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

Lecture – 11
Uninformed Search: Search Problem and Examples (Part – 2)

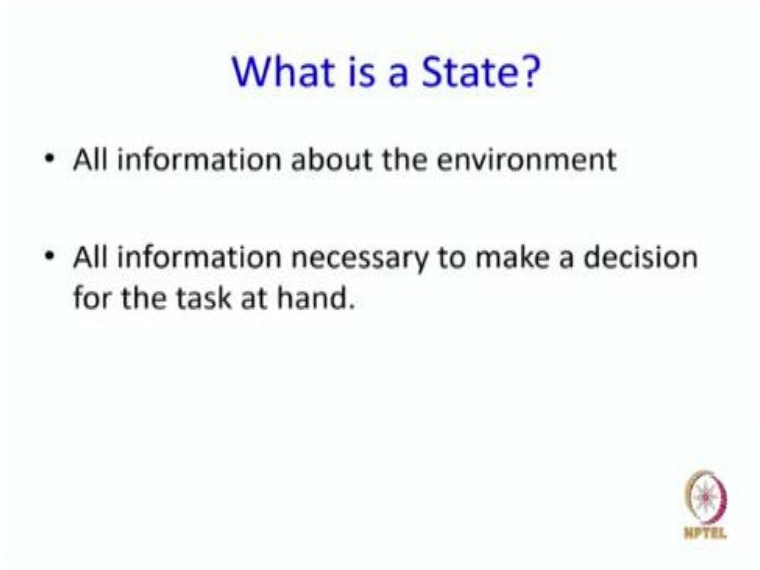
So, we started with the goal that we want to really achieve or solve every problem in the world because this is the admirable goal, it is a lofty goal as we discussed that a researchers like lofty goals so, that is one of their goals and so we wanted to formalise some representation for pretty much every problem in the world, right and of course, we will not get there but we will get to a representation that solves many problems in the world of course, right that represents many problems in the world.

In the context of that, we started out by talking about what is the state; because we will have a state centric view in all of AI, you would find that state would be central, what goes into the state will become quite important. After the class somebody came in and asked, how does human always figure out the state and my answer to the student was that for now, for the purpose of our AI class, a human is always going to figure out the state.

Now, if you are given some data and if you run some kind of a clustering mechanism later some advanced level courses, you might get to a place where you can later claimed that one cluster is one state and then I am going to do all my downstream AI algorithms on the states pays at an automatically figured out from clustering over this data, it is possible I have seen this done but it is relatively rare, okay.

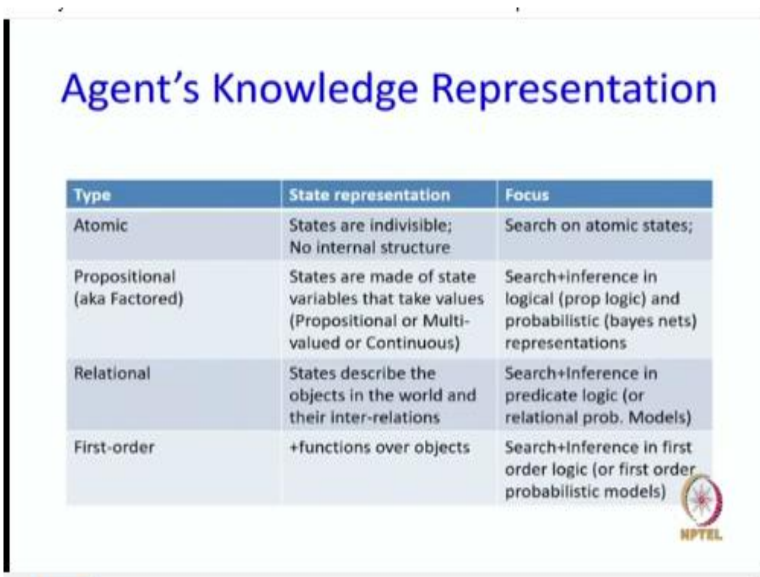
So, for most parts we would assume that a human who has the problem in mind, the domain designer; we can use the word domain designer for this, the domain designer designs the state space for us and the AI folks are just interested in solving, okay.

(Refer Slide Time: 02:03)



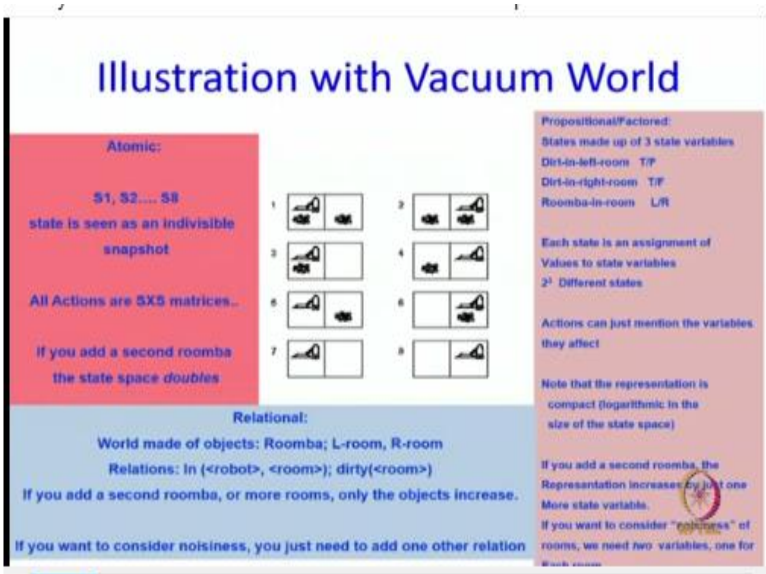
Then, we discussed that how do I represent a state, right in the world, in the AI's world, in the AI's brain, how do I give the state space to the AI agent.

(Refer Slide Time: 02:13)



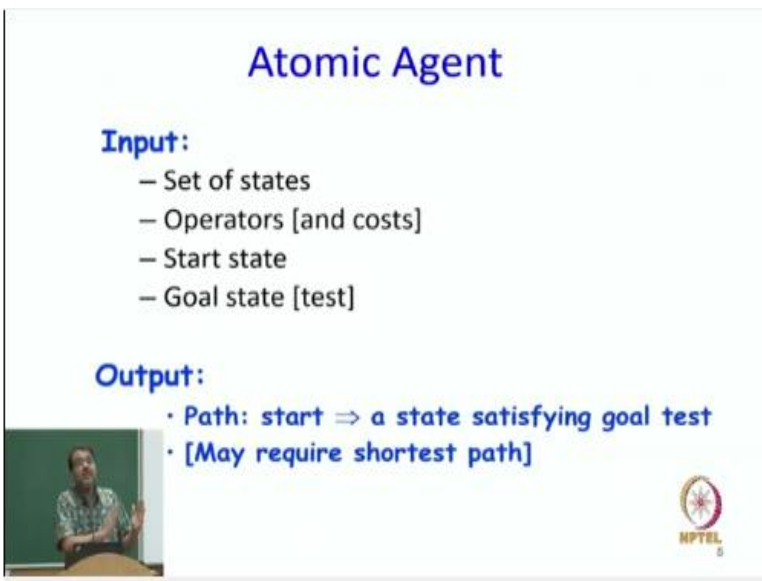
And we discussed that there are many different ways of doing this, one is atomic, where each state is the most indivisible unit in my space and let us state is a number and see you cannot really understand what is happening internally inside the state, you just say that each state, each specific configuration of physical objects or my problems setting is a number, right an unique number.

(Refer Slide Time: 02:45)



And then we also briefly talked about propositional and relational and we hear the example of the vacuum world, so this is where we were in the last class and the next thing that remains to be done is that we need to define a representation that the domain designer would give an AI agent and say look, this is my problem, please solve it for me, okay. So, of course what the domain designer is going to say is that I have a set of states.

(Refer Slide Time: 03:14)



But the interesting thing is that this set of states maybe there in domain designer's mind but here is not going to enumerate all the states forming, okay and this is going to be the fundamental difference between a lot of AI problems and a lot of other sort of graph theory problems and we will come back to that. So, let us say this domain designer writes a code for me where he says, look I will give you a black box.

I will give you 2 black boxes or 3 black boxes, right what are these black boxes? So, the first black boxes that if you ask me I can, if you ask me whether you are in as goal state, then I will give you yes or no, this is my black box 1 and we are going to call it the goal test, okay. So, notice that we are saying goal test, so if you say I have reached this state, is it a goal, is it a goal?

So, the domain designed black boxes is going to say yes or no, okay that is one component of the problem that the domain designer is going to give me. The second component is that if you are at a state and you tell me that I am at the state tell me, where all I can reach from this state in one step then I will; I as a domain designer will tell you okay from this state S1, you can reach S5 by taking action or operator 3 and S9 by taking operator 5 and so on and so forth.

So, at the internally; internally what is going on is that there is a set of states, there is a set of actions or operators and then if I am in at a certain state, if I take an action, I get to a new state, this is my graph structure internally, so think about it, fair enough, and I will be at a starting state and I want to reach a goal state, okay and my starting state will be unique more often than not there will be only one starting state that I am in.

My goal state; there can be many, many goal states, right so, if my goal state is at Vishwajeet should be sitting on this chair, then where Ansh is sitting does not matter, so there may be many, many goal states where Vishwajeet is sitting on this chair but Ansh can be sitting on many, any chair and in that way, I may have many goal states, so you cannot use the word the goal state, we have to use the word, a goal state, this is very important.

When you write, if you; when you read AI papers, if they have been careful they would not say, the goal unless there was exactly 1 goal in their problem, they will say, a goal because there can be many goals and this is what AI folks sort of thing from the perspective. So, again, so internally in domain designer's mind, the domain designer has a graph, there are states in the graph, think of each state as a node, intuitively all the differences, graph nodes.

And then there will be edges, think of them as operators, actions that I can apply in a given state, think of it as a directed graph, if I apply an action in a particular state, I get to the next state, then is that graph but this graph is possibly so huge that nobody is going to write it out for you and that is very, very important, this is actually fundamental in here because the problems are at least may be hard as we discussed last time.

So, more often than not, the problems that we will be dealing with will be very challenging in many ways and one of the ways in which they can be challenging is; I cannot even write the whole problem in as a data structure, so I will write it as a code, some kind of a black box where I will give the AI person 3 black boxes, one is an expand function. This expand function says if you are in a state, I will tell you what are the next states you can visit in one step.

One would be the goal test, if you give me a state, I can tell you what whether you are goal or not and last but not the least, I will tell you what initial state, you need to start planning from. Now, with these 3 functions, I have defined an atomic agent for you and the input would be exactly what I said these 3 particular functions and what the AI system is expected to achieve is a path from the start state to a states satisfying the goal test right.

And if my actions have costs, I may be interested in the cheapest path, if my actions do not have cost, I may be interested in the shortest path, right, everybody with me on this. Now, I started with a lofty goal that you know I will define a problem where all; I will define a representation of all problems in the world can be modelled, does it model all problems, I will just quickly ask that question, would it model all kinds of problems in the world?

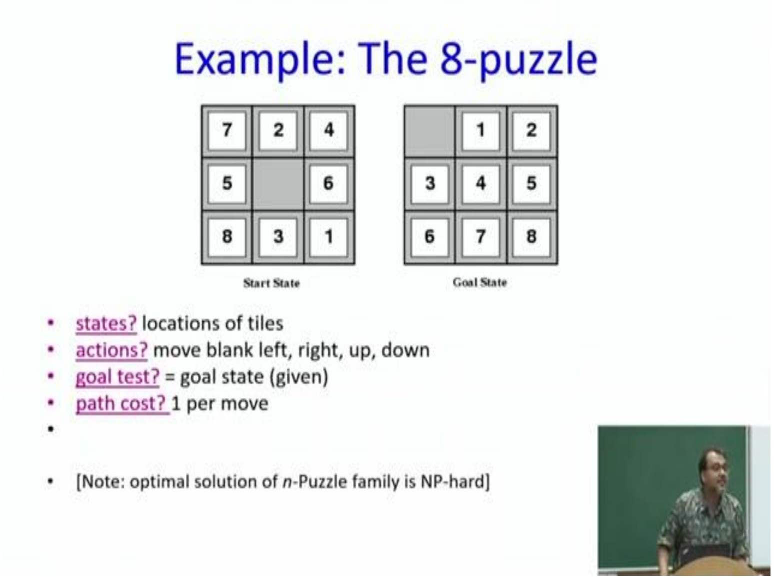
Okay, you guys have a little lost, so the answer is no, right and the reason it does not model all problems in the world is a there is no notion of uncertainty in this representation, if I take an action in a state, I have always reach the one deterministic state which is what the current formulation has, right, we will relax that, so we will reach a different kind of an agent where a stochastic actions could happen in the future or actions could have stochastic effects basically.

And alternatively, this is a single agent world of the; single agent view of the world, I am one agent, I am taking one action at a time and that is the only thing that is changing the world, it is possible that there is an adversary that I am playing against, I am playing one move, the adversary is making the other move and my goal is defeat the adversary, the adversary move is to defeat me.

Now, that kind of a setting is not naturally modelled in this representation, but we will study that too, okay. So, in some ways we started with a goal of modelling every problem in the world but what we have modelled more or less is a deterministic single agent problem, where the agent is the only one making the change to the environment because if the environment automatically changes, then I will have stochastic effects of this thing, right, does it make sense?

If does not, now this will over time, make more and more sense as we go on in the course, alright but in spite of this, the claim is that at least for this subset of problems, this will be able to formulate a lot of problems in the world, okay, a lot of problems. So, let us look at some such examples.

(Refer Slide Time: 10:27)



So, one such example is the 8 puzzle problem, everybody knows the 8 puzzle problem; I have 8 tiles in a 3 cross 3 grid, I am supposed to move these tiles such that I start from any given start state and I finally reach this particular goal configuration where all of them are number 1, 2, 3, 4, 5, 6, 7, 8, many of you may have played 16 or 25 puzzle, when you are kids. Now, a question for you; can I formulate it as a search problem?

Now, when we use the word formulated or model it, you should not think about how to solve it and this is the fundamental AI way of thinking, there is modelling and then there is algorithm, remember we discussed, so we are not talking about modelling; modelling means can I represented into a set of states, set of actions and goal test and a start state such that I will be able to give the inputs that my atomic agent needs.

And the atomic agent may have some algorithms, which are general terms and may be able to apply those general algorithms, so when we think of modelling a problem as a AI problem, we do not think about what solution we will use, we think about modelling, it is in one of the many ways to model a problem and then for that particular formulation, there are already some algorithm that have been discussed and those algorithms can be applied, okay.

So, now here is a question for you; how would you formulate a 8 puzzle problem as a search problem; atomic agent search problem, what would be the set of states, you will have to be louder, I cannot hear you, each possible configuration of the board, what is your name; Ansh okay. So, Ansh says each possible configuration of the puzzle would be my state, right, the locations have exact all these tiles like 7, 2, 4, 5, gap, 6, 8, 3, 1 is a state.

What would be my actions; yes, somebody at the back, yes what is your name; Abbura, yes Abbura, shifting one of the tiles, how many actions at most might I have; 4. How many actions at the least might I have; 2, right and instead of shifting a tile let us say, you are going to shift the gap, it will be easier to understand it, right. So, we will say that I will move gap right, I will move gap left, I will move gap to the top, I will move gap to the bottom.

What will be my gaol test? Somebody on this side, what will be my goal test, it is not very hard, yes, the final configuration that we have right, how many goal states do I have? Only 1 goal state, so if this is the problem where we can say the goal, right and is there is a notion of a path cost or the action cost, so what can we say is the path cost; the number of steps, okay. So, what we have done in this process is that we have converted our problem of going from any starting configuration to the final test; goal test.

As a search problem, we have defined a set of states, we have defined a set of actions if you start thinking about how many possible states to we have even for the small problem, what would you say, how many possible states to we have? 9 factorial, right, we will have 9 factorial states because we have 9 tiles which can be arranged in; we have arranged them in a specific sequence, gap can be thought of as a tile if you want.

So, for 3 cross 3 9 factorial, maybe we can enumerate the whole states phase but for 4 cross 4 16 factorial and then for 5 cross 5 25 factorial, we will not be able to enumerate definitely for 5 puzzle, we will not be able to enumerate the whole states phase, we will not able to give you a graph, where you can run your dijkstra's algorithm on it. So, always remember you have studied the shortest path problem, this is no different from the shortest path problem the way we have defined it.

However, the fundamental difference between the shortest path problem that you have studied and what we are going to study in AI, is that my graph would be too big that I will never be able to store in memory, so therefore I will give you the graph in some implicit form. The implicit form here is that I am giving you a successor function or expand function basically and I am giving you a goal test and I am giving you the start state.

This is my way to let you build the graph, if you want to build the graph but of course, if you build the full graph, your memory will; it will go out of memory and you will never be able to solve the problem, okay. So, this is what we will keep in mind as we study the next few weeks of lectures that you know the problems are not really hard, we understand them; dijkstra's, bellman ford, whatever we have our favourite algorithms for shortest path.

But I cannot even give you the graph, so if I cannot give you the graph intuitively, what might you do, what will you do? I have given you a state space; starting state, so I have given you the 7245 gap 6831 configuration, **"Professor – student conversation starts"** I have given you can expand function what could you do, who said randomly; yes, what is your name, Vishwajeet, yes of course, so Vishwajeet says, randomly it take actions, so just do a random walk in the space, do not think about where you came from, where you want to go, just keep searching randomly, right that in one alternative.

That is nothing wrong with it per se and we will come back to that particular alternative when we go to the local search class but for now, there are different alternatives they were interested in studying. So, what could be another alternative; yes, what is your name; give each state space a score and then what do you do with the score; how do we move closer is the question, see I am asking more fundamental question.

I am only giving you the start states, so what do I do next; yes, we move to the neighbouring states, what is your name; Chinmai, Chinmai says move to the neighbouring state but which neighbouring state do we move to; do we to move or can we just do the thinking, see look we are building in AI system; the AI system is like thinking, we do not have to play, we do not have to play right now, we are just thinking in our mind, what should be the best action to do.

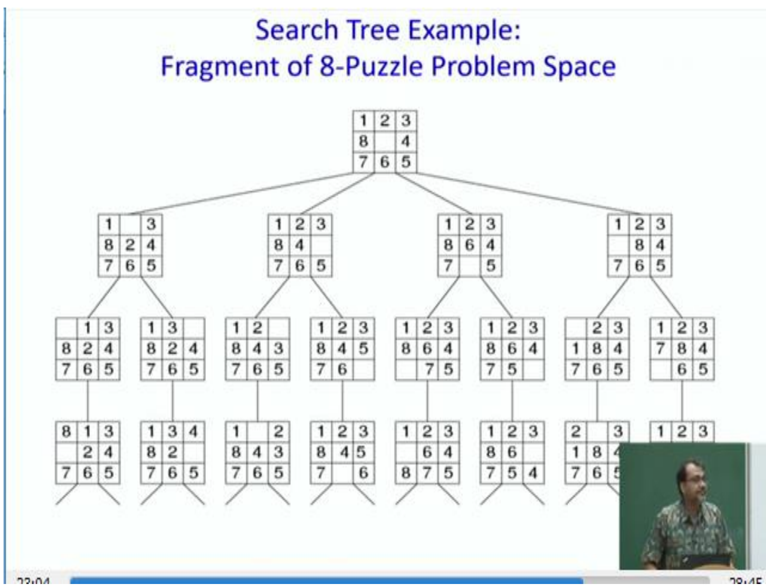
So, it is not like we have to take this move in the real physical environment, it is like we can do the planning in our brain and then decide what to do, we allow that so, we are in the thinking phase, not the acting phase, yes what is your name, Sukradhi yes, very good, you are all running slightly ahead of me but that is good, Sukradhi says let us define some heuristic and that heuristic tells me what is a good state and we will come back to that towards the end of this class, this lecture slides.

"Professor – student conversation ends" So, we are currently in the world that we have defined, so our world is that there are a set of states, set of operates, start state and a goal test, I am not giving you anything else, so notice I am not giving you anything else, this is my current version and that is why because I do not have any idea which is a better state, which is a worse state these problems are called uninformed search problems.

We are in an uninformed search problem, we have no other information and so therefore, we have to do whatever we have right now, we have to do the best we can with the information that has been provided to us. I am asking you a more, simpler question by the way, I am at a start state, what could I do, what will be the first step, I will have to do? Find the moves, what is your name, Akshay, okay, find physical moves, right, Akshay says find physical moves.

How do I do that? My domain designer has given me something, what is the domain designer given me; the expand function so, I gave my domain designer the starting state and say, please expand and when the domain designer expands or that the black box expands, what do I get; a set of states which I could reach in one step for example, in this configuration 7245, gap 6831, if I move gap to top, I should get 7 gap 4526831.

(Refer Slide Time: 20:12)



And if you keep doing this, this leads us to a search space, okay this is fairly; should be fairly intuitive to you, this was my starting state and I took the move gap to the top action then I got to this state strict and this I can do with the expand function, I got to right, I got to this state and so I now got 4 new states. Now, I can technically, recursively apply the expand function and keep expanding my search space.

Now, notice that this is different from dijkstra's algorithm not exactly but at least we are only looking at those states which we could reach from the start state, so we are not even interested in states, we may not have a be able to reach from start state or those which are very far away from start state which we do not need to reach to the goal. So, therefore we will end up expanding more often than not far fewer states than if I was giving you the full graph.

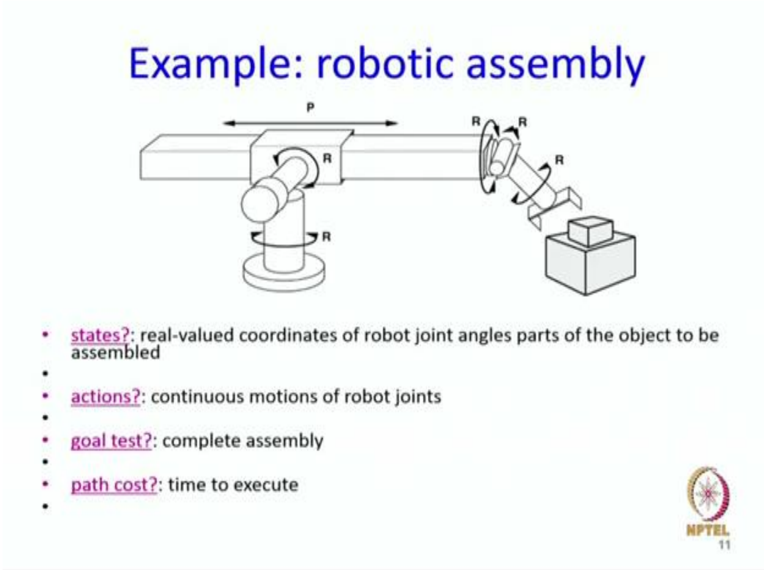
So, that is the first intuition that you have to look at, **"Professor – student conversation starts"** yes, Sukradhi, no, no, no we are getting exactly there, so the question was is it any different from the depth first search no, this is the algorithm I want you to come up with, it is not surprising in hindsight, you will say oh, I though AI was cool, it is just depth first search well, it will get cooler slowly but look, we have modelled things in a certain way.

Why have you modelled in a certain way? So that we can be general now, we pay the cost of generality, we basically expand this state space graph and we call it the search; search tree, by the way it is a search tree or a search graph, this is a good question. **"Professor – student conversation ends."** So, can states be repeated; for example, if I move the gap up and then move the gap down, will I get the same state, yes.

And now the question for you is; what do I do about the repeated states and I will come back to it but you keep thinking about it at the back of your mind, so we can have a search tree or we can have a search graph but to reach to the search graph, we have to do what is called the duplicate detection of course, because we have only giving it to the expand function, the expand function is giving us new states, we have to somehow say do have I seen this state or have I not seen this state.

And if I have to do full duplicate detection, it is possible that it will take a lot of memory anyway, so therefore in AI, at least for now, we are going to study the search tree version of it. In the search tree version, any search state could; 2 search states could represent the same words state, right, this graph does not show it but if I do a gap down action here, I will get to the same starting state and for now, we will allow it for now, okay.

(Refer Slide Time: 23:21)

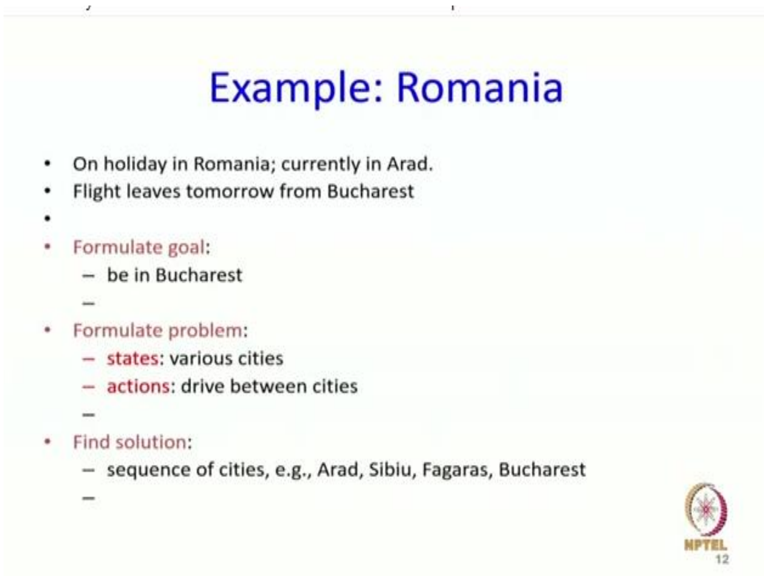


So, our goal is to make sure that this is a general enough representation, so let us take another kind of problem. This is a robotic assembly problem, I want to complete the assembly like think about an IKEA store or a store where you get all these parts and you have to assemble the table or desk at home you know, in India we do not have to do it, we have people to do it but invest in western countries, I have assembled my own table and my own TV and so on and so forth.

No, no not the TV in terms of opening up the TV but putting it on the stand and all of that, so what happens; they give you an assembly parts and they give you the screws and everything the tools and they give you the step-by-step instructions of how do you assemble, right. So, suppose this was a problem where I gave you all the tools and what will be my state; my state would be the particular pieces and how I have built my particular part.

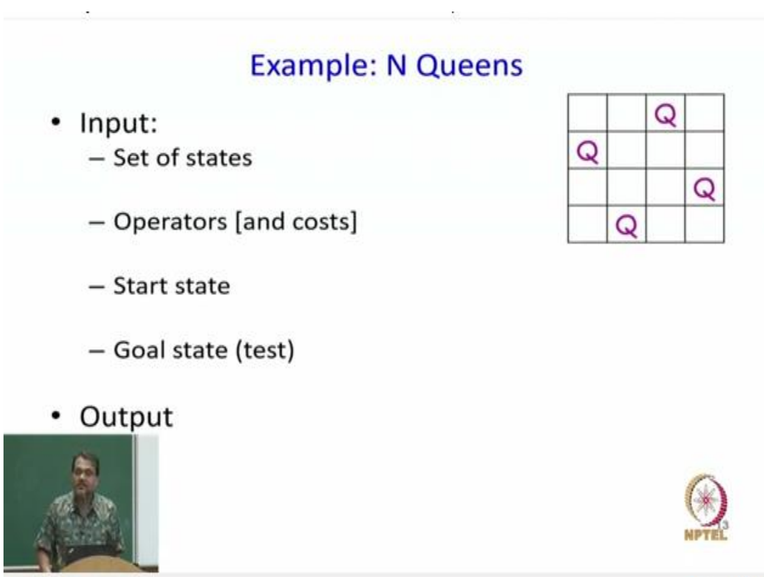
And my actions would be add another part or it could be lower level where the robots will move their joints and based on that the particular piece will go into the right place and my goal test is to check whether I have gotten the assembly completed and correctly, alright.

(Refer Slide Time: 24:39)



Or any kind of a shortest path problem like going from one city to another city, the book uses the example of going from Arad to Bucharest in Romania and there you can always think of it as a search problem where the starting state is that you are in Arad, all the states or various cities actions are drive between cities, goal is to be in Bucharest and you have to find a solution, a sequence of cities such that your total time is minimised, your total driving distance is minimised.

(Refer Slide Time: 25:12)



Now, these are easier ones but here is another fun one, okay, a slightly unusual one, so let us say I give you the N Queen's problem, do you know the N Queen's problem; the N Queen's problem is that you have to place N queens on the chess board; N cross N chess board such that the N

Queens, no 2 queens should be attacking each other and you know that a queens attack each other horizontally, vertically and diagonally.

Is this a solution to the 4 Queen's problem; yes, neither these 3 attack nor these 3 attack, nor these 3 attack and so on so forth, so nobody attacks anybody, so they are happy living in the same chess board, right. Now, if I have to solve this problem, can I still formulated as an atomic agent search problem, what will be my set of states, what will be my set of actions; **"Professor – student conversation starts",** we will place queens one by one, what is your name, Paul.

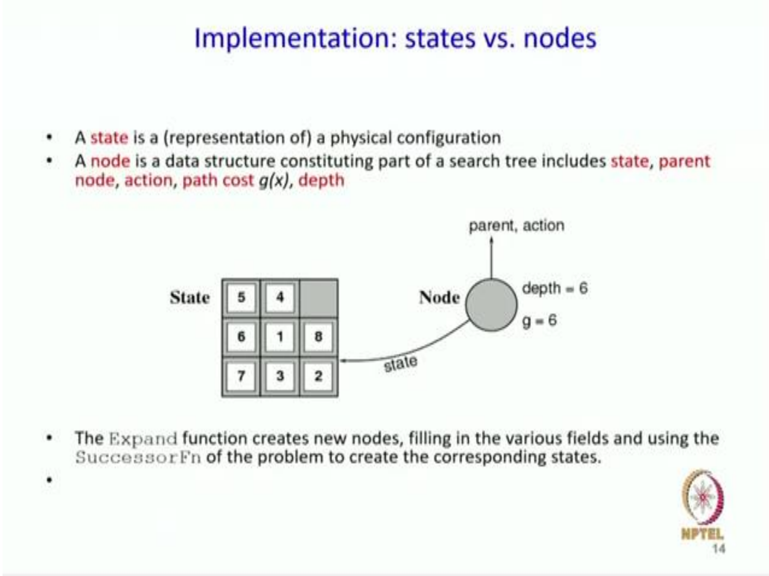
So, Paul says we will place queens one by one, so my starting state would be empty board, my starting state would be empty board, my set of states would be; not number of queens, see if you say number of queens in the board when you only have up to 4 states that is not enough but what would it be; it will be a partial configuration of the board, right and you can be clever, you can say first I will place the queen in the first column, next I will place the queen in the second column, it does not matter.

Because order does not matter, it does not matter, whether I place queen 1 first and queen 3 later and queen 3 first or queen 1 later, it does not matter, so we can become smarter in defining our search states but my operator would be adding one more queen, start state would be empty board and the goal test would be all queens placed and no queen; no 2 queens attacking each other, very good. **"Professor – student conversation ends."**

So, now you can start to think that oh, we can model many problems in the search representation and the book has many, many more, right so you should check the book for more examples of this, you should even try it yourself. So, one of the skills that you should have at the end of the year courses, if I give you a new problem you should be able to model it in one of the various problem representations or formulations we study.

And for now, you should focus on the basic search formulation, the atomic agent search formulation. Now, when we implemented there is a world state and then there is a search node and those are 2 different things.

(Refer Slide Time: 28:03)



So, state is a representation of the physical configuration and node however, is a data structure certain that has the state but also has the parent information, what actions are applicable, what is the cost up to this point, what depth in my in the search tree etc., etc., so actually when you start implementing things, a search node will have many more things that you need to keep track of but that is okay, right, we are not worrying about it for now.