Department of Electrical Engineering, IIT Madras
EE-4140: Digital Communication Systems

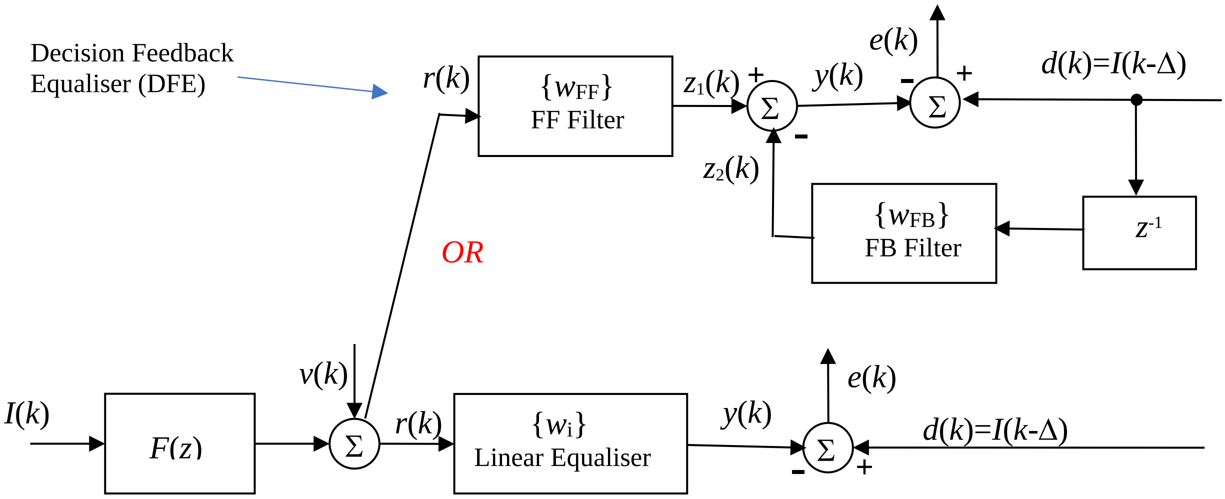
Due by Nov. 30, 2023

November 2023 | **Simulation Assignment #2** | **25 marks**

L-MMSE based Equalisation: Consider a distorting (ISI) channel defining the measurement model as:

$$r(k) = \sum_{l=0}^{L-1} f_l \; I(k-l) + v(k)$$

Here, the independent, uniformly distributed data symbols $I(k)$ entering the channel $F(z)$ are drawn from a 4-ary PAM alphabet, and the transmit signal power $E[|I(k)|^2] = \sigma_I^2 = 1$. The L-tap channel is specified for L=3 by the transfer function $F(z) = \frac{1}{\sqrt{2}}(0.8 - z^{-1} + 0.6z^{-2})$ and the additive white, Gaussian, noise component $v(k)$ is zero mean with variance σ_v^2. The signal to noise ratio (SNR) measured on $r(k)$, is then given by SNR = $1/\sigma_v^2$, and we vary the SNR on the measurements by varying σ_v^2 appropriately. In the figure below, two different equalisers based on the LMMSE formulation are described. We study the performance of LMSSE based Linear Equaliser in the first question.



(a) [1+1+1+3+3+3+3 = 15 marks] In the following, we study the performance of a linear equaliser (LE) for the above channel at SNR=10dB. Recall that to set up the N^{th} order Wiener-filter, we require the $N \times N$ auto-correlation matrix $\boldsymbol{R} = E[\boldsymbol{y}(k)\boldsymbol{y}^T(k)]$ and the $N \times 1$ cross-correlation vector $\boldsymbol{p} = E[d(k)\boldsymbol{y}(k)]$. We refer to the Weiner Filter (**WF**) determined using the true statistics as Statistical WF (**SWF**).

Now, assuming ergodicity, we also estimate $\boldsymbol{R}$ and $\boldsymbol{p}$ using time-domain averaging. For this, define: $\widehat{\boldsymbol{R}}(P) = \frac{1}{P}\sum_{k=1}^{P} \boldsymbol{y}(k)\boldsymbol{y}^T(k)$ and $\widehat{\boldsymbol{p}}(P) = \frac{1}{P}\sum_{k=1}^{P} d(k)\boldsymbol{y}(k)$ where the P snapshots can use N successive samples (i.e., where the current snapshot has an overlap of N-1 samples with the previous snapshot) and where $\boldsymbol{y}(k) = [y(k) \;\; y(k-1) \;\; y(k-2) \cdots y(k-N+1)]^T$. The WF determined using these time-domain averaged values is referred henceforth as Time-Averaged WF (**TAWF**).

For **SWF**:

(a1) The LE has order N=3, and the desired response $d(k)=I(k-\Delta)$, where the decoding delay Δ=0. Determine the SWF solution $\mathbf{w}_{opt}$. What is the corresponding J_{min}?

(a2) Repeat (a1) for N=10 and Δ=0.

(a3) Repeat (a1) for N=10 and Δ=5.

(a4) Can you find the "best choice of N and Δ" for the SWF defined for SNR=10dB by trial-and-error, to give the lowest possible J_{min}? What is this value of J_{min}?

(a5) For SNR in 2dB steps, from 0dB to 20dB, evaluate and plot $\log_{10}$SER versus $10\log_{10}$(SNR) for your best choice of N and Δ, determined for the SWF based LE in part (a4). Mark this as Fig 1. Measure SER over say 50,000 4ary-PAM symbols for each SNR step. Comment on your result.

For **TAWF**:

(a6) For N=10 and Δ=5, and for SNR=10dB, build the TAWF using time-domain averaging for the following number of snap-shots: (i) P = 20; (ii) P = 100; and, (iii) P = 500. Find the $J_{min}(P)$ in each case. Compare these results with the J_{min} of the SWF found in (a3) and comment.

(a7) Using the "best choice of N and Δ" determined for the SWF at SNR=10dB that you found in (a4), construct the TAWF with P=500 snapshots. After building the TAWF, send 50,000 more 4-ary PAM symbols for each SNR step and repeat (a5). Plot this SER curve also in Fig 1. Comment on your result.

(b) [1+3+4+2 = 10 marks] Now, consider a different FIR channel $G(z)$ with L=6 taps specified by the transfer function

$$G(z) = \frac{1}{C}(1 - 0.95z^{-1} + 0.5z^{-2} + 0.15z^{-3} - 0.2z^{-4} - 0.1z^{-5})$$

where the transmit symbols here are 2-ary PAM with $E[|I(k)|^2] = \sigma_I^2 = 1$. Normalise the channel gain to unity by appropriate choice of C in $G(z)$. Now, answer the following:

(b1) For N=20 and Δ=9, estimate the SWF Linear Equaliser. At SNR=10dB, what is the J_{min} for this LE?

(b2) We study the performance of a decision feedback equaliser (DFE) for the same channel at SNR=10dB. The feedforward filter of order N_1 produces the output $z_1(k)$ and from this, $z_2(k)$, the output of the feedback filter (of order N_2) is subtracted to produce the symbol estimate $y(k)$. The DFE has order N_1=15 and N_2=5, and the desired response $d(k)=I(k-\Delta)$, where the decoding delay Δ=1. Determine the 20×1 SWF solution for the DFE given by $\mathbf{w}_{opt} = [\mathbf{w}_{ff} \mid \mathbf{w}_{fb}]$. What is the corresponding J_{min} for the DFE? Compare with your result in (b1) and comment.

(b3) For SNR in 2dB steps from 0dB to 20dB, evaluate and plot $\log_{10}$SER versus $10\log_{10}$(SNR) for both: (i) The SWF-LE in (b1) and (ii) The SWF-DFE in (b2). Measure SER over 50,000 2ary-PAM symbols for each SNR step, and mark these plots as Fig 2.

(b4) As done in Assignment #1, but now over 0dB to 20dB, evaluate the SER of the 2^5 state Viterbi Algorithm (VA) by sending 50,004 symbols in each step of 2dB. Using decoding delay δ=30, compute the $\log_{10}$SER versus $10\log_{10}$(SNR) and plot these results of the VA also in Fig 2. Comment on your results in Fig 2.

Note: Code for this sub-part need *not* be re-submitted since it was already done in SA#1.

Instructions

Submit *both* a soft-copy and a hard-copy of your report. Latex / iPad / tablet / hand-written (and scanned) reports are fine. Your name and roll-number must appear in the first page. All plots must be numbered as specified in this assignment. Along with the soft-copy of your report, your working code (Matlab preferable) must be properly commented and be emailed to the TA, Ms. Prasikaa Shree, at ee21d700@smail.iitm.ac.in. Your report can be named "rollnumber-assignment2-report.m" and your working code can be named "rollnumber-assignment2-code.m". Else, if you have not used Matlab, we can also accept any other convenient file format for seeing the results and the code. Python submissions are also okay. The TA will get back to you if additional information is required. Please see other instructions, if any, in the WhatsApp group.