Computational Electromagnetics

ECE 4350 Final Extra Credit Project

1st Lucas Plant
Georgia Institute of Technology
Atlanta, GA
lucasplant@gatech.edu

***Abstract*—This is a review and demonstration of techniques used to simulate electromagnetics. We go over the Finite Difference Time Domain (FDTD) and Finite Element Method (FEM) techniques. I also include a demonstration using an open-source FDTD simulator and a mathematical analysis of the stability of certain discretization methods in time.**

I. Introduction

Maxwell's equations describe how Electromagnetic waves propagate through materials, however, there is no analytic solution for many more complicated real-world geometries. To solve these, we must turn our attention to approximate simulation techniques. Throughout the years many techniques have been developed including but not limited to FDTD, FEM, Method of Moments (MoM), and the Boundary Element Method (BEM). Multiple solvers solve Maxwell's equations in the frequency domain, which can be used to determine the eigenmodes of waveguides. Some examples of common simulation software are ANSYS HFSS, MEEP, and CST.

II. Finite Difference Time Domain

The FDTD algorithm solves Maxwell's equations by splitting up the simulation domain into many discrete points. This allows us to approximate the derivative using the limit definition:

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

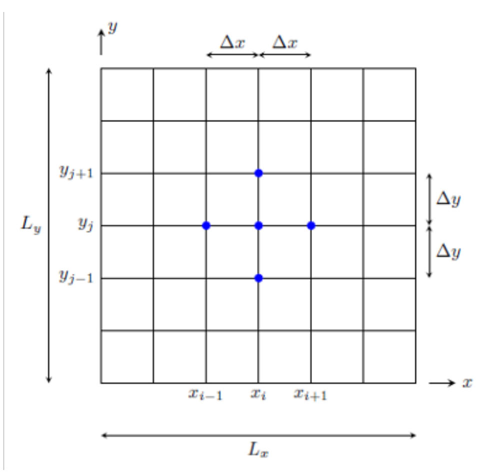

Fig. 1. Breaking up a 2D simulation domain to approximate the derivative

For finite Δx we can use a slightly different approximation to get second-order accuracy with respect to Δx:

$$f'(x) = \frac{f(x+\frac{\Delta x}{2}) - f(x-\frac{\Delta x}{2})}{\Delta x}$$

We can take this second-order approximation even further by using the Yee Lattice [1]. The Yee lattice defines Electric and Magnetic fields at staggered points that allow us to use the second-order approximation of the derivative. For example, in Maxwell's equations, the time derivative of the Electric field is proportional to the spatial derivative of the Magnetic field. Because the magnetic field is defined on either side of the electric field, we can use the difference $H(x+\frac{\Delta x}{2}) - H(x-\frac{\Delta x}{2})$ to define $\frac{dE}{dt}$ at x. In FDTD, we also stagger the electric and magnetic field calculations over time. This leads to a "leapfrog" style algorithm where we calculate the magnetic field in a half-time step in the future and then calculate the electric field. In a later section, I will show why this staggering in time is necessary using signal processing mathematics.

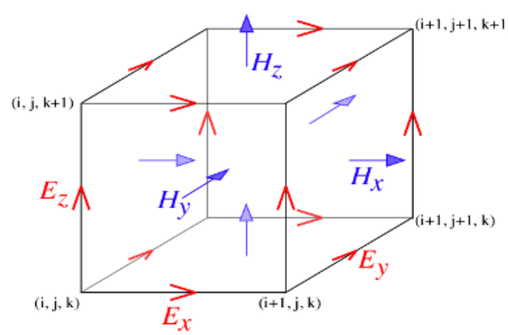

Fig. 2. Visualization of the Yee Lattice from the Meep Documentation [citation]

III. Finite Element Method

The Finite Element Method (FEM) is another commonly used technique that is more complicated than FDTD but can provide some improvements in compuation time. This technique involves splitting our simulation domain into a finite number of "Elements" or simple shapes (usually triangles). We can easily and analytically solve Maxwell's equations in these small shapes. This can turn our partial differential problem into an ordinary differential problem that can be easily solved with numerical integration techniques.

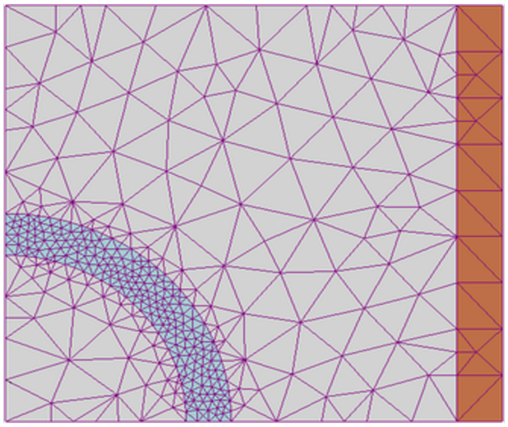

Fig. 3. FEM - Splitting the domain into many simple triangles from Wikepedia [5]

One major advantage of this technique is that the elements can vary in size based on the level of accuracy we need in one portion of the simulation domain. This allows us to have high accuracy around material transitions or complex geometries, but not waste computation power by keeping the same level of detail deep into a uniform substrate.

IV. Meep Example

Meep or MIT Electromagnetic Equation Propagation is an open-source FDTD and frequency domain solver developed by graduate students at MIT to speed up research in electromagnetics and optics by eliminating the need for labs to develop their solvers from the ground up.

I learned how to use Meep in Python and developed a basic example of simulating a wave propagating through a parallel plate waveguide. Below shows the simulation domain setup:

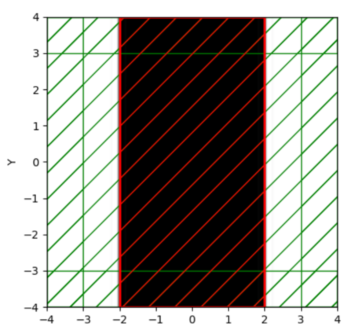

Fig. 4. Cross Section of the waveguide in the XY plane

In this simulation domain, the wave propagates in the Z direction. Black represents free space, white represents the metal that makes up the waveguide, green represents perfectly matched layers that are used to prevent reflections, and red represents our wave source.

First, let's stimulate the TEM mode with our waveguide source:

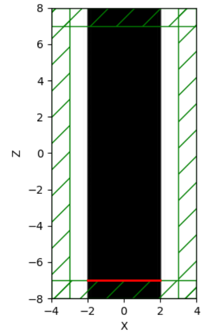

Fig. 5. Cross Section of the waveguide in the XZ plane

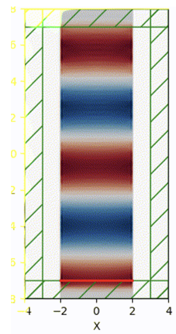

Fig. 6. The TEM mode propagating (Ex field)

By viewing the Ex field, we can easily see the oscillating TEM wave propagating. If we turn our attention to the Ez field, we can see how these time domain simulations have noise and other errors to them. We can see this as small transients when the wave first starts propagating, and we can also see a small amount of excitation of the TM1 mode.

Now, by changing our source, we can try to stimulate other modes:

We can observe how some modes have cutoff frequencies by simulating the TM2 mode. The operating frequency of the previous examples is too low to stimulate wave propagation of the TM2 mode. In the figure below, we can see this wave being heavily attenuated. When we increase the frequency, we can then observe propagation of the TM2 wave:

V. Stability analysis of FDTD

In this section, I will derive the math for a 1-dimensional FDTD simulation of a transmission line and also explain the math behind why normal time stepping is unstable.

To start off, we can model transmission lines with the telegrapher's equations:

$$\frac{\partial v(z,t)}{\partial z} = -Ri(z,t) - L\frac{\partial i(z,t)}{\partial z}$$
$$\frac{\partial i(z,t)}{\partial z} = -Gv(z,t) - L\frac{\partial v(z,t)}{\partial z}$$

R is the resistance per unit length, G is the conductance per unit length, L is the inductance per unit length, and C is the capacitance per unit length. We can now expand out our spatial derivatives as limits using the Yee Lattice:

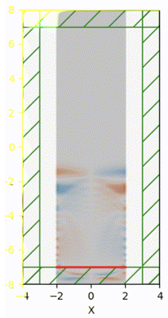

Fig. 7. Small transient when simulating the TEM wave (Ez field)

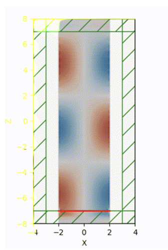

Fig. 8. The TM1 wave propagating through the waveguide (Ex field)

$$\frac{\partial i}{\partial t}(t, z+\tfrac{\Delta z}{2}) = -\frac{R}{L} i(t, z+\tfrac{\Delta z}{2}) - \frac{1}{L}\lim_{\Delta z\to 0}\left(\frac{v(t,z+\Delta z)-v(t,z)}{\Delta z}\right)$$
$$\frac{\partial v}{\partial t}(t, z) = -\frac{G}{C} v(t,z) - \frac{1}{C}\lim_{\Delta z\to 0}\left(\frac{i(t,z+\frac{\Delta z}{2})-i(t,z-\frac{\Delta z}{2})}{\Delta z}\right)$$

If we take Δz to be non-zero zero we now have a finite-dimensional continuous-time system. We can express this differential equation in state space form, which makes this easy to vectorize and simulate using linear algebra libraries in Python. To derive the A matrix for our system, consider the following setup of our state vector:

$$\vec{x}(t) = \begin{bmatrix} v(t,0) \\ v(t,\Delta z) \\ \dots \\ v(t, N\Delta z) \\ i(t, \frac{\Delta z}{2}) \\ i(t, \frac{\Delta z}{2}+\Delta z) \\ \dots \\ i(t, \frac{\Delta z}{2}+N\Delta z) \end{bmatrix}$$

We can now define our state space A matrix as a partitioned matrix:

$$\dot{\vec{x}}(t) = Ax(t)$$
$$A = \begin{bmatrix} \mathcal{G} & \mathcal{C} \\ \mathcal{L} & \mathcal{R} \end{bmatrix}$$

$\mathcal{G}$ represents the conductance of the system, $\mathcal{C}$ represents the capacitance of the system, $\mathcal{L}$ represents the inductance of the system, and $\mathcal{R}$ represents the resistance of the system. We define these submatrices as follows:

$$\mathcal{G} = -\frac{G}{C}\mathcal{I}_N$$
$$\mathcal{R} = -\frac{R}{L}\mathcal{I}_N$$

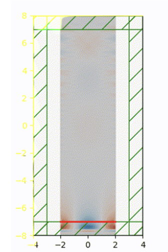

Fig. 9. Stimulation of the TM2 wave below the cutoff (Ex field)

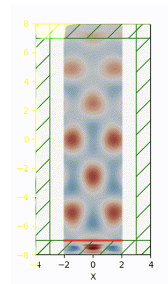

Fig. 10. Stimulation of the TM2 wave with a higher frequency (Ex field)

$$\mathcal{C} = \frac{1}{C\Delta z}\begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ -1 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ & & & \dots & & & \\ 0 & 0 & 0 & \dots & 0 & -1 & 1 \end{bmatrix}$$
$$\mathcal{L} = \frac{1}{C\Delta z}\begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ & & & \dots & & & \\ 0 & 0 & 0 & \dots & 0 & -1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 \end{bmatrix}$$

Where $\mathcal{I}_N$ is the NxN identity matrix. Now, we can use the forward Euler method to discretize in time to get a discrete-time system of the form:

$$\vec{x}(t+\Delta t) = A_d\vec{x}(t)$$
$$A_d = \begin{bmatrix} \mathcal{G}_d & \mathcal{C}_d \\ \mathcal{L}_d & \mathcal{R}_d \end{bmatrix}$$
$$\mathcal{G}_d = \Delta t\mathcal{G} + I_N,\ \mathcal{R}_d = \Delta t\mathcal{R} + \mathcal{I}_N,\ \mathcal{C}_d = \Delta t\mathcal{C}, \text{ and}$$
$$\mathcal{L}_d = \Delta t\mathcal{L}$$

Simulating this system with Python, we observe that this is unstable:

Initially, I thought that this instability would need to be solved with filtering; however, after further investigation, this error comes from the fact that we used the staggered Yee discretization in space but not time. To solve this, we can implement an algorithm that first calculates the current/magnetic field at a half-time step in the future and then calculates the voltage/electric field:

$$\vec{i}(t+\tfrac{\Delta t}{2}) = \mathcal{L}_d\vec{v}(t) + \mathcal{R}_d\vec{i}(t-\tfrac{\Delta t}{2})$$
$$\vec{v}(t+\Delta t) = \mathcal{G}_d\vec{v}(t) + \mathcal{C}_d\vec{i}(t+\tfrac{\Delta t}{2})$$

We can substitute the current update equation into the voltage equation:

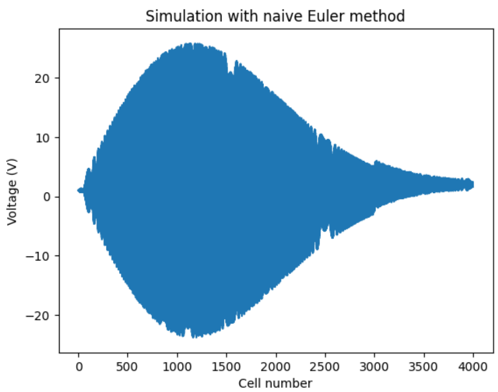

Fig. 11. Visualization of instabilities in forward Euler

$$\vec{v}(t+\Delta t) = \mathcal{G}_d\vec{v}(t) + \mathcal{C}_d(\mathcal{L}_d\vec{v}(t) + \mathcal{R}_d\vec{i}(t - \tfrac{\Delta t}{2}))$$

We can now formulate the stable discrete A matrix A_{Yee}:

$$x(t+\Delta t) = A_{Yee}x(t)$$
$$A_{Yee} = \begin{bmatrix} \mathcal{G}_d + \mathcal{C}_d\mathcal{L}_d & \mathcal{C}_d\mathcal{R}_d \\ \mathcal{L}_d & \mathcal{R}_d \end{bmatrix}$$

Simulating with the following update, we can now see that the simulation is stable:

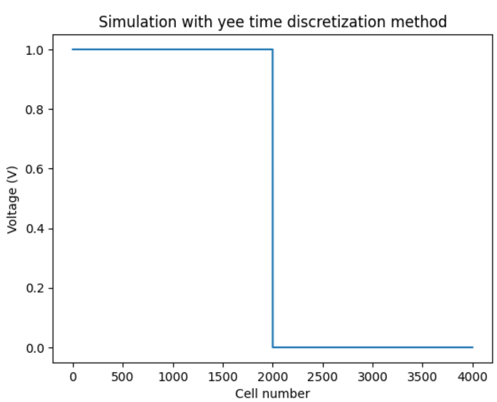

Fig. 12. A stable simulation of a step function propagating down a transmission line

To explore why forward Euler is unstable but the Yee discretization is stable, we can look at the eigenvalues of the discrete A matrix. Signal processing theory tells us that for our system to be stable, all eigenvalues must be inside the unit circle. In the figure below, we observe that the Yee discretization has all of its eigenvalues on the unit circle corresponding to pure undamped oscillatory behavior. The forward Euler discretization has a line of Eigenvalues that are mostly outside of the unit circle, leading to instability.

VI. Conclusion

Large and/or complicated simulations can be very computationally expensive and require a lot of time to run. There are, however, ways to improve this. Simplifying the simulation and recognising symmetries can help to speed up the simulation

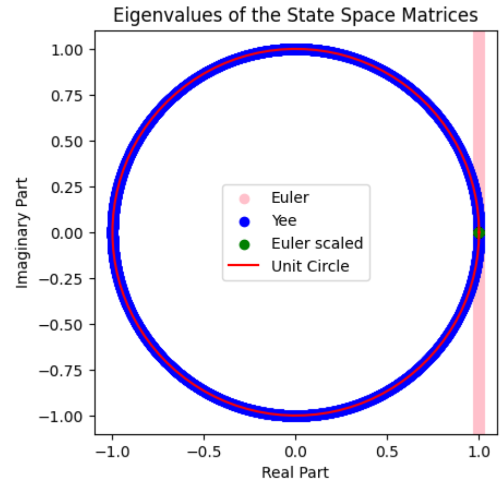

Fig. 13. Plot of the eigenvalues of the different time discretization methods

by preventing unnecessary calculations. Additionally, many simulation software, such as Meep, can integrate into high-performance computers and GPU clusters. These computers are constantly getting faster and faster, allowing engineers to perform more complex simulations.

All in all, these simulation tools have been very useful in the field of electromagnetics as they have allowed scientists and engineers to approximately solve many problems that don't have analytic solutions.

References

[1] A. Oskooi, D. Roundy, M. Ibanescu, P. Bermel, J.D. Joannopoulos, and S.G. Johnson, "MEEP: A flexible free-software package for electromagnetic simulations by the FDTD method," Computer Physics Communications, Vol. 181, pp. 687-702 (2010)

[2] Steven G. Johnson and J. D. Joannopoulos, Block-iterative frequency-domain methods for Maxwell's equations in a planewave basis, Optics Express 8, no. 3, 173-190 (2001)

[3] Mei Song Tong; Weng Cho Chew, "Computational Electromagnetics," in The Nystrom Method in Electromagnetics , IEEE, 2019, pp.75-97, doi: 10.1002/9781119284857.ch2. keywords: Wave functions;Computational electromagnetics;Maxwell equations;Dielectrics;Method of moments,