ADVANCED DIGITAL SIGNAL PROCESSING LAB

Experiment – 4

Problem Statement:

To perform power spectrum estimation using the MUSIC algorithm.

Name & Roll No. : Arkajit Pal 22SP06008

Research Paper :

P. Janik and Z. Waclawek, "MUSIC algorithm for estimation of parameters of signals in power system," 2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC), Rome, Italy, 2015, pp. 2236-2240, DOI :10.1109/EEEIC.2015.7165530

Motivation Behind Study :

- Spectrum analysis, also referred to as frequency domain analysis or spectral density estimation, is the technical process of decomposing a complex signal into simpler parts.
- As described above, many physical processes are best described as a sum of many individual frequency components.
- Any process that quantifies the various amounts (e.g., amplitudes, powers, intensities) versus frequency (or phase) can be called spectrum analysis.
- The goal of spectral density estimation or simply spectral estimation is to estimate the spectral density (also known as the power spectral density) of a signal from a sequence of time samples of the signal.
- Intuitively speaking, the spectral density characterizes the frequency content of the signal. One purpose of estimating the spectral density is to detect any periodicities in the data, by observing peaks at the frequencies corresponding to these periodicities.

Summary of Methods:

A. In order to find the power spectrum of a signal using the MUSIC method, we first generate a simple multi-tone sinusoidal signal and then add noise to it.
B. Then we find the auto-correlation matrix for the generated signal using the equations given in the next section.

C. The obtained square auto-correlation matrix is decomposed into its Eigenvalues and eigenvectors.
D. Based on the number of sinusoids present in the input signal we select the signal and noise subspace Eigenvectors.
E. Using the noise subspace vectors we find the pseudo-spectrum using the equation given in the next section.

Required Equation:

To find the auto-correlation matrix of the given signal we use the following equation –

$$\hat{R}x = \frac{1}{N} X^H X$$

It is assumed that the given signal has length L = N+M-I. The auto-correlation matrix Rx has size MxM. The matrix X is called the sample correlation matrix and is given by

$$= \begin{bmatrix} x(0) & x(1) & \cdots & x(M-1) \\ x(1) & x(2) & \ldots & x(M) \\ \vdots & \vdots & \ddots & \vdots \\ x(N-1) & x(N) & \ldots & x(N+M-2) \end{bmatrix}$$

The superscript ( ) ^{H} denotes the Hermitian operator.

To calculate the pseudo-spectrum the following equation is used -

$$P_{music}(e^{j\omega}) = \frac{1}{\sum_{i=p+1}^{M} |e^H s_i|^2}$$

$$e = [1 \quad e\wedge j\omega \quad \cdots \quad e\wedge j(M-1)\omega\,]\,T$$

Here, p is the number of frequency components in the signal if the signal is composed of complex sinusoids. p is twice the number of frequency components in the signal if the signal is composed of real sinusoids.

e is the steering vector and si is the ith Eigenvector. Since it goes from p+1 to M, it is always a noise subspace Eigenvector.

Result : x = (0.8*cos(0.5*pi*t) + 5*cos(0.2*pi*t))

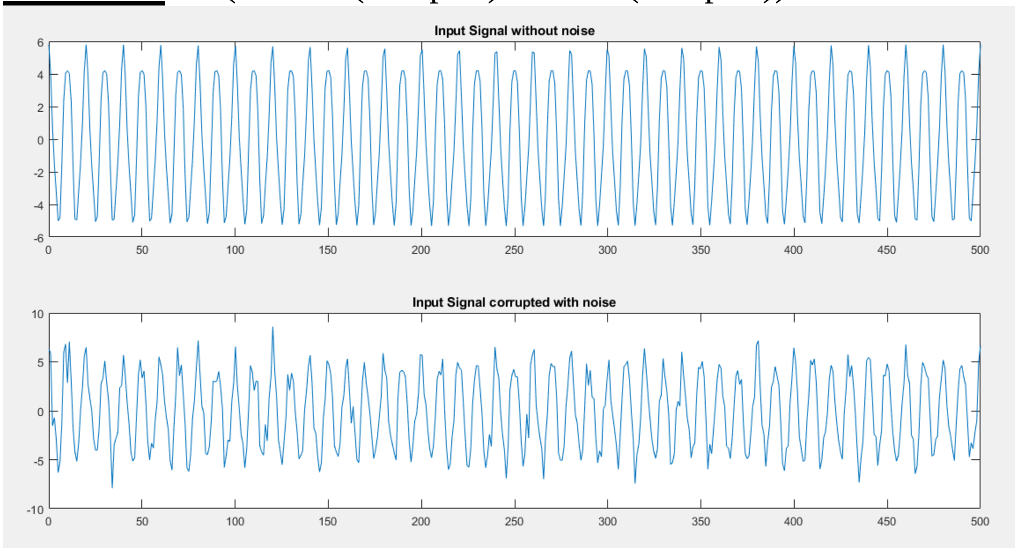

Fig 1 Input signal with and without noise

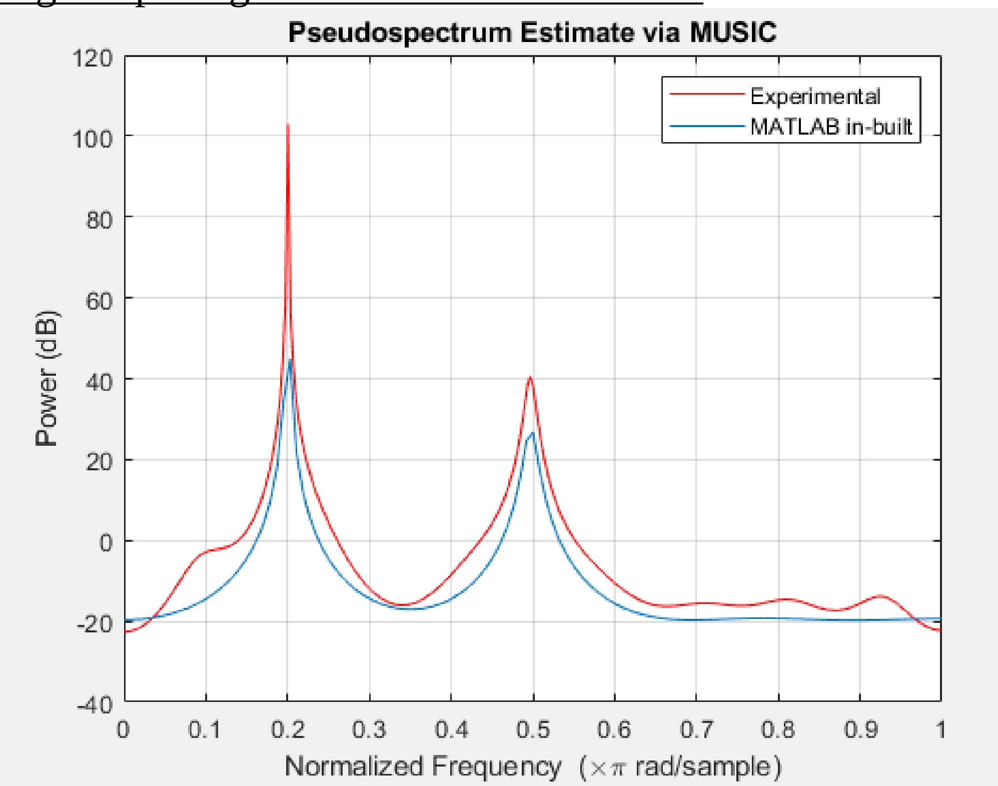

Fig 2 output PSE vs frequency

Conclusion:

x = (0.8*cos(0.5*pi*t) + 5*cos(0.2*pi*t))
here we have implemented the above x value as signal and we got the power spectral analysis using MUSIC properly.

Appendix:

```
clear all
close all
clc

L = 500;
r = 500;
n = ceil(0.98 * L);
m = L-n;

t = linspace(0,r,L);

x = (0.8*cos(0.5*pi*t) + 5*cos(0.2*pi*t));
figure; subplot(2,1,1); plot(t,x); title("Input Signal without noise");

p = 2;
p_ = 2*p;

% add noise
noise_var = 1;
for i = 1:L
    x_n(i) = x(i) + sqrt(noise_var)*randn;
end

subplot(2,1,2); plot(t,x_n); title("Input Signal corrupted with noise");

% find r_x autocorrelation matrix using method given in paper

X = zeros(n,m); % sample correlation matrix

for i=1:n
    for j=1:m
        X(i,j) = x_n(i-1+j-1 + 1);
    end
end

% X = hankel(x_n(1:m),x(m+1:end));

r_x = X' * X;

[V,D] = eig(r_x);

% V_exp = V;
% D_exp = D;

% sort eigen vectors and eigen values in descending order

for i = 1:floor(size(D,1)/2)
    temp = D(i,i);
    D(i,i) = D(size(D,1)-i+1,size(D,1)-i+1);
```

```
    D(size(D,1)-i+1,size(D,1)-i+1) = temp;
end

for i = 1:floor(size(V,2)/2)
    temp = V(:,i);
    V(:,i) = V(:,size(V,2)-i+1);
    V(:,size(V,2)-i+1) = temp;
end

omega = 0:0.01:pi;

e = zeros(m,size(omega,1)); % a single column of e corresponds to the steering
vector of a given frequency

for c=1:size(omega,2)
    for r_=1:m
        e(r_,c) = real(exp(1i*omega(c)*(r_-1)));
    end
end

d_squared = zeros(1,size(omega,2));

for c=1:size(omega,2)
    steering_vector = e(:,c);
    sum = 0;
    for k=p_+1:m
        % V_ = fftshift(fft(V(:,k)));
        sum = sum + ( abs( steering_vector' * V(:,k) ) ) ^ 2;
    end
    d_squared(c) = sum;
end

pseudo_spect = 1 ./ d_squared;

pseudo_spect_db = 10*log(abs(pseudo_spect));

% figure; subplot(1,2,1); plot(omega/pi,abs(d_squared)); title("d^2");
subplot(1,2,2); plot(omega/pi,abs(pseudo_spect)); title("Pseudo Spectrum");

figure; plot(omega/pi,pseudo_spect_db,'r'); title("Pseudo Spetrum in dB"); hold
on;

[~,R_X] = corrmtx(x_n,m-1,'mod');
pmusic(R_X,p_);
legend("Experimental","MATLAB in-built");
```