Chapter 5
Alternative Time-Domain Methods

Takuya Oshima, Takashi Ishizuka and Kan Okubo

Abstract The finite-difference time-domain (FDTD) method has been the only choice of time-domain methods for practical applications with its simplicity and efficiency. However, the simple discretization of the simple wave equation model in which the method has its basis is not sufficient for modeling more complex wave propagation phenomena, high-accuracy simulations, or acoustic fields with complex geometries. In this chapter, alternative time-domain methods that may be applied to such situations are discussed as follows: the linearized Euler equation (LEE) method, the constrained interpolation profile (CIP) method, and the finite-volume time-domain (FVTD) method. The LEE method is applicable to wave propagation phenomena under the influence of arbitrary background flows. The main application of the method is sound propagation simulations outdoors where wind effects are not negligible. The CIP method is characteristic in that the method is in principle free from numerical dispersion. The characteristic allows simulations with high phase accuracy. The FVTD method is constructed on an unstructured grid system. The method thus has an advantage in modeling complex geometries compared to the FDTD method where orthogonal structured grid is used.

T. Oshima (✉)
Faculty of Engineering, Niigata University, 8050 Ikarashi-Ninocho, Nishi-ku,
Niigata-shi, Niigata 950-2181, Japan
e-mail: oshima@eng.niigata-u.ac.jp

T. Ishizuka
Institute of Technology, Shimizu Corporation, 3-4-17 Etchujima, Koto-ku,
Tokyo 135-8530, Japan
e-mail: ishiduka@shimz.co.jp

K. Okubo
Graduate School of System Design, Tokyo Metropolitan University, 6-6 Asahigaoka,
Hino-shi, Tokyo 191-0065, Japan
e-mail: kanne@tmu.ac.jp

T. Sakuma et al. (eds.), *Computational Simulation in Architectural and Environmental Acoustics*, DOI: 10.1007/978-4-431-54454-8_5,
© Springer Japan 2014

Keywords Linearized Euler equation (LEE) method · Wind effects · Constrained interpolation profile (CIP) method · Method of characteristics (MOC) · Finite-volume time-domain (FVTD) method · Unstructured grid

5.1 Linearized Euler Equation Method

The linearized Euler equation (LEE) method is an approach to solve sound propagation under an arbitrary background flow by an addition of the advection effect of the medium to the basic equations of the finite-difference time-domain (FDTD) method. The main application of the method is sound propagation simulations outdoors with wind effects.

5.1.1 Governing Equations

The medium velocity U is represented as a composition of the background flow component $\bar{U}$ and the acoustic particle velocity component U' as

$$U = \bar{U} + U'.$$

Other physical quantities are similarly represented. For example, pressure deviation from the average medium air pressure is represented by decomposing into the airflow component $\bar{p}$ and the acoustic pressure component p' as

$$p = \bar{p} + p'.$$

For outdoor acoustic propagation problems, a simplified linearized Euler model under adiabatic, barotropic, and non-buoyant conditions can be employed. In the simplified model, the fundamental equations of fluid flow, the equations of motion, and continuity, are linearized by the background flow components of velocity, pressure, and density, $\bar{U}$, $\bar{p}$ and $\bar{\rho}$ respectively [1, 2].

$$\frac{\partial U'}{\partial t} = -\left(\bar{U} \cdot \nabla\right) U' - \left(U' \cdot \nabla\right) \bar{U} - \frac{1}{\bar{\rho}} \nabla p', \tag{5.1}$$

$$\frac{\partial p'}{\partial t} = -\bar{\rho}\bar{c}^2 \nabla \cdot U' - \bar{U} \cdot \nabla p', \tag{5.2}$$

where $\bar{\rho}$ and $\bar{c}$ are the medium air density and the speed of sound respectively.

5.1.2 Discretization

Equations (5.1), (5.2) are discretized using a three-dimensional orthogonal staggered grid where p' is located at the cell centroid and each component of U' is located at the cell face of the respective coordinate direction. The pressure and velocity variables are located at staggered positions in time as well. Hereafter, the components of the background flow vector and the acoustic velocity vector are denoted as follows:

$$\bar{U} = \{\bar{u}\ \bar{v}\ \bar{w}\}^{\mathrm{T}},\ U' = \{u'\ v'\ w'\}^{\mathrm{T}}.$$

where ^{T} denotes the transpose operator.

5.1.2.1 Temporal Discretization

Time advancement is performed using the prediction-step staggered-in-time (PSIT) technique [3], which allows a simple implementation and smaller memory footprints. The technique calculates firstly a predicted value of the acoustic velocity at time step n, $\tilde{U}'^{n}$, using the acoustic velocity at time step $n - 1/2$, $U'^{n-1/2}$, and the acoustic pressure at time step n, p'^{n}, with the following equation

$$\tilde{U}'^{n} = U'^{n-1/2} - \frac{\Delta t}{2\bar{\rho}} \nabla p'^{n}, \tag{5.3}$$

which is a backward Euler discretized form of Eq. (5.1) between time step $n - 1/2$ to n, but without the contribution of the background flow velocity $\bar{U}$

$$\frac{\partial U'}{\partial t} = -\frac{1}{\bar{\rho}} \nabla p'. \tag{5.4}$$

The predicted value is used to calculate the acoustic velocity at the next time step $n + 1/2$, $U'^{n+1/2}$, using a timewise discretized form of Eq. (5.1).

$$U'^{n+1/2} = U'^{n-1/2} - \frac{\Delta t}{\bar{\rho}} \nabla p'^{n} - \Delta t\ \left(\bar{U} \cdot \nabla\right) \tilde{U}'^{n} - \Delta t\ \left(\tilde{U}'^{n} \cdot \nabla\right) \bar{U}. \tag{5.5}$$

Similarly, for Eq. (5.2) the predicted value of the acoustic pressure at time step $n+1/2$ is calculated by using the acoustic pressure at time step n, p'^{n}, and the acoustic velocity at time step $n+1/2$, $U'^{n+1/2}$. The predicted value in turn is used to calculate the acoustic pressure at the next time step $n + 1$, p'^{n+1}.

$$\tilde{p}'^{n+1/2} = p'^{n} - \frac{\bar{\rho}\bar{c}^{2} \Delta t}{2} \nabla \cdot U'^{n+1/2}, \tag{5.6}$$

$$p'^{n+1} = p'^{n} - \bar{\rho}\bar{c}^{2} \Delta t\, \nabla \cdot U'^{n+1/2} - \Delta t\, \bar{U} \cdot \nabla \tilde{p}'^{n+1/2}. \tag{5.7}$$

5.1.2.2 Spatial Discretization

The spatial derivatives in the prediction-step equations, namely Eqs. (5.3), (5.6), are discretized by the second-order central finite difference scheme that are widely used in usual FDTD simulations. For example, the x component of $\nabla p'^n$ in the right hand side of Eq. (5.3) is written as

$$\left.\frac{\partial p'^n}{\partial x}\right|_{i+1/2,j,k} \approx \frac{p'^n_{i+1,j,k} - p'^n_{i,j,k}}{\Delta x}, \tag{5.8}$$

where Δx is the x directional grid spacing.

On the other hand, Eqs. (5.5), (5.7) are simplified assuming that the background flow $\bar{U}$ is a uniform x-directional flow of speed $\bar{u}$

$$\bar{U} = \{\bar{u}\ 0\ 0\}^{\mathrm{T}}.$$

By substituting this equation to Eq. (5.5), only the x-directional derivative remains in the third term in the right hand side of Eq. (5.5). Besides, the fourth term in the right hand side is cancelled out due to the omnidirectional zero gradient of $\bar{U}$. Hence, by representing each component of $\tilde{U}'$ by

$$\tilde{U}' = \{\tilde{u}'\ \tilde{v}'\ \tilde{w}'\}^{\mathrm{T}},$$

each component of Eq. (5.5) and Eq. (5.7) are written as follows.

$$u'^{n+1/2}_{i+1/2,j,k} = u'^{n-1/2}_{i+1/2,j,k} - \frac{\Delta t}{\bar{\rho}} \left.\frac{\partial p'^n}{\partial x}\right|_{i+1/2,j,k} - \Delta t\, \bar{u} \left.\frac{\partial \tilde{u}'^n}{\partial x}\right|_{i+1/2,j,k}, \tag{5.9}$$

$$v'^{n+1/2}_{i,j+1/2,k} = v'^{n-1/2}_{i,j+1/2,k} - \frac{\Delta t}{\bar{\rho}} \left.\frac{\partial p'^n}{\partial y}\right|_{i,j+1/2,k} - \Delta t\, \bar{u} \left.\frac{\partial \tilde{v}'^n}{\partial x}\right|_{i,j+1/2,k}, \tag{5.10}$$

$$w'^{n+1/2}_{i,j,k+1/2} = w'^{n-1/2}_{i,j,k+1/2} - \frac{\Delta t}{\bar{\rho}} \left.\frac{\partial p'^n}{\partial z}\right|_{i,j,k+1/2} - \Delta t\, \bar{u} \left.\frac{\partial \tilde{w}'^n}{\partial x}\right|_{i,j,k+1/2}, \tag{5.11}$$

$$\begin{aligned} p'^{n+1}_{i,j,k} = p'^n_{i,j,k} - \bar{\rho}\bar{c}^2 \Delta t \Bigg(& \left.\frac{\partial u'^{n+1/2}}{\partial x}\right|_{i,j,k} + \left.\frac{\partial v'^{n+1/2}}{\partial y}\right|_{i,j,k} \\ & + \left.\frac{\partial w'^{n+1/2}}{\partial z}\right|_{i,j,k} \Bigg) - \Delta t\, \bar{u} \left.\frac{\partial \tilde{p}'^{n+1/2}}{\partial x}\right|_{i,j,k}. \end{aligned} \tag{5.12}$$

The spatial derivatives in Eqs. (5.9)–(5.12) are discretized by two strategies: one is to apply the second-order central finite difference scheme, which is denoted hereafter as Type FD-2nd. Another is to apply a compact difference scheme to the third term in the right hand side of each equation and to apply the second-order central finite difference scheme to all other terms, which is denoted hereafter as Type FD-Compact.

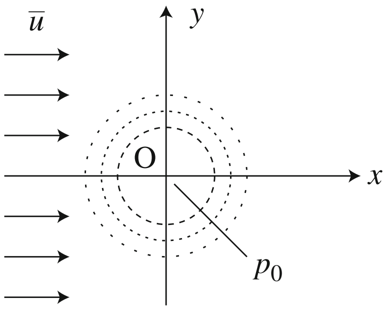


Fig. 5.1 Acoustic pulse in a uniform flow

Type FD-2nd. Taking the x-directional momentum equation Eq. (5.9) as example, the spatial derivative of the second term in the right hand side is discretized by Eq. (5.8). The third term is discretized using the finite difference of double grid spacing $2\Delta x$ as follows:

$$\left.\frac{\partial \tilde{u}'^n}{\partial x}\right|_{i+1/2,j,k} \approx \frac{\tilde{u}'^n_{i+3/2,j,k} - \tilde{u}'^n_{i-1/2,j,k}}{2\Delta x}.$$

Type FD-Compact. Taking Eq. (5.9) as example, the spatial derivative of the second term in the right hand side is identically discretized to Type FD-2nd by Eq. (5.8). The third term, on the other hand, is discretized with the Padé scheme [4], which is a fourth-order compact difference scheme that represents the approximation of $\partial \tilde{u}'^n/\partial x$, ϕ'^n, as follows:

$$\frac{1}{4}\phi'^n_{i-1/2,j,k} + \phi'^n_{i+1/2,j,k} + \frac{1}{4}\phi'^n_{i+3/2,j,k} = \frac{3}{4}\frac{\tilde{u}'^n_{i+3/2,j,k} - \tilde{u}'^n_{i-1/2,j,k}}{\Delta x}.$$

The matrix equation of ϕ'^n obtained by applying the Padé scheme to the entire grid is solved by a tridiagonal matrix algorithm (TDMA) solver [5].

5.1.3 Computational Setup

A sound propagation problem in a uniform medium under a uniform background flow within a two-dimensional free space is solved in order to compare the formulations and the analytical solutions.

An x-directional uniform background flow of speed $\bar{u}$ is given as the background flow. The initial acoustic pressure and acoustic velocity are given as

$$
\begin{aligned}
p' &= p_0 \exp\left\{-\alpha\left(x^2+y^2\right)\right\},\\
U' &= 0,
\end{aligned}
$$

as shown in Fig. 5.1. Under the initial conditions, the analytical solution of the acoustic pressure is given by Ref. [6] as

$$
\begin{aligned}
p' &= \frac{p_0}{2\alpha}\int_0^\infty \exp\left(-\frac{\xi^2}{4\alpha}\right)\cos\left(\bar{c}\xi t\right) J_0\left(\xi\eta\right)\xi d\xi,\\
\eta &= \sqrt{(x-\bar{u}t)^2+y^2},\\
\alpha &= \frac{\log 2}{b^2},
\end{aligned}
$$

where p_0 is the initial pressure amplitude, b is the half width at half maximum of the initial pressure amplitude, and J_0 is the Bessel function of the first kind of order 0. In the tests, b is set to 3 and p_0 and $\bar{c}$ are set to unity. The Mach number is assumed to be 0.1, namely

$$
M = \frac{\bar{u}}{\bar{c}} = 0.1.
$$

The computational domain is a square of diagonal points $(x, y) = (-100, -100)$ and (100, 100), which is divided by an orthogonal uniform grid of 201 × 201 × 1 cells in $x \times y \times z$ directions. The time step Δt is set to 0.5.

5.1.4 Results

The contours of instantaneous pressure amplitudes $|p'|$ at time $t = 40$ and 80 for the analytical solutions and the discretized formulations are shown in Fig. 5.2. Slow advection in x-direction following the uniform background flow of $M = 0.1$ is seen in all cases. At $t = 40$, all computed results show good agreements with the analytical case. However, as the propagation distance grows, at $t = 80$ we see differences. At the downstream direction, a numerical dispersion is seen in Type FD-2nd. On the contrary, Type FD-Compact shows no such error and agrees well with the analytical case.

5.2 Constrained Interpolation Profile Method

The constrained interpolation profile (CIP) method was developed in the field of fluid dynamics as a kind of the method of characteristics (MOC) [7–9]. This method has been applied to numerical simulations of sound field in time domain [10–14] as it

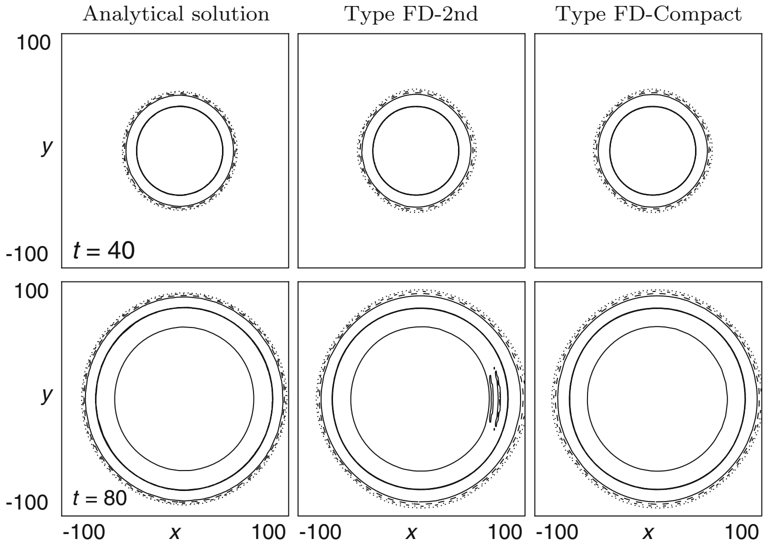
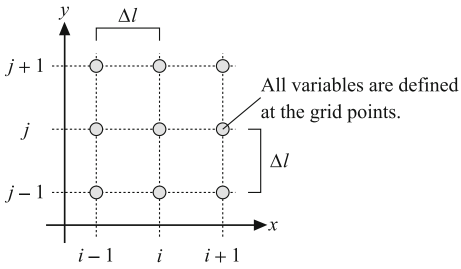


Fig. 5.2 Propagation of acoustic pulse in uniform flow at $t = 40$ and 80. *Dotted*, *dash-dotted*, *dashed* and *solid iso-contours* denote $|p'|$ values defined by a geometric ratio of 4 (which corresponds to about 12 dB in relative SPL) from 6.25×10^{-5} to 0.004

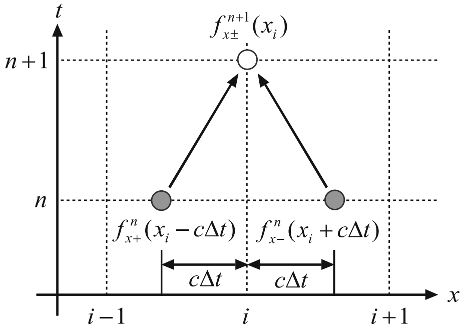
Fig. 5.3 Collocated grid system in the CIP method



has an advantage of low numerical dispersion over the FDTD method. Additionally, a spatial grid size, which the CIP method requires for adequate accuracy, is larger than that the FDTD method requires. High accuracy is achieved by a technique called "multi-moments" where not only physical values, such as the sound pressure and the particle velocity, but also their spatial derivatives are explicitly and simultaneously calculated at all grid points. In the MOC, including the CIP method, a time step size is free from Courant condition. The CIP method does not require a small time step size other time-domain methods require for accuracy and stability, resulting in a reduction of total computational time.

A collocated grid system is employed in the CIP method. All discrete physical values and their spatial derivatives are defined at the same grid point as shown in Fig. 5.3.

Fig. 5.4 Advection of values in the method of characteristics



5.2.1 Formulation

The CIP method for acoustic simulation computes advection equations derived from the governing equations of sound fields. The equation of continuity and the equation of motion for a lossless linear sound field are transformed into

$$\partial_t p + cZ\partial_x v_x = 0, \quad Z\partial_t v_x + c\partial_x p = 0, \tag{5.13}$$

where ∂_α represents an operator $\partial/\partial\alpha$, and p, v_x, c, and Z are the sound pressure, the particle velocity in the x-direction, the speed of sound, and the characteristic impedance of a medium, respectively. It should be noted that these equations express one-dimensional wave propagation in the x-direction. Addition and subtraction of the two equations in Eq. (5.13) lead to the following advection equations:

$$\partial_t f_{x\pm} \pm c\partial_x f_{x\pm} = 0, \quad \text{for} \quad f_{x\pm} = p \pm Zv_x. \tag{5.14}$$

f_{x+} and f_{x-}, which are called “characteristic curves”, represent forward and backward components of the sound field in the x-direction and Eq. (5.14) expresses propagation of them with the velocity of c.

In the MOC, the advection equations expressing wave propagation are computed by advection calculations schematically illustrated in Fig. 5.4. As shown in the diagram, advecting $f_{x\pm}$ at points $\mp c\Delta t$ distant from a grid point x_i at a time step n gives those at x_i at the next time step:

$$f_{x\pm}^{n+1}(x_i) = f_{x\pm}^{n}(x_i \mp c\Delta t), \tag{5.15}$$

where Δt is the time step size, and the superscripts n and $n+1$ denote time steps. When the points $x_i \mp c\Delta t$, called advection sources, are not at grid points, $f_{x\pm}^{n}(x_i \mp c\Delta t)$ are obtained by using interpolations.

In the CIP method, a kind of the MOC, values at advection sources are interpolated with high accuracy by using the Hermite interpolation [7, 15], normally the third-order Hermite interpolation. These calculations need spatial derivatives of $f_{x\pm}$ at each

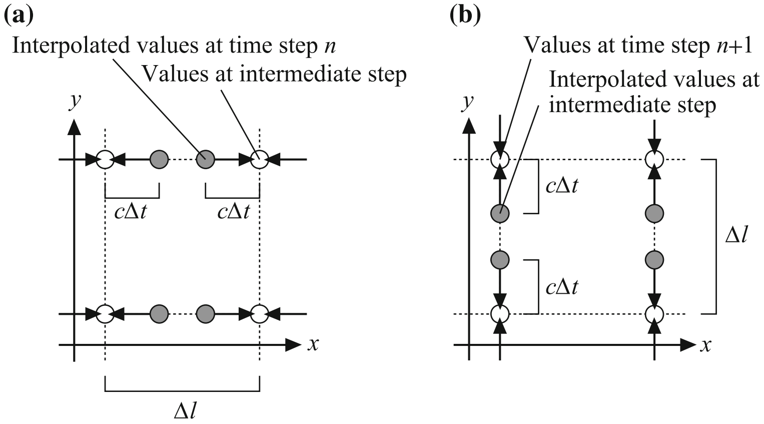

(a)
Interpolated values at time step n
Values at intermediate step
y
x
$c\Delta t$
$c\Delta t$
Δl
(b)
Values at time step n+1
Interpolated values at intermediate step
y
x
$c\Delta t$
$c\Delta t$
Δl

Fig. 5.5 Directionally separated advection of values in the multi-dimensional CIP method. **a** x-directional advection. **b** y-directional advection

grid. Advection equations for the derivatives are derived from the differentiation of Eq. (5.14):

$$\partial_t g_{x\pm} \pm c\partial_x g_{x\pm} = 0, \quad \text{for} \quad g_{x\pm} = \partial_x p \pm Z\partial_x v_x. \tag{5.16}$$

The CIP method simultaneously computes Eq. (5.14) and (5.16) using the third-order Hermite interpolation.

Multi-dimensional wave propagation is computed by the directionally separated advection formulation, where one-dimensional advections for each axis are alternately computed as shown in Fig. 5.5, for a two-dimensional simulation for example. Lettig results of x-directional advections be values at an intermediate step, values at the next time step is obtained by advecting the values at the intermediate step in the y-direction. This technique requires additional advection calculations for spatial derivatives with respect to the direction perpendicular to the advection direction. For the x-directional advection in a two-dimensional field, for example, the CIP method simultaneously computes the following advection equations as well as Eqs. (5.14) and (5.16):

$$\partial_t \eta_{x\pm} \pm c\partial_x \eta_{x\pm} = 0\ , \quad \text{for} \quad \eta_{x\pm} = \partial_y p \pm Z\partial_y v_x, \tag{5.17}$$

$$\partial_t \mu_{x\pm} \pm c\partial_x \mu_{x\pm} = 0\ , \quad \text{for} \quad \mu_{x\pm} = \partial_x \partial_y p \pm Z\partial_x \partial_y v_x, \tag{5.18}$$

where $\eta_{x\pm}$ and $\mu_{x\pm}$ are derivatives of $f_{x\pm}$ and $g_{x\pm}$ with respect to the y-direction, respectively. A kind of the CIP method, called the "type-C" CIP method, employs the third-order Hermite interpolation to obtain the perpendicular derivatives $\eta_{x\pm}$ and $\mu_{x\pm}$ at the advection sources [9, 11, 12]. Additionally, in a three-dimensional simulation, advection equations for the derivatives of $f_{x\pm}$, $g_{x\pm}$, $\eta_{x\pm}$, and $\mu_{x\pm}$ with respect to the z-direction have to be taken into calculations. We can calculate the y-directional (and the z-directional in three-dimensional simulations) advections in a similar procedure to that for the x-direction described above.

Although the type-C CIP method employing the Hermite interpolation for the perpendicular derivatives has high accuracy, it is computationally expensive. A simplified scheme, called the "type-M" CIP method, has been proposed [8, 10, 12]. The type-M CIP method calculates advection of the perpendicular derivatives only for Eq. (5.17) using the first-order Lagrange interpolation, i.e., the upwind scheme. Omitting calculations of Eq. (5.18), this scheme saves the computer memory and computational time, but has lower accuracy than that the type-C CIP method has.

5.2.2 Implementation

The followings explain an implementation procedure of the type-C CIP method for a two-dimensional sound field simulation. We assume here that the sound field is discretized in the x- and the y-direction by the same grid size of Δl and i and j denote the grid point number in each direction as shown in Fig. 5.3.

Firstly, we carry out advection calculations in the $\pm x$-direction. At a time step n, the characteristic curves and their derivatives at a grid point (i,j) are derived from the physical values and their derivatives at the same point:

$$f^n_{x\pm}(i,j) = p^n(i,j) \pm Zv^n_x(i,j), \tag{5.19}$$

$$g^n_{x\pm}(i,j) = \partial_x p^n(i,j) \pm Z\partial_x v^n_x(i,j), \tag{5.20}$$

$$\eta^n_{x\pm}(i,j) = \partial_y p^n(i,j) \pm Z\partial_y v^n_x(i,j), \tag{5.21}$$

$$\mu^n_{x\pm}(i,j) = \partial_x\partial_y p^n(i,j) \pm Z\partial_x\partial_y v^n_x(i,j). \tag{5.22}$$

The third-order Hermite interpolation simultaneously calculates $f_{x\pm}$ and $g_{x\pm}$ at an intermediate step, which are values at the advection sources at the time step n:

$$f^*_{x\pm}(i,j) = a_\pm\xi^3_\pm + b_\pm\xi^2_\pm + g^n_{x\pm}(i,j)\xi_\pm + f^n_{x\pm}(i,j), \tag{5.23}$$

$$g^*_{x\pm}(i,j) = 3a_\pm\xi^2_\pm + 2b_\pm\xi_\pm + g^n_{x\pm}(i,j), \tag{5.24}$$

where superscript $*$ represents the intermediate step and

$$a_\pm = \frac{g^n_{x\pm}(i,j) + g^n_{x\pm}(i\mp 1,j)}{(\mp\Delta l)^2} + \frac{2\left(f^n_{x\pm}(i,j) - f^n_{x\pm}(i\mp 1,j)\right)}{(\mp\Delta l)^3}, \tag{5.25}$$

$$b_\pm = \frac{3\left(f^n_{x\pm}(i\mp 1,j) - f^n_{x\pm}(i,j)\right)}{(\mp\Delta l)^2} - \frac{2g^n_{x\pm}(i,j) + g^n_{x\pm}(i\mp 1,j)}{\mp\Delta l}, \tag{5.26}$$

$$\xi_\pm = \mp c\Delta t. \tag{5.27}$$

Equations (5.23)–(5.27), which are polynomial forms of the third-oder Hermite interpolation, can be transformed into simple sum-of-product forms for efficient computation:

$$\begin{aligned} f^*_{x\pm}(i,j) = {} & C_{1\pm} f^n_{x\pm}(i \mp 1, j) + C_{2\pm} f^n_{x\pm}(i,j) \\ & + C_{3\pm} g^n_{x\pm}(i \mp 1, j) + C_{4\pm} g^n_{x\pm}(i,j), \end{aligned} \tag{5.28}$$

$$\begin{aligned} g^*_{x\pm}(i,j) = {} & C'_{1\pm} f^n_{x\pm}(i \mp 1, j) + C'_{2\pm} f^n_{x\pm}(i,j) \\ & + C'_{3\pm} g^n_{x\pm}(i \mp 1, j) + C'_{4\pm} g^n_{x\pm}(i,j). \end{aligned} \tag{5.29}$$

The coefficients will be obtained before starting computations:

$$C_{1\pm} = -2\chi^3 + 3\chi^2, \tag{5.30}$$

$$C_{2\pm} = 2\chi^3 - 3\chi^2 + 1, \tag{5.31}$$

$$C_{3\pm} = \xi_\pm(\chi^2 - \chi), \tag{5.32}$$

$$C_{4\pm} = \xi_\pm(\chi^2 - 2\chi + 1), \tag{5.33}$$

$$C'_{1\pm} = 6(-\chi^3 + \chi^2)/\xi_\pm, \tag{5.34}$$

$$C'_{2\pm} = 6(\chi^3 - \chi^2)/\xi_\pm, \tag{5.35}$$

$$C'_{3\pm} = 3\chi^2 - 2\chi, \tag{5.36}$$

$$C'_{4\pm} = 3\chi^2 - 4\chi + 1, \tag{5.37}$$

where $\chi = c\Delta t/\Delta l$. Advection of the perpendicular derivatives $\eta_{x\pm}$ and $\mu_{x\pm}$ are also calculated in the same way:

$$\begin{aligned} \eta^*_{x\pm}(i,j) = {} & C_{1\pm} \eta^n_{x\pm}(i \mp 1, j) + C_{2\pm} \eta^n_{x\pm}(i,j) \\ & + C_{3\pm} \mu^n_{x\pm}(i \mp 1, j) + C_{4\pm} \mu^n_{x\pm}(i,j), \end{aligned} \tag{5.38}$$

$$\begin{aligned} \mu^*_{x\pm}(i,j) = {} & C'_{1\pm} \eta^n_{x\pm}(i \mp 1, j) + C'_{2\pm} \eta^n_{x\pm}(i,j) \\ & + C'_{3\pm} \mu^n_{x\pm}(i \mp 1, j) + C'_{4\pm} \mu^n_{x\pm}(i,j). \end{aligned} \tag{5.39}$$

Results of advection calculations in the x-direction are restored to the physical values and their derivatives at the intermediate step:

$$p^*(i,j) = \left(f^*_{x+}(i,j) + f^*_{x-}(i,j)\right)/2, \tag{5.40}$$

$$v^*_x(i,j) = \left(f^*_{x+}(i,j) - f^*_{x-}(i,j)\right)/2Z, \tag{5.41}$$

$$\partial_x p^*(i,j) = \left(g^*_{x+}(i,j) + g^*_{x-}(i,j)\right)/2, \tag{5.42}$$

$$\partial_x v^*_x(i,j) = \left(g^*_{x+}(i,j) - g^*_{x-}(i,j)\right)/2Z, \tag{5.43}$$

$$\partial_y p^*(i,j) = \left(\eta^*_{x+}(i,j) + \eta^*_{x-}(i,j)\right)/2, \tag{5.44}$$

$$\partial_y v^*_x(i,j) = \left(\eta^*_{x+}(i,j) - \eta^*_{x-}(i,j)\right)/2Z, \tag{5.45}$$

$$\partial_x \partial_y p^*(i,j) = \left(\mu^*_{x+}(i,j) + \mu^*_{x-}(i,j)\right)/2, \tag{5.46}$$

$$\partial_x \partial_y v^*_x(i,j) = \left(\mu^*_{x+}(i,j) - \mu^*_{x-}(i,j)\right)/2Z. \tag{5.47}$$

The particle velocity in the x-direction and their derivatives are directly assigned to those at the next time step: $v_x^{n+1}(i,j) = v_x^*(i,j)$, for example.

Secondly, advection calculations in the $\pm y$-direction are carried out in a similar procedure described above. The y-directional characteristic curves and their derivatives are derived from the physical values and their derivatives at the intermediate step:

$$f_{y\pm}^*(i,j) = p^*(i,j) \pm Zv_y^*(i,j), \tag{5.48}$$
$$g_{y\pm}^*(i,j) = \partial_y p^*(i,j) \pm Z\partial_y v_y^*(i,j), \tag{5.49}$$
$$\eta_{y\pm}^*(i,j) = \partial_x p^*(i,j) \pm Z\partial_x v_y^*(i,j), \tag{5.50}$$
$$\mu_{y\pm}^*(i,j) = \partial_x\partial_y p^*(i,j) \pm Z\partial_x\partial_y v_y^*(i,j). \tag{5.51}$$

The third-order Hermite interpolation gives the values at the next time step $n+1$:

$$\begin{aligned} f_{y\pm}^{n+1}(i,j) = {} & C_{1\pm} f_{y\pm}^*(i,j\mp 1) + C_{2\pm} f_{y\pm}^*(i,j) \\ & + C_{3\pm} g_{y\pm}^*(i,j\mp 1) + C_{4\pm} g_{y\pm}^*(i,j), \end{aligned} \tag{5.52}$$
$$\begin{aligned} g_{y\pm}^{n+1}(i,j) = {} & C'_{1\pm} f_{y\pm}^*(i,j\mp 1) + C'_{2\pm} f_{y\pm}^*(i,j) \\ & + C'_{3\pm} g_{y\pm}^*(i,j\mp 1) + C'_{4\pm} g_{y\pm}^*(i,j), \end{aligned} \tag{5.53}$$
$$\begin{aligned} \eta_{y\pm}^{n+1}(i,j) = {} & C_{1\pm} \eta_{y\pm}^*(i,j\mp 1) + C_{2\pm} \eta_{y\pm}^*(i,j) \\ & + C_{3\pm} \mu_{y\pm}^*(i,j\mp 1) + C_{4\pm} \mu_{y\pm}^*(i,j), \end{aligned} \tag{5.54}$$
$$\begin{aligned} \mu_{y\pm}^{n+1}(i,j) = {} & C'_{1\pm} \eta_{y\pm}^*(i,j\mp 1) + C'_{2\pm} \eta_{y\pm}^*(i,j) \\ & + C'_{3\pm} \mu_{y\pm}^*(i,j\mp 1) + C'_{4\pm} \mu_{y\pm}^*(i,j). \end{aligned} \tag{5.55}$$

Results of advection calculations in the y-direction are restored to the physical values and their derivatives at the time step $n+1$:

$$p^{n+1}(i,j) = \left(f_{y+}^{n+1}(i,j) + f_{y-}^{n+1}(i,j)\right)/2, \tag{5.56}$$
$$v_y^{n+1}(i,j) = \left(f_{y+}^{n+1}(i,j) - f_{y-}^{n+1}(i,j)\right)/2Z, \tag{5.57}$$
$$\partial_y p^{n+1}(i,j) = \left(g_{y+}^{n+1}(i,j) + g_{y-}^{n+1}(i,j)\right)/2, \tag{5.58}$$
$$\partial_y v_y^{n+1}(i,j) = \left(g_{y+}^{n+1}(i,j) - g_{y-}^{n+1}(i,j)\right)/2Z, \tag{5.59}$$
$$\partial_x p^{n+1}(i,j) = \left(\eta_{y+}^{n+1}(i,j) + \eta_{y-}^{n+1}(i,j)\right)/2, \tag{5.60}$$
$$\partial_x v_y^{n+1}(i,j) = \left(\eta_{y+}^{n+1}(i,j) - \eta_{y-}^{n+1}(i,j)\right)/2Z, \tag{5.61}$$
$$\partial_x\partial_y p^{n+1}(i,j) = \left(\mu_{y+}^{n+1}(i,j) + \mu_{y-}^{n+1}(i,j)\right)/2, \tag{5.62}$$
$$\partial_x\partial_y v_y^{n+1}(i,j) = \left(\mu_{y+}^{n+1}(i,j) - \mu_{y-}^{n+1}(i,j)\right)/2Z. \tag{5.63}$$

In the type-M CIP method, the third-oder Hermite interpolation of the perpendicular derivatives, Eqs. (5.38) and (5.54), are replaced with the first-order Lagrange interpolation as follows:

$$\eta^*_{x\pm}(i,j) = C^L_{1\pm}\eta^n_{x\pm}(i\mp 1,j) + C^L_{2\pm}\eta^n_{x\pm}(i,j) \tag{5.64}$$

$$\eta^{n+1}_{y\pm}(i,j) = C^L_{1\pm}\eta^*_{y\pm}(i,j\mp 1) + C^L_{2\pm}\eta^*_{y\pm}(i,j), \tag{5.65}$$

where

$$C^L_{1\pm} = \chi, \tag{5.66}$$

$$C^L_{2\pm} = 1-\chi. \tag{5.67}$$

The above equations correspond with the upwind scheme. Advection calculations for Eqs. (5.39) and (5.55) are omitted and the second derivatives of the physical values, $\partial_x\partial_y p$, $\partial_x\partial_y v_x$, and $\partial_x\partial_y v_y$, are not defined at the grid points, resulting in reduction of required computer memory and computational time.

5.2.3 Boundary Conditions

In the CIP method, boundary conditions at the interface between two media are given using the reflection coefficient [16]. When f_{x+} and g_{x+} are incident on the boundary at x_b, for example, the boundary conditions are represented as

$$f_{x-}(x_b) = \Gamma f_{x+}(x_b), \tag{5.68}$$

$$g_{x-}(x_b) = \Gamma' g_{x+}(x_b), \tag{5.69}$$

where Γ and Γ' denote the reflection coefficient of p and v_x and that of the spatial derivatives $\partial_x p$ and $\partial_x v_x$, respectively.

Here, we designate a medium in the $-x$ side of the boundary as medium-1 and that in the opposite side as medium-2. The reflection coefficient Γ is given as

$$\Gamma = \frac{Z_2 - Z_1}{Z_2 + Z_1}, \tag{5.70}$$

where Z_1 and Z_2 are the characteristic impedance of each medium (hereafter, values with a subscript 1 and 2 denote those in each medium). Eq. (5.70) is derived from Dirichlet conditions of the sound pressure and the particle velocity on the boundary:

$$p_1 = p_2, \tag{5.71}$$

$$v_{x1} = v_{x2}. \tag{5.72}$$

Meanwhile, Neumann conditions on the boundary are

$$\frac{1}{\rho_1}\partial_x p_1 = \frac{1}{\rho_2}\partial_x p_2, \tag{5.73}$$

$$\rho_1 c_1^2 \partial_x v_{x1} = \rho_2 c_2^2 \partial_x v_{x2}, \tag{5.74}$$

where ρ_1 and ρ_2 are the density of each medium. These conditions lead to the following relation:

$$\Gamma' = -\frac{Z_2 - Z_1}{Z_2 + Z_1} = -\Gamma. \tag{5.75}$$

We can explicitly give the reflection coefficients Γ and Γ' as constants as long as they satisfy Eq. (5.75). For example, $\Gamma = 1$ and $\Gamma' = -1$ are set to a perfectly reflective surface.

In multi-dimensional simulations, boundary conditions for the perpendicular derivatives are given in similar forms to Eqs. (5.68) and (5.69). In a two-dimensional simulation, for example, the boundary conditions for the perpendicular derivatives are represented as

$$\eta_{x-}(x_b) = \Gamma \eta_{x+}(x_b), \tag{5.76}$$

$$\mu_{x-}(x_b) = \Gamma' \mu_{x+}(x_b). \tag{5.77}$$

The equations above are derived from the differentiation of Eqs. (5.71)–(5.74).

5.2.4 Perfectly Matched Layer

The CIP method computes a sound field formed into a finite discrete domain. When we apply the CIP method to a simulation of an infinite sound filed, such as an outdoor sound field, an absorption treatment is required to suppress reflection waves from outer boundaries of the finite simulation domain as the FDTD method requires. The CIP method computes wave propagation dividing it into forward and backward components. Consequently, the CIP method naturally provides an automatic-absorbing boundary condition (A-ABC) on the outer boundaries letting incoming components be zero. However, the A-ABC is similar to the Mur's ABC of the first kind in the FDTD method [17]. In multi-dimensional simulations, the A-ABC has insufficient absorption performance for oblique-incident waves on the outer boundaries although it is effective for a normal-incident wave.

The perfectly matched layer-absorbing boundary condition (PML-ABC) [18] is often employed in multi-dimensional FDTD simulations to achieve high absorption performance on the outer boundaries for oblique-incident waves as well as a normal-incident wave. The followings explain formulation to implement the PML-ABC in multi-dimensional sound field simulations using the type-C CIP method [19, 20].

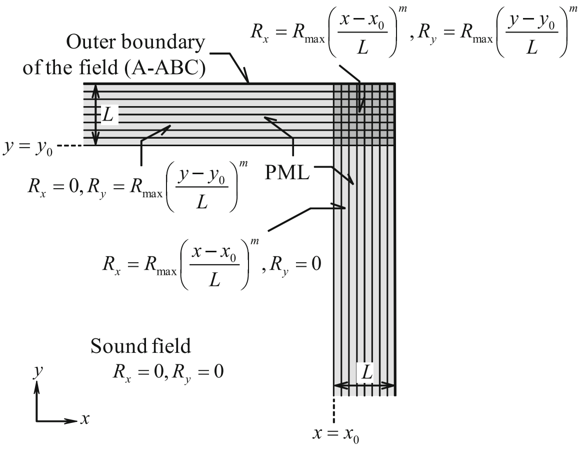

$R_x = R_{\max}\left(\frac{x-x_0}{L}\right)^m, R_y = R_{\max}\left(\frac{y-y_0}{L}\right)^m$
Outer boundary of the field (A-ABC)
L
$y = y_0$
$R_x = 0, R_y = R_{\max}\left(\frac{y-y_0}{L}\right)^m$
PML
$R_x = R_{\max}\left(\frac{x-x_0}{L}\right)^m, R_y = 0$
Sound field
$R_x = 0, R_y = 0$
y
x
L
$x = x_0$

Fig. 5.6 Perfectly matched layer surrounding a sound field

Although, for simplicity, two-dimensional formulation is shown below, it can be simply extended to three-dimensional simulations.

Figure 5.6 schematically illustrates the PML, which surrounds a calculated sound field to suppress reflection waves from the outer boundaries. In the PML region, nonzero attenuation parameter reduces sound waves and results in suppression of incoming waves. Discontinuous change of the attenuation parameter causes reflection. Therefore, increasing attenuation parameter is given as a function of the distance from the PML surface, that is the interface between the sound field and the PML:

$$R = R_{\max}\left(\frac{d}{L}\right)^m, \tag{5.78}$$

where $R_{\max}$ is the maximum value of the attenuation parameter, L is the thickness of the PML, and m decides distribution of the attenuation parameter. d denotes the distance from the PML surface and can be expressed by coordinates as shown in Fig. 5.6. It should be noted that nonzero attenuation parameter is given only in the direction where the PML intends to suppress reflection waves; $R_x = R$, $R_y = 0$ in the PML for the x-direction, $R_x = 0$, $R_y = R$ in that for the y-direction, and $R_x = R$, $R_y = R$ in that at a corner of calculated sound field.

In the PML region with nonzero attenuation parameter R_x, the governing equations for acoustic fields corresponding to Eq. (5.13) are [21]

$$\partial_t p + cZ\partial_x v_x = -\frac{R_x}{\rho}p, \quad Z\partial_t v_x + c\partial_x p = -\frac{R_x}{\rho}Zv_x. \tag{5.79}$$

Addition and subtraction of them lead to the following equations letting $r_x = R_x/\rho$:

$$\partial_t f_{x\pm} \pm c\partial_x f_{x\pm} = -r_x f_{x\pm}. \tag{5.80}$$

Although Eq. (5.80) has a non-advection term, that is a dissipation term, in the right-hand side, it can be computed by separating into advection and non-advection phases [8]:

$$\partial_t f_{x\pm} \pm c\partial_x f_{x\pm} = 0, \tag{5.81}$$

$$\partial_t f_{x\pm} = -r_x f_{x\pm}. \tag{5.82}$$

Equation (5.81) is a normal advection equation. Therefore, it can be calculated by the procedure previously described for lossless fields. We let here results of the advection calculations at a time step n be $f_{x\pm}^{n(A)}$, $g_{x\pm}^{n(A)}$, $\eta_{x\pm}^{n(A)}$, and $\xi_{x\pm}^{n(A)}$.

In the non-advection phase, the analytical solution of Eq. (5.82) gives $f_{x\pm}$ at the next time step using $f_{x\pm}^{n(A)}$:

$$f_{x\pm}^{n+1} = f_{x\pm}^{n(A)} \exp(-r_x \Delta t). \tag{5.83}$$

The derivatives $g_{x\pm}$ at the next time step are also analytically obtained by the spatial differentiation of Eq. (5.83) with respect to x:

$$g_{x\pm}^{n+1} = \left\{ g_{x\pm}^{n(A)} - \Delta t(\partial_x r_x) f_{x\pm}^{n(A)} \right\} \exp(-r_x \Delta t), \tag{5.84}$$

where the spatial derivative of r_x is given by

$$\partial_x r_x = (\partial_x d) \frac{m R_{\max}}{\rho L} \left(\frac{d}{L} \right)^{m-1}. \tag{5.85}$$

Furthermore, the differentiations of Eqs. (5.83) and (5.84) with respect to y derive the solutions of the perpendicular derivatives $\eta_{x\pm}$ and $\mu_{x\pm}$:

$$\eta_{x\pm}^{n+1} = \eta_{x\pm}^{n(A)} \exp(-r_x \Delta t), \tag{5.86}$$

$$\mu_{x\pm}^{n+1} = \left\{ \mu_{x\pm}^{n(A)} - \Delta t(\partial_x r_x) \eta_{x\pm}^{n(A)} \right\} \exp(-r_x \Delta t). \tag{5.87}$$

As described above, the values at the next time step, that is the results of the non-advection phase, can be straightly calculated by using those previously obtained in the advection phase.

We can compute y-directional wave propagations in the PML region in a similar procedure to the above.

Figure 5.7 exhibits sound pressure distributions in a two-dimensional free field with and without the PML over the lower outer boundary. The PML consists of 64 layers and the outer boundaries including the lower one behind the PML employ the A-ABC. The illustrated area is a part of an adequately large calculated field, which

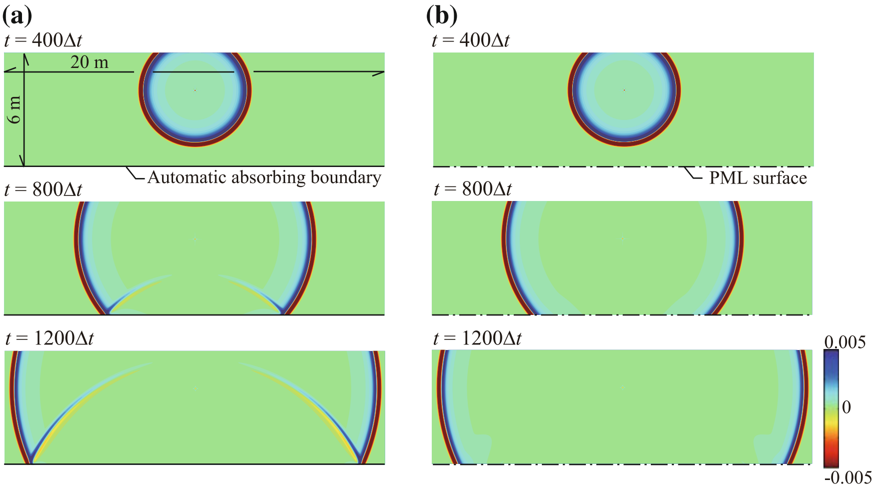


Fig. 5.7 Comparison of sound pressure distributions in the two-dimensional free field with and without the PML over the lower outer boundary. Calculation parameters are: $c = 340.5\,\text{m/s}$, $\Delta t = 0.25 \times 10^{-4}\,\text{s}$, $\Delta l = 0.025\,\text{m}$, $R_{\max} = 2.0\rho c$, $m = 2$, and $N = 64$. **a** Without PML. **b** With 64-layer PML

has dimensions of 65.6 × 27.2 m. Therefore, no reflection waves from the outer boundaries except the lower one appear in the illustrated area in a time range shown in Fig. 5.7. In the results without the PML, we can observe the reflection wave from the A-ABC and find that the amplitude of it increases as the incident angle increases. In contrast, in the results with the PML, the reflection wave almost disappears.

5.3 Finite-Volume Time-Domain Method

The FDTD approach has widely been accepted as a simple, fast, and proven method for numerical sound propagation prediction. However, the method requires an orthogonal structured grid system for a discretized representation of an acoustic field. The use of the orthogonal grid system is recognized as one of shortcomings of the FDTD method because it is generally not suitable for the representation of complex geometries. Meshing (preprocessing) and postprocessing complex geometries with orthogonal structured grids often require extensive human works.

The issue is addressed here through an introduction of a full finite-volume time-domain (FVTD) method meant as a replacement for the FDTD method. The main strength of the full FVTD approach is the capability to utilize unstructured grids, which contain arbitrary polyhedral cells. The strength opens possibility of using a vast variety of general-purpose pre- and post-processors designed for finite volume or finite element grids.

5.3.1 Finite-Volume Formulation

The wave propagation equation represented in velocity potential ϕ is written as follows.

$$\frac{\partial^2 \phi}{\partial t^2} = c_0^2 \nabla^2 \phi, \tag{5.88}$$

where t, c_0, ∇^2 are time, the propagation speed of the wave, and the Laplacian operator, respectively. Using ϕ, pressure p and particle velocity u are written as follows.

$$p = \rho \frac{\partial \phi}{\partial t}, \tag{5.89}$$

$$u = -\nabla \phi. \tag{5.90}$$

Equation (5.88) is discretized under an unstructured grid system as shown in Fig. 5.8 where the definition point of physical quantities are taken at the barycenter of each control volume (CV). For the left hand side of Eq. (5.88), by integrating over the CV with time-invariant volume V and applying central time-differential scheme, we get

$$\frac{\partial^2}{\partial t^2} \int_V \phi \, dV \approx \frac{\phi^{n+1} - 2\phi^n + \phi^{n-1}}{\Delta t^2} V,$$

where ϕ^{n-1}, ϕ^n, ϕ^{n+1} denote the values of ϕ at the $(n-1)$th, nth, $(n+1)$th steps of time step Δt. For the right hand side, by integrating Eq. (5.88) within a CV and applying divergence theorem, we get

$$\begin{aligned}\int_V c_0^2 \, \nabla^2 \phi dV &= c_0^2 \int_S dS \cdot \nabla \phi \\ &\approx c_0^2 \sum_{\mathrm{f}} S_{\mathrm{f}} \cdot (\nabla \phi)_{\mathrm{f}},\end{aligned} \tag{5.91}$$

where S_{f} denotes the face area vector of the f-th face that constitutes the polyhedral CV in question as follows:

$$S_{\mathrm{f}} = S_{\mathrm{f}} n_{\mathrm{f}}, \tag{5.92}$$

where S_{f} and n_{f} are the area and the unit outward normal vector of the face f, respectively.

If a vector connecting the centers of the CV P and its adjacent CV N, d_{PN}, is parallel to S_f, S_{f} $(\nabla \phi)_{\mathrm{f}}$ is written in terms of $\partial \phi / \partial n_{\mathrm{f}}$, the surface-normal gradient

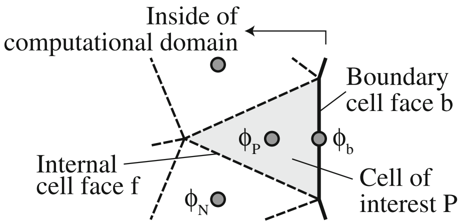


Fig. 5.8 Unstructured mesh system

of ϕ. Thus, the term within the summation in the rightmost hand side of Eq. (5.91) is discretized as follows.

$$\begin{aligned} S_\mathrm{f} \cdot (\nabla\phi)_\mathrm{f} &= S_\mathrm{f} \frac{\partial \phi}{\partial n_\mathrm{f}} \\ &\approx S_\mathrm{f} \frac{\phi_\mathrm{N} - \phi_\mathrm{P}}{|d_\mathrm{PN}|}. \end{aligned} \tag{5.93}$$

However, if d_{PN} is nonorthogonal to S_f, S_f has to be decomposed into its orthogonal part $\varDelta_\mathrm{f}$ and nonorthogonal part k_f.

$$S_\mathrm{f} \cdot (\nabla\phi)_\mathrm{f} = \varDelta_\mathrm{f} \cdot (\nabla\phi)_\mathrm{f} + k_\mathrm{f} \cdot (\nabla\phi)_\mathrm{f}.$$

The first term of the right hand side of the equation above, the orthogonal part, is discretized similarly to Eq. (5.93) as follows:

$$\varDelta_\mathrm{f} \cdot (\nabla\phi)_\mathrm{f} \approx |\varDelta_\mathrm{f}| \frac{\phi_\mathrm{N} - \phi_\mathrm{P}}{|d_\mathrm{PN}|}.$$

The nonorthogonal part, $(\nabla\phi)_\mathrm{f}$ in the second term, is given by interpolating the gradient of ϕ at the centers of CVs P and N.

$$(\nabla\phi)_\mathrm{f} = f_x (\nabla\phi)_\mathrm{P} + (1 - f_x)(\nabla\phi)_\mathrm{N}. \tag{5.94}$$

Here, the interpolation coefficient f_x and the gradient $(\nabla\phi)_\mathrm{P}$ are given as follows.

$$\begin{aligned} f_x &= \frac{\overline{\mathrm{fN}}}{|d_\mathrm{PN}|}, \\ (\nabla\phi)_\mathrm{P} &= \frac{1}{V}\int_S dS\,\phi \\ &\approx \frac{1}{V}\sum_\mathrm{f} S\phi_\mathrm{f}, \end{aligned}$$

where ϕ_f is a face-interpolated value of ϕ at the center of CVs.

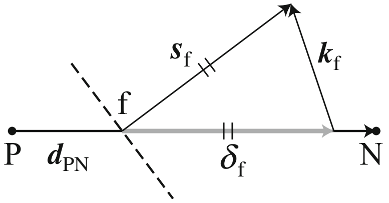

$\boldsymbol{s}_\mathrm{f}$
$\boldsymbol{k}_\mathrm{f}$
f
P
$\boldsymbol{d}_\mathrm{PN}$
$\boldsymbol{\delta}_\mathrm{f}$
N

Fig. 5.9 Nonorthogonal mesh correction vectors

The orthogonal and nonorthogonal component vectors Δ_f and k_f can be calculated arbitrarily [22]. An orthogonal correction as follows is applied here.

$$\Delta_\mathrm{f} = \frac{d_\mathrm{PN}}{|d_\mathrm{PN}|} S_\mathrm{f}, \tag{5.95}$$

$$k_\mathrm{f} = S_\mathrm{f} - \Delta_\mathrm{f}. \tag{5.96}$$

The vectors are schematically drawn in Fig. 5.9.

5.3.2 Rigid Boundary Conditions

On acoustically rigid boundaries b, the normal component of particle velocity u_b is fixed to zero.

$$u_\mathrm{b} = n_\mathrm{b} \cdot u_\mathrm{b} = 0. \tag{5.97}$$

Substituting the relationship above to Eqs. (5.90) and (5.92) leads to the equation below, which represents the surface normal gradient of ϕ being zero.

$$S_\mathrm{b} \cdot (\nabla\phi)_\mathrm{b} = 0.$$

5.3.3 Computational Setup

In order to validate the proposed FVTD technique under unstructured meshes, a comparative test using a sound propagation problem in a closed cube of 1 m × 1 m × 1 m, one of the AIJ-BPCA (Benchmark Platform on Computational Methods for Architectural and Environmental Acoustics) [23] problems, was carried out. The detail of the tested cases are shown in Table 5.1.

Case 1 The problem was solved using a conventional FDTD code written in Fortran 77 employing a pressure-particle velocity leapfrog scheme. The case is meant to be the benchmark case to which the results obtained by the proposed technique is compared for validation. Each edge of the cube was divided to

Table 5.1 Computational setup

Case	1	2	3	4
Approach	FDTD	FVTD		
Type of mesh	–	Hexahedral	Unstructured tetrahedral	
Number of cells/CVs	$81^3 = 531\,441$		531 333	
Δx [m]	0.0123		–	
l_c [m]	–		0.025 (40 elements per edge)	
Δt [ms]	0.02		0.0049	
c_0 [m/s]	343.7			
Courant number	0.96		0.99 (max)	
Nonorth. correction	–	Uncorrected		Orthogonal
Initial condition	A single wave of offset cosine (Eqs. (5.98), (5.99))			

81 subedges to create a mesh of cell width $\Delta x = 0.0123$ m and the number of cells 531,441 (Fig. 5.10b). The time step Δt and the Courant number C_o were set to 0.02 ms and 0.96, respectively.

Case 2 The problem was solved with the proposed technique under a hexahedral orthogonal mesh and setup both identical to the ones for Case 1.

Case 3 The problem was solved with the proposed technique under a nonuniform tetrahedral unstructured mesh automatically generated by an open-source mesher, Gmsh [24]. The characteristic length l_c (the length with which each edge of the cube is divided) is set to 0.025 m, to make a mesh with the number of CVs 531,333 (roughly the same as Cases 1 and 2). The ratio of maximum and minimum CV edge lengths of the generated mesh was 6.32. The time step Δt was set to 0.0049 ms to keep the maximum Courant number to 0.99. In this case no nonorthogonal techniques was applied.

Case 4 The setups are same as Case 3, except that the orthogonal correction technique was applied.

For all cases, the initial values of ϕ were set to represent the pressure and particle velocity conditions of

$$p^{-1/2}(r) = \begin{cases} \dfrac{\cos 8\pi r + 1}{2} & (r < 0.125) \\ 0 & (\text{otherwise}) \end{cases} \text{[Pa]}, \tag{5.98}$$

$$u^0(r) = 0, \tag{5.99}$$

where r [m] is the distance from the center of the cube. All cases were run up to $t = 0.04$s.

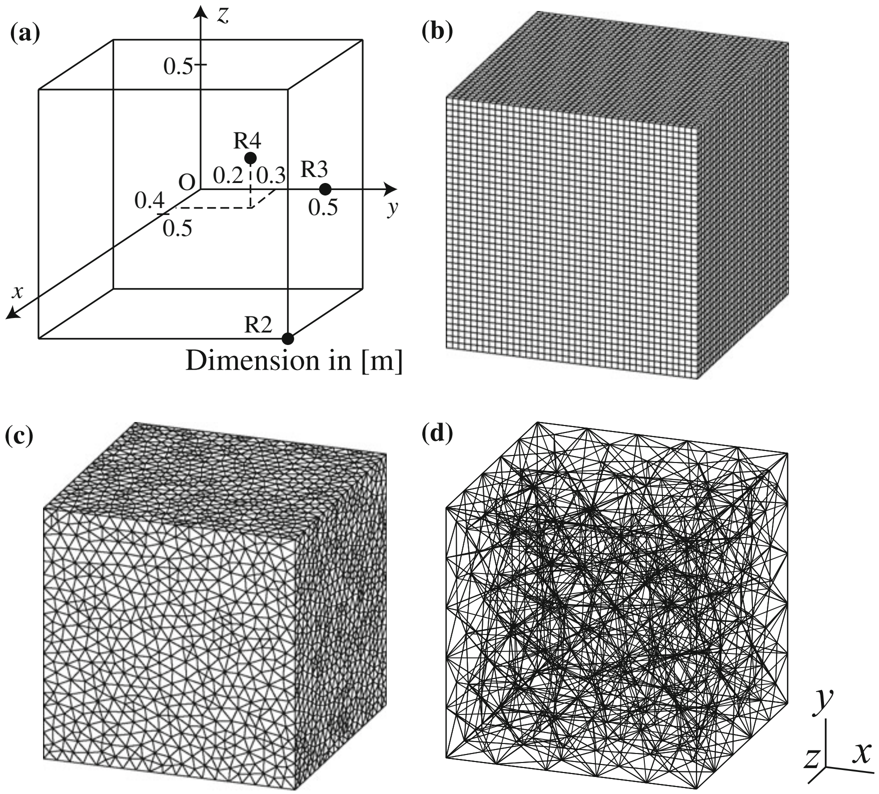


Fig. 5.10 **a** Problem geometry of the benchmark problem AIJ-BPCA B0-1T, **b** Surface mesh for Cases 1 and 2 (coarsened by factor of 2 for visibility), **c** Surface mesh for Cases 3 and 4 (coarsened by factor of 2) and **d** Surface and internal mesh (coarsened by factor of 8) for Cases 3 and 4

5.3.4 Results and Discussion

The transient sound pressure waveforms at the receiving point R2 shown in Fig. 5.10a are plotted in Fig. 5.11, using the result of Case 1 as the benchmark case for comparison with other cases. From Fig. 5.11a, one can see that the results of FDTD and FVTD techniques agree so precisely that they can be regarded as virtually identical results. From the results one can verify the proposed FVTD technique has the same accuracy as the conventional FDTD under identical geometry, mesh, and computational setups.

On the other hand, from the comparison of Cases 1 and 3 in Fig. 5.11b, the waveform obtained by the FVTD technique under the tetrahedral mesh is phasing forward in about 1.5 %. Also, the overall waveform is gradually dispersing over time. In addition, the results of Case 4 is shown in Fig. 5.11c. Despite the employment of the

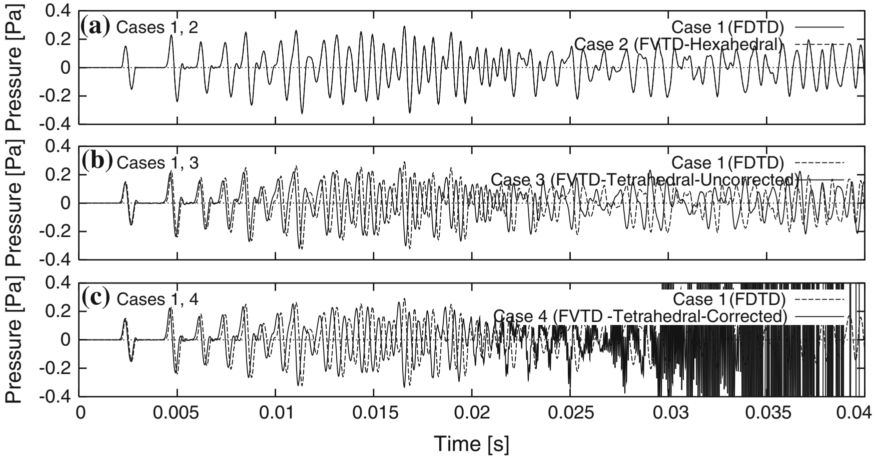


Fig. 5.11 Transient sound pressure waveforms at the receiving point R2

Table 5.2 Processor and memory usages

Case	1	2	3	4
Processor [s]	28.0	343	865	2013
Per time step [s]	0.0140	0.172	0.106	0.247
Memory [MB]	18	301	260	260

correction technique, the drift of the phase did not improve. Even worse, waveforms started to oscillate and diverged eventually.

From the results one can conclude that while one can expect identical results between FDTD and FVTD techniques under identical setups, there remains works for the FVTD technique in reducing the phase error coming from nonorthogonalities of unstructured grids.

5.3.5 Computational Loads

To compare the proposed technique with the conventional FDTD from the standpoint of computational loads, processor times and memory usages were instrumented for Cases 1–4, as shown in Table 5.2. The instrumentations were carried out on an Opteron 2.4 GHz 64-bit Linux platform. The FVTD computations turned out to require more than ten times of processor and memory usages. Hence it should be noted that, from the computational load of view, the proposed FVTD technique is not meant to completely replace FDTD especially in large cases, but should rather be used for small to medium cases where rapid preprocessing (case setup) and post-processing have of particular importance.

Although Cases 2 and 3 have roughly the same number of cells, one may notice that Case 3 requires smaller amount of computational time per time step. This is because the computational load required in calculating the Laplacian term is determined mostly by the number of faces per CV, as shown in Eq. (5.91). It is also shown that, from Cases 3 and 4, applying nonorthogonal correction technique more than doubles the processor usage.

References

1. D. Heimann, R. Blumrich, Time-domain simulations of sound propagation through screen-induced turbulence. Appl. Acoust. **65**(6), 561–582 (2004)
2. V.E. Ostashev, D.K. Wilson, L. Liu, D.F. Aldridge, N.P. Symons, D. Marlin, Equations for finite-difference, time-domain simulation of sound propagation in moving inhomogeneous media and numerical implementation. J. Acoust. Soc. Am. **117**(2), 503–517 (2005)
3. T. van Renterghem, D. Botteldooren, Prediction-step staggered-in-time FDTD: An efficient numerical scheme to solve the linearized equations of fluid dynamics in outdoor sound propagation. Appl. Acoust. **68**(2), 201–216 (2007)
4. S.K. Lele, Compact finite difference schemes with spectral-like resolution. J. Comput. Phys. **103**, 16–42 (1992)
5. C. Arakawa, in *Computational Fluid Dynamics for Engineering* (University of Tokyo Press, Tokyo, 1994) (in Japanese)
6. J.C. Hardin, J.R. Ristorcelli, C.K.W. Tam, in *ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics (CAA). NASA Conference Publications*, pp. 3300 (1995)
7. T. Yabe, F. Xiao, T. Utsumi, Constrained interpolation profile method for multiphase analysis. J. Comput. Phys. **169**, 556–593 (2001)
8. T. Yabe, T. Utsumi, Y. Ogata, *The Constrained Interpolation Profile Method* (Morikita, Tokyo, 2003). (in Japanese)
9. T. Aoki, Multi-dimensional advection of CIP (cubic interpolated propagation) scheme. Comput. Fluid Dynamics J. **4**, 279–291 (1995)
10. K. Okubo, S. Oh, T. Tsuchiya, N. Takeuchi, The examination on the three-dimensional acoustic field analysis using the CIP method. IEICE Technical Report, US2006-52:25–30, (2006) (in Japanese)
11. S. Oh, K. Okubo, T. Tsuchiya, N. Takeuchi, Analysis of acoustic field using the type-C CIP method. IEICE Technical Report, US2006-98:17–22 (2006) (in Japanese)
12. M. Konno, K. Okubo, T. Tsuchiya, N. Tagawa, Performance of various types of constrained interpolation profile method for two-dimensional numerical acoustic simulation. Jpn. J. Appl. Phys. **47**, 3962–3963 (2008)
13. Y. Tachioka, Y. Yasuda, T. Sakuma, Application of the constrained interpolation profile method to room acoustic problems: Examination of boundary modeling and spatial/time discretization. Acoust. Sci. Tech. **33**, 22–32 (2012)
14. T. Ishizuka, K. Okubo, Y. Ara, Variable-grid technique for sound field analysis using the constrained interpolation profile method. Acoust. Sci. Tech. **33**, 387–390 (2012)
15. G. Szego, in *Orthogonal Polynomials* (American Mathematical Society, Providence, 1939)
16. K. Okubo, S. Oh, T. Tsuchiya, N. Takeuchi, Consideration of the boundary condition between two media in acoustic field analysis using the constrained interpolation profile (CIP) method. IEICE Trans. Fundam. **E90-A**, 2000–2005 (2007) (in Japanese)
17. G. Mur, Absorbing boundary conditions for the finite difference approximation of the time domain electromagnetic-field equation. IEEE Trans. Antennas Propag. **45**, 377–382 (1981)
18. J.P. Berenger, A perfectly matched layer for the absorption of electromagnetic waves. J. Comput. Phys. **114**, 185–200 (1994)

19. Y. Ando, M. Hayakawa, Implementation of the perfectly matched layer to the CIP method. IEICE Trans. Electron. **E98-C**, 645–648 (2006)
20. T. Ishizuka, K. Okubo, Formulation and examination of the perfectly matched layer for sound field analyses using the constrained interpolation profile method. Acoust. Sci. Tech. **34**, 378–381 (2013)
21. Q. Qui, T.L. Geers, Evaluation of the perfectly matched layer for computational acoustics. J. Comput. Phys. **139**, 166–183 (1998)
22. H. Jasak, Error analysis and estimation for the finite volume method with applications to fluid flows. PhD thesis, Imperial College, June 1996
23. Architectural Institute of Japan, Benchmark platform on computational methods for architectural/environmental acoustics. http://gacoust.hwe.oita-u.ac.jp/AIJ-BPCA/
24. C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. Int. J. Numer. Methods Eng. **79**, 1309–1331 (2009)