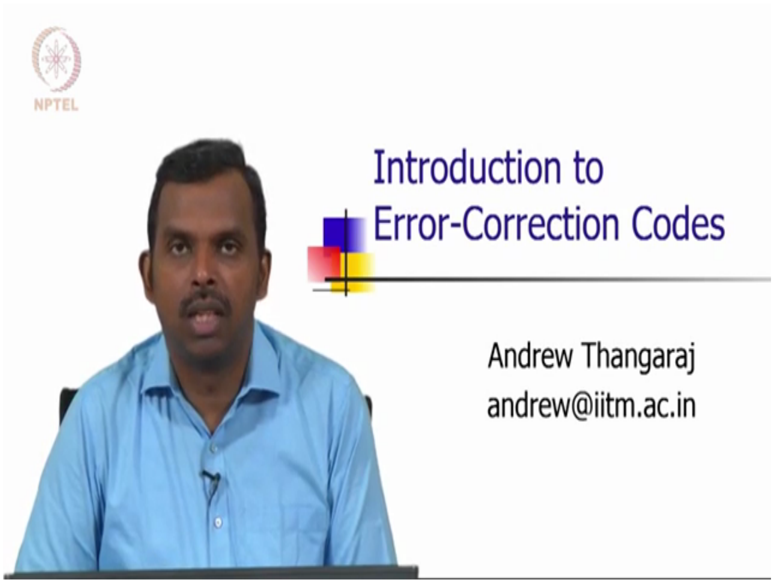
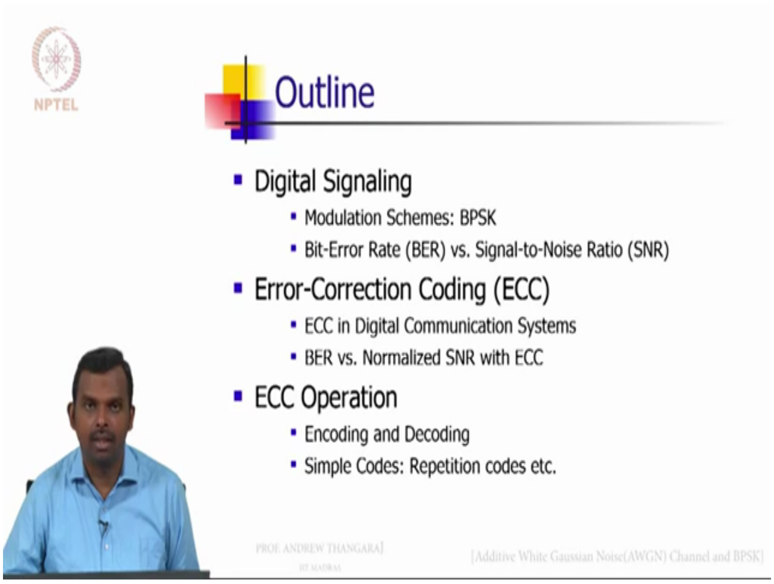
LDPC and Polar codes in 5G Standard
Additive White Gaussian Noise (AWGN) Channel and BPSK
Professor Andrew Thangaraj
Department of Electrical Engineering
Indian Institute of Technology, Madras
Introduction to Error-Correction Codes

(Refer Slide Time: 0:18)



In this lecture we will see introduction to Error-Control Codes and the role they play in digital communication systems, we will use a very simple model and describe the role and provide one or two simple examples of coding situations and show what happens there, so that is the basic objective of this lecture, so let us begin.

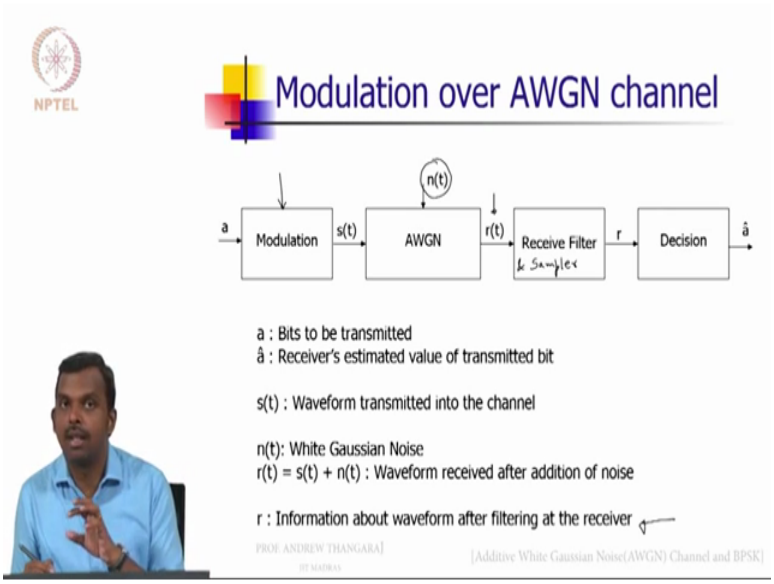
(Refer Slide Time: 0:42)



So here is a brief outline, we will start by talking about digital communication system models in particular we will talk about the modulation scheme which is called BPSK and then we will describe how to think of Bit-Error Rate versus Signal-to-Noise Ratio (SNR) now this is the fundamental trade-off in most digital communication systems, we will talk about that for a little while.

And then we will introduce how Error-Control Coding or Error-Correction Coding works in a digital communication system, we will specifically talk about a plot of Bit-Error Rate versus Normalized SNR or E b over N not in the presence of an Error-Control Code in the digital communication system, then finally I will conclude with how encoding and decoding is typically implemented and provide simple examples basically the repetition code is the example we will.

(Refer Slide Time: 1:36)



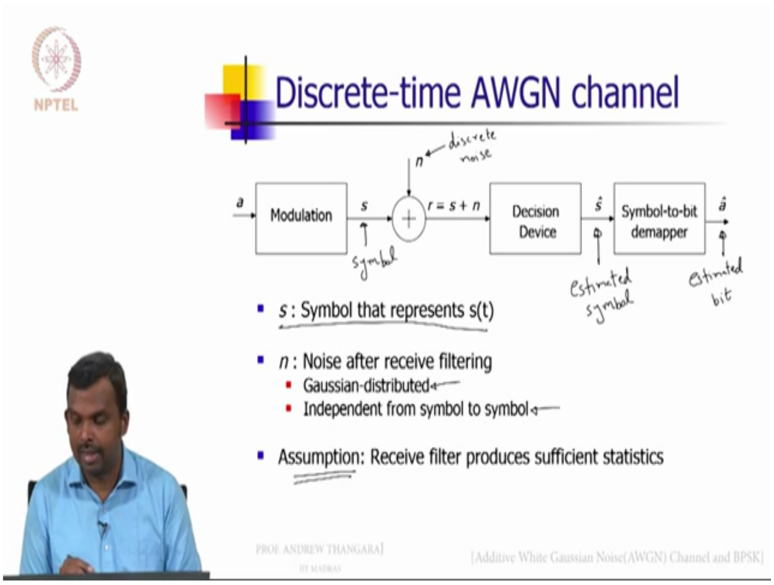
So this is the picture of what is called an AWGN channel Additive White Gaussian Noise Channel and the use of modulation over such a channel. So we do not have the time in this course to go into details of how this picture comes about, but I will start with this as an assumption and I will assume you have familiarity with digital communication to know what I am talking about, okay.

So as you know mostly the communication medium (is) will transmit signals electrical signals and in the digital communication system one needs to transmit bits, bits could be 0 or 1 and how this is done is usually the bits are converted into a signal, so that is what is called modulation and that is what is there on the first block here modulation, modulation refers to converting bits into signals, okay and once the signal is ready it is sent across the channel and for our purposes we will assume that what the channel does over all is to add a noise signal which is additive in the sense at adds which is also white in the sense it is sort of independent from time to time and it is Gaussian distributed, so this is Additive White Gaussian Noise Model.

The noise signal n of t this is the noise signal here it adds to the signal transmitted signal to obtain received signal r of t, so you get a receive signal r of t and this is usually filtered and sampled, okay so there is receive filter and sampler here and usually one can design this so that you do not lose any information and you get samples, okay. So as it turns out one can make a direct connection between the bit that was transmitted a and the sample r that was obtained, okay so this is the theory of how you model a digital communication system in the discrete time domain and once you do that you can make a decision to get a hat, okay.

So this is sort of crucial, so r the received sample is the information about all that you did in the digital communication system after you have filter, okay. So all the information about a is in r and you can do a decision.

(Refer Slide Time: 4:04)



So let us look at this in the discrete time domain we had the continuous time domain in the previous slide, this is the discrete time domain, you can see I have got an rid of the variable t, so what people usually do is to represent the transmitted signal s of t in terms of a symbol, so this is done using a bases representation and all that, this is quite standard we are not going to look at it in detail in this class but let us assume we do that.

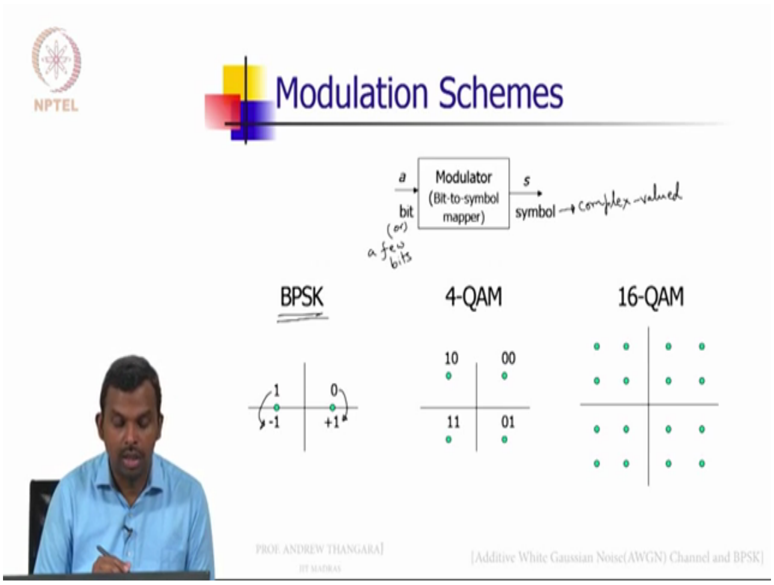
So this modulation converts the bit a into a symbol s, so this is the call the symbol is quite important to know how this works, okay and instead of a noise signal n of t adding to the signal s of t we can have a discrete version of the noise, okay so you have a discrete noise version and like mentioned here it is Gaussian distributed and it is independent from symbol to symbol, okay. So these are things that you can prove by carefully looking at the model.

So at the end of the day we have a really simple model for the entire digital communication process, you have a bit you converted into a symbol s and it is sent through a channel whatever happens to it all that you are concerned about is this received value r and that received value r is simply the transmitted symbol s plus a Gaussian distributed noise which is independent from symbol to symbol, okay. So this is a very simple and elegant model for a physical complicated system and it is quite important and because of such models we are able to do advanced theoretical studies of the systems and improve them very nicely, okay.

So once you have this value r you can run a decision device and come up with an estimate of the symbol, this is the estimated symbol, once you have an estimated symbol at the receiver you can do a demapping and go back to the bit, okay so this is the estimated bit. So this is the model that we will use throughout this course for capturing the behaviour of a digital communication system, we will not go into any other details then this this is the model that we will use.

So there are assumptions here on how we assume the system has to work, the timing has to be right all of that we assume yes that is true but those are engineering problems that we will not focus on in this class, we will work with this model and try to see how to get the best out of this model, what is the best performance that one can extract from such a system, okay so that is going to be the goal of this class at least.

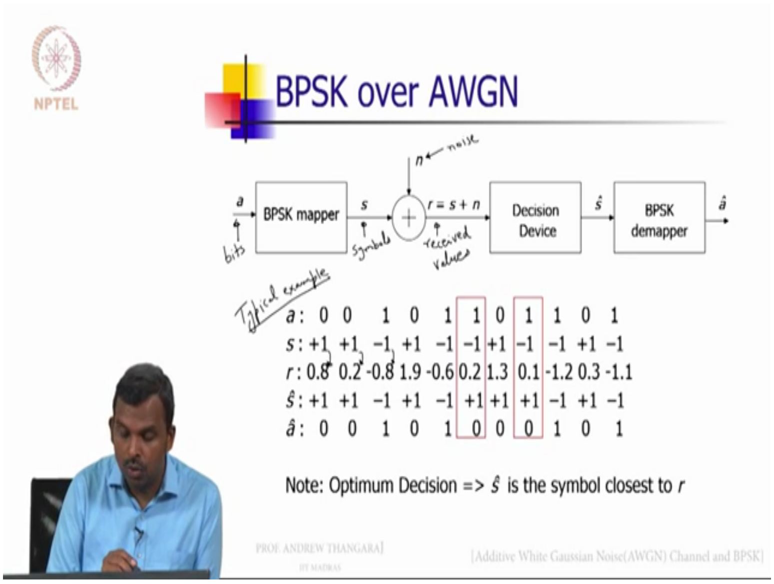
(Refer Slide Time: 6:50)



So what are the various modulation schemes that are used and practised today? So like I said you have a bit or may be a few bits, few could be you know 2, 3, 4, 5, 6 today people use even 10 bits sometimes for doing modulation and you take these bits and convert them into a symbol. Now the symbol is typically can be complex valued, okay so how do complex values convert to a actual transmitted signal s of t which is real? That is again theory we will not see, but one can assume that the symbol is complex valued, this is the idea of QAM Quadrature Amplitude Modulation which may be you are familiar with, okay.

So I have shown here 3 pictures, the first one is BPSK, I am underlining this because this is the only modulation that we will use in this class, but I want to point out there are other more advanced modulation schemes like 4-QAM, 16-QAM which are actually used today in communication systems, we will not talk about them in this class, we will focus solely on BPSK. So how does BPSK works? Bit 0 is mapped to the symbol plus 1, okay and bit 1 is mapped to the symbol minus 1, so that is BPSK. Like I said there are more advanced modulation schemes which actually use complex values, we are not going to focus on those kind of schemes here, we will only use BPSK over AWGN, okay so that is the system we are going to study next.

(Refer Slide Time: 8:28)



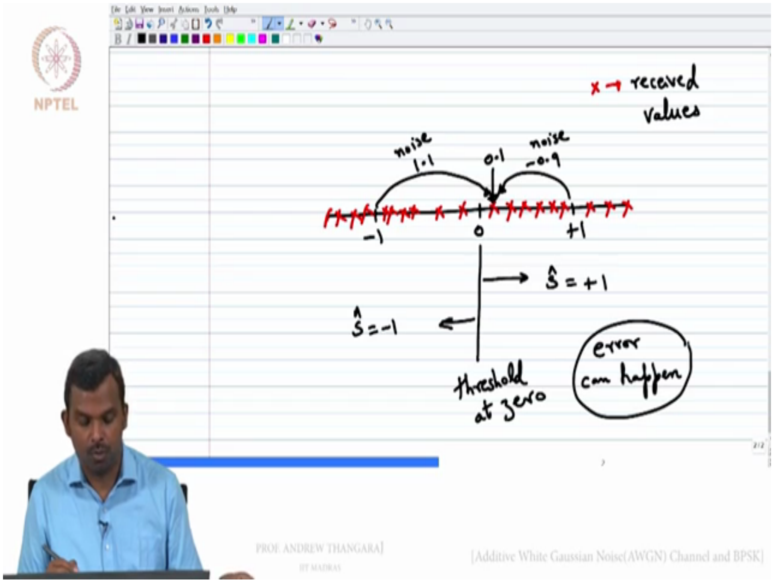
So let us look at BPSK over AWGN, it is important to know how this system sort of works and go through it with an example and imagine what is going to happen because this is the model we will use throughout this class, till the end of this class we will use the same model, okay. So you have bits coming in from the left hand side these are bits, what I have shown here is a typical example, okay so this is a typical example here of what can happen. So I have shown here bit sequence which is 0 0 1 0 1 1 0 1 1 0 1 this could be a bit sequence that you are sending through your system, okay.

So the first step is you map the bit into a symbol, okay. Now for this particular sequence how you are going to do you are going to do a BPSK mapper, so 0 goes to plus 1, 0 goes to plus 1, 1 goes to minus 1, 0 goes to plus 1, and so on, okay so that is what happens, alright and then noise gets added, okay and you get received values, okay noise gets added and you get received values and I have shown a typical possibility for a received value here, I am showing only one decimal place it could actually be more than one decimal place depending on how much resolution you have on the system.

Here is an example, so the symbol plus 1 was transmitted it got received as 0.8 which means what? The noise was minus 0.2, okay the Gaussian Noise it can be positive, negative typically you think of noise as being small, okay. So the noise could have been minus 0.2 and you got 0.8 at the received value, the second symbol is plus 1, the noise was minus 0.8 this time, so you got 0.2, the third symbol was minus 1, the noise was plus 0.2 at this time, you got minus 0.8, so on and so forth.

So you go through and you send all the symbols, some noise happened you cannot predict the noise you do not know what noise it will be ahead of time, so even if you send the same symbol vector again you will not get the same received value, right so you will get some other received value and that is in essence the problem in digital communication how do you work against that noise, okay so what do you do with the received value? So if you think about what is going on?

(Refer Slide Time: 11:06)



So here is an axis, here is 0, here is plus 1 your symbol corresponding to bit 0 and here is minus 1, okay. You are transmitting either plus 1 or minus 1 and noise gets added and your received values are going to be may be here, here, here, here, here, here, here may be most of the received values may be will be here, okay but a few received values can also be here, okay. So this x indicates received values, okay.

So at the receiver you get only these red values, all over the place 0.1, 0.2, minus 0.2, minus 0.3, minus 0.5 like that you will be getting. Now you have to decide for every received value what would have been the symbol that was transmitted, okay so that is the decision you have to make, okay we have to look at the received value and decide was the transmitted symbol plus 1, was the transmitted symbol minus 1?

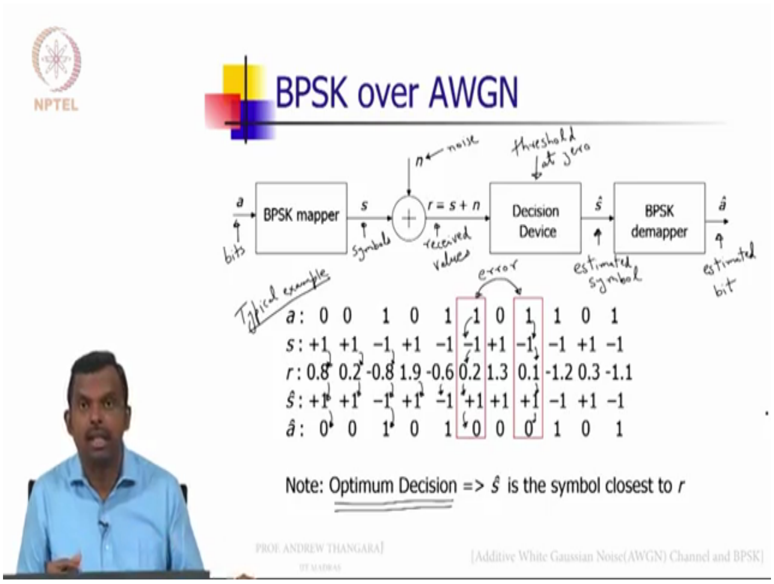
Now remember the noise is Gaussian it is symmetric and all that, it has very easy description. So one very obvious thing to do is to draw a threshold at 0, okay what do I mean by threshold at 0? Any received value to the right I will decide any received value to the right of 0 or any received value which is positive I will decide for that received value the transmitted symbol to be plus 1, okay so that is the thresholding at 0, if the received value is negative it is to the left of 0 then I will decide that the transmitted symbol was minus 1, okay so that is my decision, I am going to use the simple thresholding rule to decide whether it was plus 10.2, or minus 1, okay.

So on this side s hat which is the estimate of the transmitted symbol at the receiver is plus 1 and on the left side s hat is minus 1, okay. Now I want to point out one very important thing, okay so this thresholding can cause error, why is that? So let us take a typical let us take a received value which is closed to 0, let us say this one is 0.1 let us say, okay. If you have a received value of 0.1 the transmitted symbol could have been plus 1 and the noise could have been minus 0.9, okay or the transmitted symbol could have been minus 1 and the noise could have been plus 1.1, okay and there is no way at the receiver that you can know which of these things happened, right you do not know both of these could happen.

Now you are looking at the noise value and you are saying okay 0.9 is smaller in magnitude than 1.1, so it is more likely that the noise of point 0.9 happened than that the noise of 1.1 happened and you are deciding that you are going to go to plus 1, that is the choice that you make and that choice can be erroneous, okay it can be wrong. So it can happen that the noise was actually very bad and you were not in luck and noise was actually 1.1 and it was actually minus 1.

In those kind of cases you will make an error, okay so error can happen that is something to remember so alright, okay so any thresholding you do error can happen, so the question is can we characterize the error, can we find the probability of error?

(Refer Slide Time: 14:52)



So here what I have done after the received values are received is this is a thresholding at 0, okay and here I have estimated symbol, okay and this is estimated bit, okay. So you can see what I have done here, whenever the received value is positive I am estimating it as plus 1, whenever the received value is negative I am estimating it as minus 1, okay so that is what happens here, okay.

And once you have the estimated symbol going to the estimated bit is very easy if the symbol is plus 1 you say it is 0, if the symbol is minus 1 you say it is 1 that is it, okay. So the last part is quite easy because you went from 0 to plus 1, from plus 1 you go to 0, right so that is what you do and other way also 1 you go to minus 1, so from minus 1 you go back to 1, okay so that is what you are doing here in the last part, okay.

So now what are these two things? In these two instances of transmission there was an error, okay. So notice what is happened you transmitted a bit 1 it gave you the symbol minus 1 and then the noise was so high the noise was plus 1.2 that you went to the other side of 0, okay 0.2 and your thresholding device has no other way than to conclude from 0.2 to plus 1 and from there you made an error, okay.

So let us look at this other case you transmitted a bit 1 again it was minus 1 the noise was so high it was 1.1, it went off to the other side of 0, and you made an error, okay. So like I said you can make an error but it turns out you cannot do anything better in this scenario, so this thresholding at 0 is an optimal way to decide, okay optimal in the sense that it produces the lowest probability of error on average, okay. So that is something we can prove, we will not look at the proof in this course but this is what you can do.

So once again the idea behind this slide is to show you what happens in a BPSK, AWGN system and this is a good understanding that you should have so depending on the noise, depending on the noise values you can make errors but this is the best way to operate in BPSK, AWGN. So now the question is can you do anything better? Can you improve on the error rate and all that? So that is what error control codes bring in, we will look at that as we go along.