Post-Routing Redundant Via Insertion for Yield/Reliability Improvement

Kuang-Yao Lee and Ting-Chi Wang

Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan

Outline

- Redundant via and Double via
- Post-routing redundant via insertion
  - Maximum bipartite matching
  - Maximum independent set (MIS)
- Our MIS-based approach
- Extension to the consideration of on-/off-track redundant via
- Experimental results
- Conclusions

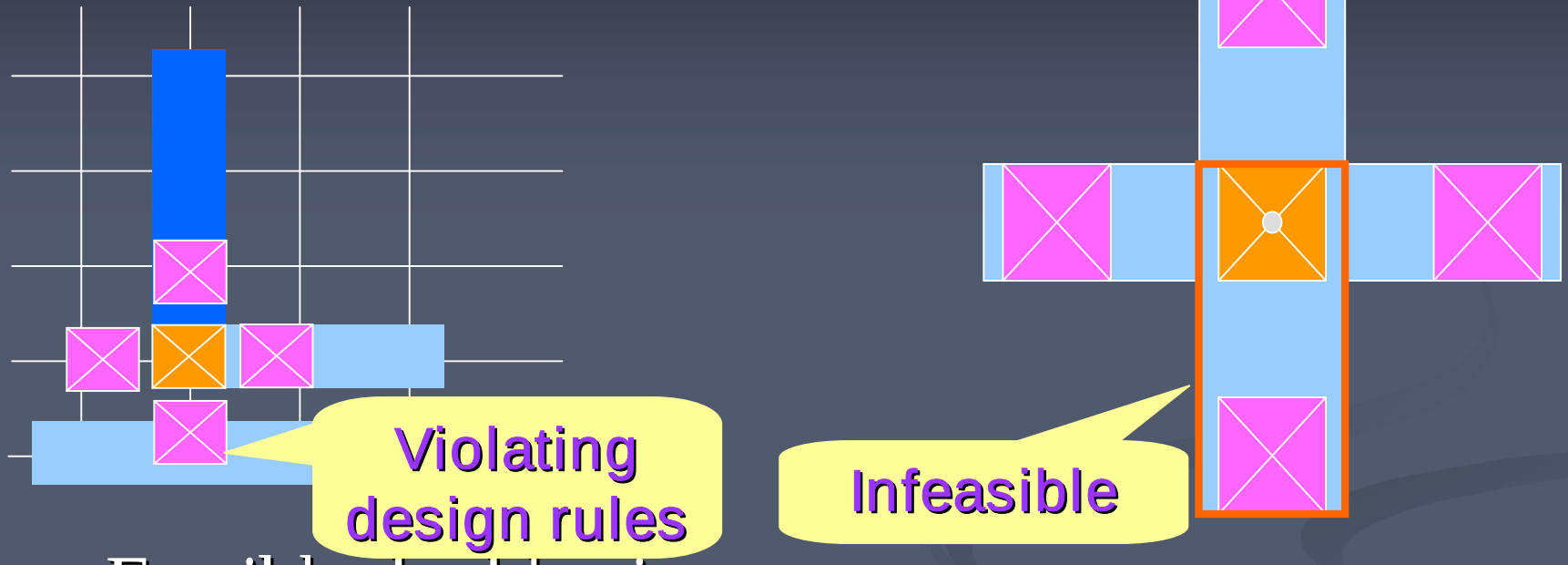
Outline

- Redundant via and Double via
- Post-routing redundant via insertion
  - Maximum bipartite matching
  - Maximum independent set (MIS)
- Our MIS-based approach
- Extension to the consideration of on-/off-track redundant via
- Experimental results
- Conclusions

Redundant Via

- Yield loss caused by via failure becomes critical and requires a good control
- A good solution is to add a redundant via adjacent to a single normal via as a backup
  - The extra via enables a single via failure to be tolerated
- Redundant via insertion can be considered in the routing stage or post-routing stage
  - We only consider post-routing redundant via insertion

Double Via



- Feasible double via
  - A double via is said to be feasible if it will not violate any design rule, assuming none of the other single vias has a redundant single via inserted in the design

Outline

- Redundant via and Double via
- Post-routing redundant via insertion
  - Maximum bipartite matching
  - Maximum independent set (MIS)
- Our MIS-based approach
- Extension to the consideration of on-/off-track redundant via
- Experimental results
- Conclusions

Problem Formulation

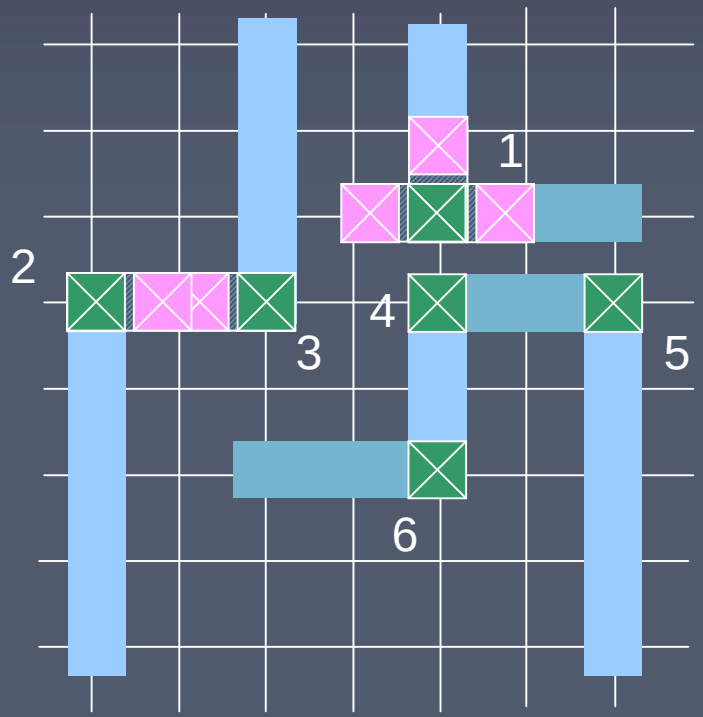
- Input
  - A routed design and a set of design rules
- Goal
  - To replace single vias on signal nets with double vias as many as possible
- Constraints
  - Do not re-route any signal net
  - Each single via either remains unchanged or is replaced by a double via
  - After double via replacement, no design rule is violated.

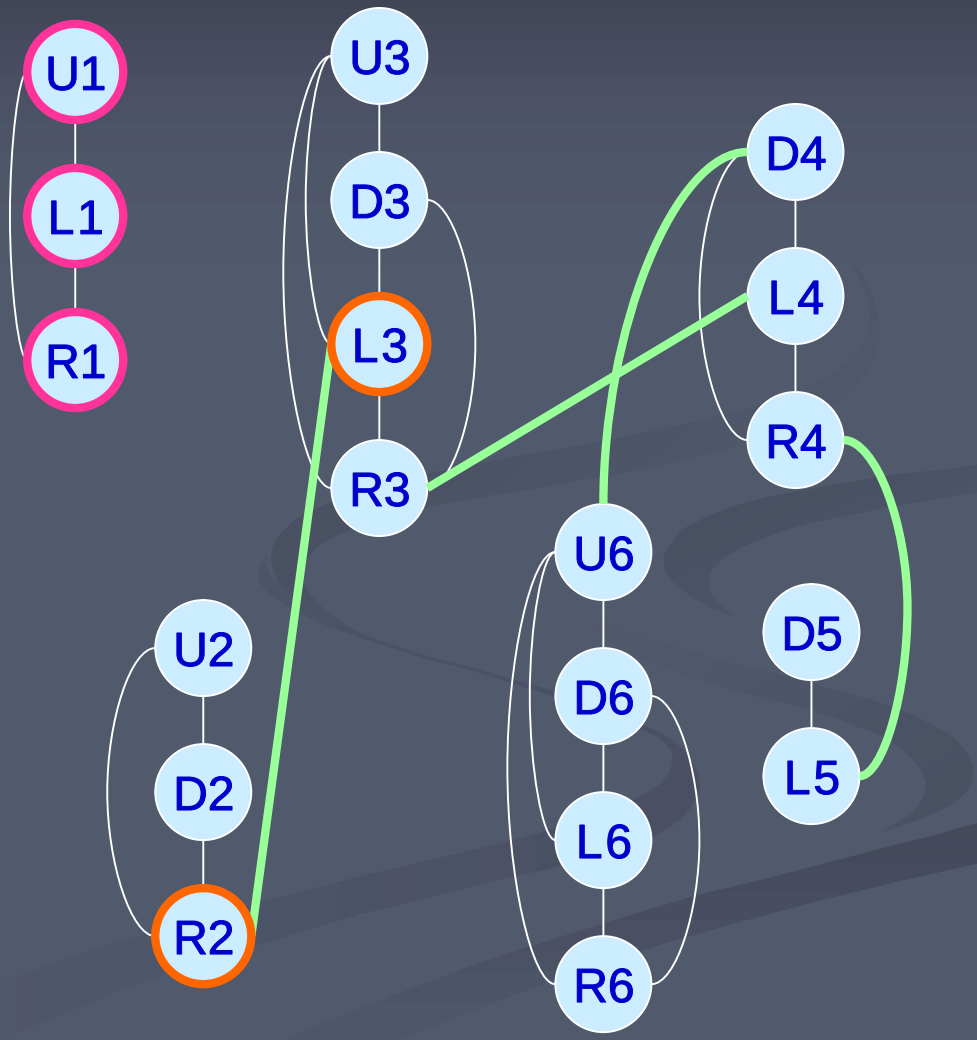
Maximum Bipartite Matching Formulation

- We showed that the maximum bipartite matching formation may not formulate the post-routing redundant via insertion problem correctly

H. Yao, Y. Cai, X. Hong and Q. Zhou, “Improved Multilevel Routing with Redundant Via Placement for Yield and Reliability”, Proc. of GLSVLSI, 2005.

MIS Formulation – Conflict Graph





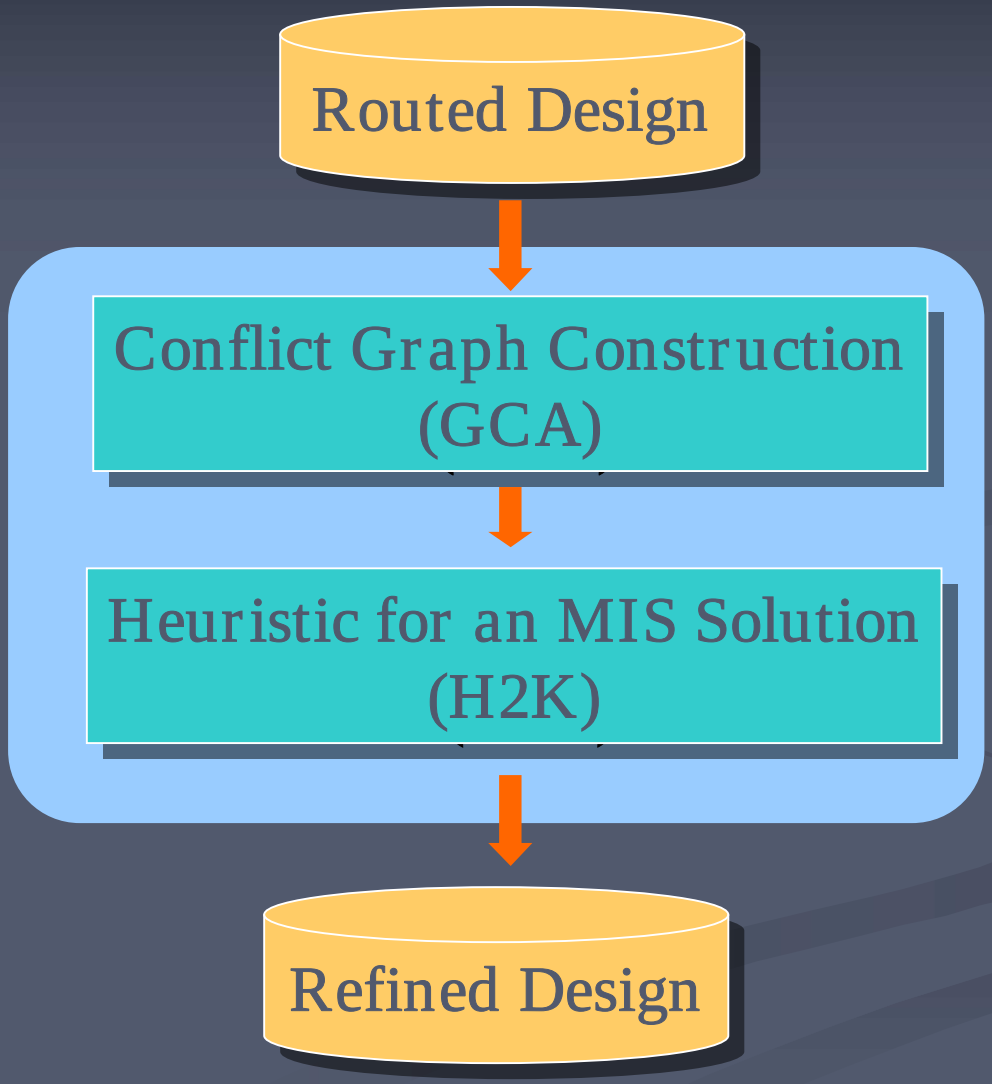
MIS Formulation – Problem Transformation

- The post-routing redundant via insertion problem can be reduced into the **maximum independent set problem**

Outline

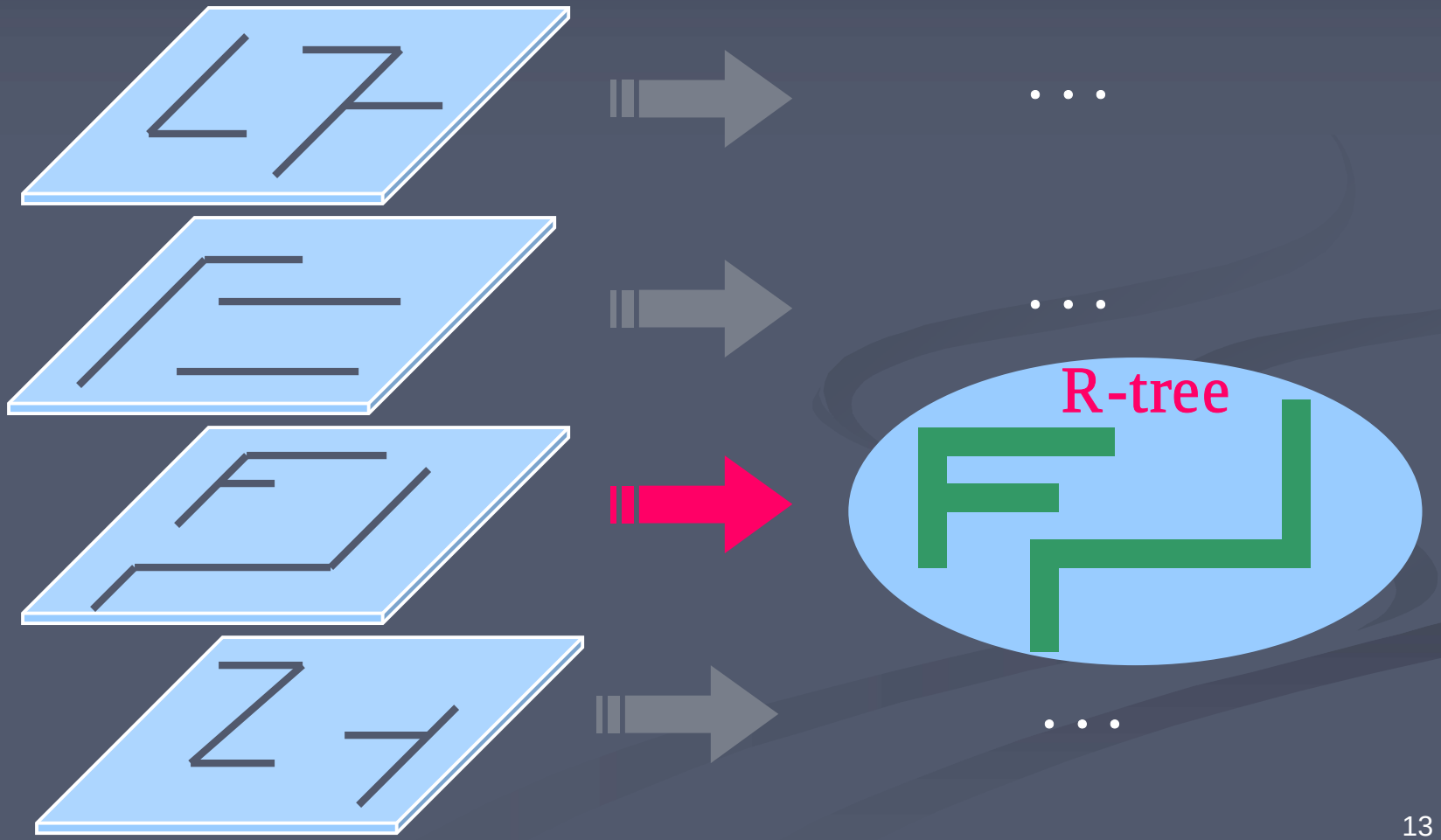
- Redundant via and Double via
- Post-routing redundant via insertion
  - Maximum bipartite matching
  - Maximum independent set (MIS)
- **Our MIS-based approach**
- Extension to the consideration of on-/off-track redundant via
- Experimental results
- Conclusions

Our MIS-based Approach

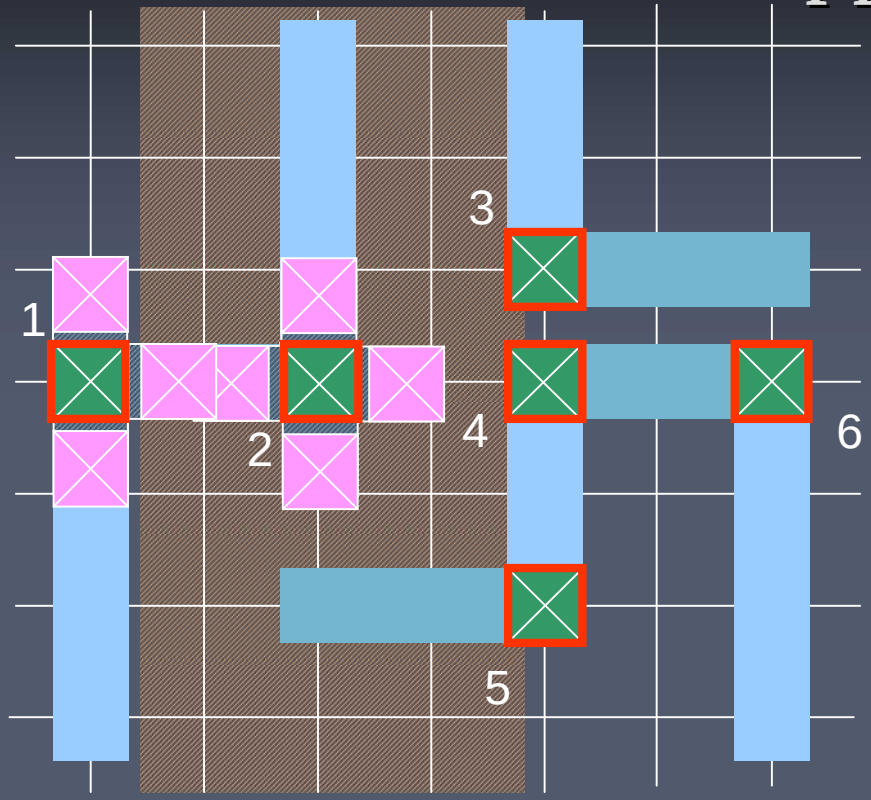


Conflict Graph Construction (cont'd)

- In this work, we use R-trees for indexing the 2D information of layout objects

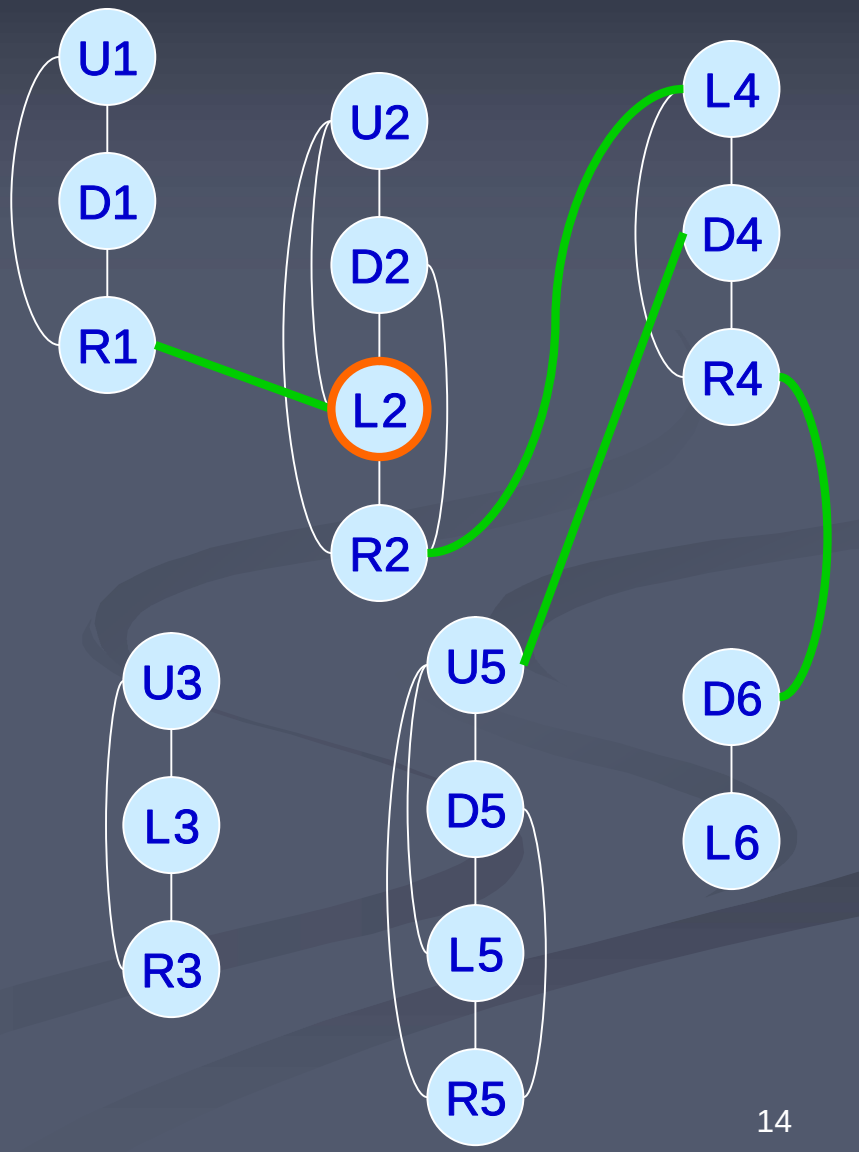


Conflict Graph Construction (cont'd) - Our approach GCA



R-tree

U3	L3	R5	D5
R3	D6	L6	U5
L4	D4	R4	

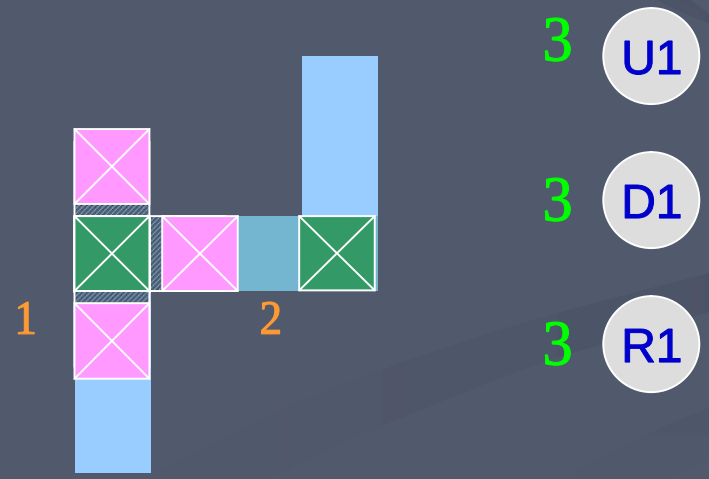


Heuristic for Solving the MIS Problem – Our algorithm H2K

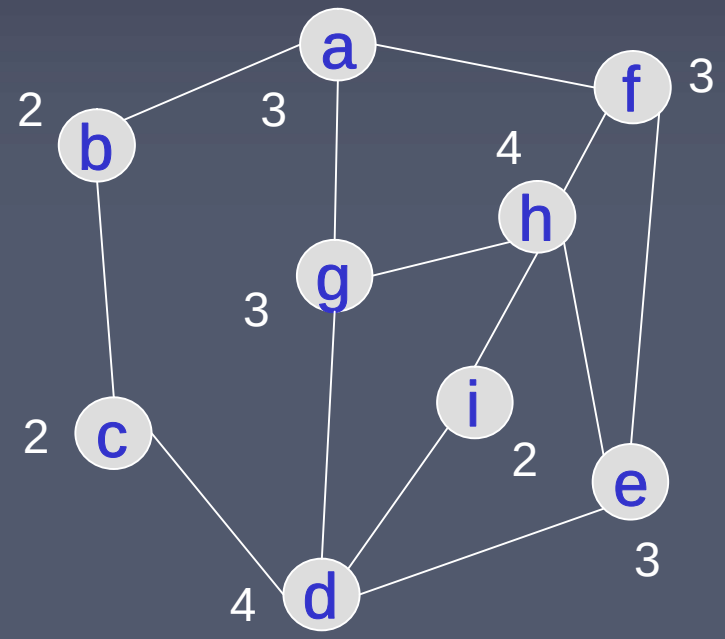
- H2K solves the MIS problem in an iterative manner
- In each iteration, a subgraph is extracted from the conflict graph, a maximal independent set solution to the subgraph is sought and added to the final solution, and the conflict graph is updated

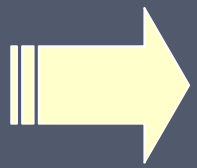
H2K (cont'd)

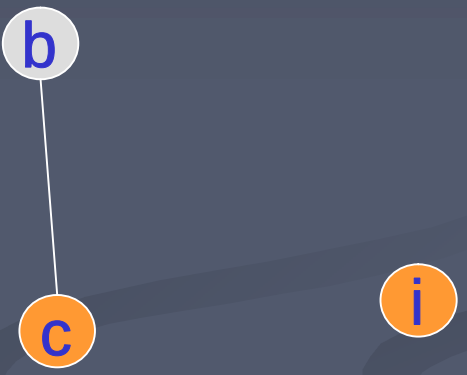
- For the conflict graph G(V ,E), we construct a priority queue Q of V by using the feasible number and degree of a vertex as the first and second keys.
  - We give a vertex a higher priority if it has smaller feasible number and degree.
- Feasible number



H2K (cont'd)







high priority low

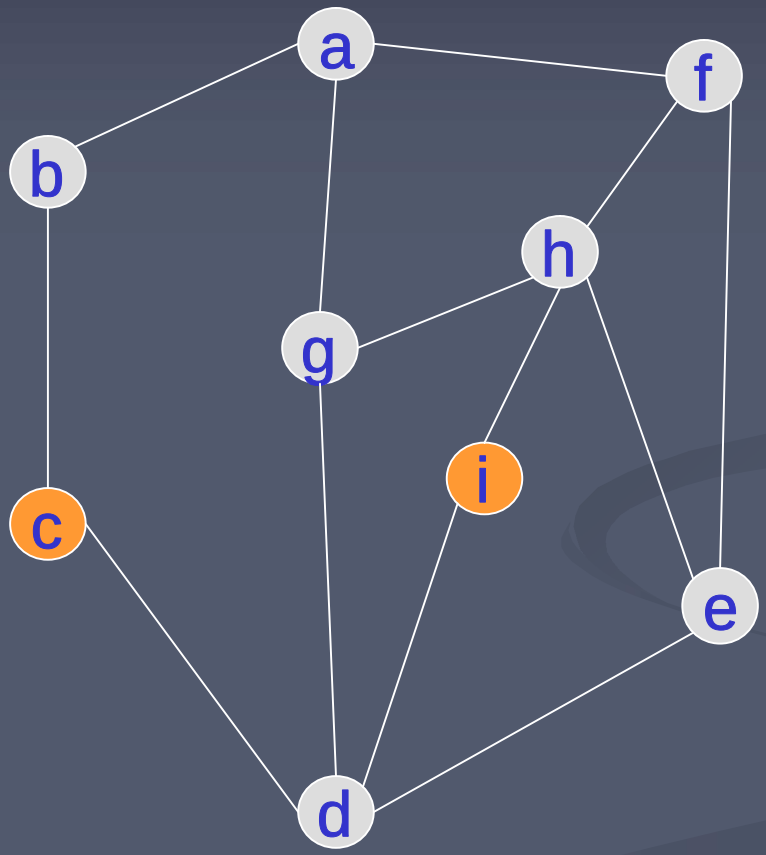
b	c	i	a	e	f	g	d	h

Priority queue

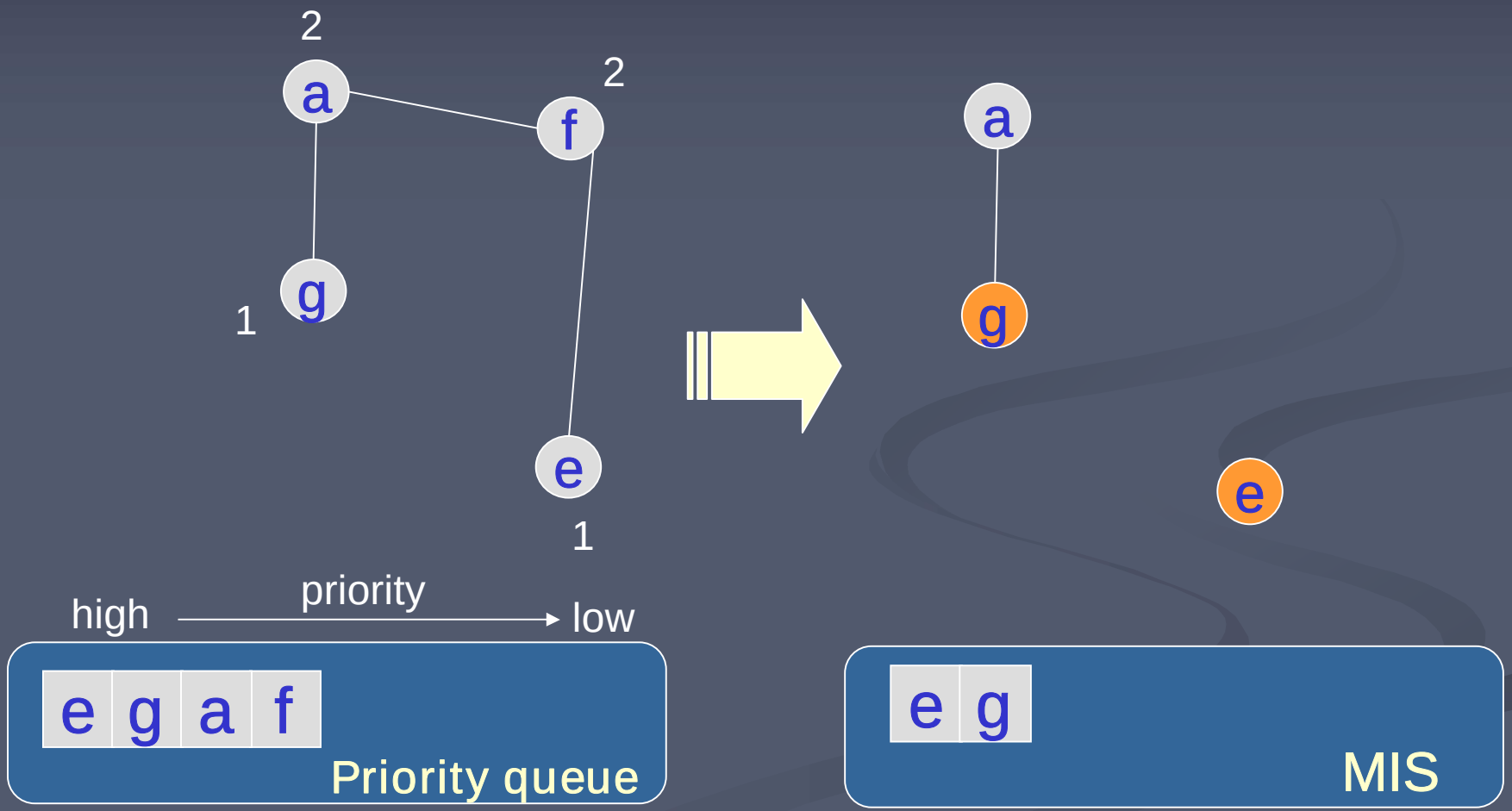
c	i

MIS

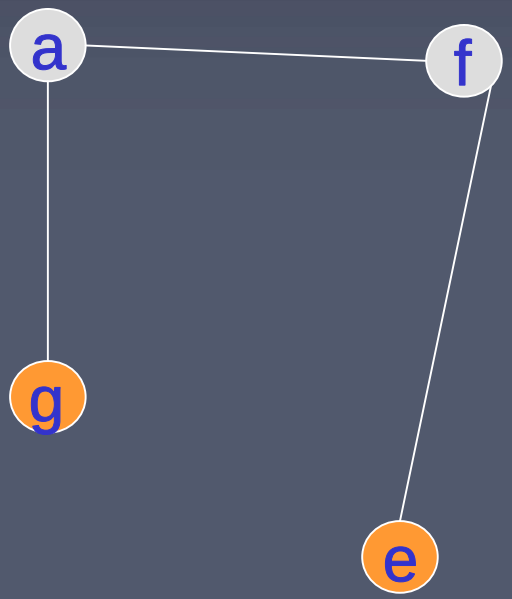
H2K (cont'd)



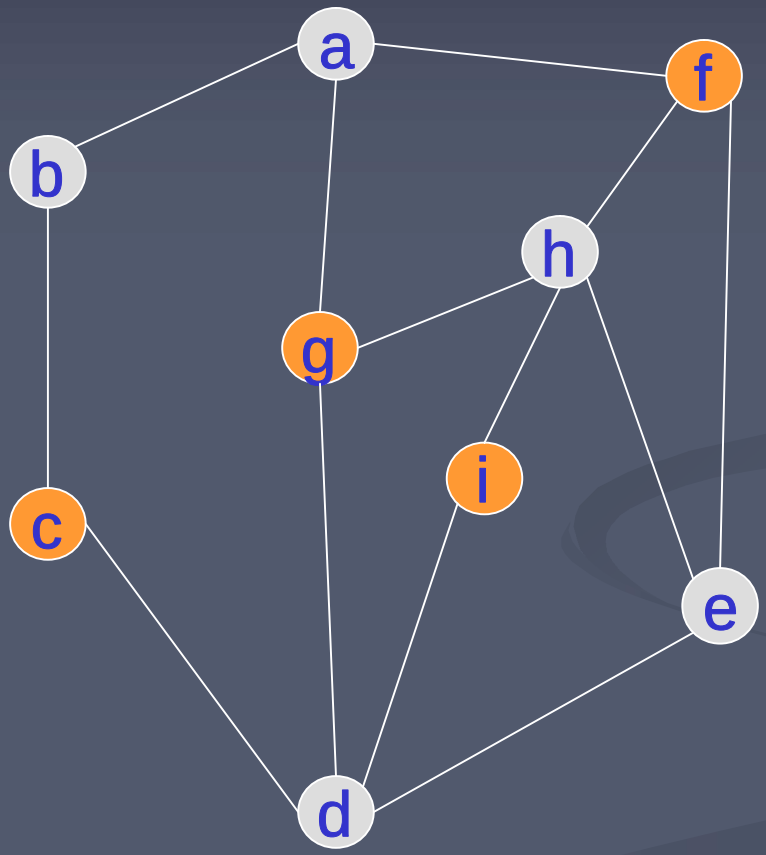
H2K (cont'd)



H2K (cont'd)



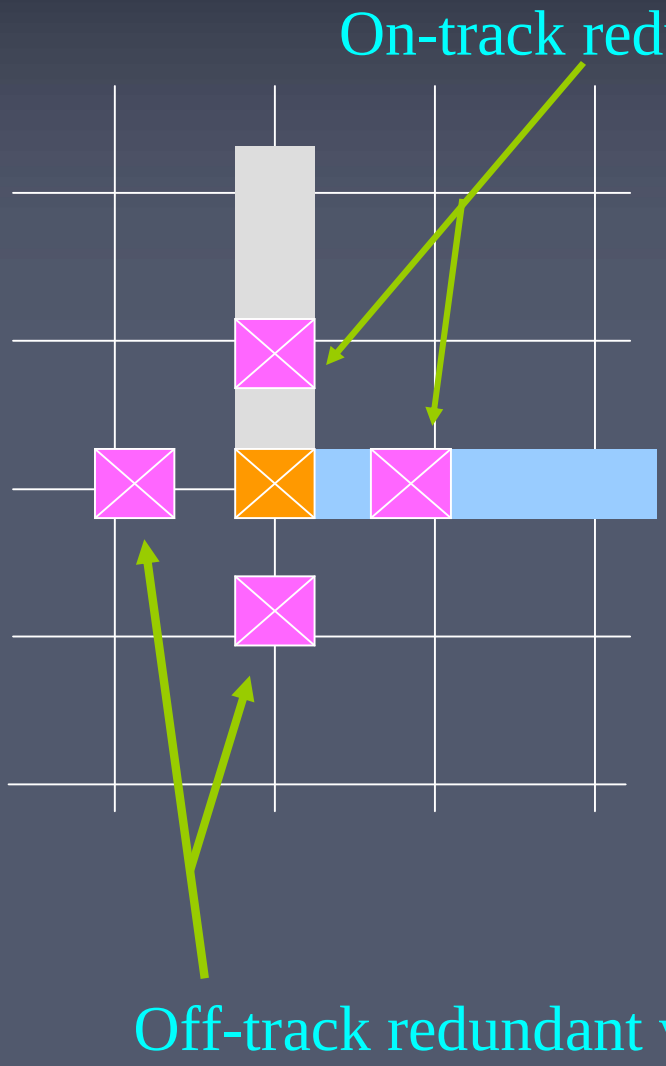
H2K (cont'd)



Outline

- Redundant via and double via
- Post-routing redundant via insertion
  - Maximum bipartite matching
  - Maximum independent set (MIS)
- Our approach
- **Extension to the consideration of on-/off-track redundant via**
- Experimental results
- Conclusions

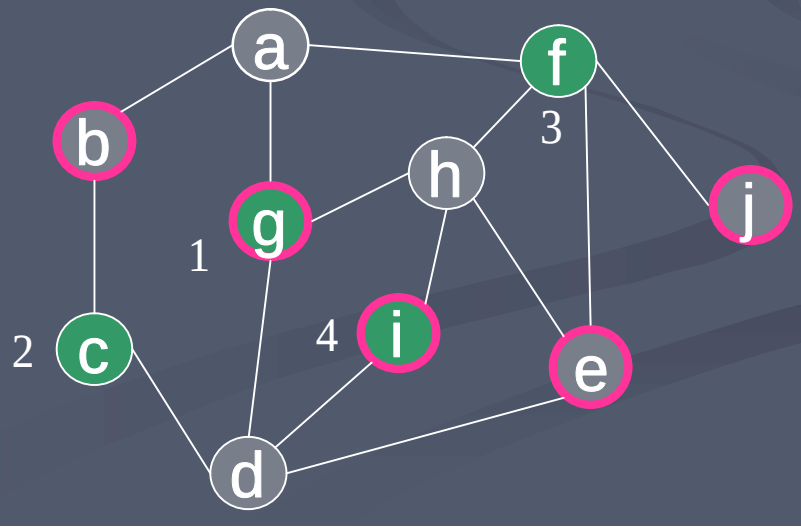
On-/Off-Track Redundant Via



- On-track redundant vias are more preferable since they
  - take less routing resource
  - have better electrical characteristics
- If two solutions contain the same number of redundant vias, we prefer the one with more on-track redundant vias

On-/Off-Track Redundant Via (cont'd)

- H3K
  - We add the third key to each vertex in the priority queue. If a vertex corresponds to an on-track double via, it will have a higher priority on this key.
- PPH (a post processing heuristic)
  - To increase the amount of on-track double vias as many as possible while at the same time without decreasing the total number of double vias.



Outline

- Redundant via and double via
- Post-routing redundant via insertion
  - Maximum bipartite matching
  - Maximum independent set (MIS)
- Our approach
- Extension to the consideration of on-/off-track redundant via
- **Experimental results**
- Conclusions

Experimental Results

Case	Size(μm)	#Nets	#Pins	#Vias	#Layers	#Objects
C1	350.000 *350.000	4309	20	24594	5	218215
C2	419.433 *413.28	5252	211	41157	5	268669
C3	799.124 *776.16	18157	85	127059	5	933852
C4	691.272 *680.400	17692	415	151912	5	934073
C5	1383.482 *1375.92	44720	99	357386	5	2851612

Experimental Results – GCA+H2K

Cases	C1	C2	C3	C4	C5
Original	24594	41157	127059	151912	357386
Upper	17522	28591	91727	102347	255301
Tool	14402	25918	80827	91574	225142
H2K	17461	28507	91461	101765	254428
Imp(%)	21.24	9.99	13.16	11.13	13.01

Experimental Results – H3K and PPH

C1					
	#Total RV		#On-track RV		Imp
	Original	Modified	Original	Modified	
H2K+PPH	17461	17461	7167	8552	19.3%
H3K	17461	-	11848	-	-
H3K+PPH	17461	17461	11848	11878	0.25%

Experimental Results – H3K and PPH (cont'd)

C2					
	#Total RV		#On-track RV		Imp
	Original	Modified	Original	Modified	
H2K+PPH	28507	28507	13406	16047	19.7%
H3K	28506	-	20508	-	-
H3K+PPH	28506	28506	20508	20519	0.05%

Experimental Results – H3K and PPH (cont'd)

C3					
	#Total RV		#On-track RV		Imp
	Original	Modified	Original	Modified	
H2K+PPH	91461	91461	42397	50275	18.6%
H3K	91461	-	66205	-	-
H3K+PPH	91461	91461	66205	66212	0.01%

Experimental Results – H3K and PPH (cont'd)

C4					
	#Total RV		#On-track RV		Imp
	Original	Modified	Original	Modified	
H2K+PPH	101765	101765	48073	57946	20.5%
H3K	101765	-	70696	-	-
H3K+PPH	101765	101765	70696	70696	0%

Experimental Results – H3K and PPH (cont'd)

C5					
	#Total RV		#On-track RV		Imp
	Original	Modified	Original	Modified	
H2K+PPH	254428	254428	118557	142251	19.9%
H3K	254428	-	180512	-	-
H3K+PPH	254428	254428	180512	180513	0.00%

Outline

- Redundant via
- Post-routing redundant via insertion
- Our approach
  - GCA
  - H2K
- Extension to the consideration of on-/off-track redundant via
  - H3K
  - PPH
- Experimental results
- **Conclusions**

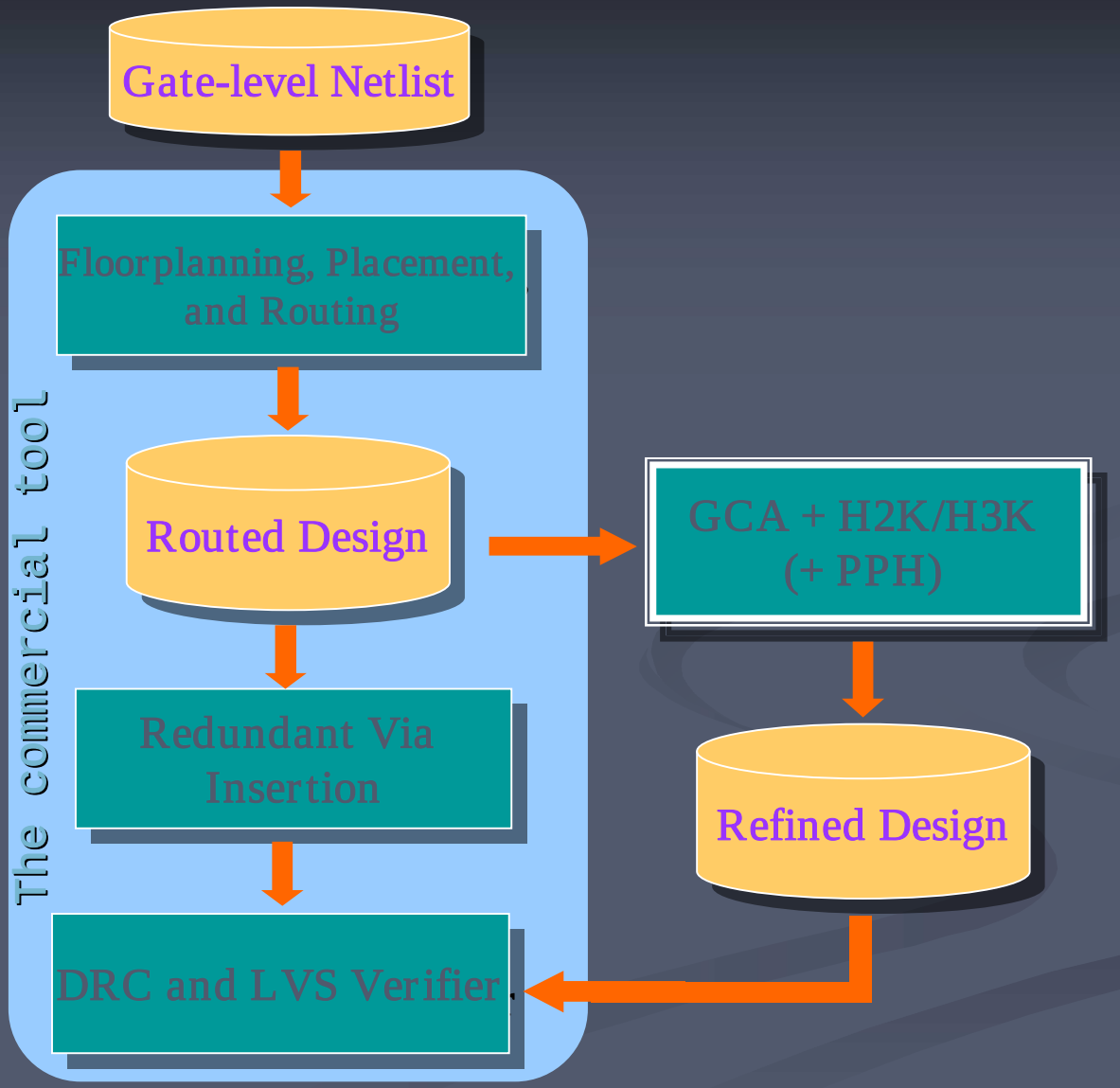
Conclusions

- We study the problem of post-routing redundant via insertion and discuss two possible formulations
  - Maximum bipartite matching (which might not work)
  - Maximum independent set (MIS)
- We present an MIS-based approach to solve the problem
- We also present two approaches to increase the amount of on-track redundant vias
- The experimental results are provided to support all our approaches

Thank You!

Q&A

Experimental Setup



Experimental Setup (cont'd)

- The commercial tool was executed on a Sun Fire V440 machine with 4CPUs and 8GB memory
- All our algorithms were implemented in C++ language running on a Linux based machine with 2.4G processor and 2GB memory

- We directly used the R*-tree package for indexing 2-dimensional information of each metal layer
- The qualex-ms was utilized as our MIS solver
- A subgraph can consist of 1500 vertices at most

The experimental result – CPU times

Testcase	Tool	Our
C1	19	32
C2	28	43
C3	101	192
C4	120	203
C5	311	710