

```

%MUSIC Algorithm for DOA
clc;
clear;

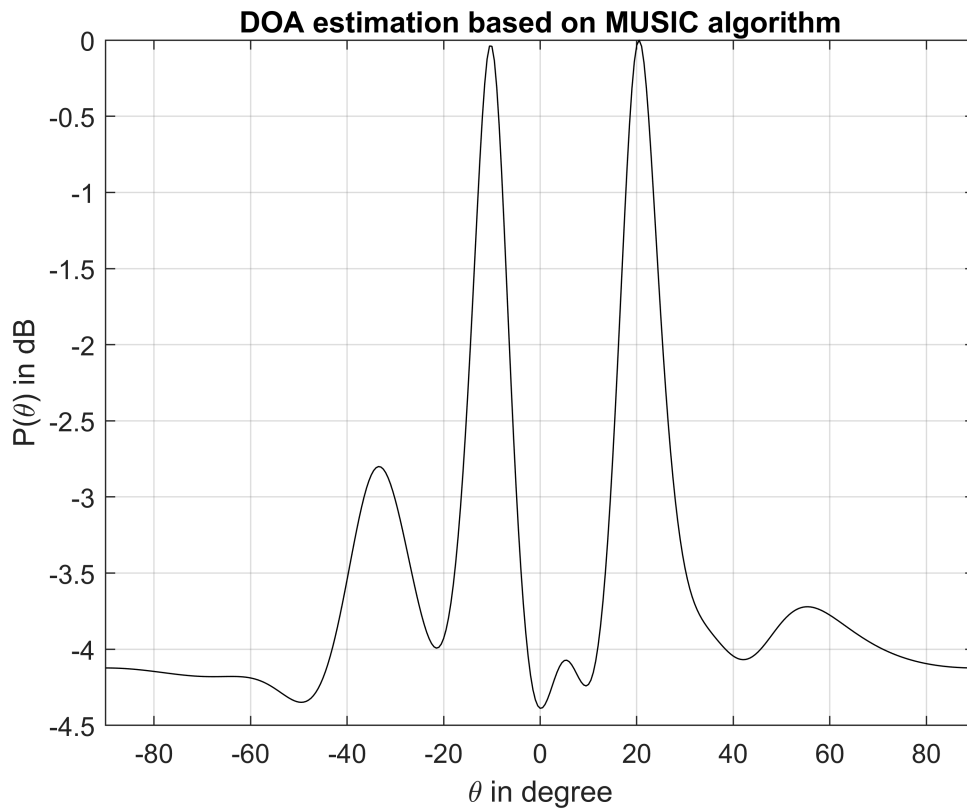
azimuth = [-10 20]/180*pi;
doa = azimuth;

N = 4500;
f = 2*10^9;
snr=5;
w = 2*pi*f*[1 1]'; %Angular frequency
M = 10; %Number of array elements
P = length(w); %Number of signal
lambda = 150/1000; %Wavelength
d = lambda/2; %Element spacing
D = zeros(P,M); %Creating a zero matrix with P rows and M columns

for k=1:P
    D(k,:) = exp(-1i*2*pi*d*sin(doa(k))/lambda*(0:M-1));
end
D=D';
%Generating Signals and Noise
Xs = 2*exp(1i*(w*(1:N))); %Generating the signal
X = D*Xs;
X = awgn(X,snr); %Insert gaussian White Noise
R = X*X';
[N,V] = eig(R); %Find Eigenvalues and Eigenvectors of R
NN = N(:,1:M-P); %Estimate Noise subspace

%Theta search for peak finding
theta = -90:0.5:90; %peak search
Pmusic = zeros(length(theta),1); %P music function
for ii=1:length(theta)
    SS = zeros(1,length(M));
    for jj=0:M-1
        SS(1+jj) = exp(-1i*2*jj*pi*d*sin(theta(ii))/180*pi)/lambda);
    end
    PP = SS*(NN*NN')*SS';
    Pmusic(ii) = abs(1/PP);
end
%%Plotting the results of theta and Pmusic function
figure;
Pmusic = 10*log10(Pmusic/max(Pmusic));
plot(theta,Pmusic,'-k');
xlabel('\theta in degree');
ylabel('P(\theta) in dB');
title('DOA estimation based on MUSIC algorithm');
xlim([-90 90]);
grid on;

```



```
%Modification in MUSIC Algorithm for Coherent Sources
J = fliplr(eye(M)); %anti Matrix
R = R+J*conj(R)*J; %Modified R matrix
[N,V] = eig(R); %Find Eigenvalues and Eigenvectors of R
NN = N(:,1:M-P); %Estimate Noise subspace

%Theta search for peak finding
theta = -90:0.5:90; %peak search
Pmusic = zeros(length(theta),1); %P music function
for ii=1:length(theta)
    SS = zeros(1,length(M));
    for jj=0:M-1
        SS(1+jj) = exp(-1i*2*jj*pi*d*sin(theta(ii))/180*pi)/lambda);
    end
    PP = SS*(NN*NN')*SS';
    Pmusic(ii) = abs(1/PP);
end
%%Plotting the results of theta and Pmusic function
figure;
Pmusic = 10*log10(Pmusic/max(Pmusic));
plot(theta,Pmusic,'-k');
xlabel('\theta in degree');
ylabel('P(\theta) in dB');
title('DOA estimation based on MUSIC algorithm');
xlim([-90 90]);
```

```
grid on;
```

