Electromagnetic Wave Propagation
Lecture 5: Finite differences in the time domain

Daniel Sjöberg

Department of Electrical and Information Technology
Outline

1. Finite differences
2. Generalization to 3D
3. Dispersion analysis in 3D
4. Example applications
5. Conclusions
1 Finite differences

2 Generalization to 3D

3 Dispersion analysis in 3D

4 Example applications

5 Conclusions
Some problems are too complicated to do by hand, such as a pacemaker in the human body. Numerical methods help.
Finite difference approximation

To make numerical simulations, we approximate the derivatives with finite differences (where the field is evaluated in grid points $E|_{nr}^n = E(r \Delta z, n \Delta t)$:

$$\frac{\partial E}{\partial z} \bigg|_{r}^{n} = \frac{E|_{r+1/2}^n - E|_{r-1/2}^n}{\Delta z}$$

$$\frac{\partial^2 E}{\partial z^2} \bigg|_{r}^{n} = \frac{\partial E}{\partial z} \bigg|_{r+1/2}^n - \frac{\partial E}{\partial z} \bigg|_{r-1/2}^n = \frac{E|_{r+1}^n - E|_{r}^n}{\Delta z} - \frac{E|_{r}^n - E|_{r-1}^n}{\Delta z}$$

Thus, the wave equation $\frac{\partial^2 E}{\partial z^2} - \frac{1}{c^2} \frac{\partial^2 E}{\partial t^2} = 0$ becomes

$$\frac{E|_{r+1}^n - 2E|_{r}^n + E|_{r-1}^n}{(\Delta z)^2} - \frac{1}{c^2} \frac{E|_{r}^{n+1} - 2E|_{r}^n + E|_{r}^{n-1}}{(\Delta t)^2} = 0$$

We expect the error in the approximation to decrease as $\Delta z$ and $\Delta t$ become small.
Time stepping

The finite difference approximation of the wave equation is

$$\frac{E_{r+1}^n - 2E_r^n + E_{r-1}^n}{(\Delta z)^2} - \frac{1}{c^2} \frac{E_{r+1}^{n+1} - 2E_r^n + E_r^{n-1}}{(\Delta t)^2} = 0$$

By solving for the field at time step $n + 1$, we find

$$E_r^{n+1} = 2E_r^n - E_r^{n-1} + \left(\frac{c\Delta t}{\Delta z}\right)^2 (E_{r+1}^n - 2E_r^n + E_{r-1}^n)$$

Thus, the solution at time $(n + 1)\Delta t$ can be found if the solution at two previous time steps is known.
Initial conditions: $E|_r^0 = f_r$ and $E|_r^1 = g_r$ (known functions).
Boundary conditions: $E|_0^n = 0$ and $E|_R^n = 0$ (metal walls).
Python program demonstration
A harmonic wave propagating through the lattice can be written

\[ e^{j(\omega n \Delta t - kr \Delta z)} \]

For a “real” wave we should have \( \omega = ck \). Inserting the exponential function into the difference approximation we obtain

\[ e^{j\omega n \Delta t - jkr \Delta z} \left( \frac{e^{-jk \Delta z} - 2 + e^{jk \Delta z}}{(\Delta z)^2} - \frac{1}{c^2} \frac{e^{j\omega \Delta t} - 2 + e^{-j\omega \Delta t}}{(\Delta t)^2} \right) = 0 \]

which can be rewritten as

\[ \left( \sin \frac{\omega \Delta t}{2} \right)^2 = \left( \frac{c \Delta t}{\Delta z} \right)^2 \left( \sin \frac{k \Delta z}{2} \right)^2 \]

This relates the spatial frequency \( k \) to the temporal frequency \( \omega \) on the *numerical grid*, and is called the *dispersion relation*. 
Time step and stability: \( R = \frac{c \Delta t}{\Delta z} \)

**\( R = 1 \):** Magic time step, \( \omega = \pm ck \), no dispersion.

**\( R < 1 \):** Stable solution, \( \omega \neq \pm ck \), different frequencies travel with different speed (lower than \( c \)).

**\( R > 1 \):** Unstable solution, growing exponentially.
Example: square wave

\[ R = 1 \]

\[ R < 1 \]

\[ R > 1 \]

(Figs. 5.3–5.5 in Bondeson et al)
Example: smooth wave

(Figs. 5.6–5.7 in Bondeson et al)

Upper graph: 12 points across $1/e$ width.
Lower graph: 6 points across $1/e$ width.
Computation of resonance frequencies

Even though FDTD solves problems in the time domain, it can be used to compute resonance frequencies. In a rectangular box with metal walls (cavity), only waves with particular resonance frequencies can exist.

\[ f_{mnp} = \frac{c}{2} \left[ \left( \frac{m}{L_x} \right)^2 + \left( \frac{n}{L_y} \right)^2 + \left( \frac{p}{L_z} \right)^2 \right]^{1/2} \]
To compute the resonance frequencies in a cavity using FDTD, you can do the following:

- Discretize the cavity with finite differences.
- Set up PEC boundary conditions.
- Give random data as initialization in order to excite all frequencies.
- Run simulation for a long time.
- Fourier transform the field, sampled at a point.
- The resulting spikes correspond to the resonances of the cavity.

Explicit matlab code in Bondeson, Rylander, Ingelström. See also the python demo fdttdt1d.py.
1 Finite differences

2 Generalization to 3D

3 Dispersion analysis in 3D

4 Example applications

5 Conclusions
Maxwell’s equations are

\[
\begin{align*}
\frac{\partial E_x}{\partial z} &= -\mu \frac{\partial H_y}{\partial t} \\
\frac{\partial H_y}{\partial z} &= -\epsilon \frac{\partial E_x}{\partial t}
\end{align*}
\]

Instead of discretizing \( E_x \) and \( H_y \) at the same grid points, shift one field half a grid point in space and time:

\[
\begin{align*}
\frac{E_x|_{r+1}^n - E_x|_r^n}{\Delta z} &= -\mu \frac{H_y|_{r+1/2}^{n+1/2} - H_y|_{r+1/2}^{n-1/2}}{\Delta t} \\
\frac{H_y|_{r+1/2}^{n+1/2} - H_y|_{r-1/2}^{n+1/2}}{\Delta z} &= -\epsilon \frac{E_x|_r^{n+1} - E_x|_r^n}{\Delta t}
\end{align*}
\]

This makes it possible to use central differences for all derivatives.
Graphical point of view

\[ n = \frac{t}{\Delta t} \]

\[ r = \frac{z}{\Delta z} \]

(Fig. 5.8 in Bondeson et al)
Generalization to 3D

E-field along edges, H-field along sides. Also staggered in time.
Integral interpretation

\[ \oint E \cdot dr = \int_S \nabla \times E \cdot \hat{n} \, dS = - \int_S \mu_0 \frac{\partial H}{\partial t} \cdot \hat{n} \, dS = - \Delta x \Delta y \mu_0 \frac{\partial}{\partial t} H_z \]

\[ \oint E \cdot dr = E_x |_{p+1/2,q} \Delta x + E_y |_{p+1,q+1/2} \Delta y \]

\[ - E_x |_{p+1/2,q+1} \Delta x - E_y |_{p,q+1/2} \Delta y \]

\[ H_z \] can be updated from knowledge of surrounding \( E_{xy} \)-fields. Similarly for \( \nabla \times H = \partial D/\partial t \).
For a PEC boundary, $\hat{n} \times E = 0$ and $\hat{n} \cdot H = 0$. Note different number of points for $E$ and $H$ for different components in different directions.
The staggered grid

- Invented by K. S. Yee in 1966.
- Boundary conditions involve only half of the field components (usually $\hat{n} \times E$ and $\hat{n} \cdot H$)
- Completely dominating in commercial codes
- Based on decoupling of electric and magnetic fields: $\nabla \times E = -\partial B / \partial t$ and $\nabla \times H = \partial D / \partial t$: time differences of $E$ depend only on space differences of $H$ and vice versa
- For bianisotropic materials, where $D$ and $B$ depend on both $E$ and $H$, the fields may need to be evaluated at the same points in space and time
Staircasing

The rectangular grid does not conform to curved surfaces:
Grid refinement

Simple method: refine along planes where the geometry has small scale

Advanced method: implicit time stepping in regions with unstructured grid (Rylander & Bondeson 2000)
Outline

1. Finite differences
2. Generalization to 3D
3. Dispersion analysis in 3D
4. Example applications
5. Conclusions
Harmonic waves and difference approximations

When considering harmonic waves

\[ u = e^{j(\omega t - k \cdot r)} = e^{j(\omega t - k_x x - k_y y - k_z z)} \]

the difference approximation of the derivatives become

\[ \frac{\partial u}{\partial t} \bigg|_n \approx \frac{u|_{r+1/2}^n - u|_{r-1/2}^n}{\Delta t} = u|_r^n \frac{e^{j\omega \Delta t/2} - e^{-j\omega \Delta t/2}}{\Delta t} \]

\[ = u|_r^n \frac{2j}{\Delta t} \sin \frac{\omega \Delta t}{2} \]

\[ \frac{\partial u}{\partial x} \bigg|_n \approx \frac{u|_{r+1/2}^n - u|_{r-1/2}^n}{\Delta x} = u|_r^n \frac{e^{-j k_x \Delta x/2} - e^{j k_x \Delta x/2}}{\Delta x} \]

\[ = u|_r^n \frac{-2j}{\Delta x} \sin \frac{k_x \Delta x}{2} \]
Dispersion relation

This means derivatives can be replaced by algebraic expressions as

\[
\frac{\partial}{\partial t} \rightarrow D_t = \frac{2j}{\Delta t} \sin \frac{\omega \Delta t}{2} \\
\frac{\partial}{\partial x} \rightarrow D_x = -\frac{2j}{\Delta x} \sin \frac{k_x \Delta x}{2}
\]

and so on. Eliminating the magnetic field, Maxwell’s equations reduce to the wave equation \( \nabla^2 E - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} E \), or

\[
\left( D_x^2 + D_y^2 + D_z^2 - \frac{1}{c^2} D_t^2 \right) E = 0
\]

Using the expressions for \( D_x, D_y, D_z \), and \( D_t \), we find

\[
\frac{\sin^2(\omega \Delta t/2)}{(c \Delta t)^2} = \frac{\sin^2(k_x \Delta x/2)}{(\Delta x)^2} + \frac{\sin^2(k_y \Delta y/2)}{(\Delta y)^2} + \frac{\sin^2(k_z \Delta z/2)}{(\Delta z)^2}
\]
In order to have a stable scheme we need real $\omega$, that is

$$\sin^2(\omega \Delta t/2) \leq 1$$

for all wave vectors $\mathbf{k} = k_x \hat{x} + k_y \hat{y} + k_z \hat{z}$. The dispersion relation then implies

$$c\Delta t \leq \left[ \frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} + \frac{1}{(\Delta z)^2} \right]^{-1/2}$$

For a cubic grid with $\Delta x = \Delta y = \Delta z = h$,

$$\Delta t \leq \frac{h}{c \sqrt{3}} \approx 0.577 \frac{h}{c}$$

Thus, the maximum time step is almost half as short in three dimensions as in one.
From the dispersion relation
\[
\frac{\sin^2(\omega \Delta t/2)}{(c \Delta t)^2} = \frac{\sin^2(k_x \Delta x/2)}{(\Delta x)^2} + \frac{\sin^2(k_y \Delta y/2)}{(\Delta y)^2} + \frac{\sin^2(k_z \Delta z/2)}{(\Delta z)^2}
\]
we can compute \(\omega(k)\) and hence the phase and group velocities by Taylor series (allowing 1% error):

\[
v_p = \frac{\omega}{k} = c \left( 1 - \frac{(kh)^2}{36} + O((kh)^4) \right) \quad \text{10.5 cells per } \lambda
\]

\[
v_g = \frac{\partial \omega}{\partial k} = c \left( 1 - \frac{(kh)^2}{12} + O((kh)^4) \right) \quad \text{18 cells per } \lambda
\]

The total phase error in domain of diameter \(L\) is

\[
e_{\text{phase}} = (k_{\text{num}} - k)L = \left( \frac{\omega}{c(1 - (kh)^2/36 + \cdots)} - \frac{\omega}{c} \right) L \approx \frac{k^3 h^2 L}{36}
\]

meaning that in addition to the above requirements, the cell size \(h\) must scale as \(1/(kL)\) to control the total phase error, or \(h \sim \omega^{-3/2}\).
Outline

1. Finite differences
2. Generalization to 3D
3. Dispersion analysis in 3D
4. Example applications
5. Conclusions
Typical flow chart for an FDTD code

- **Preprocessing**
  - Load grid and geometry
  - Memory allocation
  - Declare constants
  - Compute parameters

- **Main loop (iterate the following steps)**
  - Evaluate sources
  - Update internal H fields
  - Update internal E fields
  - Update boundary conditions
  - Save some data, typically on the boundary

- **Postprocessing**
  - Fourier transform to frequency domain
  - Near-to-farfield transformation
  - Scattering coefficients
  - ...
Typical complexity

- Size of computational region in wavelengths
- At least 10–20 points per wavelength
- Small geometries may require high discretization
- Number of time steps (becomes large at resonances)
Wireless communication with a pacemaker

- Frequency? \( f = 402\text{–}405 \text{MHz} \)
- Permittivity? \( \epsilon_r \approx 64 \)
- Conductivity? \( \sigma \approx 0.94 \text{S/m} \)
- Wavelength? \[
\lambda = \frac{c}{f} \approx \frac{c_0}{\left( f \sqrt{\epsilon_r} \right)} = \frac{3 \cdot 10^8}{400 \cdot 10^6} \approx 0.09 \text{m}
\]
- Geometry? \( 1 \cdot 1 \cdot 2 \text{m}^3 = 2 \text{m}^3 \)
- Sampling? \( \lambda/15 \approx 0.006 \text{m} \)
- Number of unknowns? \[
6 \frac{2}{0.006^3} = 55 \cdot 10^6
\]

The number 6 stems from 3 components in \( \textbf{E} \) and 3 components in \( \textbf{H} \). With double precision floats, this means \( 55 \cdot 10^6 \cdot 8 = 0.4 \text{GB} \).
Wireless communication with hearing aids

- **Frequency?**  \( f = 2.45 \text{ GHz} \)
- **Permittivity?**  \( \varepsilon_r \approx 39.2 \)
- **Conductivity?**  \( \sigma \approx 1.80 \text{ S/m} \)
- **Wavelength?**
\[
\lambda = \frac{c}{f} \approx \frac{c_0}{(f \sqrt{\varepsilon_r})} = \frac{3 \cdot 10^8}{2.45 \cdot 10^9 \cdot \sqrt{40}} \approx 0.02 \text{ m}
\]
- **Geometry?**
\[
0.164 \cdot 0.312 \cdot 0.234 \text{ m}^3 \approx 0.01 \text{ m}^3
\]
- **Sampling?**  \( \lambda/15 \approx 0.0013 \text{ m} \)
- **Number of unknowns?**
\[
6 \cdot \frac{0.01}{0.0013^3} = 27 \cdot 10^6
\]

(Figures by Rohit Chandra)
Scattering of light against a blood cell

- Wavelength in vacuum?
  \[ \lambda_0 \approx 0.632 \mu m \]

- Refractive index plasma?
  \[ n_p \approx 1.34 \]

- Refractive index in cell?
  \[ n_c \approx 1.41 \]

- Minimum wavelength?
  \[ \lambda = \lambda_0 / n_c \approx \frac{0.632 \mu m}{1.41} \approx 0.45 \mu m \]

- Diameter?
  \[ 7.76 \mu m \approx 17\lambda \]

- Height?
  \[ 2.50 \mu m \approx 6\lambda \]

- Geometry?
  \[ 8^2 \cdot 3 (\mu m)^3 = 192 (\mu m)^3 \]

- Sampling?
  \[ \lambda / 15 \approx \frac{0.45}{15} \mu m \approx 0.03 \mu m \]

- Number of unknowns?
  \[ 6 \cdot \frac{192}{(0.03)^3} \approx 42 \cdot 10^6 \]
Broadband scattering for array antenna, 1–20 GHz

- **Wavelength in vacuum?**
  \[ \lambda_{\text{min}} = 1.5 \text{ cm}, \lambda_{\text{max}} = 30 \text{ cm} \]

- **Smallest features in antenna?**
  \[ \approx 0.2 \text{ mm} < \lambda_{\text{min}}/20 = 0.75 \text{ mm} \]

- **Size of one antenna element?**
  6.67 mm

- **Number of elements?** 33 × 33

- **Side length?**
  \[ 22 \text{ cm} \approx 15\lambda_{\text{min}} \approx 0.8\lambda_{\text{max}} \]

- **Height?** 2 cm \( \approx 1.3\lambda_{\text{min}} \approx 0.07\lambda_{\text{max}} \)

- **Geometry?**
  \[ 30 \times 30 \times 2 \text{ (cm)}^3 = 1800 \text{ (cm)}^3 \]

- **Sampling?** \( \approx 0.2 \text{ mm} \)

- **Number of unknowns?**
  \[ 6 \frac{1800}{(0.02)^3} \approx 225 \cdot 10^6 \]
What about larger problems?

Sometimes the problem is too large to fit in one computer. How does FDTD perform on clusters? (http://www.lunarc.lu.se)

Alarik cluster in Lund, $208 \times 2 \times 8 = 3328$ processors, 64 bit 3.0 GHz 2–4 GB per core, started Dec 2011.

- Each cell “communicates” only with its nearest neighbors; a block of $N^3$ cells only exchanges $N^2$ data with others.
- Well suited for parallelization!
- Current trend: computation on GPU:s (hardware adapted to parallel operations).
Outline

1. Finite differences
2. Generalization to 3D
3. Dispersion analysis in 3D
4. Example applications
5. Conclusions
Conclusions

- FDTD is a simple numerical scheme for time domain.
- It is vital to choose a smaller discretization in time than in frequency. The relation between time and space determines the numerical dispersion.

Some pros and cons:

+ Easy to implement
+ Can handle advanced material models (next lecture!)
+ Easy analysis
+ Broad band
+ Parallelizable
  - Stair case approximation of geometry
  - Global time step (but local implicit time stepping is possible)
  - Numerical dispersion