.

EC431H1 Digital Signal Processing
LAB 4.
Linear Array Processing: Beaforming, Bearing estimation and Spatial filtering.

Introduction

The purpose of this experiment is to become familiar with some aspects of modeling and operation of linear antenna arrays. It is recommended that you use MATLAB to program and execute the required steps of the experiment. Your lab report should include: a)printouts of your programs, b) printouts/plots of the obtained results, and c) answers to questions and brief but critical discussion of the obtained results.

Experiments

PART A. Linear array modeling and Beam-forming

A uniform linear array is a set of M sensors in a line configuration with spacing d between sensors (see Fig. 1). Thus, the total length of the array is $(M-1)d$. A SNAPSOT is a set of M sensor readings (outputs) at a given time instant. In array processing many SNAPSHOTS taken at different time instants are combined appropriately to produce a desired result.

Now consider a linear array of M equispaced sensors with intersensor spacing d. Taking as reference the sensor 1 and by denoting as $w_m[n]$ the excitation or weight signal at sensor m at time instant n, then the far field radiation pattern (or equivalently, the response of the array to an incident signal) at an angle θ is given by

$$A(\theta, n) = \sum_{m=1}^{M} w_m[n] \cdot e^{\frac{j2\pi(m-1)d\sin(\theta)}{\lambda}}$$

where λ is the wavelength of the excitation. Usually, we want to estimate $A(\theta)$ for $-90^o \leq \theta \leq 90^o$. Notice that this, at time instant n , is nothing else but the Fourier transform of the SNAPSOT signal at time instant n at frequencies $\frac{d\sin(\theta)}{\lambda}$.

1. Assume for the moment that $M = 12$ and all array sensors are equally weighted, that is,

   $$w_m[n] = 1, \qquad m = 1, \ldots, M, \qquad \forall n$$

   Calculate the Power radiation (or response) pattern of the array $P(\theta) = |A(\theta)|^2$ for $d = \frac{\lambda}{2}$ and $-90^o \leq \theta \leq 90^o$ in increaments of $\Delta\theta = 1^0$. Can you explain how to use FFT algorithms to calculate $A(\theta)$? Draw $P(\theta)$ vs θ.

2. Consider now to use unequal weighting at different sensors for example by applying a triangular distribution of weights instead of uniform (in similarity to Bartlet vs rectangular window. Calculate and draw $P(\theta)$ vs θ making the same assumptions as in item 1.

3. Consider now 'steering' the power pattern of the array to an angle $\phi \neq 0$. To achieve this choose $\phi = 30^o$, for example, and generate the weight vector

   $$w_m[n] = e^{\frac{j2\pi(m-1)d\sin(\phi)}{\lambda}}, \qquad m = 1, \ldots, M, \qquad \forall n$$

   Calculate and draw $P(\theta)$ vs θ making the same assumptions as in item 1.

The above procedure that shapes the power pattern of the array is called BEAMFORMING.

PART B. Discrete-Time Butterworth Filter Design Using the Bilinear Transformation

This problem explores the design of discrete-time filters from continuous-time filters using the bilinear transformation. The bilinear transformation is a mapping that can be used to obtain a rational *z*-transform $H_d(z)$ from a rational Laplace transform $H_c(s)$. This mapping has some important properties, including the following:

1. If $H_c(s)$ is the system function for a causal and stable continuous-time system, then $H_d(z)$ is the system function for a causal and stable discrete-time system.
2. If $H_c(j\Omega)$ is a piecewise constant frequency response, then $H_d(e^{j\omega})$ will also be piecewise constant.

These two properties in combination imply that causal and stable ideal frequency-selective filters will map to causal and stable ideal frequency-selective filters $H_d(z)$. The bilinear transformation of $H_c(s)$ is given by

$$H_d(z) = H_c(s) \qquad \textit{evaluated with} \qquad s = \frac{2}{T_d}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) \tag{1}$$

For the frequency response, the mapping is given by

$$H_d(e^{j\omega}) = H_c(j\Omega)\Big|_{\Omega=\frac{2}{T_d}\tan(\omega/2)} \tag{2}$$

Basic Problems

In this set of problems, you will analytically design a discrete-time lowpass Butterworth filter using the bilinear transformation. The discrete-time filter must meet the following specifications:

- Passband frequency $\omega_p = 0.22\pi$,
- Stopband frequency $\omega_s = 0.4\pi$,
- Passband tolerance $\delta_1 = 0.3$,
- Stopband tolerance $\delta_2 = 0.25$.

This implies that

$$1-\delta_1 \le \left|H_d(e^{j\omega})\right| \le 1, \qquad 0 \le |\omega| \le \omega_p,$$
$$\left|H_d(e^{j\omega})\right| \le \delta_2 \qquad \omega_s \le |\omega| \le \pi.$$

An Nth-order continuous-time Butterworth lowpass filter has a frequency response whose magnitude satisfies

$$|H_c(j\Omega)|^2 = \frac{1}{1+\left(j\Omega / j\Omega_c \right)^{2N}} \qquad (3)$$

For a Butterworth filter with a real-valued impulse response, $b(t)$, the system function satisfies

$$H_c(s)H_c(-s) = \frac{1}{1+\left(s / j\Omega_c \right)^{2N}} \qquad (4)$$

The design method that you will use is

- Map the specifications for the discrete-time filter into specifications for a continuous-time filter using the bilinear transformation.
- Design a continuous-time Butterworth filter to meet or exceed these specifications.
- Map the continuous-time filter back to discrete-time using the bilinear transformation.

The following problems guide you through this design procedure.

a. Map the corner frequencies ω_p and ω_s to continuous-time corner frequencies Ω_p and Ω_s using the following equation

$$\Omega = \frac{2}{T_d}\tan(\omega/2)$$

choose $T_d = 1$, for convenience.

b. Using Equation 3, solve for the integer filter order N and the cutoff frequency Ω_c that will meet this specification. You would like to find the lowest order N that can meet these specifications, since lower order filters are easier and cheaper to implement. Also determine $H_c(s)$, the system function for the filter.

c. Make a pole-zero plot for the continuous-time filter. Also plot the frequency response magnitude of the filter. Be sure to check the passband and stopband edges to verify that your filter meets or exceeds the continuous-time specifications.

d. Map the continuous-time system function $H_c(s)$ to a discrete-time system function $H_d(z)$ using the bilinear transformation given in Equation 1 and store the resulting numerator and denominator polynomial coefficients. Plot the frequency response magnitude and verify that your filter meets the original discrete-time specifications.

e. Make a pole-zero plot for the discrete-time filter and verify that the causal and stable continuous-time filter maps to a causal and stable discrete-time filter.