

Performance Evaluation of Computer Systems
Prof. Krishna Moorthy Sivalingam
Department of Computer Science and Engineering

Indian Institute of Technology, Madras
Lecture No. # 12 (B)
Queuing Theory – I (Continued)

We computed p_1 to be ρ into p_0 .

(Refer Slide Time: 00:10)

$$\begin{bmatrix} p_0 & p_1 & p_2 & \dots \end{bmatrix} \begin{bmatrix} -\lambda & \lambda & 0 & \dots \\ \mu & -(\lambda+\mu) & \lambda & \dots \\ 0 & \mu & -(\lambda+\mu) & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix} = 0$$
$$-\lambda p_0 + \mu p_1 = 0$$
$$\therefore \mu p_1 = \lambda p_0$$
$$p_1 = \frac{\lambda}{\mu} p_0 = \rho p_0$$

Let $\rho = \lambda/\mu$ & $\rho < 1$

And, this is the second balance equation.

(Refer Slide Time: 00:16)

$$\begin{aligned}\lambda p_0 - (\lambda + \mu) p_1 + \mu p_2 &= 0 \\ \mu p_1 - (\lambda + \mu) p_1 + \mu p_2 &= 0 \\ \mu p_2 &= \lambda p_1 \\ p_2 &= \left(\frac{\lambda}{\mu}\right) p_1 = \left(\frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right) p_0 \\ &= \rho^2 p_0.\end{aligned}$$

This if... Again, just working out simply, we get back p_2 equals ρ square p naught. And, since the same equation is repeating for all subsequent values or whatever be the value of n , n minus 1 and n plus 1 I can simply work this out.

(Refer Slide Time: 00:34)

$$\begin{aligned}\therefore p_n &= \rho^n p_0 \\ \sum_{i=0}^{\infty} p_i &= 1 \\ \therefore \sum_{i=0}^{\infty} \rho^i p_0 &= 1, \rho < 1 \\ p_0 \cdot \frac{1}{(1-\rho)} &= 1, \rho < 1 \\ p_0 &= 1 - \rho\end{aligned}$$

In full glory, we will come back to this p_n equals ρ to the power n into p naught. So, to calculate the value of the p naught, we need to use the normalizing equation that all the probabilities p_i will add up to 1. So, we have sigma i equals 0 to infinity ρ to the power i p naught equals 1. And, that only will work if ρ is less than 1. (()) already. So,

if rho equals 1, it goes to infinity; if rho is greater than 1, it definitely goes to infinity. So, that is the mathematical way of saying why this rho has to be less than 1. Intuitively people have asked me why should rho equal to 1, why will system become unstable if rho becomes 1? And, because the number of customer, because this any way... These are exponential variables; it is not that it is a deterministic system. And, I still have not yet found intuitive explanation, why rho cannot equal to be equal to 1. Mathematically, we can say yes, if rho equal to 1... Then, what will happen if rho equals 1? Actually, what will be the system? If rho equals 1, what will happen?

[Not audible] (Refer Slide Time: 01:29)

For every arrival... Yeah, that is correct, but that is exponentially distributed. So, if lambda equals mu equals 1 and we look at our distribution..., look at the same Markov chain, where all the values go to 1, what will happen is, it will be simply 1 over infinity.

(Refer Slide Time: 01:50)

$$[p_0 \ p_1 \ p_2 \ \dots] \begin{bmatrix} -\lambda & \lambda & 0 & \dots \\ \mu & -(\lambda+\mu) & \lambda & \dots \\ 0 & \mu & -(\lambda+\mu) & \dots \end{bmatrix} = 0$$

$$-\lambda p_0 + \mu p_1 = 0$$

$$\therefore \mu p_1 = \lambda p_0$$

$$p_1 = \frac{\lambda}{\mu} p_0 = \rho p_0$$

Let $\rho = \lambda/\mu$ & $\rho < 1$

See what will happen is? We will see why it becomes an unstable system. See mu equals lambda. So, basically, p naught equals p 1. So, all the probabilities become equal.

That will also remain in one state only, because one is coming and one is going (Refer Slide Time: 02:10).

So, you are saying the system say... No, that is not correctly said; that is, it will be unstable; it will actually go towards infinity; it will be... See you have p naught equals p

1 all the way up to p infinity. So, if we apply $\sum p_i = 1$, what will happen? p naught will be 1 over infinity. So, it will become 0 . So, probability of being at any state is 0 . So, the system will never stay in a given state. So, it keeps on moving to the next state, next state, next state and so on.

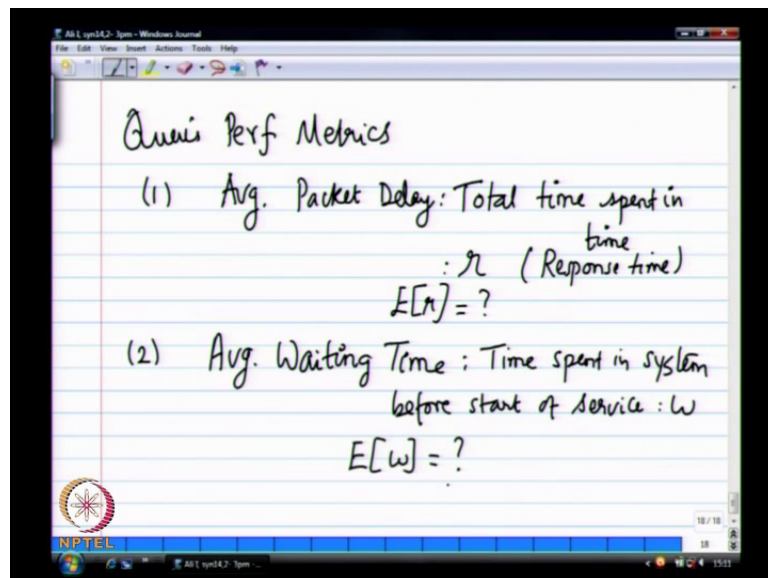
But, it is like mathematically $(())$ take it otherwise, while saying that λ equal to μ ; that means, the number of $(())$ which are coming, same are being take out.

Only thing is the arrivals times are different.

But, it is exponentially distributed (Refer Slide Time: 02:51) no?

See the times of actual arrival; even though it is 1 and 1 , the actual times of arrival is random. And, because of that, again, the system keeps on building up and it ends up being in none. It will be infinite state system; or, a queue size will be infinite in that particular system. So, we derive that (Refer Slide Time: 03:10) p naught equals 1 minus ρ . So, what does this mean now?

(Refer Slide Time: 03:25)



Now, let us look at what should we do with all these probabilities at your calculator. For that, we wanted to define some metrics for the queue. Queue's performance metrics – one – in terms of the customer there is a delay; we said average delay. We will say average packet delay and this is represented by the total time spent in the system. So, total time – from arriving to the system to departure from the system; entire time spent in

the system is 1. We will use this variable to be r ; so, the expected value will be E of r to follow the book notation. So, r stands for response time. This is the response time. So, what is this E of r , so that we need to compute?

Then, we also want to know what is the waiting time in the system, because the service for a particular customer we know; it is $1/\mu$, because μ packets per second or μ customer per second are being serviced. So, the average service time is $1/\mu$. Therefore, each customer we know it is $1/\mu$. We are sometimes interested in the waiting time of a particular system. So, in some cases, what will happen is, in some systems, the waiting will be less for one system compared to the other, but the total time will be different. So, we have to look at sometimes waiting time also. This is the (Refer Slide Time: 05:12) average waiting time that the book uses w we hope. This is the time spent in the system before start of service. So, waiting time is simply the time spent before start of service. And, we will call this to be w . So, I need to know what is the average waiting time. And, that average...

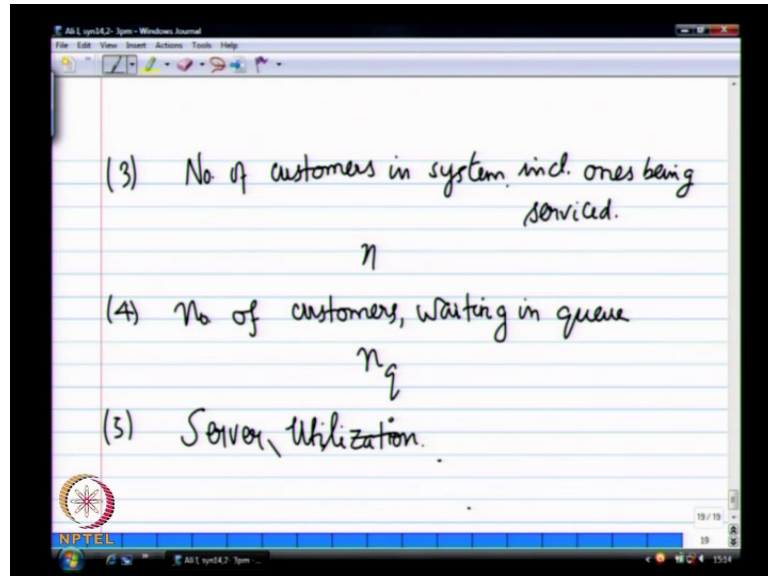
Lot of service because the queue can also packet can also go back (Refer Slide Time: 06:00).

This is the simple open-ended queue. There is no going back to this queue; you enter, you leave. Since you know, it is an open system; it is not a closed loop system. We will come to queue networks where you actually go back; the one that we saw last time where we actually go back. So, that will be a closed network system. So, this is simply an open system; customer arrives and leaves; that is all. Then, we need to know what the average waiting time is. So, that is one more thing that we can try to figure out. And then, another metrics; so, this relates to know buffer size.

Again, looking at the system, we would like to know what is the average buffer occupancy, because if you are deciding a router and then you are putting in giga byte of memory, you find that most of the time only about of you kilo bytes or mega bytes is being used. So, why invest in a giga byte of memory, because memory is expensive in routers and also power consuming. So, you sometimes want to know what is the average number of customers in the system, average number of customers in the queue; number of customers in the system equals the ones waiting plus the ones being serviced or the

average number of customers simply waiting in the buffer; that is also something, which we should look at.

(Refer Slide Time: 07:15)



So, this is the number of customers in the system including the ones being serviced. I am saying ones because a way of multi-server system; then, I can have more than one customer being currently serviced. And, unfortunately, the variable for that used is n , which is appearing all over the place. But, when I say E of n , you know what I am talking about. And, the number of customers – this is now only in the queue; this is just of the buffer occupancy. So, what is the average number of customers simply sitting in the queue? That it is called n_q . So, the first two are from the customers perspective, what is the waiting time. The customer does not care how many customers are there in a particular queue or whether the queue is fully occupied or less/half occupied and so on, except in the cases where sometimes you go to a queue, multiples queues are there. Each of them, each counter having its own queue; and, all of them are giving the same service. And, they say pick anyone; you will always pick the shortest one. Only at that time, the customer can visually see what the current queue length is; and, based on that, you can take some decision. But, the queue length is more of a system level parameter or metric. System wants to know how many customers are waiting. Based on that, the manager has to come in and add more tellers or make this guy go faster; processes being faster, so we can have the queue size to be less.

Then, there is also one more, which is the utilization of the server. So, what is the server utilization? Server utilization is simply the fraction of time that the server is being utilized. That is also metric, because if I am going to put in five people for the job in a bank and if I find that half the time or most of the time, an average, only two people are actually servicing the customers. So, utilization is 2 out of 5. So, I am paying five people salary and getting two people worth of work, which means I can remove some customers or clients or remove some servers and put them in some other branch, where there is no **(C)** So, these are some of the metrics. And then, here (Refer Slide Time: 09:30) it is not just the total time. This E of r is simply the expected time. Sometimes, I also want know the distribution of response time and what is the variance in response time; all those things are there.

Now, we know that what we want to measure. So, let us see how we can use this p s. Ultimately, all are computed so far is the p i. How can I use the p i's to determine any one of those metrics? Does any of those metrics look calculatable based on the p i values that we know? Number of customers; at every state... I know in state n, there are n customers in the system. Therefore, I can first compute E of n; so, I know the probability of being in state n; and, I know that these are rewards that we saw. In the last tutorial, you calculated for... If you know p i, then the reward of that state is the reward of 10 dollars, 1 dollar and so on.

(Refer Slide Time: 10:47)

$$\begin{aligned}
 (1) \quad E[n] &= \sum_{n=0}^{\infty} n p_n \\
 &= \sum_{n=0}^{\infty} n p^n (1-p), \quad p < 1 \\
 &= p(1-p) \sum_{n=0}^{\infty} n p^{n-1} \\
 &= \frac{p}{1-p}, \quad p < 1
 \end{aligned}$$

And, this is n going from 0 to infinity. Again, ρ should be less than 1.

[No audio] (Refer Slide Time: 10:50 to 11:21)

So, what is this work out equal to?

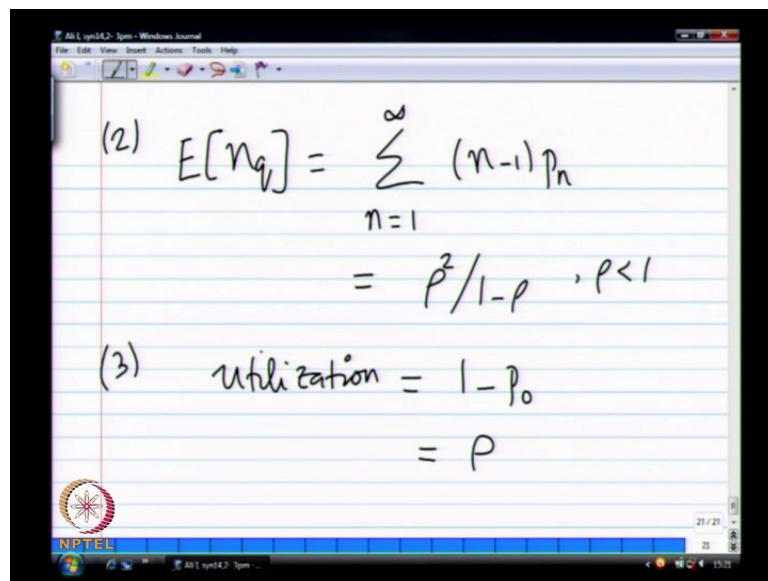
[No audio] (Refer Slide Time: 11:25 to 11:57)

So, this is again similar to the geometric variable derivation. We will not go through all that, but simply ρ by $1 - \rho$. So, this is the average number of customers in the system. So, we have found something, one metric. One good thing is the closed form. Now, what else can I derive? How do I go forward? Can we derive the E of n_q , number of customer waiting?

[Noise – not audible] (Refer Slide Time: 13:02)

So, I could derive that too.

(Refer Slide Time: 13:13)



The image shows a handwritten slide with two equations. The first equation is labeled (2) and shows the expected number of customers in the queue, $E[n_q]$, as a sum from $n=1$ to infinity of $(n-1)p_n$. This is simplified to $\rho^2 / (1-\rho)$ for $\rho < 1$. The second equation is labeled (3) and shows that utilization is equal to $1 - p_0$, which is also equal to ρ . The slide includes a Windows taskbar at the bottom with the NPTEL logo and a date/time display of 21/21.

$$(2) \quad E[n_q] = \sum_{n=1}^{\infty} (n-1)p_n$$
$$= \rho^2 / (1-\rho), \quad \rho < 1$$
$$(3) \quad \text{utilization} = 1 - p_0$$
$$= \rho$$

This minus 1 sir...

You know minus 1?

I am asking.

If I am state n with five customers, what does it mean? There are four being waiting and one being serviced, because of the $m=1$. So, it is always $n-1$ into p^n , except that n goes from 1 to infinity, because $n=0$; there is nobody waiting. So, the summation goes from 1 to infinity, which we can arrive again, derive like. so we can go through the derivation for this one too, which will turn out to be ρ^2 by $1-\rho$. You are saying something? This after write simple derivation, you will get. So, this is something (()) that we can derive.

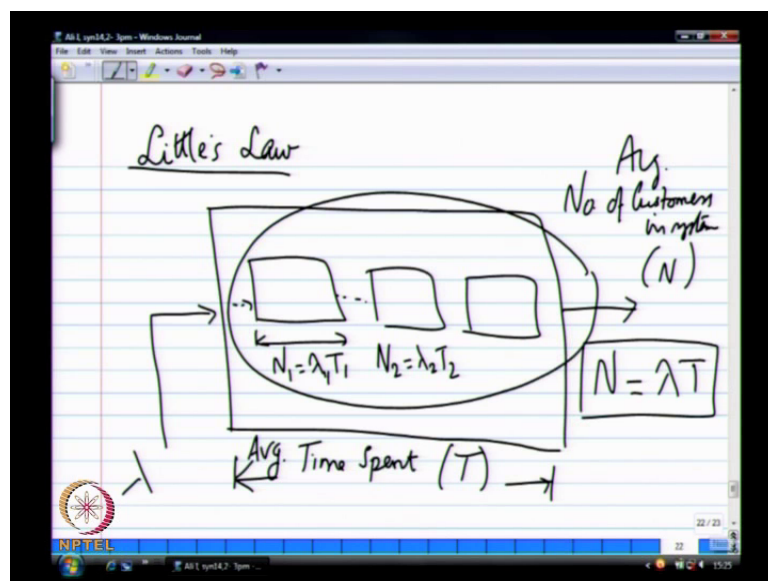
Now, what do we do? Now, what is the utilization of the system? For what fraction of time is the server being utilized?

Except when the state is 0.

So, expect when the state is 0.

Or, even when the server is being utilized. So, it is basically $1-p$ is what you are saying. p is the state the system is being idle. No customer's is p . Therefore, $1-p$ is the ... and p by the $1-\rho$. So, this is simply ρ (Refer Slide Time: 14:46). I simply said ρ , because λ/μ ; fraction of packets arriving to the fraction of packets leaving is simply ... But, you can derive at this fate. It is not always ... So, utilization of the system is simply ρ .

(Refer Slide Time: 15:30)



Now, we want to calculate delay. So, how on earth you get delay? So far, there is nothing that gives us delay for a particular packet. So, for that, we need to take a side track and look at something called Little's law. You look at any system as a black box and let the system have sub blocks also inside. So, this is the system that is handling some processing whether it is cash request or customers arriving. So, into this black box, I have customers arriving at rate λ . So, λ is the rate of something entered. This is not even poised on nothing; it is simply plain λ . This is simply arrivals per second; that is all. Then, I can measure the total time spent in the system. This is the average time spent. So, I entered the system; I spent some amount of time, which we will call that T ; average time spent in the system is equal to T . So, this is arrivals per unit; this is average time spent in the system in the same T units.

And then, I will simply count; I have not shown how to show. But, let us say the total number of customers at a given point in time; so, is the average number of customers. We call this N . So, N is simply number of arrivals into average time spent into the system. The average number of customers is simply arrivals into average time spent in the system, which can do with the simple integration equation and show that this is true, but will not go back. So, all I need to know is if I know the arrival rate and if I know the average time spent in the system, I can compute the average number of customers in this black box or vice-versa. Any two of these I have, I can calculate the third one. And, this is true of even sub blocks. So, if the sub blocks $(())$ also, I can apply the same thing. So, for any subsystem inside the system, even for that, if I say this is N_1 , this is $N_1 \lambda_1 T_1$ (Refer Slide Time: 18:37). So, for all sub blocks also, this relationship codes. So, in general, any black box where customers are being, this will $(())$. So, this is $N_2 = \lambda_2 T_2$.

Remember, we saw that in that previous example, very CPU, we go to one of those queues. So, even though the individual queues, you can apply little slot to find out; if I know 2 or 3 parameters for the small device queue, I can again calculate the third parameter for each of those queues; or, you can look at the entire system and calculate the overall time spent in the system; so, for even of the CPU queue a lot. So, this applicable to any sub part of the system; I can use it as needed. This little slot is what will now help us move forward after that E of n q .

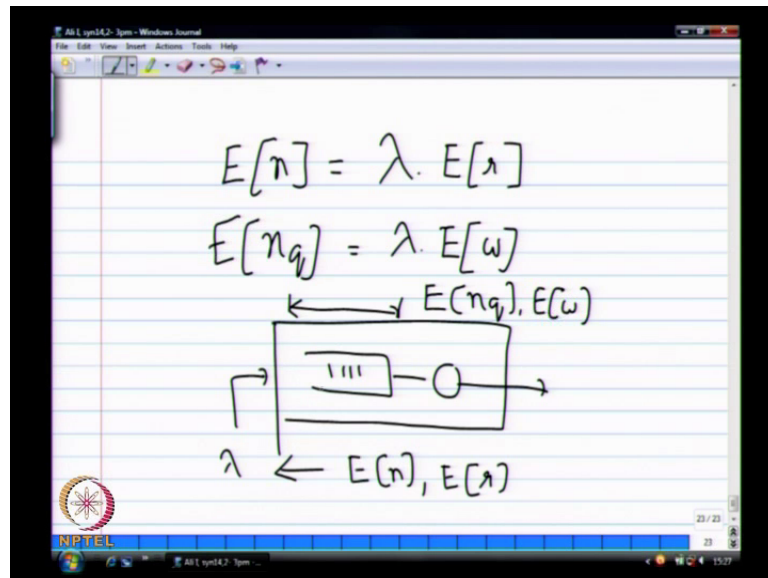
Sir, is there any assumption $(())$ (Refer Slide Time: 19:22)

No, there is no assumption or distribution at all. Only looking at the mean time spent in the system to the mean number of customers in the system.

[Not audible] (Refer Slide Time: 19:27)

I will have to draw that. So, nobody staying in the system; true.

(Refer Slide Time: 20:01)



With that little slot, now I can start equating. So, E of n is the total number of customers in the system; that is now equated to total arrivals to this queue into the average response time. The total time spent in the system is r . So, E of n equals λ into E of r . And, E of n q equals λ into E of n q is w . So, n q is the number of customers in the queue and w is the time spent in the queue. Therefore, E of n q equals λ E of w .

[Noise – not audible] (Refer Slide Time: 20:45)

E of n q ; so, this is sub system that we talked about. If I look at the queue...

How you calculated the number of customers in...

How? Let me just draw this picture for the sake of completion. So, this is E of n and then E of r . This is λ ; this little part of the system, where including the server is where E of n q and E of w occur. To both systems, the packets arriving are at rate λ ; and, there is no packet draft in the system. λ is effective. All the packets entered this

queue. And therefore, n q and E w can be equated; likewise, E of n and E of r can be equated. Questions?

Have you got that sir, p n ?

P n is a probability of being in state n . So, when I am in state n , how many customers are waiting for service? State n means I have 5 customers in the system; one is being serviced. So, there are 4 cases waiting for service.

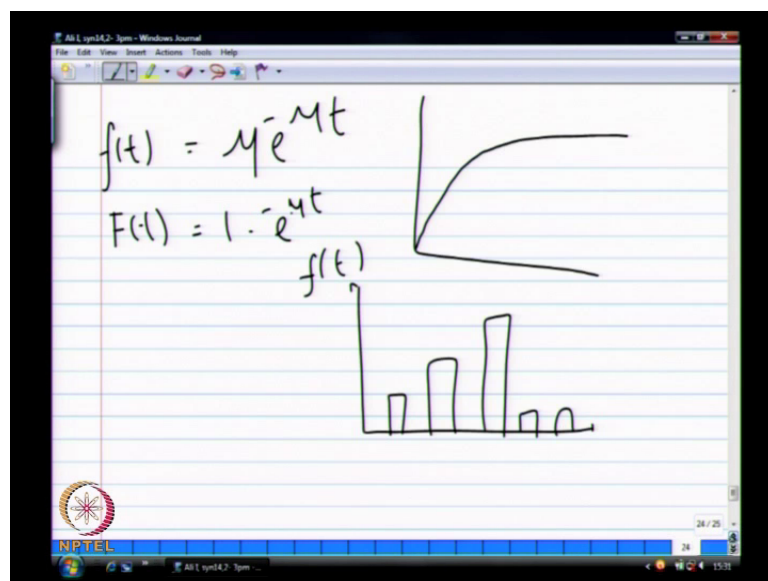
[Not audible] (Refer Slide Time: 21:51) ...when we have only one server.

Yes, correct. When you go, we can do the same thing. It will be n equals m to infinity and it will be n minus the number of ... If it is less than m , that will be nobody waiting. If I have 5 queues, 5 servers, then when the system is in state 0, 1, 2, 3, 4, 5, there will be no waiting at all; only when the state goes to 6th, we will start having waiting in the queue. So, it will be simply n minus m into p m ; that is all. And, n will go from 0 to infinity. Also, you want infinity or other n plus m ; m plus 1.

When we say service time distribution is μ , what do we actually mean?

When I say the service time distribution is exponential?

(Refer Slide Time: 22:56)



That again if you ... So, this is μ . It means that the service time distribution would be given by f of t . This is what the definition is. Where there is ... Remember that if you

draw the f of t ... that is correct. So, whenever you find a system, where you are measuring the cumulative CDF and if it is having this particular... If it fits this pattern, then you say that it is exponential.

That I understood; physically, I am asking.

Physically, that is what.

In real science, how do you say k service time distribution?

You will have to do reverse fitting. So, use it and measure. You measure the service times for all the customers in this system. And, you take the probabilities of whether the time for each of those service times being 1 unit, being 2 units, 3 units, and so on. And then, you try to see if it fits this particular distribution. If it does, then you say it is exponential.

We plotted like this number of customers...

Yes, you have to do; essentially, you are trying to fit set of sample points to some known distribution. If you are measuring the service times and let us say that you have... This is the F of t that you are trying to do or F of whatever you say $(())$ service time. So, we are saying what is the probability that the service time will be one unit. Let us say that this is just discrete set of values; it is not... We try to get something like this (Refer Slide Time: 24:16). There are only five possible values; it is not exactly a continuous distribution; it is read only five possible values; everything else goes to 0. So, this is the distribution. You cannot say this is exponential, because this is what you observed from say million samples; you go to 2 million samples may be this will change. You got to know $(())$ set. So, that is basically looking at the service time and seeing a bit maps to this nice convenient distribution, because if it does, we get this nice properties of memory less and so on. But, in real life, most likely it will not, but we just make the assumption. We did that assumption and we tried to do for example, packet queues.

Packet lengths are never exponential. First of all, packet lengths are also only discrete set; it will not go from 0 to infinity; it is not going to be the case; whereas, here anything greater than 0 is it is defined for. But, we are simply making that assumption just for the sake of convenience. You are saying my atm multiplexer is an mm 1 queue. Then, you just use that assumption to get some closed form results for what the delay will be.

If you calculate response time, suppose the packet is serviced at i th state and expected time for all the packets to be serviced in that state. Probability of the packet coming in i th state, entering a i th state is equal to the steady state probability of the i th state into the expected time in that i th state, service time for n customers in that i th state. **So, you are saying it is...**

n by μ and n minus μ , n by μ . If you are in state n , the varying time is n by μ . n by μ because μ is the average service time. So, it is n by μ into steady state probability. OK, we will check it out; we can calculate that. So, we will try both ways and see if it gives the same equation. I have not tried that, but we can try.

(Refer Slide Time: 26:19)

The image shows a whiteboard with the following handwritten equations:

$$E[n] = \frac{\rho}{1-\rho}, \quad \rho < 1$$

$$E[r] = \frac{1}{\lambda} \cdot E[n]$$

$$= \frac{1}{\lambda} \cdot \frac{\lambda}{\mu(1-\rho)}$$

$$= \frac{1}{\mu - \lambda}, \quad \rho < 1$$

At the bottom left, there is a note: "DF of r , $F(r) = 1 - e^{-r(\mu - \lambda)}$ ".

Now, E of n we know; E of n is ρ by 1 minus ρ for ρ less than (ρ) . Then, E of r is now 1 over λ into E of n . So, 1 over λ ; this is λ by μ into 1 minus λ by μ . So, what do we have? So, this (Refer Slide Time: 27:08) is an equation that you should never be able to forget. So, expected time is simply 1 over μ minus λ . That is why again ρ should be less than λ ; otherwise, it does not make sense. If ρ equals λ , again you will find that E of r goes to infinity, which is again correct if ρ ... Again mathematically speaking, truly speaking, I will think about it. Mathematically, if ρ equals μ , you will land up ρ equals 1 ; it will land up with infinite delay. In fact, if we look at the response time distribution itself as a random variable, I will just state that the CDF of r itself, F of r is actually exponential. The

response time is exponentially distributed with parameter $\mu - \lambda$. That is an interpretation of that; I will just leave it at that for now; let us not go any further (()) But, that this main thing is this – the average delay is $\frac{1}{\mu - \lambda}$.

(Refer Slide Time: 29:00)

The image shows a whiteboard with the following handwritten equations:

$$E[r] = E[w] + \frac{1}{\mu}$$

$$\therefore E[w] = E[r] - \frac{1}{\mu}$$

$$= \frac{1}{\mu - \lambda} - \frac{1}{\mu}$$

$$= \frac{1}{\mu} \frac{\rho}{1 - \rho}$$

$$E[w] = \frac{E[nq]}{\lambda} = \frac{1}{\lambda} \cdot \frac{\rho^2}{1 - \rho} = \frac{1}{\mu} \frac{\rho}{1 - \rho}$$

Now, there is another relationship that exists between E of r and E of w . So, the total response time is simply the waiting time plus the average service time for this particular (()). Therefore, I can calculate E of w as E of r minus $\frac{1}{\mu}$; $\frac{1}{\mu} - \frac{\lambda}{\mu(\mu - \lambda)}$, which we can sort of massage this and actually end up with $\frac{1}{\mu - \lambda}$. We can also have derived E of w from E of nq (Refer Slide Time: 30:27). E of nq we said was $\frac{\rho^2}{1 - \rho}$. And then, we could have applied the same thing – basically, $\frac{1}{\lambda} \cdot \frac{\rho^2}{1 - \rho}$. So, if I take out $\frac{1}{\lambda}$ from there, $\frac{\rho^2}{1 - \rho}$ by λ is nothing but $\frac{1}{\mu}$.

There is an extra...

There is an extra?

Rho square is lambda square over mu square.

If take out one lambda, one rho will still have rho by $1 - \rho$; and, rho by lambda is nothing but $\frac{1}{\mu}$. So, we can derive this way. So, in this form, if you look at E of w is $\frac{1}{\mu} \frac{\rho}{1 - \rho}$ into this particular fraction, this is one way of looking at $\frac{1}{\mu}$ as (()) easy to remember. But, actually to interpret this, I would rather use this one (Refer Slide

Time: 32:01) – $1/\mu$ into $1/(1-\rho)$. So, what this says is, $1/\mu$ is the average service time per packet. So, the average service time, the system response time depends upon is inversely proportional to μ or, actually know what is μ . If it is $1-\rho$, then the higher the utilization, higher the delay, is directly proportional to the utilization. But, this $1/\mu$ is a multiplicative factor that appears for both $E[r]$ and $E[w]$. So, when you are trying to design systems, always make sure that the $1/\mu$ factor goes to as small as possible, because everything is proportional to $1/\mu$. So, smaller the service time, which means faster the link, faster the server. Then, the overall response time is going to be much better, will be lower. Now, that you have seen all these equations, unless we see some numbers, probably it will not be fun.

(Refer Slide Time: 33:04)

E.g. M/M/1 queue
 $\lambda = 125 \text{ pps}$
 Avg. service time = 2 ms
 $\mu = \frac{1}{2 \text{ ms}} = 500 \text{ pps.}$
 $E[r] = \frac{1}{375} = 2.66 \text{ ms}$
 $P = 0.25$
 $E[w] = 2.66 - 2 = 0.66$
 $E[n] = \rho/(1-\rho) = 1/3$

We will take one example. I will just start off by simply saying arrival is 125 packets per second and the average service time is 2 milliseconds. Assume that this is an M M 1 queue, is a packet queue. So, if you remember the internals of a router, router will have queues and for each queue packets are arriving; and, let us pretend that the packets are arriving at some poisson rate. That is given by lambda. And, average service time, which is 2 milliseconds. Nobody said that the average time for packet is exponential, which is never the case. But, if somebody says what is the average delay spent by a packet in this particular queue, we will say well, because it is not deterministic, I presume that it is exponentially distributed and then I will try to get some close formed solution. So, μ is simply $1/2$ milliseconds; 500 packets per second.

Now, I can say E of r is 1 over 375. So, what is E of r? 2.66 milliseconds. It is good that it is greater than mean service time; otherwise, something went wrong with the formula. Therefore, E of w is simply 2.66 minus 2; that is, 0.66 and so on. So, E of n is rho by 1 minus rho. So, technically, always calculate rho first. But, sometimes we ignore that rho (()) So, what is here? So, here rho is 0.25; 1 by 4.

[Noise – Not audible] (Refer Slide Time: 35:54)

We can do lamda here also. In this case, the different ways calculate that. So, this is simply 0.25 by 0.75. Therefore, that is 1 by 3 and so on. Now, we get a hang of numbers. Now, we have the purpose of actually going through all these calculations. So, either we do not learn about mark question and simply use a formulae directly. But, it is good to know how we got these, so that we can use this for analyzing other systems. So, these are systems that have been analyzed to several decades ago. But, this is the way that you got some of these close form equations. Like there are some other ways of computing.

There is one more example. Questions? This is one example here that is interesting. You should look at this; there is a long list of formulae; I am only giving you some of the expressions.

(Refer Slide Time: 37:05)

Box 31.1 on p 525 of Jain's book

q -percentile of response time = $E(r) \ln \left[\frac{100}{100-q} \right]$

$q = 90^{\text{th}}$ percentile

$= 2.3 E(r)$

In prev. example, 90^{th} percentile of $E(r) = 6.1 \text{ ms}$

If you back and look at this box 31.1 on page 525 of Jain's book, usually we summarize this. For every queue type, this is a good reference. I do not want to just know what the results are; do not care about how to derive it. This is the good way to find.

[Noise] (Refer Slide Time: 37:30)

We only know for example, the mean delay. But, what if I want to know the ninetieth percentile of delay. So, we will have to go back our statistics definitions. But, here... So, we will just give you the result for the qth percentile of response time or packet delay (Refer Slide Time: 38:37). So, people are familiar with percentile after having written... So, if you want to find out... because sometimes the mean delay alone does not tell me much. I would like to know like the some set of worst case delay see what we have what is going to happen. That gives me some... If I am trying design some systems, for example, buffering, I want to know what the average, (()) what the worst case delay is. So, buffers have to be proportional to delay; then, I try to get some of these estimates. So, in this case, if q equals 50, what do you get? What is q equals 50? 50th percentile. 50th percentile is what? 50th percentile is?

50th percentile is a median. Mean. Median

50th percentile will be... So, that will be into (Refer Slide Time: 39:53) natural log 2. Means the mean is E of r. So, if you want q equals 90, what is the 90th percentile of delay? Which means that 90 percent of the times, the delay values will fall below this. That is simply natural log of 100 by 10. So, natural log of 10. Natural log of 10 is 2.3. So, that is a fairly large number. So, the delay can have wide variation in this particular case. If in this previous example, the mean was 2.66, then the 90th percentile is 2.3 times that, which is fairly large. Then, in the previous example, what is 2.3 into 2.66? Any one has got calculator?

6.1 [Noise] (Refer Slide Time: 41:24)

6.1 milliseconds. So, if the application designer requires to know the worst case delay also, then you can get that again here. And again, this because you are making an assumption that its exponential distribution time for the service and so on, sometimes that is not... So, it just gives you an idea of how much will be the worst case delay.

Again, I can put 95 also and see what is that going to be and so on. So, there are other expressions also here. I just want you to look at them.

Now, if I say if rho equals 0.9 and idealistically, I only have capacity for 1000 packets in the system total, what is the probability that I will exceed 1000 times? What is the probability that the q occupancy will be more than 1000 packets?

[Noise]

(Refer Slide Time: 43:03)

$\rho = 0.9$
 What is prob that there will be more than 1000 packets?
 $P(\geq K \text{ customers})$

$$\sum_{n=K}^{\infty} P_n = \sum_{n=K}^{\infty} \rho^n \cdot (1-\rho)$$

$$= \rho^K$$

Just try to compute this. So, what I want to try; let us say that some n is what I am looking at. So, probability of they are being greater than equal to K customers. That is simply summation of all probabilities p_n since we $(())$ n going from K to infinity. And then, what is this? This is rho to the power n into 1 minus rho (Refer Slide Time: 45:27).

Sir, why we are $(())$ why are taking it like p_n to... (Refer Slide Time: 46:01)

This is simply the probability of that being... We are trying to find out E of n q. E of n q is only the average number of customers in the system. Here I am trying to find out probability that will be more than n customers or k customers in the system.

Nobody $(())$ (Refer Slide Time: 46:24) like how many customers $(())$ it is in that state is like.

Yes, like probability of that being more than 1000, 1001 1002 and so on. That is why we are trying to compute it here.

We know that sigma in that (()) Put n is equal to 1000 into that; probably...

Just making that as K; and then, from K to infinity, anything more than K is what I am interested in.

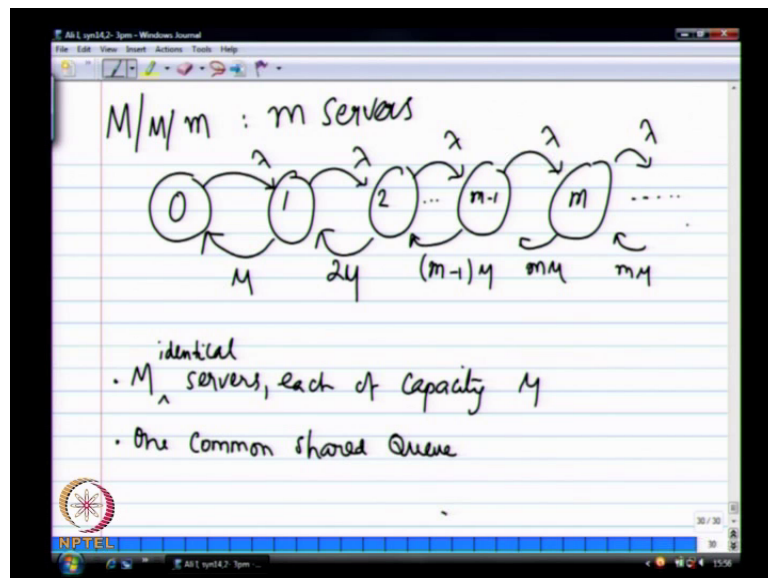
More than K

Yes, more than 1000. Greater than and equal to K customers. So, what does it converge to?

[Noise – not audible] (Refer Slide Time: 46:50)

So, we can also compute that. In some cases, it is easy to find this out. And, if you have forgotten the high school derivation, you should go back and try it out. Now, in the M M m, the next 5 minutes. Now, that I know M M 1, M M m is very similar. So, what if I have m servers? Again infinite capacity. So, what will be the Markov chain and what will be the balanced equations and what will be the final solution?

(Refer Slide Time: 48:06)



As m servers, everything else is the same as before. Now, the states again go from 0 through m minus 1, m and so on. So, this transition is still at the rate lambda, because there is only... There is only one queue being serviced by several clients. There is only

one single queue and there are several servers or clients. Now, the return service rate – if there is only one customer, one server active, what we are assuming is... So, there are m servers each of capacity μ ; and, they are identical servers. We will make sure that... And, there is one common shared queue. So, many times when we go and see queues being arranged, sometimes I put separate queues. This service is the same with any teller. Sometimes they have a common queue, sometimes they have individual queues. And, we can look at which of these is actually better. So, if there are 10 queuing $(())$, then we would figure out that one of these two is better and... because if we go to SBF, there is a single queue. It is really the token system in a single queue and $(())$ to get serviced by the next available server that is waiting.

When I have one server active, the return rate is μ ; when I have two servers active, the return rate is 2μ . And, this is again, similar to that $\lambda_1 + \lambda_2$. Either of the two servers can finish and then you will go down to the next lower state. So, this is $m - 1$. Then, subsequently, after m , we only have m servers active. So, it will always be... and so on (Refer Slide Time: 50:26). So, this is the generalized birth-death process, where $\lambda_1, \lambda_2 (())$.

Sir $(())$ (Refer Slide Time: 50:41) $m - 1$. For example, from 2 to 1, why can it only come to 1 if there are 2 servers?

As long as either of those two servers finishes, you come down the state 1.

Simultaneously... [Not audible] (Refer Slide Time: 50:54)

Assume that there is a two step process. That is where the memory less thing again comes in. So, you go from two to...

[Not audible] (Refer Slide Time: 51:07) why like this?

Because this is the effective rate of transitioning from two servers to from two customers being handled, one customer being handled.

But, it can be 0 also. No sir, from m to 2 also can be the same. Say suppose you have two customers being the same, what is the rate at which I can go to 0 customers? It can be once in a $(())$ only. Both of them $(())$

As you said, the exponential and then say... What we are looking at, we are looking at two independent processors with rate μ . And then, you are trying to find out the min of those two. Whenever one of those two events occur, the same λ_1 , λ_2 , they wait for that; we are looking at the minimum time for either or those to finish is what I am looking at; the rate for that. That is again exponential with rate $\mu + \mu = 2\mu$. That is what it is coming to. As long as either of those to finishes, you go to...

Remember, we did that derivation. Even if they are equal... x_1 equals x_2 or $x_1 \times 2$ and so on; that is taking care of that (()) So, conceptually, yes, they can go at the same time, but the rate of transition is still... It is still an exponential process with rate 2μ . That is where sometimes you try to our set of... On a daily basis, yes, I can go instantaneously; both of those can be... But, if you look at the probabilistic process that is differing, it is an exponential process with rate 2λ And that is what we trying to capture. That is OK; for now, we will just use that and then we take some time to see what that process... Look it from the probability process; not as a specific instant that is taking place at given point in time.