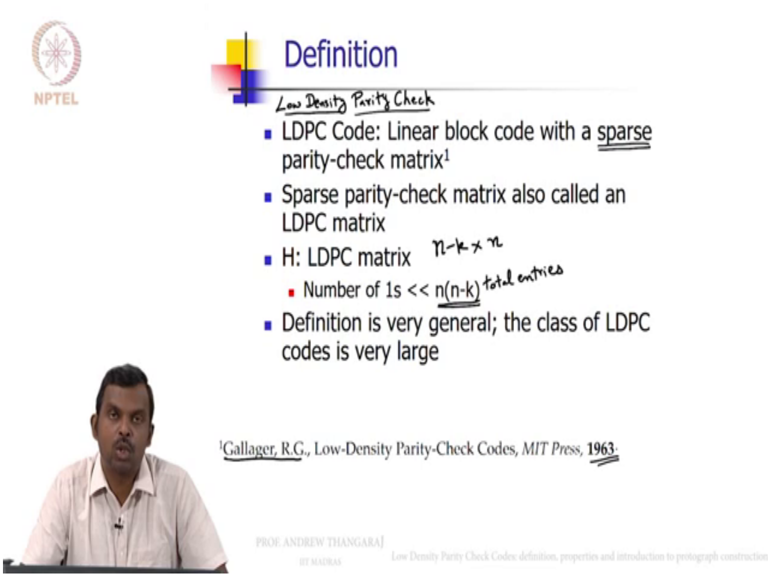
LDPC & Polar codes in 5G Standard
Prof. Andrew Thangaraj
Department of Electrical Engineering
Indian Institute of Technology Madras
LDPC Codes in 5G: definition, properties, & introduction to protograph construction

Hello and welcome to this lecture, we are going to start at the LDPC codes in the 5G standard so far we have been looking at background on linear block codes, the basics definition, so interestingly, that is all the background we need to define LDPC codes, we do need to do anything more, so in this lecture, we will define LDPC codes and in particular we will also look at the definition of the LDPC codes in the 5G standard. It is easy enough to define, it is just matrix definition and we will do that in this lecture.

(Refer Slide Time: 0:52)



Okay, let us get started, so what is low-density parity, LDPC code? It is low-density parity check code. Okay, so what is parity check? Any code like I saw as a parity check code, it has a parity check matrix, so we basically define his LDPC code through a parity check approach, we will pro specify the parity check matrix for this code. Okay, so this why is called parity check code and what is low-density, low-density means when you have a binary matrix, remember a parity check matrix is a binary matrix, it has zeros and ones and when someone says it is low-density basically, they mean it is sparse, so what is sparse? Sparse means there are many more zeros than ones. Okay, there are very few ones. Okay, the number of ones has to be considerably smaller and any time you have a parity check matrix which are a lot of zeros and very few ones, you have a low-density parity check code.

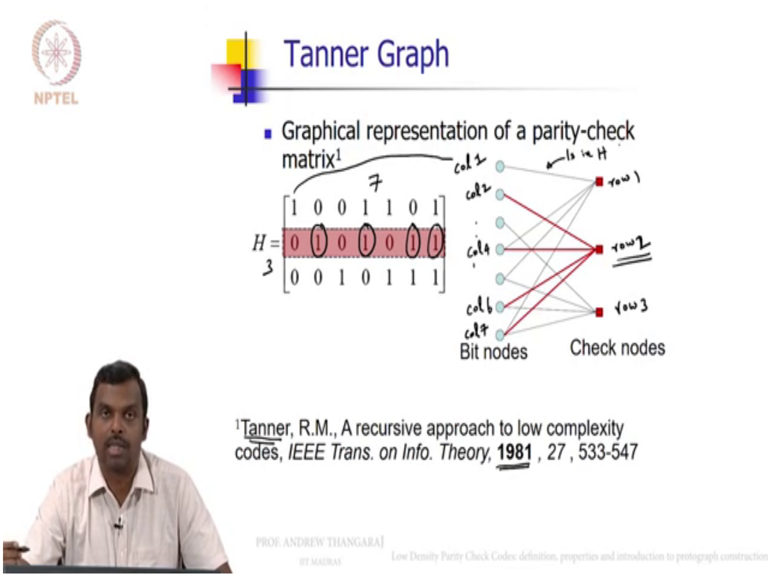
So it is a very generic definition and this looks like it is sort of week, you know mean maybe you can take one or two ones here, there and do it but this is not how one needs to do it, we have to be little bit more careful in specifying the exact sparse matrix, we will do that in this later on, as we go along in this lecture, but for now, the generic definition is that okay, low-density parity check is a parity check matrix with very few ones, so usually you think of it as every column of the parity check matrix will have very few ones and the whole matrix is made up like that. Okay.

So here is a, so remember the matrix is a N minus K cross N matrix right, so this is the H matrix, now what is this notion of very few, few in comparison to what, few in comparison to the total number of entries in the matrix, now N into N minus K is the total number of entries in the matrix. Okay, and the number of ones should be considerably lesser, so usually instead of having N into N minus K you will have you know, three times N, four times N or two times N. Some constant number into N small number into N. Okay, so that is the idea behind this definition, so for instance to give you typical numbers. If K is 1000 and N is 2000. Okay, you have 1000 by 2000 matrix. Okay that is got like, lots of entries 2 into 10 power 6 number of entries, now the number of ones is going to be 2000 into 3 say 6000. Okay, so that it, so that is much much smaller compared to the total number of entries such a matrix is called a low-density parity check matrix and that can define in LDPC code.

Okay, so the definition goes back to Robert Gallager, so he is the person who invented this codes back in the 60s. Okay, so it is a very old code in some sense. Okay, and at that time I think people did not really have the technology or realise the importance of this codes and recently in the 90s, this codes have been reselected and today, so much researches been done and it is made into the 5G standard for instance.

Okay, so we will not look at the theory of LDPC codes and all that very deeply in this class. Like I said, our focuses on decoding and building decoders, our focuses on the 5G codes. Okay, so we will simply accept the codes in the 5G standard as the LDPC codes we work with. Okay, so in this lecture, I will mostly give you definitions for the codes in the 5G standard. Okay, but before that, there are couple of concepts associated with the LDPC codes which is very important.

(Refer Slide Time: 4:41)



The first is that of a Tanner graph. Okay, so turns out a Tanner, back once again in the 80s looked at this codes and came up with the nice pictorial notation, which captures the essence of what happens in decoding and all that, captures the code very well. Okay, here is a parity check matrix. Okay, so you might look at this and say, maybe it is not really low-density, but this is just an example parity check matrix to illustrate the Tanner graph idea. Okay, so you have a parity check matrix here it is a 3 cross 7 parity check matrix and you can see it has ones and zeros. It has 1 here 1101, this is the parity check matrix of a code.

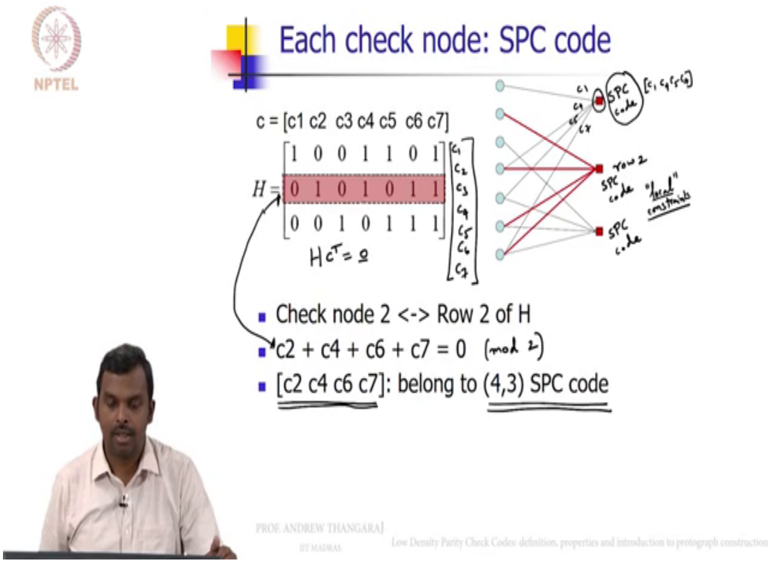
Okay, now, there is a Tanner graph associated with a parity check matrix, so the Tanner graph is nothing but some sort of a graphical representation of the parity check matrix, how do we do this, now when you go from the matrix to the Tanner graph. This is what you do. Okay, so every column. Okay, maybe I should write the note here. Okay, so this first note here, the blue note on the left side represents column 1. Okay, so this column is represented by that note, next one is column 2, so on. Okay, so this is column 7. Okay, so for every column we put a circle which is coloured in blue in this picture, but in general let us say we put a circle for every column, now every row we put a square, this is row 2, this is row 3. Okay, and these are the two types of nodes in the Tanner graph. Okay, graphs usually have nodes and edges, nodes are connected by edges in the graph and in this particular type of graph we have two types of nodes, one is what is called the bit node, it represents the columns in the parity check matrix, the other is what is called a check node, it represents the row in the parity check matrix. Okay, so to distinguish between these two types of nodes we draw circles for bit nodes and squares for check nodes.

Okay, and how do we connect, let me take row 2 is an example, now row two in the parity check matrix has connections to column 2, column 4, column 6 and column 7. Okay, so that is why, the row 2 node here is connected to column 2, column 4, column 6 and column 7, so the edges correspond to ones in the parity check matrix. Okay, wherever you have a 1 in the row which column it has a 1 you connect that row to that column. Okay, so it is very easy and straightforward way to associate a graph with a binary matrix, this is quite standard in different areas and in the (())() coding area, this is called a Tanner graph after the person who came up with this notation.

Okay, now this Tanner graph is very useful, particularly when you look at decoding for LDPC codes, most decoders are described on the Tanner graph, a lot of properties of the code is studied on the Tanner graph, etc, for us it will not play too crucial role, we will worry too much more about matrix as such, but it is good to know Tanner graph because they show up a lot in the terminology for LDPC codes and if you are interested in studying analysis of LDPC codes you need to know about Tanner graphs.

Okay, so hopefully this is clear, later on I will reinforce this idea with some more examples, but this is the Tanner graphs, so with every parity check matrix or in general any binary matrix, zero on matrix, you can associate a Tanner graphs right, you can draw some nodes for the columns, some nodes for the rows and connect the rows and columns, wherever there are ones.

(Refer Slide Time: 8:31)



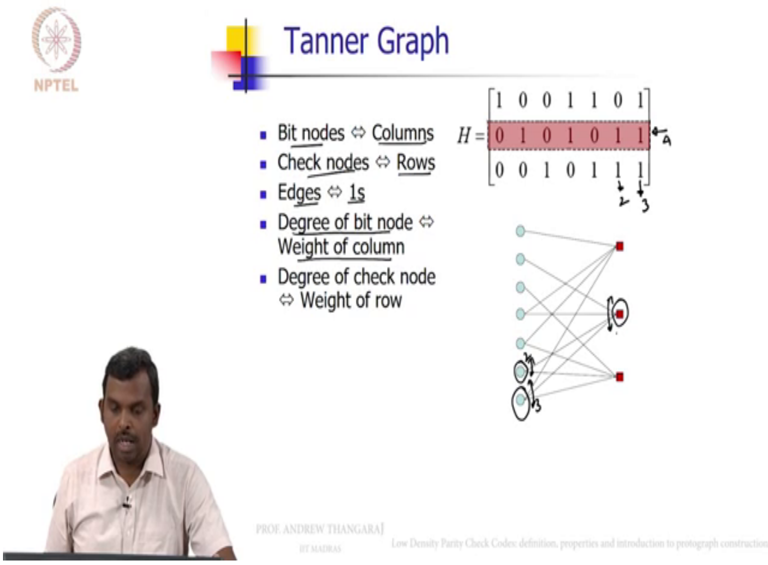
Okay, now what is special about the Tanner graph and what is the connection to the single parity check code. Okay, so this is something that is very important. This will play a crucial role in all of our decoders, now if you look at a code word C. Okay with generated from a code with parity check matrix H, what do I know? I know H times C transpores equals 0. Okay, so the vector 0, three 0. Okay, so any code words satisfies this condition, H times C transpores equals 0. So, now let us once again focus on the second row here, row 2, which corresponds to check node 2 okay and this graph, now you look at H time C transpores is 0, I have to multiply H with C on the right, C1, C2, C3, C4, C5, C6, C7 okay, when I do this product. What will the second row tell me, the second row times, this column will give me this equation, do you agree, you take that second row and multiply with this column, I am going to get C2 plus C4 plus C5 I am sorry, C2 plus C4 plus C6 plus C7 equals 0. Okay, so of course this is all modular 2 or exhort.

Okay, so in short another way of viewing this is that instead of looking at a whole code word of length seven, if you look at only these four bits, what are the four bits I am going to look at C2, C4, C6 and C7 only these four bits. Okay, what is the condition on those four bits imposed by the second row, it says that these four bits have to exhort to 0, which is, in other words, that is four bits C2, C4, C6 and C7 belong to the 4, 3 SPC code, single parity check code, right, this four bits have to exhort to 0. Okay, so each of these nodes here is a SPC code. Okay, so, in what sense is the SPC code define, you go to this node, look at the bits that this connected to, okay, so this is what, this is C1, C4 and then C5, C7 okay, so what is this SPC code? C1, C4, C5, C7 are in the 4, 3 SPC code, enforced by this check node.

Okay, so every check node in the Tanner graph enforces a single parity check code on the bits it is connected to. Okay, so these, these kinds of constraints are called local constraints. Okay so you have a code word which is global, you know all the bits together satisfy every parity check in the constraint. Okay, now the number of parity checks could be many right, so if you think of a 1000 by 2000 matrix, you have 1000 parity checks, enforcing all of them at the same time may not be a very easy thing to study on the other hand, if you go on focus on one check node, one particular check node and particular, if this is a sparse matrix then, that check node is connected to very few bit nodes. Okay, and what this local constraints says, there are few bits in the code word which satisfy the SPC constraints, single parity check constraints and we know how to decode the single parity check constraint very well.

Okay, so this is one of the crucial ideas, which is at the corniest own of the success of the LDPC codes. Okay, so to decode them, it is enough if decode these SPC constraints, the single parity checks cleverly in some order and combine them very smartly. Okay, so how to do that is what we will see in the decoding section, but for now in this lectures, we will focus on how the parity check matrices are defined in the 5G standard. Okay, so hopefully this part is clear, this idea of associating a single parity check code with a every parity check node is very crucial and will come again and again to this. Okay, so another way of thinking about it is every row in the parity check matrix enforces one parity check constraint right, on the positions of the ones. Okay, wherever you have ones there, that row of the parity check matrix enforces a parity check constraint and we will do decoding row by row. Okay and cleverly combine how we do the processing and end up with very implementable decoders which also have very good performance. Okay, so that is the crucial idea in some sets.

(Refer Slide Time: 13:20)

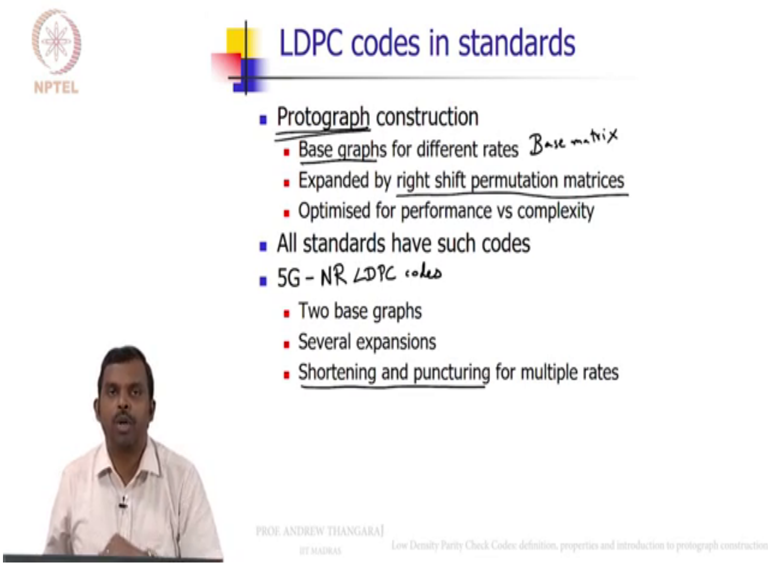


Okay, so let us also look at the Tanner graph I want to summarise some of the notations and this I am sure you will see in many, if you read a book on LDPC codes, this terminologies will come again and again, once again I want emphasise will not doing a measure theoretical study on LDPC codes here will be focused on implementation details, but nevertheless it is good to know some of these terms, in case you come across them later on in your studies.

Okay, so the bit nodes in the Tanner graph correspond to the columns, check nodes correspond to the rows, I mentioned this edges correspond to ones, now this degree of a bit node is weight of the column, so know what is weight of the column, this has weight 3, okay, number of ones, weight of the columns is number of ones and that is 3 here, that is 2 here and if go and look at the corresponding node here, it has 3 edges coming out of it, right, this has 2 edges coming out of it, so the number of edges coming out of a node is called the degree of the node in the graph, so the degree of the bit node is the weight of the column.

So likewise the degree of the check node is the weight of the row. Okay, so this one has weight 4 and the degree is 4. Okay, so most of the analysis and design of the LDPC codes focuses on the degrees. Okay, so turns out the degrees control the performance of the code. Okay, so you have to do an analysis and study of it and like I mention once again, we will not be doing analysis in this class. Nevertheless, this degree is an, degree distributions are very crucial. Okay, so that is a summary of what happens in the Tanner graph.

(Refer Slide Time: 14:51)



Okay, so now will jump into LDPC codes in standards. Okay, so in the standards it turns out 5G it is not the first standard to incorporate LDPC codes, they have being there for a long time now and many standards which may be worth not that popular, the most popular I would guess is the DVB S2 standard which has an LDPC code, Digital video broadcast standard, but now the 5G standard has made them much more prevalent than before. Okay, so it turns out the LDPC in standards are almost always Protograph constructions.

Okay, so this is not photograph in the sense of taking pictures, this is protograph. Okay, so what is a protograph, protograph is sort like an example graph which you repeat to make a bigger graph. Okay, so the way they would define a protograph is as follows, so protograph construction is basically a way of constructing your parity check matrix, remember the parity check matrix is usually big binary matrix right, so 1000 by 2000 or some such matrix, how do you specify that, how do you specify whereas the ones are right, how do you specify the Tanner graph.

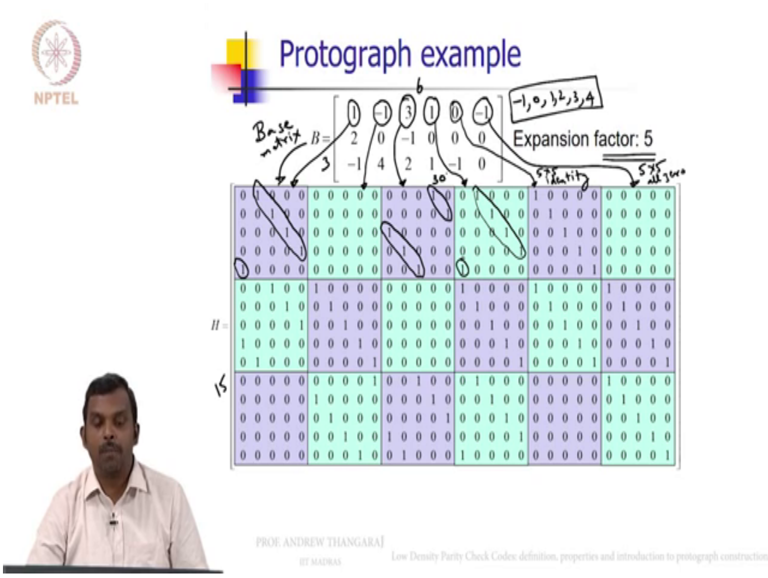
So for that the standards always use this protograph construction, there a lot of theoretical reasons for why protograph constructions are very good, but let us just look at how they are defined and how they are specified. Okay, first thing will specify is the base graph or a base matrix. Okay, and base graph or base matrix, so this base graph is also called the protograph base matrix is the proto-matrix, so speak. Okay and this base matrix is expanded or blown up. Okay, to get the actually parity check matrix. Okay, so this expansion is by what is called right shift permutation matrices, I will show you precise examples in the next few slides, but you take a small base matrix and then you blow it up. Okay, replace each base matrix entry with a larger matrix to get the overall parity check matrix.

So you can see this sort of controls complexity a lot, if you have a very large matrix specifying where ones are, can take a lot of memory, now mean it, it will be not too much, but still it takes memory and you want to minimise all of that, so to do that, you can do this base matrix which gets blown up, so you only need to specify the base matrix and how much you blow it up by, okay and this blowing up a also controlled by this right shift permutation matrices, so you do not have to specify too much that.

Okay, so overall game is to optimise performance in complexity and the theory is quite deep that we will not go too much into it and only point is all standards have such codes. Okay, maybe sometimes the expansion is not by a permutation matrix, maybe it is by something else, but all standards have such codes. Okay, now specifically in the 5G standard, these are called NR LDPC codes. Okay, think NR stands for New Radio. Okay, so there are two base graphs, there are defined and there are several expansions, the expansions can be by as short as 2 by 2 to 384 by 384, okay, so it is a very big expansions also possible and to achieve a lot of rates, they do something called shortening and puncturing, I will not talk too much about it and this lecture, later on I will come back to it and mention it, when dealing with these multiple rates and so-called rate matching. So, the rate matching is a very big deal in a standards, people will talk about that a lot, so I will briefly mention that is not very critical for us. Okay, so the most critical thing is there are two base graphs, and there are several ways of expansions.

Okay, so what is this base graph and expansion? Let start by looking at a very simple, small example where I will be able to show you the complete thing of what is going on and then I point out to what happened in the standard.

(Refer Slide Time: 18:49)



Okay, so here is an example, this is a base matrix. Okay, so an example of a base matrix. Okay and the expansion factor is 5. Okay, so if the expansion factor is 5, the base matrix will have entries -1, 0, 1, 2, 3, 4 okay, so these are the possible entries of the base matrix, so you can see that is what the entries are, there are a few minus 1 few 0, 1, 2, 3 and 4 okay, here and there, just they are thrown in, is that okay, so this is a just an example base matrix I am not claiming that it is particularly good base matrix, it is just an examples, may not be too bad actually.

Okay, so this is an example of a base matrix that we are going to use to illustrate what happens in this expansion. Okay, so now the base matrix itself is not a parity check matrix, you can see clearly it is not binary, it has minus 1 entries and 3 and 4 and all we cannot do anything, so from the base matrix B, we will generate this expanded matrix. Okay and this expanded matrix you can see it is clearly a binary matrix and it can be a parity check matrix for some code. Okay, so this is how the parity check matrix is actually specified. Okay, so have shown the expansion, I will illustrate it with this coloured blocks, of course the coloured blocks are just there for our convenience, the actual matrix is only zeros and ones.

Okay, now how do we see these expansion, let me show you what happens to each entry here. Okay, so each entry in the base matrix gets replaced by a 5 cross 5 binary matrix is that okay, so each entry replaced by a 5 cross 5 binary matrix. Okay, now if the entry is minus 1, you replace it with a 5 cross 5 all zero binary matrix. Okay, this what happens here, so minus 1, got replaced by a 5 cross 5 all zero. Okay. Is that right, what about zero? The entry zero gets replaced by 5 cross 5 identity matrix. Okay, what about the entry one? The entry one get replaced by a 5 cross 5 identity matrix shifted right by one position. Okay, so this is the right shift permutation matrix I was talking about, it is a 5 cross 5 matrix, but you start with the identity and then you shift it right by one position, of course the shift is circular, so the last entry will go of all the way to the right. Okay, you can see the shift happening here. Okay, so it is been the identity matrix has been shifted right to obtain this ones here, right, and then the last one got shifted out to this side. Okay, you can see the shift by one to the right.

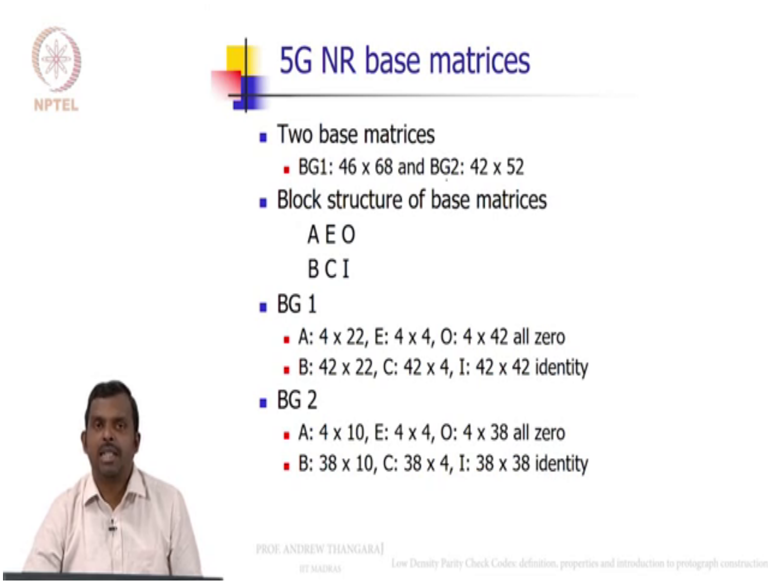
Okay, what about 3, any other value 3 is the same thing as what happened to 1, you take the identity matrix and then you shift it right by 3 positions, shift all the columns, when I say shift mean the shifting of the columns. Okay, so you take the identity matrix and rotate shift the columns by 3 positions. Okay, so you can see what has happened here, so you will get 1, 1 here and then that 1, 1, 1 here. Okay, so then you have minus 1, which gets replaced by all 0 and then a plus 1, which get replace by identity matrix shift at right by one position.

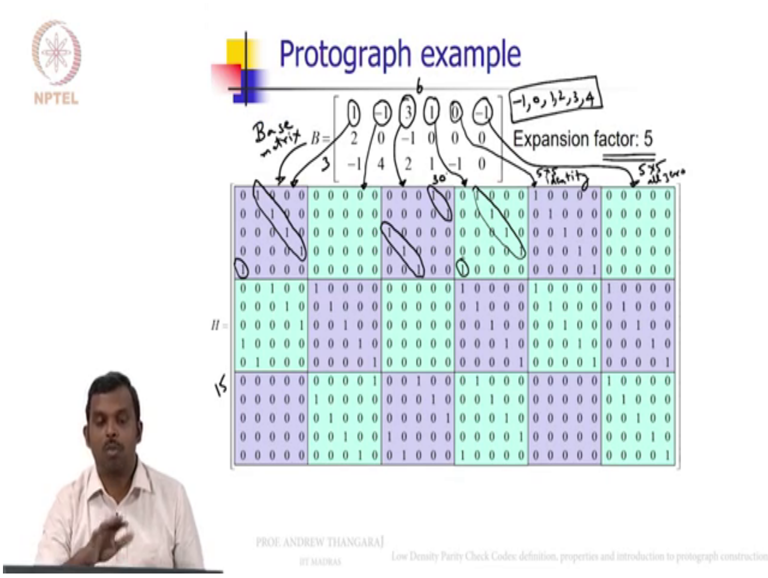
So this is what happens to every entry in the base matrix. Okay, so your original base matrix was 3 by 6 after you expanded, after you blow it up is you are going to get 15 by 30, right 6 into 5, 3 into 5, a 15 by 30 matrix. Okay and you can see, 15 by 30 matrix is quite sparse, right so very few ones because they using identity matrix and all zero matrix. Okay, and shifting etc, so not too many ones in the overall final parity check matrix and this is what happens in every row, in the second row also the same things is being done, third row also the same thing is been done, you can look at it and convince yourself that it is true.

Okay, so once again, let me repeat, the base matrix will have entries, integer entries minus 1 up to the expansion factor, expansion factor here is 5 in the 5G standard the largest expansion factor is 384. Okay, so the entries will go from minus 1 all the way till the expansion factor minus 1, minus 1, 0, all the way till 4, if the expansion factors is 5 and what you do when you expand or blow it up every entry of the protograph is replaced by 5 by 5 identity matrix, so the 5 by 5 all zero matrix or the 5 by 5 identity matrix shift at right, rotate shift at right suitably as depending on a number that you have is that okay, so this is the way in which you blow it up and you get a larger matrix from the smaller protograph for a base matrix.

Okay, so this is used in every standard that defines LDPC code in one way or the other. That is easy to specify, it also has very good properties, code properties and analysis is possible also to prove and design, you can design all this very, very well.

(Refer Slide Time: 24:47)





Okay, so what we are going to see in the next lecture is the actual base matrices in the 5G new radio standard, but let us close this lecture for now, we will stop with the example of the protograph matrix here in the next lecture, we will see the actual base matrices in the 5G standards, how they are defined. It is just mostly specification, but I quickly run through that and then I will point you to the actual base matrices themselves. Okay, let us stop here in this lecture.