ECE 6133
Implementation of “Efficient Algorithms for Channel Routing” - Yoshimura & Kuh

SHREEPAD A PANTH

Channel Routing Problem

- Route between a top row and bottom row, pins with the same number have to be connected
- Using only two metal layers
- Overall area of the channel (ie height of the channel) has to be minimized.

The Algorithm

- L = { }
- For z_s to z_t do :
- L = L + {nets that terminate at zone z}
- R = {nets that begin at zone z+1 }
- Merge L and R so as to minimize the increase in the longest path in VCG
- L = L - { merged nets in previous step }

The Algorithm (contd)

Merging is done heuristically

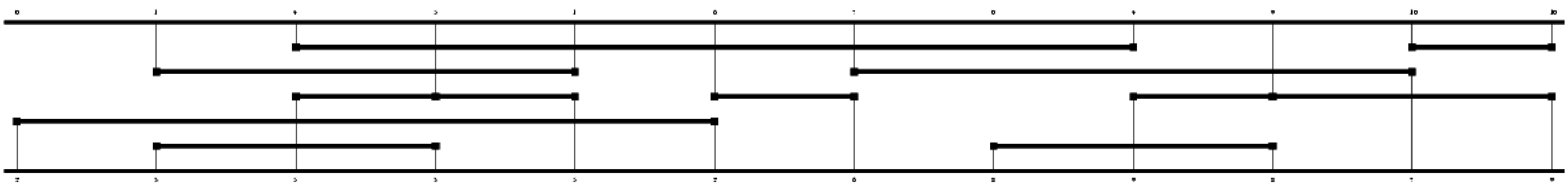
- Pair to be merged can be computed by knowing the longest source-node (u) and sink-node(d) paths for every node.
- By maximizing f(m) and minimizing g(n,m) the pair to be merged is obtained heuristically

Implementation

- If no cycles are present in the VCG , route the given problem without inserting any doglegs

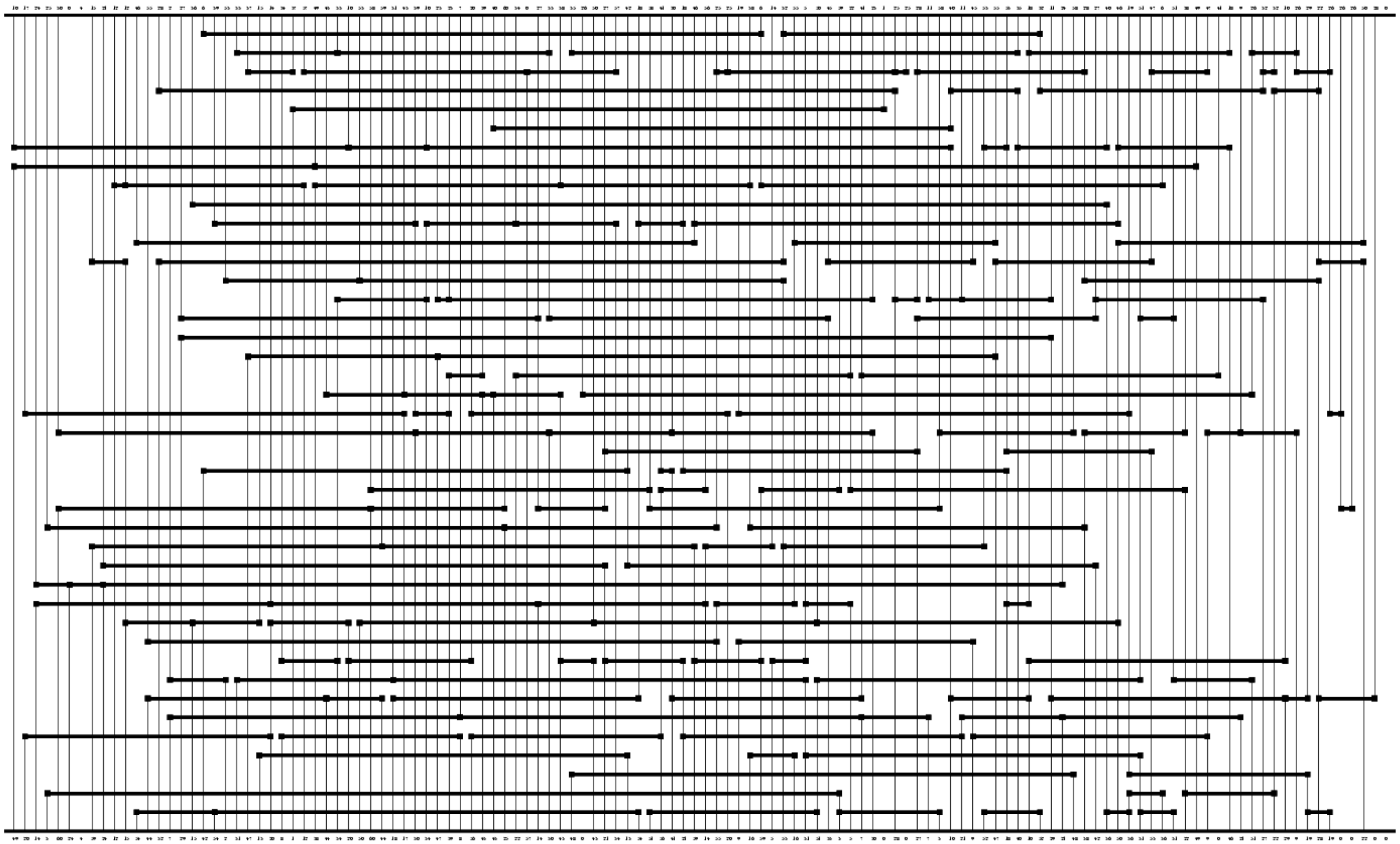
- If VCG contains cycles, then break all nets into two terminal nets and perform channel routing.

Results – small circuit



No of nets = 10
Average netsize = 2
Max density = 5
No of tracks = 5
No of vias used = 22

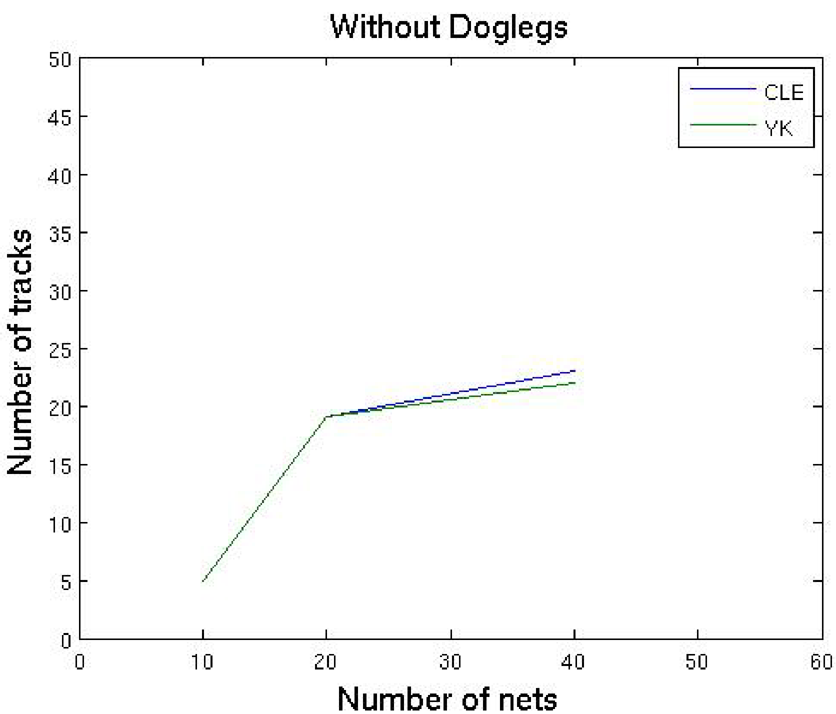
Results – Larger Circuit

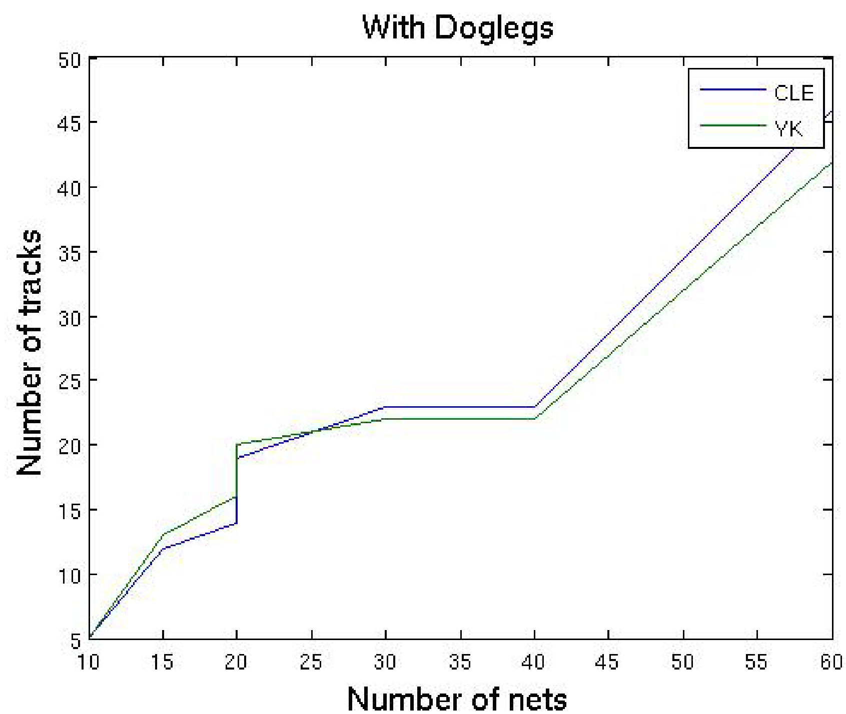


No of nets = 60
Average netsize = 4
Max density = 41
No of tracks = 42
No of vias used = 331

Results

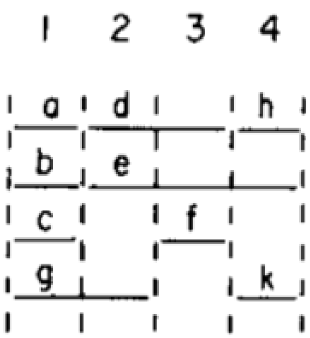
			CLE		YK	
Circuit	***Nets***	***Max Den***	***No doglegs***	***Doglegs***	***No Doglegs***	***Doglegs***
dr1	***10***	***5***	***5***	***5***	***5***	***5***
dr6	***15***	***9***	***-***	***12***	***-***	***13***
dr2	***20***	***14***	***-***	***14***	***-***	***16***
dr7	***20***	***19***	***19***	***19***	***19***	***20***
dr3	***30***	***20***	***-***	***23***	***-***	***22***
dr8	***40***	***22***	***23***	***23***	***22***	***22***
dr4	***60***	***41***	***-***	***46***	***-***	***42***

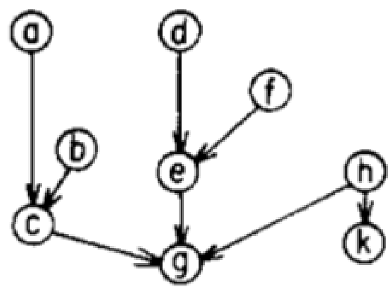




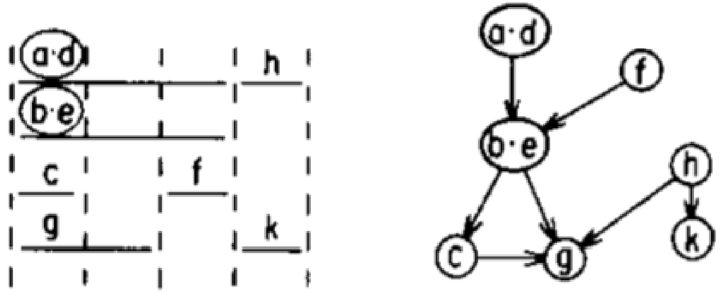
Reasons for discrepancies

- Too many doglegs worsen the solution (shown by dr7)
- With a lot of nets , merging of nets sometimes blocks the merging of other nets





- Suppose we merge nets a&d ; b&e



- Net f cannot be merged with either net c or net g as cycle will be formed in VCG
- On the other hand if we merge a&d; c&e , net f can be merged with net b

Solutions

- Inserting Doglegs only where necessary to break cycles in VCG, however potential improvement may be lost.
- Algorithm #2 proposed by Yoshimura & Kuh based on bipartite graph and matching of nets.