System Level Pin Assignment Algorithm Using OpenAccess Database

Junhao Hua and Silei Ma
University of California, Los Angeles
Email:{neilair, pseuuuuu}@g.ucla.edu

***Abstract*—Pin assignment in today's VLSI design flow has shown significant importance, as it fundamentally determines the later routing performance. Coarse routing estimation is often deployed during pin assignment to achieve an optimized result. This work shows a simple algorithm that implements pin assignment in C++ using OpenAccess database under various constraints: maximum perturbation, number of layers, wire-length optimization.**

I. Progress Report

We have finished extracting information from given circuit library and re-constructed our own data structure to avoid repetitively iteration. We deployed an algorithm to find the most optimized point under the perturbation constraint.

II. Approach

A. Data Structure

In this work, we extract useful information from given library to form our own data structure, consisting the following *structs*: **net, macro and side.** Doing so allows us to keep most frequently used information without receptively iterating through the OpenAccess database. Some details are shown in the table below.

The struct **net** is the main accessing portal which provides direct access to each instTerm. Each **macro** is constructed with 4 sets for validating the arrival point from the algorithm. Each arrival point will be checked and registered into the corresponding *side*, therefore no two arrival points share the same location nor in the illegal vicinity. Four *sides* in a macro is **circularly double linked** in the case the arrival point needs to have a transition to neighbor side. [1]

net	macro	side
name	name	set list
number of InstTerm	oaInst * pU	set * pNext
number of Term	oaCoord l,r,t,b	set * pPrev
number of InstTerm	set {l,r,t,b}_list	
instTermP (oaInsterm **)	...	
termP (oaTerm **)		
reset functions	.	

B. Basic Algorithm (Port Algorithm)

Assume that you are living on several islands in an Ocean. For some reasons, what you live on are supplies air-dropped at a fixed buoy in the ocean near the islands every week. On each island, you need to build a port on the boundary of the island to make the total distance shortest for all islands to get to the buoy.

In our case, macros are like islands which are rectangles. All pins of InstTerms are ports to be built. For net connecting one Term and several InstTerms, Term act as fixed buoy. For net connecting only InstTerms, buoy is located in the middle of all macros.

To decide the position of buoy for a certain net, as Fig.1 shows, assume the net contains InstTerms from several macros. In this case the buoy should be assigned somewhere is the shade.

For an InstTerm with a decided buoy, we divide the macro in to 12 parts. Each part corresponds to an implement to find the port. With this method, we can get the ports for the case in Fig.1. As what is shown in Fig.3, the buoy is actually the stein point of RSMT. So the total wirelength (distance) is guaranteed to be shortest.

Under the consideration of perturbation and step size, the coordinate of ports may be unreachable or illegal. And the final coordinate of an InstTerm may be occupied by another already.To solve these problems, we build up a map to check if the final position is legal and available. If its not, find the closest satisfied position before reaching the destination.

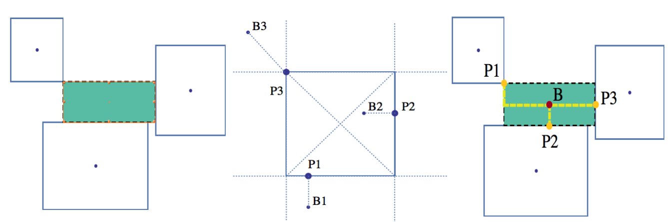

Fig 1 Fig 2 Fig 3

III. Problem

It remains a problem for us to determine the buoy in the case of InstTerm connected with InsTerm. We still need to improve our method of pin legalization, including taking metal track pitch into consideration. Also, an efficient coarse routing and congestion estimation is needed to roughly testify if the improved pin assignment can satisfy routing rules.

References

[1] https://en.wikipedia.org/wiki/Linked_list