VLSI Physical Design with Timing Analysis

Dr. Bishnu Prasad Das

Department of Electronics and Communication Engineering

Indian Institute of Technology, Roorkee

Week - 06

Lecture 30

Wirelength estimation techniques

Welcome to the course on VLSI Physical Design with Timing Analysis. In this lecture, we will discuss about various wire length estimation technique that is useful for VLSI placement. The content of this lecture includes wire length estimation, then we will discuss about maximum cut size, how we can find the maximum cut size. The wire length estimation is one of the most important part whenever you are doing any kind of VLSI placement. So, how we can do that wire length estimation? Let us say for example, we have two pins, we have two pins, two pin nets. So, we can find out the Manhattan distance dm between the two points. Let us say I have one point P1, which is having the coordinate x1, y1 and there is a point P2 which is having x2, y2. Then, how I can find the Manhattan distance is that here in case of VLSI routing, we can place the metal either horizontally or vertically. So, let us say the horizontal metal is basically going like this from here to here. Then the vertical metal is going like this.

So, what is the distance from this point to this point is nothing but y2 minus y1 mod. And what is the distance from here to this point? So, this distance is nothing but your x2 minus x1 mod. So, the sum of these two lines will give me the estimate of my wire length.

$$d_M(P_1, P_2) = |x_2 - x_1| + |y_2 - y_1|$$

So, here the Manhattan means that either horizontal metal or the vertical metal. So, we do not calculate the Euclidean distance at this point. So, the Euclidean distance at this point will be denoted by E of P1, P2 is basically x2, x1 whole square plus y2 minus y1 whole square root over which is basically the diagonal. But we cannot place a diagonal metal in this case. We always look for the Manhattan distance whenever we are doing VLSI placement. Hence this is the methodology for wire length estimation in case of any kind of VLSI routing. So, how we can estimate the wire length in different methods? So, there are different models are there.

So, we will discuss all of these models. So, because each step of our placement algorithm there was a requirement is that what is the wire length after doing this kind of placement. So, the complexity of the wire length estimation technique will indirectly guide the complexity of the placement algorithm. So, here we have different types of wire length estimation techniques are there. For example, first one is the half perimeter wire length estimation SPWL. The second one is complete graph or clique model. Then the third one is monotone chain model. Then the fourth one is star model. The fifth one is called rectilinear minimum spanning tree RMST. Then we have the sixth one is rectilinear Stainer Minimum Tree RSMT model.

Then the seventh one is basically rectilinear Stainer arborescence RSA model. Then the eighth one is basically single trunk Stainer Tree STST model. So, we will discuss each one of the model in detail. So, let us say I have a bounding box with P pins. We have a bounding box global bounding box of the placement with P pins.

It is the smallest rectangle that encloses the pin locations all the pin locations. Then the wire length whatever we will estimate is the half perimeter half of the perimeter of the bounding box. So, let us take an example here. So, here how many pins are I have? I have 1, 2, 3, 4 pins are there. But if I connect the outer pins and create a rectangular boundary bounding box, which is denoted by this one, then the estimated wire length is basically half of the perimeter. So, here half of the perimeter here it is 1, 2, 3, 4 and here it is 1, 2, 3, 4, 5. So, it is half perimeter of wire length becomes 9. So, now we will discuss about the complete graph or clique model. So, the complete graph or the clique whatever how we have already discussed about the clique or the complete graph whenever we discuss graph algorithms. So, the complete graph is a type of graph.

There is an edge from each node to the every other node. So, similar concept is applicable here. So, in this case, each pin is directly connected to every other pin. So, each pin is connected to every other pins. So, basically this is the number of edges of a complete graph. This is the number of edges of a complete graph.

$$\binom{p}{2} = \frac{p!}{2!(p-2)!} = \frac{p(p-1)}{2}$$

Now we can estimate the wire length of a complete graph or the clique model. So, that it will be basically summation of all the dm of the each node, dm will be calculated the way I have explained in the previous slide. Then there is a correction factor is there 2 by p. There is a correction factor of 2 by p is there.

$$L(net) = \frac{2}{p} \sum_{e\ \varepsilon\ clique} d_M(e)$$

So, which will add to actually estimate the length of the wire or the length of the net. So, basically first we will calculate all the dm of each of the edge. Then we will multiply that with a 2 by p correction factor to find out the L of the net. So, this is basically the correction factor. So, now let us take the same pin configuration and find what is the value of the wire length in case of a clique model. So, if you can see here every pin is connected with every other pins actually. So, if you can see here we have for example I will take this node it is connected to this one, this is connected to this one, this is connected to this one and this is connected to also this one. So, in this manner I found out this factor and we will multiply 2 by p, p is here the 4 here. So, it founds out to be 14.5

This is another wire length estimation model. Now we will discuss about the monotone chain model. So, what is the basically background behind or the motivation behind this monotone chain model is that it connects the pins on a net using a chain topology. So, here what is the concept is that every pin has 2 degree, every intermediate pin has degree 2, but the first and the last pin has degree 1. Degree means number of line incidenting to that pin.

So, then we need to what we have to know what we have to do here we sort the pin by either the x or y coordinate and connect them accordingly. So, we will sort the pin by either x or y coordinate and connects them accordingly. So, then what we found is that it is a very simple model, but the thing what we are finding is it overestimates the actual wire length. We can see the model here. This is the same pin configuration what we need to estimate.

So, this is basically the start pin and this is the end pin whose degree, degree is number of edges connecting to that pin is 1. And other pins intermediate pins such as this one and this one has degree equals to 2, but this one and this one has degree equals to 1 because only one edge is connecting to do that pin. So, here if you can estimate the chain length, so then it is found out to be 12. We will start from one node and connect to the other and we will continue like this. So, here if you can see that in this case we have connected the pins according to the x coordinate.

Sorting the first coordinate, the minimum x coordinate is having this, this pin has the minimum x coordinate. Then this is the first pin. Then comes the second x value is increased. Then the third, then the fourth. So, we sort the pins according to the x coordinate in this example.

However, we can do the similar approach for y coordinate also. Now we will go into the star model. So, it considers a one pin which is the source node and all other are the sink nodes. So, it is very useful for timing optimization sinke it captures the direction of the signal how it is flowing from the source node to the all the sink nodes. So, it is very useful for timing optimization and every star model uses only p-1 edges.

It only uses p-1 edges. Basically number of edges is less here. So, it is very useful for modeling high pin count nets. Number of edges in this case is only p-1 edges. So, it can be useful for modeling high pin count nets. So, let us say you have a number of pins is 1000, number of edges will be 999. So, it is very simple, but it always overestimates the wirelength. Let us take how this star model works. So, this is my source pin. This is my source pin and all other pins are my sink pins.

These are my sink pins. So, from the source pin we are connecting to the all the sink pins. For example, if you can see here there is edge from the source pin to this pin, there is edge from the source pin to this pin, there is edge from source pin to this pin. So, here I have 4 pins are there and number of edges is 3 which is p-1, 4-1 is 3. So, in this method if you do the wirelength estimation total wirelength is basically 15. Now we look into the rectilinear minimum spanning tree. We have already discussed the minimum spanning tree in case of graph in the first week of the lecture. Here we are doing the concept of rectilinear means it is not any line, it should follow the Manhattan geometry. Manhattan geometry means either horizontal or vertical lines. So, it always follow Manhattan geometry that is why it is called rectilinear. And earlier we discussed how to find the minimum spanning tree.

The minimum spanning tree we found using greedy algorithms like Prim's and Kruskal's algorithms, what we discussed in our graph theory discussions in the first week of the lecture. So, now we found the minimum spanning tree of a graph using Prim's and Kruskal's algorithm and we exploited the Manhattan geometry to find out the rectilinear minimum spanning tree. Then there is one more point here is that if you decompose the p-nets to 2-pin nets and connects the p-nets with p-1 edges. Why in case of if I have pins why I should have p-1 edges? If I add one more edges it will lose the property of a tree.

So, it will create a cycle. So, if I have a p-pins I should have maximum p-1 edges or connections. Because if I add one more edge then it will create a cycle then it will lose the property of a tree and it is no more a spanning tree. Let us take an example of the rectilinear minimum spanning tree. So, what is happening here is that so here we are creating a basically spanning tree first then we are converting that to a rectilinear version and here if you can see there is an edge between this pin, pin 1 to pin 2 and pin 2 to pin 3, pin 2 to pin 4.

So, the RMST length is found out to be 11. This is close to basically half perimeter estimation model. Now we have a rectilinear stainer minimum tree. Basically here it is connects all the p-nets as many as p-2 additional branching points actually. So, the stainer points are basically the branching points. We have p-2 additional stainer points in case of a p-pins and finding the optimal set of the stainer points for any given set is a NP-hard problem.

So, we will discuss how this rectilinear stainer minimum tree model will give me the results. So, here if you can see here this point is my stainer point. This point is my stainer point because here from here we have branching. So, here we have a branching. So, here the total wire length using RSMT is 10. We will discuss about how to find the rectilinear stainer tree in case of global routing in future lectures. So, now there is a concept called rectilinear stainer arbor sense RSA model. See the RSA model have of a p-pin net is also a tree and it has a single source node. It has a single source node S0 or S0. It connects to p-1 sink nodes actually. It connects to p-1 sink nodes. Then we will find out the path, the distance or the length from this S0 to each of the sink nodes. Okay. Let us say for example, Si is one of them. So, I am finding the distance S0 to Si for each of the sink node.

From the source node to each of the sink node I am finding the distance. Then I am adding all of them. Using the minimum length RSA is also an NP-hard problem. It is also NP-hard problem. For the same configuration, pin configuration, the RSA can be found out. So, this is a source node, source pin and rest all these are sink pins. Okay. So, now what we are doing here, now what we are doing here is that from the source pin S0 to each of the Si I am finding the Manhattan distance. So, from here let us say I am finding from here to here. So, this is found to be 1, 2 and 3. Okay. So, this distance is 3. Now if I have another pin here, so what will happen is that I have 1, 2. So, this is basically 4, 5. So, 3 plus 2 is 5.

Now from here from this point I have 1, 2, 3, 4, 5. So, this is addition of 5 from this point. From this point till this point is 5. Okay. Let us say this is 1, pin 1, this is pin 2, this is pin 3. So, now what is happening is that here what we are doing is that we are finding the distance 3 plus 2, 3 plus 2 plus 5 equals to 10. Okay. So, the rectilinear Stainer arborescence RSA model will give me total wire length to be 10. Now basically we have the last model, wire length estimation model which is basically your single trunk Stainer tree STST model. It consists of a one vertical segment that is trunk and it connects all the pins to that trunk using a horizontal or a vertical segment or branches.

That is very interesting. We will look into an example here. And both RSA and STSTs are constructed in order V log P time where P is the number of pins. Now let us take the same pin configuration. So, if you can see here I have a line. This is the line. Whatever I talked about the trunk. Okay. Either it is a one vertical or a horizontal segment. But in this case this is a horizontal segment. Now all the other pins will have a projection to that segment. So, we have this is one, this is one and this is one.

Now I can find out the total wire length. So, this method give me a total wire length of 10. Now basically we will have several different wire length estimation technique which is plays the heart of our placement algorithm because whenever I am doing any kind of placement I am going and evaluating the wire length how much I am getting if I do this such kind of placement. So, which will give me the best output with means I do not need a very accurate estimation of wire length but I want to do that in less time. So, the time is important. So, what is the most preferred model in basically wire length estimation is the half perimeter wire length estimation HPWL. Why because it is fast. So, this is the plus point. It is fast, orders of magnitude faster than RSMT and equal to the length of the RSMT for 2 and 3 pin nets. Its value comes close to RSMT for 2 and 3 pin nets but the margin of error for a real circuit is approximately 8 percent. The RSMT, the rectilinear tenor minimum tree is basically more accurate but it takes more time to find out the wire length. But in case of HPWL it is faster but there is error of 8 percent. So, there is a trade-off between the time it takes to evaluate versus the accuracy.

So, if you need more accuracy you can go for RSMT. If you want to fast evaluate the wire length you should go for the HPWL. So, if you can see here the RSMT which is giving me the wire length of 10 my HPWL model give me a wire length of 9. So, here the error is close to 10 percent. So, we discussed about the total wire length and what we are giving here is that basically the priority to each of the nets actually.

So, here what we are doing priority to certain nets. So, here how we are giving priority basically we give weight assigned to each of the wire. So, the total weighted wire length is basically L of P is basically W of net into L of net. W of net is the weight of the net basically how much priority I am giving to that net and L of net is the estimated wire length of that net. So, now if I have a placement P of block A to F and their pins are given here the nets basically N1 to N3 and their net weights are given here.

So, what we need to do here is that I need to find out the L of P. So, what I need to do estimate the total weighted wire length of P using a RMST model. So, what is the basically given here L of N1. So, L of N1 is basically I have DM of A1 to B1 plus DM of B1 to D2. So, this is the sum of DM of A1 to B1, DM of B1 to D2. So, here if you can see this is 1, this is 2, this is 3, this is 4, hence this is 4 here. So, this is 1, this is 2 and this is 3 here. So, these 3 corresponds to these 3. So, L of N1 is found to be 7. So, basically similarly I can find for L of N2, L of N3 then L of P. So, whenever I am finding L of P I have W of N1 which is given in the previous slide. W of N1 is 2, W of N2 is 4 and W of N3 is 1. So, if I substitute here then my L of P is found to be 32. Now we will discuss about the maximum cut size. So, here what is happening here is that basically if you can see you have a uncut net. So, what is uncut net? If I have a pin in L, so I have this is my total area, this is basically the vertical cut line. So, this one is your this basically this line is my vertical cut line, this is a vertical cut line and I have a pin here in the left hand side, I have a pin here in the right hand side. So, this L is left and R is right. So, if there is a net connecting the pin from L to R then it is a cut net because it is cut by the vertical cut line here. So, cut net has at least one pin. If I define a net as a cut net, so it has at least one pin in each L and R side at least it can have more pins also. And if it is a uncut net, let us say I have another pin here and let us say this is connected with this one.

So, this is basically uncut net. This is very simple. Now I will define two parameters here. Basically one is called V of P, one is called H of P. So, the V of P is called the global vertical cut line, V of P is vertical cut line and H of P is the horizontal cut line for the placement P. Now we have two things, one is the set of the nets cut by the cut line, basically this cut. So, this is the cut net. So, the set is represented by this Ψ P of cut, upper case of the Ψ . So, this is the upper case of the Ψ . And this is the upper case, do not confuse, upper case will give me the set of the nets. And the lower case of the Ψ will give me the number of cut nets cut by the cut line. So, your Ψ of P, the lower case Ψ of cut, Ψ of P is basically the cardinality of that set. So, now we can define X of P and Y of P are the lower bounds on the routing capacity needed by the horizontal and vertical directions. So, in the horizontal direction we have X of P, how many routing capacity is there? Y of P in the vertical direction. So, which is found out from the number of vertical cuts and horizontal cuts. The vertical cuts will give me X of P and the horizontal cut will give me the Y of P. So, you can find the max V of P, lower case Ψ P of V will give me X of P and similarly Y of P.

So, we have two types of design methodology, one is the gate array and standard cells. In case of gate array your horizontal and vertical tracks are already predefined actually, already predefined because whenever you manufacture that lines are already made. So, basically this X P and Y of P are within this capacity. We cannot route something more than that. But in case of standard cell based design it is basically the designer is designing that X of P and Y of P are given the lower bound on the demand. Basically what is the lower bound on the demand of the horizontal and vertical routing tracks. So, now here what we are doing is that to improve the total cut size separately calculate the number of crossing. So, we need to calculate the number of crossing of global vertical and horizontal cut lines and minimize that. So, here what we are finding is that what is the number of cut in the basically vertical direction, what is the number of cuts in the horizontal direction and we are minimizing this L of P.

$$L(P) = \sum_{v \, \varepsilon \, V_P} \psi_P(v) + \sum_{h \, \varepsilon \, H_P} \psi_P(h)$$

This is my objective is to reduce the number of cut. So, let us take an example here. This is the placement. The placement of P is given and these are the blocks. I have three nets N 1, N 2, N 3 and I have in case of vertical I have two cut line V 1 and V 2. In case of horizontal I have H 1 and H 2. Now all the nets are given N 1, N 2 and N 3 is given how they are connected. Now what is the goal is to determine the cut size X of P and Y of P for a placement given here using this RMST model. Now what is happening here is that if you can see this lower case will give me the number lower case Ψ will give me the number of element in that set. So, if I have this one, this is my V 1. If this is my V 1, how many lines is cut by this V 1? So, this is 1.

So, this one corresponds to this one. Now Ψ of V 2, if I draw the V 2, this is my V 2. So, how many nets are cut by the V 2? This is 1 and this is 2. So, that is why it is 2. Similarly for H 1. So, H 1 is this one. If I do H 1, then this is 1, this is a 1 cut, this is the second cut by the H 1 and this is the third cut.

This line is the third cut. Now that is why this is 3. Now H 2, H 2 is this one. So, if this is the case, this is 1 and this one is 1. So, that is why Ψ of P of H 2 is basically 2. Now I can find the total number of crossing. Total number of crossing is just add all of them. This is 1 plus 2 plus 3 plus 2, it is basically 8. So, now in X direction what is the maximum? What is Y direction maximum? So, whenever I am finding X direction, I need to consider the vertical cuts. I need to consider whenever I am finding the X direction because these are X direction. So, in case of X of P, I have V 1, I will consider V 1 and V 2 which is giving me the maximum cut. So, this if you can see at this net and this net is giving me the maximum cut. That is why X of P is 2. Now I have Y of P. Y of P I have considered in the horizontal cut. So, this is my horizontal cut H of 2 and H of 1. If I go here which is giving in the maximum cut. If you can see here, this is 1, this is 2 and this one is 3.

So, Y of P is giving me 3. So, this is basically the total X of P is 2. Here the X of P is basically 2 and Y of P is basically 3. Now I just do a slight change, placement change and it will reduce the cut cost. So, if you can see here, I move the block B, move the block B little bit in the vertical direction.

You can see the previous, here it is located here. In this case, this B is located here. Now I move this one in the vertical direction. Now what we are doing is that now we move that B block in the vertical direction. What is happening here is that if I move this one in the vertical direction, then the number of cut is reduced by 2.

So, if you can look into this H1 is now cutting only one net. Earlier it was cutting 3 nets. So, it is just cut H1 is cutting only one net. This is the only one net it is cutting, but earlier it was cutting 3 nets. So, placement plays a vital role in determining the max cut sizes in a partition whenever you have vertical and horizontal cuts. So, it is very important how the placement plays a vital role finding the maximum cut size. That is the reason we need to estimate the cut size and do the placement such a way that number of cut will reduce in the final placement.

So, in this lecture, we discussed about several wire-length estimation techniques.

Thank you for your attention.