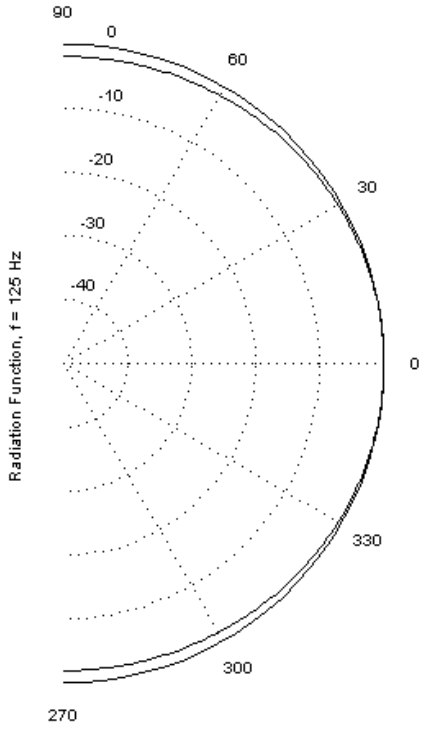
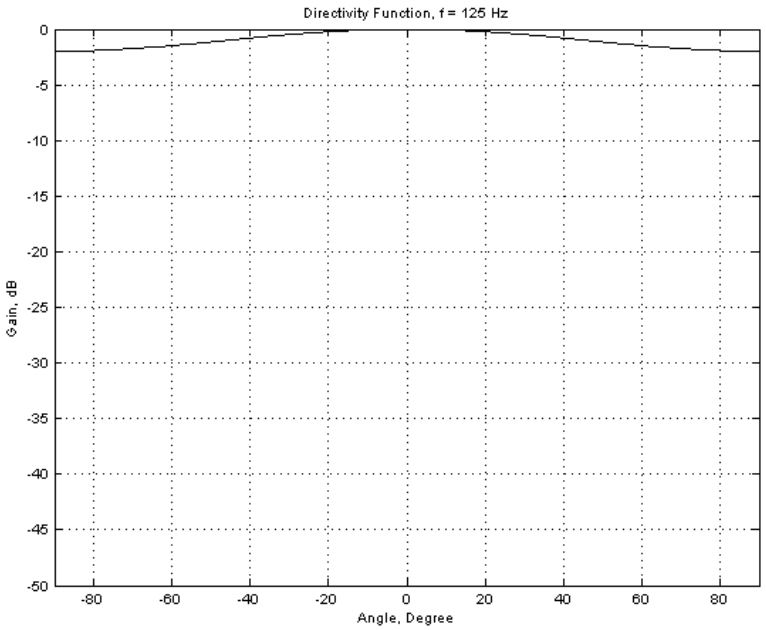
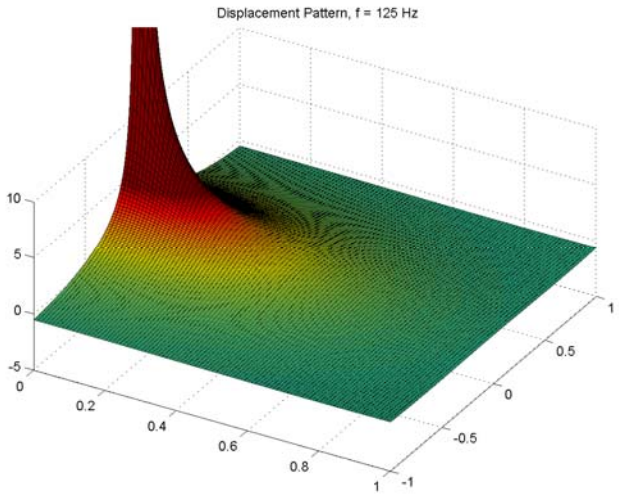
Part 1a: Array Length = 1m, Uniform volume velocity

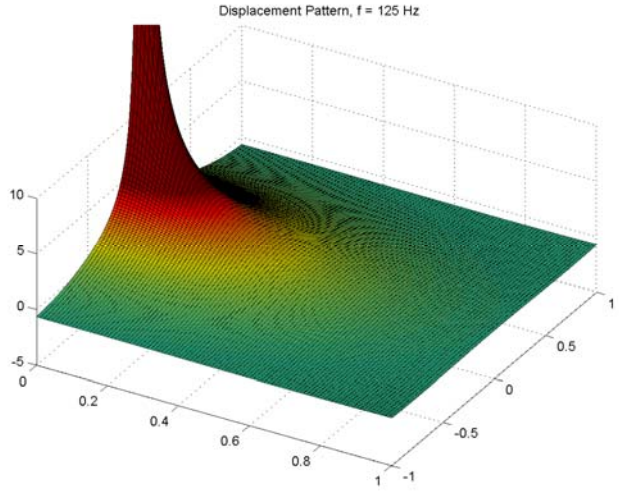


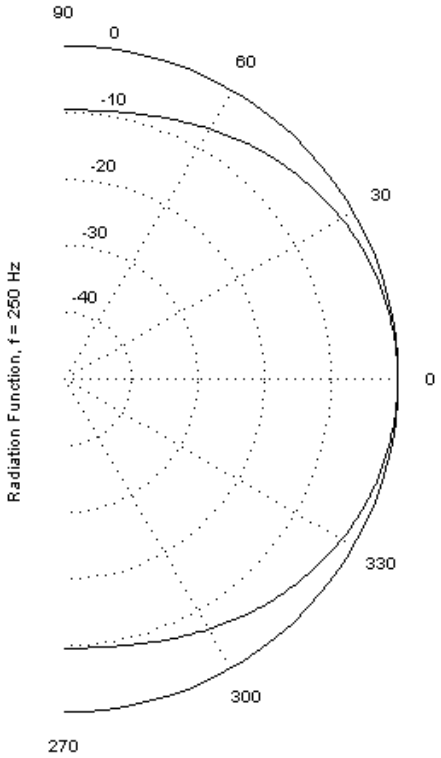


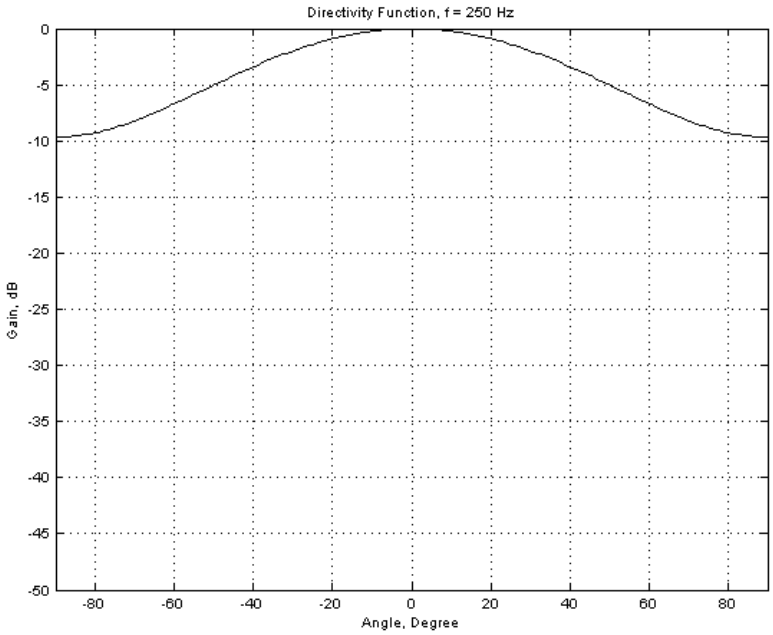
Elevation Plane



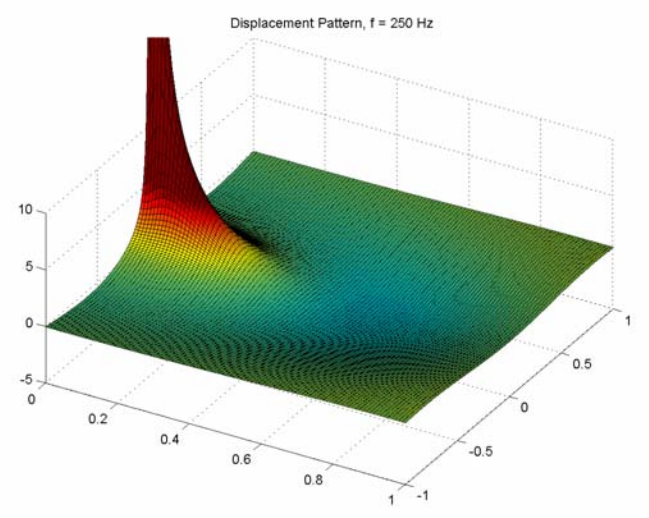
Azimuth Plane



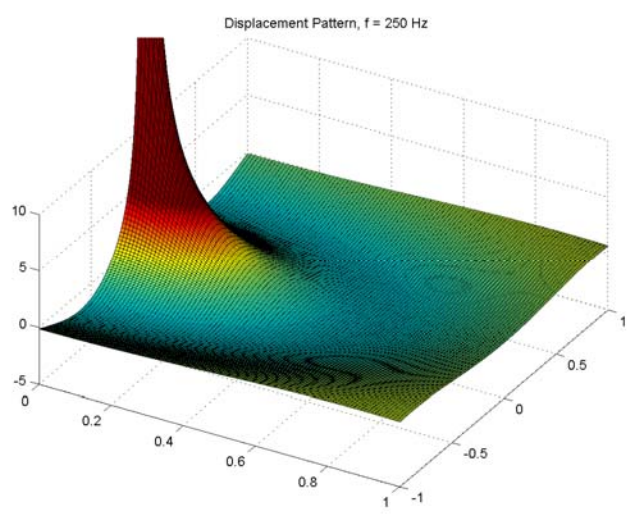


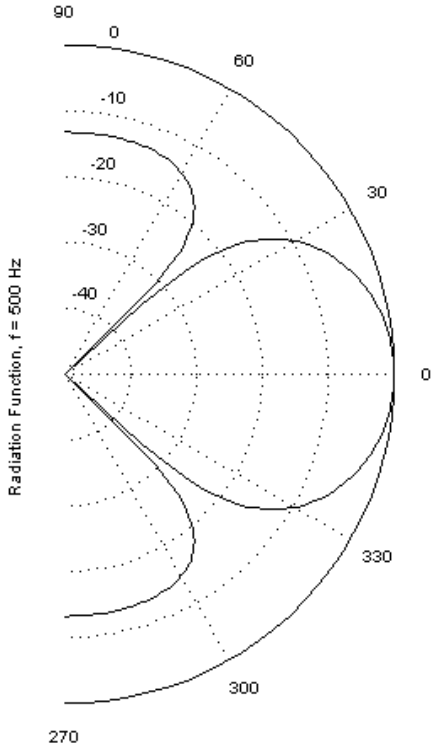


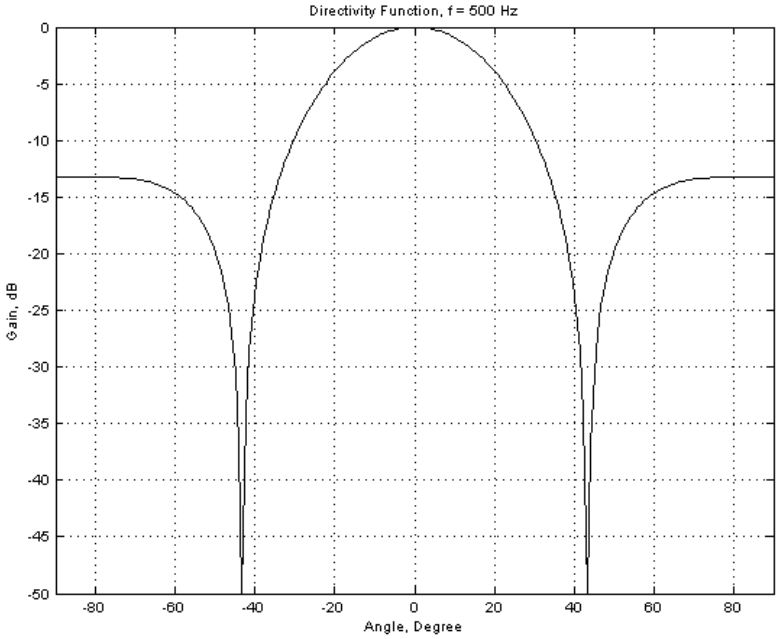
Elevation Plane



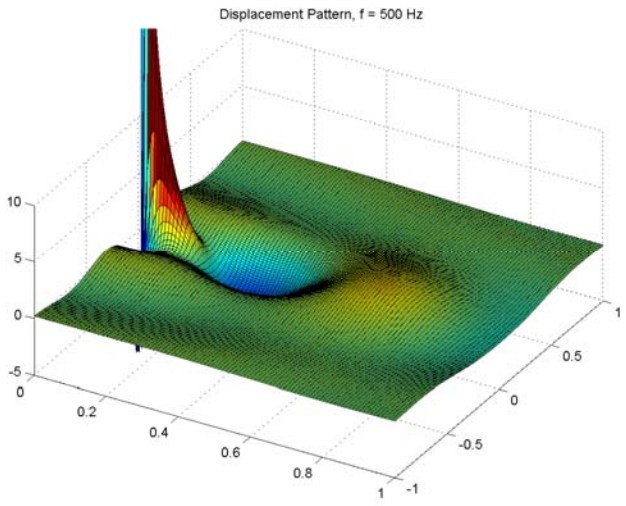
Azimuth Plane



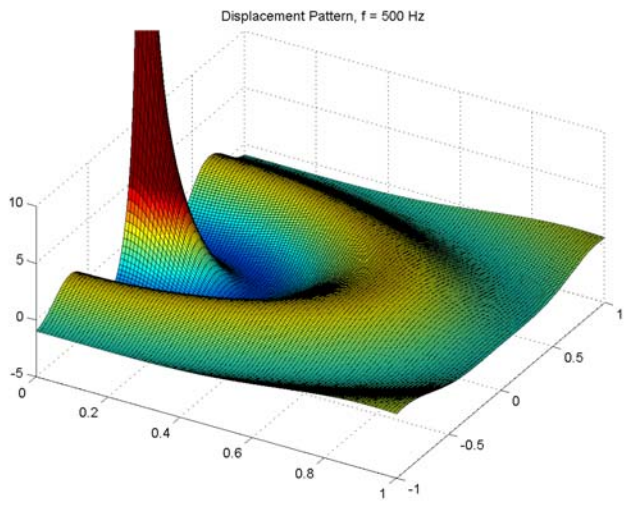


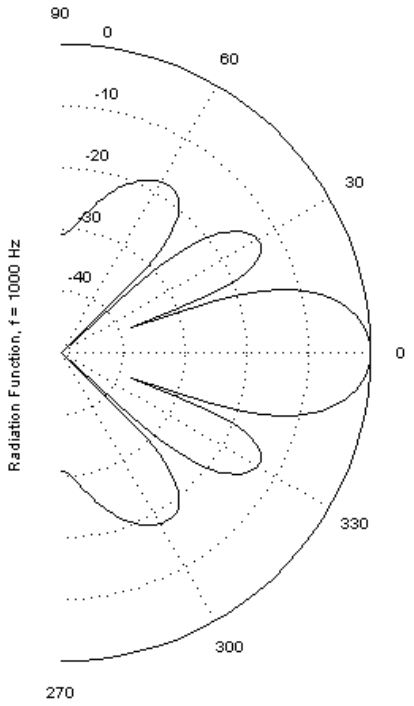


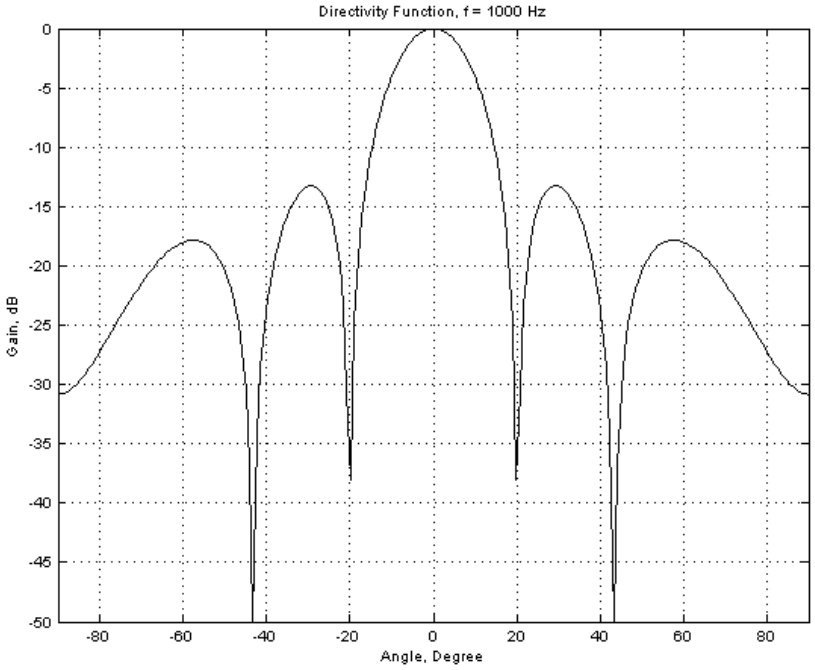
Elevation Plane



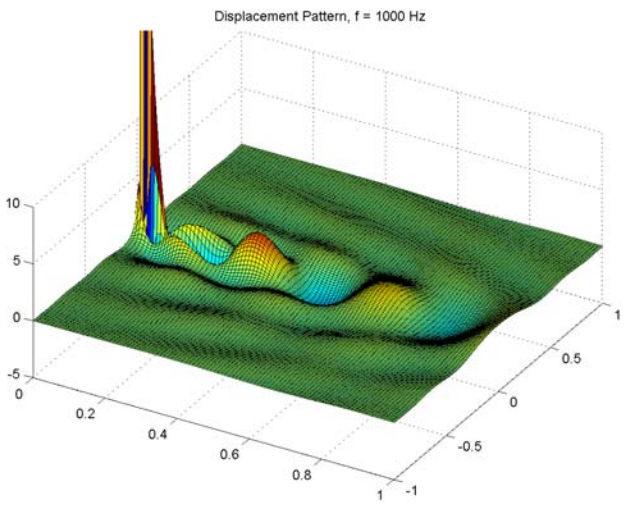
Azimuth Plane



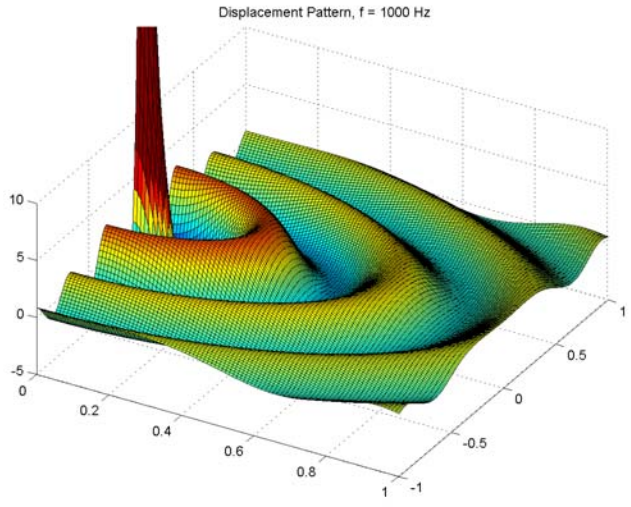


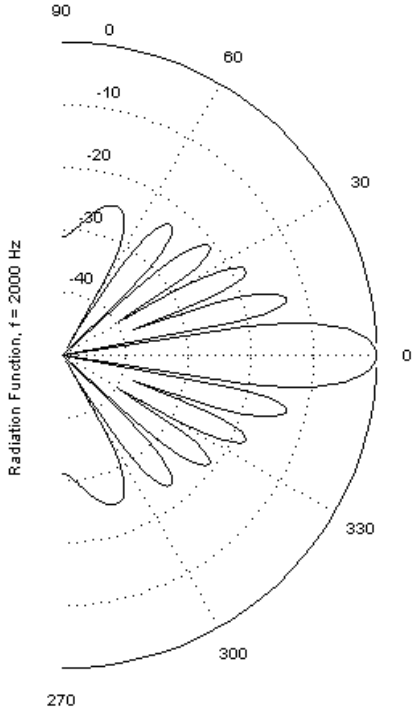


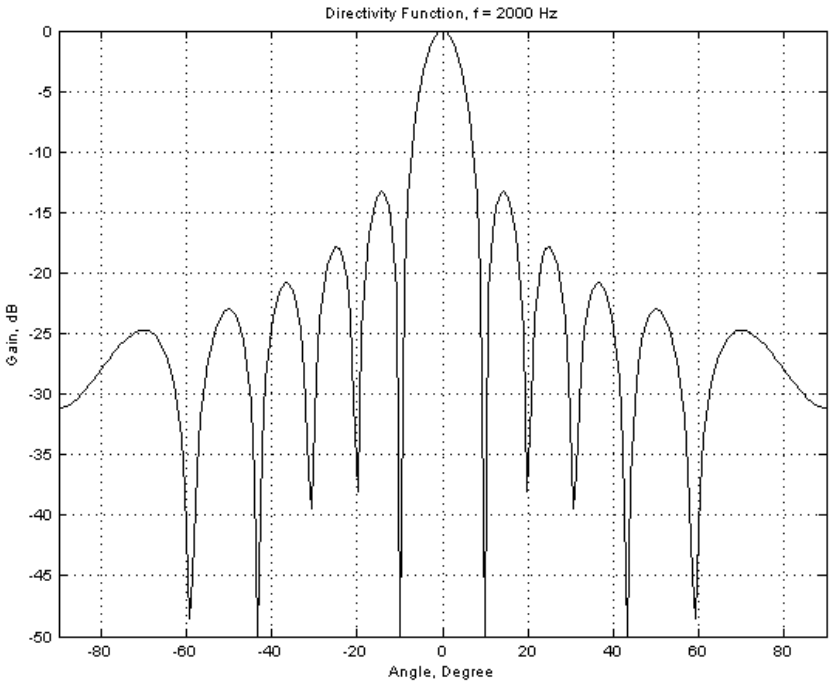
Elevation Plane



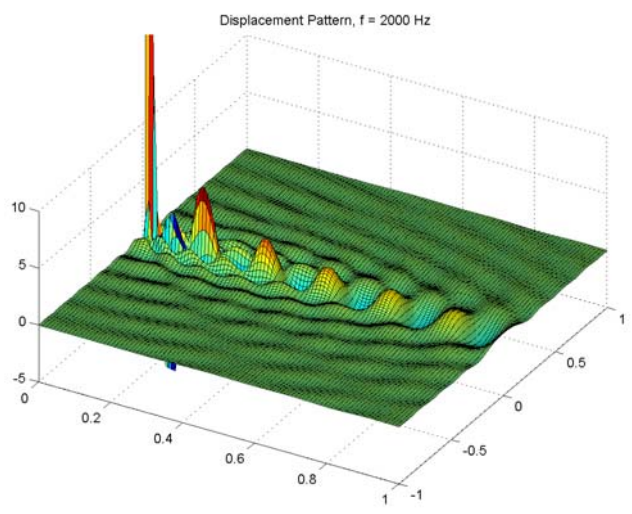
Azimuth Plane



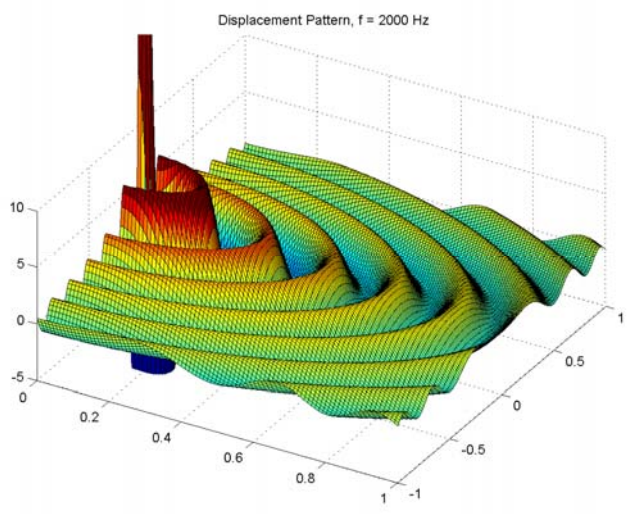




Elevation Plane



Azimuth Plane

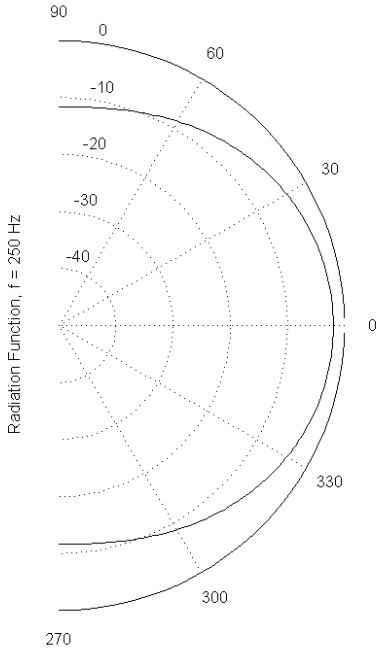


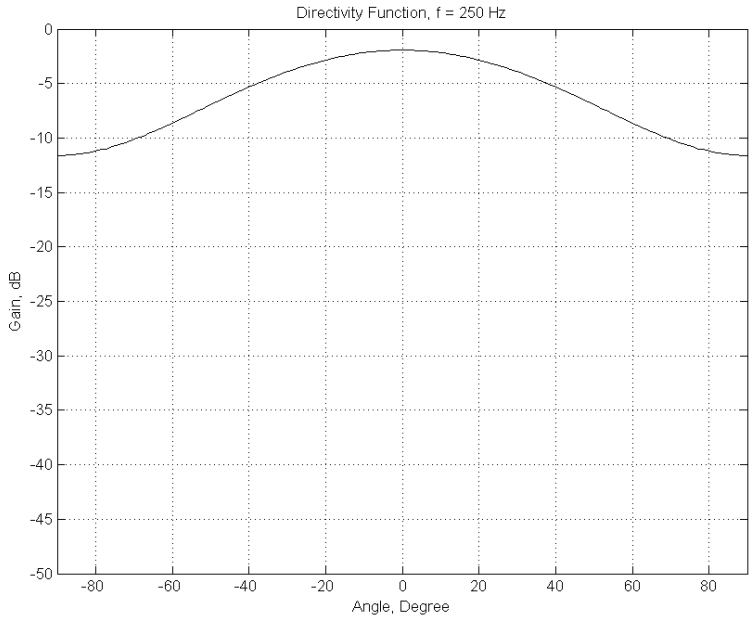
Part 1a Source Code:

```
% Ross Penniman
% EEN 502
% Project 2, Part 1a

clear all, close all
b=-1/2:(1/200):1/2;
thetaN=b*pi;
```

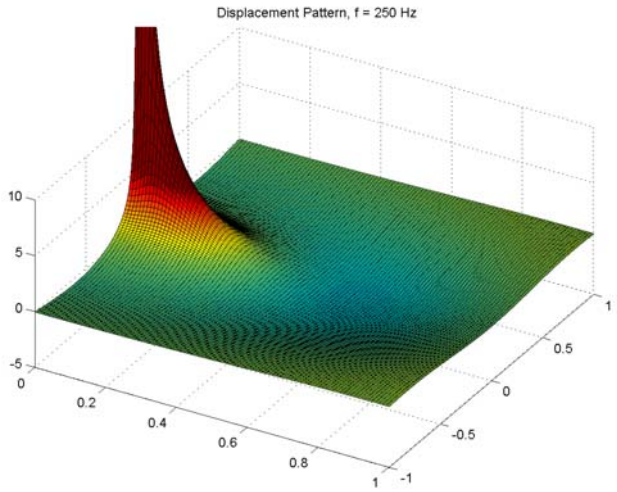
```
l=1; % length of array in meters
c=343;
f= 125;  % 125, 250, 500, 1000, 2000 Hz
lambda=c/f;

for j=1:length(thetaN)
    u = (l/lambda)*sin(thetaN(j));
    if u == 0
        p(j) = 1;
    else
        p(j)=sin(pi*u)/(pi*u);
    end
end

pdB = 20*log10(abs(p));
pdB(pdB < -50) = -50; % limit lower level of p to -50 dB
posdB = pdB + 50;

figure; % subplot(121);
polar(thetaN, posdB,'k-');  axis([0 50 -50 50]);
ylabel(['Radiation Function, f = ',num2str(f),' Hz']);

ph=findall(gca,'type','text');  ps=get(ph,'string');
% disp([num2cell(1:numel(ps)).',ps]); % see the content's indices...
ps([5,7,9,12,14,16,17,18,19,20])={'', '', '', '', '', '0', '-10',
     '-20', '-30', '-40'};
set(ph,{'string'},ps);
% print(gcf, ['prob1a_p_' num2str(f)], '-dpng', '-r80')

figure; % subplot(122);
plot(b*180, pdB,'k'); axis([-90 90 -50 0]); grid on;
title(['Directivity Function, f = ',num2str(f),' Hz']);
xlabel('Angle, Degree'); ylabel('Gain, dB');
% print(gcf, ['prob1a_d_' num2str(f)], '-dpng', '-r80')

[X,Y] = meshgrid(0:0.01:1, -1:0.01:1);
k=2*pi*f/c;
R=sqrt(X.^2+Y.^2);
m=(pi*l/lambda)*(abs(Y)./R); % Y/R = sin(thetaN)
m(m == 0) = 0.00001;
q=(sin(m)./m).*cos(k*R)./R;  % Equation 3.38 from Moser
figure;
surf(X,Y,q);  axis ([0 1 -1 1 -5 10]);  caxis([-5 5]);
title(['Displacement Pattern, f = ',num2str(f),' Hz']);
view (30, 50);
% print(gcf, ['prob1a_3D_' num2str(f)], '-dpng', '-r160')

q2 = cos(k*R)./R;  % azimuth half plane at thetaN = 0
figure;
surf(X,Y,q2);  axis ([0 1 -1 1 -5 10]);  caxis([-5 5]);
title(['Displacement Pattern, f = ',num2str(f),' Hz']);
view (30, 50);
% print(gcf, ['prob1a_3Daz_' num2str(f)], '-dpng', '-r160')
```

Part 1b: Array Length = 1m, Volume velocity profile has gaps to simulate frames around drivers of array.

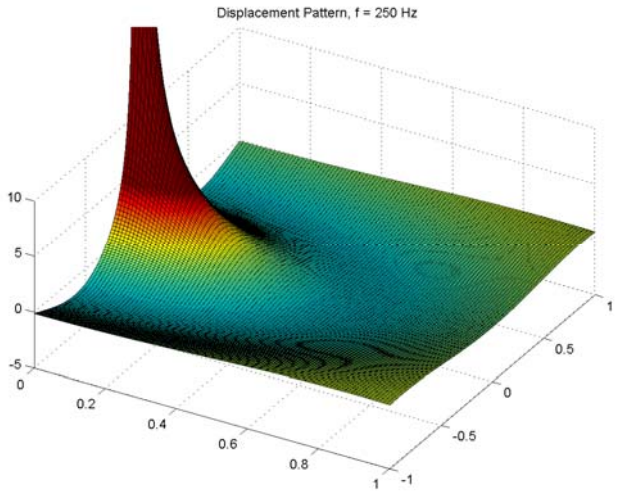


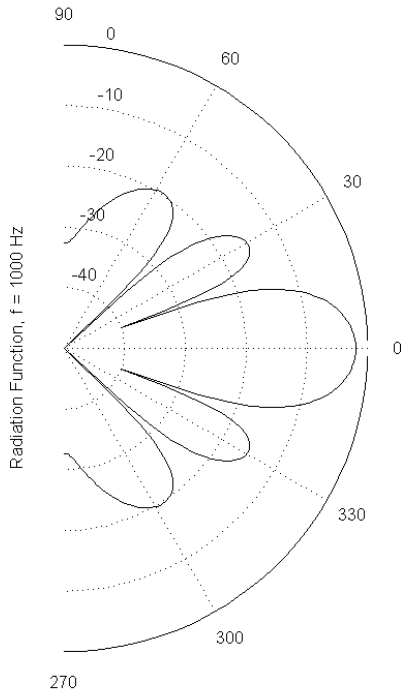


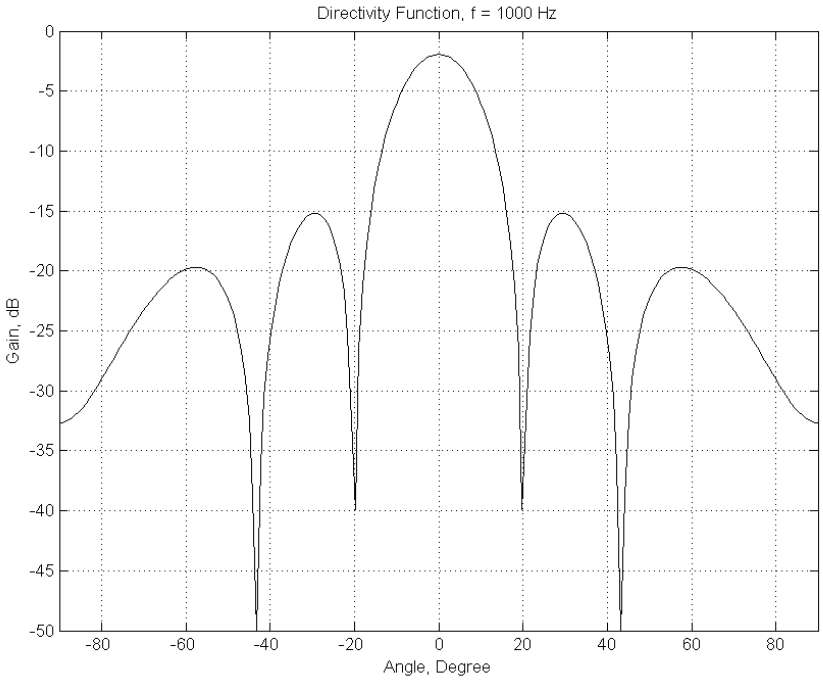
Elevation Plane



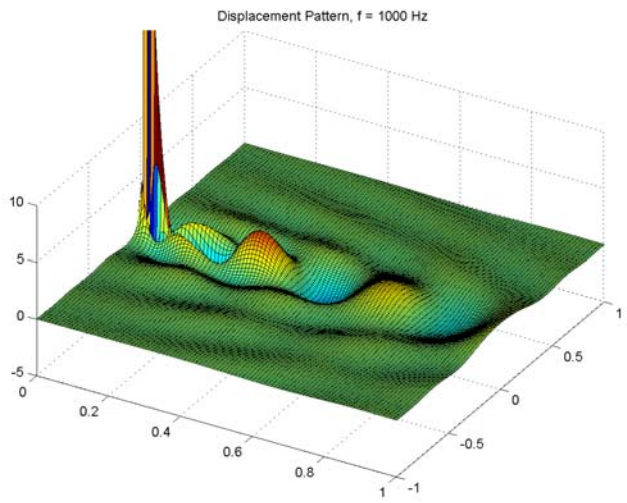
Azimuth Plane



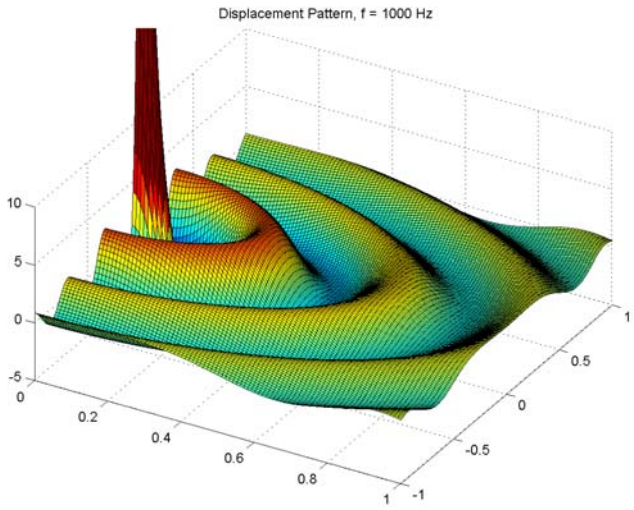




Elevation Plane



Azimuth Plane



Note: Only the graphs for frequencies 250 Hz and 1000 Hz are shown for part 1b, as they are nearly identical to the graphs for part 1a.

Part 1b Source Code:

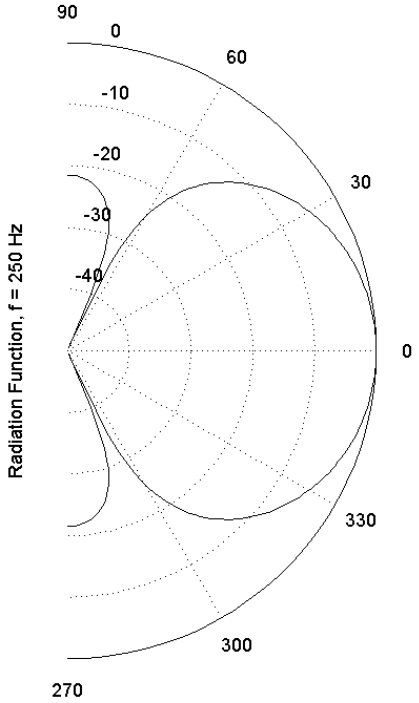
```
% Ross Penniman
% EEN 502
% Project 2, Part 1b

clear all; close all;
b=-1/2:(1/200):1/2;
thetaN=b*pi;
l=1; % length of array in meters
c=343;
f= 125;  % 125, 250, 500, 1000, 2000 Hz
lambda=c/f;
k=2*pi*f/c;

zQ = -l/2:0.001:((l/2)- 0.001);
vzQ = ones(size(zQ));  % constant volume velocity function

% Modify vzQ function for frames around each driver
for j = 1:length(zQ)
    if mod(j, 50) < 6
        vzQ(j) = 0;
    elseif mod(j, 50) < 46
        vzQ(j) = 0.001;
    else
        vzQ(j) = 0;
    end
end
% stem(vzQ);
% axis([1 1001 0 0.002]);

for j=1:length(thetaN)
    temp = 0;
    for t = 1:length(zQ)
        w = k*zQ(t)*sin(thetaN(j));
        temp = temp + (vzQ(t)*cos(w));
    end
    p(j) = temp;
end
% p = p./max(p); % normalize to 1

pdB = 20*log10(abs(p));
pdB(pdB < -50) = -50; % limit lower level of p to -50 dB
posdB = pdB + 50;

figure;  % subplot(121);
polar(thetaN, posdB,'k-');  axis([0 50 -50 50]);
ylabel(['Radiation Function, f = ',num2str(f),' Hz']);

ph=findall(gca,'type','text');  ps=get(ph,'string');
% disp([num2cell(1:numel(ps)).',ps]); % see the content's indices...
ps([5,7,9,12,14,16,17,18,19,20])={'', '', '', '', '', '0', '-10', '-20',
 '-30', '-40'};
set(ph,{'string'},ps);
% print(gcf, ['prob1b_p_' num2str(f)], '-dpng', '-r120')
```

```
figure; % subplot(122);
plot(b*180, pdB,'k'); axis([-90 90 -50 0]); grid on;
title(['Directivity Function, f = ',num2str(f),' Hz']);
xlabel('Angle, Degree'); ylabel('Gain, dB');
% print(gcf, ['prob1b_d_' num2str(f)], '-dpng', '-r120')

[X,Y] = meshgrid(0:0.01:1, -1:0.01:1);
k=2*pi*f/c;
R=sqrt(X.^2+Y.^2);
m=(pi*l/lambda)*(abs(Y)./R); % Y/R = sin(thetaN)
m(m == 0) = 0.00001;
q=(sin(m)./m).*cos(k*R)./R;  % Equation 3.38 from Moser
figure;
surf(X,Y,q);  axis ([0 1 -1 1 -5 10]);  caxis([-5 5]);
title(['Displacement Pattern, f = ',num2str(f),' Hz']);
view (30, 50);
% print(gcf, ['prob1b_3D_' num2str(f)], '-dpng', '-r160')

q2 = cos(k*R)./R;  % azimuth half plane at thetaN = 0
figure;
surf(X,Y,q2);  axis ([0 1 -1 1 -5 10]);  caxis([-5 5]);
title(['Displacement Pattern, f = ',num2str(f),' Hz']);
view (30, 50);
% print(gcf, ['prob1b_3Daz_' num2str(f)], '-dpng', '-r160')
```

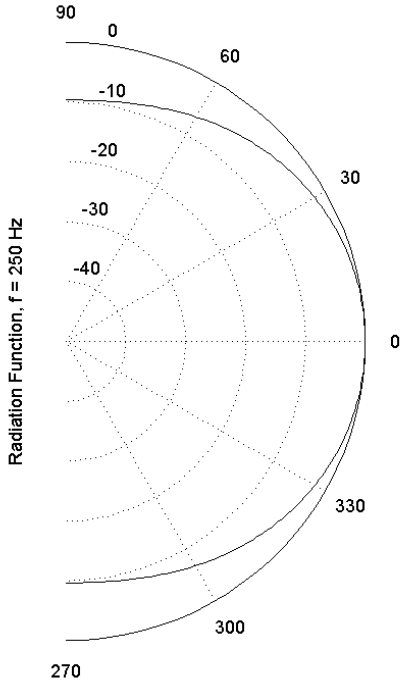
Part 1c: The acoustic performance of the arrays with and without frames around the drivers is nearly identical. The only noticeable difference is the slightly lower gain in part b due to the reduced surface area of the drivers.

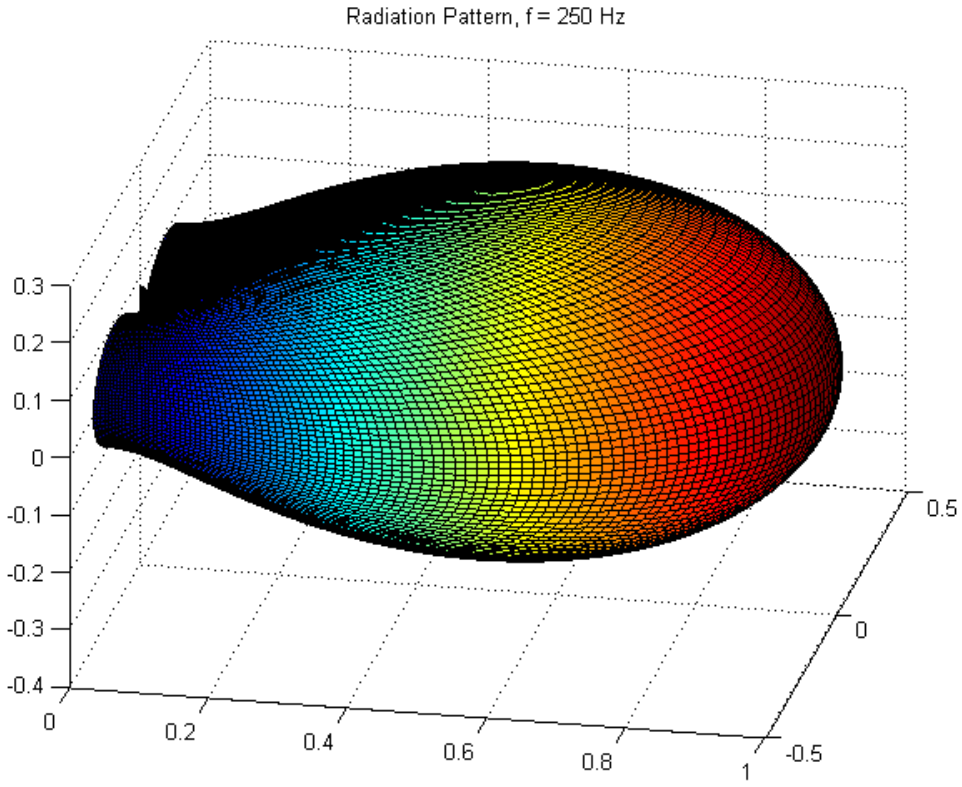
Part 2/3a:

Elevation Plane

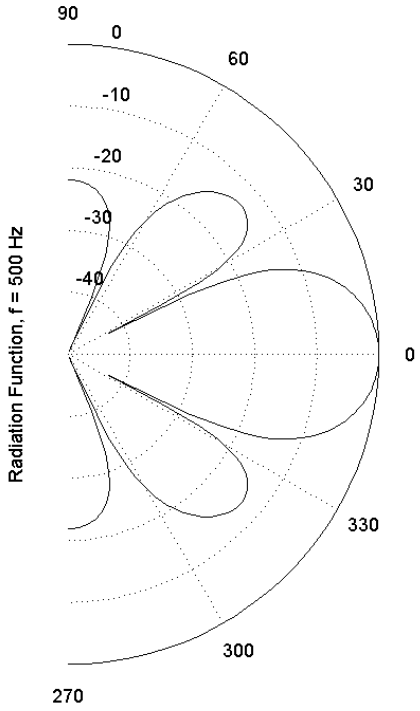


Azimuth Plane

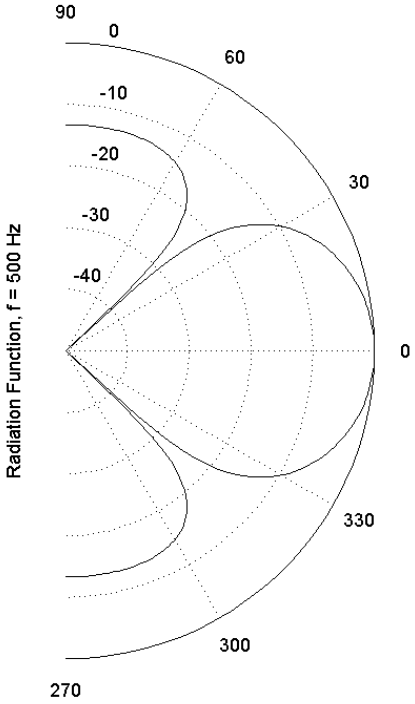


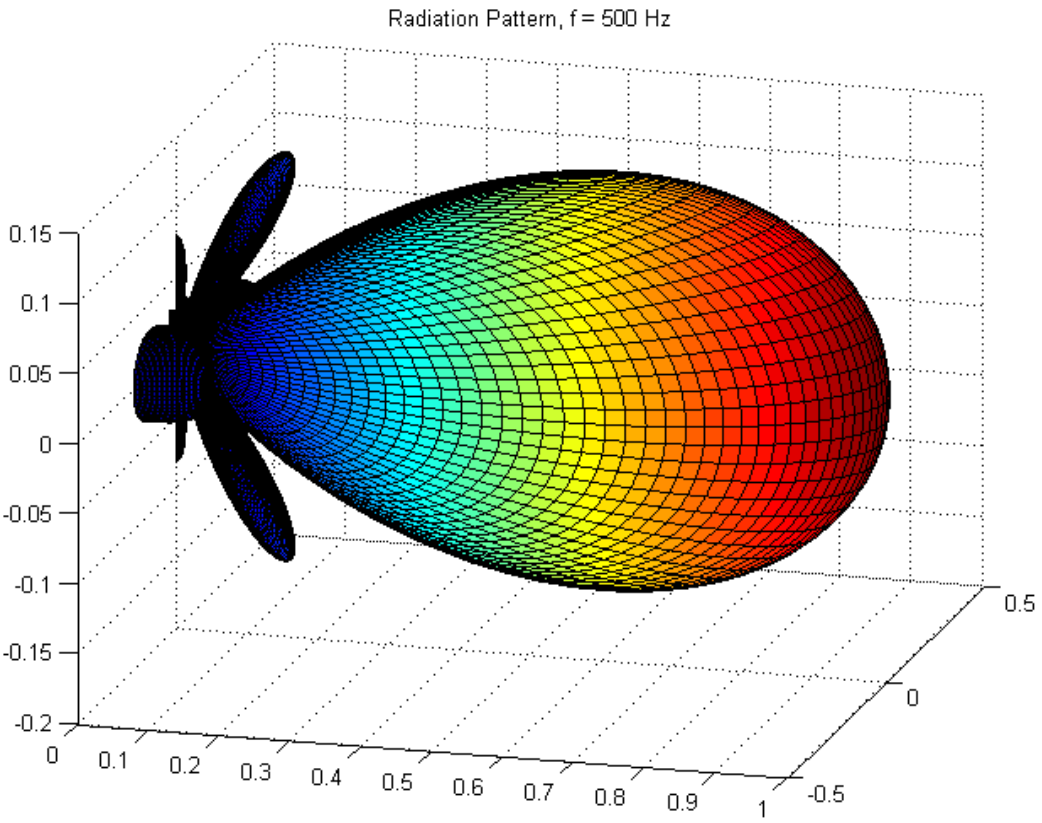


Elevation Plane

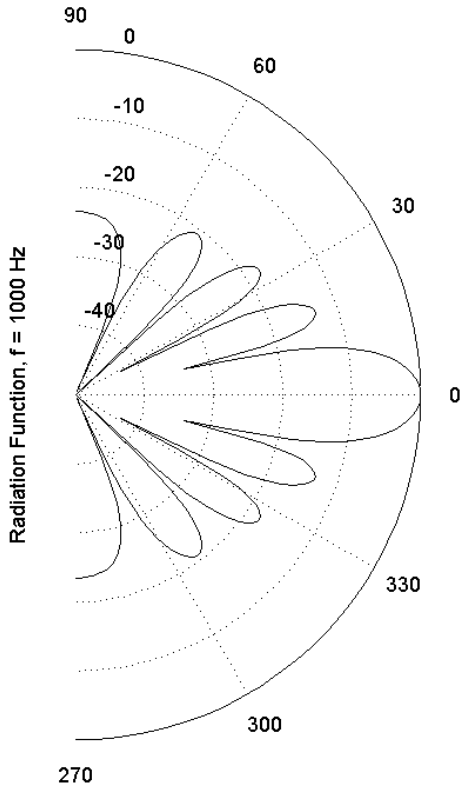


Azimuth Plane

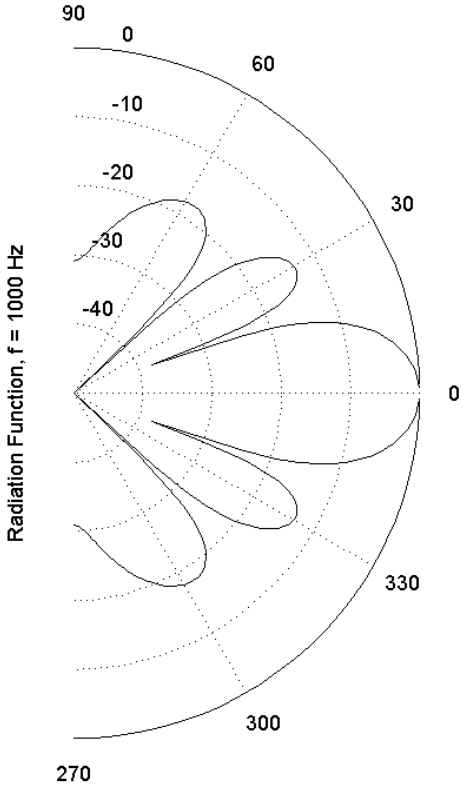


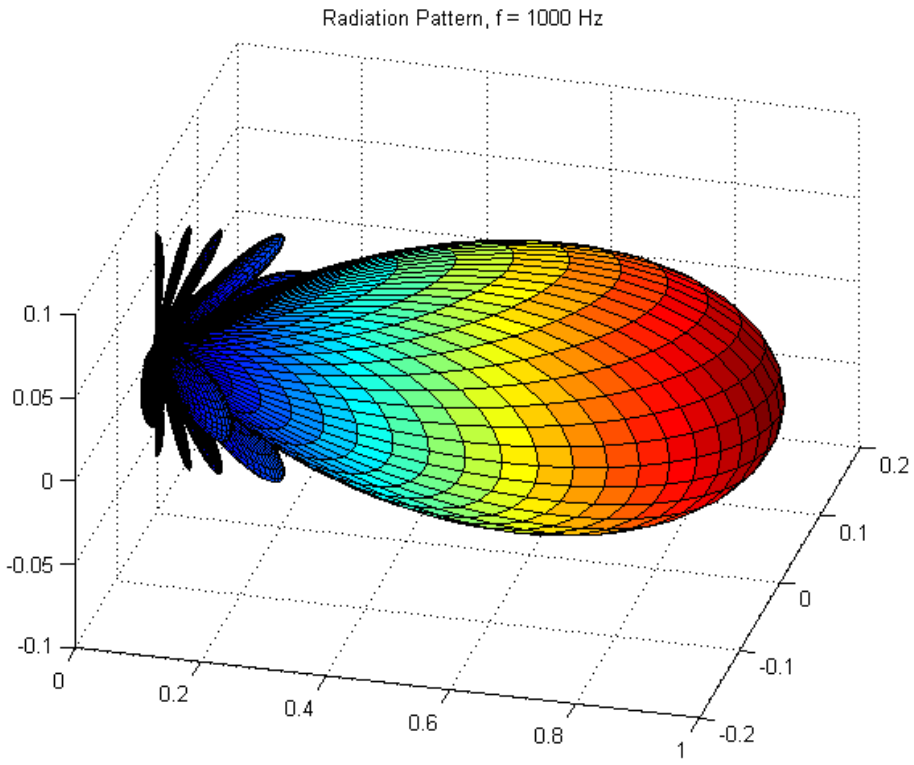


Elevation Plane

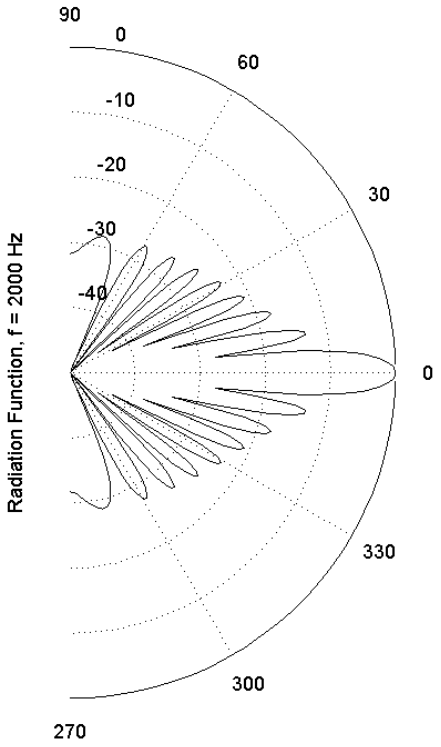


Azimuth Plane

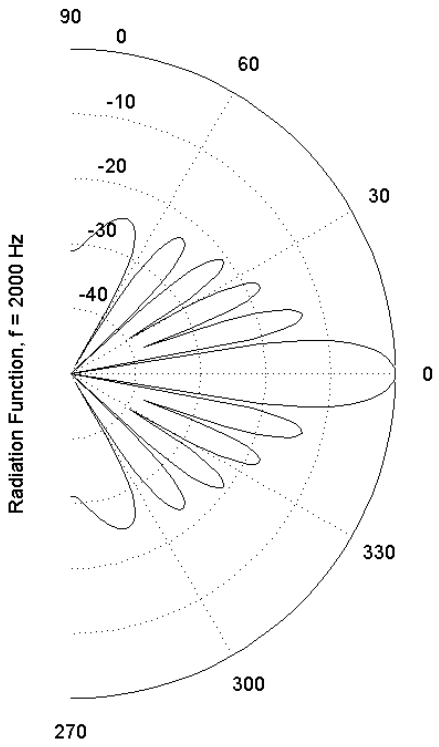


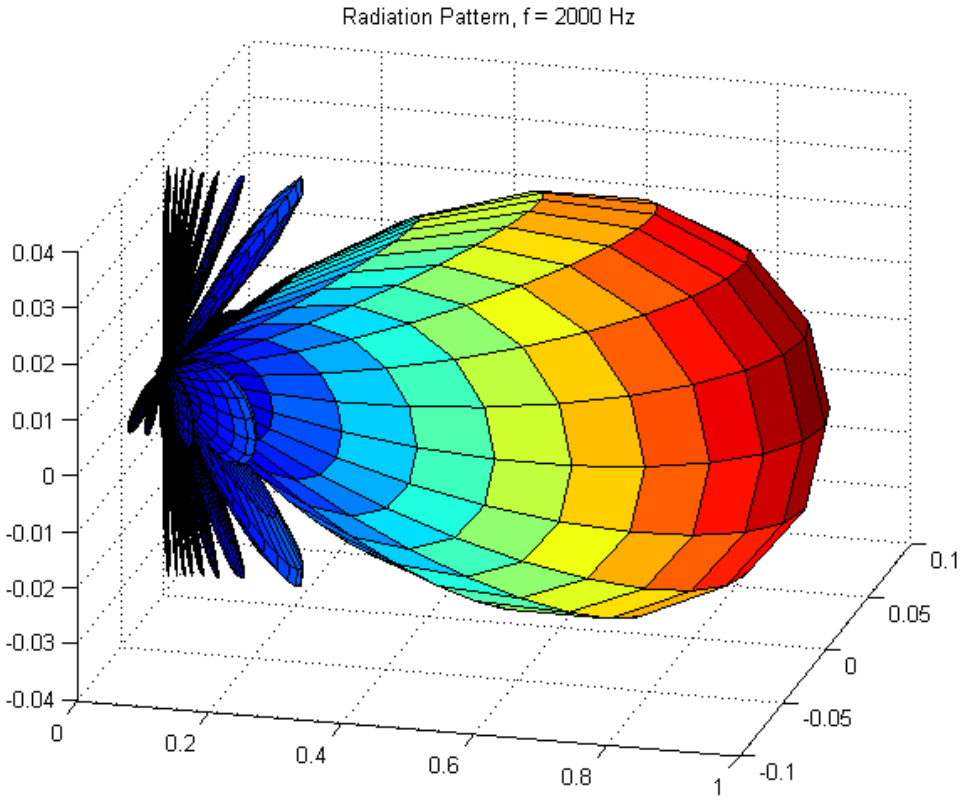


Elevation Plane



Azimuth Plane





Part 2/3a Source Code:

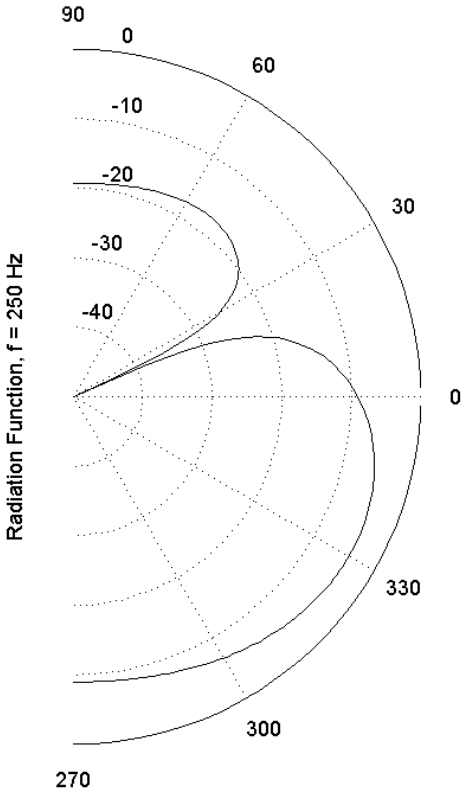
```
% Ross Penniman
% EEN 502
% Project 2, Part 3a

clear all, close all
b=-1/2:(1/200):1/2;
thetaN=b*pi;
lx=1; % length of array in meters
ly=1.5; % height of array in meters
c=343;
f= 2000;  % 250, 500, 1000, 2000 Hz
lambda=c/f;

% ************************************************************
phi=0;  % projected on x=0 plane (azimuth plane)
for j=1:length(thetaN)
    ux = (lx/lambda)*sin(thetaN(j))*cos(phi);
    uy = (ly/lambda)*sin(thetaN(j))*sin(phi);
    pAz(j)=sinc(ux)*sinc(uy);
end

pdBAz = 20*log10(abs(pAz));
pdBAz(pdBAz < -50) = -50; % limit lower level of p to -50 dB
pdBAz = pdBAz + 50;

figure;
polar(thetaN, pdBAz,'k-');  axis([0 50 -50 50]);
ylabel(['Radiation Function, f = ',num2str(f),' Hz']);

ph=findall(gca,'type','text');  ps=get(ph,'string');
% disp([num2cell(1:numel(ps)).',ps]); % see the content's indices
ps([5,7,9,12,14,16,17,18,19,20])={'', '', '', '', '', '0', '-10', '-20', '-30',
'-40'};
set(ph,{'string'},ps);
print(gcf, ['prob3a_az_' num2str(f)], '-dpng', '-r120')
% ************************************************************

% ************************************************************
phi=pi/2; % projected on y=0 plane (elevation plane)
for j=1:length(thetaN)
    ux = (lx/lambda)*sin(thetaN(j))*cos(phi);
    uy = (ly/lambda)*sin(thetaN(j))*sin(phi);
    pEl(j)=sinc(ux)*sinc(uy);
end

pdBEl = 20*log10(abs(pEl));
pdBEl(pdBEl < -50) = -50; % limit lower level of p to -50 dB
pdBEl = pdBEl + 50;

figure;
polar(thetaN, pdBEl,'k-');  axis([0 50 -50 50]);
ylabel(['Radiation Function, f = ',num2str(f),' Hz']);
```

```
ph=findall(gca,'type','text');  ps=get(ph,'string');
% disp([num2cell(1:numel(ps)).',ps]); % see the content's indices
ps([5,7,9,12,14,16,17,18,19,20])={'', '', '', '', '', '0', '-10', '-20', '-30',
'-40'};
set(ph,{'string'},ps);
print(gcf, ['prob3a_el_' num2str(f)], '-dpng', '-r120')
% ************************************************************

R = abs(pEl' * pAz);
[theta, phi] = meshgrid(thetaN, thetaN);
[X, Y, Z] = sph2cart(theta, phi, R);

figure;
surf(X, Y, Z, X);
title(['Radiation Pattern, f = ',num2str(f),' Hz']);
```

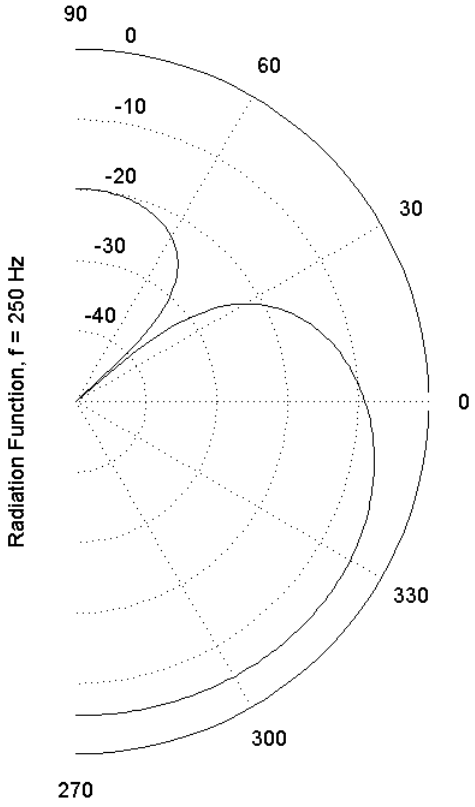
Part 3b:

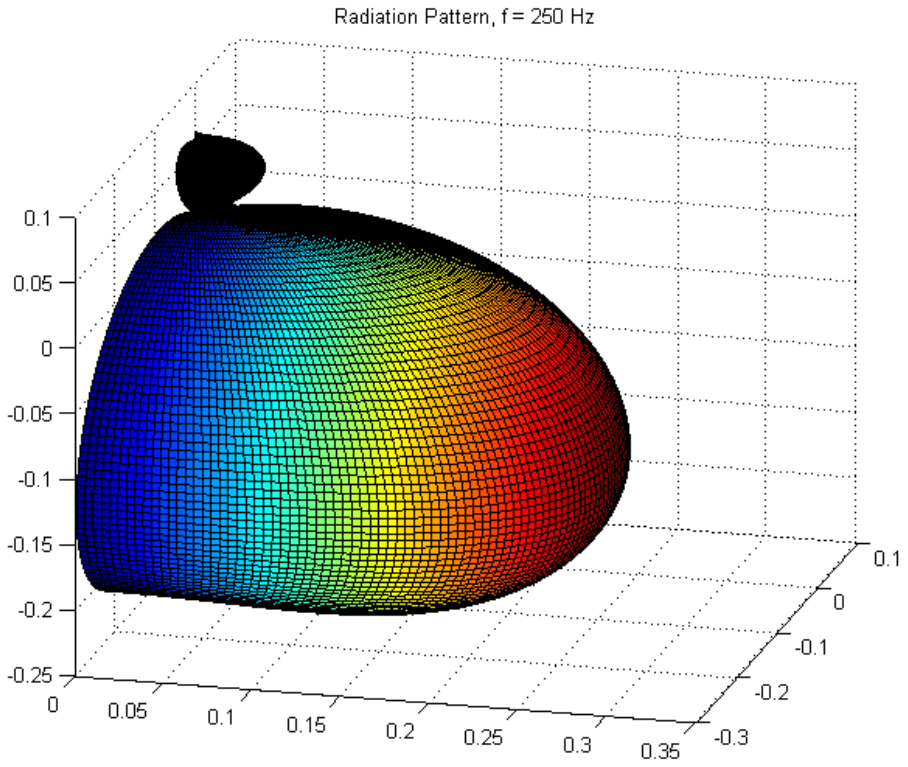
By applying a delay across the elements of the loudspeaker array, the beam can be steered in both the elevation and azimuth planes.

Elevation Plane

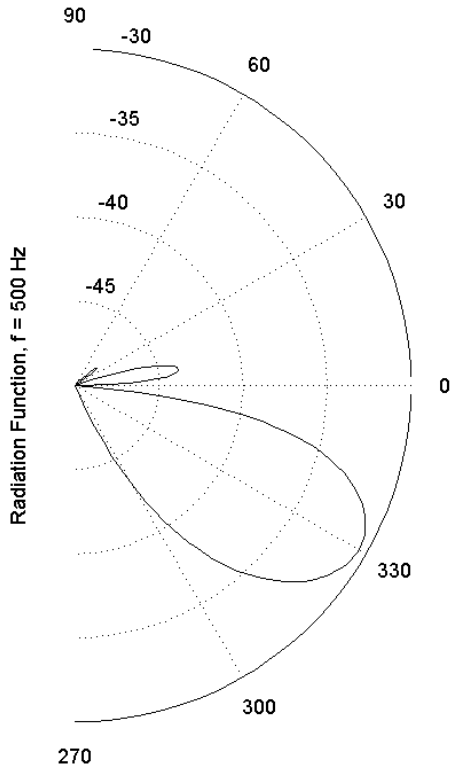


Azimuth Plane

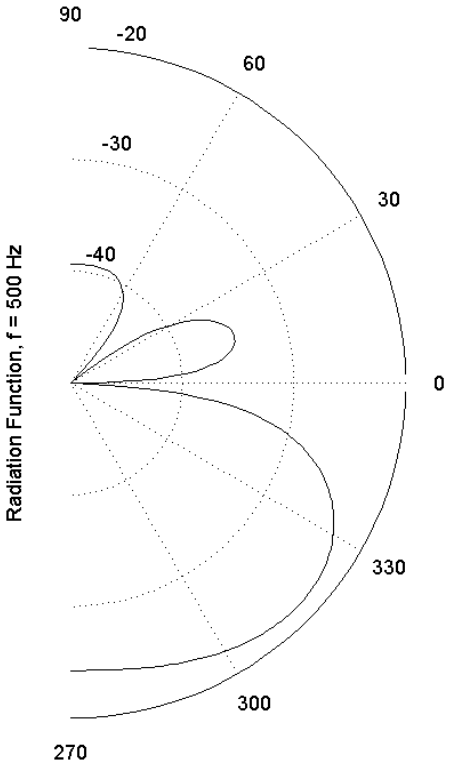


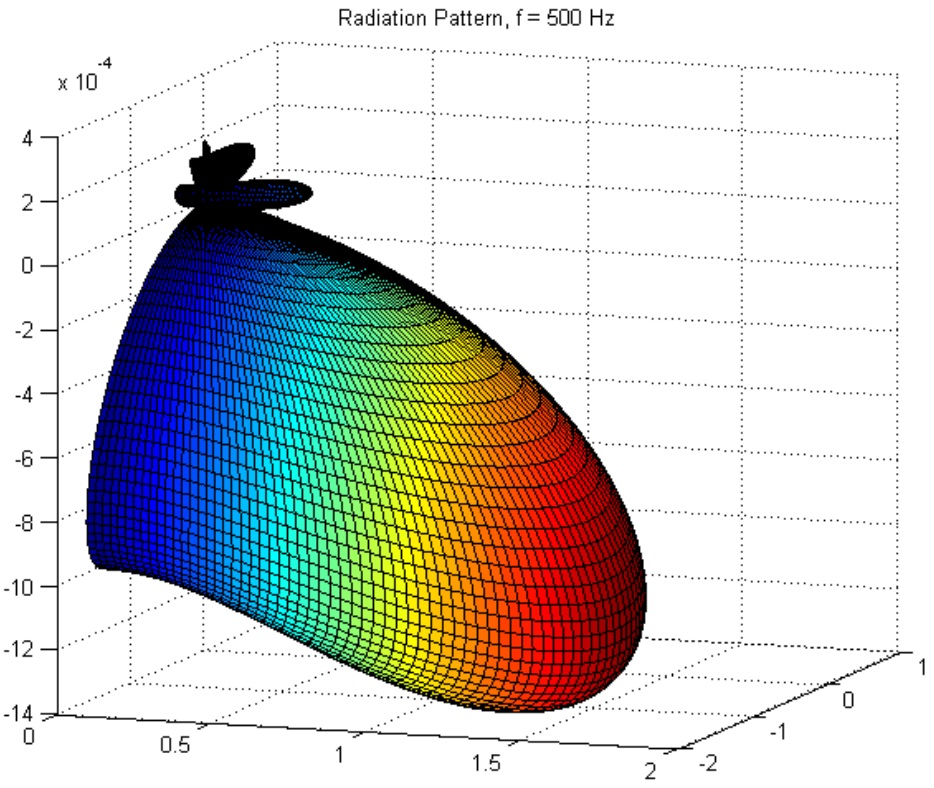


Elevation Plane

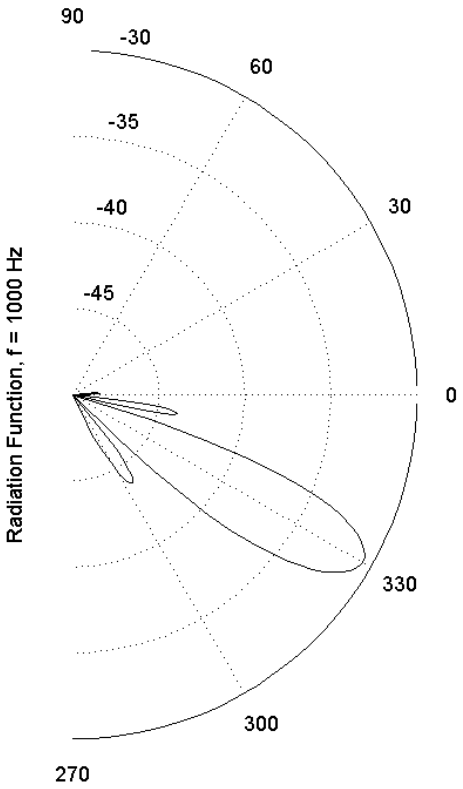


Azimuth Plane

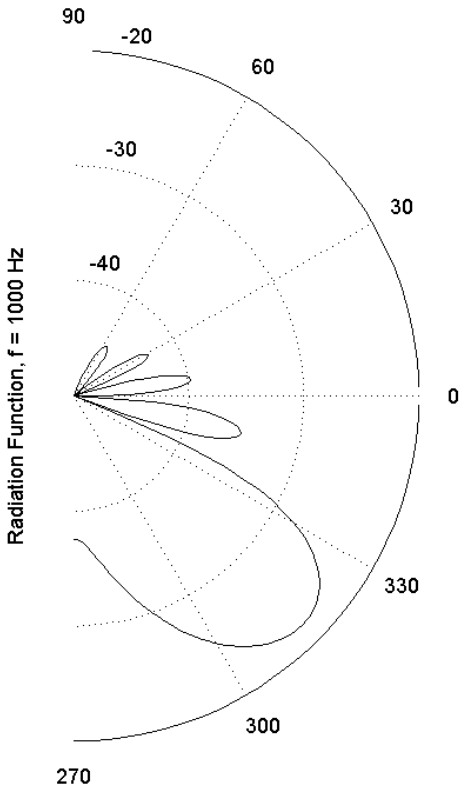


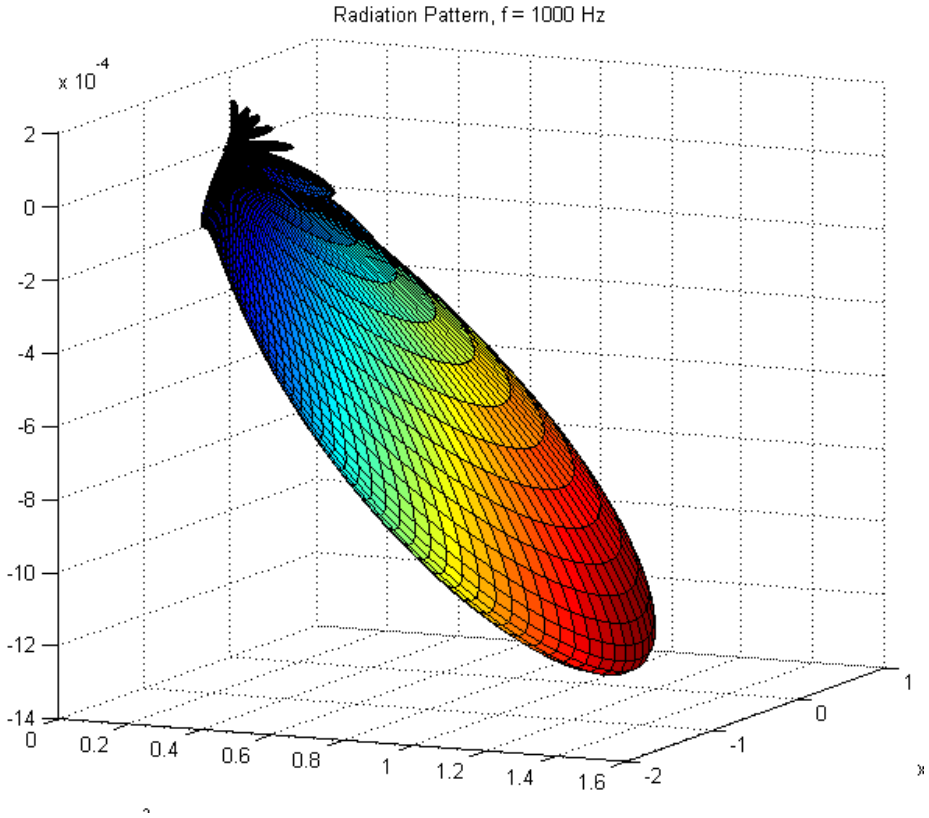


Elevation Plane

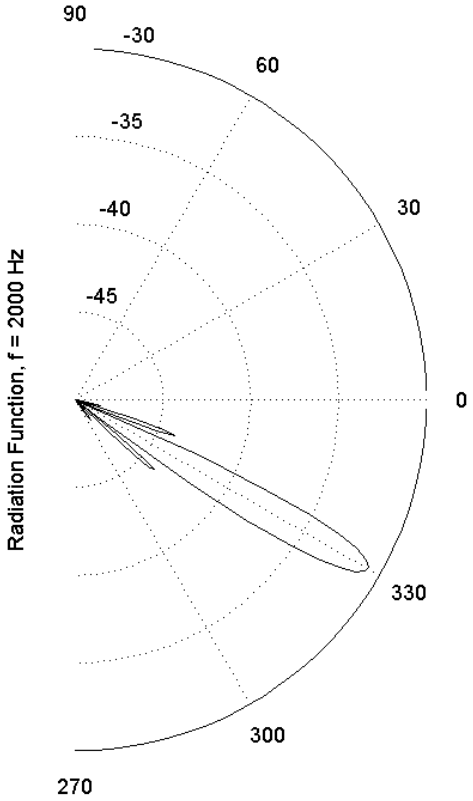


Azimuth Plane

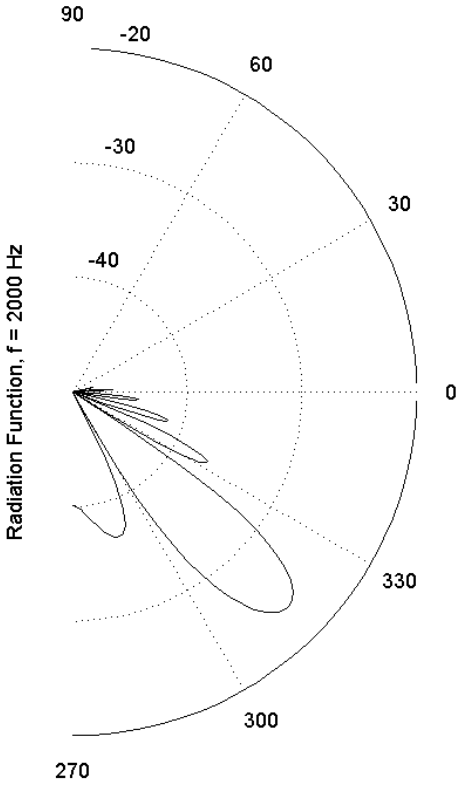


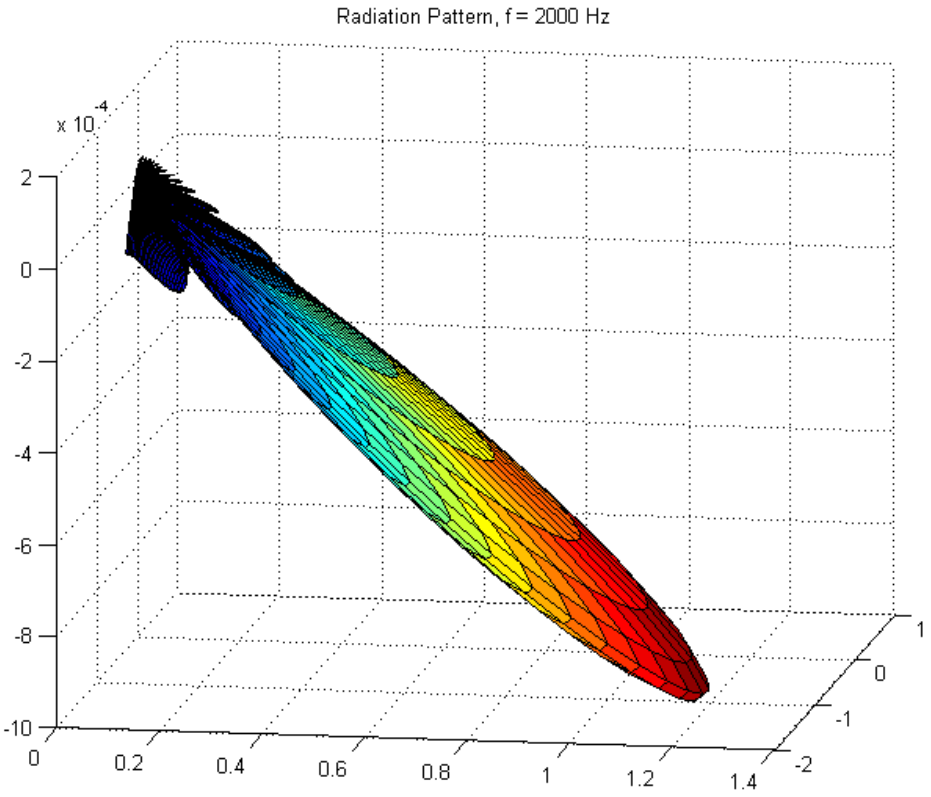


Elevation Plane



Azimuth Plane





Part 3b Source Code:

```
% Ross Penniman
% EEN 502
% Project 2, Part 3b

clear all, close all
b=-1/2:(1/200):1/2;
thetaN=b*pi;
lx=1; % length of array in meters
ly=1.5; % height of array in meters
c=343;
f= 2000;  % 250, 500, 1000, 2000 Hz
lambda=c/f;

elSteer = -30*pi/180; % convert to radians
azSteer = -45*pi/180;

lambEl = lambda/sin(elSteer);
lambAz = lambda/sin(azSteer);

% ************************************************************
phi=0;  % projected on x=0 plane (azimuth plane)
for j=1:length(thetaN)
    ux = ((lx/lambda)*sin(thetaN(j))*cos(phi)) - (lx/lambAz);
    uy = ((ly/lambda)*sin(thetaN(j))*sin(phi)) - (ly/lambEl);
    pAz(j)=sinc(ux)*sinc(uy);
end

pdBAz = 20*log10(abs(pAz));
pdBAz(pdBAz < -50) = -50; % limit lower level of p to -50 dB
pdBAz = pdBAz + 50;

figure;
polar(thetaN, pdBAz,'k-'); axis([0 30 -30 30]);    % axis([0 50 -50 50]);
ylabel(['Radiation Function, f = ',num2str(f),' Hz']);

ph=findall(gca,'type','text');  ps=get(ph,'string');
% disp([num2cell(1:numel(ps)).',ps]); % see the content's indices
ps([5,7,9,12,14,16,17,18])={'', '', '', '', '', '-20', '-30', '-40'};
% ps([5,7,9,12,14,16,17,18,19,20])={'', '', '', '', '', '0', '-10', '-20', '-30',
'-40'};
set(ph,{'string'},ps);
print(gcf, ['prob3b_az_' num2str(f)], '-dpng', '-r120')
% ************************************************************

% ************************************************************
phi=pi/2; % projected on y=0 plane (elevation plane)
for j=1:length(thetaN)
    ux = ((lx/lambda)*sin(thetaN(j))*cos(phi)) - (lx/lambAz);
    uy = ((ly/lambda)*sin(thetaN(j))*sin(phi)) - (ly/lambEl);
    pEl(j)=sinc(ux)*sinc(uy);
end
```

```
pdBEl = 20*log10(abs(pEl));
pdBEl(pdBEl < -50) = -50; % limit lower level of p to -50 dB
pdBEl = pdBEl + 50;

figure;
polar(thetaN, pdBEl,'k-');  axis([0 20 -20 20]);   % axis([0 50 -50 50]);
ylabel(['Radiation Function, f = ',num2str(f),' Hz']);

ph=findall(gca,'type','text');  ps=get(ph,'string');
% disp([num2cell(1:numel(ps)).',ps]); % see the content's indices
ps([5,7,9,12,14,16,17,18,19])={'', '', '', '', '', '-30', '-35', '-40', '-45'};
% ps([5,7,9,12,14,16,17,18,19,20])={'', '', '', '', '', '0', '-10', '-20', '-30',
'-40'};
set(ph,{'string'},ps);
print(gcf, ['prob3b_el_' num2str(f)], '-dpng', '-r120')
% ************************************************************

R = abs(pEl' * pAz);
[theta, phi] = meshgrid(thetaN, thetaN);
[X, Y, Z] = sph2cart(theta, phi, R);

figure;
surf(X, Y, Z, X);
title(['Radiation Pattern, f = ',num2str(f),' Hz']);
```