Electromagnetic Wave Propagation Simulation by Meshless Time-Domain Method Embedding Modified RPIM-Based Shape Functions

Taku Itoh\textsuperscript{1}, Soichiro Ikuno\textsuperscript{1}, and Susumu Nakata\textsuperscript{2}

\textsuperscript{1}School of Computer Science, Tokyo University of Technology, Japan
\textsuperscript{2}Department of Media Technology, Ritsumeikan University, Japan

Abstract

To improve the performance of the Meshless Time-Domain Method (MTDM) for electromagnetic wave propagation simulations, a strategy for embedding the Modified Radial Point Interpolation Method (MRPIM)-based shape functions to the MTDM has been proposed. Numerical experiments show that the electric field determined by the MTDM embedding the MRPIM-based shape functions is smoothly distributed throughout a waveguide. In addition, the computational time of the MRPIM-based MTDM is always less than that of the conventional MTDM.

Keywords – Electromagnetic propagation, FDTD methods, Meshless methods, Maxwell equations

1 Introduction

The Finite-Difference Time-Domain (FDTD) method has generally been applied to electromagnetic wave propagation simulations. However, to apply the FDTD method to electromagnetic wave propagation simulations, the numerical domain has to be divided into rectangle meshes, and it is difficult that an arbitrary-shaped domain is accurately represented by rectangle meshes.

On the other hand, the Meshless Time-Domain Method (MTDM) \cite{1} does not require finite elements or meshes of a geometrical structure, i.e., the node alignment of the MTDM is more flexible than that of the FDTD method. Thus, the MTDM can handle complex-shaped domains more easily than that of the FDTD method. However, the computational time of the MTDM is larger than that of the FDTD method. This is because, in the MTDM, shape functions based on the Radial Point Interpolation method (RPIM) \cite{2} are usually employed, i.e., linear systems whose size depends on the number of nodes have to be solved to determine the shape functions before starting the simulations.

Recently, the Modified RPIM (MRPIM) \cite{3} has been proposed. In the MRPIM, the linear systems for determining the shape functions are constructed by using the local nodes that is contained in the locally defined subdomain. Hence, if the MRPIM-based shape functions are embedded to the MTDM, the performance of the MTDM may be improved.

The purpose of the present study is to propose a strategy for embedding the MRPIM-based shape functions to the MTDM.

2 Meshless Time-Domain Method

To simulate electromagnetic wave propagation, we consider Maxwell equations in case of the 2D TM mode described as

\begin{align}
\frac{\partial E_y}{\partial t} &= -\sigma E_z + \frac{\partial H_z}{\partial x}, \quad (1) \\
\frac{\partial H_z}{\partial t} &= -\frac{\partial E_y}{\partial x}, \quad (2) \\
\frac{\partial E_z}{\partial t} &= -\frac{\partial H_y}{\partial y}. \quad (3)
\end{align}

where \( E \) denotes the \( z \) component of the electric field, and \( H_y \) and \( H_z \) denote the \( x \) and \( y \) components of the magnetic field, respectively. In addition, \( \varepsilon, \sigma, \) and \( \mu \) denote the permittivity, the electrical conductivity and the magnetic permeability, respectively.

To discretize (1), (2) and (3) by MTDM, nodes \( x_i \) (\( i = 1,2,\ldots,N_x \)) for \( E_z \) and \( x_i \) (\( j = 1,2,\ldots,N_y \)) for \( H_y \) and \( H_z \) are first aligned in a domain, where \( N_x \) denotes the number of nodes for \( E_z \) and \( N_y \) denotes the number of nodes for \( H_y \) and \( H_z \). In MTDM, the leap-frog method is employed to discretize the time-domain. In addition, the space domain is discretized by using the shape functions of the Meshless Methods. The discretized forms of (1), (2) and (3) are as follows:

\begin{align}
E_y^{n+1} &= E_y^n - \frac{\Delta t}{\varepsilon} \sum_{j=1}^{N_y} \left( H_z^{n+1} \frac{\partial \phi_j}{\partial x} - H_z^n \frac{\partial \phi_j}{\partial x} \right), \quad (4) \\
H_z^{n+1} &= H_z^n - \frac{\mu \Delta t}{\varepsilon} \sum_{j=1}^{N_y} E_y^n \frac{\partial \phi_j}{\partial y}, \quad (5) \\
H_y^{n+1} &= H_y^n + \frac{\mu \Delta t}{\varepsilon} \sum_{j=1}^{N_y} E_y^n \frac{\partial \phi_j}{\partial y} \quad (6)
\end{align}

where \( n \) is the time step, \( E_y^n \equiv E_y(x_i^j), H_z^{n+1} \equiv H_z(x_i^j), \) and \( H_y^{n+1} \equiv H_y^j(x_i) \). In addition, \( \phi_i^j(x) \) and \( \phi_i^j(x) \) denote the shape functions corresponding to \( x_i^j (j = 1,2,\ldots,N_y) \) and \( x_i^j (j = 1,2,\ldots,N_y) \), respectively. Note that, in the MTDM, the shape functions have to be chosen so that \( \phi_i^j(x) \) and \( \phi_i^j(x) \) have the Kronecker delta function property as follows \cite{2}:

\begin{align}
\phi_i^j(x_k) &= \begin{cases} 1 & \text{for } i = j, \\
0 & \text{for } i \neq j, \end{cases} \\
\phi_i^j(x_k) &= \begin{cases} 1 & \text{for } i = j, \\
0 & \text{for } i \neq j. \end{cases} \quad (7)
\end{align}

3 Shape Functions for MTDM

To satisfy (7), the RPIM-based shape functions are usually employed \cite{1}. However, since the RPIM-based shape functions are not trivial, linear systems have to be solved to determine the shape functions. Note that the size of linear systems depends on the number of nodes. Hence, in the MTDM, the computational cost for determining the shape functions is required before starting simulations. For this reason, the computational cost of the MTDM is much larger than that of the FDTD.

On the other hand, in the MRPIM, rectangular subdomains \( \Omega(k = 1,2,\ldots,M) \) are first made as shown in Fig. 1(a) \cite{3}. In addition, shape functions are determined in each subdomain. To
Fig. 1. (a) Rectangular subdomains $\Omega_k (k = 1, 2, \ldots, M)$ for determining shape functions of the MRPIM. The $k$th support domain $\Omega'_k$ is defined by enlarging $\Omega_k$. (b) Schematic view of node alignment of $x^e_i$ and $x^h_i$ in a line-shaped waveguide. Here, $x^e_i$ and $x^h_i$ are represented as red quadrilaterals and blue triangles, respectively. To embed the MRPIM-based shape functions to the MTDM, subdomains $\Omega_k (k = 1, 2, \ldots, M)$ are defined together with $x^e_i$ and $x^h_i$ such as this figure. In addition, $\Omega'_k$ is defined as $\Omega_k$ with two lower and upper subdomains.

this end, in the $k$th subdomain $\Omega_k$, the enlarged subdomain $\Omega'_k$ is considered as the $k$th support domain (see Fig. 1(a)). Namely, the linear systems for determining shape functions in $\Omega'_k$ are constructed by using the nodes contained in $\Omega'_k$. By solving the above linear systems, although shape functions are determined on nodes in $\Omega'_k$, shape functions corresponding to nodes in $\Omega_k$ are only employed for simulation. Since the linear systems are constructed in each support domain, the computational cost for determining shape functions in the MRPIM can be saved in comparison with that in the RPIM. Thus, we employ the shape functions based on the MRPIM to improve the MTDM.

4 Strategy for Embedding MRPIM-Based Shape Functions to MTDM

To embed the shape functions based on the MRPIM to the MTDM for the electromagnetic wave propagation simulation in a waveguide, we first assume that the subdomains $\Omega_k (k = 1, 2, \ldots, M)$ are made by dividing along the waveguide. In addition, we also assume that the nodes $x^e_i (i = 1, 2, \ldots, N^e)$ and $x^h_i (j = 1, 2, \ldots, N^h)$ are aligned such as shown in Fig. 1(b). This figure also shows the subdomains $\Omega_k (k = 1, 2, \ldots, M)$.

To generate the linear systems for determining shape functions in $\Omega_k$, we consider the subdomains $\Omega_{k-2}, \Omega_{k-1}, \ldots, \Omega_{k+2}$ as the support domain $\Omega'_k$ (see Fig. 1(b)). Note that if the subdomain number is less than 1 or greater than $M$, the subdomain does not adopt for definition of $\Omega'_k$, e.g., for $\Omega_1$, the subdomains $\Omega_2, \Omega_3, \Omega_4$ and $\Omega_5$ are considered as the support domain $\Omega'_1$.

In $\Omega_k (k = 1, 2, \ldots, M)$, the linear systems for determining $\phi^e_i (x)$ corresponding to $x^e_i$ are generated by using the nodes $x^e_i(j = 1, 2, \ldots, N^e)$ that are contained in $\Omega'_k$. Similarly, in $\Omega_k (k = 1, 2, \ldots, M)$, the linear systems for determining $\phi^h_i (x)$ corresponding to $x^h_i$ are also generated by using the nodes $x^h_i(j = 1, 2, \ldots, N^h)$ that are contained in $\Omega'_k$. Here, $N^e$ and $N^h$ are the numbers of nodes $x^e_i$ and $x^h_i$ in $\Omega'_k$, respectively.

5 Numerical Experiments

In this section, for a 2D electromagnetic wave propagation simulation, numerical experiments were conducted to investigate the performance of the MTDM embedding the MRPIM-based shape functions. To this end, the line-shaped waveguide illustrated in Fig. 2(a) is used for this simulation. In this simula-

tion, we assume that the wave source is a sine wave whose amplitude, frequency and speed are $1.0 \text{ (V/m)}$, $1.0 \times 10^9 \text{ (Hz)}$ and $299792458 \text{ (m/s)}$, respectively.

First, the distribution $E_t$ determined by the MTDM embedding the MRPIM-based shape functions is shown in Fig. 2(b). We see from this figure that $E_t$ is smoothly distributed throughout the waveguide. Hence, we consider that the MRPIM-based shape functions can be embedded to the MTDM.

Next, the dependence of the computational time for determining shape functions on the number $N$ of nodes is shown in Fig. 3. Here, $N = N^e + N^h$. We see from this figure that the computational time for determining shape functions based on the MRPIM is always less than that of the original RPIM. Therefore, we conclude that the performance of the MTDM can be improved by embedding the MRPIM-based shape functions to the MTDM.

Fig. 2. (a) Schematic view of the line-shaped waveguide. (b) Distribution of the electric field $E_t$ for $t = 1000\Delta t$. Here, $\Delta t = 5.0 \times 10^{-9}$. In addition, the value of $E_t$ corresponds to the color distribution on the right-hand side.

Fig. 3. Dependence of the computational time for determining shape functions on the number $N$ of nodes, where $N = N^e + N^h$.

References