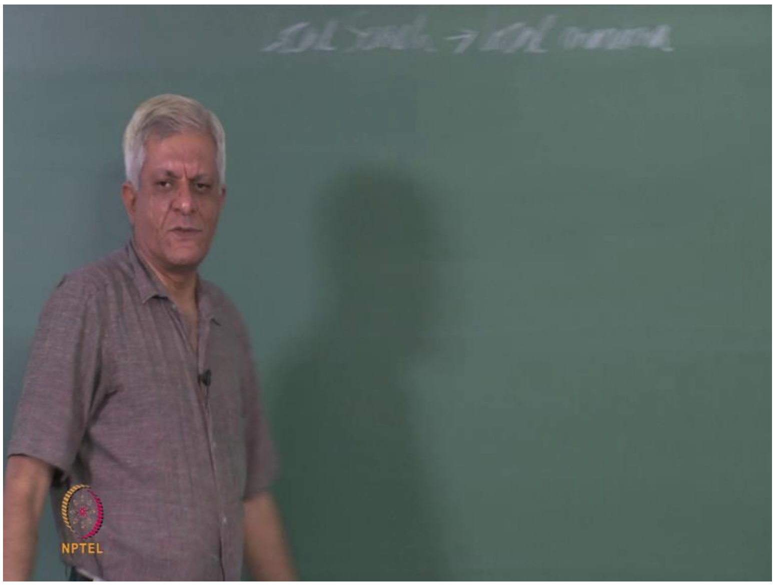
Artificial Intelligence
Prof. Deepak Khemani
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

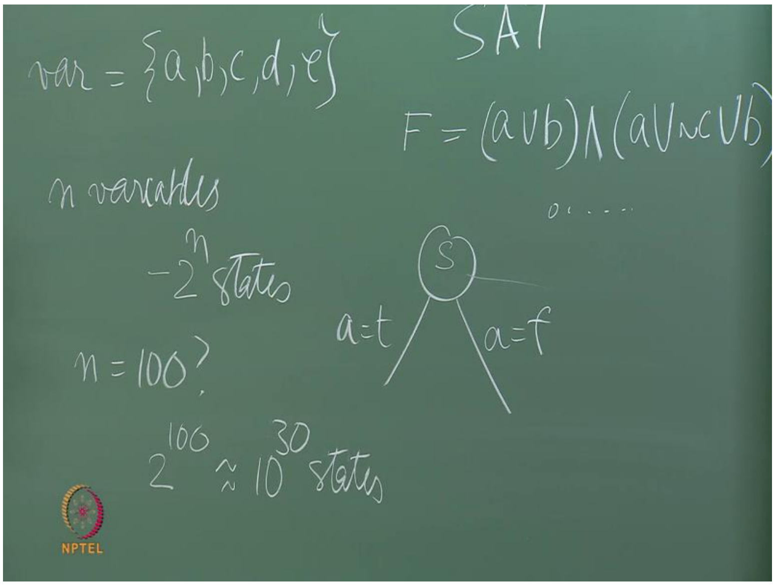
Lecture No - 11
Solution Space Search, Beam , Search

(Refer Slide Time: 00:15)



So, we are looking at this local search and your seen that which leads us to a problem of local maximum because the algorithm does not have the excess to the entire things space. It could get started; it could only see some options and not all options and could get stuck there essentially. So, you want to see how to get around and we saw that that again which this algorithm is out of navigating is determine by the Heuristic function that actualities this determines also by the more than function. We will see in a short life well given set of states it is a move them function will tells you from which state you can go to which other states essentially and we will see that we can play along that as well essentially.

(Refer Slide Time: 01:35)



So, I want now introduce this problem today which well known problem here familiar with no doubt call sat this problem everybody familiar with satisfiability. So, we just over quick recap that there is a formula and you we will assume for simplicity that this formula is in conductive normal form we does not have to be, but it is contempt. So, it is a Boolean formula we can say, so something like a orb and all not all band, so on a formula like that were a, b, c, d, e, all Boolean variables we can take two values either true or false and the semantics of and on or is defined and naught is defined . We want to find valuations for a, b, c and all the variables such that the formula evaluate is true essentially.

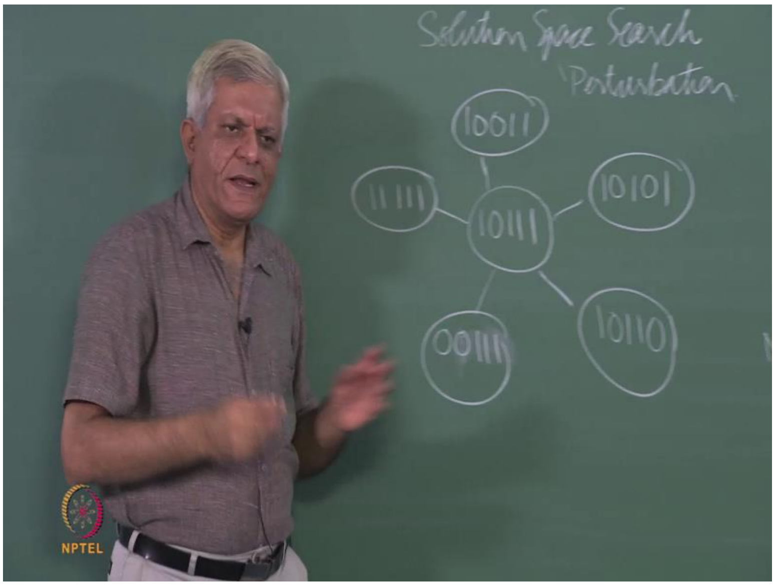
So, that is a satisfiability formula, which says that can you find values for all the Boolean variables such that the overall formula evaluates it true and obviously you can see that this is a kind of search problem you can say something like you start. You say a is equal to true or a is equals to false that is one way of looking at it that you that you pick a variable a, sign a value to it, take another value variable sign a value to it. You would generate a search tree in the process or you can say pick you can have branches for b at the sense you can formulate it was the search problem at various stages. Essentially, we want to do today is a different way of exploring the space, what the space is, so let us say we have variables are five variables.

So, we had some formula which is construct over this 5 variables what is the space the space is for each variable a or b or c or d or e value of true or false. So, 2 raise to 5 possible states in this case if you have n variables, then we have two raise to n states what if n equal to100. So, it is a similar problem that if you have dealing with a large number of variables then this problem could become hard. In fact, you know that sat problem was a first problem which was a first problem which was used by cook to talk about n p complete next essentially where n is equal to 100 we have 2 raise to 100 which is about 10 raised to 30 states.

So, we have a 10 raised to 300 bit number essentially and we you has to see in earlier that we cannot trifled bit numbers like 10 raise to 30, it is it is a huge number, little take a billions of years to search through all these things. Even if you could inspector billions states in a second, we have seen that argument earlier essentially. So, it is a difficult problem to solve, of course you must familiar with the fact that there are variations or special classes of sat problems which for example, talk about the length. So, this is calling a clause and each is call a little, so the number of literals in a clause actually is characteristics take of how hard a problem is.

So, there is a variation called two sat, which means the there is only two little literals for clause we can have many clauses has you want, but only two literals per clause. You can have as many variables as you want, but, if there only two literals per clause then the problem is actually quite easy to solve and some of you must be knowing there well known will go some Davis Putnam method for solving two sat. Essentially, 3 sat is m p complete the 3 movement you go from 2 to 3 that you allow 3 variables per clause, then the problem becomes m p complete essentially.

(Refer Slide Time: 06:36)



So, we want to explore something which we will call a solutions space search which is lot radically different from what we called as state space search, but except for the fact that we will in the search space every note will be a candidate solution essentially. So, for example, further 5 variables my state could be something like this one 0, 1, 1, 1 and it is a candidate solution. This means that I am thinking weather if I put a equal to 1, 1 meaning true and b is equal to 0 and c, d, e and equal to one Boolean formula we true on notice easily. So, this is a candidate solution and we will look at a process of perturbation which means we will take a candidate solution and perturbation to give us a new candidate solution.

It is not other side; well not radically different from what you have doing earlier the earlier my said could be called as constructive mechanism where you assemble a solution piece by piece. So, that small diagram that I have drawn on there is illustrative of that, that first you choose a variable a, then you choose a variable b, then you choose a variable c and so on and so forth. We are doing that here, we are saying that we take all the variables at one shot takes that as a candidates and if it not the solution will do something with this candidate to generate more candidates. Essentially, so in that sense we call is a solution space search.

Again, I would say it is not radically different it is just a different way of looking at things essentially even in a for example, city problem, you could say that if I had gone from IIT Madras to Marina beach the path from here to Adyar is that… That is a solution or not you can think of it like that also as a candidate in some sense essentially, but here we are saying that we have all the values or all the features of variables such we can about. So, another example is a n queens problem, you can have a constructive solution for the problem you can say place the first queen then place the second queen. Then, place a third queen that and you can build a solution step by step all you can say that I will have a candidate placement for all the queens.

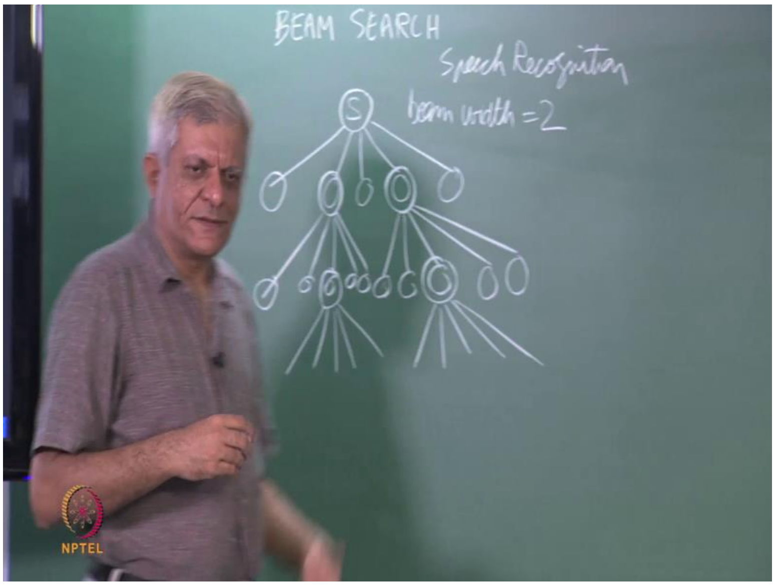
So, for example, in en queens in mike set let say 6 queens I could say this is a candidate 6, 1, 3, 4, 5, 2 and I would interpret this as saying that the that this is the first column this is the second column third column. So, I am saying the first column queen is a 6 row, the second column queen is in the first row the third column queen is in a third row and so on and so forth. This is my candidate solution and I will do something with this may be I will do formulation of some sort to generate other candidates solutions. So, that is the solutions space way of looking at things not radically different, but it sort of helps us to visualize the space a little bit different.

Now, remember that for this problem there are 2 raise to n possible states essentially and what move and function should give us is how to navigate this spaces. So, one way we can divides several move then functions essentially for this one simple, so we will call them as follows I will call this function one, which says change one bit. Essentially, if you look at this notation where they convey it is the big thing, you allowed changing one bits. So, I have tables like 0, 1, 0, 0, 0, sorry 0, 0, 1, 1, 1, so have change the first bit then 1, 1, 1, 1, 1, I change the second bit 1, 0, 0, 1, I came the third bit here, so this is one move than function that I have all her neighborhood function.

So, that is why am use a name n have, it is a neighborhood function a given a candidate solution a this be a neighborhood of the solution which is the same as what we called is a move then the start essentially. So, back to our problem of local maximum why did we get into the silk limbic algorithm because we wanted to save on space essentially? Now, a days space is hardly a problem essentially I am in computer system I remember that in 1990 I have the pc in the department which had 30 megabits of hard disk and other day I was talking to someone and he said this machine was only 2 GB RAM.

Essentially, you know look space is now days is not so much, so, problem you had a heard this is militates famously supposed to has said a some point that who will need more than 64 k of RAM. So, space of course has change essentially, so we should also add adopt to the changing world essentially. So, at a before we come to this till the very simple variation to claiming this why do want to restrict yourself to only one success essentially why not more than one essentially, I will just will illustrate that with a such three as follows.

(Refer Slide Time: 13:27)



It starts with a search the start note, then a next a there are five successes for a every note may be like this. So, you generate five successes fill claiming says move to the best one what is what is, so about moving to only 1 and a move to 2 essentially. So, let us take say I move to this one or I move I consider both I do not throw I throw either I keep this to best to once generate five children for this five children's. For this again thing the best two cumulate five children and move on this algorithm is called main search and this elastration has beam bits width into 2, so what have it done. Instead of hills claiming which explore only one path in the space they are saying we are allowed to explore more than one part simultaneously this space.

So, in this example beam beats as to be called as to, so this beam is like you can think of a like a beam of thoughts light know you should shining into and you can have more than one. So, at every level we have to notes this, so we will came going down the search space, but you will keep more than one option available essentially listen algorithm. It has been use very successfully in speech recognition and it imagines that is useful in situations where there are a few options which are likely to be candidates' solution essentially. So, if you can keep more than one option, so what is the problem in speak recognition that you start with the speak signal then converted into phonemes, then syllabus then words and so on.

Now, it turns out that very often different word combinations sounds similar essentially, so very often we think you said this or did you say this, we know the such a thing is possible impact, this in the chapter on Maxwell language processing. They are given an example of women who goes to New York and says women come from New York. I do not remember and we say everything here cause to nominal egg essentially. So, what was she saying what did a here what did the listener here the listener said cost whatever a nominal eggs the listener thought that women is saying that everything is cause a nominal.

Essentially, the women was actually saying work cost, but intended to say was everything a calls on arm in a egg, but you know people speak with different actions and that kind of I am sure you can see make both are them sound the similar. I cannot do that so easily, but that is what and what the twist a heard was cost nominal egg we can see that this kind of problem is actually very the case in speech recognition the people speak with different actions. The listener cannot always make sense of what at the same time in the number of variation options that one has. I am not too many essentially and then algorithm. Then, I will go some which can keep more than one options alive is likely to succeed in recognizing the correct sequence of word.

Essentially, in signal processing circles called is a Viterbi algorithm basically is beams search essentially now what is the complexity of beam search space complexity level yes considered that it is a constant because the beam builds is speaks instead of one known. You are going to keep beam known at every stage alive and so times number of neighbors is a maximum have to deal with essentially that is a constant space algorithm.

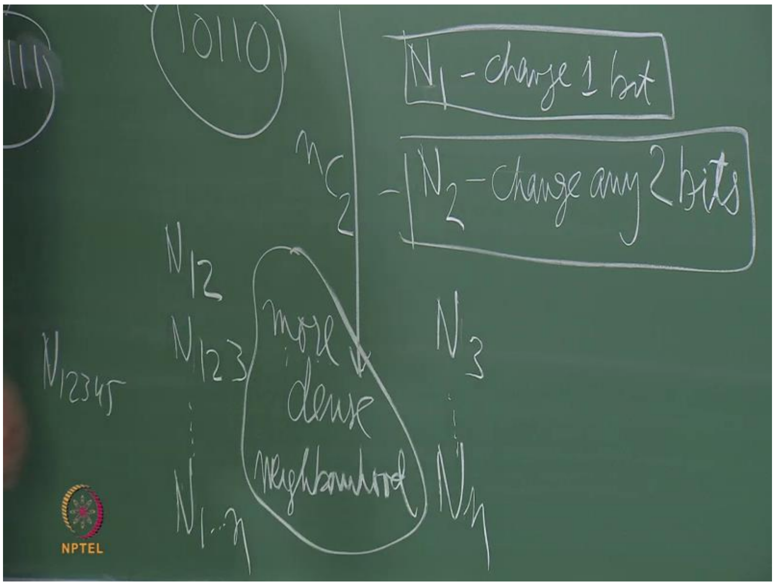
It makes around the hill local maximum problem where you know one bars may not have a good successor, but for example, all these may be bad, but this may have too good successes. So, may be some were have this solution will find which would not have worked out in this was better than this if this was better than this only this would I am in selective only this launch should have gone. It should be because we have kept two options both these branches are there so that one variation of how to get around this local maximum.

Next, come back to this solutions space sat problem essentially, so I want to talk about another algorithm which is. So, remember that I said that, so what is this is an about as a move than function it connects givens state in this case a given a candidate solution to other candidates solutions, but why should I choose this neighborhood function. Now, that is only cap what does the hill claiming algorithm do, it generates all this five neighbors and picks the best amongst them if it is better than it moves, so otherwise it is where it is. So, the complexity of the algorithm will also move them function will depend on how many in move how many in neighbors, we have will have to keep this.

Essentially, I can think of another move them function neighbor function, so just call it n two and it says change any two bit. So, n 1 says change one bit see that for this small problem of five variables sat problem, next my end to will give me a different neighbor function. So, for example, one of the neighbors would be 1, 1 or let me just see in the first 2 bits 0, 1, 1, 1, 1 another neighbor could be I change first and third bit 0, 0, 0, 1, 1.

It gives me a different set of neighbors not only he keeps a different set of neighbors it gives me more neighbors. So, this n 1 gives me 5 neighbors or n neighbors if n variables are n there this n two gives me n c 2 neighbors in this example that is ten neighbors. So, it is a different neighborhood function, I can choose between different functions how can I explore this information can I device an algorithm which it will try to make use of different neighborhood function.

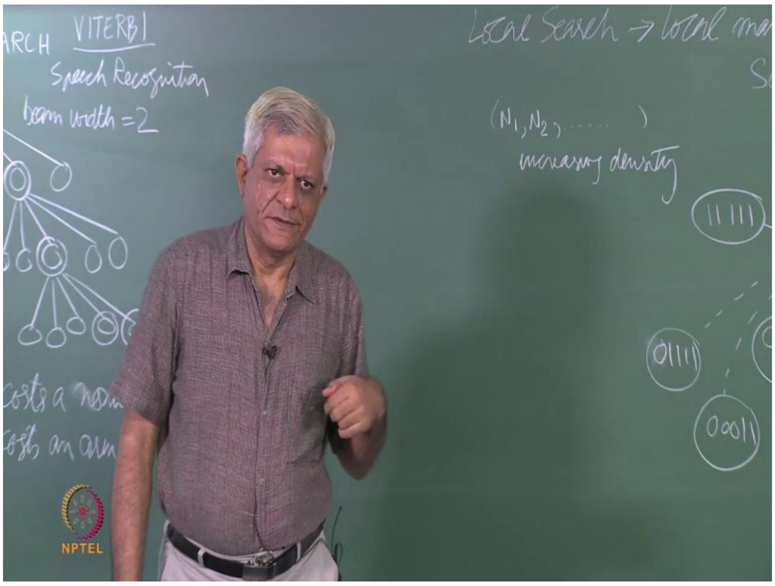
(Refer Slide Time: 22:11)



So, I can have n one n two then I can have n three and up to n if I have n variables and this case up to n five I can again say change all five essentially of course, that will give me only one neighbor. It move at me far from their all I can have combination I can device n 1 2 which says change 1 or 2 bits or change up to 2 bits essentially, I can say change up to 3 bits essentially or n 1, 2, 3, I can go up to none n which in our example would be n 1, 2, 3, 4, 5.

So, the first thing I want you to observe is that they are if you go down this path they are more dense the neighborhood is more bun dense as you go down this set of neighborhood function. So, I cannot write this here, so I can choose a sequence of neighborhood functions, which are in cleansing lead dense, which means they give me denser and denser neighborhood. Thus most simplex other most parts is n one which allow we taken get by changing one bit and then I can get n 2, we say 10, there I can get n 1 2 which as 10 plus this 5 which is 15 and so on.

(Refer Slide Time: 24:19)



I can create a sequence of neighborhood function n one n two which is increasing density how should I choose bits neighborhood function to use remember neighborhood function same as move then essentially. So, we have now the option of choosing move gen function on and we on the keeping mind that we role on the gets struck in the local maxima, so what will be the heuristic function here? If have a clause like this for a formula like this h of n for a given candidate solution these are all candidate solution say how do you give them a value any suggestion number of clauses number of clauses it makes a true that is a simplest you have doing it.

So, will keep that in mind or we can modify this to say weighted number, so if a clause has more literals then give it more weight. So, another possibility and basically it gives you an idea how much of the solution is solved in some sense if I have twenty clauses an I one candidates satisfies 7 and another one satisfies is 12. Then, may be the 12 is better than 7, just again a take remember just takes a not perfect this is only. We will basically assume that the most clauses the better the heuristic function, so we want to know device a algorithm it will search in this solution space and try to device in algorithm, which will not get struck in a local maximum.

So, what which what is a criteria one should use for choosing a neighborhood function. So, two things one is that we want to also keep a list on time complexity is time complexity going to all these going to affect may time complexity all in what manner these functions will affect my time complexity. So, that is why I am saying remember the hill claiming algorithm what will be do it generates all the neighbors and then picks the best amongst them and moves to them moves that neighbor then generates all the neighbors. Then, thus that what about this function this one what is the neighborhood function take this all example five variables a, b, c, d, e and this one is saying.

You can change any number of bits you want change 1 or 2 or 3 a when I say one or 2 I mean any one any 2, any 3, any 4, any 5, what is the neighborhood function entire space is the neighborhood function. Every note is connective directly to every other notes it can you can move from any state to another any other state in one step. How many maxima will this have? One only one which is the global maxima which is the solution, they may be more than one maxima. So, that one is the point, but what is going to be the complexity of the move gen function remembers that we said generates all the neighbors this is choosing this the move gen function and using hill climbing.

What are you doing? You are just doing a force saying generates all possible states and then pick the one best pronounce them, obviously how many states are there we have major observation that there are two states to n states essentially. If the neighborhood has to a to n states then you have actually the problem and may be not a group force search which is no point solving because you know to which instates part to many inspect any way. So, this hill claiming algorithm if you want to use this use they function the of 5, but may be for 30 or 40 or 100 variables would not even start of it would spend this whole life then waiting the neighborhood function done that.

So, what is the next base thing we can do, so we cannot choose the dense function that what I am trying to say why all we in president dense function because dense functions will not have this difficulty of getting struck in local maximum? What is the advantage of choosing a farce neighborhood function faster the neighborhood generate it faster essentially and choose us remember that hill claiming algorithm. Basically, it says generate the neighbors pick the best amongst them if is better move to that then generate the neighbors and so on.

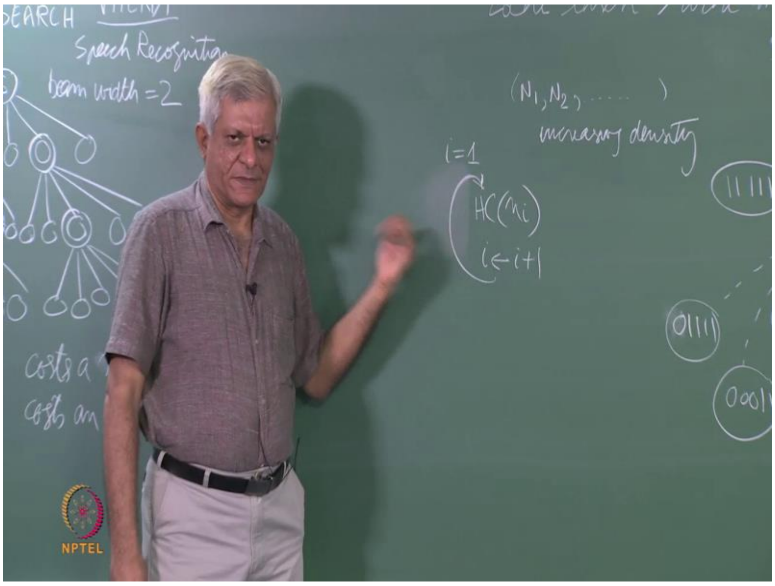
So, what do we do can we device the nice algorithm which will give us which allow us to see thus parcel the algorithm the neighborhood function the lesser the state you can. It is true that is obliviously the milling of the parts the mold likely it is that you not connected to the better state that denser the algorithm. The mold likely it is that for any given say there is always a better state which the extreme example is this algorithm because any state which is not a optimal solution a global maxima has is connected to the global maxima.

Every state has a better state and a only state is do not have better state of the global maxima states. So, if you can try imagine this that choosing move gen function is also devising the landscape. It is a difficult for me to visualize it here, but it is what a move gen function telling you will that one function is saying from here you can go to this to this to this to this or to this one. Another one is saying that from here you can go to some other set or together essentially and what is the lotion of a local maxima lotion of a local maxima is that this state is better than always neighbors that is a lotion of a local maxima if all the neighbors are the entire set like in the example then it is a global maxima.

So, in another way of looking at it is that denser function in many if you to the denser neighborhood function then probability of the likely hood of a state being as local maxima becomes lesser because it has to be a better than more neighbors. Essentially, it just a very naïve way of looking at it, but you can get some intention or that essentially. So, we would like to use a denser function because it is not likely to have states which are local maxima which mean the surface, it generates would be smoother and a cooler surface is more amenable to hill claiming.

Essentially, if a surface has no local maxima, then you will restore global maxima, so that will a simple approach to and is given in this book which has mention it also given in my book, but, I first edit here which is how to solved by modern in. It will really go and look at this book because begging in a every chapter a given nice interesting puzzle that you can solve. Essentially, may be one of these days I will give you one of the puzzles the algorithm is simple that you have access to a set up neighborhood functions and you do the following.

(Refer Slide Time: 33:37)



So, I will just write it very briefly call hill claiming, so the meaning of this is, so I call on sorry we just say hill claiming. So, H c stands for hill claiming hills hill claiming with a neighborhood function and you put i is equal to 1 and in a put i is equal to i plus 1 and put it into a move, what is the meaning of this algorithm? So, we should work out the details I am not going to write the details the meaning of this algorithm is and depending on what the resource is is allow you will go into this loaf. Once you run out of resource is you say stop something here, but inside this is a another loaf which is the hill claiming loaf and that is saying use the I as neighborhood function.

I am assuming that their order crossing to increasing density use the i h neighborhood function and when you terminate when you come out of hill claiming remember there we no longer have the notional gold test. So, in fact it is of course you can put in a gold test check, so for example, in a sat problem you can say yes if all my clauses are satisfied then stop you can put then that extra check, but we are looking at optimization more generally know. So, we want to find the maximum value of this heuristic function, so what we are saying is it start with the most sparse hill claiming most sparse neighborhood function claim us as claim up as much as you can, using a hill claiming analogy.

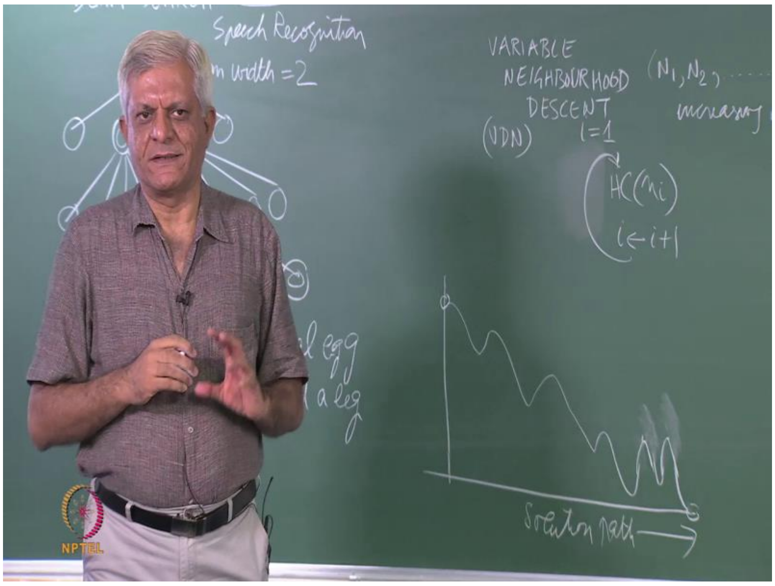
Then, switch to a different neighborhood function the next one an en claim up again and then keep doing that using neighborhood function. So, if you can visualize this somehow you can say think of the neighborhood function as you know giving you connection with ladles from one state to another state so on. So, each of them is a ladder and if the ladder is going up you claim up if you does not going up you stop essentially and many stop suddenly get a different set of ladders. Let us say this launch and see if you have better state then you keep claiming the different set of ladders and so on.

So, what is the advantage the intention behind this is that you do most of the claiming in the early stages that you early sparse a function will take you up quite a bit. Then, when you gets struck on some local maxima suddenly you have different ladder take you two another this is the algorithm its call variable neighborhood descent well they are talking about minimizing. So, it is imagined you are going down a value into the claiming hill or v d n were this is saying is that use sequence of denser neighborhood functions. Each stage you do hill claiming and you can have variation on this you can work out variations.

For example, one could say that after having using n to should that n 1 again allowed because now I am in a different states may be a n one will help from there is a. So, this a way essence that you try think of essentially for the basic idea is simple that if you us talk on a local maxima use a denser function and try that. Essentially, this means that depending on how were the resources you have you will explore it as many of this neighborhood function as possible essentially.

So, I was on the end with one example that we of discuss earlier, which is the example and what with a I want into a visualize what will the terrain look like men. Let us see call when expert and by expert we mean not necessarily who will find optimal solution, but who will solve it quickly according to alone solution, how will the terrain how will that journey look like.

(Refer Slide Time: 38:31)



So, the idea that I am trying to talk about is this that this is the solution, so this means you know what the solution that move 1 is, move 2, move 3, move 4 and so on. Somebody else gave you a solution how will the terrain look like given some heuristic function like the number of tiles in place or something like that initially the number of tiles out of place. Let us see initially little be some value which that a out of place finally, of course it choose be 0, but how was the journey and remember that we said we makes does observation. When, you have partially solve the rubrics you many mode to move further from their you have to disrupts some of the earlier partial solution that is reflect by saying that the journey is like this.

You do like this and you have solve the top row, but now you have on do the middle row you have to disrupt this, but you do something better then like this then there a some variation then a. So, it will be something like this, so whatever trying to illustrate here is that if a expert the solving is the rubrics cube, then at each state on the way if you want to flirt the heuristic value it would something like this. Of course, it means that may be this is an excavation maybe it is like this function like that that real one problems a search there it is not. So, easy to devise heuristic function which will generates most of surface essentially, so Rubik’s cube is just an example of that which mean that I cannot solve.

I cannot even hope to solve rubric cubic hopes an using a hill climbing like algorithm because I will find it extremely difficult to device a heuristic function whose surface would be monotonically decreasing towards the goal state essentially. In practice since I have to disrupt the cube on the way I will be first two go against the heuristic function essentially. So, going with a heuristic function we will call exploitation of the heuristic knowledge and as oppose to a exploitation is following. Somebody told you do this sense, so you do that essentially as oppose to exploitation is the notion of exploration to explore new idea new areas of the space essentially.

So, for our only attempted exploration has been this beam search in which we have said you will keeps many candidates alive, but in the next class then we meet we will see how to try to increase component of exploration in search. The reason behind that is at exploit pure exploitation which means purely following the heuristic function is going to end of in local maxima even in the surface has local maxima what kind of algorithm. Can we think of that will go beyond the local maxima and to the global maxima. So, I will stop here and will take that of next week essentially.