VLSI Physical Design with Timing Analysis

Dr. Bishnu Prasad Das

Department of Electronics and Communication Engineering

Indian Institute of Technology, Roorkee

Week - 06

Lecture 33

Placement algorithms and legalization

Welcome to the course on VLSI Physical Design with Timing Analysis. In this lecture, we will discuss about some of the placement algorithms and finally, we will discuss about legalization. So, the content of this lecture includes force directed placement, then we will discuss about most popular algorithm called simulated annealing, then the third point what is most important which is called the legalization and detail placement. So, what is force directed placement? In this type of placement algorithm, the cells and the wires are modeled using a mechanical analogy of mass spring system. So, there whenever you have a mass and spring system, there is an attraction force between the cell, it is directly proportional to the distance. So, this attraction force, so let us say I have a spring which is connected with a mass that this the mass and springs has a force.

So, that attraction force is directly proportional to the distance. If you do that analysis of that system, finally, it will settle to a equilibrium position which is force equilibrium position. So, finally, all the cells eventually settle to a configuration that is called force equilibrium position. The force directed placement is a special case of quadratic placement. It is quite similar to that because here the potential energy is proportional to the square of the distance between the blocks, then the derivative of that potential energy is the force between the two blocks. So, what it does is basically it optimizes the minimum energy point, so which is equivalent to minimizing the sum of the squared Euclidean distance. So, here what is happening you have two connected blocks A and B which is having an attraction force FAB between the blocks A and B is denoted by this one.

$$\overrightarrow{F_{ab}} = c(a,b).(\vec{b} - \vec{a})$$

So, this C A and B is the connection weight between the cell A and the cell B. So, and this is B minus A is the vector difference of the position of the block A and block B in the Euclidean plane.

So, what we are doing is that we are basically summing all the forces on a particular cell I connected to the other cell J. So, if I have C ij is not equals to 0 that means that C ij is not equals to 0 means there is a connection from i to j, then that force will take into account. So, we have f of i, so this vector f of i is expressed using this formula.

$$\overrightarrow{F_i} = \sum_{c(i,j)\neq 0} \overrightarrow{F_{ij}}$$

So, what we are doing here is that we are minimizing that sum. So, the position that minimizes the sum of their forces is called a zero force target JFT.

So, zero force target is a location in the layout where my force is minimum. So, let us take an example here to understand the principle behind this force directed placement. So, what we are doing here is that we have cells A, B, C, D and I and the cell I is connected with each of the blocks, then we are finding the minimum force point where the cell I should be placed such that my ZFT position is achieved. The ZFT position is basically zero force target position of the cell I. So, what we are doing here is that we have a C iA means the connection between I and A.

So, this connection into vector A minus vector I. Similarly, I can do it for other blocks B, C and D. So, this is for B block, this is for C block, this is for D block and this one is for the A block. Now we have a combined force, I want to minimize that one. So, there is an algorithm which placement algorithm iteratively moves all the cells to its ZFT positions. So, what we are doing here is that we are minimizing the force in both the direction. As I told earlier, your x coordinate is independent of y coordinate. So, we are minimizing x coordinate independently with respect to y coordinate. So, this is the expression for x coordinate.

$$\sum_{c(i,j)\neq 0} c(i,j).(x_j^0 - x_i^0) = 0$$

this is the expression for y coordinate.

$$\sum_{c(i,j)\neq 0} c(i,j).(y_j^0 - y_i^0) = 0$$

Now what we can find here is that I want to find my x i coordinate and y i coordinate from this equation. My x i coordinate will be this one, my y i coordinate will be this one, just simplifying that expression. So, this x i and y i will give me the ZFT position of the cell I. This x i and y i will provide me my ZFT position of the cell I. So, we have an example to explain this force directed placement. So, these are the given things which is giving to the input to your placement algorithm.

So, we have a NAND gate, we have this NAND gate need to be placed such a way that the ZFT position of that cell can be found out. So, what is the inputs here? So, we have four IOPADs, but whatever I told you in the previous case, so whenever you are going for a global placement, the IOPAD and other macro pin locations should be fixed. So, here what is happening all the four pins, four IOPAD and the output locations are fixed. So, we have a 3 cross 3 grid, where my NAND gate will be placed, we need to find the JFT position for that one. So, the coordinate of the IOs are also given, in one is this coordinate, in two is this coordinate, in three is this coordinate and out is this one. Now we have some weight assigned to each of the nets actually. So, from node A to IN, the weight is 8, this is 10, this is 2 and this is 2. So, now, IN needs to find the JFT location of the cell A, so which will be the JFT location of the cell A. So, this is the layout, 3 cross 3 layout. So, IN1, IN2, IN3 is having the fixed location and out is already fixed. So, we have how many locations available to us? 1, 2, 3, 4 and 5. I need to find out which location is suitable to place this NAND gate, such that my JFT position will be achieved. So, this is the formulation, I need to find XA of 0, YA of 0 and this value comes out to be 0.9 and 1.6. So, here how it is found out, just I will explain one of them, then you can do the same thing repeatedly. So, let us say CA, 1 is 8, this is 8 and X coordinate of IN1. So, the X coordinate of IN1 is basically 2. Here if you can see, this is 2, 2, this coordinate. So, what we are doing is that, so 8 is the weight and 2 is the X coordinate.

Similarly if you can see CA, IN1, so what is the weight of this one? Weight is 8 and this is also 2. So, here if you can see 8 into 2, 8 into 2 and here 8 into 2. So, these dots are not floating point number, it is just multiplications. So, finally we will get 20 by 22 which is found with 0.9. So, if I do rounding up, then it will be 1, this will be 1 and this will be 2. So, if I go to the location, my X coordinate will be 1 and Y coordinate will be 2. So, 1, 2. So, in case of a post directed placement algorithm, the inputs are the set of cells which is denoted by V and your output is the placement of the cells. P is basically initial placement, then coordinates of each cell is given to us. Similarly what we do, we assume that all the cells are not fixed actually. So, this is done using this for loop. This is run for one time. This for loop will run for one time based on the number of nodes in the set V and all the nodes are unmoved. Unmoved means not fixed. Now what will happen is that we will go inside the algorithm. So, what we are doing here is that, so if all the move, all moved cells are not 0 or stopping criteria is not reached, then we will go inside the block. So, what is done here is that we pick the vertex which is having the maximum degree. Maximum degree is basically the cell or the vertex which is connected with more number of cells. So, that cell will be taken into account first which is having the maximum degree.

Then that one is passed to our zero position algorithm to find its position first. The cell having more number of connections should be find its JFT position first. Now so the location whatever we found from the this JFT position, if it is empty means it is available

or unoccupied, empty, available or unoccupied are the same thing. So, then we place that cell in that location and fix it. So, now that cell is fixed. Else if the JFT position of the cell C is occupied by another cell, then we have to move the cell C to another location. We have to then go through, relocate algorithm to find those locations. Finally we can mark the status of the cell C as moved. So, how we can move? So, there are some couple of techniques are there. How let us say some position is already occupied, how can you move? There are couple of techniques are there. Let us say I have a basically incoming cell is P and cell that is currently in P is JFT position is Q. So, incoming cell is P, already that JFT position of P is occupied by Q. If possible move that P to a cell position close to Q that is one technique. So, you can move that P to a cell position close to Q. Compute the cost difference, then we can find the cost difference.

If P, Q were to be swapped, if the cost reduces then we can swap the position of P with that of the Q. So, this is the first technique, technique 1. Then we have the second technique which is called the chain move. In this one what happens if the cell P is moved to the cell Q's location, cell P which is coming that will be moved to the Q's location. So, now Q is not fixed. It is basically not fixed. Then what you have to do? We have to move that cell Q is shifted to the next position. Cell Q is moved to the next position. If a cell R is occupying that same space, if the R is occupying the same space then the R is shifted to the next position. So, this process is continued. Let us say for example, I will explain this. Let us say Q is there and P is coming here. So, what we do in the first step? We will place the P here and move the Q to another position. But let us say the Q position is occupied by R. Then the second step is that P is already fixed then move the R to another location and place the Q in R location like this.

So, this process is called the chain move because we are doing this process repeatedly. So, that is why it is called the chain move. Then there is a procedure called the third technique is called the ripple move. What is this ripple move? Cell P is moved to the Q's location. And so once it is moved, the new JFT position of the Q is computed. So, here we need to do that optimization technique again to find the suitable location of Q and this is called the ripple effect, continues until all the cells are placed. So, that is why it is called the ripple move. In this slide, we are discussing one of the examples of a force directed placement. So, we have three blocks, block 1, block 2 and block 3. This is B1, B2 and B3 whatever here it is written. Then we have net, there are two nets are there, N1 and N2. Then you have C of N1 and C of N2 is there. So, what we are doing is that our task is to find what is the position of the blocks such that my minimum wire length can be achieved in the placement. How I can do that? So, here we have B1, B2, B3 which is the initial placement and we are finding the JFT position of block 3. So, this is the incoming cell, B3 is the incoming cell. So, there I need to find the X coordinate. Why I am only finding X coordinate? Because the X coordinate information is given to me, Y coordinate information is not given to me. So, if the Y coordinate information will be given to me,

then I can do same type of calculation for Y coordinate. So, here if you can see, you have C B3, B1, B3, B1 is 2 and X B1 is 0 which is leading to 0. So, this 2 is same as this 2 and this 0 is same as this 0. Similarly you have C B2, B3 which is 1, this one corresponds to this one.

Similarly X B2 which is 1 is basically this will result in 1. So, this is for this numerator, this is for the numerator calculation. This is 0 plus 1 and the denominator is basically 2, this 2 is same as this 2 and this 1 is same as this 1. So, this will be basically 1 by 3, answer is actually it should be 1 by 3, but if you round down then it will be 0. So, basically speaking the B3 ZFT position is 0, 0. So, this initial placement whatever it is there, B3 is in 2, 0 but after calculating the ZFT position for B3 is 0, 0. So, since there is a block B1 is there already in this position and B3 is also coming to that place then we need to find L of P for before move and after move. So, the before move case is this one, the initial placement the case 1, this L of P is basically C of n1 into D of n1, C of n1 into D of n1 plus C of n2 into D of n2. Now this C of n1 is 2 here into D of n1 is basically distance from this to this, this is 2 here and similarly C of n2, C of n2 is basically 1 here multiplied by D of n2, multiplied by D of n2. D of n2 is this distance, middle of this one to middle of this one, this distance is 1. So, the L of P is basically 5, 4 plus 1 is 5. So, this is for the initial placement, so before the move. Now we can do it for after the move, what is my L of P? After the move the case 2, after the move what is L of P? So, this L of P in this case is basically C of n1 multiplied by D of n1, C of n2 multiplied by D of n2. So, this will be C of n1 is 2, D of n1, this is the n1 net, so the distance is 2 and C of n2, this is the n2 net and this weight is 1 multiplied by middle of this one to middle of this one, this distance is 1, so this is 1, so it is 4 plus 1 it is 5. So, the before the move this is 5, L of P and after the move L of P is 5. So, it is not good decision to swap B3 with B1, swap B3 with B1. In this slide we are finding the ZFT position of the incoming cell B2. So, ZFT position of B2 we can find it out if we have CB2, B1 this is 0 and X of B1 is 0 CB2, B3 is 1 and XB3 is basically 2. So, basically if we sum these two the denominator is basically summation of these two is basically 1 and the numerator here is basically the multiplication of this into this, this is 0 and multiplication of this into this which is 2, so here it will be your answer is 2 by 1, so this is 2. So, the ZFT position of B2 is found to be 2, 0.

So, ZFT position of B2 is 2, 0. Now if I do this move what is the change in L of P? So, the change of L of P after the move let us say the case 3, so after the move is that so your L of P is basically C of n1 into D of n1, C of n2 into D of n2. So, the C of n1 is 2 multiplied by 1, 1 into 1, so this will be 3. So, now L of P is found to be 3, so wire length is reduced by 2, so we are swapping B2 and B3, this method is called force directed placement. In case of force directed placement it finds the ZFT position of the block then it checks that after doing that movement or the swap whether there is an improvement in

the wire length. If there is an improvement in the wire length then we basically apply the technique to move and place the block into their ZFT position.

So, what are the main technique here? It is very simple, easy to implement and it is used for global placement but can also be adopted for detail placement also. So, what are the disadvantages that it does not scale for a large scale placement, large scale instances it takes very long time. It is not very effective in spreading the cells in the denser regions. So, it is a poor trade-off solution for quality and runtime.

So, these are the disadvantages of force directed placement. Then we will discuss about most popular algorithm which is called simulated annealing. Here it takes the temperature into account while optimizing the function, objective function. So, here what is this is basically iterative optimization algorithm. Here what we are doing is that we are improving the objective function by exploring the complete solution space. It is unlike a greedy algorithm, it accepts the simulated annealing, accepts inferior solution in the search of finding the global optimum. So, in case of simulated annealing, the input to the algorithm is set of all the cells V. So, all the cells are input to the algorithm and the output is the placement of all the cells. So, the simulated annealing algorithm is based on a temperature profile. What is this temperature profile? Basically we start with a very high temperature and then we will reduce the temperature till we will get a optimal solution. So, initially we will set the temperature T equal to T0, then we have an initial placement. Then we have a minimum temperature. So, if temperature is greater than the T minimum, then we will do this operations. So, what we do, we have a putter function, what we will discuss later. Then we will have a cost, delta cost which is the cost of the new variable minus the present variable. So, this is a new point in the solution space, this is the old point. So, this is a new point in the solution space, this is the old placement point or the old point in the solution space.

So, this is new point is a new placement and this is the old placement. Now what you are doing, if my delta is less than 0, means I am doing the increment. So, I am getting better solution. So, then I will accept the solution. There is no issue. If my wirelength is reduced, then I will accept that solution. Else what you have to do, I will not reject the solution, but I can accept that with some probability. So, what we are doing is that we have finding a random number between 0 to 1 and if this one is less than e to the power minus delta cost by T, then I will accept that solution P, I will accept that placement. And when the temperature is high, there is more probability of accepting the solution. When the temperature is low, there is less chance of accepting the solution. The reason is that when the temperature is high, we are not close to the actual solution. When the temperature is low, we are very close to the global minimum. So, we do not accept when your temperature is low. Now what we are doing, we are reducing the temperature by alpha factor which is lie between 0 to 1.

So, this is all about simulated annealing. Then we have a Timber-Wolf algorithm which is an academic package developed by University of California by C. Sechen and this algorithm produces an initial placement with all the cells in a row. It will find an initial solution with all the cells in a row from the width of the cells and target row length. How it works? It has a few steps, three steps. First step is the placement of the standard cell with minimum wire length using the simulated annealing.

So, this Timber-Wolf algorithm, the first step is the placement of the standard cell with minimum total wire length using simulated annealing approach. Then second step is basically globally route the placement by introducing the routing channel while minimizing the total wire length. Then the third step is basically locally optimize the placement to minimize the channel height. So, these are the three steps of the algorithm. So, now we are talking about Perturb which is used in case of simulated annealing. It generates a new placement. So, what is this Perturb function does? It generates a new placement from an existing placement using one of the following actions. There are three actions are there, one, two and three. The move is one action. What it does is it shifts a cell to a new location or new position in another row. So, shift a cell from one location to a new position in another row. Then swap, swap is basically you are in the position of a cell with another cell that swapping the cell. Then we have a mirror, mirror means that we reflect the cell orientation. So, we can change the orientation of cells around the y-axis. So, if move and the swap actions are not feasible, then we can do the mirror action which is basically orientation of the cell will be changed. So, the scope of this Perturb is limited to the small window size of basically wt cross ht.

So, we have a window size which is wt is the width of the window and ht is the height of the window. What we are doing is that for move a cell can only be moved within that window. So, whenever we are moving the cell that cell can move within that window. Whenever we are swapped to cells A and B, we can exchange only if this condition is satisfied. So, we have a basically window size wt ht depends upon the current temperature like whatever you are discussing about the simulated annealing. So, it window size will depend upon the current temperature and decreases as temperature reduces. So, this is the window size definition which depends upon the temperature value. The size of the window reduces as temperature decreases.

Now we have a cost function. This is gamma. So, this cost function has three things gamma 1, gamma 2 and gamma 3. This gamma 1 is one of the cost function which finds the total estimated wirelength. The second one is the amount of overlap. It defines the amount of overlap between the cells and the third one is basically row inequality in the length. So, we will discuss each one of them individually. So, gamma 1 is computed using the half perimeter wirelength model.

$$\Gamma_1 = \gamma_1 . \sum_{net\ \varepsilon\ Netlist} w_H(net).x_{net} + w_v(net).y_{net}$$

We discussed about the half perimeter wirelength model. So, HPWL wirelength estimation model will be used to find the total wirelength of the placement. So, this gamma, so your gamma 1 is the priority of weight for each net is a priority weight for gamma 1. So, what is the priority you can assign to the wirelength estimation since we have three different objective, how much priority you are giving to the wirelength estimation.

So, this is the factor gamma 1. Then WH and WV are the weights in the horizontal and vertical directions. Now we have the second objective function gamma 2 which defines the sum of the squares of all cells overlap.

$$\Gamma_2 = \gamma_2 . \sum_{i\ \varepsilon\ V, j\ \varepsilon\ V, i \neq j} o(i, j)^2$$

So, this is the priority assigned to this objective basically cost function and this is defines the square of all cells overlap, represents the area of the overlap between cell I and J. So, how much overlap is there that is found out and included in the objective function. Then the third one is defined as the sum of the row length deviation for all rows.

$$\Gamma_3 = \gamma_3 . \sum_{row\ \varepsilon\ Rows} |\, L(row) - L_{opt}(row) \,|$$

So, how much row length deviation is there. So, the gamma 3 is a priority for gamma 3 for this objective function. Lrow is the length of the row and Lopt is the goal length of the row. So, that difference will be taken into account in this objective function. So, we have basically a factor cooling factor alpha is empirically chosen basically depending upon our experiences and that alpha will help us to find out our solution easily. So, initially the temperature is reduced quickly using alpha equal to 0.8. So, now we will discuss about legalization and detail placement. What is this legalization means that it assigns the continuous coordinate to the standard cells and the macro blocks. So, what is the legalization is that it aligns the cells to their power rails and assign the discrete legal co-locations within the predefined rows actually. So, the main aim of legalization is required to align the power rails and it will assign the discrete legal location within the predefined rows.

If you look into this slide we have multiple standard cells are there. For example this is an inverter, this is a NAND, this is a NOR and they have a power grid or power line is there. For each of the cell your VDD and ground is constructed in metal one. So, it is by design if you place them together in a row then it will automatically create the power grid in the metal one. So, these cells whenever you are cascading or putting next to each other

without overlapping them then they will satisfy the DRC rule and also automatically creates the power and ground network. So, this is your VDD line and this is your ground line which is automatically created in metal one which is possible in case of a standard cell based design which is not possible in case of a full custom design style.

Here all the cells are placed in their legal locations and there is no overlap of the cells and all of the cells are getting the proper supply voltage. So, these are the legal position of the standard cells. So, this is the process of bringing the cells and putting into its power grid is called the process of legalization. So, legalization aims to find the non-overlapping legal placements to minimize the adverse impact on wire length, timing and other objective functions, other design objectives. So, here what is the point here is that we improve the basically the detail placement improves the can further be improved with legal placement. It considers the objective like reducing the total wire length and this detail placement techniques include basically two things. One is the swapping of the neighboring cells to reduce the total wire length or the second one is sliding the cells to one side of the row when a new space is available. So, in this lecture we discussed about several placement algorithm and legalization phases of the VLSI physical design.

Thank you for your attention.