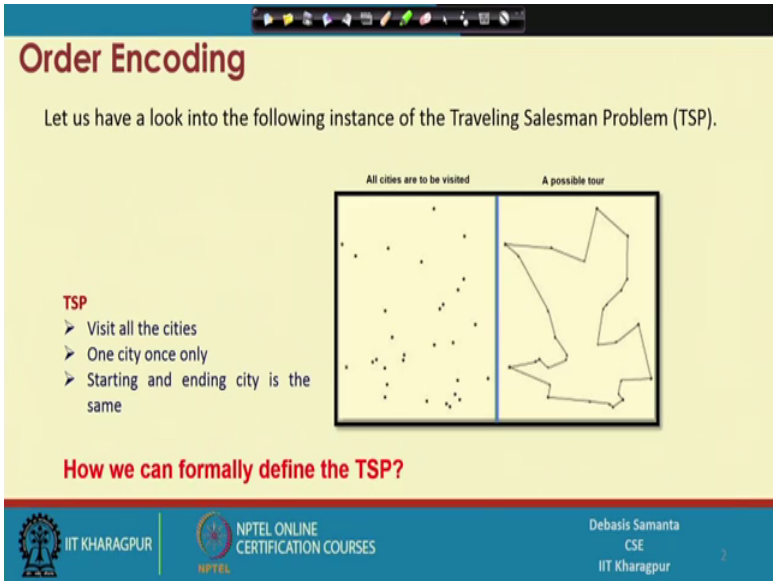
Introduction to Soft Computing
Prof. Debasis Samanta
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture - 17
GA Operator: Encoding schemes (Contd.)

In the last lecture we have learned about 2 encoding scheme the binary encoding and real value encoding. Today, we are going to learn other 2 encoding scheme namely order encoding and tree encoding.

(Refer Slide Time: 00:36)



So, the order encoding can be better understood with an example. So, I want to give 1 example this example we cited from the famous problem it is called the travelling salesman problem.

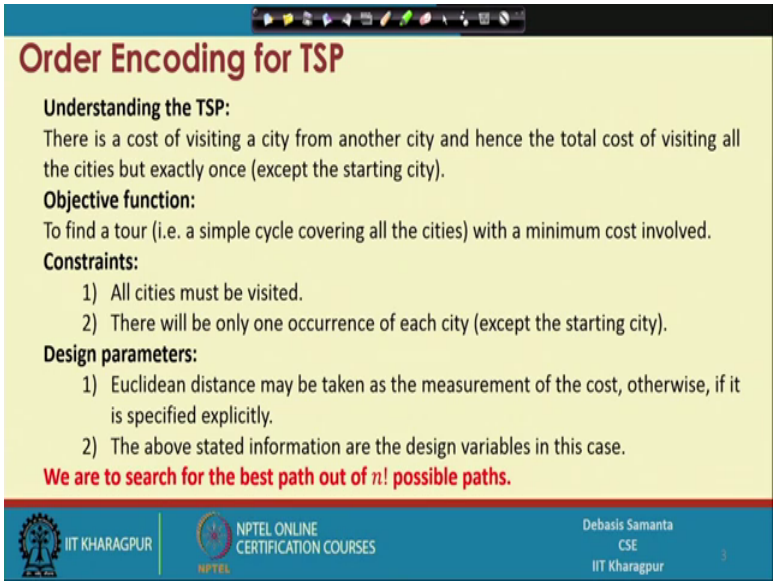
So, in the travelling salesman problem, the problem is defined like this, there are n number of cities and some cities are connected to other some cities. So, that one travellers he can travel from one city to another and so for the travelling from a city to another city there is a cost involved. So, the problem is to find a path or a route that the traveller should follow. So, that he will incur minimum cost of travelling, but the constant is that he should travel all the cities, but exactly once and he should return to the starting city at the end.

So, this is the problem and this problem is called travelling salesman problem and famously it is called the TSP and in the TSP, this figure shows basically a simple way of representing the different cities. So, that location of the different cities on the surface of the earth and one path that it is one tour rather we can say one tour for a travellers is shown here, say is this is a tour.

So, if there n cities are there, then essentially we have 2 to the power n different tours are possible, that means, there are 2 to the power n number of solutions are possible and out of these 2 to the power n number solutions we have to search for the optimum tour; optimum tour in the sense that the tour which required minimum cost for the traveller.

Now, this is a problem and optimization problem also little bit understood. So, now, let us see how this optimization problem can be defined and then corresponding it is encoding.

(Refer Slide Time: 02:44)

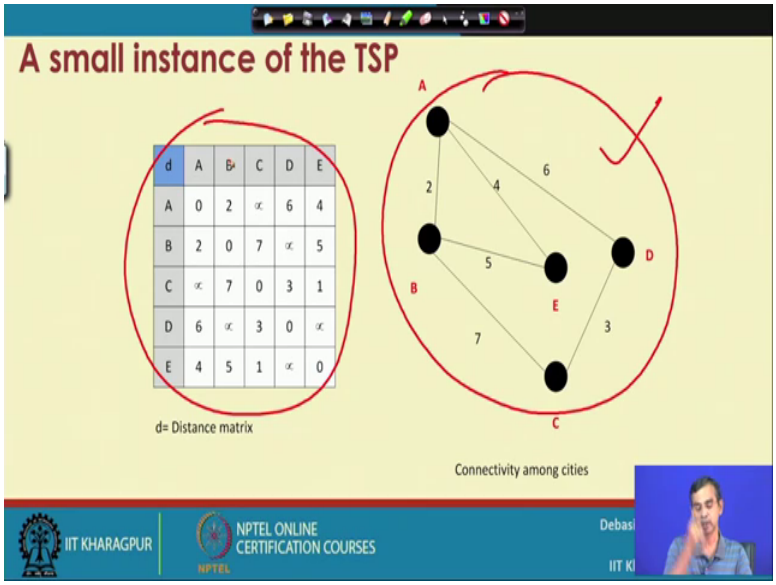


So, here is the idea about how the optimization problem can be express, that is fine. So far the optimization problem is concerned; the input is we have given the different cities and the cost of travelling from one city to another. So, this is the input that is given and also it is to mention that which is the starting city. Then, objective function is basically to find a tour or rather we can say a cycling covering all the cities exactly once, except the first cities and with the minimum cost involved and the constant in this problem are all

cities must be visited and there will be only one occurrence of each city, that means, he should not visit the same city more than once except the starting city.

And here, design parameter we can consider about if the location of the cities are given and then Euclidean distance between the 2 cities can be taken as the cost, other I it is spaces if it is specified explicitly there what is the cost of travelling from one city to another and then this problem can be stated in terms of mathematical representation, it is like this.

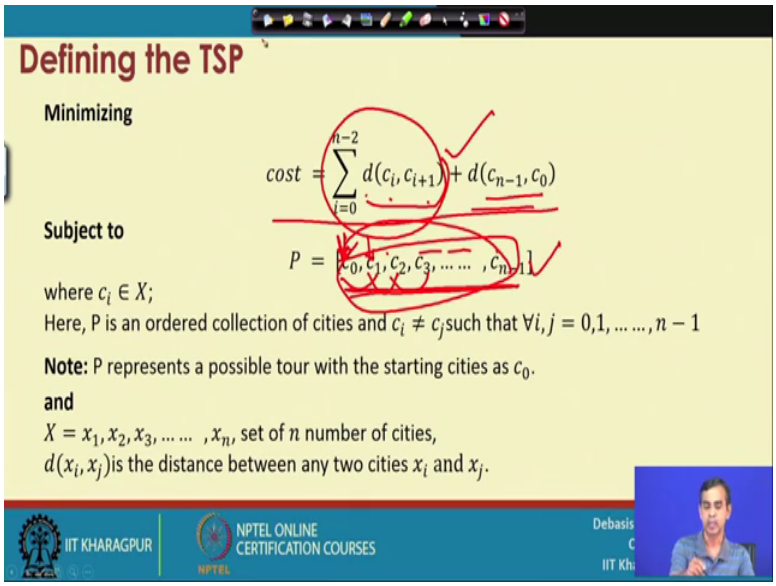
(Refer Slide Time: 03:59)



So, this is basically the input, this input say here this problem is simplified for 5 cities, the cities are termed as ABCDE and then is the cost of travelling from city C to D it is basically is a unit is 3 likewise and if there is no path from one city to another, then the cost of travelling is a very large, so it is infinite for example, so from city A to city C there is no path. So, here this is the same representation which is shown in the form of a matrix it is represented here, from city A to city C we do not find any path, that means, we can say that this is the cost of is huge or infinite cost like this.

So, the idea is that, if this is essentially the pictorial description of the city map and the same concept it is stored here in the form of a matrix. So, this basically the input to the problem it is the cost matrix like this one. Now, having this is the problem statement, we can define one encoding scheme like, so order, that means, what are the different possible ordering that it can have.

(Refer Slide Time: 05:21)

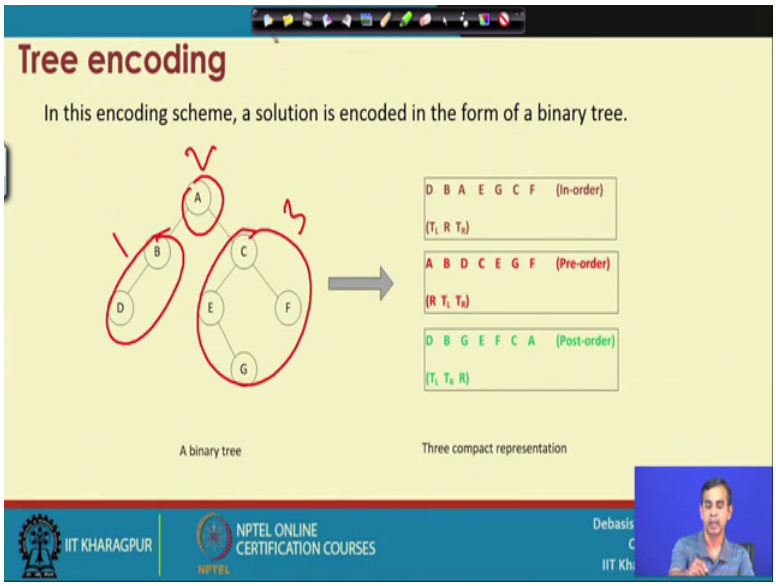


Now, regarding this ordering as a simple example, fine first before going to this ordering scheme again let us first little bit explain about the objective function, here the objective function is defined using this formula; here d the distance from any city c i to c i plus 1 and if we at any instant, so this is basically the at any instant, this is the solution, where the solution is that that order first start with c 0, then c 1, then the city in sequence.

So, this is basically ordering of the different cities are there and then if this is the solution then we can evaluate the cost of this even this formula. So, c0 to c1 what is the distance, then c 1 to c 2, then c 2 to c 3 these are the so sum of all distance and then the finally, c n minus 1 to c 0 because he has to return to the starting city then. So, this is the distance of covering from n minus 1 to this one.

Now, so if this is the 1 solution and then we can obtain the cost of the solution. So, here encoding scheme, this is the encoding scheme that we have followed, it basically the sequence of visit, visiting the cities. So, sequence means if city A is first, then city D, then city B, then city C. So, it is ADCB like this, so it is the different sequence of there.

(Refer Slide Time: 06:45)



Now, so this way we can encoded and this is the major the main concept in the order sequencing. So, this is pretty simple actually, so what are the different sequence it is possible and then that is the order encoding scheme. Now, after this order encoding scheme we will discuss about another encoding scheme it is called the tree encoding scheme.

Now, the concept of tree encoding scheme it follows the concept of tree in this slides I have shown one tree, here A is basically starting node and form A there will be 2 what is called the child the B and C, similar C being the node it has 2 children E and F, for the E only one children it is called the right children and there is no left children, for B similarly it has left children and no right children. So, it is a typical form of tree and here we can note that one parent here has at most 2 children. So, it is in some sense it is called the binary tree.

Now, having this is the binary tree, it has basically representation of certain what is called the way of I mean storing the data, if all the values in this node represent some symbols or data for example, ABCD in this case, then in the order they will be visited it basically called one sequence or one encoding.

Now, I have mentioned 3 different ways the tree can be visited and the first method is called in order. So, first we will visit, so in order means to visit this things, we have to visit this things first, then we will visit this one and then we will visit this one, again the

same way to visit this one we have to visit this one, then visit this one, then this one likewise.

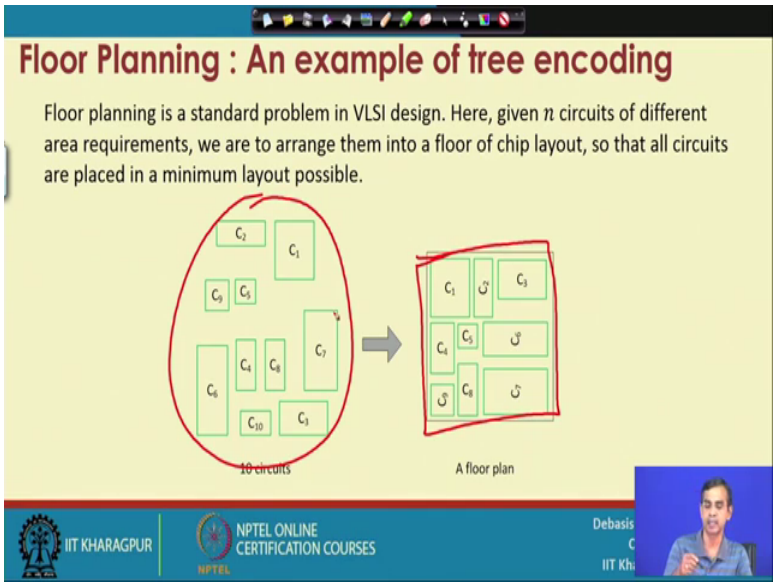
So, if you follow the same procedure, same policy for each then the visiting will be there and then it will be listed for example, so if we consider the visit that we will visit this things first, then this thing and then this things. So, visit this one, then this one and then this one, it is called the one sort of visiting and this is one sort of encoding in fact. Now, for using this one, so the idea it is that we will visit in order to visit the tree we have to visit this one, to visit this tree we have to visit this one. So, first these visited, then we have to visit this one it is empty. So, no nothing to mention here, so the next B, so this completes the visit.

So, these visit is basically D and B, so it is basically the left tree is visited left part visited, then we visit this one. So, it is A, then the right part will be visited again visiting right part we have to visit this one, now visiting this part means, so it is empty. So, no need to go there, so we will visit E then G, so basically E and G. So, visiting this one, then we will visit C, so it is C and then finally, the F will be visited. So, this is the order, so it is called the in order traversers.

Now, similarly pre order traversal means we will visit this one first, then this one and then this one. If you follow it, we visit this one first, then visiting this one mean B first, then D, so BD. Similarly, visiting this one means C first C, then visiting this one means E first E and G and then finally A, so this way the different thing.

So, what I want to say is that if this is the tree that is given to you, then these are the different way the tree can be visited and each way representing one, what is called the encoding form. So, this way it is the tree encoding come into the picture and it is also some way resembles with the order encoding scheme that we have discussed for the genetic algorithm.

(Refer Slide Time: 10:47)

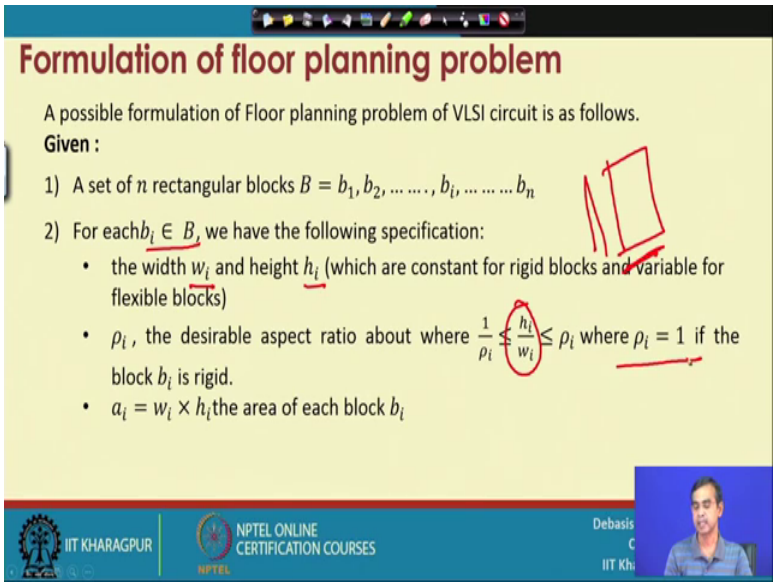


Now, so this is the tree encoding scheme and this is 1 encoding scheme is there and next I want to give 1 example of this tree encoding scheme, this example is basically a is a problem it is called the floor planning problem. So, floor planning problem is basically the idea is that the different what is called the blocks are there.

So, these are the input blocks of different sizes and you have a floor and then you have to arrange all these blocks into this floor. So, that the there will be no wastage of space and within the minimum floor area we can place all these blocks into the floor. So, it is basically layout of the floors needs to be decided by placing all the blocks into that floor. So, that it takes the minimum area or something else like.

Now, this is the 1 example called the floor planning and we will see exactly how the tree encoding can be applied to solve this kind of problem.

(Refer Slide Time: 11:54)

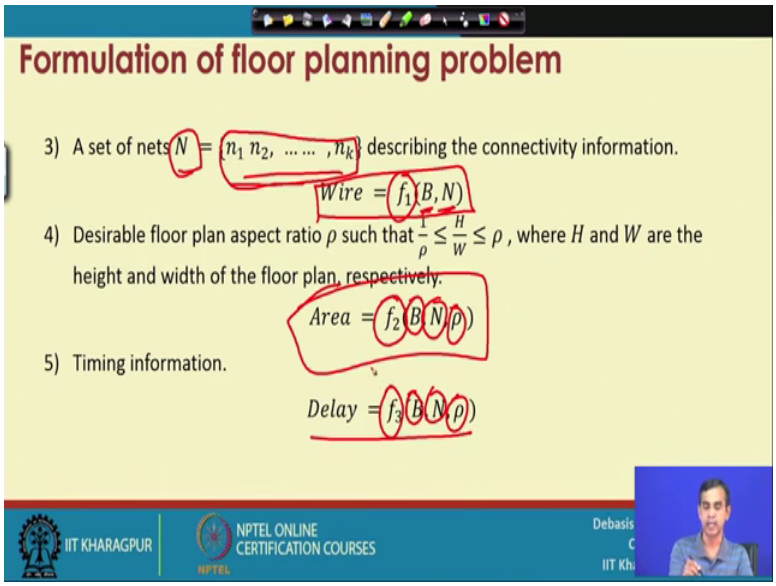


Now here, to be more specific for the problem is concerned we want to make a little bit complex. So, that the problem has takes it is own strength like.

So here, the input is a set of blocks; n number of blocks and we assume that all blocks are of rectangular size and these blocks are denoted as b1, b2 up to bn. So, these are the n number of blocks are the input to this problem and for each block there is some specification, the specification is there; for each block it is specified by width and height. That mean, if this is a block it basically width w and height w, so it is specified and there is also another specification. So, and then blocks are rigid, rigid means no a, this width and height cannot be changed and rho i for each block it basically states the aspect ratio; that means, it is basically decided rho i this one, hi by wi.

So, if rho i equals to 1, if the blocks bi is rigid, otherwise it is n. So, we assume that rho i is equals to 1 this case and then given the w and h i we can easily understand our area of the block.

(Refer Slide Time: 13:19)



So, a i denotes the area of the ith block or block bi. So, this is the specification of the problem at n and then we will see exactly how the problem or what is the objective function for this problem is there.

Now here, so there are n number of blocks and we assume that the blocks are connected from there is a connection or you can say line from one, what is this is basically the problem related to the VLSI problem, it is called the very last scale integration where we have to place the different circuits, a circuits resembles to one block and then there is a input output connections. So, this connection is basically called the net, so there are n number of connections which is shown here, the set of connections; n is a set of connection, these are the connections.

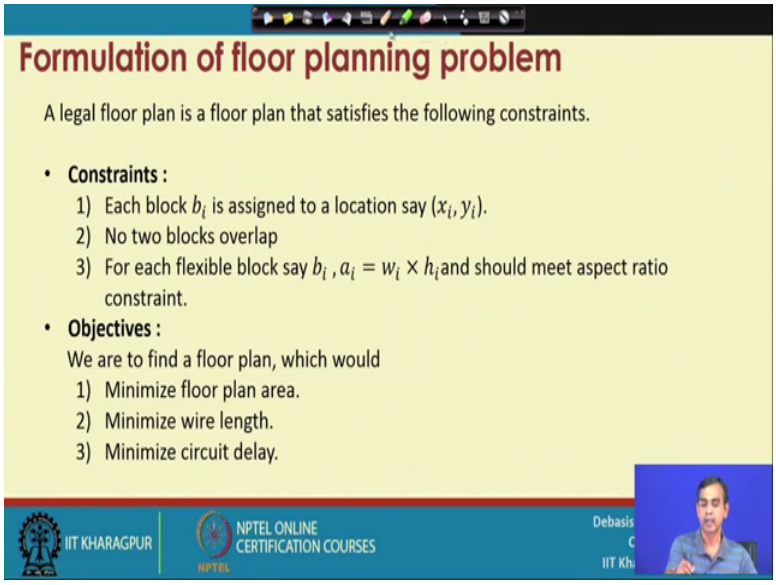
So, the idea is that if B is given there and N is the connection and f1 is 1 objective function it is known there, by which it can say what is the total wire that is required, wire means how many connectivity are there, how many cost of networking is required and also if rho the aspect ratio is given for each block and then set of block is given and then this is a network connection then what is the area that is required it is also given by this function f 2.

Similarly, f 3 is another function it will basically calculate delay of the entire things for so far the circuit is concerned. So, it is given the set of block and then connection and then rho the aspect ratio then, then f3 this will give the measurement about the delay

involved in the layout design. So, these are the 3 functions f1, f2, f3 it is obtained from the VLSI specialist, they given this B and N; B and N rho or B N rho, they will be able to calculate f 1, f 2 and f 3, namely the wire length, the area and the delay of the block that is required.

Now, so our objective, so far the genetic algorithm is concerned find a network connection out of the so many networks and from the B, so that this f1 can be minimised, f2 can be minimised and f3 can be minimised. So it is basically the multiple objectives here, unlike the single objective that we have discussed so far.

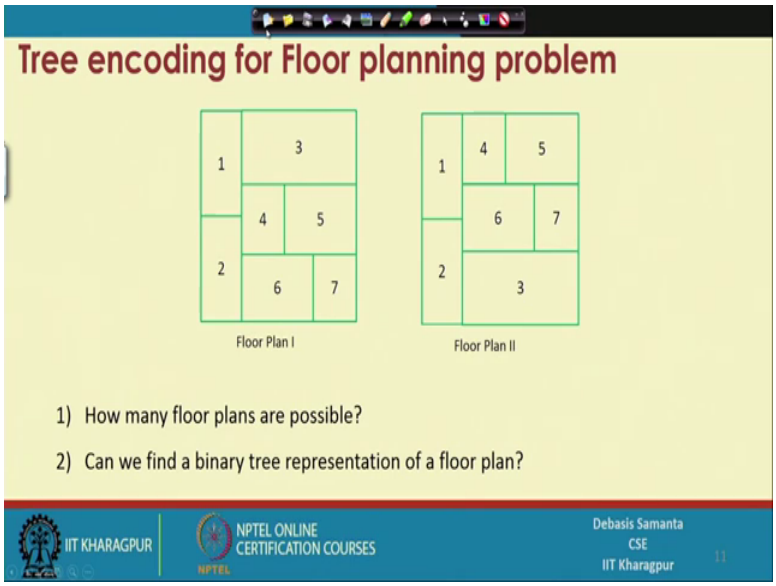
(Refer Slide Time: 15:34)



Now, in this multiple objective, we have to we have to minimise 3 objectives, the area and length and then circuit delay. So, these are the thing defined by the function it is already discussed.

Now, we will come to the encoding scheme rather. So, how the encoding scheme can be there?

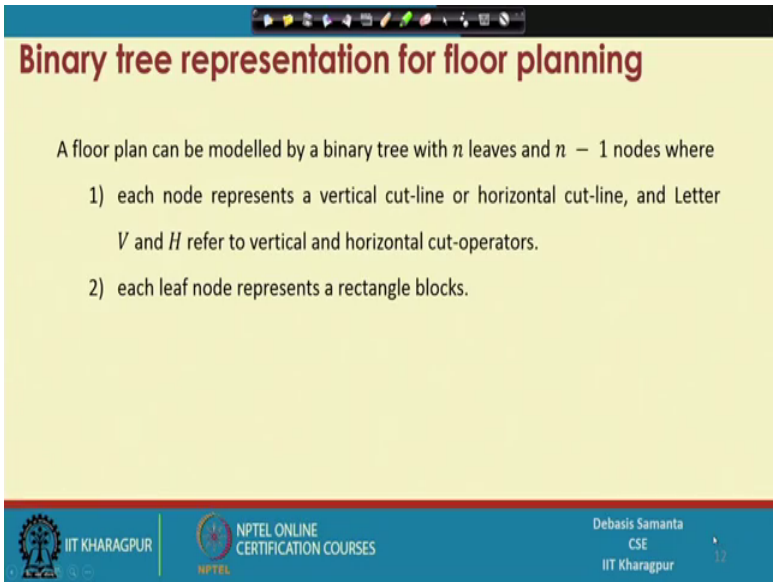
(Refer Slide Time: 15:53)



Now, before going to the encoding scheme let us consider 2 instances here. So, these are the different block 1, 2, 3, 4, 5 like this on and these are the 1 plan, so that if it is plan like this. So, for the same set of blocks 2 plan we have mentioned it was and this is the 1 plan. So, for this is the 2 floor plan are given here and then we have to see whether for these floor plan area is minimum or delay is minimum or the length is minimum whatever it is there. So, basically the idea it is there.

Now, given a set of block can you imagine how many different floor plans are possible? There in fact, many, that is a, and searching for all the floor plan and then ultimately finding the minimum is a very time containing problem. So, this can be represented in GA frame work and then finally, can be solved using genetic algorithm.

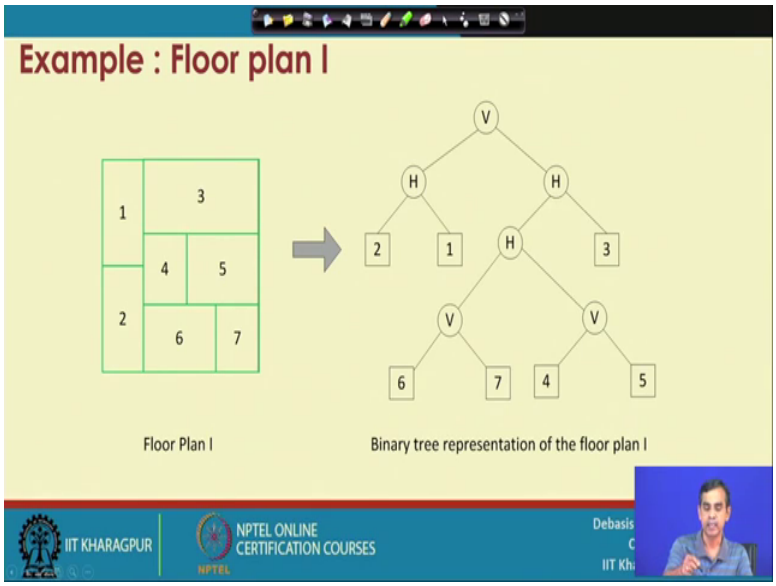
(Refer Slide Time: 16:54)



Now, so to do this thing, we have to do certain encoding scheme. Now, so encoding scheme that we are going to propose it basically like this, it can be modelled in the form of a binary tree because we are going to realise it is in the tree encoding with n number of what is called the leaf nodes; that means, the last node does not have any children and n minus 1 the non leaf nodes and each node represents a vertical cut line or it is called the horizontal cut line and later they can be represented by V and H respectively and each leaf node represents a rectangular blocks, so this is the specification.

Now, let see how the same thing can be represented in the form of a tree given floor plan like. So, idea it is like this.

(Refer Slide Time: 17:40)



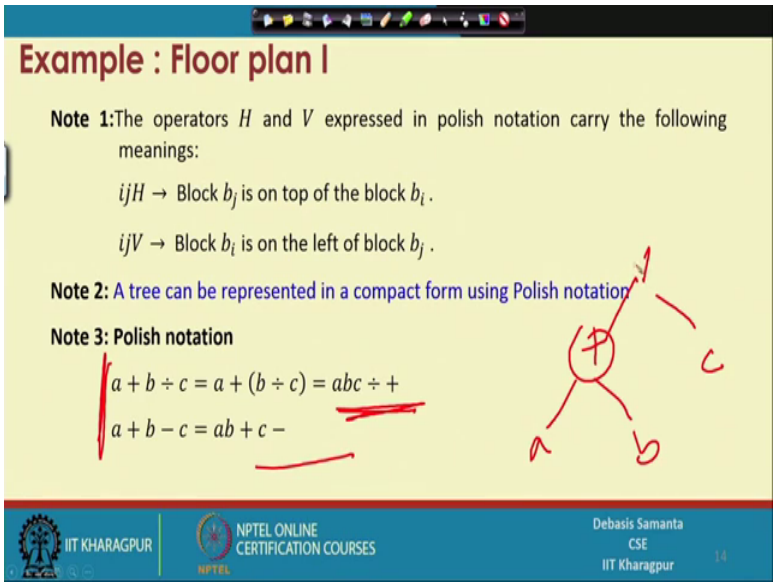
Now, here you can see it. So, this is the 1 floor plan and this floor plan can be represented equivalently in the form of a tree, so here V basically vertical line. So, it is basically the vertical line and then there is a 1 horizontal line, so it is V then H and then 2 and 1, so these are leaf node.

Similarly, if we come to this part, so this is basically left part of the tree. Now, these are right part again from 1 vertical, so right part, this is the right part, so these are right part. Now, in this right part we first have the horizontal cut. So, this is the 1 horizontal, so this H and these 3, so this is the 1 and then again this part, so this part again another tree, so it is represented by this part. So, here 1 horizontal this is the horizontal cut and this horizontal has 1 vertical. So, it is vertical 4, 5 and another vertical 6, 7, so this one.

So, idea is that, this kind of floor plan can be represented by means of a tree whose structure is here. Here, these are the leaf node leaf node represents the block ultimately and other the non-leaf nodes VH, they are basically whether the vertical cut or horizontal cut. So, it is a vertical cut, it is a horizontal cut back on so.

Now, so the idea is that any such floor plan if it is given you, we can find is a equivalently the corresponding tree and this is the concept of tree encoding in this case, now as a few example further.

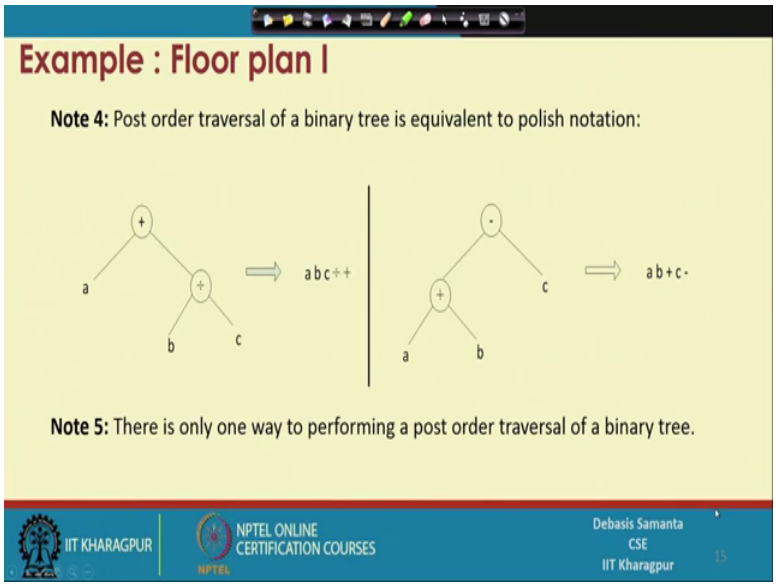
(Refer Slide Time: 19:22)



The different floor plan if it is given to you, you will be able to represent it and the tree can be represented using the different traversal that we have discussed in order, post order and tree order. So, specifically 1 particular order is used that is called the post order traversal; post order traversal can be expressed in this form.

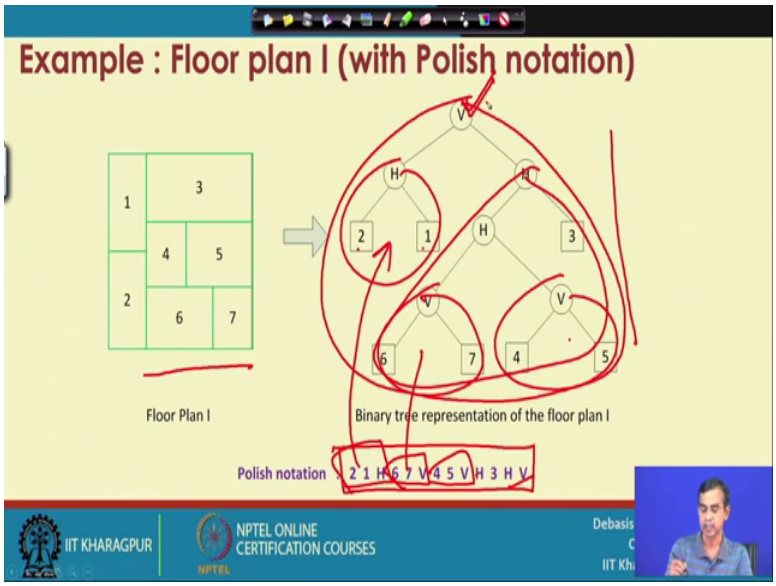
So, if it is like plus a, b then divide, then c, so is a plus b divided by c this kind of thing. So, it is basically a, b plus then c this one, so it is like that. So, this is the way that can be, so anyway some tree representation can be used and then the 1 traversal the post order traversal can be followed and then it can be obtained.

(Refer Slide Time: 20:14)



We can represents one example it is like this.

(Refer Slide Time: 20:18)



Here for example, here say, suppose this is the floor plan and this is the equivalent tree. So, this tree is a graphical display, this can be also alternatively represent in this form, it is basically called the polish notation or post order traversal; post order traversal means, so in order to transverse it first left, then right and then finally this one.

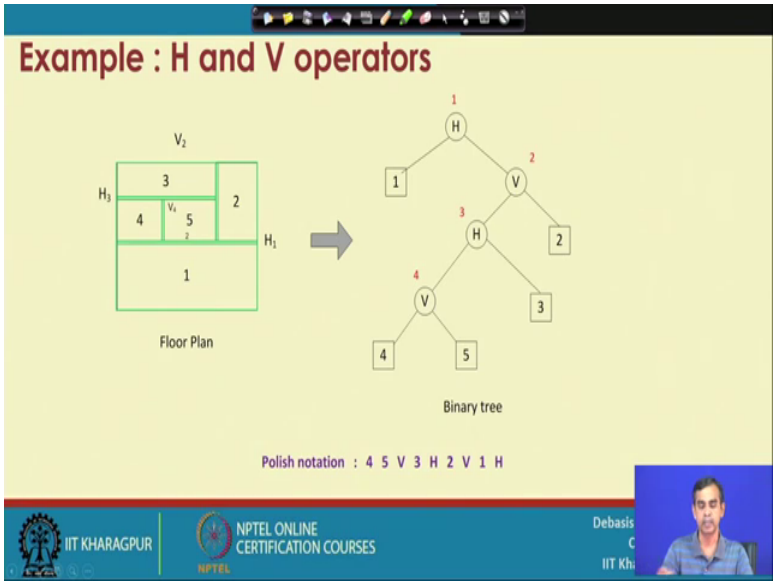
So, here for example,. So, first left means we have to find this 1, now to do this again left, right this one, so 2 1 H so far the this traversal is concerned, then we have to

traverse it. So, to traverse it first left and this again left, so 6 7 V, so 6 7 V corresponds this traversal first, then this one it is 4 5 V and then finally, it is H this 1 and then finally, will visit 3 and H. So, it is this one and finally, the last one. So, this way, so this basically if this is the tree given to us, then it has the equivalent representation which is shown here.

So, what I want to say that, if this is the floor plan and if this is the tree encoding, then this tree encoding can be come back to a represented by this one. So what, in other way, other what is it is that, this is basically is a encoding scheme for a solution. So, this is the solution and the solution is represented in this form and this is the encoding scheme and this in fact is a solution means it is the chromosome.

So, in this case the length of the chromosomes same as the number of nodes involved here. Now, the number of nodes can vary if the different order is followed, so obviously, it is the variable length chromosome can be obtained if it is like this.

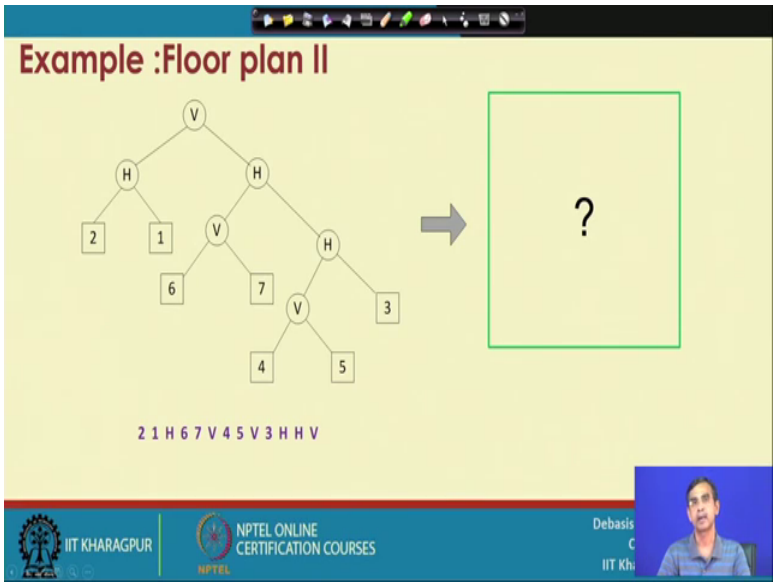
(Refer Slide Time: 22:02)



Now, so the same idea can be extended and then you can for example, if this is the 1 layout it is given then corresponding for your practice you can check it, if this is the like one solution this is the tree and this is the solution representation or encoded form of the solution or a chromosome, so the idea it is there.

Now, the problem here, our object is to solve the optimization problem.

(Refer Slide Time: 22:30)

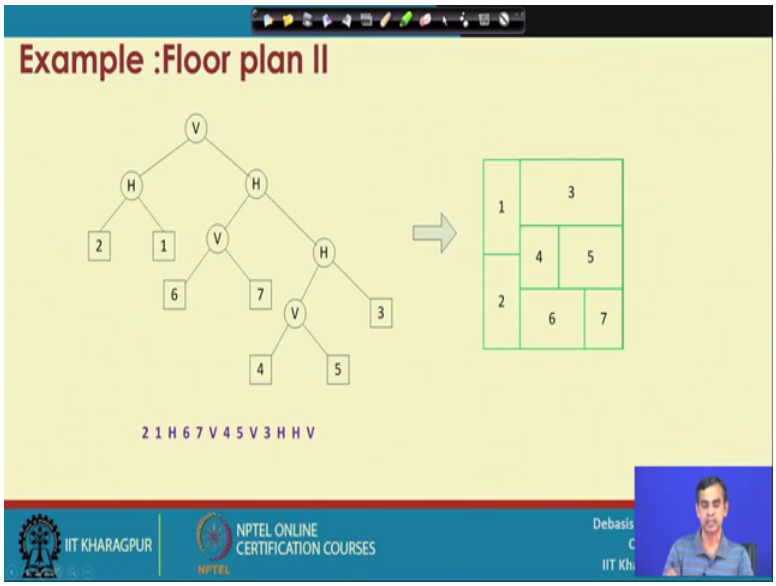


So, that how we have to find the best solution here. Now if, so alternative also if this is given we can draw the floor plan also; that means, if it is given we can do it or if it is given we can obtain it and then we can obtain the floor plan or directly from the floor plan. So, all the representation are basically same and then same thing can be obtain uniquely.

Now, so idea here is that, this is the one floor plan we have to obtain, then one solution is tree that means one representation, that can be thought of; that means, how many way this representation can be thought of, of if the n number of nodes are there, so basically the number of solutions are, basically how many different tree that can be formed.

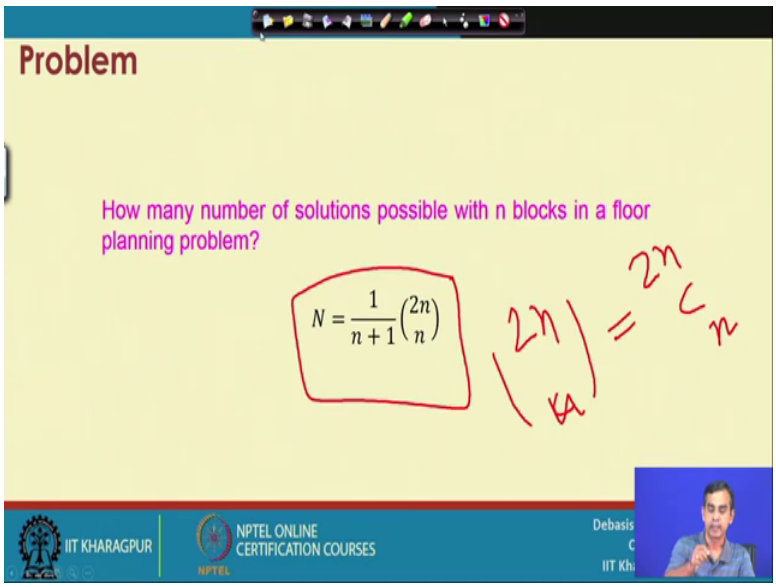
Now, so if it is only fewer number, then no problem, but actually this number is a large number; that means, for n number of blocks the number of trees that is possible which represents 1 solution is a very large number.

(Refer Slide Time: 23:32)



So, this number is, so this is the, another example you can understand whether this is correct or not, this is the tree representations, this is the floor plan and this is this one.

(Refer Slide Time: 23:39)



Now, the question is that how many number of solutions possible if there an n number of blocks in a 4 floor. So, if floor plan with n number of blocks, then how many solutions are there? Now, it is very difficult to calculate the number of trees that is possible and it can be shown that the number that is possible is this one. So, so this, this is even

eventually a very large number for n where 2 n, n basically it is it represents 2 n; n, this is equals to basically combination 2 n c n.

So, this is very large number in fact, and then, so it is a large number means we cannot solve in real time. So, it require some other approach or hard programing approach cannot dry out all possible trees and then finally, finding the optimum tree; optimum tree corresponding to the, the best floor plan in this case.

So, the genetic algorithm approach is, we have to randomly choose some solutions, randomly choose solutions means, whatever the pattern that we have discussed about there one pattern or one path or tour and then decide the solution and this way we can try out all possible in a random or probabilistic search manner.

So, this is the concept of tree encoding that we have discussed about it and then we will discuss, so this is the encoding scheme that is followed in genetic algorithm and we will discuss about other operators, the next operator that we will discuss in the next lecture slide is the selection scheme; that means, how to select the best solution out of many solutions are there. So, we will be discuss in the next lecture.

Thank you.