Lecture #8 – Outline

• Formulation
  – Prepare Maxwell’s equations
  – Finite-difference approximations
  – Reduction to one dimension
  – Derivation of Update Equations
  – Bells and whistles
    • Grid resolution, time step, sources, boundary conditions, Fourier transforms, transmittance and reflectance

• Implementation
• Sequence for Code Development
• Walkthrough
Formulation of 1D FDTD

Prepare Maxwell’s Equations

We satisfied the divergence equations by adopting the Yee grid scheme. We now only have to deal with the curl equations.

\[ \nabla \times \vec{E} = -\left[ \mu \right] \frac{\partial \vec{H}}{\partial t} \]

\[ \nabla \times \vec{H} = \left[ \varepsilon \right] \frac{\partial \vec{E}}{\partial t} \]

The \( \vec{E} \) and \( \vec{H} \) fields are two to three orders of magnitude different. This will cause rounding errors in your simulation and it is always good practice to normalize your parameters so they are all the same order of magnitude. Here we choose to normalize the magnetic field.

\[ \vec{H} = \mu_0 \vec{H} \]

\[ \nabla \times \vec{E} = -\frac{\mu_r}{c_0} \frac{\partial \vec{H}}{\partial t} \]

\[ \nabla \times \vec{H} = \frac{\varepsilon_r}{c_0} \frac{\partial \vec{E}}{\partial t} \]

Note: \[ c_0 = \frac{1}{\sqrt{\mu_0 \varepsilon_0}} \]
Expand the Curl Equations

\( \nabla \times \vec{E} = \left[ \mu_r \right] \nabla \times \vec{H} \)

\( \nabla \times \vec{H} = \left[ \varepsilon_r \right] \nabla \times \vec{E} \)

Here we assumed linear and non-dispersive materials with diagonal tensors.

Formulation of 1D FDTD

Finite-Difference Approximations
Finite-Difference Approximations

\[ \frac{df_{1.5}}{dx} \approx \frac{f_2 - f_1}{\Delta x} \]

This derivative is defined to exist at the mid point between \( f_1 \) and \( f_2 \).

Finite-Difference Approximation of Time Derivatives

We approximate all of the derivatives in Maxwell’s equations with finite-differences.

\[ \nabla \times \vec{E} = \frac{[\mu_r] \partial \vec{H}}{c_0} \partial t \]
\[ \nabla \times \vec{H} = \frac{[\varepsilon_r] \partial \vec{E}}{c_0} \partial t \]

\[ \nabla \times \vec{E} \bigg|_{t+\Delta t/2} \approx \frac{[\mu_r]}{c_0} \vec{H} \bigg|_{t+\Delta t/2} - \frac{\vec{H}}{\Delta t} \bigg|_{t-\Delta t/2} \]
\[ \nabla \times \vec{H} \bigg|_{t+\Delta t/2} \approx \frac{[\varepsilon_r]}{c_0} \vec{E} \bigg|_{t+\Delta t/2} - \frac{\vec{E}}{\Delta t} \bigg|_{t} \]

The electric and magnetic fields are staggered in time by \( \Delta t/2 \) so that every term in the finite-difference equations exists at the same instant in time.
Representing Functions on a Grid

Example physical (continuous) 2D function

A grid is constructed by dividing space into discrete cells

Function is known only at discrete points

Representation of what is actually stored in memory

Yee Cell for 1D, 2D, and 3D Grids

1D Yee Grid

2D Yee Grids

3D Yee Grid

Benefits
- Implicitly satisfies divergence equations
- Naturally handles physical boundary conditions
- Elegant approximation of the curl equations using finite-differences

Consequences
- Field components are in physically different locations
- Field components may reside in different materials even if they are in the same unit cell
- Field components will be out of phase
Finite-Difference Approximations on a Yee Grid

Finite-Difference Equation for $H_x$

$$
\frac{E_x^{i,j+1,k} - E_x^{i,j,k}}{\Delta y} = \frac{\mu_e}{\varepsilon_0} \left[ H_z^{i,j,k+1} - H_z^{i,j,k} \right]
$$

Finite-Difference Equation for $H_y$

$$
\frac{E_y^{i,j+1,k} - E_y^{i,j,k}}{\Delta z} = \frac{\mu_e}{\varepsilon_0} \left[ H_z^{i,j,k+1} - H_z^{i,j,k} \right]
$$

Finite-Difference Equation for $H_z$

$$
\frac{E_z^{i,j+1,k} - E_z^{i,j,k}}{\Delta x} = \frac{\mu_e}{\varepsilon_0} \left[ H_z^{i,j,k+1} - H_z^{i,j,k} \right]
$$

Finite-Difference Equation for $E_x$

$$
\frac{H_x^{i,j+1,k} - H_x^{i,j,k}}{\Delta y} = \frac{\varepsilon_0}{\mu_e} \left[ E_y^{i,j,k+1} - E_y^{i,j,k} \right]
$$

Finite-Difference Equation for $E_y$

$$
\frac{H_y^{i,j+1,k} - H_y^{i,j,k}}{\Delta z} = \frac{\varepsilon_0}{\mu_e} \left[ E_y^{i,j,k+1} - E_y^{i,j,k} \right]
$$

Finite-Difference Equation for $E_z$

$$
\frac{H_z^{i,j+1,k} - H_z^{i,j,k}}{\Delta x} = \frac{\varepsilon_0}{\mu_e} \left[ E_z^{i,j,k+1} - E_z^{i,j,k} \right]
$$

Formulation of 1D FDTD

Reduction to One Dimension

Lecture 8 Slide 11
Reduction to One Dimension

We saw in Lecture 3 that some problems composed of dielectric slabs can be described in just one dimension. In this case, the materials and the fields are uniform in two directions. Derivatives in these uniform directions will be zero. We will define the uniform directions to be the $x$ and $y$ axes.

\[
\frac{\partial}{\partial x} = \frac{\partial}{\partial y} = 0
\]

$x$ and $y$ Derivatives are Zero

We approximated the derivatives with finite-differences on a Yee grid.

\[
\frac{\partial E_x}{\partial x} = \frac{\partial H_y}{\partial z} = \frac{\varepsilon_0}{c_0} \frac{\partial E_x}{\partial t}
\]
\[
\frac{\partial E_y}{\partial x} = \frac{\partial H_z}{\partial z} = \frac{\mu_0}{c_0} \frac{\partial H_y}{\partial t}
\]
\[
\frac{\partial E_z}{\partial x} = \frac{\partial H_x}{\partial z} = \frac{\mu_0}{c_0} \frac{\partial H_z}{\partial t}
\]
\[
\frac{\partial H_x}{\partial y} = \frac{\partial E_z}{\partial z} = \frac{\varepsilon_0}{c_0} \frac{\partial E_z}{\partial t}
\]
\[
\frac{\partial H_y}{\partial y} = \frac{\partial E_x}{\partial z} = \frac{\mu_0}{c_0} \frac{\partial E_x}{\partial t}
\]
\[
\frac{\partial H_z}{\partial y} = \frac{\partial E_y}{\partial z} = \frac{\mu_0}{c_0} \frac{\partial E_y}{\partial t}
\]
Summary of 1D FDTD Modes

**Ex/Hy Mode**

\[
\frac{E_x^{k+1} - E_x^k}{\Delta z} = \frac{\mu}{c_0} \frac{\tilde{H}_y^{k+\frac{1}{2}} - \tilde{H}_y^{k-\frac{1}{2}}}{\Delta t}
\]

\[
\frac{\tilde{H}_y^{k+\frac{1}{2}} - \tilde{H}_y^{k-\frac{1}{2}}}{\Delta z} = \frac{\varepsilon}{c_0} \frac{E_x^{k+1} - E_x^k}{\Delta t}
\]

**Ey/Hx Mode**

\[
-\frac{E_y^{k+1} - E_y^k}{\Delta z} = -\frac{\mu}{c_0} \frac{\tilde{H}_x^{k+\frac{1}{2}} - \tilde{H}_x^{k-\frac{1}{2}}}{\Delta t}
\]

\[
\frac{\tilde{H}_x^{k+\frac{1}{2}} - \tilde{H}_x^{k-\frac{1}{2}}}{\Delta z} = \frac{\varepsilon}{c_0} \frac{E_y^{k+1} - E_y^k}{\Delta t}
\]

---

Formulation of 1D FDTD

**Derivation of Update Equations**
**E_y/H_x Mode: Update Equation for E_y**

Start with the finite-difference equation which has \( E_y \) in the time-derivative:

\[
\frac{\tilde{H}_x^{i+\frac{1}{2}} - \tilde{H}_x^{i-\frac{1}{2}}}{\Delta z} = \frac{\varepsilon_{y0}}{c_0} \frac{E_y^{i+\frac{1}{2}} - E_y^{i-\frac{1}{2}}}{\Delta t}
\]

Solve this for the field at the future time value.

\[
E_y^{i+\frac{1}{2}} = E_y^{i-\frac{1}{2}} + \frac{c_0 \Delta t}{\varepsilon_{y0}} \left( \frac{\tilde{H}_x^{i+\frac{1}{2}} - \tilde{H}_x^{i-\frac{1}{2}}}{\Delta z} \right)
\]

**E_y/H_x Mode: Update Equation for H_x**

Start with the finite-difference equation which has \( H_x \) in the time-derivative:

\[
-\frac{E_y^{i+1} - E_y^{i}}{\Delta z} = \mu_{x0} \frac{\tilde{H}_x^{i+\frac{1}{2}} - \tilde{H}_x^{i-\frac{1}{2}}}{\Delta t}
\]

Solve this for the field at the future time value.

\[
\frac{\mu_{x0}}{c_0} \frac{\tilde{H}_x^{i+\frac{1}{2}} - \tilde{H}_x^{i-\frac{1}{2}}}{\Delta t} = \frac{E_y^{i+1} - E_y^{i}}{\Delta z}
\]

\[
\tilde{H}_x^{i+\frac{1}{2}} = \tilde{H}_x^{i-\frac{1}{2}} + \frac{c_0 \Delta t}{\mu_{x0}} \left( \frac{E_y^{i+1} - E_y^{i}}{\Delta z} \right)
\]
Update Equations and Update Coefficients

The update coefficients do not change their value during the simulation. They should be computed only once before the main FDTD loop and not at each iteration inside the loop.

The finite-difference equations in terms of the update coefficients are:

\[
\tilde{H}_y^{k+1} = \tilde{H}_y^k + \left( m_{Hx}^k \right) \left( \frac{E_y^{k+1} - E_y^k}{\Delta z} \right)
\]

\[
E_y^{k+1} = E_y^k + \left( m_{Ey}^k \right) \left( \frac{\tilde{H}_x^k - \tilde{H}_x^{k-1}}{\Delta z} \right)
\]

\[
m_{Ey}^k = \frac{c \Delta t}{\varepsilon_{yy}^k}
\]

\[
m_{Hx}^k = \frac{c \Delta t}{\mu_{xx}^k}
\]

Formulation of 1D FDTD

Bells and Whistles
Computing Grid Resolution

1) Resolve Wavelength
\[ \lambda_{\text{max}} = \frac{c_0}{f_{\text{max}} n_{\text{max}}} \]
\[ \Delta_j = \frac{\lambda_{\text{min}}}{N_j} \quad N_j \geq 10 \]

2) Resolve Features
- \( N_j << 1 \)
- \( N_j < 1 \)
- \( N_j = 1 \)
- \( N_j = 4 \)
\[ \Delta_j = \frac{d_{\text{min}}}{N_j} \quad N_j \geq 1 \]

3) Initial Resolution
\[ \Delta' = \min[\Delta_x, \Delta_y, \Delta_z] \]

4) “Snap” Grid to Critical Dimensions
\[ N = \text{ceil}(d_c / \Delta') \]
\[ \Delta = d_c / N \]

Computing the Time Step

Generalized Courant Stability Condition
\[ \Delta t < \frac{n_{\text{min}}}{c_0 \sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}}} \]

For 1D Grids with Perfectly Absorbing Boundary Condition
\[ \Delta t = \frac{n_{bc} \Delta z}{2c_0} \quad n_{bc} \equiv \text{refractive index at boundaries} \]
The Gaussian Source

The Gaussian source approximates an impulse so that a structure can be characterized over an enormous range of frequencies in a single simulation.

\[
g(t) = \exp \left[ -\left( \frac{t - t_0}{\tau} \right)^2 \right]
\]

duration of simulation

\[
\tau \approx \frac{0.5}{f_{\text{max}}} \quad t_0 \approx 6\tau
\]

Estimating Total Number of Iterations

Total Simulation Time

\[
T = 12\tau + 5t_{\text{prop}}
\]

\[t_{\text{prop}} = \frac{n_{\text{max}}N_z\Delta z}{c_0}\]

Time it takes for a wave to propagate across the grid one time.

Allow for 5 bounces.
Highly resonant devices will need much more.
Allow for the entire pulse without cutting it off.

Total Number of Iterations

\[
\text{STEPS} = \text{round} \left[ \frac{T}{\Delta t} \right]
\]

This must be an integer quantity.
The Total-Field/Scattered-Field Framework

1D FDTD Grid

Animation of TF/SF in 1D-FDTD
Correction to Finite-Difference Equations at the Problem Cells (1 of 2)

On the scattered-field side of the TF/SF interface, the finite-difference equation contains a term from the total-field side. Due to the staggered nature of the Yee grid, this only occurs in the update equation for a magnetic field.

\[
\vec{H}_s^{(u)} = \vec{H}_s^{(u-1)} + \left( m_{tx} \right)_s^{(u-1)} \left[ \frac{E_x^{(u)} - E_y^{(u-1)}}{\Delta z} \right]
\]

This is an equation in the scattered-field, but \( E_y^{(u-1)} \) is a total-field quantity.

We must subtract the source from \( E_y^{(u-1)} \) to make it look like a scattered-field quantity.

\[
\vec{H}_s^{(u)} = \vec{H}_s^{(u-1)} + \left( m_{tx} \right)_s^{(u-1)} \left[ \frac{E_x^{(u)} - E_y^{(u-1)}}{\Delta z} \right] - \left( m_{tx} \right)_s^{(u-1)} E_y^{(u-1)}
\]

This is a correction term that can be implemented after the standard update equation to inject a source.

Correction to Finite-Difference Equations at the Problem Cells (2 of 2)

On the total-field side of the TF/SF interface, the finite-difference equation contains a term from the scattered-field side. Due to the staggered nature of the Yee grid, this only occurs in the update equation for an electric field.

\[
E_x^{(u)} = E_x^{(u-1)} + \left( m_{tx} \right)_s^{(u)} \left[ \frac{\vec{H}_s^{(u)} - \vec{H}_s^{(u-1)}}{\Delta z} \right]
\]

This is an equation in the scattered-field, but \( \vec{H}_s^{(u-1)} \) is a total-field quantity.

We must add the source to \( \vec{H}_s^{(u-1)} \) to make it look like a total-field quantity.

\[
E_x^{(u)} = E_x^{(u-1)} + \left( m_{tx} \right)_s^{(u)} \left[ \frac{\vec{H}_s^{(u)} - \vec{H}_s^{(u-1)}}{\Delta z} \right] + \left( m_{tx} \right)_s^{(u)} \vec{H}_s^{(u-1)}
\]

This is a correction term that can be implemented after the standard update equation to inject a source.
The Two Source Terms

From the previous slides, we now know that we need to calculate two source functions before the main FDTD loop. These are:

\[ \text{src}^{\text{src}}_k y t \quad E_x \mid_{t+\Delta t/2} \quad \text{src}^{\text{src}}_k y t \quad E_y \mid_t \]

We need to make a few observations that must be accounted for before we can calculate these source functions correctly.

1. The amplitude of these functions can be different as E and H are related through the material impedance.
2. These functions are a half grid cell apart and have a small time delay between them.
3. These functions exist at different time steps.

Calculation of the Source Functions

**\( E_x/H_y \) Mode**

We calculate the electric field as

\[ E_x^{\text{src}} \mid_{t+\Delta t/2} = g(t) \]

We calculate the magnetic field as

\[ \tilde{H}_y^{\text{src}} \mid_{t+\Delta t/2} = \left[ \frac{c}{\mu''_r} \right]^{\text{src}} \cdot \left( t \right) \]

**Amplitude due to Maxwell's equations**

\[ \left[ \frac{c}{\mu''_r} \right]^{\text{src}} \]

**Delay through one half of a grid cell**

**Half time step difference**

\[ \cdot \]

\( E_y/H_x \) Mode

We calculate the electric field as

\[ E_y^{\text{src}} \mid_t = g(t) \]

We calculate the magnetic field as

\[ \tilde{H}_x^{\text{src}} \mid_{t+\Delta t/2} = -\left[ \frac{c}{\mu''_r} \right]^{\text{src}} \cdot \left( t + \frac{n_{\text{src}} \Delta z}{2 c_0} + \frac{\Delta t}{2} \right) \]

\[ \cdot \]

**Amplitude due to Maxwell's equations**

\[ \left[ \frac{c}{\mu''_r} \right]^{\text{src}} \]

**Delay through one half of a grid cell**

**Half time step difference**

\[ \left( t + \frac{n_{\text{src}} \Delta z}{2 c_0} + \frac{\Delta t}{2} \right) \]

\( \mu''_r, \epsilon''_r, n_{\text{src}} \equiv \) material properties where source is injected
Dirichlet Boundary Condition

Dirichlet boundary conditions assume that all field quantities outside of the grid are zero.

We modify the update equations as follows.

\[ \begin{align*}
\tilde{H}^N_{\xi} \big|_{z-\frac{\Delta}{2}} &= \tilde{H}^N_{\xi} \big|_{z-\frac{\Delta}{2}} + m_{Hz} \left( \frac{E^{N-1}_E - E^N_E}{\Delta z} \right) & k < N_z \\
\tilde{H}^N_{\psi} \big|_{z-\frac{\Delta}{2}} &= \tilde{H}^N_{\psi} \big|_{z-\frac{\Delta}{2}} + m_{Hx} \left( \frac{0 - E^N_E}{\Delta z} \right) & k = N_z \\
E^N_E \big|_{z,\Delta z} &= E^N_E \big|_{z,\Delta z} + m_{Ex} \left( \frac{\tilde{H}^{N-1}_{\psi} - \tilde{H}^N_{\psi}}{\Delta z} \right) & k > 1 \\
E^N_E \big|_{z,\Delta z} &= E^N_E \big|_{z,\Delta z} + m_{Ex} \left( \frac{\tilde{H}^N_{\psi} - 0}{\Delta z} \right) & k = 1
\end{align*} \]

Perfectly Absorbing Boundary Condition

**Conditions**
- Waves at the boundaries are only travelling outward.
- Materials at the boundaries are linear, homogeneous, isotropic and non-dispersive.
- Time step is chosen so physical waves travel 1 cell in two time steps.
\( \Delta t = n\Delta z/(2c_0) \)

**Implementation at \( z \)-Low Boundary**
At the \( z \)-low boundary, we need only modify the E-field update equation.

\[ h_2 = h_1 \quad h_1 = \tilde{H}^N_{\psi} \big|_{z-\frac{\Delta}{2}} \quad E^N_E \big|_{z,\Delta z} = E^N_E \big|_{z,\Delta z} + m_{Ex} \left( \frac{\tilde{H}^{N-1}_{\psi} - h_2}{\Delta z} \right) \]

**Implementation at \( z \)-High Boundary**
At the \( z \)-high boundary, we need only modify the H-field update equation.

\[ e_2 = e_1 \quad e_1 = E^N_E \big|_{z} \quad \tilde{H}^N_{\psi} \big|_{z+\frac{\Delta}{2}} = E^N_E \big|_{z+\Delta z} + m_{Hz} \left( \frac{e_2 - E^N_E}{\Delta z} \right) \]
Efficient Fourier Transform (1 of 2)

The standard Fourier transform is defined as

\[ F(f) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi ft} dt \]

If the function \( f(t) \) is only known at discrete points, the Fourier transform can be approximated numerically as

\[ F(f) \approx \sum_{m=1}^{M} f(m\Delta t) e^{-j2\pi f m\Delta t} \Delta t \]

This can be written in a slightly different form.

\[ F(f) \approx \Delta t \sum_{m=1}^{M} \left( e^{-j2\pi f \Delta t} \right)^m \cdot f(m\Delta t) \]

Example

\[ \Delta t = 33.3564 \text{ ps} \]
\[ f = 1.0000 \text{ GHz} \]
\[ K = 0.9781 - 0.2081i \]

Efficient Fourier Transform (2 of 2)

The final form on the previous slide suggests an efficient implementation. The Fourier transform is updated every iteration so by the end of the main loop:

\[ F(f) \approx \Delta t \sum_{m=1}^{M} \left( e^{-j2\pi f \Delta t} \right)^m \cdot f(m) \]

This multiplication can be done after the main FDTD loop in a post-processing step.

\[ e^{-j2\pi f \Delta t} \]

This “kernel” can be computed prior to the main FDTD loop for each frequency of interest. The kernels can be stored in a 1D array.
Fourier Transforms in FDTD

The easiest, but least memory efficient, method to compute a Fourier transform is to perform a simulation and record the desired field as a function of time. After the simulation is finished, these functions can be Fourier transformed using an FFT.

Post-Processing the Fourier Transforms

We must normalize the spectra to calculate transmittance and reflectance. We do this by dividing the reflection and transmission spectrum by the source spectrum.

It is ALWAYS good practice to check for energy conservation by adding the reflectance and transmittance and ensuring the sum equals 100% (assuming no loss or gain in your device).
Implementation of 1D FDTD

1D-FDTD Grid

Note: A real grid would have 200 or more points.
Initializing the FDTD Simulation

**Initialize Simulation**
- Initialize MATLAB
- Define units
- Define constants

**Define Simulation Parameters**
- Frequency range \( f_{\text{max}} \)
- Device parameters
- Grid parameters (NRES, etc.)

**Compute Grid Resolution**
Initial resolution
\[ \Delta' = \min \left[ \frac{\lambda}{N_y}, \frac{\lambda}{N_z} \right] \]
\( N_y, N_z \geq 10 \)
Snap grid to critical dimension
\( N = \text{ceil} \left[ \frac{d}{\Delta'} \right] \)
\( \Delta = d / N \)

**Build Device on Grid**
Refer to Lecture 3.

The Main FDTD Loop

**Record E at Boundary**
\[ e_1 = e_1, \quad e_1 = E_{\text{in}}^e \]

**Record H at Boundary**
\[ h_0 = h_0, \quad h_0 = H_{\text{in}}^h \]

**Update H (Perfectly Absorbing Boundary)**
\[ H_{\text{in}}^h + \Delta\frac{\partial H}{\partial t} \big|_{x=0} = -H_{\text{out}}^h \big|_{x=0} \quad k \leq k_1 \]
\[ H_{\text{out}}^h \big|_{x=0} = H_{\text{in}}^h \big|_{x=0} \quad k > k_1 \]

**Update E (Perfectly Absorbing Boundary)**
\[ E_{\text{in}}^e + \Delta\frac{\partial E}{\partial t} \big|_{x=0} = -E_{\text{out}}^e \big|_{x=0} \quad k \leq k_1 \]
\[ E_{\text{out}}^e \big|_{x=0} = E_{\text{in}}^e \big|_{x=0} \quad k > k_1 \]

**Handle H Source**
\[ H_{\text{in}}^h \big|_{x=0} = H_{\text{in}}^h \big|_{x=0} - \frac{m_0}{\Delta t} E_{\text{in}}^e \]

**Handle E Source**
\[ E_{\text{in}}^e \big|_{x=0} = E_{\text{in}}^e \big|_{x=0} - \frac{m_0}{\Delta t} H_{\text{in}}^h \]

**Update Fourier Transforms**
\[ \tilde{F}_{x=0} = \tilde{F}_{x=0} + \Delta t \left( K \right) \tilde{F}_{x=0} \]

**Visualize Simulation**
- Superimpose fields on materials
- Show reflectance, transmittance and conservation
- Update only after some number of iterations
Post Processing

Compute Response

\[ R(f) = \left( \frac{E_{\text{ref}}}{\text{FFT}[E_{\text{src}}(t)]} \right)^2 \]
\[ T(f) = \left( \frac{E_{\text{src}}}{\text{FFT}[E_{\text{ref}}(t)]} \right)^2 \]
\[ C(f) = R(f) + T(f) \]

Visualize Results
- Superimpose fields on materials
- Show reflectance, transmittance and conservation
- Show response on linear and dB scale

Done? yes no

Finished!

Sequence for Code Development
Step 1 – Basic FDTD Algorithm

- Basic update equations

Step 2 – Add Soft Source

- Basic update equations
- Add a soft source
### Step 3 – Add Absorbing Boundary

- Basic update equations
- Add a soft source
- Add perfect boundary condition

![Graph](image1.png)

### Step 4 – Add TF/SF

- Basic update equations
- Add a soft source
- Add perfect boundary condition
- Incorporate TF/SF “one-way” source

![Graph](image2.png)
Step 5 – Move Source & Add T/R

- Basic update equations
- Add a soft source
- Add perfect boundary condition
- Incorporate TF/SF “one-way” source
- Move position of source
- Calculate transmittance and reflectance

Step 6 – Add Device (Complete Algorithm)

- Basic update equations
- Add a soft source
- Add perfect boundary condition
- Incorporate TF/SF “one-way” source
- Move position of source
- Calculate transmittance and reflectance
- Add a real device
Summary of Code Development Sequence

Step 1 – Implement basic FDTD algorithm

Step 2 – Add the source

Step 3 – Add absorbing boundary

Step 4 – Add “one-way” source

Step 5 – Calculate transmittance and reflectance

Step 6 – Add a device

FDTD Analysis Walkthrough
Outline of Steps for FDTD Analysis

• Step 1: Define problem
  – What device are you modeling?
  – What is its geometry?
  – What materials is it made of?
  – What do you want to learn about the device?

• Step 2: Initialize FDTD
  – Compute grid resolution
  – Assign materials values to points on the grid
  – Compute time step
  – Initialize Fourier transforms

• Step 3: Run FDTD

• Step 4: Post-process the data

Step 1: Define the Problem

What device are you modeling? – A dielectric slab
What is its geometry? – 1 foot thick slab
What materials it is made from? – $\mu_r = 2.0$, $\varepsilon_r = 6.0$ (outside is air)
What do you want to learn? – reflectance and transmittance from 0 to 1 GHz
Step 2: Compute Grid (1 of 2)

Initial Grid Resolution (Wavelength)

\[ N_\lambda = 20 \]
\[ n_{\text{max}} = \sqrt{\mu_r \varepsilon_r} = \sqrt{(2.0)(6.0)} = 3.46 \]
\[ \lambda_{\text{min}} = \frac{c_0}{f_{\text{max}} n_{\text{max}}} = \frac{299792458}{(1.0 \text{ GHz})(3.46)} = 8.6543 \text{ cm} \]
\[ \Delta \lambda = \frac{\lambda_{\text{min}}}{N_\lambda} = \frac{8.6543 \text{ cm}}{20} = 0.4327 \text{ cm} \]

Initial Grid Resolution (Structure)

\[ N_d = 4 \]
\[ \Delta d = \frac{d}{N_d} = \frac{30.48 \text{ cm}}{4} = 7.6200 \text{ cm} \]

Initial Grid Resolution (Overall)

\[ \Delta z' = \text{min} [\Delta \lambda, \Delta d] = 0.4327 \text{ cm} \]

Step 2: Compute Grid (2 of 2)

Snap Grid to Critical Dimension(s)

The number of grid cells representing the thickness of the dielectric slab is

\[ N' = \frac{d'}{\Delta z'} = \frac{30.48 \text{ cm}}{0.4327 \text{ cm}} = 70.44 \text{ cells} \]

It is impossible to represent the thickness of the slab exactly with this grid resolution.

To represent the thickness of the slab exactly, we round \( N' \) up to the nearest integer and then calculate the grid resolution based on this quantity.

\[ N = \text{round up} \left[ N' \right] = 71 \text{ cells} \]
\[ \Delta z = \frac{d'}{N} = \frac{30.48 \text{ cm}}{71} = 0.4293 \text{ cm} \]
Step 2: Build Device on the Grid (1 of 2)

Determine Size of Grid

We need to have enough grid cells to fit the device being modeled, some space on either side of the device (10 cells for now), and cells for injecting the source and recording transmitted and reflected fields.

\[ N_z = 71 + 2 \times 10 + 3 = 94 \] cells

To calculate the size of the grid, we need to include:

- Enough grid cells to fit the device being modeled.
- Some space on either side of the device (10 cells for now).
- Cells for injecting the source and recording transmitted and reflected fields.

\[ \Delta z = \frac{d}{n} \]

where \( d \) is the thickness of the slab and \( n \) is the refractive index.

Step 2: Build Device on the Grid (2 of 2)

Compute Position of Materials on Grid

\[ n_{z,1} = 2 + 10 + 1 = 13 \]

\[ n_{z,2} = n_{z,1} + \text{round}\left(\frac{d}{\Delta z}\right) - 1 = 13 + 71 - 1 = 83 \]

Add Materials to Grid

\[ \text{UR}(nz1:nz2) = ur; \]
\[ \text{ER}(nz1:nz2) = er; \]
Step 2: Initialize FDTD (1 of 2)

Compute the Time Step $\Delta t$

$$\Delta t = \frac{n_w \Delta z}{2 c_0} = \frac{(1.0)(0.4293 \text{ cm})}{2(299792458 \frac{\text{cm}}{\text{s}})} = 7.1599 \times 10^{-12} \text{ sec}$$

Compute Source Parameters $t_0$ and $\tau$

$$\tau = \frac{1}{2 f_{\text{max}}} = \frac{1}{2(1 \text{ GHz})} = 5.00 \times 10^{-10} \text{ sec}$$

$$t_0 = 6 \tau = 3.00 \times 10^{-9} \text{ sec}$$

$t_0 = 6 \tau$ Rule of thumb

Compute Number of Time Steps, $\text{STEPS}$

$$t_{\text{prop}} = \frac{n_w N_w \Delta z}{c_0} = \frac{(3.46)(94)(0.4293 \text{ cm})}{(299792458 \frac{\text{cm}}{\text{s}})} = 4.6629 \times 10^{-9} \text{ sec}$$

Time it takes a wave to propagate across the grid.

$$T = 12 \tau + 5 t_{\text{prop}} = 12 \left(5 \times 10^{-10} \text{ s}\right) + 5 \left(4.6597 \times 10^{-9} \text{ s}\right) = 2.9314 \times 10^{-8} \text{ sec}$$

$$\text{STEPS} = \text{round} \left( \frac{T}{\Delta t} \right) = 4095$$

$T = 12 \tau + 5 t_{\text{prop}}$ Rule of thumb

STEPS must be an integer

---

Step 2: Initialize FDTD (2 of 2)

Compute the Source Functions for $E_y/H_x$ Mode

$$\delta t = \frac{n_w \Delta z}{2 c_0} + \frac{\Delta t}{2} = 1.0740 \times 10^{-11} \text{ sec}$$

$$A = -\frac{\epsilon_{\text{src}}}{\mu_{\text{src}}} = -\frac{1.0}{1.0} = -1$$

$$E_y(t) = \exp\left[-\left(\frac{t - t_0}{\tau}\right)^2\right]$$

$$H_x(t) = A \exp\left[-\left(\frac{t - t_0 + \delta t}{\tau}\right)^2\right]$$

% COMPUTE GAUSSIAN SOURCE FUNCTIONS

t = [0:STEPS-1]*dt; %time axis
delt = nsrc*dz/(2*c0) + dt/2; %total delay between E and H
A = -sqrt(ersrc/ursrc); %amplitude of H field
Ersrc = exp(-((t-t0).^2)); %E field source
Hsrc = A*exp(-((t-t0+delt)/tau).^2); %H field source

Initialize the Fourier Transforms

% INITIALIZE FOURIER TRANSFORMS
NFREQ = 100;
FREQ = linspace(0.1*gigahertz,NFREQ);
K = exp(-i*2*pi*dt.*FREQ);
REF = zeros(1,NFREQ);
TRN = zeros(1,NFREQ);
SRC = zeros(1,NFREQ);

% compute frequency
Frequency, $f$ | Kernel, $\lambda$
|-----------------|-----------------
| 0.0 MHz | 1.0 |
| 50.5 MHz | 1.0 - $i0.0023$ |
| 202.0 MHz | 1.0 - $i0.0091$ |
| 747.5 MHz | 0.9994 - $i0.0336$ |
| 1.0 GHz | 0.9990 - $i0.0450$ |
Step 3: Run FDTD (3 of 3)

Step 4: Post-Process the Data

Normalize the Data to the Source Spectrum

$$R(f) = \left( \frac{F_{ref}(f)}{\text{FFT}\left[E_{src}(t)\right]} \right)^2$$

$$T(f) = \left( \frac{F_{trn}(f)}{\text{FFT}\left[E_{src}(t)\right]} \right)^2$$

$$C(f) = R(f) + T(f)$$

% COMPUTE REFLECTANCE
% AND TRANSMITTANCE
REF = abs(REF./SRC).^2;
TRN = abs(TRN./SRC).^2;
CON = REF + TRN;