NPTEL

NPTEL ONLINE COURSE

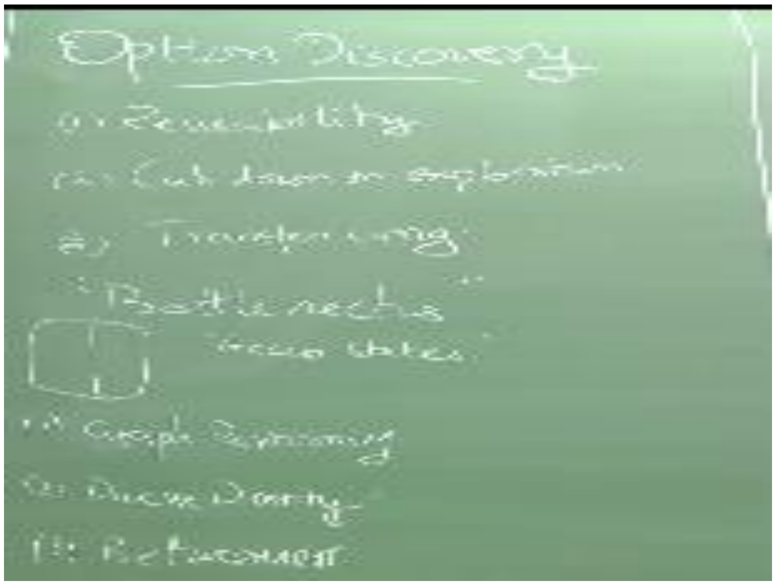
REINFORCEMENT LEARNING

Option Discovery

Prof. Balaraman Ravindran
Department of Computer Science and Engineering
Indian Institute of Technology Madras

I want to very quickly talk about option discovery right I mean forget about how we define max Q hierarchies and things like that the way the more complex the hierarchical formulation the harder it becomes to automatically discover them right but with options it is kind I easy right so you can actually talk about option discovery rather in a straightforward fashion.

(Refer Slide Time: 00:50)



Option Discovery
(i) Reusability
(ii) Cut down on exploration
(iii) Transfer learning
"Bottlenecks"
"access states"
(i) Graph Symmetry
(ii) Diverse Density
(iii) Betweenness

So what would be a good option for you to discover when we recall an option good okay so one thing is reusability okay second thing would be I would say that it cuts down cuts down on exploration right for all see to learn faster right and it aids in plans of learning but concerns and all of these are actually hard ways to evaluate how good an option is that if I want to say it cuts down on exploration and other things how will you how will I evaluate it and how will you find out an option that cuts down on exploration see little sticky right.

Where you have to run multiple times like with the one setup option then have to pick another set of options run multiple times to see if it cuts down exploration and things like that so what people typically tend to do is same stuff looking at the actual thing you want from options they try to use some kind of a surrogate function right like surrogate measures so what could be a surrogate measure that they use so the first thing that they use this what is called bottlenecks.

So what are bottlenecks something that is limiting that is a bottleneck right so you have a bottle in the next one bottlenecks are usually narrow right so something that kind of has like this funnel kind of an effect I have a lot of things but then they had all passed through a very narrow constriction to reach from one point to another right so for example have you seen any bottlenecks in any of the examples you have seen so far bringing into and then city of in terms of rebuilding doorways that even that simple to have problem that we looked at earlier right.

See doorways are actually bottlenecks because to go from one half of the MDP to the other half of the MDP we have only two states which acted as this bottle legs or sometimes called sometimes called access states because they allow access to some other part of the MDP right so people said we are going to find bottlenecks so why bottlenecks why are bottlenecks a useful heuristic to have well they obviously will cut down on exploration because if you know how to get to through this door very effectively then I can explore from here I can go to the states here very easily.

Otherwise if I am trying to do random exploration I have to hit those just exactly those two states to get from one room together so it makes it hard so it cuts down on exploration great what about reusability even if I keep moving the goal around mania I will probably have to go through this so it is probably something that I am going to reuse a lot even while learning right and also for solving multiple problems what about transfer learning again the same thing if I move the goal around also knowing where the bottleneck states are will be useful.

So essentially people say oh okay so bottlenecks are good things to find right so how do you find bottlenecks so depending on the level of information that you have about the problem right if I give you the p if I give you the transition structure of an MDP how will you find out bottlenecks okay let us say since you what you do so essentially try to segment the MDP we can model it as a graph right so two states are connected if there is an action that will take me to take you from one state to another right.
Now you model this as a graph and then try to find graph cuts let try and try to find the components of the graph that which are very weakly connected and then those states where the weak connection happens would be your bottleneck states right so there are several approaches that actually use this basic idea and so some kind of I am going to say there is some kind of some kind of graph partitioning ideas right so there is work by Shimon or way back 2003 or 2002, 2001 or something.

On where they do this kind of graph clustering to find the connected components and not connected component of crystalline to find the different components and then do it so we have done some work on using graph partitioning to find the options right and it is all of what Ram Anandan is doing now is essentially this right so a lot of lot of approaches for doing this okay what if you did not have this graph what if we did not have them DP how will you do it one simple thing is to say do a lot of just walk around randomly in the MDP collect data from the MDP and then do a graph partitioning is there anything else that you can do.

Turns out that you can do even simpler things right so one of the simplest thing is to look at what people call diverse density so what did I do in the diverse density approach is they're going to assume that you have many trajectories that you have generated solving a problem in the MDP many trajectories that you have generated solving a problem in the MDP right and many trajectories where you have generated where you have aborted you have not me reach the goal right.

So you have successful trajectories and you have unsuccessful trajectories right so you look at states that appear a lot on the successful trajectories but do not appear a lot on the unsuccessful trajectories again it is a heuristic but the heuristic essentially tells you that ok this is the state that I needed to get through to reach to the goal right if the times when it did make it through this state and ever reach the good but it is also possible.

For example in the rooms case it is possible that it came from one room to the other but still I aborted because I was taking too long in the other room that's also possible right but that is why I am saying it is not 0 in the unsuccessful trajectory is also the statement occur but it occurs disproportionately more in these successful trajectories you have to keep track of the trajectory we need lot of successful trajectories and a lot of unsuccessful trajectories and then on those successful trajectories I count I keep track of all the trajectories see all those extras have seen a few hundred few thousand trajectories how many ever you want right.

And from that you can do this is called the diverse density from so you it is more often occurred in the successful trajectories and less offer any unsuccessful trajectories right in fact there was the first option discovery paper that was written and unfortunately people are not given to much thought into more principled approaches to discovering options right and so that turned remains one of the most implementable approaches because it requires I mean however few data you give it our little data you give it can throw back some options at you right.

This kind of graph partitioning approaches you need a good model of the MDP before you get reasonable options out of it and people haven't really looked at you know what happens if you are having ill specified model and so on so forth we are doing some work on it now so hopefully something interesting would come out of it soon right and the third approach is essentially it is like graph partitioning right.

I would say it is still a graph partitioning approach even though the author would disagree with that it is based on between this right so between this is a concept from the network analysis community where a node has a high between nests if a lot of shortest paths pass through that node right so I take a pair of vertices on the graph that I find the shortest path between the pair of it is I find all say shortest paths that could be many shortest paths.

If we suppose you take a grid right I want to go from one co one point or another I can go any combinations of left is and rights right as long as the it covers the Manhattan distance between the two points that will be a shortest path Venus is a peculiar centrality measure that loves you to find bottlenecks in fact there is a very strong correlation between us and more in fact it can define between bottlenecks through between this right.

So really high between this means that that not necessarily that all the other centrality measures will yield a direct analogue for example degree centrality does not necessarily mean it is a bottleneck right but between the centrality means it is a bottleneck high between the centrality means it is important and so essentially what you do is you run your between centrality algorithms on the graph and then find the states with high between the centrality and then call them as your bottleneck states once you find bottleneck states.

Once you have ways of finding options defining options you need I π and β so the bottleneck states give you the beta bottleneck states give you the β they will tell you where you want to stop right but which are the appropriate states to start from what is the policy to follow to get there all of those things must be determined right so in the diverse density case they do it very naturally by using experience replay.

So I have so many trajectories from which I have learnt figured out which are the bottleneck states right I just replay those trajectories and learn the cue functions for going to the option policy for going to those bottlenecks States right and in fact were so we did some we did a fun project in the social networks course when I was teaching where we looked at random options which we call small world options essentially we inserted options at random into the world but the expected length of transition of the options followed a certain probability distribution okay.

Which is what was needed to convert the MDP into a small world mdp small world is essentially have you reach from any node to any other node in a small number of hops especially login hops right so you want to reach from any point to any other point so n is the size of the graph I want to be able to go from any point or any other point on the graph in log n hops actually log n squared hops login into log in hops right so that is a short this is a small number in comparison with n right so that is called a small world graph and there are results that say that if you manage to insert edges into the graph that have a certain length distribution you can convert them at a small world class.

So essentially we just threw in small world options and then showed that that cuts down the exploration time significantly right so it cut improves the learning time significantly so for these you do not really need any prior experience or anything you do not need to have even solve the problem right all you need is some random trajectories and any number of trajectories random trajectories will give us we can define useful options out of it.

So in fact we have a data equivalence test so I give you a certain number of experiments that you can do okay with that data that you gather you can compare against the between this kind of a method right and also a graph partitioning kind of an approach and see which one makes better use of the data it turns out the small world options makes best use of the data that is arguing you better things so.

So that is a completely of the completely off the wall think for trying to discover this of course I am not sure how well it will do in transfer learning but it should do in well in terms of learning because it only worries about the dynamics of the problem it has nothing to do with the reward functions right so the small world option so it should do well in transfer learning it certainly cuts down on exploration and certainly has a high reusability because it just connects different points in the world right so these are these are some of the approaches that people have out there so I will stop here so this is all about hierarchies.

IIT Madras Production

Funded by