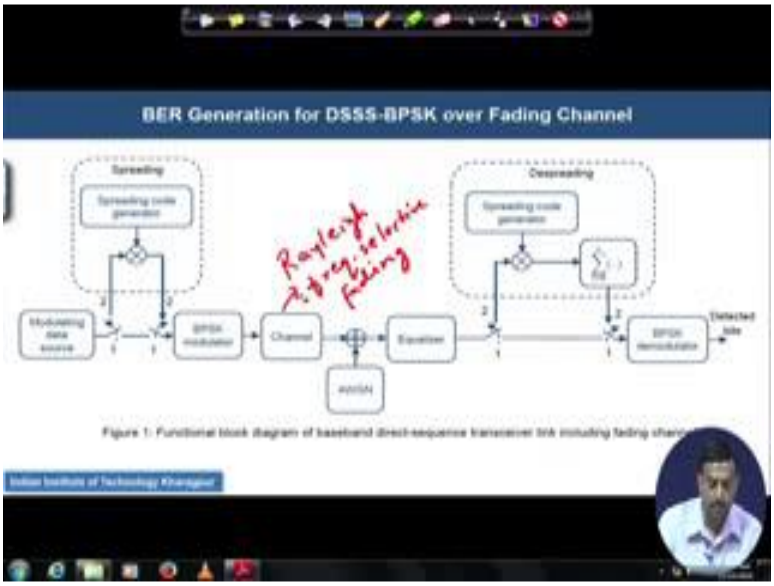
Spread-spectrum Communications and Jamming
Prof. Kutty Shajahan M
G S Sanyal School of Telecommunications
Indian Institute of Technology, Kharagpur

Lecture - 52
Tutorial – VII

Hello friends, so today we will discuss tutorial-7, which is basically dealing with MATLAB code for BER generation we will evaluate BER performance of a spread-spectrum BPSK system. Specifically, we are going to do it over a fading channel as against the plane AWGN channel, which we are dealt with in the last tutorial. So, this tutorial is specifically to evaluate the performance over a fading channel and we will see what kind of fading channel we are going to consider.

(Refer Slide Time: 00:59)

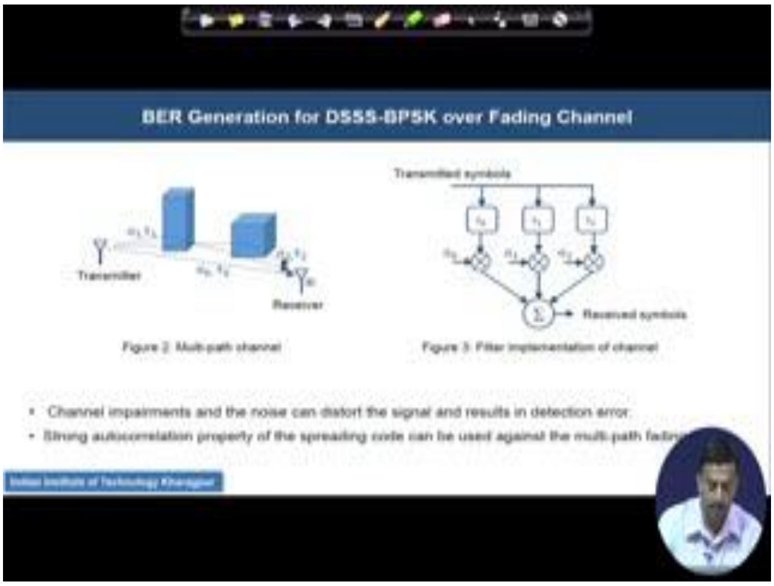


So, as was explained in the previous tutorial as well, we begin with the modulating data source which will generate bipolar data from binary data. And the switches as explained was the operation of the switches are such that if it is on, switches are on in position one and we have a conventional BPSK modulated system without spreading whereas switching to two incorporates the spreading process, and corresponding with despreading process at the receiver. So, we are going to analyze the performance of the spread-spectrum system. So, we are going to switch over to position two, and then we have a spreading code generator which will be multiplied with incoming bipolar data. So, this as we shall see is going to be a Walsh Hadamard code once again. And we are going to take a length of 4. We have a BPSK modulator followed by a channel and this channel to be very specific is going to be a Rayleigh frequency selective fading channel.

So, the modulated signal is going to be distorted due to frequency selective fading, which is nothing but a type of distortion that occurs when at the band width of a signal is greater than the coherence bandwidth of the channel. Basically what it means is that the different frequency components of the modulated signal will undergo different gains. So, the spectral property of the original signal is going to be modified by the channel. So, it is a slightly more complicated fading process as compared to the relatively simple flat fading case where all frequency components of the modulated signal undergo the same gain. However, the fading amplitudes are different, but in frequency selective fading the distortion is more because the spectral properties of the modulated signal itself changes.

So, you add AWGN noise which is universally present at the receiver; because of frequency selective fading we need to employ an equalizer in order to mitigate the effects of frequency selective fading. So, in our case, we are going to use what is called as the MMSE equalizer that is a minimum mean square error equalizer. And finally, after equalization, we despread the signal using the same spreading code which was used at the transmitter; and we demodulate the signal as we had done in the previous case.

(Refer Slide Time: 05:00)

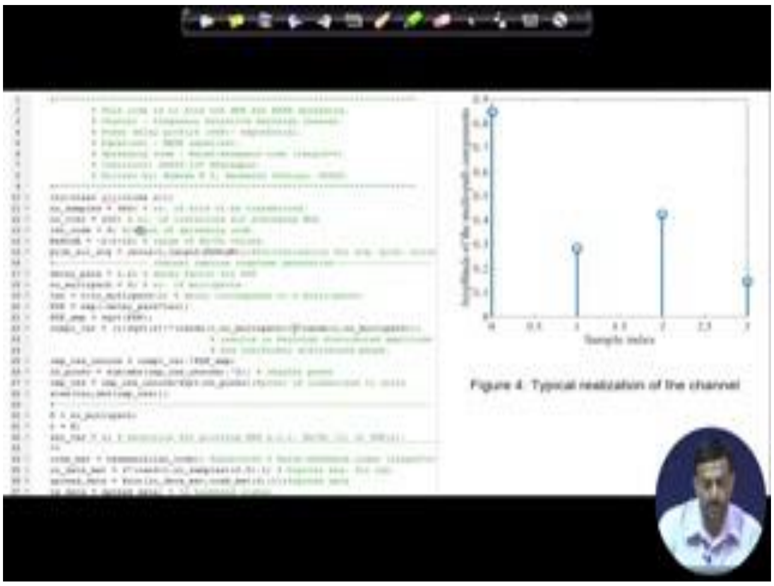


So, basically this diagram highlights the multi-path fading effect that happens in typical wireless channels. So, we can see that different version multiple versions of the same signal are received through different paths, for example, the path corresponding to the parameters a naught and tau naught where a naught denotes the path in and tau naught denotes the path delay between the transmit and the receiver is a line of sight path. Whereas, due to reflections from objects, we have additional paths presented by parameters a 1, tau 1 and a 2 tau 2. So, typically wireless channel would inculcate more than one such paths which is essentially the cause for distortion.

And a simple filter implementation or interpretation of this effect can be in terms of this linear filter that has been modeled. So, we can model this effect by introducing the corresponding delays and map this effect to this model, and implement it in order to run our code. We assume that we will have some kind of integrate random detector, which will be able to resolve these paths and some them up in order to facilitate detection.

Now, one of the advantages of using a spread-spectrum system over conventional BPSK is to a large extent it can help mitigate multi-path fading, because after all the spreading process spreads that signal over a wide bandwidth. But the strong autocorrelation property of the spread signal can the spreading code can be made use of in order to mitigate the multi-path effect. Essentially what I am trying to say is spreading basically gives you diversity which helps in reducing the effect of multi-path channel.

(Refer Slide Time: 07:39)



So, that is what is our objective to find out the performance of such a wide spectrum system over a multi-path frequency selective channel. So, much of this code at least the initial part and towards the end is the same description that all was there for the previous code that we had covered in the last tutorial. So, I will quickly go through it, we generate a certain number of samples and we fix a certain number of iterations in order to average over the these number of iterations in order to find our BER. We can take, vary this code length we could make it 4, 8 etcetera.

So, in this case, I have set it to 8 then we have a range of will be the E b by N naught values from minus 2 to 12. We have the probability of the error the error probability the average error probability which can be which is expressed as a variable over here. And we are going to collect the number of errors and store it in this variable subsequently. Now, the key parts of this code are one is the generation of the frequency selective Rayleigh fading channel, and secondly the equalization and spreading, despreading operation.

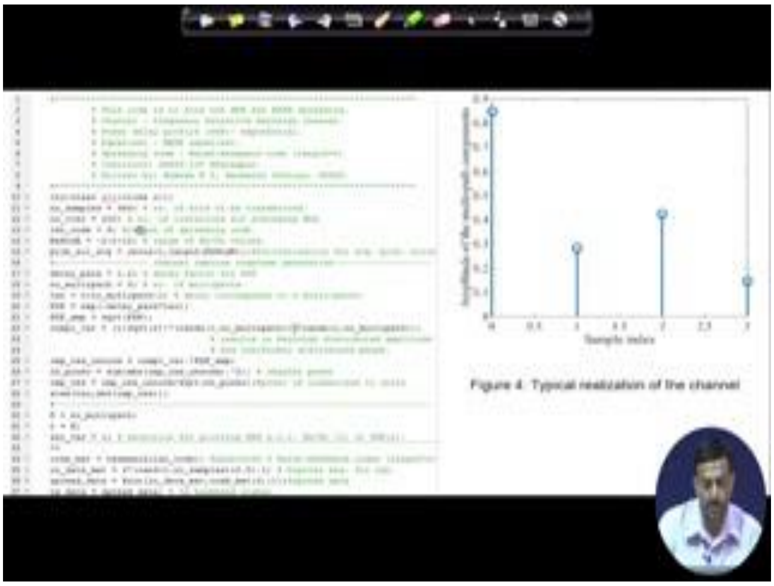
So, this first part over here deals with the generation of the channel. So, we have typically mobile communication channel would have what is called as an exponential power delay profile and that is characterized by a decay parameter which we have chosen a typical value of this exponential decay parameter to be 1.2 of course, you could change this and play around. We have chosen four number of multi-paths once again you could change this, just like how you could change the length of code and observe the effect.

So, these are parameters settings the amplitude of this PDP is nothing but a square root of this decay profile. The Rayleigh distributed amplitude parameters are generated by taking the absolute value of two Gaussian random variables, which here is expressed as a complex Gaussian random variable as you can say. This is a standard technique, in order to generate Rayleigh distributed amplitude; and ensure at the same time, that the phase of these components the Rayleigh components are uniformly distributed over 0 to 2 pi.

We generate the impulse response, which is not normalized because we need to do that subsequently, but the impulse response is nothing but the power delay profile amplitude which, we obtained in line number 21 multiplied by this complex variable which as an envelope which is Rayleigh distributed. So, we can see and then we can of course, normalize the impulse response by finding the channel power and dividing the previous impulse response that we got by this channel power. So, what we get end up getting as a normalized impulse response of a typical Rayleigh frequency selective multi-path fading channel which in our case or this run of the code looks like this. So, please remember that this is the impulse response normalized by channel power.

So, this is the plot that is in the number 28. The next part of the code basically deals with spreading. So, we had already seen this in the last tutorial. So, I will not go into the details we are using as I said Hadamard code of length 8 in this case, but you could experiment with length 4 because I think that was what was covered in the last tutorial. Nevertheless, the idea is to generate a bipolar-sequence and spread the data by multiplying the bipolar data with the spreading code that is chosen from the matrix Hadamard matrix basically which is generated in line number 34.

(Refer Slide Time: 12:35)



This part of the program that is line number 41 to 60 deals with the in addition of noise that is we generate the range of E b by N naught values. But importantly what needs to be noted over here is line number 46, which now is not merely the addition of the signal with AWGN, which will come later of course. But, prior to that because it is a frequency selective channel we convolve the transmitted data that is a spread data with the impulse response which was generated in the previous part of the code.

So, convulsion process models frequency selectivity. So, this is a k r important difference between this program our code and the previous one. We add noise as we get generated in the code in the last tutorial, so that part remains the same. Of course, the difference being that for E b by N naught to be scale the noise factor by the length of the code because as we had said the energy gets spread over the chips that is a code chips. Whereas, for SNR, there is no square root length code term because the power essentially remains same, there is no motion of time for SNR. The convolved data is added with AWGN noise and we calculate the variance of the noise, which is to be used for the equalizer.

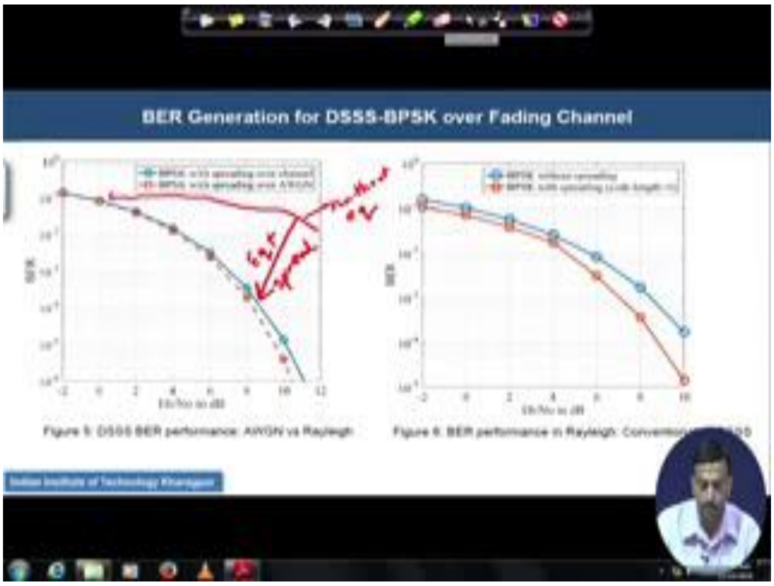
So, now at the receiver we use MMSE based equalization. And actually the description of this equalizer is beyond the scope of this tutorial; however, I will just tell you some important points. The parameters required to implement this MMSE equalizer is first of all we need to generate the ideal response that we would get if you would convolve the channel impulse response with equalizer impulse response. And an ideally that should look like a delta function if we need perfect equalization, so that operation or rather that requirement of the delta function is given in line number 72; other than that we need to specify the length of the equalizer in terms of its order or the number of tabs. So, normally we would choose the number of tabs greater than the number of the samples that we have for the impulse response, so that is what is chosen over here that is in line number 66. You could increase this length and see the effect of equalization.

I will cut it short and advise you to kindly go through this implementation of this equalizer by reading the theory from the book Filter bank Transceiver for OFDM and DMT systems. So, this book is written by the authors Lin, Phoong and Vaidyanathan; so it is been basically a Cambridge university press publication. So, I would request you to go through the theory first for this MMSE equalizer, and you will notice that the implementation follows the theory very closely.

(Refer Slide Time: 17:17)

So, up to this part is the equalization part, and we also do the synchronization certain number of bits initially of the convolution and then subsequent equalization is discarded that is for the sake of synchronization. And the received data which is the equalized signal is stored in the variable specified in line number 82. The rest of the code is similar to what we had discussed in our previous tutorial where in we consider the received signal vector convert into a matrix. Take the Krnocker in order to generate the periodic repeated code sequence, and we simply multiply the code sequence, which with the set of data corresponding to each spread bit. And then we dispread the signal, sum it up, and we measure the errors and we determine the probability of errors. So, this is pretty straightforward and was discussed previously and the last part of the code deals with a bit error rate plot versus E b by n naught in this case.

(Refer Slide Time: 18:46)



So, we see that there are two important diagrams over here. The first one highlights or depicts the performance of this DSSS system that is a BER performance with spreading over AWGN channel, which is usually the bond and in a fading environment. So, we clearly see that the BPSK BER performance with spreading over a fading channel is pretty close to that of AWGN channel, but remember that is only because we have used the equalizer in this case, and we have also used spreading. So, spreading and conjunction with equalization has managed this kind of performance. However of course, if where if we were not to use equalizer for example, then we will we will get a kind of flat kind of curve. So, this kind of curve is without equalization. So, equalization is extremely important to start with and when we employ equalization plus spreading, we are able to achieve a very good performance.

The second diagram shows the performance - BER performance comparison between the BPSK system with a code length of 4. So, I would also advise you to plot the same for a code length of a 8, because that is what we have done in our code, it should be better than this performance, so that is left is in exercise to you. But with a code length of 4, we see that with spreading we have a better performance in a fading channel. So, here we are not shown the AWGN channel because that is already discussed in figure 5. So, both these curves are to do with frequency selective Rayleigh fading channel and we clearly see that there is a performance improvement when we use spreading as compared to when we do not use spreading.

The one problem here is that it is difficult to quantify by whatever or rather by to what extent this performance improvement occurs or to give a value to the gain. The simple reason being that there are two factors involved in we realizing this kind of a performance; one is the effect of equalization and of course, the second one is the effect of the frequency selective Rayleigh fading channel. So, there are mathematical tractability issues due to which there is no value that can be associated with the gain of obtained in terms of the BER performance improvement. However, the improvement can clearly be seen between the BER performances of BPSK with spreading and without spreading in a Rayleigh faded frequency selective multi-path environment. So, this concludes tutorial-7.

Thank you.