VLSI Physical Design with Timing Analysis

Dr. Bishnu Prasad Das

Department of Electronics and Communication Engineering

Indian Institute of Technology Roorkee

Week 05

Lecture 27

Floorplanning Algorithms - II

Welcome to the course on VLSI Physical Design with Timing Analysis. In this lecture we will discuss about some more flow planning algorithms. So, the content of this lecture includes some of the flow planning algorithm which is useful for VLSI physical design. So, we will discuss about linear ordering algorithm and then we will discuss cluster growth algorithm and finally we will discuss about the simulated annealing how it is useful for flow planning. So, linear ordering algorithm is employed for generating the initial placement. So, it is used in iterative improvement of placement algorithm.

So, the objective of the linear ordering algorithm is to how we can place the blocks in a single row such a way that we can minimize the total well length between the connecting blocks. So, here there are some terminologies used in case of linear ordering. So, there are three types of nets one is called terminating net one is called the new net and the third one is the continuing net. So, what is the difference between each of them? We will discuss with one on one example.

So, the terminating nets have no other incident blocks that are unplaced means that this whenever we discuss about the terminating net here for example the N1 and N2 is terminated in the block C. So, it is not going after block C. So, when we place the block C then the nets N1 and Nare called the terminating nets. Then what is new net? New net has no pins or on any block from the partially constructed ordering. So, what does that mean? So, first we place the block A then we place the block B then we are placing the block C.

So, if you can see in this diagram the N3 and N4 is started from the block C and it has no connection from the previous partially constructed ordering. So, that is why this N3 and N4 are called the new nets. Then there is a terminology called continuing nets. The continuing nets have at least one pin on a block from the partially constructed ordering

and at least one pin on an unordered blocks. So, partially constructed ordering means the A and B are called the partially constructed ordering.

So, it has partially made. So, then the basically the continuing net has at least one pin on a block from the partially constructed ordering. So, if you can see here the N5, N5 has some nets on B. So, it has some nets on B. So, and it is also continuing to the next unplaced block or unordered blocks.

So, the blocks which is coming after the block C are called the unordered blocks. So, that is why this N5 is called the continuing nets. So, we will have a linear ordering algorithm. What are the inputs to that algorithm? It takes a set of blocks set up all the block M and the outputs are the ordering of the blocks order. Basically, we have set of blocks and connection between them the nets between the blocks are given to us.

Then we will place the block such a way that our interconnect distance will be minimum. So, how it is possible? We have a gain formula. So, the gain at this point is defined by this one. So, the gain of a block is equals to terminating nets minus new nets. So, how many nets are terminating minus new nets.

So, we will take an example then we will come to the algorithm later. So, we have these are the given to us. What are the things given to us? We have a net list with the 5 blocks A B C D E are given. Then we have to assume one of them as a starting block. Let us say the block A is the starting block.

So, that is given to us and the net list are given by these statements. So, what it says that I have a net N1 which is connecting between block A and block B. So, let us say that N6 take an example of this N6 what it is telling is that there was a net or interconnect or a wire between the block C and block E. So, now this is our initial placement. So, this is our initial placement what is given to us.

Now we need to find the placement of the blocks such a way that our interconnect distance will be minimum. How can you do that? So, find a linear ordering using the linear ordering algorithm. So, how I can do that? So, first block so this information what is given to us with that we have created the placement of the blocks but how I can do the optimization in the initial placement of the blocks. So, we will start with the first block block A is the initial block we are placing. So, if I go to the block A what are my new nets N2 and N1 is my new net.

N2 and N1 are my new nets. So, our first block is the block A the block A is the first block. We assume that the block A will be placed first. So, what are the new net and at this point the N1 and N2 are the new nets and what is the terminating net? Since it is the first block there is no terminating net. So, our gain is basically new net minus the terminating net.

Hence my gain is minus 2. This is a gain in the first iteration. So, this is a gain at the first iteration. Then I will look into if I place my next block B after A so what are my new nets? So, if I place the block B after A the N3 is my new net and the my gain is basically minus 1 and my continuity is N1. Let us go and check that diagram.

So, if B will be placed here then the N1 is my continuing net. My N1 is my continuing net. And basically what are my new net after B will be placed the N3 is my new net. We will go there N3 is my new net and N1 is my continuing net. So, similar approach I need to do. So, if my C will be placed next to block A if block C will be placed after block A what are my new nets? So, if the block C will be placed after A then the new nets will be your N4 and N5. So, if you can go here this N4 and N5 are my new nets. And what are my continuing net? Obviously this N1 is my continuing net because this N1 is connecting between A B and C we have seen in the previous diagram. So, now similarly we can do it for D and E and we can find the gain for all of them. So, I need to choose the block which is offering me the highest gain.

In this case the block B is giving me the highest gain. After block A the block B will be placed. This is my placement of the block after A the block B will be placed. So, now we are going to the second iteration. Our first iteration is over.

This is the placement of the block after the first iteration. Now we are moving to the second iteration. In the second iteration our block A and B are fixed. So, we need to find out our gain for rest of the block C D and E. So, for C if I place the block C after A B then my gain is minus 1.

If I place the block D then my gain is minus 1. And if I place the block E then the gain is minus 2. So, I have two choice either C or D both are having the same gain. So, we have basically two possible solutions are there either C or D because both of them having the same gain. But if you can see if I place the block D then it has one continuing net but the block C has no continuing net.

So, the gain which has having the lowest gain as well as have continuing nets that will be chosen. So, after A and B we need to place the D. So, we have this N3 continuing net is there for the D. So, why it is useful because if I place the C here in place of D then the length of my continuing net will increase. So, length of my continuing net will increase.

To avoid that I need to place the D first such that my distance of N3 will reduce. So, now we have the third iteration. So, we have already placed A B and D. I have two more blocks block C and block E. So, for block C and block E I need to find out my gain. So, the block C has gain of 1 block E has gain of 1. So, I can choose any one of them because both are having no continuing nets. So, you can choose any one of them. So, let us we will choose the block E here. We can choose the block E here. After I took the block E now I have the last block C is left. So, which is having a gain of 3 then it has no

continuing net and there is no other option. So, only one option is there. So, I have to choose the block C only. So, this is the final linear ordering of the blocks which is basically very much optimized in terms of wire length. So, this is called the linear ordering of the blocks which is used for the floor planning.

Now we will go for the second algorithm cluster growth algorithm. So, it construct a floor plan by adding the blocks iteratively. So, it will create the floor plan by adding the blocks iteratively until all the blocks are assigned. So, now let us start with the initial block.

We have an initial block is there. In this case we have chosen block A which is chosen to which is placed at lower left corner LLX LLY lower left corner. So, then we need to subsequently add one block at a time and find out the area such way that my overall bounding box of my overall floor plan will be as minimum as possible. So, in this case we are adding one block at a time. And it is most horizontally vertically or diagonally with the cluster. It can be added horizontally, it can be added vertically or diagonally with the cluster.

So, the next blocks placement is based on the current cluster shape and objective function. What objective we want to look for it. Let us the block B is placed let us the block C is placed. So, this is my overall bounding box global bounding box.

This is my overall global bounding box. So, this is an example of a cluster growth algorithm. So, here in case of cluster growth algorithm only the different orientation of the individual blocks are taken into account. So, the order in which the blocks are added is typically determined by the linear ordering algorithm. We have already studied the linear ordering algorithm then the placement of the blocks are determined by the linear ordering algorithm then we will take one block from there and place it using the cluster growth algorithm. So, we will see how we can add one of them using the cluster growth algorithm.

So, this is the pseudo code of our cluster growth algorithm. What is the input to this pseudo code of the cluster growth algorithm is that a set of blocks M and a cost function C then final output is basically optimized for plan F based on C based on the cost may be area wire length or whatever we want what is our target. So, first we run that linear ordering algorithm whatever we discussed earlier. So, it will generate the placement of the blocks means after A which block will come this order will be determined by the linear ordering algorithm. So, then we will take one block at a time and we will place add to the floor plan.

So, we will take one block at a time and we will add that block either horizontally or vertically or diagonally to the floor plan. So, take an example. So, this is basically the blocks A to E and these are the basically linear ordering. Whatever we discuss this

example I will go to that example. This is the final ordering we have the block A block B then block D E and C.

So, the final ordering is A B D E C. So, this is my ordering of the blocks A B D E C. So, here if you can see the same order is used A B D E C. So, this is the output from our linear ordering algorithm. Then what are the things given to us? We have given all the blocks given to us ordering is given to us then the aspect ratio aspect ratio means in different orientation how can you place the same block. Let us consider a block A it has two orientation one is horizontal orientation one is vertical orientation.

Let us say my WA of the block is 2 then the height is 3 or my width is 3 height is 2 but the area of the block is same. Area of the block A is basically 6 area of the block is 6. Similarly for all other blocks each of the block has two orientation. So, our final task is to find a floor plan with minimum global bounding box area. So, we will place the block such a way that we will have minimum bounding box area.

So, first block is A which area is basically 6. So, we will take this order 2 and 3. Now we will add the block B. Block B is WB is 2 and HB is 1. Then my area is 2 into 4 is 8. So, there is no wasting of area here. So, if I place the block in the horizontal direction at this point my area will be more. There will be wasting area. Let us say I will put this block sizes 2 and 1. So, if I put the block like this if I put the blocks 2 or 1 like this my overall bounding box will be like this. So, this is the wasting of area. This is not these are the there is nothing block is placed here.

So, this is not a good placement good floor plan. So, the A and B is placed such a way that my area is minimum. So, we need to check that we will place the block such a way that my area is minimum. So, then we will go to the block D. WD is 3 and HD is 3. So, we will place the block here. The bounding box the bounding box area is basically 20 here. We have some unused area is there. Then we will place the block E. After placing the block E at this point my bounding box area basically this area becomes 6 into 5 is 30. So, the bounding box area becomes 30. Now I have placed A B D E then I have block C. So, the block C is basically this is the aspect ratio of the block C. Now if I place this block my total bounding box area is 35 and this is the placement using the cluster growth algorithm. So, we will discuss about the simulated annealing. The simulated annealing is an iterative optimization algorithm. So, the main objective of the simulated annealing is to explore the solution phase. So, it will improve the objective function by exploring the solution space. So, there is a algorithm called greedy algorithm. So, if I go by the greedy algorithm I will always go to the local optimal value. In case of greedy algorithm it will never accept a solution with inferior value. The greedy algorithm always look for a best solution at each step of the algorithm but the simulated annealing accept inferior solution in the search of the global optimum. So, if you can look into this diagram your x axis is your different solution space basically and y axis is your cost. If I plot this one if I plot

this one and let us say this my starting solution is I my starting solution is I that is the initial solution.

Then if I go by the greedy approach, I will go to the local optimum local optimum but our actual optimum for the local minimum. So, here if you can see this is a minimization problem. So, it will go to the local minimum but actually if you look into this diagram I have another solution here which is better solution than the whatever I got using the local minimum. But if I go by the greedy method I will always fall into this local optimum but in case of simulated annealing it can accept a solution with basically inferior cost but it has a hope to get a global solution in future that is the main objective of simulated annealing. So, this is whatever this is for the minimization problem this is for the maximization problem. So, in case of maximization problem if I go by the greedy method this will go and stick to this point and this is called a local maximum this is called local maximum but that is not the best solution. We need to go for this global maximum. So, this slide basically summarizes what is the difference between our greedy algorithm versus simulated annealing. Greedy algorithm always accept improving solution which can lead to local optima. However, in case of simulated annealing it allows acceptance of inferior solution.

So, increasing the chance of going to the global optimum. So, what is the philosophy behind the simulated annealing is based on the annealing process. What is annealing? Annealing is basically the controlled cooling of high temperature material to achieve a minimum energy configuration. So, whenever you heat any material it got energized and it goes to high energy state and whenever we are cooling down the material then what will happen is that it will get to a minimum energy configuration and in the process of getting the minimum energy configuration basically we will get a better solution. So, here what is happening is that we involve transition from chaotic state to the structure state. When there is no unstructured state to the structure state we are moving from chaotic state to the structure state while we are going through the annealing process or cooling down of the temperature.

So, this annealing basically involves gradual cooling and basically whenever I am keeping a temperature constant I am exploring that different solution space for that problem. So, here the probability of drastic change in the atomic configuration decrease with the temperature. Basically, we have this probability of getting into the best solution comes at the lower temperature but we start with very high temperature and cool down till we get a lower temperature and get a better solution. So, slower cooling with small temperature increase increases the chance of reaching a global minimum. So, basically the process of cooling with small change in the temperature leads to getting into a global minimum.

So, we have a short space we are exploring the short space with different possible condition of the temperature. The simulated annealing basically applies the principle of annealing to a combinatorial optimization problem. It basically applies the process of annealing to a combinatorial optimization problem. It finds a lowest cost solution which is similar to finding the minimum energy state. So, what it says is this is most interesting point here is that whenever you are basically heating the material to high energy state then we reduce the temperature gradually then the atoms in the material will cool down and come to the minimum energy state.

So, that is for the material case. That is for the material case. But our objective is to find a minimum cost solution. We are exploring the solution space. So, in our case we are getting the lowest cost solution to our objective function. So, we will start with the initial solution then evaluate its cost then we will go to the new solution through a random work with perturbation. So, we have a random number generator which is giving me a new solution then we can accept or reject a solution basically based on the temperature.

So, we can accept a solution or reject a solution is based on the temperature parameter temperature. So, there was a Boltzmann acceptance criteria this is the Boltzmann acceptance criteria where we have a current solution. So, this is the current solution and we have a next solution which is the solution what we will get after the random work then this is the temperature current temperature and this R is a random number. If this is satisfied if this criteria is satisfied then we will accept that solution. The rate of temperature decrease the rate of temperature decrease is crucial for the success of simulated annealing.

And remember that the simulated annealing is most popular algorithm in VLSI physical design it uses quite frequently a different part of physical design to optimize various objective functions because it uses a heuristic driven algorithm and it is not greedy based algorithm. And it must enable sufficient high temperature exploration it actually basically explore the high temperature space also to check for all possible combination of the solutions at the beginning and allows enough time at low temperature to have a sufficient probability to settle to a near optimal solution. But there is no guarantee that I am getting the best optimal solution. However, if I run the algorithm for a multiple iterations there is a chance I can go to the near optimal solution because these problems are designed such that is very difficult to get into the optimal solution.

So, near optimal solution is also a good solution for the problem. So, it is basically a stochastic meaning two runs can yield two different results basically because it is driven by the random numbers. So, if the seed is different if I run the program two different time I can get two different results. So, the difference arise from the probabilistic decision in generating a new solution and accepting or rejecting a moves. So, it depends upon how we can accept and basically reject a move based on that our results can also vary. So, this

is a pseudo code of the simulated annealing. It starts with t equals to 0 and this is the first iteration. Then if I get the initial solution for the first iteration then what I have to do I need to find the cost. The cost is a function for the algorithm will pass through a cost function to find the cost of the present configuration. So, this present configuration what will be the cost? Now we have a minimum temperature T is greater than T minimum.

So, if I reach to a certain temperature I can stop it. This is stopping criteria. Then I have basically increasing the iterations. Then we can select a pair select two objects to putter then we can do small local change. So, this is a trial solution. Then the trial solution is go into the cost function. Like I did it for the initial solution. Now I have a trial solution then I pass it to a cost function to find what is my cost if I go by this trial solution. So, now I have a current cost. This is the current cost actually which I found from the initial solution. Then I have a trial cost. Then I find the change in the cost. This is basically the change in the cost. Now if my change is less than zero means that next if there is an improvement then I will change my current cost to the trial cost. Then I can allow the current cost to the trial cost. Then this move is executed because I am getting a better solution.

So, I will apply that move and I will go to a new solution. So, since it is a minimization problem I am getting a better cost means there is an improvement. Since this is applied for a minimization problem trial cost minus current cost delta cost is negative. So, I am getting an improvement. So, the trial cost is going to a better solution.

So, now the trial cost is my current cost and I will accept that move. If my trial cost is greater than my current cost in this case and delta cost is basically positive I am not getting any improvement then I can accept the cost with some probability. So, the random number R is basically zero to one. I need to check a random number between zero to one. What it says that zero is allowed but one is not allowed. So, this says that zero can be chosen as a random number but one cannot be taken as a random number.

This bracket says that one is not allowed. Now what I have to do I need to find R less than e to the power delta cost by t. So, if it meets the threshold let us say my t is basically high temperature. High temperature means this is basically close to zero. If it is close to zero this value becomes one. So, what means that if your R is less than one means whenever it is high temperature the probability of accepting the solution is higher.

When T is high then delta cost by T is close to zero and e to the power delta cost by T is close to one then R less than one because one is not included. So, when the temperature is high the probability of accepting the solution is more because I am exploring the complete solution space. Even if there is a penalty in the cost I can accept that move. So, T is basically reduced because alpha between zero to one your T is basically reduced temperature is reduced.

So, temperature is reduced and finally what will go? It will close to zero. If it is close to zero so whenever high this is the case we explain this one for the high condition but whenever it is close to zero whenever your T is close to zero then what will happen to minus delta cost by T? Delta cost minus by T is close to infinity. So, now if this is the case now e to the power minus delta cost by T should be close to zero because e to the power minus infinity will be one by infinity which is close to zero. So, now what it says that my R is less than zero. Basically, here R is less than zero means that here the cost is whenever your temperature is coming closer to zero then you are not getting any points between zero to one because R cannot be negative. So, in that case what we will do? There is a less chance of acceptance when your temperature is getting zero.

So, this is the main principle of simulated analysis. So, this is all explanation of this whatever I discussed in the previous slide. So, this is for the low temperature. This is the output. What is the main motive of this algorithm? There are two things.

One is basically if you increase the time x-axis is time and y-axis is the area. If I increase the time then I will get a better solution that is one. Then the second thing is that I have three configurations three different configuration if I run I have three different solutions. So, it tells that if you run it for a longer period you will get a better solution that is one thing. If I run it for different configurations I will get different solution at the output. So, we discussed about different flow planning algorithm in this lecture.

Thank you for your attention. Thank you very much.