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

Lecture – 17
Informed Search: Analysis of A*Algorithm (Part- 3)

(Refer Slide Time: 00:19)

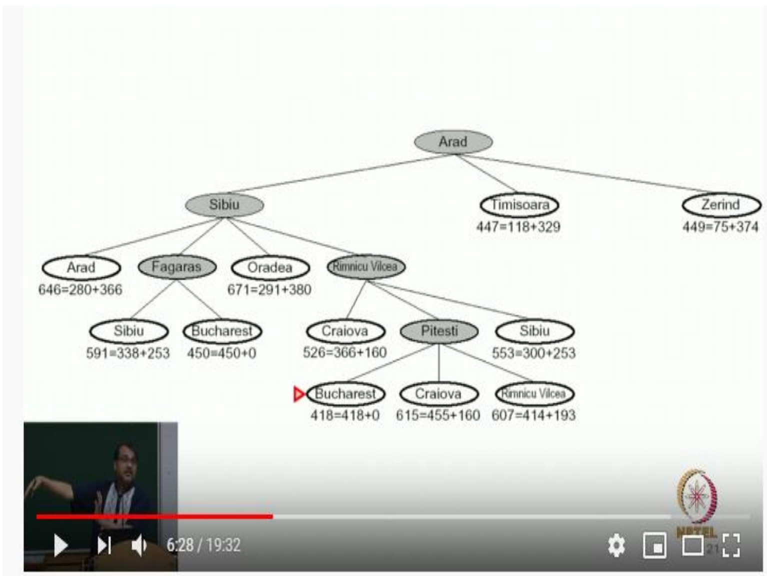
So let us look at a sample run of the A star algorithm. So it starts from Arad.

(Refer Slide Time: 00:28)

And currently its G is G function is 0. I want everybody to answer these simple questions so that you know you get used to the A star algorithm. So G function is 0 because cost from the start state to me, which is start state is 0 and my H function is a straight line distance between Arad and Bucharest which is 366. So initially this is the only node in the fringe, but then when I expand, I will expand to 3 possible nodes in the fringe. The 3 nodes is would be there Zerind, Sibiu and Timisoara everybody with me. This is the first child, second child and third child.

Now we need to figure it out which node to expand further. So let us look at them for Zerind my F function would be 75 + 374 for Sibiu my F function will be 140 + 253 and for the Timisoara my F function would be 118 + 329.

(Refer Slide Time: 01:45)



And therefore I am going to expand Sibiu because it is the shortest. It is 393 it basically says even though the current price that I am giving is high the expected estimated distance further down is low. And so let us try to go there. Okay so far so good so expand Sibiu when I expand Sibiu of course I also get Arad lot back because I can always go back to Arad but notice now the F cost has become 140 + 140, 280 plus the original 366 which was the heuristic distance.

So it has become fairly high. So hopefully the algorithm will figure it out. There is no point going from Arad to Sibiu and then coming back to Arad then following up from there and it will not get to it will not be picked in the fringe. And even if it gets big, it not be big many, many times right. So now in my fringe I have 6 States 6 nodes Zerind, Timisoara. Oradea, Fagaras, Arad and Rimnicu Vilcea. You can see I have done my homework. See this will be very helpful to you when you go to the Romania you should plan your vacation in Romania.

Now that you are so familiar with its map and I am willing to pick the node, which looks the best for now, right? It could be Timisoara, or it could be a so which is the child of Arad or it could be one of the children of Sibiu. And in this case, it is another children of Sibiu. So I expand Rimnicu Vilcea and I get to these 3 States, 3 cities, and notice that the best is now Pitesti no Fagaras. So at this point, what is my algorithm saying? You are just saying maybe the best path is Pitesti, but for now it seems like the best path might be through Fagaras.

So let me first expand it and double check that. Is there one or is there not one? So I expand Fagaras, so I get to these 2. Now as soon as I hit Bucharest, am I done? It is important so what I found is have I found a path to Bucharest very good. I have found a path to Bucharest and the cost to the path to Bucharest is 450 however there are nodes in the fringe? Whose F value is less than 450 what does that tell you?

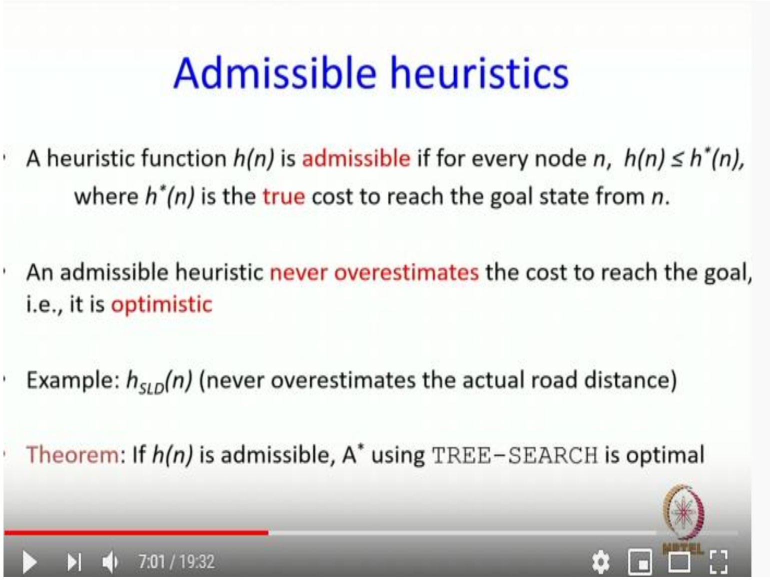
They may be another path through those nodes, which may have lower total cost. So let us not stop right here. Let us keep going. So then I expand Pitesti and once I expand Pitesti I get to another node of Bucharest. At the time I remove the goal from my fringe that is the time I know that I have found her the best path. And again, well prove some of these things. Why best in what sense?

So when I enabled Bucharest, that is when I found the path that the algorithm returns and the algorithm returns Arad, Sibiu, Rimnicu Vilcea, Pitesti and Bucharest as my final path, which is if you go back to the map Arad to Sibiu to Rimnicu Vilcea to Pitesti to Bucharest. And the alternative path we have found is Arad to Sibiu to Fagaras to Bucharest. But even though it has fewer numbers of cities, the distances are high and so this is not the path that we return. Everybody with me on A star functioning any questions?

So now comes the bigger question and the bigger question is, is it going to be optimal? Okay and here I am not going to ask you for intuitions because there is a theory behind it. And it depends on how good the heuristic function is. It is possible that my heuristic functions so confused you that you ended up finding a path, which is not the optimal, the optimal path was here, but the heuristic function was so bad that I never even expanded some many of those steps it is possible. In fact, I urge you to construct such an example it will be very easy to construct good.

So now the question is when would this be optimal and what do we know theoretically about it? And for that I need to define some notions of goodness of heuristic functions. And I am going to define two notions.

(Refer Slide Time: 06:59)



The first notion is quite admissibility. A heuristic function at hn is admissible. And I have written many words here but first thing I want you to learn is that a heuristic function is admissible if it is optimistic. Okay I want you to first learn this word. And what is optimistic? What is most optimistic? What is most optimistic? I am goal what will be the heights of optimism for the Node, the node will say no not need to search for the I am goal enough I am god, right?

So the equivalent of that I am goal. In other words, h will be equal to come on h will be equal to 0 if I am goal then distance to myself is 0. So the most optimistic heuristic would be 0. But however, what would be the meaning of optimism? The meaning of optimism would be that the cost to the goal is estimated as less than what it actually is. You feel that I will have to pay less cost. Then you actually end up paying. That is the meaning of admissibility.

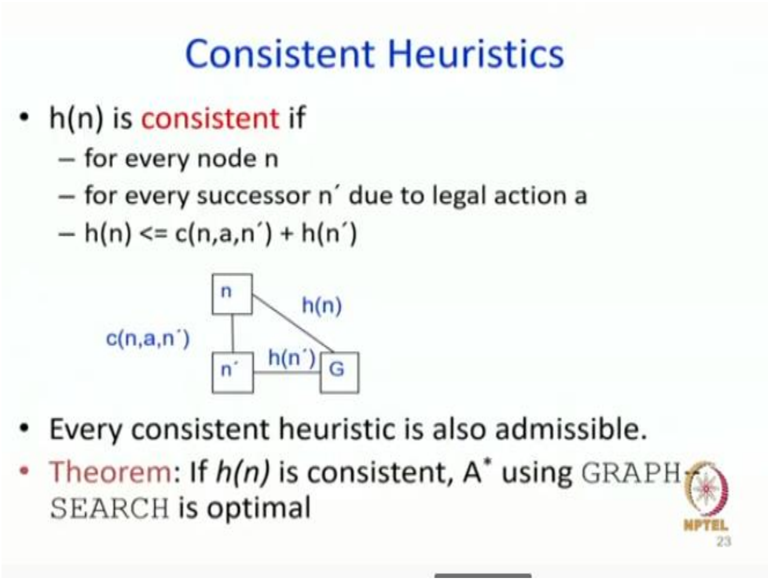
If however, you are in a place where you want to make money and higher is better, what is an admissible heuristic one which makes you believe that you will earn more, not less. So any day if somebody asks you is us a function means it is always less than the optimal, say no. Admissible heuristic is not always that is less than optimal. It is less than optimal for or less than equal to optimal for minimization problems, but it is greater than equal to optimal for maximization problem.

Everybody with me so far so good but because we are in the minimization setting an heuristic function hn is admissible if for every node n, hn is less than equal to the optimal true cost to reach a goal state from n that is an admissible heuristic never overestimates the cost to reach the goal ie it is optimistic and now you can ask the question, is this straight line distance admissible? because the shortest flight you can have is the curves flight.

After that you will end up diverging from the state line and you will end up paying more. So therefore the straight line distance is actually an admissible heuristic and moreover and there is a theorem and you have to do all proofs. But I will discuss one of them only if hn is admissible, then the tree search version of A star is optimal. Okay, we will prove it soon but the interesting thing is that if its the graph search version, then admissibility is not sufficient.

It is a necessary condition but not a sufficient condition. And so we need to define a new condition to figure it out. When would the graph search version of A star the optimal and that condition is called consistency okay.

(Refer Slide Time: 10:43)

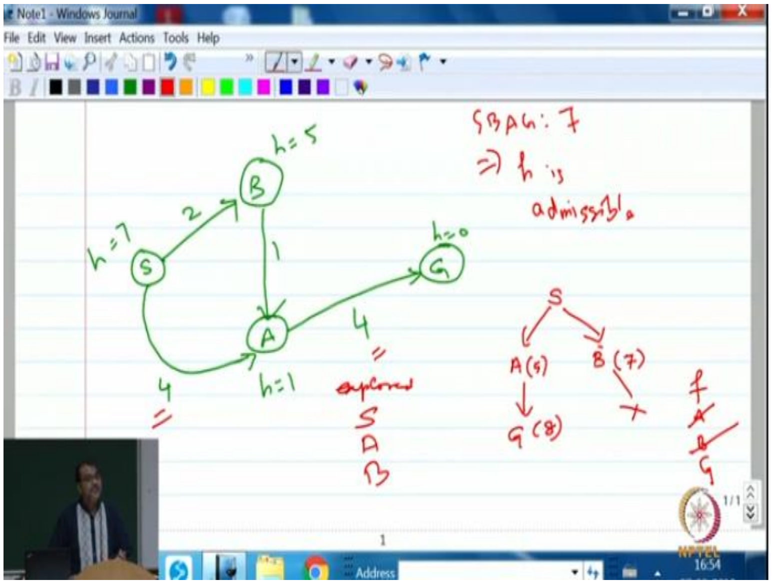


An hn is consistent if for every node n and for every successor n prime due to some action the triangular inequality is satisfied, which basically means that the estimated costs from end to the goal is going to be less than equal to the actual costs that you pay right now when going from n to n prime and the estimated cost from n prime to goal okay and in fact you can prove, and I would let you do this as homework, you can do it by induction or whatever, that every consistent heuristic is also admissible.

And now you can also prove that if heuristic is consistent, then A star using graph search is optimal okay. So there are 3 things to learn here okay and understand one when is tree search of A star optimal when the heuristic is admissible, when is graph search version of A star optimal when heuristic is consistent. But believe it or not, consistency subsumes admissibility. So, if you are consistent you are any way admissible and moreover, we need to get some intuitions as to why this is the case.

Okay I will do some intuitions there are more, but we will not study all of them. So first I want to do one simple example of what happens if you have an admissible heuristic, but we are doing graph search version of A star. This example will help clear your concepts on graph search version of A star so let us do that.

(Refer Slide Time: 12:28)



And we will do this with this example. So let us say my objective is to go from start state to goal and I can go to A or B, I can go from B to A and then I can go from A to goal. It is a relatively simple example and the h costs are mentioned 2, 4, 1 and 4 and the heuristics are mentioned on the nodes h = 7 for state S, h=5 for state B h=1 for state A and of course 0 for goal. The first thing we need to check is whether h is admissible.

What is the shortest path? By the way, what is the shortest path? We are looking for SBAG. So SBAG is the shortest path we are looking for. And what is the cost for that? 7 and now is this heuristic admissible? So what is the shortest path from A to G? What is the cost? 4 and what is the heuristic 1 is less than 4 good what is the shortest path from B cost 5 what is the heuristic 5 less than equal to the optimal heuristic is admissible. The best cost from state S is 7 and what is the heuristic 7.

So therefore we can prove that h is, or we can verify that h is admissible. Now what we are going to do is we are going to run the A star algorithm on this particular example just to show that it messes up tree search version of A star okay? So let us do that so we will start from the start state and when we expand it, we can go to a or we can go to B and when we go to A what is the F value? 5, 4 +1 and what is the F value for B? 7 what is in my fringe right now? A and B and which one I am going to expand? A now notice that this is the graph search version.

So when I expanded S what did I put in my explore list? I already put S in my explore list, right? And my open list or my fringe currently is A and B. Now when I remove A then I remove it from the fringe, and I added to the explore list. And what are the states where can I go to from A, I can go to G. what will be the F value of G? This is careful be careful. What would be the F value of G 8 4 + 4 + 0. So my F value of G would be 8 so I will add G into the fringe.

Now what is in my fringe? B and G is in my fringe, right? I do not stop until I move goal. So B and G is in my fringe. Now comes the key step then I expand B I will get to A and when I go to A its already in explored list because it is already an explored list. Am I going to add it in the Fringe no.? So basically, I just remove B and expansion leads to nothing. And I added B to explore list.

So what just happened? I lost my optimal path, which was SBAG in this process. And why did I lose it? Because earlier I have already expanded A and why did I expand A well, I expanded it because it was looking too close to goal. But now that I have expanded it, there is no way to reserve it a newfound path to the node that was already expanded new found path to A.

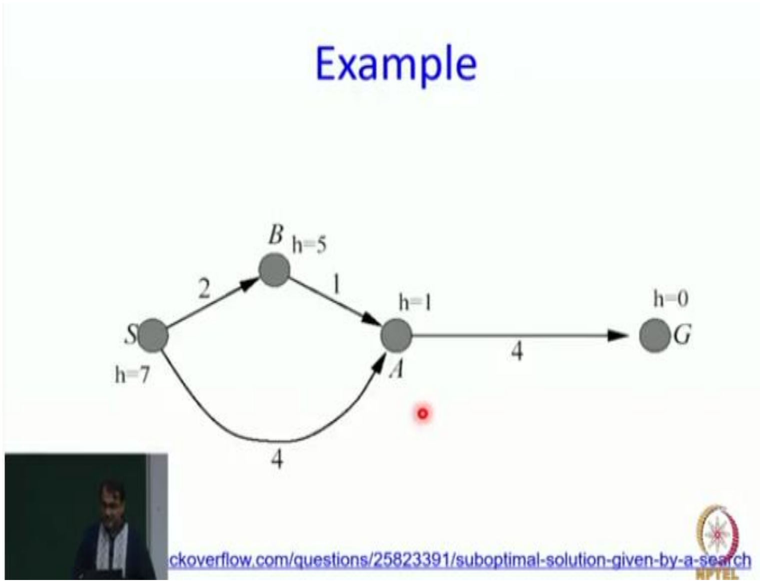
So in order for this to work, I need to somehow prove that no new, shorter path to this node can be found when Im moving it from fringe and adding it to explore it. And this can happen if your heuristic is consistent, but this does not always happen if your heuristic is admissible. Okay? So there is a proof in the book, which shows that the graph search version of A star is optimal if you expand using a consistent heuristics.

But the goal, the proof says that I will in one path Fs are always in non-decreasing order. And that is sort of the fundamental basis of that which I would encourage you to read. But this example shows you that the algorithm can be fooled when I put something in an explode list and a later better path is found, which can happen if your heuristic is only admissible but not consistent.

“Professor - student conversation starts” Yes question say that again? Why is SAG 8 why is it 8? Because S to A is 4 and A to G is 4 so that is why it is 8 see this is 4 and this is 4 from starting to end heuristic. No this is not heuristic this is F, G function h of the goal is going to be 0. Does that make sense? This is important why would F value for goal be 8 because F value is always g+h g is the actual cost to the goal or to the node and h is the estimated cost in the future, estimated cost in the future is going to be 0.

The actual cost is going to be 8 because I am following this h, and this is h okay. Good so that was one example "**Professor - student conversation ends"**.

(Refer Slide Time: 19:09)



This example is from stack overflow. That is a nice example I like it we can stop here. Thank you.