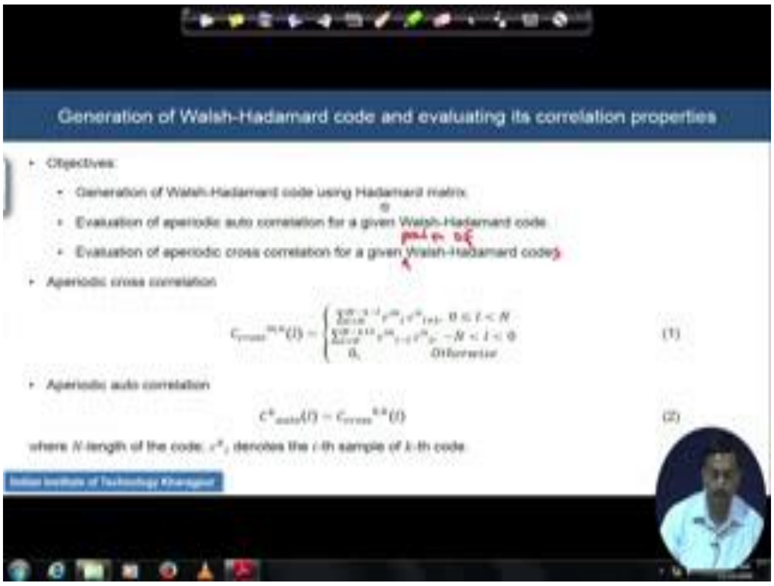
Spread Spectrum Communications and Jamming
Prof. Kutty Shajahan M
G S Sanyal School of Telecommunications
Indian Institute of Technology, Kharagpur

Lecture - 23
Tutorial- III

Hello friends. So, today we will take up tutorial 3, which is basically dealing with MATLAB code for generation of Walsh-Hadamard code. And we will check it is correlation properties in order to establish how these kinds of codes are useful for spread spectrum communications. In fact, this tutorial is the beginning of the series of tutorials where we will discuss MATLAB codes essentially. The first 2 tutorials in fact, were dealing with certain numerical on DSSS, FHSS. So, from now onwards till tutorial 8, we will be dealing with MATLAB codes and performance evaluation of systems, and we will try and gain a insight into how to simulate spread spectrum systems and evaluate the performance.

(Refer Slide Time: 01:49)



So, I assume that most of you are familiar with MATLAB or at least certain aspects of MATLAB, because that will prove useful as a pre requisite to our discussion. So, the objective of this tutorial is to generate a Walsh-Hadamard matrix. And the rows of this matrix will be corresponding to Walsh-Hadamard codes. We will check a periodic auto correlation property, in the sense that we will choose a given row of this Hadamard matrix and we will test it is auto correlation property, will plot it and check whether it follows any trend. And we will also check the cross correlation for a given pair of such codes. So, we will choose any 2 rows of the Hadamard matrix, and we will check the cross correlation property of these 2 codes.

So, that is the idea a aperiodic because we will not assume that the codes keep repeating, but we will take only one instance of the code and we will check the correlation property. So, the aperiodic cross correlation equation can be expressed using this relation. So, we have cross correlation term which is denoted by C cross. And it is between 2 different codes. So, therefore, we denote the codes by the notations m and n. And as we know that correlation has to be plotted versus lag over the shift. So, we denote the shift by l. We have the definition such that for all positive lag values or shift up to N minus 1 because this is l, lesser than N.

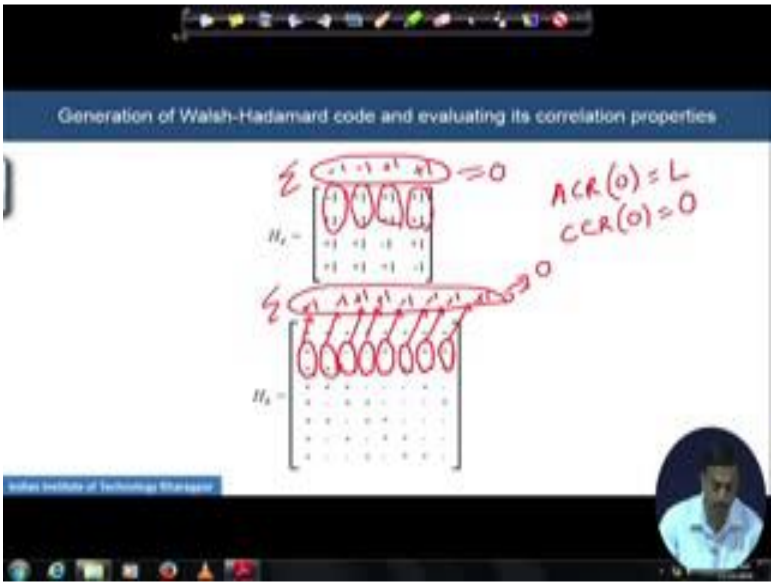
So, we have a definition whereby we keep one code as it is and we shift the other code based on this index. So, for all positive values we will have a shift in one direction and for all negative values we will there is for all negative values of lag we will shift in the other direction. And we will check as to what is the corresponding product of the code values. And ultimately we will sum it over the product values. Because this is a well-known definition of correlation I will try and show it to you with an example in the next slide.

So, turns out that the values of this cross correlation term for this lag will exist as we can see over here. From on one hand that is on the positive side it will exist from 0 to N minus 1, and on the negative side that is for the negative shifts it will exist from minus 1 to minus N minus 1. So, otherwise the value outside this range is equal to 0 which is essentially the meaning of aperiodic cross correlation. So, aperiodic is because of this reason we want to test the property. The code does not repeat themselves says that once overlap between the codes is over, then the value of the corresponding correlation term turns out to be 0. So, that is the significance of the correlation being aperiodic.

The auto correlation term is just special case of this cross correlation definition. So, auto correlation is it is nothing, but checking the similarity between the same code. That is a code with itself that is just a shifted version of the same code. And so, therefore, now the indices m and n are the same now, because it is the similarity with itself. So, we have only one index. And we check the correlation property of the code with shifted version of itself. So, we assume the n length code and specifically small c k superscript k, subscript i denote the ith sample of the kth code. So, this is the notation that we are going to follow.

So, of early simple definition to begin with and let us see what kind of a matrix is a Hadamard matrix.

(Refer Slide Time: 07:13)



So, as we see we have Hadamard matrix which can be of a given size. And I think this has been dealt with is in a previous lecture. So, I am not going to get in to it, but generally whenever we have Hadamard matrix, the first row of Hadamard matrix will be of all 1s, but here we see that we have a special case where the first row is not all 1s, but it is a particular sequence of this of course, it is a special case that is only known case of a cyclic circulate Hadamard matrix. But other than that in general this will be all 1s, that is the first row and it will not be used as a spread spectrum sequence because it serves no purpose basically.

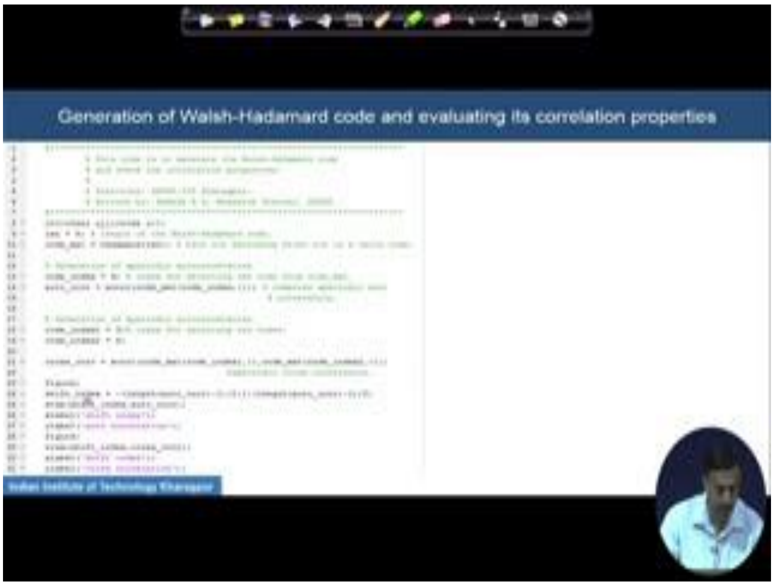
You will need a pseudo a kind of a pseudorandom sequence, but this is a you know sequence of all 1s. So, if you multiply it with a data bit there is no change in or there is no spreading essentially data occurs. So, so usually this first 2 is a redundant row, but then there are other properties which are of interest. So, to begin with all rows will have an auto correlation value that is basically, as I said we auto correlation we take the code which is a row of this matrix, and we will try and shift or rather try and find out the correlation between that row and a shifted version of itself. So, at 0 lag that is at auto correlation equal to 0, we will have a multiplication of this each of this bits with itself and summation over it.

So, it is quite apparent that we if we for 0 lag we will end up getting a value of auto correlation which will be equal to the length of the code. For example, in this case the length of the code is four. So, auto correlation at 0 will be equal to 4 and in this case at 0 it will be equal to 8. Now as we shift the signal, what will happen is we will start getting certain values of auto correlation. And that is what we intend to plot and analyze. As far as cross correlation is concerned, the idea is to multiply 2 different rows or 2 different codes. So, to say and then multiply it element by element and then summate up which will give us the cross correlation. So, we will also see is to what peculiar property the Hadamard set of codes follows. In fact, if it is straight away look over here at cross correlation value across that is between any 2 codes at lag zero; that means, without shifting it will straight away be this product of this will be equal to minus 1. So, this product gives you minus 1. This product also gives you minus 1. This will be plus 1 and this will be plus 1.

So, take any other 2 rows as well. For example, we take these 2 rows. So, we see that this is plus 1, this gives us minus 1, this again gives us plus 1, and this this is plus 1. This will be minus 1, this will be minus 1, this is again minus 1 finally, and this will give me plus 1. Of course, I am assuming that plus here minus here stands for plus 1 and minus 1 respectively. So, what we notice that, the summation of this if I sum over all this, then the result is equal to 0. Similarly, here also we can say that there are 4 1s, 4 minus 1s and summing up over this entire range of product value is gives us 0.

So, one thing is quite apparent and you can try it out across any 2 rows from this given Hadamard matrix somebody you realize it that cross correlation at 0 is equal to 0. So, we see 2 important things that we need to establish over here. One is that auto correlation at lag 0 is equal to the length of the code. So, this was discussed first. And the cross correlation at lag 0 turns out to be equal to 0. So, these are 2 properties that we see straight away. And the rest of the task now is to verify this. And also find out auto correlation and co cross correlation for the various values of lag. So, and then plot it. So, let us have a look at it, but before that let us see the program that we use in order to plot these values.

(Refer Slide Time: 12:58)



So, this is the first MATLAB code in the series of tutorials for implementation of spread spectrum system. So, I will be a little bit more comprehensive in my explanation. So, we first declare the length of the code we store it in a variable. So, in this code we have chosen a length of 8. And we generate the code matrix using the function Hadamard, which is library function in MATLAB. And it takes parameter of length. So, if we specify length it will generate a matrix, which will have the number of rows equal to the length.

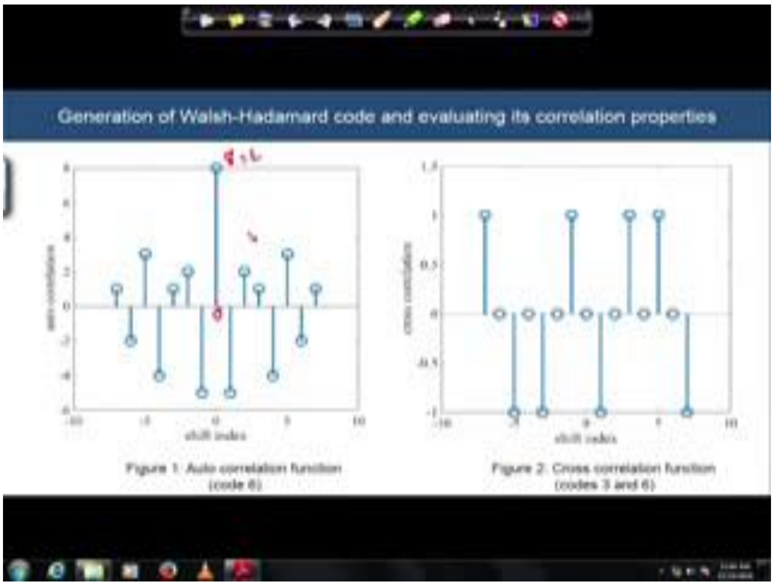
So, in this case we will be in a position to generate a Hadamard matrix that is a matrix with Walsh-Hadamard rows corresponding to different Walsh-Hadamard codes except the first row which I said in the all one row. And in this case the matrix will have a I have a number of rows equal to 8. So, we assign value of which is a basically an integer, and we have a variable which we call as a code index. So, we can actually assign a specific value to this variable. And we were going to make use of this in order to select a certain row from this Hadamard matrix.

For example, in this case we have set the code index to 6, which means I will be choosing the 6th row of the Hadamard matrix, in order to plot with an auto correlation or my cross correlation as the cases. So, line number 14 we see that we use another inbuilt

MATLAB function which plots the correlation of it is argument. So, if there is only one way of argument given, then it will plot the auto correlation and when we have 2 values or 2 arguments then it will cross plot the cross correlation between the 2. So, in the first case we are interested in the in plotting auto correlation of any given row that is any given Walsh code from this Hadamard matrix. So, we choose that as row number 6, you could change this and try out the plot, but in this case we choose the 6th code. And further for cross correlation we choose we need 2 codes. So, we choose row number 3 and row number 6. So, make sure that code index should not be 1, because as I said the first row is the row of all 1s, except a special case, but otherwise it is all 1s. So, we find the cross correlation by once again using the same function, but now we have 2 arguments.

So, that is choosing the row number 3 and row number 6. We need to plot the values of this auto correlation and cross correlation. So, therefore, we need to assign index, a time index basically or a shift index to be most specific in this case. So, we do that by generating x axis index, from length auto core which will give you the overall length of this correlation. So, we know that if I have 2 entities, which have lengths of m and n respectively then the correlation length will be m plus n minus 1. So, therefore, that will be the length of this auto and cross correlation are results or vector. And then we stem which is a descript plot of this auto correlation versus the shift index which will be on the x axis. So, this commands from 25 or rather from 25 to 31 or in order to plot the figure.

(Refer Slide Time: 18:08)



So, let us see what the plot is. So, for code number 6 which we have chosen the auto correlation function looks something like this. As we can see at shift index 0, this is shift index 0. So, for shift index 0 we have a value of 8, which is equal to the length of the code. That is what we had established in the previous slide that auto correlation at shift 0 should be equal to the length. And then the rest of the values of auto correlation will depend upon the code that you have chosen in order to.

So, all codes will not have the same auto correlation plots as you shall see, but the importance of having a good auto correlation property, and when I say good auto correlation property it means, the lag at lag 0 you should have high magnitude of auto correlation. And that should gradually decrease or rather should be very small for other lags. As far as applications such as synchronization is concerned. Because what we are trying to do is we in synchronization as we are trying to find out our synchronize 2 entities. And so, for the desired signal if you in a spread spectrum system as you are aware, want to synchronize the incoming signal with the code that is being locally generated at the receiver, then usually we compare the auto correlation at lag 0 with some threshold.

So, in order for us to meet that requirement the auto correlation at lag 0 should be as high as possible. On the other hand, sometimes it is also desirable to have a high value of auto correlation at 0, and reasonably high value after just after 0 as well. Because sometimes in cases of lots of synchronization the second value also is important in order to detect the signal and synchronize it. Nevertheless, usually the case is that we need the very high auto correlation at 0 lag, and low values around it on the positive and negative side.

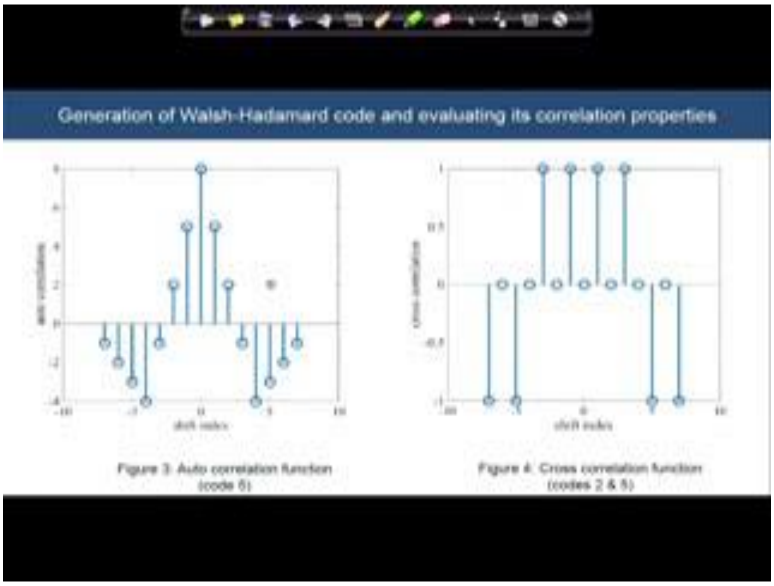
Usually such codes which give you such good auto correlation properties as we are aware, are we have the p n sequences you might have learnt about maximal length sequences pseudorandom noise sequences, and also codes such as barker sequences. So, they are having codes which have particular auto correlation properties, but then the catch is that usually it is seen that, it is a compromise between auto and cross correlation properties of a given set of codes. So, codes set of codes which have code auto correlation properties it is show turns out will have bad cross correlation properties and vice versa. So, it is always a trade of our compromise and it depends upon what kind of application these codes are used for.

On the other hand, let us locate the cross correlation properties and of code 3 and 6 which we would I think we are chosen for the program and we straightaway say that at lags 0, we have a value of 0. So, that is what I told you can try it out with the other codes as well. This particular feature will remain the same, property will remain the same. And now we notice that as we shift one of the codes like in this case if we keep code 3 in the same and we shift code number 6, for different lags we end up meeting values sometimes it is 0 and at other times the value turns out to be 1.

So, this is the very interesting property of Walsh-Hadamard codes and it will sort, it is not always necessary that this will be the pattern of cross correlation between any 2 arbitrary codes. The pattern may change, but 2 things we will notice will remain the same. One is that the cross correlation between any 2 arbitrary code that lags 0 will be 0, and otherwise it may either be 0 or it turns out to be 1 or minus 1. The interesting thing is it does not go beyond this value. And that is applicable with any 2 hybrids arbitrary set of codes chosen to be Hadamard matrix. And this is important because what we desire is also and this is important in a multi user environment. Where there is going to be a lot of interference, let say is what we desire is that for good interference rejection properties or in order to discriminate between different users that is desired users and then unwanted users the codes should assign to this user should be orthogonal and moreover it should have very good cross correlation properties. When I say very good cross correlation properties it means it should have 0 value at lag 0. And it should have low values around this 0 that is for positive shifts and negative shifts.

So, here there is some kind of a cap on those cross correlation values. We see that they do not go beyond 1 which is good. And so, this is a pretty good code to be used both for synchronization as well as in a multi user scenario, because it has early good auto correlation properties. I would not say it extremely good, because in generally for synchronization. We as I said we use p n sequences or barker codes, but definitely it has early code cross correlation properties, which can be used for multi user spread spectrum communications. So, let me wind up by showing you another example wherein we change these codes.

(Refer Slide Time: 24:49)



So, we see that the auto correlation function for code number 5 looks something like this. Now we see straightaway let me go back to the previous slide that, for code 6 it is different. So, it is not necessary that all codes will have the same kind of auto correlation function. So, you can change the value of this code index and try and plot the values for different code indices, that is you can try and plot auto correlation and cross correlation. We see another example where we have codes 2 and 5, but again this plot is different from the plot for cross correlation between 3 and 6.

However, the features remain the same. And by that I mean that auto correlation at 0 it turns out is equal to the length of the code, cross correlation at 0 lag again is 0. So, those properties remain the same also cross correlation does not exceed 1 and minus 1 as you can see. So, these are the peculiar features or properties of Walsh-Hadamard codes that we in this tutorial try to understand our implement and plot. And just summarize codes with good auto correlation properties like p n sequences barker sequences and we used for synchronization.

Whereas, codes for with good cross correlation properties like in this particular case can be used in order to discriminate between multiple users in spread spectrum communication system. So, these conclude tutorial 3. And in the next tutorial we will try and make use of these codes and specifically the Walsh-Hadamard code in order to evaluate the performance of certain spread spectrum modulation systems. So, in that sense this tutorial assumes importance, because we want to make use of these codes throughout the remaining tutorials for generation of bitrates specifically for performance evaluation of spread spectrum systems.

Thanks a lot.