

**Decision Making Under Uncertainty**  
**Prof. Natarajan Gautam**  
**Department of Industrial and Systems Engineering**  
**Texas A & M University, USA**

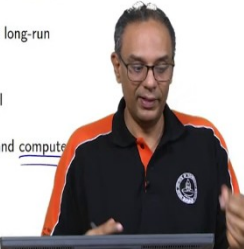
**Lecture – 32**  
**Markov Decision Process Set Up**

In this lecture, we will talk about Markov Decision Processes. We will essentially present a quick setup like a problem description and then move on in the next lecture to analysis.

(Refer Slide Time: 00:25)

From DTMC to Markov Decision Process

- In the DTMC example we were told the rule: order when inventory is less than 2 and order to bring the quantity to 5
- However in a Markov decision process (MDP) those two numbers, frequently written as  $(s, S)$ , are results of decisions or actions
- At the end of a day the inventory level is assessed and then one decides whether to order, and if yes, how much
- Note that the observation time in the DTMC is different from when the MDP decisions are made
- The rule is a "policy" but one can think of other policies
- The question is what is the optimal policy
- Say there is a cost of placing an order (irrespective of how many get delivered), there could be a constraint on how many washing machines can be stored on the floor, and a loss of revenue from unmet demand
- We could formulate an MDP and determine the optimal policy that minimizes the long-run average cost per unit time
- It is known that the structure of the optimal policy is of the  $(s, S)$  type
- Notice that the policy is stationary, deterministic, and monotone in inventory level
- Optimal values of  $s$  and  $S$  would depend on the costs and constraints
- For each  $(s, S)$  we can build a DTMC with  $S - s + 1$  states, obtain the  $\pi$  values and compute long-run average cost per day



NPTEL

So, we are going to take a leap from Discrete Time Markov Chains to Markov decision processes. So, the example that we saw, we were told a rule: whenever the inventory level of the washing machines goes below to, go ahead and place an order and order enough to bring the quantity to 5.

So, either you have 1 or 0 and then you order 4 or 5. So, that was the policy or rule we were told. Now, Markov decision processes are essentially once, where you come up with such policies. For example, this policy is called the  $(s, S)$  policy, this little  $s$  is when you observe the system being less than that, you place an order and this upper case  $S$  is the order up to level. So, this  $s$  is our 2, this  $S$  is our 5.

So, this 2 and 5 are typically written as little  $s$  and big  $S$ . Now, notice that the big  $S$  here is not the same as the state space. So, I know there are just letter  $s$  people seem to like a lot.

However, this is different and just be a little bit careful about it. So, at the end of the day remember you assess the inventory level and then make a decision of whether to order and if yes how much to order. Now, this is a situation where the discrete time Markov chain state is different from when the MDP decisions are made.

So, that means, in the Markov Decision Process- you observe the system at the end of a day and take an action such as should we order more or should we let it go. Now, that decision is made at the end of the day. However, your state observation is at the beginning of a day. Now, that is purely for convenience reason.

I just wanted to clarify that for you clearly. Now, the rule like I said a little while ago is what we call as a policy, you could think of other policies, there is no reason that your policy should be between 2-5 or for that matter even be of the  $(s, S)$  type: little  $s$ , big  $S$  type; does not have to be. Now, the question is what is the optimal policy, what policy is going to minimize your costs.

So, for example, if we have a cost, a physical cost for placing an order, it does not matter how many get delivered. It is just the physical cost of ordering a vehicle to take the stuff from the warehouse and bring them to the floor of the store. That is going to be a fixed placing order placing cost, there could also be some type of constraints and how many washing machines can be stored on the floor right; you cannot store 10,000 on them, maybe you have enough space to store 5 or 6 or 7.

And another issue is we have to be concerned about the loss of revenue from demand that does not get met, some people sometimes use an inventory cost. There are many other costs that one can consider and based on all that what one would do is formulate a Markov Decision Process, from now on I am going to call it MDP and find what is the optimal policy. A policy that minimizes the long run average cost per unit time. Now, if you were to do something like what is written here in this and go ahead and solve the MDP, it is a little bit beyond the scope of this course.

But, let us say you are able to do that then. In fact, you can show that the optimal policy is of  $(s, S)$  type. So, the policy that we saw was not just picked out from thin air, it was the policy that was derived as an optimal solution to a setting, where there are costs- for ordering there was a cost, for storage there were a cost, for loss of revenue and so on. Now, notice the following attributes of these policies, the policy is stationary, what does that mean? That

means, irrespective of what happens if my inventory level is below  $s$ , I place an order, if my inventory is above little  $s$ , I do nothing.

And stationery does not depend on whether this is a particular time of the year or anything like that. It is also deterministic, it is not like if my inventory level goes below  $s$ , then I flip a coin and if I get heads I order, if I get tails I do not order. That is not what we are doing, I have a deterministic policy, I do this or I do that.

Also the policy is monotonic in inventory level, that means, I have a threshold which is little  $s$  and I say from below that I order, if it is above that I do not order. That means, there is a monotonicity. For example, you would never get a situation where you would order when the inventory level is 4, you will not order it is 3 and again order at 2, you will not go up and down like that. You would either not order above a certain point, and below a certain point is always order. Some monotonic properties are typically expected in this system.

Now, the optimal values of little  $s$  and big  $S$  of course, would depend on costs and constraints. So, usually in an MDP exercise, people will typically show that the policy is stationery, determination in monotonic. The policy has a certain structure, then we will go and do some analysis to compute what is the optimal values a little  $s$  and big  $S$ .

So, one thing that we could do in this problem is using the costs build a discrete time Markov chains, compute obtain the  $\pi$  values we saw a little while ago, and compute the long run average cost per day and among all little  $s$  and big  $S$ , pick the one that minimizes my cost, we could clearly do something like that.

(Refer Slide Time: 06:24)

Markov Decision Process: Example

- We now consider a maintenance example where a company has two identical machines ✓
- At the end of a day each machine can be in one of three states: good (G), deteriorated (D) and bad (B).
- In state G one typically does nothing and the machine produces quality parts the next day
- In state B the machine does not produce anything the next day if one does nothing, so in practice one would repair
- In state D one has the choice of doing three things (nothing, tinker or repair), when nothing is done, the parts produced next day would be of lower quality
- If the policy is to do nothing in all three states, then the state of a single machine follows a Markov chain with transition probabilities

State	G	D	B
G	0.85	0.1	0.05
D	0	0.75	0.25
B	0	0	1

→ Note that if we do nothing, B becomes an absorption state and the DTMC is not irreducible



Let us look at an example, I do want to take a different example because I do not want you all to think the only example we can do is one in inventory. Now, we are going to look at an example in maintenance, which is a slightly different type of a setting. So, we are looking at maintenance situation, where a company has 2 identical machines, both these machines are identical to each other. At the end of each day now, this part is similar to what we had before, end of each day, we observe the machine and we figure out what state the machine is.

The machine could be in one of 3 states: it could either be good for G, it could be deteriorated or it could be bad, these are the 3 possible states. Good is what you would like it to be; deteriorated- you can live with; bad- you do not want it. So, what do you do, if you observe the system in a good state, you typically do nothing and the machine will continue to produce quality parts. Now, we have 2 machines, but I am going to talk about 1 machine because the 2 machines are identical.

So, we will pick 1 machine and talk about it and if this machine is observed to be good, you do not touch it, we will let this be the way it is. We are not going to mess it up. Now instead B, which is the bad state, it is not the middle state, but the last state. The machine would not be able to produce anything of any quality that you could do anything with, there is no reason for it to even produce if you are in state B.

So, if you do nothing, then basically is going to produce rotten parts. So, in practice you would always repair in state B. So, you do not have much of a choice, but you would perform

a repair in state B. However, the third state D is really the interesting state, in that you have the choice of doing one of 3 things. If you observe the system in state D that means it is deteriorated: it is not as good as good, it is not as bad as bad.

So, it is somewhere in between and that is called deteriorated and in that situation you have 3 choices: you could either do nothing, which is just fine; or you can tinker, by tinker what we mean is somebody locally might just take some quick stuff and fix something quick and you do not have to bring someone from outside to do a major repair. So, 3 choices: bring somebody from outside to a major repair, do a little tinkering job or do nothing. Now, when nothing is done, then the parts that are produced would be of lower quality because the machine itself deteriorated. So, the quality of the parts that are being produced will go down.

Now, imagine we have collected historical data and we can come up with a matrix like structure or a table in the following way. If today's observation, a machine- let us not worry about the 2 machines, let us think about 1 machine. One of those machines and they are identical, they behave exactly the same way, if one of those machines is in a good state today, then tomorrow also it will be good with probability 0.85.

So, when I observe tomorrow, there is an 85% chance that it will still be good. However, there is a 10% chance that it would have deteriorated and there is a 5% chance that it would have gone bad. So, today if I started with good, this is the evening I observe and I observe the system to be in a good state, then tomorrow evening when I observe, there is an 85% chance that I would still be good, there is a 10 percent chance that I would have deteriorated and there is a 5 percent chance that I would have gone bad.

Now, from deteriorated you can never become good because the policy right now is to do nothing in all the 3 states. So, if you are going to be doing nothing in all the 3 states, you will never go from deteriorated to good, you will have to do some type of tinkering or some type of repair for it to go from deteriorated to good.

So, that the probability is 0. However, from deteriorated, it can stay deteriorated with a 75% chance and it could go bad with a 25% chance. So, from good, you can go to deteriorated or you can go to bad. From deteriorated, you could only go to either deteriorated or go to bad; from bad you are stuck with bad. Now, that is very bizarre, we have never seen something like this. Notice that we are not doing anything. So, if you do not do anything, the bad machine continues to stay bad.


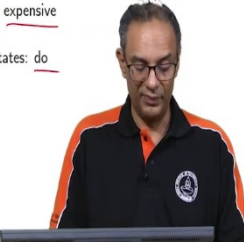
So, this state B is what is called an absorption state, if your policy is to do nothing in all the states. However, if you look here, we have stated very clearly that in the bad state, we would typically repair. So, in practice we will not be in that situation, but I just wanted to show this to you to let you know that, yes you could have an irreducible DTMC.

If you went with something like this and in the long run the machine will eventually be back. So, that is what this means. So, we do not want this, we will try and repair.

(Refer Slide Time: 11:30)

Modeling Two-Machine System States

- ▶ When a machine is in state B, the only action (or decision) is to repair
- ▶ Also, in state G, the only action (or decision) is to do nothing
- ▶ The question is what action to take in state D? Do nothing, tinker or repair
- ▶ Since the state is observed at the end of the day, the machines are tinkered or repaired the next day during working hours
- ▶ Now consider both machines as a system which can be in six states GG, DG, DD, GB, DB and BB at the end of each day
- ▶ Note that since the machines are identical we do not specify in states DG, GB and DB which machine is in which state
- ▶ Clearly we would do nothing in GG and perform repair in GB, DB and BB
- ▶ Note that tinkering is done by mechanics in the facility while a repair would be an expensive operation by outside technicians
- ▶ From an economic standpoint there are only two options in each of DG and DD states: do nothing or tinker
- ▶ There are three types of costs
  - ▶ Loss of quality cost (by using a D machine for a day)
  - ▶ Maintenance cost (for both tinkering and repair)
  - ▶ Loss of productivity cost (when repair/tinkering is performed)

So, now let us talk about the two machine system. Now, we have two machines and they are identical. So, I am not going to make a big difference deal about which machine is which.

So, I am going to make the following restriction- if a machine is in a bad state, then the only action I take is to repair it. If the machine is in a good state, then the only action I take is to do nothing. So, I do nothing to a good machine, I always repair, that means I call and bring the fancy repair people from outside and do a little repair. Now, what about in D, what do I do? I either do nothing or I tinker or I repair.

So, I have these options. Now, I have to make some assumption. The first assumption that I make is: if I observe this at the end of the day, the machine is in a state where I have to repair it or tinker it, then the machine the tinkering or the repair happens only the next day because you are observing at the end of the day, people have gone home, you get a chance to repair it only the next day.

And so, that is one of the important things for us to remember. Now, look at this way the system state of all. There are six possible states, either both machines can be good; one machine could be deteriorated, while the other is good; or both machines could be deteriorated; or one machine is good, one machine is bad; or one machine is deteriorated and one machine is bad; or both machines are bad at the end of each day. Now, the machines are identical. So, we are not going to make a big deal out of which machine is in which 6.

So, when it is in DG, I am not saying the first machine is D, second in G, it could as well be the first machine in G and the second machine in D; it could have been either ways. So, the order in which I list is not important, I want you to clearly think about that because I could do that, that will increase unnecessarily increase the state space by 3 more states. We do not need that, yes it will make the P matrix computation a little bit more tricky.

So, in these 3 states, where the machines are in different state, the other state is not going to be a big deal because the both machines are in the same state, in these 3. The other 3 states and they are in different states, we are not going to say which machine is in which state. Now, in the state GG, we are not going to anything. In these 3 states, when there is at least one bad machine, we will always call someone and perform a repair. So, the actions are pretty straightforward. Yes, I did pick an example that is easy to show in an online class and in small examples, the example is pretty small. So, your action is obvious.

Now, I also want to say the following- tinkering by itself is done by people inside the facility, inside the company and it is going to be not so expensive. However, when you call someone from outside, you would do an expensive repair and these are done by outside technicians. So, the only two situations, where you would do any type of tinkering or repair is in DG or in DD. However, notice that in the deteriorated stage, and at least one of them is deteriorated and the other is either good or deteriorated, then it makes sense. There too your 2 options are either to do nothing or to tinker, you would not call an outside technician because the outside technician costs are going to be much higher.

So, from an economic standpoint, if you are just in a deteriorated state, you would either do nothing or you would tinker. So, I am going to make the options even simpler and in fact, we saw this even in some of the other cases like the decision trees, where we had explicitly not worried about this situation, where you had a more expensive option to even consider.

So, we are not even going to be bothered with that. So, turns out that there are three types of cost. The first cost is a cost of quality loss, right remember if you had decided to do nothing to deteriorated machine, next day it is going to be produced parts that are of lower quality. That would mean that the company itself will incur a cost. There is also a maintenance cost. Now, there is a different maintenance cost for tinkering, there is a different maintenance cost for repair.

Now, remember that when you perform a repair, it is only done the next morning. So, the machine is not going to be available for a little while and you will lose productivity during the time. So during that productivity lost, you lose some potential for making profits. So, that is called loss of productivity.

(Refer Slide Time: 16:13)

Quality, Maintenance and Productivity Costs

Decision	State	Quality Cost	Maintenance Cost	Productivity Cost	Total Cost
Do Nothing	GG	0	0	0	0
	DG	10	0	0	10
	DD	20	0	0	20
Tinker	DG	0	15	10	25
	DD	0	30	10	40
Repair	GB	0	40	15	55
	DB	0	45	20	65
	BB	0	50	20	70

- ▶ The costs are multiple of ₹ 1000
- ▶ The costs are total for both machines
- ▶ The states and decisions are at the end of a day, the costs are for the next day
- ▶ Actions of tinker and repair would both result in GG at next observation
- ▶ Four policies: (1) Do nothing in DG and DD; (2) Tinker in DG and DD; (3) Do nothing in DG, tinker in DD; (4) Do nothing in DD, tinker in DG
- ▶ Which policy would result in the lowest long-run average cost per day?



So, I am going to represent that in a graphical manner in a table. So, turns out that these two states or the states that gets repeated, all the others are the same. So, when you are in state GG both machines are good, your decision is always to do nothing. When you are having one machine that is in a bad state, you cannot afford to not call a repair person, you will surely call them and you would perform repairs.

So now, the ones that could either be in the tinker option or the do nothing option are these states DG, you could either tinker or do nothing; and DD you could either do nothing or tinker. Let me tell you a little bit about the cost, remember that you only incur a quality costs if you do nothing and you have a deteriorated machine. So, in the first stage, there is only one



deteriorated machine. So, you lose quality of only 1, in here there are 2 deteriorated machines. So, you lose quality for both machines products.

So, you are going from 10, the quality cost goes to 20. You do not have any quality cost in the other states because as soon as the observant in there let us say the next morning is going to be a real good nice machine. So, you are not going to lose quality. Now, if you do nothing, there is no maintenance cost right. If you tinker, there is this maintenance cost. So, when you are in DG, you only do one tinkering, so it is 15; if it is DD, both machines need to be tinkered, so the cost is double.

However, the maintenance cost is more tricky, you need to have someone come over, maybe you might have to pay for their airplane fare or something like that. It is going to be expensive to bring them, plus the cost to perform the repair. So, if you think about it, the 3 cases here there is a maintenance cost, although that is the only decision. It is much higher than tinkering, that is why I said in general you would not repair, if you can tinker because the state of the machine is going to go back to being good, even by the tinkering people.

But, if it is a rip, if it is in a bad state, the local people cannot tinker. So, here also if you see when there are 2 machines that are in a bad state, it is more expensive to do maintenance. And it is cheapest to do maintenance of one machine is good, the other is bad because only one machine needs to be maintained. Whereas, if one is in deteriorated, that will also be taken care of by the repair people. So, we are not going to be tinkering, but you go to a proper repair.

So, it is a little bit more expensive. Now, let us come to the productivity cost, there is no loss in productivity if you did nothing, but the next day the machine is not going to be in downtime for a while. So, there you lose some productivity costs. I am going to assume that the repairs happen in parallel and therefore, the productivity cost is only is exactly the same, whether you tinker one machine or both machines because you are going to be doing it in parallel.

However, in this situation notice the following: these 2 machines are being repaired in parallel and therefore you lose some productivity cost, and however in this situation, we are saying you do not lose much productivity cost on one machine, on just on the other one because one of the machines is good, it continues to produce.

So, if I add up all the numbers in these states, it tells me what it would cost under these various situations. So, for example, if you look at this 70 number, it is basically the sum of 50 and 20; if you take this 40 number, it is the sum of 30 plus 10 and so on. So, these costs, the total costs are essentially sum of all the cost. So, at the end of the day, if you are in this state and you take this action then this will be the cost. Notice that the costs are multiples of 1000 rupees and the costs are in total for both machines together.

Also, if the states and actions, a decision or end of a day and the costs are applied for the next day. Further, if I were to tinker or repair, the next day's observation is guaranteed to be in GG. This is an assumption we make, that is the machine does not deteriorate the day it is repaired and fixed is really good, it will at the end of the day it will be in GG, both machines will be good.

So, we are looking at basically 4 policies- I either do nothing in the DG and GG states. So, I do nothing in both, my 2<sup>nd</sup> option is tinker in both states; my 3<sup>rd</sup> option is do nothing in DG and tinker in DD- this is my option 3; my 4<sup>th</sup> option is this, that is do nothing in DD, but tinker in DG. Which policy do you think will reduce the long run average cost- which one would be the lowest long run average cost per day? This is something that we wish to compute, we will do that next.

Thank you.