Class Project Proposal: Length Matching Routing with Given Topology

Tan Yan

April 7, 2008

Routing for PCB (printed circuit boards) is full of challanges. This is mainly due to the constraints that must be satisfied. One of the major constraint on PCB routing is the length constraint. That is, the routing length of each wire must be within some bounds. Since general routing problem is already NP-hard, routing with the length constraints becomes even harder. To make things simpler, we assume that the topology of the routing is already given and our task is to transform the topology into detailed routing that satisfies the constraints.

Now let me describe the problem in better details. The length matching routing problem can be formulated as follows. An example can be found in Figure 1.

Input:

- A rectilinear piecewise-linear complex as the domain of the routing. Its definition is almost the same as the piecewise-linear complex introduced in the triangular mesh generation lecture. The only difference here is that all the segments in the complex are either horizontal or vertical. This is why the word "rectilinear" is there. As can be seen in the example, the domain can have "holes" which represents the previously placed devices and "segments" which represents the previously routed wires.
- A constant f as the minimum feature size of the routing. Any point in the domain should have a local feature size larger than f.
- A set of points P inside the domain. They are the pins.
- A relation $N \subseteq P \times P$. This specifies the connections between the pins (we call the connection net). For simplicity, we can assume that one pin is connected to only one other pin, i.e., no two nets can share the same pin.
- For each net $n \in N$, we are given a path between its two pins. All the paths given are guaranteed to have no intersections. You can assume that the paths are given in any form you want. In our example, the paths are given as rectlinear (that is, containing only horizontal and vertical segments) paths.
- For each net $n \in N$, we are given a lower length bound l_n and an upper bound u_n. Different nets might have different bounds.

Output: A rectilinear path p_n for each net $n \in N$ that connects the two pins of the net. Again, "rectilinear" means that the segments of the path can only be horizontal or vertical. The length of p_n must satisfy $l_n \leq length(p_n) \leq u_n$. Finally, all the paths must be within the domain given and must be disjoint (that is, no intersections between the paths).

Since the topology is given, the task is to meet the length bounds. First of all, we should try to route all nets as short as possible. This can be done by many existing algorithms, e.g., [1]. We check if the upper bounds of all the nets are satisfied. If not, we know that all the nets are already routed as short as possible yet the upper length bounds are violated. This means there is no feasible solution.

If the upper bounds can all be satisfied, we need to extend the routing so that the lower bounds are satisfied. A common way to do so is to detour the wire as in Figure 2. But please don't be restricted to this way of extending length. Notice that due to the minimum feature size constriant, we cannot make the vertical segments infinitely close to each other.

It seems that even satisfying the lower bound is NP-hard. The rational comes from the similarity between this problem and the Hamilton path problem. For a problem with only one net, suppose we apply a grid on the domain with minimum feature size as the grid size. Then we make the continous domain descrete. If we restrict our routing on the grid, our problem becomes "find a simple path between two nodes such that its length is longer than a given constant". If the given constant is exactly the total amount of grids in the domain, our problem becomes find a

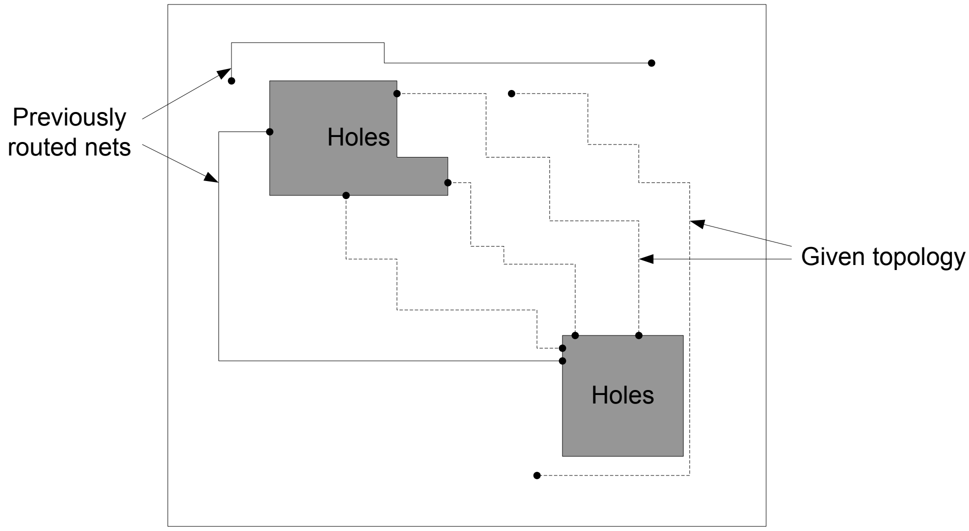

Figure 1: An example of the input.

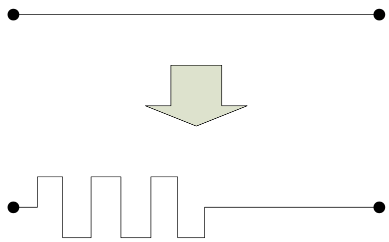

Figure 2: Detour the wire for longer length.

simple path between two nodes that goes through all the nodes in the grid once and only once. This is the Hamilton path problem. Therefore, there is a high probability that the problem we try to solve is NP-hard.

References

[1] H. F. Steven Chen and D. T. Lee, "A Fast Algorithm for Rubber-Band Equivalent Transformation for Planar VLSI Layouts," IEEE Trans. on CAD, Vol. 15, No. 2, Feb. 1996.