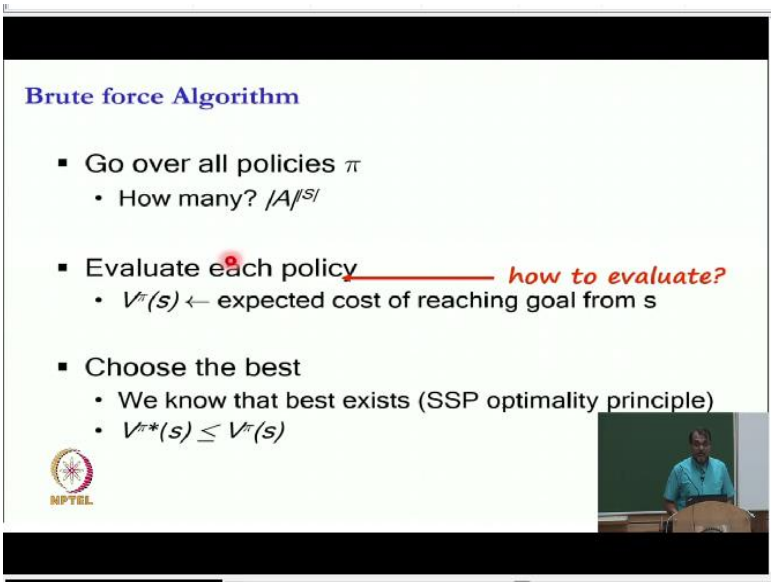
Artificial Intelligence
Prof. Mausam
Department of Computer Science and Engineering
Indian Institute of Science Education and Research-Delhi

Lecture-72
Markov Decision Processes:
Policy Evaluation Using System of Linear Equations
PART-3

Now suppose we wanted to start creating an algorithm for an MDP. What is an algorithm, an algorithm would be computing the policy which is the best policy, the best action to take in each state. Now I can create a very brute force algorithm. A very brute force algorithm could be something like this.

(Refer Slide Time: 00:48)



Go over all policies pi, evaluate each policy, figure out how good each policy is, choose the best policy. Now first question for you to answer is what do we not know here or can we easily compute this, can we achieve this or is there a step that we do not know so far. So the step that we do not know is given a policy. How do we compute, how do we evaluate it okay, that is a step, we so for do not know. So we will work on that next.

Another question for you to answer is how many policies are there or do we have infinite policies or finite number of policies. Let us say I give you a finite number of states. Let us say I

give you a finite number of actions. I told you that policies are mapping some states to actions. Now do I have a finite number of policies or infinite number of policies. Finite number of policies and how many policies do I have.

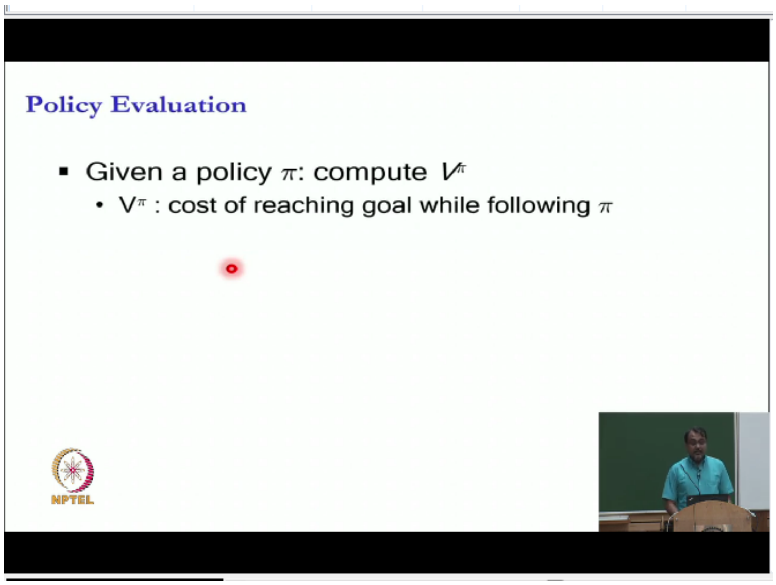
How many, what is your name Prathiush, Prathiush says A raise to S and why does Prathiush say A raise to S. Because for the first state how many actions can I have A, for the second state how many actions can I have A. So I keep multiplying A for every state I get A to the power S. So this is as many policies as I have access to. Now the next thing is I need to somehow think about how do I evaluate each policy and take the best right.

And for evaluating each policy I am going to define a new concept called V of pi. This is a value function V pi of S which says let us say I start in state S, what is the expected cost of reaching the goal if I am always following the policy part, moreover we need to figure out how to answer this, how to compute this V pi, the third thing we do not know is if a best policy even exists in this model, now that requires us to go into theory of a Markov decision process which we will not go into for this entry level class.

But I will tell you that A there is a best policy or many but there is at least A best policy and the definition of best policy. Let us say that policy is called pi star then the definition of the best policy is that V pi star of S is always less than equal to V pi of S for any policy pi and every state S. So a best policy is best starting in every state or as good, as good or better. Then any policy starting in every state.

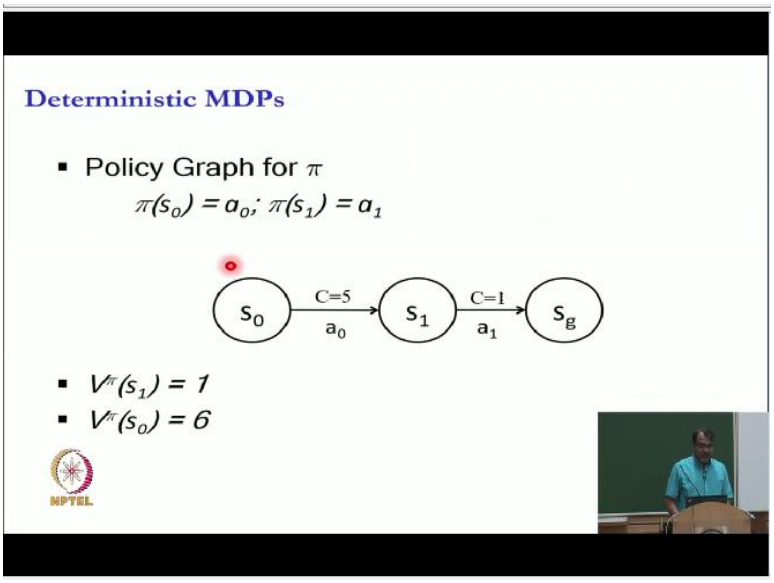
So this we will take for granted in this class that there exists a best policy pi star which has this characteristic that its value function is the lowest for every state, okay. Now we can still do the brute force algorithm. Unfortunately, we still have to figure out this second step. How do I compute V pi of S, that is what we are going to sub goal on next.

(Refer Slide Time: 04:38)



And we will call this particular problem the problem of policy evaluation okay, the policy evaluation problem is given a policy pi which is a mapping from states to actions compute V of pi, that is cost of reaching the goal following policy pi, starting in every state. Now let us build it up again. So this is not going to be very hard.

(Refer Slide Time: 05:04)

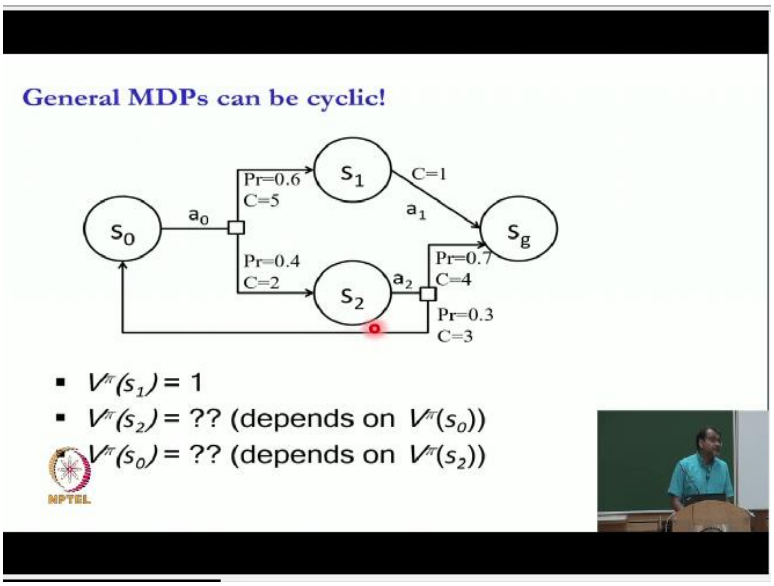


So if I am giving you a deterministic MDP, with every policy I can write down a graph for it. A graph where from every node only one edge comes out, every state node, only one edge comes out which edge the action right. That policy pi tells me, that I should take in that particular state. So let us say policy pi says in S 0 do a 0 in S 1 do a 1. And let us say that is the policy graph I get, you know in S 0 you have to pay cost 5, you get to S 1.

In S 1 you have to pay cost 1, you have to get to the goal. So then you can easily compute V pi of you know, S 0 S 1 and goal. What is V pi of goal, 0 how much cost do I have to pay to reach the goal, nothing. I am already in goal. What is V Pi of S 1. One because I pay cost 1 and reach the goal. What is V pi of S 0, I pay costs 6 and reach the goal. This is a deterministic MDP. This is exactly how I can backup the values in any kind of a search algorithm.

In fact in a path, this is the path, I am giving you a path. Now if I give you an acyclic MDP, it will not be much harder.

(Refer Slide Time: 06:27)



So let us say I give you this acyclic MDP in S 0 do a 0, there is AND node via because you know, there are probabilities okay, no problem with 0.4 you pay cost 2 get to S 2 with 0.6 you pay cost 5 get to S 1, then you can take different actions here. You can get to the goal. If I give you such a graph, is it going to be very hard for you to compute this no, V pi a goal will still be 0. V pi of S 1 will be 1, V pi of S 2 will be 4.

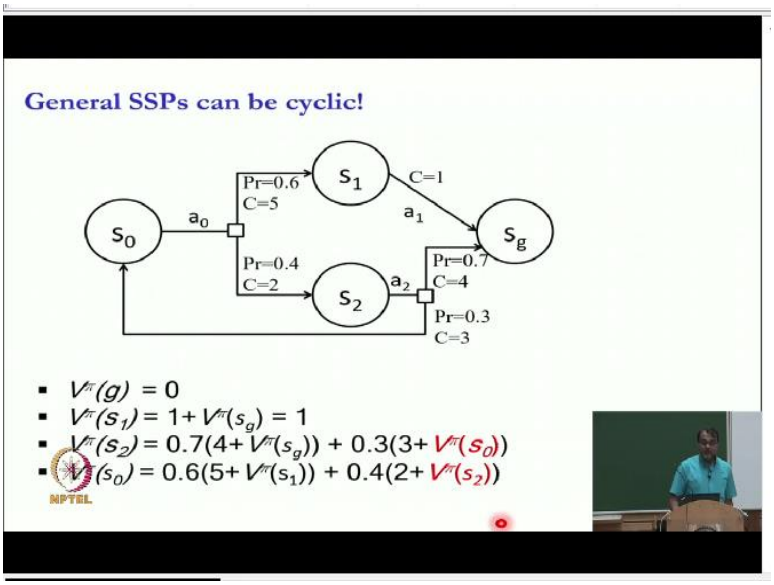
Because these are deterministic edges and even V pi of S 0 will be nothing but 0.6 times 5 cost + 1 which is the V pi of S 1 and 0.4 times cost 2 + 4, which is the V pi of S 2. And that will still be 6. This is not very hard. In fact, you can still backup values like expectimax and you can get to

the policy evaluation for the acyclic problem. However, life becomes interesting when we have a cyclic MDP, right.

So let us say we give you this kind of an MDP. Here with S 0, you get to S 2 with some probability, but when you take an action a 2 and S 2 with small probability you come back to S 0. Now here you can compute V pi of S 1 easily which will be 1 and goal which will be 0 but you do not know how to easily compute V pi of S 2 and V pi of S 0 because V pi of S 2 depends on V pi of S 0 and V pi of S 0 depends on V pi of S 2.

Now there are 2 ways to deal with this. The first is let us write down the system of equations. So if I want you to write down the equation for V pi of S 2 what would you say 0.7 times 4 + 0.3 times 3 + V pi of S 0 0.7 times I pay cost 4 get to the goal. So no more cost 0.3 times I pay 3 and get to S 0 and what is the cost I pay long-term V pi of S 0. Similarly, I can write down the equation for V pi of S 0 0.6 times 5 + V pi of S 1 which is 1. And 0.4 times 2 + V pi of S 2.

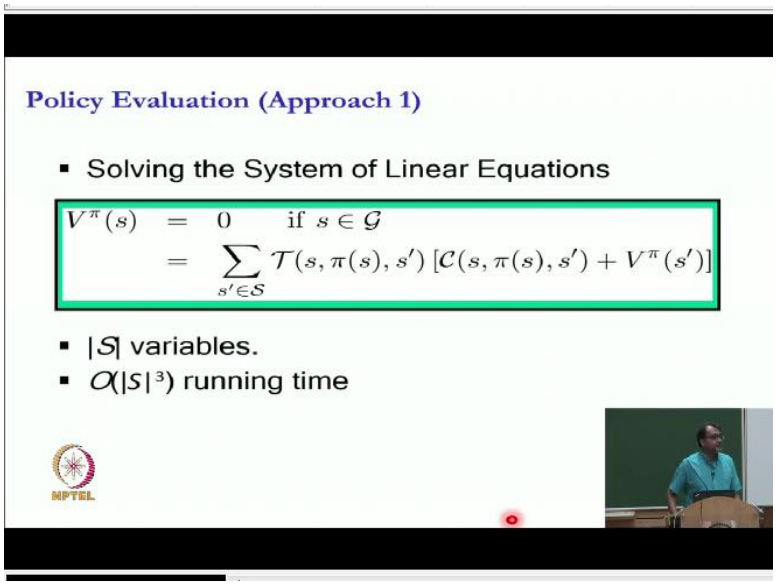
(Refer Slide Time: 09:22)



So technically I can write this down as a system of equations. This system of equations is in 4 variables and these are linear right, because that every state I am computing the expectation which is summation of some probabilities which are given to us times, some costs which are given to us plus some long-term values which are using the variables that are in my problem. So

I can write this down as a system of equations. These are S equations in S variables. I can solve it.

(Refer Slide Time: 10:07)



So, if I have to do the general version, the general version would be look, V pi of S will always be 0 if S is a goal. Work with me on this slide. This is an important slide. However, if I am not in goal, I am starting in state S which action will I take we are evaluating policy pi. So which action will I take. Come on, this is easy, policy pi is a mapping from states to actions. So in state S which action to be take pi of S, very good.

If I take action pi of S I made each any state. I do not know which state am I going to reach. Let us say I reach state S prime. What is the cost that I will be paying, cost immediate cost, what cost will I pay right there, C of S, pi S, S prime, it is important. Let us make sure we understand this. Cost function takes s a s and gives me a number. The first argument is the current state. The second argument is the action which is pi S.

The third argument is the next state which is S. This is the cost I will pay right away. Plus there is some long-term cost I am going to pay. What is the long-term cost I am going to pay V pi of S prime, why because V pi of S prime has been defined to be the cost to reach goal starting in state S prime. And now what have we done. We have reached S prime. And we have made all these Markov assumptions and so on so forth.

So current state is the only state determines the future nothing else in the past matters. So we will additionally pay the long-term cost of V pi S prime. This is the total cost we pay if we take action pi S in S and we reach S prime, but do we know which S prime do we reach. Do we know the S prime. We do not know S time, in the worst case how many S primes can we reach, modest cardinality of the state S.

So we have to take an how do we deal with lots of S's S prime's, we take an expectation right because we have to minimize the expected cost and this is the point we take an expectation. So we take an expectation like this and that is it. This is my system of equations. And now I can use your favorite, you know, gauss-seidel method as you know, whatever Gauss elimination methods or any of your favorite methods and compute V pi of s.

This would be a system of equations in S variables. So it will take an S cube running time. You can argue that that will be too slow, you will be right, but for now we are not talking about scalability. Any questions on this.