Generation of Pseudo-Random Numbers

Dr. D. Rawal

Dept. of ECE, The LNMIIT Jaipur

January 09, 2015

1 Objective

- To generate the uniformly distributed random numbers by linear congruent generator (LCG).
- To generate the standard Normal distributed numbers by Box-Muller transformation.
- To generate the standard Normal distributed numbers by central limit theorem (CLT).
- To generate the standard Normal and α-μ distributed numbers by acceptance-rejection method.

2 Introduction

2.1 Uniformly Distributed Random Numbers

The pseudorandom numbers with probability distributions given as:

$$p(x) = \begin{cases} 1 & \text{, if } 0 < x < 1; \\ 0 & \text{, otherwise} \end{cases} \tag{1}$$

are known as uniform random numbers and denoted by $U(0,1)$ [1]. Moreover, notation $U(a,b)$ is used to denote absolutely continuous uniform distribution over the interval (a,b). D. H. Lehmer in 1948 proposed a simple *linear congruential generator (LCG)* as a source of random number. The form of LCG is [1]:

$$x_{n+1} = [ax_n + c] \bmod m \tag{2}$$

where,

$$x = \text{sequence of uniformly distributed random numbers}$$
$$a = \text{multiplier}$$
$$c = \text{increment}$$
$$m = \text{modulus}$$
$$x_0 = \text{seed}$$

Each x_n is scaled into the unit interval (0,1) by division by m, that is,

$$u_n = x_n/m \tag{3}$$

The period of a general LCG is at most m, and for some choices of factor a much less than that. Provided that the offset c is nonzero, the LCG will have a full period for all seed values if and only if [2]:

- c and m are relatively prime.
- $a - 1$ is divisible by all prime factors of m.
- $a - 1$ is a multiple of 4 if m is a multiple of 4.

2.2 Box-Muller Transformation

A simple method to generate standard Normal distributed random numbers, is Box-Muller transformation [3]. It states that, if U_1 and U_2 are independent uniform random variables on the interval $[0, 1]$, then,

$$X_1 = \sqrt{-2ln(U_1)}sin(2\pi U_2) \tag{4}$$

will be exactly $\mathcal{N}(0, 1)$. The method is relatively fast, and provides exact normal random variables. Sometimes $cos(\cdot)$ in place of $sin(\cdot)$ is also used.

2.3 Central Limit Theorem (CLT) Approach

CLT is an important, but less accurate approach to generate Normal random numbers. The theorem states that, if $X_1, X_2, \ldots, X_n$ are independent distributed uniform random variables according to some common distribution function with mean μ and finite variance σ^2, then as the number of random variables increases indefinitely, the random variable,

$$Y = \frac{\sum_{i=1}^{n} X_i - n\mu}{\sqrt{n\sigma^2}} \tag{5}$$

converges to $\mathcal{N}(0, 1)$ [4].

2.4 Acceptance-Rejection Method

Acceptance-rejection method is a powerful tool for generating random numbers, if the probability density function (pdf) of the distribution is known. Basically this method works by generating uniform random numbers repeatedly, and accepting only those that meet certain conditions. These conditions for accepting the uniform random numbers are so designed that the accepted random numbers follow the given distribution. For the rejection method to be applicable, the pdf $f(x)$ of the distribution must be non zero over an interval, say (a, b). Let function $f(x)$ is bounded by the upper limit f_{max} which is the maxima of $f(x)$.

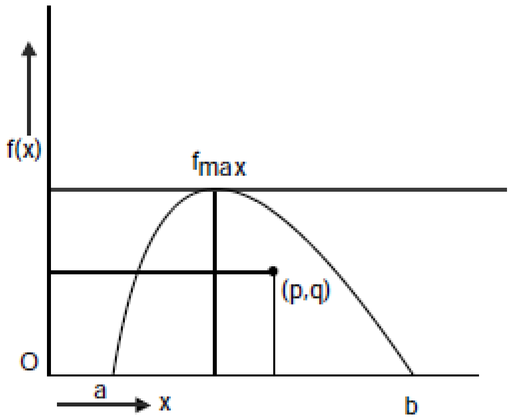


Figure 1: Rejection method

The method can be implemented following the given steps:

- *Step 1*: Generate a uniform random number u lying between $(0, 1)$. Let us define p as
$$p = a + (ba)u \tag{6}$$
This means p lies between a and b.

- *Step 2*: Generate another uniform random number v lying between $(0, 1)$. Let us define $q = f_{max}v$.

- *Step 3*: If $q > f(y)$, then reject the pair (u, v), otherwise accept u as the random number following the distribution $f(x)$.

The pdf for Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$is given as:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[-\frac{(X-\mu)^2}{2\sigma^2}\right] \tag{7}$$

where, μ and σ denote the mean and standard deviation of the distribution respectively.

One of the most important distribution, useful for modeling of short-term signal variation (fading) is generalized α-μ distribution. The model can closely represent some well-known fading distributions such as Rayleigh, Nakagami-m, Exponential, Gamma etc for certain value of α and μ. The pdf of the envelope for α-μ distribution is given as [5]:

$$f_R(r) = \frac{\alpha \mu^{\mu} r^{\alpha\mu - 1}}{\tilde{r}^{\alpha\mu} \Gamma(\mu)} exp\left(-\mu \frac{r^{\alpha}}{\tilde{r}^{\alpha}}\right) \tag{8}$$

where, $\Gamma(\cdot)$ is the well-known Gamma function and $\tilde{r}$ is the α-root mean value of the envelope random variable.
Table 1 gives the values of α and μ for which the α-μ distribution converges to certain well-known fading distributions.

Table 1: Values of α and μ for different known distributions

Type of distribution	α	μ
One-sides Gaussian	2	0.5
Rayleigh	2	1
Exponential	1	1
Nakagami-m	2	m
Gamma	1	a

3 Tasks

- **Task 1:** Generate a sequence of 100000 uniformly distributed random numbers. Plot the histogram of the generated numbers (without using any *in-built* function) and the pdf of the generated distribution. Also find the mean and variance of the generated random variate. Use the following values:

  (i) $a = 5$, $c = 1$, seed $= 1$, $m = 16$.
  (ii) $a = 5$, $c = 1$, seed $= 1$, $m = 32$.
  (iii) $a = 1$, $c = 0$, seed $= 1$, $m = 16$.
  (iv) $a = 7^5$, $c = 0.231$, seed $= 1$, $m = 2^{31}$.

  Put your comments about the periodicity of the generated random numbers

- **Task 2:** Generate a sequence of 100000 random numbers following $\mathcal{N}(0, 1)$ by Box-Muller transformation, CLT and Acceptance-rejection method. Plot the histogram and pdf of the generated random variables. Compare the simulated pdf with the theoretical one and find the mean-square error (MSE).

- **Post-Lab Task:** Generate a sequence of 100000 random numbers following α-μ distribution with $\alpha = 2$, $\mu = 1$ and $\tilde{r} = 1$. Find out the pdf of the generated random numbers and compare the result with the theoretical pdf. Find out the MSE between theory and the simulation.

4 Instructions for the submission of the report

Prepare a detailed report of the performed tasks including the MATLAB code, strictly in LaTeXand submit the same before the next lab session.

References

[1] JE Gentle, *Random Number Generation and Monte Carlo Methods*, Springer, 2003.

[2] DE Knuth, *The Art of Computer Programming,* Addison-Wesley, 1997.

[3] CEP Box, and ME Muller, "A Note on the Generation of Random Normal Deviates," *Ann. Math. Stat.*, vol. 29, 1959, pp. 610-11.

[4] VP Singh, *System Modeling and Simulation*, New Age International Publishers, 2009.

[5] MD Yacoub, "The α-μ Distribution: A Physical Fading Model for the Stacy Distribution," *IEEE Tran. Veh. Technol.,* vol. 56, no. 1, pp. 27-34, Jan. 2007.

Lab Handout #2
Signal Modeling using Padé and Prony Approximation

Instructor: Dr. D. Rawal

Dept. of ECE, The LNMIIT Jaipur

January 23, 2015

1 Objective

- Pole-Zero modeling of a given sequence using Padé approximation.
- Filter design using Padé approximation.
- Pole-Zero modeling of a given sequence using Prony approximation.
- Filter design using Prony approximation.

2 Introduction

The term *signal modeling* refers to the task of describing a signal with respect to an underlying structure - a model of the signal's fundamental behavior [1]. Signal modeling is used for signal compression, prediction, reconstruction and understanding. In data compression, for example, a waveform $x(n)$ consisting of N data values, $x(0), x(1) \dots x(N-1)$, is to be transmitted over a communication channel or archived on a tape or disk. One method that may be used for transmission or storage is to process the signal on a point-by-point basis, i.e., transmit $x(0)$ followed by $x(1)$ and so on. However it is possible to model the signal with a small number of parameters $k \ll N$, then it would be more efficient to transmit or store these parameters instead of the signal values [2]. There are various methods to model a signal:

(a) Parametric signal modeling: Used to reduce a complex process with many variables to a simpler one with few parameters.

(b) Direct modeling: To match impulse response of a rational system $H(z)$ to arbitrary signal $x(k)$.

(c) Indirect modeling: Minimize least squares error generated by filtering x(k) with FIR filter. Since the FIR filter will attempt to remove poles of X(z) it should be a good estimate of denominator.

Some of the most popular methods for signal modeling include Padé approximation, Prony's method, Auto-correlation modeling / All-pole modeling, Covariance method, Levinson Durbin method etc.

2.1 Padé Approximation

Signal modeling based on least square error criterion leads to a mathematically intractable solution [2]. Unlike the least squares solution, the Padé approximation only requires solving a set of linear equations. Let $x(n)$ be the signal to be modeled as the unit sample response of a causal linear shift-invariant filter. The system function with p poles and q zeros is given as:

$$H(z) = \frac{B_q(z)}{A_p(z)} = \frac{\sum_{k=0}^{q} b_q(k) z^{-k}}{1 + \sum_{k=1}^{p} a_p(k) z^{-k}} \tag{1}$$

It is clear from (1), that the system has $p+q+1$ degrees of freedom, and hence, it is plausible to force the filter output, $h(n)$, to be equal to the given signal $x(n)$ for $p+q+1$ values of n. The numerator and the denominator coefficients for the Padé approximation for modeling a signal as the unit sample response of a linear shift-invariant system with p poles and q zeros is given as [2]:
Denominator coefficients:

$$\begin{bmatrix} x(q) & x(q-1) & \dots & x(q-p+1) \\ x(q+1) & x(q) & \dots & x(q-p+2) \\ \vdots & \vdots & \vdots & \vdots \\ x(q+p-1) & x(q+p-2) & \dots & x(q) \end{bmatrix} \begin{bmatrix} a_p(1) \\ a_p(2) \\ \vdots \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} x(q+1) \\ x(q+2) \\ \vdots \\ x(q+p) \end{bmatrix} \tag{2}$$

Numerator Coefficients:

$$\begin{bmatrix} x(0) & 0 & 0 & \dots & 0 \\ x(1) & x(0) & 0 & \dots & 0 \\ x(2) & x(1) & x(0) & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ x(q) & x(q-1) & x(q-2) & \dots & x(q-p) \end{bmatrix} \begin{bmatrix} 1 \\ a_p(1) \\ a_p(2) \\ \vdots \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} b_q(0) \\ b_q(1) \\ b_q(2) \\ \vdots \\ b_q(q) \end{bmatrix} \tag{3}$$

2.2 Prony's Method

The limitation with the Padé approximation is that it only utilizes values of signal $x(n)$ over the interval $[0, p+q]$, to determine the model parameter. Over the range of $[0, p+q]$ coefficients, the model of the signal is error free, but there is no guarantee on the behavior of the model for signal approximation with $n > p+q$. Prony's method relax the requirement that the model to be exact over the interval $[0, p+q]$, in order to produce better approximation to the signal for all values of n [2]. The governing equations are given by:

Denominator coefficients:

$$\begin{bmatrix} r_x(1,1) & r_x(1,2) & r_x(1,3) & \dots & r_x(1,p) \\ r_x(2,1) & r_x(2,2) & r_x(2,3) & \dots & r_x(2,p) \\ r_x(3,1) & r_x(3,2) & r_x(3,3) & \dots & r_x(3,p) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_x(p,1) & r_x(p,2) & r_x(p,3) & \dots & r_x(p,p) \end{bmatrix} \begin{bmatrix} a_p(1) \\ a_p(2) \\ a_p(3) \\ \vdots \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} r_x(1,0) \\ r_x(2,0) \\ r_x(3,0) \\ \vdots \\ r_x(p,0) \end{bmatrix} \tag{4}$$

where,

$$r_x(k,l) = \sum_{n=q+1}^{\infty} x(n-l)x^*(n-k) \tag{5}$$

Numerator Coefficients:

$$b_q(n) = x(n) + \sum_{k=1}^{p} a_p(k)x(n-k); \quad n = 0, 1, \dots, q \tag{6}$$

3 Tasks

- **Task 1:** Model the sequence $h = [1, 1.5, 0.75]$ via Padé approximation with the linear shift-invariant system specifications as follows:

  (a) $p = 2$, $q = 0$

  (b) $p = q = 1$

  Find the numerator and denominator coefficients.

- **Task 2:** Model the sequence $x(n)$ as an approximation with linear shift-invariant system via Prony's method.

$$x(n) = \begin{cases} 1, & \text{if } 0 \le n < 22 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Find the numerator and denominator coefficients for the following system specifications:

(a) $p = q = 1$

(b) $p = q = 3$

- **Post-lab Task 1:** Design a linear phase low-pass filter having cut-off frequency of $\pi/2$. The frequency response of the ideal low pass filter is:

$$I(e^{j\omega}) = \begin{cases} e^{jn_d\omega}, & |\omega| < \pi/2 \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

where, n_d is the filter delay. The unit sample response of the filter is:

$$i(n) = \frac{sin\left[(n - n_d)\pi/2\right]}{(n - n_d)\pi} \tag{9}$$

The system specification is as follows:

$$\text{Filter delay} = 5$$
$$p = q = 5$$

Find the first $p + q + 1$ values of $i(n)$, the numerator and the denominator coefficients with the Padé approximation. Draw the frequency response of the filter and plot the approximation error $e'(n) = i(n) - h(n)$.

- **Post-lab Task 2:** Solve the above problem via Proby's method and comment on the error in approximation via the two methods.

4 Instructions for the submission of the report

Prepare a detailed report of the performed tasks including the MATLAB code, strictly in LaTeXand submit the same before the next lab session.

References

[1] MM Goodwin, *Adaptive Signal Models: Theory, Algorithms, and Audio Applications*, PhD Thesis, University of California, Berkeley, 1997.

[2] MH Hayes, *Statistical Digital Signal Processing and Modeling,* John Wiley & Sons, Inc. New York, USA, 1996.

Lab Handout # 3
Noise Addition and Cancellation

Modeling and Simulation LAB

Instructor: D. RAWAL

Dept. of ECE, The LNMIIT, Jaipur

Time : 3:00 Hour | Maximum Marks : 10

Instructions and information for students

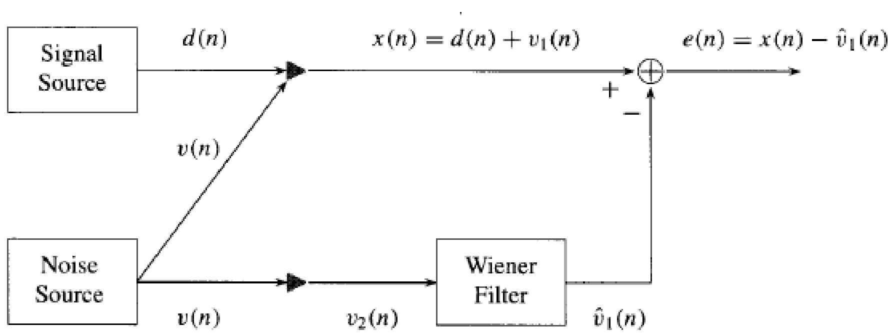
- This Lab Handout consists of 2 pages. Please check that you have a complete copy.
- Simulate in matlab or any other Software.

Objective:

1) Additive White Gaussian Noise cancellation using Wiener filter.
2) Noise cancellation in music file using wiener filter.

1) **Do as following.**
   a) **Wiener filter based noise cancellation**
      i) Generate a desired signal $d(n) = sin(n\omega)$ take f = 100 hz, 10khz,
      ii) Generate noise
      $$v1(n) = 0.8 \cdot v1(n-1) + g(n)$$
      iii) Generate noise
      $$v2(n) = -0.6 \cdot v2(n-1) + g(n)$$
      Where g(n) is white noise with mean zero variance 1.
      iv) Find out autocorrelation r_{v2} and generate $Rv2$ autocorrelation toeplitz matrix.
      v) Find the cross correlation r_{xv2} vector.
      vi) Use wiener-hopf eq.
      $$w = (Rv2)^{-1} \cdot r_{xv2}$$
      and find out wiener filter coefficients for noise cancellation.
      vii) Repeat the above experiment for wiener filter with p = 6 and p = 12 poles.
      viii) plot the input and output results.

      i) **Wave file noise addition(Do as following).**
         A) Take a wave file and store the samples in d(n).
         B) Generate noise
         $$v1(n) = 0.8 \cdot v1(n-1) + g(n)$$
         C) Generate noise
         $$v2(n) = -0.6 \cdot v2(n-1) + g(n)$$
         Where g(n) is white noise with mean zero variance 1.
         D) Find out autocorrelation r_{v2} and generate $Rv2$ autocorrelation toeplitz matrix.
         E) Find the cross correlation r_{xv2} vector.
         F) Use wiener-hopf eq.
         $$w = (Rv2)^{-1} \cdot r_{xv2}$$
         and find out wiener filter coefficients for noise cancellation.



b) **Simulink based Modeling. (do not use inbuilt function)**

i) Repeat the whole experiment in Simulink model.

————————————————WELL, It's DONE————————————————-

Lab Handout # 5
Estimation, System Identification and Adaptive Equalization.

Modeling and Simulation LAB

Instructor: D. RAWAL

Dept. of ECE, The LNMIIT, Jaipur

Time : 3:00 Hour | Maximum Marks : 10

Instructions and information for students

- This Lab Handout consists of 2 pages. Please check that you have a complete copy.
- Simulate in matlab or any other Software.

Objective:

1) System Identification.
2) Noise cancellation using LMS adaptive filter.

Useful Theory stepwise:

1) The weights are continuously adapting using
$$W_{n+1} = W_n - \nabla(E[e(n)^2])$$
2) The error between desired signal d(n) and system generated signal x(n) is
$$e(n) = d(n) - {W_n}^H \cdot \mathbf{x(n)}$$
3) The Mean square error is given by $E[e(n)^2]$.
$$E[e(n)^2] = E[e(n) \cdot ((e(n)^*)] = E[e(n) \cdot (d(n) - [{W_n}^H \cdot \mathbf{x(n)}]^*)]$$
4) Differentiate with respect to filter coefficient W_n.
$$\nabla E[e(n)^2] = \frac{\partial}{\partial {W_n}^*}(E[e(n)^2]) = E(e(n) \cdot \mathbf{x(n)}^*)$$
5) Initially, any adaptive filter starts with coefficient w(n) with zero values.
6) It is updated at every next sample such that the mean square error e(n) is minimized.
$$W_{n+1} = W_n + E[e(n) \cdot \mathbf{x(n)}^*]$$
7) However since Expectation as in Eq.(1) is not known generally at the receiver, The W_n updating is done with current sample value and is given by
$$W_{n+1} = W_n + (Stepsize) \cdot [e(n) \cdot \mathbf{x(n)}^*]$$
8) Take stepsize between 0 and 1, (here take Stepsize = 0.595 or 0.995)

1) **Do as following.**
   a) **System Identification:**
      i) Generate a random binary sequence of 10000 values. Let's call it x sequence.
      ii) Pass the above sequence through h = [1, 0.5] system.
      iii) The generated output sequence is d with 10001 values.
      iv) Now consider that first 200 values of integer sequence is known to you (Which is called training sequence).
      v) Find out the optimized coefficient of the adaptive filter w with this 200 values using Eq.(3)
      vi) Plot the mean square error between h and w given by $E[(h - w)^2]$.

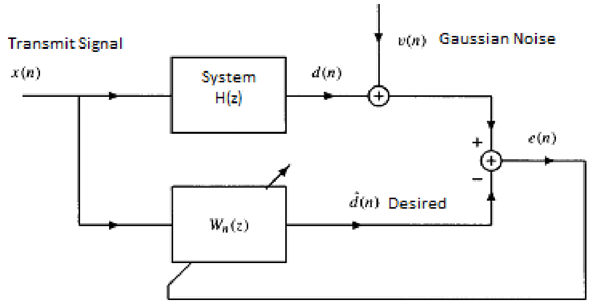


Fig. 1. System Identification

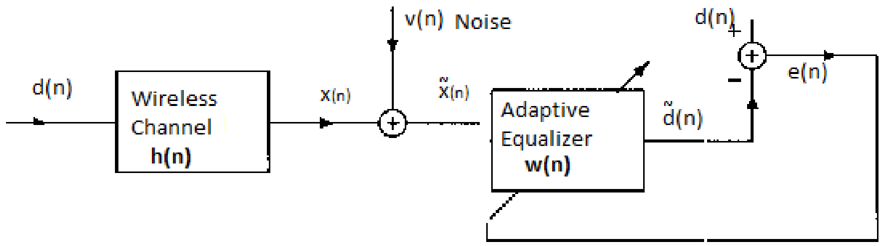


Fig. 2. Adaptive Noise Cancellation

b) **Adaptive Noise Cancellation(Do as following).**
   i) Exchange the role of x and d in the above system identification problem, it will become adaptive equalizer.
   ii) Follow all the above steps of previous part.
   iii) Find out the total error between input d and recovered sequence $\hat{d}$.

c) **Repeat above Problem with noise added/ snr varied from 0-24 in step of 4 dB.**
   i) Plot the error vs snr graph.

d) **Do the following. (Audio signal analysis and noise reduction)**
   i) Take an audio wav file. (you can convert any mp3 file to wav file using mp3towav converter) Also use "mp3 splitter" to select smallest part of the audio file.
   ii) Use inbuilt "wavread" function to store its samples into some variable X.
   iii) Pass the sequence x through h = [1, 0.5, 0.8, 0.15]
   iv) Add Gaussian noise (or snr varying from 0 to 24 in step of 4, assume signal power is 1).
   v) After noise added let's call it sequence xbar.
   vi) Take training seq. as first 200 samples from d.
   vii) Find out optimum adaptive equalizer coefficient w.
   viii) Filter or equalize the Noisy sequence xbar with adaptive equalizer w.
   ix) Write wavfile using "wavwrite".
   x) Conclude which snr range is better for this equalizer.

e) **Repeat the experiment in CCS and on DSK TMS320C5416.**

——————————————WELL, It's DONE——————————————

Lab Handout # 6
Wireless Channel Modeling: Flat, Multipath

Modeling and Simulation LAB

Instructor: D. RAWAL

Dept. of ECE, The LNMIIT, Jaipur

Time : 3:00 Hour | Maximum Marks : 10

Instructions and information for students

- This Lab Handout consists of 2 pages. Please check that you have a complete copy.
- Simulate in matlab or any other Software.

Objective:

1) Channel Modeling : Indoor, Outdoor channel.
2) Doppler spread based fast fading channel modeling.

1) **Do as following.**
   a) **Useful Practice I**
      i) Take $h = [1, 0.5, 0.3]$
      ii) Find the autocorrelation function of above step.
      iii) Find the power spectrum of step(i) and (ii).
      iv) Conclude your remarks.

   b) **Useful Practice II**
      i) Generate Gaussian random variable with mean zero and variance 1.
      ii) Add two or more above generated complex Gaussian random variables.
      iii) Comment on the resultant distribution.

   c) **Useful Practice III**
      i) Generate Gaussian random variable with mean one and variance 1.
      ii) Add two or more above generated complex Gaussian random variables.
      iii) Comment on the resultant distribution.
      iv) Distinguish the difference between results from above steps.

   d) **Flat fade channel: Rayleigh, Rice**
      i) Generate two Gaussian random variables X and Y with mean zero and variance $\sigma^2 = \frac{1}{2}$.
      ii) Make a complex random variable h using above two gaussian random variable results in rayleigh random variable.
      iii) Repeat above steps for 1000 times and plot the pdf of magnitude of the resultant variable h.
      iv) Verify the PDF resembles the function given belwo by varying a from $[0 : 3]$ in steps of 0.01.

$$f_A(a) = 2a \cdot e^{-a^2}$$

      v) Repeat all the above steps and add an LOS component (fix real number) in real random variable X.
      vi) Comment on the resultant distribution.

e) **Multipath Channel Modeling and useful models**

i) Take h = [1, 0.01].
ii) Find the spectrum of channel h.
iii) Take h = [1, 0.3, 0.4, 0.2, 0.5, 0.21, 0.6, 0.49].
iv) Find the spectrum of channel h.
v) Comment on the types of the channel.
vi) Find out the flat bandwidth(Coherence Bandwidth) of both channels.
vii) Find out a way to increase the flat bandwidth.
viii) Consider all the power delay profile given below, calculate delay spread τ_{ds} and coherence bandwidth B_c.
ix) Take $fc = 1Ghz,\ V = 30, 60, 120, 300km/hr,\ Fs = 11.2Mhz.$ calculate doppler spread f_d and coherence time T_c.
x) Verify the multipath channels using simulink block inbuilt channel blocks.

f) **Multipath Channel Models from PDP**

i) Indoor A channel:
delay=[0 50 110 170 290 310]; nsec
power=[0.0 -3.0 -10.0 -18.0 -26.0 -32.0]; dB
ii) Indoor B channel:
delay=[0 100 200 300 500 700];
power=[0.0 -3.6 -7.2 -10.8 -18.0 -25.2];
iii) Pedestrian A channel:
delay=[0 110 190 410];
power=[0.0 -9.7 -19.2 -22.8];
iv) Pedestrian B channel:
delay=[0 200 800 1200 2300 3700];
power=[0.0 -0.9 -4.9 -8.0 -7.8 -23.9];
v) Vehicular A channel:
delay = [0 310 710 1090 1730 2510];
power = [0.0 -1.0 -9.0 -10.0 -15.0 -20.0];

g) **Doppler spread**

i) Channel h is represented by

$$h = a \cdot e^{j \cdot 2\pi \cdot f_{dmax} \cdot t}$$

where a is attenuation constant and f_{dmax} is maximum doppler spread around carrier frequency. The doppler frequency f_d is given by

$$f_d = \frac{V \cdot f_c}{c} \cdot (Cos(\theta))$$

————————————WELL DONE————————————-

Lab Handout # 7
Wireless Channel Modeling: Receive Diversity

Modeling and Simulation LAB

Instructor: D. RAWAL

Dept. of ECE, The LNMIIT, Jaipur

Time : 3:00 Hour | Maximum Marks : 10

Instructions and information for students

- This Lab Handout consists of 1 page. Please check that you have a complete copy.
- Simulate in matlab or any other Software.

Objective:

1) Receive Diversity : MRC, SC.

1) **Do as following.**
   a) **Receive Diversity: Maximal Ratio Combining**
      i) Generate a rayleigh distributed flat fade SIMO channels.
      ii) Generate a BPSK sequence x and transmit through the above generated SIMO channels.
      iii) Combine received signal using matched channel response $\frac{h^*}{||h||}$.
      iv) Vary the SNR in Range of 0 to 35 in steps of 3 dB and plot the BER vs SNR performance.
      v) Compare it with theoretical exact BER expression given by

$$Average(P_e) = \left(\frac{1-\lambda}{2}\right)^L \sum_{l=0}^{L-1} \text{(L+l-1)}\mathbf{C}_l \left(\frac{1+\lambda}{2}\right)^l$$

      where $\lambda = \sqrt{\frac{SNR}{2+SNR}}$ and L si the total no. of branches.
      vi) The Asmptotic BER expression is given by

$$Asymptotic(P_e) = \text{(2L-1)}\mathbf{C}_L \left(\frac{1}{2SNR}\right)^L$$

   b) **Selection Combining**
      i) Generate a rayleigh distributed flat fade SIMO channels.
      ii) Generate a BPSK sequence x and transmit through the above generated SIMO channels.
      iii) Choose the highest received SNR branch and apply thresholding to estimate the transmitted signal x.
      iv) Vary the SNR in Range of 0 to 35 in steps of 3 dB and plot the BER vs SNR performance.
      v) Compare it with theoretical exact BER expression given by

$$Average(P_e) = \left(\frac{1}{2}\right) \cdot \sum_{l=0}^{L-1} \mathbf{LC}_l \left(1 + \frac{l}{SNR}\right)$$

      vi) Conclude your remarks.

————————————————————WELL DONE————————————————————-

Lab Handout # 8
Performance over Selective Decode and Forward(DF) based Cooperative Communication

Modeling and Simulation LAB

Instructor: D. RAWAL
Dept. of ECE, The LNMIIT, Jaipur

Time : 3:00 Hour | Maximum Marks : 10

Instructions and information for students

- This Lab Handout consists of 1 page. Please check that you have a complete copy.
- Simulate in matlab or any other Software.

Objective:

1) BER peroformance over Selective DF.

1) **Do as following.**
   a) **Selective DF**
      i) Generate a BPSK sequence x and transmit through the single relay with single antenna system.
      ii) Vary the SNR in Range of 0 to 35 in steps of 3 dB and plot the BER vs SNR performance.
      iii) Compare it with theoretical Asymptotic probability of error/BER expression given by

$$P_e = \frac{1}{SNR^2}\left(\frac{1}{4\alpha^2\delta_{s,d}^2\delta_{s,r}^2} + \frac{3}{4\alpha(1-\alpha)\delta_{s,d}^2\delta_{r,d}^2}\right)$$

      iv) Repeat all the above steps for a cooperative selective DF relay system with relay and Destination node has Nr and Nd multiple receiving(i.e, 2,3....) antennas respectively.
      v) In the above case, The asymptotic BER is given by

$$\begin{aligned} P_e = (2Nd\text{-}1)\mathbf{C}_{Nd} \cdot (2Nr\text{-}1)\mathbf{C}_{Nr} \cdot \frac{1}{2^{Nd+N_r}}\frac{1}{\delta_{s,d}^{2Nd}\delta_{s,r}^{2Nr}}\left(\frac{1}{\alpha^{Nd+Nr}}\frac{1}{SNR^{Nd+Nr}}\right) \\ +(4Nd\text{-}1)\mathbf{C}_{2Nd} \cdot \frac{1}{2^{2Nd}}\frac{1}{\delta_{s,d}^{2Nd}\delta_{r,d}^{2Nd}}\left(\frac{1}{\alpha^{Nd}(1-\alpha)^{N_d}}\frac{1}{SNR^{2Nd}}\right) \end{aligned} \tag{1}$$

      vi) Compare theoretical and simulation results.
      vii) Repeat the whole experiment in simulink using Embed Matlab block.

——————————————————WELL DONE——————————————————-

Lab Handout # 9
BER Performance of Multiuser MIMO(MU-MIMO) using Block Diagonalization-Zero Forcing (BD-ZF) precoding

Simulation LAB-II

Instructor: D. RAWAL

Dept. of ECE, The LNMIIT, Jaipur

Time : 2:00 Hour | Maximum Marks : 5

Instructions and information for students

- This Lab Handout consists of 1 page. Please check that you have a complete copy.
- Simulate in matlab or any other Software.

Objective:

1) Cancellation of Inter user Interference using Block Diagonalization.
2) Mitigate Inter antenna interference using Zero Forcing.

1) **MU-MIMO System:** Use of BD-ZF precoding in MU-MIMO system to mitigate inter-user, intra-user interference.
2) Consider downlink scenario with N users, each user having N_u antennas such that total Base station Antennas $N_B = N \times N_u$.
   a) Partiion the channel matrices into corresponding downlink channel matrices set. i.e if user having 2 antennas and 2 users are there, than H_1^{DL} corresponding to dwonlink channel 2×4, similar for the other user.
   b) Apply SVD on the corresponding set of Downlink Matrices, let's say $H_1^{DL} = [U_1, S_1, V_1]$, similarly for all the partiitoned matrices.
   c) Use the null space of all the V matrices to create null in the other users direction. For ex. continuing above case,

$$H_1^{DL}{}_{2\times 2} = \begin{bmatrix} \bar{U}_1 & \bar{U}_2 \end{bmatrix}_{2\times 2} \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \end{bmatrix} \begin{bmatrix} \bar{V}_1 \\ \bar{V}_2 \\ \bar{V}_3 \\ \bar{V}_4 \end{bmatrix}_{4\times 4}$$

   d) The Null space vectors i.e $\bar{V}_3, \bar{V}_4$ are used as precoding vectors W_2. Similarly one can find W_1 from null space vector set of H_2^{DL}.
   e) Using percoding the interference contribution of one user towards other user is made zero as can be verified by

$$H_1^{DL}{}_{2\times 2} \cdot W_2 = \begin{bmatrix} \bar{U}_1| & \bar{U}_2 \end{bmatrix}_{2\times 2} \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \end{bmatrix} \begin{bmatrix} \bar{V}_1 \\ \bar{V}_2 \\ \bar{V}_3 \\ \bar{V}_4 \end{bmatrix}_{4\times 4} \begin{bmatrix} \bar{V}_3 & \bar{V}_4 \end{bmatrix}_{4\times 4} = 0$$

   f) Using Precoding $W_1 \cdot X_2 + W_2 \cdot X_1$ interuser interference is made zero. and channel is block diagonalized.
3) Use ZF method to overcome inter antenna interference, Change the SNR and find out the BER performance.

——————————————————WELL, It's DONE——————————————————-