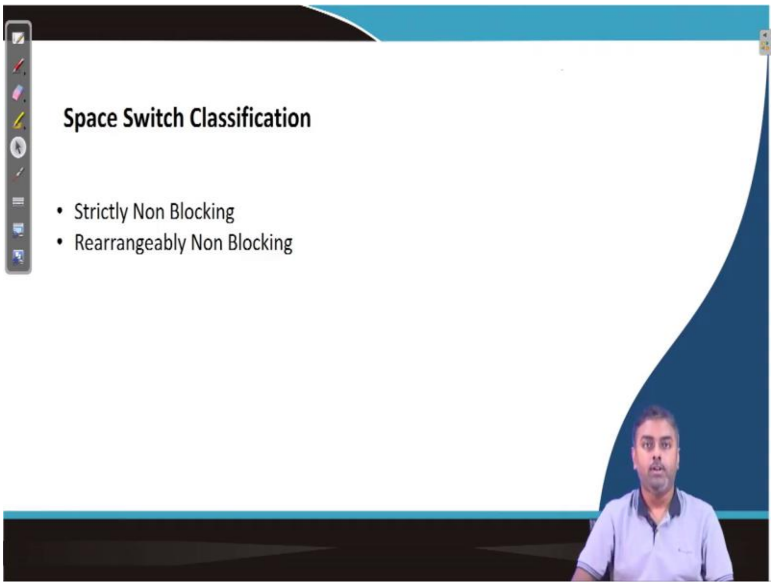
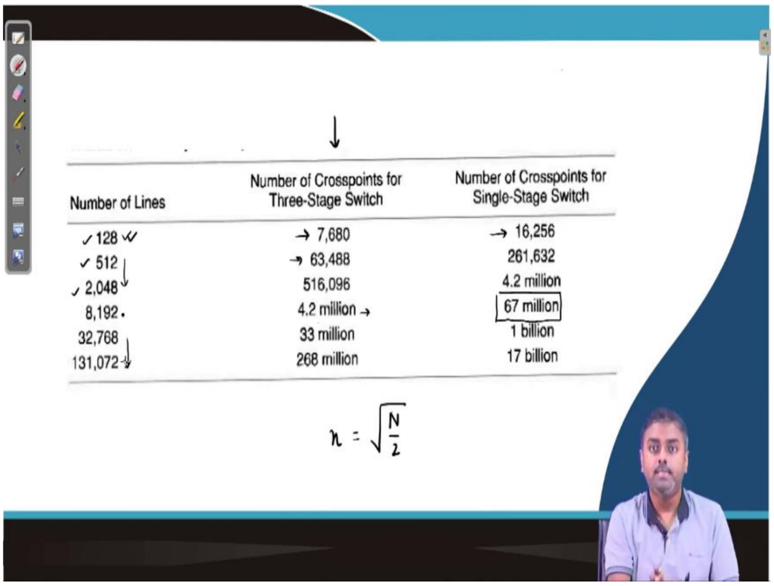
Communication Networks
Prof. Goutam Das
G. S. Sanyal School of Telecommunication
Indian Institute of Technology, Kharagpur

Module - 02
Circuit Switched Networks
Lecture - 08
Space Switching Architecture cont'd

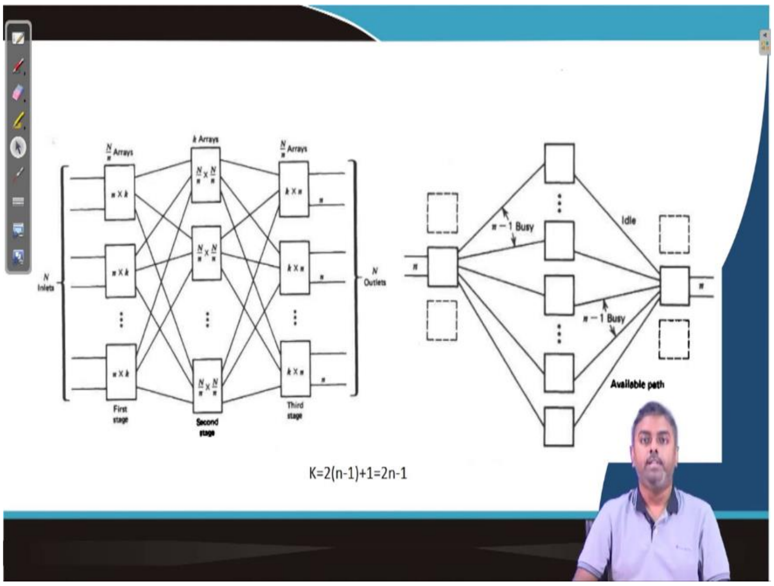
(Refer to Slide Time: 00:37)



(Refer to Slide Time: 00:44)

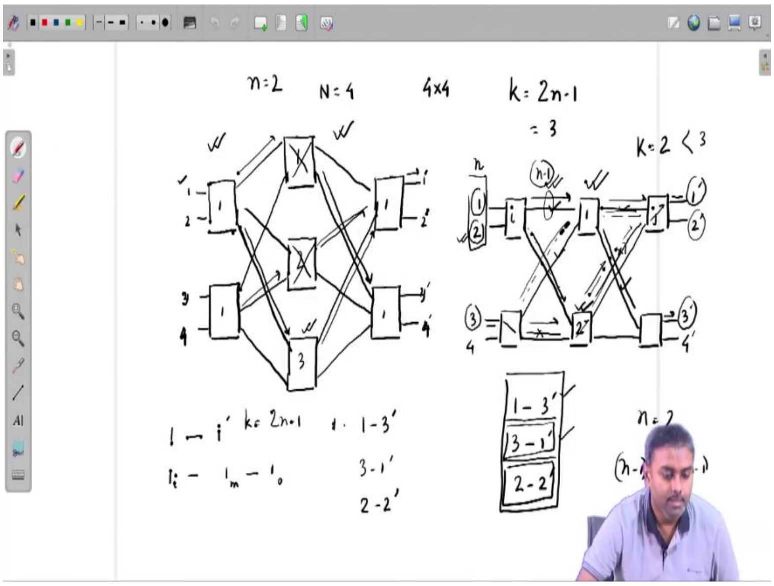


(Refer to Slide Time: 00:46)

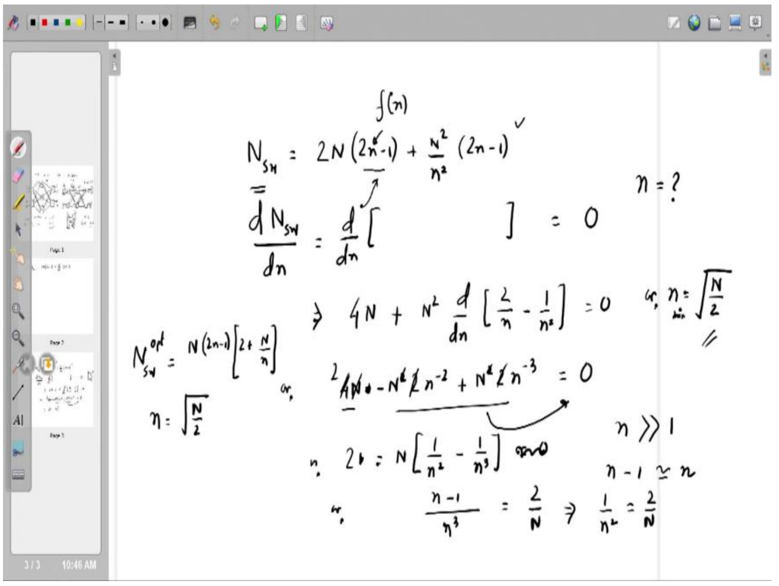


Ok, so we have been discussing Space Switch Architecture. So, we will continue on that in this lecture as well. So, what we have already discussed up strictly non-blocking and rearrangeably non-blocking switch elements, and from there, we also started for the strictly non-blocking we tried to I mean started describing how do I optimize the switch. Knowing that for a strictly non-blocking middle stage number of middle stage requirements is 2 n minus 1 ok.

(Refer to Slide Time: 00:59)



(Refer to Slide Time: 01:02)



So, let us try to see what we can do, ok. So, basically, we have already discussed this part, which is the number of switching elements that are required, and then from that number of switching elements, we would like to minimize it with respect to now only independent variable is there is a small n. Because capital N is my desired parameter, that is, that is the one switch I will have to design. So, that is fixed. I cannot change that, but small n is the particular things input that I have in my hand; I can vary that.

And I would like to actually minimize the number of switching elements that were my target strictly because I have taken the middle stage as 2 n minus 1. So, that ensures that it is a strictly non-blocking switch. So, that has been ensured already. So, now, my target is with respect to small n, which is my designed parameter.

So, I would like to get a small n which gives me a minimum number of switching elements. So, that is very easy because this n switch is a function of this small n. So, it is a function of the small n. So, all I have to do is derive, differentiate, and put it 0. So, that should be either maxima or minima; we will be able to show that; that is the minima, actually.

And at that point, we will be trying to evaluate the number n. So, all we have to do is do this derivative right. So, let us do this differentiation of these things. So, what do I get? So, of course, the second term will be gone because it is not a variable of n. So, this will be coming as 4 N plus N square that is coming this is d d n 2 by n minus 1 by n square ok. So, this should be 0, or I can write 4 N plus N square. So, now, do this derivative. So, it is actually n to the power minus 1. So, this should be basically that should be minus 2 into n to the power minus 2.

And this is again n to the power minus 2 differentiation that should be minus 2. So, basically, this should become plus ok N square 2 n minus 3; this must be 0 from here; all we have to do is we have to evaluate this n right. So, if I see it is, I can take this 4 N, I can take this whole thing on this side. So, basically, if we get 4 N, 1 N might be canceled 2 might be canceled. So, I get 2. So, basically, I get 2 n. Sorry 2 equals to this N that is common; this becomes plus. So, 1 by n square and this is minus 1 by n cube is equal to 0, sorry.

So, this is the case. So, from here, if I just put n cube, this is n minus 1, that is equal to 2 by N ok. Now what I can do this n generally is much, much bigger than 1 because I am generally trying to look for design a big switch. So, whenever I am trying to design a big switch, n will be much, much bigger than 1 most of the time ok. So, in that case, I can take this approximation that n by n minus 1 is almost equal to n. So, this approximation I can take, immediately what do I get?

So, I can write approximately as n minus 1 I can write as n. So, n by n square n cube that would be 1 by n square that is equal to 2 by capital N; from there, I get the value of n to be N by 2 square and will be able to also show that this is the point where it is actually becoming minima. So, there that is the n minima that I will be taking; of course, it might not always be an integer. So, that depends on n ok. So, it might not be an integer; it is all the integer if I have a perfect square. So, let us say n by 2 is a perfect square.

So, if n is 50, then it will probably become 25. So, small n becomes 5. So, something like that is ok. So, otherwise, we take the closest integer. So, that it still satisfies that condition ok. So, once I do this, this particular calculation.

So, from there, I can also find out what will be the element N switching element. So, in this particular expression, n, I put that minimum. So, basically, N switch optima will be in this. So, I can write it as N gets common to n minus 1 this gets common. So, I have 2 plus N by n, and then in place of n, I put this 2 by N.

So, this I put, and then I get optimal switching elements ok. So, this is something I can easily calculate, and I can get the value of it ok. So, if I just try to put that value, let us try to see what happens and what is the optimality that we have designed. So, this is where I started putting those. So, over here, we have taken the number of lines to 128. As you can see, it is very clearly seen that I had that root over N by 2 right n equals too. So, accordingly, we have taken the number 128. So, that it becomes 64 square root of 64, I get an integer value.

Again, 512. So, that is 256. I get 8 or something like that. So, we have taken that we have taken these numbers accordingly, and by taking those numbers accordingly, we have tried to see what is the optimal number of switching elements that are required ok. So, if I take 128 cross 128 switches, we could see 7680 switch ports are required. Instead of doing this three-stage switch, if we had taken a full-blown switch, then 16,000 switching ports would have been required; as you can see, more than double for only 128 cross 128 switches ok.

So, as I keep on increasing the number, you can see that the values are becoming more and more. Suppose you see an 8192 port switch. So, there here, if I do a multistage switching, I only require 4.2 million switches, whereas, if I go for this single stage, we can see already 67 million. So, it is more than 10 times increased.

So, we can see the benefit of it as we keep on increasing the switch dimension. So, this is something we could easily see. So, we could also appreciate why this Clos switch is so important as the switch dimension keeps on increasing. Even for a strictly non-blocking switch, we can reduce the switch number of switching elements quite heavily; that is what we are seeing.

If we go for rearrangeably non-blocking, then we can further reduce it. We have already seen that for a very simple structure of 4 crosses 4 switches if we give in the middle stage either 3 switches instead of that 2 switches, we can still have rearrangeably non-blocking parts, but as we are reducing the number of middle stages we could already appreciate that we might enhance the switching means a number of switching elements that are required. So, we might reduce that part. So, this is something we have seen.

Now the next part of this switching design is the blocking probability. So far, whether it is strictly non-blocking or rearrangeably non-blocking, we have been designing switches that are giving no blocking, and this blocking is with respect to not user blockage. So, users are available to input and output only the switch structure is giving me blocking.

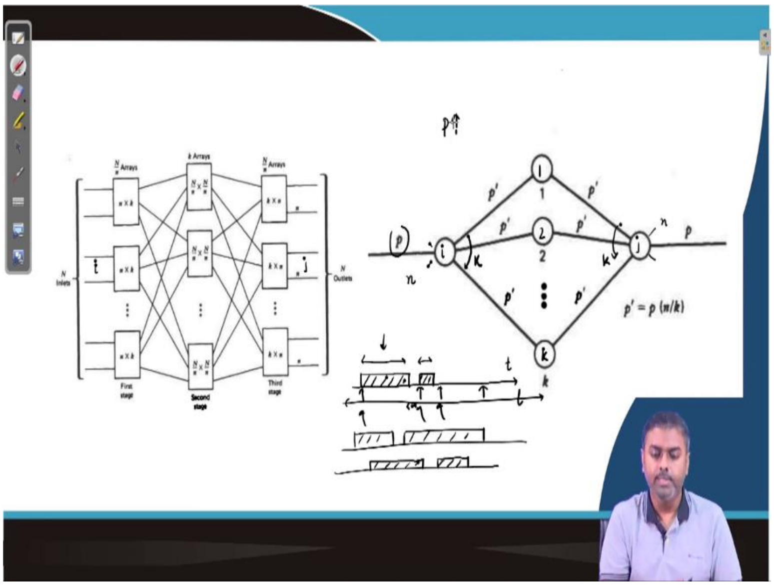
So, now, can I introduce some amount of blocking because sometimes what might happen is users might say, ok, I am making a call 100 times. I make a call if you are blocking me twice in 100 times or maybe 1 time in 100 times, so; that means, 1 percent or 2 percent such blocking I can withstand; I am not having any issue with that.

So, 100 times I will be making a call in a day, probably 1 or 2 times, and you say ok, it is blocked right now. You cannot make a call. Just come back later on. So, I will be ok with that. If the user says that, I probably need to also see if I can go and design my switches in such a manner that I ensure that this much blockage will be happening, but with that, what will be my extra facility?

Again, the same target number of the switching elements can I reduce, but now you have to come up with some engineering decision, not arbitrary switching elements; I will take out 1 or 2 middle switching elements, and whatever happens, I should not do that it should be engineering decision so, that I can actually give a guarantee on the quality that users or I am demanding right.

So, that means users might say that I can withstand up to 2 percent blocking probability, then I have to design the switch in a manner in perfect engineering design so that it fits that 2 percent blocking. So, can I do this design or redesign my switches in such a manner that I can actually give a switch design or number of middle stages same three-stage switch number of middle stages so that people can map what is the blocking probability from there, this should be our next target.

(Refer to Slide Time: 11:37)



So, let us try to see how we can do that. So, now, in this particular scenario, we are probably targeting a blocking switch, not a non-blocking switch; that means we no longer guarantee that middle stage k will be 2 n minus 1; it might be less than 2 n minus 1 this is what we are targeting. But now, what we have to do is we have to evaluate the process of blocking. Can we calculate this blocking probability? So, that is something that we will now be targeting ok.

So, let us try to see again if our target is that we have a switch this kind of a particular switch that we are showing on the left-hand side. So, we have a switch which is a three-stage switch middle stage; there is k number of stages now; k is arbitrary. It is not 2 n minus 1. It might be less than that ok. So, with that, can this switch structure can we now try to evaluate the blocking probability?

When we try to evaluate the blocking probability, we need to see this blocking probability cannot be given without knowing what is coming at the input so; that means the user traffic without knowing that we cannot evaluate the blocking probability; this is something we will be discussing heavily when we will be doing a little bit of queueing theory analysis also for the trunk switches.

But right now, let us try to understand this particular concept ok. So, this particular thing will be described with respect to a specialized graph called Lee’s graph; this was famously given by Lee who in his seminal paper, constructed a graph to represent these kinds of networks.

So, what he has done is suppose I have a connectivity requirement from the ith input stage, similarly like Clos was doing ith input stage to the jth output stage switches. So, if we just construct it. So, let us say this is the ith input, and this is the jth 1 some user from ith to jth wish to connect.

He has taken some simplified assumptions also that all users are a homogeneous case; that means all users are similar ok in terms of you will. Now ask what does that mean? Will say in terms of traffic ok or the stochastic nature of the traffic, they are similar we. Will we will come back and describe what does; that means, ok. So, these users, whenever they are trying to connect. So, this might be completely represented by Lee's graph; what is this? So, this ith stage, that is the ith one, this jth stage, that is the jth one.

Of course, this ith stage has multiple inputs. So, how many of them? n number of inputs are there, the jth stage also will have multiple outputs n number of ok. So, out of them, one of them we are picking, we are seeing it because all users are similar. So, we are trying to see from a single user's perspective one user what he will be experiencing in the network.

So, now, for him, what is happening? As you can see, he goes via this ith switching stage from there are k number of output stages each of one of them goes to this 1 2 k switching element from there is an output port k number of such thing goes to the jth one and then from there my user is connected.

So, he has simplified this and given a graphical representation of a particular user from the ith stage input stage to a particular user in the jth stage, and all users are symmetric. So, let us now try to understand this symmetric nature of the user. So, let us say I am seeing some user; ok, user l let us say ok.

So, this is the actually, this is the time over which some events are occurring to the user, and I am trying to capture that time and try to see what he is doing in user activity actually. So, what will happen? If you target that user over the day, you try to observe what he is doing.

He will have some random arrivals of calls. So, he will be trying to make some calls, and that will be random; it really is not very clear when he will be exactly making the calls this is not deterministic; it might be random at 5 O’ clock he might wish to call somebody at 7 O’ clock he might again wish to call somebody, and then after initiating the call, he will be actually occupying for some amount of duration the link that telephony link he has.

So, basically, that much duration he will be occupied in talking to the other party, whomever he is calling, then again, there will be some free period that is the period when he will not be doing anything. So, he is inactive. Again, he might initiate another call, and some amount of duration, he might occupy the channel.

So, this is how much time he will be occupying the call duration that is also random. So, there are two random events over here which is happening; one is when he will be making the call. So, this call initiation is called the arrival, this arrival is a random thing, and then how much time he will be occupying the line or the call duration that is also random. So, these two random events actually make the user’s timeline.

And as you can see, if you clearly see the user, what is happening? He has alternative active and inactive periods in the line. So, if you try to see the entire timeline of a user, it has some alternative active and inactive period. Now it is completely stochastic as this duration is random, and when he will be arriving, that is also random. So, it will be a different day if you observe it will look different. So, another day it might look like this, another day it might look like this. So, something like this activity and inactivity is a completely random thing.

But what we are trying to say is the users are homogeneous and then they have some stationarity in this process; what is the stationarity of that? The arrival process is the underlying stochastic process or the underlying distribution of their arrival that does not change with time or with different days ok. So, that will be multiple users; if you take among the users that do not change, the arrival process remains the same; what does; that mean? They are still arriving randomly, but probably the statistical distribution that describes this does not change ok.

So, this is one thing that will be happening again in the service; that means the amount of time he is the call duration that is a random different time that will be a different day that will be a different user that will be different, but probably the amount of time they take for service this whatever distribution it follows that does not change with time or with the user. So, if they remain the same then we can say it is a stationary process and it is also users are homogeneous; they have similar kinds of distribution.

If these two are happening, that means two underlying processes are stochastic in nature, then probably the activity that you will be seeing that also has some stationarity ok. Now if I try to describe something called the probability that he will be active. So, what does; that mean? You observe over a longer duration, and then you say see a person because he has active and inactive periods.

No the amount of time or the probability that he is remaining active, he is remaining busy; he is keeping the channel busy. So, that probability, if I take it, will be less than 1, of course, because if he makes it entirely busy, then it is one, anything less than that, it will be some fraction.

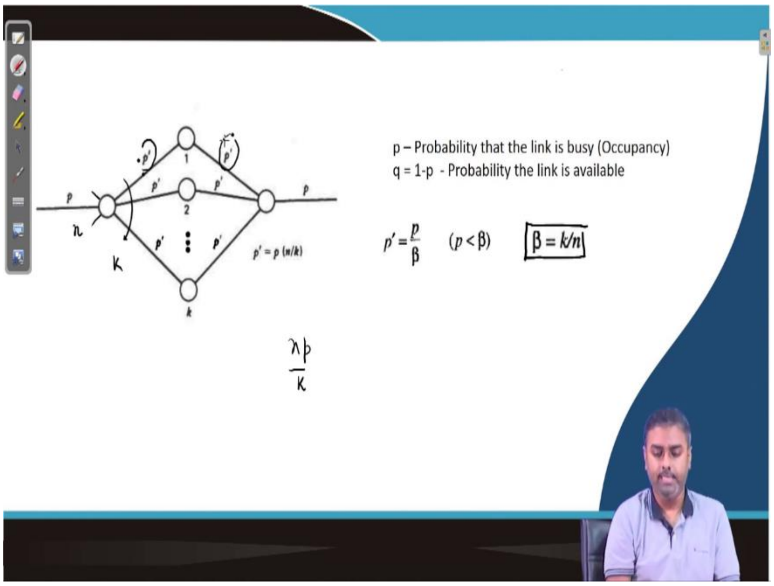
So, it will give a probability that he is busy. So, we are defining that p, or we are saying that p is that particular number of his activity; that means this is the amount of time he is keeping the channel busy. So, the more the p-value is; that means, the more the users are bringing traffic.

So, if you go to a probably home, this one user and one guy who is not very much interested in talking to others, then you will probably see his activity is much lower if you go to a particular user, if you go to business cases and you try to see a telephone in a business case, or if you try to see a guy who is interested in talking to others then you will see his activity is higher. So, this amount of activity a particular user is bringing in is very important, that is, to characterize a particular network ok.

So, this p is that particular value and what we are also assuming for a switch that all the users are homogeneous; we are trying to characterize a switch where users will be homogeneous. This might be the case might not be the case; if they are inhomogeneous, there will be little bit slight differences in switch designing, so that something else we will not be discussing right now.

We will be discussing at least the switch designing perspective; all the inputs are kind of homogeneous, so; that means they bring the same amount of p ok the probability that he will be active.

(Refer to Slide Time: 21:36)



So, once we have this now, this should be our design ok. So, what is happening? If we see this p is the probability that one of the links will be occupied ok, then what or p is the probability that the link is busy. So, that is called occupancy; what is the probability that the link will be free or the user input link will be free? That is 1 minus p.

So, this is something we already know; if this is busy, this one probability is p 1 minus p will be that the link is free. So, this is something we know. Now, let us try to characterize that if a user is bringing p amount of traffic, what will be the traffic if they are equally distributed among all the, let us say, all the output k links?

So, what will be the traffic in each of the links? So, p is the probability that it’s getting at the input. What will be the probability that each of these links is busy and we will be assuming that all links are chosen randomly with equal probability if that is the case? So, what is the maximum traffic we have already said that there is n number of such inputs.

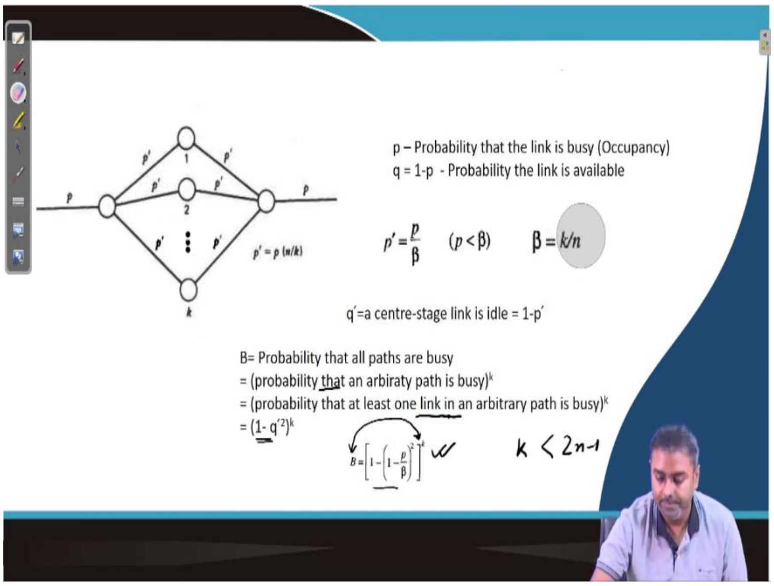
So, each one of them has p amount of traffic ok. So, n such will be bringing n such link; what is the overall traffic that is coming in? So, p is the traffic that is coming by one user, or that is being injected by one user n into p will be the traffic brought in by n number of users, right?

Because there are n number of users who are getting connected, each of them is getting distributed among these k links equally likely. So, n p by k is the amount of traffic that will be coming into each of these links. So, that is the now if we write beta to the power to be k divided by n.

So, then this becomes p by beta; as long as p is less than beta, we ensure that then p' will always be a probability because that is a fraction; now, p is a fraction and we are dividing that ok with something and p is less than beta. So, this will always be a fraction.

So, this p' is also a probability that this particular input link will be occupied because of the symmetric structure you will see and because all the users are homogeneous. So, basically, if this is having p', then this also will be having p'. So, we can already see that, ok.

(Refer to Slide Time: 24:15)



So, that is the p' we have discussed so far, ok. So, now, if p' is the occupancy probability of a particular link, then q' is the availability probability or the probability that a link will be idle. So, if p' is the probability that a link is busy, q', which is 1 minus p', is the probability that a center stage link means these links ok. So, these links. So, these links are idle ok.

Now what is the blocking probability, then? The blocking probability will be when a particular call is blocked that I generate a call over here and none of these center stages are free ok; that means all the center stages are busy. So, how the center stage will be busy? If none of the, as you can see, there are k paths 1 2 like this k paths if all k paths are busy and all these k paths are equally likely, means they are in terms of statistical nature they are equally likely.

So, if that is the case and they are independent also because I can choose any one of them independently. So, the probability that all paths are busy is the blocking probability that all k paths are busy, then only I will be blocked, that can be written as the probability that an arbitrary path is busy to the power k because they are all independent. So, their probabilities will be multiplied. So, if I can find out one path is busy and they are equally likely because I am choosing them with equal probability and they are all independent one path is busy does not actually have influence on the other paths.

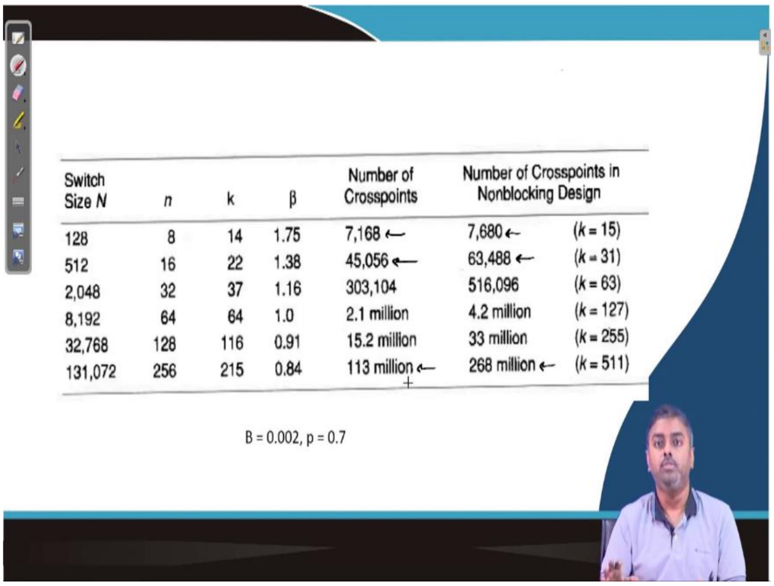
So, whether the other path will be busy or not that is all independent. So, basically, I can say that probability that for an arbitrary path, one of the paths is busy is that to the power k because all of them are equal probably now what is the probability that an arbitrary path is busy?

So, that is the probability that at least one link in that arbitrary path is busy. So, when a particular, let us say, one of the paths this path I am targeting when it will be busy, either this is busy, or this is busy, or both of them are busy, so; that means, either of these paths is busy, what is that probability?

That probability is the probability that one of these paths is busy is actually 1 minus q’ square. How we have come to this? It is actually q’ is one of the paths is available q’ square is this either this is available, and this is available both of them available; that means that entire path is available 1 minus that is actually that is not available. So, 1 minus this q’ q’ is 1 minus p’. So, 1 minus q’ square ok. So, that is that this particular path is busy whole to the power k that is the blocking probability.

So, therefore, the overall blocking probability will be this much. So, 1 minus q’; q’ is nothing, but 1 minus p’; p’ is p by B. So, after replacing all these things, we get this nice formula that B is equal to 1 minus 1 minus p by B whole square whole to the power k that is a very nice relationship between blocking probability and my design stage k. Now I can come up with arbitrary k, which is less than 2 n minus 1, and from there, I can get a blocking probability targeted blocking probability; this is something I can actually do, and if you just try to see the effect of this ok.

(Refer to Slide Time: 28:05)



Switch Size N	n	k	β	Number of Crosspoints	Number of Crosspoints in Nonblocking Design	
128	8	14	1.75	7,168	7,680	(k = 15)
512	16	22	1.38	45,056	63,488	(k = 31)
2,048	32	37	1.16	303,104	516,096	(k = 63)
8,192	64	64	1.0	2.1 million	4.2 million	(k = 127)
32,768	128	116	0.91	15.2 million	33 million	(k = 255)
131,072	256	215	0.84	113 million	268 million	(k = 511)

B = 0.002, p = 0.7

So, if you just try to see the effect of this, this is what summarizes that effect. So, what have we done? We have targeted a blocking probability of 0.002; that means 0.2 percent blocking, which is very less, and my input traffic that p which characterizes each user input traffic is a homogeneous case. So, every user will be having 0.1 traffic. What does that mean? That means 10 percent of the time, he is keeping the line busy. So, the user is not very occupied with calls.

So, 10 percent of the time, he is making calls. If this is the scenario, now we can see that for again 128 or 512 same number of switch dimensions 128 crosses 128 if I now take n equal to 8 and I take k equal to 5 deliberately. As you can see it is much lesser than 2 n minus 1; 2 n minus 1 should be 15; this is the case.

So, 2 n minus 1 is 15, actually, sorry. So, this 2 n minus 1 is actually 15 for that number of cross points that is required; we have already seen through Clos switch ok. So, with that, one optimal number of switch elements is required 7680, but if I now make my blocking probability just 0.2 percent, I can reduce the switching element quite a lot; you can see almost 3 to 4 times I am reducing it 3 times actually I am reducing it.

And as the number increases more and more reduction as you can see for 512, these are the elements that will be required if I try to build a strictly non-blocking switch, but whereas, with some amount of blocking that is just 0.2 percent blocking, I will require much less number of switching elements. If we now go for let us say if we just go for a little bit of higher this one higher amount of traffic, you will see that is the amount of traffic we are talking about 0.7 now the traffic has been 7 times increased.

So, now we will see that switching elements will be a little bit higher. So, now, according to my design, I could see that if the traffic is a little higher now probably the benefit I have got with respect to switching elements for at least k means 128 I do not get much benefit, but still I get benefit if I keep on increasing I can see the benefit. So, here you can see that it is 268 million whereas, I only need 113 million for a very low amount of blocking probability. So, now, I can induce blocking probability and design the switch according to the blocking targeted blocking ok.

So, this is so far; we have very nicely seen how to apply Lee's graph for a blocked switch design; we have also seen how to design a Clos switch which is a non-blocking switch with some sharing middle-stage sharing, and we have also seen how to make with little less number of switching elements how to make rearrangeably non-blocking switch with some amount of reconfiguration in between whenever there is a blockage.

So, with these three kinds of switching structures we have seen, we will further see if this middle stage switch can further enhance. So, from the three stages to the five stages to the seven stages, those things, how do we do that? How do we analyze those things for blocking? So, these things we will try to see, and then later, we will try to see how to design trunk switches. So, I think that it is from this particular class we will continue on this switch design further.

Thank you.