EE5027 Adaptive Signal Processing
Homework Assignment #2

Notice

- Due at 9:00pm, November 17, 2020 (Tuesday) = T_d for the electronic copy of your solution.
- Please submit your solution to NTU COOL (`https://cool.ntu.edu.tw/courses/3062`)
- ***All answers have to be fully justified.***
- All the figures should include labels for the horizontal and vertical axes, a title for a short description, and grid lines. Add legends and different line styles if there are multiple curves in one plot.
- No extensions, unless granted by the instructor one day before T_d.

Problems

1. (Linear prediction, 35 points) Consider a wide-sense stationary random process $x(n)$ whose autocorrelation function satisfies

$$r(0) = 0.1482, \qquad r(1) = 0.0500, \qquad r(2) = 0.0170, \qquad r(3) = -0.0323. \tag{1}$$

   Find the following quantities *by hand*

   (a) (6 points) The reflection coefficients κ_1 κ_2, and κ_3.

   (b) (6 points) The quantities Δ_0, Δ_1, and Δ_2.

   (c) (8 points) The tap-weight vector of the forward prediction-error filter $\mathbf{a}_0$, $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$.

   (d) (8 points) The minimum mean-square prediction error P_0, P_1, P_2, and P_3.

   (e) (7 points) Draw a signal-flow graph for the lattice model of the prediction errror filters of order $M = 3$. The input is $x(n)$ and the outputs are $f_M(n)$ and $b_M(n)$.

2. (Prediction error bound and spectral flatness, 15 points) In the lecture, we studied a lower bound for the the minimum forward prediction error power for a one-step forward linear prediction with order m, denoted by P_m, as follows:

$$P_m \geq \exp\left(\int_{-1/2}^{1/2} \log S_x(e^{j2\pi f})\, \mathrm{d}f\right). \tag{2}$$

The right-hand side of (2) is the prediction error bound. We also define the spectral flatness measure γ_x^2 as

$$\gamma_x^2 \triangleq \frac{\exp\left(\int_{-1/2}^{1/2} \log S_x(e^{j2\pi f})\,\mathrm{d}f\right)}{\int_{-1/2}^{1/2} S_x(e^{j2\pi f})\,\mathrm{d}f}. \tag{3}$$

(a) (10 points) Consider an AR process with the following relation:

$$x(n) = \alpha x(n-1) + v(n), \tag{4}$$

where $\alpha \in \mathbb{C}$, and $|\alpha| < 1$. The wide-sense stationary complex random process $v(n)$ has zero mean and the power spectral density $S_v(e^{j2\pi f}) = 1$. Find the prediction lower bound for $x(n)$.

(b) (5 points) Find the spectral flatness measure γ_x^2 for $x(n)$ in (4).

3. (LMS Algorithms, 18 points) Consider the input signal $x(n)$ and the desired signal $d(n)$ to be

$$x(n) = e^{j2\pi f_1 n}, \qquad d(n) = e^{j2\pi f_2 n}. \tag{5}$$

We assume that the number of taps M is 1. The initial weight vector of the LMS adaptive filter is $\widehat{\mathbf{w}}(0) = 0$. The step-size parameter satisfies $0 < \mu < 1$. Calculate the following quantities *by hand*

(a) (2 points) The error signal $e(0)$.

(b) (4 points) The weight vector $\widehat{\mathbf{w}}(1)$.

(c) (4 points) The weight vector $\widehat{\mathbf{w}}(2)$.

(d) (8 points) The weight vector $\widehat{\mathbf{w}}(n)$ for $n \geq 3$.

4. (The least-perturbation property, 12 points) The update equation for the ϵ-NLMS algorithm is given by

$$\widehat{\mathbf{w}}_{\epsilon\text{-NLMS}}(n+1) = \widehat{\mathbf{w}}_{\epsilon\text{-NLMS}}(n) + \frac{\widetilde{\mu}}{\epsilon + \|\mathbf{x}(n)\|_2^2}\mathbf{x}(n)e^*(n), \tag{6}$$

where $0 < \widetilde{\mu} < 1$ and $\epsilon > 0$.

(a) (5 points) Write down the least-perturbation property for the ϵ-NLMS algorithm.

(b) (7 points) Show that the solution to the optimization problem in Problem 4a is (6). *Hint:* Use the method of Lagrange multipliers and matrix derivatives.

5. (Prediction error filters, 20 points) We consider a system diagram in Figure 1. The random process $v(n)$ is a zero-mean, circularly-symmetric complex Gaussian, white, wide-sense stationary random process with unit variance ($\sigma_v^2 = 1$). The transfer function $H(z)$ is given by

$$H(z) = \frac{1 + z^{-1}}{\left(1 - \frac{1}{2}z^{-1}\right)\left(1 + \frac{1}{3}z^{-1}\right)}, \tag{7}$$

with the region of convergence $|z| > \frac{1}{2}$.

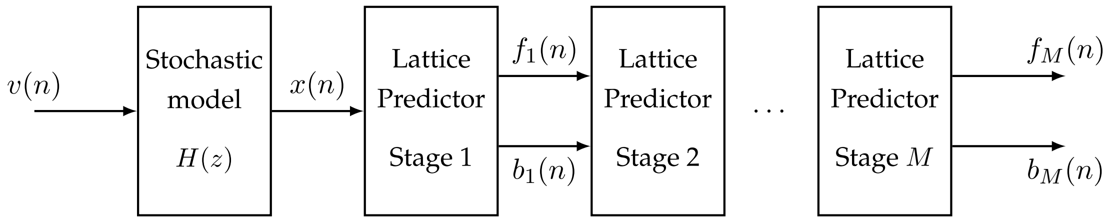

Figure 1: A system diagram for Problem 5.

(a) (4 points) First we implement the Levinson-Durbin algorithm using MATLAB. The autocorrelation function is specified by a column vector r $= [r_x(0), r_x(1), \ldots, r_x(M)]^T$. Write the following MATLAB function for the Levinson-Durbin algorithm:

$$\texttt{[a, P, kappa] = ASP\_Levinson\_Durbin(r);} \tag{8}$$

The output arguments are specified as follows.

- a is a MATLAB cell array of size M. The entries in a contain the coefficients of the forward prediction error filter. More specifically, we have a{1} $= \mathbf{a}_1$, a{2} $= \mathbf{a}_2$, and a{M} $= \mathbf{a}_M$.
- P is an $(M+1)$-by-1 vector for the prediction errors. We have P $= [P_0, P_1, P_2, \ldots, P_M]^T$.
- kappa is an M-by-1 vector for the reflection coefficients. We have kappa $= [\kappa_1, \kappa_2, \ldots, \kappa_M]^T$.

Note and hints:

- This function returns error messages if r does not correspond to a valid autocorrelation.
- First derive an expression for the autocorrelation function of $x(n)$ from (7).
- Test your program using the autocorrelation function of $x(n)$.

(b) Next we move on to the the simulation of these random processes. You can read the file ASP_Problem_5.mat for the sequence $v(n)$ for $n = 0, 1, \ldots, L-1$. The vector v is defined as v $\triangleq \begin{bmatrix} v(0) & v(1) & \ldots & v(L-1) \end{bmatrix}$, where $L = 1000$. Then we compute the associated $x(n)$ and $f_1(n)$. Plot the real parts and the imaginary parts of $f_1(n)$ against the time index n. Note that the MATLAB function filter helps to find the output signal of a linear-time invariant system. Estimate the average power of $f_1(n)$ from these measurements by

$$\widehat{P}_{f,1} \triangleq \frac{1}{L} \sum_{n=0}^{L-1} |f_1(n)|^2, \tag{9}$$

where the subscript $f, 1$ denotes the association with $f_1(n)$.

(c) We consider the same realization of $v(n)$ in Problem 5b. Repeat Problem 5b for $b_1(n)$. The average power of $b_1(n)$ is estimated by

$$\widehat{P}_{b,1} = \frac{1}{L} \sum_{n=0}^{L-1} |b_1(n)|^2. \tag{10}$$

(d) Plot the prediction error power over the index $m = 1, 2, \ldots, 10$. This plot contains four curves:

i. One curve for the forward prediction error power $\widehat{P}_{f,m}$ (from the realizations of $f_m(n)$).

ii. Another curve for the backward prediction error power $\widehat{P}_{b,m}$ (from the realizations of $b_m(n)$).

iii. Another curve for the prediction error power P_m from the Levinson-Durbin algorithm.

iv. The other curve for the prediction error bound of $x(n)$. You may use numerical integration in MATLAB for the value of this bound.

Comment on your results with possible explanations.

Note: Please submit a compressed file including two MATLAB scripts with the following file names:

(a) `ASP_Levinson_Durbin.m`

(b) `ASP_HW2_Problem_5.m`

These MATLAB codes should generate your final results and plots directly, which have to be identical to those in your solution.

Last updated October 28, 2020.