Digital Communication
Prof. Bikash Kumar Dey
Department of Electrical Engineering.
Indian Institute of Technology, Bombay

Lecture - 22
Probability of Error Calculation for Digital Modulation
(Part-1)

We have been discussing digital modulation techniques so, far we have discussed what are the different module digital modulation techniques which are used in practice PAM QAM PSK FSK PPM all these techniques we have discussed. Also we have classified digital modulation schemes. And discussed different types of modulation schemes like: orthogonal modulation, biorthogonal modulation then simplex signal set.

And then we have discussed different type of receiver structures for first we have discussed also digital different, generic receiver structures for any given signal set matched filter receiver correlation receiver. We have discussed matched filter in detail we have shown that it is optimum. We have shown that sampling at multiples of capital T seconds is optimal. And in the last class, we have discussed I diagrams which is a way of seeing how much i psi is there on the oscilloscope.

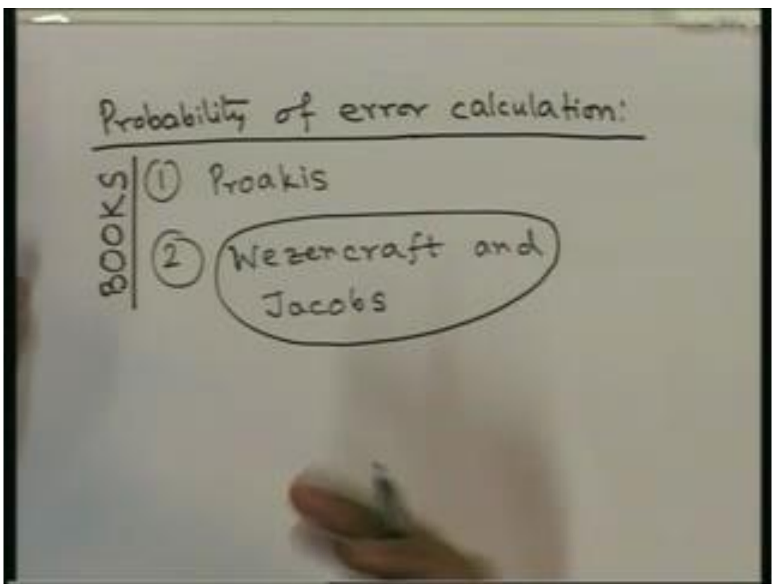
Then we have discussed mapper in detail and we seen that by using grey coding we can, we can achieve probability of bit errors almost equal to probability of symbol errors. So, for many modulation schemes, because most of the times the symbol errors happen because of detection of neighboring points of the transmitted point. So, the grey coding ensures difference of only one bit between neighboring points. And thus most of the symbol errors will result in most only 1 bit error with grey coding.

In this class, we will and then also in the last class we have this seen the relevance of discrete memory channel and also a real vector channel discrete time even for continuous time channels, even for communication systems with continuous time channel. In this class we will start probability of error calculation for different kind of signal sets whenever, there is it is not easily possible to compute the probability of error.

We will also see ways of findings an upper bound of the probability of error upper bound of probability of error will also be useful because then we can guarantee that, the probability of error for the system we are designing will be less than so much. So, less than 10 to the power minus 5 so, that is, that is nice. So, lower bound does not make sense it does not it is not useful upper bound of the probability of error is useful. So, if you say probability of error is greater than 10 to the power a minus 5 then it can be even 0.1 it can be 0.2.

So, anything may greater than 10 to the power minus 5. So, that is not useful, but if you have an upper bound, if we say that less than 10 to the power minus 5 then that is nice because we know that, out of 10 to the power 5 bits about 1 bit will go wrong. So, for this class and for the next class we will discuss probability of error calculation and for this topic I will advise you to read out of these 2 books.

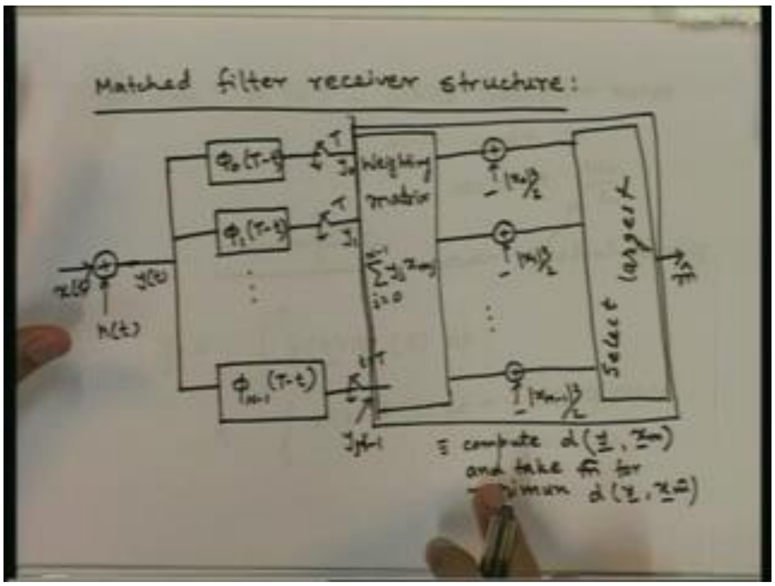
(Refer Slide Time: 04:45)



Proakis digital communication and Wezencraft and Jacob I will, specially recommend this book for this topic specially, for the systematic treatment of. So, treatment for a generic receiver structure and also the discussion of signal set in terms of vector spaces in a very systematic manner please read this book. And also probability of error calculation, which we are going to do for this, in this class and the next class this book will be very nice.

And in fact the approach of vector space approach for signal design was introduced for the first time in this book. So, is a very classic book on this topic. So digital communication, by Wezencraft and Jacobs. So, let us start, let us see how we can compute the probability of error for different signal sets? So, we have seen that so, we have seen that a generic matched filter receiver structure is this.

(Refer Slide Time: 06:02)

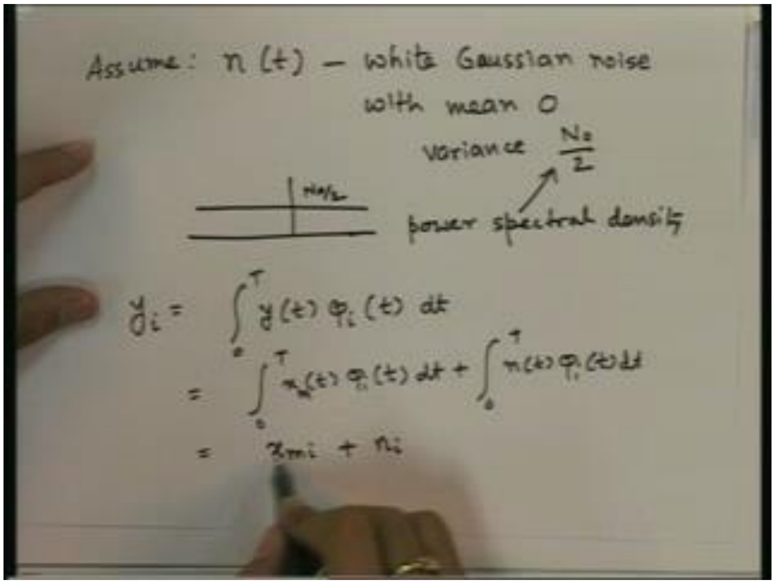


Probability of error will certainly depend on how we do the detection. So, we have to keep the detection in mind how we have implemented the receiver to compute the probability of error. If we do the detection in the not optimum way then; obviously, the probability of error will go up. So, we are going to analyze the probability of error which is minimum that, is the best receiver structures will give that probability of error.

So, matched filter is an optimum matched filter receiver is an optimum receiver. So, we are going to analyze this receiver. So, we are seen that, this channel output yt is fade to 2 matched a bank of matched filters matched to the basis signals, orthonormal basis signals. So, we will sample at capital T for 1 for the first symbol and then again at multiple sub T. So, at these instances this is yt and then we have this weighting matrix this basically computes j equal to 0 to n minus 1 yj xmj for the m’th output.

So, here we have and out of this, we will select the largest which will give us the estimate of m is the receiver structure. The channel is assumed to be ideal AWGN channel which adds only noise, this the receiver structure for generic signal set in msnl signal set.

(Refer Slide Time: 10:14)



Now, we will assume as we have done before that nt is white Gaussian noise with mean 0 and variance N naught by 2 why it is written in this form because it is the variance of the noise with power spectral density N naught by 2. If you draw the power spectral density of the noise, it will look like this with N naught by 2 and what is the a this is the y y1, y2 and so, on y0 y1 ym minus n minus 1.

Now, we will see what is yi? You have seen it before already. So, what is yi? Yi is the output of the filter at capital T and it can be written as yt phi I t dt. And this is equal to xt phi it where yt is xt plus nt plus nt phi it dt. We have also seen before that this part, this part this block is equivalent to equivalent to equivalent to finding the distance of compute distance of the vector y with different xm vectors the signal vectors and take m hat for minimum d.

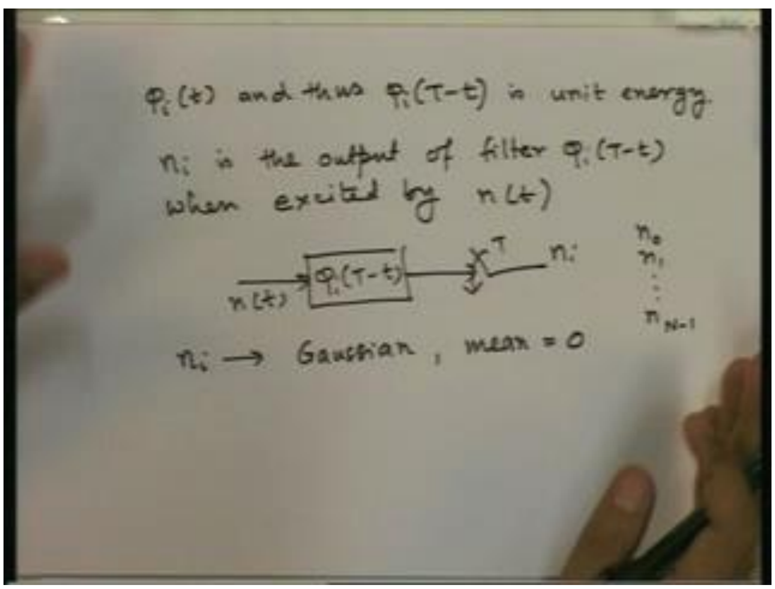
So, that is we originally actually had this as the receiver structure. Then we have seen that computing the distance and finding the minimum distance that is the picking of the minimum the nearest signal point from this vector y is equivalent to performing these operations. So, to find the probability of error we can assume that we are doing this because the result will be same.

So, this is xmi suppose, xm was transmitted then we will have m here. Then xmi i'th component when we do this plus this will be the noise in fact we have seen before that, if we take an ortho normal basis and take a signal in that space then you can compute the component i'th component of this vector signal by it simply taking the inner product of xm with phi i.

Now we have this, now this is the 1 ith component that was transmitted, but this is a noise which is which has corrupted this. Now we need to characterize we need to find out, what is the distribution of ni to know the probability of errors. So, we need to know how often this will corrupt it. So, much that, these noise component ni(s) n naught to n capital N minus 1 components will corrupt the received vector the transmitted vector x so, much that we will detect wrongly.

So, to do that, let us see what ni? What kind of noise ni is? What kind of random variable ni is?

(Refer Slide Time: 15:57)



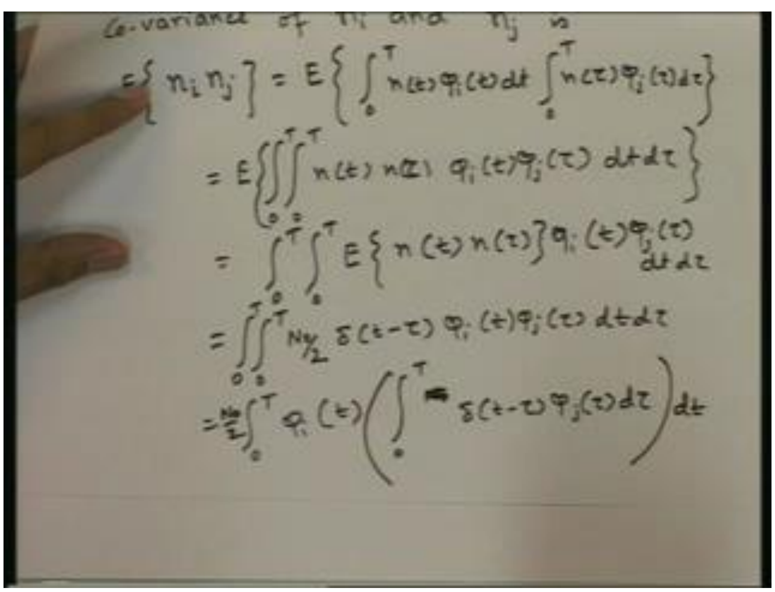
So, this let us see, Phi it first of all we know that is orthonormal and. So, that is it is no doubt yeah so, it is unit energy we know. So, phi I T minus t is also unit energy because it is nothing but flipped versio n of phi it. So, this is unit energy and ni, is the output of this filter with impulse response. When the input is when excited by nt. So, can see here that, nt is the inner product a ni is the inner product of nt and phi it. So, it will be the output of the filter with impulse response phi I T minus t when the input is nt this is also the convolution of nt with phi I T minus t at T time.

So, ni is the output of the filters. So, that is this filter is there phi T minus t there is yt no this is nt and this output at T is ni. We have seen it here also. So, because this yt is a xt plus nt. So, nt part it goes through here and that capital T it is the ni component. So, this is the, this is a filter with unit energy impulses response nt is white Gaussian noise with 0 mean and variance n naught by 2.

So, we need to characterize what is ni? What is the distribution of ni? What is the mean of ni? What is the variance of ni. First of all ni is known that, it is known that if nt is white Gaussian noise, if it is Gaussian ni is also Gaussian. So, it first of all ni is Gaussian. And because nt is 0 mean ni is also 0 mean and variance we will compute, but we should compute a not only compute variance we will compute the covariance between any 2 of these noise components n naught 2 n n capital N minus 1.

If you take any 2 noise components we will compute the covariance between them. So, that will give us variance of ni if, you take covariance of with itself and covariance of different noise component pairs also. So, let us compute the covariance. What is covariance?

(Refer Slide Time: 19:22)

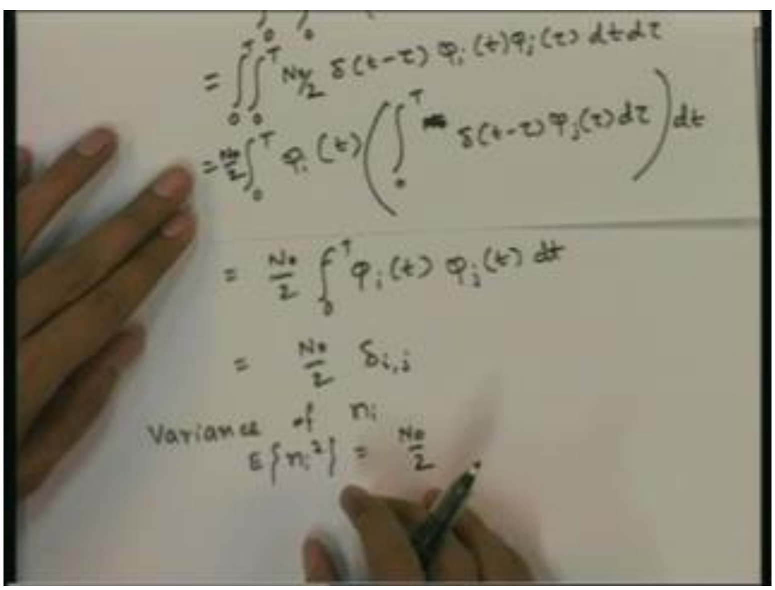


Covariance of ni and nj inj may be same or different is: it is we know that covariance of for 2 random variables is given by this because these are 0 mean: we need not subtract the means: this is 0 to T nt phi it dt 0 to T n tau phi j tau d tau. We are just taking 2 different variables in these 2 integrals because we are going to combine these 2 integrals into 1 double integral.

So, this is expectation of 0 to T 0 to T nt n tau phi it phi j tau dt d tau expectation of this whole thing. Now, we can take the expectation inside and these are fixed functions. So, we do not these can be taken outside the expectation we have inside the expectation only n T and n tau phi it phi j tau dt d tau. Now, what is the expectation of nt and n tau product? This is the autocorrelation function of a white Gaussian noise at t minus tau.

So, because it is white we will with power spectral density n naught by 2 we know that this result is nothing but naught by 2 n naught by 2 delta t minus tau and phi it phi j tau dt d tau. Now let us write, this in a little different way. We take the t parts separately outside phi it this is independent of tau and then do integration on tau n naught by 2 is constant. So, n naught by 2 outside then delta t minus tau phi j tau d tau dt. Now this is equal to n naught by 2 0 to T phi it

(Refer Slide Time: 22:20)

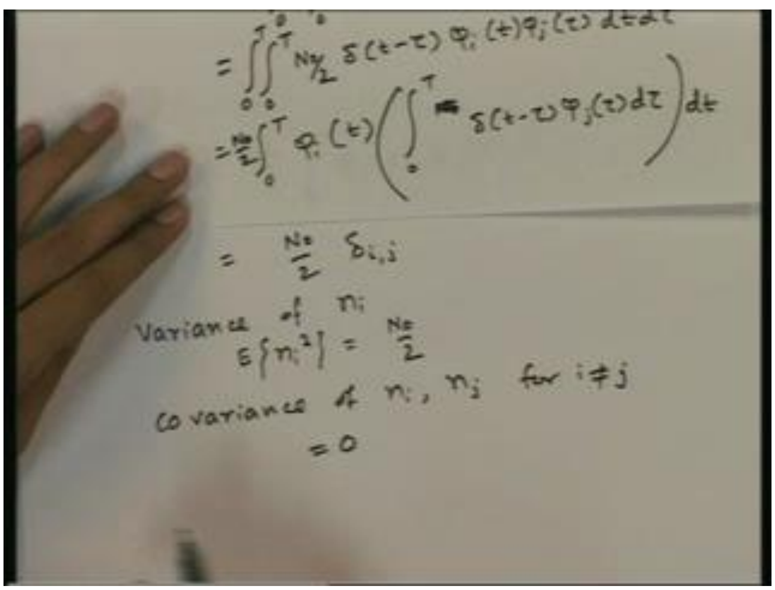


Now, what is this integration? This is a delta function shifted by t flipped and shifted by 2 flipped version is same Delta t minus tau is same as delta tau minus t. So, delta tau minus t minus t times phi j tau integrated. So, it will be nothing but phi j of t this whole integration that is the way. In fact, that is a part of the definition of the delta function.

This is delta j t dt and this is equal to n naught by 2 and what is this integration? We know that these functions are orthonormal. So, if i is equal to j this integration is 1, if they are not equal then the integration is 0. So, in other words this is the Kronecker delta. If I is equal to j this delta is 1, if I is not equal to j this delta is 0 this is called the Kronecker delta function.

This is different from Dirac function well Dirac delta is not a function first of all, but any we loosely call them function. Delta this delta is Dirac delta which is continuous whereas, this delta is discrete for it is valid only on discrete values of inj. So, this is the thing. So, what is the variance of ni? For any I it is expectation ni square and that is; that means, ni times I that is I equal to j. So, for I is equal j is 1. So, it is n naught by 2.

(Refer Slide Time: 24:26)



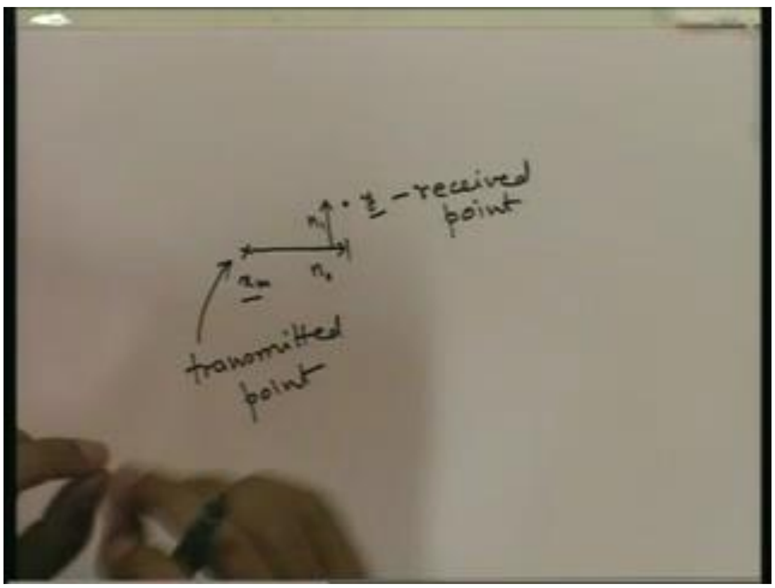
So, this is the variance and covariance of ni and nj for I naught equal to j is 0. So; that means, these noise components are uncorrelated and we all we already know that they are Gaussian. So, if you have 2 Gaussian random variables they are uncorrelated then we also know that, they are independent because we know the joint distribution of 2 Gaussian random variables there is some correlation coefficient which comes in the distribution.

If you make that 0 we can see that, if can be the the whole distribution probability density joint density can be broken into 2 parts as product of 2 different densities

So, is the density of ni times density of nj. So, they are independent, if they are uncorrelated. It is not true for any other distributions was not all distributions, but for Gaussian it is true. So, here we have seen that the noise components that, we are getting N naught 2 n N minus 1 as part of this vector y naught 2 N minus 1 are independent Gaussian with variance 0 mean and variance N naught by 2. So, now what are the? What is the noise doing to the points we are transmitting?

Suppose, we transmitting a vector x xm and we are receiving y it will not be for exactly xm, but it will be somewhere else where will it be that, depends on the noise vector.

(Refer Slide Time: 26:18)



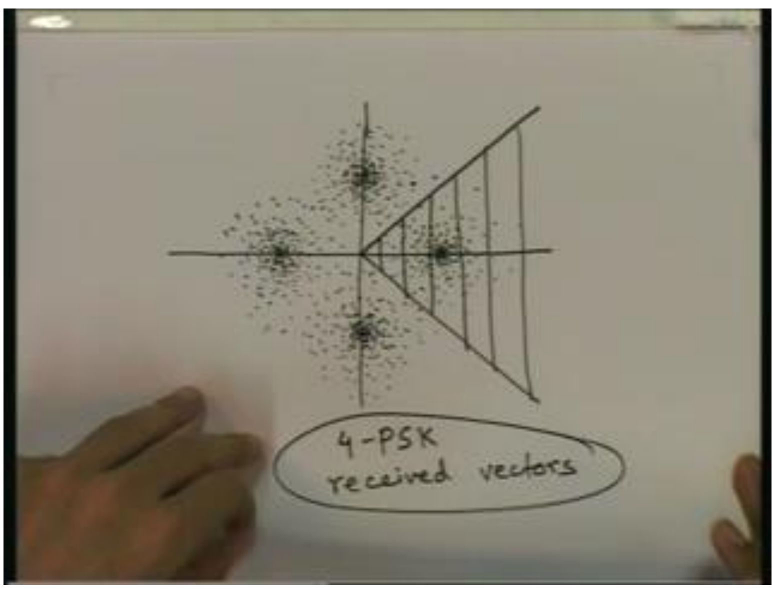
So, if this is a, this is the point xm we are transmitting we have transmitted. We will receive somewhere probably here which is, which is the received vector y and this vector is xm plus the noise vectors that is n naught n1. We are considering 2 dimensional. So, n naught n1. So, n naught will be added to the x1 the first component x naught then the second component will be added with n1.

So, if this is the received point then from here to here this vector is n naught. Similarly, this vector is n 1 this much and this is n naught. So, this is the received point, this is the transmitted point. So, noise deflects the noise vector deflects the transmitted vectors. So, this way, if we take the, if we take, if we transmit a sequence of symbols symbol after symbol every T seconds you transmit 1 symbol and at the receiver you extract the vectors y for different symbols and.

Suppose, you plot those points, in the vector space. So, because we can plot nicely for 2 dimensional we can plot for 2 dimensional vector spaces. Let us consider only 2 dimensional vector space for timing. Suppose, we plot all for every symbol you extract that vector you generate that, vector using matched filter and sampler and then you plot that vector on the 2 dimensional plane the you do that for all symbols you keep doing that, then there will be more and more points.

If for every symbol you have 1 points somewhere on the plane and what will be ultimate look of the plane plot. So, it will some this like this, if you use 4 PSK.

(Refer Slide Time: 28:26)



So, if you use 4 PSK received points will be like this received vectors will be this. We will transmit either this or this or this or this sometimes this sometimes this sometimes this or sometimes this these points. Now, when we transmit this point the noise at different symbol interval will be different. So, noise here sometimes it will be here it will deflect the point to here sometimes it will deflect the point to here sometimes here sometimes here and so, on.

But because the noise is Gaussian the probability that the noise is of smaller value is more. So, most of the times, the noise value will be small. So, the received points we transmit only this point always. The received points will be more points will be near this then few more points then again even fewer points further.

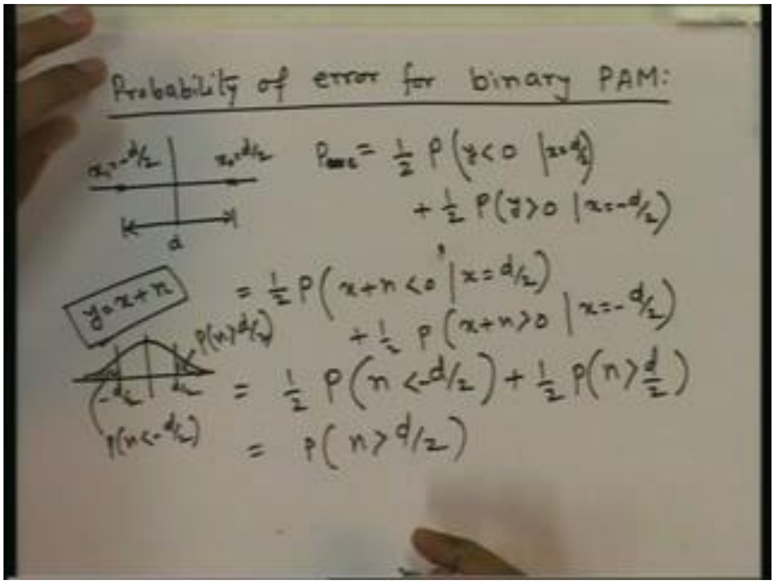
So, as you go further there will less and less points that is seen in this diagram this plot because you can see as you go further from this point the points are becoming less dense. Density of the points is decreasing that is because the noise a value is most of the times small. So, similarly if you when you transmit this points you will receive this points they will scattered towards this point and similarly for these points.

Now, it may happen thus when you transmit these point sometimes very rarely the points go very far from her and somewhere here or somewhere here. Then we will make a discussion error at the receiver because this is nearer up to this point and we will think that this was transmitted. So, there will be decision errors whenever, the point goes very far and when exactly will be the errors. Suppose, we transmit this then the decision region for this point we have discussed already before is this region this part this region.

So, whenever the noise is so, much that you transmitted this and you received somewhere outside this region, then there will be decision error. If it is on this side you will think this was transmitted. If it on this side you will think this is transmitted. If it is on this side you will think it is on this 1 was transmitted.

So, there will be decision error, if the received point is met not in this region. So, the points will look like. Now, we will start now computing probability of error for different modulation techniques. Let us compute for first for the simplest modulation technique that is PAM.

(Refer Slide Time: 31:14)



So, probability of error probability of error, for first binary PAM the simplest, binary PAM. So, we are transmitting suppose, we transmit these 2 1 of these 2 points suppose this distance is d. So, this is basically x naught equal to d by 2 and x1 is minus d by 2. We will receive somewhere here. If you transmit this you will receive somewhere here nearby some, if you transmit this you will receive some point somewhere nearby. Now what is the average probability of error? P average probability of average probability of error.

Let us say Pe average. This probability is see half of the times we assume that, these 2 points are transmitted with equal frequency then half of the time this is transmitted half of the time this is transmitted. Then the probability of error will be half times the probability of error when you transmit this and half times the probability of error when you transmit this the average of the 2 probability of errors.

So, half probability of errors when you transmit this that is: if you transmit x equal to x naught that is d by 2. If you transmit d by 2 value when will the error it will the error will when the received vector is less than 0 it is negative then you will detect this and the as a result there will be error. So, the probability of error when this is transmitted is nothing but, probability of y less than 0 when x is equal to d by 2 y is equal to x plus noise remember that.

So, and plus half times the probability of error when this is transmitted. So, probability of errors when x is minus d by 2 given x is minus d by 2. So, error is error will happen when? y is greater than 0 for this case. So, y is greater than 0. Now, this is half P as you said y is x plus n. So, y is less than 0 means x plus n is less than 0, but x is equal to d by 2 for this case.

So, x plus n less than 0 given x equal to d by 2 plus half probability x plus n greater than 0 when x is minus d by 2. Now, if the x value is d by 2 when will this happen x plus n is less than 0 when n is less than minus d by 2 x is d by 2. So, you take d by 2 on this side n less than minus d by 2. So, this is half probability n less than d minus d by 2 then, if x is minus d by 2 what is the probability of this?

The probability of this the probability of n being a greater than d by 2 probability n greater than d by 2.

Now, you can see the noise is n is Gaussian it has 1 component n naught only. So, we are denoting it by n itself N probability of n, n has Gaussian distribution with 0 mean. So, the probability that n is greater than say this d by 2 or less than minus d by 2 that is the area here has probability of n greater than d by 2. Area here is probability of n less than minus d by 2. This area is probability n greater than d by 2. Both are same because this is noise is symmetric.

So, for symmetric noise, if you have a such a symmetric signal set probability of error for this will be same as probability of error for this as you have seen here.

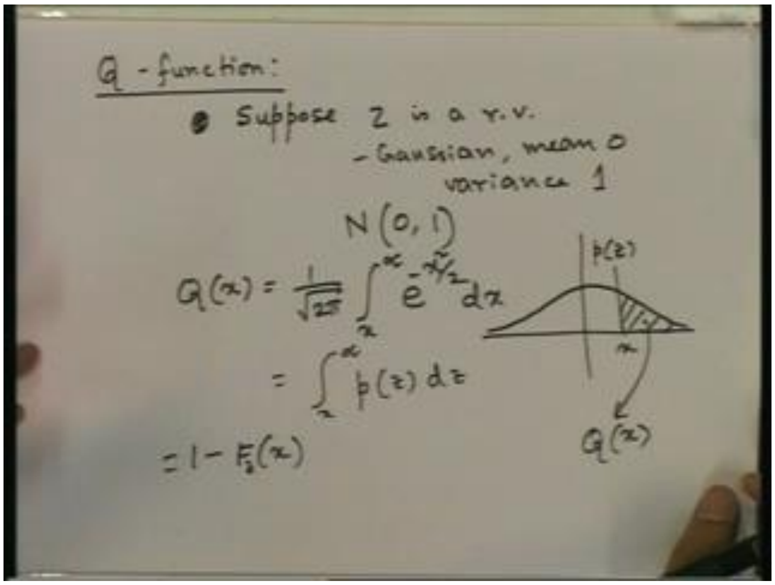
This is same as this or this will be same as this. Now so, we can write this as any of this because half times this plus times the same thing. This is same as this. So, this is probability of we can write n greater than d by 2 only this term. So, now what is this?

Now this is difficult to actually compute in closed form this is actually the integration of the probability density function. This is the integration of d by 2 to infinity 1 by root over 2 pi then root over a n naught by 2 and e to the power minus. So, minus x square by 2 sigma square that is, n naught by 2 again 2 2 cancels 2 2 cancels dx. So, this is this is probability this is the area here we are integrating this curve from here d by 2 to infinity.

So, this needs to be computed, but this is no closed from computation for this. And there is a function defined which has which is implemented which is, which is there as sub subroutines in

MATLAB which is used very much which will enable you to compute this kind of probability of errors. The function is as following the function is called Q function.

(Refer Slide Time: 37:19)

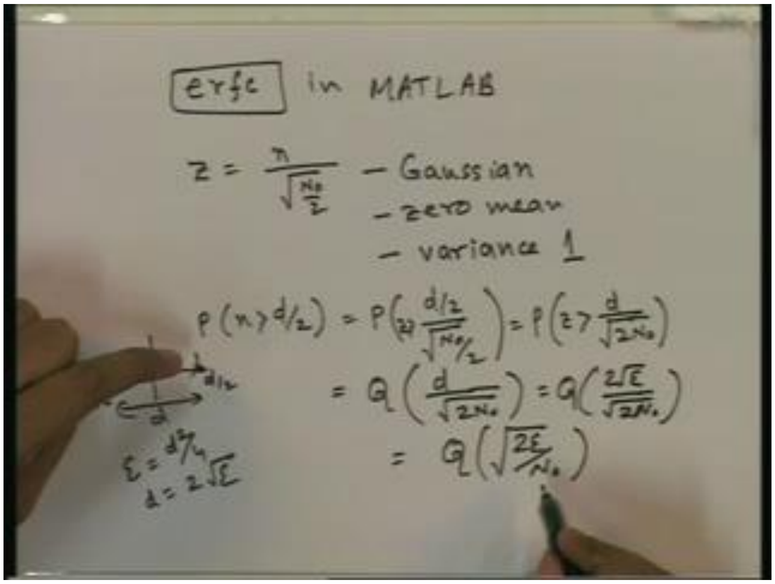


It is defined to be defined in the following way. Suppose, Z is a random variable which is Gaussian mean 0 variance 1 such random variable is denoted by N 0 1 N for normal you know that, Gaussian distribution is also called normal distribution. So, no N for normal 0 mean variance 1. So, variance n naught by 2 which we had for this n, this n random variable is denoted by N 0 n naught by 2 the variance n naught by 2 mean 0.

So, suppose we have a random variable this 0 mean 1 variance Gaussian then Qx is denoted a is defined to be root is x to infinity e to the power minus x square by 2 dx. Basically, the integration of this x to infinity P of z dz. So, if you plot the probability density function of z is Gaussian 0 mean variance 1 take x anywhere this area is called the Q. This is Pz this is Q of x. Q of x is defined as the probability of unit variance 0 mean Gaussian random variable taking value greater than x.

So, this can also be written as 1 minus the cumulative distribution function of z at x that is 1 minus F of Fz of x the random variable z at x. So, this is, Q function and let us now see, what we need to compute and try to express the Q function. So, Q function cannot be really computed in closed form numerically then integration can be found and. So, routines for programs for finding this integration for different value of x is available in MATLAB not directly Q function, but some other function called erfc from which you can now derive Q function. So, please look at erfc function in MATLAB and try to express Qx in terms of erfc of x.

(Refer Slide Time: 40:24)



So, erfc e r f c function, in MATLAB because later we will give, will I will give exercise to do in MATLAB and there you will need this erfc function. Now, we were trying to find out the probability of the error for binary signal set and we have come to this level that, it is probability n greater than d by 2 where n is a Gaussian random variable with 0 mean variance n naught by 2.

Now, let us derive another random variable from n that, random variable z is n by root over n naught by 2. We are just scaling this random variable n by this scalar. Now, this is a just scaling of a Gaussian random variables. So, this is also Gaussian mean of n is 0. So, 0 mean: variance of n is N naught by 2 that is standard derivation of n is root over N naught by 2.

So, the standard derivation of z will be standard derivation of n by this. So, it will be 1. So, variance is also variance is also 1. So, we have derived a random variable z from n which is 0 mean Gaussian with variance 1 which was used to define the Q function you remember the Q function was defined with respect to density of such a random variable.

So, now the integral we want to the probability we want to find is probability n greater than d by 2. N is greater than d by 2, if and only if z is z is if n is greater than d by 2 z is greater than d by 2 by this. So, it is d by 2 by root over N naught by z is greater than this, that is P z is greater than d by root over 2 N naught. So, now this is of this form the definition of Q function, if you see is that Q of x is defined to be probability z greater than x this is also probability z greater than x that is; Q of x.

So, here z is same random variable that is 0 mean Gaussian with radiance 1. So, probability of z greater than d by this is nothing but, Q of d by root over 2 N naught. So, this can be computed in MATLAB or table for this also exist Q function table. So, we can find this value and then find out the value of Q at that point. Now, now we have expressed this in terms of d the distance between the points this is d.

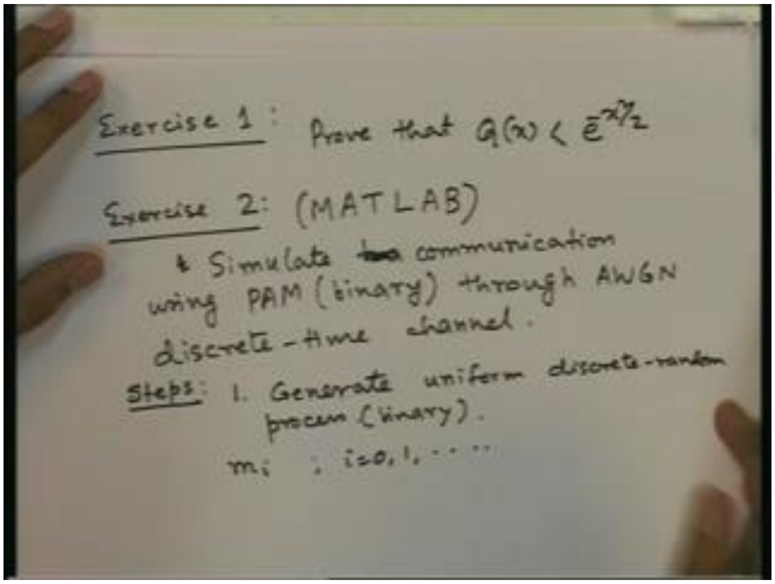
Now, if we want to express in terms of the energy of the point because we know that, increasing the energy will energy is something really you can compute the energy. It is same as the energy of the signals. So, in terms of energy can you express it. If we increase energy we should be able to decrease the probability of error and vice versa.

So, in terms of energy we will express the probability of error, but that is very simple because the energy of average energy E is nothing, but energy of one point if energy of the both the points are same because they are same distance from 0 the distance from 0 is same. So, this d by 2 point and this is minus d by 2. The energy of both the points is d square by 4. So, d is 2 root E.

So, here we can put in place of d 2 root E and then Q of 2 root E by root over 2 N naught that is: Q of root over 2 E by N naught. So, we have expressed the probability of error for binary PAM in terms of the average energy that is transmitted and the noise power spectral density. Now, we will see some a we will see some exercise.

Exercise 1, I will advise that you please solve this exercise as soon as possible because this will also make these ideas clear in your mind. There are far many more exercises are in Wezencraft and Jacob and I encourage you to try to solve those problems because then you will be able to understand the next class better.

(Refer Slide Time: 45:22)



So, exercise 1 is prove that, Qx is less than e to the power minus x square by 2. So, this is an upper bound of Qx and using this upper bound you can get an upper bound of this probability and such probability is which we will compute later also probability for a different like QAM signals will also be expressed in terms of Q functions.

Now, if we are not, if we are not given MATLAB, if you do not want to do 2 numerical integration we can at least approximate Qx we can upper bound Qx by this and then we will get some closed from expression which will be the upper bound of the probability of error. So, this is use useful that way. Then exercise 2, exercise 2 is using MATLAB please simulate a PAM communication system based on PAM using MATLAB that is the exercise.

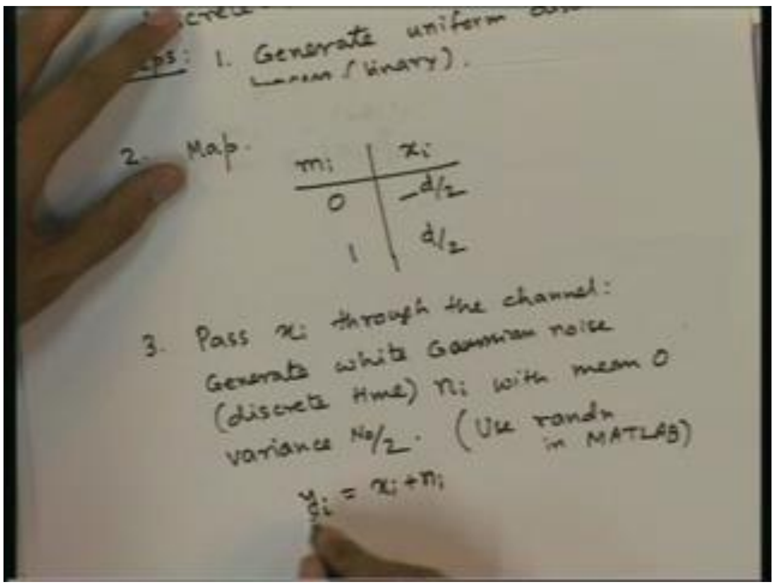
The details I will work out now first step is. So, the idea is to exercise is to simulate transmission or communication transmission and reception communication using PAM another binary PAM through AWGN discrete time. So, remember the relevance of discrete time channel model in the context of continuous time channel. If you take the modulator block and matched filter and sampler block at the receiver together with the channel they form a discrete time channel and I am not asking you to simulate the inside just assume a discrete time channel.

So, do not go into continuous time just take the discrete time channel and simulate the outside blocks. I will now, explain step by step what is to be done?

First the steps are first generate the generate the information you want to transmit generate uniform, the distribution is uniform, discrete random process, discrete binary random process, because the values will be binary meaning by. So, binary meaning by generate a sequence of bits mi randomly.

So, mi is o the I'th bit that is the massage 1 bit is 1 symbol I equal to 0 1 and so, on till say 1000 or 1,00,000 you can take. Then each mi I said is from 0 and 1 that is it. is a bit value is either 0 or 1 next step, next step is, next step is do mapping.

(Refer Slide Time: 49:26)

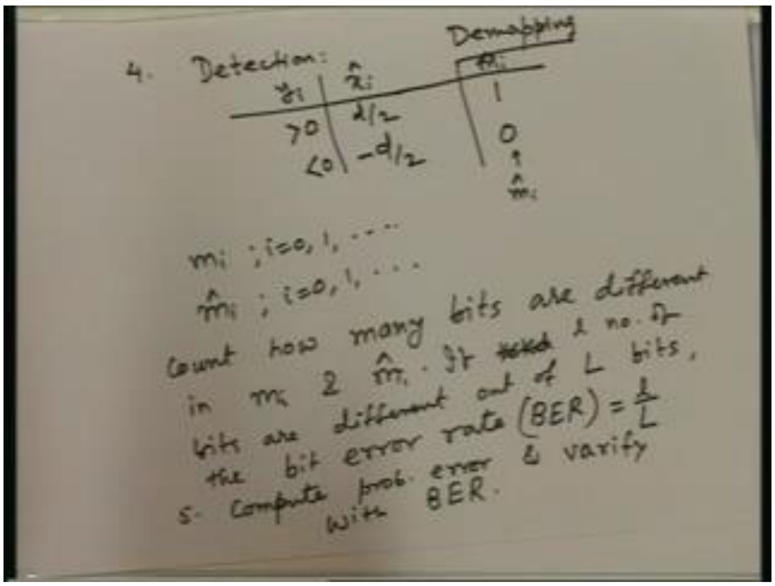


Mapping of 0 1 to some level map. So, you mi from there you consider another variable another sequence of variables there is a random process. If mi is 0 for any I if the mi the bit is 0 take xi equal to minus d by 2. If it is 1 take d by 2 that is the binary PAM mapping then we want to pass xi through the channel discrete time channel inside, which all the modulation and as matched filter all these are there they are inside the channel.

So, we will not simulate the channel, but we will simulate only the output that as the whole channel as a black box. So, how do we generate the output of the channel. So, to generate that, we have to generate the first the noise process. So, generate first generate a white Gaussian noise, discrete time, white Gaussian noise ni with mean 0 variance N naught by 2. So, once we have generated ni. So, you can generate using randn function use randn in MATLAB.

So, using this function you can generate a random sequence of random numbers with distribution, Gaussian distribution mean 0 variance N naught by 2. So, once you generate that, you already have the sequence xi x naught x1 and. So, on till x say 1,00,000. So, add for each I add xi with ni and that is your yi that is the received vector xi plus ni. So, you will get y naught y1 and so, on that you have got by adding the noise sequence to the transmitted sequence then at the receiver what do we do the fourth step is the detection.

(Refer Slide Time: 52:18)



Detection means: detecting which value was transmitted, which value of xi was transmitted from yi. So, if you have now the mapping the the detection will be we know that for binary PAM this is the estimate of xi hat. If is greater if yi is greater than 0 take it to be d by 2. If, it is less than 0 take it to be minus d by 2. And then you can subsequently do demapping from here for yi greater than 0 xi hat is d by 2 and mi is 1 for minus d by that is less than 0 yi is 0.

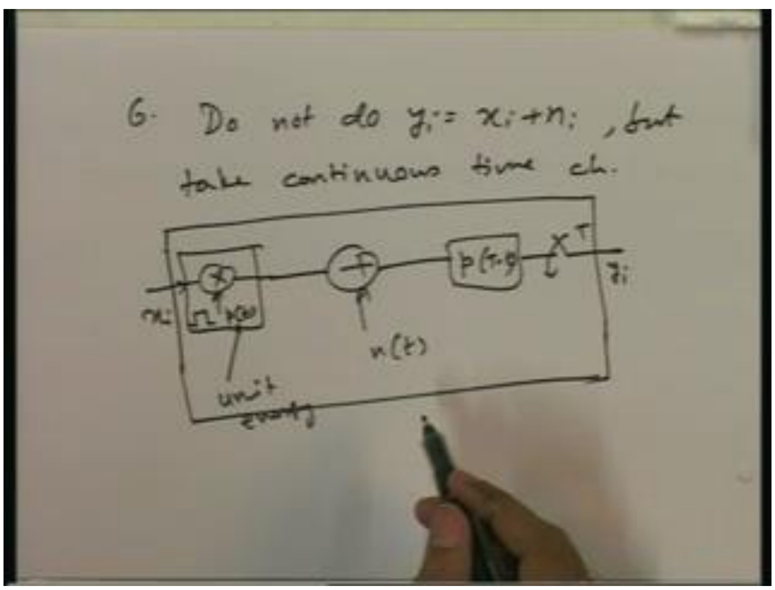
So, you have got mi hat, this is not mi hat this is the estimate of mi hat at the receiver this is the mi hat. Now, count you have the sequence mi I equal to 0 1 and. so, on you have the estimated sequence mi hat at the receiver for I equal to 0 1 and so, on see the difference between these 2 sequences see how many bits are different in these sequence 2 sequences.

So, count how many bits are different in mi and mi hat. If total, If say a if n, if a say l number of bits are different that is: if l number bits were received wrongly out of say L bits the out L bits l bits are received wrongly. So, the bit error rate also called BER in short in most of the literature BER is l by L this is the estimate. If, you should take sufficiently large L to get a clear estimate of the bit error rate.

And bit rate you should compare with the probability of error that should they should be ideally same, but they should be very near for large L. If you take L equal to 10 to the power 10 you will have almost same the bit error rate will be same as the probability of error you have computed using the formula using Q function.

So, compute the compute probability of error and verify with BER they should be almost same. Next step of the exercise is to simulate the continuous time channel itself take xi modulated with a pulse pt and then received pt and through pass it through matched filter and then get yit instead of just generating a noise sequence discrete noise sequence and adding with xi to get yt yi you implement the inside of the of thing also inside of the discrete channel also in terms of the continuous time channel.

(Refer Slide Time: 56:15)



So, that also can be done. So, that is the next step. sixth do not do yi equal to xi plus ni, but take continuous time channel that is: now nt this is the channel. Here you have modulation this is xi it is multiplied by pulse pt. Pt you should take you need energy to verify with your probability of error. This you should take unit energy pt then pass through matched filter and then sample at T or multiples of capital T and then get a get yi. So, previously you just did this as a black box

Now, implement these also. So, this will this way you will able to verify whether the formula using which you are computing the probability of error in MATLAB will actually whether, it gives you the correct a really a an impression of how many bits will go wrong in practice, if you use that modulation scheme and that kind of receiver system.

So, in the next class, we will continue discussion on probability of error for other modulation techniques and whenever, not possible to compute the probability of error easily we will try to see, if can get an upper bound on the probability of error in an easy manner. See you in the next class.

Thank you.