FDTD D-H FORMULATION

The E-H formulation we have been using is ideal for learning the basics, but has the drawback that mixes the iterative equations with the material model. The D-H formulation allows for separate treatment of these two aspects.

Starting from

\[
\begin{align*}
\frac{\partial \mathbf{D}}{\partial t} &= \nabla \times \mathbf{H} \\
\mathbf{D}(\omega) &= \varepsilon_0 \varepsilon'(\omega) \cdot \mathbf{E}(\omega) \\
\frac{\partial \mathbf{H}}{\partial t} &= -\frac{1}{\mu_0} \nabla \times \mathbf{E}
\end{align*}
\]

where \( \mathbf{D} \) is the electric flux density vector, and

\[
\varepsilon'(\omega) = \varepsilon_f + \frac{\varepsilon' - \varepsilon_f}{\text{Im} \varepsilon_0}
\]

(this is just one of the possible formulations for \( \varepsilon' \)).

As before, we normalize the \( \mathbf{E} \) (and \( \mathbf{D} \)) field by

\[
\mathbf{E} = \sqrt{\frac{\varepsilon_0}{\mu_0}} \cdot \tilde{\mathbf{E}}
\]

\[
\mathbf{D} = \sqrt{\frac{1}{\varepsilon_0 \mu_0}} \cdot \tilde{\mathbf{D}}
\]

which results in

\[
\begin{align*}
(1) \quad \frac{\partial \mathbf{D}}{\partial t} &= \frac{1}{\sqrt{\varepsilon_0 \mu_0}} \nabla \times \mathbf{H} \\
(2) \quad \mathbf{D}(\omega) &= \varepsilon'(\omega) \cdot \mathbf{E}(\omega) \\
(3) \quad \frac{\partial \mathbf{H}}{\partial t} &= -\frac{1}{\sqrt{\varepsilon_0 \mu_0}} \nabla \times \mathbf{E}
\end{align*}
\]
1 and 2 result in very simple finite difference expressions. But 3 is expressed in the frequency domain, and we need it in the time domain.

We can write 2 as:

\[ D(w) = E_r \cdot E(w) + \frac{V}{j\omega E_0} \]

In time domain, \( \frac{1}{j\omega} \rightarrow \text{integration, so} \)

\[ D(t) = E_r \cdot E(t) + \frac{V}{E_0} \int_0^t E(t')dt' \]

In sampled-time domain, we approximate the integral as summation over time,

\[ D^n = E_r \cdot E^n + \frac{V \cdot \Delta t}{E_0} \sum_{i=0}^{n-1} E^i \]

where \( D^n \)s are calculated at \( t = \Delta t \cdot n \).

Since for we need \( D_0 \) to calculate \( E_0 \), and vice versa, we rewrite the last expression as:

\[ D^n = E_r \cdot E^n + \frac{V \cdot \Delta t}{E_0} E^n + \frac{V \cdot \Delta t}{E_0} \sum_{i=0}^{n-1} E^i \]

And finally,

\[ E^n = \frac{D^n - \frac{V \cdot \Delta t}{E_0} \sum_{i=0}^{n-1} E^i}{E_r + \frac{V \cdot \Delta t}{E_0}} \]

Then, we can calculate \( E^n \) from the current value of \( D^n \) and previous values of \( E^i \). We define an auxiliary value

\[ I^n = \frac{V \cdot \Delta t}{E_0} \sum_{i=0}^{n} E^i \]

Note: \( I^n \) is not in units of current.
\[ E^n = \frac{D^n - I^{n-1}}{\varepsilon_f + \sigma^* d_l/\varepsilon_0} \]
\[ I^n = I^{n-1} + \frac{D^n}{\varepsilon_0} \]

WHERE 5 SIMPLY ACCUMULATES THE VALUE AT EACH TIME-STEP.

WRITING THIS IN TERMS OF COMPUTER CODE, ASSUMING OUR 1D \( E_x, D_x, H_y \) FORMULATION,

\[
\begin{align*}
D_x(k) &= D_x(k) + 0.5 \times (H_y(k-1) - H_y(k)) \\
E_x(k) &= \gamma_{ax}(k) \times (D_x(k) - I_x(k)) \\
I_x(k) &= I_x(k) + 0.5 \times \gamma_{ax}(k) \times E_x(k) \\
H_y(k) &= H_y(k) + 0.5 \times (E_x(k) - E_x(k+1))
\end{align*}
\]

WHERE

\[
\begin{align*}
\gamma_{ax}(k) &= \frac{1}{\varepsilon_0 \sigma \Delta t + (\Sigma \sigma \Delta t) / \varepsilon_0} \\
\gamma_{dx}(k) &= \frac{\Sigma \sigma \Delta t}{\varepsilon_0}
\end{align*}
\]