Lecture 37

Finite Difference Method, Yee Algorithm

In this lecture, we are going to introduce one of the simplest methods to solve Maxwell's equations numerically. This is the finite-difference time-domain method. Because of its simplicity, and that a simple Maxwell solver can be coded in one afternoon, almost every physics or electrical engineering laboratory has a home-grown version of the finite-difference time-domain solver. This method is the epitome of theat "simplicity rules."[1] Professor Hermann Haus at MIT used to say: find the simplest method to do things. Complicated methods will be forgotten, but the simplest method will prevail. This is also reminiscent of Einstein's saying, "Everything should be made as simple as possible, but no simpler!"

37.1 Finite-Difference Time-Domain Method

To obtain the transient (time-domain) solution of the wave equation for a more general, inhomogeneous medium, a numerical method has to be used. The finite-difference time-domain (FDTD) method, a numerical method, is particularly suitable for solving transient problems. Moreover, it is quite versatile, and given the present computer technology, it has been used with great success in solving many practical problems. This method is based on a simple Yee algorithm [224] and has been vastly popularized by Taflove [225, 226].

In the finite-difference method, continuous space-time is replaced with a discrete space-time. Then, in the discrete space-time, partial differential equations are replaced with finite difference equations. These finite difference equations are readily implemented on a digital computer. Furthermore, an iterative or time-stepping scheme can be implemented without having to solve large matrices, resulting in a great savings in computer time. Moreover, the matrix for the system of equations is never generated making this a matrix-free method. There is no need for matrix management as one writes this numerical solver. More recently,

[1] "rule" is used as a verb.

the development of parallel processor architectures in computers has also further enhanced the efficiency of the finite-difference scheme [227].

The finite-difference method is also described in numerous works (see, for example, Potter 1973 [228]; Taflove 1988 [225]; Ames 2014 [229]; Morton 2019 [230].

37.1.1 The Finite-Difference Approximation

Consider first a scalar wave equation of the form

$$\frac{1}{c^2(\mathbf{r})}\frac{\partial^2}{\partial t^2}\phi(\mathbf{r},t) = \mu(\mathbf{r})\nabla\cdot\mu^{-1}(\mathbf{r})\nabla\phi(\mathbf{r},t). \tag{37.1.1}$$

The above equation appears in scalar acoustic waves or a 2D electromagnetic waves in inhomogeneous media [34, 231].

To convert the above into a form that can be solved by a digital computer, first, one needs to find finite-difference approximations to the time derivatives. Then, the time derivative can be approximated in many ways. For example,

$$\begin{aligned}
\text{Forward difference:}\quad & \frac{\partial\phi(\mathbf{r},t)}{\partial t} \approx \frac{\phi(\mathbf{r},t+\Delta t)-\phi(\mathbf{r},t)}{\Delta t}, && (37.1.2)\\
\text{Backward difference:}\quad & \frac{\partial\phi(\mathbf{r},t)}{\partial t} \approx \frac{\phi(\mathbf{r},t)-\phi(\mathbf{r},t-\Delta t)}{\Delta t}, && (37.1.3)\\
\text{Central difference:}\quad & \frac{\partial\phi(\mathbf{r},t)}{\partial t} \approx \frac{\phi(\mathbf{r},t+\frac{\Delta t}{2})-\phi(\mathbf{r},t-\frac{\Delta t}{2})}{\Delta t}, && (37.1.4)
\end{aligned}$$

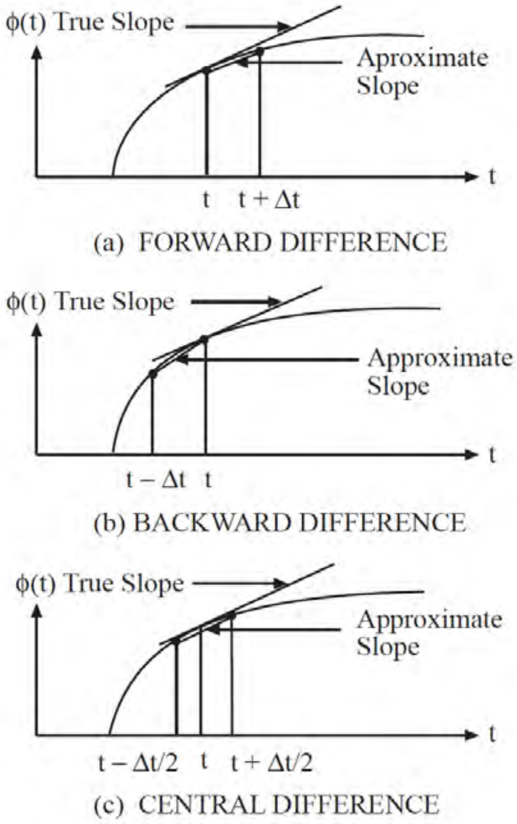

Figure 37.1: Different finite-difference approximations for the time derivative.

where Δt is a small number. Of the three methods of approximating the time derivative, the central-difference scheme is the best approximation, as is evident in Figure 37.1. The errors in the forward and backward differences are $O(\Delta t)$ (or first-order error) while the central-difference approximation has an error $O[(\Delta t)^2]$ (or second-order error). This can be easily illustrated by Taylor series expanding the right-hand sides of (37.1.2) to (37.1.4).

Consequently, using the central-difference formula twice, we arrive at

$$\frac{\partial^2}{\partial t^2}\phi(\mathbf{r},t) \approx \frac{\partial}{\partial t}\left[\frac{\phi(\mathbf{r},t+\frac{\Delta t}{2}) - \phi(\mathbf{r},t-\frac{\Delta t}{2})}{\Delta t}\right] \tag{37.1.5}$$

$$\approx \frac{\phi(\mathbf{r},t+\Delta t) - 2\phi(\mathbf{r},t) + \phi(\mathbf{r},t-\Delta t)}{(\Delta t)^2}. \tag{37.1.6}$$

Next, if the function $\phi(\mathbf{r},t)$ is indexed on discrete time steps on the t axis, such that for $t = l\Delta t$, then $\phi(\mathbf{r},t) = \phi(\mathbf{r},l\Delta t) = \phi^l(\mathbf{r})$, where l is an integer. Using this notation, Equation

(37.1.6) then becomes

$$\frac{\partial^2}{\partial t^2}\phi(\mathbf{r},t) \approx \frac{\phi^{l+1}(\mathbf{r}) - 2\phi^l(\mathbf{r}) + \phi^{l-1}(\mathbf{r})}{(\Delta t)^2}. \tag{37.1.7}$$

37.1.2 Time Stepping or Time Marching

With this notation and approximations, Equation (37.1.1) becomes a time-stepping (or time-marching) formula, namely,

$$\phi^{l+1}(\mathbf{r}) = c^2(\mathbf{r})(\Delta t)^2\mu(\mathbf{r})\nabla\cdot\mu^{-1}(\mathbf{r})\nabla\phi^l(\mathbf{r}) + 2\phi^l(\mathbf{r}) - \phi^{l-1}(\mathbf{r}). \tag{37.1.8}$$

Therefore, given the knowledge of $\phi(\mathbf{r},t)$ at $t = l\Delta t$ and $t = (l-1)\Delta t$ for all $\mathbf{r}$, one can deduce $\phi(\mathbf{r},t)$ at $t = (l+1)\Delta t$. In other words, given the initial values of $\phi(\mathbf{r},t)$ at, for example, $t = 0$ and $t = \Delta t$, $\phi(\mathbf{r},t)$ can be deduced for all subsequent times, provided that the time-stepping formula is stable.

At this point, the right-hand side of (37.1.8) involves the space derivatives. There exist a plethora of ways to approximate and calculate the right-hand side of (37.1.8) numerically. Here, we shall illustrate the use of the finite-difference method again to calculate the right-hand side of (37.1.8). Before proceeding further, note that the space derivatives on the right-hand side in cartesian coordinates are

$$\mu(\mathbf{r})\nabla\cdot\mu^{-1}(\mathbf{r})\nabla\phi(\mathbf{r}) = \mu\frac{\partial}{\partial x}\mu^{-1}\frac{\partial}{\partial x}\phi + \mu\frac{\partial}{\partial y}\mu^{-1}\frac{\partial}{\partial y}\phi + \mu\frac{\partial}{\partial z}\mu^{-1}\frac{\partial}{\partial z}\phi. \tag{37.1.9}$$

Then, one can approximate, using central differencing that

$$\frac{\partial}{\partial z}\phi(x,y,z) \approx \frac{1}{\Delta z}\left[\phi\left(x,y,z+\frac{\Delta z}{2}\right) - \phi\left(x,y,z-\frac{\Delta z}{2}\right)\right], \tag{37.1.10}$$

Consequently, using central differencing two times,

$$\begin{aligned}\frac{\partial}{\partial z}\mu^{-1}\frac{\partial}{\partial z}\phi(x,y,z) \approx \frac{1}{(\Delta z)^2}\bigg\{&\mu^{-1}\left(z+\frac{\Delta z}{2}\right)\phi(x,y,z+\Delta z)\\ &-\left[\mu^{-1}\left(z+\frac{\Delta z}{2}\right) + \mu^{-1}\left(z-\frac{\Delta z}{2}\right)\right]\phi(x,y,z)\\ &+\mu^{-1}\left(z-\frac{\Delta z}{2}\right)\phi(x,y,z-\Delta z)\bigg\}.\end{aligned} \tag{37.1.11}$$

Furthermore, after denoting $\phi(x,y,z) = \phi_{m,n,p}$, $\mu(x,y,z) = \mu_{m,n,p}$, on a discretized grid point at $x = m\Delta x$, $y = n\Delta y$, $z = p\Delta z$, we have $(x,y,z) = (m\Delta x, n\Delta y, p\Delta z)$, and then

$$\begin{aligned}\frac{\partial}{\partial z}\mu^{-1}\frac{\partial}{\partial z}\phi(x,y,z) \approx \frac{1}{(\Delta z)^2}\Big[&\mu^{-1}_{m,n,p+\frac{1}{2}}\phi_{m,n,p+1}\\ &-\left(\mu^{-1}_{m,n,p+\frac{1}{2}} + \mu^{-1}_{m,n,p-\frac{1}{2}}\right)\phi_{m,n,p} + \mu^{-1}_{m,n,p-\frac{1}{2}}\phi_{m,n,p-1}\Big].\end{aligned} \tag{37.1.12}$$

This cumbersome equation can be abbreviated if we define a central difference operator as[2]

$$\bar{\partial}_z \phi_m = \frac{1}{\Delta z}\left(\phi_{m+\frac{1}{2}} - \phi_{m-\frac{1}{2}}\right) \tag{37.1.13}$$

Then the right-hand side of the (37.1.12) can be written succinctly as

$$\frac{\partial}{\partial z}\mu^{-1}\frac{\partial}{\partial z}\phi(x,y,z) \approx \bar{\partial}_z \mu_{m,n,p} \bar{\partial}_z \phi_{m,n,p} \tag{37.1.14}$$

With similar approximations to the other terms in (37.1.9), Equation (37.1.8) becomes

$$\begin{aligned}\phi^{l+1}_{m,n,p} = (\Delta t)^2 c^2_{m,n,p}\mu_{m,n,p}\left[\bar{\partial}_x \mu_{m,n,p}\bar{\partial}_x + \bar{\partial}_y \mu_{m,n,p}\bar{\partial}_y + \bar{\partial}_z \mu_{m,n,p}\bar{\partial}_z\right]\phi_{m,n,p} \\ + 2\phi^l_{m,n,p} - \phi^{l-1}_{m,n,p}.\end{aligned} \tag{37.1.15}$$

The above can be readily implemented on a computer for time stepping. Notice however, that the use of central differencing results in the evaluation of medium property μ at half grid points. This is inconvenient, as the introduction of field values at half grid points increases computer memory requirements. Hence, it is customary to store the medium property at the integer grid points, and to approximate

$$\mu_{m+\frac{1}{2},n,p} \simeq \frac{1}{2}(\mu_{m+1,n,p} + \mu_{m,n,p}), \tag{37.1.16}$$

$$\mu_{m+\frac{1}{2},n,p} + \mu_{m-\frac{1}{2},n,p} \simeq 2\mu_{m,n,p}, \tag{37.1.17}$$

and so on. Moreover, if μ is a smooth function of space, it is easy to show that the errors in the above approximations are of second order by Taylor series expansions.

For a homogeneous medium, with $\Delta x = \Delta y = \Delta z = \Delta s$, (37.1.15) written explicitly becomes

$$\begin{aligned}\phi^{l+1}_{m,n,p} = \left(\frac{\Delta t}{\Delta s}\right)^2 c^2 \big[\phi^l_{m+1,n,p} + \phi^l_{m-1,n,p} + \phi^l_{m,n+1,p} + \phi^l_{m,n-1,p} + \phi^l_{m,n,p+1} \\ + \phi^l_{m,n,p-1} - 6\phi^l_{m,n,p}\big] + 2\phi^l_{m,n,p} - \phi^{l-1}_{m,n,p}.\end{aligned} \tag{37.1.18}$$

Notice then that with the central-difference approximation, the value of $\phi^{l+1}_{m,n,p}$ is dependent only on $\phi^l_{m,n,p}$, and its nearest neighbors, $\phi^l_{m\pm1,n,p}$, $\phi^l_{m,n\pm1,p}$, $\phi^l_{m,n,p\pm1}$, and $\phi^{l-1}_{m,n,p}$, its value at the previous time step. Moreover, in the finite-difference scheme outlined above, no matrix inversion is required at each time step. Such a scheme is also known as an explicit scheme. The use of an explicit scheme is a major advantage of the finite-difference method compared to the finite-element methods. Consequently, in order to update N grid points using (37.1.15) or (37.1.18), $O(N)$ multiplications are required for each time step. In comparison, $O(N^3)$ multiplications are required to invert an $N \times N$ full matrix, e.g., using Gaussian elimination. The simplicity and efficiency of these finite-difference algorithms have made them very popular.

[2] This is in the spirit of [232].

37.1.3 Stability Analysis

The implementation of the finite-difference scheme using time-marching does not always lead to a stable scheme. Hence, in order for the solution to converge, the time-stepping scheme must at least be stable. Consequently, it is useful to find the condition under which a numerical finite-difference scheme is stable. To do this, one performs the von Neumann stability analysis (von Neumann 1943 [233]) on Equation (37.1.18). We will assume the medium to be homogeneous to simplify the analysis.

As shown previously, any wave can be expanded in terms of sum of plane waves in different directions. This is the spirit of the spectral expansion method. So if a scheme is not stable for a plane wave, it would not be stable for any wave. Consequently, to perform the stability analysis, we assume a propagating plane wave as a trial solution

$$\phi(x,y,z,t) = A(t)e^{ik_x x + ik_y y + ik_z z}, \tag{37.1.19}$$

In discretized form, it is just

$$\phi^l_{m,n,p} = A^l e^{ik_x m\Delta s + ik_y n\Delta s + ik_z p\Delta s}. \tag{37.1.20}$$

Using (37.1.20), it is easy to show that for the x space derivative,

$$\begin{aligned}\phi^l_{m+1,n,p} - 2\phi^l_{m,n,p} + \phi^l_{m-1,n,p} &= 2[\cos(k_x\Delta s) - 1]\phi^l_{m,n,p} \\ &= -4\sin^2\left(\frac{k_x\Delta s}{2}\right)\phi^l_{m,n,p}.\end{aligned} \tag{37.1.21}$$

The space derivatives in y and z directions can be similarly derived.

The time derivative can be treated and it is proportional to

$$\frac{\partial^2}{\partial t^2}\phi(\mathbf{r},t)(\Delta t)^2 \approx \phi^{l+1}_{m,n,p} - 2\phi^l_{m,n,p} + \phi^{l-1}_{m,n,p}. \tag{37.1.22}$$

Substituting (37.1.20) into the above, we have the second time derivative being proportional to

$$\frac{\partial^2}{\partial t^2}\phi(\mathbf{r},t)(\Delta t)^2 \approx (A^{l+1} - 2A^l + A^{l-1})e^{ik_x m\Delta s + ik_y n\Delta s + ik_z p\Delta s} \tag{37.1.23}$$

To simplify further, one can assume that

$$A^{l+1} = gA^l. \tag{37.1.24}$$

This is commensurate with assuming that

$$A(t) = A_0 e^{-i\omega t} \tag{37.1.25}$$

where ω can be complex. In other words, our trial solution (37.1.19) is also a time-harmonic signal. If the finite-difference scheme is unstable for such a signal, it is unstable for all signals.

Consequently, the time derivative is proportional to

$$\frac{\partial^2}{\partial t^2}\phi(\mathbf{r},t)(\Delta t)^2 \approx (g - 2 + g^{-1})\phi^l_{m,n,p} \tag{37.1.26}$$

We need to find the value of g for which the solution (37.1.20) satisfies (37.1.18). To this end, one uses (37.1.21) and (37.1.24) in (37.1.18), and repeating (37.1.21), which is for m variable, for the n and p variables, one obtains

$$\begin{aligned}(g-2+g^{-1})\phi^l_{m,n,p} &= -4\left(\frac{\Delta t}{\Delta s}\right)^2 c^2\left[\sin^2\left(\frac{k_x\Delta s}{2}\right)+\sin^2\left(\frac{k_y\Delta s}{2}\right)\right.\\ &\qquad\left.+\sin^2\left(\frac{k_z\Delta s}{2}\right)\right]\phi^l_{m,n,p}\\ &= -4r^2s^2\phi^l_{m,n,p},\end{aligned} \tag{37.1.27}$$

where

$$r=\left(\frac{\Delta t}{\Delta s}\right)c, \qquad s^2=\sin^2\left(\frac{k_x\Delta s}{2}\right)+\sin^2\left(\frac{k_y\Delta s}{2}\right)+\sin^2\left(\frac{k_z\Delta s}{2}\right). \tag{37.1.28}$$

Equation (37.1.27) implies that, for nonzero $\phi^l_{m,n,p}$,

$$g^2-2g+4r^2s^2g+1=0, \tag{37.1.29}$$

or that

$$g=(1-2r^2s^2)\pm 2rs\sqrt{(r^2s^2-1)}\,. \tag{37.1.30}$$

In order for the solution to be stable, it is necessary that $|g|\le 1$. But if

$$r^2s^2<1, \tag{37.1.31}$$

the second term in (37.1.30) is pure imaginary, and

$$|g|^2=(1-2r^2s^2)^2+4r^2s^2(1-r^2s^2)=1, \tag{37.1.32}$$

when (37.1.31) is true. Therefore, stability is ensured. Since $s^2\le 3$ for all k_x, k_y, and k_z, from (37.1.31), one concludes that the general condition for stability is

$$r<\frac{1}{\sqrt{3}}, \qquad \text{or} \qquad \Delta t<\frac{\Delta s}{c\sqrt{3}}. \tag{37.1.33}$$

The above analysis is for 3 dimensional problems. It is clear from the above analysis that for an n-dimensional problem where $n=1,2,3$, then

$$\Delta t<\frac{\Delta s}{c\sqrt{n}}. \tag{37.1.34}$$

One may ponder on the meaning of this inequality further: but it is only natural that the time step Δt has to be bounded from above. Otherwise, one arrives at the ludicrous notion that the time step can be arbitrarily large thus violating causality. Moreover, if the grid points of the finite-difference scheme are regarded as a simple cubic lattice, then the distance $\Delta s/\sqrt{n}$ is also the distance between the closest lattice planes through the simple cubic lattice. Notice that the time for the wave to travel between these two lattice planes is $\Delta s/(c\sqrt{n}\,)$. Consequently, the stability criterion (37.1.34) implies that the time step Δt has to be less than the shortest travel time for the wave between the lattice planes in order to satisfy causality. In other words, if the wave is time-stepped ahead of the time on the right-hand side of (37.1.34), instability ensues. The above is also known as the CFL (Courant, Friedrichs, and Lewy 1928 [234]) stability criterion. It could be easily modified for $\Delta x \neq \Delta y \neq \Delta z$.

The above analysis implies that we can pick a larger time step if the space steps are larger. A larger time step will allow one to complete generating a time-domain response rapidly. However, one cannot arbitrary make the space step large due to grid-dispersion error, as shall be discussed next.

37.1.4 Grid-Dispersion Error

When a finite-difference scheme is stable, it still may not produce good results because of the errors in the finite-difference approximations. Hence, it is useful to ascertain the errors in terms of the size of the grid and the time step. An easy error to analyze is the ***grid-dispersion error***. In a homogeneous, dispersionless medium, all plane waves propagate with the same phase velocity. However, in the finite-difference approximation, all plane waves will not propagate at the same phase velocity due to the grid-dispersion error.

As a consequence, a pulse in the time domain, which is a linear superposition of plane waves with different frequencies, will be distorted if the dispersion introduced by the finite-difference scheme is intolerable. Therefore, to make things simpler, we will analyze the grid-dispersion error in a homogeneous free space medium.

To ascertain the grid-dispersion error, we assume that the solution is time-harmonic, or that $A^l = e^{-i\omega l\Delta t}$ in (37.1.20). In this case, the left-hand side of (37.1.27) becomes

$$\left(e^{-i\omega\Delta t} - 2 + e^{+i\omega\Delta t}\right)\phi^l_{m,n,p} = -4\sin^2\left(\frac{\omega\Delta t}{2}\right)\phi^l_{m,n,p}. \tag{37.1.35}$$

Then, from Equation (37.1.27), it follows that

$$\sin\left(\frac{\omega\Delta t}{2}\right) = rs, \tag{37.1.36}$$

where r and s are given in (37.1.28). Now, Equation (37.1.36) governs the relationship between ω and k_x, k_y, and k_z in the finite-difference scheme, and hence, is a dispersion relation for the approximate solution.

But if a medium is homogeneous, it is well known that (37.1.1) has a plane-wave solution of the type given by (37.1.19) where

$$\omega = c\sqrt{k_x^2 + k_y^2 + k_z^2} = c|\mathbf{k}| = ck. \tag{37.1.37}$$

where $\mathbf{k} = \hat{x}k_x + \hat{y}k_y + \hat{z}k_z$ is the direction of propagation of the plane wave. Defining the phase velocity to be $\omega/k = c$, this phase velocity is isotropic, or the same in all directions. Moreover, it is independent of frequency. But in (37.1.36), because of the definition of s as given by (37.1.28), the dispersion relation between ω and $\mathbf{k}$ is not isotropic. This implies that plane waves propagating in different directions will have different phase velocities.

Equation (37.1.36) is the dispersion relation for the approximate solution. It departs from Equation (37.1.37), the exact dispersion relation, as a consequence of the finite-difference approximation. This departure gives rise to errors, which are the consequence of grid dispersion. For example, when c is a constant, (37.1.37) states that the phase velocities of plane waves of different wavelengths and directions are the same. However, this is not true for (37.1.36), as shall be shown.

Assuming s small, (37.1.36), after using Taylor series expansion, can be written as

$$\frac{\omega\Delta t}{2} = \sin^{-1} rs \cong rs + \frac{r^3 s^3}{6}. \tag{37.1.38}$$

When Δs is small, using the small argument approximation for the sine function, one obtains from (37.1.28)

$$s \simeq \frac{\Delta s}{2}(k_x^2 + k_y^2 + k_z^2)^{1/2} \tag{37.1.39}$$

Equation (37.1.38), by taking the higher-order Taylor expansion of (37.1.38), then becomes

$$\frac{\omega\Delta t}{2} \simeq r\frac{\Delta s}{2}(k_x^2 + k_y^2 + k_z^2)^{1/2}\,[1 - \delta] \tag{37.1.40}$$

where (see [34])

$$\delta = \frac{\Delta s^2}{24}\frac{k_x^4 + k_y^4 + k_z^4}{k_x^2 + k_y^2 + k_z^2} - \frac{r^2 \Delta s^2}{24}(k_x^2 + k_y^2 + k_z^2) \tag{37.1.41}$$

Since $\mathbf{k}$ is inversely proportional to wavelength λ, then δ in the correction to the above equation is proportional to $\Delta s^2/\lambda^2$. Therefore, to reduce the grid dispersion error, it is necessary for δ to be small or to have

$$\left(\frac{\Delta s}{\lambda}\right)^2 \ll 1. \tag{37.1.42}$$

When this is true, using the fact that $r = c\Delta t/\Delta s$, then (37.1.40) becomes

$$\frac{\omega}{c} \approx \sqrt{k_x^2 + k_y^2 + k_z^2}\,. \tag{37.1.43}$$

which is close to the dispersion relation of free space. Consequently, in order for the finite-difference scheme to propagate a certain frequency content accurately, the grid size must be much less than the wavelength of the corresponding frequency. Furthermore, Δt must be chosen so that the CFL stability criterion is met. Hence, the rule of thumb is to choose about 10 to 20 grid points per wavelength. Also, for a plane wave propagating as $e^{i\mathbf{k}\cdot\mathbf{r}}$, an error $\delta\mathbf{k}$ in the vector $\mathbf{k}$ gives rise to cumulative error $e^{i\delta\mathbf{k}\cdot\mathbf{r}}$. The larger the distance traveled, the larger the cumulative phase error, and hence the grid size must be smaller in order to arrest such phase error due to the grid dispersion.

37.2 The Yee Algorithm

The Yee algorithm (Yee 1966 [224]) is specially designed to solve vector electromagnetic field problems on a rectilinear grid. The finite-difference time-domain (FDTD) method (Taflov 1988) when applied to solving electromagnetics problems, usually uses this method. To derive it, Maxwell's equations are first written in cartesian coordinates:

$$-\frac{\partial B_x}{\partial t} = \frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z}, \tag{37.2.1}$$

$$-\frac{\partial B_y}{\partial t} = \frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x}, \tag{37.2.2}$$

$$-\frac{\partial B_z}{\partial t} = \frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y}, \tag{37.2.3}$$

$$\frac{\partial D_x}{\partial t} = \frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} - J_x, \tag{37.2.4}$$

$$\frac{\partial D_y}{\partial t} = \frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} - J_y, \tag{37.2.5}$$

$$\frac{\partial D_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - J_z. \tag{37.2.6}$$

Before proceeding any further, it is prudent to rewrite the differential equation form of Maxwell's equations in their integral form. The first equation above can be rewritten as

$$-\frac{\partial}{\partial t} \iint_{\Delta S} \mu H_x dS = \oint_{\Delta C} \mathbf{E} \cdot d\mathbf{l} \tag{37.2.7}$$

where $\Delta S = \Delta x \Delta z$. The approximation of this integral form will be applied to the surface that is closest to the observer in Figure 37.2. Hence, one can see that the curl of $\mathbf{E}$ is proportional to the time-derivative of the magnetic flux through the suface enclosed by ΔC, which is ΔS. One can see this relationship for the other surfaes of the cube in the figure as well: the electric field is curling around the magnetic flux. For the second half of the above equations, one can see that the magnetic fields are curling around the electric flux, but on a staggered grid. These two staggered grids are intertwined with respect to each other. This is the spirit with which the Yee algorithm is written.

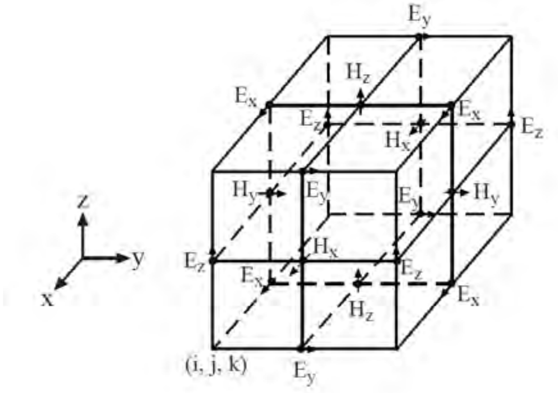

Figure 37.2: The assignment of fields on a grid in the Yee algorithm.

After denoting $f(m\Delta x, n\Delta y, p\Delta z, l\Delta t) = f^l_{m,n,p}$, and replacing derivatives with central finite-differences in accordance with Figure 37.2, (37.2.1) becomes

$$\frac{1}{\Delta t}\left[B^{l+\frac{1}{2}}_{x,m,n+\frac{1}{2},p+\frac{1}{2}} - B^{l-\frac{1}{2}}_{x,m,n+\frac{1}{2},p+\frac{1}{2}}\right] = \frac{1}{\Delta z}\left[E^{l}_{y,m,n+\frac{1}{2},p+1} - E^{l}_{y,m,n+\frac{1}{2},p}\right] - \frac{1}{\Delta y}\left[E^{l}_{z,m,n+1,p+\frac{1}{2}} - E^{l}_{z,m,n,p+\frac{1}{2}}\right]. \tag{37.2.8}$$

Moreover, the above can be repeated for (37.2.2) and (37.2.3). Notice that in Figure 37.2, the electric field is always assigned to the edge center of a cube, whereas the magnetic field is always assigned to the face center of a cube.

In fact, after multiplying (37.2.8) by $\Delta z \Delta y$, (37.2.8) is also the approximation of the integral forms of Maxwell's equations when applied at a face of a cube. By doing so, the left-hand side of (37.2.8) becomes

$$(\Delta y \Delta z/\Delta t)\left[B^{l+\frac{1}{2}}_{x,m,n+\frac{1}{2},p+\frac{1}{2}} - B^{l-\frac{1}{2}}_{x,m,n+\frac{1}{2},p+\frac{1}{2}}\right], \tag{37.2.9}$$

which is the time variation of the total flux through an elemental area $\Delta y \Delta z$. Moreover, by summing this flux on the six faces of the cube shown in Figure 37.2, and using the right-hand side of (37.2.8) and its equivalent, it can be shown that the magnetic flux adds up to zero. Hence, $\frac{\partial}{\partial t}\nabla \cdot \mathbf{B} = 0$ condition is satisfied within the numerical approximations of Yee algorithm.

Furthermore, a similar approximation of (37.2.4) leads to

$$\frac{1}{\Delta t}\left[D^{l}_{x,m+\frac{1}{2},n,p} - D^{l-1}_{x,m+\frac{1}{2},n,p}\right] = \frac{1}{\Delta y}\left[H^{l-\frac{1}{2}}_{z,m+\frac{1}{2},n+\frac{1}{2},p} - H^{l-\frac{1}{2}}_{z,m+\frac{1}{2},n-\frac{1}{2},p}\right] - \frac{1}{\Delta z}\left[H^{l-\frac{1}{2}}_{y,m+\frac{1}{2},n,p+\frac{1}{2}} - H^{l-\frac{1}{2}}_{y,m+\frac{1}{2},n,p-\frac{1}{2}}\right] - J^{l-\frac{1}{2}}_{x,m+\frac{1}{2},n,p}. \tag{37.2.10}$$

Also, similar approximations apply for (37.2.5) and (37.2.6). In addition, the above has an interpretation similar to (37.2.8) if one thinks in terms of a cube that is shifted by half a grid point in each direction. Hence, the approximations of (37.2.4) to (37.2.6) are consistent with the approximation of $\frac{\partial}{\partial t}\nabla \cdot \mathbf{D} = -\nabla \cdot \mathbf{J}$. This way of alternatively solving for the **B** and **D** fields in tandem while the fields are placed on a staggered grid is also called the leap-frog scheme.

In the above, $\mathbf{D} = \epsilon\mathbf{E}$ and $\mathbf{B} = \mu\mathbf{H}$. Since the magnetic field and the electric field are assigned on staggered grids, μ and ϵ may have to be assigned on staggered grids. This does not usually lead to serious problems if the grid size is small. Alternatively, (37.1.16) and (37.1.17) can be used to remove this problem.

By eliminating the **E** or the **H** field from the Yee algorithm, it can be shown that the Yee algorithm is equivalent to finite differencing the vector wave equation directly. Hence, the Yee algorithm is also constrained by the CFL stability criterion.

The following figures show some results of FDTD simulations. Because the answers are in the time-domain, beautiful animations of the fields are also available online:

https://www.remcom.com/xfdtd-3d-em-simulation-software

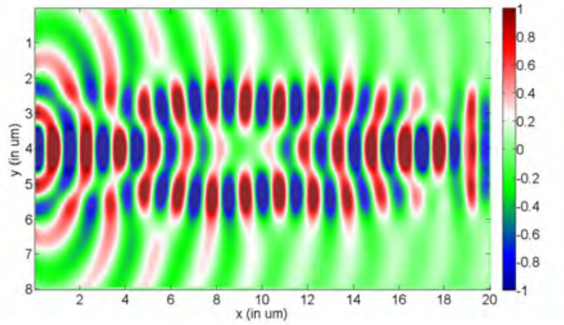

Figure 37.3: The 2D FDTD simulation of complicated optical waveguides (courtesy of Mathworks).

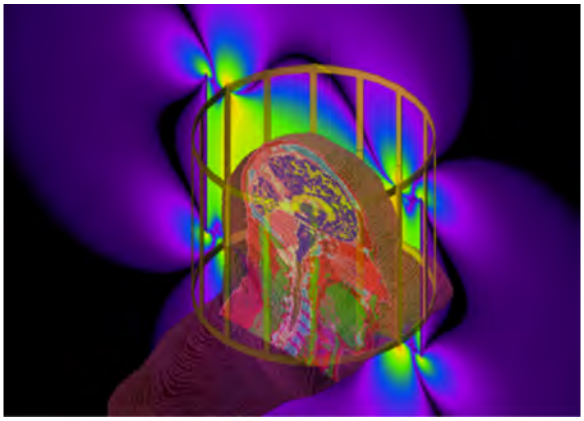

Figure 37.4: FDTD simulation of human head in a squirrel cage of an MRI (magnetic resonance imaging) system (courtesy of REMCOM).

37.2.1 Finite-Difference Frequency Domain Method

Unlike electrical engineering, in many fields, nonlinear problems are prevalent. But when we have a linear time-invariant problem, it is simpler to solve the problem in the frequency domain. This is analogous to perform a time Fourier transform of the pertinent linear equations.

Consequently, one can write (37.2.1) to (37.2.6) in the frequency domain to remove the time derivatives. Then one can apply the finite difference approximation to the space derivatives using the Yee grid. As a result, one arrives at a matrix equation

$$\overline{\mathbf{A}} \cdot \mathbf{x} = \mathbf{b} \tag{37.2.11}$$

where $\mathbf{x}$ is an unknown vector containing $\mathbf{E}$ and $\mathbf{H}$ fields, and $\mathbf{b}$ is a source vector that drives the system containing $\mathbf{J}$. Due to the near-neighbor interactions of the fields on the Yee grid, the matrix $\overline{\mathbf{A}}$ is highly sparse and contains $O(N)$ non-zero elements. When an iterative method is used to solve the above equation, the major cost is in performing a matrix-vector product $\overline{\mathbf{A}} \cdot \mathbf{x}$. However, in practice, the matrix $\overline{\mathbf{A}}$ is never generated nor stored making this a matrix-free method. Because of the simplicity of the Yee algorithm, a code can be written to produce the action of $\overline{\mathbf{A}}$ on $\mathbf{x}$.

37.3 Absorbing Boundary Conditions

It will not be complete to close this lecture without mentioning absorbing boundary conditions. As computer has finite memory, space of infinitely large extent cannot be simulated with finite computer memory. Hence, it is important to design absorbing boundary conditions at the walls of the simulation domain or box, so that waves impinging on it are not reflected. This mimicks the physics of an infinitely large box.

This is analogous to experments in microwave engineering. In order to perform experiments in an infinite space, such experiments are usually done in an anechoic (non-echoing or non-reflecting) chamber. An anechoic chamber has its walls padded with absorbing materials or microwave absorbers so as to minimize the reflections off its walls (see Figure 37.5). Figure 37.6 shows an acoustic version of anechoic chamber.

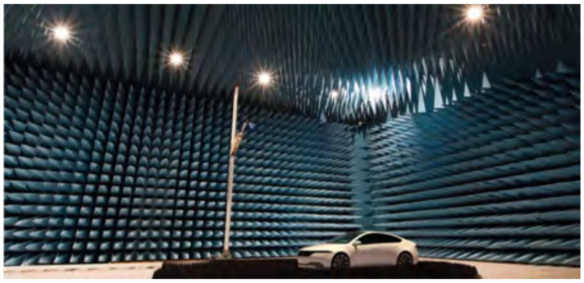

Figure 37.5: An anechoic chamber for radio frequency. In such an electromagnetically quiet chamber, interference from other RF equipment is minimized (courtesy of Panasonic).

Figure 37.6: An acoustic anechoic chamber. In such a chamber, even the breast-feeding sound of a baby can be heard clearly (courtesy of AGH University, Poland).

By the same token, in order to simulate an infinite box with a finite-size box, absorbing boundary conditions (ABCs) are designed at its walls. The simplest of such ABCs is the impedance boundary condition. (A transmission line terminated with an impedance reflects less than one terminated with an open or a short circuit.) Another simple ABC is to mimick the Sommerfeld radiation condition (much of this is reviewed in [34].

A recently invented ABC is the perfectly matched layers (PML) [235]. Also, another similar ABC is the stretched coordinates PML [236]. Figure 37.7 shows simulation results with and without stretched coordinates PMLs on the walls of the simulation domain [237].

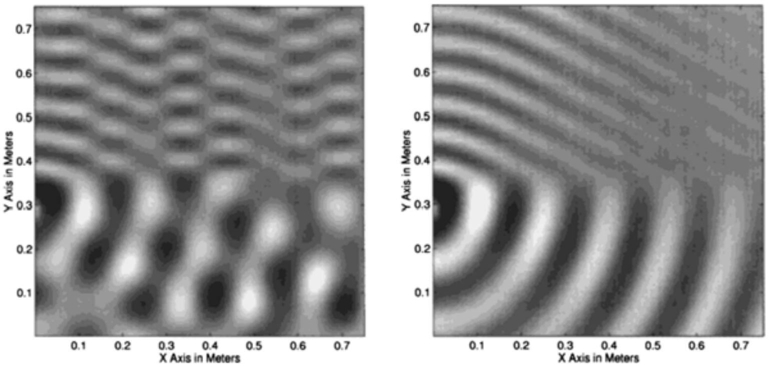

Figure 37.7: Simulation of a source on top of a half-space (left) without stretched coordinates PML; and (right) with stretched coordinats PML [237].