NPTEL

NPTEL ONLINE COURSES

RENIFORCMERNT LEARNING

Policy Iteration

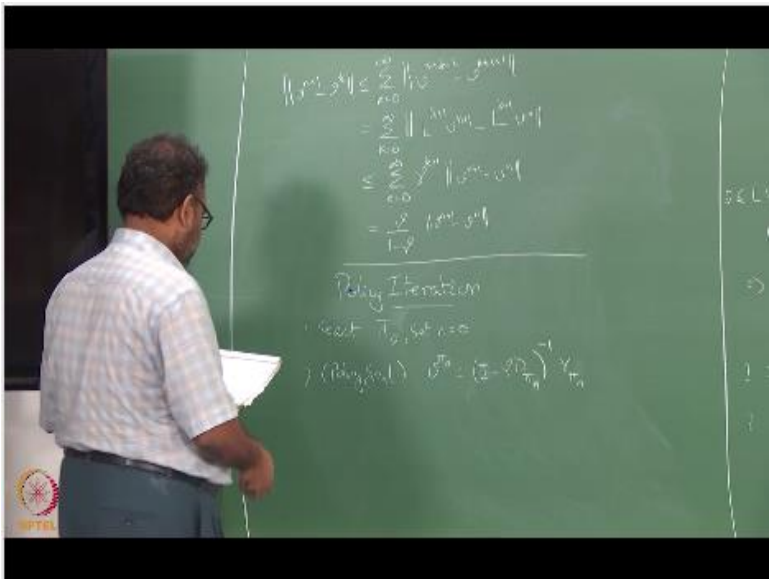
Prof. Balaraman Ravindran

Department of Computer Science and Engineering

Indian Institute of Technology Madras

So we need to do policy iteration so shall we are, should we not when we started late I know it's my mistake here oh okay, so that leaves us with 15 minutes I think, we could do it in 15 minutes, so policy ration the idea is very simple, right so I start off with some arbitrary policy let us say π_0 may start off with some arbitrary policy π_0 I find weep I not, right then what do, I do I be greedy with respect to $v\pi_0$ I will find π_1 right.

(Refer Slide Time: 02:25)

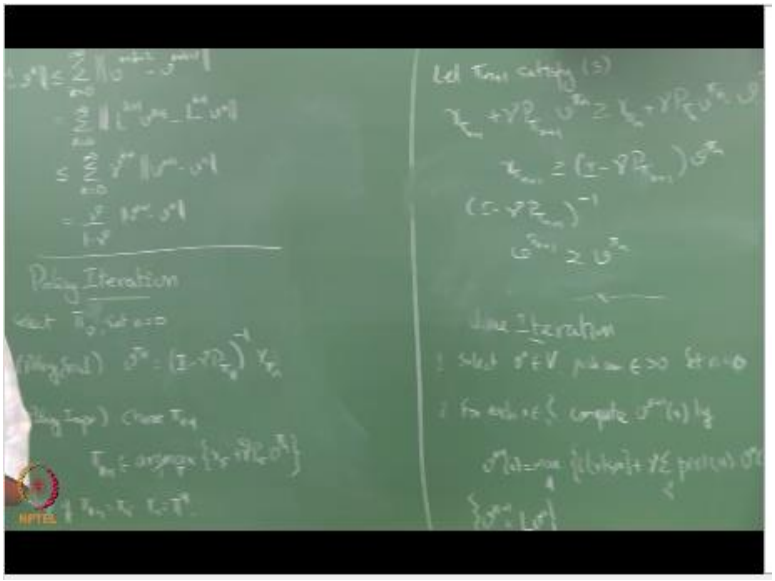


So being greedy is this right so I will take some $\pi 0$ solve fine $v\pi_0$, right then do this and I will find $\pi 1$, right and then I find $v\pi 1$ plug that backend here right, so essentially I will plug in VN

here find πn+1 right, so keep going until I converge and convergence here is more straight forward if you stay with the same policy you converge where unusually convergence is much faster the rate of convergence much faster with policy, iteration but, for variety of reasons value iteration remains a very favorite algorithm for many war people don't like policy iterations for, some reasons.

So but people get the idea for the policy iteration is right you can go implement policy patients rather easy, right so first so let's of arbitrary sorry I hate going back and forth select an arbitrary π0, right second step is called, policy evaluation so where you say, refined vπ0 right so policy evaluation is just find vπn right so we can just do this by solving the system of equations I can just take the inverter inversion in the inverse of that matrix and solve it right or if you like it you could do the iterative fashion just just like you did in value iteration instead of iterating on L I can iterate on lπ, right and then I let have some stopping criterion for stopping it but I can do this as well next thing is called policy.

(Refer Slide Time: 02:25)



Policy improvement okay this is a second step is called policy evaluation third step is called policy improvement, just like we mentioned earlier it says mark Maxes then component-wise, right for every state yes you do the max, it is whatever I wrote here before erased it at the beginning at exactly that, only difference is here I am plugging in vπn it is not V* right so this vπn and I do this argmax over π and at every step because I am checking for equality for stopping right.

So there could be multiple actions that give you the max here right, so if at step n you chose an action for giving a max like step n there is some action your chosen as policy for a state s in step n+1 if the same action belongs to the max set you will pick that actually you do not pick an arbitrary action from the maxim, suppose let us say going up and going right right both are optimal, right but in the previous step I had chosen going up for the state this step I should chose I should choose going up also in otherwise I will just keep oscillating between up and right up and right I might not figure out that I have stopped right.

So I shouldn't choose an action from the maximizes randomly right are arbitrarily I have to have some mechanism by which I consistently break ties so that, policies do not change when they do not have to, πn+1 =πn stop and declare that πn is 5* otherwise go too, well otherwise increment n and go to step 2, right so it's rather easy, right so if the stopping condition holds can you see why πn should be π*, that doesn't mean anything it just means it's a fixed point, yπn should be π* well it is a fixed point but it is a fixed point of L.

If you think about it max over π rπ+γpπvπ ray that operation that looks like L that is yell operating on vπ then if I get that π in the same beep I same π again, right so that means the value function hasn't changed, right so that means V=lv, so that means it is the optimal value function right so if πn=πn+1, because of the way I am generating the sequence of π and if πn=πn+ 1 then πn is π*, also I haven't changed right because the apology has it changes the value function will interchange fiesta no because of the way we are generating here right remain so.

So if the value function hasn't changed and this is essentially this part of it is essentially L operating on vπ and it does not change so that essentially means, that it is the optimal value function, okay and what else do we need to show I said if πn=πn+1 then πn is V* πn is a π* by the way right ,what else we have to show, that I have to show that happens right I will have to show that that happens so we will do that in two ways one we will have to show that so vπn+1 ≥ vπn, right.

So why is that why you think that will be greater, so basically πn+1 is a greedy policy according to be πn, okay so taken vπn and action greedily with respect to it and I get πn+1, so that is what I mean like πn+1 satisfy 3 then, right this is clear right so because πn+1 satisfies 3 so that is being greedy with respect to vn well πn is some arbitrary policy which is the same as this then that will give the same answer otherwise this will be greater, because this is the one obtained by maximizing this expression, wait I took this expression I am maximizing it and whatever gives me the maximum I am saying is πn+1, right.

So this is the maximum on the left hand side so the maximum has to be greater than or equal to whatever value I will get by plugging in some other πn here because this is the max okay this is clear right one more step, so I can say so why can I say this because this is actually equal to vn that the right hand side will be equal to vπn because it's a fixed point of lπ, so this is lπ operating the lπn operating on vπn such a fixed-point so this is vπn so essentially this is it and if I multiply both sides by, what do I get so that into this is what, right.

So every step I will keep improving okay so the last bit we need is to show that, there are only a finite number of policies that you can search through, if you are having a finite MDP finite number of deterministic policies we are searching through only deterministic policies here, only finite a number of deterministic policies you can search through if you have a finite MDP so that is basic approved it so policy iteration will converge, but every time I have to become better and

there are only finite number of things I can search through at some point so I 'll have to stop, or is it okay the component ways so every component, of v is greater than sorry I didn't define it earlier I should have so this essentially means set vn+1 of s1 is greater than or equal to v1 of s1 of s2 s3 for all s, right I think there are people clamoring to come in so that's why I stopped we can go out and you can ask questions.

IIT Madras Production

Funded by

Department of Higher Education

Ministry of Human Resources Development

Government of India

www.nptel.ac.in