

Fields & Waves II

Project III - An Introduction to the Finite Difference Time Domain

Aaron Rosen

Team Partners: Jade Gullic & Sydney Northcutt

Oklahoma State University

Fields & Waves II - ECEN-3623

22 February 2022

A20198898

Contents

1	Introduction	3
2	Problem Set 1.1	3
2.1	1.1.1	3
2.2	1.1.2	5
3	Problem Set 1.3	6
3.1	1.3.1	6
4	Problem Set 1.4	8
4.1	1.4.1	8
4.2	1.4.2	11
4.3	1.4.3	11
5	Appendix	13
5.1	1.1.1	13
5.2	1.1.2	14
5.3	1.3.1	16
5.4	1.4.1	18
5.5	1.4.3	20

1 Introduction

This assignment will stretch your ability to program in MATLAB, and to develop a basic simulation tool that demonstrates plane wave interactions. Develop a brief report with figures and captions to demonstrate your success. Provide a brief description of your responses to questions in the problem sets. Organize your report by section numbers corresponding to the problem set numbers.

All plots must have figure numbers and captions. In a paragraph below each plot provide a discussion of the plot highlighting important characteristics. You need to find three things to say about each plot you provide. Include as an appendix, your MATLAB code used to obtain your results. Remember - although I am encouraging you to work together to get the code working this is an individual assignment and you are responsible to provide a written discussion of the problem sets above. More importantly, you need to understand what the code is telling you about the interactions and explain why the results make sense.

2 Problem Set 1.1

2.1 1.1.1

Get the program `fdld_1.1.c` running. What happens when the pulse hits the end of the array? Why?

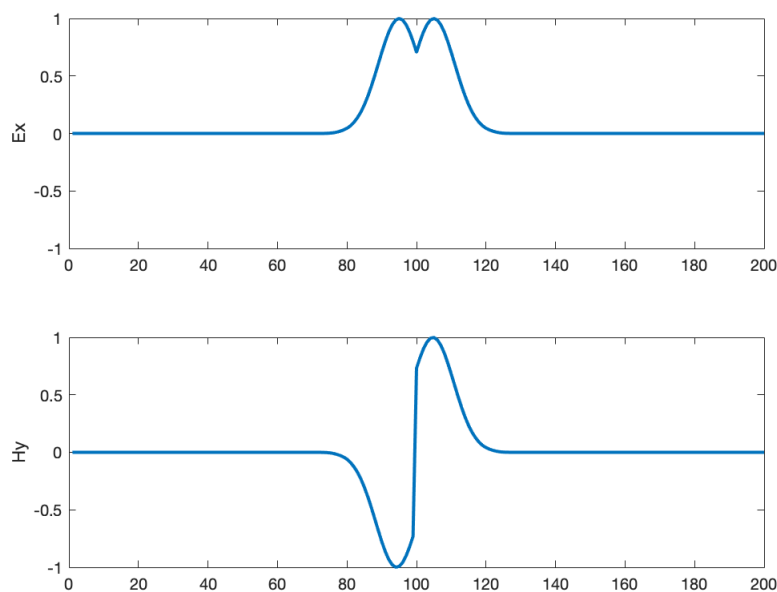


Figure 1: Gaussian Pulse Initial

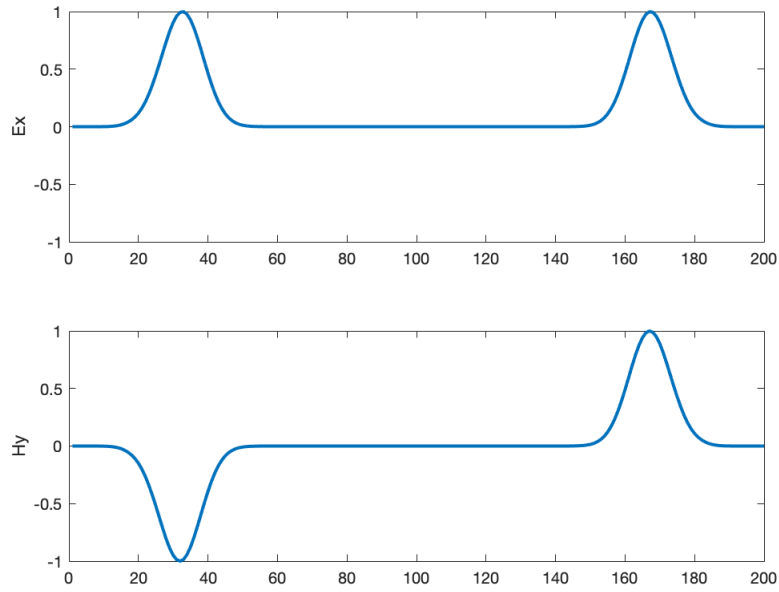


Figure 2: Gaussian Pulse After Some Time

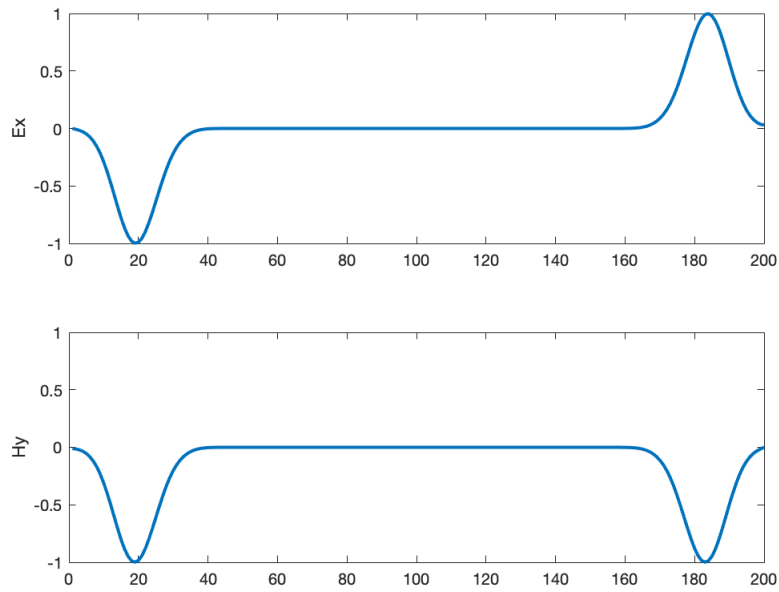


Figure 3: Gaussian Pulse After Hitting Array Boundaries

Following getting the program running, the wave of the Gaussian pulse for $NSTEPS = 500$ demonstrates the wave beginning propagation at $kc = 100$, traveling towards (0 & 200) array bounds, and reflecting. The wave acts as expected. The phase change after the array boundary/ reflection is caused by the equal and opposite force exerted by the medium on the wave.

2.2 1.1.2

Modify the program so it has two sources, one at $kc - 20$ and one at $kc + 20$ (Notice that kc is the center of the problem space). What happens when the pulses meet? Explain this from basic EM theory.

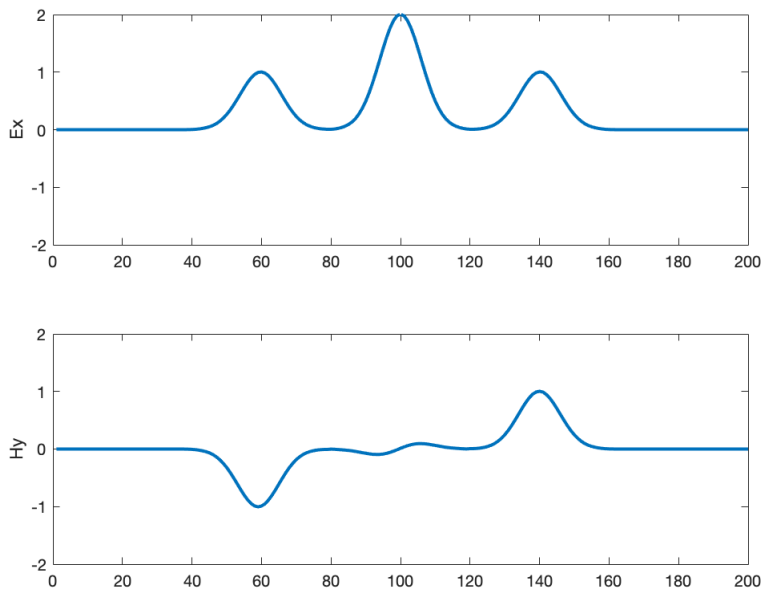


Figure 4: Gaussian Pulse Slightly After Initial

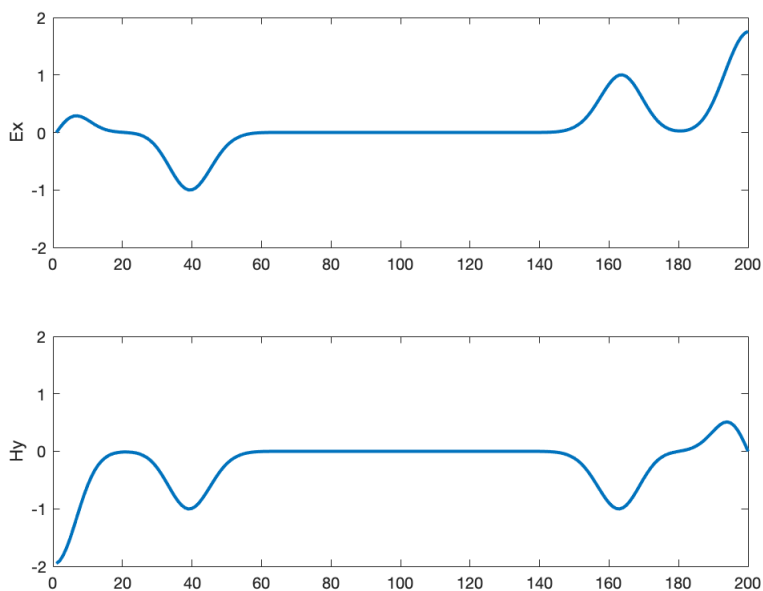


Figure 5: Gaussian Pulse After Some Time

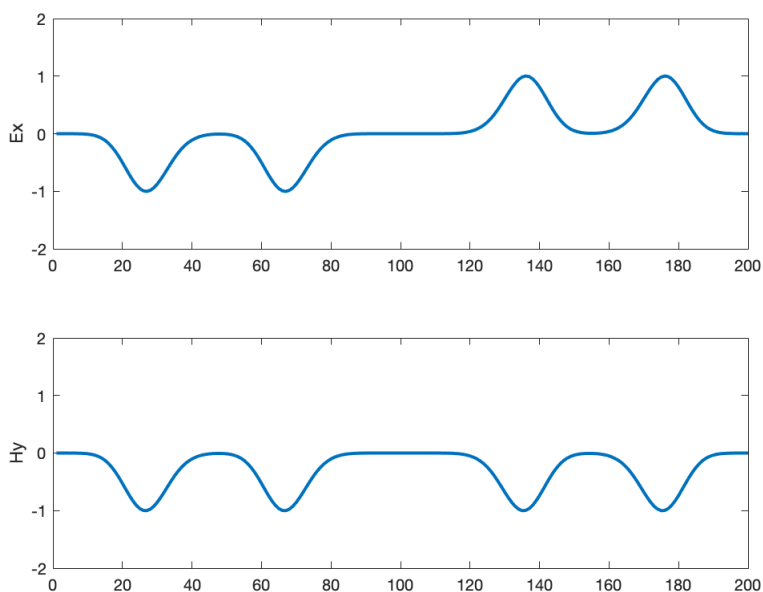


Figure 6: Gaussian Pulse After Hitting Array Boundaries

This demonstration showed two Gaussian Pulses with a width of 40 propagating. The increase in amplitude is expected because of constructive interference. Following the wave hitting the boundaries/ array limits, the wave(s) reflect, similar to problem 1.1.1.

3 Problem Set 1.3

3.1 1.3.1

The program `fdld_1.2.c` has absorbing boundary conditions at both ends. Get this program running and test it to ensure the boundary conditions are completely absorbing the pulse.

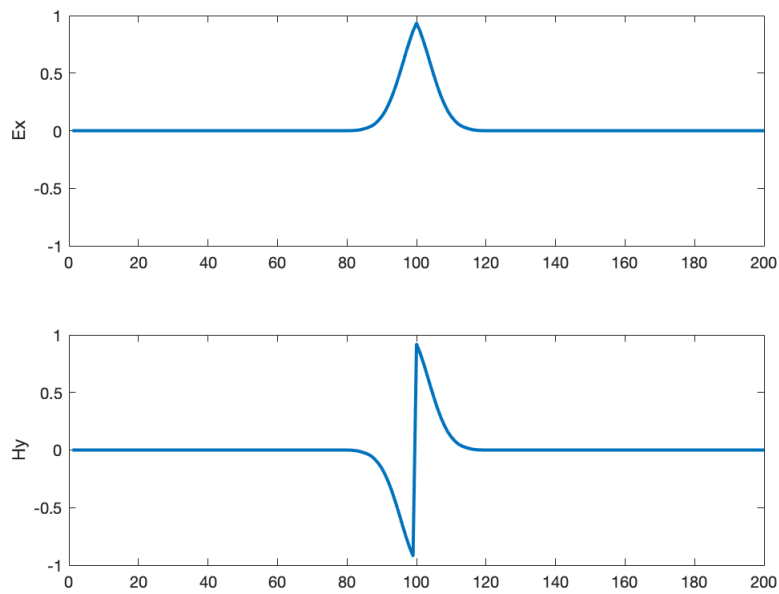


Figure 7: Gaussian Pulse Slightly After Initial

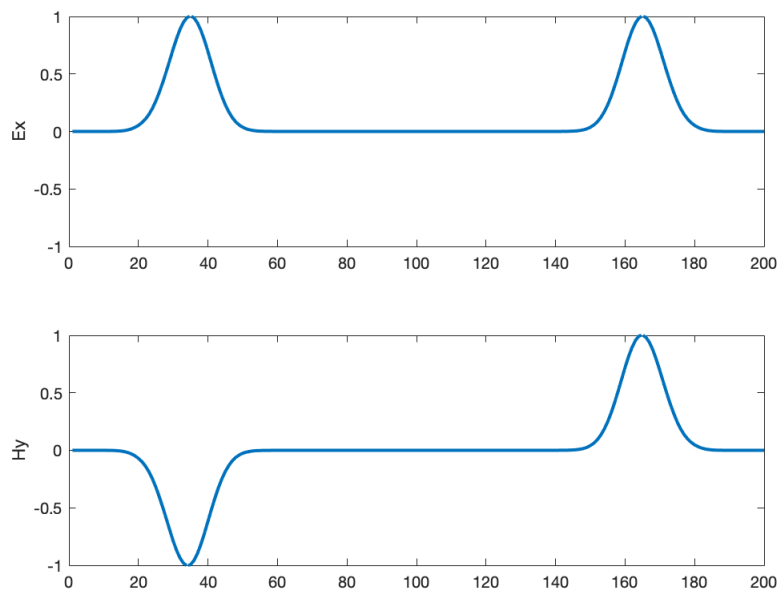


Figure 8: Gaussian Pulse After Some Time

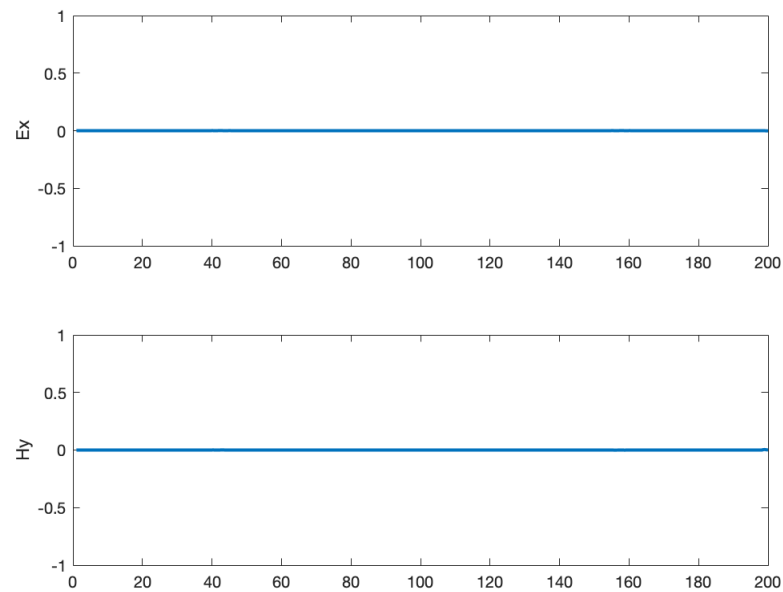


Figure 9: Gaussian Pulse After Hitting Array Boundaries

The program `fd1d_1.2` acts similarly to `fd1d_1.1`; however, in this program the Gaussian wave is absorbed by during the boundary conditions/ array limits. This absorption is shown in Figure 9 where the pulse goes to zero and remains at zero. This confirms the program is working correctly.

4 Problem Set 1.4

4.1 1.4.1

The program `fdld_1.3.c` simulates a problem partly containing free space and partly dielectric material. Run this program and duplicate the results of Fig. 1.4.

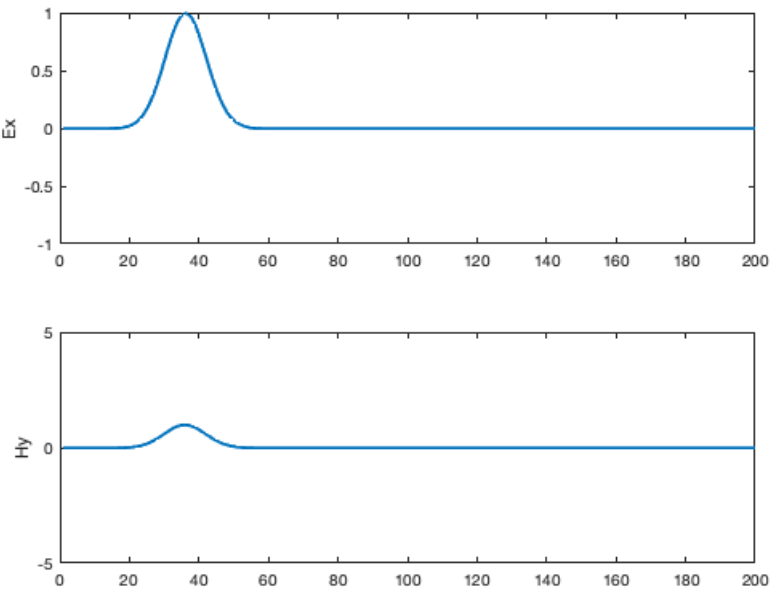


Figure 10: Gaussian Pulse @ $T = 100$

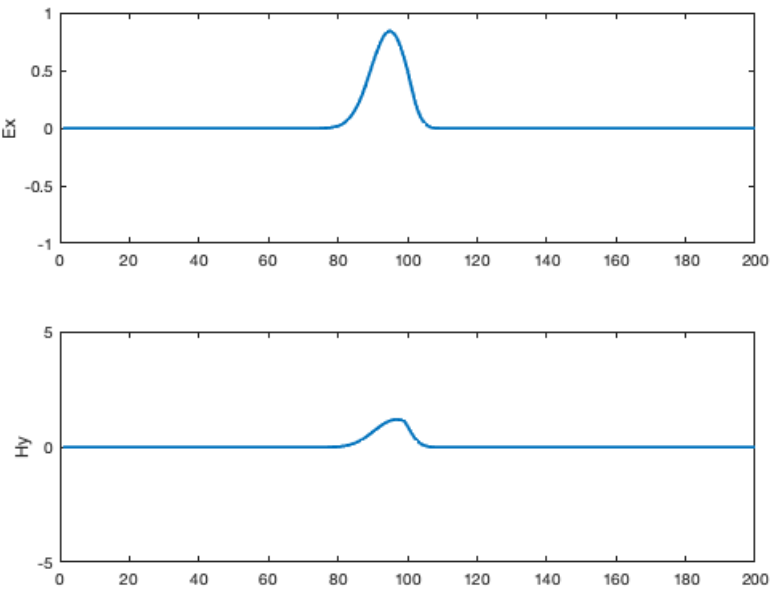


Figure 11: Gaussian Pulse @ $T = 220$

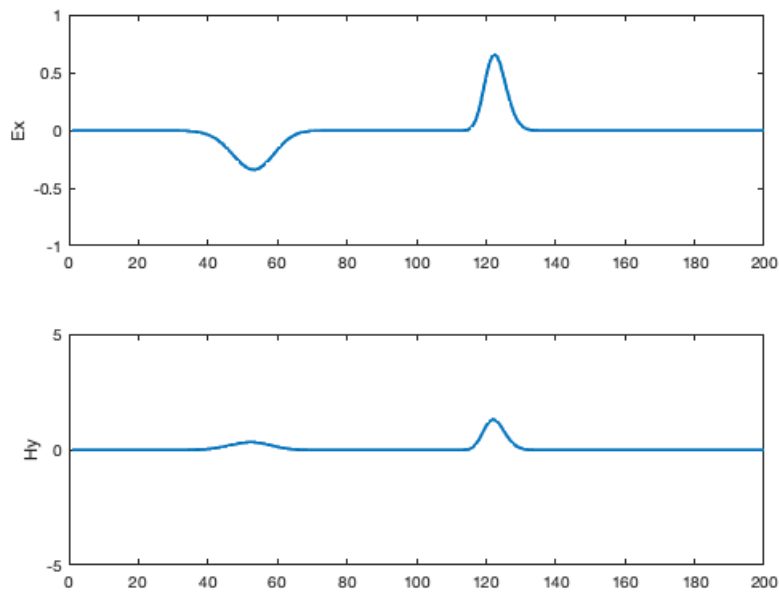


Figure 12: Gaussian Pulse @ $T = 320$

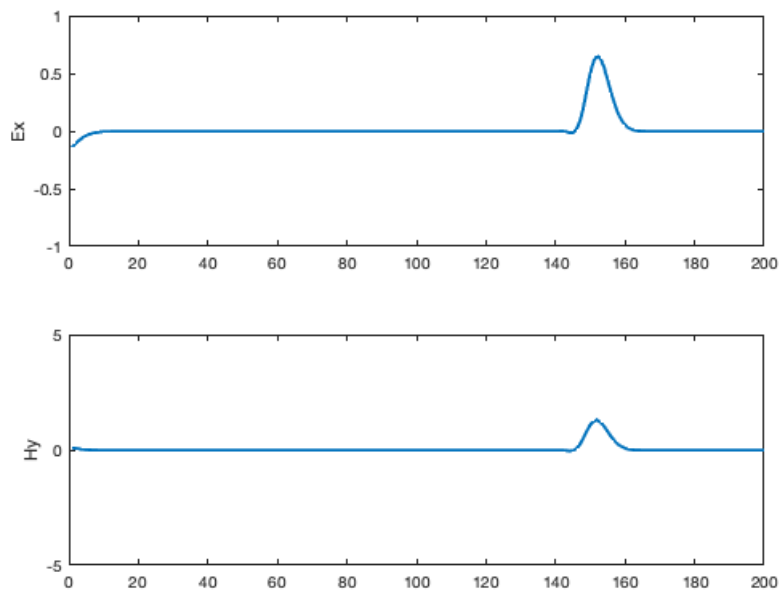


Figure 13: Gaussian Pulse @ $T = 440$

The figures [Figure 10-13] are duplicates of the images provided in the Sullivan PDF. Given these photos are the same, the code was replicated properly.

4.2 1.4.2

Look at the relative amplitudes of the reflected and transmitted pulses. Are they correct? Check them by calculating the reflection and transmission coefficients. (See the appendix at the end of this chapter.)

Reflection Coefficient

$$\Gamma = \frac{\eta_2 - \eta_1}{\eta_2 + \eta_1}$$

Transmission Coefficient

$$\tau = \frac{2\eta_2}{\eta_2 + \eta_1}$$

Calculation for Eta

$$\eta = \frac{\eta_o}{\sqrt{\epsilon_{r1}}}$$

Observed Values

$$\eta_o = 120\pi, \epsilon_{r1} = 1, \epsilon_{r2} = 4$$

Calculations

$$\eta_1 = \frac{120\pi}{\sqrt{1}} = 120\pi$$

$$\eta_2 = \frac{120\pi}{\sqrt{4}} = 60\pi$$

$$\Gamma = \frac{120\pi - 60\pi}{120\pi + 60\pi} = -\frac{1}{3}$$

$$\tau = \frac{2(60)\pi}{120\pi + 60\pi} = \frac{2}{3}$$

Following calculating these values, the wave propagation observed is correct. After hitting the second medium, the wave reflects, reduces $\frac{1}{3}$, and changes phase demonstrating the negative portion of the calculation. The wave continues propagating at a reduction of $\frac{2}{3}$ showing the transmission coefficient is also correct.

4.3 1.4.3

A type of propagating wave function that is of great interest in areas such as optics is the "wave packet," which is a sinusoidal function in a Gaussian envelope. Modify your program to simulate a wave packet.

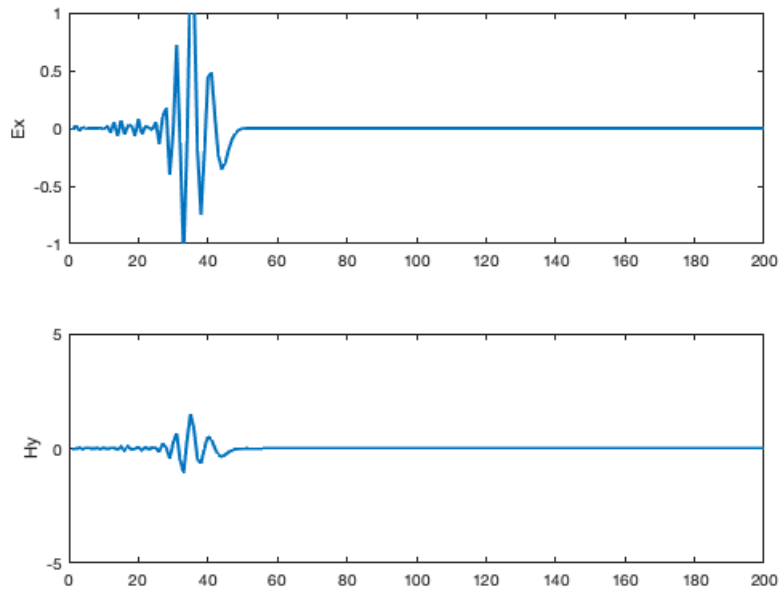


Figure 14: Wave Packet Propagation

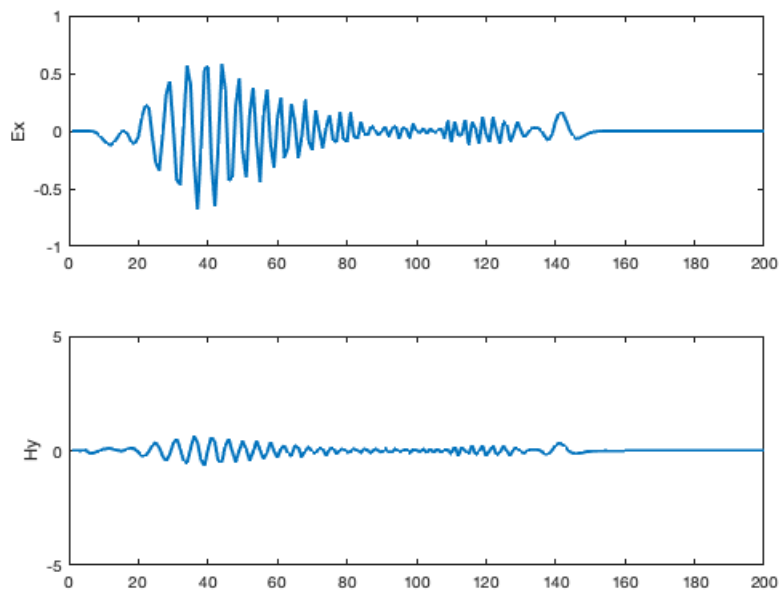


Figure 15: Wave Packet after Reflection/ Transmission off Second Medium

In Figure 14 & Figure 15 a wave packet is shown propagating. The wave packet interferes with the second medium, reflects back, and transmits. I expected more reduction in the reflection amplitude. This may be due to the wave packet having constructive interference with itself though. I also expected more transmission amplitude, but I am not sure of an explanation for this.

5 Appendix

5.1 1.1.1

```
1 clear all; close all; clc;
2
3 %Aaron Rosen — Fields & Waves II
4 %Project 3 — An Introduction to the Finite Difference Time Domain
5
6
7 %Rebuilding FD1D_1.1.c in MATLAB
8 KE = 200;
9 Ex = zeros(KE, 1);
10 Hy = zeros(KE, 1);
11 kc = KE/2; %Center of problem space
12 t0 = 40; %Center of the incident pulse
13 spread = 12; %Width of the incident pulse
14 NSTEPS = 1; %Number of Steps
15 T = 0;
16
17
18 while (NSTEPS > 0)
19     %Enter the Steps
20
21     prompt = "NSTEPS——> ";
22     NSTEPS = input(prompt);
23     disp(NSTEPS);
24     n = 0;
25
26
27     for n = 1:NSTEPS
28         T = T+1;
29         %Calculating the Ex Field
30         for k = 2:KE
31             Ex(k) = Ex(k) + 0.5*(Hy(k-1) - Hy(k));
32         end
33         %Gaussian Pulse in the middle
34         pulse = exp(-0.5*((t0-T)./spread).^2);
35         Ex(kc) = pulse;
36         disp(t0-T);
37         disp(Ex(kc));
38         %Calculate the Hy field
```

```
39         for k = 1:KE-1
40             Hy(k) = Hy(k) + 0.5*(Ex(k) - Ex(k+1));
41         end
42
43
44         figure(1)
45         subplot(2,1,1)
46         plot(Ex, 'LineWidth', 2)
47         ylim([-1 1])
48         ylabel('Ex')
49
50         subplot(2,1,2)
51         plot(Hy, 'LineWidth', 2)
52         ylim([-1 1])
53         ylabel('Hy')
54
55         for k = 1:KE
56             disp(Ex(k));
57             disp(Hy(k));
58         end
59
60
61
62         %
63     end
64 end
```

5.2 1.1.2

```
1 clear all; close all; clc;
2
3 %Aaron Rosen — Fields & Waves II
4 %Project 3 — An Introduction to the Finite Difference Time Domain
5
6
7 %Rebuilding FD1D_1.2.c in MATLAB
8 KE = 200;
9 Ex = zeros(KE, 1);
10 Hy = zeros(KE, 1);
11
12 kc = KE/2;
13 k1 = kc - 20;
14 k2 = kc + 20;
```

```
15 t0 = 40;
16 spread = 12;
17 T = 0;
18 NSTEPS = 1;
19
20 while (NSTEPS > 0)
21
22     prompt = "NSTEPS —> ";
23     NSTEPS = input(prompt);
24     disp(NSTEPS);
25
26     for n = 1:NSTEPS
27         T = T+1;
28         %Calculate the Ex Field
29         for k=2:KE
30             Ex(k) = Ex(k) + 0.5*(Hy(k-1) - Hy(k));
31         end
32
33         %Guassian Pulse
34         pulse = exp(-0.5*((t0-T)/spread).^2);
35         Ex(k1) = Ex(k1) + pulse;
36         Ex(k2) = Ex(k2) + pulse;
37         disp(t0-T);
38         disp(Ex(k1));
39         disp(Ex(k2));
40
41         %Calculate the Hy Field
42         for k=1:KE-1
43             Hy(k) = Hy(k) + 0.5*(Ex(k)-Ex(k+1));
44         end
45
46         figure(1)
47
48         subplot(2,1,1)
49         plot(Ex, 'LineWidth', 2)
50         ylim([-2 2])
51         ylabel('Ex')
52
53         subplot(2,1,2)
54         plot(Hy, 'LineWidth', 2)
55         ylim([-2 2])
56         ylabel('Hy')
```

```
57
58     for k = 1:KE
59         disp(Ex(k));
60         disp(Hy(k));
61     end
62     % writematrix(Ex, Ex);
63
64
65     %writematrix(Hy, Hy);
66
67     end
68 end
```

5.3 1.3.1

```
1 clear all; close all; clc;
2
3 %Aaron Rosen — Fields & Waves II
4 %Project 3 — An Introduction to the Finite Difference Time Domain
5
6
7 %Rebuilding FD1D_1.2c in MATLAB
8 KE = 200;
9 Ex = zeros(KE, 1);
10 Hy = zeros(KE, 1);
11
12 kc = KE/2;
13 t0 = 40;
14 spread = 12;
15 T = 0;
16 NSTEPS = 1;
17
18 ex_low_m1 = 0;
19 ex_low_m2 = 0;
20 ex_high_m1 = 0;
21 ex_high_m2 = 0;
22
23 while (NSTEPS > 0)
24     n = 0;
25     prompt = "NSTEPS——>";
26     NSTEPS = input(prompt);
27     disp(NSTEPS);
28
```



```
29     for n = 1:NSTEPS
30         T = T+1;
31
32         %Calculate the Ex Field
33         for k = 2:KE
34             Ex(k) = Ex(k) + 0.5*(Hy(k-1) - Hy(k));
35         end
36
37         %Put a Gaussian Pulse in the Middle
38         pulse = exp(-0.5*((t0-T)/spread)^2);
39         Ex(kc) = Ex(kc) + pulse;
40         disp(t0-T);
41         disp(Ex(kc));
42
43         %Define Boundary Conditions
44         Ex(1) = ex_low_m2;
45         ex_low_m2 = ex_low_m1;
46         ex_low_m1 = Ex(2);
47         Ex(KE-1) = ex_high_m2;
48         ex_high_m2 = ex_high_m1;
49         ex_high_m1 = Ex(KE-2);
50
51
52         %Calculate the Hy Field
53         for k = 1:KE-1
54             Hy(k) = Hy(k) + 0.5*(Ex(k) - Ex(k+1));
55         end
56
57         figure(1);
58         subplot(2,1,1)
59         plot(Ex, 'LineWidth', 2)
60         ylim([-1 1])
61         ylabel('Ex')
62
63         subplot(2,1,2)
64         plot(Hy, 'LineWidth', 2)
65         ylim([-1 1])
66         ylabel('Hy')
67
68         for k=1:KE
69             disp(Ex(k));
70             disp(Hy(k));
```

```
71         end
72     end
73 end
```

5.4 1.4.1

```
1  clear all; close all; clc;
2
3  %Aaron Rosen — Fields & Waves II
4  %Project 3 — An Introduction to the Finite Difference Time Domain
5
6
7  %Rebuilding  FD1D_1.3c in MATLAB
8  KE = 200;
9  Ex = zeros(KE, 1);
10 Hy = zeros(KE, 1);
11 CB = zeros(KE, 1);
12
13
14
15 kc = KE/2;
16 t0 = 40;
17 spread = 12;
18 T = 0;
19
20 NSTEPS = 1;
21
22 ex_low_m1 = 0;
23 ex_low_m2 = 0;
24 ex_high_m1 = 0;
25 ex_high_m2 = 0;
26
27 for k = 2:KE
28     CB(k) = 0.5;
29 end
30
31 die = "Dielectric starts @ —> ";
32 kstart = input(die);
33 disp(kstart);
34 eps = "Epsilon —>";
35 epsilon = input(eps);
36 disp(epsilon);
37
```

```
38 for k = kstart:KE
39     CB(k) = 0.5/epsilon;
40 end
41
42
43
44 while (NSTEPS > 0)
45     n = 0;
46     prompt = "NSTEPS---->";
47     NSTEPS = input(prompt);
48     disp(NSTEPS);
49
50     for n = 1:NSTEPS
51         T = T+1;
52
53
54
55         %Calculate the Ex Field
56         for k = 2:KE
57             Ex(k) = Ex(k) + CB(k)*(Hy(k-1) - Hy(k));
58         end
59
60
61         %Define Boundary Conditions
62         Ex(1) = ex_low_m2;
63         ex_low_m2 = ex_low_m1;
64         ex_low_m1 = Ex(2);
65         Ex(KE-1) = ex_high_m2;
66         ex_high_m2 = ex_high_m1;
67         ex_high_m1 = Ex(KE-2);
68
69
70
71         %Put a Gaussian Pulse in the Middle
72         pulse = exp(-0.5*((t0-T)/spread)^2);
73         Ex(6) = Ex(6) + pulse;
74         disp(t0-T);
75         disp(Ex(6));
76
77
78
79
```

```
80
81
82 %Calculate the Hy Field
83 for k = 1:KE-1
84     Hy(k) = Hy(k) + 0.5*(Ex(k) - Ex(k+1));
85 end
86
87
88
89
90 figure(1);
91 subplot(2,1,1)
92 plot(Ex, 'LineWidth', 2)
93 ylim([-1 1])
94 ylabel('Ex')
95
96
97
98 subplot(2,1,2)
99 plot(Hy, 'LineWidth', 2)
100 ylim([-5 5])
101 ylabel('Hy')
102
103
104
105 for k=1:KE
106     disp(Ex(k));
107
108 end
109 end
110 end
```

5.5 1.4.3

```
1 clear all; close all; clc;
2
3 %Aaron Rosen — Fields & Waves II
4 %Project 3 — An Introduction to the Finite Difference Time Domain
5
6
7 %Rebuilding FD1D_1.3c in MATLAB
8 KE = 200;
9 Ex = zeros(KE, 1);
```

```
10 Hy = zeros(KE, 1);
11 CB = zeros(KE, 1);
12
13
14
15 kc = KE/2;
16 t0 = 40;
17 spread = 12;
18 T = 0;
19
20 NSTEPS = 1;
21
22 ex_low_m1 = 0;
23 ex_low_m2 = 0;
24 ex_high_m1 = 0;
25 ex_high_m2 = 0;
26
27 for k = 2:KE
28     CB(k) = 0.5;
29 end
30
31 die = "Dielectric starts @ ——> ";
32 kstart = input(die);
33 disp(kstart);
34 eps = "Epsilon ---->";
35 epsilon = input(eps);
36 disp(epsilon);
37
38 for k = kstart:KE
39     CB(k) = 0.5/epsilon;
40 end
41
42
43
44 while (NSTEPS > 0)
45     n = 0;
46     prompt = "NSTEPS---->";
47     NSTEPS = input(prompt);
48     disp(NSTEPS);
49
50     for n = 1:NSTEPS
51         T = T+1;
```

```
52
53
54
55 %Calculate the Ex Field
56 for k = 2:KE
57     Ex(k) = Ex(k) + CB(k)*(Hy(k-1) - Hy(k));
58 end
59
60
61
62
63 %Put a Gaussian Pulse in the Middle
64 %pulse = exp(-0.5*((t0-T)/spread)^2);
65 %Ex(6) = Ex(6) + pulse;
66 %disp(t0-T);
67 %disp(Ex(6));
68
69 %wave packet
70 if T<20
71     pulse2 = (sin((T)*2*pi) - sin((T)*1.1*2*pi));
72     Ex(6) = (Ex(6) + pulse2);
73
74 else
75
76     Ex(6) = 0;
77     Ex(k) = 0;
78
79 end
80
81
82 %Define Boundary Conditions
83 Ex(1) = ex_low_m2;
84 ex_low_m2 = ex_low_m1;
85 ex_low_m1 = Ex(2);
86 Ex(KE-1) = ex_high_m2;
87 ex_high_m2 = ex_high_m1;
88 ex_high_m1 = Ex(KE-2);
89
90
91
92
93
```

```
94
95
96     %Calculate the Hy Field
97     for k = 1:KE-1
98         Hy(k) = Hy(k) + 0.5*(Ex(k) - Ex(k+1));
99     end
100
101
102
103
104     figure(1);
105     subplot(2,1,1)
106     plot(Ex, 'LineWidth', 2)
107     ylim([-1 1])
108     ylabel('Ex')
109
110
111
112     subplot(2,1,2)
113     plot(Hy, 'LineWidth', 2)
114     ylim([-5 5])
115     ylabel('Hy')
116
117
118
119     for k=1:KE
120         disp(Ex(k));
121
122     end
123 end
124 end
```