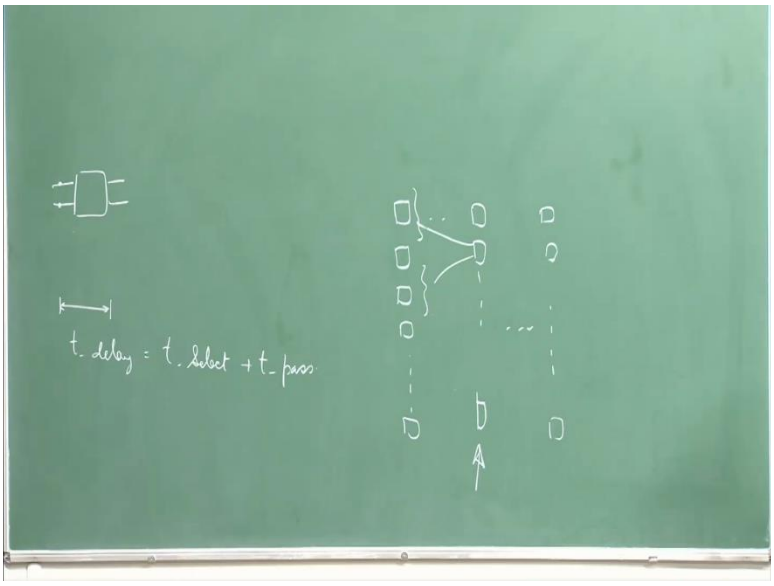
Digital Switching
Prof. Y. N. Singh
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture – 26

So, we will start from where we left in the previous lecture. So, I will now give how you actually iterate [FL] these are particular specific scenario of how you solve a Markov chain; technically, it is actually solution of a Markov chain by doing iterative computation actually. Because sometime when these number of states are extremely large, it is very difficult to get a solution. So, usually the way we would always teach the solution is we always say, okay, let us build up a Markov chain, where we have some state probabilities.

And based on that then you start building up what you called balance equations and then using a fundamentalism; that all state probabilities sum has to be equal to 1. You actually get all simultaneous equations, solve them, get state probability; that is what usually we do. Here actually I am not interested in that; I am more interested in what is going to be my throughput performance, what is going to the outflow but that actually depends on the states.

(Refer Slide Time: 01:36)



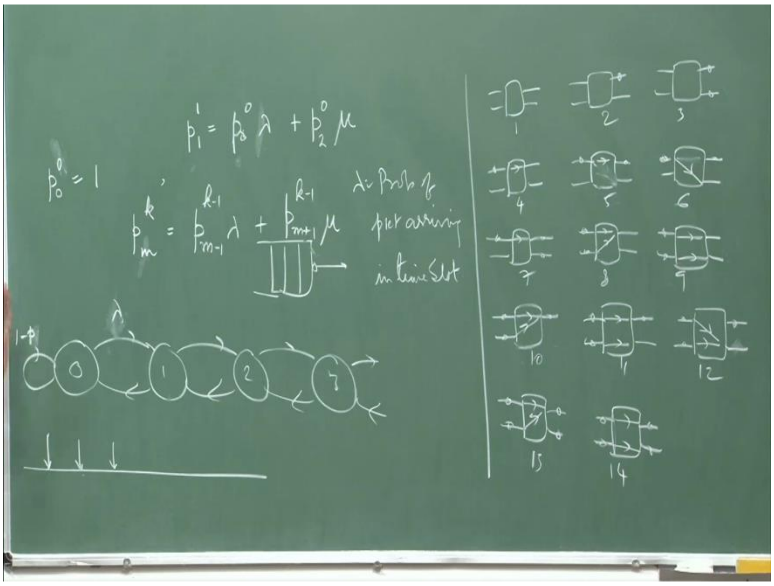
So, last time I had drawn fourteen states of a 2 by 2 switch, okay, and we will take two modes once one case will be when. In fact, for every switch when the packets are there at the input, there are two ways I can actually understand; two ways the analysis can be done. One is the total period which is required in this case is what we call, just a minute I just t delay actually. So, t delay is one step delay from here to here. This delay usually will consist of t select plus t pass.

T select is the time required to identify to which outgoing port these packets have to go and then when actually the transfer happens that requires this much time. So, you will take actually first the case when t pass will be t select will be 0, and then t pass will be 0; there are two cases actually. So, now one important thing I have given you fourteen states. I will assume that because now switch will actually consist of buffer delta consists of many switches in state, and there many states actually and so on. So, all states probabilities within a stage are going to be same for all switches; all state probabilities for all the switches in one single stage is going to be same; that is the assumption.

So, I need not bother about all these switches, because, otherwise, you see fourteen states I have given, and then you have how many switches? So, fourteen probabilities raise to power number of switches; those many number of states will be there. And even if it is a very small number, say, you are going to have 8 by 8 switch, for example, 8 by 8 switch requires twelve such switches. So, 14 raise to power 12 is a huge number actually, and mentoring those kinds of things in a Markov chain or any finite state machine is going to be complicated; you cannot handle that actually. So, analysis cannot be done in that way.

So, first simplification which was done is this. This is usually true because number of ports which are coming in, these are independent sets, and that is true for all switches, because what is true for first switch resulted for the second, so on, iteratively. So, this is all switches at technically see the same thing. So, their state probability has to be actually same; now this is going to be true. So, I am going to take this case. This is the one important thing; you have to keep your state diagram handy, all states actually.

(Refer Slide Time: 04:49)



So, I will just still put them here. I will not erase them during the lecture, because I have to refer again and again back to these states. So, I think I can redraw them. So, this is one, okay. Now before I go onward with the iterative method of computing the state probabilities, method is going to be pretty simple. For example, if you want to find for a queue, for a single queue I think I can just explain how it will be done for that. For that actually usually, I can find out every simple solution because my balance equations can be formed. So, with this if I actually being given you zero transition probabilities, one transition probabilities and so on; for example, this is an infinite queue.

So, usually actually you can always run an algorithm on this. So, you can start with a queue when it is empty, and now we talk about state probabilities at various time instance; that is very important. While, when we did a queuing analysis of a simple queue, I was not using time instance for state probabilities. So, t is equal to 0. So, I will always define p 0 as the state at time t. So, this is k. So, k will be 0 in the beginning. So, your initial boundary condition can be that there is no queue; that is the probability, and then you will have transition rates. So, you have to define transition probabilities.

So, lambda is a rate for going into one direction and mu is not going into that; mu is coming back. So, you will define this as lambda, and you are not going to go there is 1 minus lambda actually. Important thing is transition is happening at every time instant; it is a differential. So, I am not making a very different kind of queue. Here on a time scale at, t is equal to zero packet will come, t is equal to 1 something will happen, t is equal to 2 this will keep on happening; it is not as such a Markovian thing.

So, lambda I can replace this thing by actually p the probability p it is a discreet a time system now. So, with probability p the event packet comes, 1 minus p packet does not come; with packet again the same thing is going to happen at every time instant. So, from here actually you can find out what is going to be your p 1. So, from here you can actually find out whatever was the value p 1 at instant one. In fact, you can build up an equation based on this.

So, for one I will show, then I can write it for general. I will use a very similar procedure there. This one will be that you have to be in this state in the previous time instant, then there is a transition probability which was I call it something else I think; let us call it lambda. Lambda is not arrived at; it is a probability, you have to understand it. So, lambda is probability, but this is a computational procedure; you cannot have a closed form solution here. So, with this you will be able to come here.

Second thing which you can get is p of two; from here you also can come, and this probability you have defined as mu. Mu is the probability with a packet will go out of the packet actually out within that slot. It still requires same slot; it is technically is not arrived at, it is a probability; it is a conditional probability technically. So, if you are in this state, chances that you go here is this, but it also turns out to be same as rate. So, this method is not used while doing for a simple mm 1 infinity queue. So, I have modified the queue somehow.

So, you get one equation through iteration; remember my time instance is different here. And to begin with my initial condition is that this is one, rest everything has to be 0 because of this; rest everything has to be 0, okay, and q is also in state 0. So, I will be able to get p 11 from here. I can in fact solve for all these in general. So, I can get p of, say, m in state k as p of m minus 1 lambda k minus 1 plus p of m minus 1 m plus 1 actually k minus 1 into mu.

So, now actually you have to just run a computer program. It will start with this initial condition, and as your time moves, you are computing every time p 1 k, p 2 k, p 3 k, p 4 k; you are just iterating. And p 0 k we will also get adjusted correspondingly; you have to understand p 0 k will also be changing will be a function of p 0 k minus 1 1 minus lambda. So, what I am doing is we are adjusting controls; we are reducing it by some amount and increasing all other state probabilities.

And after sometime in the computational procedure itself, it is always bound to converge. You will find there is a steady state condition which will operate, and steady state condition is when the flow rate almost becomes constant out of a system. So, when for all of them, you will find out that it happens that p of any value any m and, say, k; this almost dot lambda, lambda is actually constant here departure rate. So, the flow rates here statistically have to become constant; they will not be varying or when the flow rate becomes gets balanced that also you can figure out. So, in this case condition is pretty simple here.

Student: Sir, here when we are drawing probability of zero, are we not transferring the probability from state one to zero.

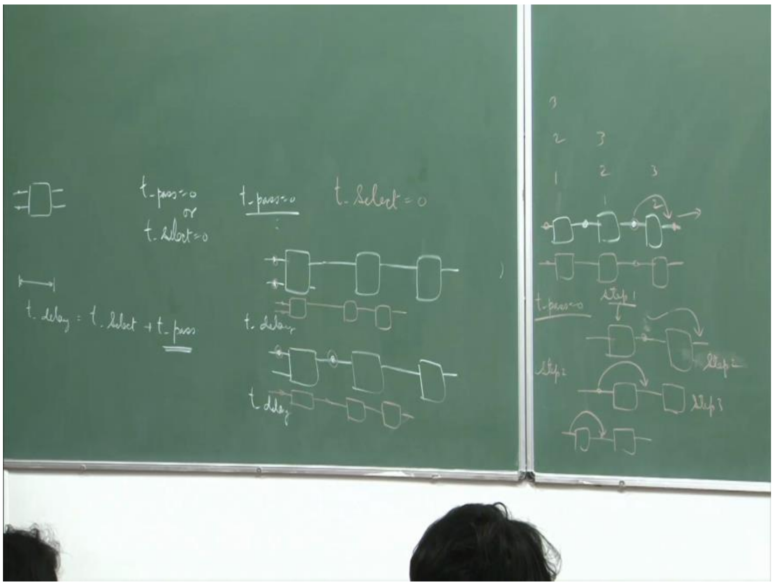
Yeah, that also need to be there, you are right; that also has to be there, you are right, you are right, I agree, k minus 1; yeah, that also you have to make that. So, when this condition or the balance conditions actually you see, start seeing your computation, things have converged; take those state probabilities, and you have got your results. So, with various values of lambda and mu actually, you can find out that thing. It will turn out to be almost very same what we did for the simple queue. Now this same procedure we are going to invoke here, because here we cannot get the closed form solution. This Markov chain was very simple; it could have been solved, but that one we cannot do.

But just to give an example I have just simplified this thing into this particular. This is a computational procedure, but this comes in the category of still analysis; it is a not simulation, okay, because you are not simulating anything. We are just computing, and it is an iterative computation till the computation stabilizes or converges. So, we are technically solving balance equations, and balance equations are when the flow rates from in to out and out to in, both are same.

So, we are just finding out when the flow rate converges, and everything is going to be keep on it will actually balance. Remember, it is like a first order equation; it is not a second order. So, there is no question of oscillations here, and if you carefully observe, it is a first rates order actual equation. It is always trying to move to a stable condition. So, convergence is guaranteed actually; this is fine.

Now this one was a simple queue. Now I am going to come to complex one. So, state transition probabilities are complicated. I think one important thing is you have to understand how state transition probabilities are estimated, and how you actually understand the whole extraction; that is the important thing. So, far the things were I think little simpler.

(Refer Slide Time: 16:11)



Now coming to this; first case switch we will take is when t pass is 0. So, there are two cases, when I can take t pass is equal to 0 or I can take t select is equal to 0; this is or condition. Now what is the meaning of this actually, and what is the implication of this? So, I think once you understand this, then it will be much more clear. In the paper it has been given; later I am doing it early actually. I somehow feel this should be done. So, I have to take the cascades of the switch; let us take this case.

So, all switches are empty. There is no nothing in the buffer, and they are having exactly one buffer at every input port. It does not matter; you can actually then do the simulation even for two buffers or infinite buffers, whatever way you feel. So, if you have two packets coming in, now you have to look at the time instance. So, this is at one particular time instant, it has come. Then the t delay will elapse after this, and then the next state will come. So, how the state transition will happen? And then in this case, I am taking as t pass is equal to 0.

So, this t delay will be nothing but equal to t select. This is the time required for identifying to which outgoing port the packet has to go, but packets move almost instantaneously from input to output, from any buffer to any buffer. So, once the decision will take this much time, you know there is a packet. At the end of this t delay, what will happen? One packet will be almost instantaneously just before my time period finishes just before that 0 minus t minus actually. So, if it is zero instant, then just 1 minus.

At that instant, the packet would have moved here; your situation will be something like this. If both where I am assuming containing for the same port, you will come in this situation. Another t delay will happen; you will get a situation like this now. So, in this t delay, they will figure out it has to go to which port; this has to go to this port. Earlier also we tried, but there was a contention. So, it was buffered. This will also be identified and just before the next time slots just minus just slightly earlier, the packets will be moved instantaneously.

So, you will actually have this packet coming onto this position and this packet coming onto this position. So, you will have this situation. So, that is when your t pass is 0; I am drawing with a different color when t select is 0, then what will happen? So, it actually means when this particular period from here to here that slot which is starting within 0 time, I am able to figure out which packet has to be going to the outgoing port. Then it takes one full slot for the packet to be transferred to the outgoing buffer, okay.

So, after this situation, this is what is going to be there, perfect. So, this is what will be the situation, another time slot delay. So, in the beginning of this particular slot, they will know this packet has to go here, and this packet has to go here. But the problem is this packet cannot move, because this buffer is not empty, because in the beginning of the slot itself, we have identified where the packet has to go. But when you start transferring this packet will take full one slot for transmission to the outgoing buffer, and unless this buffer is empty, this cannot move. So, this will still remain struck there.

So, you will end up with a situation here where this packet will be here, and this packet will be here. So, that is the difference between the two. So, unless when t select is equal to 0. it is simpler actually. So, I will do a complicated version first when t pass is 0.

Student: Sir, final state will be until there will be no packet in the middle stage.

The one with the red once you look at or maybe I can draw it separately if you wish. So, colors you have to match. So, red one corresponds to red one, white one corresponds to white, okay. So, now for when t pass is 0, you have to understand now if there are packets which are queue dub like this in a string, all of the packet will move in the same slot to the outgoing port. Decision will take the same time, one slot period, then there is an instantaneous movement. So, all of them will move in a chain. Now solving this is slightly tricky. So, what happens is each time slot or each event is being now identified into three phases.

So, first phase we call when the packet actually goes out when this is the outgoing buffer. When the packet from your outgoing buffer or the input of the next stage; input buffer of the next stage is pushed out. That is the phase one, when the phase one is happening. So, this is step one actually; we do not call it phase one, this is a step one here. Step two will be when the packet from here to here will come; see I have broken things into step, so that when this goes out, I can immediately move the packet in the same slot here, which is not possible when t select is 0.

For t pass for simulation or iteration, I need to do this. So, that is the step two and step three is that packet comes to your input buffer. So, you have to understand when step one is running here, which step is running here? Step three; When step two is running here, no, this is not step three; this is step two actually. Packet is going to from input to output of a buffer; this is the step two. This same thing is happening here for this. So, while at this same time, the step three is happening for this person.

Student: Sir that is for t select 0.

These are for t pass; t select is equal to 0 you do not require this definition. This breakup is not required per slot. Bala, if you carefully think I think this is difficult to explain, but this is the way it happens; this is the only way it can happen actually.

Student 1: Sir, once again sir.

Student 2: Sir, more complicated is t select is equal to 0.

More complicate is t pass is equal to 0.

Student: So, t pass is equal to zero, everything is moving in a chain?

But then I have to split it when I am doing iteration. See it is a problem of how we implement the computation; their computation is simple every one step. One time slot one step, but here every time slot broken into three steps, and this actually does allow all the chain to move in one go. So, when a time slot starts you will say step one here, okay, then step two, then step three. So, everywhere you will do the step one. This step one will happen first, then step two, then step three for whole switch.

So, usually what will happen is this is the packet at the outgoing buffer. Step one; this packet will go out instantaneously here. When this step one happens here, then you will apply step one here, and at this point step two is going to happen here; it is a chaining which is happening. So, when one is happening here, two is going to happen here, and then one is going to happen here, then two will happen here, and three will happen here. And then of course, two will happen here, three will happen here, last three will happen here and one full chain of event.

So, one complete slot will be simulated to one two three; all three has happened. For this there is actually nothing to do with one or two; one as happened in the previous time slot. So, currents and time slot I can implement this way; otherwise, it is not feasible. So, now coming to the formal definition which are required here.

Student: Sir, t pass equal to zero we require minimum three steps to three.

T pass is equal to zero, three steps per slot.

Student: And three stages also, sir.

Any number of stages there can be just three.

Student: Minimum three minimum three stages

Not necessary, even with two also you can. It is like a state machine; one full slot gets done when all three steps are executed.

Student: Sir, what happens in each second step?

Second step packet is coming from your input to your output; first step packet from your output has to go.

Student: Output has to go outside?

From your output, the packet is moving further, going out, it passed from your input to output, then packet has to come, then from the packet has to come to your input. These three things will give a slot for a switch.

Student: Sir, which is my inputs at this case out of these one, any particular switch in particular?

Which one?

Student: In step two and three, then what is it?

This is it. I am talking about this switch. When for this switch, this step one is happening, this packet is moving out; for this switch, step two is happening at that time.

Student: Sir, which is out sir or in step two and three?

Which one? This step two is for this.

Student: Yet a packet is going.

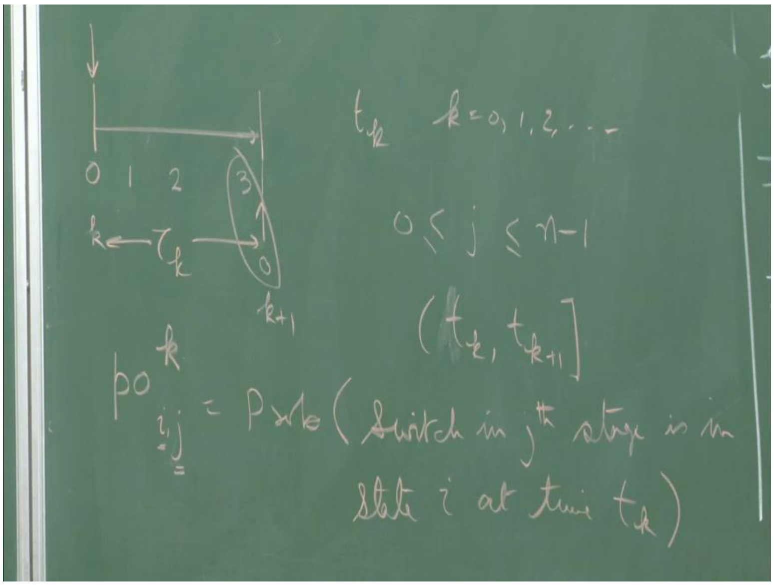
Packet is going from input to output, but then for this switch, the packet is coming to the input. So, there is a step three for this and step two for this. Last one packet is coming to my input. So, the previous switch you are having a.

Student: Two.

Step two happening. So, it is always one two three, and it actually propagates backward; that is what is happening.

So, when full one two three set propagates from output to input, one slot has finished. Now that is the way we assume. T delay is one delay; t delay is delay here, remember, and every t after t delay, packet moves further. So, one t delay requires three instances, three steps, and selection time is full t delay in this case, but packet passing is instantaneous packet transfer. So, we define now what we call about probabilities before we come to the states. So, notation is very simple.

(Refer Slide Time: 28:46)



So, what happens at time when slot starts? You are at state you are at step zero, okay, then step one will happen, then step two will happen, step three will happen step; three will be nothing but step zero for k plus first slot. So, step three for k slot in step zero for k plus one are same. So, now we will actually define at time instant t k now k can go from 0, 1, 2, and so on, because it is an iterative calculation. These are stages in the switch. So, I am not worried about how many input ports are there 2 raise to power n by 2 raise to power n, but number of stages matter.

Because I have made an assumption which is the correct assumption for almost all delta networks that in a stage all switches are equivalent; their state probabilities are going to be same. So, far they are in the same stage. So, under that assumption, and we will define now a probability we called it p 0; this is after step zero actually in time instant in time slot k k starts from. So, that is a slot, and I think do you understand what is the meaning of this closed and opened intervals? So, in case t k is included, t k plus 1 is not included; that is what it means. It is open on this side and closed on this side.

So, this particular probability is probability that your switch in j th stage, this is this is in state; this is not stage, this is state i this one. So, notation you have to remember. So, do not change it; it has to be consistent actually at time t k. So, if this is the t k instant, when the time starts this that value. Now this whole interval is known as tau k; tau k is this interval. This side is not included, this is included.

Student: Sir, in tau k I am assuming that stage one two three will be?

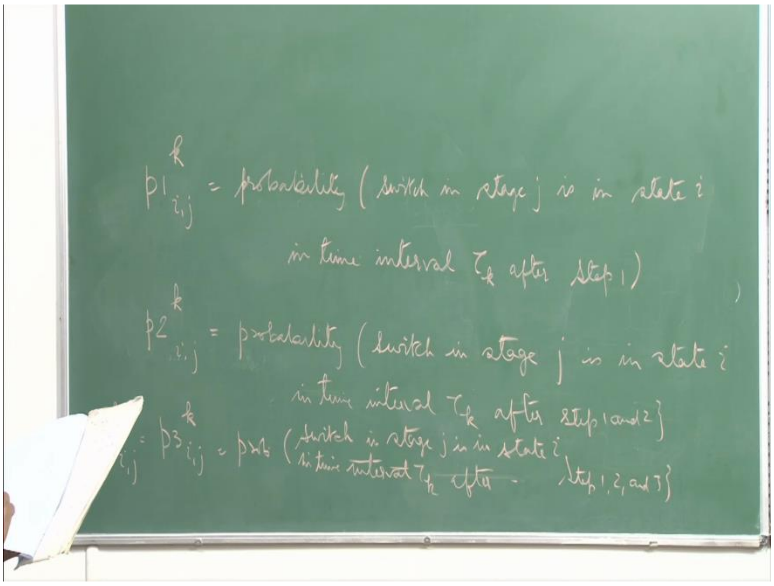
Their states are there, steps are there; step one two three are happening. I am not going to write down those probabilities p 0, then p 1, p 2 and p 3. P 3 will be nothing but become p 0 for the next time step, next t delay.

Student: Time t k, the switch is in state i.

Switch in j th stage at state I, there are many stages.

So, for multiple stages we have to solve. I am just giving the formalism; actual calculation has to be done through a computer.

(Refer Slide Time: 32:21)



P1, so this number identifies the step after which you are looking at the probability stage time instant time slot, okay. This is pretty common in most of the switching analysis; you will use lot of subscripts superscripts and all kind of notations. So, this is the probability that your switch in stage j is in state i in time interval. I am not writing time instant now; time interval tau k after step one.

Student: Step zero?

There is no step zero; third step is step zero for the next time.

Student: Sir, but that notation should be q 2 then.

Which one?

Student: In this p zero.

This is after zero th step, just before the first step.

Student: Just before the first step?

Professor: When first step is taken, second has happened, third has happened. Third is nothing but zero th of the next one next time slot. So, I will change there; k will be changed there.

Student: But it counts zero only we start counting zero one two three.

I am counting one two three.

Student: Zero is same as three?

Zero is same as three but time instant changes. K will become k plus 1; see while at steps I would not count zero one two three. It is the objects which are counted or ports which are counted zero one two three.

Now I am being consistent with the paper, because ultimately I know this is difficult to communicate in the class; you will anyway have to depend on the paper. And if my notations are not same, there is going to be a problem in that case in understanding. So, I have to be exactly same what is given in the paper. But certain nuances which have not been explained; for example, t pass and t select difference has not been explicitly told. The procedure of computation is not explicitly stated, because the author assumes that is what happens in most of the paper; we do not give details sometimes.

We assume these are kind of a common knowledge and person can figure out on the other side, but I know the class cannot figure out as of now. Unless you are a PhD student with a good amount of experience, it is going to be a difficult to figure it out, okay. So, I could do that; that does not mean that everybody else can do it. So, I have to be explicit, okay, and at least you have an option to fallback; you can always ask me if there is problem. I never had that option; with this paper is pretty old I do not know where are these guys.

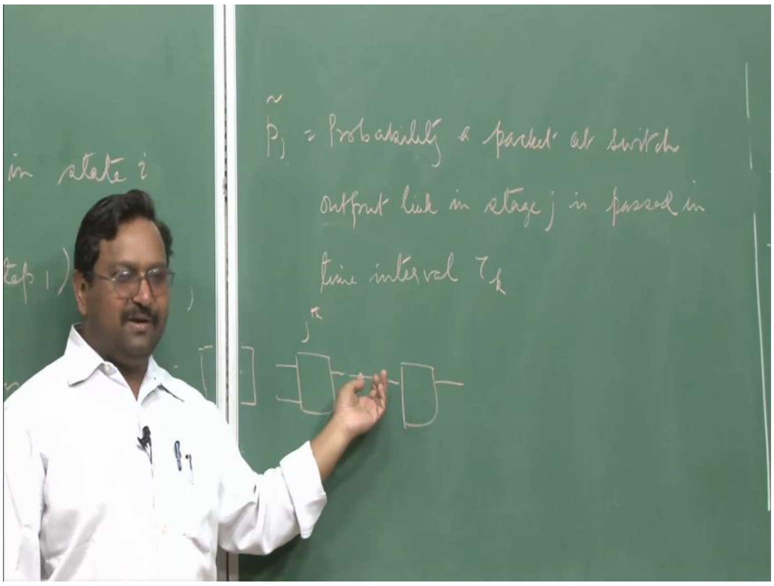
But I think this was the first one which k buffer buffer delta the analysis part and it matches with the simulation; that was a good thing. And this has taught me a different method of solving Markov chains iterative computation which is again usually not taught at most of the places. So, switch is in switch in stage j is in state i in time interval tau k after step one and two both has happened. Similarly, you will have p 3 k, and this has to be equal to p 3 i j k plus 1, okay. Rest everything, follow exactly except in the last you will write step one two and three, rest everything is same.

Student: Next time interval will be tau k plus 1, sir?

Yeah, next will be tau k plus 1.

So, this, let me write it; I think this is better, because switch in stage j is in state i in time interval after step one two and three, and we will define one more probability actually. Now come back to trickiest part. Now what I am defining is actually will be the probabilities which are used for computing what we call transition probabilities at every state. And these will also keep on changing with time as time evolves, and they will stabilize after some time in the Markov chain.

(Refer Slide Time: 38:31)



So, first probability is p j tilde. So, I am going to use three of them. I think one has to be careful, I will be using this, I will be using this; these three have all different meanings. So, first I will define the top most one. This is the probability that a packet at switch output link this stage j; this j actually identifies this is passed in time interval tau k. So, this actually is indeed a function of k. I am not explicitly mentioning it, but all these p j tilde, p j bar and p j will be function of k. I will not be writing them; paper also has not written it. So, I am keeping it as it is. Ideally, I should have actually put a k subscript k superscript there for doing this.

So, this actually means this will be in terms of now the states; whatever those fourteen states are there, and I have already defined how the probability will be defined after every step. And we know what these probabilities actually mean; what happens in step one, what happens in step two, and what happens in step three, okay. So, from here if they are two stages, if this is a j th stage, what is the probability that a packet will go out? I am trying to estimate that, okay.

Now usually what will be the function? This will depend on at the start of the time slot what was the state of this switch, okay? And then the step one will be happening for this and step one for this and step two will be happening for this, and there is a switch previous to this; that also you have to take into account, because their index j actually will change when I am going to estimate for p j. I will be using in this case j plus 1 actually, and that decision will happen is what happens after step one here, because step two outcome will be decide by what has happened in the step one here, okay.

So, that is why the expression actually will contain all j plus first terms, but what is going to happen after step one? So, let us see what is going to happen.

Student: At expression tau k is passed in interval t q.

Time interval tau k.

Student: I mean it is passed in the second line

Second line, the probability that a packet at switch output link in stage j, packet is here is passed in time interval tau k passed to the output port. It will never pass anywhere else; it can only move forward. So, it is being moved in time interval tau k; that is what it means.

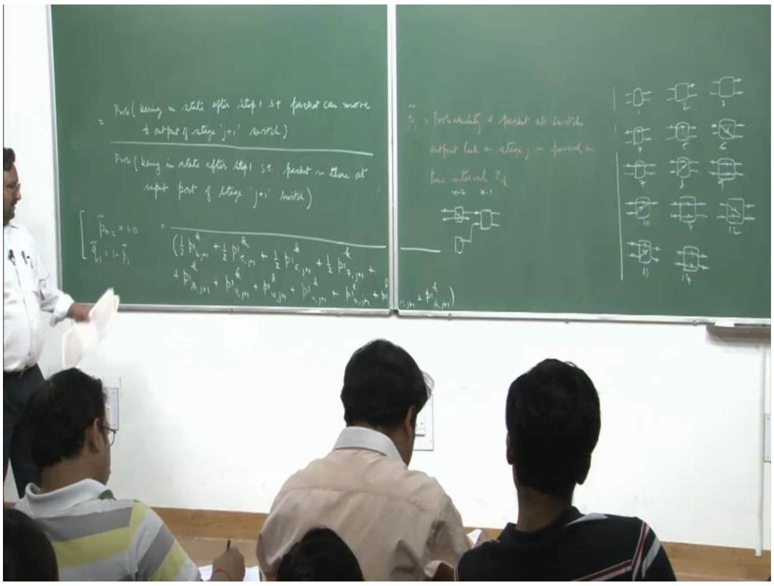
Student: Transition comes step one to step two, sir?

Step one two three we have to identify.

Currently, I am only looking at if there is a packet sitting in here, what are the chances it will move? It is the conditional probability, remember; it is the conditional probability that packet is sitting here. What are the chances that it will move here? You have to identify basically these states for this. What is the earlier state, and after step one what is happening here? That will decide the probability of transition for this.

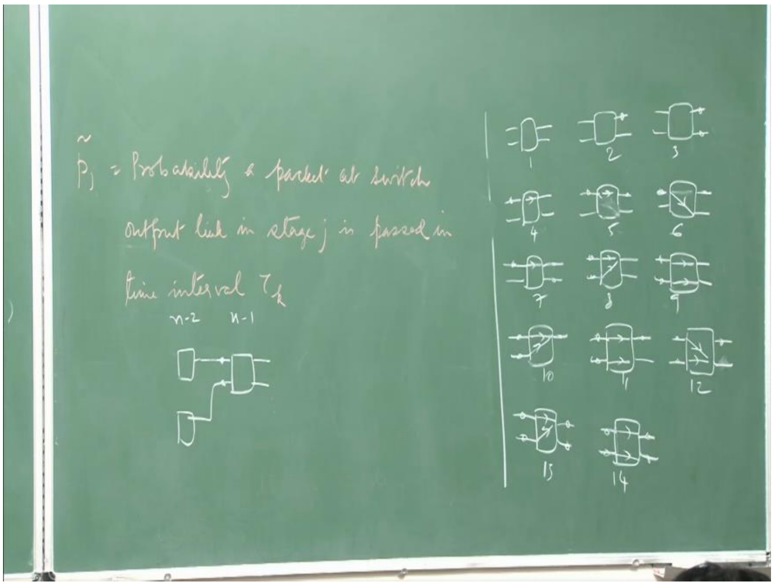
See whatever happens here will have a impact here. If packet moves here, then only state will change here; if packet does not move, state does not change here. So, these are transition; this actually will be used for computing transition probability here. This is the outgoing packet probability, but that will be function of what happens in step one here after step one. If the packet goes out or does not go out, packet does not move out from here, then nothing will happen; this packet cannot move. So, essentially what I have to do is I have to now list this thing can be written as.

(Refer Slide Time: 43:27)



So, this will be nothing but probability of this such that actually. I have to essentially identify states when this is possible, and remember this is a conditional probability. So, I have to identify all these states when the packet is there at the input of j plus first stage, okay. This will become more clear when I will give the example now build up all that. Now I will give you the example how this actually works out. And of course, for this the boundary condition before that is this at every stage, you will identify q of j by this.

(Refer Slide Time: 45:52)



These two are why these two? In the last stage if the switch is there, there is one buffer here; packets are instantaneously removed. So, there is no packet queue dub here; there is no buffer required at the output. If these two packets are there in the last stage and they both have to go to any port; even if they have to go to the same port. There is only time required is t delay which is for selection. Selection is done when both of them will be instantaneously put here, and they will be instantaneously taken out. So, probability that this particular packet will be which is there in this is n minus first stage remember; this is n minus 2.

So, probability that the packet at the output of n minus 2 will be passed out; it has to be always equal to 1 if the packet exists condition on that. So, that is the initial condition. So, there is one of boundary condition of the solution; you have to start with certain boundary condition always, okay. Initial condition will be when all switches will be in this state, the state one, no packet in the system when you start your iteration.

Student: Sir, boundary condition p tilde n minus 2 is equal to 1.0; they say there is no output contention at the end point.

Not possible because packet is taken out instantaneously. T select is equal to zero when t select is equal to something is equal to zero, then this will be problem will be there; you cannot take. Then it has to be p n minus 1 tilde is equal to 1; it is not p n minus tilde.

Student: If this t pass is equal to 0, t select is what we are modifying; t select is equal to t tilde.

So, in full slot, both them will be selecting the one output.

Student: What if there is output?

It does not matter, because time t is zero for reading out the packet.

So, both of them can be read out instantaneously, and they can be read out from the output port is zero interesting case.

Student: Since I have only considered with the packets at the input; that is why I am taking only step one.

Packets at the input, I am bothered about j plus first stage, what is happening here.

See what, I am in some state. I have some packet queue dub, what happens to my outgoing packet; that will decide what will be my next state, whether packet can move or not move. I am looking into transition probabilities now. Whatever is my current state here, this can only get modified if my packet moves, and that depends on what is the state probability after step one here. So, that is why this has been done this way.

Student: If the contention is in that case, there would not be any delay, sir? Both the packet will travel simultaneously in case there is a contention.

Only at the last port, only at the last stage, not here. See what happens, for example, if you put packets here; if these two packets does not go out or see both of these wants to come here. So, you have consumed the full pair in finding out they have to come here, but this buffer is not free. Buffer is not free, they will remain there. Remember the patterned model of the switch; there is backward flow of signal which is there. Packet can only move if the buffer is empty at the input of the next switch.

Student: Here we do not consider packet dropping?

There is no packets are dropped; they are all buffered. It is a full loading condition; remember the initial assumption is there. There is a maximum loading condition. I am only estimating what is the maximum achievable throughput, but I am actually using a trick. I can actually modify the input probabilities, but input probability initial conditions or boundary condition I am taking such that it is a maximum loading condition, okay. I will write actually more; this is not only the expression. I have to write p j bar also.

So, now this will be I have to first of all write down all the states when the packets are there at the input. See then only they can go out; if the packet is not there at this port, they cannot go out. I am worried about here. So, I have to look into all the states in j plus first stage with the packet is there at the input. So, first one; can I put one? I cannot; there is no packet at the input, okay. So, one two three cannot be used, four can be used; there is only one packet.

So, if the next stage switch is in four, there is a chance I might be connected to the one where there is no input buffer is not occupied or I might be connected to the top one where the buffer is occupied. Both of these can be connected to my switch in j th stage with equal probability. So, I will use half of this p 1 4 j plus 1; I have to write all that, time instant k. Half is because I can be equally likely connected to anyone of those if the switch in the next stage in this state.

So, my probability is half; 50 percent of time I will be connected to the one where input buffer is occupied, 50 percent of time buffer is not occupied. Even if I flip these input ports, this I can take on top, this on bottom is still the same state, remember, plus what is the next one? You have five also similar situation, but remember in five, the next stage in that state the packet will not move. Packet cannot go, because the next buffer is not empty after step one; this is after step one actually. Step one has already taken place in j plus first the outgoing packet has not moved out. So, my packet also cannot move in.

So, four will be sitting in the numerator, but five will not be sitting in the numerator; it is only in the denominator for this conditional probabilities. So, I will write p 1 5. So, I am intentionally writing it first thing here, then I will only take some of them on the above part; all of them will not go p 1. Same is with the sixth; there is only one input port. I can get connected to anyone of these, but six packet will go out. So, it will go in the numerator also.

Similarly, next one will be half p 1 7 j plus 1 k. I think you have to take a large copy if you take a a4 sheet print of the paper that tilde and bar are not clearly written on the paper actually. So, if there is a confusion, then you ask me, because there tilde will be actually visible as a bar in the paper.

Student: Sir in six why tilde half factor will come because now the two states are not?

[FL] but my stage j th stage is going to connect to j plus 1 either here or here.

Student: Half factor?

With f50 probability I can be connected here; fifty probability I can be connected here. So, when I am connected here, input buffer is not occupied.

Student: Sir, eight it should not come step eight

Step eight it will be half, no step eight it will not be half.

Student: Will not be half.

Will not be half, it will be full; it will be full.

See this is the probability that packet is here at the input port. So, here only 50 percent time packet will be there at my outgoing port or the input port of the next stage. There it will be sure that packet will be there in the port. So, p 7 and then of course, yeah, then after that I will not be putting half here as we have currently identified p eight. Similarly, you will have p 1 9, p 1 10 will be there? Yes, it will be there, I think now almost all of them will be there 11 p 1 12 p 1 13 p 1 14. So, okay, I think we are now going to close.

So, only I will put the numerator part, remaining we will do it in the next lecture. So, numerator part will be? You can quickly put. In case of four, yes, packet is going to go out. So, I am going to write it positively here when packet can move half of p 1 6 j plus 1, packet can move in case of 6 also, it is free; five packet cannot move actually go out. So, that should not count to my probability by definition. So, half of p 18 p 19 I think both the packets can move out. So, it is 100 percent probability half of p1 11 half of p 1 of 12.

So, this gives you the probability of p j tilde to you in terms of the state probabilities after step one in the next stage. So, I will use this p j tilde to essentially compute the. So, we will start from one edge and then compute all the way in the forward and backward direction, two ways we will do the computation actually. So, I think here I will close today and then we will move forward; we will find out what is p j bar and so on in the next lecture.