PRÁCTICA 4 RESUELTA + PLOTS: radar cross section

Radiolocalització (Universitat Politècnica de Catalunya)

PRÁCTICA 4

MAIN

```
clear all;
clear all figures;
%% 5.1 RCS of an sphere --> dependency: frequency of the radar
%% 5.2 RCS of a circular flat plate --> dependency: aspect angle
%INPUTS: frequency [Hz], radius [m]
%OUTPUT: aspect angle [°], RCS [dB]
r=10000;
freq=10;
[vaspect_deg, rcs_dB] = rcs_circ_plate (r, freq);
lambda=3e8/freq;
area=pi*r^2;

%comprovem que per theta=0°-->RCS=(4pi*Area^2)/(lambda^2)
normal_incidence_RCS=10*log10((4*pi*area^2)/(lambda^2));

%% 5.4 Two spheres RCS
%==============RCS COMPUTATION FUNCTION===================
%-----I N P U T S------
%RCS_scatters = Column vector of RCSs of spheres [dBsm=dBm2]
%scatters_rect_coord = Matrix having the rectangular coordinates of the
scatterers positions:
%           x-coordinates in first column [m]
%           y-coordinates in second column [m]
%           z-coordinates in third column [m]
%radar_sphere_coord = Matrix having the spherical coordinates of the
different radar locations
%           r-coordinates in first column [m]
%           theta-coordinates in second column [rad]
%           phi-coordinates in third column [rad]
%carrier_freq=Carrier frequency [Hz]

%-----O U T P U T S------
% monostatic_RCS = Column vector of monostatic RCSs seen from the
different radar locations [m^2]

%==============FOR ONE VALUE OF L=========================
%--> i n p u t s   a j u s t a b l e s
L=1; %-->electrical separation between scatters
carrier_freq=1.5*10^9;
RCS_scatters=[0,0]; %0dBsm-->1m^2 isotropical scatters
    %coordinates of scatters
scatters_rect_coord=[-L/2 0 0; L/2 0 0];
    %coordinates of radars
r_gran=10^6;
if(r_gran<(2*L^2)*carrier_freq/(3e8))    %   condition   to   fulfill
r_gran<(2L^2/lambda)
    disp('Distance between scatters and radar stations too small.');
end
theta=pi/2; %-->horizontal plane
phi=0:pi/1000:2*pi; %--> all angles of the horizontal plane

%--> e x e c u c i ó
for i=1:length(phi)
    radar_sphere_coord(i,:)=[r_gran theta phi(i)];
end
```

```
monostatic_RCS=RCS_computation(RCS_scatters,scatters_rect_coord,radar_
sphere_coord,carrier_freq);

figure(1);
plot(phi,10*log10(monostatic_RCS)); %RCS [dBsm]
xlabel('Phi angle (rad)');
ylabel('RCS (dBsm)');
str = sprintf('Resulting RCS for two spheres (1m^2) separated %0.2f
m',L);
title(str);
figure(2);
polarplot(phi,monostatic_RCS); %RCS [m^2]
str = sprintf('Resulting RCS for two spheres (1m^2) separated %0.2f
m',L);
title(str);


%===============FOR DIFFERENT VALUES OF L====================
%--> i n p u t s   a j u s t a b l e s
L=[1 0.1 0.01];
j=1;
carrier_freq=1.5*10^9;

while(j<=length(L))
RCS_scatters=[0,0]; %0dBsm-->1m^2
scatters_rect_coord=[-L(j)/2 0 0; L(j)/2 0 0];
r_gran=10^6;
theta=pi/2;
phi=0:pi/1000:2*pi;

%--> e x e c u c i ó
for i=1:length(phi)
    radar_sphere_coord(i,:)=[r_gran theta phi(i)];
end

monostatic_RCS=RCS_computation(RCS_scatters,scatters_rect_coord,radar_
sphere_coord,carrier_freq);

figure(3);
subplot(length(L),2,j+(j-1))
plot(phi,10*log10(monostatic_RCS)); %RCS [dBsm]
xlabel('Phi angle (rad)');
ylabel('RCS (dBsm)');
subplot(length(L),2,(j+1)+(j-1))
polarplot(phi,monostatic_RCS); %RCS [m^2]
str = sprintf('Resulting RCS for two spheres with 1m^2 separated %0.2f
m',L(j));
title(str);
j=j+1;
end

%% 5.5 Swerling model statistics I and II
%==============RCS COMPUTATION FUNCTION===================
%-----I N P U T S------
%RCS_scatters = Column vector of RCSs of spheres [dBsm=dBm2]
%scatters_rect_coord = Matrix having the rectangular coordinates of the
scatterers positions:
%            x-coordinates in first column [m]
%            y-coordinates in second column [m]
%            z-coordinates in third column [m]
```

```
%radar_sphere_coord = Matrix having the spherical coordinates of the different radar locations
%           r-coordinates in first column [m]
%           theta-coordinates in second column [rad]
%           phi-coordinates in third column [rad]
%carrier_freq=Carrier frequency [Hz]

%-----O U T P U T S------
% monostatic_RCS = Column vector of monostatic RCSs seen from the different radar locations [m^2]

%--> i n p u t s   a j u s t a b l e s
L=1;
carrier_freq=3*10^9;
num_scatters=10; %10 scatters in the scenario
RCS_scatters=ones(1,num_scatters)*0.0000001; %dBsm RCSs close to 0 dBsm;

    %coordinates of scatters DISTRIBUTED RANDOMLY
circle_radius=10; %circle of 10m
r_scatter=0+circle_radius*rand(1,num_scatters);
theta_scatter=ones(1,num_scatters)*pi/2;
phi_scatter=0+2*pi*rand(1,num_scatters);
x_scatter=r_scatter.*sin(theta_scatter).*cos(phi_scatter);
y_scatter=r_scatter.*sin(theta_scatter).*sin(phi_scatter);
z_scatter=r_scatter.*cos(theta_scatter);
scatters_rect_coord=[x_scatter; y_scatter; z_scatter]';

    %coordinates of radar positions
r_gran=1000; %1km
theta=pi/2; %-->horizontal plane
phi=0:pi/1000:2*pi;%--> all angles of the horizontal plane

%--> e x e c u c i ó
for i=1:length(phi)
    radar_sphere_coord(i,:)=[r_gran theta phi(i)];
end

monostatic_RCS=RCS_computation(RCS_scatters,scatters_rect_coord,radar_
sphere_coord,carrier_freq);

figure(1);
histogram(monostatic_RCS,'Normalization','pdf'); %RCS [m^2]
data_hist_1=histcounts(monostatic_RCS,'Normalization','pdf');
ylabel('RCS probability density function');
xlabel('RCS (m^2)');
title(sprintf('PDF of RCS resulting from %g spheres with 0dBsm distributed randomly',num_scatters));

figure(2);
subplot(2,1,1)
plot(phi,10*log10(monostatic_RCS)); %RCS [dBsm]
xlabel('Phi angle (rad)');
ylabel('RCS (dBsm)');
subplot(2,1,2)
polarplot(phi,monostatic_RCS); %RCS [m^2]
suptitle(sprintf('Resulting RCS for %g spheres with 0dBsm distributed randomly',num_scatters));

%% 5.6 Swerling model statistics III and IV
%==============RCS COMPUTATION FUNCTION====================
%-----I N P U T S------
```

```
%RCS_scatters = Column vector of RCSs of spheres [dBsm=dBm2]
%scatters_rect_coord = Matrix having the rectangular coordinates of the
scatterers positions:
%           x-coordinates in first column [m]
%           y-coordinates in second column [m]
%           z-coordinates in third column [m]
%radar_sphere_coord = Matrix having the spherical coordinates of the
different radar locations
%           r-coordinates in first column [m]
%           theta-coordinates in second column [rad]
%           phi-coordinates in third column [rad]
%carrier_freq=Carrier frequency [Hz]

%-----O U T P U T S------
% monostatic_RCS = Column vector of monostatic RCSs seen from the
different radar locations [m^2]

%--> i n p u t s   a j u s t a b l e s
L=1;
carrier_freq=3*10^9;
num_scatters=10; %10 scatters in the scenario
RCS_scatters=ones(1,num_scatters)*0.0000001; %dBsm RCSs close to 0 dBsm;
RCS_scatters(8)=RCS_scatters(8)+17; %one scatter 17dB greater

    %coordinates of scatters DISTRIBUTED RANDOMLY
circle_radius=10; %circle of 10m
r_scatter=0+circle_radius*rand(1,num_scatters);
theta_scatter=ones(1,num_scatters)*pi/2;
phi_scatter=0+2*pi*rand(1,num_scatters);
x_scatter=r_scatter.*sin(theta_scatter).*cos(phi_scatter);
y_scatter=r_scatter.*sin(theta_scatter).*sin(phi_scatter);
z_scatter=r_scatter.*cos(theta_scatter);
scatters_rect_coord=[x_scatter; y_scatter; z_scatter]';

    %coordinates of radar positions
r_gran=1000; %1km
theta=pi/2; %-->horizontal plane
phi=0:pi/1000:2*pi;%--> all angles of the horizontal plane

%--> e x e c u c i ó
for i=1:length(phi)
    radar_sphere_coord(i,:)=[r_gran theta phi(i)];
end

monostatic_RCS=RCS_computation(RCS_scatters,scatters_rect_coord,radar_
sphere_coord,carrier_freq);

figure(3);
histogram(monostatic_RCS,'Normalization','pdf'); %RCS [m^2]
data_hist_1=histcounts(monostatic_RCS,'Normalization','pdf');
ylabel('RCS probability density function');
xlabel('RCS (m^2)');
title(sprintf('PDF of RCS resulting from %g spheres with 0dBsm (except
one 17dB greater) distributed randomly',num_scatters));

figure(4);
subplot(2,1,1)
plot(phi,10*log10(monostatic_RCS)); %RCS [dBsm]
xlabel('Phi angle (rad)');
ylabel('RCS (dBsm)');
subplot(2,1,2)
```

```
polarplot(phi,monostatic_RCS); %RCS [m^2]
suptitle(sprintf('Resulting RCS for %g spheres with 0dBsm (except one
17dB greater) distributed randomly',num_scatters));
```

FUNCTION: RCS_computation

```
function
[monostatic_RCS]=RCS_computation(RCS_scatters,scatters_rect_coord,rada
r_sphere_coord,carrier_freq)

%coord esfèriques dels radars
r=radar_sphere_coord(:,1);
theta=radar_sphere_coord(:,2);
phi=radar_sphere_coord(:,3);

%coord cartesianes dels radars
x_radar=r.*sin(theta).*cos(phi);
y_radar=r.*sin(theta).*sin(phi);
z_radar=r.*cos(theta);

%coord cartesianes dels scatters
x_scatter=scatters_rect_coord(:,1);
y_scatter=scatters_rect_coord(:,2);
z_scatter=scatters_rect_coord(:,3);

c=3*10^8;
lambda=c/carrier_freq;
RCS_scatters=10.^(RCS_scatters/10);

%per cada radar-->tots els scatters
for k=1:length(x_radar)
    for j=1:length(x_scatter)
    r_n(k,j)=sqrt((x_scatter(j)-x_radar(k)).^2+(y_scatter(j)-
y_radar(k)).^2+(z_scatter(j)-z_radar(k)).^2);
    phi_n(k,j)=2*2*pi/lambda*r_n(k,j);
    end

monostatic_RCS(k)=abs(sum(sqrt(RCS_scatters).*exp(1i*phi_n(k,:)))).^2;
end

end
```

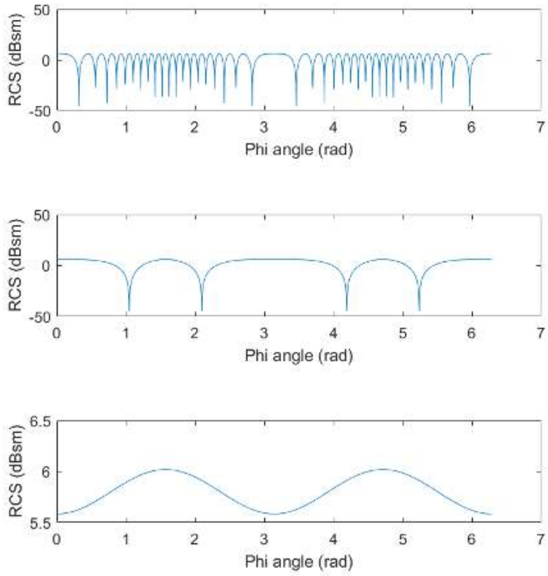
FUNCTION: rcs_circ_plate

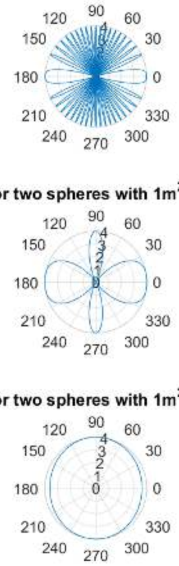
```
function [vaspect_deg, rcs_dB] = rcs_circ_plate (r, freq)
eps = 0.000001;
cdeg2rad=pi/180;

% Compute wavelength
lambda=3e8/freq;
index=0;
vaspect_deg=[0:.1:180];
vaspect_rad=cdeg2rad*vaspect_deg;
vx=(4*pi*r/lambda)*sin(vaspect_rad);
vval1=4*pi^3*r^4/lambda^2;
vval2=2*besselj(1,vx)./vx;
rcs_po2=vval1*(vval2.*cos(vaspect_rad)).^2+eps;
% vval1m=lambda*r;
% vval2m=8*pi*sin(vaspect_rad).*(tan(vaspect_rad).^2);
% rcs_mu2=vval1m./vval2m+eps;
[vindex]=find(or(vaspect_deg==0,vaspect_deg==180));
```

```
rcs_po2(vindex)=(4*pi^3*r^4/lambda^2)+eps;
% rcs_mu2(vindex)=rcs_po2(vindex);
rcsdb_po2=10*log10(rcs_po2);
% rcsdb_mu2=10*log10(rcs_mu2);

figure; set(gcf,'Color','w');
plot(vaspect_deg,rcsdb_po2,'b'); hold on; grid on;
% plot(vaspect_deg,rcsdb_mu2,'b--');
xlabel ('Aspect angle [°]');
ylabel ('RCS [dBsm]');
rcs_dB=[rcsdb_po2];
```

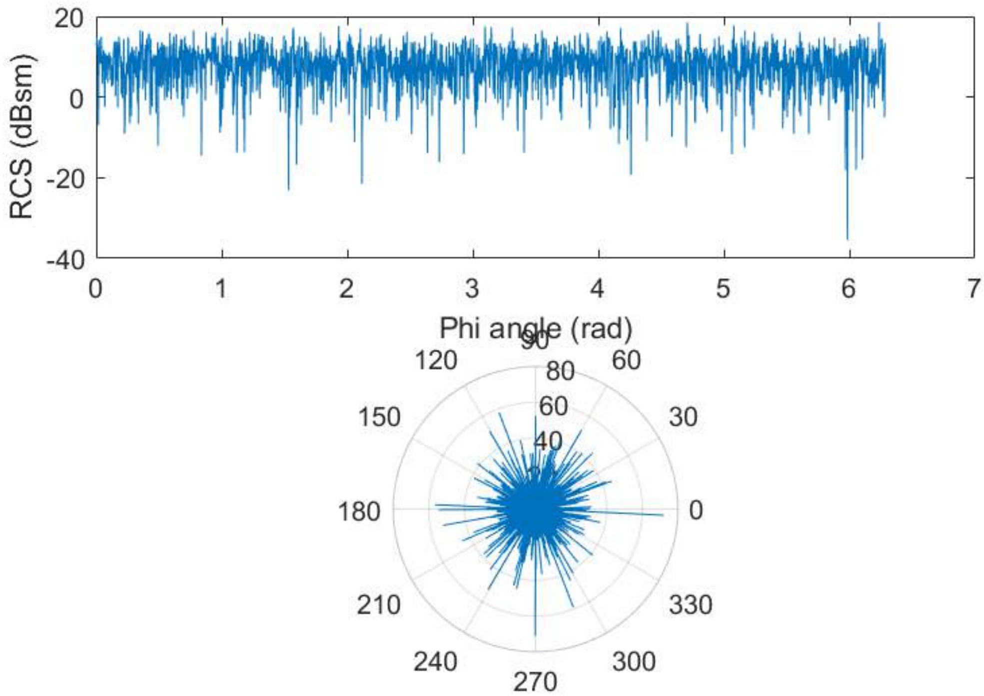
FUNCTION: rcs_sphere

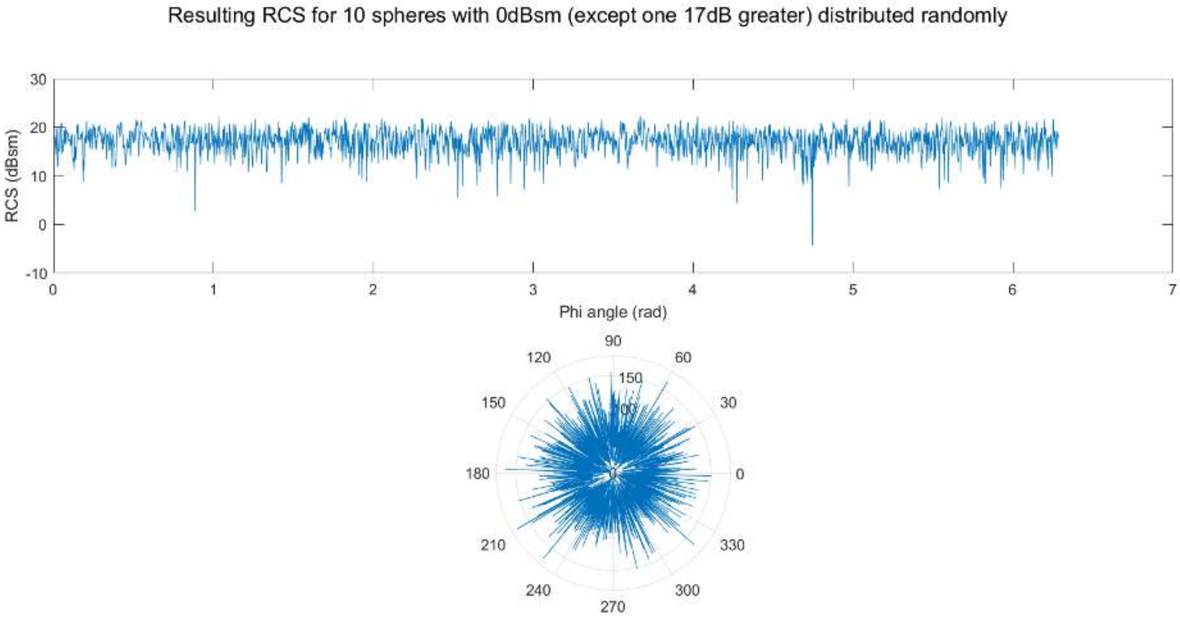
```
eps = 0.00001;
index = 0;
% kr limits are [0.05 - 15] ===> 300 points

for kr = 0.05:0.05:15
 index = index + 1;
 sphere_rcs = 0. + 0.*i;
 f1 = 0. + 1.*i;
 f2 = 1. + 0.*i;
 m = 1.;
 n = 0.;
 q = -1.;
 % initially set del to huge value
 del =100000+100000*i;

 while(abs(del) > eps)
 q = -q;
 n = n + 1;
 m = m + 2;
 del = (2.*n-1) * f2 / kr-f1;
 f1 = f2;
 f2 = del;
 del = q * m /(f2 * (kr * f1 - n * f2));
 sphere_rcs = sphere_rcs + del;
 end

 rcs(index) = abs(sphere_rcs);
 sphere_rcsdb(index) = 10. * log10(rcs(index));
end

figure; set(gcf,'Color','w');
n=0.05:.05:15;
plot (n,rcs,'k');
set (gca,'xtick',[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]);
xlabel ('Sphere circumference in wavelengths (2{\pi}r/{\lambda})');
ylabel ('Normalized sphere RCS [adim.]');
grid;
figure; set(gcf,'Color','w');
plot (n,sphere_rcsdb,'k');
set (gca,'xtick',[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]);
xlabel ('Sphere circumference in wavelengths (2{\pi}r/{\lambda})');
ylabel ('Normalized sphere RCS [dB]');
grid;
figure; set(gcf,'Color','w');
semilogx (n,sphere_rcsdb,'k'); grid on;
xlabel ('Sphere circumference in wavelengths (2{\pi}r/{\lambda})');
ylabel ('Normalized sphere RCS [dB]');
```

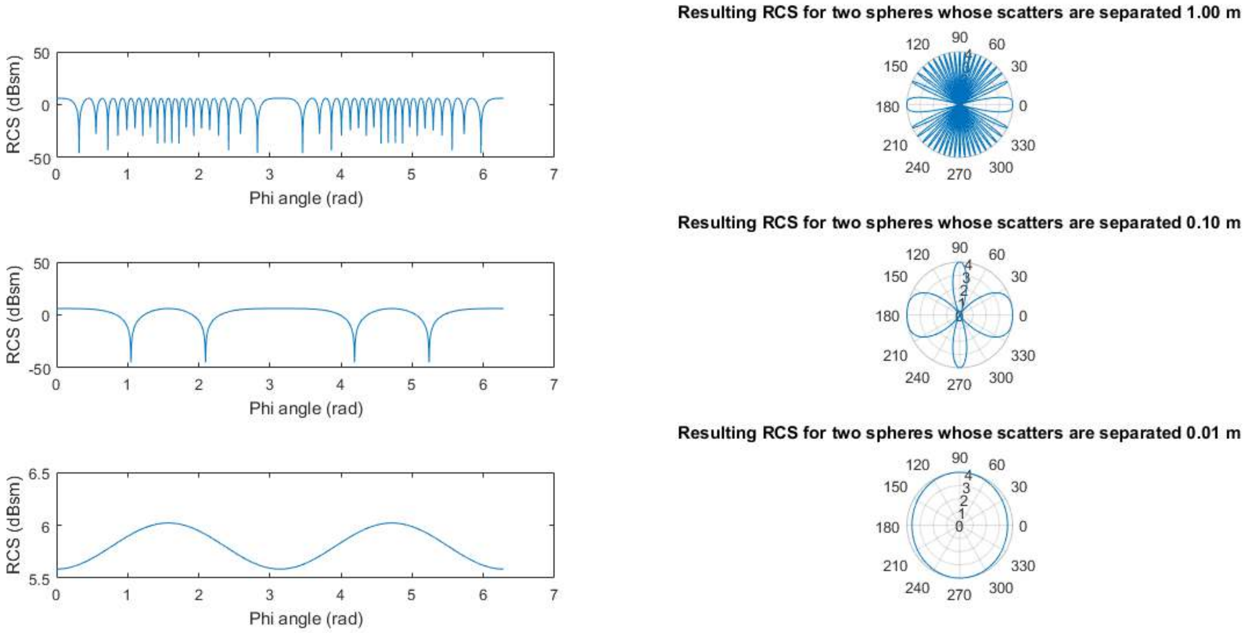


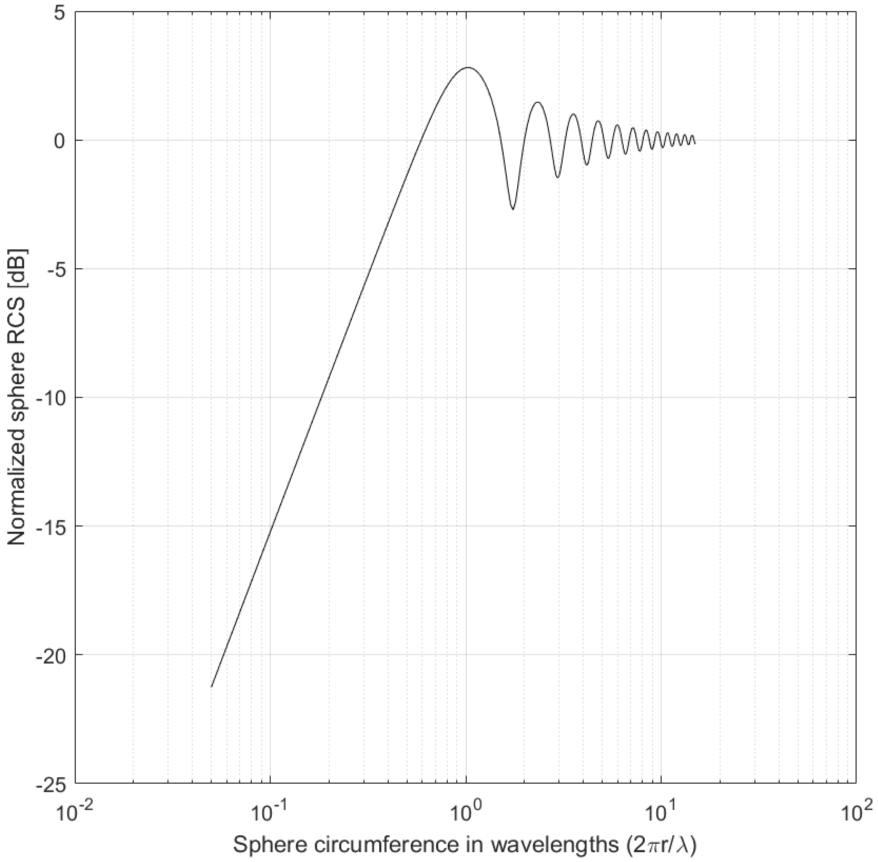


Resulting RCS for 10 spheres with 0dBsm distributed randomly

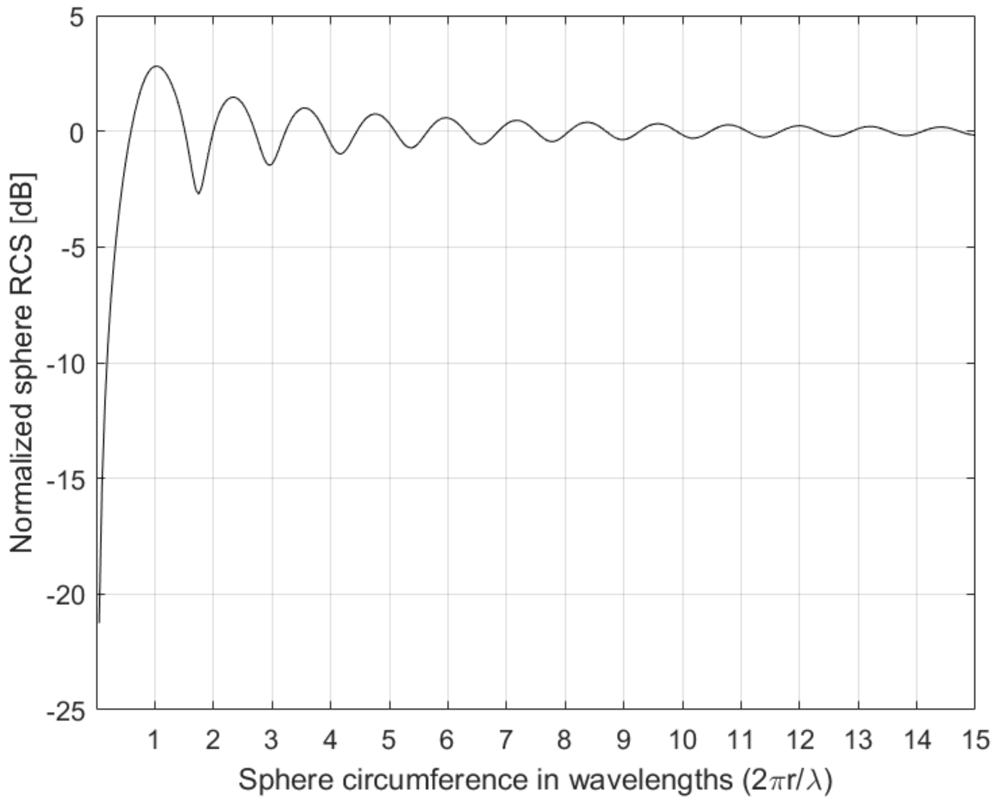



Resulting RCS for 10 spheres with 0dBsm (except one 17dB greater) distributed randomly
RCS (dBsm)
Phi angle (rad)

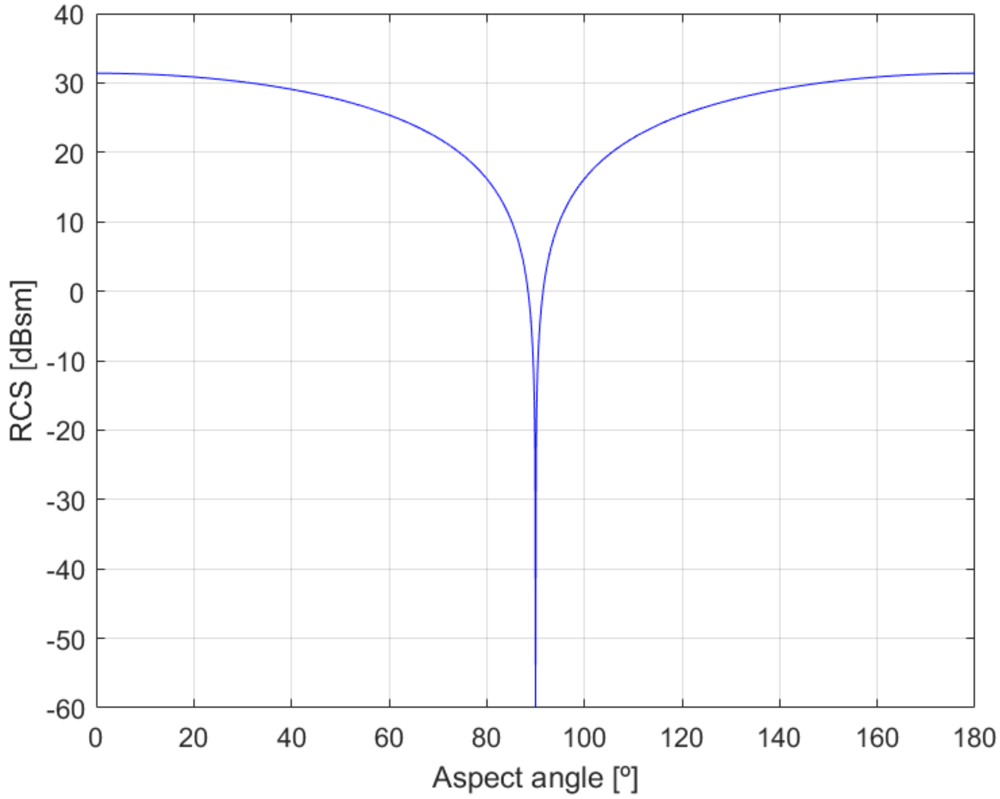

Resulting RCS for two spheres whose scatters are separated 1.00 m
Resulting RCS for two spheres whose scatters are separated 0.10 m
Resulting RCS for two spheres whose scatters are separated 0.01 m
RCS (dBsm)
Phi angle (rad)

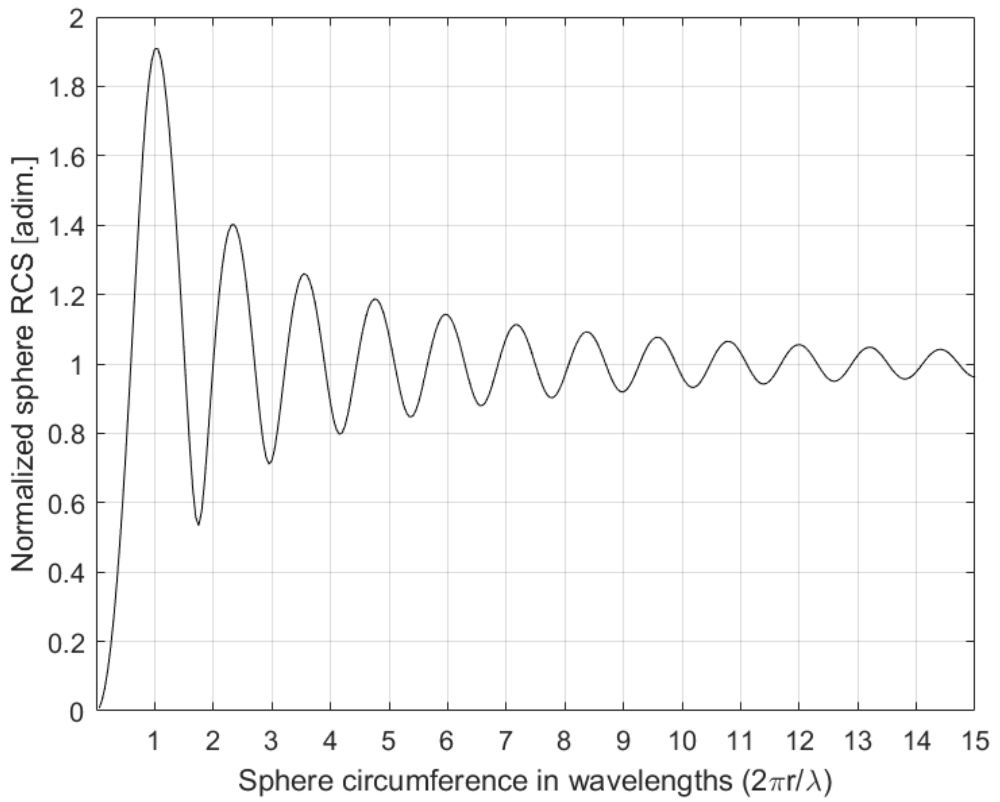


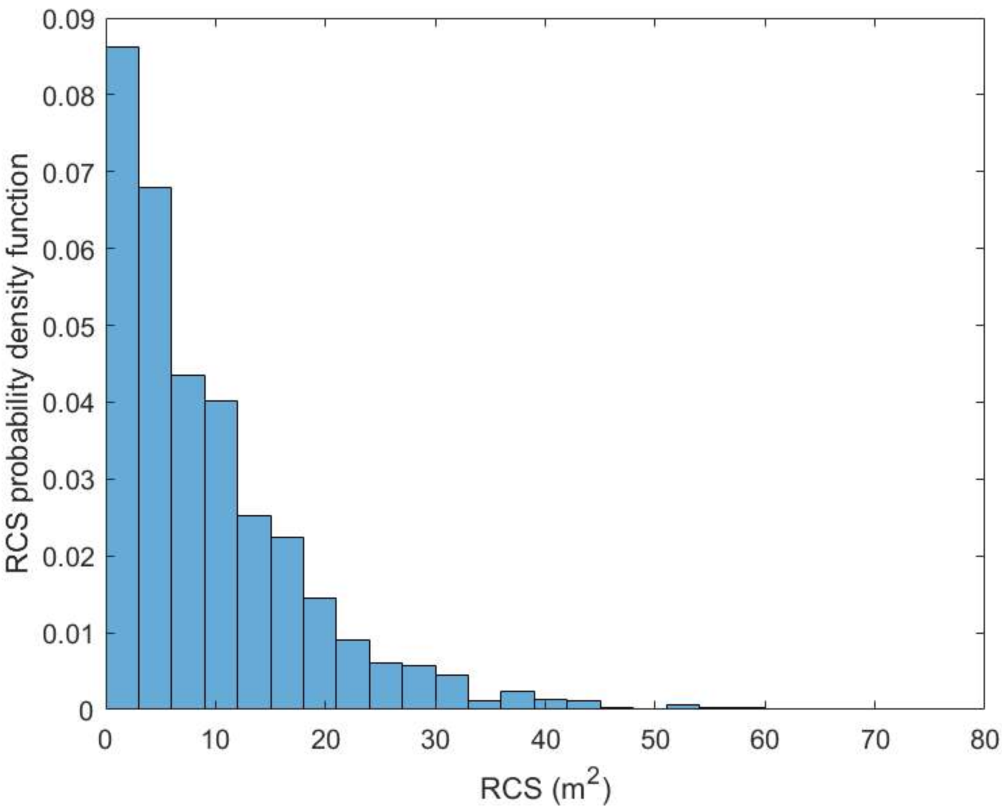
Normalized sphere RCS [dB]
5
0
-5
-10
-15
-20
-25
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
Sphere circumference in wavelengths ($2\pi r/\lambda$)

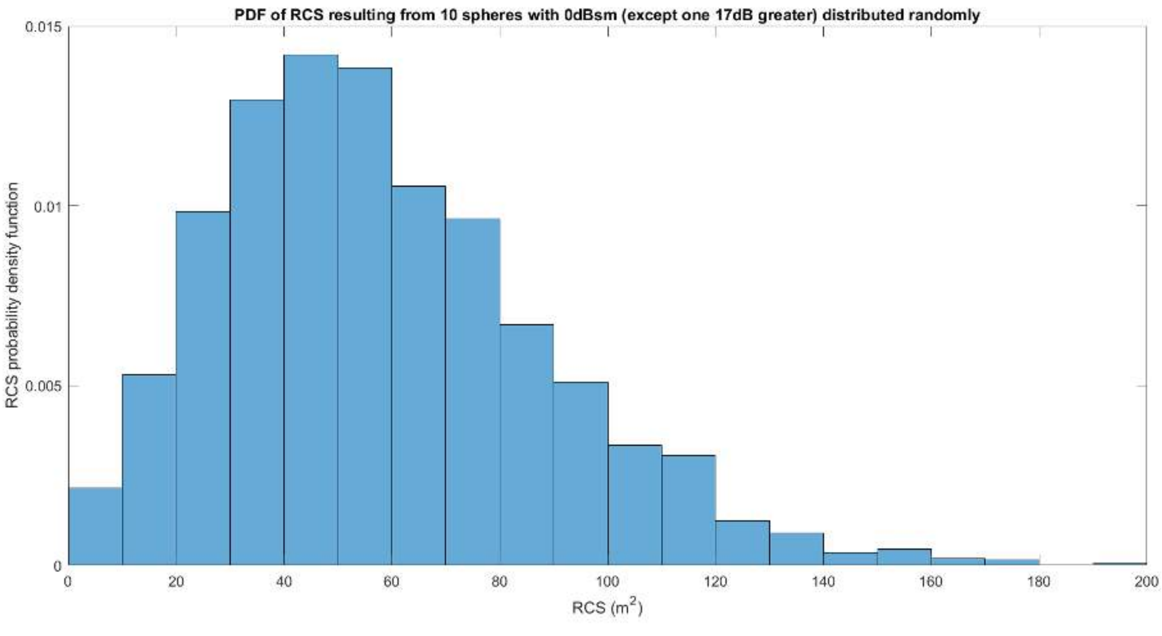


Normalized sphere RCS [dB]
5
0
-5
-10
-15
-20
-25
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
Sphere circumference in wavelengths ($2\pi r/\lambda$)


RCS [dBsm]
40
30
20
10
0
-10
-20
-30
-40
-50
-60
0
20
40
60
80
100
120
140
160
180
Aspect angle [º]


Normalized sphere RCS [adim.]
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
Sphere circumference in wavelengths (2πr/λ)


RCS probability density function
RCS (m2)


PDF of RCS resulting from 10 spheres with 0dBsm (except one 17dB greater) distributed randomly
RCS probability density function
RCS (m2)