Communication Networks

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

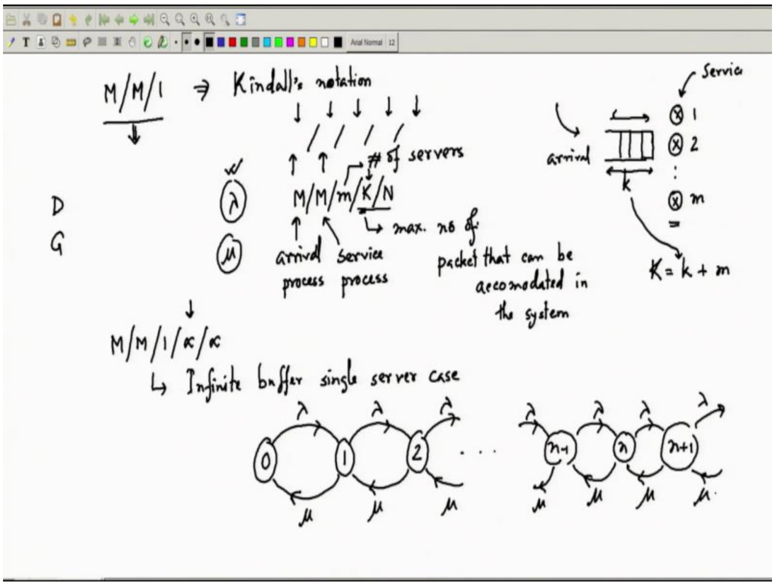
Module - 06
Erlang Formulas
Lecture - 26
M / M / 1 Queue

So, far we have discussed the discrete-time Markov chain and then the corresponding continuous-time Markov chain. So, we have seen how the guiding formulas, the characteristic of the Markov chain, the corresponding rate matrix; all those things we have already discussed.

Now, what we will do is we will see the application of this continuous-time Markov chain for our case, that is the trans switch analysis that is where we started this detour of queuing analysis. So, now we are in a position to launch that aspect. So, there are two important formulas are very important which are called Erlang formulas. Erlang B and C formula actually.

So, we will see how to derive that, but before that in this lecture particularly we will initiate the whole process and we will discuss some very famous queueing models or very elementary queueing models we should say that is called the M M 1 queue. So, let us try to understand.

(Refer Slide Time: 01:20)



First of all, this notation when we say this M M 1 queue; what do we mean by this notation? So, let us try to understand this. This is actually according to Kendall’s notation ok. So, what is Kendall’s notation let us try to understand. It is actually with a slash there are five spaces. This is 1, 2, 3, 4, 5; actually ok.

So, there are 5 spaces; over here, you can see 3 I have populated. We will come to that why 3 has been populated ok. So, the first space actually. So, this is actually a typical queueing system analysis. So, for the queueing, we know that the arrival process is very important and the service process is very important these are the two things that have to be specified.

So, the first basically, first location specifies what are the characteristics of an arrival process, for which we are doing an analysis. So, this is the arrival process if we say M over there. So, if we write M in the first place; that means, the arrival process is Markovian with of course, there will be some arrival rate that will be specified if it is Markovian; that means, it has to be Poisson.

So, it must have an associated rate which is lambda ok. And, we all already know that it is a homogeneous system and it is a stationary system, for that only we are trying to analyze ok. So, this is the first thing that has to be specified. So, if I say M, it is Markovian. If I say D that means, it is deterministic.

So that means, it is a regular interval with some arrival interval ok, some specified arrival interval. If I say G; that means, it's generalized. So, it can take any distribution ok, so, accordingly. So, we will right now concentrate on Markovian because we know that we want to do a trans-switch analysis where the call arrivals generally follow the Poisson process with some arrival rate lambda.

So, it is always Markovian; so, we can write it Markovian. The second place talks about the service process ok. So, again we have a Markovian service process, and again we can write M. There also we can have other options like deterministic generalized any distribution you can take Weibull, Gaussian whatever comes to your mind.

So, for all kinds of things we can take any distribution, it can be even deterministic also, but for our case already call durations are again Poisson distributed or sorry exponentially distributed with rate mu; so, we will take that ok.

The third place so, basically is the arrival process, this is the service process. What is the third one? So, the third one gives a number ok. I can put some small m ok this gives a number, and what does this characterize now? We have characterized the arrival and service process now we have to characterize how many servers are there. So, this specifies the number of servers.

The fourth place tells me. So, basically, now what has happened? For a particular queue I have a queue I have a queue followed by a server or a server system multiple servers also can be there. I have arrived with some distribution I also have for each packet there is a service. So, the service has some distribution ok.

I have now defined how many servers are there. So, this is m 1, 2 up to m, m servers are there ok, but I still have not decided in the queue, how many places are there how many customers, or how many packets I can keep them. So, what is my buffered size? Ok. So, that is what is defined over here, but remember when I define this, this actually tells me that how many customers.

So, the maximum number of sorry maximum number of packets that can be accommodated in the system is very important. So, it talks about overall in the system how many packets or how many customers I can accommodate maximum. So, if the queue has a length of k then how many total customers I can accommodate? So, k can wait over here and m can be solved. So, this should be m plus k.

So, if I specify a number capital K that capital K must be small k which is this; the number of buffer places plus m number of servers. So, this specifies in the system at max how many customers I can hold. The last one gives me another interesting thing that probably we would not be mostly taking into account.

This says how many customers from outside are making arrivals this is something we have never talked about. This is a finite customer system, if we specify that number if we do not specify it is infinite. So, generally, our assumption will be infinite customers. So, outside the queue who are making arrival, how many potential customers are there? Who will be making the arrival? So, suppose I talk about a particular railway counter ok railway ticket booking counter ok.

So, in that railway ticket booking counter, how many customers potentially; that means, the overall popularity population of the area who will be making potential arrival to this railway booking center ok. If that I can approximate or assume to be infinite then that is infinite. The general assumption will be infinite, but I can also have a finite population over there ok?

So, I can say that some n number of customers are there outside the system who can potentially make an arrival to this particular system this particular queueing system ok. So, if they are not specified otherwise, then the value can be taken to be infinity especially these last two. So, if I do not specify over here M M 1 queue.

So; that means, the arrival process is Markovian, and the service process is Markovian only one server is there. So, it is a single server queue and then the last two things are not specified; that means, I can take potentially there to be infinity. So; that means, there is no restriction in the buffer I can have an infinite buffer. So, it is an infinite buffer single server case.

So, whenever I say M M 1 queue, it is actually an infinite buffer single server case ok. So, that is what we are describing over here. So, this is the system we want to analyze; that means, there is only a single server and there are infinite buffer customers arriving with Poisson distribution with an arrival rate lambda and being served with independent exponential distribution with rate mu. This is what is happening in the system I want to analyze this system.

I want to see what will be the corresponding average number of customers and the average delay that it gets all those things distribution of all those key parameters. So, how do I see all those things, how do we do analysis applying the Markov chain that is our target ok? So, let us now try to see because everything is Markovian.

So, I what can be the state description now. The state description as we have already so far discussed can be because all are Markovian. So, I can just describe the states by number of customers in the system. So, that is countable. So, the state is defined by only one value 0, 1, 2. Now, you can see how far this state can go because the queue length is not finite.

So, basically state also potentially can grow up to infinity, because as many customers if too many customers are coming the server is not capable of serving all of them. So, it will it will just grow and go towards infinity. So, therefore, potential state can go up to infinity. If my server was finite then this potential state would go up to sorry, not server my amount of customers who can be kept in the system that was finite then the state will go up to that finite number only.

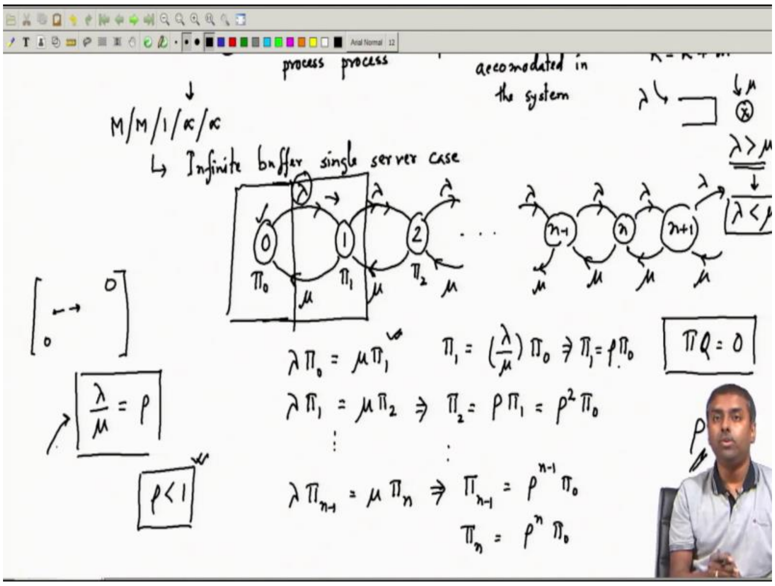
So, accordingly, your Markov chain will be either infinite or finite. So, it depends on what description we are giving. So, if there is a finite number in the fourth column or fourth place then always it will be a finite Markov chain. If I do not have any specification; that means, it must be infinitely like for the M M 1 queue.

So, it is infinity. So, therefore, it goes to infinity and it is infinite I mean infinite Markov ok? So, now, graphically I can represent. So, you remember we had this rate and due to the nonsimultaneity principle, it can only go in the next state and it can come back to the next previous state only. So, there will be arrows over here that is the transition, rate transition matrix.

So, if you have some n sorry n minus 1, n, n plus 1. So, from every state, it can grow to the next state, and from every state, it can come back to the immediate previous state; that is the overall Markov chain. Now, in the rate matrix, what should be the associated rate? This will be we have already seen this with the arrival rate.

So, if that is lambda. So, always it will be lambda and this is with service state. So, always this is mu ok. Now, can I draw means I have drawn the Markov chain? Can I now put the set of equations ok?

(Refer Slide Time: 12:48)



process process

accomodated in the system

M/M/1/∞/∞

↳ Infinite buffer single server case

$\lambda > \mu$

$\boxed{\lambda < \mu}$

$\boxed{\frac{\lambda}{\mu} = \rho}$

$\boxed{\rho < 1}$

$\boxed{\pi Q = 0}$

$$\lambda \pi_0 = \mu \pi_1 \qquad \pi_1 = \left(\frac{\lambda}{\mu}\right)\pi_0 \Rightarrow \pi_1 = \rho \pi_0$$

$$\lambda \pi_1 = \mu \pi_2 \Rightarrow \pi_2 = \rho \pi_1 = \rho^2 \pi_0$$

$$\vdots \qquad \vdots$$

$$\lambda \pi_{n-1} = \mu \pi_n \Rightarrow \pi_{n-1} = \rho^{n-1} \pi_0$$

$$\pi_n = \rho^n \pi_0$$

I can do this in one way which is actually by writing pi Q equals 0, and then the Q matrix will be with this lambda mu you will be able to populate the way we have discussed it will be an infinite dimensional matrix.

Forget about the diagonal term you put the off-diagonal term 1 on this side it will be all lambda 1 on this side it will be all mu and then this will be minus lambda plus mu we have already seen that. So, the rest of the things will be all 0 ok, and then put all the set of linear equations this is one way of doing it.

Another way, we have discussed the total outgoing rate because it is stationary so; that means, the total outgoing rate of probability should be balanced. So, incoming should be balanced with outgoing. So, if this has a probability of pi 0, this has a probability of pi 1, this has a probability of pi 2; then lambda pi 0 must be equal to mu into pi 1, because that is the rate; rate into associated probability that is the rate of probability that will be coming in and rate into associated probability.

So, the rate is lambda and the associated probability is pi 0, which should be going out this should be balanced similarly if I construct a system over here with a combined system state of 0 and 1, what is the outgoing rate? So, that should be this one; and what is the incoming rate? That should be this one.

So, again I can put lambda into pi 1 must be equal to mu into pi 2, and so on. So, lambda into pi let us say n minus 1 must be equal to mu into pi n. Like this, you would be able to put certain means series of equations. Now, from here I can see the relationship pi 1 must be equal to.

So, lambda by mu pi 0, let us call this call lambda by mu you will see this is a very important thing that is rho ok. So, let us talk about that that part. Now, over here we need to also characterize something that is called the stability condition. We have already told stationary so, but there is a condition that it will remain stationary.

Very generally think about it, a particular queue is there ok and a single server is there. The server is serving at a rate mu arrival is happening at a rate lambda if somehow this arrival rate lambda it is greater than mu, then what will happen? So; that means, the rate at which it is arriving he is not able to handle that because he is serving at a lower rate.

So, the queue will keep on building up, and what will happen? It will remain intransient it will never reach a steady state it will go up to infinity because the queue will always build up because it is not able to handle the incoming rate. So, therefore, the queue will always build up. So, the stability condition is the reverse of that it must be less than ok even equal we will later on see that equal also will not hold there also queue will not become stable ok.

So, for stability condition what we need is this lambda must be less than mu ok. This is one condition that has to be true; that is something has to be true if I have to do a steady state analysis. If this equation is true then this condition must be satisfied that our arrival rate must be less than the service rate; otherwise, there is the queue will not give me any satisfactory answer. It will always grow up to infinity the delays will go to infinity eventually and it will never reach a steady state.

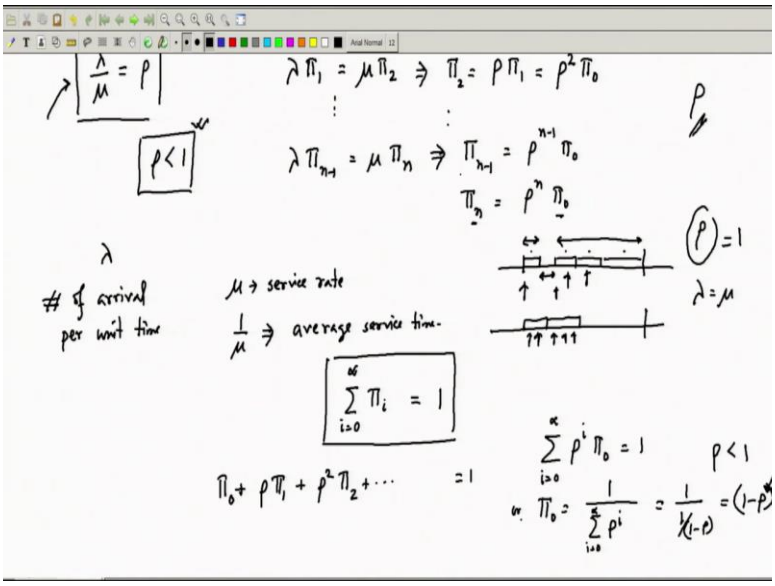
So, this will always happen. So, at least that condition has to be satisfied, what does; that mean? This condition this rho less than 1 must be always satisfied because lambda is less than mu. So, therefore, lambda by mu must be less than 1. So, this condition has to be true.

So, over here if I just replace this definition of lambda with mu, I can write from here pi 1 is equal to rho into pi 0. Similarly, from here what do we get? Pi 2 equals to some rho into pi 1. Now, I can replace this definition of pi 1. So, basically it becomes rho square pi 0. Similarly, it happens. So, finally, this pi n minus 1 will be rho to the power n minus 1 into pi 0, and pi n in general will be rho to the power n pi 0.

So, in these sets of equations we know who is something that is the characteristics of the traffic I must be knowing that. Because, if I do not know the characteristics of the traffic what is the arrival rate what is the amount of service it is bringing average on an average. If I do not know that, then there is no point in characterizing this whole thing ok.

So, that rho must be known and rho is actually this lambda by mu this is told to be the amount of service that a particular input or arriving traffic is bringing in let us try to understand this part a little bit more carefully.

(Refer Slide Time: 18:49)



$\boxed{\frac{\lambda}{\mu} = \rho}$

$\boxed{\rho < 1}$

$\lambda \pi_1 = \mu \pi_2 \Rightarrow \pi_2 = \rho \pi_1 = \rho^2 \pi_0$

$\vdots$

$\lambda \pi_{n-1} = \mu \pi_n \Rightarrow \pi_{n-1} = \rho^{n-1} \pi_0$

$\pi_n = \rho^n \pi_0$

ρ

λ

# of arrival per unit time

$\mu \rightarrow$ service rate

$\frac{1}{\mu} \Rightarrow$ average service time.

$\boxed{\sum_{i=0}^{\infty} \pi_i = 1}$

$\pi_0 + \rho \pi_1 + \rho^2 \pi_2 + \cdots = 1$

$(\rho) = 1$

$\lambda = \mu$

$\sum_{i=0}^{\infty} \rho^i \pi_0 = 1 \quad \rho < 1$

$\therefore \pi_0 = \frac{1}{\sum_{i=0}^{\infty} \rho^i} = \frac{1}{1/(1-\rho)} = (1-\rho)$

So, what we are saying? What is lambda? Lambda is the average number of arrivals per unit time we know that. Number of because it is the rate number of arrivals per unit of time. So, basically, if you talk about unit time as 1 second. So, 1 second lambda number of arrival will be happening.

What is mu? Mu is the service rate we also know what the definition of 1 by mu is according to the Poisson this one, is the average service time. So, lambda number of average arrivals are occurring if I serve them the average service time is 1 by mu.

So, on average how much time per unit time, if I can serve all of them? So, suppose some of them are arriving I am actually putting them one after another. So, it just says that if I take the average of these things multiplied by lambda that is within unit time what is the fraction of time being occupied as you can see ok.

So, the average fraction of time that is being occupied. So, that is why because rho is less than 1, generally if rho is greater than 1 then it will be more than 1. Basically, the fraction of time it requires from the system or from the server will be actually greater than 1 which is not possible. The server will not be able to handle that is why the queue keeps on growing ok.

So, it will be something like this. The situation will be that there are multiple arrivals. So, you are trying to accommodate them, but by that time some more arrivals are happening. So, you are not capable of handling all of them ok? So, it will always be difficult to handle all these things because the arrivals are happening too fast; whereas if you do an arrival which is which is below the service then there will be some gaps in between.

So, rho actually characterizes the amount of time out of that unit time ok or you if you take a very long this one, what is the fraction of time that the server is getting occupied; that means, this is the amount of service that is brought in for the server.

You remember for the lees graph also we discussed how much amount of time the server will be occupied or a particular the things switch port will be occupied. So, that is actually rho and this is characterized as the amount of traffic, and the unit is Erlang ok. So, basically when I say 1 Erlang traffic; that means, the server will be completely occupied. The amount of traffic that is being brought in or it means actually 1 Erlang means, lambda is equal to mu ok.

So, the server targeted server will be completely occupied if I say 1 Erlang. If I say point five Erlang; that means, the server will be 50 percent occupied. So, that is the basic definition of rho ok. So, once we have defined this rho, now can I just try to see how do we solve this equation in a recursive manner.

So, first thing we will have to solve as you can see each of them are getting linked to this pi 0. So, therefore, I have to first solve pi 0 because everybody is defined with respect to pi 0. So, everybody will be defined if I can define the pi 0 ok. How do I define pi 0 now? I have another equation left I have always talked about that equation which is the normalization equation.

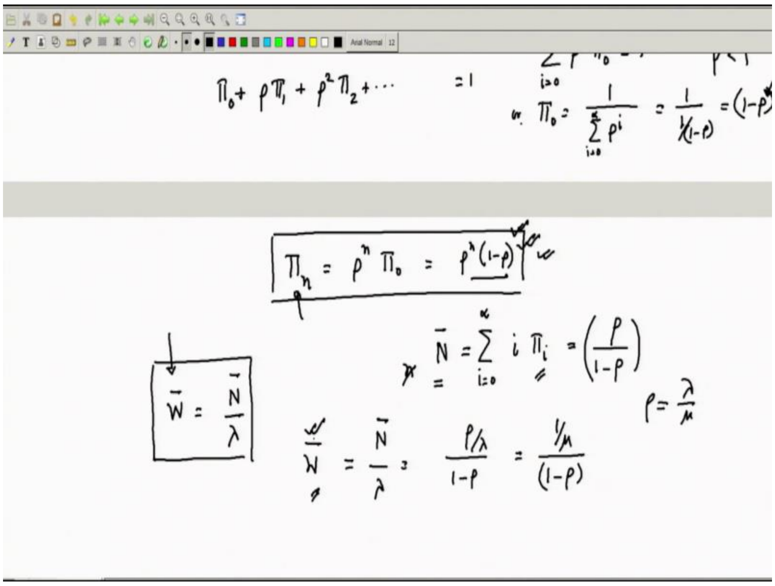
The summation of pi I goes from 0 to infinity because there are infinite states and the state starts from 0. This is equal to 1, this is something I know. If I put all those pi I over here then I will be probably getting my pi 0 solution because that is the only unknown that would be remaining.

So, let us try to do that. So, basically, this will be pi 0 plus rho into pi 1 sorry rho into pi 0, that is pi 1 plus rho square into pi 2, and so on right? This is equal to 1 or I can write summation i called to 0 to infinity pi I can write rho to the power i pi 0 is equal to 1 or from here pi 0 goes out because that does not depend on i.

So, pi 0 will be equal to 1 by summation rho to the power I, i goes from 0 to infinity. This is a geometric series and rho is less than 1. This is something I know already if that is the case this is a geometric series we know that its summation is 1 by 1 minus rho. So, therefore, this will be 1 by 1 by 1 minus rho or this is 1 minus rho.

So, pi 0 happens to be 1 minus rho again rho is a given parameter ok. So, rho is something that is already given to me, once I know this rho I will be able to calculate pi 0. Once I know pi 0, now you can see all the pi n’s are calculable. So, basically what we can do now? Wait a second ok.

(Refer Slide Time: 24:30)



$$\Pi_0 + \rho\Pi_1 + \rho^2\Pi_2 + \cdots = 1$$

$$\sum_{i=0} \rho^i \Pi_0 \quad \rho < 1$$

$$\Pi_0 = \frac{1}{\sum_{i=0}^{\infty} \rho^i} = \frac{1}{1/(1-\rho)} = (1-\rho)$$

$$\boxed{\Pi_n = \rho^n \Pi_0 = \rho^n(1-\rho)}$$

$$\bar{N} = \sum_{i=0}^{\infty} i\,\Pi_i = \left(\frac{\rho}{1-\rho}\right)$$

$$\rho = \frac{\lambda}{\mu}$$

$$\boxed{\bar{W} = \frac{\bar{N}}{\lambda}}$$

$$\bar{W} = \frac{\bar{N}}{\lambda} = \frac{\rho/\lambda}{1-\rho} = \frac{1/\mu}{(1-\rho)}$$

So, now I can define any pi n as equal to rho to the power n pi 0 which is nothing but rho to the power n 1 minus rho. I have solved the state equations and I have got the corresponding every state. So, I have got the analytical formula for this. So, this is the famous formula of M M 1 queue.

So, in the M M 1 queue, the states are defined by this as you can see this is a very well-known distribution, this is called geometric distribution again you can see a memoryless thing has been reproduced. So, this is a geometric distribution with parameter rho right. And, of course, it is a, it is a P M F because n values are discrete it can take 0, 1, 2, 3 those values are only ok.

So, I have got the distribution of a number of customers. Now, all we have to do is we have to take the average of this; average number of customers ok. So, how do I take the average? I know the PMF is average so I will have to take it. So, n or you can even write i pi i summation i cross to 0 to infinity that should be my n bar ok.

So, all we will have to do now? We have to calculate this, this summation is a very easy means you have done that in your 12th standard math. So, this summation can be easily calculated as pi I just replace that should be rho to the power I pi 0 and pi 0. You can put or you can pi I you can put this rho to the power n 1 minus rho you put that and then you do that summation and the basic formula will come down to be ok.

So, once I know the n bar it is very easy to calculate know average number of customers in the system. Now, how do I get the delay? You remember my delay was the average number of customers divided by lambda which is the Little’s formula. So, I put Little’s formula I get my delay also average delay.

So, average delay will be this n bar divided by lambda which is rho 1 minus rho and divided by lambda. Now, rho is lambda by mu divided by lambda that should be. So, this is rho. So, divided by lambda that should become 1 by mu. So, I can write it 1 by mu that is my average waiting time ok?

So, as you can see once I know the construction and how to construct ACTMC from there I can easily analyze the average condition that we have talked about the average waiting time. So, that is something we can calculate an average number of customers in the system or the average number of packets in the system. If it is an M M 1 queue that is characterizing the circuit switching or call switching then I can see the average number of calls that are waiting in the system I can characterize that.

The average amount of waiting time for each call to be established. So, I can characterize that as W bar. So, these are the parameters we can characterize we can also see the number of customers their basic number of customers, their distribution, and their PMF that we can do.

Even waiting time also we can calculate the associated distribution that will be exponentially distributed, but right now probably we will skip that because we are not trying to do the M M 1 queue over here. So, what we will do? With this background now, we can go ahead and do the derivation of other important trans-switching calculations.

So, next class we will try to do that, we will try to see what is the system that is being associated with it and how we analyze that system ok.

Thank you.