

## Lab #3 Random Signal Transformations and Quantization

### 1 Purpose

In this lab, you will be introduced to transformations of random variables. You will learn how transformations can be used to generate samples for a random variable with a different probability density function. You will also implement (i) an uniform pulse code modulator (PCM) that uses a uniform quantizer; and (ii) a nonuniform PCM with a nonuniform quantizer. The nonuniform quantizer is implemented by first distorting the signal using a logarithmic compression characteristic, followed by a uniform quantizer. The logarithmic compression characteristic used in the project is the  $\mu$ -law compander.

You will complete the following three simulations in MATLAB and submit your solutions along with a soft copy of the code in the form of a report.

### 2 Objectives

By the end of this project, you will be able to:

- 1) Generate samples of a random variable for any given probability density function by transforming uniformly distributed random variables.
- 2) Implement a uniform quantizer used in the pulse code modulation (PCM) system to represent sampled analog signals.
- 3) Augment the PCM system with a compander used to transmit signals with a large dynamic range over communication channels with smaller dynamic range capability.

### 3 References

- 1) Bernard Sklar text: Sections 2.6 - 2.7. Pages 79 – 84.

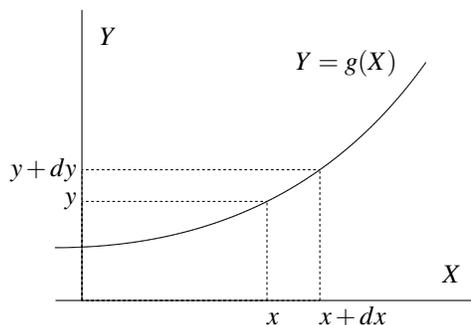


Figure 1: Transformations of variables.

## 4 Transformations of Random Variables

A situation that frequently occurs in the analysis of communication systems concerns a random variable  $Y$  that is functionally related to another random variable  $X$ , for example via some function  $g(\cdot)$

$$Y = g(X). \quad (1)$$

Typically the probability density function (PDF)  $p_X(x)$  of  $X$  is known and it is desired to determine the PDF  $p_Y(y)$  of the resulting random variable  $Y$ .

An example of a typical situation that a designer considers is one where the PDF of a transmitted signal is known (after all we are the ones that transmitted it) and it is often required to evaluate the PDF of the signal received through a nonlinear channel (we don't know exactly the channel that the received signal went through). A complete discussion of this problem is beyond the scope of this lab but a few elementary concepts can be presented and will be useful for subsequent discussions.

To gain a better footing, the general relations between our signals of interest are sketched in Fig. 1. It is clear from Fig. 1 that whenever the random variable  $X$  lies between  $x$  and  $x + dx$ , the random variable  $Y$  will lie between  $y$  and  $y + dy$ . Since the probabilities of these event ranges are equal we can state:

$$p_Y(y)dy = p_X(x)dx \quad (2)$$

from which the required PDF can be expressed as

$$p_Y(y) = p_X(x) \frac{dx}{dy} \quad (3)$$

$$= p_X(x) \frac{d[g^{-1}(y)]}{dy} \quad (4)$$

where  $g^{-1}(\cdot)$  is the inverse of  $g(\cdot)$  (like  $\tan^{-1}$  is the inverse of  $\tan$ ) and where we obviously took into account the relation

$$X = g^{-1}(Y). \quad (5)$$

In Fig. 1 we implied that  $g(X)$  is a monotonically increasing function of  $X$ . A similar result to that shown in (3) is obtained when  $g(X)$  is a monotonically decreasing function of  $X$ . In this latter case the derivative  $\frac{dx}{dy}$  will be negative. Since a PDF must be positive, we take the absolute value of the derivative  $\frac{dx}{dy}$  in (3). Hence, for either situation

$$p_Y(y) = \left| \frac{dx}{dy} \right| p_X(x)|_{X=g^{-1}(Y)} \quad (6)$$

In summary, the expression (6) is used to calculate the PDF of any random variable that is a function of a random variable with a known PDF.

**Example 1:** Consider now a specific example of the transformation of random variables by assuming that the random variable  $X$  has the PDF

$$p_X(x) = e^{-x}U(x) \quad (7)$$

where  $U(x)$  is the unit step starting at  $x = 0$ . This is a so-called “exponential distribution”. Now, also assume that the random variable  $Y$  is related to  $X$  by

$$Y = X^3 \quad (8)$$

$$\text{or } X = Y^{1/3}. \quad (9)$$

That is for any realization of the random variable  $X$  we have

$$x = g^{-1}(y) = y^{1/3}. \quad (10)$$

Having noted our functional relation between  $y$  and  $x$  we can calculate the derivative  $\frac{dx}{dy}$

$$\frac{dx}{dy} = \frac{d[g^{-1}(y)]}{dy} = \frac{1}{3y^{2/3}}. \quad (11)$$

Using (6) the PDF of  $Y$  is thus

$$p_Y(y) = \frac{1}{3}y^{-2/3}e^{-y^{1/3}}U(y). \quad (12)$$

■

## 5 Computer Generation of Random Variables

A digital computer is often used to simulate systems in order to estimate their performance with noise prior to the actual construction of that system. These simulations usually require that random numbers be generated to represent both the signal and the noise in the system. As we have already discussed in class these are values (realizations) of random variables having prescribed probability distributions.

If software exists for the specified distribution, there is no problem; just use the given code to run your simulation. However, if a computer library does not contain the desired random generation program, it is necessary for the simulation to generate its own random variables. Here we briefly describe how to generate a random variable with any known distribution, given mainly that the computer is able to generate random numbers that are values of a random variable with uniform distribution on the interval  $[0,1]$ .

As you have already seen, in MATLAB, the function `rand` generates random numbers with a uniform distribution. The problem then is to find the transformation

$$Y = g(X) \quad (13)$$

that will create a random variable  $Y$  with PDF  $p_Y(y)$  when  $X$  has the PDF  $p_X(x)$  that is uniform. Both PDFs  $p_Y(y)$  and  $p_X(x)$  are therefore known (i.e. we know what we have in using a uniform  $X$  and we know what we want  $Y$  to be). Since  $X$  has uniform distribution,  $p_X(x) = 1$  for  $0 \leq x \leq 1$ .

Referring to Fig. 1, the probability of the event  $\{Y \leq y\}$  equals the probability of the event  $\{X \leq x\}$  because of one-to-one correspondence between  $X$  and  $Y$ , i.e.,

$$P\{Y \leq y\} = P\{X \leq x\} \quad (14)$$

$$\text{or } F_Y(y) = F_X(x) \quad (15)$$

where  $F_X(x)$  and  $F_Y(y)$  are the cumulative distribution functions (CDFs) for the random variables  $X$  and  $Y$ . For uniform distribution, with  $p_X(x) = 1$  for  $0 \leq x \leq 1$ ,

$$F_X(x) = x \quad \text{for } 0 < x < 1 \quad (16)$$

therefore (15) becomes

$$F_Y(y) = x \quad \text{for } 0 < x < 1 \quad (17)$$

from (17) it is a simple matter of taking the inverse of our CDF to get an expression for  $y$

$$y = F_Y^{-1}(x) \quad \text{for } 0 < x < 1. \quad (18)$$

The above expression is our principal result. It states that, given a specified cumulative distribution  $F_Y(y)$  for  $Y$ , we find the inverse function by solving  $F_Y(y) = x$  for  $y$ .

**Problem 1** (A theoretical exercise asking for the generation of random numbers with exponential PDF) An exponential random variable is defined by the PDF

$$p_Y(y) = \begin{cases} ae^{-ay} & y \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

where  $a > 0$  is a given constant.

(a) Show that the distribution  $F_Y(y)$  is given by

$$F_Y(y) = (1 - e^{-ay})U(y). \quad (20)$$

(b) Using  $F_Y(y)$  and (17), evaluate the relationship  $Y = g(X)$  that transforms a random variable  $X$  with uniform PDF between  $[0, 1]$  to a random variable  $Y$  with exponential PDF.

(c) Determine the mean and variance of the exponential random variable defined in (19).

**Problem 2** Using the result of **Problem 1**, generate an independent identically distributed (i.i.d.) sequence  $\{y_n\}$  of  $N = 10000$  random variables with exponential distribution. Assume  $a = 1$ . Compute the mean and variance of the generated sequence and compare with the theoretical values obtained analytically from the distribution (19). Plot the histogram of  $y$  and show how it relates to the PDF of  $y$  in (19). Note that MATLAB has a histogram function, `hist`, that you may find useful.

## 6 Uniform Quantization

The uniform quantizer is discussed in section 2.5 of the Sklar text. In uniform quantization, the quantization regions are chosen to have equal length, referred to as the *quantile interval*. Review section 2.5 of the text and focus on the implementation of a uniform quantizer.

**Problem 3** Write a Matlab function `myunfquantizer` with the following calling syntax

```
function [yq] = myunfquantizer(y,n)
```

that quantizes the input vector  $y$  with a uniform quantizer having  $L = 2^n$  levels. Take the interval of interest to be  $[0, 1]$ . The output is returned in the column vector  $yq$ .

**Problem 4** Continuing to assume that  $a = 1$ , quantize the random numbers generated in **Problem 2** with uniform quantizers having  $L = 2, 4, 8, 16, \dots, 1024$  levels for the source. You may want to scale the random numbers between the range  $[0, 1]$  by dividing with the maximum value present in you sequence before applying the uniform quantizer. After quantization, multiply with the maximum value to restore the original scale.

**Problem 5** Compute the quantization error (QE) between the original vector  $y$  and the quantized vector  $\hat{y}$  defined as

$$QE = \sum_{i=1}^N [y(i) - \hat{y}(i)]^2 \quad (21)$$

for  $L = 2, 4, 8, 16, \dots, 1024$ . Note  $\hat{y}(i) = yq$  in your case and  $i$  is the index for the samples. Plot QE as a function of  $n$ .

## 7 Nonuniform Quantization

Nonuniform quantization is a more efficient means of sampling; why do we have to quantize everything uniformly if the signals we are processing tend to stick to a certain range most of the time. It is wasteful to dedicate lots of bits to a certain range that is rarely reached by our signal. For example in speech you'll notice that smaller signal amplitudes are more common than large signal amplitudes, so why not somehow quantize small amplitudes more finely than large amplitudes.

One way to achieve nonuniform quantization while still using a linear quantizer (which is easier to build) is to first pass the input signal through a nonlinear system, a compressor, that reduces its dynamic range and then applying the input to a uniform quantizer. At the receiving end, the output is passed through the inverse of the nonlinear element used in the transmitter. Here we will use the  $\mu$ -law characteristics for the *compression*

$$z = \frac{\log(1 + \mu|y|)}{\log(1 + \mu)} \operatorname{sgn}(y) \quad (22)$$

where  $y$  is the normalized input ( $|y| \leq 1$ ),  $z$  is the nonlinear output, and  $\mu$  is a positive constant. The inverse of the  $\mu$ -law nonlinearity is the *expansion*

$$y = \frac{(1 + \mu)^{|z|} - 1}{\mu} \operatorname{sgn}(z) \quad (23)$$

By compression, then linear quantization and then expansion you achieve a quantization method called *companding*.

**Problem 6** Derive equation (23) from (22).

**Problem 7** Plot the input-output characteristics of the nonuniform quantizer obtained from (22) using  $\mu = 255$ .

- Problem 8** Repeat **Problem 4** for the random variables  $y$  that have been compressed using (22). In other words, generate random variables  $y$  with the exponential distribution, compress them using (22), and then use uniform quantization to quantize them. Compare your results with the results of **Problem 4** and comment on the differences.
- Problem 9** List the first 32 samples of the random variable  $y$  with exponential PDF, the output of the uniform quantizer in **Problem 4**, and the output of **Problem 8**. Make sure that you use the same ensembles of  $y$  in each case. Comment on your results.
- Problem 10** Using equation (23), uncompress your output. Compare the first 32 samples with the first 32 samples of the output of the uniform quantizer in **Problem 4**.
- Problem 11** As in **Problem 5** compute and compare the quantization errors for the compressed and linear schemes for  $L = 2, 4, 8, 16, \dots, 1024$ . As before, you should use more than 32 samples in order to get a better comparison.
- Problem 12** Repeat **Problems 8–11** for two values of  $\mu$  (other than 255) in the range  $0 \leq \mu \leq 255$ .