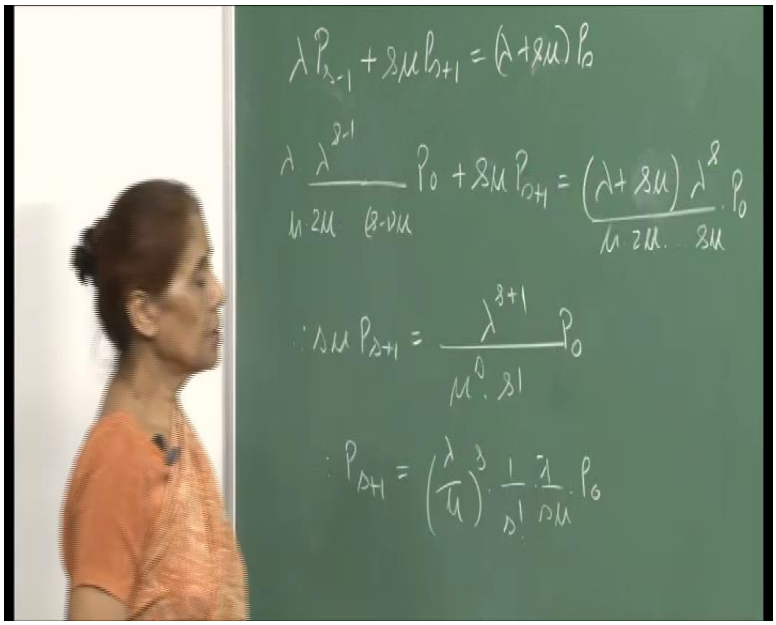
Introduction to Probability Theory and its Applications
Prof. Prabha Sharma
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture - 36
M/M/S M/M/I/K Models

(Refer Slide Time: 00:15)



So, continuing with our solutions of the balanced equations for multiple servers, so M M S system; maybe, I can again write here this because we are continuing with the same system with multiple servers. So, then you may solve equations for P 1 and P 2. And now, in general, if you want to solve it for, so at the point when you have, you know, more than s people, or it just at the boundary, so this is been lambda P s minus 1 when you have s minus 1 people.

So, in coming, the way you can reach P s would be through 1 arrival. And then here, when you have s plus 1 people, then, you know, your service rate is s mu. So, it will be, you know, again 1 person departs. So, then you will be back to P s; and here it will be 1 arrival and 1 departure. So, then again you remain at P s, right. So, therefore, this is the balanced equation at this stage.

And, we can now substitute for, because, yeah; so I had shown you that the formula for P s, upto P s minus 1 would be this. So, I just substitute for this in terms of P naught. So, this would be lambda into lambda s minus 1 upon mu into 2 mu into s minus 1 mu, right;

then, plus s mu P s plus 1, and lambda plus s mu into lambda raise to s. So, upto P s the same solution will go; the same formula; so mu into 2 mu into s mu.

So, let us just, yes, we simplify; you take this to this side and then, you know, without spending time because surely you can do this calculation yourself. So that, from here you will subtract this expression, right; and you have s minus 1 mu here. So, you will get a s mu in the denominator from you multiply, and so when you multiply by s mu. So, lambda s, s mu term will cancel out, right; and you will be left with lambda s plus 1. So, this is the simplification; you can see it right away from here, right.

So, you will be left with only lambda s plus 1 upon, so this will be mu raise to s, and then 1 into 2 into 3 upto s, so that is s factorial. So, that gives you s mu into P s plus 1 equal to this. And therefore, P s plus 1 will be lambda by mu raise to s, and the lambda, so the s factorial, and then divided by s mu. So, 1 lambda upon, yeah, this is my lambda; you should be able to write it; yeah, this is lambda, lambda upon s.

So, that means, when it is s plus 1, you have this term into lambda upon s mu times 1 at when your rho is lambda, utilization factor is s mu, lambda by s mu now, right, s servers; and just want to show you that the same formula will continue, and then you can generalize for n, P n. So, now if you write lambda P s, then s mu P s plus 2, because after you have more than s people then if the same service state will continue. So, s mu into p s plus 2 is equal to, lambda plus s mu into P s plus 1.

And, this again when you substitute for p s and p s plus 1 from here, then you will get the expression that p s plus 2 is equal to lambda by mu raise to s into, lambda upon s mu raise to 2. So, this is the factor which goes on increasing; this remains the same; s factorial lambda by mu raise to s is common; and then it is lambda by s mu square. So, now, you can generalize for, you know, you can write the general formula, and that will be P n is lambda by mu raise to n, n factorial, P naught, no, that yeah, for n between 0 and s. So, this formula is ok for n between 0 and s that is what we were using here, right; and here, for P s.

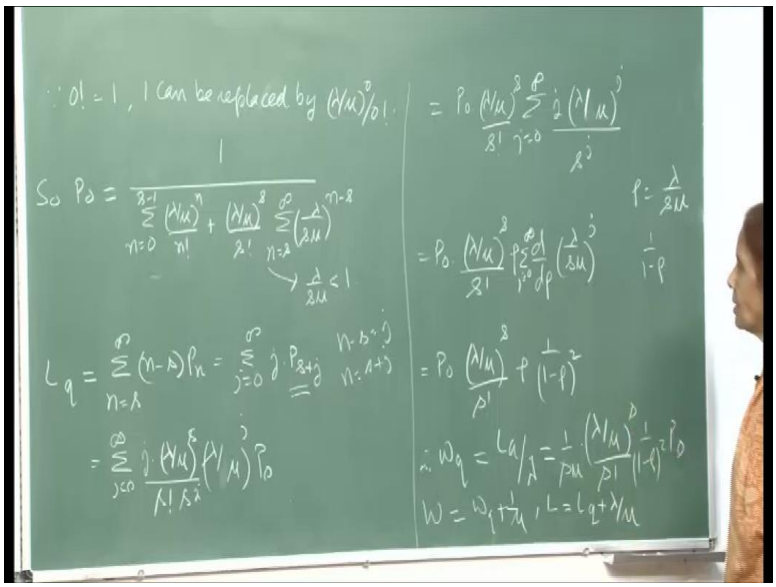
So, then, but n greater than s, you are going to have lambda mu raise to s upon s factorial. And then, see, we have, let me show me you. So, this is then this is lambda by s mu square. So, I am writing lambda by s mu, no; I do not have to write s here. So, this is lambda by mu raise to n minus 1 upon s raise to n minus s. So, either s you can include with this, and it will be lambda by mu raise to n minus s.

So, if you are writing the formula for P n lambda by mu raise to s upon s factorial, this is the common thing, and then it will be lambda by mu s raise to n minus s, which I choose to write as lambda by mu raise to n minus s, and divided by s raise to n minus s. So, it is same thing, into p naught if n is greater than or equal to s, right. And so now, since we have the general expression you can use the convention that 0 factorial is 1, and therefore 1 can be replaced by lambda by mu because when you add up, see now you want to use the fact that summation P i, i varying from 0 to infinity is 1.

So, when you write P 0, so therefore, n all are this are also in terms of P 0. So, the series will become 1 plus and so on, divided by the P 0. So, 1, we are replacing by 0 factorial. So, in that case, you see, this will be your this thing; and here, the later part, that means, from s onwards, s plus 1, s plus 2, to infinity, the terms set you will get, you will have this power series; and, this will be convergent if lambda upon s mu is less than 1.

So, as we have anyway c in that, you know, if this is greater than 1 then your things will blow up; if your arrival rate is more than s mu then the q size in everything will blow up. So, we have to anyway work this under the system; and this is less than 1.

(Refer Slide Time: 06:49)



So, for any feasible if you want to consider a balanced form and stationary system, so therefore; so this then can be written as s to infinity, lambda by s mu raise to n minus s, under the condition that lambda upon s mu is less than 1. So, this is the expression. And of course, we are not going to see this no other close form for this. And, when you are given the values lambda, mu, and s, you can just compute this value, right.

So, now start computing the expressions for L, L q, W, W q, and so on. So, let us just look at the expression for L q which will be n minus s. And, you can see the reason because there is a neat expression for this thing enough for the probabilities when n is s or more than s. So, I am therefore, writing L q. And since, I can get L from L q, therefore, it is enough that I compute this. And, once I get q, I will get L, and then I get W, I get W q. So, that is the thing.

So, let L q; so we will write as n minus s P n where n is varying from s to infinity. And here, if you want to write n minus s s j, then n is s plus j, right. So, therefore, you can then say that j varies from 0 because n is varying from s; so n plus s. The j will vary from 0 to infinity, and this will be n minus s j into, P s plus j. So, we can, we have a nice way of writing the probability for this.

So, which will be, see here, the s part is lambda by mu raise to s upon s factorial; and then this for the j this will be lambda by mu raise to j upon s raise to j, right. And, this j is there because n minus s. So, you are computing L q, right; expected number of people in the q, right; and this is P 0, fine. So, I can remove lambda by mu raise to s upon s factorial, take it outside the, and P naught outside the summation sign; then I am left with this j into lambda by mu raise to j upon s raise to j.

And, here this I can; see, lambda by s mu, one s I can take outside; then, I will be left with lambda by s mu raise to j minus 1, right; one lambda by s mu I take outside which I am writing as rho; then, it will be lambda by s mu raise to j minus 1. So, j times this which is the, you know, derivative of lambda upon s mu raise to j. So, this whole thing by taking rho outside is equivalent to the derivative of this, right.

And since, lambda upon s mu is less than 1, we know that this series is also convergence; this is the arithmetic co geometric form with common ratio is lambda upon s mu which is less than 1. So, therefore, this is the convergent series. So, I can take the, first was, I should have written the derivative outside; and then, since this is a convergent series I can bring the derivative sign inside, and this is what you have.

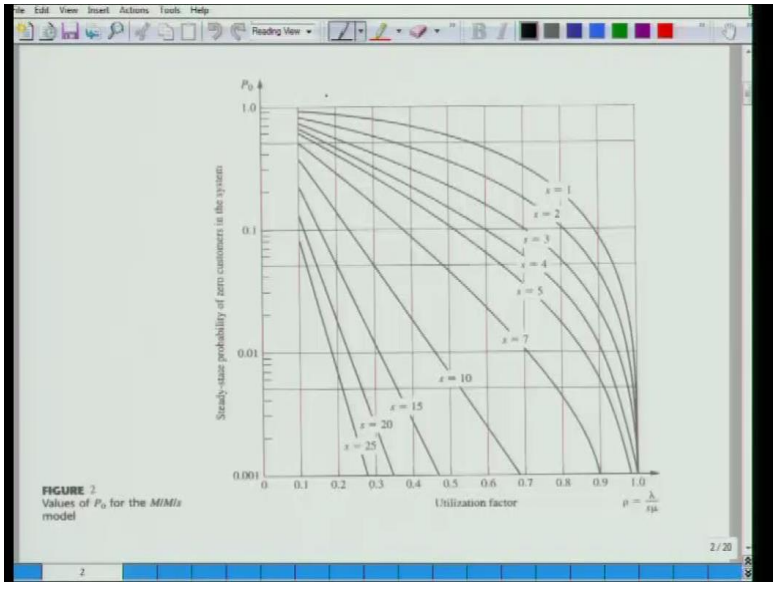
And therefore, now this is the geometric series which adds up to 1 upon 1 minus rho, right; this part summation, and then derivative of this, will be 1 upon 1 minus rho whole square. So, actually, no no; what I should have done is I, first I write derivative inside and then I take it outside because it is a convergent series, so you can interchange summation sign and derivative sign.

So, now I take the derivative outside; then this sums up to 1 upon 1 minus rho; and the derivative gives me 1 upon 1 minus rho whole square. So, therefore, now you have a neat expression, except the P naught is a little complex one. So, then you have this expression for L q. And then, I can get my W q which will be L q by lambda; and, because we said that these relationships are valid even in the general case.

So, 1 upon; so therefore, this is given 1 upon s mu; when you divide this by lambda, this will be left, though 1 upon s mu, and you have this expression. Hence, then after that you will say, W, W q plus; and remember, the difference between W and W q will still be that of 1 service because your departure from service is 1 at a time. So, therefore, this we will not 1 by s mu, it will be 1 by mu, right. And similarly, here this lambda will, L will be L q plus lambda by mu. So, keeping that in mind, you have all this relationships.

Now, let us just look at some of the; no these, yes, and I want it to show; the tables that I am going to show you have been taken from this book and I will give you the proper references all at the Ravindran and Philips books. So, I have taken some tables where they have plotted the L with respect to, you know, the rho sign, rho, where s is vary for different values of s. So, rho changes; so they have plotted.

(Refer Slide Time: 12:03)



So, let me show you the graph here. So, figure 2, I would just want to explain that here you see the horizontal axis is the utilization factor; that means, rho equal to lambda by s mu; so different values of rho going from 0 to 1. And, the vertical axis is the steady state probability of 0 customer in the system. So, look at the curves in the diagram, in the

graph, then you see that the utilization factor is coming down drastically as s increases, ok.

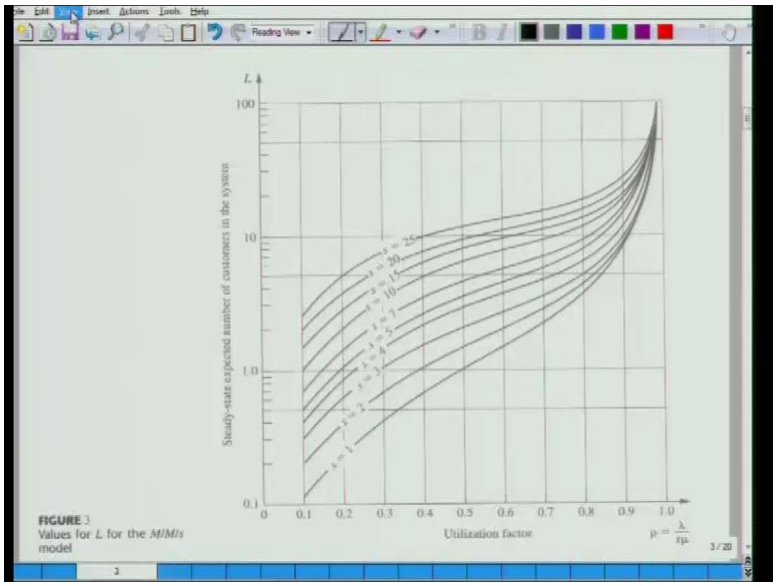
So, for; that means, for 0.8, for example, the top; we have top line shows you the utilization factor the, you know, utilization factor versus the probability steady state, probability of nobody no has customer in the system. So, these are the different lines. And, you see that the neutralization factor when s is 25 is barely, is even less than 0.3. So, that of course, we also except because more the server, the lesser the utilization factor. And, but there again as we said that it is always conflict between, you know, have it congestion or having more servers; and, that depends on what your priorities are.

So, anyway, this, the graph actually show you what we expect; that as this will go up the utilization factor will come down. But, the other important contribution of this graph is that, you know, you can plot because, remember, the expression for the steady state probability when you had more than 1 server, the expression was lengthy one.

So, now you can actually plot for different say; that means, if you know the utilization factor, say, for example I know the my utilization factor is 0.7, and then if I want to find out what the corresponding value of P naught will be, so then I will go along this vertical line of 0.7. And, if my number of servers are 4, then you see whereever this curve s equal to 4 cuts the vertical line 0.7, so that point, and I go horizontally then across to the vertical line, so I can then find out the corresponding value of the P naught.

And so that will help me because then I have to make my computations for L, L q, and so on, or even for P n s, I will need the value of P naught. So, then it will help me to just plot the value, given utilization factor and the number of servers; then I can find out the corresponding value of P naught. So, this is the contribution of figure 2, and later on when I work out, when I worked out any, I will use the values.

(Refer Slide Time: 14:53)



Now, I will show you another graph. This is the utilization factor, the verses. So, we are plotting the utilization factor on the horizontal axis, and steady state expected number of customers; so number of customers in this system, right. So, utilization factor and therefore, it says the n for different servers as you except. So, for example, if your utilization factor is 0.3 for s equal to 25 you can see that; well, let us see; this is at utilization factor 0.4; that means, your lambda bar, lambda upon s mu is 0.4.

And then, if you go up then you see that for s equal to 25 the number of people will be 10. So, now, total number of people; but you have this is the steady state expected number of customers which is L. So, the number L would be around 10, winning of 25 servers in the system.

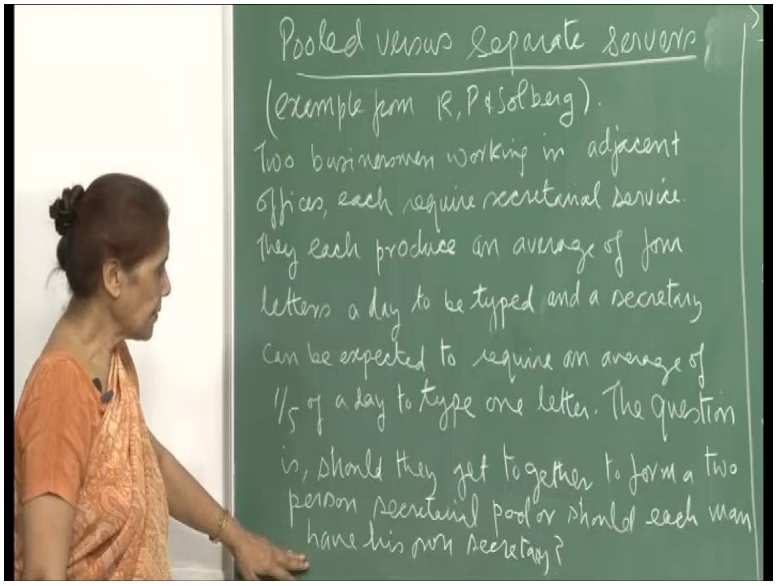
Now, if you come down, that means, if you come down to or if you go higher, then of course, these as the utilization factor increases, then we know that the number of people in the system will go to infinity, and the, because it is never advisable to have your s mu equal to lambda. So, therefore, this should never come very close to 1, the, your rho, right.

So, that of course, is depicted by as we saw it for ones server, the same phenomena is shown, is repeated here also. But, this again gives you an idea is to the utilization factor verses the number of servers you have and the number of customers you will have in the system. So, for different values you can just plot and see, right; for what is the number of; so if for example, for 0.5 if you go again the number of people in the system will

come, will be again round 11 or something; you can say that for 25. But, then if you have only s equal to 1 and 0.5, then you expect only 1 person to be in the system.

So, this is the kind of; so therefore, this, I mean the conclusions are the ones that you except, right; but they, sort of, give you more accurate you, and you can accurately find out for number of the utilization factor, number of servers, what will be the expected number. Because computing P naught, so you can from here only just find out the different values of rho; you can find out n for number of servers; you can find out the excepted number of people in the system. So, once you can do that, then if you have computed L, then you can find out L q, you can find out W q, and you can find out your W. So, this is just to give you a feeling about the multiple servers, and how these quantities L, L q, W, W q, behave.

(Refer Slide Time: 18:00)



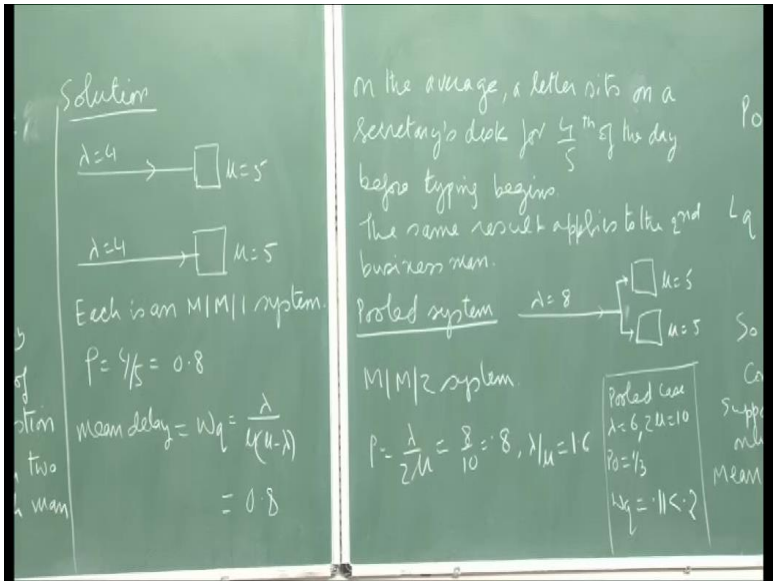
So, you see, now interesting example from Ravindran, Phillips and Sol berg, actually there are 3 authors; so Ravindran, Phillips, and Sol berg. So, now they are trying to show you that pool verses separate servers, and what would be the conclusion. So, let us look at this example. So, the 2 business working, business men working in adjacent offices; each requires secretarial service; and they each produce an average of 4 letters a day; well, this is simplification.

But anyway, whatever the work, what we are saying is, average of 4 letters a day to be typed, and a secretary can be excepted to require an average of 1/ 5 of a day to type 1 letter; that means, the rate of typing letters by secretaries 5 letters a day, and the business

men each is producing 4 letters a day to be typed. So, the question is should they get together to form a 2 person secretarial pool; the pool means that whenever anybody has a letter ready, when anyone of the 2 secretaries who are free, they will type the letter.

So, it is not that, you know, 1 secretary to the 1 business man, and so she will only work for particular, for her boss only, and only when the letter is ready by the boss she will type it. So, by pooling, it will be possible for each of the business men to access both the secretaries; that is the idea. Or should each men have his own secretary; so this is the question and let us try to answer through this model of M M S which we have just now talked about.

(Refer Slide Time: 19:39)



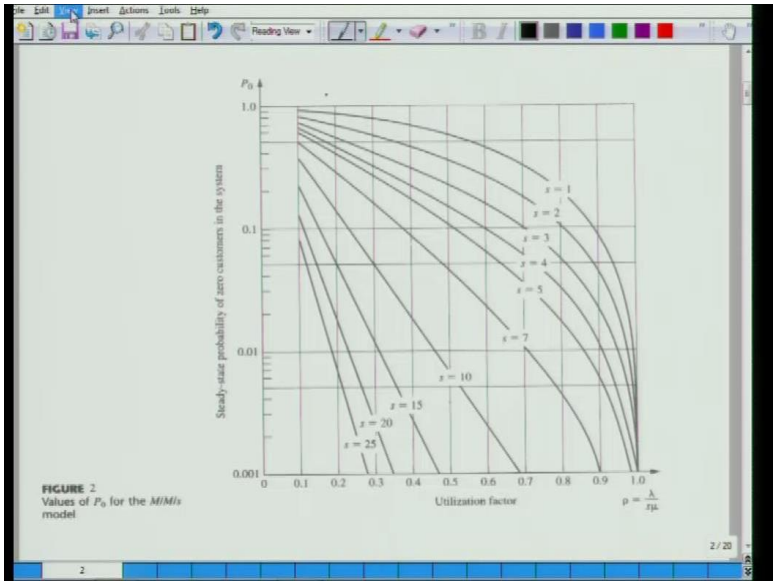
And you see, so let us see. So, if you take a single system then it will be just input is in 4 letters a day and their secretary is typing 5 letters a day. So, each system can be considered as a M M 1 system, right. So, and therefore, your rho will be 0.8; the utilization factor 4 by 5 is 0.8; and the mean delay which is W q will be lambda upon mu; mu delay means that the letter has to wait for sometime before it gets started to be typed by a secretary.

So, then W q is lambda upon mu into mu minus lambda which is 0.8; so that means, on the average a letter will sit for, you know, 4 th, 5 th of the day, on the secretaries desk before it is being typed, before it starts being typed, typing begins on that letter. So, the mean delay is 4th 5th of the day, right. Now, and the same applies to the second business man also because they are identical systems M M 1 systems with the same data.

So, therefore, the second business men also will have the same thing happening to his letters that for 4 5th of the day now the letter will be waiting on the average, and then just typing begins. Now, suppose, you pool the system then your input will be, you know, 8 letters a day, and you have 2 servers now, each them producing, typing 5 letters a day. So, then this is a M M 2 system, right.

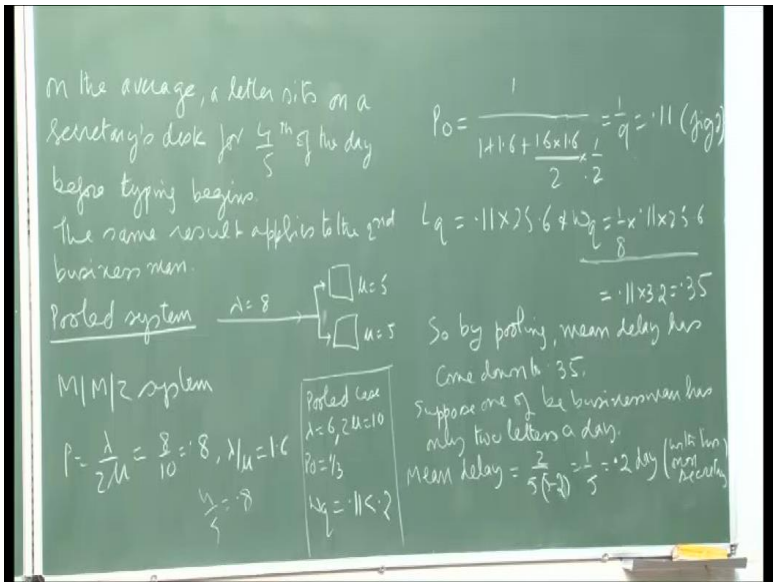
And, your rho will be again lambda upon 2 mu which is 8 by 10. So, therefore, this is 0.8. And your lambda by mu is 1.6, right. So, then P 0; now this is where you are, the graph that I had talked about come in handy; of course, this is the small system, and so therefore, I have shown you the calculation even otherwise; this is this, right. I have shown you the calculation, but see, you can see from, that means, for lambda by s 2 mu, s mu is 0.8; so corresponding to 0.8 and 2 servers.

(Refer Slide Time: 21:52)



If you look at the graph here, figure 2. So, 0.8, and you want to go up to 2. So, you see this is little above 0.1. So, 0.11, right. You can see in the graph, lambda by s mu equal to 0.8 and 2 servers. So, this is just above 0.1; so 0.11, right; and which also by our calculation comes out be 1 by 9. So, this is 0.11. So, larger systems your graph is plotted; you can just check the value, you know, look up the value for P naught without having to do the lengthy calculation.

(Refer Slide Time: 22:13)



And therefore, your L q would be again by the formula and then so W q; that means, so by formula we have W q s, this number which is 0.35. So, therefore, by pooling the mean delay has come down to 0.35; earlier the mean delay was we have computed it was 4th 5th of the day, right; which was, right. And so here now it is 0.35 which is much less than 4 by 5, right. Because, if you multiply this is, mean delay here, what is it, 4 5th of the day. So, if you want to compute it in; this is 0.8; mean delay is 0.8, so which is much less than point, which is much very very high compared to 0.35.

So, by pooling definitely your mean delay will come down. So, it might not be, you know, if you put the ego side apart; you know, like having your own secretary; and of course, probably this goes against intuition also. Because you may feel that if you have 1 person to exclusively to do your job, then you should get it done faster, but certainly the data here shows that this is not the case; by pooling it will always help you to get your work done faster.

Now, mean delay we had computed that 0.8. So, therefore, this is very high compared to this number. Now, let us see, we can again play around with few numbers. Suppose you say that this data was particularly till I had, so that their difference is so much- 0.8 and 0.35. Now, suppose 1 of the business men has only 2 letters a day then the mean delay for is the letter getting type will be 2 upon; so this is the number of let mu mu upon lambda, lambda minus mu, sorry; I have said the wrong way.

2 is the number of letters that arrive per day. So, lambda is 2, mu is 5, because 5 letters get typed; the secretary can type 5 letters a day; mu is 5. So, this is, lambda is 2; so lambda upon mu into mu minus lambda. So, this will be 2 by 15 which is 0.133. So, with his own secretary the mean delay would be of the order of 0.133.

Now, if we pool the 2 secretaries then your lambda becomes 6 because the first business men is, one of the business men is sending, getting 4 letters to be typed; and the other one has only 2. So, lambda equal to 6; and 2 mu will be 10, because each of them can type 5 letters a day.

So, therefore, P 0 is 1 by 3. Now, this we get from the figures that I have, should, given you. So, therefore, for lambda and mu, for these values of lambda and mu, you find out p naught which it comes out to be 1 by 3 from the figure. And therefore, your W q would be 0.11; that means, the mean delay would be 0.11 which is still less than 0.133. So, you see pooling definitely is a better option with 2 letters and the secretary and the man using his own secretary, even then the delay that he encounters is more than what he would encounter if he, if the 2 secretaries service is a pooled up and then they type letters as they come.

So, you see, even though as I said in the beginning in somewhere in the last lecture that, you know, you cannot take the values that we compute through this such models as exact, but they do definitely give you, you know, they have good guiding; they give you, provide you with good parameters to make, to help you make your decisions, right. See, even though the numbers may not be so exact like 0.35 and 0.8, but it definitely shows that the difference is there.

And so you can, the efficiency of the system gets improved by pooling. So, your services that you have; you know, like so many banks in so many other places, in public places if you see, that sometimes even at airport cannot counters and so on, you feel that, you know, separate queues because once you, once up the customer joins a queue then he or she cannot change the queue.

So, you see, that way you can immediately see that, I mean, this kind of model shows you that lot of time is being wasted; I mean the system is not working efficiently because you are not pooling the resources. So, somehow the feeling that separate queues will be more efficient and your work will get done faster, so that belief is not supported by this

model; and it is a reasonable correct model, in the sense that it gives you ideas to what happens when you pool up the resources.

So, this is a all about it; and we will continue with the. So, figure 2, I would just want to explain that here you see the horizontal axis is the utilization factor; that means, rho equal to lambda by s mu; so different values of rho going from 0 to 1; and the vertical axis is the steady state probability of 0 customers in the system.

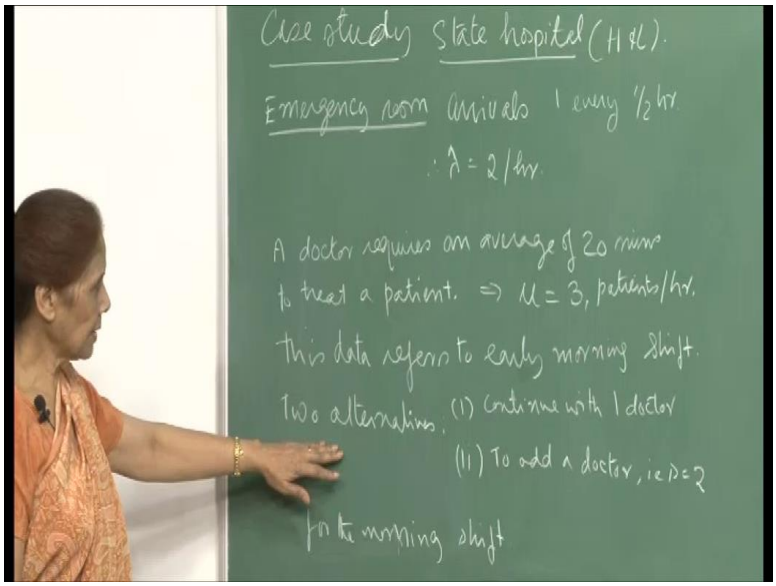
So, first of all if you look at the curves in the diagram, in the graph then you see that the utilization factor is coming down drastically as s increases. So, for, that means, for 0.8, for example, s equal to 1 you, the top, way of top line shows you the utilization factor the, you know, utilization factor verses the probability, steady state probability of nobody no has customer in the system.

So, these are the different lines. And, you see that the neutralization factor when s is 25 is barely is even less than 0.3. So, that of course, we also expect because more the server the lesser the utilization factor, and, but their again as we said that it is always conflict between you know have a congestion or having more servers, and that depends on what your priorities are. So, any way this, the graph actually show you what we expect; that as the number of servers will go up, the utilization factor will come down.

But, the other important contribution of this graph is that, you know, you can plot because, remember the expression for the steady state probability when you had more than 1 server, the expression was lengthy one. So, now, you can actually plot for different; that means, if you know the utilization factor. So, for example, I know the, my utilization factor is 0.7, and then if I want to find out what the corresponding of P naught will be, so that I will along this vertical line of 0.7.

And, if I number of servers are 4, then you see wherever this curve s equal to 4 cuts the vertical line 0.7, so that point; and I go horizontally then across to the vertical line, so I can then find out the, in a corresponding value of the P naught. And so that will help me because then I can make my computations for L, L q and so on, or even for P n s I will need the value of P naught. So, then it will help me to just plot the value, given my utilization factor and the number of servers then I can find out the corresponding value of P naught. So, this is the contribution of figure 2 and later on when I work out, when I worked out an example, I have used the, I will use the values of P naught from this graph.

(Refer Slide Time: 30:42)



So, I will take up this, taken up this case study from the state hospital in; and this is from the book Hillier and Leeber men; again this reference will also will be given at the end of the course. And, see, the state hospitals in the US are called county hospital. So, the data was collected from the county hospital, and the emergency room is considered because that can be modeled really well. So, here emergency room and the arrivals, and we are considering the morning shift; have I written something from where here, yes. So, this data refers to the early morning shift.

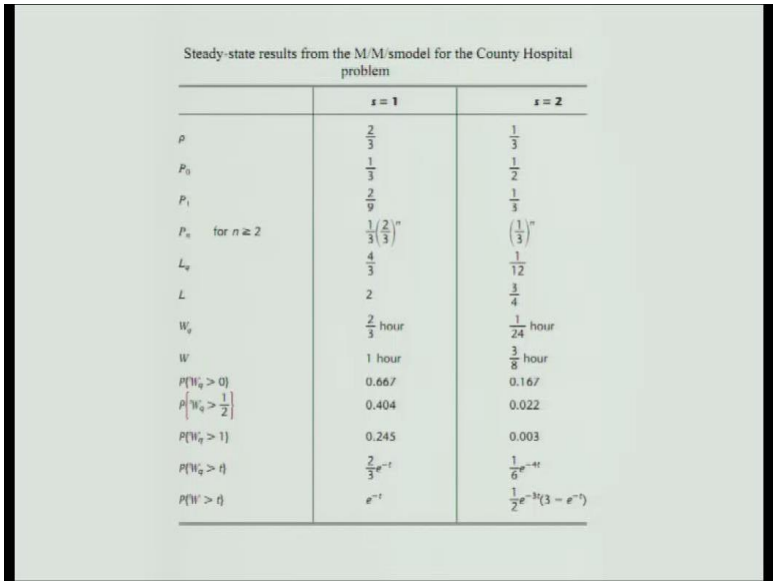
So, early morning shift and that, I do not know, for some reason this is what happens; that often the emergencies are in the early hours of the day, early hours of the morning. So, arrivals are 1 per half hour, right; so that means, your arrival rate is lambda equal to 2. And, here again, it is, it was found suitable to model the thing y Poisson arrival; and of course, the service process is exponential, negative exponential.

So, that means, since 1 every half hours, so 2 arrivals per hour; then doctor requires an average of 20 minutes to treat a patient which means that mu is 3 patients per hour. So, your this thing also; if it is negative exponential then the inter arrival times, the services, the service times would follow negative exponential distribution. Now, there are 2 alternatives which the hospital management has to consider.

They have 1 doctor to manage the emergency room, so either they continue with the, with 1 doctor, or to add another doctor; that means, your number of servers will go up to 2. So, these are the 2 alternatives; this for a morning shift; it is not for the whole day

because for the rest of the day, your lambda may change and even your mu may change. So, for the morning shift this is the, these are 2 alternatives which are being considered.

(Refer Slide Time: 32:55)



So, now let us look at the data. And so all the calculations have been made with s equal to 1 and s equal to 2. And so let us look at the data. So, this is steady state results from the M M S model. So, there should be a gap between S and model, for the M M S model, for the county hospital problem, right. Now, for s equal to 1, the rho, the traffic intensity is 2 by 3. But, when it is s equal to 2, it will become mu lambda by 2 mu. So, this will be 1 by 3. So, intensity will come down to 1 by 3.

P naught, your probability, when there is no patient in the system, in the emergency room, this is 1 by 3, and here it is 1 by 2. Then, P 1, the computed value of P 1 also, at 1 patient it will be 2 by 9 for s equal to 1, and 1 by 3 for s equal to 2. So, again the number of patients, the probability of P 1 will be 1 by 3. Then, P n for n greater than or equal to 2 is 1 by 3 into 2 by 3 raise to n; and here it will be for s equal to 2 it will be 1 by 3 raise to n.

So, everything, obviously, we except all these numbers to come down, but the drastic difference is seen where in L c, L q is 4 by 3. So, a patient has to wait, right; the q is 4 by 3. Whereas, for s equal to 2 it is 1 by 12; so that is really remarkable difference. Then your number of people in the system on the average would be 2; whereas, here it will be 3 by 4 for s equal to 2. Then, W q, the time, average time spent in the queue waiting for to be treated by doctor, here its 2 by 3 hour; and for s equal to 2 it becomes 1 by 24.

And so you see in an emergency room it is very, very crucial that a patient gets treatment as fast as possible because it is a matter of life and death. So, here W q being 2 by 3 if only one doctor is attending to the patients, then it is a, then the waiting time is high; whereas, it comes down drastically to 1 by 24 hours if your s is 2. Then, W, the number of people waiting; that means, q n service is 1 hour; whereas, here it is 3 by 8 hour. So, you see these are the figures which immediately tell you that it will definitely be, to the, it will be advantages to have 2 doctors because after all in an emergency room we do not want patient, patients to die because they have not got immediate attention.

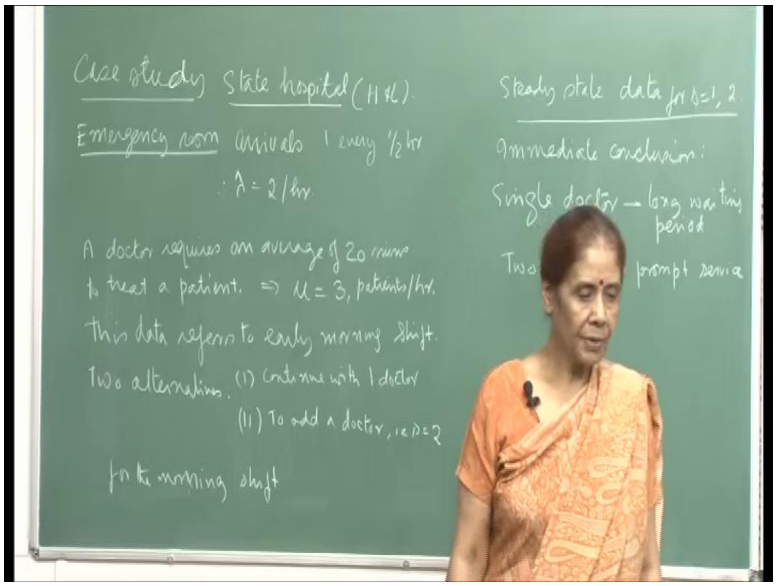
And then, probability W q greater than 0 is 0. 667 when s is 1, but it is 0.167 when s is 2 ok. So, that means, the patient coming into the emergency room will have to wait; that is high; 0.667 when s is 1. But, the moment you have 2 doctors attending it comes down to 0.167. And similarly, probability W q greater than half would be 0.404, for s equal to 1; whereas, it will be 0.022 for s equal to 2.

So, therefore, this is also you know damaging because they, in a hospital emergency room if you have to wait for more than 1 by 2 of an hour then this is bad, and the probability is 0.404 when s is 1. So, this is not acceptable. And, similarly, W q greater than 1 hour is 0.245 which will be less than half, right. So, that is the waiting time. So, that will be 0.245, but for s equal to 2 it will be 0. 003.

See, if you just look at the numbers the other 2 are not that important. But, anyway, so this data; so therefore, we are able to then conclude that single doctor will give you a long waiting period which is not very desirable for a emergency room, for hospital emergency room, but 2 doctors you expect from service.

And so therefore, anybody looking at this data; and this is what this model has helped us to generate the data and see that it will compare very, you can compare the performance of the emergency room, and there is 1 doctor remain, and when there are 2 doctors. So, if financier is not a consideration. It would be very helpful to have 2 doctors.

(Refer Slide Time: 37:31)

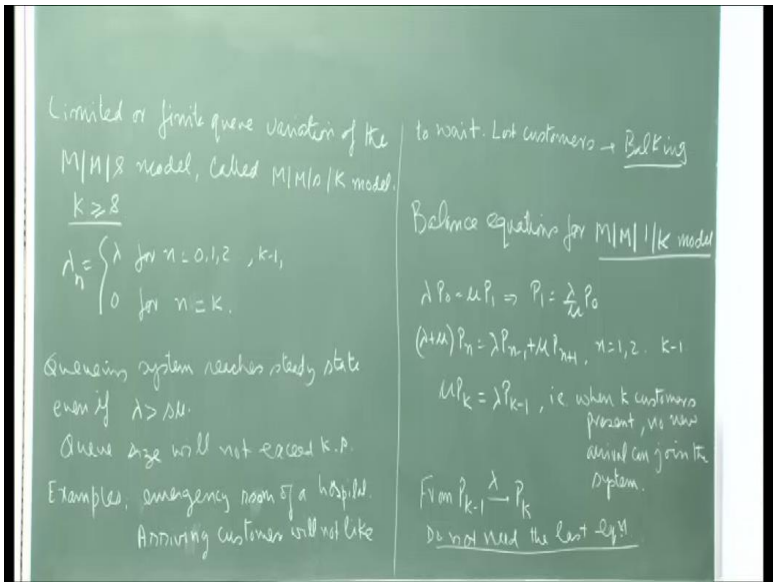


And then, again I just want to make a point about pooling that what we were talking the example I gave you in the earlier. So, pooling, we saw that if the waiting time is the main consideration then pooling will always have, because we saw that even with, you know, when 2 business men having their own secretaries they had to, the letters had to wait longer, but when the after when the services of the 2 secretaries were pooled the waiting time for the letters to be typed came down, right.

And, if there is some registration or something then it is a different thing; that you cannot, you have to have separate queues, but otherwise if the main consideration is to avoid long delays, then pooling is the answer. So, there, another, we know, again the model helped us to arrive at that conclusion.

Now, let us look at another kind of model which is limited of finite q variation of the M M S K model. So, here the ideas that you cannot allow queue, more than k people in the system. So, you could just take the example of an emergency room in hospital; you know, it may not be possible to because people come, they need beds, they are, it is an emergency, or they are on stretches. So, then you definitely need room for these patients arriving for emergency service.

(Refer Slide Time: 38:58)



So then, you, the space is limited; and therefore, you cannot allow for infinite q size, right. Then, you can also have many other; and as we said that, you know, a petrol station, if you may not, you could not have infinite number of cars waiting to be serviced. So, therefore, again you have limited space for the cars to be waiting and that the number usually; if you are big, even that it cannot be more than 5 to 6 cars which can be accommodated by the petrol station where they are waiting to be serviced, depending on the number of pumps the petrol station has, anyway.

So, this is very reasonable; and this model realistic situations where your limited space; that means, your finite q; you cannot allow for infinite queue. So, the only change that you would have to make in the M M S model would be that your lambda n will be lambda, for n varying from 0, 1, to k minus, 1; and for n equal to k, it will be 0. So, that means, you will not allow people to come; even once you have k people in the system then you will not allow people to enter the system essentially, right.

And here, the q will reach a steady state even if lambda is greater than s mu. See, remember, for infinite size we had to restrict lambda less than s mu, but, because otherwise the number would have blown up, right; your L, the average number of people in the system, and so on, would blow up if lambda was greater than s mu, in case you allowed infinite q size.

So, here, because you are not permitting your queue size to be more than k, so then lambda greater than s mu is also permissible, right. And so the q size will never exceed k

minus s; and the total number of people in the system will not exceed k. See, your s servers and your k size cannot be more than k minus s. Also, so as I came with example is emergency room of a hospital; also you see there are situation the places where the customers are choosy. And they would not like to wait; they would not like to enter the system as they are more than k people already; I mean, they might consider that k number to be a crowd, and therefore, there would go away.

Now, such phenomena is called bulking because you are losing outer customers, since you have limited space. So, you are losing customers; and you may also be losing out good will. So, we will look at this aspect. And, essentially, one would like to know what kind of business you are losing now, because your people are, your customers are being turned away because there you do not have enough room.

And, of course, for emergency rooms in a hospital registration requires, that if you cannot accommodate a patient right away, then you have to send them to another person, another hospital. So, there the registration requires that you turn away patient if you do not have enough room for them. So, all these considerations are there.

So, we will look at a bolking, and we will try to 3 examples try to see how you estimate the loss of revenue because you have lost customers. And then, of course, that might also encourage you to invest in increasing the waiting space, waiting room, so that to compensate for the loss in business.

So, now of course, I will give you the expressions for M M S K also, but right now it will be easier to just write down the balance equations from M M 1 K model; and then the arguments for M M S K will also not be much be different except that you will have to take care of the s server. So, for M M 1; that means, 1 server, but you have finite q. So, the balance equations will be; of course, when there is nobody there in the system then 1 arrival comes, and then you can go from P 1 to 1 departure. So, therefore, this is the balance equation, right.

And, so that gives you P 1 equal to lambda by mu P naught. And so here you see the transition diagram is no different from M M 1 except that there will be no state after K. And so therefore, you will have only that many balanced equations. So, that is only the difference; that is why I did not draw the transition diagram, anyway.

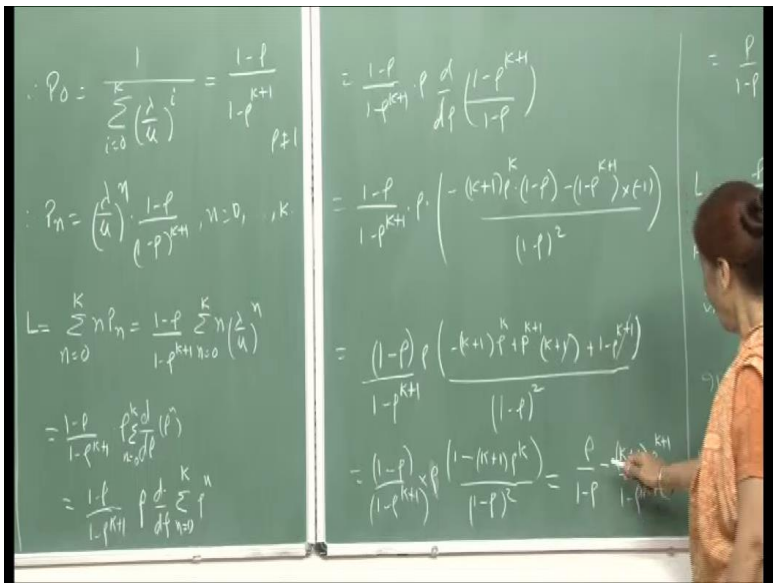
And so for, when there are n people in the system, then lambda plus mu, 1 arrival, 1 departure; again you are back with n people in the system; then this will be n minus 1

people; you can reach p n, you can reach n by 1 arrival. And, when you have n plus 1 people then you can reach again state n by 1 departure. And, this will be valid for n, 1, 2, to k, minus 1, right.

And then, the last one when you have k people, when only departure is allowed, because no arrival, right. And so from P k minus; so that means, from k minus 1, again you can reach k by 1 arrival. So, this will be the last equation, right. Surprisingly, you do not answer therefore, that is ok; this makes sense because you cannot go away from here; and you cannot have any, allow any arrival here. So, this will be the equation.

And, the interesting thing is that we will not leave this equation actually, when you are obtaining values of P 1, P 0, P 1, P 2, because from here, see, when you put n equal to k minus 1, P k value will be available U from here. So, we do not write we needed, but just for completeness sake we want to write it down. And so now, one can solve these balanced equations to get the corresponding relevant probability.

(Refer Slide Time: 45:10)



So, just as for M M 1 model, we will solve this balanced equation; and, you will get P n is equal to lambda by mu raise to n P naught, n varying from 1 to k. So, let me just show you the calculations for; as I told you that the last equation will not be needed; the last but 1 equation can, will give us the value of P k. So, the last, but one equation is lambda plus mu, P k minus 1, is equal to, lambda p k minus 2, plus mu P k, right; because when there k people in the system, no arrivals are allowed.

So, this is your last but 1 equation; and since you have obtained the formula for P k minus 1, and P k minus 2, so I just substitute. So, therefore, your mu P k is equal to lambda plus mu, times lambda by mu raise to k minus 1, minus lambda times lambda by mu raise to k minus 2, into P 2, P naught, right. Everything is in terms of P naught.

So, therefore, just simplify; lambda by mu raise to k minus 2, you can outside. So, then you will be left with this expression; and here you can simplify. So, mu lambda; and lambda again you can take out side. So, it will be lambda plus mu, minus mu by mu; lambda is outside here. So, this gives you lambda by mu; and therefore, this becomes lambda square by mu.

But then, you have your mu p k here, so therefore, P k will be lambda by mu raise to 2 into lambda by mu raise to k minus 2. So, the whole thing is there. So, therefore, you, there is no problem in solving your balanced equation. And then, since all the probability is must add up to 1, so you get the expression for P naught which is the geometric series here.

And, of course, except that rho should be not equal to 1; and otherwise you can add this and that gives you 1 minus; because when rho is equal to 1 you will have a simplification. So, that can be written down immediately. So, therefore, this is your value of P naught; and so you get first form for the P n which is lambda by mu raise to n into 1 minus rho, upon 1 minus rho raise to k plus 1. So, this is valid for n varying from 0 to k, right.

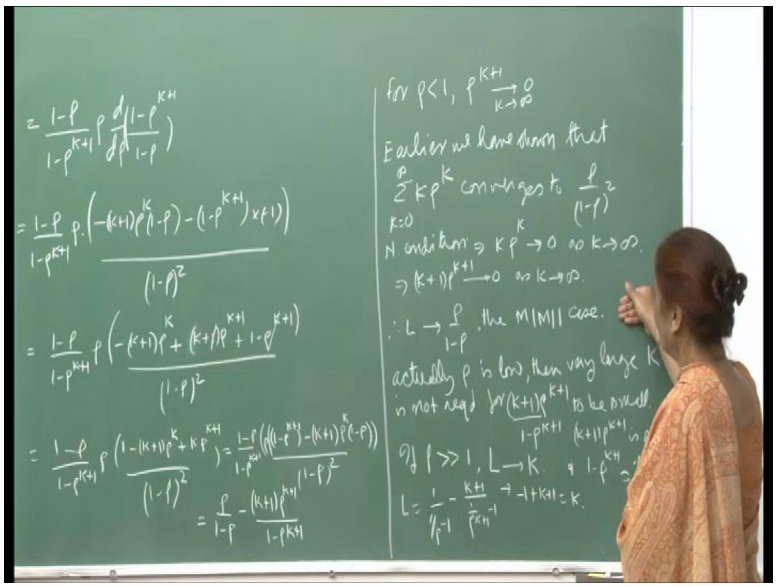
Now, you want to find out the average number of people in the system. So, this will be sigma n P n, n varying from 0 to k. And, so you just substitute for P n; that is what you get. And again, we will use the same trick that lambda, n lambda by mu raise to n. So, this can be written as derivative of lambda by mu raise to n, respect to rho; so rho raise to n. So, n rho raise to n; so that will be an summation of course, you are n varying from 0 to k.

So, if you take a rho outside, then it will be; so this will be then rho, derivative of rho is 2 n. And then again, so finite series I can interchange. So, d by d rho outside the summation, n varying from 0 to k rho n; and that gives you again geometric series; and the summation is this. So, derivative of this d by d rho on minus rho raise to k plus 1 upon 1 minus rho.

So, differentiate the numerator; this is minus k plus 1 rho raise to k, into 1 minus rho, minus, 1 minus rho raise to k plus 1, into derivative of this which is minus 1, right; divided by 1 minus rho whole square. And just simplify; and finally, you will get this as. So, I have just separated out the 2 terms, rho upon 1 minus rho minus, k plus 1 rho to k plus 1 divided by, 1 minus rho raise to k plus 1, ok.

So, this 1 minus rho cancels here. So, you are left with 1 minus rho, the power 2 is gone, and then the rho part here. So, rho 1 minus rho, I have written out here; and there is a 1 here. So, then this gets coupled with this. So, minus k plus 1 rho raise to k plus 1; rho is outside here; divided by 1 minus rho k plus 1. So, this usually you can see, simplifies to this expression, fine.

(Refer Slide Time: 49:34)



Now, we just want to look at the long, the behavior, if you allow k to become large, you want to look at this. So, for rho less than 1, rho k plus 1 will go to 0 as k goes to infinity. And earlier, we have shown that this series is, this converges k sigma raise to, sorry; rho raise to k, summation; the series converges to rho upon 1 minus rho whole square. We have already seen this because this arithmetic co geometric series, right. And so if a series converges then the necessary condition is that the n th term must go to 0 as s k goes to infinity, right.

So, therefore, k rho raise to k, must go to 0 which implies that k plus 1 into rho k plus 1 goes to 0, as k goes to infinity. So, now, if you look at this here, this is going to 0, so this reduces to 1; and k plus 1, rho k plus 1 goes to 0. So, therefore, your limiting value of L

when rho is less than 1 and s k goes to infinity. So, the limiting value of L is rho upon 1 minus rho which is the M M 1 case.

So, you see, you can immediately conclude that, you know, when you have the M M K s; that means, you have a limited space for people to wait; that is a k people can wait; units can wait; then we, so this model relates to that. But, if you make this space unlimited; that means, there is no restriction on how many people can wait in the system then the system reduce, that then the whole process reduce this to the M M 1 case.

So, this validates the M M K K s; that means, whatever we have derived, the values of L and so on, they are valid in the sense that they correspond to the M M 1 case, in case your space becomes unlimited, so as many people as you want can wait for to be serviced. So, in that case, it will be M M 1 case, right. So, you can, you know, so there are many ways in which you can also try to validate the model that you have constructed. So, this is one of them.