

Faster Generation of Shape Functions in Meshless Time Domain Method^{*)}

Yoshiharu OHI, Yoshihisa FUJITA¹⁾, Taku ITOH²⁾, Hiroaki NAKAMURA³⁾ and Soichiro IKUNO²⁾

RIKEN Advanced Institute for Computational Science, Chuo-ku, Kobe, Hyogo 650-0047, Japan

¹⁾*Department of Energy Engineering and Science, Nagoya University, Toki, Gifu 509-5292, Japan*

²⁾*School of Computer Science, Tokyo University of Technology, Hachioji, Tokyo 192-0982, Japan*

³⁾*National Institute for Fusion Science, Toki, Gifu 509-5292, Japan*

(Received 11 December 2013 / Accepted 3 September 2014)

The finite difference time domain method (FDTD) is a robust numerical scheme for time-dependent electromagnetic wave propagation phenomena that uses orthogonal meshes, like staggered meshes, also known as Yee lattices. However, treating complex shaped domains is challenging for the FDTD. Meshless methods, in contrast, do not require meshes for a geometrical structure. The meshless time domain method (MTDM), based on the radial point interpolation method, can be used for numerical simulations in computational electromagnetics. In MTDM, shape functions have to be generated before the time-dependent calculation, and the computational cost involved can be very large. We herein propose a new method for reducing the computational cost of generating shape functions and we confirm the effectiveness of the proposed method by numerical experiments.

© 2014 The Japan Society of Plasma Science and Nuclear Fusion Research

Keywords: finite difference time domain method, meshless time domain method, radial point interpolation method

DOI: 10.1585/pfr.9.3401144

1. Introduction

In the electron cyclotron heating system used for helical plasma heating in the Large Helical Device (LHD), a long corrugated waveguide transmits electrical power generated by a gyrotron system to the LHD. The length of the waveguide is about 100 meters and the waveguide is bent at right angles several times between the gyrotron and the LHD. Miter bends are implemented to reduce transmission loss in the bends. It is not clear what effect the shape of the corrugated waveguide has or what the performance of the miter bends is. It is important that the shape of the waveguide and the miter bends are evaluated by numerical simulation.

Generally, the finite difference time domain method (FDTD) is applied for time-dependent electromagnetic wave propagation simulations. The FDTD provides a direct solution of Maxwell's equations. Orthogonal meshes, like staggered meshes, also known as Yee lattices, are adopted for the standard FDTD [1]. The FDTD has great advantages in terms of discretization and parallelization, as well as other advantages [2]. In order to treat problems in complex shaped domains, a fine mesh or an adaptive mesh is adopted. The computational cost of this approach is high and treating domains with different mesh sizes is complicated. Hence dealing with complex shaped domains is a challenge.

A meshless method, in contrast, does not require finite elements or meshes with a geometrical structure, and allows solutions for arbitrary shaped domains to be easily obtained [3]. Various meshless methods, such as the radial point interpolation method (RPIM), have been developed [4]. The meshless time domain method (MTDM), based on RPIM, can be applied to time-dependent electromagnetic wave propagation simulations [5–7]. In the MTDM, a shape function is generated before the time-dependent calculation. The computational cost of generating shape functions can be very large, and should be reduced as much as possible.

The purpose of the present study is to propose a new method to reduce the computational cost of generating shape functions. The effectiveness of the proposed method will also be verified by numerical experiments.

2. Meshless Time Domain Method

Various meshless methods have been developed, which employ a governing equation that is discretized using a shape function or a partial derivative of the shape function. In the present study, we consider the RPIM meshless method. In the RPIM [4], an approximate function $u^*(\mathbf{x})$ and its partial derivative $\partial u^*(\mathbf{x})$ are expanded using a shape function $\phi(\mathbf{x})$, its partial derivative $\partial \phi(\mathbf{x})$ and a known vector \mathbf{u} as:

author's e-mail: yoshiharu.ohi@riken.jp

^{*)} This article is based on the presentation at the 23rd International Toki Conference (ITC23).

$$\mathbf{u}^*(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x}) \cdot \mathbf{u} = \sum_i \phi_i(\mathbf{x}) u_i, \quad (1)$$

$$\partial \mathbf{u}^*(\mathbf{x}) = \partial \boldsymbol{\phi}(\mathbf{x}) \cdot \mathbf{u} = \sum_i \partial \phi_i(\mathbf{x}) u_i, \quad (2)$$

where \mathbf{x} is the position vector and \cdot denotes an inner product. ∂ means $\partial/\partial x$ or $\partial/\partial y$ in 2D. The shape function and its partial derivative are generated by solving systems of linear equations generated from a polynomial basis function (PBF) and a radial basis function (RBF) [4]. The PBF and RBF are functions of the relative distances between the nodes in the calculation of the shape function. RBFs of various types have been proposed [8, 9].

A shape function based on RPIM satisfies the Kronecker delta function property:

$$\phi_i(\mathbf{x} = \mathbf{x}_j) = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases} \quad (3)$$

where i and j denote node numbers. From this property, the approximate function can be written in a simplified form:

$$\mathbf{u}^*(\mathbf{x}_i) = \boldsymbol{\phi}(\mathbf{x}_i) \cdot \mathbf{u} = u_i. \quad (4)$$

This property is very important for discretization.

In the present study, an electromagnetic wave propagation simulation of a 2D TM-mode is applied for a numerical examination of the method. Maxwell's equations in a vacuum domain are written as:

$$\varepsilon \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}, \quad (5)$$

$$\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y}, \quad (6)$$

$$\mu \frac{\partial H_y}{\partial t} = \frac{\partial E_z}{\partial x}, \quad (7)$$

where E_z denotes the z -component of the electric field, H_x and H_y denote the x - and y -components of the magnetic field, and ε and μ denote the electric permittivity and permeability, respectively. Equations (5)–(7) can be discretized with respect to time by the leapfrog method, and are then written as:

$$\frac{\varepsilon}{\Delta t} (E_z^{n+1} - E_z^n) = \frac{\partial H_y^{n+\frac{1}{2}}}{\partial x} - \frac{\partial H_x^{n+\frac{1}{2}}}{\partial y}, \quad (8)$$

$$\frac{\mu}{\Delta t} (H_x^{n+\frac{1}{2}} - H_x^{n-\frac{1}{2}}) = -\frac{\partial E_z^n}{\partial y}, \quad (9)$$

$$\frac{\mu}{\Delta t} (H_y^{n+\frac{1}{2}} - H_y^{n-\frac{1}{2}}) = \frac{\partial E_z^n}{\partial x}, \quad (10)$$

where n and Δt denote a time step and an interval of the discrete time. The derivation up to this point is the same as for the FDTD. In the present study, equations (8)–(10) can be discretized with respect to space by using the shape function, and are then written as:

$$E_{z,i}^{n+1} = E_{z,i}^n + \frac{\Delta t}{\varepsilon} \left(\sum_j \frac{\partial \phi_{i,j}^{H_y}}{\partial x} H_{y,j}^{n+\frac{1}{2}} - \sum_k \frac{\partial \phi_{i,k}^{H_x}}{\partial y} H_{x,k}^{n+\frac{1}{2}} \right), \quad (11)$$

$$H_{x,i}^{n+\frac{1}{2}} = H_{x,i}^{n-\frac{1}{2}} - \frac{\Delta t}{\mu} \sum_j \frac{\partial \phi_{i,j}^{E_z}}{\partial y} E_{z,j}^n, \quad (12)$$

$$H_{y,i}^{n+\frac{1}{2}} = H_{y,i}^{n-\frac{1}{2}} + \frac{\Delta t}{\mu} \sum_j \frac{\partial \phi_{i,j}^{E_z}}{\partial x} E_{z,j}^n, \quad (13)$$

where k denotes the node number and $\partial \phi^H$ and $\partial \phi^E$ denote the partial derivatives of the shape function for the magnetic and electric fields, respectively. It is important that the term in the left side and the first term in the right side can be written in a simplified form by satisfying the Kronecker delta function property (4). By solving (11)–(13) and updating n , the time-dependent solutions for an arbitrary shaped domain are obtained. This method is known as the MTDM.

3. Faster Generation of Shape Functions

In the MTDM, shape functions have to be generated before the time-dependent calculation can be performed. As described in the previous section, it is necessary to solve a system of linear equations as $A\mathbf{x} = \mathbf{b}$ to generate a shape function, where A is a coefficient matrix, \mathbf{x} is an unknown vector and \mathbf{b} is a known vector. The sizes of the matrices are $n_{\text{infl}} \times n_{\text{infl}}$, where n_{infl} denotes the number of nodes in the domains of influence. The conceptual diagram of the node distribution based on a Yee lattice, which is applied in the FDTD, is shown in Fig. 1. Here R denotes a support radius which determines the domain of influence and W and L denote the width and length of the analysis domain. The computational cost to generate shape functions is $O(n_{\text{infl}}^3 \times N)$ using a direct method to solve systems of linear equations, where N denotes the number of systems in generating the shape functions. If N is very large, the computational cost also becomes very large.

Even if a simulation is performed in a complex shaped domain, it is not usually the case that the entire analysis domain is complicated. Taking as an example the corrugated waveguide introduced above, the interior surface of the waveguide is corrugated and complicated, but most of the interior of the waveguide is a vacuum domain and simple. Therefore, we propose a new method to reduce the computational cost of generating shape functions in a domain, which assumes a regular nodal distribution, as shown in Fig. 1 left. Due to translational symmetry, the relative distances between nodes involved in the calculation of shape functions are the same. Consequently, instead of calculating shape functions for each node, they are calculated only once and are then translated to all other nodes (i.e., they are reused).

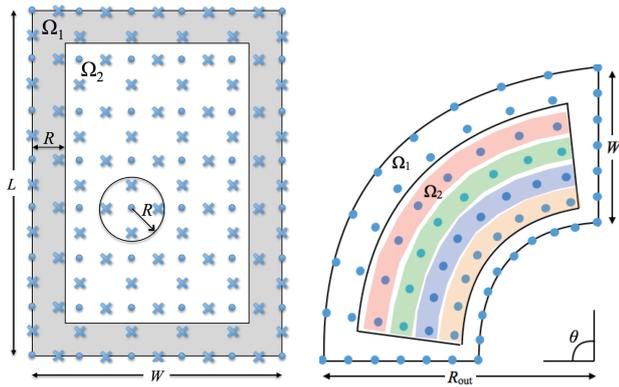


Fig. 1 Node distributions based on a Yee lattice (left) and a circular sector (right).

Let us consider a specific example. A rectangular domain is divided into two domains Ω_1 and Ω_2 , as shown in Fig. 1 left. Ω_1 is the domain within R of the boundary, and Ω_2 is the interior domain. The shape function is generated only once in Ω_2 , and it is reused at the other nodes. Therefore, the computational cost in the interior domain becomes $O(1)$. In applying this method, it is expected that the computational cost of generating the shape functions is cut drastically.

The proposed method is also applicable to domains other than Yee lattices. For example, a circular sector domain having constant curvature is shown in Fig. 1 right. The analysis domain is divided into two domains Ω_1 and Ω_2 , as in the rectangular domain. Nodes on the same colored line in Ω_2 have the same shape functions. Therefore, the shape function is generated only once on a colored line and is reused along the colored line. In the next section, the proposed method and the conventional method are compared by numerical experiments to verify the effectiveness of the proposed method.

4. Numerical Experiments

A rectangular waveguide in which the nodes are arranged based on a Yee lattice and a circular sector waveguide having constant curvature are applied for an electromagnetic wave propagation simulation of a 2D TM-mode. A direct method is adopted to solve the systems of linear equations. The computation environment is shown in Table 1.

First, to investigate the effectiveness of the proposed method for a domain in which the nodes are arranged based on a Yee lattice, it is applied to the rectangular domain shown in Fig. 1 left. The waveguide parameters are shown in Table 2. The distances between the nodes are all 2 mm. W and L are determined so that the areas of both domains are the same ($WL = 1$). The support radius R is 6 mm. The computational time to generate shape functions and the actual speedup rates are given in Table 2. We can see that the computational time for generating shape functions

Table 1 Computational environment.

OS	CentOS 5.9
CPU	Intel Xeon E5-2660 2.20GHz
Memory	64GB
Compiler	gcc 4.1.2

Table 2 Parameters for rectangular waveguide, computational time for generating shape functions and actual speedup rates.

#	W[m]	L[m]	Comp. time [sec]		Actual speedup rates
			Conventional	Proposed	
1-1	0.1	10	192	24.2	7.93
1-2	0.2	5	193	12.1	16.0
1-3	0.5	2	196	6.05	32.4
1-4	1	1	196	5.80	33.8

Table 3 Computational cost, cost reduction rate and theoretical speedup rate for a rectangular waveguide.

#	Computational cost		Cut rates [%]		Theoretical speedup rates
	Conventional	Proposed	Theoretical	Numerical	
1-1	1015152	161691	87.9	84.1	8.24
1-2	1007802	80778	93.7	92.0	16.0
1-3	1003752	39528	97.0	96.1	33.2
1-4	1003002	36714	97.6	96.3	41.4

using the proposed method is significantly lower than for the conventional method. In addition, the speedup rate increases if the area of Ω_2 becomes large. Note that it has been confirmed that solutions using the proposed method and the conventional method corresponds exactly. It is thus shown that the proposed method is very effective for domains in which the nodes are arranged based on a Yee lattice.

Let us estimate the computational cost of the first experiment. It is considered that the computational cost is in proportion to the area of the domain. The areas of the whole domain, Ω_1 and Ω_2 are $S_{\Omega_1 \cup \Omega_2} = WL$, $S_{\Omega_1} = 2R(W + L) - 4R^2$ and $S_{\Omega_2} = (W - 2R)(L - 2R)$, respectively. By using the proposed method, the computational cost in Ω_2 becomes $O(1)$. Therefore, it is estimated that the computational cost is reduced to $S_{\Omega_2}/S_{\Omega_1 \cup \Omega_2}$. These estimations and the results of the first experiment are compared to verify the validity of the proposed method. The computational cost, cost reduction rate and theoretical speedup rate are given in Table 3, where the computational cost means the number of systems used to generate the shape functions, the theoretical cost reduction rate is calculated using $S_{\Omega_2}/S_{\Omega_1 \cup \Omega_2} \times 100[\%]$ and the numerical and theoretical reduction rates are calculated from the rates for the computational cost. These results show that the theoretical and numerical reduction rates largely correspond. The actual speedup rates (Table 2) are slightly lower than the theoret-

Table 4 Width of a circular sector waveguide, computational time to generate shape functions and actual speedup rate.

#	W[m]	Computational time [sec]		Actual speedup rates
		Conventional	Proposed	
2-1	0.1	31.5	4.06	7.76
2-2	0.2	60.4	4.06	14.9
2-3	0.4	109	4.22	25.8
2-4	0.8	171	4.32	39.6

Table 5 Computational cost, cost reduction rate and theoretical speedup rate for a circular sector waveguide.

#	Computational cost		Cut rates [%]		Theoretical speedup rates
	Conventional	Proposed	Theoretical	Numerical	
2-1	159507	25958	87.9	83.7	8.23
2-2	316657	26151	93.8	91.7	16.2
2-3	630957	31510	96.8	95.0	31.4
2-4	1259557	35916	98.3	97.1	59.3

ical speedup rates (Table 3). It is believed that this is because memory copying occurred when the functions were reused. If memory copying could be eliminated, greater speedup is expected, and this will be explored in future work. The above quantitative analysis, demonstrates the validity of the proposed method for a rectangular domain.

In the second experiment, the proposed method is applied to a circular sector domain having constant curvature, as shown in Fig. 1 right. The distances between nodes along the radial direction are 2 mm. The distances between the nodes along an angular direction and the support radius R along the outside of the curve are 2 mm and 6 mm, respectively, and become gradually smaller inside the curve. The radius of the outside of the waveguide R_{out} and the central angle θ are fixed at 1 m and $\pi/2$, respectively. The width of the waveguide W can be varied from 0.1 to 0.8 m. The computational time to generate shape functions and the actual speedup rates are given in Table 4. These results show that the computational time to generate shape functions using the proposed method is significantly lower than when using the conventional method, as in the first experiment. In addition, the effectiveness of the proposed method increases if the area of Ω_2 becomes large. It is thus confirmed that the proposed method is very effective for a circular sector domain having constant curvature.

Let us estimate the computational cost of the second experiment in the same way as for the first experiment. The area of the whole domain is $S_{\Omega_1 \cup \Omega_2} = (WR_{out} - W^2/2)\theta$, and the areas of Ω_2 and Ω_1 are $S_{\Omega_2} \approx (WR_{out} + RW - 2RR_{out} - W^2/2)(\theta - 2R/R_{out})$ and $S_{\Omega_1} \approx S_{\Omega_1 \cup \Omega_2} - S_{\Omega_2}$, respectively. Here, the central angle includes few approximate errors. By using the proposed method, the computational cost in

Ω_2 becomes $O(n_l)$, where n_l is the number of colored lines in Ω_2 . Therefore, if W increases, the computational cost in Ω_2 also increases. These estimations and the results of the second experiment are compared to verify the validity of the proposed method in a circular sector domain. The computational cost, cost reduction rate and theoretical speedup rate are given in Table 5. These results show that the theoretical and numerical reduction rates largely correspond in the second experiment.

The above quantitative analysis demonstrates the validity of the proposed method for a circular sector domain.

5. Conclusion

In the present study, a new method to reduce the computational cost of generating shape functions has been proposed and its effectiveness for a rectangular domain in which the nodes are arranged based on a Yee lattice and for a circular sector domain having constant curvature have been confirmed by numerical experiments using the MTDM. The conclusions obtained in the present study are summarized as follows.

- The computational cost of generating shape functions decreases significantly by using the proposed method.
- The effectiveness of the proposed method increases if the area of the interior domain is large.
- The proposed method can be applied to a Yee lattice or a circular sector domain.
- The actual computational cost of the proposed method largely corresponds to the theoretical estimations.

Future works will examine automating the search for nodes that can reuse the shape function and applying the proposed method to 3D simulations and real problems such as a corrugated waveguide.

Acknowledgement

We wish to acknowledge the great help received from Prof. Takashi Kako at the University of Electro-Communications.

- [1] K.S. Yee, IEEE Trans. Antennas Propag. **14**, 3, 302 (1966).
- [2] T. Hanawa *et al.*, IEEE Trans. Magn. **43**, 1545 (2007).
- [3] T. Belytshko *et al.*, Int. J. Numer. Methods Eng. **37**, 2, 229 (1994).
- [4] J.G. Wang *et al.*, Int. J. Numer. Methods Eng. **54**, 11, 1623 (2002).
- [5] T. Kaufmann *et al.*, Microwave Symposium Digest, 2008 IEEE MTT-S International (2008) pp.61-64.
- [6] T. Kaufmann *et al.*, IEEE Trans. Microw. Theory Tech. **58**, 12, 3399 (2010).
- [7] S. Ikuno *et al.*, IEEE Trans. Magn. **49**, 5, 1613 (2013).
- [8] M. Powell, The theory of radial basis function approximation in 1990 (Oxford, 1992) pp.105-203.
- [9] G.E. Fasshauer, Proceedings of Chamonix (Vanderbilt University Press: Nashville, 1996) pp.1-8.