*Department of Electrical Engineering, IIT Madras*

**Due by Monday, 10am, Oct. 30, 2023**

# EE-4140: Digital Communication Systems

**Oct. 20, 2023** **Simulation Assignment #1** **50 marks**

**1. [1.5+1.5+3+4 = 10 marks]** Consider a raised cosine (RC) pulse shaped transmission. The impulse response of the pulse-shaping continuous time filter g(t) is given as follows:

$$g(t) = Sinc(\frac{\pi t}{T}) \times \left( \frac{Cos(\frac{\pi \beta t}{T})}{1 - \frac{4\beta^2 t^2}{T^2}} \right)$$

where the "excess bandwidth" factor $0 \leq \beta \leq 1$ and T is the symbol duration and Sinc(x)=Sin(x)/x. We use a discrete-time filter g(nTs) = g(n) where the sampling rate Ts = T/J, where the oversampling factor J is a positive integer. Also, the filter is truncated to 2LT symbol durations, i.e., LT symbol durations on *each side* of t=0, where L is a positive integer. The input bit (or symbol) stream I(kT) into this filter will be zero-interleaved with J-1 zeros in order to create the output sequence x(kTs) which is sent into a DAC.

Choose your own N-bit (with N=32) random sequence where I(k)$\in$ {+1, -1} in order to create a BPSK signal. Assume all the bits before and after this 16-bit pattern are zeros. You are to plot the output sequence (samples) x(kTs) using both "symbol-plot" as well as join them using "line-plot" using Matlab.

(1.1) Plot as Fig.1, x(kTs) for J=4, L=2, and β=0.
(1.2) Repeat (a) and include in Fig.1 another plot with the change L=4.
(1.3) Plot as Fig.2, *three* different plots of x(kTs) where all of them have J=8, L=4, but with *three* different excess bandwidth factors, namely, β=0, β=0.5, and β=1. Comment on your results in (1.2) and (1.3).

(1.4) Plotting the PSD $S_X(f)$ using Periodogram: We now perform Monte-Carlo simulations and averaging to approximate the PSD of x(kTs). For this Periodogram approach, generate R=100 runs, each of bit-sequence length N=1024, for the RC pulse-shape with β=0.5, J=8, L=4. This would give in each MC run, 1024xJ = 8192 samples of x(kTs). Then, take a 8192-point FFT, square the magnitude response, and average it point-wise over the R=100 trials. Divide this by R (=100) to get an estimate of the PSD of the x(kTs), namely $S_X(f)$. Plot this as Fig.3. Now, also estimate in a similar manner the PSD of x(kTs) when the following "rectangular pulse-shape" is used on the bit-stream, namely: $g(t) = \begin{cases} \frac{1}{\sqrt{T}}, 0 < t \leq T \\ 0, \text{otherwise} \end{cases}$. Plot this PSD estimate using the periodogram approach, also on the same Fig.3, and comment.

**2. [3+3+3+3+3 = 15 marks]** In this question, we compare the theoretical average probability of symbol error $P_E$ value and the computer simulated symbol error rate, SER, for three popular linear modulation schemes. The approximation on $P_E$ using bounds will also be studied.

Consider the same energy per bit, $E_B$, for the 2-PAM (or BPSK), 4-QAM (or QPSK), and 16-QAM signals. Assume the complex baseband AWGN measurement model r(k)=I(k)+v(k), where i.i.d. symbols I(k) are uncorrelated with the white Gaussian noise v(k). For the two complex signals (4-QAM and 16-QAM), the noise will be complex with variance $\sigma^2 = N_O/2$ per dimension. Vary the signal to noise ratio, SNR, in the log-scale as $10\log_{10}(E_B / \sigma^2)$ in 2dB steps from 0dB to 14dB, by changing the noise variance. Chose $E_B$=1 for all the signal sets. The *Q*(.) function or the *erfc*(.) function can be evaluated using Matlab. *Hint:* Note that the energy per symbol will be different for each *M* and is given by $E_S = \log_2(M) E_B$ where *M*=2, 4, or 16, for the given three signal sets given above, respectively. Express "2*d*", the distance between the 2 nearest points in the constellation as described in the class, as a function of $E_B$, in order to compute "*q*". From *q*, get $P_E$.

(2.1) Compute and plot as Fig.4, the $P_E$ for the 3 signals when sent thro the AWGN channel model. Plot the SNR against $\log_{10}(P_E)$ on the y-axis, for all the 3 signals.

(2.2) For the 4-QAM signal set, compute using the following approximations to $P_E$. Plot each of these values (bounds) on $P_E$ for the same range of SNRs and call this Fig.5.

(2.2.1) Union bound using *all* the pairwise symbol errors
(2.2.2) Union bound using only the *nearest neighbours*
(2.2.3) In part (2.2.2) above, replace the *erfc*() function with the Chernoff bound and replot.
(2.2.4) Plot also the accurate $P_E$ obtained in part (2.1) also in Fig.5.
(2.2.5) Comment on these results in Fig.5.

(2.3) Now, we wish to simulate the SER for 4-QAM (QPSK). For generating the transmit symbols, use uniform random variables between [0,1]. If the r.v. takes a value between 0 and 0.5, it is mapped to say -*d*; else, it is mapped to +*d*, where 2*d* is the distance between the symbols. Note that $d = \sqrt{E_S} = \sqrt{2E_B} = \sqrt{2}$ in this case. We will need two uniform r.v.s, one of the real part and one for the imaginary part. For noise, generate 2 Gaussian r.v.s with appropriate variance, one for the real and one for the imaginary part. SNR is varies by changing the noise variance. (Explain your approach). Generate about $10^5$ measurements r(k) to measure the SER over the same range of SNRs, i.e., 0dB to 14dB. Plot $\log_{10}$(SER) on the y-axis. Include this SER plot also into Fig.5 and comment.

(2.4) Repeat part (2.3) above for the 16-QAM signal set. Chose the uniform r.v. to symbol mapping appropriately. (Explain your approach). This SER plot should be called Fig.6. Plot the true $P_E$ for 16-QAM found in part (2.1) also in Fig.6.

(2.5) Compute the union bound on $P_E$ using *only the nearest neighbours* for the 16-QAM signal set. Plot this result also into Fig.6. Comment about all your results in Fig.6.

**3. [2+2+2+2+2 = 10 marks]**: 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 $\sigma_v^2$. The signal to noise ratio (SNR) measured on *r*(*k*), is then given by $1/\sigma_v^2$ (why?), and we vary the SNR on the measurements by varying $\sigma_v^2$ appropriately. We study the performance of a MLSE based approach implemented using the Viterbi Algorithm (VA).

Define a VA based sequence estimator, assuming perfect information about *F*(*z*) is available at the receiver. Generate for each SNR (in 2dB steps from 0dB to 16dB), say 100,002 symbols of 4-PAM using uniform random variables appropriately to get *I*(*k*). The noise *v*(*k*) is generated using the normal pdf, and the value returned is scaled using the current $\sigma_v$ (for the given SNR). The last 2 symbols (note: *L*-1 = 2 here for *F*(*z*)) of *I*(*k*) may be viewed as "tail symbols" which are known to the receiver, to terminate the VA in a known state. Measure SER only over the (unknown) remaining 100,000 symbols, for the following choices of "trace-back length" or decoding delay δ, namely:
(3.1) δ=3;
(3.2) δ=6;

(3.3) $\delta$=10;
(3.4) $\delta$=20;
(3.5) $\delta$=40;
Compute the $\log_{10}$SER versus $10\log_{10}$(SNR) for each choice of $\delta$ and plot all results in Fig. 7.

**4. [1+5+4+5 = 15 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 $E[|I(k)|^2] = \sigma_I^2 = 1$. Now, answer the following:
(4.1) Normalise the channel gain to unity by appropriate choice of $C$ in $G(z)$. What is this $C$ ?
(4.2) As in part (a), vary SNR by changing noise variance appropriately. Over 0dB to 20dB, evaluate the SER of the $2^5$ state VA by sending 100,004 symbols in each step of 4dB. Using decoding delay $\delta$=30, compute the $\log_{10}$SER versus $10\log_{10}$(SNR) and plot the results in Fig.8.
(4.3) Now, suppose, the VA is constructed by considering a truncated impulse response, where only the first 3 taps of $G(z)$ are taken into consideration, to give a $2^2$ state VA (i.e., the weaker taps, namely 0.15, -0.2, and -0.1 are ignored in constructing the VA). Note, however, that the measurements would have contribution from all the 6-taps of $G(z)$. For this case, evaluate and plot $\log_{10}$SER versus $10\log_{10}$(SNR), on the same Fig.8 as in part (4.2), for $\delta$=30.
(4.4)* Now, can you incorporate a decision-feedback mechanism to improve the performance of this $2^2$ state VA? *Hint*: Sub-optimal low-delay decisions can be made at each state based on the corresponding survivor sequence for that state, and this contribution can be subtracted before computing the transition metrics. If you need more details, read the DDFSE paper by Duel-Hallen and Heegard (IEEE Trans. on Communications, 1989). Plot this result for $\delta$=30 also in Fig.8, and comment on your results.
*indicates higher order to difficulty

---

## *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-assignment1-report.m” and your working code can be named “rollnumber-assignment1-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.