A decorative graphic on the left side of the slide, consisting of a vertical, teardrop-shaped area filled with a colorful, multi-colored grid pattern. The colors transition from purple and blue at the top to green and yellow at the bottom.

DSAR: DSA aware Routing with Simultaneous DSA Guiding Pattern and Double Patterning Assignment

Jiaojiao Ou¹, Bei Yu², Xiaoqing Xu¹, Joydeep Mitra³, Yibo Lin¹, David Z. Pan¹

¹ECE Department, University of Texas at Austin

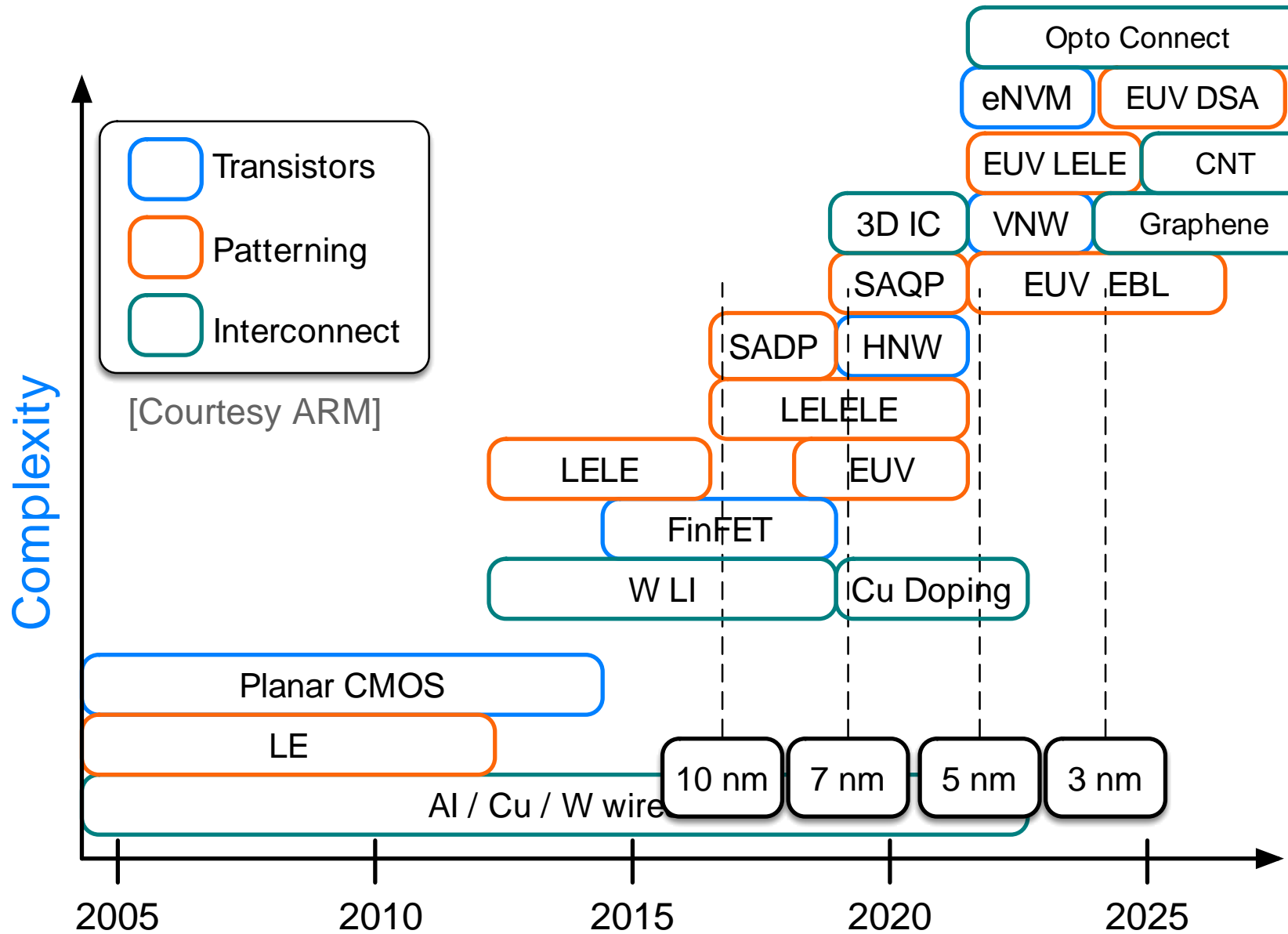
²CSE Department, CUHK

³Mentor Graphics Inc.

Outline

- ◆ Introduction
- ◆ Problem formulation
- ◆ Detailed routing algorithms
- ◆ Experimental results
- ◆ Conclusion

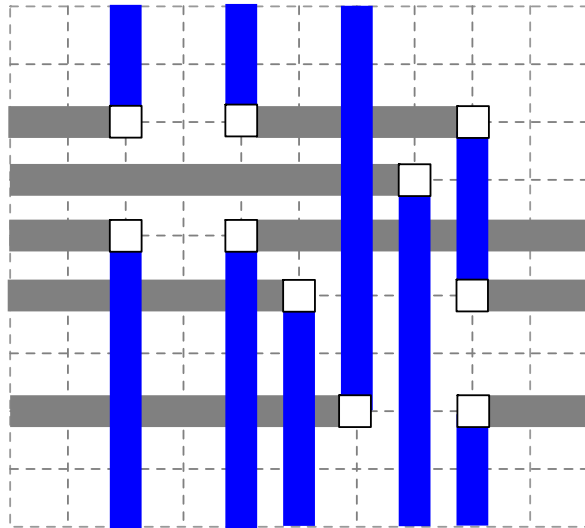
Introduction: Technology Scaling



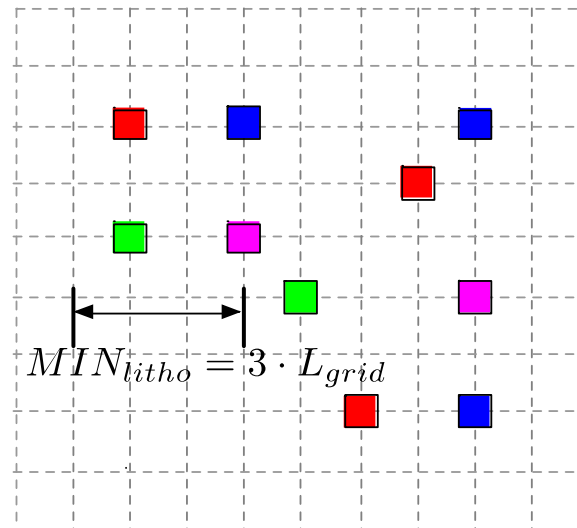
Technology Scaling: More Masks

◆ Via density increases

- ✓ Triple/Quadruple patternings are required
 - Placement error problem
 - Cost increases
- ✓ More via: 1D design



(a) Original layout



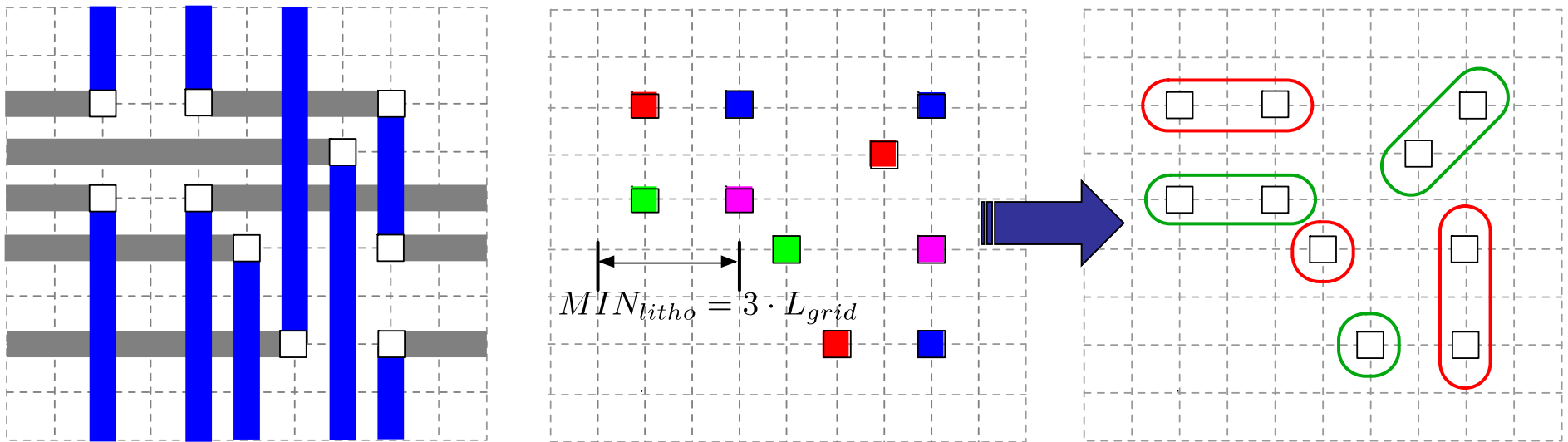
(b) Via layer with quadruple patterning

- Metal 2
- Metal 3
- Via
- Mask 1
- Mask 2
- Mask 3
- Mask 4



Motivation of DSA on Via Layer

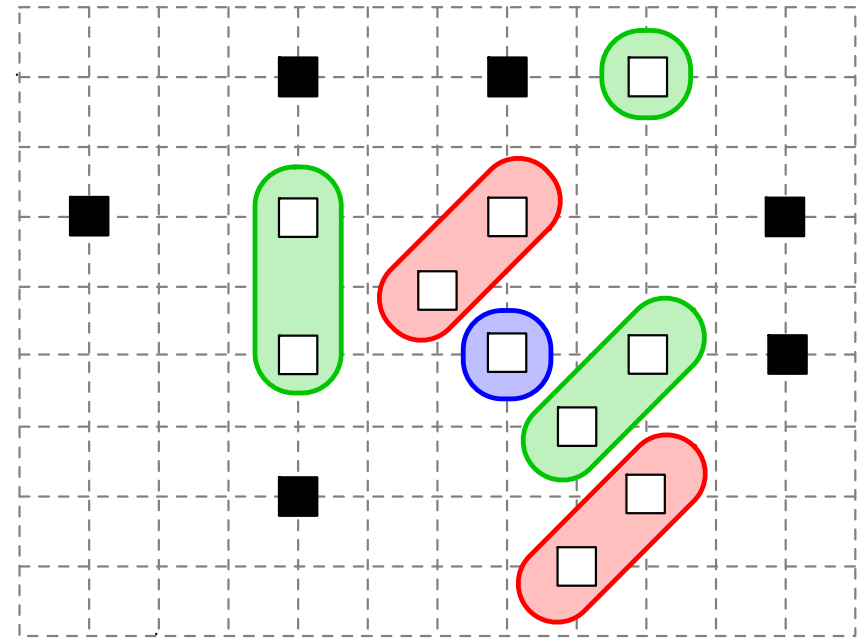
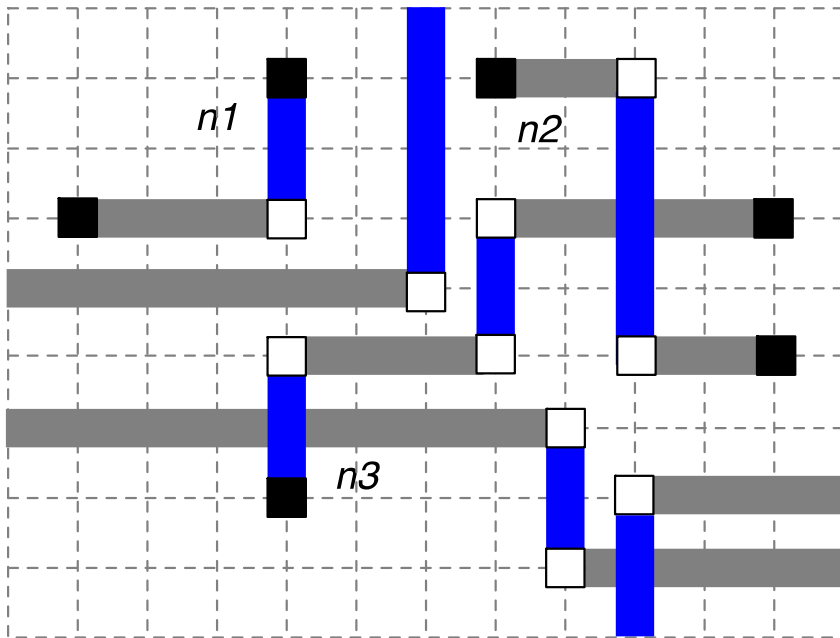
- ◆ Reduce mask number by grouping vias in the same guiding pattern



Reduce 2 mask with DSA 😊

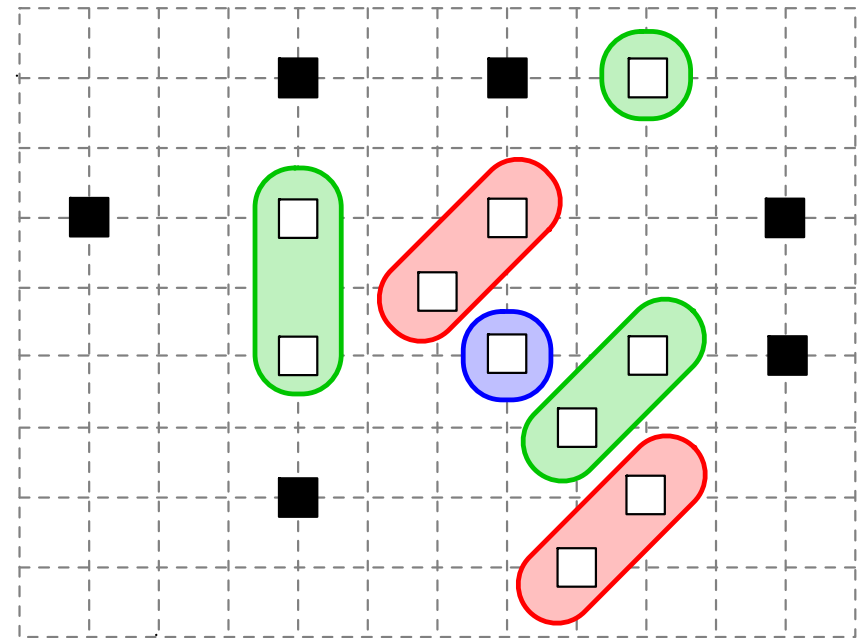
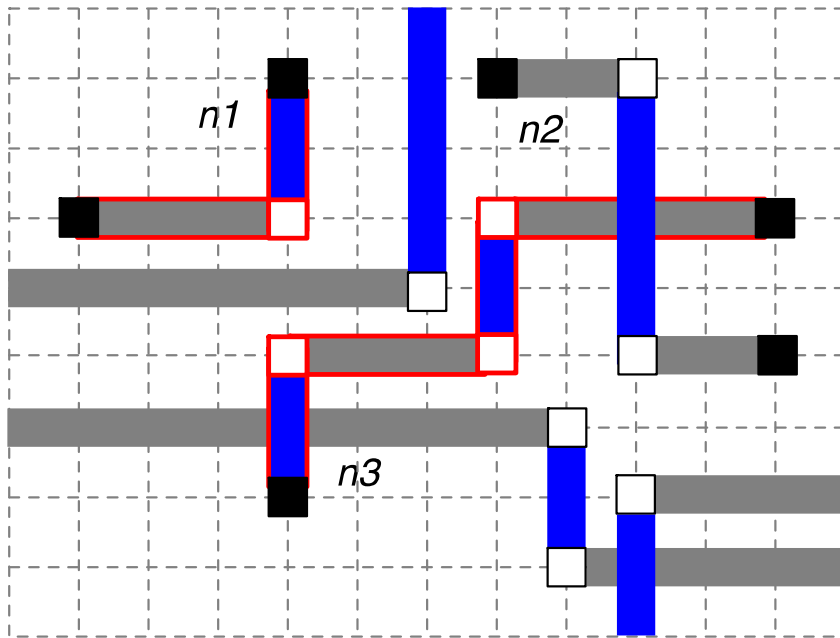
Consider DSA during Routing 1

- ◆ Initial detailed routing without DSA consideration
 - ✓ 3 masks are required



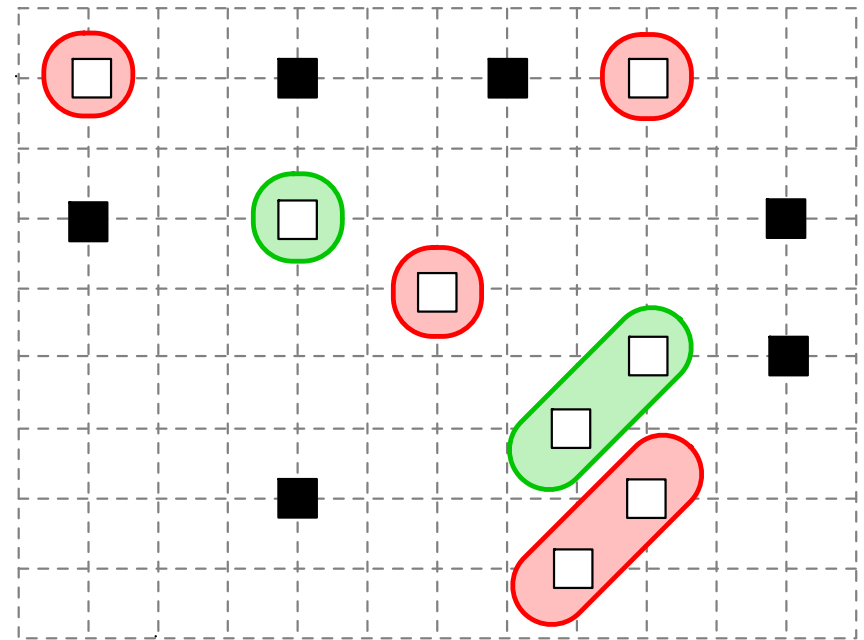
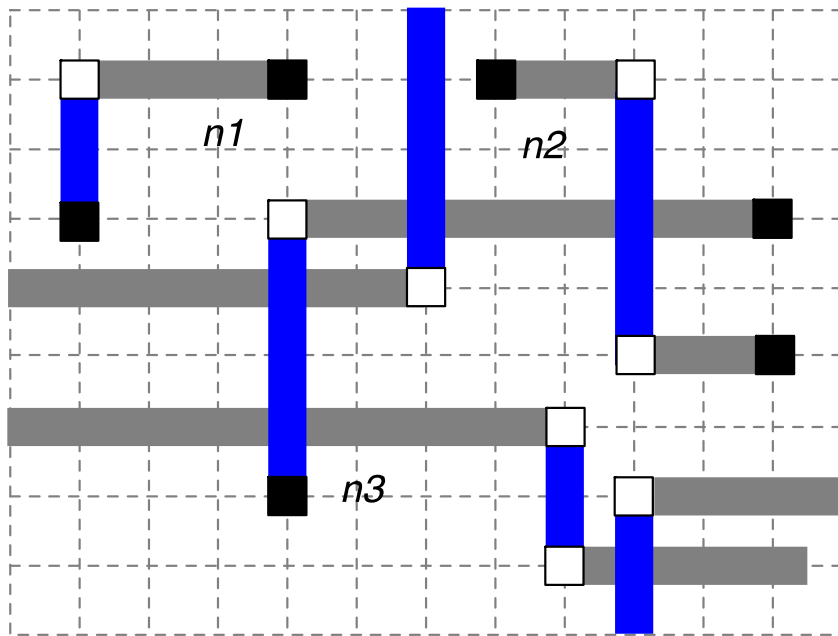
Consider DSA during Routing 2

- ◆ Initial detailed routing without DSA consideration
 - ✓ 3 masks are required
- ◆ Reroute $n1$ and $n3$



Consider DSA during Routing 3

- ◆ Initial detailed routing without DSA consideration
 - ✓ 3 masks are required
- ◆ Reroute $n1$ and $n3$
 - ✓ Reduce 1 more mask ☺



Previous Works

- ◆ DSA-aware detailed routing for via layer optimization [Du+, SPIE'14]
 - ✓ Resolve conflicts and infeasible via patterns during rip-up and reroute with negotiated congestion based scheme
 - ✓ Incapable to handle DSA with multiple patternings
 - ✓ More wire length may be introduced
- ◆ Redundant Via insertion consideration [Lin+, ASPDAC'17]
 - ✓ Simultaneously consideration of redundant via insertion and guiding template feasibility
 - ✓ Increase redundant via insertion rate
 - ✓ Multiple patterning for via is not considered, not compatible for 1D design

Problem Formulation: DSAR

DSA and double patterning aware detailed routing

Input:

- ▷ Netlist with source/target pins
- ▷ Feasible DSA patterns
- ▷ Design rules

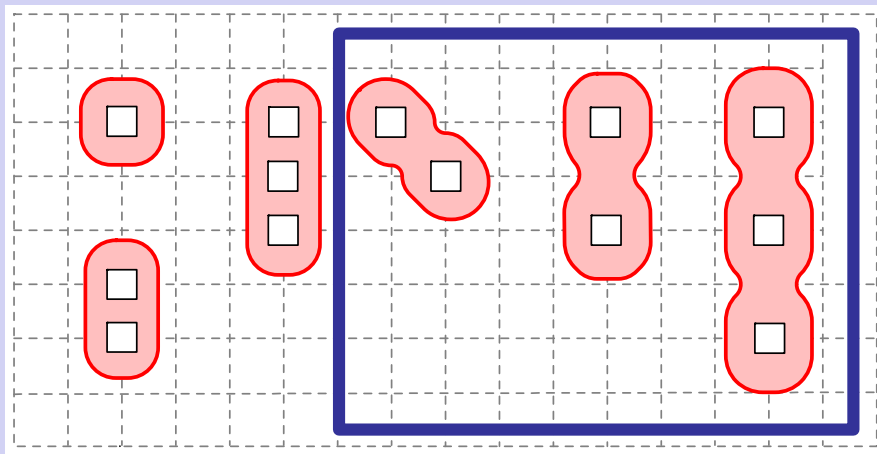
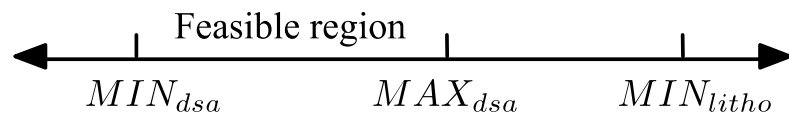
Output:

- ▷ Minimize wirelength, unroutable nets
- ▷ DSA-DP compatible via layer

Design Rules

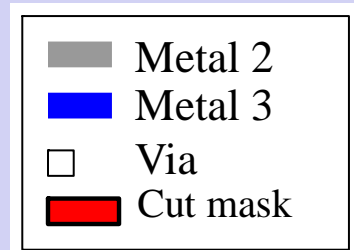
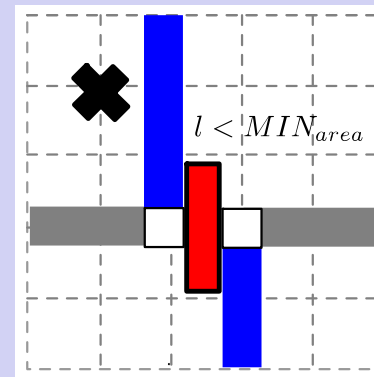
DSA design rules

$$L_{grid} > MIN_{dsa} \quad 2 \cdot L_{grid} \leq MAX_{dsa} < 3 \cdot L_{grid}$$



More complex
guiding templates

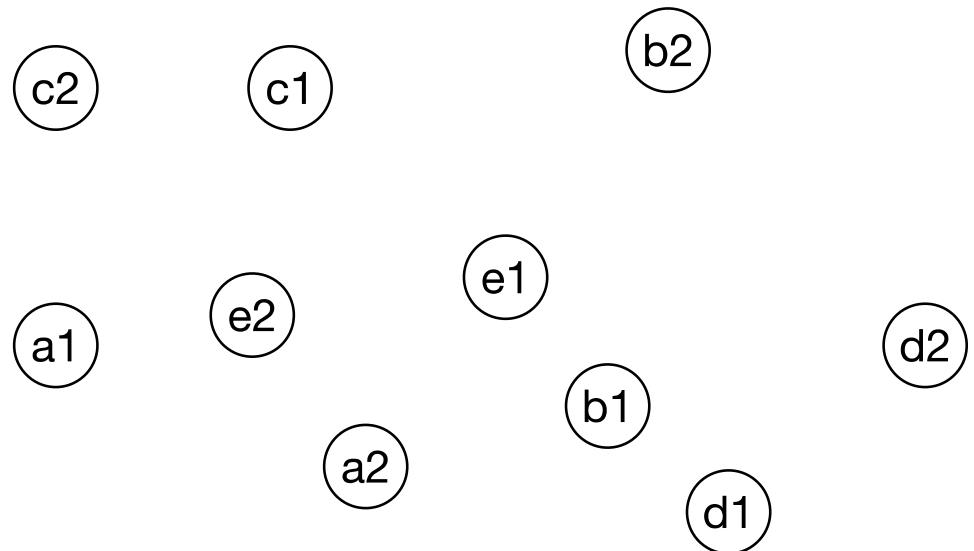
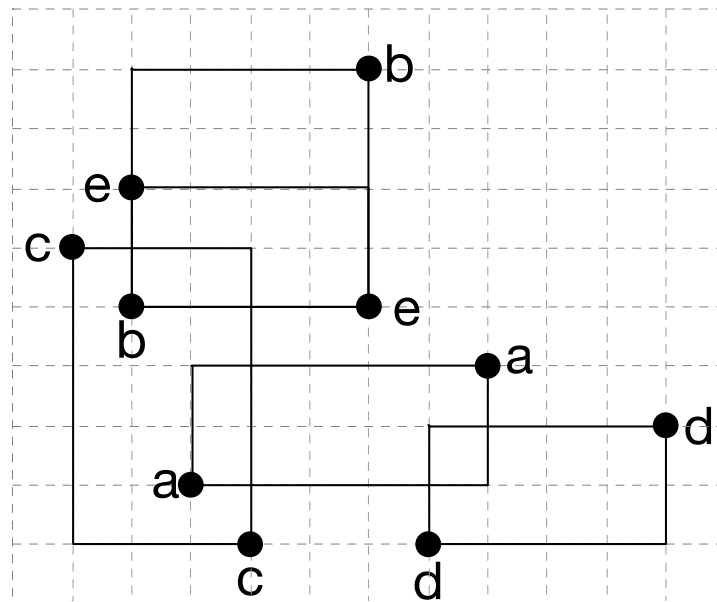
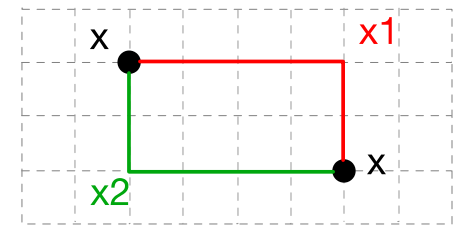
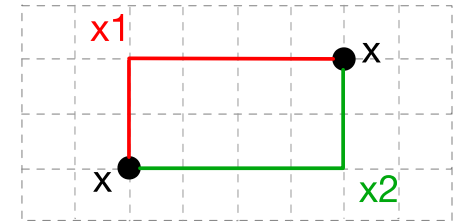
Forbidden Via distribution



Pre-route Net Planning - 1

◆ Construct conflict graph

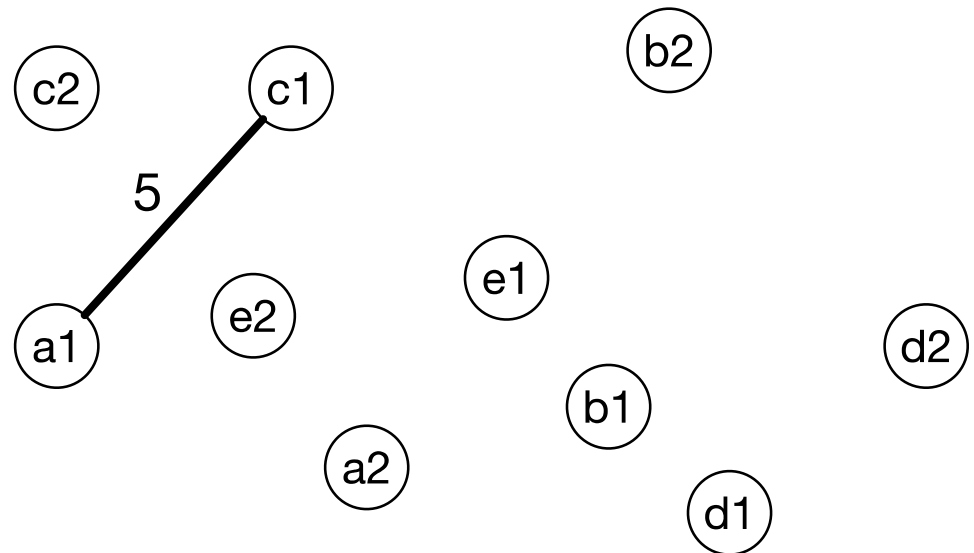
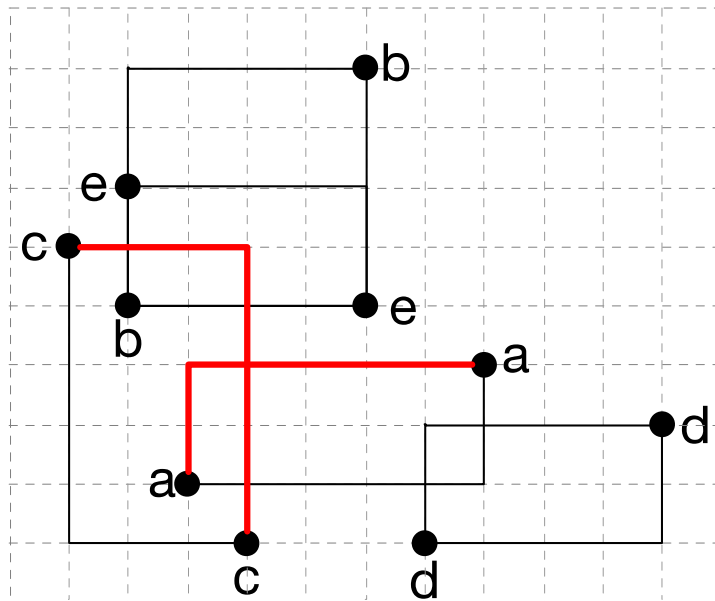
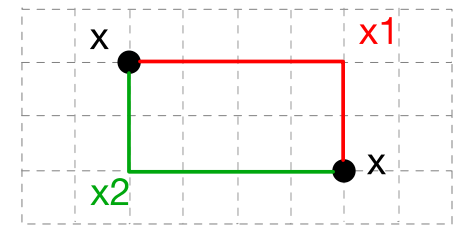
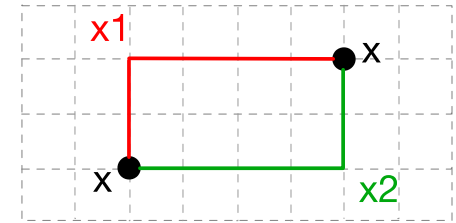
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 1

◆ Construct conflict graph

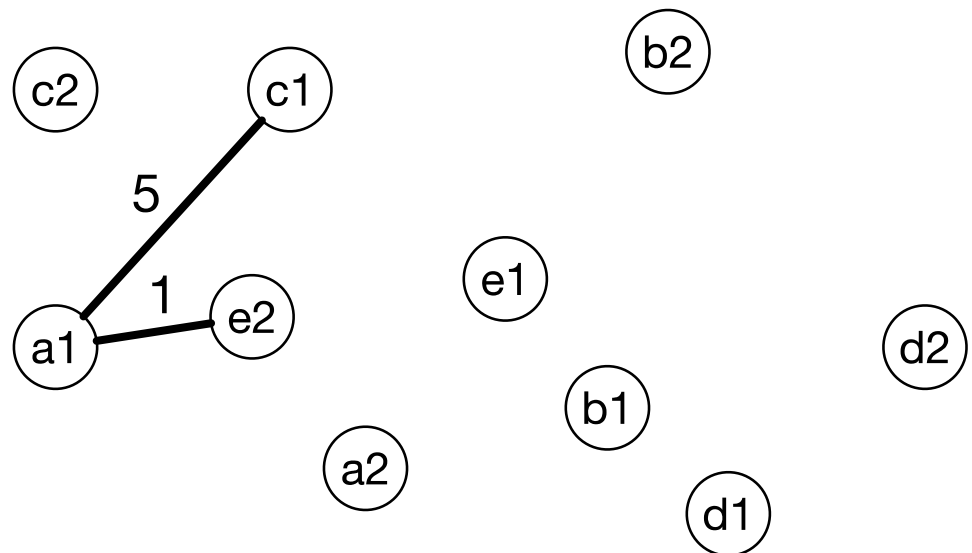
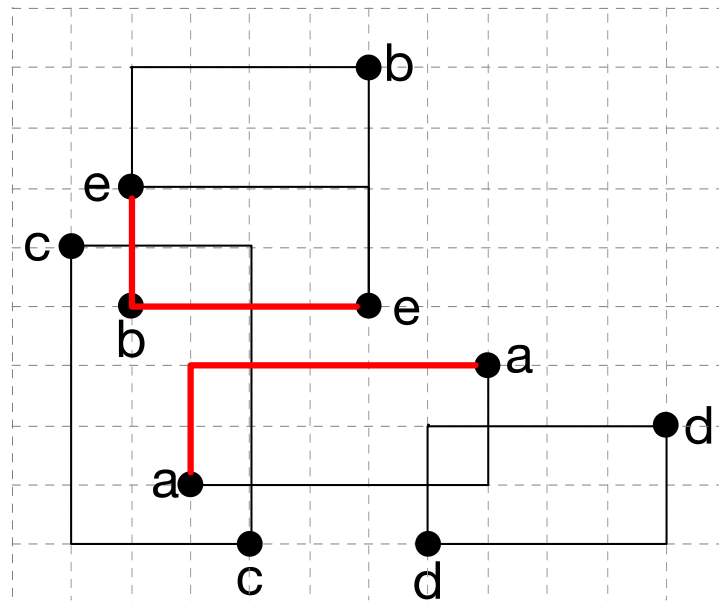
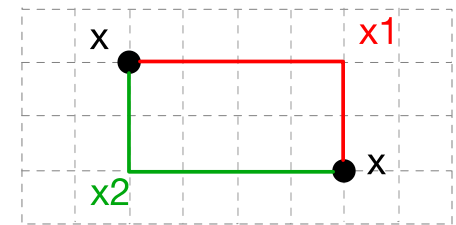
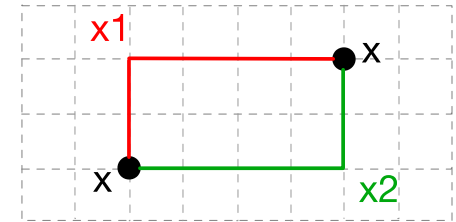
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 1

◆ Construct conflict graph

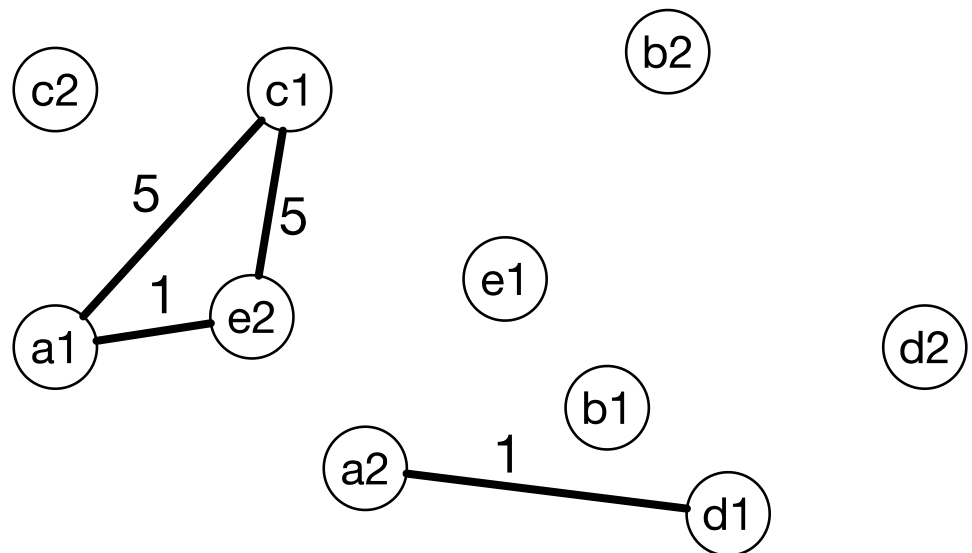
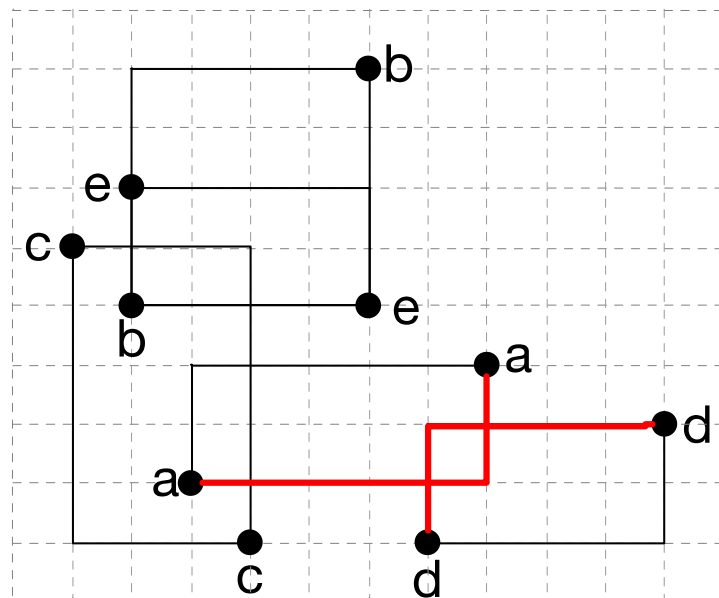
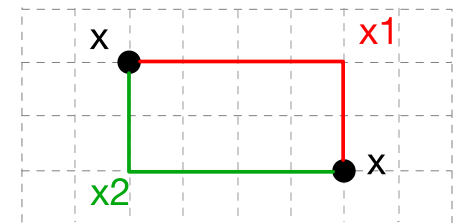
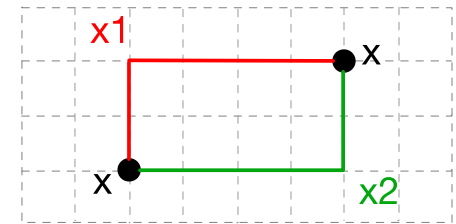
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 1

◆ Construct conflict graph

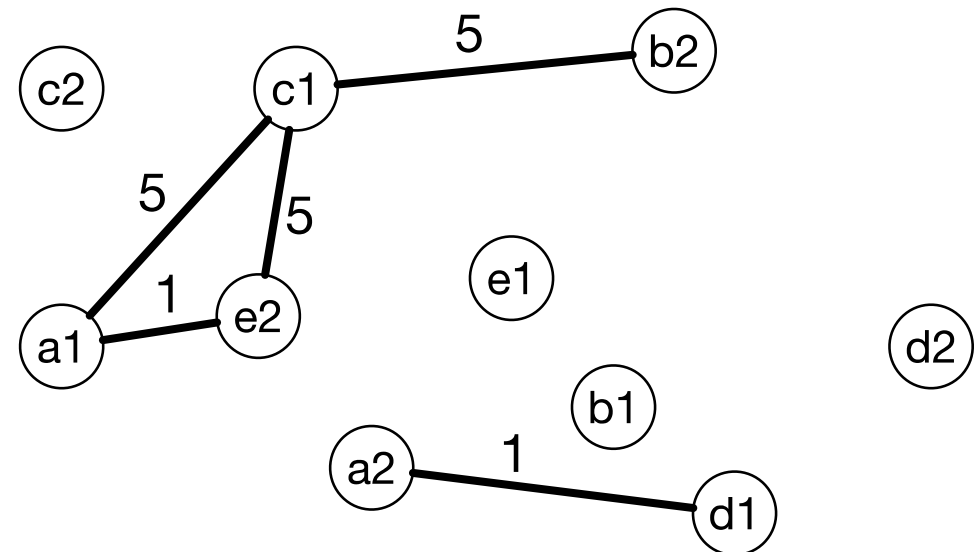
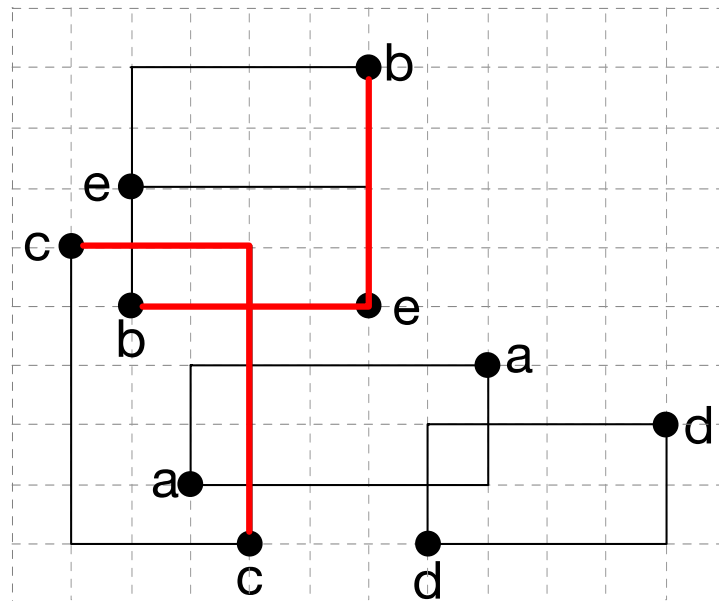
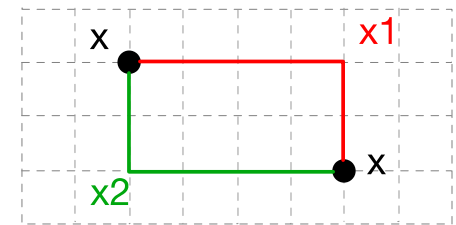
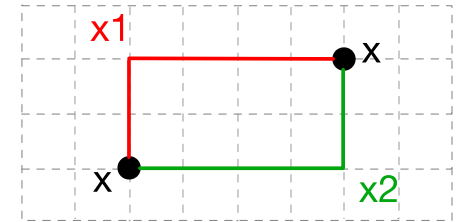
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 1

◆ Construct conflict graph

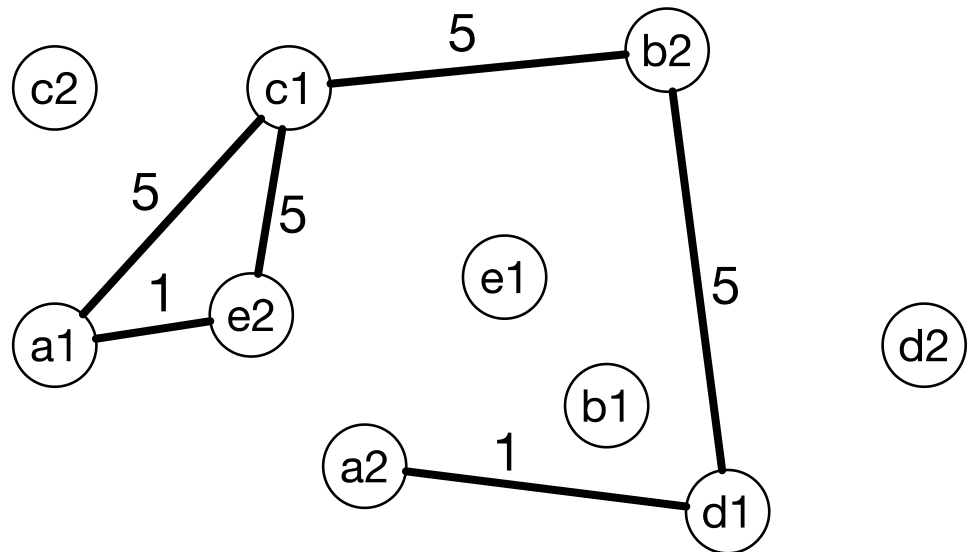
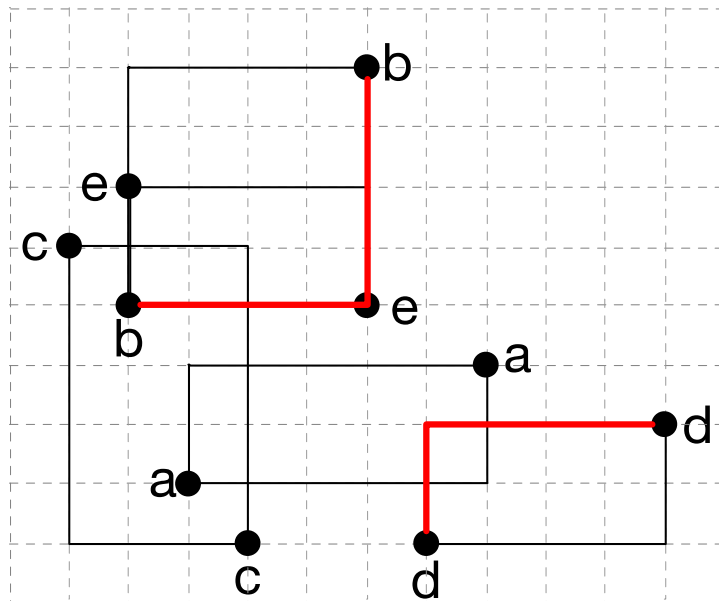
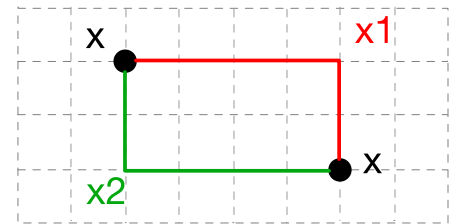
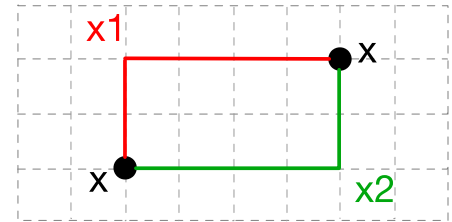
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 1

◆ Construct conflict graph

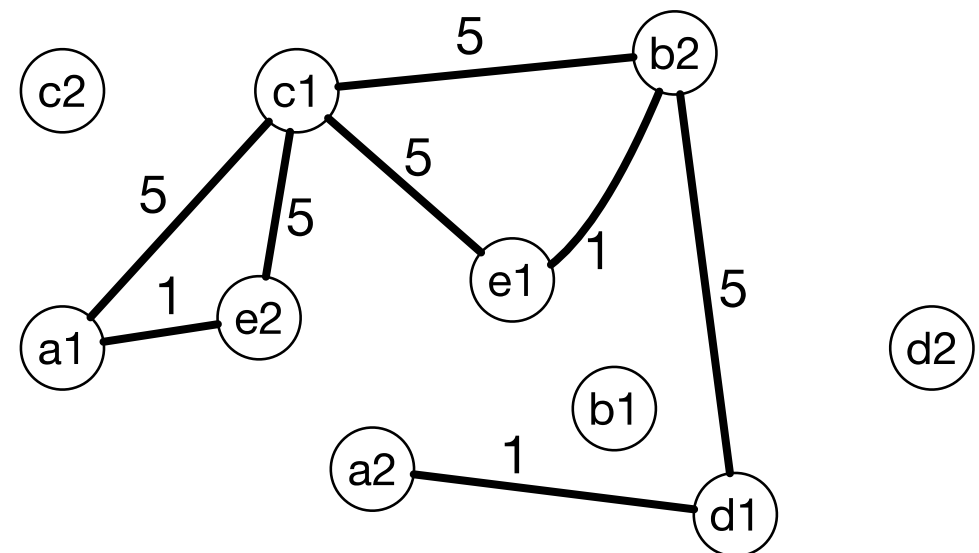
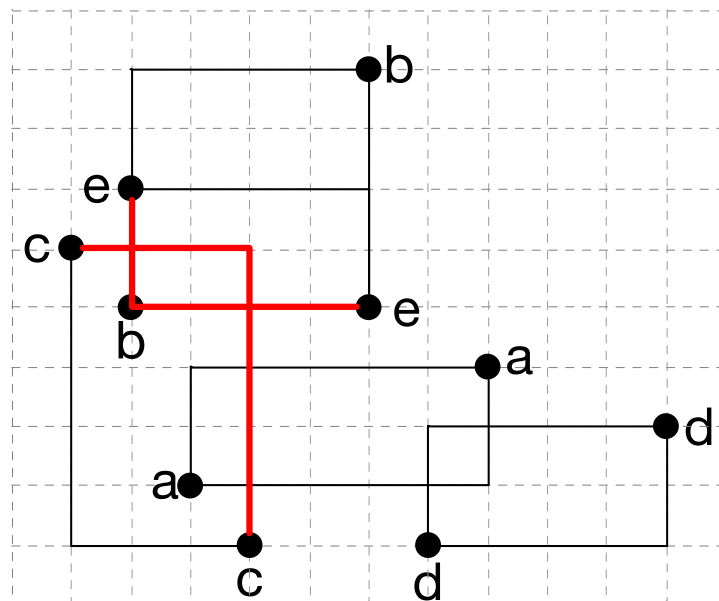
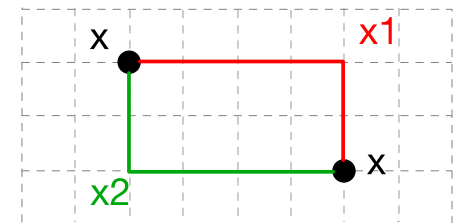
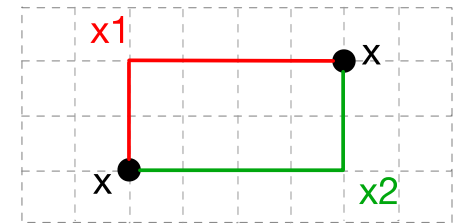
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 1

◆ Construct conflict graph

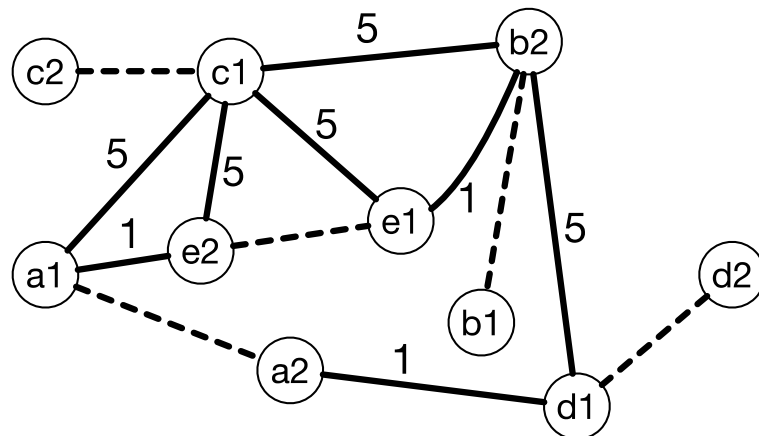
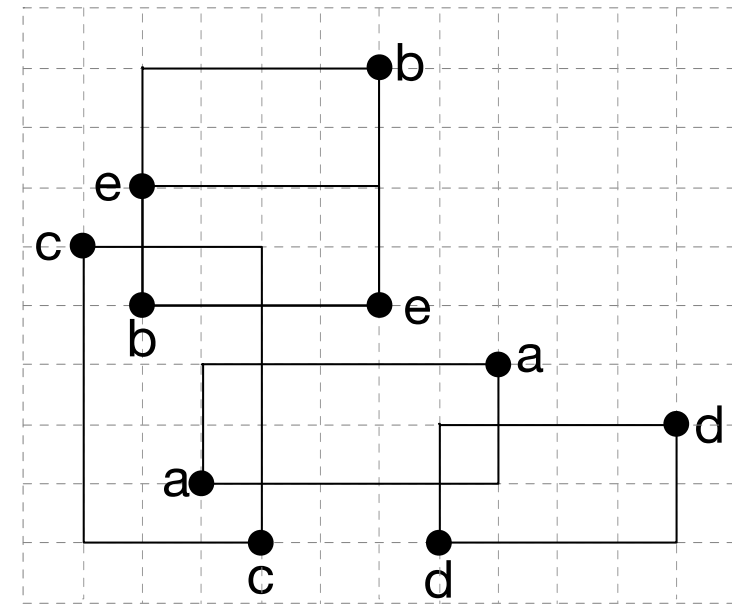
- ✓ Vertices: bbox corners
- ✓ Edge weight: DSA friendly? 1: 5



Pre-route Net Planning - 2

◆ Conflict graph constraints

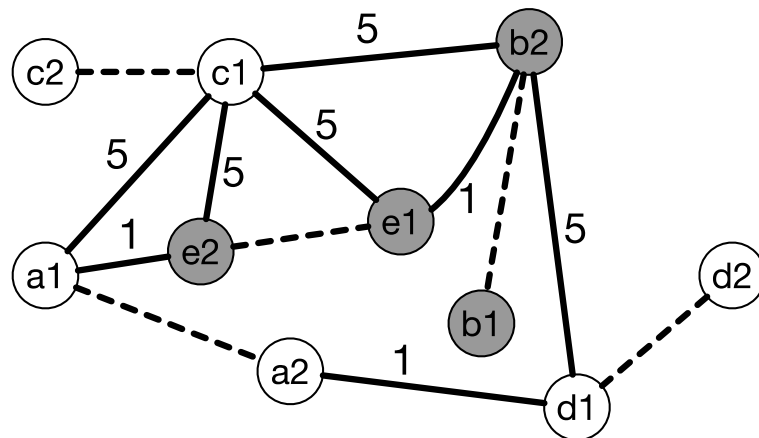
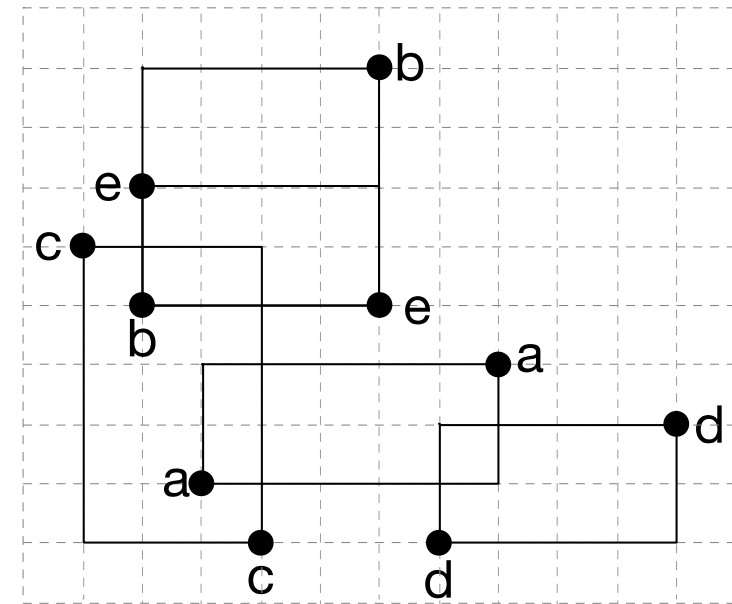
- ✓ At most 1 corner of each net



Pre-route Net Planning - 2

◆ Conflict graph constraints

- ✓ At most 1 corner of each net
- ✓ Corners cross the pins of other nets



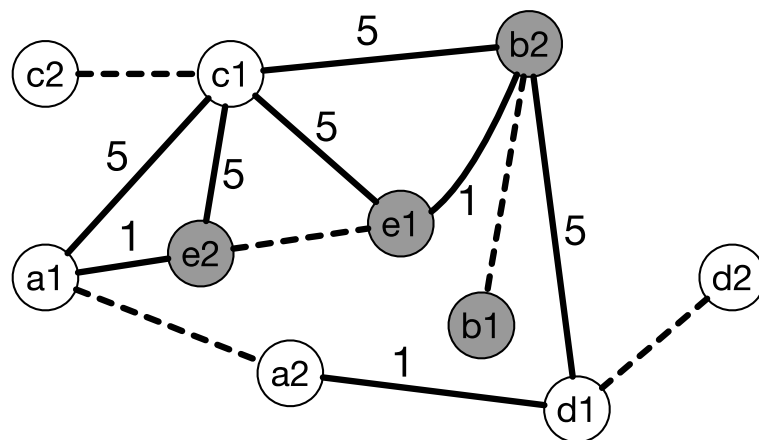
Pre-route Net Planning - 3

◆ Conflict graph constraints

- ✓ At most 1 corner of each net
- ✓ Corners cross the pins of other nets

◆ Conflict graph bipartization

- ✓ Pre-determine (estimate) the routing paths for nets



$$\min \sum_{i=1}^N (\alpha_i \times c_i) \quad \alpha_v = \frac{1}{\sum_{e \in V} Cost_e}$$

$$\mathbf{s.t.} \quad \begin{aligned} s_v + s_u + (c_v + c_u) &\geq 1, & \forall \{v, u\} \in E_s, \\ s_v + s_u - (c_v + c_u) &\leq 1, & \forall \{v, u\} \in E_s, \\ c_v + c_u &\geq 1, & \forall \{v, u\} \in E_d, \\ c_v &= 1, & \forall v \in V_f. \end{aligned}$$

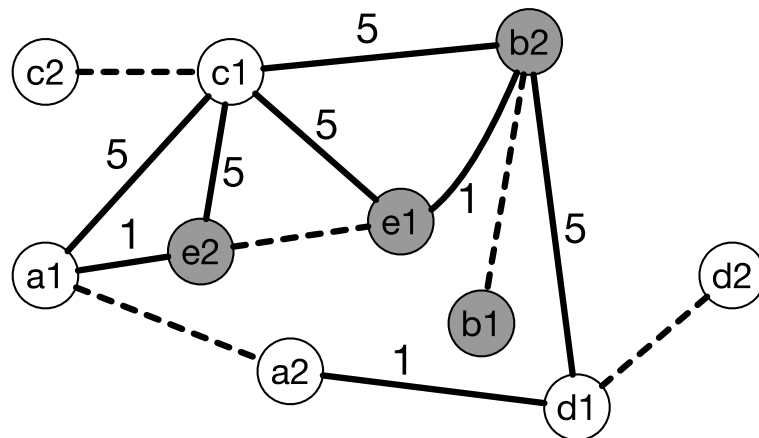
Pre-route Net Planning - 3

◆ Conflict graph constraints

- ✓ At most 1 corner of each net
- ✓ Corners cross the pins of other nets

◆ Conflict graph bipartization

- ✓ Pre-determine (estimate) the routing paths for nets
- ✓ Minimize deleted vertices from conflict graph
 - DSA unfriendly vertices



$$\min \sum_{i=1}^N (\alpha_i \times c_i)$$

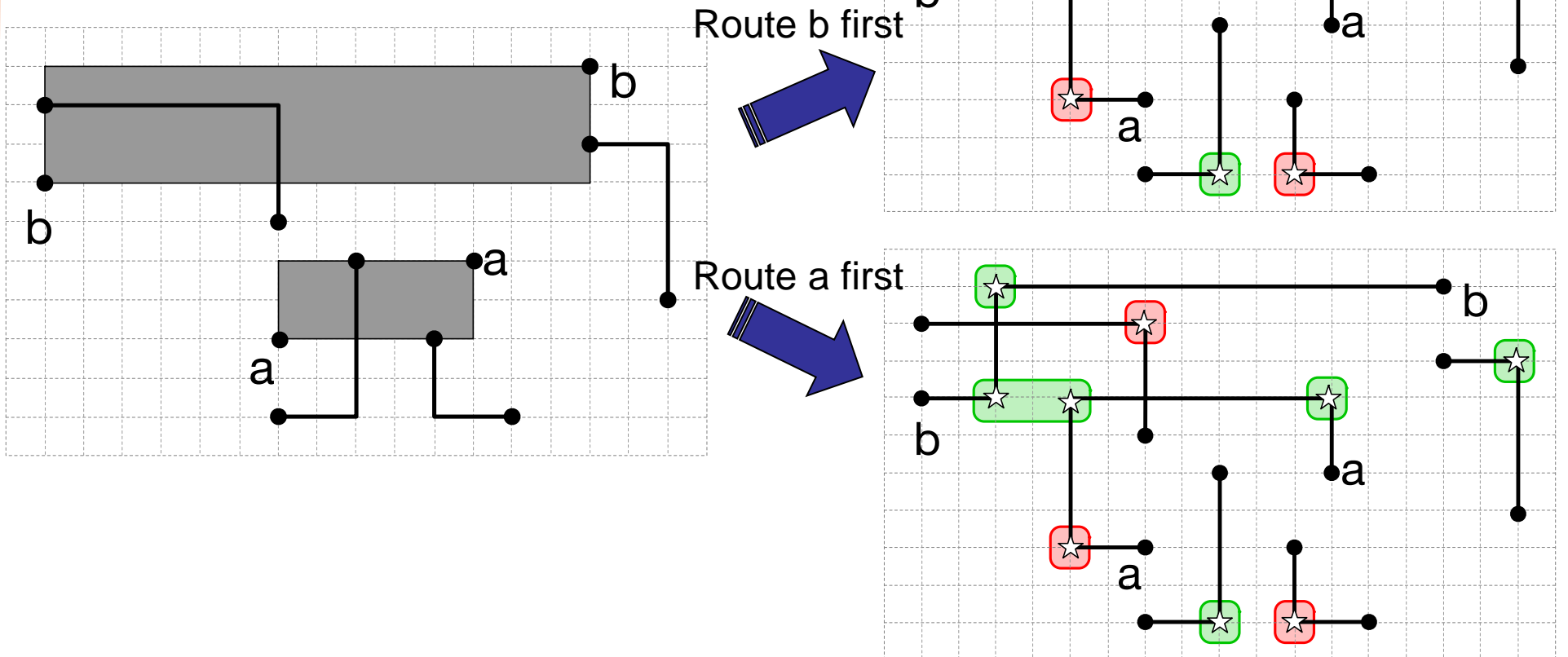
$$\alpha_v = \frac{1}{\sum_{e \in V} Cost_e}$$

$$\begin{aligned} \mathbf{s.t.} \quad & s_v + s_u + (c_v + c_u) \geq 1, & \forall \{v, u\} \in E_s, \\ & s_v + s_u - (c_v + c_u) \leq 1, & \forall \{v, u\} \in E_s, \\ & c_v + c_u \geq 1, & \forall \{v, u\} \in E_d, \\ & c_v = 1, & \forall v \in V_f. \end{aligned}$$

Pre-route Net Planning - 4

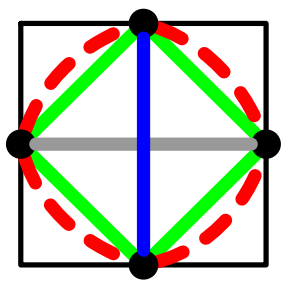
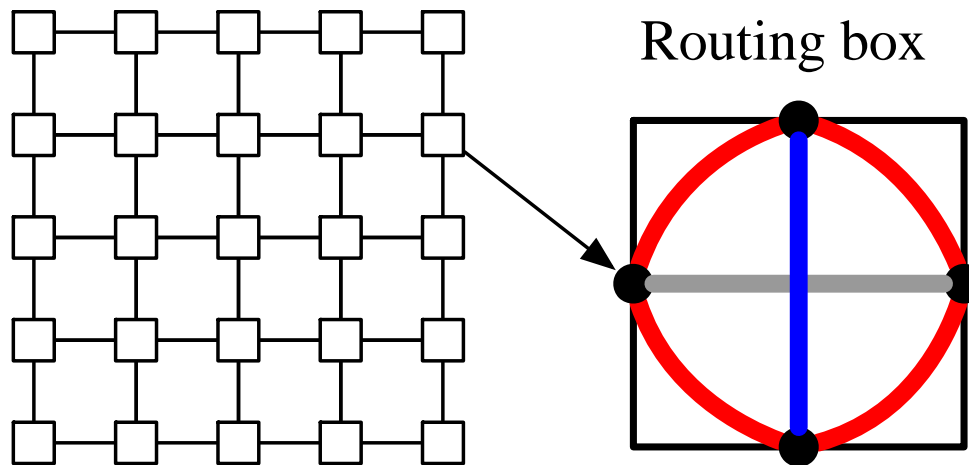
◆ Net ordering for undetermined nets

- ✓ Smaller bbox (HPWL)
- ✓ Overlaps

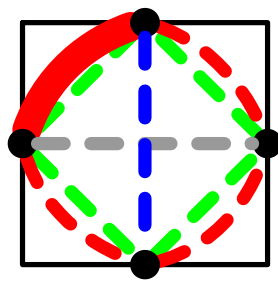
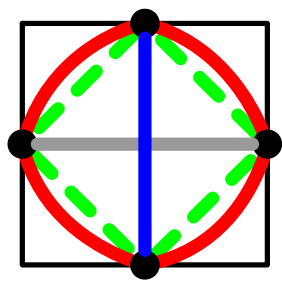


Detailed Routing - 1

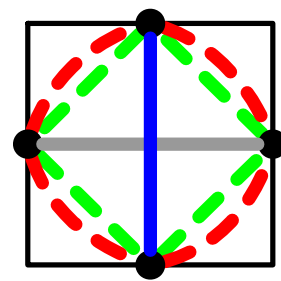
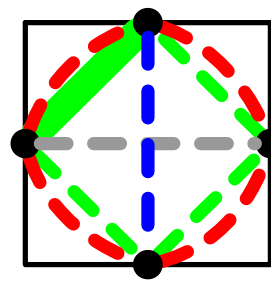
◆ Routing graph model



One color assignment forbidden



Via inserted

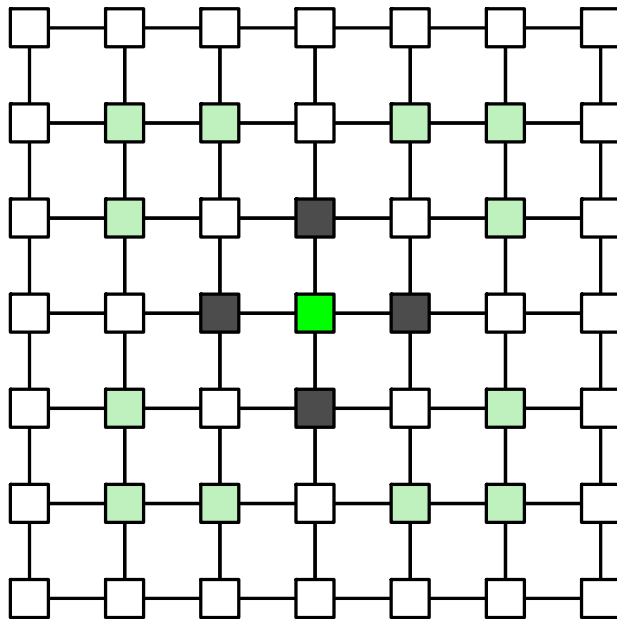


All vias are forbidden₂₅

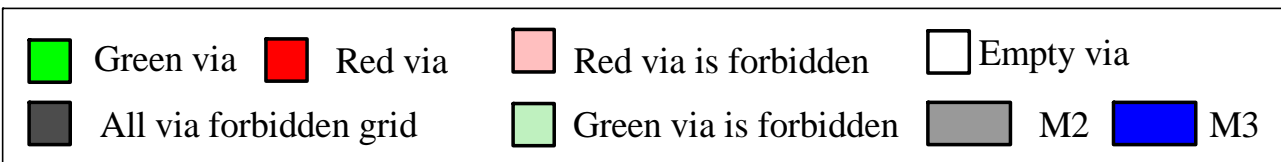
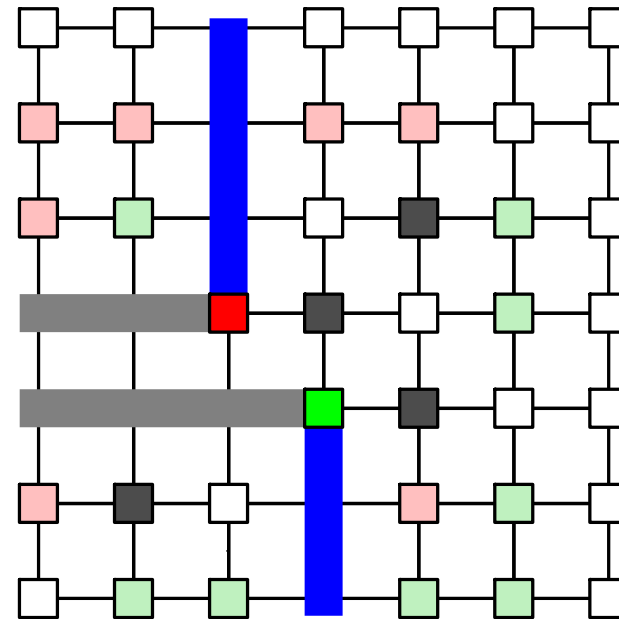
Detailed Routing - 2

◆ Routing box state update

Nearby routing box update



Example



Detailed Routing - 3

◆ Routing scheme

- ✓ Negotiated congestion based
- ✓ A* search

$$p_i(s, t) = c(i) + \sigma \times dist_i^t$$

$$c(i) = cost_s^{i-1} + c_{i-1}^i + h(i)$$

$$h(i) = h(i)' + A \times usage(i) + B \times h_{dsa}(i)$$

$$\sigma = \begin{cases} 1, & c(i) \leq l_{HPWL}, \\ 1 + \frac{c(i)}{HPWL}, & c(i) > l_{HPWL}. \end{cases}$$

Algorithm 1 DSA+DP aware detailed routing

Input: Netlists from net planning algorithm.

Output: Routed nets with DSA friendly via layer.

```
1: Route determined nets;
2: Update grids cost;
3: Initial routing iteration;
4: Q ← nets in violated grids;
5: while !Q.empty() do
6:    $g(i) \leftarrow Q.pop()$ ;  $Nets \in g(i)$ ;
7:   for each net  $k \in Nets$  do
8:     Pre-route with cost evaluation;
9:   end for
10:  Rip-up net  $k$  that has maximum cost improvement;
11:  Route net  $k$ ;
12:  for each grid  $g(j)$  of net  $k$  do
13:    Update grid cost;
14:    if  $g(j)$  is violated then
15:      Q ← nets in this  $g(j)$ ;
16:    end if
17:  end for
18:  if  $g(i)$  is still violated then
19:    Q ←  $g(i)$ ;
20:  end if
21: end while
```

Detailed Routing - 3

◆ Routing scheme

- ✓ Negotiated congestion based
- ✓ A* search

$$p_i(s, t) = c(i) + \sigma \times dist_i^t$$

$$c(i) = cost_s^{i-1} + c_{i-1}^i + h(i)$$

$$h(i) = h(i)' + A \times usage(i) + B \times h_{dsa}(i)$$

$$\sigma = \begin{cases} 1, & c(i) \leq l_{HPWL}, \\ 1 + \frac{c(i)}{HPWL}, & c(i) > l_{HPWL}. \end{cases}$$

Algorithm 1 DSA+DP aware detailed routing

Input: Netlists from net planning algorithm.

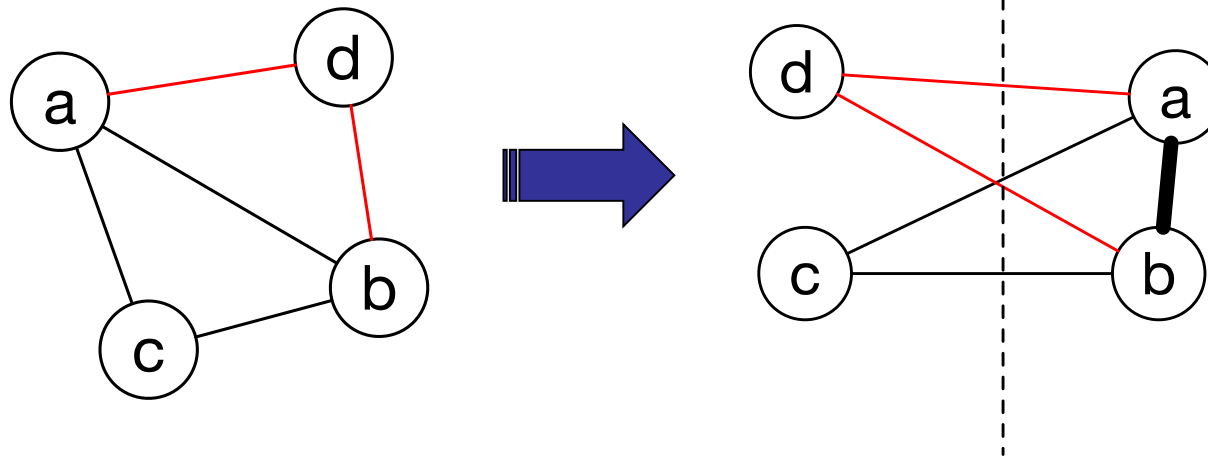
Output: Routed nets with DSA friendly via layer.

```
1: Route determined nets;
2: Update grids cost;
3: Initial routing iteration;
4: Q ← nets in violated grids;
5: while !Q.empty() do
6:    $g(i) \leftarrow Q.pop()$ ;  $Nets \in g(i)$ ;
7:   for each net  $k \in Nets$  do
8:     Pre-route with cost evaluation;
9:   end for
10:  Rip-up net  $k$  that has maximum cost improvement;
11:  Route net  $k$ ;
12:  for each grid  $g(j)$  of net  $k$  do
13:    Update grid cost;
14:    if  $g(j)$  is violated then
15:      Q ← nets in this  $g(j)$ ;
16:    end if
17:  end for
18:  if  $g(i)$  is still violated then
19:    Q ←  $g(i)$ ;
20:  end if
21: end while
```

Post Routing Optimization

◆ Assign DSA guiding patterns

- ✓ Minimize DSA groups and conflicts
- ✓ Edge bipartization



Post Routing Optimization

◆ Assign DSA guiding patterns

- ✓ Minimize DSA groups and conflicts
- ✓ Edge bipartization

$$\mathbf{min} \quad \sum e_{v_i v_j} + M \cdot \sum e_{v_k v_h} \quad (7a)$$

$$\mathbf{s.t.} \quad t_{v_i} + t_{v_j} + e_{v_i v_j} \geq 1, \quad \forall \{e_{v_i v_j}\} \in E, \quad (7b)$$

$$t_{v_i} + t_{v_j} - e_{v_i v_j} \leq 1, \quad \forall \{e_{v_i v_j}\} \in E, \quad (7c)$$

$$e_{v_i v_j} + e_{v_i v_k} \leq 1, \quad \forall (i, j, k) \text{ infeasible}, \quad (7d)$$

$$e_{v_i v_j} + e_{v_j v_k} + e_{v_k v_h} \leq 2, \quad \forall (i, j, k), (j, k, h) \text{ groupable}, \quad (7e)$$

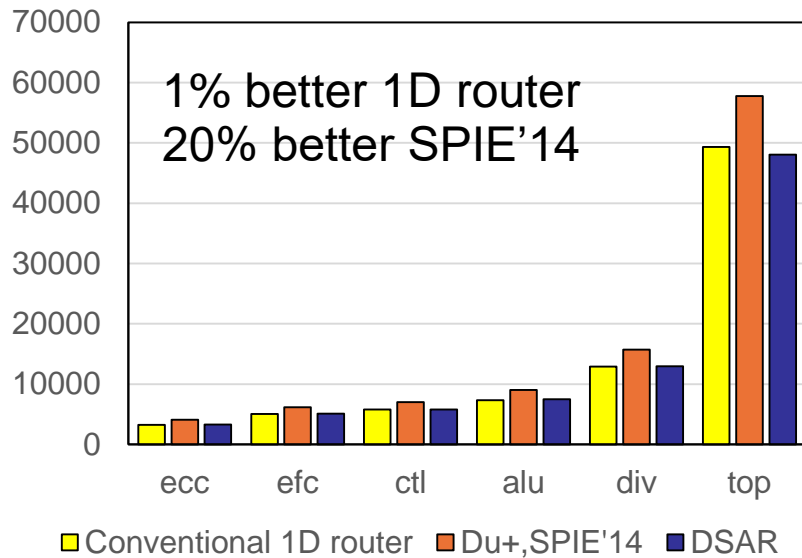
$$e_{v_i v_j}, t_{v_i} \in \{0, 1\}. \quad (7f)$$

Experimental Setup

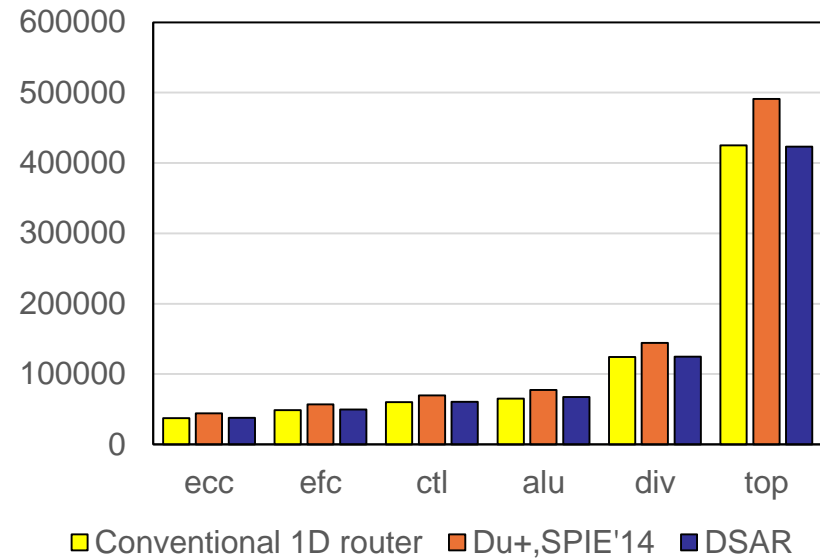
- ◆ Implemented in C++
- ◆ 3.4GHz Linux server, 32GB RAM
- ◆ ILP solver: GUROBI 6.5
- ◆ OpenSparc T1 design:
 - ✓ M2, M3 for routing
 - ✓ [Du+, SPIE'14], 1D router

bench	#net	#pin	Grid size
ecc	1671	3342	436×446
efc	2219	4438	406×421
ctl	2706	5412	496×503
alu	3108	6216	406×408
div	5813	11626	636×646
top	22201	44402	1176×1179

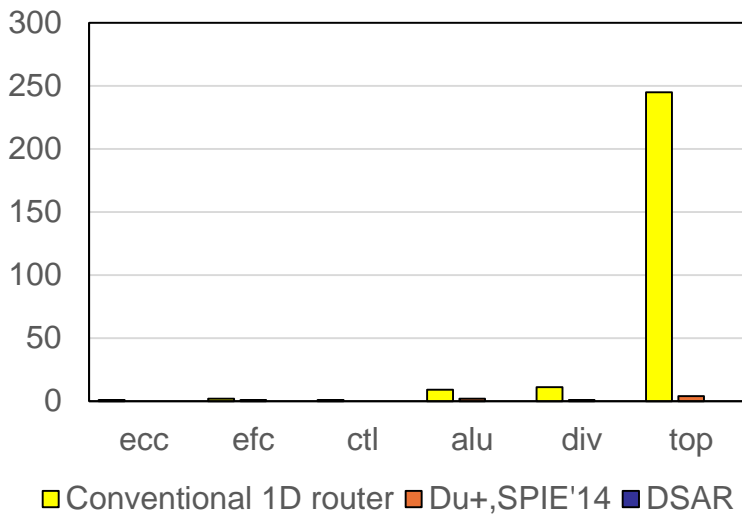
Routing Result Comparison



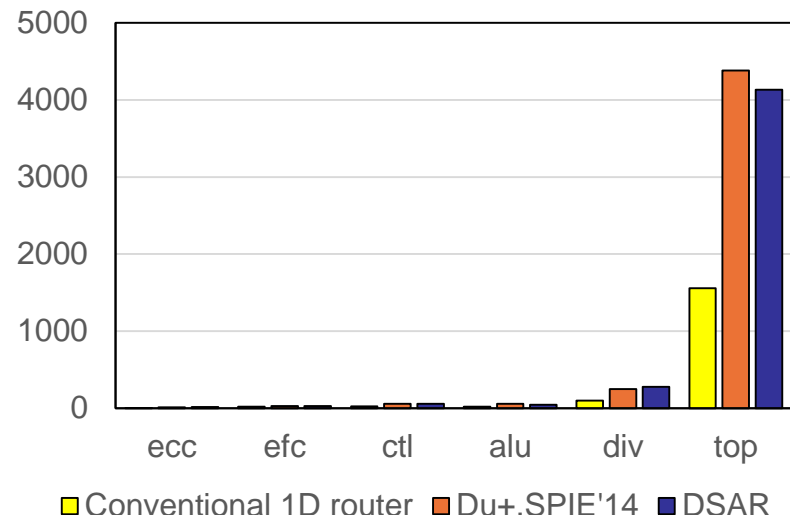
(a) Number of Vias



(b) Wirelength

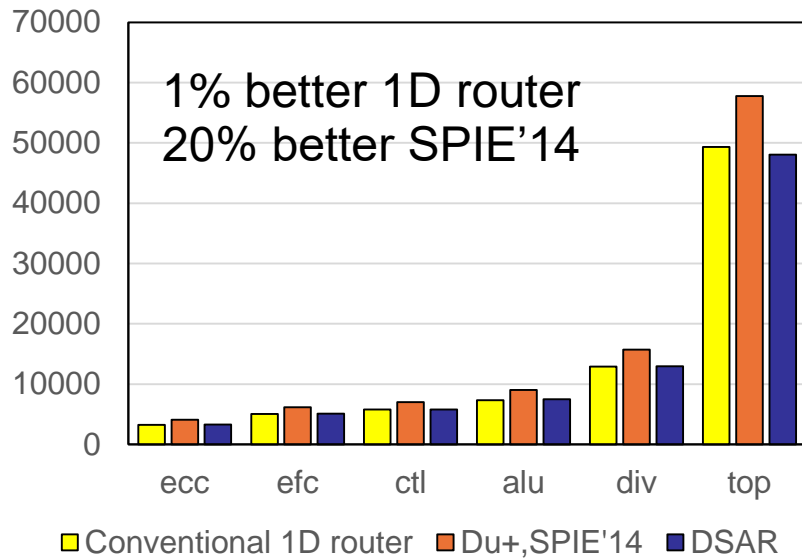


(c) DSA conflicts

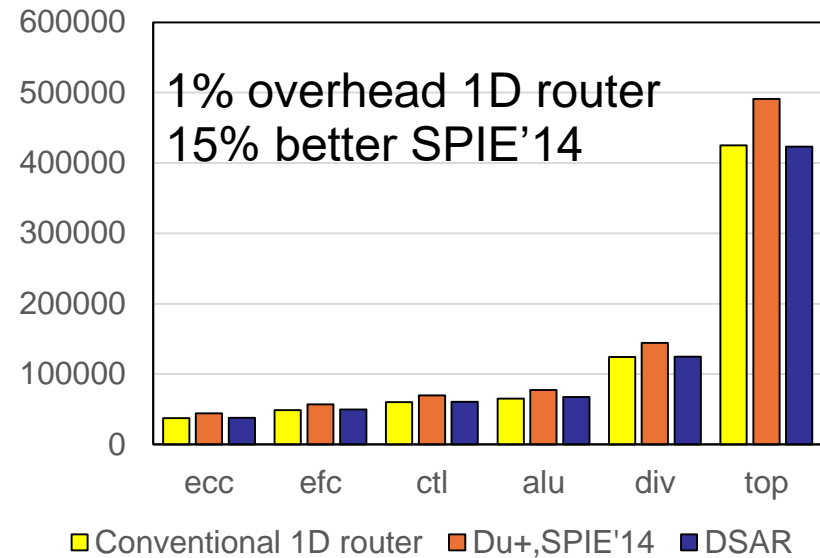


(d) CPU(s)

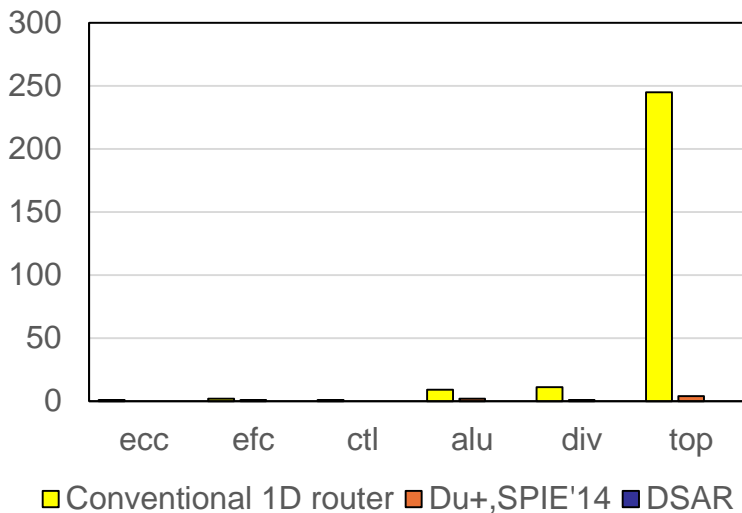
Routing Result Comparison



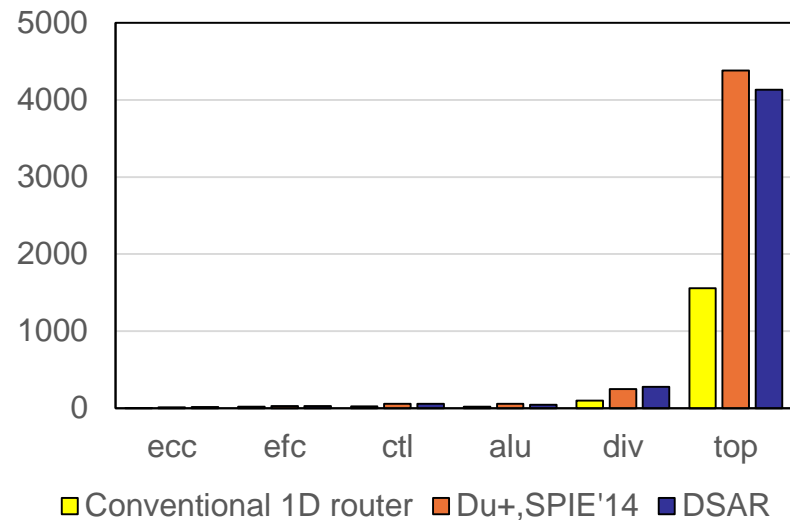
(a) Number of Vias



(b) Wirelength

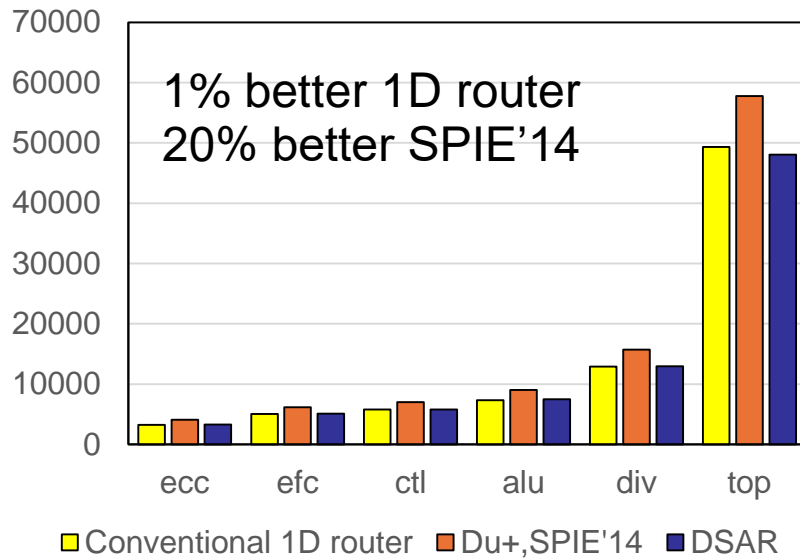


(c) DSA conflicts

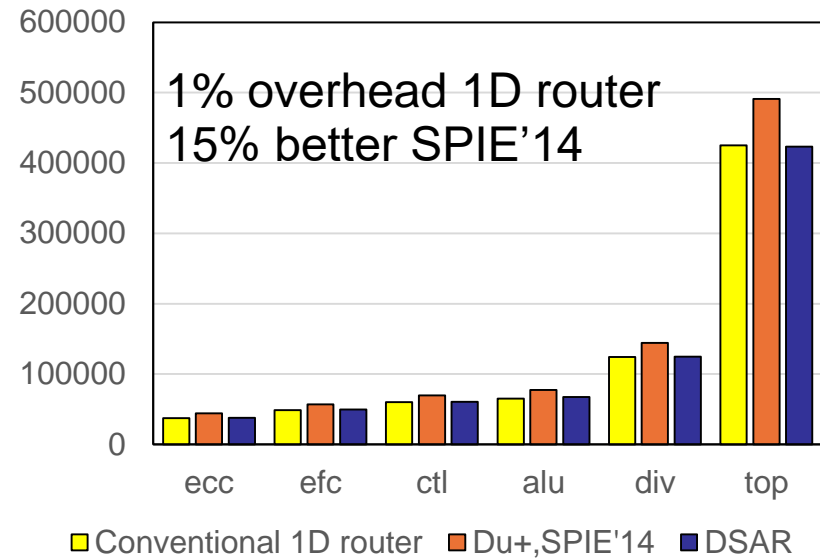


(d) CPU(s)

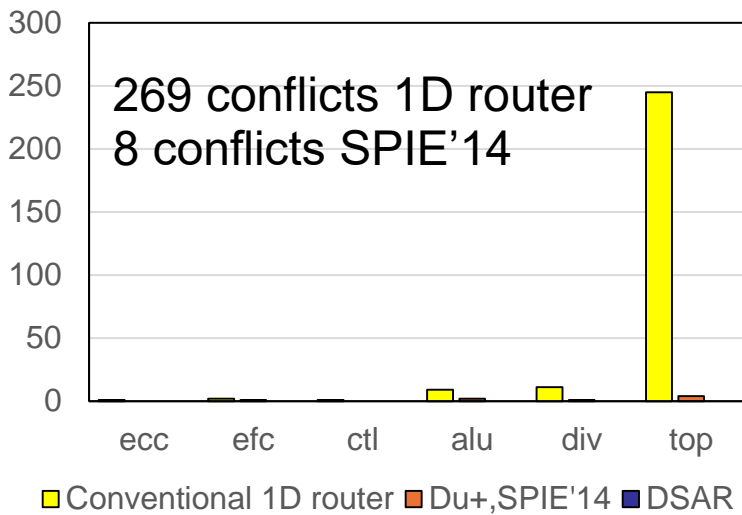
Routing Result Comparison



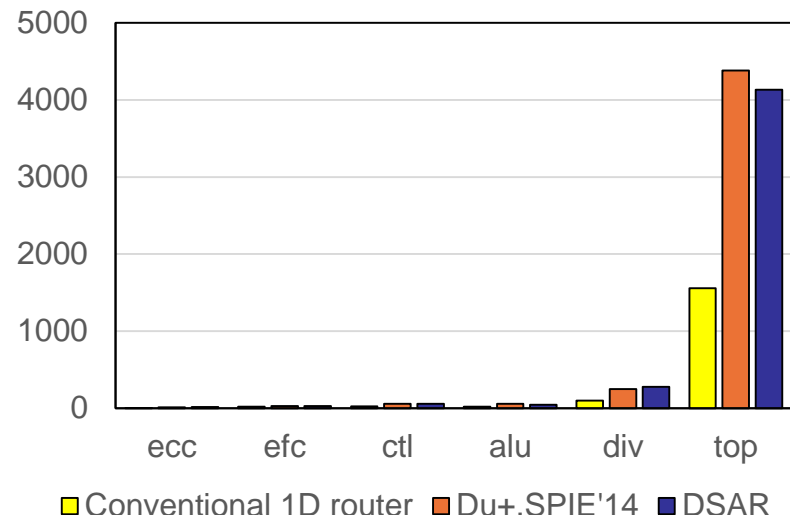
(a) Number of Vias



(b) Wirelength

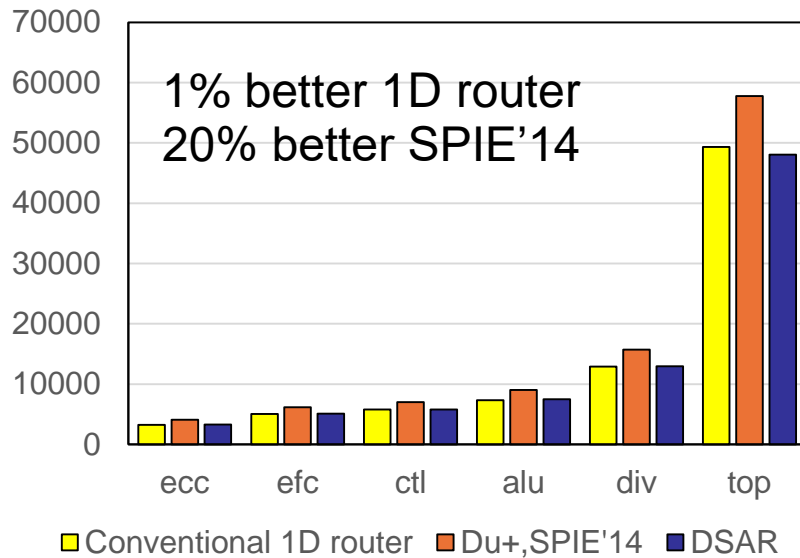


(c) DSA conflicts

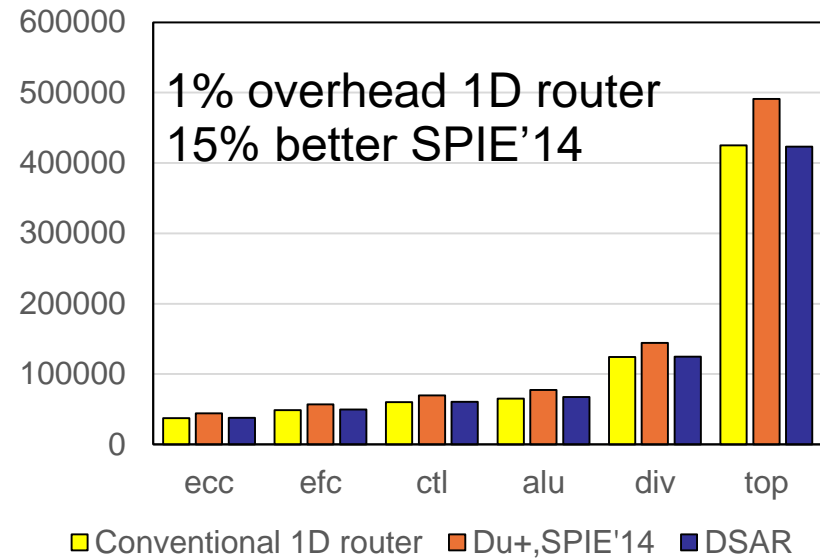


(d) CPU(s)

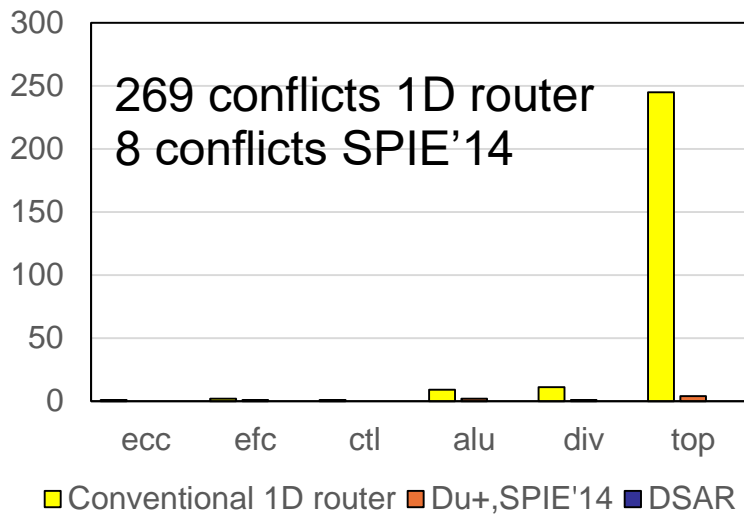
Routing Result Comparison



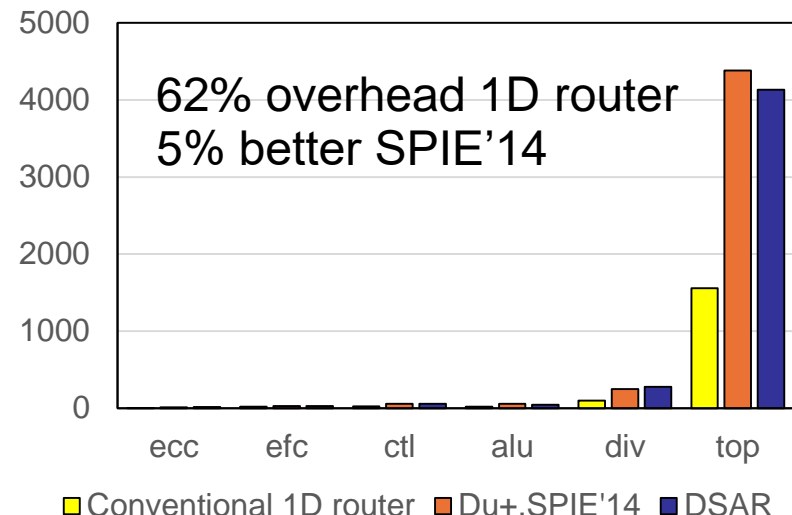
(a) Number of Vias



(b) Wirelength

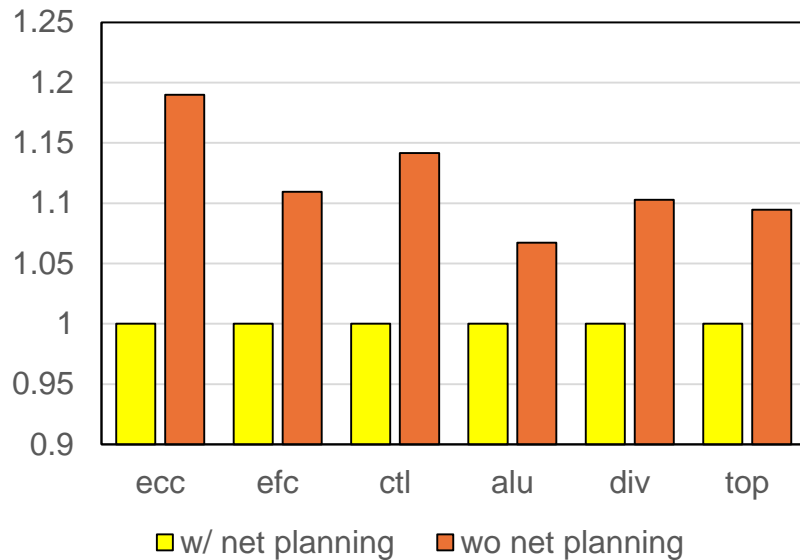


(c) DSA conflicts

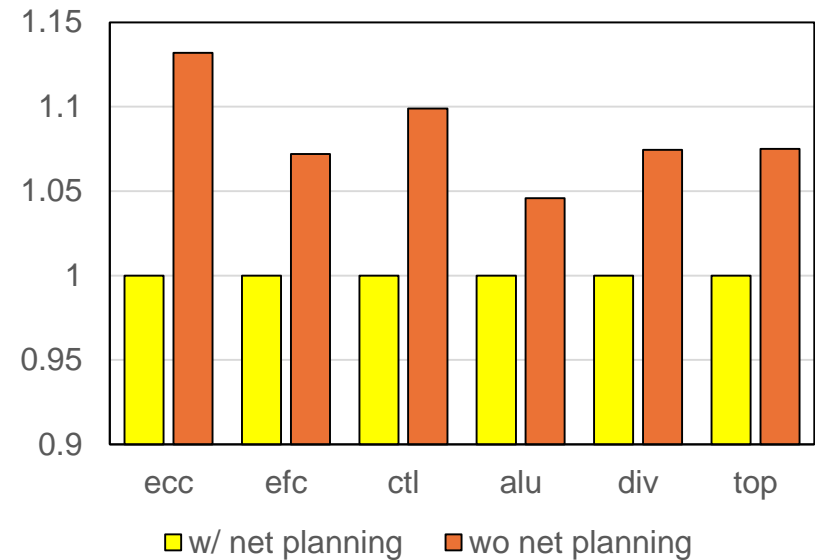


(d) CPU(s)

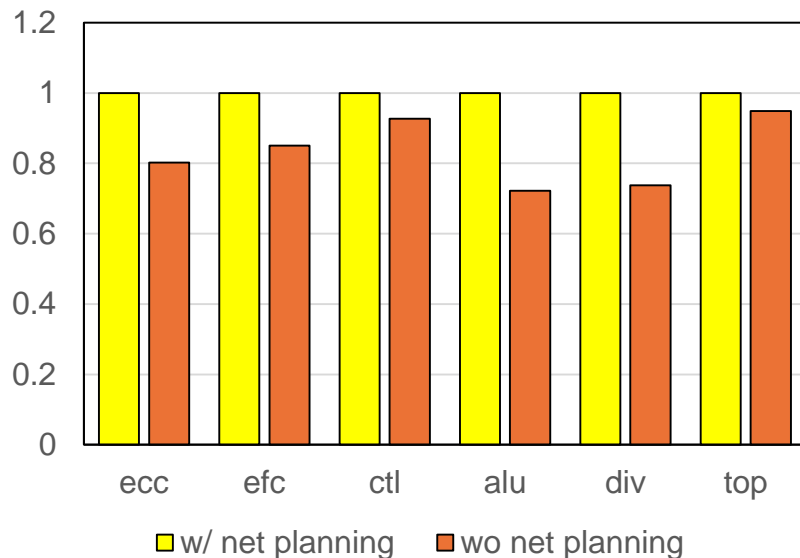
Comparison between W/ and WO Net Planning



(a) Number of Vias



(b) Wirelength



(c) CPU(s)

With v.s. without net planning

- ▷ **19% less via number**
- ▷ **8% less wirelength**
- ▷ **7% more runtime**

Conclusion

- ◆ DSA and double patterning for via layer in detailed routing
 - ✓ Pre-route net planning
 - ✓ Routing model with DSA-DP consideration
 - ✓ Post-routing optimization to improve DSA guiding pattern assignment and decomposition
- ◆ Future work
 - ✓ Adaptive to more routing layers
 - ✓ General to more DSA and multiple patterning considerations



Q&A

THANK YOU