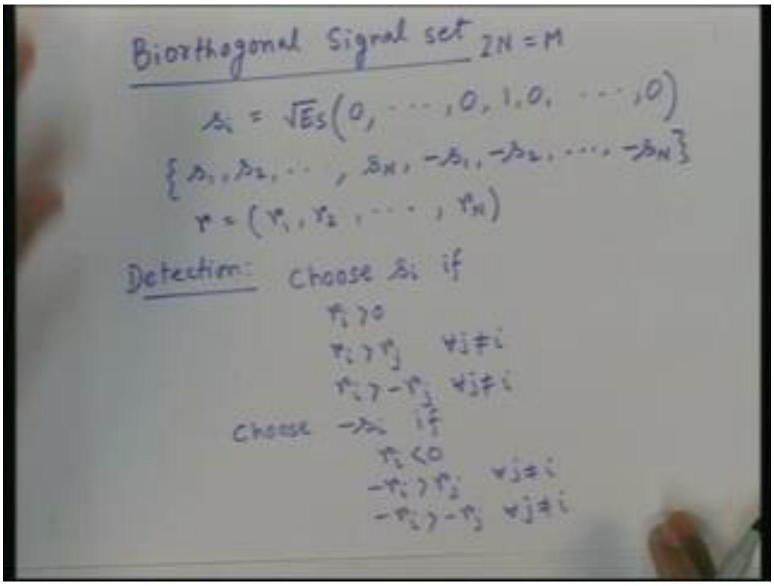
Digital Communication
Prof. Dr. Bikash Kumar Dey
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Lecture No. #2 4
Calculation of Probability of Error and Modulation with Memory (Part - 3)

Hello everyone, in the last class we have discussed probability of error calculation for QM constellation and also for orthogonal signal sets like: FSK, PPM etcetera. In this class we will continue our discussion on probability of error calculation and see how to compute probability of error for bi-orthogonal signal set. And then, we will also see how to compute an upper bound on probability of error for some signal set for which, computing the exact probability of error is not so simple like: orthogonal and bi-orthogonal signal sets.

For orthogonal signal sets we have already seen that the that the expression is too complicated to be computed in closed form. So, for that kind of signal sets only way to compute the probability of error is by numerical integration. And for such signal sets often we can compute a reasonably good upper bound on the probability of error. So, that will give us also some idea about how good the signal set is. So, let us see how to compute the probability of error probability of error for bi-orthogonal signal set first.

(Refer Slide Time: 02:27)



So, here for the bi-orthogonal signal set we have some N N number of signal points of this type and then so this has, so you know that the dimension is N and number of points is 2 times N. So, these there are N number of components and the i'th signal point for I equal to 1 to capital N is this. And then we the whole signal set is s1 s2 to s N then, the negative of that minus s1 minus s2 till minus sN.

And how do we detect we know that the ML detection can be done by seeing the distance of the received, vector from all the points. But, when we try to look at the condition carefully for the bi-orthogonal signal set we will see that a simple rule comes out from that condition. And that is we simply take the received vector r which has N components r1, r2, rN. Then, see which of these components has maximum absolute value.

That is if it is negative we take the, we take the negative of that to make it positive. We make all the components positive and then take that component which is the maximum. And then if that component is positive then, we take suppose ri is maximum the absolute value of ri is maximum. Then if ri is positive we take si we detect si if ri is negative we take minus si. So, that is the simple detection rule and that is the ML detection rule.

So, what is the detection rule we can say that the detection rule is that choose si if. So, si you will be chosen if. So, remember that si has all the components zeros and only the i'th component 1. So, si will be chosen if if ri the ith component is greater than 0. If it not greater than 0; obviously, the point is nearer to minus si than si. So, there is no reason to choose si at all because a minus si itself it is a better signal point nearer to the received vector. So, ri has to be greater than 0.

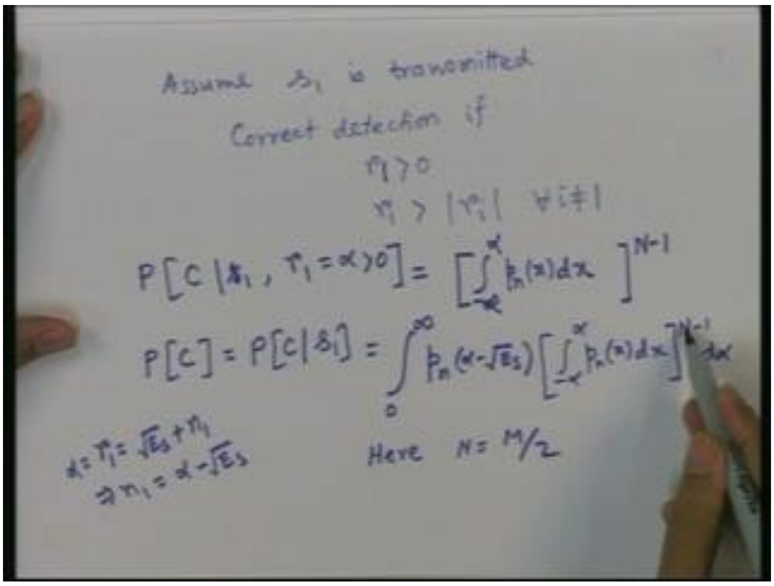
So, if ri is not greater than 0 also the others components of the received vector should be less than these value because, otherwise some other signal point is nearer to the received vector. So; that means, ri is greater than rj for all j not equal to I and also it should be greater than the negative of rj that is, they are should not be any other component whose absolute value is greater than ri. Even though that value is negative and it is less as a result it is less than ri where, its absolute value itself should not be greater then ri.

Then if it is so then, if it is not if it is if the value is negative and its absolute value is greater than ri then minus of that corresponding signal is a better choice. So, ri should be greater than minus rj also for all j not equal to i. Similarly, we should choose minus si as the signal point if ri is less than 0 and then minus ri remember ri is less than 0. So, minus ri is positive and it should be greater than rj and minus ri should be greater than minus rj also.

So, these two conditions together here actually means that ri is greater than the absolute value of rj; these two together means. And here it means the minus ri is greater than absolute value of rj. So, this for all j not equal to i, this for all j not equal to i. So, this is the detection criteria. Now, like the orthogonal case this bi-orthogonal signal set is also a symmetric signal set. So, as a result the probability of error for all the signal points will be same.

So, it is sufficient to compute the probability of error assuming that 1 of them is transmitted. That is, just compute the probability of errors for 1 of the signal points and that is going to be the probability of error for all the signal points. As a result that is going to be a average probability of error of the signal set. So, we will compute the probability of error for 1 signal point that is s 1. So, we will assume that s1 is transmitted and try to compute the probability of error. That will be the average probability of error.

(Refer Slide Time: 08:48)



So, we want to compute the probability of we assume that s1 is transmitted and we also know that the correct decision is made correct detection is made if r 1 is r1 is greater than 0. Because, it for s1 to be chosen at the detector r1 has to be greater than 0 0 and also r1 has be greater than mod ri for all I not equal to 1. So, what is the probability of correct detection if s1 is transmitted and we know that r1.

So, r1 is greater than this and r1 is greater than mod ri for all i not equal to 1. And let us take the value of r1 to be alpha just like orthogonal signal set the way we computed we assume that r1 is equal to alpha. And alpha is greater than 0 because, r1 is to be greater if r1 is alpha is not greater than 0 then, there is a error. So, suppose we know that r1 is equal to alpha and alpha is greater than 0. Then, what is the probability of correct detection? Probability of correction detection is the probability that all the r other ri's absolute values are less than r1.

So, the all the components arise where i is not equal 1 are nothing, but the noise values in those components noise components. And those noise components are independent of each other. So, the probability of there being less than r 1 factorizes into individual probabilities because the noise components are independent. So, this becomes the probability of a noise components being. So, that the absolute value is less than r1 that is, the noise component being in the range minus r1 to r1 that is minus alpha to alpha.

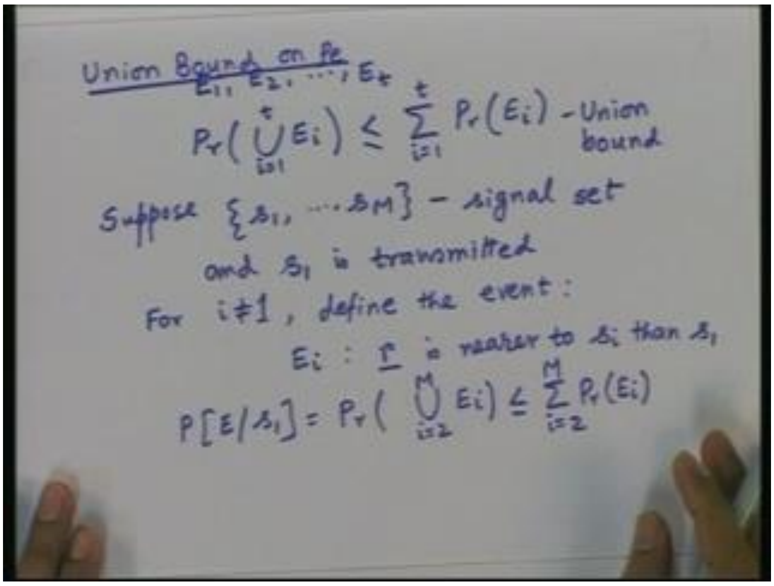
So, that probability is obtained by integrating the density function of the noise component in the range minus alpha to alpha. And then there are N minus 1 such value of i i not equal to 1. So, excluding on there are there are N minus 1 other dimensions, so N minus 1 is power. So, probability of correct detection is known to be probability of correct detection given that s1 is transmitted. Now, this is obtained by averaging this over all values of alpha above 0.

So, we have to integrate this from 0 to infinity. Alpha varying from 0 to infinity and then we have to multiply this by the density of r1 at alpha and then integrate that. So, now r 1 is nothing, but the first component that was transmitted is root over Es. So, root over Es plus n1, so if r 1 is alpha that means, that n 1 must be alpha minus root over Es. So, the density of n 1 is known so we have to take that evaluate that density at this point. For r 1 the be alpha n 1 must be this value so pn alpha minus root over Es.

Then take this minus alpha to alpha pnxdx N minus 1 d alpha and here this N is M by 2. So, we can if you want to write in terms of N then we have to write this as N by 2. So, here also the probability of correct detection is written as an integration and this is not so easy to compute. So, but this can be computed numerically and there is no other simple way of computing this. So, of both for orthogonal and bi-orthogonal signal set the form of the expression of the probability of error that we have computed is similar and they can be computed numerically.

But, can we get a reasonably good estimate of the probability of error. This is the exact probability of error, but can we get reasonably good estimate of the probability of error in a simpler manner. That is the question, so, the answer is we can get an upper bound of the probability of error by using what is known as the union bound. The you remember that we discussed when we discussed the background on probability that there is something called union bound on the probability

(Refer Slide Time: 15:15)



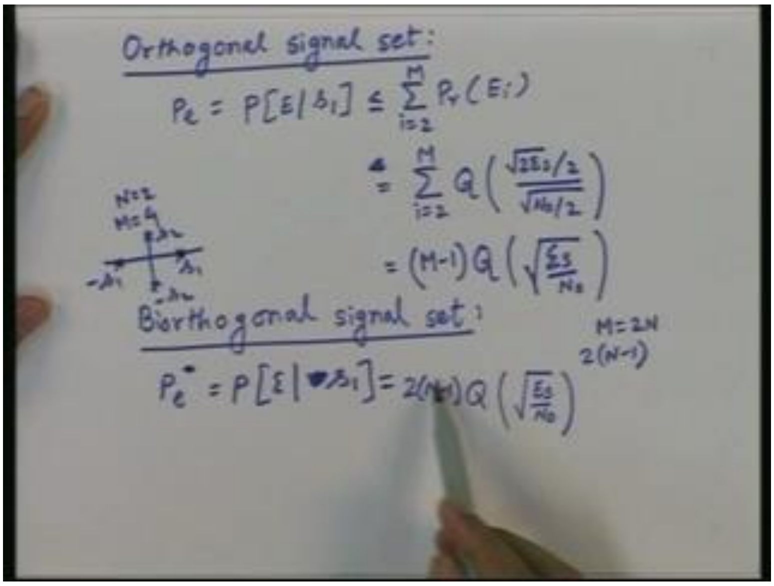
That is if there are some events E 1 E 2 to E say t. Then the probability of the union of the events I equal to 1 to t Ei is less than equal to the sum of the individual probabilities of the events. This is the union bound. So, we will use this to get an upper bound on the probability of error. So, suppose that so this we are discussing union bound on probability of error. And now suppose s1 s2 till sm is the signal set and s1 is transmitted.
Now, for i not equal to 1 we define the following event define the event Ei to be the event that the received vector r is nearer to si than s1 So, I is not equal 1 and if si the i'th

signal point is nearer to the received vector than s1 we say that E1 has happened. So, Ei is that event. Now, what is the probability of correct decision probability of the probability of error given that s1 is transmitted. We know that, if s1 is transmitted there will be an error in detection if we detect something other than s1. So that means, that 1 of the other signal points is nearer to the received vector than s1.

So, that means, 1 of the events E 2 to Em has happened; only then the error can happen. So, there is a error in detection if s1 is transmitted then; that means, that 1 of the other signal points is neared to s1. That means 1 of the events just now defined it is correct true. So that means, this probability is nothing but, the probability of the union of these events I equal to 2 to M Ei. And from the union bound on probability this is less than equal to the summation. I equal to 2 to M probability Ei.

So, and often this is easier to compute; probability of individual Ei is often easier to compute. And we will see for some particular examples for which the exact probability of error calculation was not easy that this becomes easy to compute. So, let us take those 2 examples orthogonal signal set and bi-orthogonal signal set.

(Refer Slide Time: 19:49)



So, orthogonal for orthogonal signal set the probability of error is same as the probability of error given s1 is transmitted. So, we need to compute this now, the event Ei in this case is the event that the received vector is nearer to si than s1. Now, the distance between s1 and si is same for all i’s not equal to 1. So, and that distance is root 2 times root over Es; root over Es times root 2. So, we know the distance between 2 points and what is the probability that the received vector 1 is transmitted.

What is the probability that the received vector is nearer to the other point than the transmitted point. That we know that is the probability of error PM we have computed that, so that probability is known. So, this is by union bound this is less than equal to summation i equal to 2 to M probability of Ei and this is summation i equal to 2 to M. And probability of Ei is Q of D by 2 by root over N naught by 2 and what is D by 2 it is the D is root 2 root over 2 Es.

That by 2 means so, root over 2 Es by 2 by root over N naught by 2. So, we get this is and this does not depend on i. So, this is basically M minus 1 times the M minus 1 copies of this being added. So, this times Q of root over Es by N naught. So, this is this is an upper bound of the probability of error for orthogonal signal set. And this is the union bound on the probability of error. And this is quite simple compare to the exact probability of error and this only needs the value of Q at some point.

For bi-orthogonal signal set for bi-orthogonal signal set probability of error again is because the signal set is symmetric. We have probability of error given s1 and now the events here are slightly different. There are M points M is equal to 2 N and out of so if we take s1, there are s2 2 sN and minus s2 to minus sN. There are these 2 times N minus 1 points 2 times N minus 1 points are there which are along other dimensions, but there is 1 point which is in the opposite which is the minus s1.

So, the distance of minus s1 from s is different from the distance of the other signal points from s1. So, what is the distance of the other signal points from s1 is the same as what was here, root 2 Es. But what is the distance of s1 from minus s1 it is root 2 times root over Es. So, those 2 distances also different so this just consider, the case of 2 dimension that is N equal to 2 M equal to 4 so, we have this.

So, this becomes four PSK signal set actually like this. This is s1 this is s2 this is minus s1 this is minus s 2. So, the event this is transmitted; this distance is different this distance is same as this distance, but this distance is different. So, all the points have same distance from s1 except for minus s1 that has a different distance so that will be computed separately. So, consider all the other points 2 times and minus 1 such points for them the probability of that event of being detected as is Q of this root over Es by N naught. And there are 2 times N minus 1 such points.

There is one more point so, remember this summation is there and we have taken; so, many terms, so many events 1 more event is left and that we have to add. So, that is Q of that distance is 2 times root over Es; so we will have here 2 Es by N naught. This is the union bound on the probability of error for bi-orthogonal signal set. And this is also quite simple it is only expressed in terms of the Q; that means, to be evaluated at 2 different values.

So, we have…So, far we have seen how to compute the probability of error exactly for: Q M constellation, PM constellation and for any binary signal set. And we have seen that that all those probability of errors can be computed or expressed in terms of the Q function in a simple manner. But for orthogonal and bi-orthogonal signal set the probability of error cannot be expressed in such a simple way in terms of Q function. There is integration involved and those integrations need to be done numerically because inside integrant is not a closed form expression it has Q function inside it.

So, though those functions those probability of errors can be computed numerically often, we need some gross idea about the probability of error which need not be very accurate. So, for that this union bound on the probability of error will serve the purpose. So, now for a better understanding of to get a good feel of these calculations and also to develop confidence in these formulas one needs to verify that these formulas actually give you the probability of error and their bounds quite correctly.

So, for that 1 can do MATLAB simulations; is very helpful if you do MATLAB simulation on take a signal take a FSK signal set or take a QAM signal set and transmit from the signal set randomly with uniform distribution on all the points. Transmit the points say 1,00,000 points and find out the probability of error. The receiver you do the detection by ML detection according the rules that I just now said. And for QAM you can simply take best on the Euclidian distance or try to simplify it along each dimension you can do independently, if it is 16 QAM or any rectangular QAM constellation. Then you get a probability of error; average probability of error from the simulation.

So, that is error rate it that need not be exactly same as the probability of error, but there will be slight variation. And then you can compute this probability of errors from the formulas we have just now developed. Again for compute for computing you can use MATLAB function ERFC to compute the Q function values.

Then also to compute the integrations involved you can do numerical integration either in C or MATLAB better to in mat lab because you ready made function for Q at least. And then you can compare the exact probability of error that you have computed. The union bound you have computed and the error rate it that you have found using simulation. These three things can be compared.

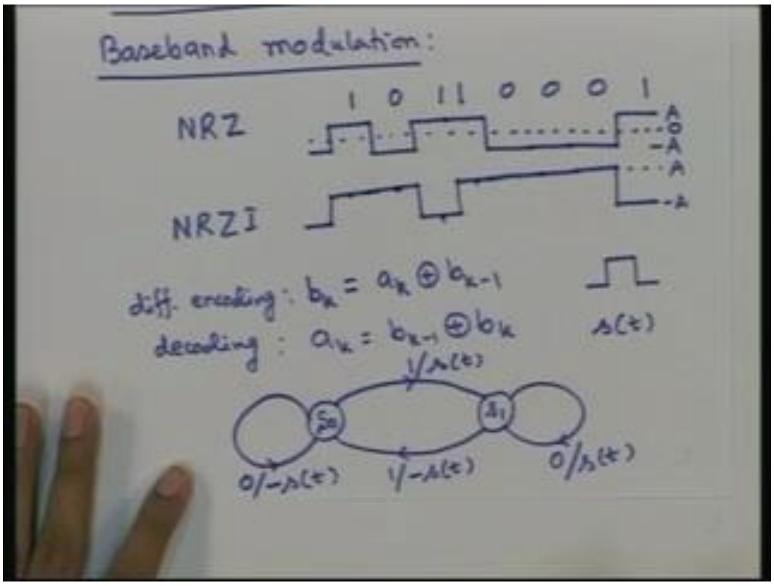
And all these three quantities can be plotted against the average signal energy or even a SNR Es root over Es by Es by N naught. You can vary Es by N naught and plot these 3 quantities. Do the simulation again and again for different Es by N naught Keep the N naught fixed and change the Es or keep Es fixed and change N naught to get different Es by N naught. And then compute and simulate all these values and plot them in a single figure and then find out how they compare.

That will give you some good idea about how good the union bound is for example and also the fact that simulation gives you a value probability of error value near enough to the computed value. We will give you some confidence about these formulas. So, this is all about probability of error calculation that we wanted to discuss in this course. And now we will we want to that move on to the next topic that is the modulation with memory

So far, we have compare we have discussed modulation techniques without memory. In each symbol what you transmit depends only on, the information bits that we wanted to transmit. It did not depend on what we transmitted before. So, what we will discuss now the modulation techniques that we will discuss now. There we will see that what will from transmit now will depend on not only the information bits that we transmit now, but also, on what waveforms we transmitted before.

So; that means, the modulator has some memory in it. Its transmitted waveform depends on what it transmitted before not only on the inputs input information bits, but also on what it transmitted before. So, that is why the modulator is called the this technique is called modulation with memory because a modulation remembers what it transmitted as some memory.

(Refer Slide Time: 32:42)



So, we are considering modulation with memory. We will just discuss some simple examples of modulation with memory first of all we will consider some base band modulation. So, you we want to transmit 0 1 a traditionally, we either transmit we assign 2 levels for 1 and 0 and then transmits transmit those voltage levels. And either it may be plus 5 volt and 0 volt for 1 it is plus 5 0 for 0 volt or, but then, there the signal value is returning to 0 whenever the bit is 0.

So, that is returned to 0 whenever there is a 0 bit that is to be transmitted, but there is another scheme where the transmission of 0 is not preferred, but you transmit the negative voltage for 0. That is called non return to 0 scheme because you do not return to the 0 voltage, but go to the negative voltage; toggle between plus and minus voltage.
So, that is called non return to 0 or abbreviated as NRZ.

So, there let us say you want to transmit 1011 then 0001 and so on. Then here we will transmit for 1 a high voltage then a low voltage then 1 another 1 then 000 then 1. So, this voltage will be same A and this will be minus A the negative voltage, the 0 voltage is in between. Now, this does not have memory because what voltage level is transmitted here does not depend what was transmitted before. So, this is not a modulation with memory, but this following scheme is a modulation with memory NR this called NRZI.

Here, the voltage level is changed from the previous level if a 1 is to be transmitted. So, for example, here initially we transmit minus A voltage then a 1 is to be transmitted so we change the voltage to plus A. Then transmit this level, but then 0 is to be transmitted. So, you do not change the voltage so whether, the voltage level will be changed to the other level or not depends on what we want to transmit.

So, 0 means no change of voltage level, then again 1 means change of voltage level, then again 1 means change of voltage level, then zeros 0 again 0 then 1 to transmit 1 we will change the voltage level. So, this is A and this is minus A. So, this scheme now it has some memory because what you are transmitting here depends on what you transmitted before. For the same bit 0 here we transmit 1 here we transmit 111, but at some other time we will be transmitting 0. For example if a 0 came after this we will be we will not change the level and we will keep transmitting 0.

So, for the same bit 0 we sometimes we transmit 1 sometimes we transmit 0. So, it does not only depend on what you want to transmit, but it also depends on what you transmitted before. So, if 0 is to be transmitted we keep the previous voltage level in this time slot also. So; that means, we have to remember the previous voltage level, so this has some memory. This encoding can be also expressed in this fashion that the present voltage level, let us call it bk.

Suppose, this is if we assume it to be 1 0 instead of plus A and minus A just assume this level to be 1 this level 0. Then the present bit is the transmitted bit is the ak, ak is the information bit k'th information bit. What is the input and then with the previous level previous bit that you transmitted. That is bk minus; 1this is called differential encoding. So, here this orbit is so for example, take this we here bk minus it so, we want to find out what we need to transmit here.

So, the encoder will decide what is to transmitted here So, it means to transmit zero. So, that is ak is 0. And then, bk minus 1 what was transmitted before is 1 here this level is 1 it is not this bit, but it is this level that is 1 and this level is 0. So, bk minus 1 is 1 and 0 is to be transmitted. So, 1 0 that is 1 so it transmits 1 then for this ak is 1 and then this level is also is 1. So, it you get 0 it transmit 0. Then this is bk minus 1 is 0 for the previous for this and this is 1 so and 1 and 0 you get 1;so, you get this.

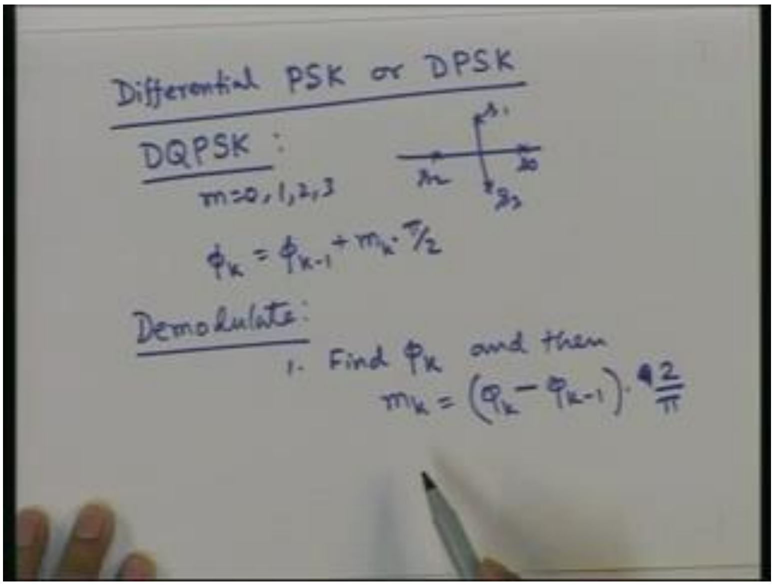
This is the differential encoding and from this 1 can see the the decoding must be can be done in a simple way as ak is bk minus 1 bk. So, you just take the 2 conjugative symbols 2 conjugative times slots XOR the levels have got and that will be the information bit. Take this XOR of this will be this bit. So, these 2 are same level so XOR will be 0 0. These 2 are different levels XOR will be one. So, 1 these 2 are different so 1 these 2 are same 0 and so on.

So, the decoding is very simple. And this encoding is expressed in terms of state diagram in a very simple manner say the encoder has 2 states s naught s 1. So, s naught denotes what was transmitted the 0 level that was transmitted and s1 denotes the 1 level. So, the encoder is at 0 level means it is in s naught level state; it is here 1 level it is s1 state. So, if the signal this level signals, A voltage level A is called suppose, we say that that is st that is this pulse value 1, in the negative is minus st.

So, if it is encoder is at state s naught that is 0 then, if it receives 0 then it will be there in that state it will not transmit the same minus st level; this is minus st. So, if it receives 0 what will be transmitted is so, it receives 0 then it will go to s naught itself because, the next level is going to transmit the same minus st waveform. And so, minus st will be transmitted and it will go to the same state. It will remain in the negative state if it receives if it wants to transmit 1 then it will transmit st.

It is the positive level, but then it will go to the next state because for the next time slot it will be in the higher state so, it the state is changing is coming to s1 state. Similarly, if it is in the higher state it will receives 0 the state will not change and it will transmit st if it is 1 it will change the state and it will transmit minus st. So, this is the start transition diagram of the encoder.

(Refer Slide Time: 42:40)



Now, similar to this scheme is there is a pass band modulation scheme called differential PSK or abbreviated as DPSK. Now, let us take an example of DQPSK Where QPSK constellation is used, but differential encoding along with QPSK constellation. So, we have QPSK constellation and traditionally, what we do is we input the 2 bits into 1 of these points and transmit. So that means, the actual phase of the signal that we will transmit depends on what bits we want to transmit.

If you want to transmit say m equal to 0 we will transmit this point: this is s1, this s2, this s 3. So, m equal to 0 1 2 3 for these we will transmit corresponding point. But in the DQPSK just like NRZI we do not transmit the exact point based on what you want to transmit, but we change the phase based on what bits we want to transmit. If we want to transmit m equal the 0 we do not change the phase whatever we transmitted in the last symbol interval you transmit that.

Now, if you want to transmit 1 we rotate the point whatever you transmitted before, whichever point we transmitted before, you rotate the point and take the next point in the anticlockwise direction and transmit that point. If we want to transmit to 2 we rotate it by 180 degree 2 times and then transmit that point and so on. So, again here just like NRZI what you transmit now depends on what you transmitted before; as well what information we want to transmit.

That is the phase difference between the previous symbol and the new symbol has the information. The information is contained not in the phase of the absolute phase of the present waveform, but between in the phase difference between the previous waveform and the present waveform. So, the information is encoded in the phase difference between the 2 consecutive symbol intervals. So, the phi k the present k'th phase of the signal that will be transmitted is phi k minus 1 plus the k'th m that is, the kith symbol that will be transmit then use to be transmitted times pi by 4 the pi by 2.

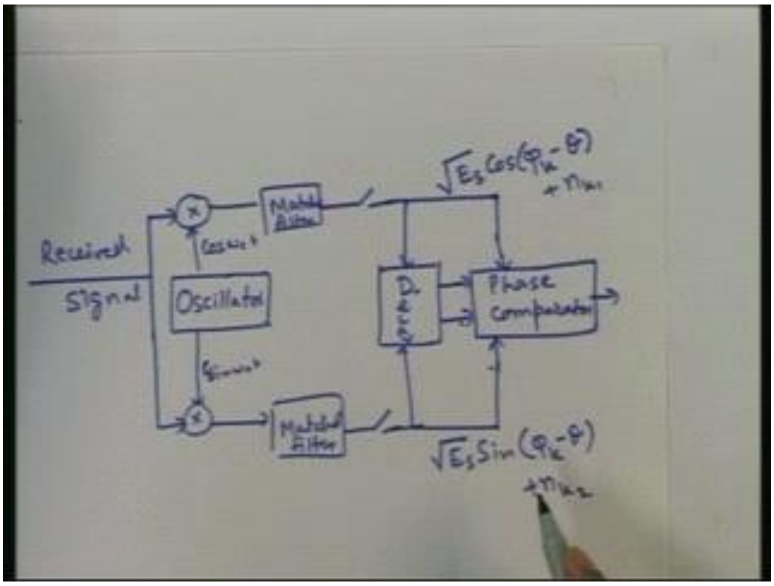
So, if mk is 0 then you transmit phi k minus 1 phase itself If mk is 1 then you transmit phi k minus 1 plus pi by 2 that is if previously you transmitted s1 and now you want transmit 1 you transmit s2 rotate it once and transmit s two. So, now how do you demodulate of course, we can do exactly the same way as NRZI. And we first find out what is the phase of the received signal that is we find phi k and then previously we actually found phi k minus 1.

So, for every k if you find phi k and then mk is determined by phi k minus phi k minus 1. Take the phase difference between the present and previous symbols and that times 4 by 2 by pi is the is the symbol that was transmitted. But here, we need to find phi k, but phi k may not be may be sometimes difficult to get exactly because, often what happens to the carrier phase is shifted when it goes through the channel out some part of the receiver.

So, it is it is often difficult to estimate at what exact phase the carrier has the transmitted signal, had at the receiver. So, in such situations when it is difficult to compute phi k or when the carrier phase is shifted by some constant. And as a result it is difficult to get the absolute phase of the signal, so you do not want to compute this. But, can we get this difference without computing this itself?

So, of course, there is a constant shift that will cancel when we subtract these 2, but even without computing this we can get this directly from the received signal in the following way. So, that is the method 2.

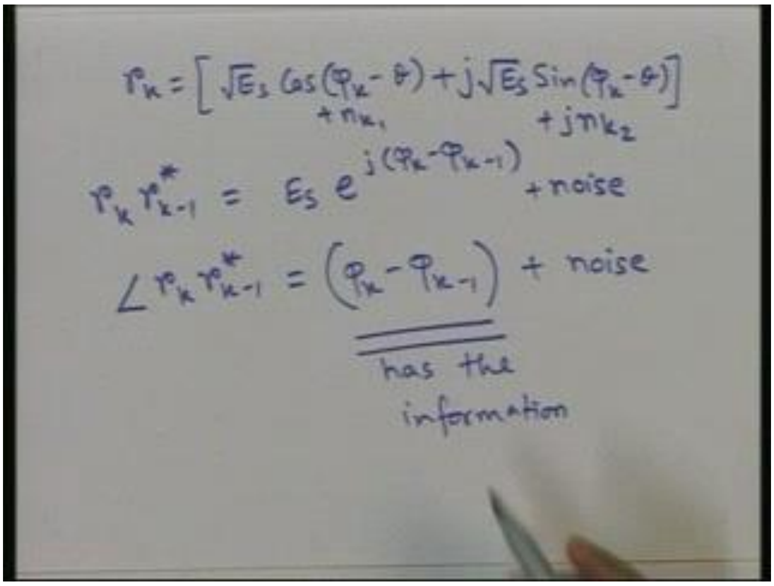
(Refer Slide Time: 47:44)



So, method 2 is this we take received signal and from an oscillator we generate cos omega ct and sin omega ct multiply pass through matched filter. And then, sampler then here we actually get the real part of the received vector that is root over Es cos phi k minus. Possibly, some constant phase shift that we were talking about and plus some noise nk 1. This is after doing the co-relation with the pulse actually.

So, this is the real part of the received vector 2 dimensional vector and then we also get here the imaginary part Es sin phi k minus theta plus n k 2. And then, we also delay it that is, we want the previous received vector. So, we delay both these pass through a delay and then this comes here, this comes here after delay. And then so, we have the present received vector here and the previous received vector here. And we compare the phase between these 2 vectors. So, we can compute the phase difference and decide on the the phi k minus phi k minus 1.

(Refer Slide Time: 50:30)



And mathematically, it means that we are computing rk the received vector. That is root over Es cos phi k minus theta plus j times root over Es sin phi k minus theta. This plus noise, this plus noise and then we have passed it through delay. So, we have got rk minus 1 also and then we a take rk; what phase comparer does is it can simply multiply this. Take the conjugate about k minus and multiply it rk. That will give us see that this signal part if you neglect the noise part it this is basically, it will root over E s times it will got j times this.

So, you get Es when you multiply these 2; you get Es e to the power j phi k minus theta minus phi k minus 1 plus theta. That is, we get phi k minus phi k minus 1 and plus some noise term. So, the if you now take the phase of this rk rk minus 1 the phase of this will be this phi k minus phi k minus 1 and plus some noise component will be there.

So, this part has the information from here we can find the information. If it is phi by 4 the bit is if it is 0 it is bit is 0. The symbol is 0 if it is phi by 2 the symbol is 1 it is phi symbol is 2 and so on. So, there is some deflection due to noise, but we will get the appropriate value. So, this is how 1 can decode the DPSK detect the DPSK signal, without explicitly computing the exact phase of the signal.

So, both DPSK and NRZ these are all modulations with memory 1 is base band other is pass band. And there are other schemes which you not discussed; these are the just basic schemes; that is all.

Thank you.