Communication Networks

Prof. Goutam Das
G. S. Sanyal School of Telecommunication
Indian Institute of Technology, Kharagpur

Module - 02
Circuit Switched Networks
Lecture - 05
Space Switching Architecture

Ok, so far in this course, we have probably discussed different kinds of networking over the past few decades that has been evolved. So, that is something we have already discussed. So, now, we will actually go inside the actual content.

So, we will start with the first thing that we have discussed, which is the Circuit Switched Network. So, we need to first understand how the circuit-switched network was designed, what the aspects of it are, and how this was realized. So, these are things we will discuss. Once we do that, then we will probably progress to other parts of networking, ok.

So, in a circuit-switched network, you will see that there are two things that are very important, one is multiplexing, and the other one is switching, ok. Of course, then synchronization is another part that we will be trying to discuss.

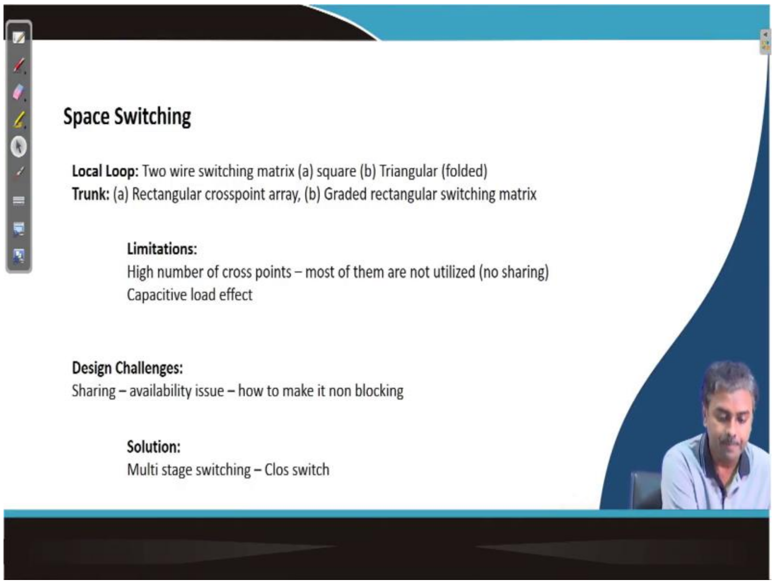
So, today probably in this class, we will be trying to discuss one of the specific switching technology, which is called space switch. So, we will see what this is and how it is realized, ok.

(Refer to Slide Time: 01:28)

So, the concept under circuit-switched networking is probably, we will be covering space and time switching. So, today mostly we will be covering space switches and most of their design aspects. Then, we will talk about synchronization, which is a very important aspect of circuit-switched networks. It has to be precisely synchronized, why that is required, we will talk about that also later on. So, that is something that we will be discussing next.

And then finally, we will be discussing multiplexing, and various multiplexing schemes which are there. And, of course, we cannot discuss all of them one after another. So, in between, there will be a mix of these things. So, multiplexing will come inside the space of time switching. So, we will see those things. But we will try to keep the concept as much clearer as possible, ok.

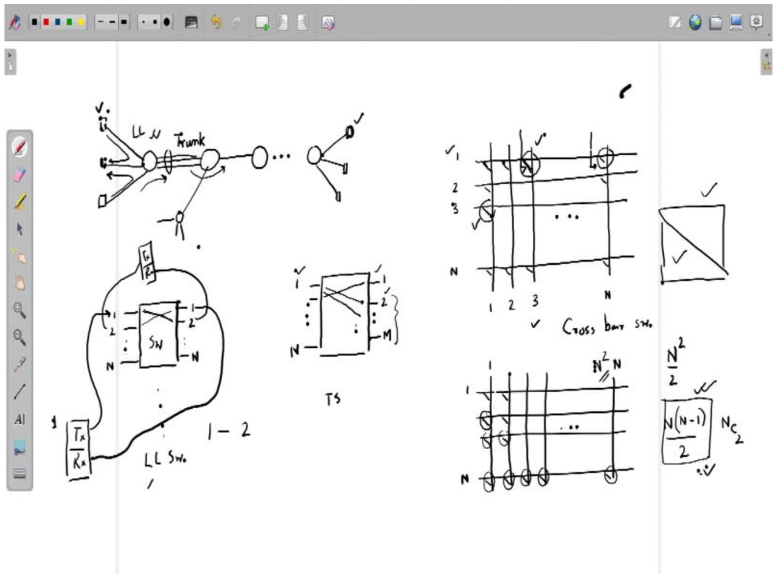
(Refer Slide Time: 02:14)



So, let us start talking about the space switching. So, let us see what is actually space switching and how it is being realized. So, two kind of space switching we will talk about, we will talk about one for local loop and another one for trunk.

So, we have already probably discussed. So, let me again give some examples of that. So, let us try to think about what is actually local loop. So, we have said that maybe there might be some subscriber which are over here. So, let us say some subscriber which are over here, ok.

(Refer to Slide Time: 02:49)



So, first, what they do is they try to connect to the local subscriber loop. And from there, we have already shown this diagram that it goes to some trunk where multiple such local subscriber loops connect to, ok. And then it goes via multiple trunks in between, and then finally, it gets connected to again some subscriber over here, ok.

Now, whenever you are switching over here in this local loop. So, what do you require? One is that you have to go out; that is a different issue. But if you have to communicate within themselves, so it is switching among themselves, ok. So, any combination might be there. So this is called the local loop.

And when you are actually switching, either going from here to this trunk, so this is the trunk portion. So, either you go from local loop to trunk or trunk to trunk. This switching when you are doing that is the trunk switching, ok. So, we will talk about the specific details of these things.

So, when you are in the local loop, what you need is something like this. So, you need a switch where let us say, N number of subscribers are connected, so 1, 2, and then they need connectivity to means among themselves. That means, if in the switch, if I see the output let us say 1, 2, all those customers are at the output, also they are connected. And what is generally being done so every user will have a transmitter and a receiver. So, it is called a transceiver. So, every user will have this block.

Now, what happens, is this transmitter I will connect to this thing suppose for user-1, and this will be taken out and connected to the receiver and so on for every user. So, for the user- 2 also, the same thing will be happening. So, his transceiver 2 will be connected, and so on. There will be N such users will be there, ok.

So, now every user who wishes to connect to other users what you need to do is his transmitter, and the other guys' receiver has to be connected and vice versa. So, this is something which you will have to do. So, that means, basically, if now user-1 wishes to connect to user, let us say 2, then you need to give connectivity through the switch like this. These are 1 to 2 and probably 2 to 1, something like this, ok.

If it is a means half duplex kind of thing, ok. So, if it means a complete duplex, then what will happen? One connectivity is ok; that carries the data from user 1 to 2 and vice versa. So, whatever it is, we need to give this spatial connectivity, and that is the switch we are talking about. So, that is the local loop switch. So, the local loop switch will probably have as many input ports as many numbers of output ports, these input ports and output port are actually corresponding to every user, ok.

So, each pair of input and output ports actually corresponds to 1, 1 user. So, if we say user-1 it will have 1 input port and 1 output port. Of course, through the switch, he will never be connected to himself. So, that is stupidity that you do not go all the way to the switch and then connect to yourself, that will not be done. But to all other users, you need to give connectivity, ok. So, this is one kind of switch that we will be talking about. That is the local loop switch. So, this is the local loop switch.

In the trunk, what happens? So, the trunk is a little bit different. If you see this example. So, let us say there are multiple trunk links that are going out, ok. Now, these users are 1 to N; so now, let us think about the switch for the trunk. So, users 1 to N wish to connect to this trunk link. So, he wishes to basically these users want to go out. So, if he has to go out, he has to take one of these links, ok. Any link is ok because with any link, he will be able to go somewhere out. So, basically, in the output of the trunk, there will be multiple trunk links. So, that might be different 1, 2 up to let us call M, ok.

Now, if any user wishes to get to this trunk, one of the links which are available right now can be given. So, basically, it does not mean that user; over here, what was happening? If user-1 wish to communicate to user 2, only one possibility of linking is possible. So, 1 to 2 connectivity you will have to give. This connectivity you will have to give because otherwise, user 1 will not be able to talk to user 2.

But over here, whenever you are transferring something for, let us say, from user-1 to some remote location over here, it does not matter which trunk you take because all of the trunk will be taking you possibly to the same destination. Therefore, you can take any of these available trunk. This is the flexibility the trunk switch will have. So, the trunk switch is a little bit different from the local loop switch.

So, now trunk switch, what I can do, is any of the available links right now, ok, some of the links might be occupied. Suppose user-2 is already talking to means somewhere outside, so it has to be connected to the trunk. So, he has already given connectivity to user-1 probably, right.

So, that is already given; now, what we need to do is now user-1 wishes to make connectivity. So, now, all the available M minus 1 connectivity, any one of them he can take. So, I can connect him to 1, sorry to 2, 1, of course, I cannot because 1 is already taken. I can even connect him to 3 and so on up to M. So, anywhere I can give him connectivity.

Now, once user 1 is connected, now next time, if user-3 wish to get connectivity now this connection is taken, and I have probably connected user 1 to 2. So, that also is taken. Now, the rest of the things I have to think about. So, this flexibility we have in trunk switching. So, these two kinds of space switching, as you have seen now, we will have to realize, ok. How can we realize these things?

So, let us first try to see this particular local loop switching, ok. So, in local loop switching, what can we do? We can do a switching matrix. So, how is that possible? I will just explain.

So, let us say user-1, user-2, and there are N users, right. So, I connect this, everybody has a line, and then I put a cross line, ok. So this is called a crossbar switch. So, I put a cross switch or cross line for each user. Now, whichever connectivity this possible, these things can be connected ok.

So, I can always give connectivity in these links. So that is why it is called a crossbar switch. So, it is a cross kind of switching matrix. And these are the switching elements, actually. This is where I can give connectivity from one line to another line, and this is digitally controlled. We will see how these switches can be realized, and how I control these things. So, these are generally digitally controlled from the outside, ok. So, whatever digital input I give accordingly they will be connected, ok.

So, now, what do I need to do? So, whenever I wish to connect 1 to some, let us say 3, then I will enable these things. So, this will be with the digital control, I will enable this immediately there will be connectivity which will be constructed from 1 to 3. So, this is called a crossbar switch, ok.

So, in the crossbar switch, you can now see how many elements are required. From N user, if I wish to make a crossbar switch, so clearly I can see N square number of elements are required, ok. So, if there are N number of inputs or N number of users who are trying to connect to a local loop switch, then we need N square number in this kind of crossbar switch. We need an N square number of switching elements.

The switching elements are the costly part because they need, you will see a digital circuit and control from outside. We will talk about that part of how those switching elements are realized in digital switching and how they are controlled. So, that is something we will be discussing later on.

But for the time being, we are just giving this representative diagram as if from the output control, we can connect any of the input to the output, ok. So, as you can see, there are N square numbers of switching elements.

Now, if we ask ourselves and this is called the square arrangement. So, because you have this cross N bar, N square number of switching elements are there. So, it is very symmetric. So, that is why it is called square. So, square switching can be facilitated where N square number of switching elements are required. Suppose this is the full duplex. So, one connectivity I give, 1 to let us say 3, then 1 can communicate to 3 as well as 3 can communicate to 1 via that same this one.

Then, can you see there is a redundancy over here? What is that redundancy? The first redundancy let us try to understand. There is also particular connectivity of 3, 3 to 1, I can give connectivity, ok. So, this switch and this switch these two are redundant if I make it a full duplex.

So, if I make sure that if 1 is connected to 3, then 3 can also, in the back direction means, transmit to means either data or voice to 1. So, if this is happening, so only this switch is required; this switch is actually a redundant switch. It is giving me no purpose. Any of these switches will do everything for me, ok.

So, as you can see, if I just explore a little bit, I can see that means if I just from this square one, if I make it triangular, then all other half of the triangle is actually redundant. So, if I just make it something like 1 up to N, let us say. So, if I just bring, suppose first of all let us try to see the diagonal element 1 to 1, 2 to 2 is that required; 1 will not be talking to 1.

So, basically, these diagonal elements are not required; that is pretty much sure. In the off-diagonal element, I can see, as you can see, if I take this one, then this one is redundant. So, basically, if I just take out the upper triangle, probably, and if I take out the diagonal things if I only connect the lower triangle, that is good enough for me.

So, if I just take this switching element, this, this, and so on like the entire these things, either I take the lower triangle, or I take the upper triangle, one of them. So, if this is my whole matrix, so either I take this or this, any one of them will serve the purpose without the diagonal, ok.

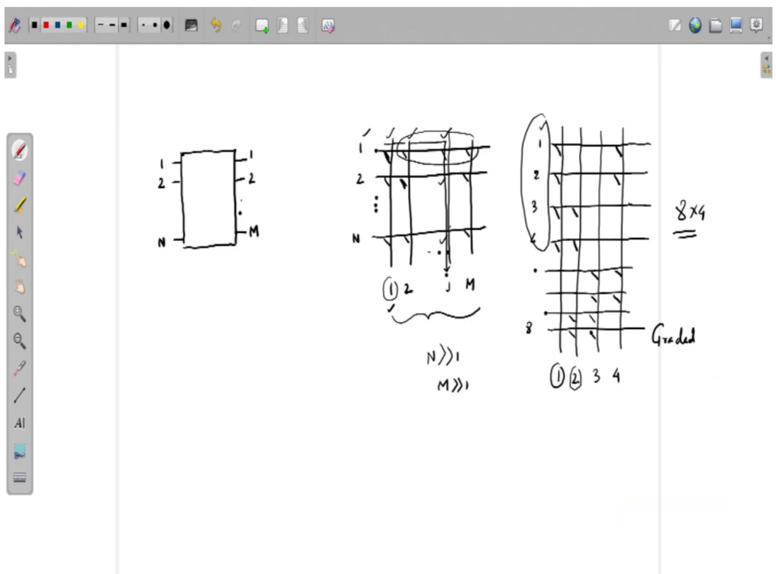
What is the number of switching elements now? I have made some reductions. So, what will that be? So, as you can see it was N square, half of them I have taken, so that must be N square by 2, ok. So, that must be the case, ok half of them if we take. But, if you see, it is just not N square by 2, because I have also taken out the diagonal 1. So, that means what will be the number? That should be N into N minus 1 divided by 2, ok.

So, if I take half, then out of diagonal half will be taken that half N by 2 you have to take out, so I get this, that is simply nothing but N C 2, as many combinations of connections you can have. This is something we already know, if there are N number of users, how many connectivities you will have to give, that is actually N C 2 because two of them will be talking to each other as many possibilities are there, that is N C 2, that many connectivities you will have to give.

In the switch also, we can see if we have to give full connectivity to each and every user; this is the bare minimum that will be required. So, that means, after reducing the switching number of switching elements, we have come to this magic figure of N C 2, N into N minus 1 divided by 2. Later on, we will talk about whether this is sufficient reduction can be reduced furthermore, and what is the disadvantage of this one, ok.

But before that, let us also talk about the trunk switching, ok. So, let us see the trunk switching part.

(Refer to Slide Time: 17:50)



Now, M trunks are there, ok. What is the connectivity? Again, I can construct a crossbar switch. At the output, I will be taking because M number of outputs are required. If I provide these connectivities, it is good enough.

Now, wherever I wish to go, so 1 suppose initially everything output everything is free. I can give connectivity to 1 by enabling this 1, ok, and so on. If now 2 comes, it needs connectivity I can give 2 to 2 connectivity. Then, 3 comes, I can like any connectivity I can give because over here I do not have a preference; any one of them is good for me, ok. So, I do not have to put it either in 1 or 2 or M, which was the local loop switching requirement, but that is not there over here.

So, over here, if I try to see what is happening? If you just try to see if there is any connectivity, I give I have so many redundant switching elements. Suppose 1 has to be given connectivity, I do it by 1, and all these other switching elements are redundant; they will further not be required, ok.

So, these are the redundant switching elements that are under-utilized; that is one thing. So, that means a lot of switching elements are not being properly utilized, ok. But you can always say, if I just give 1 to 1 connectivity then only one trunk is available for him. If he is not using that, and if anybody else is not really connected to or there is a possibility of connecting him to any other user then that will be underutilized again.

So, basically, it is a tradeoff between whether the trunk should be underutilized or the switching element should be underutilized. Which one is more costlier? Of course, the trunk is more costlier. So, that is why people have come up with these full-blown switching elements, they call it rectangular matrix or rectangular crossbar switch, right. It is rectangular because the input and output are no longer symmetric. Earlier, it was called a square because input-output in the local loop was symmetric. As many inputs are there that many outputs were there, ok.

So, over here, we call it rectangular, but in this rectangular, there is also another extra thing; that means, the output is now no longer targeted, which output I have to put, the switch does not have to decide that any of those outputs are ok. So, that gives us an extra facility. Do you know what is that extra facility? Maybe some of the switching elements I can reduce. So, that reducing let us try to give one example.

So, let us say I have 8 users, ok or let us take a simpler example. Let us take 4 users, ok. And let us say I have ok. So, for demonstration maybe let us take 8 only, 1, 2, 3, 4 up to let us say 8 number of users are there, and I have 4 trunk, ok.

So, for giving connectivity, I need some amount of means I should not fix it. So, basically, I can take all these 8 into 4 connectivity that I can do, but that is too many switching elements, ok. So, instead of that, I can reduce it a little bit. What is the method of reducing? It is called like this, I can for 1, I can give 2 connectivity, so he has 2 options. For 2, also I give 2 connectivity; he also has 2 options. For 3, I might give some other options, ok.

Similarly, for 4, I can give some other options. So, all kinds of possibilities. So, suppose 5 and 6, I might give some other possibilities. So, all kinds of shared options I am trying to give over here. As you can see, each of these switching elements or each of these trunks is shared among multiple users.

So, the first trunk is not; earlier, what was happening the first trunk was shared among all the users, and every user has the possibility of connecting him. Over here, I have made a subgroup by restricting the number of switching elements. So, what have I done? For 1, I have created among the user a subgroup of 4 users, 1, 2, 3, and 4, as you can see. For 2, I have created another subgroup. So, if 7 and 8 also I demonstrate, let us say I do this, ok.

So, as you can now see that I have created a subgroup for 1, and trunk -1, I have created a subgroup of 1, 2, 3, 4. So, these four users have the option of connecting to Trunk 1. For 2, that subgroup is different, ok. And each user also has multiple options, as you can see. User -1 does not always have to go through 1st trunk; he can either take 1st trunk; if 1st trunk is not available, he can go to the second trunk; sorry, this 4th trunk ok.

Similarly, for user-3, it is a different thing. So, with this, I have reduced the number of trunks. As you can see, I have now made it half. So, like this, there is a possibility of reducing the number of connectivity or the number of switching elements that you are putting in because you will, later on, see the switching elements are the costly costlier part. So, if a switch dimension has to be increased, so if I have to scale this switch, so N becomes very, very large, or M becomes very, very large, which will be happening in actual switching.

So, in that case, the switching elements and the amount of switching elements that I will have to put in will be dauntingly high, and I need to reduce that in my engineering design. So, these are the options for reducing things. So, this is called graded switching. So, this particular part was rectangular, which means full-blown switching elements are all given. This is called graded.

So, once I do this graded matrix, then definitely, I have to do it very carefully, who should be in which group, who should give connectivity means, and for whom I should give connectivity, so all these things have to be carefully designed, ok. So, these are the two options for, if I can say trunk switching, ok.

So, I can either take the full rectangular or I can take a graded rectangle, ok where I will be; means my target is always the same. I am just trying to reduce the number of switching elements; over here also, when it was a local loop also, I was almost trying to do the same thing. So, in both cases, I am just trying to reduce the number of switching elements.

So, now, if I just ask myself why I am reducing the switching element. One was a very obvious reason that I want to reduce the cost. Is there any other benefit of reducing the switching elements? So, let us try to see what this switching element does.

So, let us say I have this full-blown switching, ok. I want to give connectivity; let us say the 1st one I want to connect to the jth one, this is the jth one, I want to connect to jth one. Let us say all these switching elements are there. Now, what this switching element will be? They are sub-digital circuits, so they have some capacitive effect you, which means whichever digital circuit you will take, they will have those stray capacitance right.

So, each of these elements if you see how 1 goes to j, so it has to go via this route. Now, when after doing the switching, when you go via this, it has to actually cross through all these elements, all these switching elements. More switching elements will be there; it has to cross through that, and all those switching elements will give capacitive loss.

So, that means my signal will have a huge loss. As I increase the number of switching elements because in the matrix if all switching elements are there, he has to pass through all these switching elements. So, all these capacitive losses will be incurred by him and that will give the signal quality to be poor. If he has to go through multiple such switching elements or the switch dimension is very big, which will be happening practically, so then you have a problem over here.

So, as you can now see, we have two basically engineering aspects or engineering prospects that we have in mind for which I need to reduce the switching elements or the number of switching elements. One is, if I have to scale the cost of the switching element is going to be daunting. The second thing is there is this stray capacitor effect which will give me give distortion to my signal to my actual voice signal, whatever we are transmitting, and it will also give extra loss to that signal.

So, for these two reasons, we will next onwards what we will be trying to do is; is there any further way of reducing the switching element. So, that should be our major focus of the next 1 or 2 lectures, ok. So, that is something we will be targeting, towards means reducing this number of switching elements. Either it is in the trunk switch, or it is in the local loop.

You will see that the local loop and trunk switch will be a little bit different, and accordingly, while reducing them also, there might be some differences. So, that is something we will also try to appreciate in the next class, ok. So, that should be the means we will start this discussion in the next class.

Thank you.