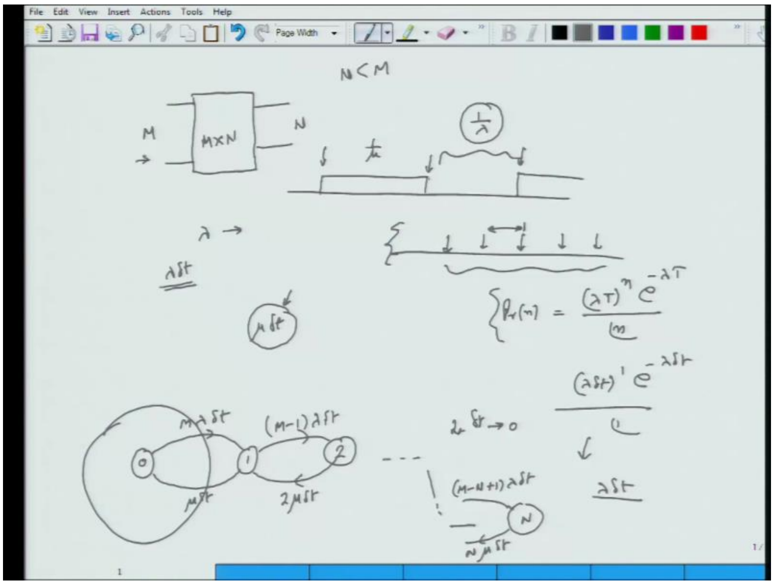
Digital Switching
Prof. Y. N. Singh
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture – 4

We will continue with the discussion where we left with the previous lecture. So, what I was doing, just a recoup of whatever has been discussed in the earlier lecture. I was actually discussing M by N composite switch.

(Refer Slide Time: 00:30)



So, there will be M inputs in this case, and there will be N outputs. So, these M inputs can come at any point of time and N outputs have to be connected to the outgoing sides; all subscribers are connected on this side. Typically, this particular calculation will actually, be required when you are actually, going to compute either for blocking probability estimates. For that of course, what I will do is I will try to use this to actually, give an understanding of what is the call congestion and what is the time congestion. I have to formally, define these terms and also convey the meaning of these two, because this is how the switch blocking probabilities are usually, estimated; versus basically, what is the amount of call congestion and what is the amount of time congestion in a switch, is an important parameter.

For this, the way I actually had told earlier that for each line, a call rate can arrive at lambda calls per unit time. This technically, means between when one call on a time scale, when a call gets finished up and when the next call arrives, the gap between these two is exponentially, distributed with mean value of one by lambda. As I have actually, mentioned in one of my lectures that exponential distributions, as well as a Poisson statistics, are kind of two faces of the same coin. So, if look at the number of events and this events are point events which are happening; so number of events which are going to happen, that probability, will be given by in a time t. So, N event is going to happen, that probability is given by this.

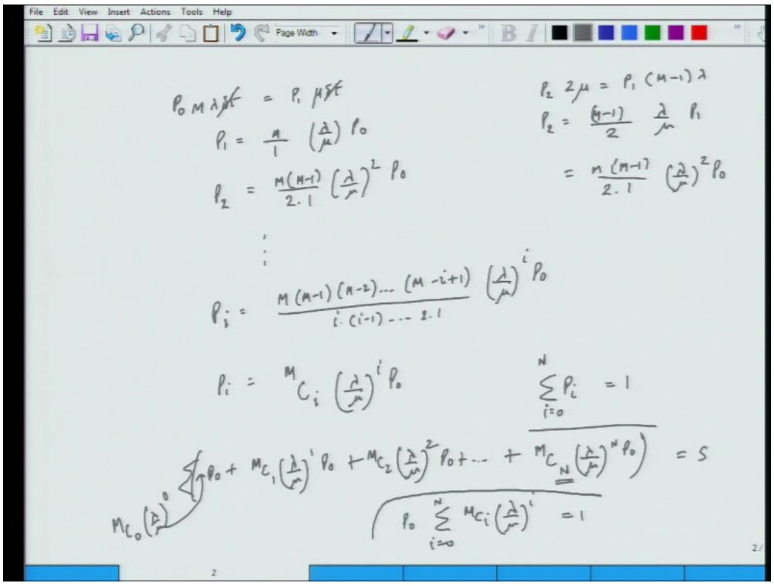
Of course, the time gap between them is going to be exponentially, distributed with mean value of one by lambda; that is what actually, it means. Whether you talk in terms of poisson distribution, or you talk in terms of exponential distribution; these are technology same things, but when we talk about the voice call, the things are slightly different. Because when a call arrives, the person actually, talks; it consumes. So, channel is blocked; no new call can arrive at this point of time. Only, when the call gets finished, then the next call only, can arrive. This inter-variable time basically, when the call gets finished, the next call arrives; that is being characterized by one by lambda, actually, for the circuit switching systems. Because this is not a point process; a call arrival is not a point process. It arrives, but then it holds on for some time, before the call closes; it does not finish it off; it cannot be served immediately, actually.

It takes some time. So, usually for this, we will have exponential distribution of one by mu and for this, one by lambda. For a small infinitesimally small time, delta t, the probability that one call will arrive, is given by lambda delta t; you can put here lambda into delta t, E raise power N; E raise power N minus lambda delta t by one factorial; You put N limit. When delta t goes to 0, this will be actually, converging on to this entity. So, this actually means if, and we have to define the state of this particular switch. So, a state of a switch is; how many calls are currently through. If you are in a state 0, you can actually, move to a state 1. This will happen, because call can arrive on any one of these lines, and that will happen with a probability of; because each line is independent. So, that will be happening with the probability of M lambda delta t, and similarly, when a call is already in process, what is the probability that call will be completed actually, in time delta t; that will be given by whatever, be the number of calls, multiplied by mu into delta t.

Because mu into delta t is that a call will get completed in a infinitely small time, delta t. If there are M calls, so, the probability with that one call will get completed; will be M mu delta t kind of thing. So, in this case, there is only one call, so, you will have mu delta t. Then, of course, this state can be two. So, in this case, it will be M minus 1, because this is already one call through. Now, second call is arriving. So, blank input ports for M minus one; it will be M minus 1 lambda delta t, and this will be two mu delta t, and so on. Then, of course, in the end, the maximum number of calls I am actually, assuming a case where N is less than m. So, the maximum calls, which can be done is n. When you are in this state, you will actually, have number blank input ports or input port lines will be M minus N.

Even, if a call arrives, it will remain in the same state; call cannot be accepted, because it cannot be connected to any outgoing port. So, just before this, it will be M minus N plus 1; there has to be at least, one output port, free for coming to this particular state. This state will be called N state, will be lambda delta t, and when you go out, it will be N mu delta t. Under (( )) state operation, the probability of being in a state, is going to be constant; is going to be same for the time to come on an average basis, actually which actually, means under (( )) state operation, if I take any close surface like this, the rate at which the transition, which will be happening for going to outside the surface and coming into the surface has to be equal; that is the balance equation.

(Refer Slide Time: 06:23)

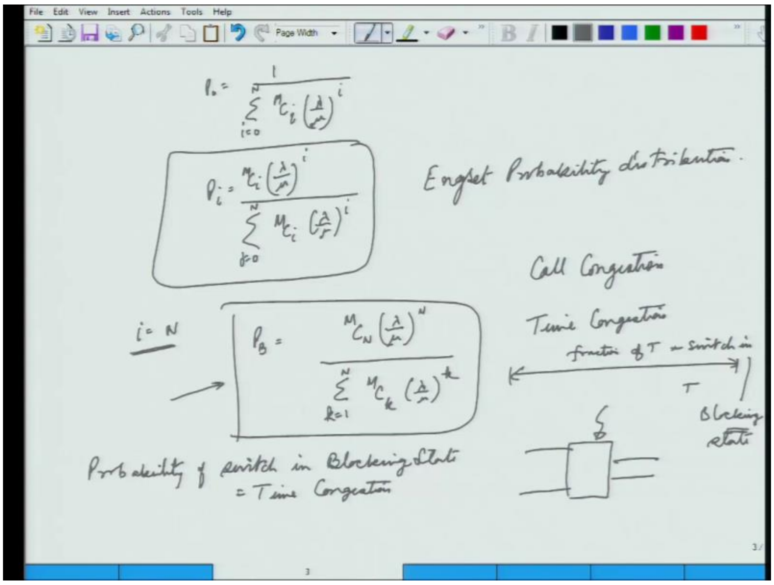


Once we create a balance equation, I can solve for this. In this case for example, it will be M lambda delta t; probability, that you have to be in a state 0. So, that is the rate at which, the transition out of the surface will be happening. This has to be equal to P 1. There is only one mu delta t. So, that is the rate, and delta t, of course, cancels. In fact, the balance equations will not have delta t, which actually implies I can now actually, need not consider this delta t is here, because you are talking about rate equations; which implies that P 1; I can now write here, P 1 will be nothing, but M lambda by mu by one. What about P 2? For P 2, if write the equation, P 2, 2 mu t; in fact, now I am not writing delta t, because those will anyway, will cancel out, and if I am there in P 1, this will be M minus 1 into lambda.

So, this will become P 2 is equal to M minus 1 by 2 lambda by mu P 1, and I can put a value of P 1 from here; this will be M into M minus 1 by 2 into 1 lambda by mu square p 0. So, that is what will be the term, and so on. So, P of I will be, in this case, will be M into M minus 1, M minus 2, and so on. M minus I plus 1, I of I minus 1 to 1, and of course, clearly this is nothing, but M combinatorial I lambda by mu raise power I into p 0. So, that is how the state probabilities for this particular mark of chain actually, can be computed. Then, we also know the rule that some of all state probabilities, because all of them are mutually, exclusive and system has to be in one of those states; this is a complete definition. So, summation of all state probabilities P of I, for I is equal to 0 to N, has to be equal to 1. Remember we have done it for M m 1 infinity q also in one of the previous lectures.

I can actually, use this axiom, and then, of course, do the summation. This will give me summation of P 0 plus M c 1 lambda by mu raise power 1, P 0 M c 2 lambda by mu raise power 2, P 0 and so on; the last will be M c N of course, lambda by mu raise power N P 0. So, of course, I can actually, put here M c 0, which is 1 lambda by mu raise power 0. So, this gives me a complete arithmetic progression, sorry, this is geometric progression, because it is binomial series actually. Once I do the summation of this, but the problem is I cannot solve it, because I need to get the terms from 0 to till m. I am actually, having only terms still n. So, I have to just keep it as it is; that is the sum, which is going to be there, and if I do this, I can write this thing as M c I lambda by mu raise power I; I goes from 0 to N, where N is less than M. P 0 is equal to 1 and from here, I can get an estimate of what is P 0.

(Refer Slide Time: 10:42)



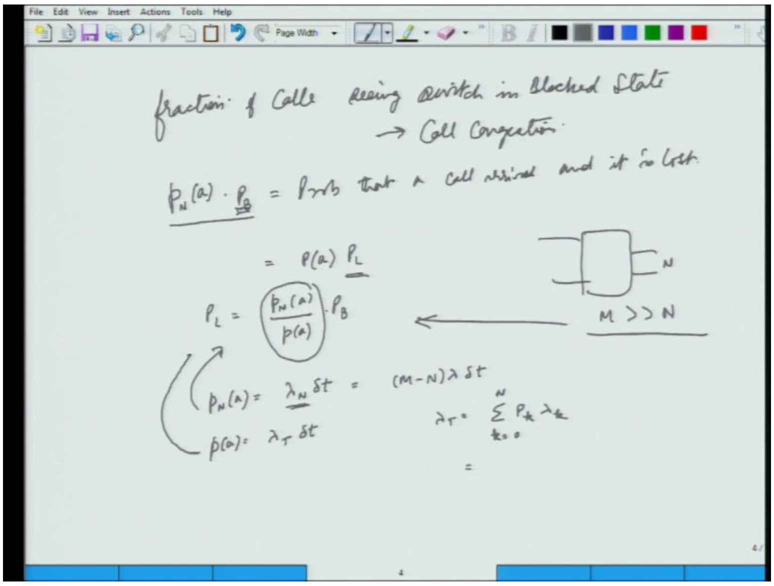
So, P 0 will turn out to be 1 over summation of I 0 goes to from 1 to N; M c I lambda by mu raise power I, and then, of course, I can actually, further solve it. I will get P I is lambda by mu raise power I M c I, whatever, was the P 0 value, which is; now, this is the probability distribution and we call it Engset probability distribution for a composite switch. So, this gives the blocking, basically, the probability that a switch will be in a state I. Now, if I put instead of I, I put N which is when, the switch will be in the blocking state; once you are in a state N, any new call which arrives; the call cannot be completed, because all the outgoing ports will be busy. So, probability of blocking can be identified as P b is equal to M c N lambda by mu raise power N; k is equal to 1 to N, M k, M c k is lambda by mu k; that is the blocking probability which we have.

Now, when I am talking about a switch, it is very important; there are two things. Even, if the switch is in this state for which, I have defined the probability, and no call arrives, for example, if you are operating a switch; switch thus goes into this state, but when it is in this state, no new call comes, actually. In fact, none of your calls will actually, see blocking. So, there is a difference. Switch, just remains in blocking state, does not mean that a call will get blocked by the caller only, get blocked, if a call arrives. So, we have to now, define something called call congestion and time congestion. Now, I will explain this thing, like this; that there was a switch, and all calls the switch is in the blocking state, because you know, if some new call arrives, it will not get through.

So, the probability; your switch is in the blocking state is what is known as time congestion, actually. So, that is time congestion. Clearly, what you are seeing here is nothing, but the time congestion; it is not the call congestion, actually. If you want to actually, make a measurement of a switch, you just keep on observing the switch. This probability will give you nothing; if you observe the switch for sometime t, it is the fraction of time, fraction of t for which, switch was in blocking state. It does not talk about, whether calls were blocked or not blocked. It is talking about only switch, being in a certain state. Since, it is in terms of fraction of time, you call it time congestion; it is a time for which, it was in blocking state.

I cannot change the definition and probably, some of you, may not be interested in this parameter. People can ask what I would like to know is if I make one million calls; how many calls switch was not able to complete; switch was not able to connect them, actually. So, the fraction of calls, which saw the switch, was in blocked state; now, that actually, depends on the call rate, when you are making the call. That fraction of calls, which will see the switch in blocked state, is what is known as call congestion.

(Refer Slide Time: 15:17)



So, this is fraction of calls, seeing switch in blocked state, is what is known as call congestion. So, we would like to, now actually find out a relation between these two; how that actually, will be done. So, what we can say is let the switch was in blocked state; that probability is P b, and when you were in this blocked state, this basically is call congestion. What is the probability condition on this that you are in state N, a new call arrives; that probability? This will give what we call in absolute sense, a call loss rate; number of calls, which will be lost on an average basis. If one makes calls, how many of them will be lost, is not conditioned, actually. It is not conditional probability, but it is in an absolute sense, probability that a call arrives, and it will be lost. It is not that what is the probability that a call will be lost, conditioned on, that a call arrives. Remember, there are two different things. If a call arrives, now; call has arrived is already sure. Now, what is the condition probability that it will be lost? So, call arrival is already there; if call is not lost, this probability of call loss is going to be 0; probability, if the call is going to be last surely, it is 1.

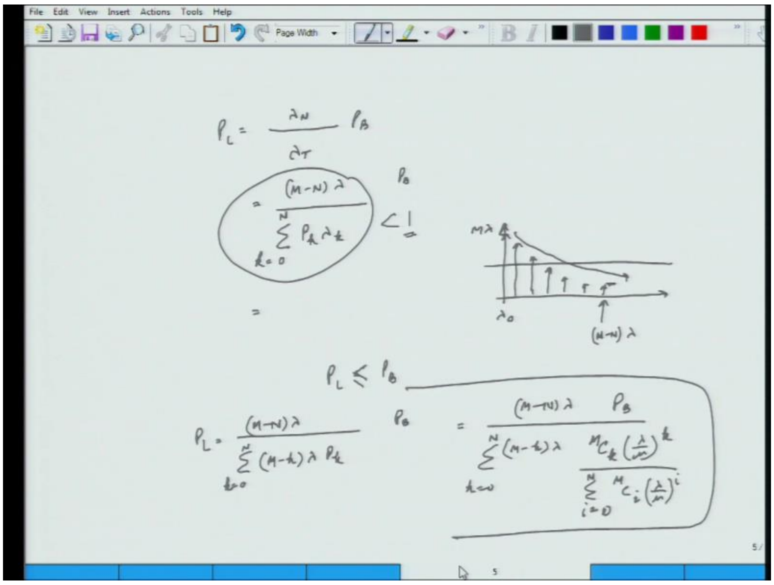
But, when I am talking about this kind of thing; probability that a call arrives and call is lost; that will never be equal to 1; that will be less than 1, actually, usually. So, this should be that probability call arrives, and it is lost; I should write this; that this is a probability that a call arrives. So, call arrival probability is also included, and it is lost; both things. This should be equal to that; what is the probability that a call will arrive, P of a, and then, it will see that switch was in blocked state. So, we call it P of l; that is basically, is probability that a call arrives, is a conditional probability that when call arrives, the call will be lost. This is what we call congestion. This is time congestion.

We need to find out the relation between these two. So, from here, we clearly; because we have estimated this P b; we have to estimate P l. I will do it for composite switch. So, P l will be given by P b. Now, if the call arrival probability, which I have written in the bottom; is the call arrival probability when switch is in the state n. When call arrival rate is independent of the switch state; switch state does not govern it; this value and this value, both will cancel. These both will cancel, and call lost probability will be equal to, or call congestion will be equal to time congestion. Both of them will be same. This usually, will happen in very large identical switch, and you are operating below a small level, actually. Very large number of things are there. It is like, if you take M and N, then M is very larger than N; then, this will be very closely, call congestion and time congestion will be very closely, actually, become equal to each other. There will be minor difference, because in absolute sense, in real sense, these two probabilities cannot be same, but you can approximate them.

We now, need to make an statement for this. We need to find out, how you will find out P and a. First, we need to find out this. Now, this is call arrival probability when switch is in blocked state. So, that will happen. What is the arrival rate, total arrival rate, because of the free incoming ports, when the switch was in state N over time delta t; this will be nothing, but you have to find out this arrival rate is M minus N, because that is the free ports into lambda into delta t, and I am talking about arrival chance within infinitesimally, small time delta t.

So, that is why I am not talking the full Poisson expression, because remaining thing actually, goes to 0. Now P of arrival call arrival; because now, I have do what we call statistical averaging over all possible states, for all arrival probabilities. So, I can write this thing as lambda t into delta t where lambda will be nothing, but it will be an average value; probability, you are in state k is this, and the arrival probability in that state is this 1. So, I will do the averaging, sorry, this will be N, because there are only N states. So, that will your lambda t from where I can get P a.

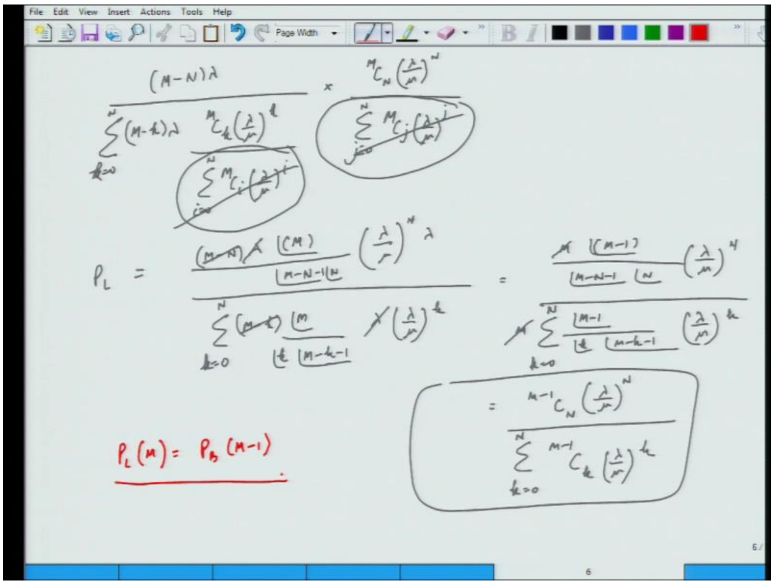
(Refer Slide Time: 20:54)



So, this also, can be done. Now, if you put it, of course, I need to put these two in this particular equation, and I will get P of l lambda N, lambda t and P of b; that is the term which, we will get from here. So, P n a will be nothing, but lambda N delta t; P a will be lambda t delta t; delta t will cancel. Again, we are talking about rates and lambda N; I can write as M minus N into lambda into P b. This I can write as summation; let me use the right same subscript; P k lambda k, P v and this. Solving this further, now, you can actually, clearly, observe one important thing; I am doing the averaging of arrival rate. So, lambda, when your 0 state, will be very high; because it will be M lambda actually, in that case. As I keep on reducing, keep on actually increasing my state value, it will be monotonically, decreasing function. In the end, you will have M minus N into lambda. So, average value, which you will be computing, will always be higher than this particular value, which actually, means this whole fraction will always be less than 1; which implies that your call congestion P l, will always be greater than or equal to P b; the time congestion. So, this will be usually, holding true; unless, your arrival process itself, is a state dependent. Then only, it will be equal; otherwise, it will be always less than condition.

Now, let us solve this for this particular thing. So, P of l will be given by M minus N lambda P b so, I can write now p k is M minus k just a minute M c k, sorry, lambda t is M minus k lambda, and I have to put a value of P k. Simplifying it now, P k will be given by Engset set probability distribution, N c k. Just, you look what we had derived; this is how you will get your M c k. So, I have to write exactly, the same expression. So, I am using subscript here, is 0; this is a different summation; that is what it should be. So, it is exactly from here, I have just taken it and you have got this expression, and of course, ultimately, P b here. Now, solving it further, and probability of blocking also, we have the relation already identified by this particular thing, and I have to use this expression. So, let me use this also.

(Refer Slide Time: 24:34)



I am just rewriting everything here, on the fresh sheet. Now, t b is going to be M c N lambda by mu raise power N, divided by summation, variable; running variable here. Now, this particular entity and this particular entity are the same, and they are independent of k. So, I can simply cancel them. This will give me nothing, but P l or the call congestion. So, you will end up with M minus N lambda. This, I can write as M, M minus N factorial, N factorial lambda by mu N into lambda M minus k, M factorial, k factorial, M minus k factorial; there is a lambda; lambda by mu raise to power k. So, this lambda, of course, can cancel with this one, without any problem, and I will end up in getting; this M minus N will cancel with this M minus 1, and I will get minus 1 here. This will be renewed. M minus k, this will become minus 1. So, I can take this M out, and it will become M minus 1 factorial, M minus N minus 1 factorial, N lambda by mu raise power N.

Similarly, M will be taken out here, and then, of course, M will cancel. I will get M minus 1 c N, summation of over, sorry; this summation is over, N actually. Now, interestingly, this whole expression, this particular expression is very similar to the expression given here; this particular one, except this M is now being replaced by M minus 1, which implies that P of l for a value M is nothing but equal to P of b for M minus 1. So, that is the relationship, which will come between call congestion probability and time congestion. So, once we understand this, we need to now, see we actually have to use it for finding out what we call in a three stage interconnection network, and how to estimate the blocking probability. So, this same concept will actually be used there. In the next lecture, we will be actually, doing a call congestion estimate for a three stage class network.