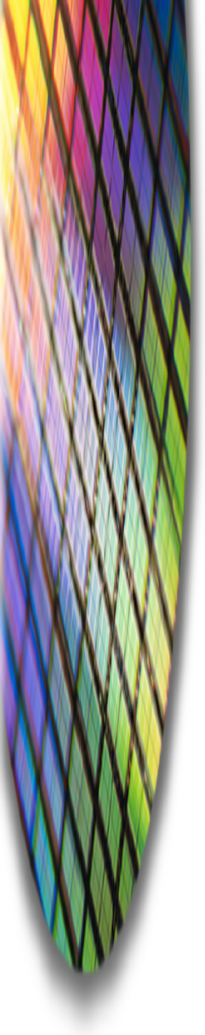


2013 Advanced Lithography

UT DA

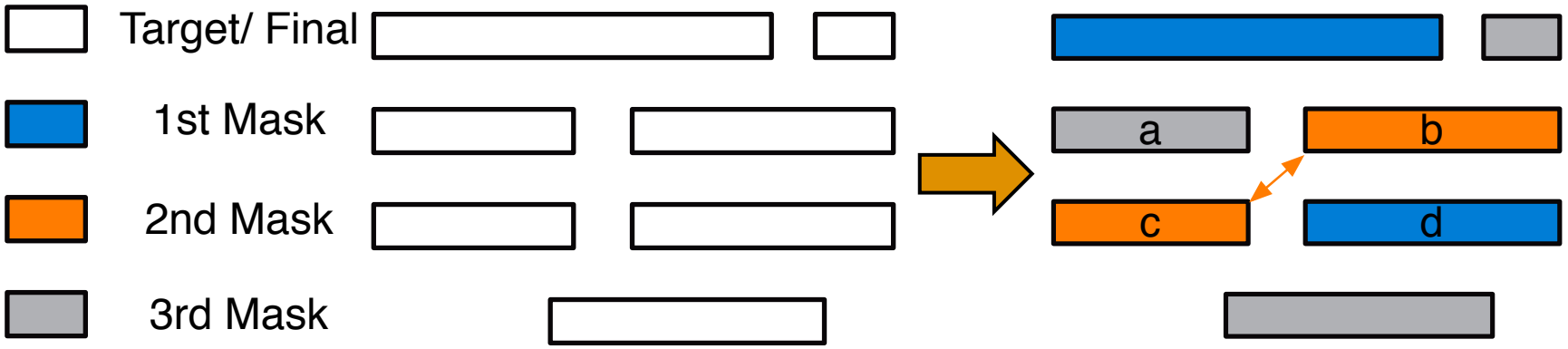
Triple Patterning Lithography (TPL) Layout Decomposition using End-Cutting

Bei Yu, Jhih-Rong Gao, and David Z. Pan
Dept. of Electrical & Computer Engineering
University of Texas at Austin

Supported in part by NSF, SRC, Oracle, and NSFC

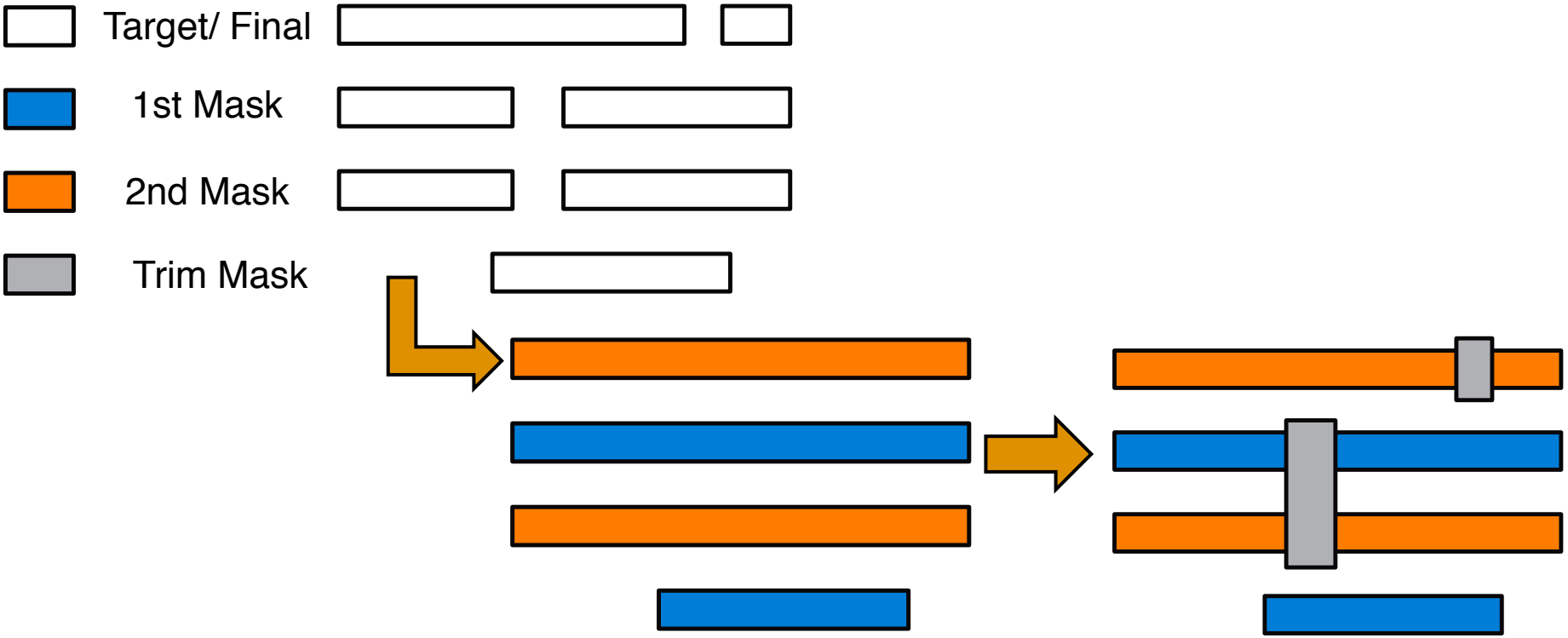
Triple Patterning Lithography (TPL)

- LELE-LE: Extend from LELE type double patterning
- Main challenge: layout decomposition
- Native conflicts



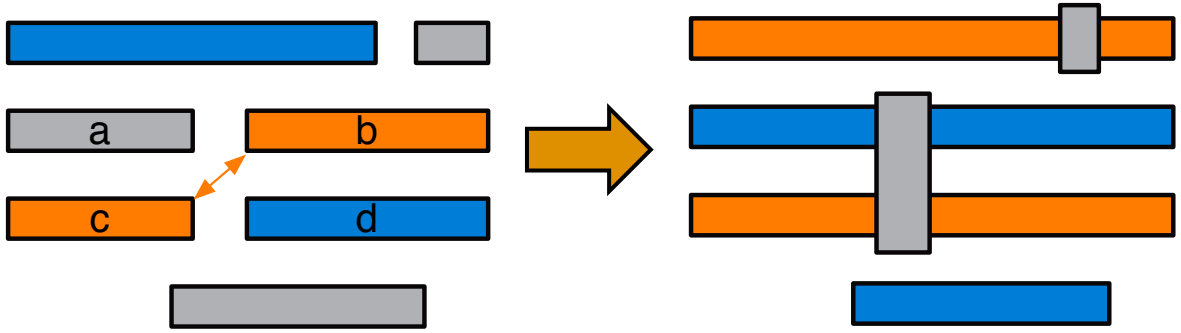
TPL with End-Cutting (LELE-EC)

- New TPL manufacturing process [Lin, ISPD'12]
- LELE + end cutting (trim mask)

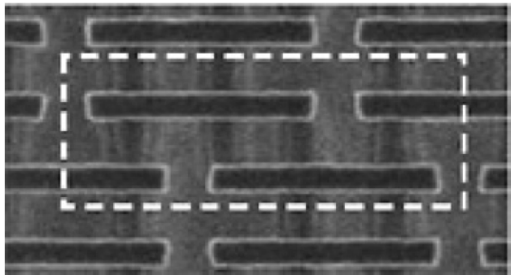


Why LELE-EC ?

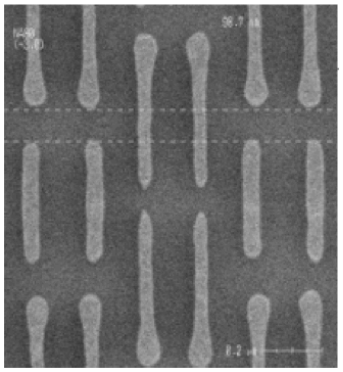
- Remove 4-clique native conflict in LELE-LE
  - Common even in regular layout

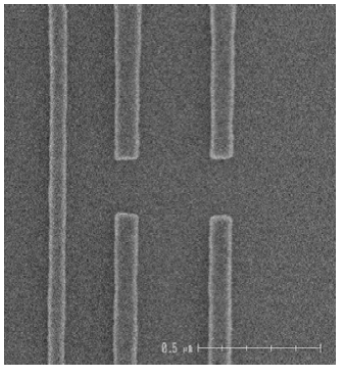


- Square-shaped Line-ends



[Y. Bordovsky, SPIE'05]

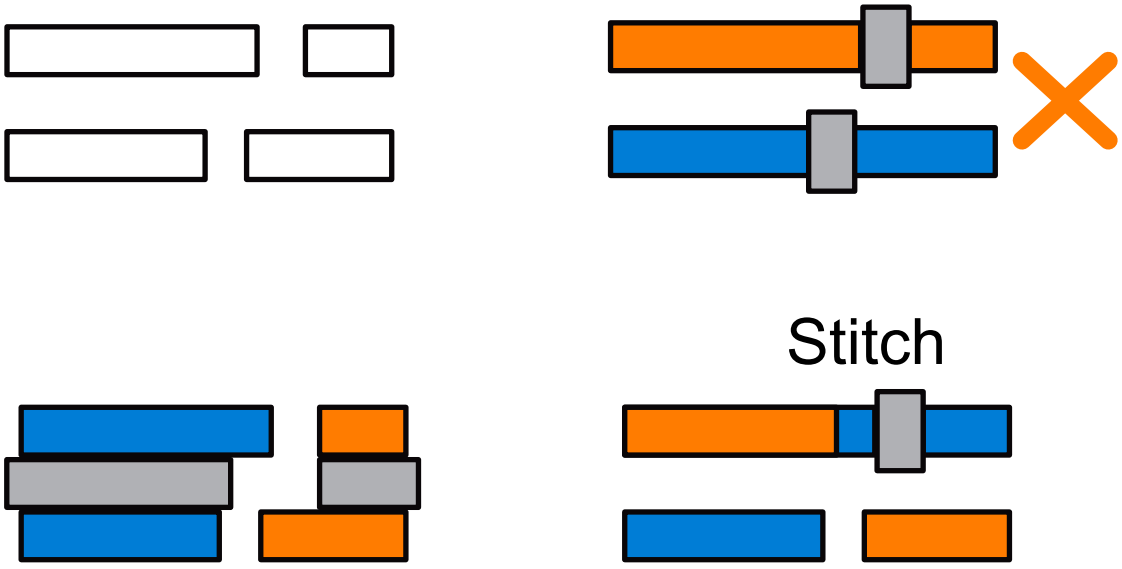




[B. Lin, ISPD'12]

LELE-EC: no free lunch

- New design constraints:
  - Min distance among end-cuts



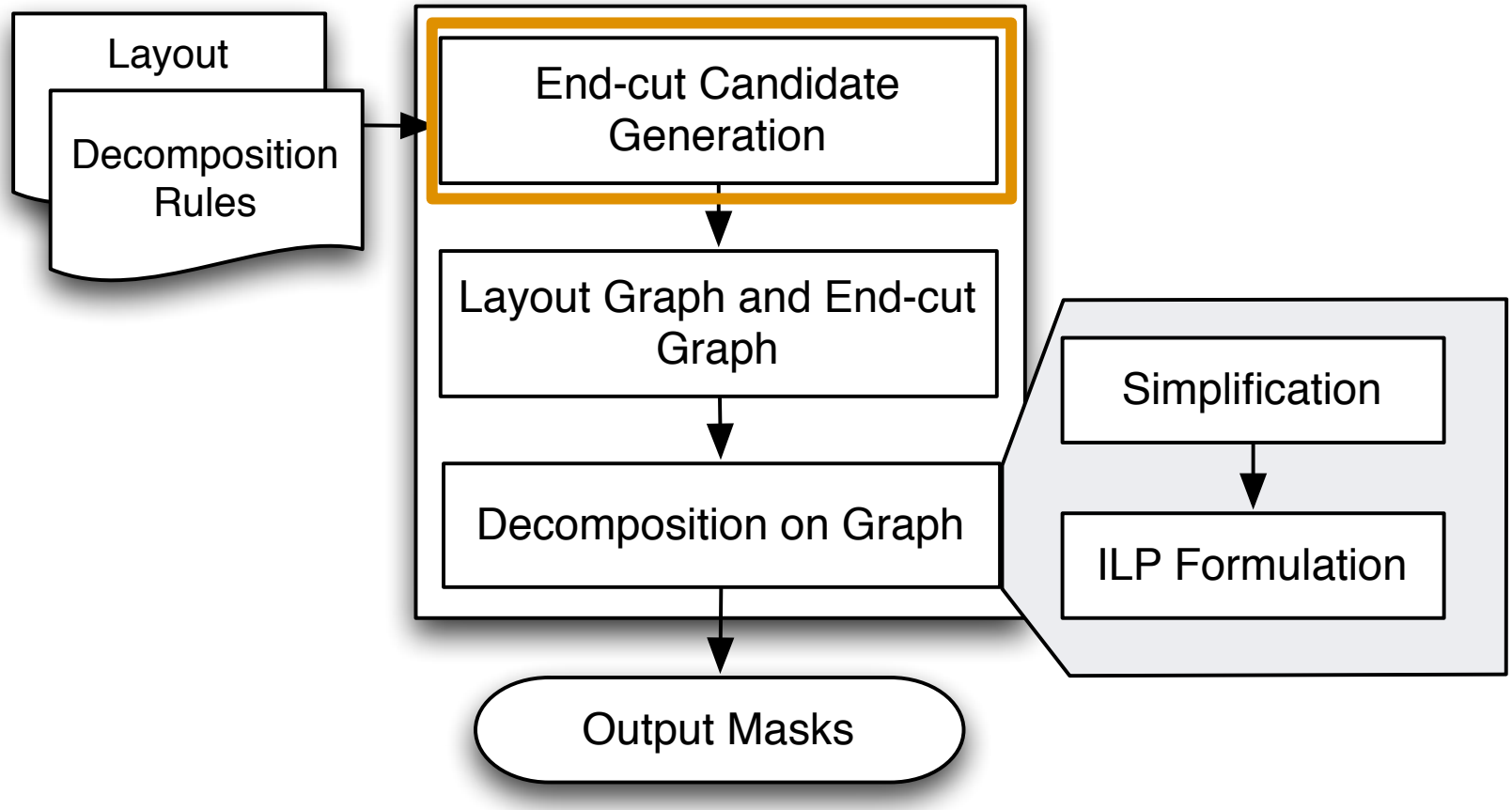
Solution to simultaneously assign colors and assign end-cuts

Previous Works

- LELE-LE layout decomposition
  - Mathematical programming [Cork, SPIE'08; Yu, ICCAD'11]
  - Heuristic methods [Ghaida, SPIE'12; Fang, DAC'12]
  - Polynomial time checking [Tian, ICCAD'12/SPIE'13]
- LELE-LE aware routing [Ma, DAC'12; Lin, ICCAD'12]

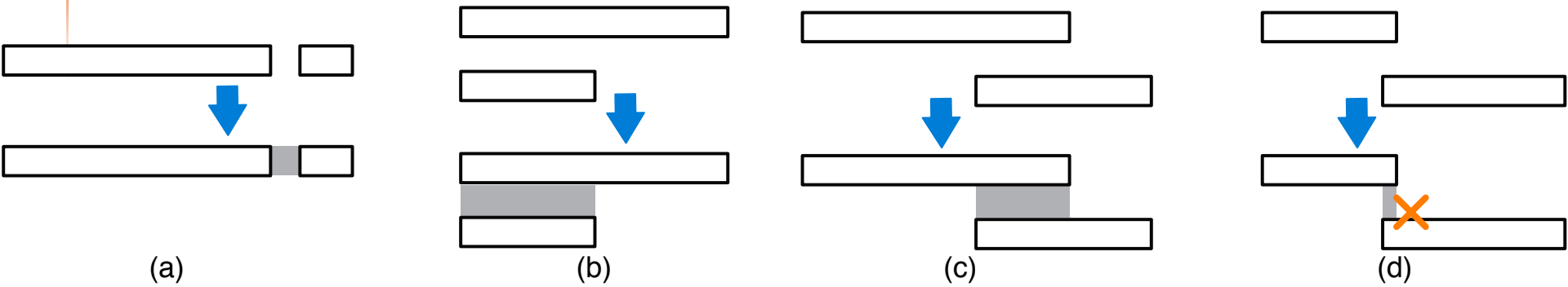
- First study for LELE-EC type triple patterning
  - Can borrow previous idea ?

Layout Decomposition Flow

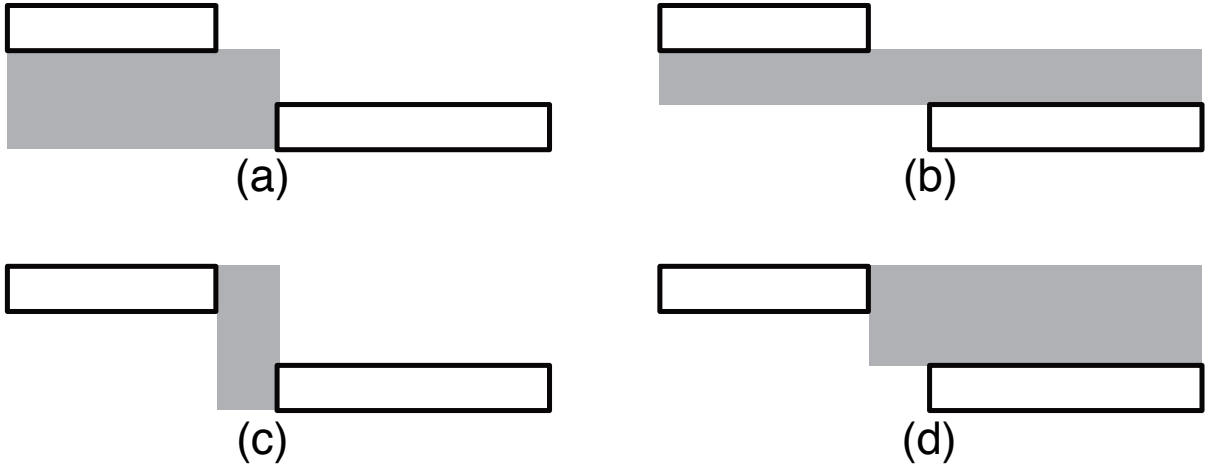


End-cut Candidate Generation

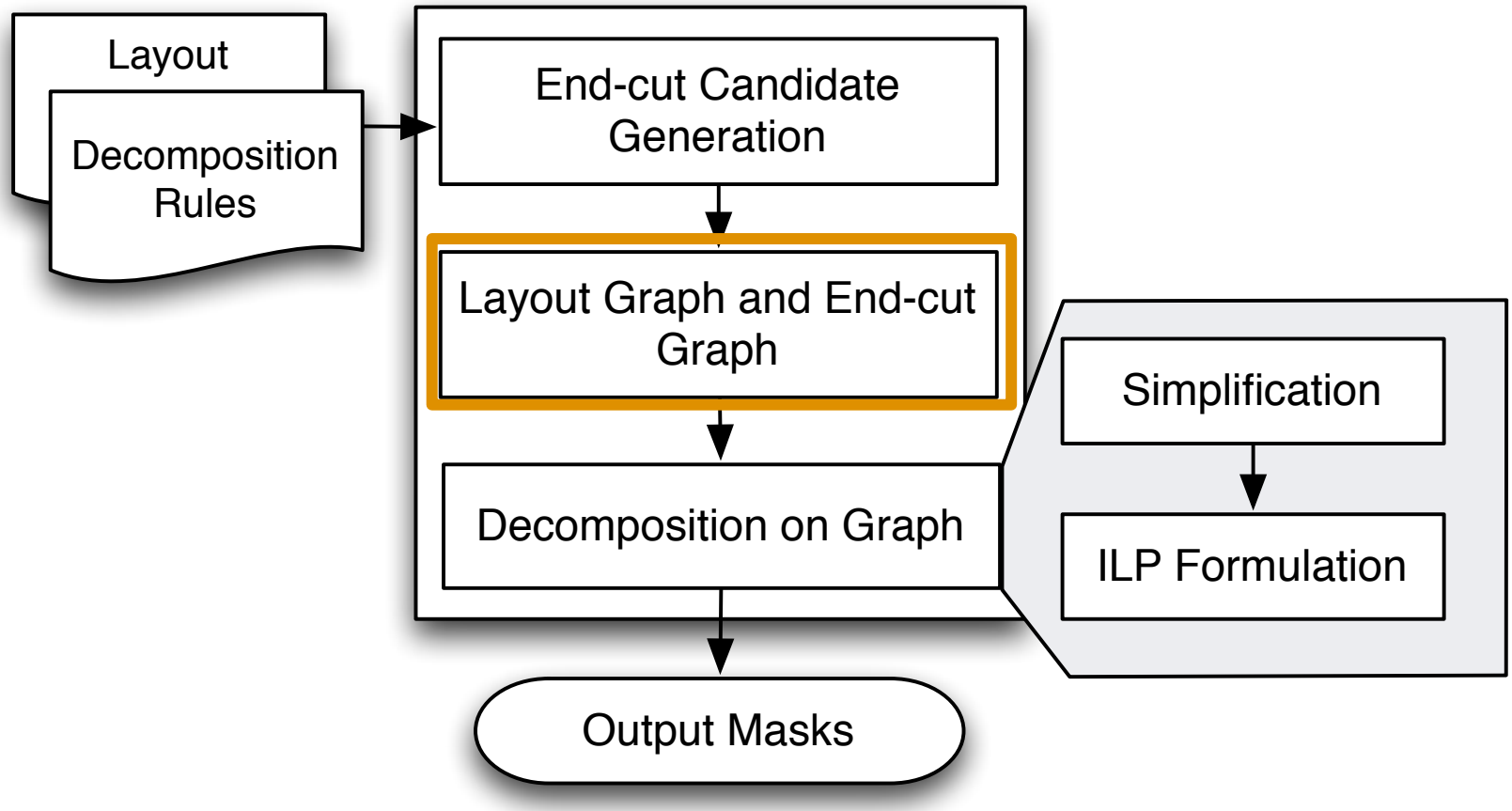
- Edge-edge



- Corner-corner

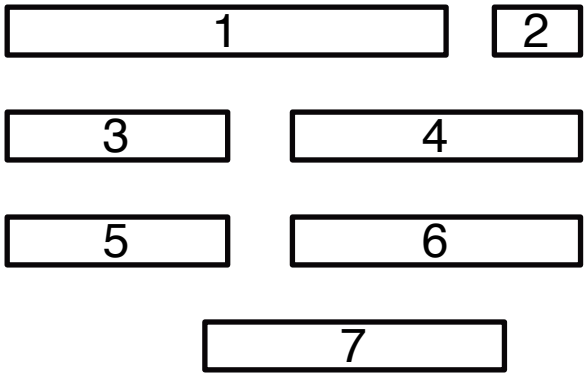


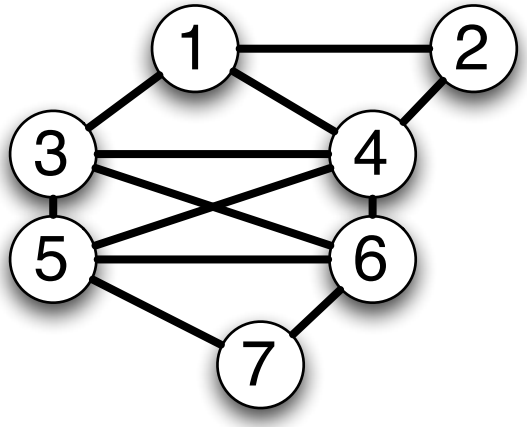
Layout Decomposition Flow

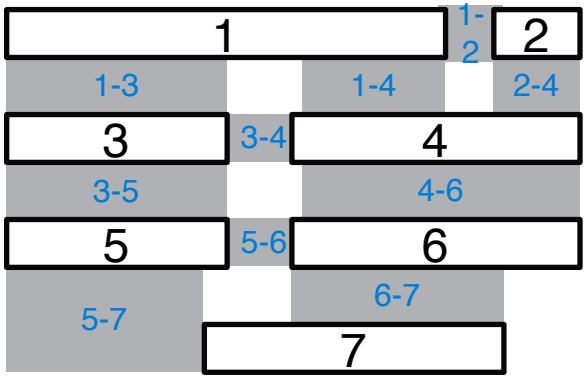


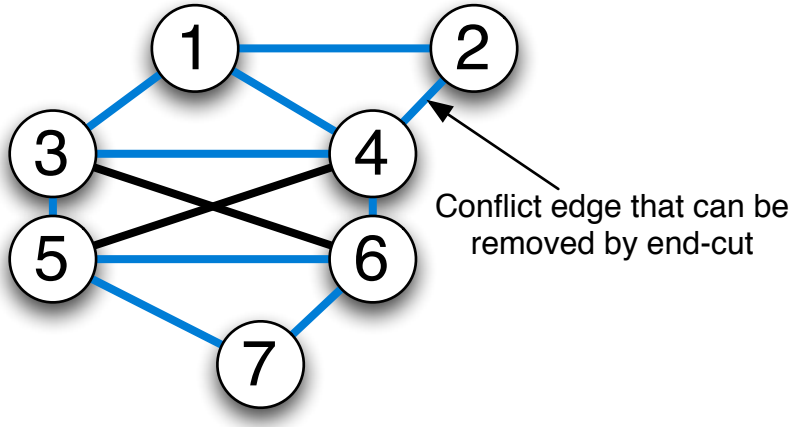
Layout Graph

- Layout topologies ➔ graph model
- Layout graph: feature info and end-cut candidate info



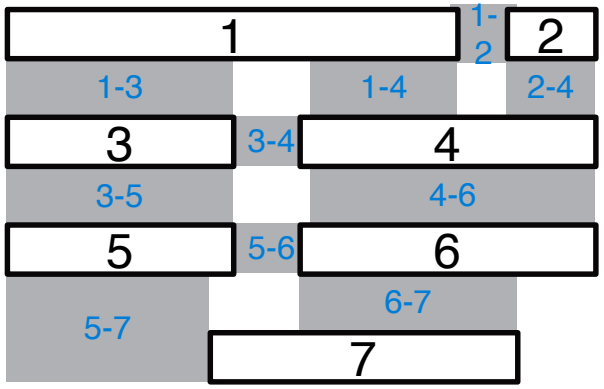


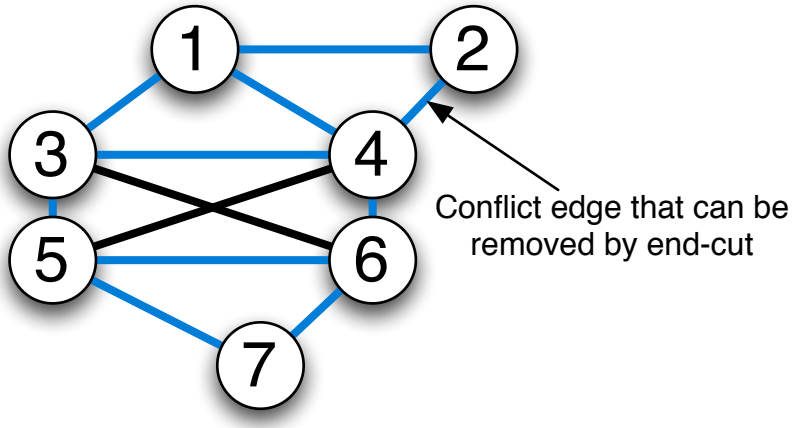


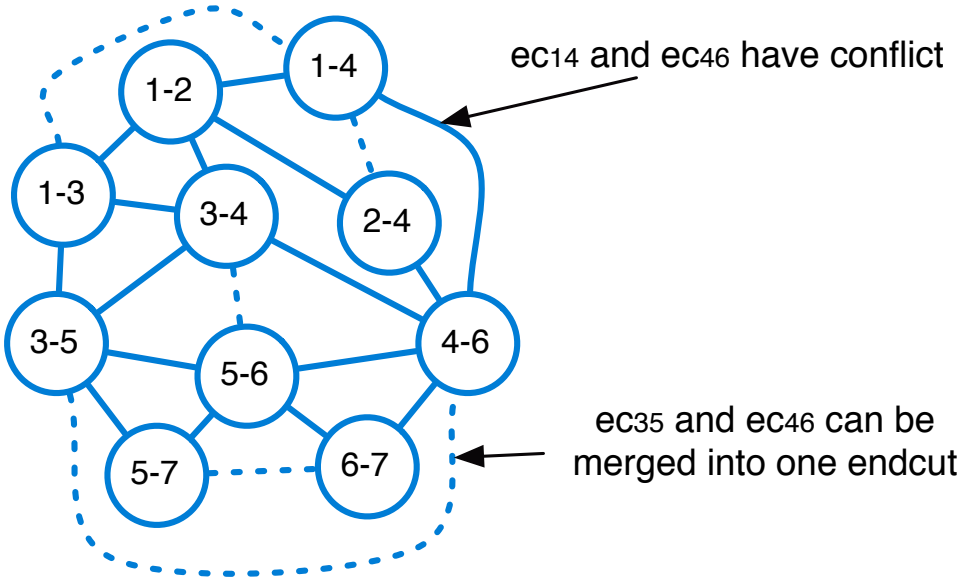


End-cut graph

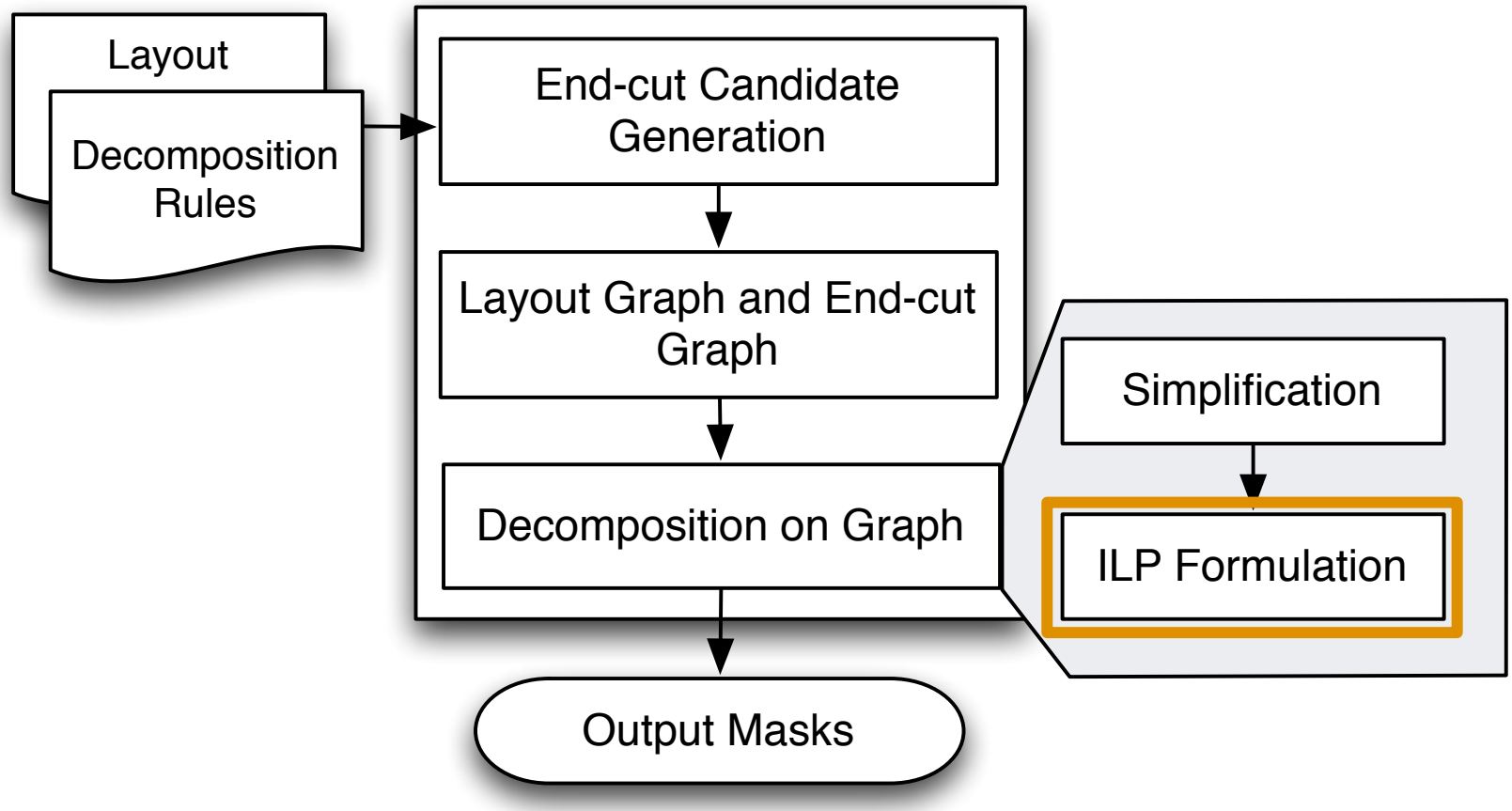
- Some end-cuts are conflict, while some can be merged
- New graph to store the end-cut relationships
  - conflict edge (solid): two candidates are conflict
  - merge edge (dash): two candidates can be merged







Layout Decomposition Flow



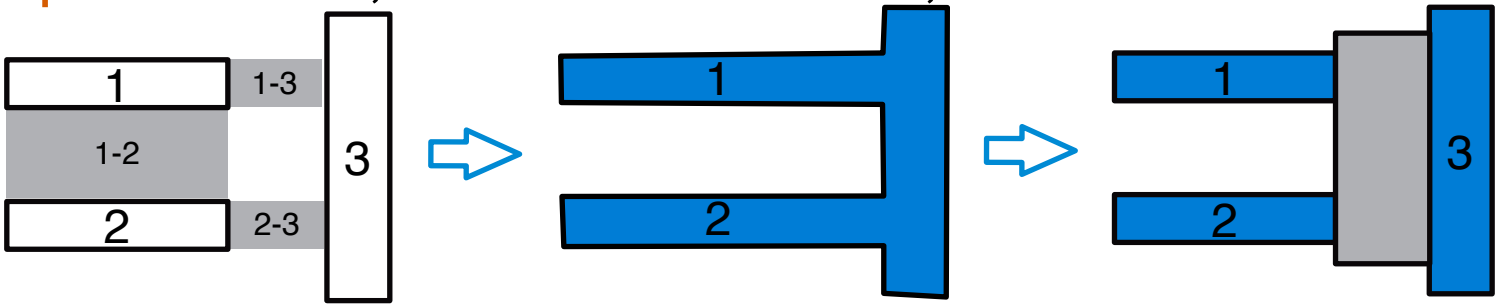
ILP Formulation

- CE: edge set of layout graph
- EE: conflict-edge set of end-cut graph

$$\min \sum_{e_{ij} \in CE} c_{ij}$$

$$\begin{aligned}
\text{s.t.} \quad & x_i + x_j \leq 1 + c_{ij} + ec_{ij} && \forall e_{ij} \in CE \\
& (1 - x_i) + (1 - x_j) \leq 1 + c_{ij} + ec_{ij} && \forall e_{ij} \in CE \\
& ec_{ij} + ec_{pq} \leq 1 && \forall e_{ijpq} \in EE \\
& ec_{ij} + x_i - x_j \leq 1 && \forall e_{ij} \in CE \\
& ec_{ij} + x_j - x_i \leq 1 && \forall e_{ij} \in CE
\end{aligned}$$

Exception: $x_1 = x_2$, since $ec_{13} = ec_{23} = 1$, ec_{12} can be 0

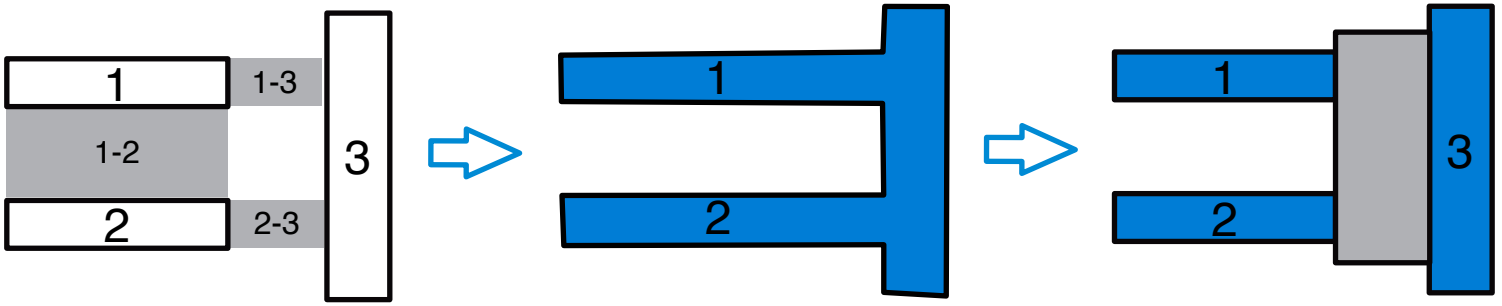


ILP Formulation (cont.)

$$\min \sum_{e_{ij} \in CE} c_{ij}$$

Non-linear

$$\begin{aligned}
\text{s.t.} \quad & x_i + x_j \leq 1 + c_{ij} + ec_{ij} + \gamma_{ik,jk} && \forall e_{ij} \in CE \\
& (1 - x_i) + (1 - x_j) \leq 1 + c_{ij} + ec_{ij} + \gamma_{ik,jk} && \forall e_{ij} \in CE \\
& ec_{ij} + ec_{pq} \leq 1 && \forall e_{ijpq} \in EE \\
& ec_{ij} + x_i - x_j \leq 1 && \forall e_{ij} \in CE \\
& ec_{ij} + x_j - x_i \leq 1 && \forall e_{ij} \in CE \\
& ec_{ik} + ec_{jk} \leq \gamma_{ik,jk} + 1 \\
& ec_{ik} \geq \gamma_{ik,jk} \\
& ec_{jk} \geq \gamma_{ik,jk}
\end{aligned}$$



ILP Formulation (cont.)

- Consider stitch insertion
- SE: set of stitch edge candidates

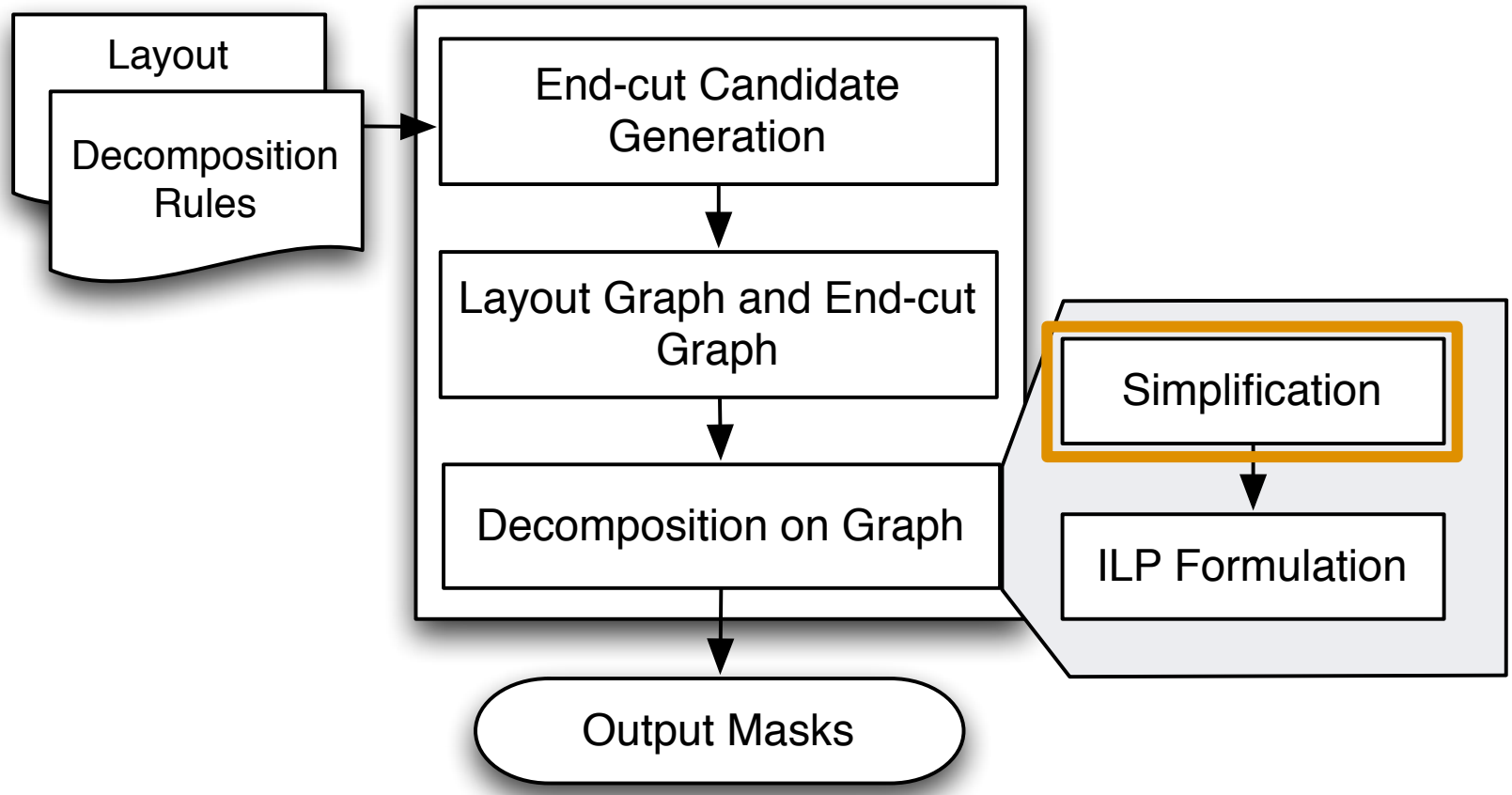
$$\min \sum_{e_{ij} \in CE} c_{ij} + \alpha \times \sum_{e_{ij} \in SE} s_{ij}$$

$$\text{s.t.} \quad x_i - x_j \leq s_{ij}$$

$$x_j - x_i \leq s_{ij}$$

Other constraints in previous ILP

Layout Decomposition Flow



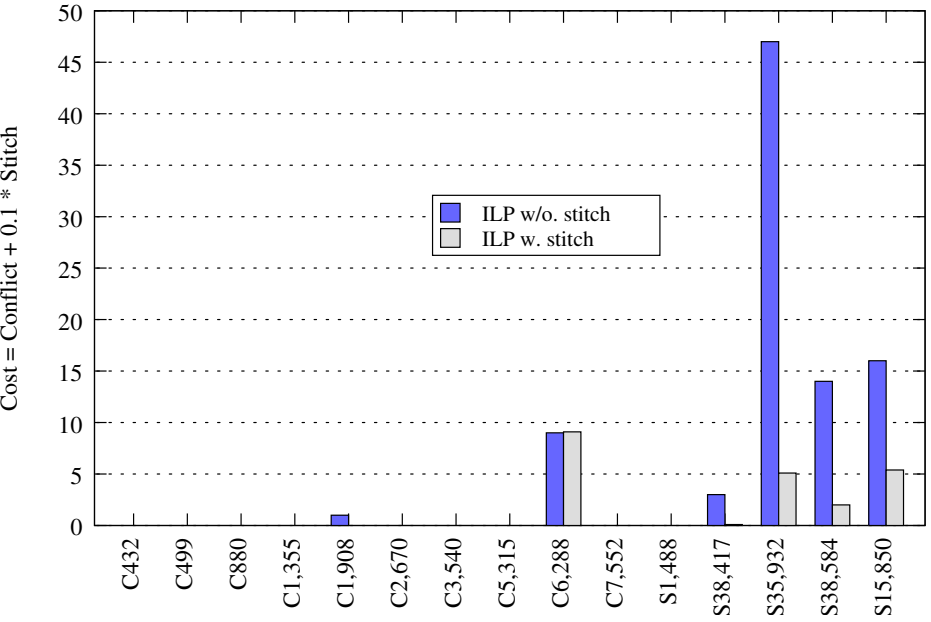
- Independent Component Computation
- Bridge Computation
- End-Cut Pre-Selection

Experimental Results

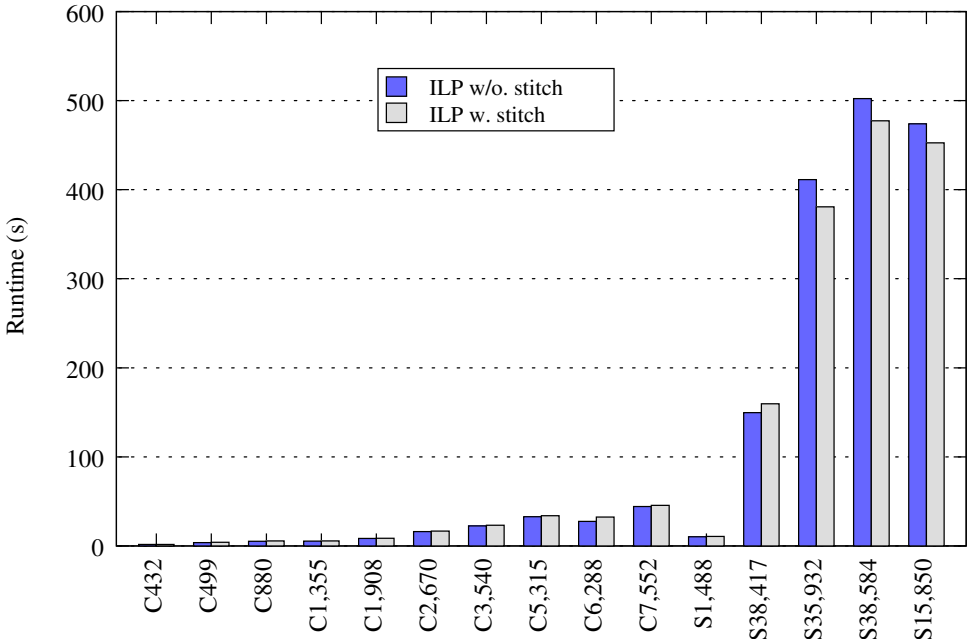
- Implement in C++
- 3.0GHz Linux machine with 32G RAM
- ISCAS 85&89 benchmarks from [Yu, ICCAD'11]
- Scaled to 14nm nodes
- ILP solver: GUROBI

Without or With stitch?

- Cost comparison (cost = conflict# + 0.1 * stitch#)
- Runtime comparison



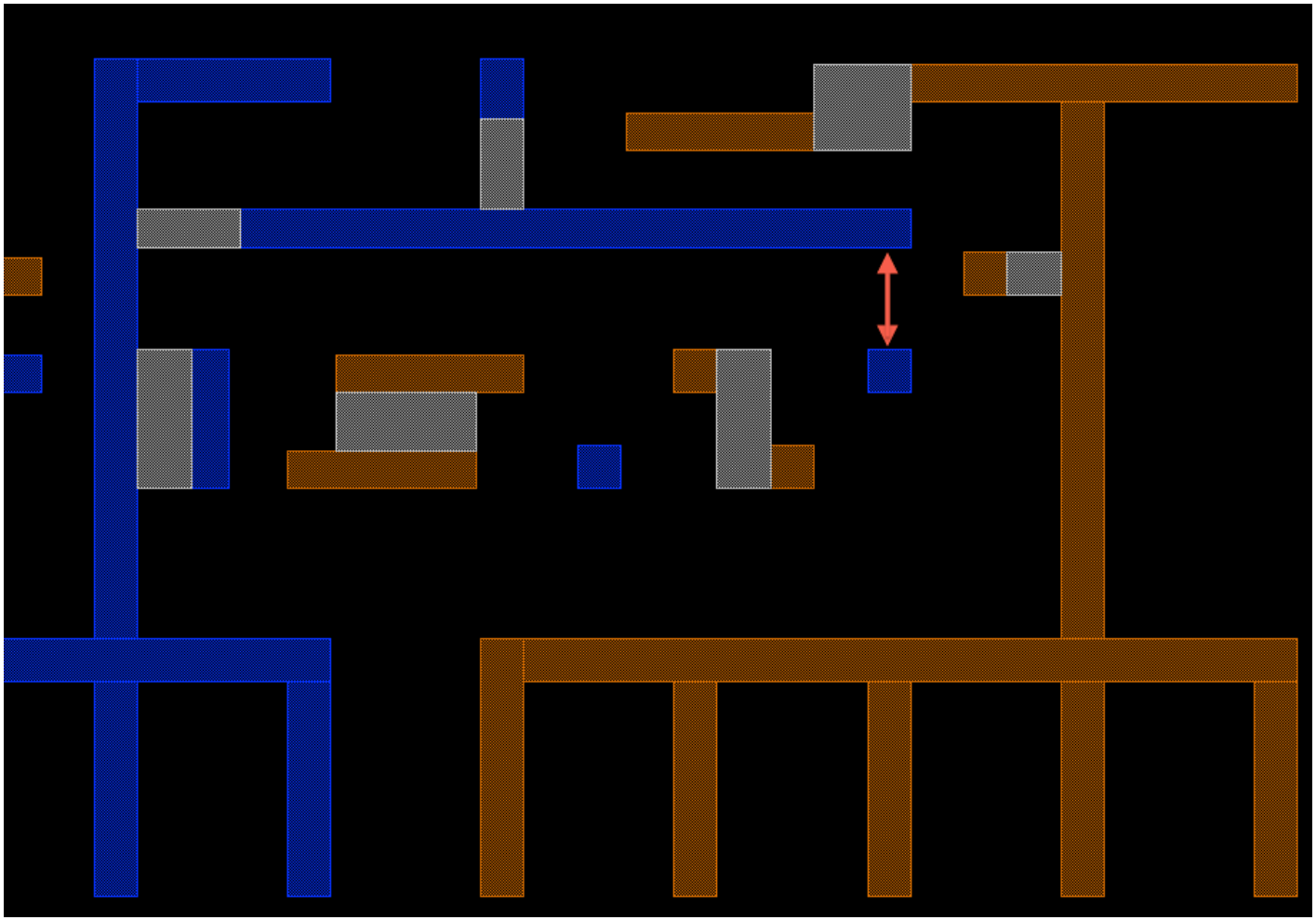
cost



runtime

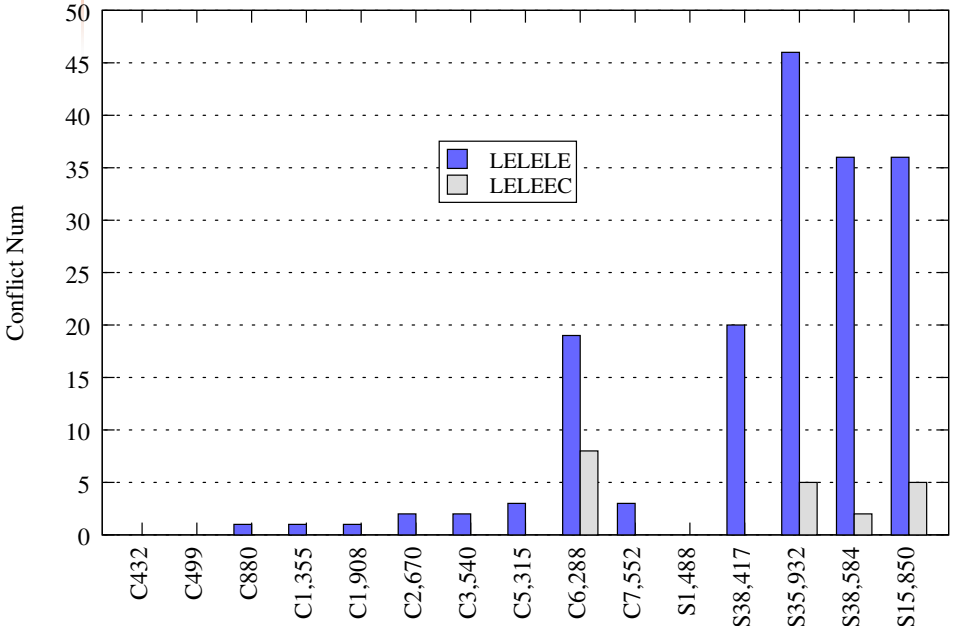
Conflict Example

- Irregular via array is dangerous

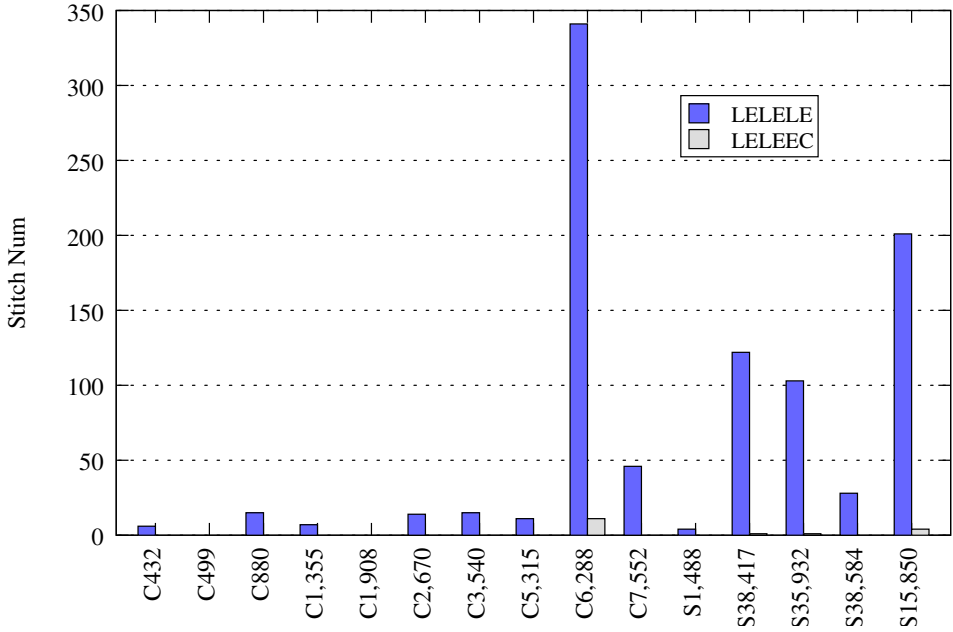


LELE-LE v.s. LELE-EC

- LELE-LE decomposer from [Fang, DAC'12]



conflict



stitch

Conclusion and Future Works

- First LELE-EC layout decomposition problem
- ILP formulation and speedup techniques
- Less conflict & stitch compared with LELE-LE

TPL is candidate for 14nm node.

- More research on TPL(LELEEC)-aware design

Thank You