Design Project

Constrained Routing in Physical Design

By :
Vedanshu Goyal (B19CSE096)
Sarthak Kumar Singh (B19CSE077)
Saurav Majoka (B19CSE079)

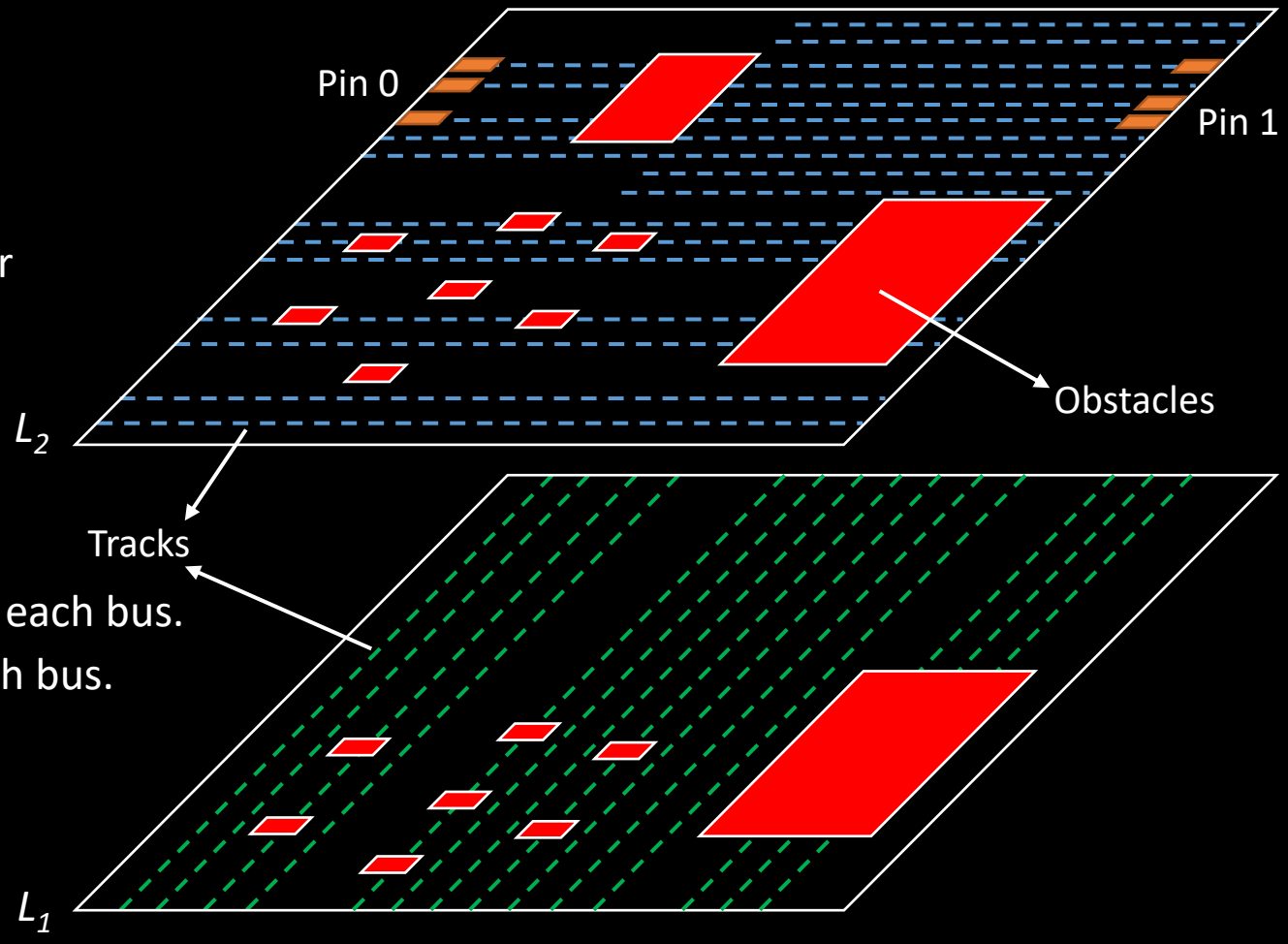
Outline:

- Problem Formulation
- Method used to solved the problem : DAG based Algorithm
- Result and Analysis
- Conclusion

Problem Formulation

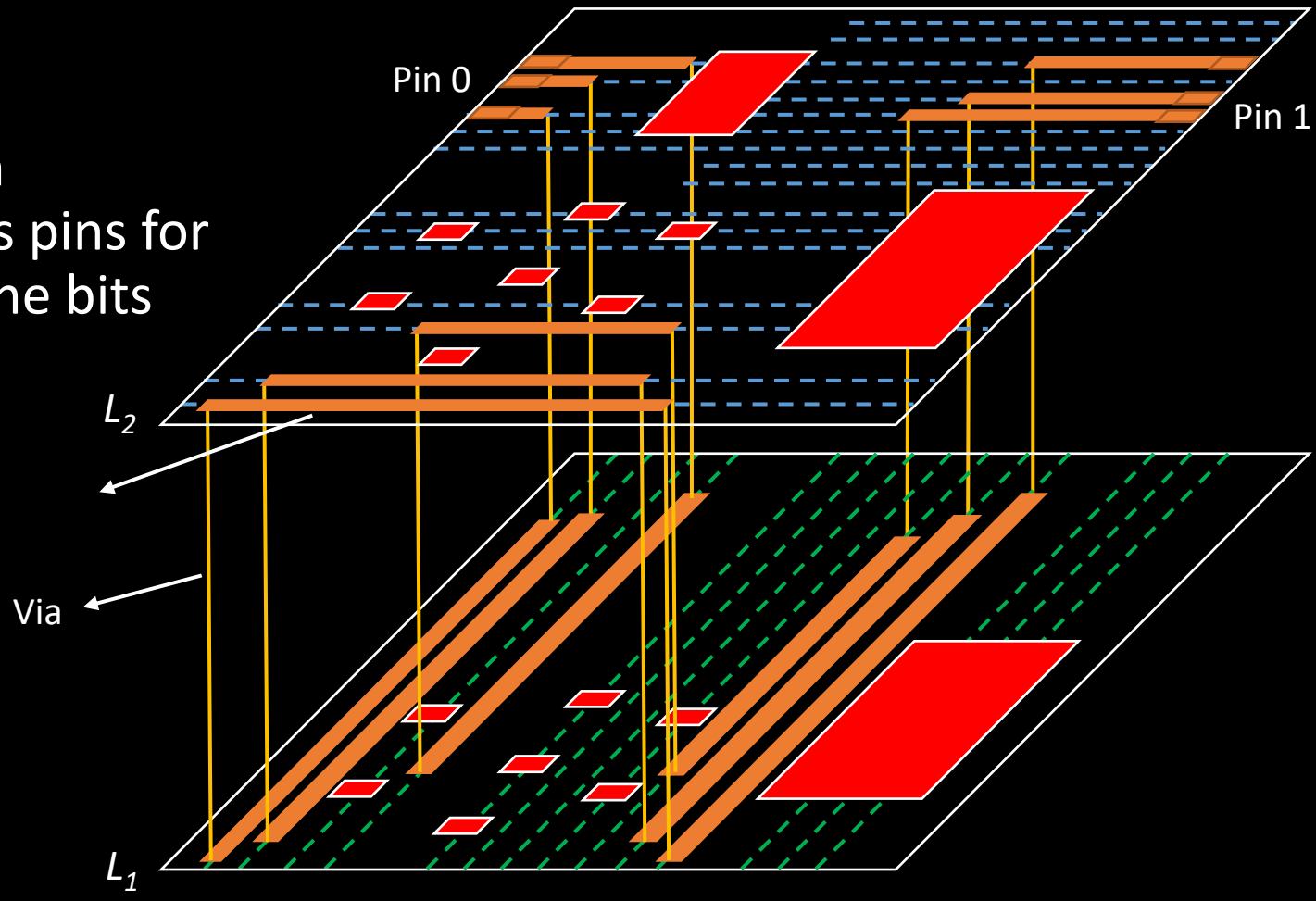
This is algorithmic challenging problem in which we have a find a routes between pin groups of a single bus.

- Input Format:
  - A set of layers
    - With preferred track direction for each layer
    - With min spacing constraint for each layer.
  - A set of routing tracks
    - With wire width constraint for each track.
  - A set of buses
    - With a set of bits to be routed on tracks for each bus.
    - With width constraint for each layer for each bus.
  - A set of obstacles for each layer



Problem Formulation

- Output Format:
  - A set of wires and vias(connection between two layers) that connects pins for all buses and in the same bus all the bits must have the same topology.



Problem Formulation

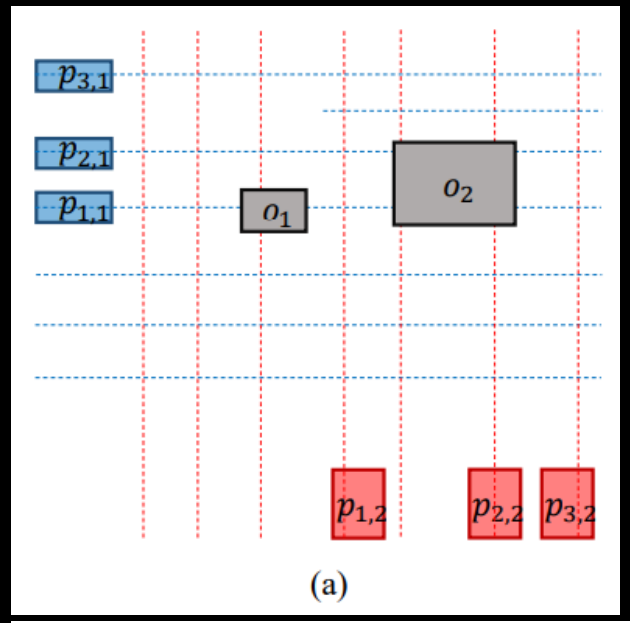
- Cost Calculation:
  - The evaluator compute the cost of our solution by the following rules:
  - Total Routing Cost will be:
    - Wire Length cost
    - Segment cost
    - Compactness cost

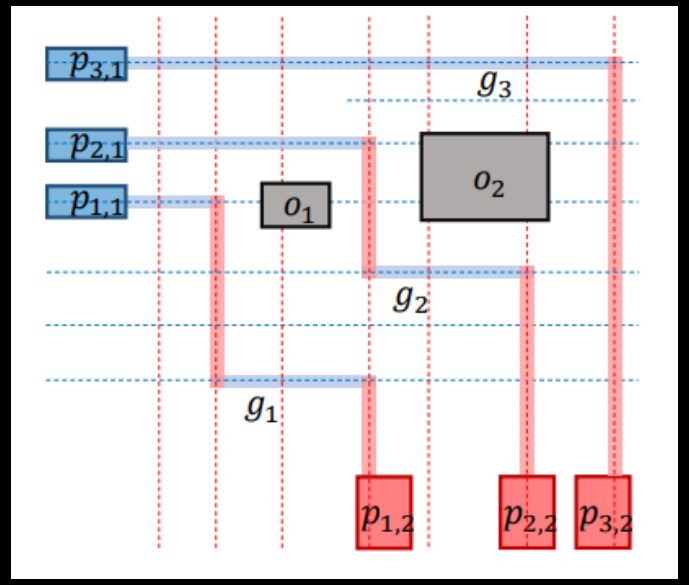
Our Methodology: DAG BASED ALGORITHM

- Choose all the bits of a bus.
- Then route each bit to its respective destination with the help of A* star algorithm.

  Sg = Set of all path of all the bits.
- Maintain the Topological candidate list(TCL) in which all the bits which have same topological guide comes under same group.

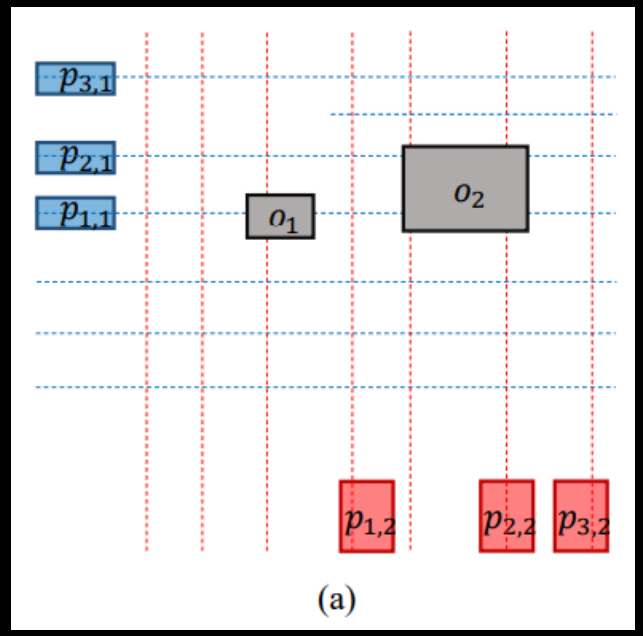
  Ex: Bits {1, 2} maintaining same topology so it comes under same TCL and {3} forms its TCL itself.

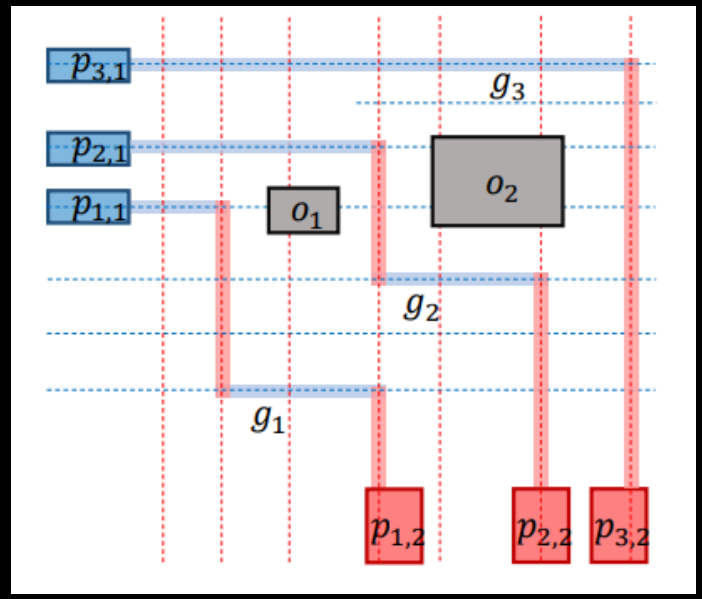




Our Methodology: DAG BASED ALGORITHM

- Sort the TCL list according to the number of bits in its group.
- Traverse the list and choose a TCL (max->min).
- For each segment in a TCL, we have to decide whether this segment is either LSB to MSB or MSB to LSB.
- So now iterate through each bits and try to find the path with the same topology of the chosen TCL-topology.
- To connect a bit in the specific topology, a routing directed acyclic graph(DAG) G(V, E) is constructed with several properties.



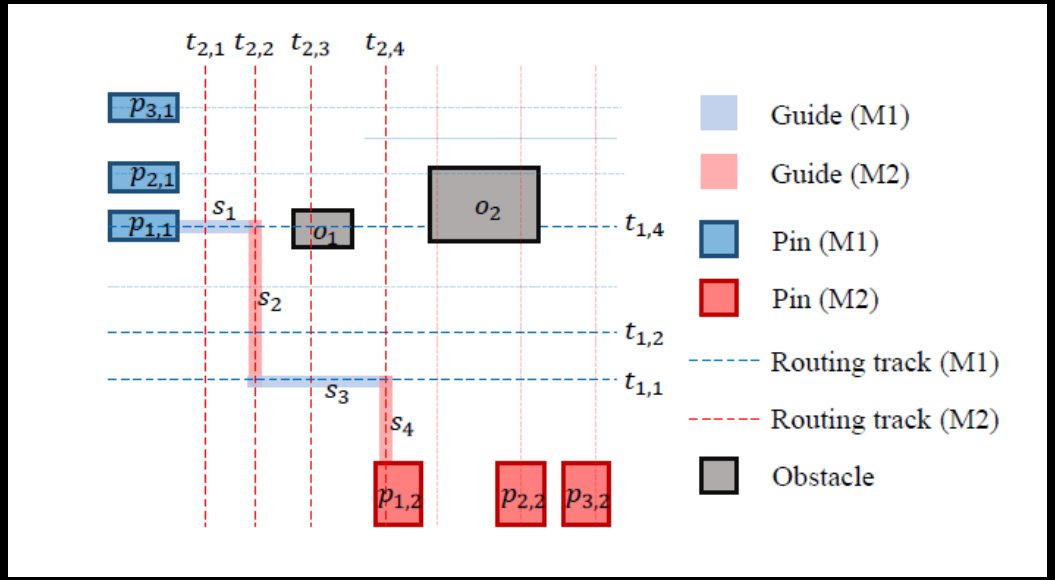


Our Methodology: DAG BASED ALGORITHM

Construction of DAG

- First we will maintain an usable track list(UTL) for each segment.
- To form the UTL_i for the first bit we have to take n_{first} nearest tracks from both sides of the segment s_i. And for the non-first bits we have to take $n_{non\text{-}first}$ nearest tracks from one side.

$$n_{first} = \min(100, 6n_{Bbit})$$
$$n_{non\text{-}first} = \min(100, 2n_{Bbit})$$

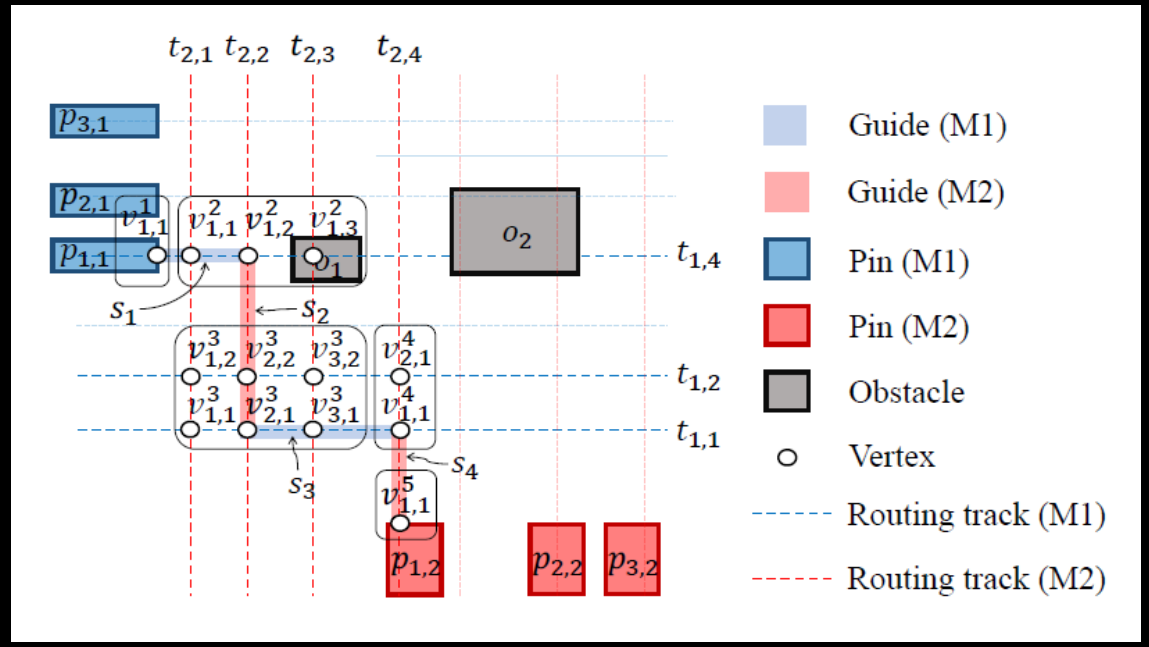


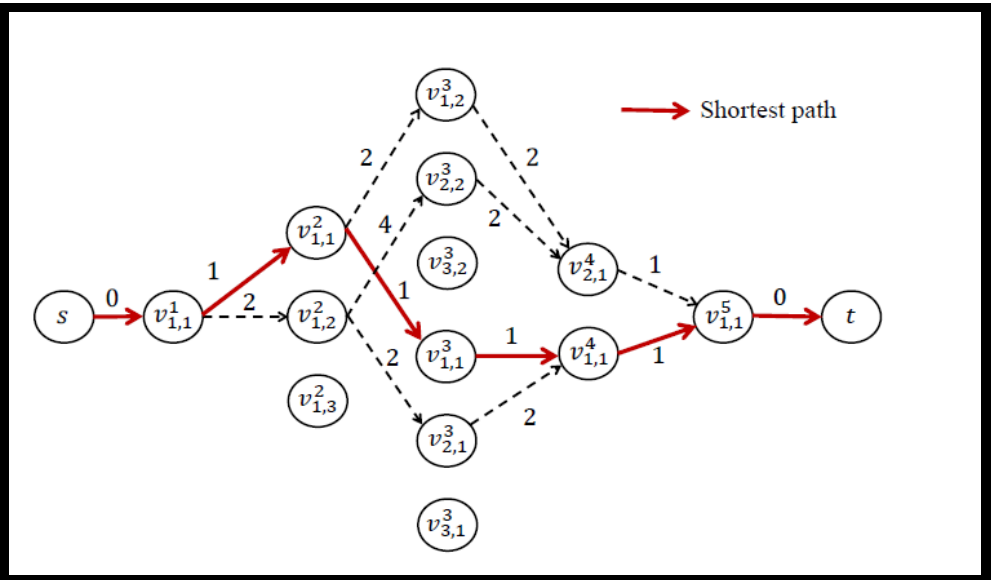
Suppose n_{first} = 2.
UTL2 is $\langle t_{2,1}, t_{2,2}, t_{2,3} \rangle$.
UTL3 is $\langle t_{1,1}, t_{1,2} \rangle$.

Our Methodology: DAG BASED ALGORITHM

Construction of DAG

- For two adjacent segments, say s_i and s_{i+1}, the intersection points of the perpendicular tracks in UTL_i and UTL_{i+1} form a box. All box points are vertices in G.
- To determine edge costs, each UTL is sorted by the track positions in increasing (decreasing) order if the bit direction of the corresponding segment is LSB(MSB). Now, the edge cost will be the multiplication of the index of two tracks in their respective UTL list.





Result And Analysis

Input

SERIAL No.	No. Of LAYERS	No. Of Tracks	DESIGN AREA	Bits in a Pin Group	No. of Obstacles
1	2	9	0.25 mm^2	3	17
2	3	22732	1.5 mm^2	9	0
3	4	54150	15.5 mm^2	375	10

Output

SERIAL No.	No. of Segments
1	27
2	108
3	2625

Conclusion

- We have implemented an effective and efficient topology matching bus routing algorithm considering obstacles and non-uniform track configurations.
- A DAG-based bus routing algorithm has been proposed to efficiently connect all bits within a bus in the specified topology.
- We have implemented our algorithm for the single bus, to use it for the multi-bus we have to implement LCS bus-clustering method to find the order of buses, and then proceed one by one with our single bus algorithm.