INDIAN INSTITUTE OF TECHNOLOGY GUWAHATHI

NPTEL

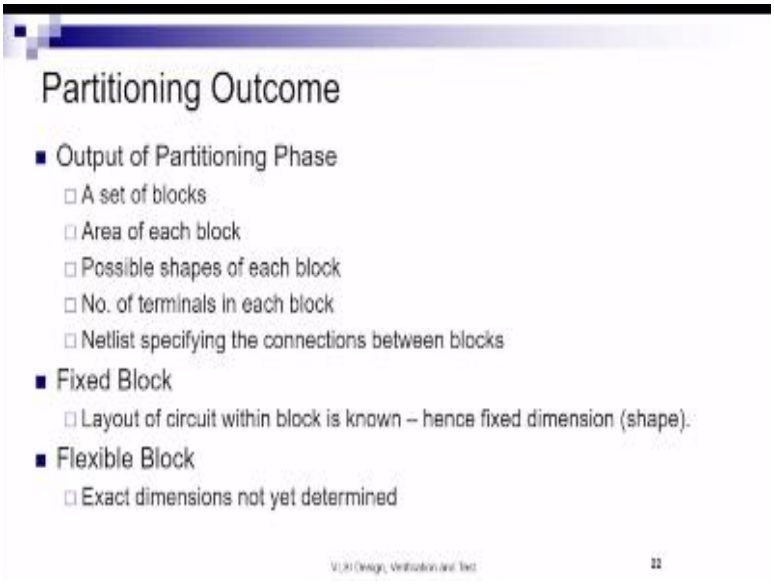
NPTEL ONLINE CERTIFICATION COURSE
An Initiative of MHRD

VLSI Design, Verification & Test

Dr. Arnab Sarkar
Department of CSE
IIT Guwahati

Welcome in the last module we took a look at the partitioning process in physical design automation in this module we will take a look at floor planning and routing at the output of the partitioning phase we have a set of blocks defined because we have divided the circuit into a set of sub circuits each being called a block here we know the area of the block we at least have an estimate of the area of the block because we know the circuit components that will go inside a block and how they will be interconnected and hence we have an estimate of the area of the block we also know the possible shapes of each block and a number of terminals.

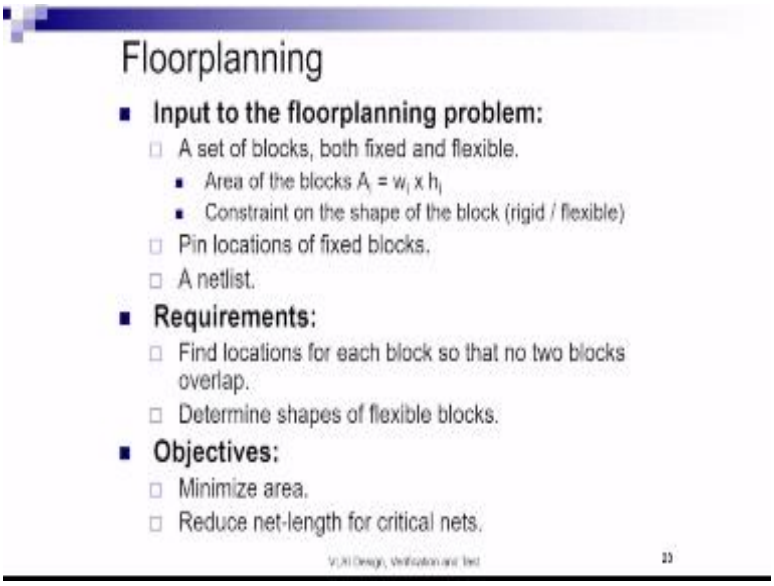
(Refer Slide Time: 01:10)



In each block because we know how many wires will cross the partition of a given partition for many wires will cross the given partition and go to other partitions and hence we know how many terminals that I will require because external connections across a partition can only be done through terminals at the periphery of each block and we have a net list specifying the connections between blocks and we consider two types of blocks here a fixed block a layout of a circuit within a block is known hence fixed dimension.

So if the complete layout of the circuit within a blog is known it is pre-designed then the block is fixed or rigid and flexible block where the exact dimensions are not yet determined we have an estimate of the area but the exact dimensions I had not yet been determined and hence their aspect ratios and sizes can be changed within a given bound.

(Refer Slide Time: 02:20)



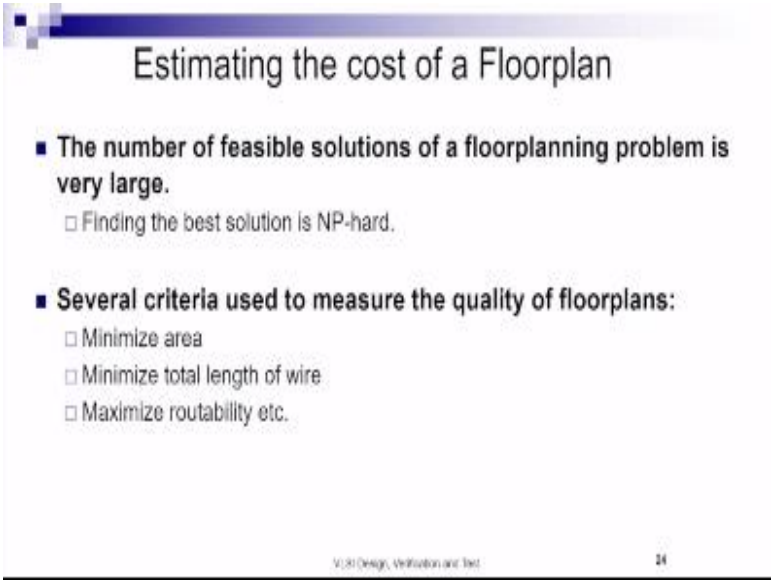
Then what do we do at the floor planning phase the input to the floor planning phase therefore is a set of blocks both fixed and flex the area of each fixed block the area of each block a I equals to wi cross H I is known with cross height the constraint on the shape on the shape of each block rigid flexible is known the pin locations of the fixed block the terminals the locations of the terminals are known and we know the interconnection among blocks as the net list.

What do we require we require to find locations of each block so that no two blocks overlap so we want to obtain a relative placement of the blocks with respect to each other so that none of them overlap okay so what is the objective the objective is to minimize he total area of the layout on the on the chip and to reduce the net length for critical Nets so we can if we place two blocks which have a high which have a high amount of interconnection among them far apart from each other there will be a lot of wires through the chip area long wires through the chip through the through the routing area of the chip connecting them.

And hence it will it will lead to a loss in the area that we can obtain will it lost in the inner it will lead to a reduction it will lead to a loss in the reduction that we can possibly achieve in the area of the chip we also want to reduce the net length for critical nature let us say that we have a net whose length cannot be more than a fixed length some K nanometers and why because if the net length is more than this then my delay the propagation delay through the wire will be such that thus the delay constraint for the signal which will pass through this wire will be violated.

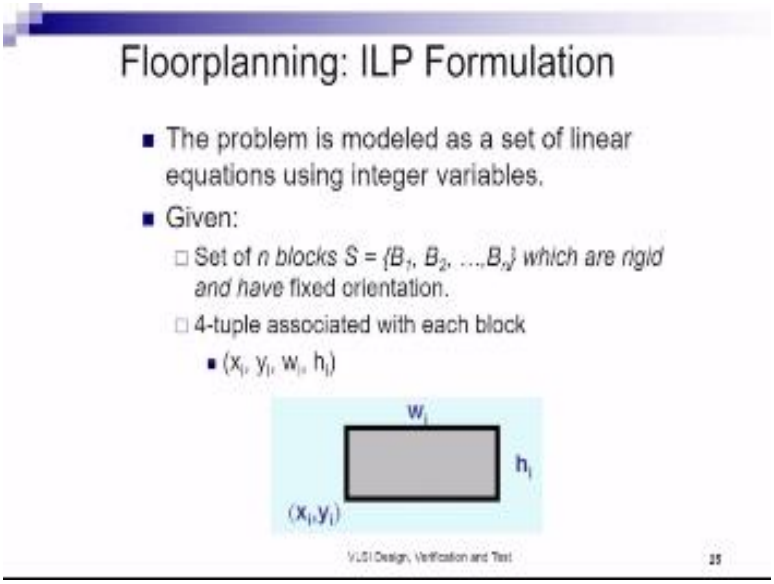
And hence we need to reduce the net length for critical next critical length nets are what they are the net which take the highest amount of the witches of which are the longest nets and hence the signals take the highest amount of time to propagate through them and the performance of the chip is often determined by the length of this critical Nets okay now an additional requirement of the for planning phase is also to freeze the shapes of flexible blocks. So we have flexible blocks we do the floor planning and then we freeze the shapes of the flexible blocks.

(Refer Slide Time: 05:39)



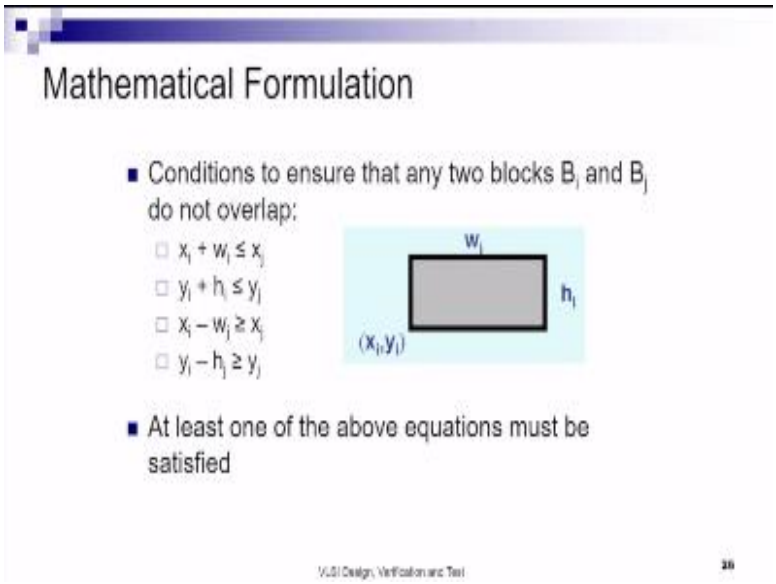
How do we estimate the quality of a floor plan the quality of a floor plan is measured by several different criteria for example minimize area minimize total wire length maximize route ability as we said of the wires through the routing regions the vacant spaces on available on the floor of the chip.

(Refer Slide Time: 06:03)



Now to understand the floor caning problem and the solution approach we will look at its LP formulation so the problem is modeled as a set of linear equations using linear variables therefore and I LP so we are given a set of n blocks s equals to be 1 V 2 dot up to PN which are rigid and have fixed orientation and each block is a for table we know the left bottom corner of the chip what the we know the coordinates of the left bottom corner of the chip we know the width of the chip and we also know the height of the chip.

(Refer Slide Time: 06:46)

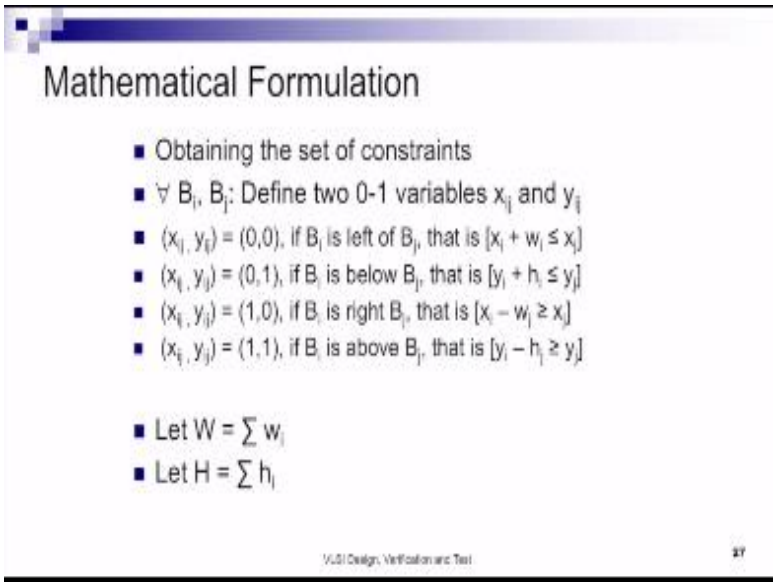


Now to ensure that any two blocks do not overlap so we want to determine now we need to determine what are the conditions that we need to ensure so that any two blocks bi and BJ never overlap with each other now let us say that we have two block BI and BJ now the first one is that let us say that we have XJ to the right of XL if that is so if XJ is to the right of X I then X I +W I must be less than equals to X jail to ensure that two blocks will never overlap then we need to meet anyone at least one of these four constraints here.

So what are they x I plus W I less than x j if x j or bj is to the right of B I on the other hand if BJ is on the top of di then why I + H I should be less than equals to y YJ if BJ is on the top of bi third one if BJ is on the left of bi so X I minus WJ has to be greater than equals 2x j why because bj now is to the left of P I now if BJ is below bi then what happens y i- AJ has to be greater than or equal to YJ.

So these four constraints need to be insured to need to hold in so at least one of these four constraints need to hold to ensure that the low to no 2 blocks bi and pj wi never overlap.

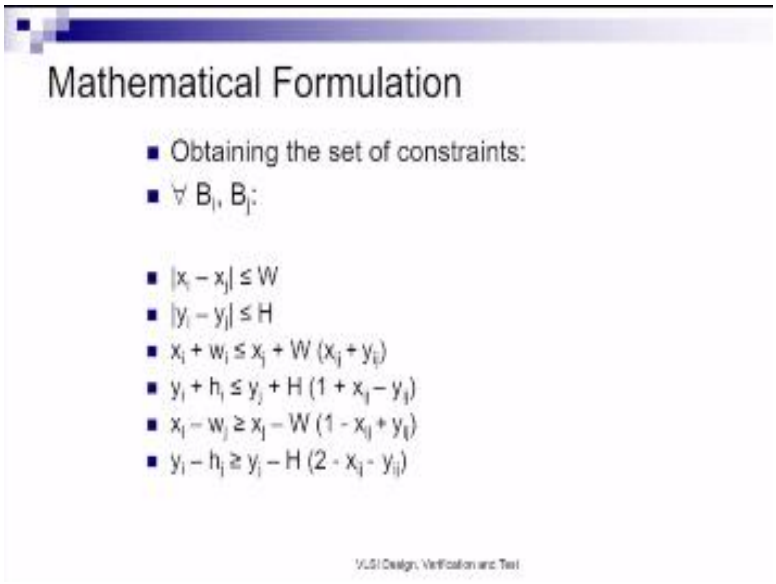
(Refer Slide Time: 08:53)



Now to obtain the set of constraints we define 201 variables exciting and y IJ for each vertex pair for each vertex pair VI VJ we define to 01 variables 0 x IJ and y IJ now excite a comma y IJ equals to 00 if bi is to the left of BJ that is in the first constraint x i + w y less than equals 2x j this is what we need to ensure right we will keep xij comma y IJ equals to 00 x IJ y IJ equal to 10 if bi is below busy excited y IJ is 10 if bi is to the right of bj and xij y IJ is 11if bi is above BJ right and let W be the sum of the wits of all the blocks that we have.

And capital H be the summation let W with let summation WI let W equals to summation WI be the be the summation of the widths of all the blocks that we have and H equals to summation HIV the summation of all the heights of the blocks that we have.

(Refer Slide Time: 10:29)



Now we write the constraints between any two pairs of vertices as follows for each pair bi PJ we need to have X I minus XJ less than equal to W so they must be separated then they must be separate their left coordinates might must be separated by less than equal to W similarly the why I minus YJ less than equals to capitulate the summation of the height so how far they can be apart it can be the summation of the heights right.

So these are trivial constraints which will always need to be satisfied now how do we write the other four overlapping constraints we said that at least one of the constraints Lee to be true so in our constraints one of the constraint as to be true and the other constraint need to be trivially true we will see how we will achieve this so X i + w I less than XJ plus capital W into X IJ plus y IJ we said that when XJ is on the right of X I then what happens then xij comma y IJ equals 200 right so if X i +w I is less than equal to XJ if this is true if this is true and this is what we want that we want.

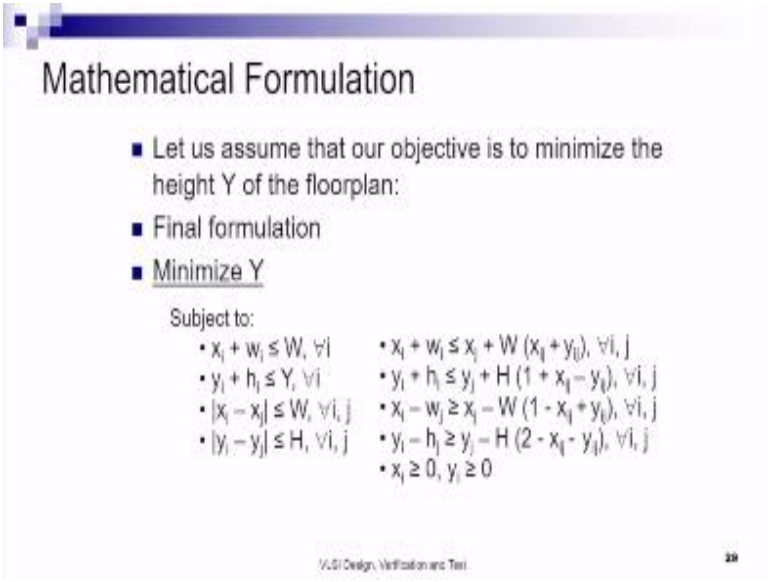
A bj to be on the right of BI if you want DJ to be on the right of bi then we want this constraint X I + w y less than equals to X 0 to be true and it should not be trivially true and see that this is what happens here if this is what we want that bj should be on the right of bi then xij plus y IJ0 0

and hence w into x IJ plus y IJ is also 0 so this will not be trivially true however for the others you see that all the other constraints will be trivially true why xij minus y IJ is 0minus zero is zero so H into one.

So H is there so for a short w the y plus h I less than equals 2y j plus h will always be trivially true right let us and in the third constraint because xij and yIJ are both 0 no W will be there so X I minus capital W will have you will not be equal to 0 so capital W into 1 minus x IJ plus y IJ will not be 0 therefore Xi minus W I x I minus WJ greater than equals 2x j minus capital w will be trivially true.

Similarly for the fourth constraint y i-AJ will be greater than equals 2y j minus 2h will be trivially true likewise we will see that one of the constraints depending on what we want we want a block to be placed on top or one on top of the other on the right of the other below the other or to the right of the other based on that one of these constraints will be non-trivial e has to be non-trivial e true we have to make it true the other constraints will be trivially true and if any one of these constraints are satisfied we know that these two blocks will never overlap right with this understanding we come to the mathematical formulation of the ILB for floor planning.

(Refer Slide Time: 14:33)



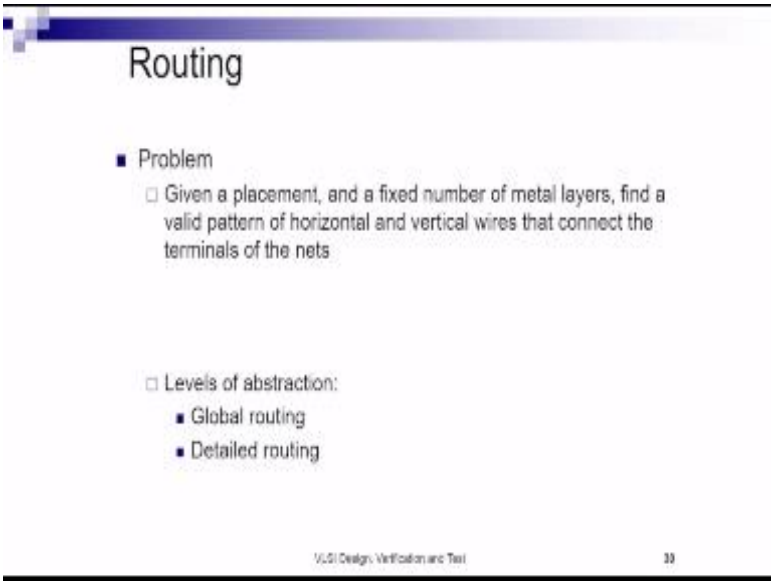
Let us assume that our objective is to minimize the height Y of the floor plan so what will be the final formulation then we have to minimize why the height of the floor plan subject to X I + w I less than equal to capital W so where can W be placed farthest on the right of the chip such that X it ís WI after the width the block should be only placed within the area of the chip and for that X I +w I less than equal to capital W needs to be true why r + H I less than equals to y so Y is what we want to minimize.

So why I + H I so if the where can you can a block we placed maximum on the on the top part of the chip we have to satisfy the constraint why I + H I less than equal to capital y also we saw these other two constraints excites minus XJ less than equals to w and y i-less than equals to H and then we satisfy on these four constraints that we just mentioned in the last slide one of these constraints will be true depending on the relative position of one block with respect to others and it has to be true floor for all pairs of blocks and for therefore for all y IJ pairs these constraints have to be chartered down.

And then we will feed this to an LP solver and the output will be an optimal solution with respect to the constraints that we have the last or constraint of course is that xi xi and why I both have to

be greater than zero right the coordinates have to be greater than zero so with this we understand a mathematical formulation for the floor planning problem with this we understand the ILB formulation for the floor planning problem with this understanding we will now move to the routing problem in physical design automation.

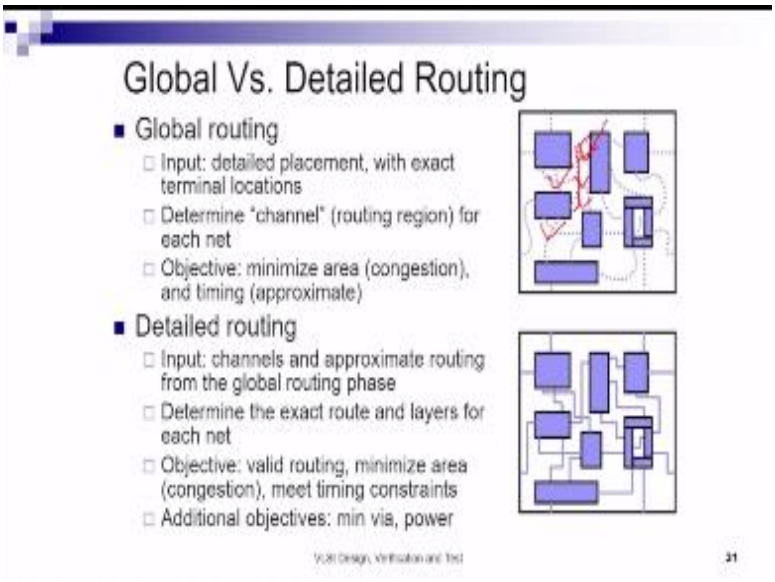
(Refer Slide Time: 17:01)



The problem of routing is as follows given a floor planning and placement and a fixed number of metal layers we need to find a valid pattern of horizontal and vertical wires that connect the terminals of the nets so we have the placed blocks their exact positions known and we know what which wires will be inter connected between these blocks so in the routing phase we need to find actual Manhattan paths through the vacant spaces within the blocks to connect these terminals.

And what are the objectives we may need or we may want to minimize the routing area we may want to minimize the area required for routing so that the total area of the chip finally will be minimized now routing is abstracted at two different levels one is the global routing the other is rowdy.

(Refer Slide Time: 17:57)



So what is global routing in the global routing the input is the detailed placement after the flow planning and placement phase with exact terminal locations known so we know the exact terminal locations of the blocks, now we determine channels for each net so we determined the channels for each net so these are the channels these are the channel these are the empty regions the channels between the blocks through which these routes route on these wires have to be routed.

So determine channel or routing region for each net but we don't find out we do not deal with who where exactly within this channel my wire will be placed so which track within this channel is through which my wire will be placed I don't determine in the global routing case I only find done an enumeration of the channels through which my wire a wire between any two terminal should pass for example let us say now if we consider if we consider this into this one this net this man if you consider this net.

We found that it should go through this channel and then this channel and then this one here and so this is the position it should take there could be other more complicated ways in which you are out a particular route a particular net so the objective is to minimize area or minimize congestion so we need to balance we have a total routing area different regions and we need to balance the

the congestion within each of the different routing regions within the different routing regions that we have and we want to minimize the congestion in these areas.

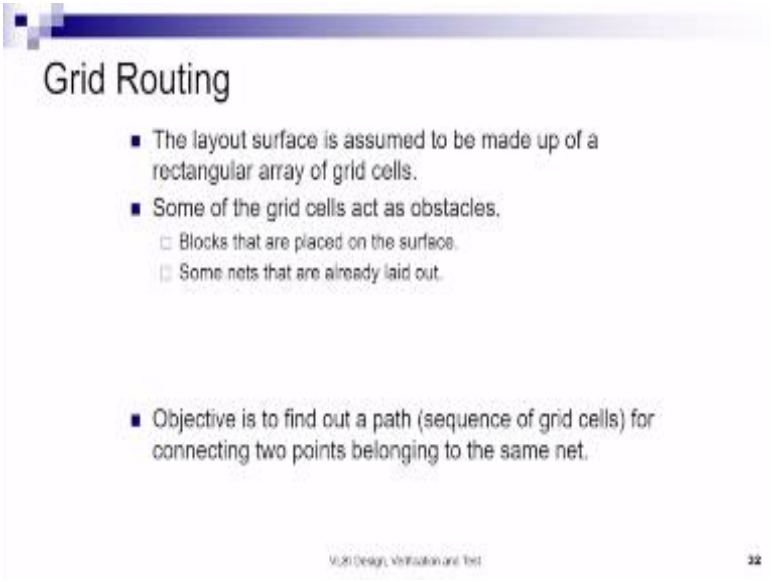
So that the overall area of the routing region will be minimized so it is minimize area or congestion and timing now what we do in the detailed routing phase in the detailed routing phase we do the routing for each channel now at the Google routings after the global routing phase we have known what are the wires that will pass through a given channel so input channels and approximate routing from the global routing phase.

So determine the exact route and layers for each net so through which route and through which layer will this wires pass right will this wires no pass is found out in the detailed routing phase so we take each channel a ta time from the global routing phase we know what are the wires that will pass through this channel and then we find an exact track for each of these wires the objective is to obtain a valid routing minimize area meet timing constraints etc.

Additional objectives could be minimizing vias are the connections between two distinct layers on a chip so now we have different layers on which the cheapest design we have abstracted the little as the cheapest a single layer but there can be multiple layers and there are connections between layers these connections are called fears and routing or wiring is typically done through metal layers.

So there are metal layers and poly silicon layers and through which a wiring can be done and the connection between these layers are called vias and one of the objectives of routing would be mean would be to minimize the number of vias.

(Refer Slide Time: 22:10)

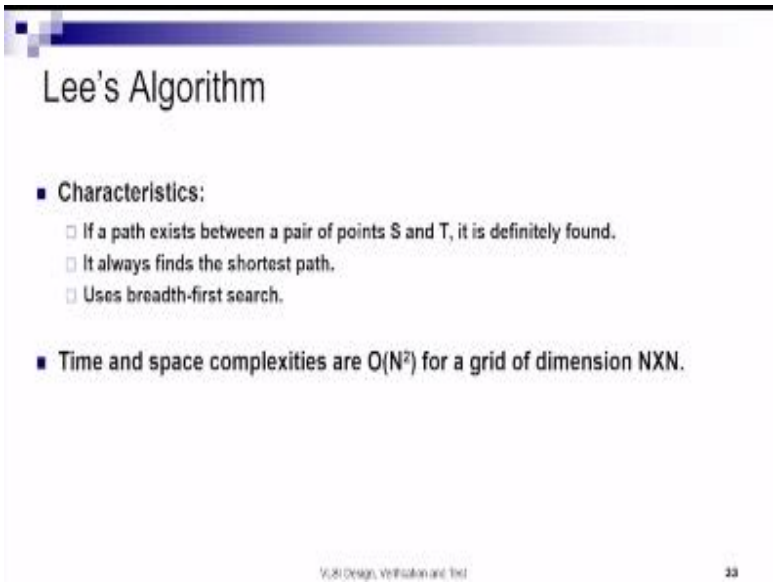


Now we take at take a look at a distinct routing type now we take a look at a distinct type of routing called grid routing it comes within the global routing phase the layout surface is assumed to be made up of a rectangular area of grid cells some of the grid cells act as obstacles which are basically circuit blocks that are placed on the surface and some nets that are already laid out so we will now understand a specific global routing strategy called grid routing.

So in the grid routing the layout surface is assumed to be made up of a rectangular area of grid cells some of these grid cells are as obstacles so what could be the obstacles they could be blocks and replaced on the surface or they could be some net that are already laid out so now we want to lay a new net we cannot bypass we cannot cross these nets because that it will create an electrical short and we cannot cross through the blocks because those are circuit blocks.

So the remaining vacant space we have to do the rerouting to the remaining vacant space available so what is the objective is to find out a bath a sequence of grid cells that is for connecting two points belonging to the same net.

(Refer Slide Time: 23:40)



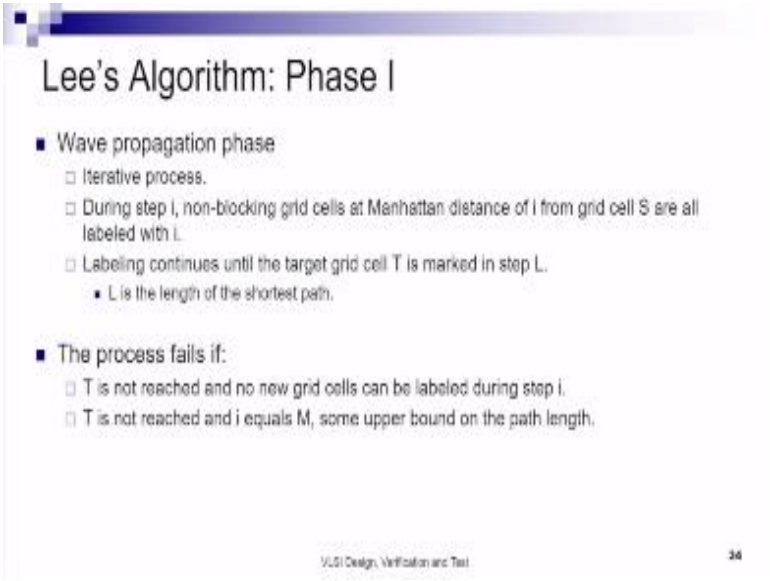
We do this through Lee's algorithm so one of the important grid routing algorithm one of the important grid routing algorithms is the Lee's algorithm what are the characteristics of the knees algorithm Lee's algorithm with respect to the grid that we have produces an optimal strategy so what do we mean if a path exists between a pair of points S&Y for two terminal Nets an optimal strategy if a path exists between a pair of points SNT it is definitely found it always finds the shortest path.

And it basically uses depth-first search the time complexities are Big O of n square for a grid of dimension n crossing so what we do we have in grid routing we assume the floor of the chip to have to be as a grid we assume the floor of the tip to be a grid a grid consisting of n columns in vertical in n vertical wires so we have in grid routing we assume the floor of the chip to be composed of a grid.

The grid has n vertical wires and in horizontal wires in an n cross n grid right so one of the important read routing algorithms is Lee's algorithm in this algorithm in this algorithm we do basically do a breadth-first search to find the path between two terminals to find a route between

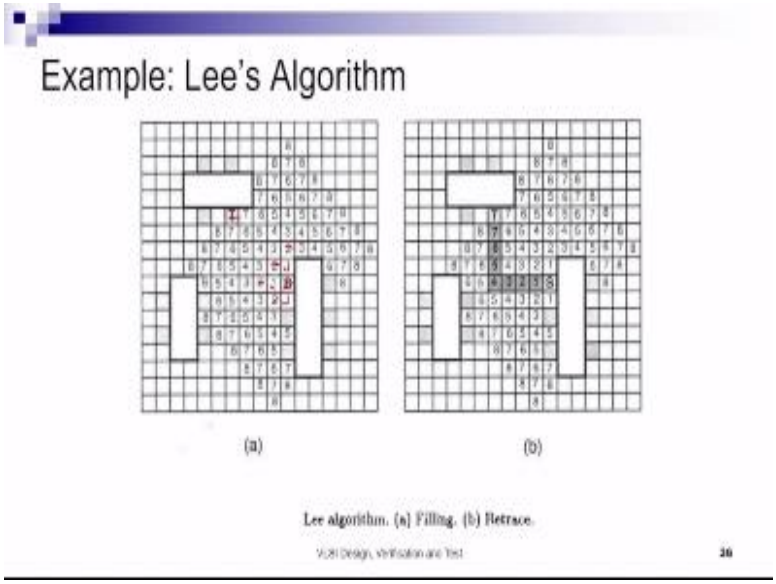
two terminals if a path exists between a pair of points SNT is definitely found and it always finds a shortest path these are these two properties.

(Refer Slide Time: 25:35)



There are three phases to the Lee's algorithm in the first phase we say this to be the wave propagation phase it's an iterative process during step I non-blocking grid cells at Manhattan distance I from grid cell s, s is the source and T is the target so at step I non-blocking grid cells at Manhattan distance I from grid cell s are all labeled with I labeling continues until the target Ritz LT is marked in step l so labeling continues until the target is reached L is l is the length of the shortest path. So it is taken example and see.

(Refer Slide Time: 26:15)



In this example we see that here is the source S is the sources now in the first it we said that it there is a wave propagation breadth-first search and it and it happens in iterations in the first step so if you go to the algorithm in step I non-blocking grid cells at Manhattan distance I from the grid cells are all labeled with I so we see that in step one from matter in the this cell is at Manhattan distance one from is this one is at Manhattan distance one from is this one is also at Manhattan distance one from is and hence these are labeled with one if you see this too is at Manhattan distance to from it this is also at Manhattan distance to form is this is also at Manhattan distance to form is this is also at Manhattan distance to promise.

Likewise we go on propagating on a wave right and at each level I we label cells with I with the number I 4222 under to understand to denote that these cells are at distance Manhattan distance I from the source is and we go on doing this until the target is found for in this case our target it is at this position and we see that the it is it is found in the eighth step week after seven we come to eight so it is at a distance eight from the source.

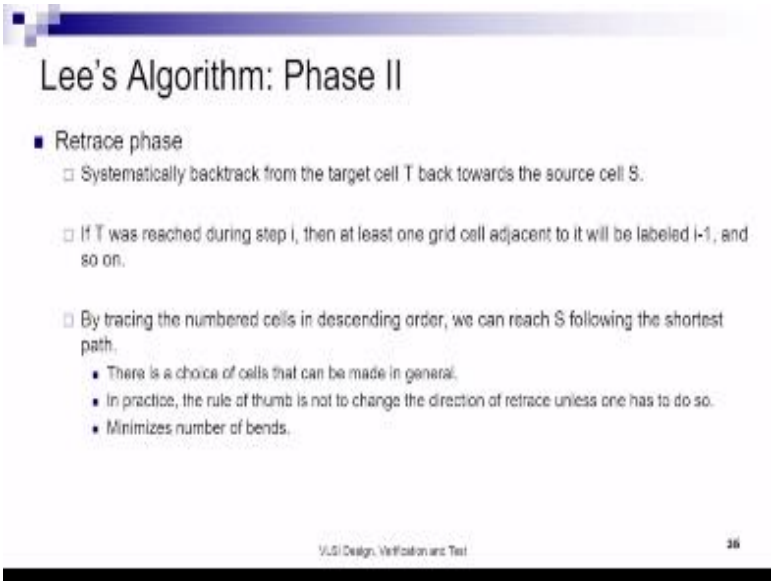
So again it is an iterative process during step I non blocking grid cells at Manhattan distance I from pretzel s are all labeled with I labeling continues until the target grid certainty is marked in

step l l is the length of the shortest path the process fails if T is not reached and no new grid cells can be labeled during step I t is not reached so when to be fail when does the Lee's algorithm fail t cannot be reached no new blood cells can be labeled.

Because it's a breadth-first search it will do an exhaustive search and if the target is not there then it will not be able to find because let us say the target is at a place where it is it is surrounded on all sides by an obstacle if it is not found surrounded by all sides by an obstacle Lee's algorithm is bound to find it right however there is another case in which these algorithm can fail if we bound the number of steps in through which the iteration can progress to at most M so T is not reached an equals M so we cannot do a breadth-first search beyond iteration m then that is some upper bound and we do not find a target T then the process fails.

However if I if I do not have such an upper bound and there is at least one available path to the target Lee's algorithm will always find it because it is an exhaustive search through wave propagation using red first search.

(Refer Slide Time: 29:41)



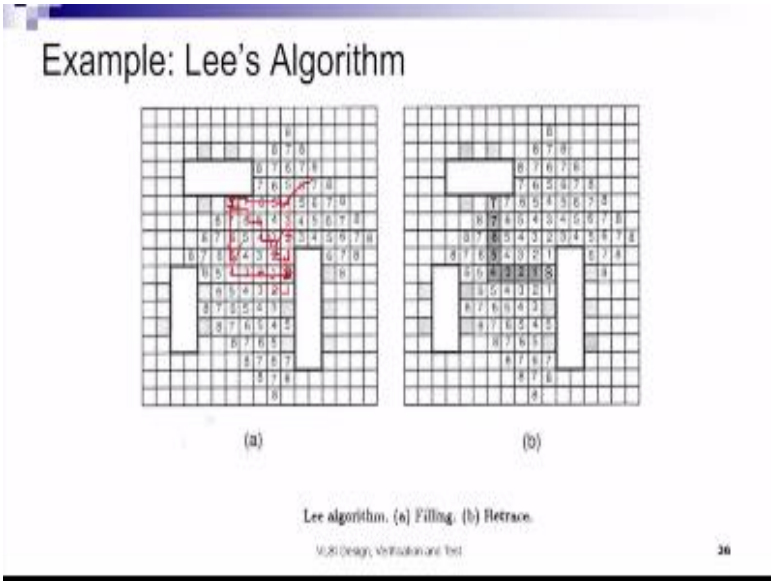
Now the second phase is the retrace face so we systematically backtrack from the targets LT towards the source s right backwards from those backwards towards the source is if T is reached if T was reached in step I then at least one grid cell adjacent to it will be labeled I minus 1 and so on so what happens is that if so now after phase one we backtrack and go back to the source so for example we can take this path.

So there is if T is reached at lake at step8 let us say then there will be at least one with the stem with a step value i-1 or here at 8-1 so we have a seven year so we could take anyone of them let us say we take this one and we progressively go back to the source s so we go back seven six five four three two one zero or we could take seven six five four three seven six five four three two one zero so not this one I have made a small mistake so it will be seven six five four three two one s right.

It could be any it could be any other path rather it could be this also seven six five four three two one is now all these paths have the same Manhattan distance all our shortest paths this is another important thing to note so if T was reached during step I then at least one grid cell adjacent to it will be labeled i minus 1 and so on by tracing the numbered cells in decreasing order we can

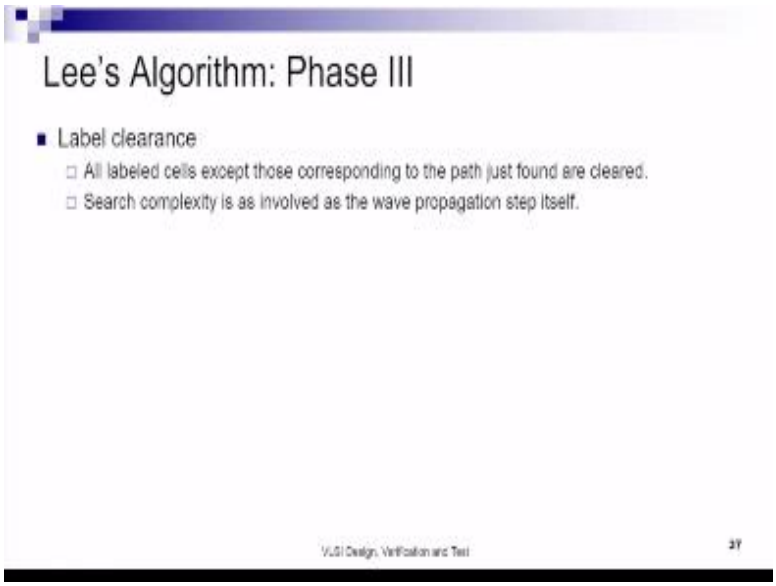
reach as following the shortest path there is a choice of sense that can be made in general in practice the rule of thumb is not to change direction of the retrace unless one has to do. So why because this reduces the number of pens in the wires right.

(Refer Slide Time: 31:56)



Now therefore although all have the same path we will favor this part for example and this part with against this path right will favor this part and this part against this path okay, so this from this was the second phase of the Lee's algorithm.

(Refer Slide Time: 32:15)



In the third phase of the DS algorithm what do we have in the third phase of the Lee's algorithm we clear the labels that we created the redundant flavors that we created all labels cells except those corresponding to the path just found out clearly the third complexity for this clearing process obviously is the same as the wave variation itself because it if that was a wave propagation.

And this will also be another wave propagation to find out all cells which I have labeled in this path to provide a connection between these two terminals in this to terminal net only thing that in this clearance face we have to keep the path that we have chosen in Phase two and delete all other steps so that I can further progress the algorithm for another two terminal net with this small introduction to routing we come to the end of this module.

Centre For Educational Technology
IIT Guwahati
Production

Head CET
Prof. Sunil Khijwania

CET Production Team
Bikask Jyoti Nath
CS Bhaskar Bora
Dibyajyoti Lahkar
Kallal Barua
Kaushik Kr. Sarma
Queen Barman
Rekha Hazarika

CET Administrative Team
Susanta Sarma
Swapan Debnath