  $\tilde{E}_{i+1,j}$  as linear combinations of  $E_{i,j-1}, E_{i,j}, E_{i,j+1}, E_{i,j+2}, [E]$ , and  $[\frac{\partial E}{\partial x}]$ . By substituting such combinations into (17) to eliminate the fictitious values, the finite difference approximations to the double  $x$  derivatives are now second order accurate, irrespective of the presence of the interface.

When one grid line intersects the interface  $\Gamma$  near a sharp corner, the interface could be cut twice within a short distance, see Fig. 2 (b). The classical MIB method [25,26] is not able to handle such a non-smooth interface case. Following the improved MIB scheme for elliptic interface problems [31], we develop a double layer MIB scheme for solving the present Maxwell–Drude interface problem. Denote two interface points as  $(x_{o_1}, y_j)$  and  $(x_{o_2}, y_j)$  with  $x_{i-1} < x_{o_1} < x_i < x_{o_2} < x_{i+1}$ . The  $x$  derivative approximations at the three nearby nodes are modified to be

$$\begin{aligned} \frac{\partial^2 E_z}{\partial x^2} \Big|_{i-1,j} &\approx \frac{E_{i-2,j} - 2E_{i-1,j} + \tilde{E}_{i,j}}{\Delta x^2}, & \frac{\partial^2 E_z}{\partial x^2} \Big|_{i,j} &\approx \frac{\tilde{E}_{i-1,j} - 2E_{i,j} + \tilde{E}_{i+2,j}}{\Delta x^2} \\ \frac{\partial^2 E_z}{\partial x^2} \Big|_{i+1,j} &\approx \frac{\hat{E}_{i,j} - 2E_{i+1,j} + E_{i+2,j}}{\Delta x^2} \end{aligned} \quad (19)$$

where  $\tilde{E}_{i,j}$  and  $\hat{E}_{i,j}$  are two fictitious values located at the same grid node  $(x_i, y_j)$  and they could take different values. The four fictitious values involved in (19) can be determined at the same time by enforcing jump conditions (10) and (12) of  $E_z$  at two consecutive interface points  $(x_{o_1}, y_j)$  and  $(x_{o_2}, y_j)$ . These four jump conditions will be discretized as in (18). In particular, the negative and positive terms at  $(x_{o_1}, y_j)$  are approximated based on  $(E_{i-2,j}, E_{i-1,j}, \tilde{E}_{i,j})$  and  $(\tilde{E}_{i-1,j}, E_{i,j}, \tilde{E}_{i+1,j})$ , respectively. Similarly, the positive and negative terms at  $(x_{o_2}, y_j)$  are approximated based on  $(\tilde{E}_{i-1,j}, E_{i,j}, \tilde{E}_{i+1,j})$  and  $(\hat{E}_{i,j}, E_{i+1,j}, E_{i+2,j})$ , respectively. The details of the discretization are omitted here to save the space.

After correcting all finite difference approximations by the MIB scheme, the spatial discretization of the Maxwell–Drude system (5) is ensured to be second order of accuracy at time  $t_k$ . One can then integrate (5) to next time level  $t_{k+1}$  to generate new values of  $E_z, H_x$ , and  $H_y$ . We will update the IADE (16) from  $t_k$  to  $t_{k+1}$  as well. For this purpose, we shall calculate  $g(t_k, x_0, y_0)$  at the interface point  $(x_0, y_0)$  first. To overcome the aforementioned difficulty due to the involvement  $\frac{\partial \tilde{E}_z}{\partial t}$  in the IADE, we propose to reformulate the expression of  $g(t, x, y)$  by substituting the last equation of (5) into (15)

$$\begin{aligned} g(t_k, x_0, y_0) &= \left[ \frac{\epsilon_0 \epsilon_\infty^-}{\epsilon_\infty^+} (\omega_1^+)^2 - \epsilon_0 (\omega_1^-)^2 \right] E_z^+ \\ &\quad + \frac{\epsilon_\infty^+ - \epsilon_\infty^-}{\mu_0 \epsilon_\infty^+} \left( \frac{\partial^2 E_z^+}{\partial x^2} + \frac{\partial^2 E_z^+}{\partial y^2} \right) + \frac{\gamma (\epsilon_\infty^+ - \epsilon_\infty^-)}{\epsilon_\infty^+} \left( \frac{\partial H_y^+}{\partial x} - \frac{\partial H_x^+}{\partial y} \right), \end{aligned} \quad (20)$$

so that the explicit calculation of  $\frac{\partial \tilde{E}_z}{\partial t}$  is not required. Nevertheless, one needs to evaluate various derivative and function values at the off-grid point  $(x_0, y_0)$  in (20). Fortunately, the MIB scheme is a robust finite difference method, which allows these values being approximated by using on-grid function values exclusively from the positive side. For example,  $(\partial^2 E_z^+ / \partial x^2)(t_k, x_0, y_0)$  will be approximated via one-sided interpolation based on several nearby derivative values of  $(\partial^2 E_z^+ / \partial x^2)(t_k, x_i, y_j)$  for some  $i$  and  $j$ . Then  $(\partial^2 E_z^+ / \partial x^2)(t_k, x_i, y_j)$  can be approximated by finite differences using  $E_z^+$  values within  $\Omega^+$ . Other terms in (20) can be similarly treated. With  $g(t_k, x_0, y_0)$  values, we integrate the IADE (16) in time to generate interface values  $\psi(t_{k+1}, x_0, y_0)$ , which will be used in the MIB scheme for next time level  $t_{k+1}$ . This completes one time step of the proposed MIBTD algorithm.

It is noted that for a straight interface case, the jump conditions can be iteratively enforced [25] so that high order MIBTD algorithms can be easily constructed. It is also noted that most computations of the MIB jump condition enforcement need to be conducted only once at the beginning, because the geometric domain, grid, and spatial discretization of the jump conditions are all time invariant. The calculated representation coefficients that express the fictitious values in terms of field values will be re-used in all time steps.

## 5. Numerical experiments

In this section, we investigate the accuracy and stability of the proposed MIBTD algorithm for solving straight and curved interface problems. Different types of curved interfaces with constant curvature, fluctuating curvature and sharp corner are studied. In all studies, a uniform mesh is employed for the  $E_z$  component, in which the number of grid points in both  $x$  and  $y$  directions is denoted, respectively, as  $N_x$  and  $N_y$ . The staggered grids for  $H_x$  and  $H_y$  can then be correspondingly formed. By setting the initial wave solutions as trivially zero at  $t = 0$ , numerical time integration is carried out for  $N_t$  steps, until a stopping time  $T$ . Here the time increment is given as  $\Delta t = T/N_t$ . Numerical errors in the  $L_\infty$  norm will be reported in all examples.

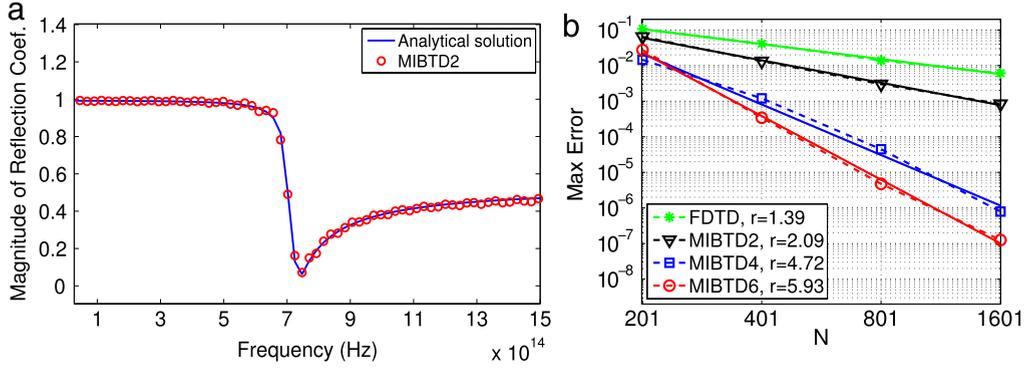


Fig. 3. Numerical results of Example 1. (a) The reflection coefficient at a straight air–gold interface; (b) Numerical convergence test.

### 5.1. Example 1: straight Drude interface

We first consider a straight interface problem, for which the MIBTD algorithm can be easily generalized to higher order. In this example, the computational domain is considered as  $\Omega = [0 \mu\text{m}, 3 \mu\text{m}]^2$  with the interface  $\Gamma$  located at  $x = 0.75\pi \mu\text{m}$ . A dispersive gold medium is placed on the right subdomain  $\Omega^-$  with the Drude parameters  $\epsilon_\infty = 9.84$ ,  $\omega_1 = 9.096 \text{ eV}$ , and  $\gamma = 0.072 \text{ eV}$  [3,32]. An incident pulse of the form  $E_z(x, y, t) = \exp(-(x + 1.5 - c_0 t)^2 / (2\alpha^2))$  is enforced as the boundary condition at the left boundary  $x = 0 \mu\text{m}$ . Here  $\alpha = 0.075 \mu\text{m}$  and  $c_0$  is the speed of light in air. We note that both the media and the incident wave are invariant along the  $y$  direction, so that the electromagnetic wave solutions to this straight interface problem will be constant along the  $y$  direction. Therefore, periodic conditions will be assumed on the top boundary  $y = 3 \mu\text{m}$  and the bottom boundary  $y = 0 \mu\text{m}$  for simplicity. As in [28], our simulation will stop before the transmitted wave hits the right boundary  $x = 3 \mu\text{m}$ . Under this assumption, we can impose the perfectly electric conducting (PEC) boundary condition [33] at  $x = 3 \mu\text{m}$  for simplicity.

Higher order MIBTD schemes are examined for this example. Following [28], away from the interface, the standard  $(2M)$ th order central finite difference approximation will be carried out in both  $x$  and  $y$  directions. Here  $M$  is the half stencil bandwidth. Near the interface, the zeroth and first order jump conditions will be discretized by using  $2L$  function values and 2 fictitious values at the first step, where one-sided approximations involving  $L$  grid nodes in one side are used to ensure the sufficiently high accuracy. One can then iteratively enforce these two jump conditions, until a total of  $2M$  fictitious values are solved [25,26]. Then, the  $(2M)$ th order central finite difference can be conducted across the interface. Three MIBTD schemes are considered, i.e., the MIBTD2 with  $(M, L) = (1, 2)$ , the MIBTD4 with  $(M, L) = (2, 4)$ , and the MIBTD6 with  $(M, L) = (3, 6)$ . Theoretically, these three schemes attain second, fourth, and sixth order of accuracy, respectively. The approximation in the  $y$  direction is not critical in this example, because the wave solutions are trivially constant along this direction. Thus, we will take  $N_y = 10$  in all tests and consider different  $N_x$  values. For a comparison, a FDTD method is considered in this paper which discretizes the Maxwell–Drude system (5) by the central finite difference in space and the classical fourth order Runge–Kutta method in time. Computationally, this FDTD method is simply attained by switching off the MIB fictitious value representation in the MIBTD2 scheme.

We first qualitatively benchmark the MIBTD algorithm by studying the wideband reflection coefficient of an air–gold interface. The MIBTD2 scheme is employed with  $N_x = 1601$ . Two time series are recorded for the  $E_z$  values at two fixed observation points in the vacuum with a time increment  $\Delta t = 0.0027 \text{ fs}$  and a total of 10,000 time steps. Through proper processing, one can generate the incident and reflected pulses based on the recorded time series. The numerical reflection coefficient can then be calculated as the ratio of the discrete Fourier transformation of the reflected and incident pulses. Fig. 3(a) compares the magnitude of the reflection coefficient as a function of frequency to the analytical one. It can be seen that the MIBTD result and the exact one are in good agreement over a wide range of frequencies. This validates the proposed MIBTD algorithm.

We next investigate the stability of the proposed MIBTD method. For the stability study, a stopping time  $T = 13.23 \text{ fs}$  is used, which is short enough such that both reflected and transmitted pulses have not reached the horizontal boundaries. The Courant–Friedrichs–Lewy (CFL) stability condition for solving Maxwell’s equations in 2D is known to be [25]

$$\Delta t \leq C \frac{1}{v_{\max} \sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}}}. \quad (21)$$

Here, we have that  $\Delta x$  is different from  $\Delta y$  and the maximum velocity  $v_{\max}$  equals to the speed of light in air, i.e.,  $v_{\max} = c_0 = 3 \times 10^8 \text{ m/s}$ . The CFL constant  $C$  is determined by both the spatial and temporal discretizations. We note that because the present FDTD approach is based on the fourth order Runge–Kutta method, instead of the Leapfrog scheme, the present CFL constant  $C$  differs from the one of the standard FDTD scheme by a factor of  $\sqrt{2}$  [25].

**Table 1**Critical CFL number  $C$  for Example 1. Here  $N_t^*$  denotes the critical  $N_t$  value.

$N_x$	FDTD		MIBTD2		MIBTD4		MIBTD6	
	$N_t^*$	$C$	$N_t^*$	$C$	$N_t^*$	$C$	$N_t^*$	$C$
101	91	1.4587	91	1.4587	105	1.2642	112	1.1960
201	182	1.4543	182	1.4543	257	1.0299	300	0.8823
401	365	1.4492	365	1.4492	483	1.0951	552	0.9583
801	739	1.4313	739	1.4313	877	1.2060	930	1.1361

**Table 2**

Numerical convergence test of Example 1.

$N_x$	FDTD		MIBTD2		MIBTD4		MIBTD6	
	Error	Order	Error	Order	Error	Order	Error	Order
201	1.07E−1		6.44E−2		1.45E−2		2.74E−2	
401	4.12E−2	1.38	1.33E−2	2.28	1.20E−3	3.59	3.42E−4	6.32
801	1.39E−2	1.57	3.00E−3	2.15	4.48E−5	4.74	4.75E−6	6.17
1601	6.10E−3	1.19	8.36E−4	1.84	7.90E−7	5.83	1.26E−7	5.24

**Table 3**CPU time in seconds of the FDTD method ( $T_F$ ), the MIBTD2 method ( $T_{M_2}$ ), the MIBTD4 method ( $T_{M_4}$ ) and the MIBTD6 method ( $T_{M_6}$ ) for Example 1.

$N_x$	FDTD	MIBTD2		MIBTD4		MIBTD6	
	$T_F$	$T_{M_2}$	$T_{M_2}/T_F$	$T_{M_4}$	$T_{M_4}/T_F$	$T_{M_6}$	$T_{M_6}/T_F$
101	0.028	0.028	1.00	0.068	2.43	0.136	4.86
201	0.104	0.168	1.62	0.368	3.54	0.468	4.50
401	0.410	0.576	1.40	0.780	1.90	1.720	4.20
801	1.620	1.952	1.20	2.412	1.49	3.320	2.05
1601	6.460	6.550	1.01	7.284	1.13	9.200	1.42

In order to examine the stability of the proposed algorithms, we numerically detect the critical  $N_t$  value for each scheme. Here, the critical  $N_t$  value is defined in the sense that a even smaller  $N_t$  value will render the computation to be unstable. In particular, for a given  $N_x$  value, we examine all possible  $N_t$  values. For a particular  $N_t$  value, say  $N_t^*$ , such that the final  $L_\infty$  error is less than a tolerance, e.g. 1, the computation is said to be stable. If the final error of  $N_t^* - 1$  is greater than or equal to the tolerance, the empirical critical  $N_t$  value is taken as  $N_t^*$ . By considering several  $N_x$  values, the critical  $N_t$  values and the corresponding CFL constants for both the FDTD and MIBTD methods are reported in Table 1. It can be seen that as  $\Delta x$  becomes smaller and smaller, the CFL number  $C$  of the present FDTD method goes to  $C = \sqrt{2}$ . In fact,  $C = \sqrt{2}$  is the analytical CFL number of the present FDTD method [25,29]. This validates the present numerical procedure for the stability analysis. For the proposed MIBTD algorithms, no analytical result is available. We thus numerically explore the CFL number  $C$  for different MIBTD schemes. It can be seen from Table 1 that the MIBTD2 scheme attains the same CFL number as the FDTD. In other words, the MIB interface treatment in the MIBTD2 has negligible impact on the stability for this straight interface problem. For the MIBTD4 and MIBTD6,  $C$  decreases as  $M$  is larger as expected, because  $C$  is known to be inversely proportional to the absolute sum of the underlying finite difference weights [25].

We then quantitatively examine the accuracy of the proposed higher order MIBTD methods. We conduct a numerical convergence analysis by considering different spatial size  $N_x$  while  $N_y$  is always fixed to be 10. To ensure the stability, we take the CFL constant to be  $C = 0.7$  in all methods. Because there is no analytical solution available for this Drude interface problem, we benchmark our numerical results by considering a numerical reference solution. This reference solution is generated by considering the most accurate MIBTD6 scheme with a very dense grid  $N_x = 12,801$  and a sufficiently small  $\Delta t$ . Based on this reference solution, the  $L_\infty$  errors in the  $E_z$  component are reported in Table 2. The numerical orders calculated based on two successive meshes are also reported. The orders of three MIBTD schemes are indeed about two, four, and six, respectively. Without the interface treatment, the FDTD scheme is obviously degraded to the first order of accuracy. For a visual comparison, the  $L_\infty$  errors in Table 2 are also plotted in Fig. 3 (b). A linear least-squares fitting [26] is conducted in the log–log scale for the purpose of determining the overall convergence rate  $r$  of the scheme. Here the fitted convergence lines are shown as solid lines in Fig. 3 (b) with  $r$  being the slopes. The averaged orders of both MIBTD and FDTD methods are close to the theoretical estimates.

We finally investigate the computational speed of the MIBTD methods. The CPU time in seconds of both the FDTD and MIBTD methods are listed in Table 3. All computations here are conducted on an SGI Xeon E5-4640 CPU core operating at 2.4 GHz and 8 GB of memory. As in the numerical convergence tests, the CFL constant is set to be  $C = 0.7$ . It is seen in Table 3 that the CPU time of the MIBTD methods becomes larger when the order is higher. The ratio of the CPU time of the MIBTD over that of the FDTD is also listed in Table 3. This ratio becomes much smaller when  $N_x$  is larger. This is because the generation of fictitious value representation needs to be carried out only once in the MIB scheme. Thus, the additional CPU

**Table 4**  
Critical CFL number  $C$  for Example 2. Here  $N_t^*$  denotes the critical  $N_t$  value.

$N$	FDTD		MIBTD	
	$N_t^*$	$C$	$N_t^*$	$C$
101	102	1.5914	112	1.4493
201	219	1.5481	236	1.3756
401	450	1.4428	508	1.2781
801	907	1.4317	1052	1.2344

**Table 5**  
Numerical convergence test of Example 2.

$N$	FDTD		MIBTD	
	Error	Order	Error	Order
101	4.12e−2		1.57e−2	
201	1.85e−2	1.16	4.60e−3	1.77
401	7.30e−3	1.34	1.00e−3	2.20
801	3.90e−3	0.90	2.28e−4	2.13

time caused by the MIB treatment becomes less significant comparing with the overall CPU time used in the Runge–Kutta time integrations, when  $N_x$  is larger. Therefore, the ratio of the MIBTD2 over the FDTD eventually goes to 1.01 when  $N_x$  is very large, while that between the MIBTD6 and FDTD is about 1.42. Even though the MIBTD methods consume more CPU time than the FDTD, the MIBTD methods are actually more cost-efficient. When one aims to achieve certain high precision, the MIBTD method based on a coarser grid is actually much faster.

## 5.2. Example 2: circular Drude interface

We next study a circular Drude interface problem with  $\Gamma$  defined as

$$\Gamma : \sqrt{(x - 0.7\pi)^2 + (y - 1.75)^2} = 0.5, \quad (22)$$

where the unit is in micrometers. The entire domain  $\Omega = [0 \mu\text{m}, 3.5 \mu\text{m}]^2$  consists of two sub-domains. The subdomain  $\Omega^-$  inside the cylinder is assigned to be a dispersive gold medium with the same Drude parameters as in Example 1. The subdomain  $\Omega^+$  is assumed to be air. An incident pulse of the form  $E_z(x, y, t) = \exp(-(x + 1.75 - c_0 t)^2 / (2\alpha^2))$  is imposed as the boundary condition at the left boundary  $x = 0 \mu\text{m}$  with  $\alpha = 0.25 \mu\text{m}$ . The computation will be terminated before the propagating wave hits the right boundary  $x = 3.5 \mu\text{m}$  and the scattered wave touches the top boundary  $y = 3.5 \mu\text{m}$  and bottom boundary  $y = 0 \mu\text{m}$ . For these reasons, the stopping time is chosen as  $T = 13.40$  fs. Moreover, for simplicity, the top and bottom boundary conditions are enforced according to the incident pulses, while the PEC boundary condition [33] is imposed on the right boundary.

For curved Drude interface problems, the generalization of the MIBTD algorithm to higher order is not straightforward. Thus, only the second order MIBTD2 scheme will be considered in the curved interface problems, and this scheme is simply referred to as the MIBTD scheme. Moreover, we will set  $N = N_x = N_y$  in the all curved interface problems for simplicity. The aforementioned FDTD method is also employed for a comparison.

The stability of the MIBTD and FDTD schemes is examined first. Because of  $\Delta x = \Delta y$  in the present case, the CFL stability condition (21) is reduced to be

$$\Delta t \leq C \frac{\Delta x}{c_0 \sqrt{2}}. \quad (23)$$

Again, the CFL constant  $C$  is determined by both the spatial and temporal discretizations. Similarly, we will numerically estimate  $C$  by detecting the critical  $N_t$  value for a given  $N_x$  value. The results are reported in Table 4. Similar to the straight interface case, the CFL number of the FDTD scheme goes to  $C = \sqrt{2}$  as  $\Delta x$  approaches zero, while that of the MIBTD method shall be a number within the interval  $C \in [1.23, 1.28]$ . In the following computations, we will take  $C = 0.7$  to ensure the stability.

We next examine the numerical orders of the MIBTD and FDTD methods. A numerical reference solution is generated by the MIBTD scheme with a dense mesh  $N = 1601$ . Maximum errors calculated with respect to this reference solution are reported in Table 5 and also plotted in Fig. 4(a). The contour plot of one MIBTD solution is shown in Fig. 4(b). A sharp change is clearly seen across the interface  $\Gamma$  in the  $H_y$  component. In general, we know that  $E_z$  is  $C^1$  continuous across  $\Gamma$ , while the regularity of  $H_y$  and  $H_x$  degrades to  $C^0$ . Without proper interface treatment, the staircase error is severe in the FDTD algorithm. Thus, the least-squares fitted convergence rate of the FDTD method is found to be 1.15 for the present example. However, the MIBTD method still successfully retains the second order convergence after enforcing jump conditions for  $H_x$ ,  $H_y$  and  $E_z$  across the interface.

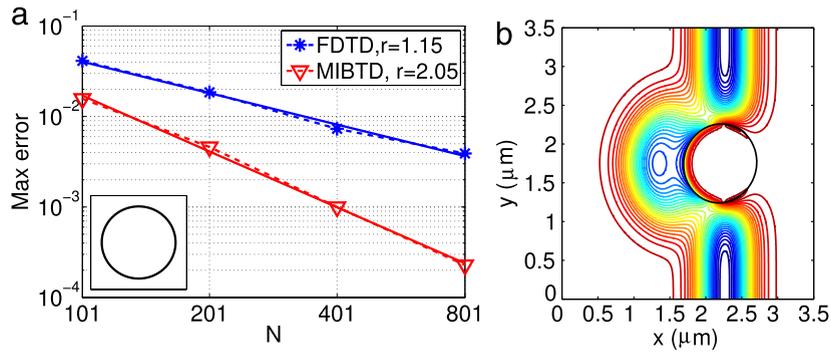


Fig. 4. Numerical results of Example 2. (a) Numerical convergence test; (b) Contour plot of  $H_y$  generated by the MIBTD method with  $N = 401$ .

Table 6

CPU time in seconds of the FDTD method ( $T_F$ ) and the MIBTD method ( $T_M$ ) for Example 2.

$N$	$T_F$	$T_M$	$T_M/T_F$
101	0.37	0.43	1.16
201	3.00	3.40	1.13
401	26.99	34.93	1.29
801	247.80	334.42	1.35

Our final test for the present example is again exploring the speed of MIBTD method. The CPU time in seconds consumed by the FDTD and MIBTD methods is reported in Table 6. Because in the curved interface case, different MIB treatments have to be conducted at different irregular points, the computational overhead of the MIB scheme becomes larger than that of the straight interface case. Thus, the CPU time ratio of the MIBTD method over the FDTD method in the present study is about 1.35 when  $N$  is large. Nevertheless, we note that the MIBTD method is still more cost-efficient than the FDTD method.

### 5.3. Example 3: complex Drude interfaces

We next examine the robustness of the MIBTD algorithm in dealing with complex interfaces with non-constant curvatures. A family of interfaces parameterized by the polar angle  $\theta$  [26,29] are studied for this purpose

$$\Gamma : \sqrt{(x - 0.7\pi)^2 + (y - 1.75)^2} = 0.5 + b \sin(m\theta), \quad \theta \in [0, 2\pi]. \quad (24)$$

The unit of this equation as well as  $b$  is in micrometers. Here the parameter  $m$  determines the number of leaves of the core region and  $b$  controls the magnitude of the curvature. As in [29], we investigate four independent cases: Case 1,  $(m, b) = (2, 1/4)$ ; Case 2,  $(m, b) = (3, 1/9)$ ; Case 3,  $(m, b) = (4, 1/9)$ ; Case 4,  $(m, b) = (5, 1/11)$ . Apart from the difference in the definition of the interface, the other problem settings including the domain dimensions, Drude parameters, incident wave, boundary conditions and stopping time are the same as in Example 2.

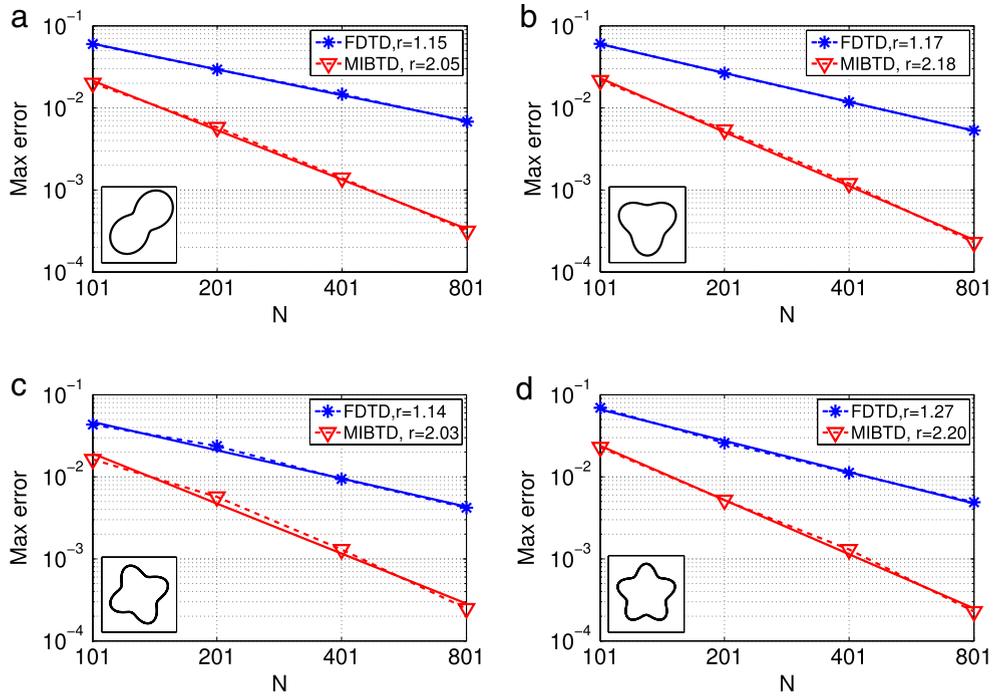
The stability and CPU behaviors of the MIBTD and FDTD methods are similar to those of Example 2, and thus are not presented here to save space. We only focus on the accuracy of the MIBTD method in solving various non-constant curvature interfaces. In order to benchmark the numerical convergences, for each case, a reference solution is generated by the MIBTD algorithm with  $N = 1601$  and a sufficiently small  $\Delta t$ . We again set the CFL number to be  $C = 0.7$ . The MIBTD and FDTD solutions based on different  $N$  values are compared with the reference solution on mutually sampled grid nodes. The results on maximum errors in  $E_z$  are reported in Table 7. It can be seen that, the order convergences of the FDTD method always stay at around one, while those of the MIBTD algorithm are often about two throughout the table. Similar to the previous examples, the errors are also depicted in Fig. 5, augmented by the linear least-squares fitting to analyze the numerical orders. Due to the loss of regularity in wave solutions near the interface, the convergence rates of the FDTD method are found to be  $r = 1.15$ ,  $r = 1.17$ ,  $r = 1.14$  and  $r = 1.27$ , respectively, for the Case 1, 2, 3, and 4. After the MIB interface treatment, the accuracies of the MIBTD method are all about second order, even though the shape of the interface becomes more and more complicated. The present study demonstrates the robustness of the proposed MIBTD algorithm. Lastly, the contour plots of the  $H_y$  components in four cases are shown in Fig. 6. A very complicated interaction between the electromagnetic waves and the Drude interface can be clearly seen, especially when the interface shape becomes more irregular.

### 5.4. Example 4: non-smooth Drude interface

In the previous examples, the interfaces are always  $C^1$  continuous, so that the regular MIB interface treatment is sufficient to treat them. In our last study, we consider a Lipschitz continuous but not  $C^1$  continuous interface to validate the proposed

**Table 7**  
Numerical convergence tests of Example 3.

N	FDTD		MIBTD		FDTD		MIBTD	
	Error	Order	Error	Order	Error	Order	Error	Order
	Case 1, (m, b) = (2, 1/4)				Case 2, (m, b) = (3, 1/9)			
101	6.00E-2		2.00E-2		6.04E-2		2.16E-2	
201	2.94E-2	1.03	5.80E-3	1.79	2.64E-2	1.19	5.40E-3	2.00
401	1.47E-2	1.00	1.40E-3	2.05	1.18E-2	1.16	1.20E-3	2.17
801	6.80E-3	1.11	3.15E-4	2.15	5.30E-3	1.15	2.30E-4	2.38
	Case 3, (m, b) = (4, 1/9)				Case 4, (m, b) = (5, 1/11)			
101	4.33E-2		1.63E-2		6.94E-2		2.29E-2	
201	2.37E-2	0.87	5.70E-3	1.52	2.59E-2	1.42	5.20E-3	2.14
401	9.40E-3	1.33	1.30E-3	2.13	1.12E-2	1.21	1.30E-3	2.00
801	4.20E-3	1.16	2.48E-4	2.39	4.90E-3	1.19	2.28E-4	2.51



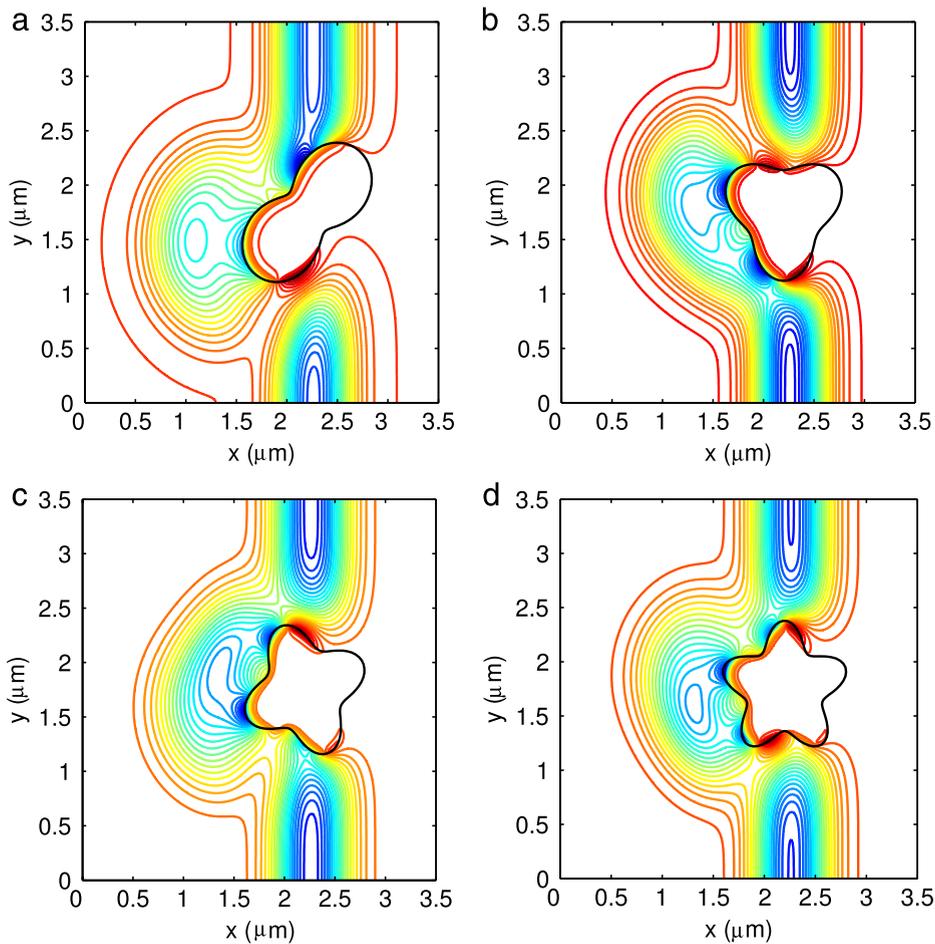
**Fig. 5.** Numerical convergence tests for Example 3. (a) Case 1, (m, b) = (2, 1/4); (b) Case 2, (m, b) = (3, 1/9); (c) Case 3, (m, b) = (4, 1/9); (d) Case 4, (m, b) = (5, 1/11).

MIB non-smooth interface treatment. For this purpose, a 3/4 circle interface parameterized by the polar angle  $\theta$  is defined as follows

$$\Gamma : (x, y) = \begin{cases} (0.7\pi + 0.5 \cos \theta, 1.75 + 0.5 \sin \theta), & \text{if } 0 \leq \theta \leq \frac{3\pi}{4} \text{ or } \frac{5\pi}{4} \leq \theta \leq 2\pi, \\ \left(0.7\pi - \sqrt{2} \left(1 - \frac{\theta}{\pi}\right), 1.75 + \sqrt{2} \left(1 - \frac{\theta}{\pi}\right)\right), & \text{if } \frac{3\pi}{4} \leq \theta \leq \pi, \\ \left(0.7\pi + \sqrt{2} \left(1 - \frac{\theta}{\pi}\right), 1.75 + \sqrt{2} \left(1 - \frac{\theta}{\pi}\right)\right), & \text{if } \pi \leq \theta \leq \frac{5\pi}{4}. \end{cases} \quad (25)$$

The unit of this equation is in micrometers. The other problem settings are the same as in the previous example. Due to appearances of two sharp corners on the left part of the interface  $\Gamma$ , the double layer MIB algorithm is indispensable. The CFL stability condition and execution time of the MIBTD algorithm remain essentially the same as in the previous curved interfaces. For this reason, those results are not presented here.

It is of interests to investigate the impact of the sharp corners on the accuracy of the MIBTD method. As usual, a reference solution is generated by the MIBTD scheme with  $N = 1601$  and a sufficiently small  $\Delta t$ . The CFL number is set to be  $C = 0.7$  for all computations. The  $L_\infty$  errors involved in the MIBTD and FDTD  $E_z$  solutions are reported in Table 8 and plotted in



**Fig. 6.** Contour plots of  $H_y$  generated by the MIBTD method with  $N = 401$  in Example 3. (a) Case 1,  $(m, b) = (2, 1/4)$ ; (b) Case 2,  $(m, b) = (3, 1/9)$ ; (c) Case 3,  $(m, b) = (4, 1/9)$ ; (d) Case 4,  $(m, b) = (5, 1/11)$ .

**Table 8**  
Numerical convergence test of Example 4.

$N$	FDTD		MIBTD	
	Error	Order	Error	Order
201	2.10E-2		6.10E-3	
321	2.23E-2	-0.13	5.70E-3	0.14
401	1.02E-2	3.52	2.70E-3	3.36
801	4.90E-3	1.06	2.90E-4	3.23

Fig. 7(a). These results reveal that the orders of both methods are highly nonuniform, due to the presence of the corner points. Indeed, the contour plot given in Fig. 7(b) shows that the wave solution displays almost singular behavior near two corner points. The graphical least-squares analysis conducted in Fig. 7(a) gives a better estimate to the numerical orders than the tabulated results shown in Table 8. In particular, the numerical order of the FDTD and MIBTD is found to be 1.14 and 2.32, respectively. Without the double layer MIB scheme, the MIBTD algorithm will not be able to restore the fully second order of accuracy in solving only Lipschitz continuous interfaces.

## 6. Conclusion

In this paper, we propose a new Maxwell–Drude formulation for studying 2D transverse magnetic problems involving Drude materials with curved and non-smooth dispersive interfaces. This hybrid system couples the wave equation for the electric component with Maxwell’s equations for the magnetic components, and allows us to track the transient change in the regularity of electromagnetic fields across the Drude interfaces. The resulting time dependent jump conditions are enforced through regular and double layer interface treatments in the proposed MIBTD algorithm. For straight interfaces,

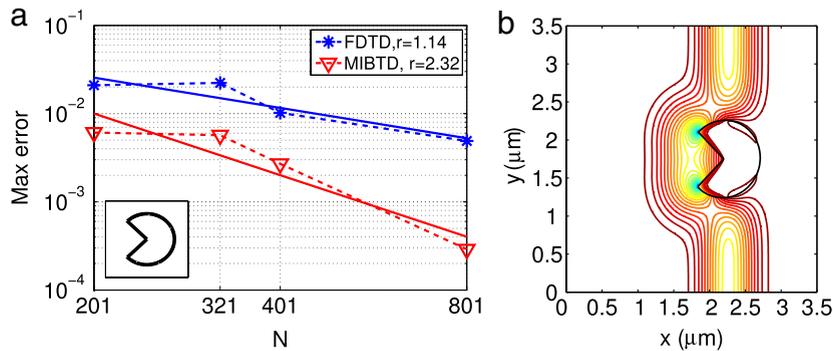


Fig. 7. Numerical results of Example 4. (a) Numerical convergence test; (b) Contour plot of  $H_y$ , generated by the MIBTD method with  $N = 401$ .

the MIBTD method can be generalized to higher order, while for both curved interfaces and interfaces with sharp corners, the MIBTD method can fully recover the second order of accuracy. Thus, the proposed MIBTD method is more accurate and cost-efficient than the classical FDTD method.

The generalization of the proposed MIBTD algorithm for solving the transverse electric system is currently under our investigation. Such a generalization is highly nontrivial and requires a new Maxwell formulation, because the field components could be discontinuous in such an interface problem.

## Acknowledgments

This work was supported in part by National Science Foundation (NSF) grants DMS-1016579 and DMS-1318898, and the University of Alabama Research Stimulation Program (RSP) award.

## References

- [1] J.M. Bourgeois, G.S. Smith, A fully three-dimensional simulation of a ground-penetrating radar: FDTD theory compared with experiment, *IEEE Trans. Geosci. Remote Sens.* 34 (1996) 36–44.
- [2] X. Li, E.J. Bond, B.D. Van Veen, S.C. Hagness, An overview of ultra-wideband microwave imaging via space-time beamforming for early-stage breast-cancer detection, *IEEE Antennas Propag. Mag.* 47 (2005) 19–34.
- [3] K. Kolwas, A. Derkachova, M. Shopa, Size characteristics of surface plasmons and their manifestation in scattering properties of metal particles, *J. Quant. Spectrosc. Radiat. Transfer* 110 (2009) 1490–1501.
- [4] R.M. Joseph, S.C. Hagness, A. Taflove, Direct time integration of Maxwell's equations in linear dispersive media with absorption for scattering and propagation of femtosecond electromagnetic pulses, *Opt. Lett.* 16 (1991) 1412–1414.
- [5] O.P. Gandhi, B.-Q. Gao, J.-Y. Chen, A frequency-dependent finite-difference time-domain formulation for general dispersive media, *IEEE Trans. Microw. Theory Tech.* 41 (1993) 658–665.
- [6] M. Okoniewski, E. Okoniewska, Drude dispersion in ADE FDTD revisited, *Electronics Lett.* 42 (2006) 503–504.
- [7] J. Shibayama, A. Nomura, R. Ando, J. Yamauchi, H. Nakano, A frequency-dependent LOD-FDTD method and its application to the analysis of plasmonic waveguide devices, *IEEE J. Quantum Electron.* 46 (2010) 40–49.
- [8] V.A. Bokil, N.L. Gibson, Analysis of spatial high-order finite difference methods for Maxwell's equations in dispersive media, *IMA J. Numer. Anal.* (2011) 1–31. <http://dx.doi.org/10.1093/imanum/drr001>.
- [9] D. Jiao, J.M. Jin, Time-domain finite-element modeling of dispersive media, *IEEE Microw. Wireless Comp. Lett.* 11 (2001) 220–222.
- [10] J. Li, Error analysis of finite element methods for 3-D Maxwell's equations in dispersive media, *J. Comput. Appl. Math.* 188 (2006) 107–120.
- [11] H.T. Banks, V.A. Bokil, N.L. Gibson, Analysis of stability and dispersion in a finite element method for Debye and Lorentz dispersive media, *Numer. Methods Partial Differential Equations* 25 (2009) 885–917.
- [12] J. Li, Y. Huang, Y. Lin, Developing finite element methods for Maxwell's equations in a Cole–Cole dispersive medium, *SIAM J. Sci. Comput.* 33 (2011) 3153–3174.
- [13] Y. Huang, J. Li, W. Yang, Modeling backward wave propagation in metamaterials by the finite element time-domain method, *SIAM J. Sci. Comput.* 35 (2013) B248–B274.
- [14] T. Lu, P. Zhang, W. Cai, Discontinuous Galerkin methods for dispersive and lossy Maxwell's equations and PML boundary conditions, *J. Comput. Phys.* 200 (2004) 549–580.
- [15] Y. Huang, J. Li, Interior penalty discontinuous Galerkin method for Maxwell's equations in cold plasma, *J. Sci. Comput.* 41 (2009) 321–340.
- [16] K. Stannigel, M. König, J. Niegemann, K. Busch, Discontinuous Galerkin time-domain computations of metallic nanostructures, *Opt. Express* 17 (2009) 14934–14947.
- [17] B. Wang, Z. Xie, Z. Zhang, Error analysis of a discontinuous Galerkin method for Maxwell equations in dispersive media, *J. Comput. Phys.* 229 (2010) 8552–8563.
- [18] S.D. Gedney, J.C. Young, T.C. Kramer, J.A. Roden, A discontinuous Galerkin finite element time-domain method modeling of dispersive media, *IEEE Trans. Antennas and Propagation* 60 (2012) 1969–1977.
- [19] A. Taflove, S.C. Hagness, *Computational Electrodynamics: The Finite-Time Domain-Method*, third ed., Artech House, Norwood, MA, 2005.
- [20] Y. Zhao, Y. Hao, Finite difference time domain study of guided modes in nano-plasmonic waveguides, *IEEE Trans. Antennas and Propagation* 55 (2007) 3070–3077.
- [21] A. Deinega, I. Valuev, Subpixel smoothing for conductive and dispersive media in the finite-difference time-domain method, *Opt. Lett.* 32 (2007) 3429–3431.
- [22] A. Mohammadi, T. Jalali, M. Agio, Dispersive contour-path algorithm for the two dimensional finite difference time-domain method, *Opt. Express* 16 (2008) 7397–7406.
- [23] J.H. Park, J.C. Strikwerda, The domain decomposition method for Maxwell's equations in time domain simulations with dispersive metallic media, *SIAM J. Sci. Comput.* 32 (2010) 684–702.

- [24] J.S. Hesthaven, High-order accurate methods in time-domain computational electromagnetics. A review, *Adv. Imag. Electron Phys.* 127 (2003) 59–123.
- [25] S. Zhao, G.W. Wei, High order FDTD methods via derivative matching for Maxwell's equations with material interfaces, *J. Comput. Phys.* 200 (2004) 60–103.
- [26] S. Zhao, High order matched interface and boundary methods for the Helmholtz equation in media with arbitrarily curved interfaces, *J. Comput. Phys.* 229 (2010) 3155–3170.
- [27] S. Zhao, High order FDTD methods for transverse electromagnetic systems in dispersive inhomogeneous media, *Opt. Lett.* 36 (2011) 3245–3247.
- [28] D.D. Nguyen, S. Zhao, High order FDTD methods for transverse magnetic modes with dispersive interfaces, *Appl. Math. Comput.* 226 (2014) 699–707.
- [29] D.D. Nguyen, S. Zhao, Time-domain matched interface and boundary (MIB) modeling of Debye dispersive media with curved interfaces, *J. Comput. Phys.* 278 (2014) 298–325.
- [30] S. Zhao, G.W. Wei, A unified discontinuous Galerkin framework for time integration, *Math. Methods Appl. Sci.* 37 (2014) 1042–1071.
- [31] S. Yu, Y. Zhou, G.W. Wei, Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces, *J. Comput. Phys.* 224 (2007) 729–756.
- [32] A. Cala' Lesina, A. Vaccari, A. Bozzoli, A novel RC-FDTD algorithm for the Drude dispersion analysis, *Prog. Electromagn. Res.* 24 (2012) 251–264.
- [33] S. Zhao, A fourth order finite difference method for waveguides with curved perfectly conducting boundaries, *Comput. Methods Appl. Mech. Engrg.* 199 (2010) 2655–2662.