Instructor
Dr. Raymond Rumpf
(915) 747-6958
rcrumpf@utep.edu

EE 5337
Computational Electromagnetics (CEM)

Lecture #13
Formulation of Finite-Difference Frequency-Domain

Outline
• Periodic boundary conditions
• Matrix wave equations
• Plane wave source by total-field/scattered-field
• Formulation summary and examples
• Calculating transmittance and reflectance
From Last Time

\[ \frac{\partial}{\partial y} E_z - \frac{\partial}{\partial z} E_y = \mu_0 \hat{H}_x \]
\[ \frac{\partial}{\partial z} E_x - \frac{\partial}{\partial x} E_z = \mu_0 \hat{H}_y \]
\[ \frac{\partial}{\partial x} E_y - \frac{\partial}{\partial y} E_x = \mu_0 \hat{H}_z \]

\[ \frac{\partial}{\partial y} \hat{H}_x - \frac{\partial}{\partial z} \hat{H}_y = \varepsilon_{\infty} E_z \]
\[ \frac{\partial}{\partial z} \hat{H}_y - \frac{\partial}{\partial x} \hat{H}_z = \varepsilon_{\infty} E_x \]
\[ \frac{\partial}{\partial x} \hat{H}_z - \frac{\partial}{\partial y} \hat{H}_x = \varepsilon_{\infty} E_y \]

**Sign Convention for FDFD**

\[ e^{+ jkz} \]
Periodic Boundary Conditions

Problem with the PBC Discussed Until Now

Suppose we wish to simulate a periodic structure with the source wave incident at an angle. Is the problem still periodic?

Field is NOT continuous
Revisiting Bloch’s Theorem

Waves in a periodic structure take on the same symmetry and periodicity as the structure itself.

Waves in a periodic structure obey Bloch’s theorem.

\[ \vec{E}(\vec{r}) = \vec{A}(\vec{r}) e^{i\vec{\beta} \cdot \vec{r}} \]

- **Overall \( E \) Field**: Combines the periodic envelope with the “plane wave” like phase term.
- **Periodic Envelope, \( A \)**: It is this part that takes on the same symmetry and period as the structure the wave is in.
- **“Plane Wave” Phase Term**: This has the same form as a plane wave with wave vector \( \vec{\beta} \). This incorporates an overall “phase tilt.”

Revised Periodic Boundary Condition

Suppose we have the following grid...

We can write expressions for \( E_i \) at each point using Bloch’s theorem.

\[ E_i = A_i e^{i\beta x_i} \]

We need to relate \( E_1 \) to \( E_6 \) in a way that is consistent with Bloch’s theorem.

\[ E_1 = A_1 e^{i\beta x_1} \]
\[ E_6 = A_6 e^{i\beta x_6} \]

Solve for \( A_1 \),
\[ A_1 = E_1 e^{-i\beta x_1} \]

Bloch’s theorem states that \( A_1 = A_6 \).

Replace \( A_i \) with expression from above.

\[ E_6 = E_1 e^{i\beta (x_6 - x_1)} \]

We now have our revised boundary condition.

We recognize that \( \Lambda_x = x_6 - x_1 \).
The Final Periodic Boundary Condition

We can think of the grid this way

$$E_x - E_z = e^{i\omega t} E_x - E_z$$

For FDFD analysis, $$\tilde{\beta} = k_{inc}$$

$$D_x^e = \frac{1}{\Delta x}$$

|   -1 1 0 0 0 |
| 0 -1 1 0 0 |
| 0 0 -1 1 0 |
| 0 0 0 -1 1 |
| $$e^{j k_{inc} \Delta x}$$ 0 0 0 -1 |

Names for this BC
- Periodic BC
- Pseudo-periodic BC
- Floquet BC
- Bloch-Floquet BC

2D Derivative Operators for 1D Grids With Source Wave at an Oblique Angle

When $$N_x=1$$ and $$N_y>1$$

$$D_x^r = D_y^b = jk_{x,inc} I$$

$$D_x^r$$ and $$D_y^b$$ are standard for 1D grid

$$D_x^r = D_y^b =$$

$$\begin{bmatrix}
jk_{x,inc} & 0 \\
0 & jk_{x,inc} \\
\cdots & \cdots \\
0 & jk_{x,inc}
\end{bmatrix}$$

When $$N_x>1$$ and $$N_y=1$$

$$D_x^r$$ and $$D_y^b$$ are standard for 1D grid

$$D_x^r = D_y^b = jk_{y,inc} I$$

$$D_x^r = D_y^b =$$

$$\begin{bmatrix}
jk_{y,inc} & 0 \\
0 & jk_{y,inc} \\
\cdots & \cdots \\
0 & jk_{y,inc}
\end{bmatrix}$$
USE SPARSE MATRICES!!!!!!

WARNING !!

The derivative operators will be EXTREMELY large matrices.

For a small grid that is just 100 x 200 points:

- Total Number of Points: 20,000
- Size of Derivate Operators: 20,000 x 20,000
- Total Elements in Matrices: 400,000,000
- Memory to Store One Full Matrix: 6 Gb
- Memory to Store One Sparse Matrix: 1 Mb

NEVER AT ANY POINT should you use FULL MATRICES in the finite-difference method. Not even for intermediate steps. NEVER!

---

Header for yeeder() Function

```matlab
function [DEX, DEY, DHX, DHY] = yeeder(NGRID, RES, BC, kinc)
%
% YEEDER Construct Yee Grid Derivative Operators on a 2D Grid
%
% [DEX, DEY, DHX, DHY] = yeeder(NGRID, RES, BC, kinc);
% Note for normalized grid, use this function as follows:
% [DEX, DEY, DHX, DHY] = yeeder(NGRID, k0*RES, BC, kinc/k0);
%
% Input Arguments
% ----------------
% NGRID [Nx Ny] grid size
% RES [dx dy] grid resolution of the 1X grid
% BC [xbc ybc] boundary conditions
% kinc [kx ky] incident wave vector
% This argument is only needed for periodic boundaries.
```

---
Block Diagram for yeeder() Function

- **NGRID**, **RES**, **BC**, and optional **kinc**

  - If **kinc** not given, let **kinc** = [0 0]

  - **Nx** = 1?
    - Yes: **DEX** = \(jk_{\text{inc}}\)
    - No: Initialize **DEX** as sparse
      - Place two diagonals (1's and -1's)
      - Make corrections
        - **PBC?**
          - Yes: Insert \(\exp(jk_{\text{inc}}\Delta_x)\) terms
          - No: Calculate \(D_X = -\{D_X^T\}^H\)
            - \(D_Y = -\{D_Y^T\}^H\)
            - Finished

  - **Ny** = 1?
    - Yes: **DEY** = \(jk_{\text{inc}}\)
    - No: Initialize **DEY** as sparse
      - Place two diagonals (1's and -1's)
      - Make corrections
        - **PBC?**
          - Yes: Insert \(\exp(jk_{\text{inc}}\Delta_y)\) terms
          - No: Calculate \(D_Y = -\{D_Y^T\}^H\)
            - \(D_X = -\{D_X^T\}^H\)
            - Finished

Note: The finite-difference equation for point \((nx, ny)\) on the grid is located in row \(m\) where \(m = (ny - 1)\times Nx + nx\)

---

3x3 Grid Numerical Example

- \(k_0 = 2\pi i\)
- \(\theta = 20\times \text{degrees}\)
- \(k_{\text{inc}} = k_0\times [\sin(\theta) ; \cos(\theta)]\)
- **NGRID** = [3 3]
- **RES** = [1 1]
- **BC** = [-2 -2]
- \([\text{DEX, DEY, DHX, DLY}] = \text{yeeder}(\text{NGRID, RES, BC, kinc});\)

\[
\begin{align*}
    k_{\text{inc}} &= k_{\text{inc}}\Delta_x + k_{\text{inc}}\Delta_y = 2.1490\Delta_x + 5.9043\Delta_y \\
    \Lambda_x &= N_x \cdot \Delta_x = 3 \cdot 1 = 3 \\
    \Lambda_y &= N_y \cdot \Delta_y = 3 \cdot 1 = 3 \\
    e^{\Delta_x \cdot \Lambda_x} &= e^{2.1490} = 0.9866 + i0.1630 \\
    e^{\Delta_y \cdot \Lambda_y} &= e^{5.9043} = 0.4205 - i0.9073
\end{align*}
\]

---

\[\begin{align*}
\text{DEX} &= \begin{bmatrix} -1.049 & 1.000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.000 & 1.000 & -1.049 & 1.000 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 & 0 & 0 & 0 \\
0 & 0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 & 0 & 0 \\
0 & 0 & 0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 & 0 \\
0 & 0 & 0 & 0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 \\
0 & 0 & 0 & 0 & 0 & 0 & -1.049 & 1.000 & -1.049 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.049 & 1.000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.049 \\
\end{bmatrix}
\]

\[\begin{align*}
\text{DEY} &= \begin{bmatrix} -1.049 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.000 & 1.000 & -1.049 & 1.000 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 & 0 & 0 & 0 \\
0 & 0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 & 0 & 0 \\
0 & 0 & 0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 & 0 \\
0 & 0 & 0 & 0 & 0 & -1.049 & 1.000 & -1.049 & 1.000 \\
0 & 0 & 0 & 0 & 0 & 0 & -1.049 & 1.000 & -1.049 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.049 & 1.000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.049 \\
\end{bmatrix}
\]
### 4x4 Grid Numerical Example (1 of 2)

\[
\begin{align*}
\bar{k}_{\text{inc}} &= k_{x,\text{inc}} \hat{a}_x + k_{y,\text{inc}} \hat{a}_y = 2.1490 \hat{a}_x + 5.9043 \hat{a}_y, \\
\Lambda_x &= N_x \cdot \Delta_x = 4 \cdot 1 = 4, \\
e^{\mathbf{k}_{x,\text{inc}} \cdot \Lambda_x} &= e^{2.1490 \Delta_x} = -0.6757 + i0.7372
\end{align*}
\]

\[
\begin{array}{cccc}
& -1 & 1 & 0 \\
-1 & -1 & 0 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
\end{array}
\]

\[
D'_{x,y} =
\begin{array}{cccc}
-1 & 1 & 0 & -1 \\
-1 & -1 & 0 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.68 + i0.73 & -1 & 0 & -1 \\
\end{array}
\]

### 4x4 Grid Numerical Example (2 of 2)

\[
\begin{align*}
\bar{k}_{\text{inc}} &= k_{x,\text{inc}} \hat{a}_x + k_{y,\text{inc}} \hat{a}_y = 2.1490 \hat{a}_x + 5.9043 \hat{a}_y, \\
\Lambda_y &= N_y \cdot \Delta_y = 4 \cdot 1 = 4, \\
e^{\mathbf{k}_{y,\text{inc}} \cdot \Lambda_y} &= e^{5.9043 \Delta_y} = 0.0551 - i0.9985
\end{align*}
\]

\[
\begin{array}{cccc}
& -1 & 1 & 0 \\
-1 & -1 & 0 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
\end{array}
\]

\[
D'_{x,y} =
\begin{array}{cccc}
-1 & 1 & 0 & -1 \\
-1 & -1 & 0 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
-1 & -1 & -1 & 1 \\
-0.05 - i0.99 & -1 & 0 & -1 \\
\end{array}
\]
Matrix Wave Equations

3D FDFD in Block Matrix Form

We can write our matrix equations in block matrix form.

\[
\begin{align*}
\text{D}_x^s \epsilon_z - \text{D}_y^s \epsilon_x &= \mu_y \hat{h}_z \\
\text{D}_x^s \epsilon_z - \text{D}_z^s \epsilon_y &= \mu_y \hat{h}_x \\
\text{D}_y^s \epsilon_y - \text{D}_z^s \epsilon_x &= \mu_x \hat{h}_y \\
\text{D}_y^s \hat{h}_x - \text{D}_z^s \hat{h}_y &= \varepsilon_x \epsilon_z \\
\text{D}_z^s \hat{h}_y - \text{D}_x^s \hat{h}_z &= \varepsilon_y \epsilon_x \\
\text{D}_z^s \hat{h}_y - \text{D}_y^s \hat{h}_z &= \varepsilon_y \epsilon_x \\
\end{align*}
\]

\[
\begin{align*}
\nabla' \times \vec{E} &= \left[ \mu \right] \vec{H} \\

\nabla' \times \vec{H} &= \left[ \varepsilon \right] \vec{E} \\
\end{align*}
\]
3D-FDFD Formulation

Block Matrix Form

\[
\begin{align*}
\mathbf{C}^e \mathbf{\varepsilon} &= \mathbf{[\mu]} \mathbf{\hat{h}} \\
\mathbf{C}^\varepsilon \mathbf{\hat{e}} &= \mathbf{[\varepsilon]} \mathbf{\hat{e}}
\end{align*}
\]

\[
\begin{pmatrix}
\mathbf{h} \\
\mathbf{\varepsilon}
\end{pmatrix} =
\begin{pmatrix}
\mathbf{p}_{xx} & 0 & 0 \\
0 & \mathbf{p}_{yy} & 0 \\
0 & 0 & \mathbf{p}_{zz}
\end{pmatrix}
\begin{pmatrix}
\mathbf{e} \\
\mathbf{\varepsilon}
\end{pmatrix} =
\begin{pmatrix}
\mathbf{e} \\
\mathbf{\varepsilon}
\end{pmatrix}
\]

\[
\mathbf{C} =
\begin{pmatrix}
0 & -D_x^\varepsilon & D_y^\varepsilon & D_z^\varepsilon \\
-D_x^\varepsilon & 0 & -D_y^\varepsilon & D_z^\varepsilon \\
-D_y^\varepsilon & D_x^\varepsilon & 0 & -D_z^\varepsilon \\
-D_z^\varepsilon & D_y^\varepsilon & D_x^\varepsilon & 0
\end{pmatrix}
\]

Matrix Wave Equations

\[
\begin{align*}
\left( \mathbf{C}^e \mathbf{[\varepsilon]}^{-1} - \mathbf{[\mu]} \right) \mathbf{\hat{h}} &= \mathbf{0} \\
\left( \mathbf{C}^\varepsilon \mathbf{[\mu]}^{-1} - \mathbf{[\varepsilon]} \right) \mathbf{\hat{e}} &= \mathbf{0}
\end{align*}
\]

\[
\begin{pmatrix}
\mathbf{A} \\
\mathbf{\hat{e}}
\end{pmatrix} = \mathbf{0}
\]

For 3D analysis, \( \mathbf{A} \) is usually too big to solve by simple means.

For more information on 3D FDFD analysis, see


3D FDFD Usually Too Big to Solve

Typical grid required to model a 3D device.

Number of points in grid: 40,000 points
Complex #’s for \( e_x \), \( e_y \), and \( e_z \): 120,000 complex #’s
Real #’s for \( e_x \), \( e_y \), and \( e_z \): 240,000 real floating-point #’s

Size of matrix \( \mathbf{A} \): 120k x 120k complex #’s
Number of complex elements: 14.4 billion complex #’s
Number of real elements: 28.8 billion real #’s
Memory to store full \( \mathbf{A} \): 214.6 Gb

Size of \( e_x \): 40k complex numbers
Memory for \( e_x \): 625 kb

Size of full \( \mathbf{D}_x \): 40k x 40k complex numbers
Memory for full \( \mathbf{D}_x \): 23.8 Gb
Density of \( \mathbf{D}_x \): 0.005% non zero elements
Memory for sparse \( \mathbf{D}_x \): 1.5 Mb

Size of full \( \mathbf{C}^{e_x} \): 120k x 120k complex numbers
Memory for full \( \mathbf{C}^{e_x} \): 214.6 Gb
Density of \( \mathbf{C}^{e_x} \): 0.0033% non zero elements
Memory for sparse \( \mathbf{C}^{e_x} \): 8.2 Mb

Memory for direct solution: 110 Gb
Recall Diffraction Configurations

**Planar Diffraction from a Ruled Grating**
Model can be reduced to a 2D simulation
- Simplest model
- Two independent modes

**Conical Diffraction from a Ruled Grating**
Requires full 3D simulation
- Analytically, about the same complexity as 3D
- No independent modes

Advice: Don’t bother developing a 2D code for conical diffraction. Develop a 3D code and use it for 2D.

Reducing Planar Diffraction to 2D

For problems uniform along the $z$-direction and propagation restricted to the $x$-$y$ plane,

\[ \frac{\partial}{\partial z} = 0 \quad \rightarrow \quad \mathbf{D}_z = \mathbf{D}_y = \mathbf{0} \]

\[ \mathbf{D}_x \mathbf{e}_x - \mathbf{D}_y \mathbf{e}_y = \epsilon_{xx} \mathbf{h}_x \]
\[ \mathbf{D}_x \mathbf{e}_x - \mathbf{D}_y \mathbf{e}_y = \mu_{xx} \mathbf{h}_x \]
\[ \mathbf{D}_x \mathbf{e}_y - \mathbf{D}_y \mathbf{e}_y = \mu_{yx} \mathbf{h}_x \]
\[ \mathbf{D}_x \mathbf{h}_x - \mathbf{D}_y \mathbf{h}_y = \mu_{xx} \mathbf{e}_x \]
\[ \mathbf{D}_x \mathbf{h}_y - \mathbf{D}_y \mathbf{h}_y = \epsilon_{xx} \mathbf{e}_x \]
\[ \mathbf{D}_x \mathbf{h}_x - \mathbf{D}_y \mathbf{h}_y = \mu_{yx} \mathbf{e}_y \]

Maxwell’s equation decouples into two distinct modes:

**E Mode:**
\[ \mathbf{D}_x \mathbf{h}_x - \mathbf{D}_y \mathbf{h}_y = \epsilon_{xx} \mathbf{e}_x \quad \mathbf{D}_x \mathbf{e}_x = \epsilon_{xx} \mathbf{h}_x \]
\[ \mathbf{D}_x \mathbf{e}_y = \mu_{yx} \mathbf{h}_y \]

**H Mode:**
\[ \mathbf{D}_x \mathbf{h}_x - \mathbf{D}_y \mathbf{h}_y = \mu_{xx} \mathbf{e}_x \]
\[ \mathbf{D}_x \mathbf{h}_y = \epsilon_{xx} \mathbf{e}_x \]
\[ \mathbf{D}_x \mathbf{e}_y = \mu_{yx} \mathbf{h}_y \]

Lecture 13
Matrix Wave Equation (E Mode)

The matrix equations for the \( E_z \) mode were

\[
\begin{align*}
D_y^e \mathbf{h}_y - D_y^h \mathbf{h}_y &= \mathbf{e}_{zy} e_z \\
D_y^e \mathbf{e}_x &= \mu_{xy} \mathbf{h}_x \\
-D_y^e \mathbf{e}_z &= \mu_{yz} \mathbf{h}_y
\end{align*}
\]

Following matrix algebra rules, we solve the second two equations for the \( H_x \) and \( H_y \).

\[
\mathbf{h}_x = \mu_{yx} D_y^e \mathbf{e}_x \quad \mathbf{h}_y = -\mu_{yz} D_y^e \mathbf{e}_z
\]

Now we substitute these expressions into the first matrix equation.

\[
\begin{align*}
D_y^h \mathbf{h}_y - D_y^h \mathbf{h}_x &= \mathbf{e}_{zy} e_z \\
D_y^h \left( -\mu_{yz} D_y^e \mathbf{e}_x \right) - D_y^h \left( \mu_{yx} D_y^e \mathbf{e}_x \right) &= \mathbf{e}_{zy} e_z \\
-D_y^h \mu_{yz} D_y^e \mathbf{e}_x - D_y^h \mu_{yx} D_y^e \mathbf{e}_x &= \mathbf{e}_{zy} e_z \\
D_y^h \mu_{yz} D_y^e \mathbf{e}_x + D_y^h \mu_{yx} D_y^e \mathbf{e}_x + \mathbf{e}_{zy} e_z &= 0 \\
(D_y^h \mu_{yz} D_y^e \mathbf{e}_x + D_y^h \mu_{yx} D_y^e \mathbf{e}_x + \mathbf{e}_{zy} e_z) e_z &= 0
\end{align*}
\]

\[\mathbf{A}_E e_z = 0 \quad \mathbf{A}_x = D_y^h \mu_{yz} D_y^e \mathbf{e}_x + D_y^h \mu_{yx} D_y^e \mathbf{e}_x + \mathbf{e}_{zy} e_z\]

AE = DX/URyj*DEX + DNY/URxx*DEY + ERzz;

Two Matrix Wave Equations

\[\mathbf{E} \text{ Mode}\]

\[
\mathbf{A}_E = D_x^h \mu_{yy}^{-1} D_x^e + D_y^h \mu_{xx}^{-1} D_y^e + \mathbf{e}_{zz}
\]

\[\mathbf{H} \text{ Mode}\]

\[
\mathbf{A}_H = D_x^e \mathbf{e}_{yy}^{-1} D_x^h + D_y^e \mathbf{e}_{xx}^{-1} D_y^h + \mu_{zz}
\]
For the $E$ mode, the electric field is always transverse to waves propagating in the $x$-$y$ plane.

$E$ Mode = TE Mode

For the $H$ mode, the magnetic field is always transverse to waves propagating in the $x$-$y$ plane.

$H$ Mode = TM Mode

We use the labels “TE” and “TM” when we are trying to describe the orientation of a linearly polarized wave relative to a device.

**TE/ perpendicular/s** – the electric field is polarized perpendicular to the plane of incidence.

**TM/ parallel/p** – the electric field is polarized parallel to the plane of incidence.

In the limit as $\theta \to 90^\circ$, we have

$E$, Mode = TM Mode

$H$, Mode = TE Mode

This is exactly the opposite of Framework #1 when propagation is restricted to be in the $x$-$y$ plane.
Our Matrix Equations Cannot Yet Be Solved

Solution to our matrix wave equation

\[ A_E z = 0 \quad \rightarrow \quad z = A_E^{-1} 0 = 0 \]

trivial solution!!

A source \( b \) must be incorporated.

\[ A_E z = b \quad \rightarrow \quad z = A_E^{-1} b \]

Plane Wave Source By Total-Field/Scattered-Field
The Total-Field/Scattered-Field Framework

Problem rows

Lecture 13 Slide 29

The Source Field

Compute the source as it would exist in a completely homogeneous grid.

Generated by meshgrid()

Unit amplitude plane wave

Lecture 13 Slide 30
Reshape data from a 2D grid to a 1D array, then insert the 1D array into the center diagonal of a sparse matrix.

\[ Q = \text{diag}(\text{sparse}(Q(:))); \]

Note: The TF/SF interface must completely enclose the source in order to completely describe it.

Calculating the Source Vector \( b \) (1 of 2)

Source field: \( f_{\text{src}} \)
SF masking function: \( Q \)
Source isolated to TF: \( f_{\text{tot}} = (1 - Q)f_{\text{src}} \)
Source isolated to SF: \( f_{\text{scat}} = Qf_{\text{src}} \)

Quantities that must be subtracted from TF terms to make them look like SF terms...

Quantities that must be added to SF terms to make them look like TF terms...

Combine corrections:
\[ A_{\text{sf}} - QA_{\text{tf}} + (1 - Q)A_{\text{sf}} = 0 \]

...but only from SF equations:
\[ QA_{\text{sf}} \]

...but only to TF equations:
\[ (1 - Q)A_{\text{sf}} \]

Note, we get non-zero numbers here where \( f_{\text{tot}} \) or \( f_{\text{scat}} \) fails to satisfy Maxwell's equations.
Calculating the Source Vector $b$ (2 of 2)

Our matrix equation incorporating the TF/SF corrections is

$$Ae - QAf_{\text{tot}} + (I - Q)Af_{\text{scat}} = 0$$

These are known quantities that can be brought to the right-hand side of the equation.

We now have an equation from which to determine the source vector $b$.

$$Ae = QAf_{\text{tot}} - (I - Q)Af_{\text{scat}}$$

We can write $b$ in terms of just $Q$, $A$, and $f_{\text{src}}$ by substituting in our expressions for $f_{\text{tot}}$ and $f_{\text{scat}}$.

$$b = QAf_{\text{tot}} - (I - Q)Af_{\text{scat}}$$

$$= QA(1 - Q)f_{\text{src}} - (I - Q)AQf_{\text{src}}$$

$$= (QA - AQ)f_{\text{src}}$$

Famous QAAQ Equation

All of the complexities of the total-field/scattered-field method are easily implemented using the famous QAAQ (pronounced “quack”) equation.

$$Ae = b \quad b = (QA - AQ)f_{\text{src}}$$

- Very easy to implement!
- TF and SF regions can be arbitrary.
- Works regardless of grid (i.e. uniform, nonuniform, unstructured, etc.).
- Works regardless of order of accuracy of the finite-differences.
- Should work independent of the numerical method used to construct $A$. 
Visualizing the Data in $b$

We see that $b$ is non-zero only at the interfaces between the total-field and scattered-field. This is where the finite-difference equations contain quantities from both regions and correction terms are needed to make the quantities compatible.

Strictly speaking, the source field $f_{src}$ really only needs to be calculated at the points immediately adjacent to the TF/SF interface.

Example Simulation #1
Example Simulation #2

<table>
<thead>
<tr>
<th>Materials</th>
<th>Field</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMI</td>
<td>PMI</td>
<td>PMI</td>
</tr>
<tr>
<td>n₁ = 1.0</td>
<td>Scattered Field</td>
<td>Scattered Field</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Materials</th>
<th>Field</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMI</td>
<td>PMI</td>
<td>PMI</td>
</tr>
<tr>
<td>n₁ = 3.0</td>
<td>Total Field</td>
<td>Total Field</td>
</tr>
</tbody>
</table>

Formulation Summary and Examples
FDFD Formulation Summary

Matrix wave equations

\[ \mathbf{A}_x = D_x^x \mathbf{\mu} \mathbf{D}_x^x + D_y^x \mathbf{\mu} \mathbf{D}_y^x + \mathbf{\varepsilon}_x^x \]

\[ \mathbf{A}_h = D_x^h \mathbf{\varepsilon} \mathbf{D}_x^h + D_y^h \mathbf{\varepsilon} \mathbf{D}_y^h + \mathbf{\mu}_x^h \]

Source vector

\[ \mathbf{b} = (\mathbf{Q} \mathbf{A} - \mathbf{A} \mathbf{Q}) \mathbf{f}_w \]

FDFD Matrix Problem

\[ \mathbf{A}_x \mathbf{e} = \mathbf{b} \]

\[ \mathbf{A}_h \mathbf{h} = \mathbf{b} \]

Periodic Structures in FDFD

- \( \zeta = 1.0 \)
- \( \zeta = 6.0 \)

- \( \Re \{ f_x(x,y) \} \)
- \( \Re \{ f_y(x,y) \} \)
This is a simulation of an anti-reflective waveguide termination. This required that the fundamental mode be launched into the waveguide as well as calculation of the energy reflected back into the fundamental mode.
Calculating Transmittance and Reflectance

Process of Calculating Transmittance and Reflectance

For each frequency of interest...

1. Perform FDFD simulation
2. Extract cross sections of the field on the reflected and transmitted side.
3. Calculate the wave vector components of the spatial harmonics
4. Calculate the complex amplitude of the spatial harmonics
5. Calculate the diffraction efficiency of the spatial harmonics
6. Calculate over all reflectance and transmittance
7. Calculate energy conservation.
Step 1: Perform FDFD Simulation

TF/SF Interface
Periodic BC
Spacer
Spacer
Reflection record plane
Transmission record plane

Step 2: Calculate Wave Vector Components

Transverse Components

\[ k_x(m) = k_{x,\text{inc}} = \frac{2\pi m}{\Lambda_x} \]

\[ m = \text{floor} \left( \frac{N_x}{2} \right), \cdots, -2, -1, 0, 1, 2, \cdots, \text{floor} \left( \frac{N_x}{2} \right) \]

Note: transverse wave vector components are the same throughout the entire grid.

Longitudinal Components

\[ k_{y,\text{ref}}(m) = -\sqrt{\left(k_{0,\text{ref}}\right)^2 - k_x^2(m)} \]

\[ k_{y,\text{sm}}(m) = \sqrt{\left(k_{0,\text{sm}}\right)^2 - k_x^2(m)} \]

Note: the longitudinal components vary throughout the grid and are calculated from the dispersion relation.
Step 3: Extract Reflected and Transmitted Fields

- Extract Reflected and Transmitted Fields

Step 4: Remove Phase Tilt

- Remove Phase Tilt

Field calculated from FDFD

Phase across grid due to angle of incidence

Field with phase tilt removed. Bloch amplitude.

\[ E_{\text{ref}}(x) = A_{\text{ref}}(x)e^{jk_{y,\text{inc}}x} \]

\[ E_{\text{trn}}(x) = A_{\text{trn}}(x)e^{jk_{y,\text{inc}}x} \]
### Step 5: Calculate Amplitudes of Spatial Harmonics

Note: This operation is performed for every frequency of interest.

\[
E(x, y) = \sum_{m=-\infty}^{\infty} S(m)e^{i[x(m)e^{-y(m)}]}
\]

\[
\text{FFT} \left[ \begin{bmatrix} S_{-M} & \cdots & S_{-2} & S_{-1} & S_{0} & S_{1} & S_{2} & \cdots & S_{M} \end{bmatrix} \right]
\]

### Step 6: Calculate Diffraction Efficiencies

Notes

1. This calculation is performed for every frequency of interest.
2. \( |\tilde{S}_{\text{inc}}(m)| \) is the amplitude of the source. This is usually set equal to 1.0 and dropped from the above equations.
3. We can neglect the negative sign in the \( R_{DE}(m) \) equation by also ignoring the negative sign when calculating \( k_{y,\text{ref}}(m) \).

\[
R_{DE}(m) = \left| \frac{\tilde{S}_{\text{ref}}(m)}{\tilde{S}_{\text{inc}}(m)} \right|^2 \frac{\text{Re}\left[-k_{y,\text{ref}}(m)/\mu_{\text{inc}}\right]}{\text{Re}\left[k_{y,\text{inc}}/\mu_{\text{inc}}\right]}
\]

\[
T_{DE}(m) = \left| \frac{\tilde{S}_{\text{inc}}(m)}{\tilde{S}_{\text{inc}}(m)} \right|^2 \frac{\text{Re}\left[k_{y,\text{inc}}/\mu_{\text{inc}}\right]}{\text{Re}\left[k_{y,\text{inc}}/\mu_{\text{inc}}\right]}
\]

This equation assumes the \( E \) mode was solved.
Step 7: Reflectance and Transmittance

Reflectance is the total fraction of power reflected from a device. Therefore, it is equal to the sum of all the reflected modes.

\[ R = \sum_{N_r} R_{DE} (m) \]

Transmittance is the total fraction of power transmitted through a device. Therefore, it is equal to the sum of all the transmitted modes.

\[ T = \sum_{N_r} T_{DE} (m) \]

Reflectance and Transmittance on a Decibel Scale

\[ R_{db} = 10 \log_{10} R \quad T_{db} = 10 \log_{10} T \]

Be careful NOT to use \[20 \log_{10} \]!

Step 8: Calculate Power Conservation

Assuming you have not include loss or gain into your simulation, the reflectance plus transmittance should equal 100%.

\[ R + T = 100\% \]

It is ALWAYS good practice to calculate this total to check for conservation of power. This may deviate from 100% when:

- The boundary conditions are not working properly and need to be corrected.
- Rounding errors are too severe and greater grid resolution is needed.
- You have included loss or gain into your materials.

The general equation for conservation must also include the absorptance \( A \).

\[ A + R + T = 100\% \]