



**Electrical Engineering Department,  
Fourth Year - Communications & Electronics.**

# **EE 466 ANTENNA**

## **Lab Assignment-2**

| PREPARED BY                  | SECTION | SEAT.NO. |
|------------------------------|---------|----------|
| Mahmoud Mohamed Kamal Ismail | 7       | 250      |
| Mahmoud AbdElHady Mahmoud    | 7       | 248      |
| Ahmed Mohamed Abdelakher     | 1       | 37       |

## ➤ Contents

|  |                |
|--|----------------|
| <b>1. Code</b>   | <b>Page 3</b>  |
| 1.1. Part 1: linear antenna (dipole of general length) | Page 3         |
| 1.2. Part 2: Uniform linear antenna array (ULA)        | Page 4         |
| 1.3. Part 3: Nonuniformly-Fed linear antenna array     | Page 5         |
| A. Binomial Arrays                                     | Page 5         |
| B. Dolph-Tschebysceff Arrays                           | Page 6         |
| <b>2. Examples</b>                                     | <b>Page 7</b>  |
| 2.1. Part 1: linear antenna (dipole of general length) | Page 7         |
| 2.2. Part 2: Uniform linear antenna array (ULA)        | Page 9         |
| 2.3. Part 3: Nonuniformly-Fed linear antenna array     | Page 10        |
| A. Binomial Arrays                                     | Page 10        |
| B. Dolph-Tschebysceff Arrays                           | Page 12        |
| <b>3. Copy of Code</b>                                 | <b>Page 14</b> |

## ➤ List of Figures

|                         |        |
|-------------------------|--------|
| Figure 1: Code – Intro. | Page 3 |
|-------------------------|--------|

|  |         |
|--|---------|
| Figure 2: Code - Part 1: linear antenna (dipole of general length).  | Page 3  |
| Figure 3: Code - Part 2: Uniform linear antenna array (ULA).   | Page 4  |
| Figure 4: Code - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).                          | Page 5  |
| Figure 5: Code - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebysceff Arrays).                | Page 6  |
| Figure 6: Example 1 - Part 1: linear antenna (dipole of general length).                                   | Page 7  |
| Figure 7: Example 2 - Part 1: linear antenna (dipole of general length).                                   | Page 7  |
| Figure 8: Example 3 (Errors) - Part 1: linear antenna (dipole of general length).                          | Page 8  |
| Figure 9: Example 1 - Part 2: Uniform linear antenna array (ULA).  | Page 9  |
| Figure 10: Example 2 - Part 2: Uniform linear antenna array (ULA).   | Page 9  |
| Figure 11: Example 3 (Errors) - Part 2: Uniform linear antenna array (ULA).                                | Page 10 |
| Figure 12: Example 1 - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).                    | Page 10 |
| Figure 13: Example 2 - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).                    | Page 11 |
| Figure 14: Example 3 (Errors) - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).           | Page 11 |
| Figure 15: Example 1 - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebysceff Arrays).          | Page 12 |
| Figure 16: Example 2 - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebysceff Arrays).          | Page 12 |
| Figure 17: Example 3 (Errors) - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebysceff Arrays). | Page 13 |

[https://github.com/MahmoudFierro98/Antenna\\_Lab/](https://github.com/MahmoudFierro98/Antenna_Lab/)

## 1. Code

```
1 %%
2 % Alexandria University - Faculty of Engineering
3 % Electrical and Electronic Engineering Department - Fourth Year - Communications & Electronics
4 %
5 % Course: Antenna Lab
6 % Antenna Lab Assignment-2
7 %
8 % Name : Mahmoud Mohamed Kamal Ismail - Mahmoud AbdElHady Mahmoud - Ahmed Mohamed Abdelakher
9 % Section : 7 - 7 - 1
10 % Seat No.: 250 - 248 - 37
11
12 %%
13 - clear;
14 - close all;
15 - clc;
16
```

Figure 1: Code – Intro.

### 1.1. Part 1: linear antenna (dipole of general length)

```
17 %% Part (1): linear antenna (dipole of general length)
18 - fprintf('linear antenna (dipole of general length)\n');
19 - Lambda = 1;
20 - B = (2*pi)/Lambda;
21 - Theta = linspace(-pi,pi,350);
22 - Phi = linspace(-2*pi,2*pi,350);
23 - L = input('enter the length of dipole relative to lambda l = ');
24 - while L < 0
25 -     fprintf('ERROR :: "l < 0"\n');
26 -     L = input('enter the length of dipole relative to lambda (l>=0) l = ');
27 - end
28 - L = L * Lambda;
29 - En = abs((cos((B*L)/2).*cos(Theta)) - cos((B*L)/2)) ./ sin(Theta);
30
31 - figure(1);
32 - polar(Theta,En);
33 - view([90 90]);
34 - title('The 2D pattern of the dipole');
35
36 - Phi_3D = meshgrid(Phi);
37 - Theta_3D = meshgrid(Theta);
38 - En_3D = meshgrid(En);
39 - X = En_3D.*sin(Theta_3D).*cos(Phi_3D');
40 - Y = En_3D.*sin(Theta_3D).*sin(Phi_3D');
41 - Z = En_3D.*cos(Theta_3D);
42
43 - figure(2);
44 - surf(X,Y,Z);
45 - shading interp;
46 - axis vis3d;
47 - axis equal;
48 - lighting gouraud;
49 - title('The 3D pattern of the dipole');
50
51 - fprintf('-----\n');
52
```

Figure 2: Code - Part 1: linear antenna (dipole of general length).

## 1.2. Part 2: Uniform linear antenna array (ULA)

```
53 %% Part (2): Uniform linear antenna array(ULA)
54 fprintf('Uniform linear antenna array(ULA)\n');
55 Lambda = 1;
56 B = (2*pi)/Lambda;
57 d = input('enter the spacing w.r.t lambda d = ');
58 while d < 0
59     fprintf('ERROR :: "d < 0"\n');
60     d = input('enter the spacing w.r.t lambda (d>=0) d = ');
61 end
62 d = d * Lambda;
63 N = input('enter number of elemente N = ');
64 while N < 0
65     fprintf('ERROR :: "N < 0"\n');
66     N = input('enter number of elemente (N>=0) N = '); % If N = 0 there's no ar
67 end
68 alpha = input('the progressive phase shift alpha = ');
69 max_angle = acos(-alpha/(B*d));
70 Gamma = linspace(-pi,pi,6000);
71 Phi = linspace(-2*pi,2*pi,6000);
72 ebsi = B*d*cos(Gamma) + alpha;
73 AF = abs(sin((N*ebsi)/2) ./ (N * sin(ebsi/2)));
74
75 figure(1);
76 plot(ebsi,AF);
77 title('array factor vs ebsi');
78 xlabel('ebsi','fontsize',10);
79 ylabel('AF','fontsize',10);
80
81 figure(2);
82 polar(Gamma,AF);
83 view([90 90]);
84 title('The 2D pattern of the array');
85
86 Phi_3D = meshgrid(Phi);
87 Gamma_3D = meshgrid(Gamma);
88 AF_3D = meshgrid(AF);
89 X = AF_3D.*sin(Gamma_3D).*cos(Phi_3D');
90 Y = AF_3D.*sin(Gamma_3D).*sin(Phi_3D');
91 Z = AF_3D.*cos(Gamma_3D);
92
93 figure(3);
94 surf(X,Y,Z);
95 shading interp;
96 lighting gouraud;
97 title('The 3D pattern of the array');
98
99 fprintf('-----\n');
100
```

Figure 3: Code - Part 2: Uniform linear antenna array (ULA).

### 1.3. Part 3: Nonuniformly-Fed linear antenna array

#### A. Binomial Arrays

```
101 %% Part (3): Nonuniformly-Fed linear antenna array
102 % A. Binomial Arrays
103 - Lambda = 1;
104 - B = (2*pi)/Lambda;
105 - d = input('enter the spacing w.r.t lambda d = ');
106 - while d < 0
107 -     fprintf('ERROR :: "d < 0"\n');
108 -     d = input('enter the spacing w.r.t lambda (d>=0) d = ');
109 - end
110 - d = d * Lambda;
111 - N = input('enter number of elemente N = ');
112 - while N < 0
113 -     fprintf('ERROR :: "N < 0"\n');
114 -     N = input('enter number of elemente (N>=0) N = '); % If N = 0 there's no array
115 - end
116 - alpha = input('the progressive phase shift alpha = ');
117 - Theta = linspace(-pi,pi,6000);
118 - Phi = linspace(-2*pi,2*pi,6000);
119 - u = (B*d*cos(Theta) + alpha)/2;
120 - AF = abs(cos(u).^(N-1));
121
122 - figure(1);
123 - plot(u,AF);
124 - title('array factor vs u');
125 - xlabel('u','fontsize',10);
126 - ylabel('AF','fontsize',10);
127
128 - figure(2);
129 - polar(Theta,AF);
130 - view([90 90]);
131 - title('The 2D pattern of the array');
132
133 - Phi_3D = meshgrid(Phi);
134 - Theta_3D = meshgrid(Theta);
135 - AF_3D = meshgrid(AF);
136 - X = AF_3D.*sin(Theta_3D).*cos(Phi_3D');
137 - Y = AF_3D.*sin(Theta_3D).*sin(Phi_3D');
138 - Z = AF_3D.*cos(Theta_3D);
139
140 - figure(3);
141 - surf(X,Y,Z);
142 - shading interp;
143 - axis vis3d;
144 - axis equal;
145 - lighting gouraud;
146 - title('The 3D pattern of the array');
147
148 - fprintf('-----\n');
149
```

Figure 4: Code - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).

## B. Dolph-Tschebysceff Arrays

```

150 %% Part (3): Nonuniformly-Fed linear antenna array
151 % B. Dolph-Tschebysceff Arrays
152 Lambda = 1;
153 B = (2*pi)/Lambda;
154 d = input('enter the spacing w.r.t lambda d = ');
155 while d < 0
156     fprintf('ERROR :: "d < 0"\n');
157     d = input('enter the spacing w.r.t lambda (d>=0) d = ');
158 end
159 d = d * Lambda;
160 N = input('enter number of elemente N = ');
161 while N < 0
162     fprintf('ERROR :: "N < 0"\n');
163     N = input('enter number of elemente (N>=0) N = '); % If N = 0 there's no array
164 end
165 M = N - 1;
166 alpha = input('the progressive phase shift alpha = ');
167 Ro = input('Mainlobe to sidelobe level Ro = ');
168 while Ro <= 1
169     fprintf('ERROR :: "Ro < 1"\n');
170     Ro = input('Mainlobe to sidelobe level (Ro>1) Ro = ');
171 end
172 Zo = cosh((1/M)*acosh(Ro));
173 Z = linspace(-Zo,Zo,6000);
174 u_up = acos(Z./Zo);
175 u_down = -u_up;
176 u = [u_down ; u_up];
177 Theta1 = acos((2.*u_down)-alpha)/(B*d);
178 Theta2 = -Theta1;
179 Phi = linspace(-2*pi,2*pi,6000);
180 AF = abs(cosh(M.*acosh(Z)));
181
182 figure(1);
183 plot(Z,AF);
184 title('array factor vs Z');
185 xlabel('Z','fontsize',10);
186 ylabel('AF','fontsize',10);
187
188 figure(2);
189 polar(Theta1,AF,'-b');
190 hold on;
191 polar(Theta2,AF,'-b');
192 view([90 90]);
193 title('The 2D pattern of the array');
194
195 Phi_3D = meshgrid(Phi);
196 Theta_3D = meshgrid(Theta1);
197 AF_3D = meshgrid(AF);
198 X = AF_3D.*sin(Theta_3D).*cos(Phi_3D);
199 Y = AF_3D.*sin(Theta_3D).*sin(Phi_3D);
200 Z_ = AF_3D.*cos(Theta_3D);
201
202 figure(3);
203 surf(X,Y,Z_);
204 shading interp;
205 axis vis3d;
206 axis equal;
207 lighting gouraud;
208 title('The 3D pattern of the array');
209
210 fprintf('-----\n');

```

Figure 5: Code - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebysceff Arrays).

## 2. Examples

### 2.1. Part 1: linear antenna (dipole of general length)

Example 1:  $l = \frac{3\lambda}{2}$

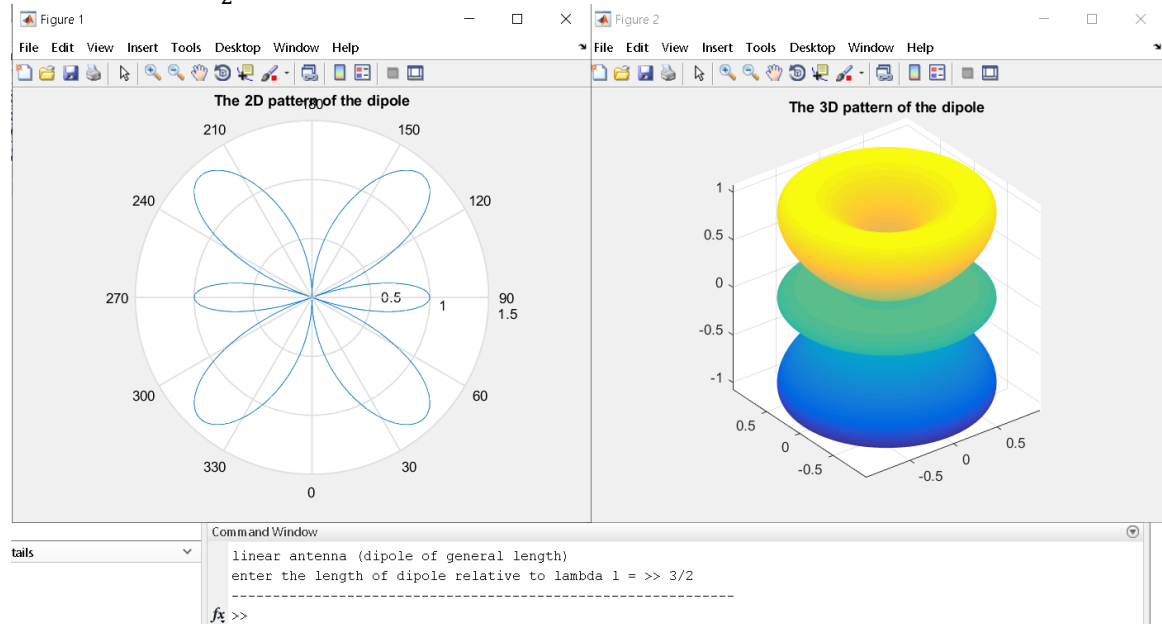


Figure 6: Example 1 - Part 1: linear antenna (dipole of general length).

Example 2:  $l = 2\lambda$

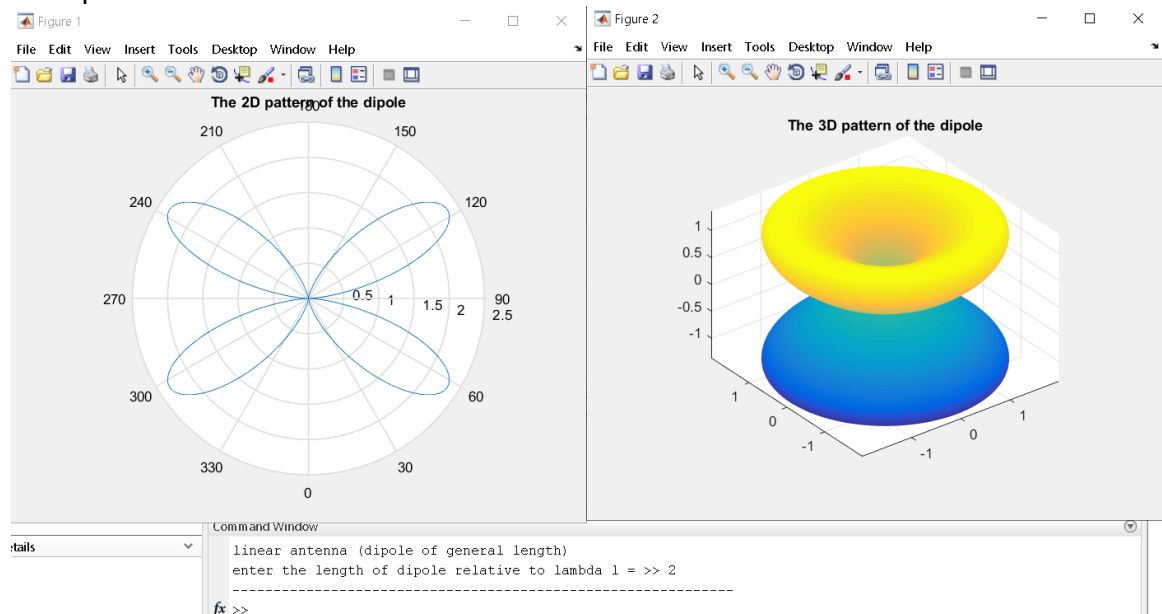


Figure 7: Example 2 - Part 1: linear antenna (dipole of general length).



### Example 3: *Errors*

if  $l < 0$ , we will ask you again  $l = ?$ , then I put it = 0.5.

And what happen if you put  $l = 0$

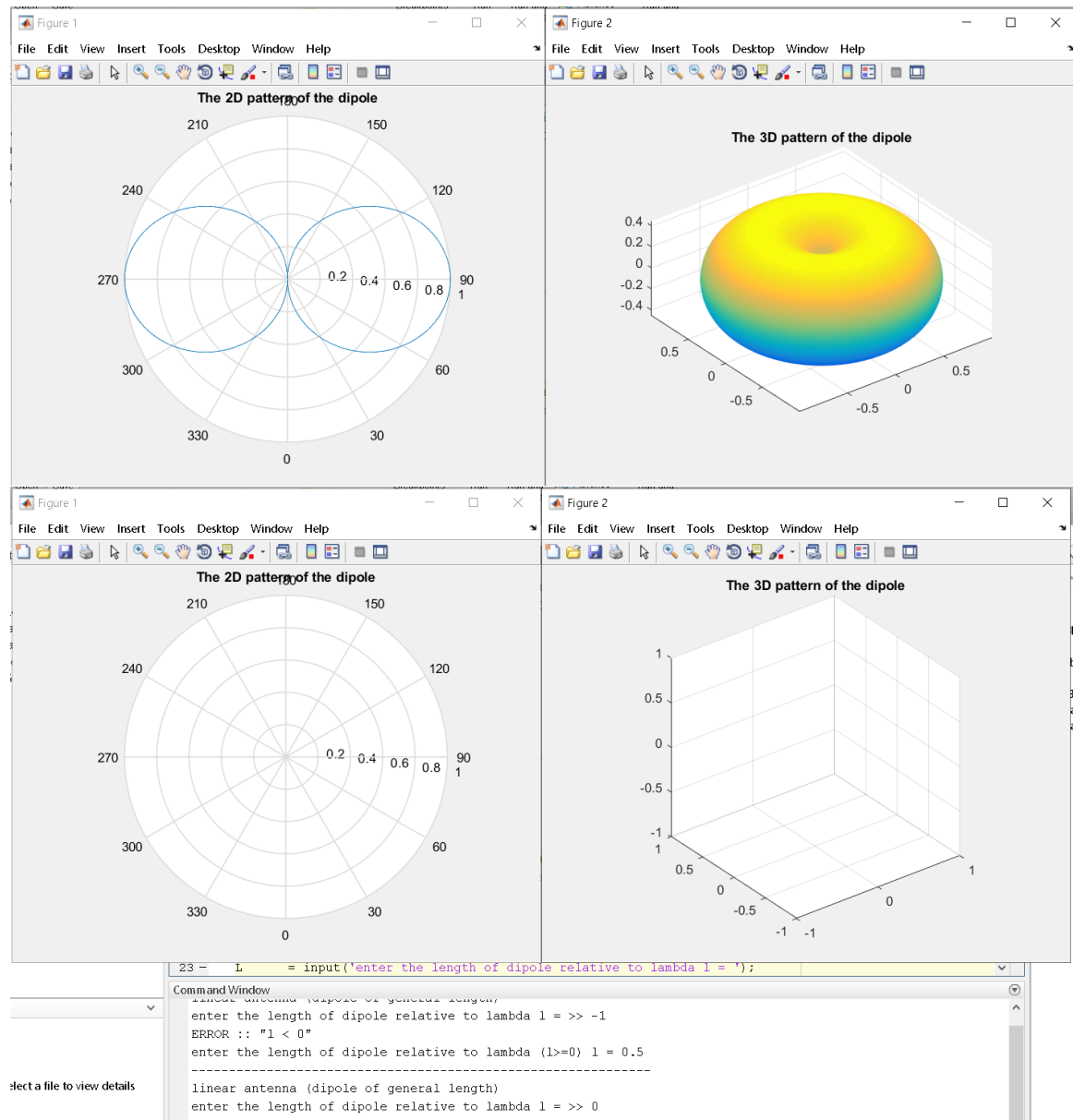


Figure 8: Example 3 (Errors) - Part 1: linear antenna (dipole of general length).

## 2.2. Part 2: Uniform linear antenna array (ULA)

Example 1:  $d = \frac{4\lambda}{7}, N = 7, \alpha = \frac{-4\pi}{7}$

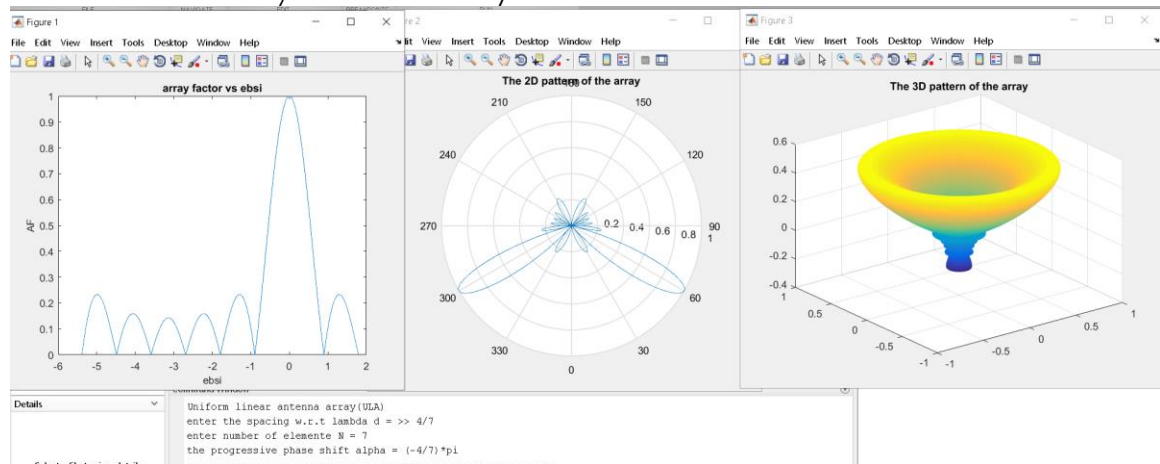


Figure 9: Example 1 - Part 2: Uniform linear antenna array (ULA).

Example 2:  $d = \frac{5\lambda}{12}, N = 6, \alpha = \frac{-5\pi}{6}$

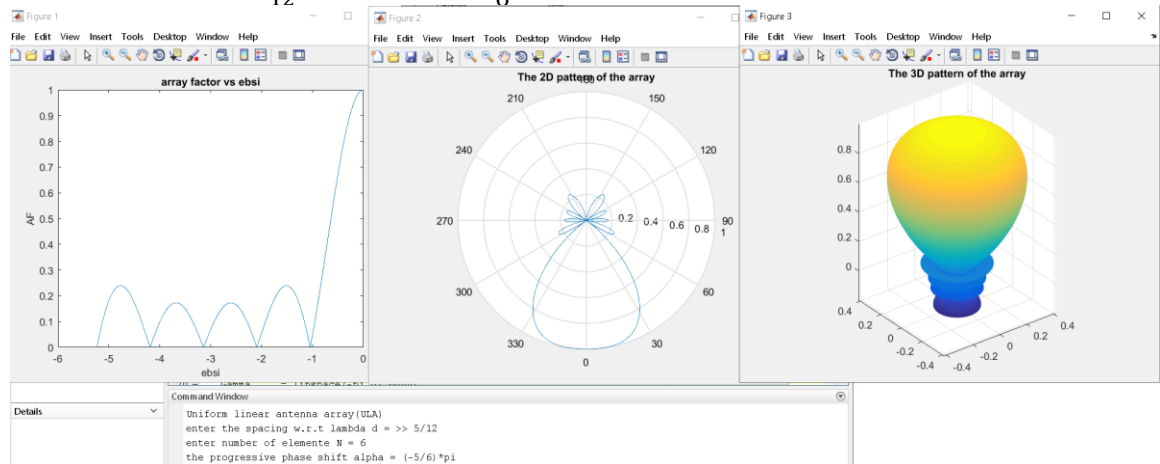


Figure 10: Example 2 - Part 2: Uniform linear antenna array (ULA).

### Example 3: Errors

if  $d < 0$  or  $N < 0$ , we will ask you again.

And what happen if you put *all* pramaters = 0

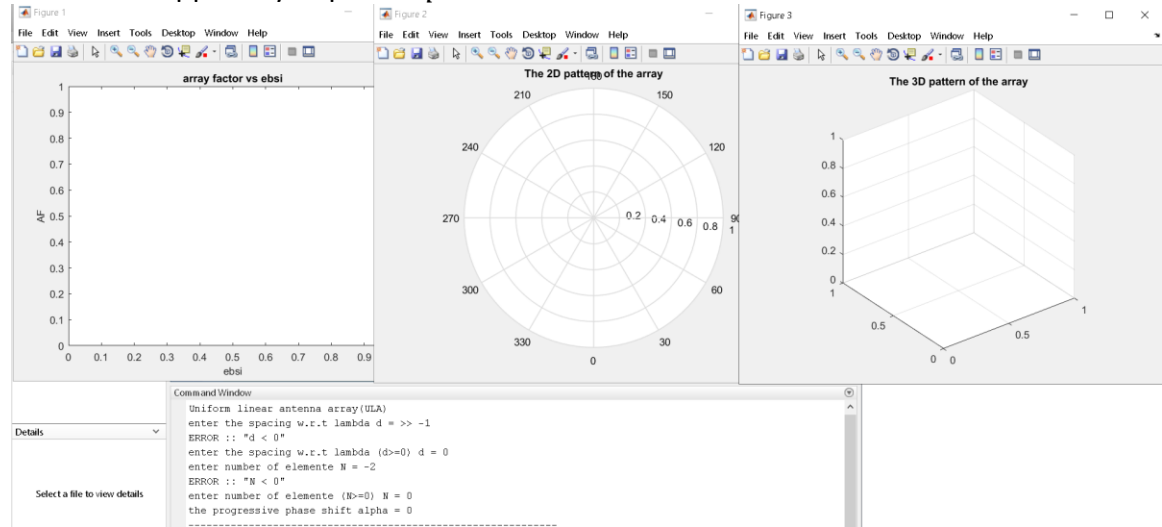


Figure 11: Example 3 (Errors) - Part 2: Uniform linear antenna array (ULA).

## 2.3. Part 3: Nonuniformly-Fed linear antenna array

### A. Binomial Arrays

Example 1:  $d = \frac{3\lambda}{4}$ ,  $N = 8$ ,  $\alpha = 0$

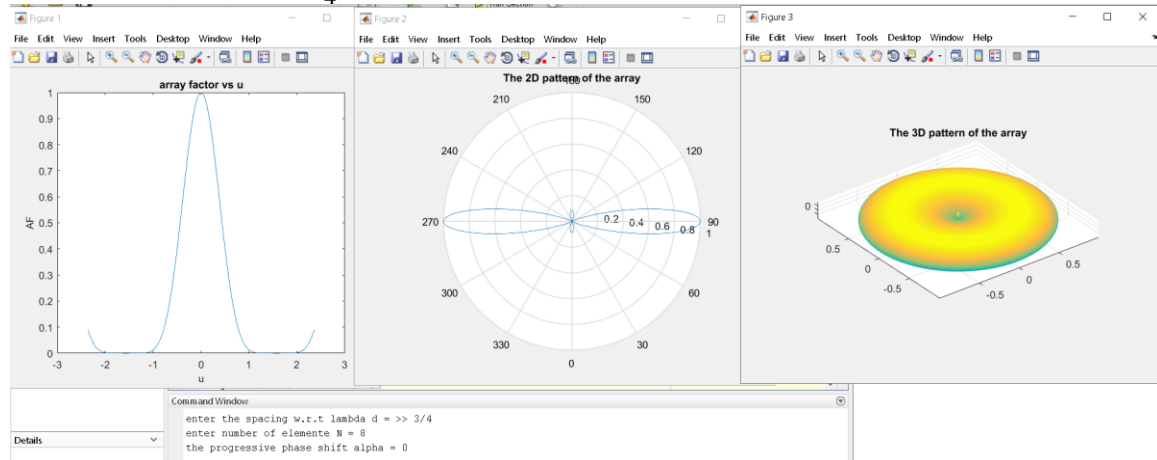


Figure 12: Example 1 - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).

Example 2:  $d = \frac{\lambda}{4}, N = 8, \alpha = 0$

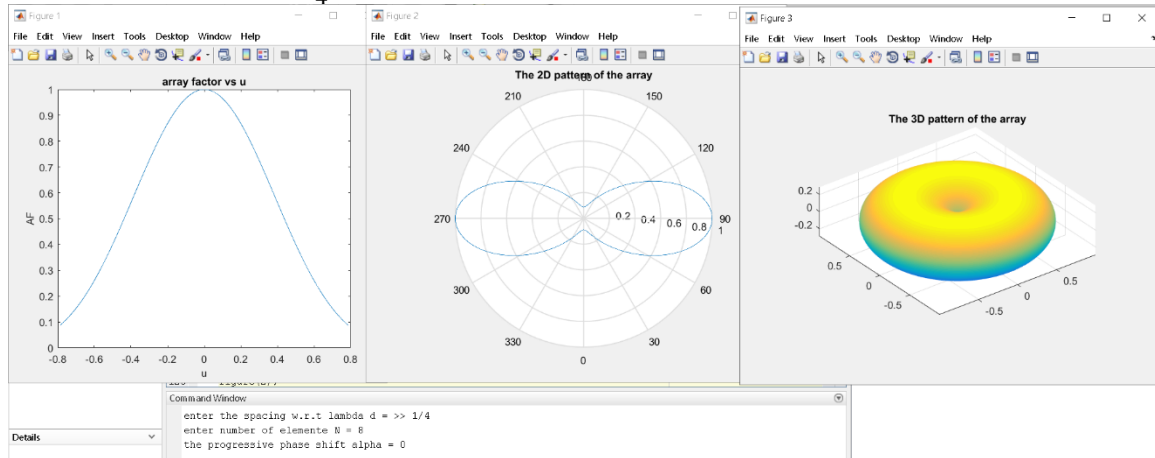


Figure 13: Example 2 - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).

### Example 3: Errors

if  $d < 0$  or  $N < 0$ , we will ask you again.

And what happen if you put *all* pramaters = 0

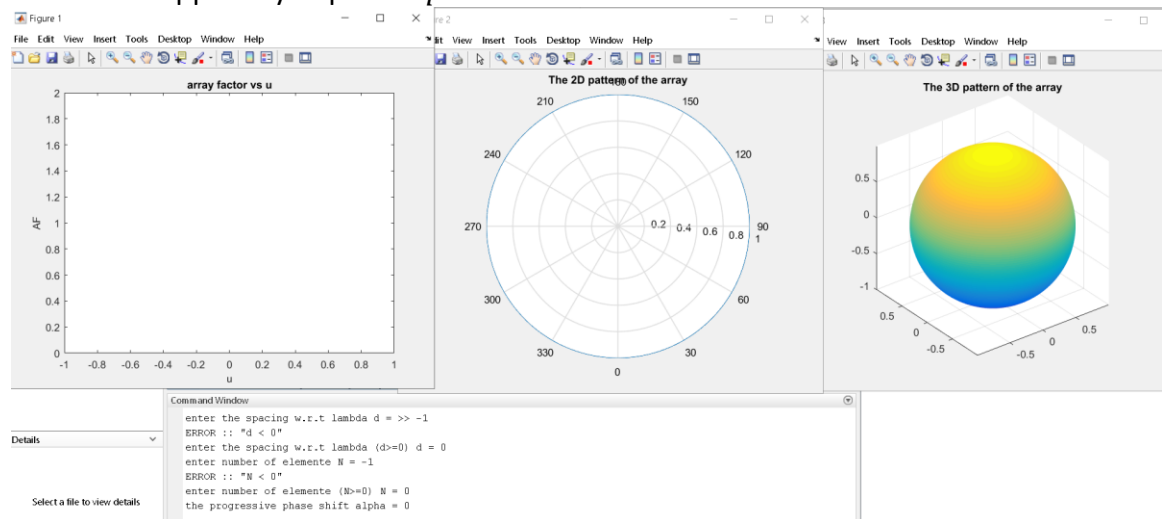


Figure 14: Example 3 (Errors) - Part 3: Nonuniformly-Fed linear antenna array (Binomial Arrays).

## B. Dolph-Tschebyscheff Arrays

Example 1:  $d = \frac{\lambda}{2}$ ,  $N = 6$ ,  $\alpha = -\pi$ ,  $R_0 = 10$ .

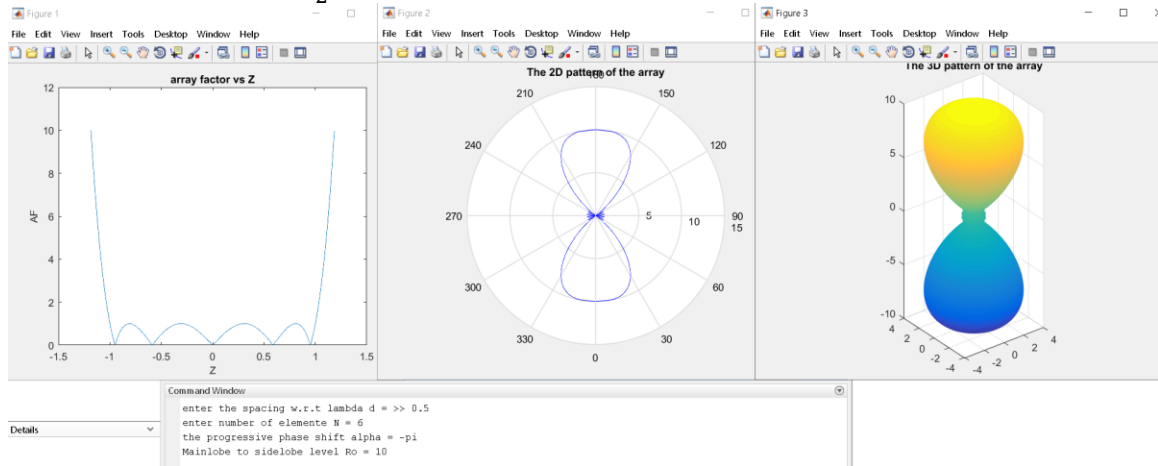


Figure 15: Example 1 - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebyscheff Arrays).

Example 2:  $d = \frac{\lambda}{2}$ ,  $N = 3$ ,  $\alpha = -\pi$ ,  $R_0 = 10$ .

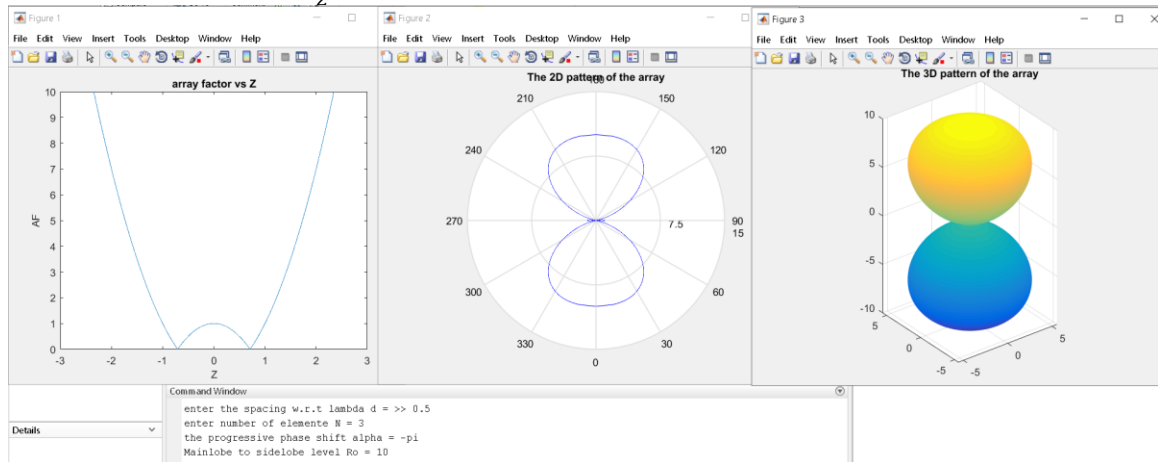


Figure 16: Example 2 - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebyscheff Arrays).

### Example 3: *Errors*

if  $d < 0$  or  $N < 0$ , we will ask you again.

Always  $R_0 > 1$ .

#### Comm and Window

```
enter the spacing w.r.t lambda d = >> -1
ERROR :: "d < 0"
enter the spacing w.r.t lambda (d>=0) d = 0
enter number of elemente N = -8
ERROR :: "N < 0"
enter number of elemente (N>=0) N = 0
the progressive phase shift alpha = 0
Mainlobe to sidelobe level Ro = 1
ERROR :: "Ro < 1"
Mainlobe to sidelobe level (Ro>1) Ro = -2
ERROR :: "Ro < 1"
Mainlobe to sidelobe level (Ro>1) Ro = 1
ERROR :: "Ro < 1"
fx Mainlobe to sidelobe level (Ro>1) Ro = |
```

Figure 17: Example 3 (Errors) - Part 3: Nonuniformly-Fed linear antenna array (Dolph-Tschebysceff Arrays).

### 3. Copy of Code

```
%%
% Alexandria University - Faculty of Engineering
% Electrical and Electronic Engineering Department -
Fourth Year - Communications & Electronics
%
% Course: Antenna Lab
% Antenna Lab Assignment-2
%
% Name      : Mahmoud Mohamed Kamal Ismail - Mahmoud
AbdElHady Mahmoud - Ahmed Mohamed Abdelakher
% Section   : 7                      - 7
- 1
% Seat No.: 250                      - 248
- 37

%%
clear;
close all;
clc;

%% Part (1): linear antenna (dipole of general length)
fprintf('linear antenna (dipole of general length)\n');
Lambda = 1;
B       = (2*pi)/Lambda;
Theta   = linspace(-pi,pi,350);
Phi      = linspace(-2*pi,2*pi,350);
L        = input('enter the length of dipole relative to
lambda l = ');
while L < 0
    fprintf('ERROR :: "l < 0"\n');
    L = input('enter the length of dipole relative to
lambda (l>=0) l = ');
end
L        = L * Lambda;
En       = abs((cos((B*L)/2).*cos(Theta)) -
cos((B*L)/2)) ./ sin(Theta));

figure(1);
polar(Theta,En);
view([90 90]);
title('The 2D pattern of the dipole');
```

```

Phi_3D    = meshgrid(Phi);
Theta_3D  = meshgrid(Theta);
En_3D     = meshgrid(En);
X         = En_3D.*sin(Theta_3D).*cos(Phi_3D');
Y         = En_3D.*sin(Theta_3D).*sin(Phi_3D');
Z         = En_3D.*cos(Theta_3D);

figure(2);
surf(X,Y,Z);
shading interp;
axis vis3d;
axis equal;
lighting gouraud;
title('The 3D pattern of the dipole');

fprintf('-----\n');

%% Part (2): Uniform linear antenna array(ULA)
fprintf('Uniform linear antenna array(ULA)\n');
Lambda    = 1;
B         = (2*pi)/Lambda;
d         = input('enter the spacing w.r.t lambda d = ');
while d < 0
    fprintf('ERROR :: "d < 0"\n');
    d      = input('enter the spacing w.r.t lambda (d>=0) d = ');
end
d         = d * Lambda;
N         = input('enter number of elemente N = ');
while N < 0
    fprintf('ERROR :: "N < 0"\n');
    N      = input('enter number of elemente (N>=0) N = ');
    % If N = 0 there's no array
end
alpha     = input('the progressive phase shift alpha = ');
max_angle = acos(-alpha/(B*d));
Gamma     = linspace(-pi,pi,6000);
Phi       = linspace(-2*pi,2*pi,6000);
ebsi      = B*d*cos(Gamma) + alpha;
AF        = abs(sin((N*ebsi)/2) ./ (N * sin(ebsi/2)));

```



```

figure(1);
plot(ebsi,AF);
title('array factor vs ebsi');
xlabel('ebsi','fontsize',10);
ylabel('AF','fontsize',10);

figure(2);
polar(Gamma,AF);
view([90 90]);
title('The 2D pattern of the array');

Phi_3D = meshgrid(Phi);
Gamma_3D = meshgrid(Gamma);
AF_3D = meshgrid(AF);
X = AF_3D.*sin(Gamma_3D).*cos(Phi_3D);
Y = AF_3D.*sin(Gamma_3D).*sin(Phi_3D);
Z = AF_3D.*cos(Gamma_3D);

figure(3);
surf(X,Y,Z);
shading interp;
lighting gouraud;
title('The 3D pattern of the array');

fprintf('-----\n');

%% Part (3): Nonuniformly-Fed linear antenna array
% A. Binomial Arrays
Lambda = 1;
B = (2*pi)/Lambda;
d = input('enter the spacing w.r.t lambda d = ');
while d < 0
    fprintf('ERROR :: "d < 0"\n');
    d = input('enter the spacing w.r.t lambda (d>=0) d = ');
end
d = d * Lambda;
N = input('enter number of elemente N = ');
while N < 0
    fprintf('ERROR :: "N < 0"\n');
    N = input('enter number of elemente (N>=0) N = ');
    % If N = 0 there's no array
end

```

```

alpha = input('the progressive phase shift alpha = ');
Theta = linspace(-pi,pi,6000);
Phi    = linspace(-2*pi,2*pi,6000);
u      = (B*d*cos(Theta) + alpha)/2;
AF     = abs(cos(u).^(N-1));

figure(1);
plot(u,AF);
title('array factor vs u');
xlabel('u','fontsize',10);
ylabel('AF','fontsize',10);

figure(2);
polar(Theta,AF);
view([90 90]);
title('The 2D pattern of the array');

Phi_3D = meshgrid(Phi);
Theta_3D = meshgrid(Theta);
AF_3D = meshgrid(AF);
X      = AF_3D.*sin(Theta_3D).*cos(Phi_3D');
Y      = AF_3D.*sin(Theta_3D).*sin(Phi_3D');
Z      = AF_3D.*cos(Theta_3D);

figure(3);
surf(X,Y,Z);
shading interp;
axis vis3d;
axis equal;
lighting gouraud;
title('The 3D pattern of the array');

fprintf('-----\n');

%% Part (3): Nonuniformly-Fed linear antenna array
% B. Dolph-Tschebyscheff Arrays
Lambda = 1;
B       = (2*pi)/Lambda;
d       = input('enter the spacing w.r.t lambda d = ');
while d < 0
    fprintf('ERROR :: "d < 0"\n');
    d = input('enter the spacing w.r.t lambda (d>=0) d = ');

```

```

end
d      = d * Lambda;
N      = input('enter number of elemente N = ');
while N < 0
    fprintf('ERROR :: "N < 0"\n');
    N = input('enter number of elemente (N>=0) N = ');
% If N = 0 there's no array
end
M      = N - 1;
alpha  = input('the progressive phase shift alpha = ');
Ro     = input('Mainlobe to sidelobe level Ro = ');
while Ro <= 1
    fprintf('ERROR :: "Ro < 1"\n');
    Ro = input('Mainlobe to sidelobe level (Ro>1) Ro = ');
end
Zo     = cosh((1/M)*acosh(Ro));
Z      = linspace(-Zo,Zo,6000);
u_up   = acos(Z./Zo);
u_down = -u_up;
u      = [u_down ; u_up];
Theta1 = acos(((2.*u_down)-alpha)/(B*d));
Theta2 = -Theta1;
Phi    = linspace(-2*pi,2*pi,6000);
AF     = abs(cosh(M.*acosh(Z)));

figure(1);
plot(Z,AF);
title('array factor vs Z');
xlabel('Z','fontsize',10);
ylabel('AF','fontsize',10);

figure(2);
polar(Theta1,AF,'-b');
hold on;
polar(Theta2,AF,'-b');
view([90 90]);
title('The 2D pattern of the array');

Phi_3D = meshgrid(Phi);
Theta_3D = meshgrid(Theta1);
AF_3D = meshgrid(AF);
X      = AF_3D.*sin(Theta_3D).*cos(Phi_3D);
Y      = AF_3D.*sin(Theta_3D).*sin(Phi_3D);

```

```
Z_      = AF_3D.*cos(Theta_3D);  
  
figure(3);  
surf(X,Y,Z_);  
shading interp;  
axis vis3d;  
axis equal;  
lighting gouraud;  
title('The 3D pattern of the array');  
  
fprintf('-----  
-----\n');
```