6.041SC Probabilistic Systems Analysis and Applied Probability, Fall 2013 Transcript – Lecture 17

PROFESSOR: All right. So today, we're going to start by taking stock of what we discussed last time, review the definition of Markov chains. And then most of the lecture, we're going to concentrate on their steady-state behavior. Meaning, we're going to look at what does a Markov chain do if it has run for a long time. What can we say about the probabilities of the different states?

So what I would like to repeat is a statement I made last time that Markov chains is a very, very useful class of models. Pretty much anything in the real world can be approximately modeled by a Markov chain provided that you set your states in the proper way. So we're going to see some examples. You're going to see more examples in the problems you're going to do in homework and recitation.

On the other hand, we're not going to go too deep into examples. Rather, we're going to develop the general methodology. OK.

All right. Markov models can be pretty general. They can run in continuous or discrete time. They can have continuous or discrete state spaces. In this class, we're going to stick just to the case where the state space is discrete and time is discrete because this is the simplest case. And also, it's the one where you build your intuition before going to more general cases perhaps in other classes.

So the state is discrete and finite. There's a finite number of states. At any point in time, the process is sitting on one of those states. Time is discrete, so at each unit of time, somebody whistles and then the state jumps. And when it jumps, it can either land in the same place, or it can land somewhere else. And the evolution of the process is described by transition probabilities.

Pij is the probability that the next state is j given that the current state is i. And the most important property that the Markov chain has, the definition of a Markov chain or Markov process, is that this probability, Pij, is the same every time that you land at state i -- no matter how you got there and also no matter what time it is.

So the model we have is time homogeneous, which basically means that those transition probabilities are the same at every time. So the model is time invariant in that sense. So we're interested in what the chain or the process is going to do in the longer run. So we're interested,

let's say, in the probability that starting at a certain state, n times steps later, we find ourselves at some particular state j.

Fortunately, we can calculate those probabilities recursively. Of course, at the first time 1, the probability of being 1 time later at state j given that we are right now at state i, by definition, this is just the transition probabilities. So by knowing these, we can start a recursion that tells us the transition probabilities for more than n steps.

This recursion, it's a formula. It's always true. You can copy it or memorize it. But there is a big idea behind that formula that you should keep in mind. And basically, the divide and conquer idea. It's an application of the total probability law. So let's fix i. The probability that you find yourself at state j, you break it up into the probabilities of the different ways that you can get to state j.

What are those different ways? The different ways are the different states k at which you might find yourself the previous time. So with some probability, with this probability, you find yourself at state k the previous time. And then with probability Pkj, you make a transition to state j. So this is a possible scenario that takes you to state j after n transitions. And by summing over all the k's, then we have considered all the possible scenarios.

Now, before we move to the more serious stuff, let's do a little bit of warm up to get a handle on how we use transition probabilities to calculate more general probabilities, then talk about some structural properties of Markov chains, and then eventually get to the main business of today, which is a steady-state behavior.

So somebody gives you this chain, and our convention is that those arcs that are not shown here corresponds to 0 probabilities. And each one of the arcs that's shown has a non-zero probability, and somebody gives it to us. Suppose that the chain starts at state 1. We want to calculate the probability that it follows this particular path. That is, it goes to 2, then to 6, then to 7. How do we calculate the probability of a particular trajectory?

Well, this is the probability-- so it's the probability of the trajectory from 1 that you go to 2, then to 6, then to 7. So the probability of this trajectory is we use the multiplication rule. The probability of several things happening is the probability that the first thing happens, which is a transition from 1 to 2. And then given that we are at state 2, we multiply with a conditional probability that the next event happens. That is, that X2 is equal to 6 given that right now, we are at state 1. And that conditional probability is just P26.

And notice that this conditional probability applies no matter how we got to state 2. This is the Markov assumption. So we don't care about the fact that we came in in a particular way. Given that we came in here, this probability P26, that the next transition takes us to 6. And then given that all that stuff happened, so given that right now, we are at state 6, we need to multiply with a conditional probability that the next transition takes us to state 7. And this is just the P67.

So to find the probability of following a specific trajectory, you just multiply the transition probabilities along the particular trajectory. Now, if you want to calculate something else, such

as for example, the probability that 4 time steps later, I find myself at state 7 given that they started, let's say, at this state. How do you calculate this probability?

One way is to use the recursion for the Rijs that we know that it is always valid. But for short and simple examples, and with a small time horizon, perhaps you can do this in a brute force way. What would be the brute force way? This is the event that 4 time steps later, I find myself at state 7. This event can happen in various ways. So we can take stock of all the different ways, and write down their probabilities.

So starting from 2. One possibility is to follow this trajectory, 1 transition, 2 transitions, 3 transitions, 4 transitions. And that takes me to state 7. What's the probability of this trajectory? It's P26 times P67 times P76 and then times P67. So this is a probability of a particular trajectory that takes you to state 7 after 4 time steps.

But there's other trajectories as well. What could be it? I might start from state 2, go to state 6, stay at state 6, stay at state 6 once more. And then from state 6, go to state 7. And so there must be one more. What's the other one? I guess I could go 1, 2, 6, 7.

OK. That's the other trajectory. Plus P21 times P12 times P26 and times P67. So the transition probability, the overall probability of finding ourselves at state 7, is broken down as the sum of the probabilities of all the different ways that I can get to state 7 in exactly 4 steps.

So we could always do that without knowing much about Markov chains or the general formula for the Rij's that we had. What's the trouble with this procedure? The trouble with this procedure is that the number of possible trajectories becomes quite large if this index is a little bigger. If this 4 was 100, and you ask how many different trajectories of length 100 are there to take me from here to there, that number of trajectories would be huge. It grows exponentially with the time horizon. And this kind of calculation would be impossible.

The basic equation, the recursion that have for the Rij's is basically a clever way of organizing this computation so that the amount of computation that you do is not exponential in the time horizon. Rather, it's sort of linear with the time horizon. For each time step you need in the time horizon, you just keep repeating the same iteration over and over.

OK. Now, the other thing that we discussed last time, briefly, was a classification of the different states of the Markov chain in two different types. A Markov chain, in general, has states that are recurrent, which means that from a recurrent state, I can go somewhere else. But from that somewhere else, there's always some way of coming back. So if you have a chain of this form, no matter where you go, no matter where you start, you can always come back where you started. States of this kind are called recurrent.

On the other hand, if you have a few states all this kind, a transition of this type, then these states are transient in the sense that from those states, it's possible to go somewhere else from which place there's no way to come back where you started. The general structure of a Markov chain is basically a collection of transient states. You're certain that you are going to leave the transient states eventually.

And after you leave the transient states, you enter into a class of states in which you are trapped. You are trapped if you get inside here. You are trapped if you get inside there. This is a recurrent class of states.

From any state, you can get to any other state within this class. That's another recurrent class. From any state inside here, you can get anywhere else inside that class. But these 2 classes, you do not communicate. If you start here, there's no way to get there.

If you have 2 recurrent classes, then it's clear that the initial conditions of your Markov chain matter in the long run. If you start here, you will be stuck inside here for the long run and similarly about here. So the initial conditions do make a difference. On the other hand, if this class was not here and you only had that class, what would happen to the chain?

Let's say you start here. You move around. At some point, you make that transition. You get stuck in here. And inside here, you keep circulating, because of the randomness, you keep visiting all states over and over. And hopefully or possibly, in the long run, it doesn't matter exactly what time it is or where you started, but the probability of being at that particular state is the same no matter what the initial condition was.

So with a single recurrent class, we hope that the initial conditions do not matter. With 2 or more recurrent classes, initial conditions will definitely matter. So how many recurrent classes we have is something that has to do with the long-term behavior of the chain and the extent to which initial conditions matter.

Another way that initial conditions may matter is if a chain has a periodic structure. There are many ways of defining periodicity. The one that I find sort of the most intuitive and with the least amount of mathematical symbols is the following. The state space of a chain is said to be periodic if you can lump the states into a number of clusters called d clusters or groups. And the transition diagram has the property that from a cluster, you always make a transition into the next cluster.

So here d is equal to 2. We have two subsets of the state space. Whenever we're here, next time we'll be there. Whenever we're here, next time we will be there. So this chain has a periodic structure. There may be still some randomness. When I jump from here to here, the state to which I jump may be random, but I'm sure that I'm going to be inside here. And then next time, I will be sure that I'm inside here.

This would be a structure of a diagram in which we have a period of 3. If you start in this lump, you know that the next time, you would be in a state inside here. Next time, you'll be in a state inside here, and so on. So these chains certainly have a periodic structure. And that periodicity gets maintained.

If I start, let's say, at this lump, at even times, I'm sure I'm here. At odd times, I'm sure I am here. So the exact time does matter in determining the probabilities of the different states. And in particular, the probability of being at the particular state cannot convert to a state value.

The probability of being at the state inside here is going to be 0 for all times. In general, it's going to be some positive number for even times. So it goes 0 positive, zero, positive, 0 positive. Doesn't settle to anything. So when we have periodicity, we do not expect the states probabilities to converge to something, but rather, we expect them to oscillate.

Now, how can we tell whether a Markov chain is periodic or not? There are systematic ways of doing it, but usually with the types of examples we see in this class, we just eyeball the chain, and we tell whether it's periodic or not. So is this chain down here, is it the periodic one or not? How many people think it's periodic?

No one. One. How many people think it's not periodic? OK. Not periodic? Let's see. Let me do some drawing here. OK. Is it periodic? It is. From a red state, you can only get to a white state. And from a white state, you can only get to a red state.

So this chain, even though it's not apparent from the picture, actually has this structure. We can group the states into red states and white states. And from reds, we always go to a white, and from a white, we always go to a red. So this tells you that sometimes eyeballing is not as easy. If you have lots and lots of states, you might have some trouble doing this exercise.

On the other hand, something very useful to know. Sometimes it's extremely easy to tell that the chain is not periodic. What's that case? Suppose that your chain has a self-transition somewhere. Then automatically, you know that your chain is not periodic.

So remember, the definition of periodicity requires that if you are in a certain group of states, next time, you will be in a different group. But if you have self-transitions, that property is not true. If you have a possible self-transition, it's possible that you stay inside your own group for the next time step.

So whenever you have a self-transition, this implies that the chain is not periodic. And usually that's the simplest and easy way that we can tell most of the time that the chain is not periodic.

So now, we come to the big topic of today, the central topic, which is the question about what does the chain do in the long run. The question we are asking and which we motivated last time by looking at an example. It's something that did happen in our example of last time. So we're asking whether this happens for every Markov chain.

We're asking the question whether the probability of being at state j at some time n settles to a steady-state value. Let's call it pi sub j. That these were asking whether this quantity has a limit as n goes to infinity, so that we can talk about the steady-state probability of state j.

And furthermore, we asked whether the steady-state probability of that state does not depend on the initial state. In other words, after the chain runs for a long, long time, it doesn't matter exactly what time it is, and it doesn't matter where the chain started from. You can tell me the probability that the state is a particular j is approximately the steady-state probability pi sub j. It doesn't matter exactly what time it is as long as you tell me that a lot of time has elapsed so that n is a big number.

So this is the question. We have seen examples, and we understand that this is not going to be the case always. For example, as I just discussed, if we have 2 recurrent classes, where we start does matter. The probability pi(j) of being in that state j is going to be 0 if we start here, but it would be something positive if we were to start in that lump. So the initial state does matter if we have multiple recurrent classes.

But if we have only a single class of recurrent states from each one of which you can get to any other one, then we don't have that problem. Then we expect initial conditions to be forgotten. So that's one condition that we need. And then the other condition that we need is that the chain is not periodic. If the chain is periodic, then these Rij's do not converge. They keep oscillating.

If we do not have periodicity, then there is hope that we will get the convergence that we need. It turns out this is the big theory of Markov chains-- the steady-state convergence theorem. It turns out that yes, the rijs do converge to a steady-state limit, which we call a steady-state probability as long as these two conditions are satisfied.

We're not going to prove this theorem. If you're really interested, the end of chapter exercises basically walk you through a proof of this result, but it's probably a little too much for doing it in this class.

What is the intuitive idea behind this theorem? Let's see. Let's think intuitively as to why the initial state doesn't matter. Think of two copies of the chain that starts at different initial states, and the state moves randomly. As the state moves randomly starting from the two initial states a random trajectory. as long as you have a single recurrent class at some point, and you don't have periodicity at some point, those states, those two trajectories, are going to collide. Just because there's enough randomness there. Even though we started from different places, the state is going to be the same.

After the state becomes the same, then the future of these trajectories, probabilistically, is the same because they both started at the same state. So this means that the initial conditions stopped having any influence. That's sort of the high-level idea of why the initial state gets forgotten. Even if you started at different initial states, at some time, you may find yourself to be in the same state as the other trajectory. And once that happens, your initial conditions cannot have any effect into the future.

All right. So let's see how we might calculate those steady-state probabilities. The way we calculate the steady-state probabilities is by taking this recursion, which is always true for the end-step transition probabilities, and take the limit of both sides. The limit of this side is the steady-state probability of state j, which is pi sub j. The limit of this side, we put the limit inside the summation. Now, as n goes to infinity, n - also goes to infinity. So this Rik is going to be the steady-state probability of state k starting from state i.

Now where we started doesn't matter. So this is just the steady-state probability of state k. So this term converges to that one, and this gives us an equation that's satisfied by the steady-state probabilities. Actually, it's not one equation. We get one equation for each one of the j's. So if we

have 10 possible states, we're going to get the system of 10 linear equations. In the unknowns, pi(1) up to pi(10).

OK. 10 unknowns, 10 equations. You might think that we are in business. But actually, this system of equations is singular. 0 is a possible solution of this system. If you plug pi equal to zero everywhere, the equations are satisfied. It does not have a unique solution, so maybe we need one more condition to get the uniquely solvable system of linear equations.

It turns out that this system of equations has a unique solution. If you impose an additional condition, which is pretty natural, the pi(j)'s are the probabilities of the different states, so they should add to 1. So you want this one equation to the mix. And once you do that, then this system of equations is going to have a unique solution. And so we can find the steady-state probabilities of the Markov chain by just solving these linear equations, which is numerically straightforward.

Now, these equations are quite important. I mean, they're the central point in the Markov chain. They have a name. They're called the balance equations. And it's worth interpreting them in a somewhat different way. So intuitively, one can sometimes think of probabilities as frequencies. For example, if I toss an unbiased coin, probability 1/2 of heads, you could also say that if I keep flipping that coin, in the long run, 1/2 of the time, I'm going to see heads.

Similarly, let's try an interpretation of this pi(j), the steady-state probability, the long-term probability of finding myself at state j. Let's try to interpret it as the frequency with which I find myself at state j if I run a very, very long trajectory over that Markov chain. So the trajectory moves around, visits states. It visits the different states with different frequencies. And let's think of the probability that you are at a certain state as being sort of the same as the frequency of visiting that state.

This turns out to be a correct statement. If you were more rigorous, you would have to prove it. But it's an interpretation which is valid and which gives us a lot of intuition about what these equation is saying. So let's think as follows. Let's focus on a particular state j, and think of transitions into the state j versus transitions out of the state j, or transitions into j versus transitions starting from j. So transition starting from that includes a self-transition.

Ok. So how often do we get a transition, if we interpret the pi(j)'s as frequencies, how often do we get a transition into j? Here's how we think about it. A fraction pi(1) of the time, we're going to be at state 1. Whenever we are at state 1, there's going to be a probability, P1j, that we make a transition of this kind. So out of the times that we're at state 1, there's a frequency, P1j with which the next transition is into j.

So out of the overall number of transitions that happen at the trajectory, what fraction of those transitions is exactly of that kind? That fraction of transitions is the fraction of time that you find yourself at 1 times the fraction with which out of one you happen to visit next state j. So we interpreted this number as the frequency of transitions of this kind. At any given time, our chain can do transitions of different kinds, transitions of the general form from some k, I go to some l.

So we try to do some accounting. How often does a transition of each particular kind happen? And this is the frequency with which transitions of that particular kind happens. Now, what's the total frequency of transitions into state j? Transitions into state j can happen by having a transition from 1 to j, from 2 to j, or from state m to j. So to find the total frequency with which we would observe transitions into j is going to be this particular sum.

Now, you are at state j if and only if the last transition was into state j. So the frequency with which you are at j is the frequency with which transitions into j happen. So this equation expresses exactly that statement. The probability of being at state j is the sum of the probabilities that the last transition was into state j. Or in terms of frequencies, the frequency with which you find yourself at state j is the sum of the frequencies of all the possible transition types that take you inside state j.

So that's a useful intuition to have, and we're going to see an example a little later that it gives us short cuts into analyzing Markov chains. But before we move, let's revisit the example from last time. And let us write down the balance equations for this example. So the steady-state probability that I find myself at state 1 is the probability that the previous time I was at state 1 and I made a self-transition-- So the probability that I was here last time and I made a transition of this kind, plus the probability that the last time I was here and I made a transition of that kind. So plus pi(2) times 0.2.

And similarly, for the other states, the steady-state probably that I find myself at state 2 is the probability that last time I was at state 1 and I made a transition into state 2, plus the probability that the last time I was at state 2 and I made the transition into state 1. Now, these are two equations and two unknowns, pi(1) and pi(2). But you notice that both of these equations tell you the same thing. They tell you that 0.5pi(1) equals 0.2pi(2).

Either of these equations tell you exactly this if you move terms around. So these two equations are not really two equations. It's just one equation. They are linearly dependent equations, and in order to solve the problem, we need the additional condition that pi(1) + pi(2) is equal to 1. Now, we have our system of two equations, which you can solve. And once you solve it, you find that pi(1) is 2/7 and pi(2) is 5/7. So these are the steady state probabilities of the two different states.

If we start this chain, at some state, let's say state 1, and we let it run for a long, long time, the chain settles into steady state. What does that mean? It does not mean that the state itself enters steady state. The state will keep jumping around forever and ever. It will keep visiting both states once in a while. So the jumping never ceases. The thing that gets into steady state is the probability of finding yourself at state 1.

So the probability that you find yourself at state 1 at time one trillion is approximately 2/7. The probability you find yourself at state 1 at time two trillions is again, approximately 2/7. So the probability of being in that state settles into a steady value. That's what the steady-state convergence means. It's convergence of probabilities, not convergence of the process itself.

And again, the two main things that are happening in this example, and more generally, when we have a single class and no periodicity, is that the initial state does not matter. There's enough

randomness here so that no matter where you start, the randomness kind of washes out any memory of where you started. And also in this example, clearly, we do not have periodicity because we have self arcs. And this, in particular, implies that the exact time does not matter.

So now, we're going to spend the rest of our time by looking into a special class of chains that's a little easier to deal with, but still, it's an important class. So what's the moral from here? This was a simple example with two states, and we could find the steady-state probabilities by solving a simple system of two-by-two equations. If you have a chain with 100 states, it's no problem for a computer to solve a system of 100-by-100 equations. But you can certainly not do it by hand, and usually, you cannot get any closed-form formulas, so you do not necessarily get a lot of insight.

So one looks for special structures or models that maybe give you a little more insight or maybe lead you to closed-form formulas. And an interesting subclass of Markov chains in which all of these nice things do happen, is the class of birth/death processes. So what's a birth/death process? It's a Markov chain who's diagram looks basically like this. So the states of the Markov chain start from 0 and go up to some finite integer m.

What's special about this chain is that if you are at a certain state, next time you can either go up by 1, you can go down by 1, or you can stay in place. So it's like keeping track of some population at any given time. One person gets born, or one person dies, or nothing happens. Again, we're not accounting for twins here. So we're given this structure, and we are given the transition probabilities, the probabilities associated with transitions of the different types. So we use P's for the upward transitions, Q's for the downward transitions.

An example of a chain of this kind was the supermarket counter model that we discussed last time. That is, a customer arrives, so this increments the state by 1. Or a customer finishes service, in which case, the state gets decremented by 1, or nothing happens in which you stay in place, and so on. In the supermarket model, these P's inside here were all taken to be equal because we assume that the arrival rate was sort of constant at each time slot. But you can generalize a little bit by assuming that these transition probabilities P1 here, P2 there, and so on may be different from state to state.

So in general, from state i, there's going to be a transition probability Pi that the next transition is upwards. And there's going to be a probability Qi that the next transition is downwards. And so from that state, the probability that the next transition is downwards is going to be Q_(i+1). So this is the structure of our chain. As I said, it's a crude model of what happens at the supermarket counter but it's also a good model for lots of types of service systems.

Again, you have a server somewhere that has a buffer. Jobs come into the buffer. So the buffer builds up. The server processes jobs, so the buffer keeps going down. And the state of the chain would be the number of jobs that you have inside your buffer.

Or you could be thinking about active phone calls out of a certain city. Each time that the phone call is placed, the number of active phone calls goes up by 1. Each time that the phone call stops happening, is terminated, then the count goes down by 1.

So it's for processes of this kind that a model with this structure is going to show up. And they do show up in many, many models. Or you can think about the number of people in a certain population that have a disease. So 1 more person gets the flu, the count goes up. 1 more person gets healed, the count goes down. And these probabilities in such an epidemic model would certainly depend on the current state.

If lots of people already have the flu, the probability that another person catches it would be pretty high. Whereas, if no one has the flu, then the probability that you get a transition where someone catches the flu, that probability would be pretty small. So the transition rates, the incidence of new people who have the disease definitely depends on how many people already have the disease. And that motivates cases where those P's, the upward transition probabilities, depend on the state of the chain.

So how do we study this chain? You can sit down and write the system of n linear equations in the pi's. And this way, find the steady-state probabilities of this chain. But this is a little harder. It's more work than one actually needs to do. There's a very clever shortcut that applies to birth/death processes. And it's based on the frequency interpretation that we discussed a little while ago.

Let's put a line somewhere in the middle of this chain, and focus on the relation between this part and that part in more detail. So think of the chain continuing in this direction, that direction. But let's just focus on 2 adjacent states, and look at this particular cut. What is the chain going to do?

Let's say it starts here. It's going to move around. At some point, it makes a transition to the other side. And that's a transition from i to i+1. It stays on the other side for some time. It gets here, and eventually, it's going to make a transition to this side. Then it keeps moving and so on.

Now, there's a certain balance that must be obeyed here. The number of upward transitions through this line cannot be very different from the number of downward transitions. Because we cross this way, then next time, we'll cross that way. Then next time, we'll cross this way. We'll cross that way. So the frequency with which transitions of this kind occur has to be the same as the long-term frequency that transitions of that kind occur.

You cannot go up 100 times and go down only 50 times. If you have gone up 100 times, it means that you have gone down 99, or 100, or 101, but nothing much more different than that. So the frequency with which transitions of this kind get observed. That is, out of a large number of transitions, what fraction of transitions are of these kind? That fraction has to be the same as the fraction of transitions that happened to be of that kind.

What are these fractions? We discussed that before. The fraction of times at which transitions of this kind are observed is the fraction of time that we happen to be at that state. And out of the times that we are in that state, the fraction of transitions that happen to be upward transitions. So this is the frequency with which transitions of this kind are observed.

And with the same argument, this is the frequency with which transitions of that kind are observed. Since these two frequencies are the same, these two numbers must be the same, and

we get an equation that relates the Pi to P_(i+1). This has a nice form because it gives us a recursion. If we knew pi(i), we could then immediately calculate pi(i+1). So it's a system of equations that's very easy to solve almost.

But how do we get started? If I knew pi(0), I could find by pi(1) and then use this recursion to find pi(2), pi(3), and so on. But we don't know pi(0). It's one more unknown. It's an unknown, and we need to actually use the extra normalization condition that the sum of the pi's is 1. And after we use that normalization condition, then we can find all of the pi's.

So you basically fix pi(0) as a symbol, solve this equation symbolically, and everything gets expressed in terms of pi(0). And then use that normalization condition to find pi(0), and you're done. Let's illustrate the details of this procedure on a particular special case. So in our special case, we're going to simplify things now by assuming that all those upward P's are the same, and all of those downward Q's are the same.

So at each point in time, if you're sitting somewhere in the middle, you have probability P of moving up and probability Q of moving down. This rho, the ratio of P/Q is frequency of going up versus frequency of going down. If it's a service system, you can think of it as a measure of how loaded the system is. If P is equal to Q, it's means that if you're at this state, you're equally likely to move left or right, so the system is kind of balanced. The state doesn't have a tendency to move in this direction or in that direction.

If rho is bigger than 1 so that P is bigger than Q, it means that whenever I'm at some state in the middle, I'm more likely to move right rather than move left, which means that my state, of course it's random, but it has a tendency to move in that direction. And if you think of this as a number of customers in queue, it means your system has the tendency to become loaded and to build up a queue.

So rho being bigger than 1 corresponds to a heavy load, where queues build up. Rho less than 1 corresponds to the system where queues have the tendency to drain down. Now, let's write down the equations. We have this recursion P_(i+1) is Pi times Pi over Qi. In our case here, the P's and the Q's do not depend on the particular index, so we get this relation. And this P over Q is just the load factor rho.

Once you look at this equation, clearly you realize that by pi(1) is rho times pi(0). pi(2) is going to be -- So we'll do it in detail. So pi(1) is pi(0) times rho. pi(2) is pi(1) times rho, which is pi(0) times rho-squared. And then you continue doing this calculation. And you find that you can express every pi(i) in terms of pi(0) and you get this factor of rho^i.

And then you use the last equation that we have -- that the sum of the probabilities has to be equal to 1. And that equation is going to tell us that the sum over all i's from 0 to m of pi(0) rho to the i is equal to 1. And therefore, pi(0) is 1 over (the sum over the rho to the i for i going from 0 to m). So now we found pi(0), and by plugging in this expression, we have the steady-state probabilities of all of the different states.

Let's look at some special cases of this. Suppose that rho is equal to 1. If rho is equal to 1, then pi(i) is equal to pi(0). It means that all the steady-state probabilities are equal. It's means that every state is equally likely in the long run. So this is an example. It's called a symmetric random walk. It's a very popular model for modeling people who are drunk.

So you start at a state at any point in time. Either you stay in place, or you have an equal probability of going left or going right. There's no bias in either direction. You might think that in such a process, you will tend to kind of get stuck near one end or the other end. Well, it's not really clear what to expect. It turns out that in such a model, in the long run, the drunk person is equally likely to be at any one of those states.

The steady-state probability is the same for all i's if rho is equal to 1. And so if you show up at a random time, and you ask where is my state, you will be told it's equally likely to be at any one of those places. So let's make that note. If rho equal to 1, implies that all the pi(i)'s are 1/(M+1) -- M+1 because that's how many states we have in our model.

Now, let's look at a different case. Suppose that M is a huge number. So essentially, our supermarket has a very large space, a lot of space to store their customers. But suppose that the system is on the stable side. P is less than Q, which means that there's a tendency for customers to be served faster than they arrive. The drift in this chain, it tends to be in that direction. So when rho is less than 1, which is this case, and when M is going to infinity, this infinite sum is the sum of a geometric series.

And you recognize it (hopefully) -- this series is going to 1/(1-rho). And because it's in the denominator, pi(0) ends up being 1-rho. So by taking the limit as M goes to infinity, in this case, and when rho is less than 1 so that this series is convergent, we get this formula. So we get the closed-form formula for the pi(i)'s. In particular, pi(i) is (1- rho)(rho to the i). to

So these pi(i)'s are essentially a probability distribution. They tell us if we show up at time 1 billion and we ask, where is my state? You will be told that the state is 0. Your system is empty with probability 1-rho, minus or there's one customer in the system, and that happens with probability (rho - 1) times rho. And it keeps going down this way. And it's pretty much a geometric distribution except that it has shifted so that it starts at 0 whereas the usual geometric distribution starts at 1.

So this is a mini introduction into queuing theory. This is the first and simplest model that one encounters when you start studying queuing theory. This is clearly a model of a queueing phenomenon such as the supermarket counter with the P's corresponding to arrivals, the Q's corresponding to departures. And this particular queuing system when M is very, very large and rho is less than 1, has a very simple and nice solution in closed form. And that's why it's very much liked.

And let me just take two seconds to draw one last picture. So this is the probability of the different i's. It gives you a PMF. This PMF has an expected value. And the expectation, the expected number of customers in the system, is given by this formula. And this formula, which is interesting to anyone who tries to analyze a system of this kind, tells you the following.

That as long as a rho is less than 1, then the expected number of customers in the system is finite. But if rho becomes very close to 1 -- So if your load factor is something like .99, you expect to have a large number of customers in the system at any given time.

OK. All right. Have a good weekend. We'll continue next time.

6.041SC Probabilistic Systems Analysis and Applied Probability
Fall 2013