

PROGRESS REVIEW

Recent techniques on sound field simulation

To cite this article: Takao Tsuchiya 2022 *Jpn. J. Appl. Phys.* **61** SG0801

View the [article online](#) for updates and enhancements.

You may also like

- [Corrigendum: Detection of nosemosis in European honeybees \(*Apis mellifera*\) on honeybees farm at Kanchanaburi, Thailand \(2019 IOP Conf. Ser.: Mater Sci Eng. 639 012048\)](#)
Samrit Maksong, Tanawat Yemor and Surasuk Yanmanee
- [\(Invited\) Characterization of Interface Defects by the Charge Pumping Technique](#)
Toshiaki Tsuchiya
- [Modified Surface Activated Bonding Using Si Intermediate Layer for Bonding and Debonding of Glass Substrates](#)
Kai Takeuchi, Masahisa Fujino and Tadatomu Suga



Recent techniques on sound field simulation

Takao Tsuchiya*

Department of Information Systems Design, Doshisha University, Kyotanabe, Kyoto 610-0321, Japan

*E-mail: ttsuchiy@mail.doshisha.ac.jp

Received January 5, 2022; revised January 25, 2022; accepted February 10, 2022; published online June 10, 2022

This is a tutorial paper on the basics and applications of the finite-difference time-domain (FDTD) method. Two types of discretization of the linear governing equations, the scalar-type FDTD method and the vector-type one, are first discussed. Then the basic concept of the compact explicit-FDTD (CE-FDTD) method is described. By considering the relationship between the cutoff frequency and the computer resources, it is shown that the interpolated wide band scheme requires the least computer resources among the derivative schemes of the CE-FDTD method. The discretization of the arbitrary shaped sound field by voxels and its boundary conditions, and the implementation of the density variation are also described. The sound field rendering and its real time renderer “Silicon concert hall” are introduced. © 2022 The Japan Society of Applied Physics

1. Introduction

Various numerical methods such as the finite-difference method,¹⁾ the finite element method,²⁾ and the boundary element method³⁾ have been proposed for the sound field analysis. In recent years, the finite-difference time-domain (FDTD) method⁴⁻¹⁹⁾ has been widely used because of its simple algorithm and easy programming.

This is a tutorial paper focusing on the FDTD method. The explanation begins with the basic handling for the analysis of the linear sound wave propagation by the standard FDTD method.⁴⁾ We first describe two types of discretization of the governing equations: a scalar-type FDTD method and a vector-type FDTD method. Then it is shown that the scalar-type FDTD method is advantageous for the sound field analysis from the viewpoint of the computer resources.

Next, the basic concept of the compact explicit-FDTD (CE-FDTD) method is described. The CE-FDTD method is a high-precision version of the FDTD method. It was first proposed by Kowalczyk in the two-dimensional field¹⁵⁾ and was soon extended to the three-dimensional field.¹⁶⁾ We first implemented the CE-FDTD method on a graphics processing unit (GPU) cluster system¹⁷⁾ and evaluated its performance in the large scale sound field analysis. By considering the relationship between the cutoff frequency and the computer resources, we also show that the interpolated wide band (IWB) scheme requires the least computer resources among the derivative schemes of the CE-FDTD method. The discretization of the arbitrary shaped sound field by voxels and its boundary conditions,¹⁸⁾ and the implementation of the density variation¹⁹⁾ are also described in this paper.

We have been implementing various applications of the FDTD simulation. Sound field rendering^{17,20)} is the most important application. It is a technology that the sound pressure waveform at the listening position is numerically calculated by the wave based 3D sound field analysis such as the FDTD method, and then the calculated waveform is auralized with the 3D audio. We have implemented various rendering techniques such as the multi-channel rendering and the binaural rendering.²¹⁾ As applications other than the sound field rendering, the CE-FDTD method has been applied to the analysis of the Mach cutoff noise during the supersonic flight²²⁾ and of the moving sources and receivers to simulate bat echolocation.^{14,23)} In this paper, the sound

field rendering and its real time renderer “Silicon concert hall” are introduced.

2. Theory

2.1. Governing equations

For analysis of the linear sound wave propagation without absorption, the following continuity equation and the equation of motion are used as the governing equations.

$$\frac{\partial p}{\partial t} + \rho c_0^2 \nabla \cdot \mathbf{u} = 0, \quad (1)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho} \nabla p = 0, \quad (2)$$

where p is the sound pressure, \mathbf{u} is the particle velocity vector, ρ is the density, and c_0 is the sound speed. Strictly speaking, Eq. (1) must be expressed in terms of the density, but it is expressed in terms of the sound pressure in this paper since the linear sound wave propagation is assumed. In this paper, the set of Eqs. (1) and (2) is called a vector-type governing equations for convenience.

On the other hand, the wave equation for the sound pressure p , which is obtained by eliminating the particle velocity vector from Eqs. (1) and (2) can be also used as the governing equation as

$$\frac{1}{c_0^2} \frac{\partial^2 p}{\partial t^2} = \nabla^2 p. \quad (3)$$

This equation is called a scalar-type governing equation in this paper. So, which governing equation should be used for the sound field analysis? The answer depends on the problem to be solved. In the most sound field analysis, the wave Eq. (3) is sufficient for the governing equation since only the sound pressure is required. Equations (1) and (2) should only be used for the problems that the particle velocity must be kept in the whole analysis domain, such as the analysis of the sound intensity distribution.

2.2. Discretization by FDTD method

The FDTD method is based on a simple and clear algorithm in which the governing equations are directly discretized based on the central finite-difference. However, it is not well known that there is an obvious difference between the discretization of the vector- and scalar-type governing

equations in the computer resources such as the memory usage and the calculation time. Here, we consider a two-dimensional sound field for the sake of simplicity. The difference between the vector- and scalar-type discretizations is described for the case that only the sound pressure is required.

2.2.1. Vector-type FDTD method. First, we consider the discretization of the vector-type governing equation, called the vector-type FDTD method. Since the vector-type FDTD method was first proposed in the analysis of the electromagnetic field⁴⁾ and was applied as it is to the sound field analysis,^{24,25)} it generally refers to the standard FDTD method in the field of acoustics.

We consider the discretization of the vector-type governing equation on the collocated grid as shown in Fig. 1(a), where Δ is the grid interval. It is assumed in the subsequent discretization that all grid interval in the x -, and y -directions (and z -direction in the 3D case) is the same as Δ . In the case of the discretization of the first derivative on the collocated grid, the difference interval becomes 2Δ in space and $2\Delta t$ in time using the central-difference because the sound pressure and the particle velocity are defined at the same grid point. This increased grid interval causes the reduction of the numerical accuracy. In order to avoid the degradation in accuracy, a staggered grid is generally used in the vector-type FDTD method as shown in Fig. 1(b). Discretizing Eqs. (1) and (2) on the staggered grid gives the following equations.

$$p_{i,j}^{n+1} = p_{i,j}^n - \chi(\bar{u}_{x_{i+\frac{1}{2},j}}^{n+\frac{1}{2}} - \bar{u}_{x_{i-\frac{1}{2},j}}^{n+\frac{1}{2}} + \bar{u}_{y_{i,j+\frac{1}{2}}}^{n+\frac{1}{2}} - \bar{u}_{y_{i,j-\frac{1}{2}}}^{n+\frac{1}{2}}), \quad (4)$$

$$\bar{u}_{x_{i+\frac{1}{2},j}}^{n+\frac{1}{2}} = \bar{u}_{x_{i+\frac{1}{2},j}}^{n-\frac{1}{2}} - \chi(p_{i+1,j}^n - p_{i,j}^n), \quad (5)$$

$$\bar{u}_{y_{i,j+\frac{1}{2}}}^{n+\frac{1}{2}} = \bar{u}_{y_{i,j+\frac{1}{2}}}^{n-\frac{1}{2}} - \chi(p_{i,j+1}^n - p_{i,j}^n), \quad (6)$$

where $p_{i,j}^n$, $u_{i,j}^n$ represent the sound pressure and the particle velocity on the grid point $(x, y) = (i\Delta, j\Delta)$ at the time $t = n\Delta t$, respectively, $\bar{u} = \rho c_0 u$ is the normalized particle velocity, and $\chi = c_0 \Delta t / \Delta$ is the Courant number.¹⁾ The difference intervals can be kept at Δ and Δt respectively by using the staggered grid, and the accuracy degradation can be avoided.

2.2.2. Scalar-type FDTD method. Next, we consider the discretization of the scalar-type governing equation, called the scalar-type FDTD method. The scalar-type governing

equation is discretized on the collocated grid as

$$p_{i,j}^{n+1} = 2p_{i,j}^n - p_{i,j}^{n-1} + \chi^2(p_{i+1,j}^n + p_{i-1,j}^n + p_{i,j+1}^n + p_{i,j}^n) - 4\chi^2 p_{i,j}^n. \quad (7)$$

This discretization is sometimes called the wave equation FDTD (WE-FDTD) method.^{26,27)}

2.2.3. Comparison between vector-type and scalar-type FDTD methods. So which discretization is better for the sound field analysis, the vector-type or the scalar-type? To answer this question, we first consider the numerical accuracy. Equation (7) can be obtained eliminating the particle velocity from Eq. (4) using Eqs. (5) and (6). So, it is found that both discretizations have the same accuracy. In the vector-type FDTD method, the particle velocity acts only as an intermediate variable in the computer program.

Next, we consider the computer memory required for the FDTD analysis. Suppose we need N memories for the entire grid points for a certain physical quantity. The scalar-type FDTD method requires N memories for the sound pressure p^n , and N for p^{n-1} , so $2N$ memories are totally required regardless the number of dimensions. Here, the memory for p^{n+1} is not required since p^{n+1} can overwrite p^{n-1} . On the other hand, the vector-type FDTD method requires N memories for the sound pressure and dN memories for the particle velocity, where d is the number of dimensions, so $(d + 1)N$ memories are totally required. The memory required is tabulated in Table I. In the three-dimensional analysis, the vector-type FDTD method requires twice as many memories as the scalar-type one. The memory usage is also related to the calculation time because most of the calculation time is spent on data transfer. This is especially noticeable when using the high-speed computing devices such as graphics processing units (GPUs). It is shown that the scalar-type FDTD method is superior to the vector-type one because it requires less computer resources with the same numerical accuracy.

2.3. CE-FDTD method

The scalar-type FDTD method has another advantage that the numerical accuracy can be controlled by using the multidirectional stencil in the case of two- or three-dimensional field analysis. The compact explicit-FDTD (CE-FDTD) method is one of methods based on the multidirectional stencil.¹⁵⁻¹⁷⁾ In this section, we will explain the basic concept of the CE-FDTD method in a two-dimensional field, then extend it to a three-dimensional field.

2.3.1. 2D CE-FDTD method. In the standard scalar-type FDTD method that is also known as the standard leapfrog (SLF) scheme, the second-order difference is evaluated by the grid points along the axial directions as shown in Fig. 2(a). It is also possible to evaluate along the diagonal directions as shown in Fig. 2(b), which is called the rotated

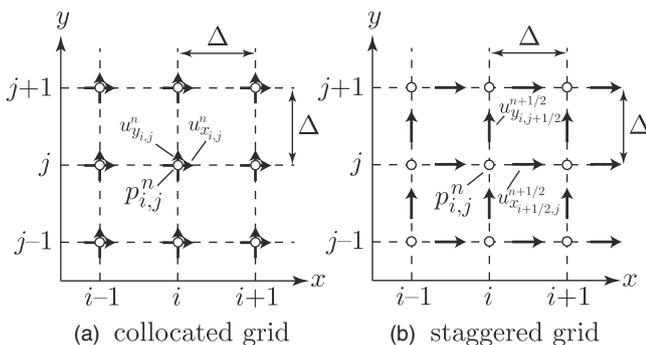


Fig. 1. Two types of grid for 2D FDTD method.

Table I. Comparison of memory required for the vector- and scalar-type FDTD methods. N is the memory required for the entire grid points for one physical quantity.

Dimension	Vector-type	Scalar-type
1D	$2N$	$2N$
2D	$3N$	$2N$
3D	$4N$	$2N$

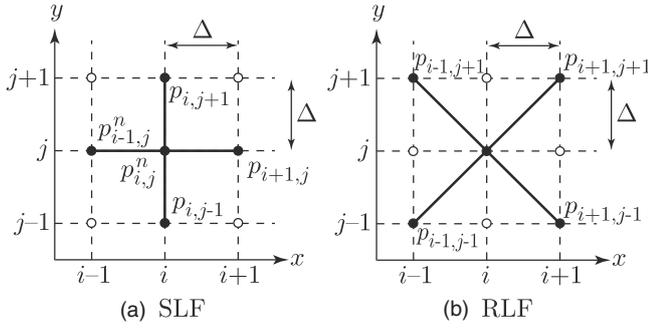


Fig. 2. Stencils for 2D CE-FDTD method.

leapfrog (RFL). The discretized equation of the RLF scheme is given by

$$p_{ij}^{n+1} = 2p_{ij}^n - p_{ij}^{n-1} + \frac{\chi^2}{2}(p_{i+1,j+1}^n + p_{i+1,j-1}^n + p_{i-1,j+1}^n + p_{i-1,j-1}^n) - 2\chi^2 p_{ij}^n. \quad (8)$$

The accuracy of the RLF scheme is different from the SLF scheme because the grid interval of the RLF scheme is $\sqrt{2}\Delta$. The numerical scheme of the CE-FDTD method is derived by combining the SLF and RFL schemes as

$$p_{ij}^{n+1} = 2p_{ij}^n - p_{ij}^{n-1} + \chi^2(1 - 2a) \times (p_{i+1,j}^n + p_{i-1,j}^n + p_{i,j+1}^n + p_{i,j-1}^n) + \chi^2 a (p_{i+1,j+1}^n + p_{i+1,j-1}^n + p_{i-1,j+1}^n + p_{i-1,j-1}^n) - 4\chi^2(1 - a)p_{ij}^n, \quad (9)$$

where a is a parameter for accuracy control. To confirm that Eq. (9) is a discretized equation of the wave equation, the Taylor series expansion is applied around p_{ij}^n .

$$\begin{aligned} & \frac{\partial^2 p}{\partial x^2} \Big|_{i,j}^n + \frac{\partial^2 p}{\partial y^2} \Big|_{i,j}^n - \frac{1}{c_0^2} \frac{\partial^2 p}{\partial t^2} \Big|_{i,j}^n \\ &= \frac{1}{6} \left\{ (1 - a) \left(\frac{\partial^4 p}{\partial x^4} \Big|_{i,j}^n + \frac{\partial^4 p}{\partial y^4} \Big|_{i,j}^n \right) + 6a \frac{\partial^4 p}{\partial x^2 \partial y^2} \Big|_{i,j}^n \right\} \Delta^2 + O(\Delta^3) \\ &+ \frac{1}{12c_0^2} \frac{\partial^4 p}{\partial t^4} \Big|_{i,j}^n (\Delta t)^2 + O((\Delta t)^3), \end{aligned} \quad (10)$$

where $\Big|_{i,j}^n$ denotes the value of the derivative of p at a point (i, j) and at a time n . The left hand side of Eq. (10) corresponds to the wave equation and the right one corresponds to the discretization error. It is found that the error term with respect to the spatial differentiation can be controlled by the parameter a . The greatest advantage of the CE-FDTD method is that the numerical accuracy can be improved by increasing the number of evaluation points without increasing the computer memory.

2.3.2. 3D CE-FDTD method. In the 3D CE-FDTD method, the second-order difference is evaluated by the

grid points along not only the axial and the face diagonal directions, but also the space diagonal directions as shown in Fig. 3. The face diagonal evaluation shown in Fig. 3(b) is called cubic close packed (CCP), and the space diagonal evaluation in Fig. 3(c) is called octahedral (OCTA). Considering all directions, the wave Eq. (3) is discretized in the CE-FDTD method as^{16,17)}

$$\delta_t^2 p_{i,j,k}^n = \chi^2 \{ (\delta_x^2 + \delta_y^2 + \delta_z^2) + a(\delta_x^2 \delta_y^2 + \delta_y^2 \delta_z^2 + \delta_z^2 \delta_x^2) + b\delta_x^2 \delta_y^2 \delta_z^2 \} p_{i,j,k}^n, \quad (11)$$

where a and b denote two independent numerical parameters, $p_{i,j,k}^n$ represents the sound pressure on the grid point $(x, y, z) = (i\Delta, j\Delta, k\Delta)$ at time $t = n\Delta t$. δ^2 is an operator on the central finite difference. For example,

$$\delta_t^2 p_{i,j,k}^n \equiv p_{i,j,k}^{n+1} - 2p_{i,j,k}^n + p_{i,j,k}^{n-1}, \quad (12)$$

$$\delta_x^2 p_{i,j,k}^n \equiv p_{i+1,j,k}^n - 2p_{i,j,k}^n + p_{i-1,j,k}^n. \quad (13)$$

δ_y^2 and δ_z^2 are given in the same manner. Equation (11) is then rewritten as

$$\begin{aligned} p_{i,j,k}^{n+1} &= d_1(p_{i+1,j,k}^n + p_{i-1,j,k}^n + p_{i,j+1,k}^n + p_{i,j-1,k}^n + p_{i,j,k+1}^n + p_{i,j,k-1}^n) \\ &+ d_2(p_{i+1,j+1,k}^n + p_{i+1,j-1,k}^n + p_{i+1,j,k+1}^n + p_{i+1,j,k-1}^n + p_{i-1,j+1,k}^n + p_{i-1,j-1,k}^n + p_{i-1,j,k+1}^n + p_{i-1,j,k-1}^n) \\ &+ d_3(p_{i+1,j+1,k+1}^n + p_{i+1,j-1,k+1}^n + p_{i+1,j+1,k-1}^n + p_{i+1,j-1,k-1}^n + p_{i-1,j+1,k+1}^n + p_{i-1,j-1,k+1}^n + p_{i-1,j+1,k-1}^n + p_{i-1,j-1,k-1}^n) \\ &+ d_4(p_{i,j,k}^n - p_{i,j,k}^{n-1}), \end{aligned} \quad (14)$$

where d_1 – d_4 are given by

$$d_1 = \chi^2(1 - 4a + 4b), \quad d_2 = \chi^2(a - 2b), \quad d_3 = \chi^2 b, \quad d_4 = 2(1 - 3\chi^2 + 6a\chi^2 - 4b\chi^2). \quad (15)$$

In Eq. (14), d_1 corresponds to the SLF scheme, d_2 corresponds to CCP, and d_3 corresponds to OCTA. Table II shows the derivative schemes in the 3D CE-FDTD method and their numerical parameters.^{16,17)} Other schemes can be configured by combining these stencils. In particular, the IWB scheme with evaluation of all directions shown in Fig. 3(d) has excellent characteristics as described later. The Courant number that determines the stability condition of the scheme has the maximum value χ_m shown in Table II.

3. Numerical characteristics of FDTD method

3.1. Numerical dispersion

Figure 4 shows the theoretical dispersion curves expressed in the sound speed for the representative propagation directions: $(1, 0, 0)$, $(1, 1, 0)$, and $(1, 1, 1)$.¹⁷⁾ In the figures, \bar{f} denotes the frequency normalized by the sampling frequency $f_s = 1/\Delta t$ and \bar{c} denotes the sound speed normalized by c_0 . In each curve, the Courant number is set to the maximum value χ_m of each scheme. As shown in the figures, the FDTD

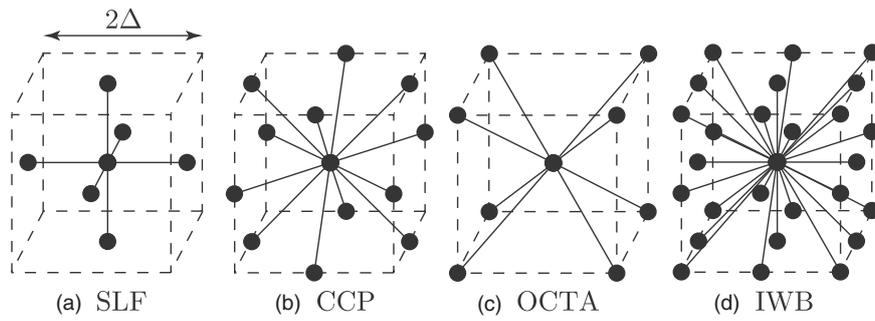


Fig. 3. Stencils for 3D CE-FDTD method.

Table II. Derivative schemes in the 3D CE-FDTD method and numerical parameters.

Method	a	b	d_1	d_2	d_3	d_4	χ_m	\bar{f}_c
SLF	0	0	1/3	0	0	0	$1/\sqrt{3}$	0.196
CCP	1/4	0	0	1/4	0	-1	1	0.333
OCTA	1/2	1/4	0	0	1/4	0	1	0.25
IWB	1/4	1/16	1/4	1/8	1/16	-3/2	1	0.5

method generally has the numerical dispersion characteristics in which the numerical sound speed becomes slow as frequency increases.

The numerical dispersion causes the collapse of the sound pressure waveform. Figure 5 shows the sound pressure waveforms calculated by the SLF and IWB schemes. In the calculation, a cubic domain is assumed and is divided into $256 \times 256 \times 256$ FDTD cells. The boundary condition is assumed to be the Mur's 1st order absorbing boundary.²⁸⁾ A differential Gaussian pulse is radiated from a point source located at the center of the domain and is received at the point $(x, y, z) = (64\Delta, 128\Delta, 128\Delta)$. Figure 5(a) shows the waveforms with the one-point driving, which is normally applied to the source driving in the FDTD method. The waveform calculated by the SLF scheme begins to collapse because of the numerical dispersion. However, the waveforms calculated by the IWB scheme show continuous high-frequency ringing in the tail of the waveform at a frequency corresponding to the cutoff frequency, as if it were diverging.

This is due to that the cutoff frequencies of the IWB scheme in each direction are concentrated at the Nyquist frequency.

To suppress this continuous ringing, an eight-points driving is devised in a FDTD cell.¹⁷⁾ This driving method acts as a spatial low-pass filter whose gain is zero at the Nyquist frequency. The results are shown in Fig. 5(b). The eight-point driving effectively suppresses the continuous ringing in the IWB scheme while it has little effect in the SLF scheme. It is found that the IWB scheme can perform accurate calculations without the numerical dispersion error by introducing the eight-points driving. For planar sources, although the numerical dispersion error depends on the source shape, this drive method is effective.

3.2. Cutoff frequency

Sound wave cannot propagate with frequency above the cutoff frequency \bar{f}_c ,^{16,29)} which is the upper limit of the normalized frequency of the dispersion curve as shown in Fig. 4. The cutoff frequency depends on the propagation direction and the lowest normalized cutoff frequency \bar{f}_c in each direction restricts the bandwidth of the sound field analysis. The lowest normalized cutoff frequency \bar{f}_c is also shown in Table II for each scheme. The IWB scheme has the comprehensively widest bandwidth in the CE-FDTD method, where the cutoff frequency corresponds to the Nyquist frequency.

Figure 6 shows the spectrograms of the impulse responses at the receiving point calculated by each scheme. The numerical model is the same with Fig. 5. In the case of

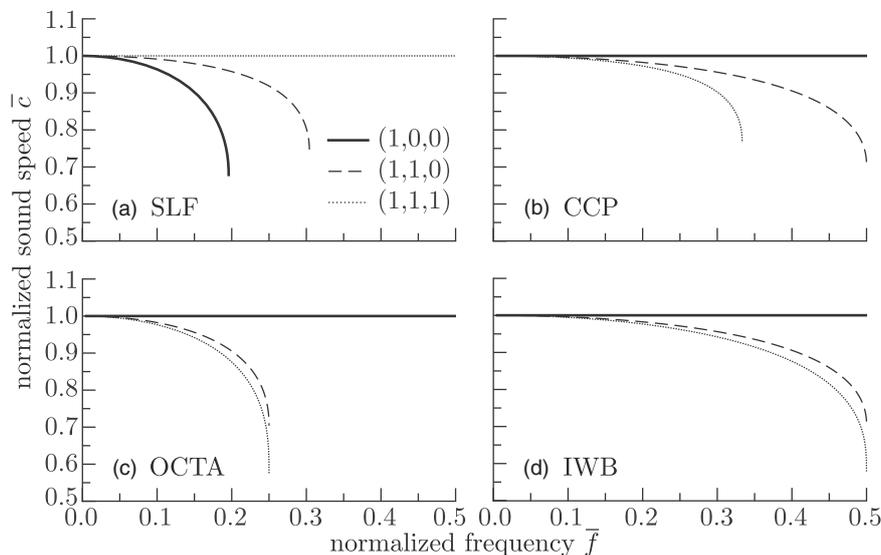


Fig. 4. Numerical dispersion curves expressed by sound speed for the representative propagation directions.

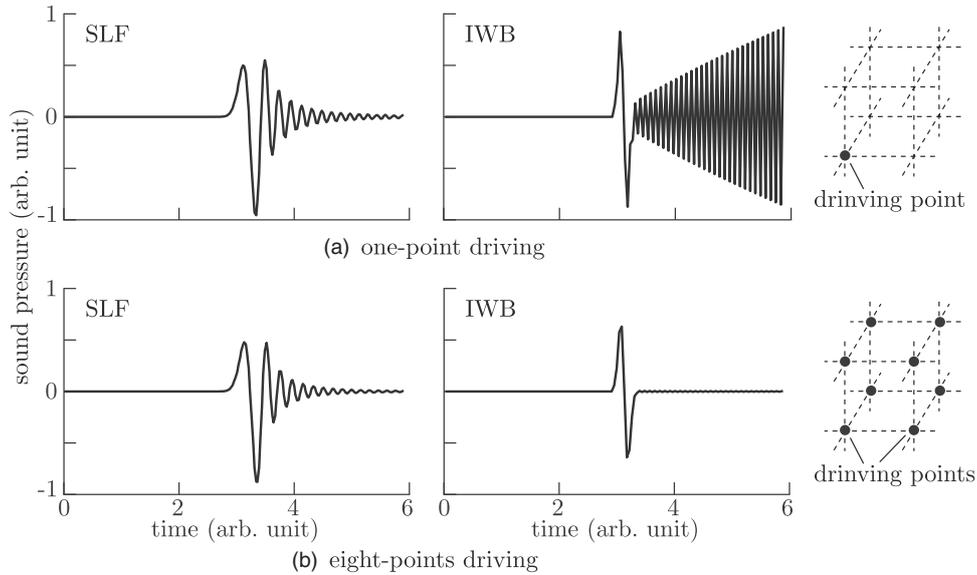


Fig. 5. Sound pressure waveforms calculated by the SLF and IWB schemes with one-point and eight-points driving.

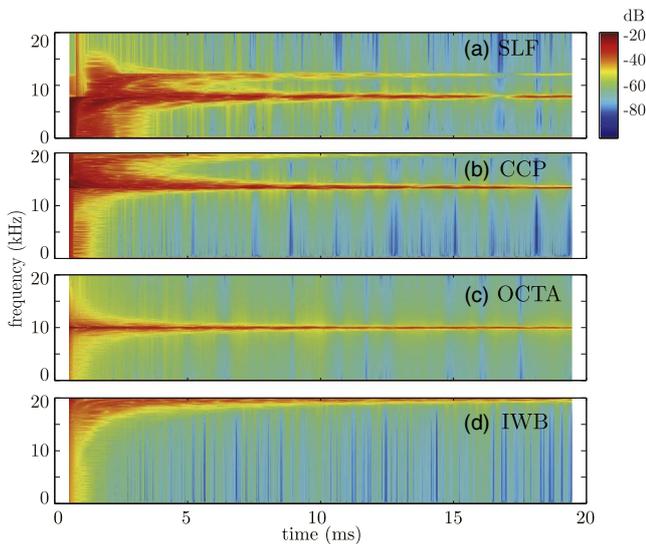


Fig. 6. (Color online) Spectrograms of the impulse responses calculated by each scheme.

impulse radiation, the ringing cannot be completely suppressed even by the eight-points driving, and some peaks are observed in the spectrogram corresponding to the cutoff frequency \bar{f}_c shown in Table II. The reason for the separation of the peaks in the SLF and CCP schemes is that the cutoff frequency depends on the direction, as shown in Fig. 4. It is found that the IWB scheme has the widest bandwidth.

The cutoff frequency also depends on the Courant number. Figure 7 shows the normalized cutoff frequency against the Courant number for each scheme. The normalized cutoff frequency becomes large as the Courant number increases for all schemes. It is found that the Courant number should be set as the upper limit of the scheme for accurate analysis.

As shown in Table II, the cutoff frequency and the maximum Courant number depend on the scheme. This means that the computer resources required for the calculation will be different for each scheme when the analysis is

performed at the same cutoff frequency. Table III shows the minimum computer resources required to analyze the response per unit volume and per unit time when the same cutoff frequency F_c is achieved. In the table, \sim indicates the ratio to the value of the SLF scheme, $f_s = F_c/\bar{f}_c$ is the sampling frequency to achieve the cutoff frequency F_c , $\Delta = c_0/(\chi_m f_s)$ is the grid interval, $N = 1/\Delta^3$ is the total number of the grid points per unit volume, n_f is the number of the floating point operations per cell and per time step, and $T_c = f_s N n_f$ is the total floating point operations per unit volume and per unit time. The amount of memory usage is proportional to the total number of grid points N , and the calculation time is proportional to the total floating point operations T_c . It is found that the amount of memory usage of the IWB scheme is less than the one-third of that of the SLF scheme to achieve the same cutoff frequency. It is also found that the calculation time of the IWB scheme is about 42% that of the SLF scheme for the same accuracy.

This is especially obvious when using the GPUs since most of the computation time is spent transferring data within the GPU. Figure 8 shows the memory usage and the calculation time for the impulse response of unit time measured using a single GPU. In the figure, the measured values are normalized by the results calculated by the SLF scheme. It is found that the memory usage of the IWB scheme is smallest and is about 31% of that of the SLF scheme, which agrees with the theoretical results in Table III. On the other hand, the measured calculation time does not agree with Table III because the calculation time is determined by not only the total number of floating point operations but also the data transfer time between GPU and the memory. So, the calculation time is proportional to $\tilde{N} \tilde{T}_c$. It is confirmed that the calculation time of the IWB scheme is shortest and is about 14% of that of the SLF scheme. It is also found that a method with a higher cutoff frequency is advantageous even if the computation is more complex, because it reduces the data transfer time due to the low memory usage.

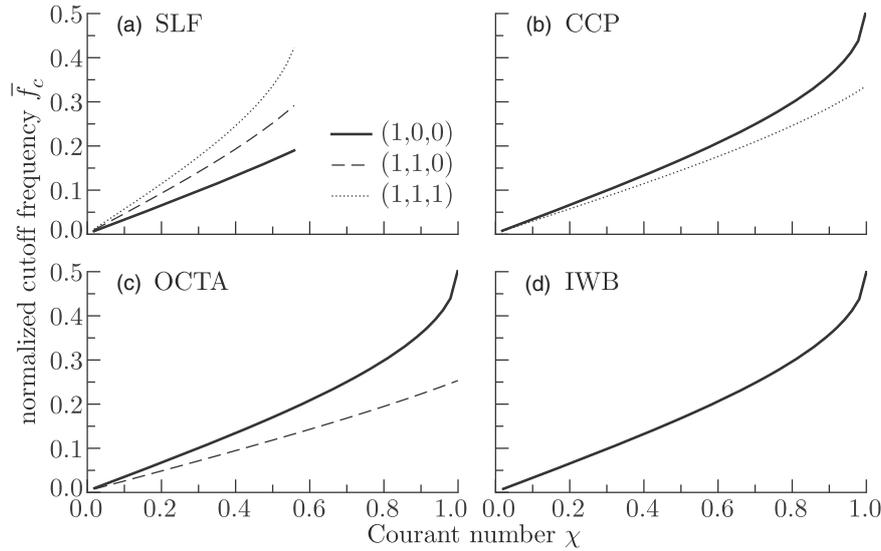


Fig. 7. Cutoff frequency against the Courant number for the representative propagation directions.

Table III. Minimum computer resources required to analyze the response per unit volume and per unit time when the same cutoff frequency is achieved. \sim indicates the ratio to the value of the SLF scheme, f_s is the sampling frequency to achieve the cutoff frequency F_c , Δ is the grid interval, N is the total number of the grid points, n_f is the number of the floating point operations per cell, and T_c is the total floating point operations.

Method	χ_m	\tilde{f}_c	\tilde{f}_s	$\tilde{\Delta}$	\tilde{N}	\tilde{n}_f	\tilde{T}_c
SLF	0.577	0.196	1	1	1	1	1
CCP	1	0.333	0.589	0.980	1.05	1.67	1.04
OCTA	1	0.250	0.784	0.736	2.51	1.22	2.40
IWB	1	0.500	0.392	1.47	0.314	3.44	0.423

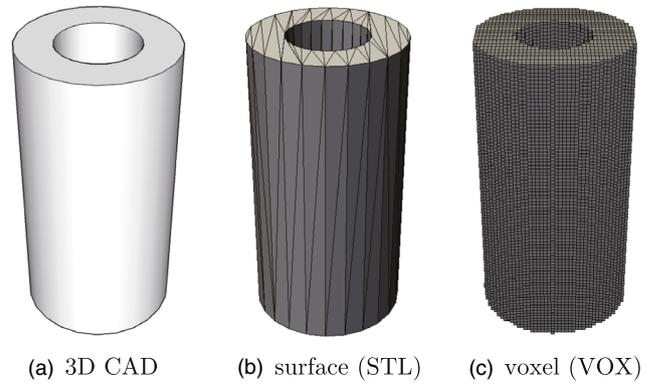


Fig. 9. (Color online) Surface model and voxel model.

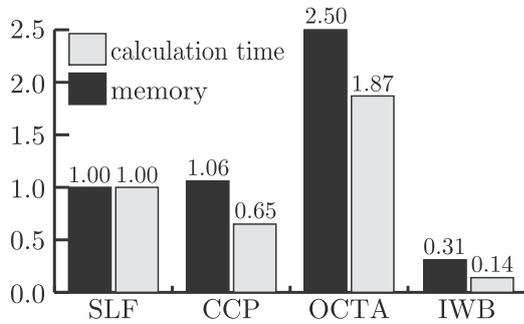


Fig. 8. Computational performances with consideration of the cutoff frequency in the case of a single GPU.

4. Techniques related to CE-FDTD method

4.1. Boundary conditions

In the CE-FDTD method, the object shape is represented by voxels. The voxel is a cube composed of the adjacent eight grid points. So, the entire interior of the analysis domain must be filled with voxels. However, in most of the 3D CAD systems, the object shape is represented only by the surface data as shown in Fig. 9(a), and the inside of the surface model is usually hollow. Therefore, it is necessary for the CE-FDTD method to create voxel data [Fig. 9(c)] from the general surface data such as the STL and OBJ formats [Fig. 9(b)]. For the large sound field analysis, voxels of 10^9 or more may be required. Since commercial voxelizers cannot handle a large

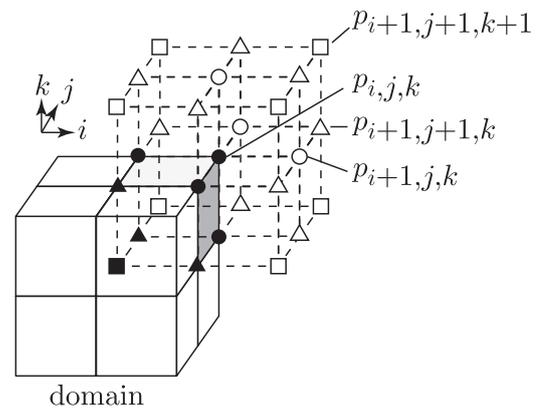
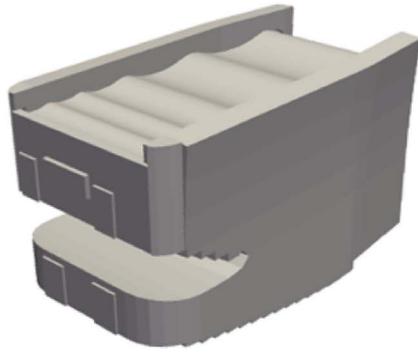


Fig. 10. Grid point on the boundary. The black points are defined in the domain or on the boundary and the white points are located outside the domain. The circles indicate grid points in the axial direction, the triangles indicate grid points in the face diagonal direction, and the squares indicate grid points in the space diagonal direction.

number of voxels, we have developed our own voxelizer “SUF2VOX”³⁰ that can generate voxels from the STL data.

To calculate the sound pressure at the grid points on the boundary of an arbitrary voxelized object, the sound pressures at the grid points outside the domain are required, but these are not defined in the computation. Therefore, it is necessary to estimate the sound pressure at the grid points



(a) exterior view



(b) interior view from stage

Fig. 11. (Color online) A Yamaha hall model.

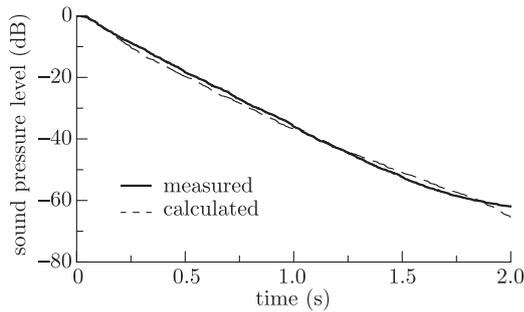


Fig. 12. Reverberation curve of the Yamaha hall model at 1 kHz.

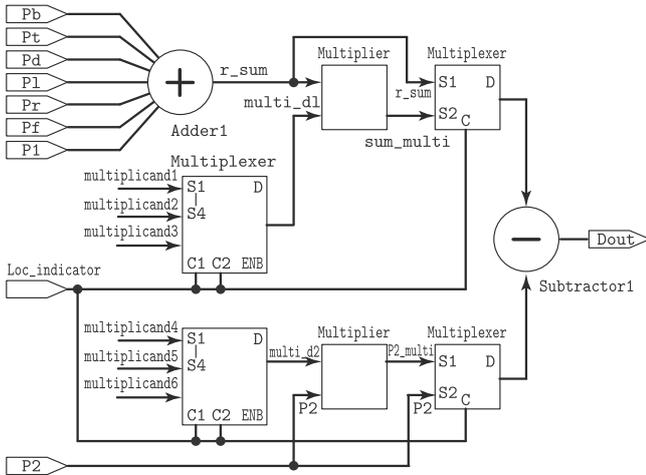


Fig. 13. A compute unit of 3D FDTD method by FPGA.

outside the domain by applying the boundary condition. For an example, we here consider the calculation of the sound pressure $p_{i,j,k}^{n+1}$ at the grid point located on the edge of the domain indicated by the black circle as shown in Fig. 10. In the figure, the black points are defined in the domain or on the boundary, while the white points are located outside the domain, and are not defined in the computation. To estimate the sound pressure at the white grid points in the SLF scheme, a simple impedance relation for the normal incidence is applied in the axial direction. However in other schemes of the CE-FDTD method, the different boundary condition is applied to other white points depending on the position of the grid point.¹⁸⁾

In many sound field analyses, the perfect matched layer (PML)³¹⁾ is used as the absorbing boundary condition. The

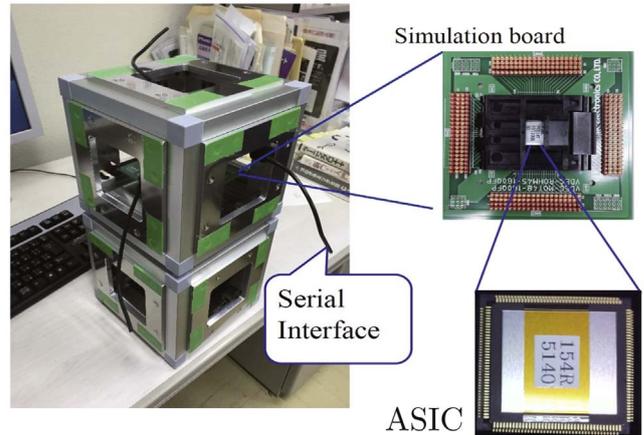


Fig. 14. (Color online) A rendering block prototyped by ASIC.

PML was considered inapplicable to the scalar-type FDTD method because it requires the particle velocity for the impedance matching between the region and the PML. However, the PML can be adjoined with the computational domain of the scalar-type FDTD method by using Eqs. (5) and (6) to calculate the particle velocity locally on the boundary.²⁶⁾ However, a PML for the CE-FDTD method has not yet been developed because of the difficulty of the multi-directional impedance matching.

4.2. Density variation

Since the density does not appear explicitly in the wave equation, it was considered to be difficult to analyze the sound propagation in a medium with density variation using the CE-FDTD method. However, by considering the difference in density of each cell when eliminating the particle velocity from the governing Eqs. (1) and (2), we can obtain a second-order difference equation that includes the density. For an example, the second-order difference equation with respect to x -direction including the density is expressed as follows.¹⁹⁾

$$\delta_x^2 p_{i,j,k}^n \equiv 2\rho_{i,j,k} \left(\frac{p_{i+1,j,k}^n - p_{i,j,k}^n}{\rho_{i+1,j,k} + \rho_{i,j,k}} - \frac{p_{i,j,k}^n - p_{i-1,j,k}^n}{\rho_{i,j,k} + \rho_{i-1,j,k}} \right), \quad (16)$$

where $\rho_{i,j,k}$ represents the density in a cube with one side Δ centered on the grid point (i, j, k) . The other directions can be obtained in the same manner. The discretized equation for the CE-FDTD method with density variation is represented as¹⁹⁾

$$\delta_i^2 p_{i,j,k}^n = \chi^2 \{ (\bar{\delta}_x^2 + \bar{\delta}_y^2 + \bar{\delta}_z^2) + a(\bar{\delta}_x^2 \bar{\delta}_y^2 + \bar{\delta}_y^2 \bar{\delta}_z^2 + \bar{\delta}_z^2 \bar{\delta}_x^2) + b \bar{\delta}_x^2 \bar{\delta}_y^2 \bar{\delta}_z^2 \} p_{i,j,k}^n. \quad (17)$$

5. Application of FDTD method

5.1. Sound field rendering

Sound field rendering is a technology that the sound pressure waveform at the listening position is numerically calculated by the wave based 3D sound field analysis such as the FDTD method, and then to make it audible in 3D audio. This is based on the same concept as the image rendering in the 3D computer graphics. The sound field rendering for the large scale sound field such as concert halls requires enormous computer resources, but the application of the IWB scheme and its implementation on a GPU cluster system have made it almost practical.

As an example of the sound field rendering, the Yamaha hall model is produced and calculated²¹⁾ as shown in Fig. 11. The hall size is $12 \times 21.5 \times 13$ meters with 333 seats. For the calculation condition of the IWB scheme, the sound speed is 340 m s^{-1} , the CFL number is 0.95, the grid interval is 8.5 mm, the sound pressure reflection coefficient of the wall is 0.93, and the number of calculation steps is $100\,000 \Delta t$. Figure 12 shows the reverberation curve at 1 kHz calculated from the normal incidence impulse response. In the figure, the solid line is the measured curve and the dashed line is the calculated one. They are in good agreement, indicating that the rendering results are reasonable. The rendering results are audible by a 24-channel hemispherical speaker array system installed in the Doshisha University, and the appropriate reverberation is reproduced, giving a sense of presence as if one were in the hall.

5.2. Silicon concert hall

Sound field rendering requires the pre-rendering, in which the impulse responses are calculated in advance for a considerable time. Therefore, a real-time renderer is essential to realize the rendering that can freely change the listening point. The silicon concert hall is such a real-time sound field renderer that can simulate the hall acoustics on silicon chips.³²⁾ For the realization of the silicon concert hall, the parallelization of the compute units and the faster memory transfers are required. The memory transfer is a bottleneck for the GPU-based computation because current GPUs have their memories outside the compute unit, and the data transfer from the memory to the unit is very costly. Therefore, if the memory can be directly connected to the compute unit, the cost problem of data transfer can be solved.

Figure 13 shows a block diagram of a 3D SLF-based sound pressure compute unit designed on a field programmable gate array (FPGA) device.³³⁾ The sound pressure can be calculated and memorized for a single FDTD cell. The compute units are arranged three-dimensionally, and exchange data with each other. Figure 14 shows a rendering block prototyped on an application specific integrated circuit (ASIC).³⁴⁾ The rendering block can calculate $4 \times 4 \times 4$ grid points. This prototype still works only as a stand-alone renderer, but in the future it may be possible to connect

multiple blocks to calculate the sound field corresponding to the physical shape of the blocks in real time.

6. Conclusions

A tutorial on the basics and applications of the FDTD method was described. For the analysis of the linear sound wave propagation by the standard FDTD method, two types of discretization of the governing equations, the scalar-type FDTD method and the vector-type one were discussed. It was shown that the scalar-type FDTD method is advantageous from the viewpoint of the computer resources. Then the basic concept of the compact explicit-FDTD (CE-FDTD) method was described. By considering the relationship between the cutoff frequency and the computer resources, it was shown that the IWB scheme requires the least computer resources among the derivative schemes of the CE-FDTD method. The discretization of the arbitrary shaped sound field by voxels and its boundary conditions, and the density variation were also described. The sound field rendering and its real time renderer “Silicon concert hall” were introduced. We hope this paper will be useful for beginners in the numerical simulations of the sound fields.

- 1) R. Courant, K. Friedrichs, and H. Lewy, *IBM J. Res. Dev.* **11**, 215 (1967).
- 2) O. C. Zienkiewicz, *The Finite Element Method* (McGraw-Hill, New York, 1977).
- 3) C. A. Brebbia and S. Walker, *Boundary Element Techniques in Engineering* (Butterworth-Heinemann, London, 1979).
- 4) K. S. Yee, *IEEE Trans. Antennas Propag.* **14**, 302 (1966).
- 5) T. Tsuchiya and A. Kumagai, *Jpn. J. Appl. Phys.* **48**, 07GN02 (2009).
- 6) T. Yoda, K. Okubo, N. Tagawa, and T. Tsuchiya, *Jpn. J. Appl. Phys.* **50**, 07HC12 (2011).
- 7) T. Yiyu, Y. Inoguchi, Y. Sato, M. Otani, Y. Iwaya, H. Matsuoka, and T. Tsuchiya, *Jpn. J. Appl. Phys.* **52**, 07HC03 (2013).
- 8) T. Yiyu, Y. Inoguchi, Y. Sato, M. Otani, Y. Iwaya, H. Matsuoka, and T. Tsuchiya, *Jpn. J. Appl. Phys.* **53**, 07KC14 (2014).
- 9) A. Hosokawa, *Jpn. J. Appl. Phys.* **57**, 07LF06 (2018).
- 10) T. Tsukui, S. Hirata, and H. Hachiya, *Jpn. J. Appl. Phys.* **58**, SGGB12 (2019).
- 11) L. Bustamante, M. Saeki, and M. Matsukawa, *Jpn. J. Appl. Phys.* **58**, SGGE20 (2019).
- 12) L. Bustamante, M. Saeki, T. Misaki, and M. Matsukawa, *Jpn. J. Appl. Phys.* **59**, SKKB05 (2020).
- 13) D. Ito, T. Oguri, N. Kamiyama, S. Hirata, K. Yoshida, and T. Yamaguchi, *Jpn. J. Appl. Phys.* **60**, SDDE11 (2021).
- 14) T. Tsuchiya and M. Kanamori, *Jpn. J. Appl. Phys.* **60**, SSDB02 (2021).
- 15) K. Kowalczyk and M. V. Walstijn, *IEEE Trans. Audio Speech Lang. Process.* **18**, 78 (2010).
- 16) K. Kowalczyk and M. V. Walstijn, *IEEE Trans. Audio Speech Lang. Process.* **19**, 34 (2011).
- 17) T. Ishii, T. Tsuchiya, and K. Okubo, *Jpn. J. Appl. Phys.* **52**, 07HC11 (2013).
- 18) O. Yamashita, T. Tsuchiya, Y. Iwaya, M. Otani, and Y. Inoguchi, *Jpn. J. Appl. Phys.* **54**, 07HC02 (2015).
- 19) T. Tsuchiya and N. Maruta, *Jpn. J. Appl. Phys.* **57**, 07LC01 (2018).
- 20) T. Tsuchiya, Y. Iwaya, and M. Otani, *Proc. Int. Congress. Acoust. ICA.*, 2013, 10.1121/1.4798996.
- 21) T. Tsuchiya, I. Takenuki, and K. Sugiura, *Conf. Paper 2018 Audio Engng. Soc. Int. Conf. on Spatial Reproduction*, PP-5, 2018.
- 22) T. Tsuchiya, M. Kanamori, and T. Takahashi, *Jpn. J. Appl. Phys.* **58**, SGGB01 (2019).
- 23) T. Tsuchiya, Y. Teshima, and S. Hiryu, *Acoust. Sci. Technol.* **43**, 57 (2022).
- 24) D. Botteldooren, *J. Acoust. Soc. Am.* **95**, 2313 (1994).
- 25) O. Chiba, T. Kashiwa, H. Shimoda, S. Kagami, and I. Fukai, *J. Acoust. Soc. Jpn.* **49**, 551 (1993) [in Japanese].
- 26) Y. Miyazaki and T. Tsuchiya, *Jpn. J. Appl. Phys.* **51**, 07GB02 (2012).
- 27) S. Nakai, T. Ishii, and T. Tsuchiya, *Jpn. J. Appl. Phys.* **51**, 07GG07 (2012).

- 28) G. Mur, *IEEE Trans. Electromagn. Compat.* **EMC-23**, 377 (1981).
- 29) S. Bilbao, *Wave and Scattering Methods for Numerical Simulation* (Wiley, London, 2004).
- 30) T. Tsuchiya, "Development of a voxelizer for finite difference time domain method with arbitrary boundary in sound field analysis," IEICE Tech. Rep. EA2016-28 (2016) [in Japanese].
- 31) J.-P. Berenger, *J. Comput. Phys.* **114**, 185 (1994).
- 32) T. Tsuchiya, *J. Virtual Reality Soc. Jpn.* **25**, 19 (2021), [in Japanese].
- 33) T. Yiyu, Y. Inoguchi, M. Otani, Y. Iwaya, and T. Tsuchiya, *Appl. Sci.* **8**, 35 (2018).
- 34) Y. Iwaya and Proc, Proc. ICT Innovation Forum 2016, 131 202 005, Strategic Info. and Comm. R&D Promotion Programme, 2016.