
Solution to Problem Set # 2.0
ECE-595, Fall 2006
Spatial Array Processing
University of New Mexico, Albuquerque

Date Assigned : 09/06/2006
Date Due : 09/13/2006

```
%  Author: Balu Santhanam
%  DATE  : 04/25/02
%  Matlab program generates synthetic data that simulates
%  plane waves propagating in free-space
%**************************************************************
% Prompt User to input some array parameters
 L = input('Number of sensors in array --->');
 M = input('Number of source signals --->');
 N = input('Number of snaphots needed --->');
 d = input('Sensor spacing in half wavelength units --->');
 nv = input('Variance of noise sources --->');
 sv = input('Vector of variances of signal sources  --->');
 ang = input('vector of DOA of the sources --->');
%
% Generate simulated data
sd = clock; ang = ang(:)*(pi/180);
randn('seed',sd(6)*1000000)
%
%  Parameters of Noise Sources to generate
%  a noise matrix of size N x L one column for
%  each sensor
 mu_v = zeros(L,1); sigv = sqrt(nv)*eye(L,L);
 n = ((1+j)/sqrt(2))*randn(N,L)*sigv;
%
%  Parameters of signal amplitude to
%  Generate N x M amplitude matrix one
%  for each source
A = ((1+j)/sqrt(2))*randn(N,M)*diag(sqrt(sv),0);
%
%  Generate directional matrix S (M x L)
%  and the observation matrix (N x L)
S = zeros(M,L); for k = 1:1:L
    S(:,k) = exp(-j*pi*sin(ang)*(k-1)*d);
end
sig = (A*S + n).';
```

```
%
%   Author: Balu Santhanam
%   Date: 04/25/02
%   This function computes the DOA estimates of incoming
%   plane waves impinging on a linear array of sensors that
%   are separated by d units arranged on the xaxis
%   We are assuming that the number of such sources are known.
%   SYNOPSIS
%    P = doa(sig,L,N,M,d)
%    L : number of sensors
%    N : number of temporal snapshots
%    d : sensor spacing in half wavelengths
%*****************************************************************
function P = doa(x,L,N,d)
%  Error checking
if nargin < 5
   error('Insufficient Info')
elseif isnumeric([x(:);L;N;M;d])~=1
   error('Non-numeric input')
elseif mod(N,1)~= 0 | mod(M,1)~= 0 | mod(L,1)~=0
   error('N,L,M have to be integers')
end
%
% Generate spatial correlation matrix of observations
% by averaging over time snapshots
% Matrix size is L x L
%
R_est = x*x'/N;
%
% Generate matrix of steering vectors
% Matrix size is L x range of angles
% Angles are measured with respect to
% normal at surface of incidence.
%
steer = zeros(L,360);
for m = 0:1:L-1
    steer(m+1,:) = exp(j*pi*(m)*d*sin(([0:1:359] - 180)*pi/180));
end
% Generate estimate of average power for different
% directions
for i = 1:360
    P(i) = abs(steer(:,i)'*R_est*steer(:,i));
end
```

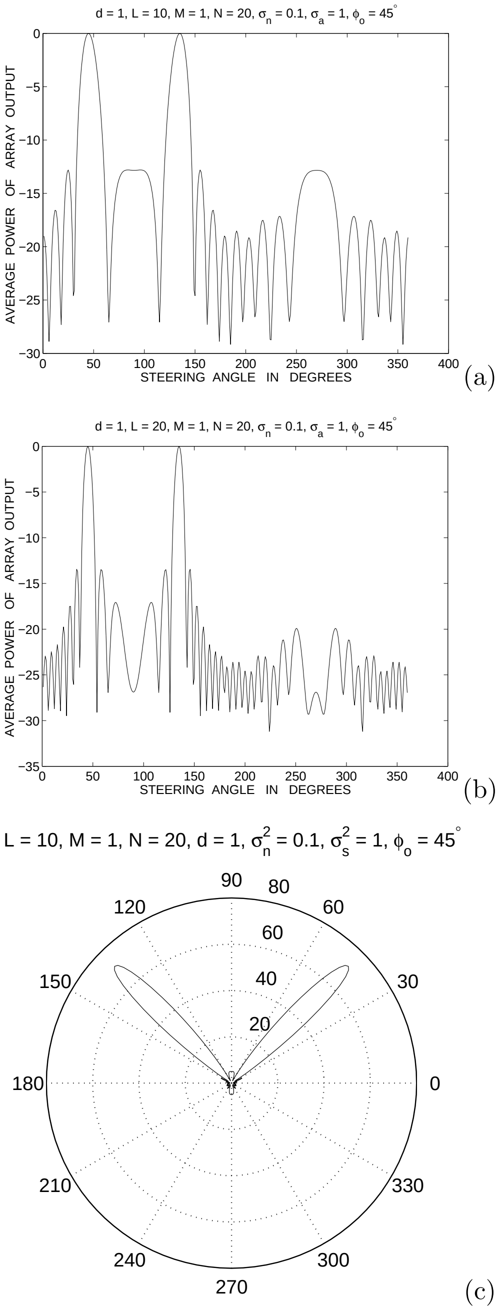

Figure 1: Effect of spectral resolution on beamformer performance for a single source. Increasing the number of sensors in the ULA decreases the width of the mainlobe and improves spectral resolution.

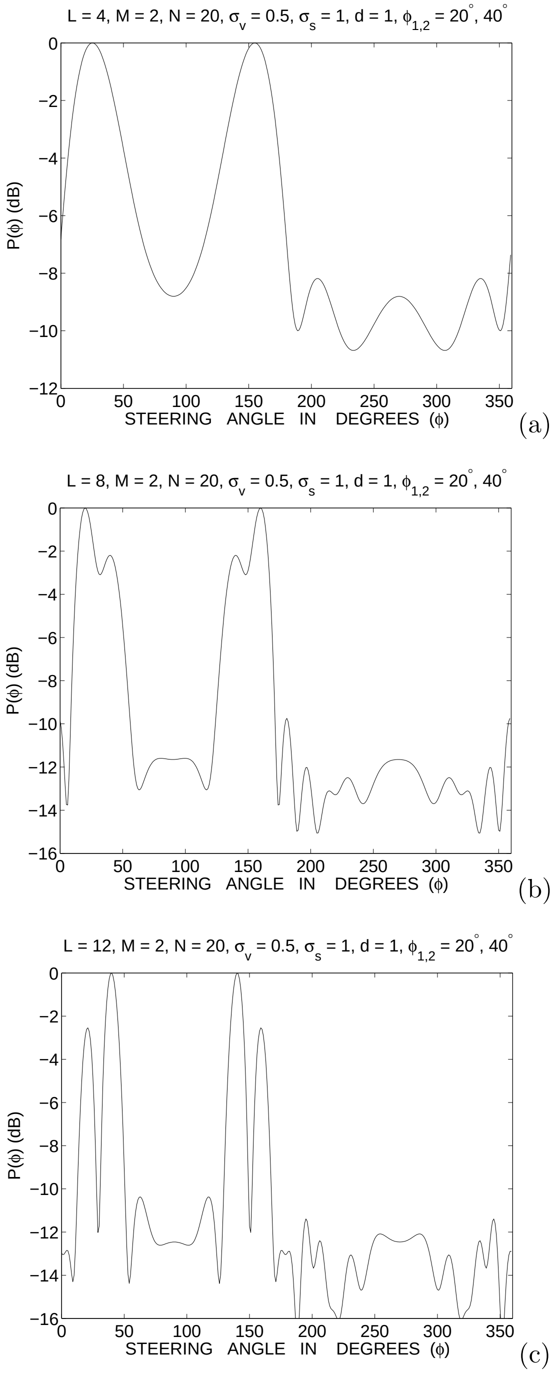

Figure 2: Effect of spectral resolution on beamformer performance for two sources. Note here that the peaks are not resolved when the resolution of the beamformer is not sufficient.

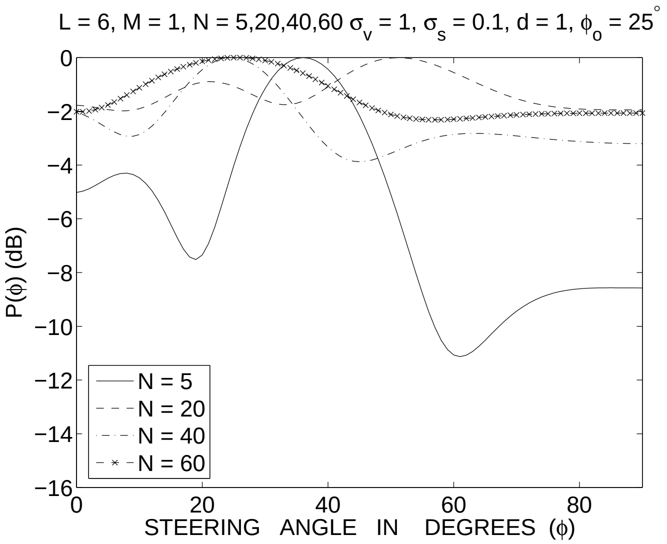

Figure 3: Effect of temporal averaging on the beamformer performance for different number of snap-shots. Note that the peak at $\phi = 25^\circ$ is not detected until $N = 40$ for a SNR of -10 dB. Lower SNR's require more averaging to be able to resolve the peak.

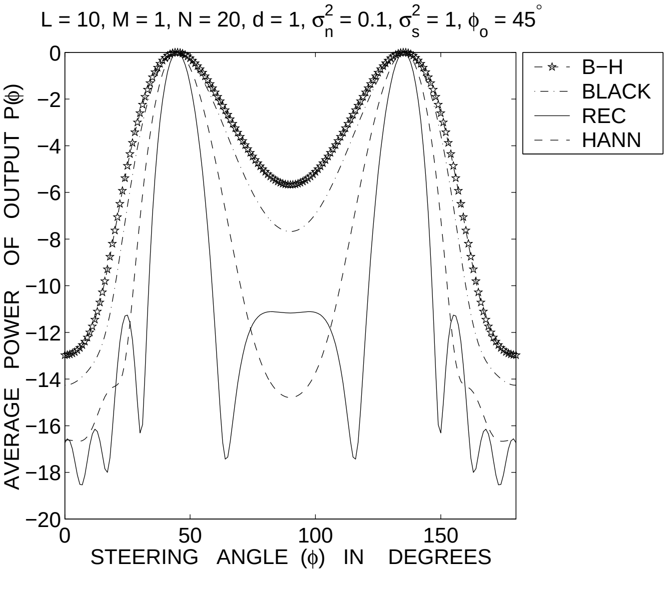

Figure 4: Effect of apodization and sidelobe suppression on beamformer performance for four different windows.