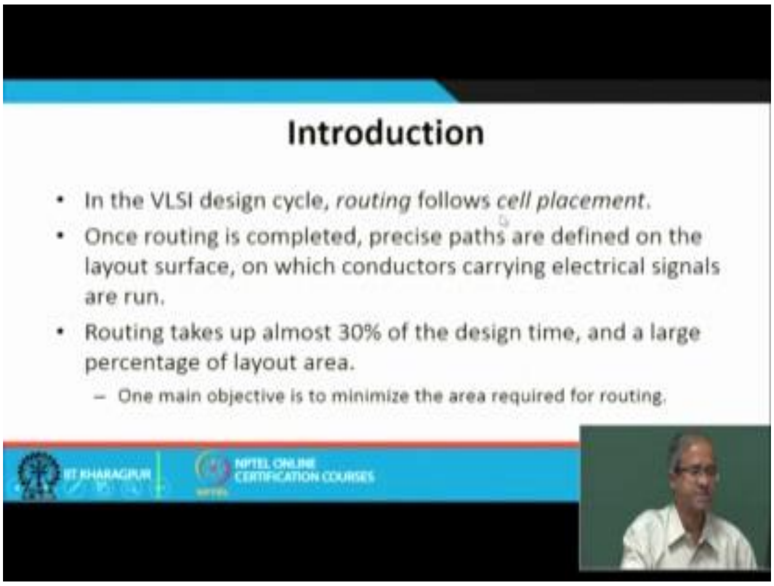
VLSI Physical Design

Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 15
Grid Routing (Part I)

Welcome back now we discuss some topics about routing, now that we have already discussed the issues about floor planning and placement of course, means with the sub problem of partitioning wherever required. Now we shall be looking at the very important step of routing, why it is required and what are the different types of routing involved.

(Refer Slide Time: 00:54)

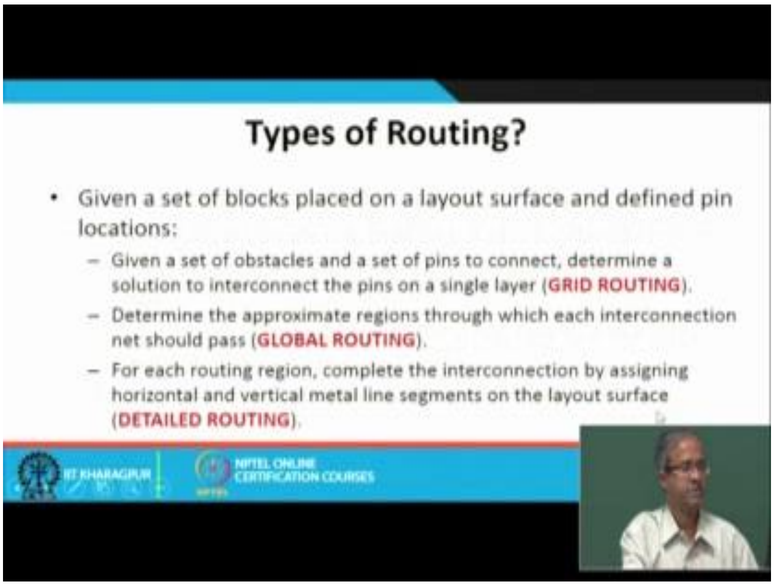


So, we start with something called grid routing. So, let us see basic problem of routing to start with. So, as I have said that after cell placement is completed the cells or the modules are placed on the silicon floor, we have to inter connect all the pins according to the specification of the nets this process is called routing.

Now, once you complete the process of routing what you do is that, you actually determine the precise paths for the wires to run between the pins so as to connect them. Now usually this routing is carried out on the metal layers in silicon. So, in the modern day VLSI technology there are several metal layers which can be used for interconnecting different pins, which are required to complete the nets.

Now, one very important issue is that, routing is important because inter connections in a typical chip can take us to an overrate of almost 30 percent or even more of the design time as well as the area over it. So, routing is an issue which needs to be given utmost importance and a good placement followed by a good routing step, will give us not only smaller design times, but also compact layout area. So, the main objective for routing is to minimize the total area required to layout to wires.

(Refer Slide Time: 02:44)

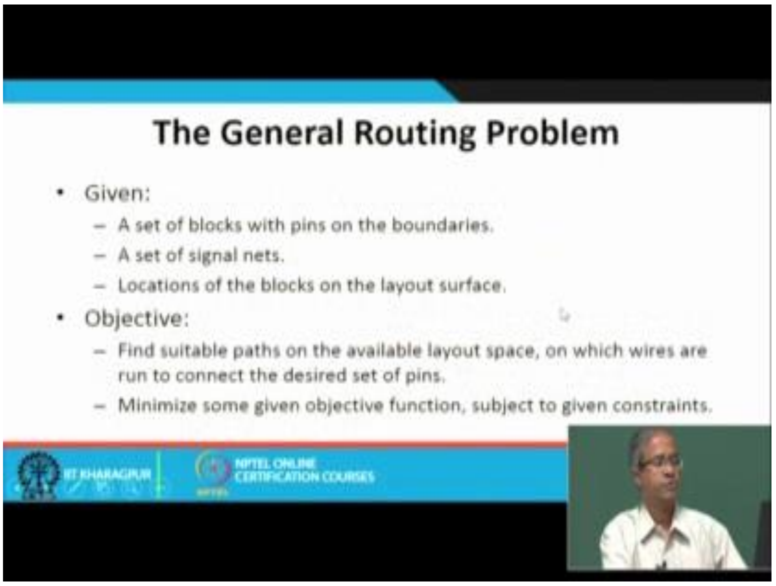


So, broadly we shall be talking about 3 different kinds of routing. So, let us see in a nutshell, what these 3 different routing processes are; now the basic problem is the same we have a set of blocks or modules which are placed on a surface 2 dimensional surface, and we have to inter connect some of the pins across them.

So, this problem can be subdivided into 3 classes or categories, one is called area routing or grid routing. Here the idea is that we have a set of blocks, which can also be regarded as obstacles in the sense that we cannot use those areas for interconnections, for running the lines and a set of pins which needs to be connected. Grid routing says that we have to find out a good path to interconnect the pins and the entire wiring has to be completed on a single layer, this is a very specific requirement.

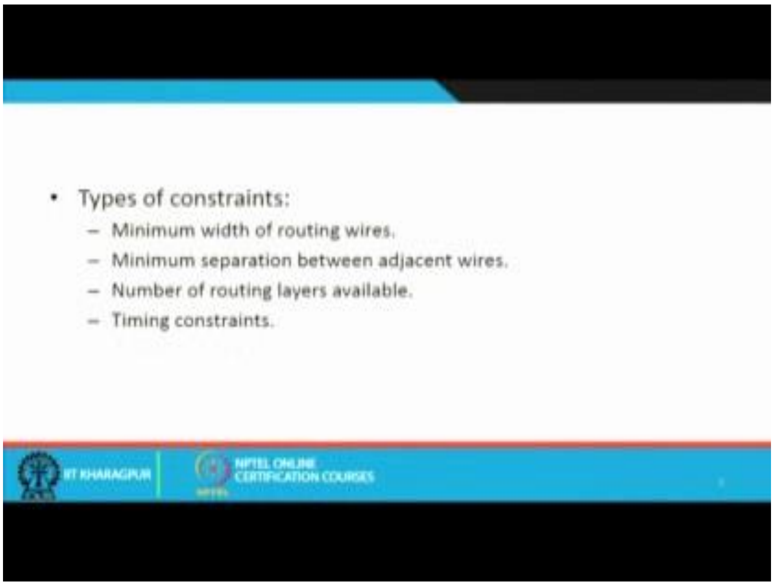
Global routing and detailed routing goes hand in hand, global routing says for the more complex problem in the chips where there are many modules or blocks there are lot of set up pins to interconnect. So, for each of the inter connection nets, you will try to determine the approximate sequence of regions through which the nets should pass through; and once this process is completed, for each of the regions we have the complete specification of the inter connection or wiring requirements. So, now, in the step of detailed routing, we actually assign exact horizontal and vertical metal segments in those routing regions. So, as to complete the inter connections.

(Refer Slide Time: 04:56)



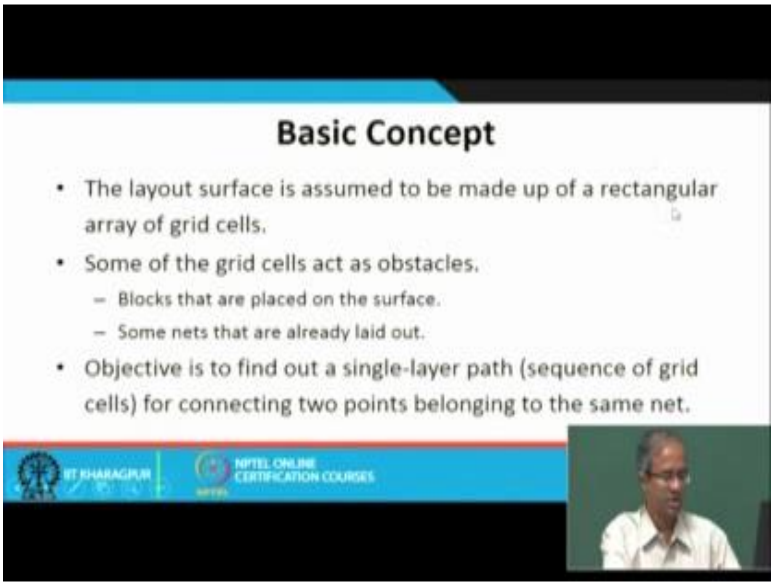
So the general routing problem you can visualize like this, you have a set of blocks with pins on the boundaries, you have a set of signal nets which need to be connected and these blocks are already placed, locations of the blocks on the layout surface. The objective is of course, to find suitable paths through which you can run the wires, to connect the pins as required and of course, you have to minimize certain objective criteria. Now depending on the scenario, depending on your priorities the objective criteria may be different. So, let us look at some of the more important objective criteria which people have tried to address and tried to optimize.

(Refer Slide Time: 05:51)



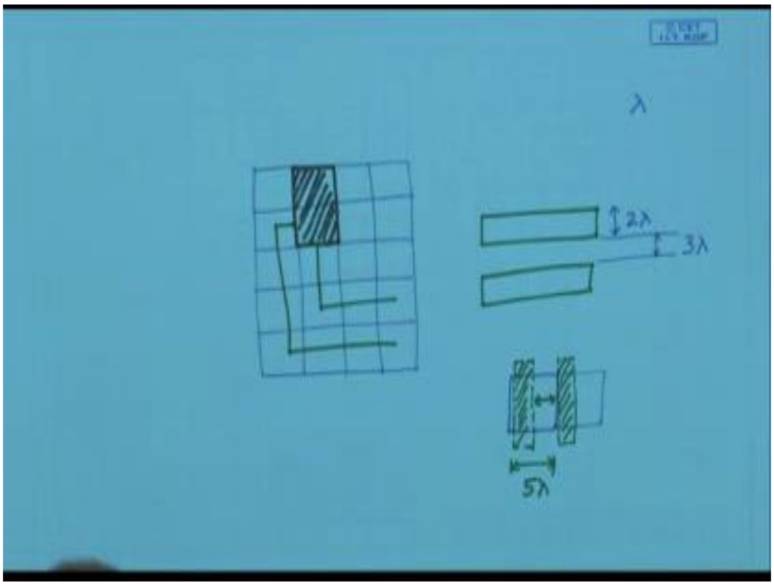
So, you may try to minimize the width of the routing wires or the total area of a routing you want to minimize, you want to minimize the separation between adjacent wires wherever possible, number of routing layers this we would may try to minimize like for example, you may say that well I can complete my routing in a single metal layer, that is the best thing that that you can have, but we will see later that singular routing is often not possible, you may require 2 layers. But if you have more than 2 layers available to you then possibly you can have more compact or area efficient routing of course, of course, the issue of inter connection across layers comes into the picture, that is another problem that we have to address; and of course, the last point is very important we shall specifically deal with this later, this is we have to complete certain timing constraints.

(Refer Slide Time: 07:05)



So let us now come to the problem of area routing or grid routing. So, what does grid routing say? It says that the layout surface is assumed to be made up of a rectangular array of grid cells.

(Refer Slide Time: 07:19)



So, it is something like this, like I have my rectangular layout surface like this. So, we assume that the total area is divided up into some grid cells arranged in rows and columns let say like this.

Now, all lines and blocks will be placed aligned to these grid cells for example, suppose I want to place a block let say here. So, it will be placed aligned to this grid cell, this is a block I have placed. Now suppose I want to run some inter connection lines, let us take some examples say I want to run a line like this, this is 1 inter connection line. I want to run another inter connection lines like this let say.

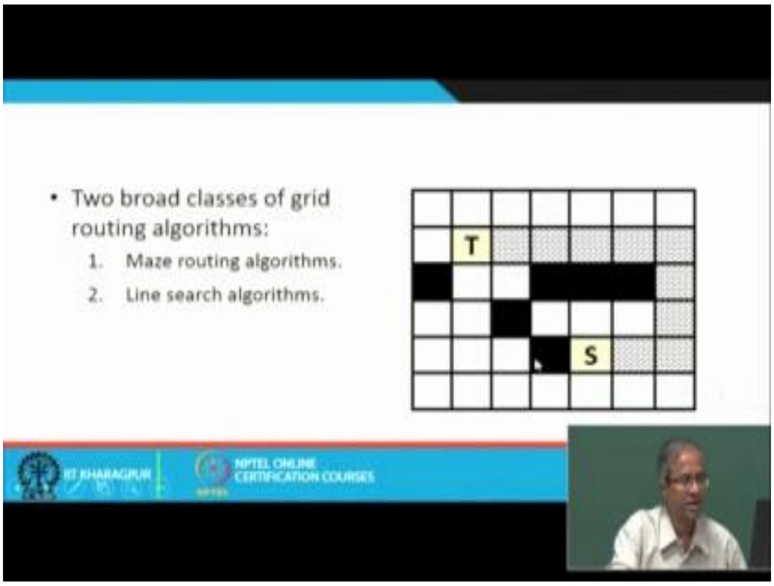
Now, means on the surface of the silicon when you run a set of lines on the same layer, let say they are running in parallel then you have to satisfy certain criteria. Criteria like you have to satisfy the minimum width restrictions of these lines, these are typically expressed in terms of a basic unit of length called lambda that is called the feature size. Let say this width restriction has been separation between the 3 lines can be 2 lines can be 3 lambda. So, these are the constraints that need to be satisfied.

Now, why I am saying is that in this grid representation, the size of each grid; this size of the each grid should be large enough so that it can contain the line that is running through it and also the separation. So, the idea is that if there are 2 adjacent grid cells like this then you can safely run 2 lines on them parallel to each other, without worrying about the width and the separation constraints. The grid cells should be large enough for example; in this example the width of the grid cell will be minimum 5 lambda, so this is the idea.

So, this cells in the grid they represent as either obstacles or as free cell through which we can run the inter connection. Like some of the blocks that are already placed on the 2 dimensional surfaces, they can work as obstacles. So, we shall show them on the grid by coloring them with the dark colour; and also some of the nets which are already laid out they will also be shown as obstacles because for future connection of the nets we should not cross the nets which are already laid out.

Now, in this routing sub problem as I mentioned earlier, we are trying to find out a single layer path; that means, we are not crossing from one metal layer to the other. So, given 2 points on the 2 dimensional grid, we have to find out the sequence of grid cells for connecting from the source point to the destination or the target point.

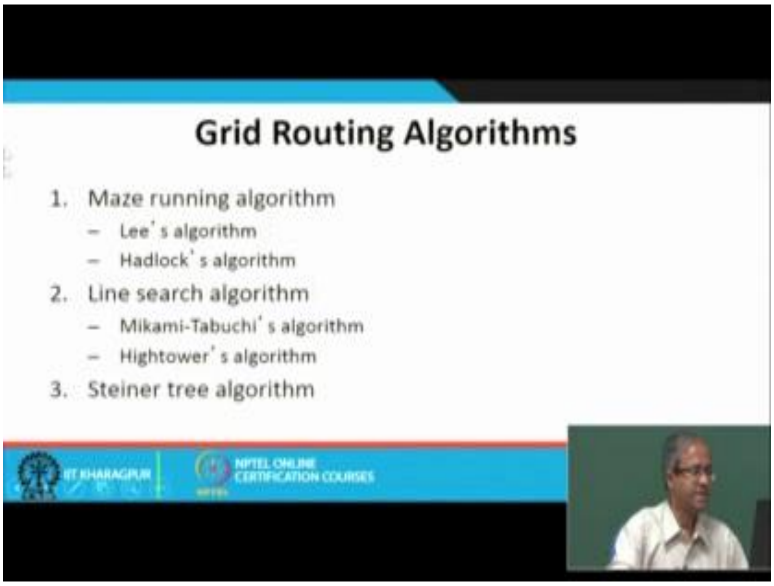
(Refer Slide Time: 11:20)



So, the problem can be viewed like this in this diagram if you see. So, here we have showed a 2 dimensional array of cell, where this dark block represent the obstacle through which you cannot lay the (Refer Time: 11:39). Suppose I have a source and a target point which I have to connect, and let say some routing algorithm has found out a path like this, which is shown shaded. Later on I may be having another set of S and T points, which we may want to interconnect. So, during that time these shaded points will be regarded as obstacles, because they have already laid out a live here. So, these cells are no longer available for laying out the other pairs of lines.

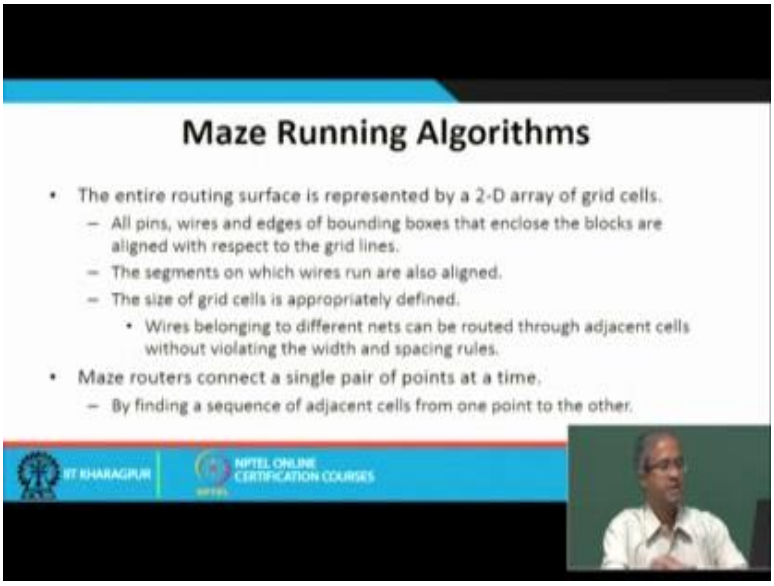
Now, grid working algorithms mainly can be classified as maze routing and line search algorithms as we shall see.

(Refer Slide Time: 12:26)



So, under the maze routing algorithm, where we are assuming that the 2 dimensional space is divided into grids or cells as I have just now said. So 2 commonly used algorithm shall be discussed lees algorithm, and hadlocks algorithm then we shall move towards line search algorithm which we will see that they are more efficient in terms of computation time that may not be that efficient in terms of the quality of solution. So, here also we shall be looking up a couple of algorithms and finally, a brief look at Steiner tree based approaches shall be discussed.

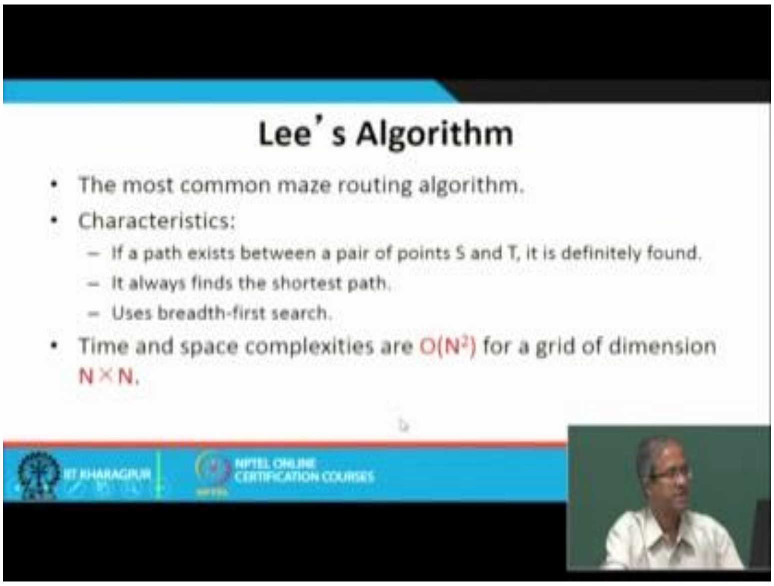
(Refer Slide Time: 13:14)



Now, starting with the maze running algorithms, here the entire routing surface is represented by a 2 dimensional grid of cells; these things are already mentioned you see here we are assuming that all the pins, all the inter connection wires and also the obstacles which are the bounding boxes for the block, they are all aligned with respect to the grid lines. Like if you are having a 2 dimensional array of grid cells, everything will be aligned to the grid lines the horizontal and vertical, you cannot place a block so that it is covering half a cell and half a cell is not covered it is not like that, everything has to be aligned with respect to the vertical and horizontal boundaries of the cells. So, the size of the cells have to be selected in an appropriate way, they should be large enough to means as I said to accommodate the wires when they are running and also the entire wire spacing, so that 2 wires can run on 2 parallel set of grid cells, without any minimum separation constraint violation.

So, the segment on which the wires run are also aligned as I had said, the size of the grid cells are appropriately defined; so that wires for different nets can be routed through adjacent cells. And if we do proper sizing, we are sure that our constraints regarding width and spacing of the lines shall not be violated and these maze routers, they actually connect a single pair of points at a time of course, we shall see later how this restriction can be avoided we can have multi point nets as well. So, let us see the process then we shall illustrate it with the help of an example.

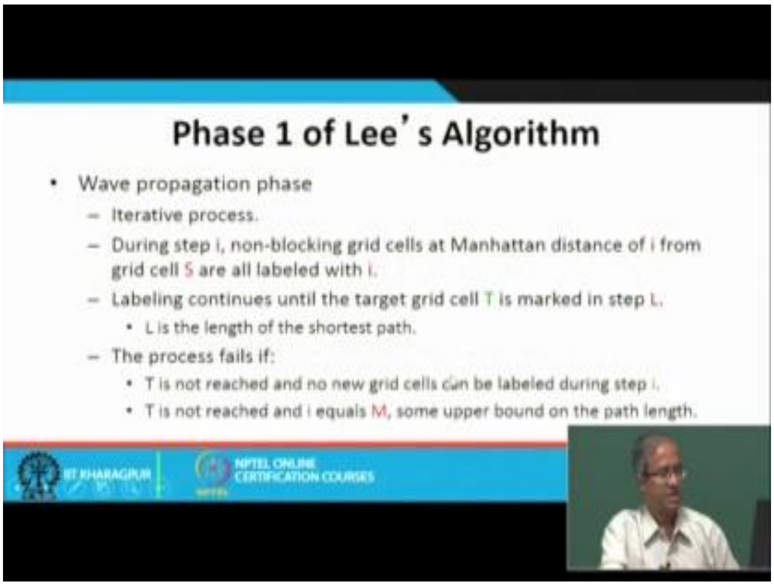
(Refer Slide Time: 15:31)



So, lees algorithm is the most basic and the classical maze routing algorithm, it is very simple in concept and some of the very interesting characteristic is that it says if a path exists between the pair of points S and T, then it will always be found. Not only that lees algorithm guarantees to find the shortest possible path, essentially it uses breadth first search. Starting from the start point S it tries to carry out a breadth first search by exploring all the adjacent cells level by level as if a wave front is emanating. So, imagine a source of a sound you make a vibration, the sound waves start emanating in all directions in the form of wave fronts. As if the wave fronts are moving in all directions and you can go on until or unless it touches or finds the target point. So, once it does that you know that this is the shortest distance of the wave front it should travel. So, as to reach the (Refer Time: 16:49) the T point the target point. So, that you can determine or find out the route through which T and S can be connected this is the basic idea.

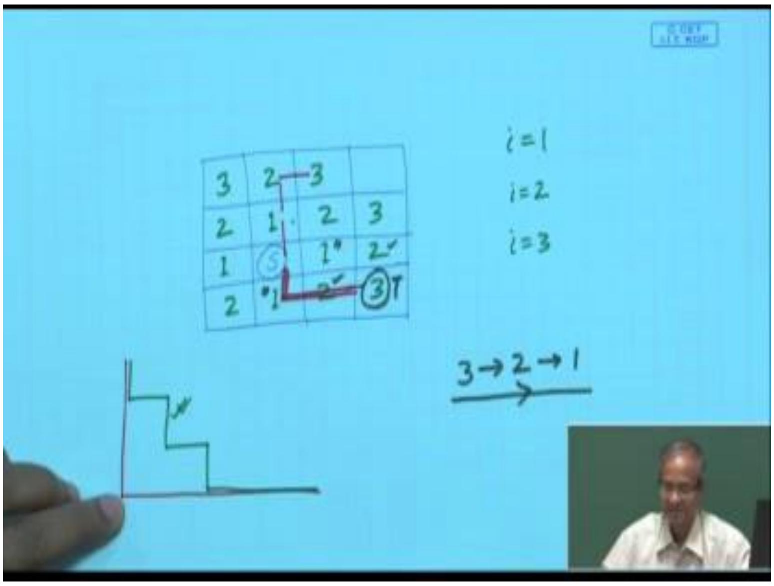
Some of the not so good features of this algorithm is that the time complexity is order N square for an N by N grid. So, you require order N square time not only that also space complexity is order N square, because we are representing the entire grid 2 dimensional grid as a data structure on which you are running the algorithm. So, both time and space complexities are order N square in the worse case.

(Refer Slide Time: 17:38)



So, this algorithm consists of broadly 3 phases; phase 1 says the wave propagation phase this is an iterative process. So, what is done is that, during step i, i starts with 1; 1 2 3 in this way we proceed. So, what we do the grid cells which are at an Manhattan distance of i from the grid cell S are all labeled with i, what does this mean? Let me just illustrate with the help of a simple diagram.

(Refer Slide Time: 18:15)



Suppose you have a simple 4 by 4 grid cell let say my point S is here. So, what this process says is that in spec i, all the cells which are at an Manhattan distance of all we will start with i equal to 1 distance of i from S will be labeled with the value i. So, the 4 cells which are labeled which are like a Manhattan distance of one are here, here, here and here they are all labeled with one.

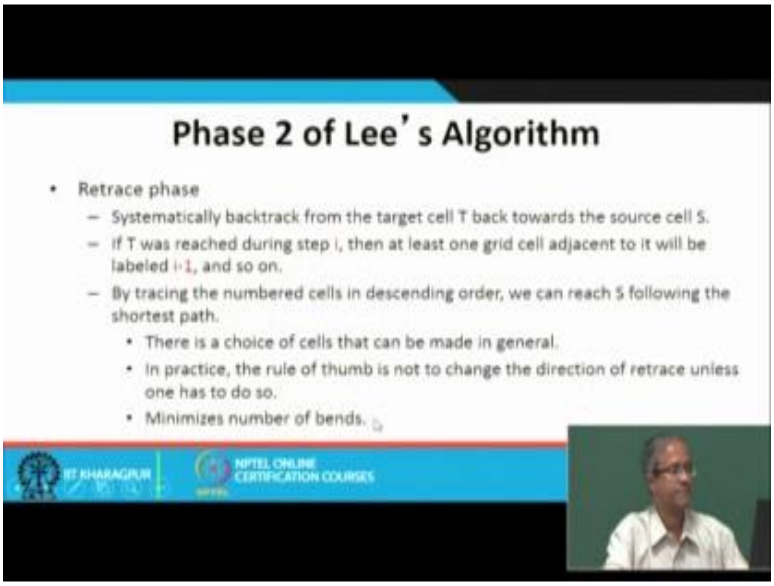
Then we move to the next step i equal to 2; starting with the cells which are marked as on you go to the neighbors. So, the neighbors will represent the cells which are at a Manhattan distance of 2 from S like this, then you move to i equal to 3. You look at the cells marked with 2, go to the neighbors 3 3 3 and 3. So, you see any cell which is marked at 3 as 3 you take any cell let say you take this, it will have a Manhattan distance of 3 up to s take any other 3 1 2 3. So, these labels indicate actually Manhattan distance from the starting point S.

So, you carry out labeling the cells like this, and this labeling continues until the target cell T has been marked; let say we need the L number of steps for doing that. So, the L will indicate the length of the shortest path, because we are doing a breadth first search with respect to the Manhattan distance, we are visiting all cells at Manhattan distance of

1, then distance 2, then distance 3 in this way you proceed; as soon as you reach the cell with label L which means we have already reached, the cell and L is the minimum number of you can say cells you have to traverse to reach there that is the shortest path

So, the process will fail if you cannot reach T and you reach a stage where you cannot label any new grid cell in the next step, which means the path does not exist or secondly, if you if you have some upper bound M with to the length of a path you say, my path should not be greater than M. Then if I see that I have reached M, but still the path is not found then you can conclude that no path of length less than or equal to M is this is the phase 1.

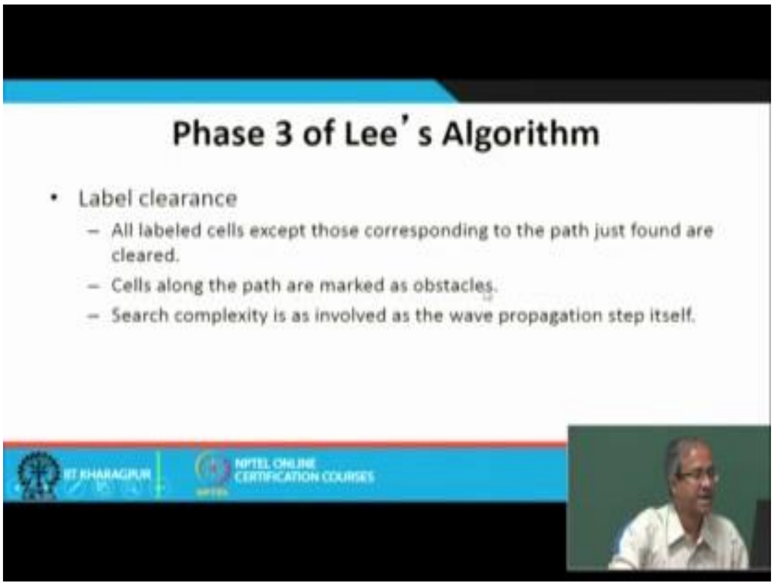
(Refer Slide Time: 21:28)



Now, after we have labeled it, phase 2 consists of the retrace phase. So, from the target cell T we systematically backtrack towards the source cell. Now here one thing you just understand let us look at this diagram again which I have drawn. Let say let us consider this cell, so let us call this was the target. So, once you have a cell with a label of 3, from 3 we will try to go to a cell with a label of 1 less 2. So, here you see that there are 2 alternatives you can either go here or you go here; let say I go here from this 2 again, you go to a cell which is marked as 1. So, here again there are 2 choices either you can go here or you can come here, let say I move here and once I have 1 I know that the adjacent cell of 1 is a s to find out where it is. So, this is the sequence you need to be back traced, starting from a cell labeled with 3 you go to a cell labeled with 2 labeled with 1 and so on and as you can see at each step there can be multiple choices or possibility. So, there is a choice of cells as I said, but you can follow some simple heuristics like you do not try to change the direction unless required; that means, we try to minimize the number of bends.

Like for example, from this T to S you can follow a path like this, but in this example of course, for any path there will be 1 bend, but you can have a scenario where let say from S to T you can have a path either like this or you can have a path like this, both are of the same length, but the second path has more number of bends. Now in various are routing less number of bends the better, because the reliability of the inter connection will be higher in that case.

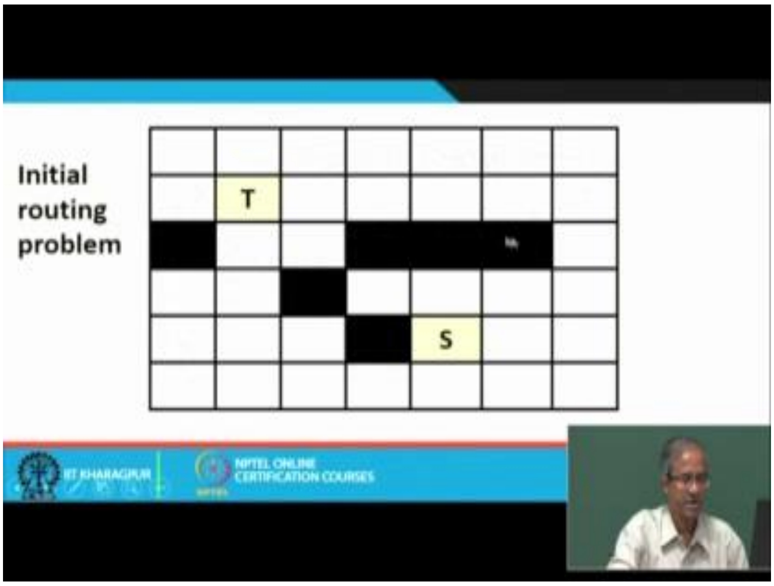
(Refer Slide Time: 24:06)



So, in phase 2 we try to intuitively minimize the number of bends; and phase 3 is once you have found out a path, you clear all the cells that have been labeled during phase 1 and the path which have been identified, you make them as obstacles now because for the next routing step these cells are already laid out and they will act as obstacles.

So, wave propagation process I have tried to explain, the search complexity is the same as the wave propagation process, it goes in a breadth search fashion.

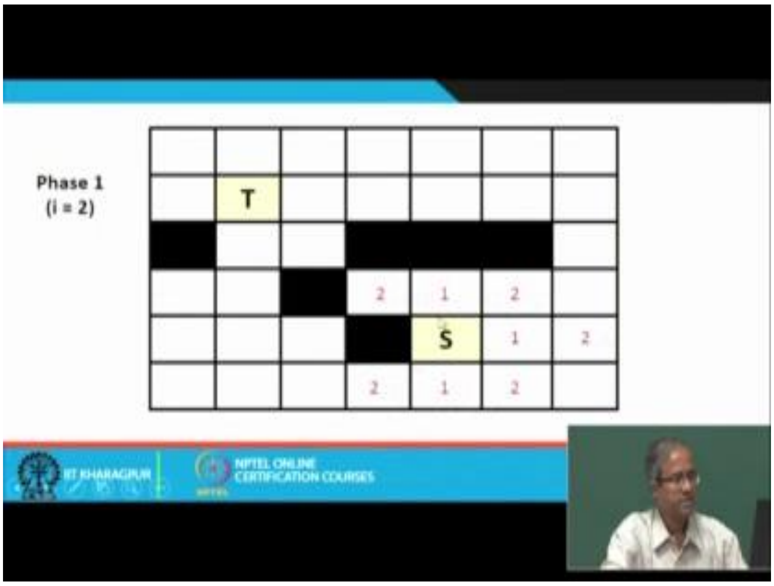
(Refer Slide Time: 24:52)



So, let us try to work out an example, let us take a problem instance like this where I have a set of grid cells: 6 number of rows and 7 number or columns. Some of the cells marked in black are already filled up they are obstacles, you cannot use them for routing this is your source, this is your target and you have to find out a path from the source to the target. So, let us start with the phase 1.

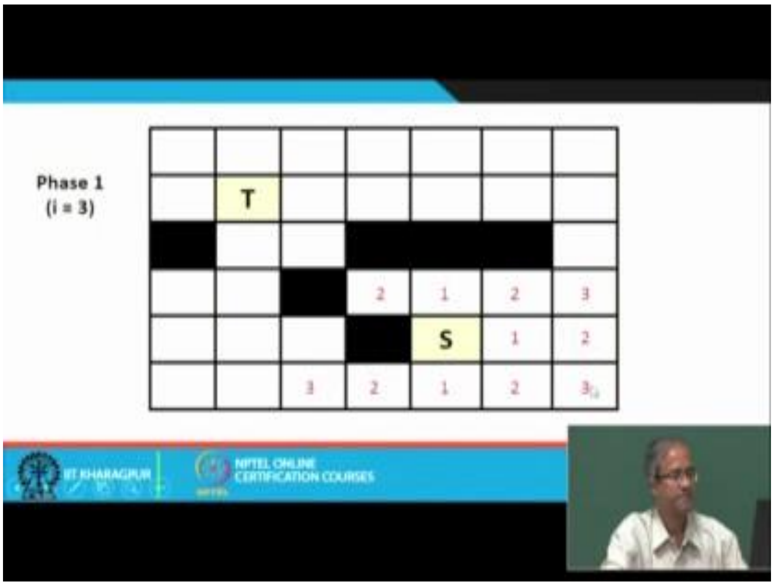
Phase 1 for i equal to 1 as I said we tried to find out the cells which are at a distance of 1 from S and label them with 1, there are 3 such cells. In the next step the neighboring cells of 1 will be marked as 2.

(Refer Slide Time: 25:48)

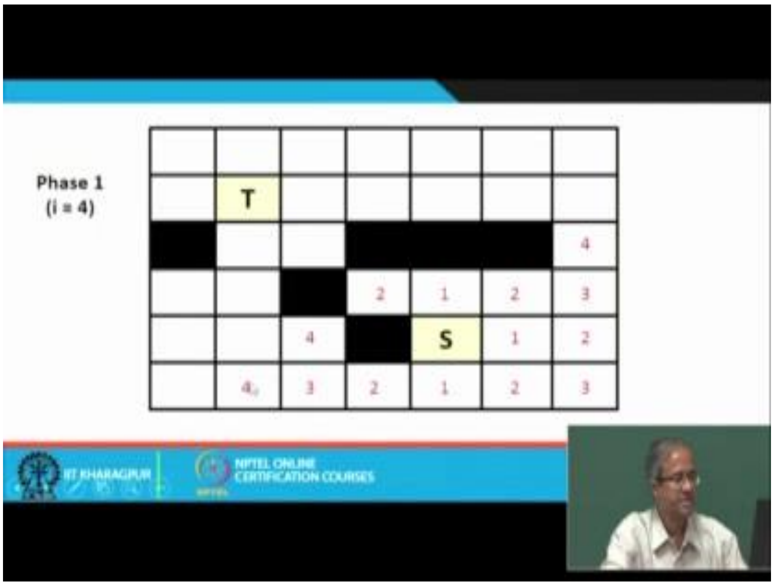


So, you see this cell will be 2, this is 2 this is 2, this 2 and this 2. So, in this way you proceed in the third step the neighboring cells for 2 like it will be in this cell this cell and this cell they will be marked as 3.

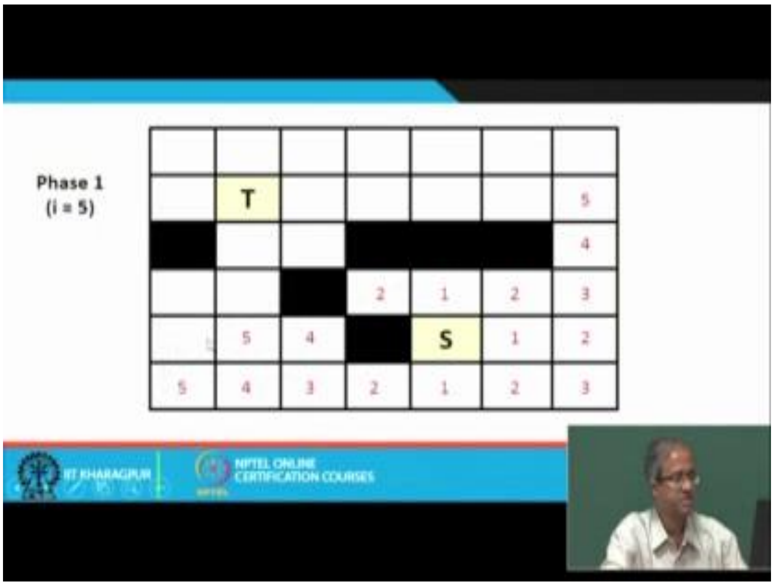
(Refer Slide Time: 26:06)



(Refer Slide Time: 26:12)

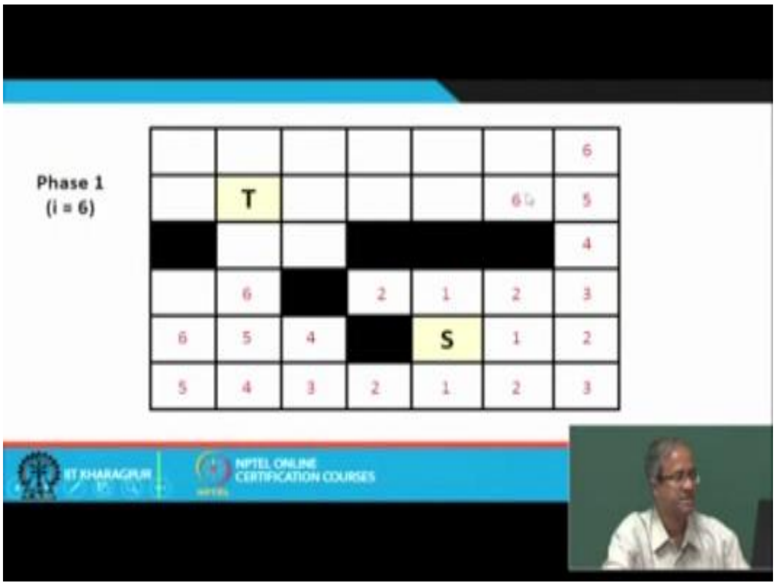


(Refer Slide Time: 26:17)

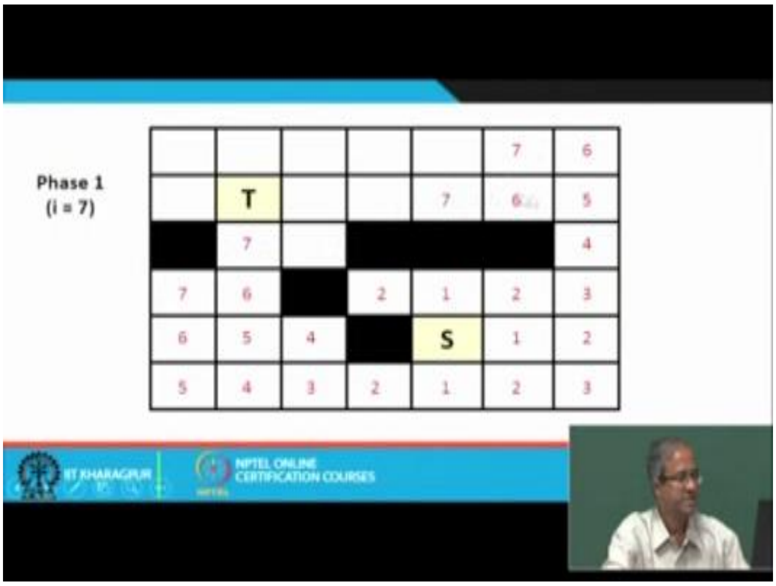


Again in the next step this will be 4, this will be 4 and this will be 4 like this then 5, this will be 5, this will be 5, this will be 5 then this will be 6, this will be 6, this will be 6 and this will be 6 these 4 cells then 7 1 2 3 4, 4 cells will be 7.

(Refer Slide Time: 26:21)



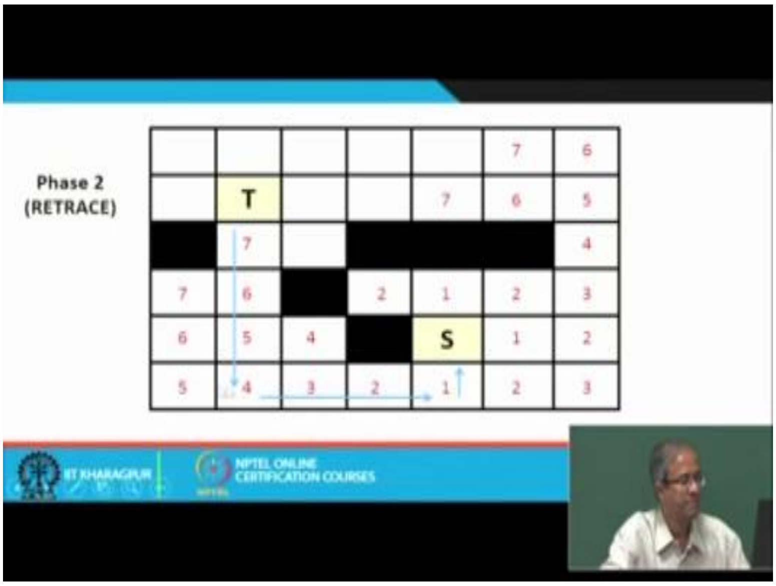
(Refer Slide Time: 26:26)



So, in this way you go on and you see that you have reached a stage where this 7 the cell you have labeled is adjacent to the target. So, you have reached the target. So, from here you can retrace the path from T to S to find out the solution.

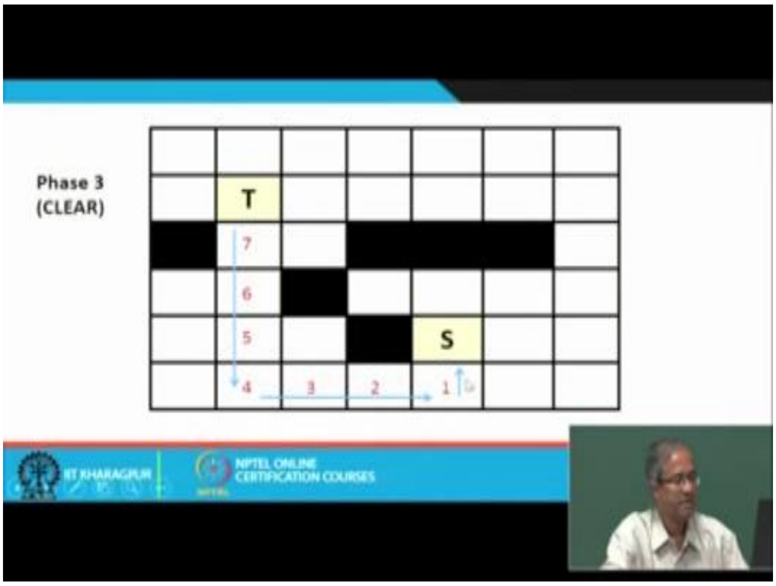
Now, you see there are 2 alternatives either you can follow a path like this, like this, like this or you can follow 7 6 5 then 4, then 3 2 1. So, here you have another done here, but as I said the rule up can be is not to change directions as long as it is not required. So, from 7, we try to find out an adjacent cell 6, which is in same direction like.

(Refer Slide Time: 27:20)



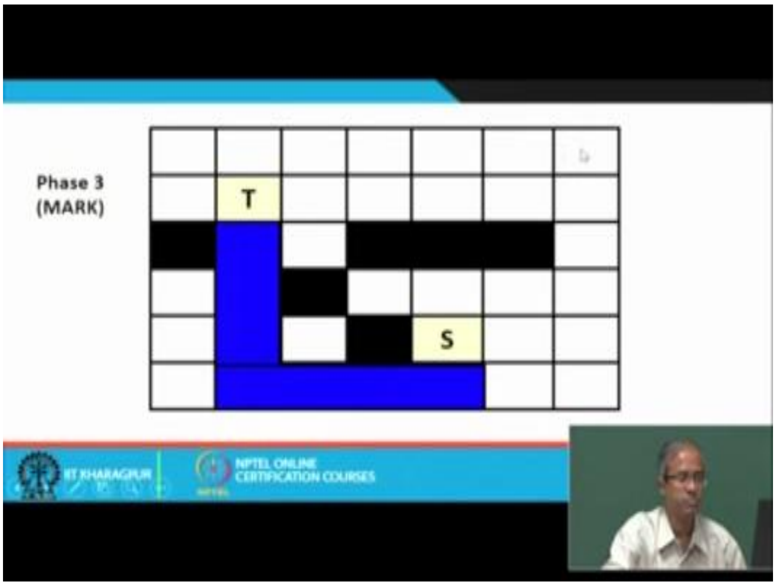
This is the retrace phase then from 6 we try to find out a cell with label 5, then 5 to 4 then you see in the same direction nothing is there then you look for a neighbor with a label of 3, it is on this side 3 then again look for 2, look for 1, then we may s is adjacent so you got a path right. So, this is your phase 2.

(Refer Slide Time: 27:53)



Now, after this phase is completed. So, what you do during the third phase? Third phase you clear all the labels and what I have said that, whatever path you have found out this will be marked as obstacles now.

(Refer Slide Time: 28:07)



Now, in the next step maybe you will be having another pair let say this cell will be your source, this cell will be your target then again we will be starting the wave form propagation from the source where these cells will be marked as target.

Now, see we have discussed a method which is simple in concept and also it guarantees that it will give you the minimum length solution or the shortest path whenever it can find 1, because you are exploring all paths parallely at each step; distance of 1, distance of 2, distance of 3, all the cells which are at a distance of I we are considering in parallel. So, if a shortest path of length one exists after one steps, you are guaranteed to find that path and it will touch that point.

So, in our next lecture we shall look at some issues regarding the memory requirements of this scheme, how much memory it can require some simple calculation and some ways in which you can reduce the time as well as the memory or in storage complexity in this algorithm. So, with this we come to the end of this lecture.

Thank you.