A Separation and Minimum Wire Length Constrained Maze Routing Algorithm under Nanometer Wiring Rules

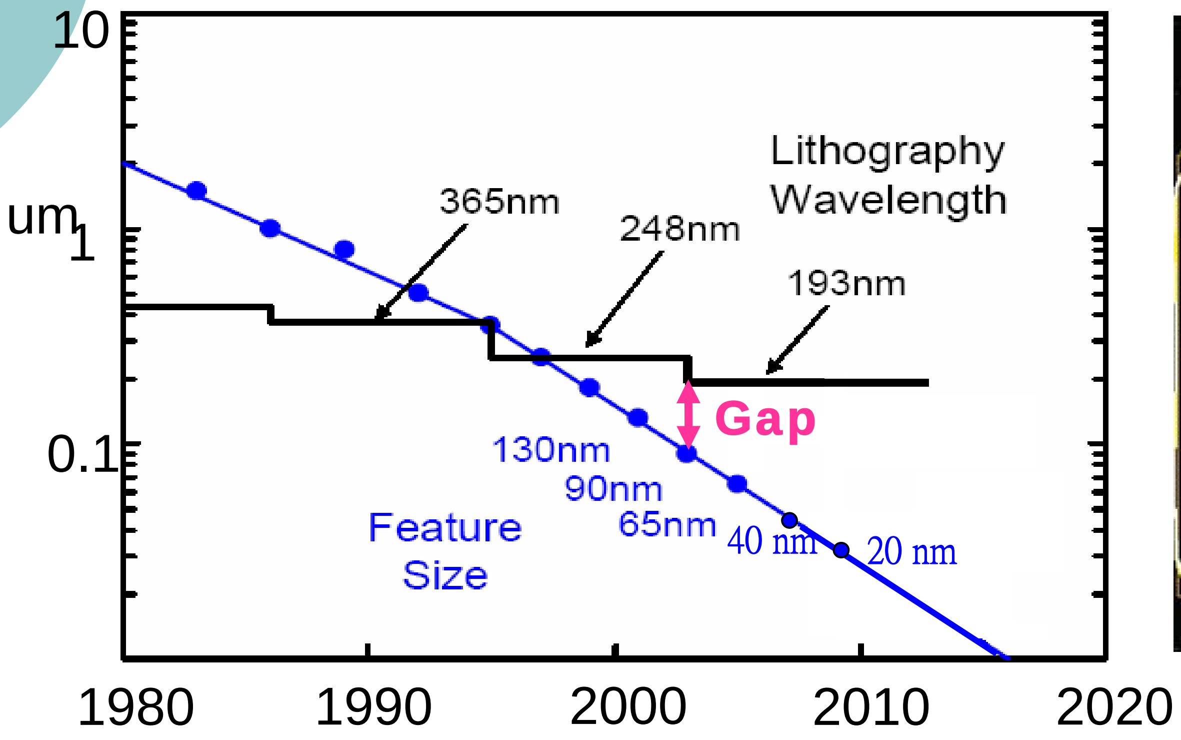
[1]National Tsing-Hua University
[2]SYNOPSYS
Fong-Yuan Chang[1,2], Ren-Song Tsay[1], Wai-Kei Mak[1], Sheng-Hsiung Chen[2]

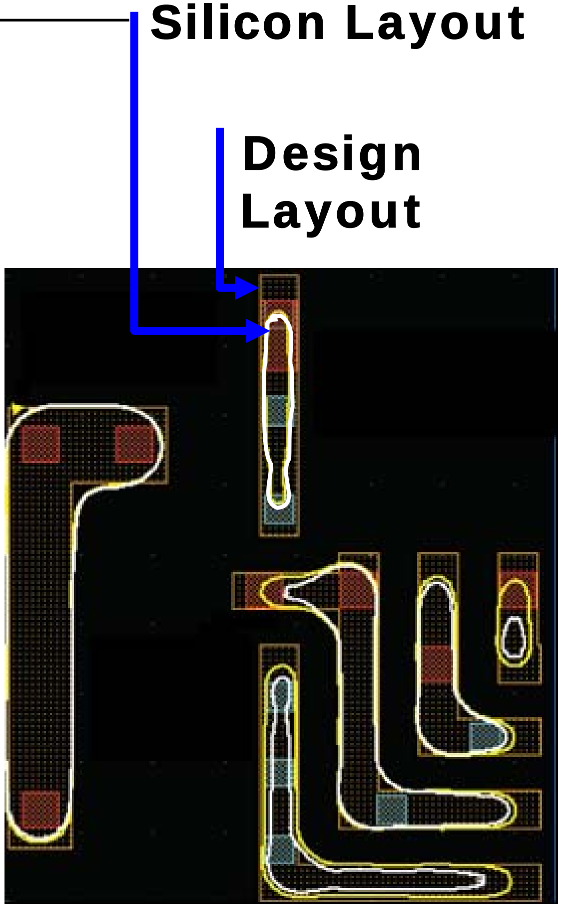
Outline

- Nanometer wiring rules
- Traditional rule handling methods
- MANA: a MAze algorithm under NAnometer rule
- Experimental results

Process Limits – Lithography

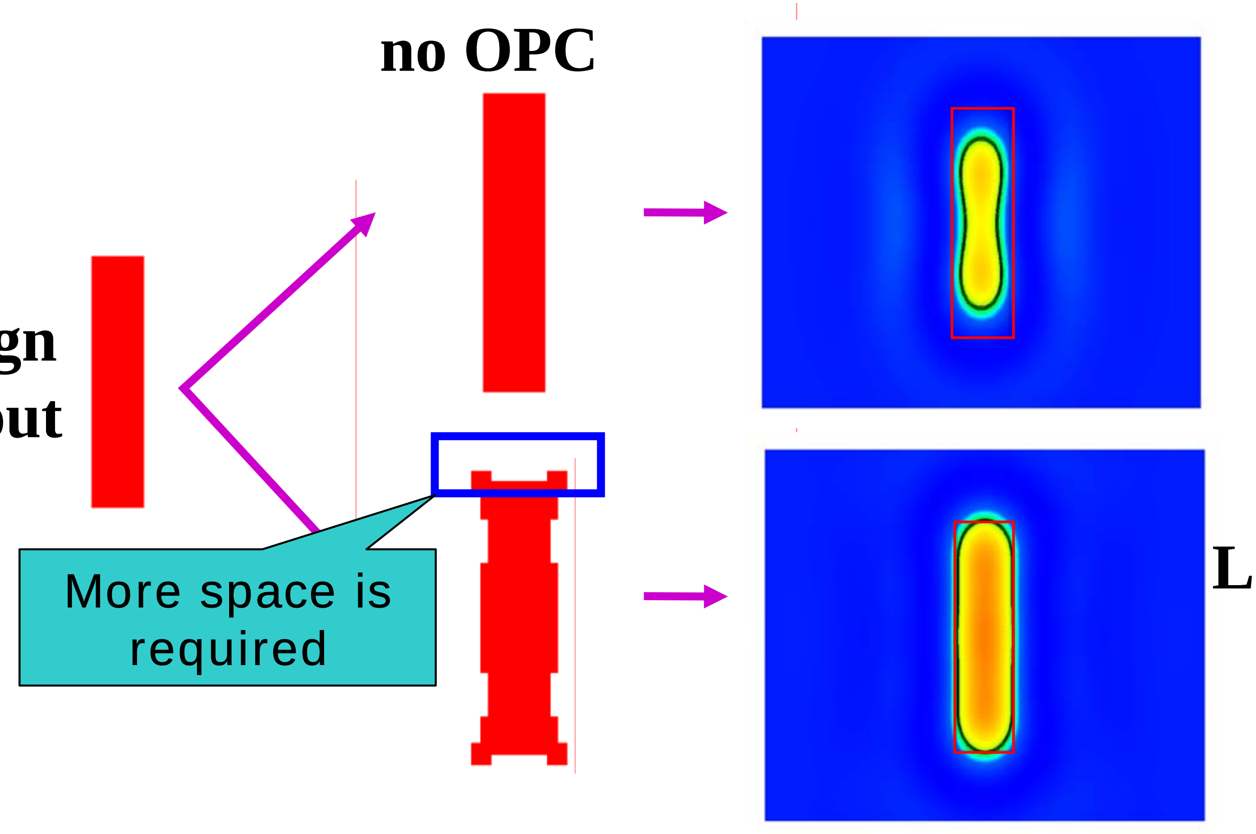
- Process shrinking





Optical Proximity Correction

- What you see is what you get



no OPC

Silicon Layout

Design Layout

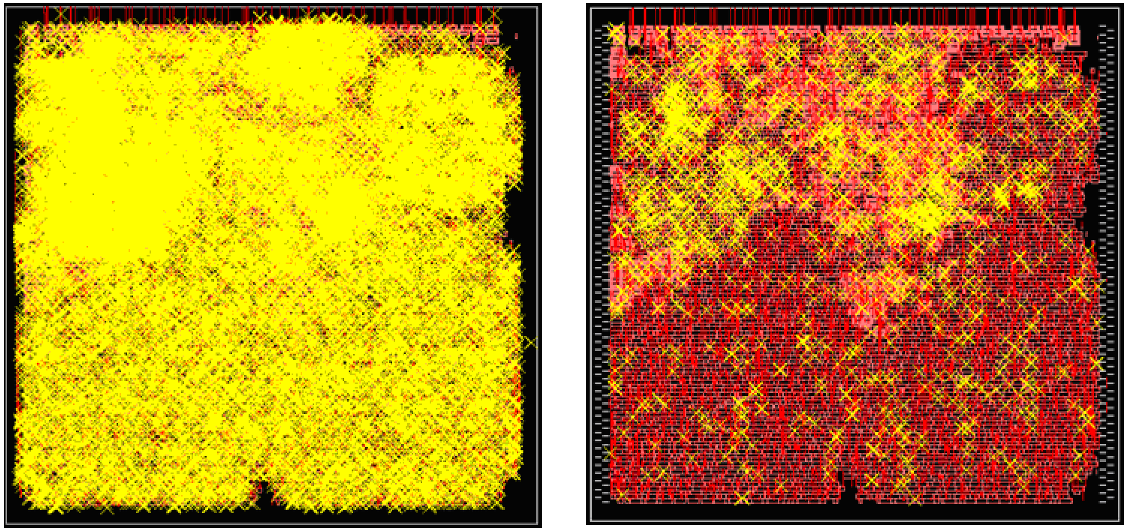
More space is required

Silicon Layout with OPC

with OPC

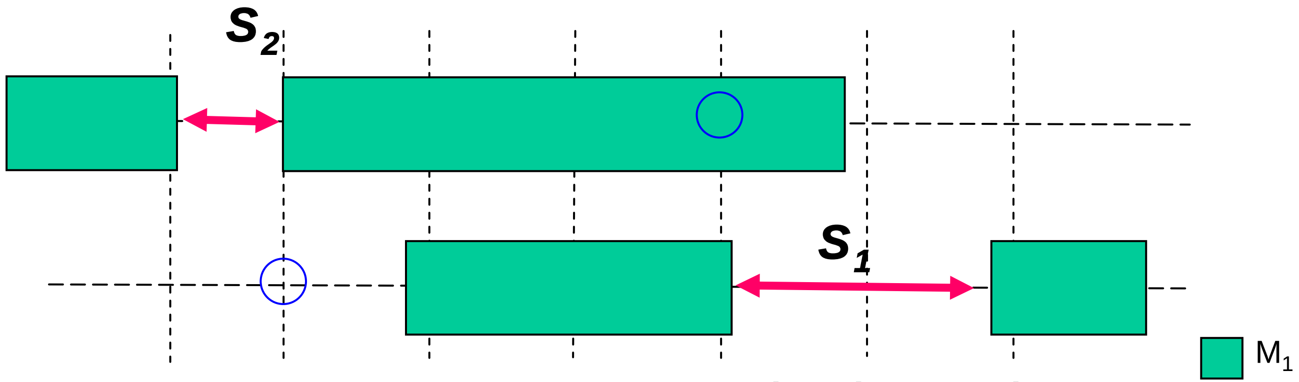
Major Violations

- 94% violations
- Rules
  - End-end separation rules
  - Minimum length rules



Design Rules (1/2)

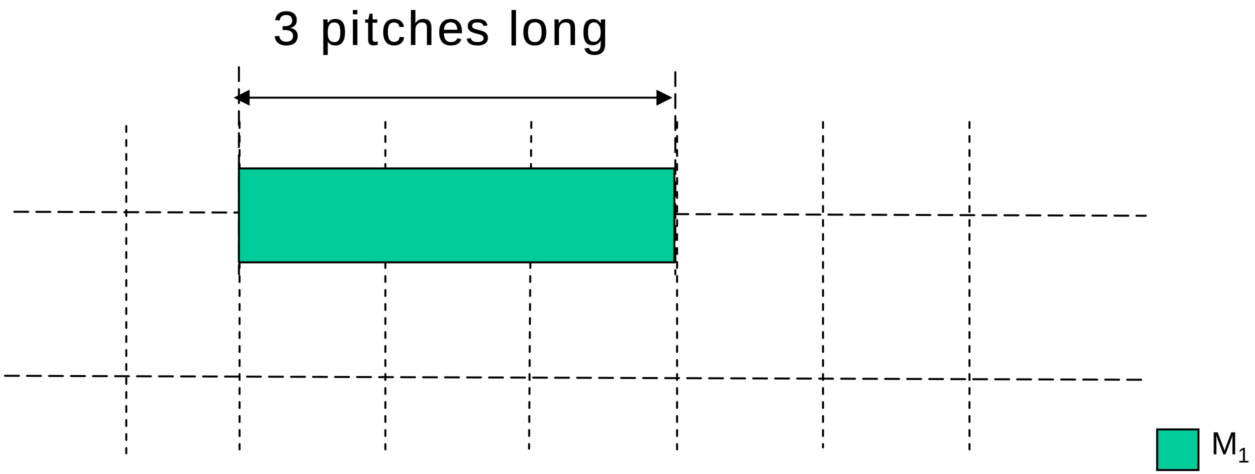
- The end-end separation rule
  - S_1 : if there is a wire near a wire end on a neighboring track
  - S_2 : others



End-End Separation

Design Rules (2/2)

- The minimum required length for each wire segment

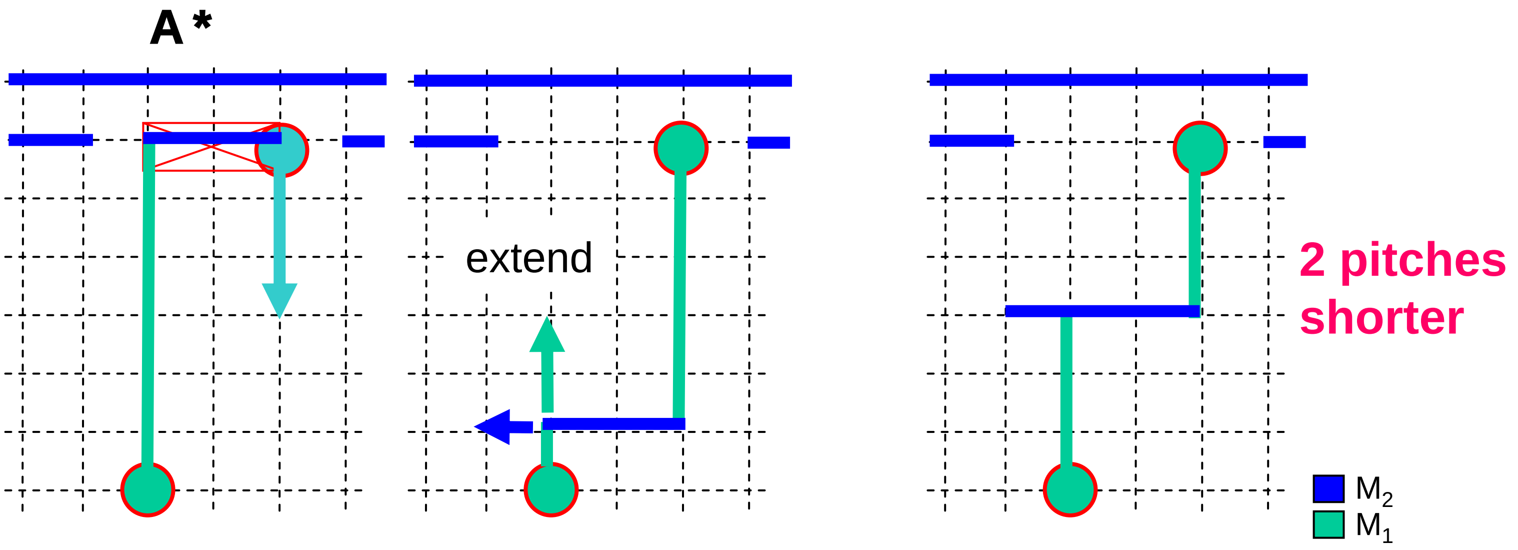


Outline

- Nanometer wiring rules
- Traditional rule handling methods
- MANA: a MAze algorithm under NAnometer rule
- Results and future work

Post-Routing Rule Handing

- A post rule handling method
  - 1. Find shortest paths without considering rules
  - 2. Extend short wires to meet the required length

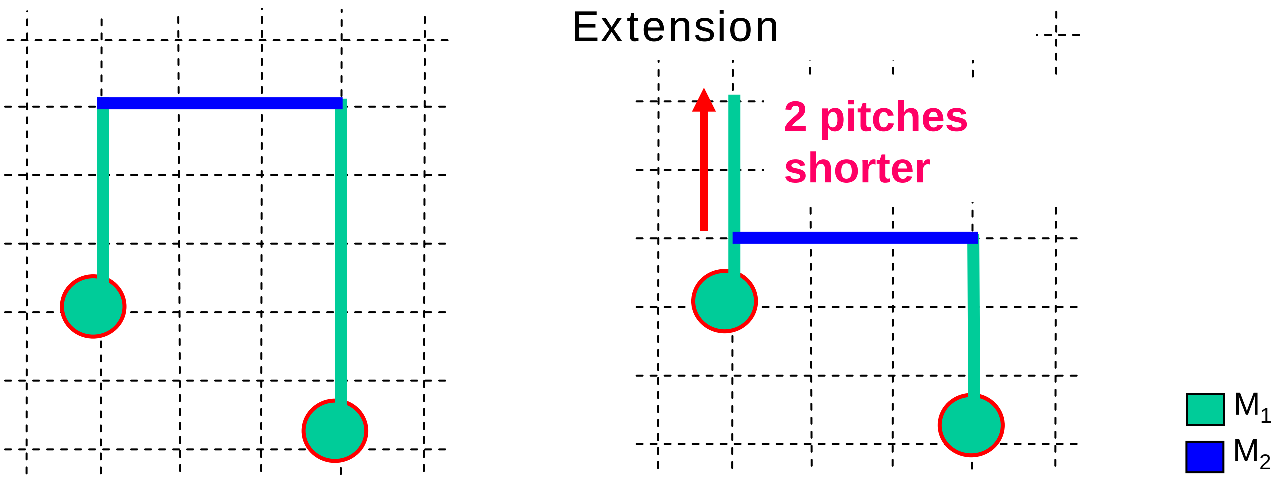


Min-Wire-Length: 3 Pitches

In-Routing Rule Handing 1/2

- A maze handling method [DAC2012]
  - A shortest path with **each segment** no less than the required length

Min-Wire-Length:
3 Pitches



without extension

with extension

Problem

- Paths found by traditional methods may contains violations or are not shortest.

Outline

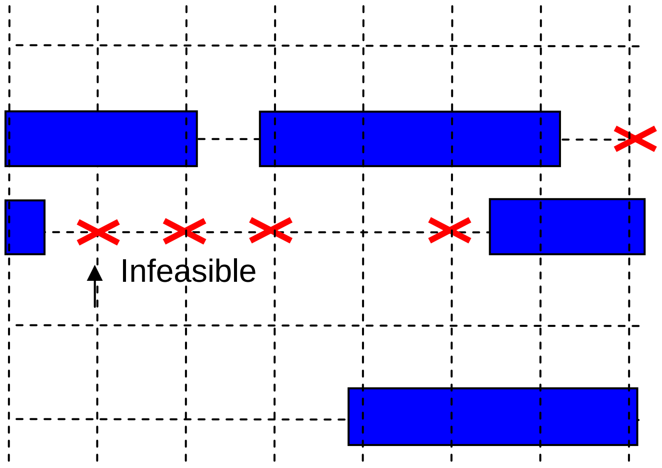
- Nanometer wiring rules
- Traditional rule handling methods
- MANA: a MAze algorithm under NAnometer rule
  - Algorithm flow
  - Runtime complexity
- Experimental results

The Proposed Algorithm

- MANA: a shortest path MAze algorithm under NAnometer Rules
  - End-end separation rules
  - The minimum wire length rule
- Goal
  - Resolve major violations

End-End Separation Rule Handling

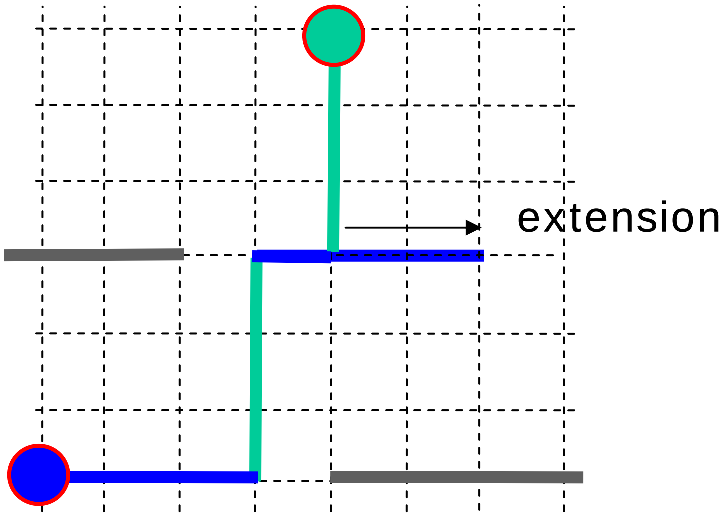
- Pre-filter out infeasible grid points
  - For each grid point, check if any end-end separation rules are violated when a wire passes through the point.



If there is a wire near a wire end, separation = 2 pitches

The Minimum Length Handling

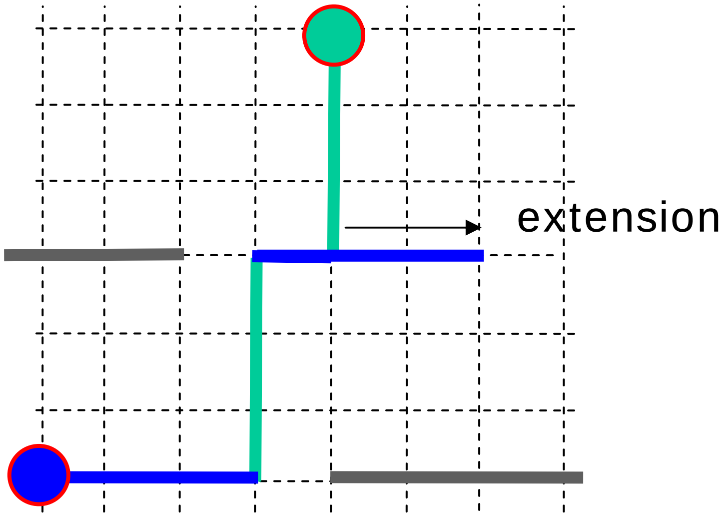
- Method
  - Extend short wires to meet the minimum wire length rule
- Extension
  - The extra extended wire segments



Min-Wire-Length: 3 Pitches

The Length of Path with Extension

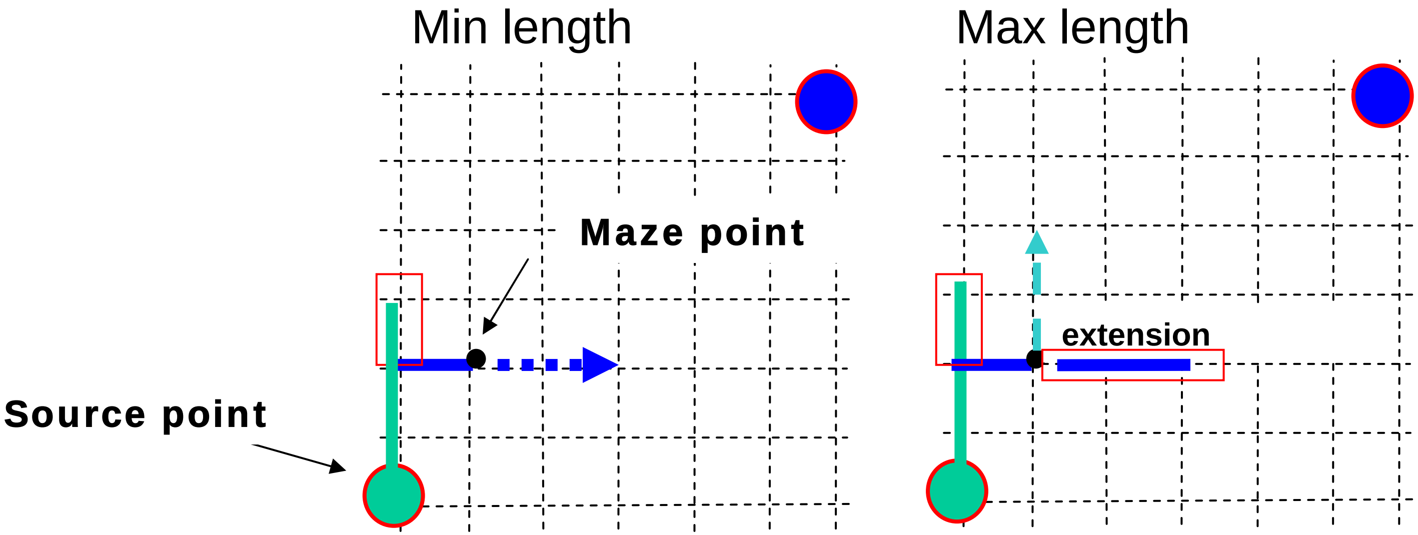
- Path length
  - The lengths of all wires and extensions



Min-Wire-Length: 3 Pitches

The Length of a Partial Path

- The length of a partial path
  - Min length : the length of last wire without extensions plus the sum of lengths of all previous wires with extensions.
  - Max length: the length of last wire with extension to meet the required length plus the sum of lengths of all previous wires with extensions.

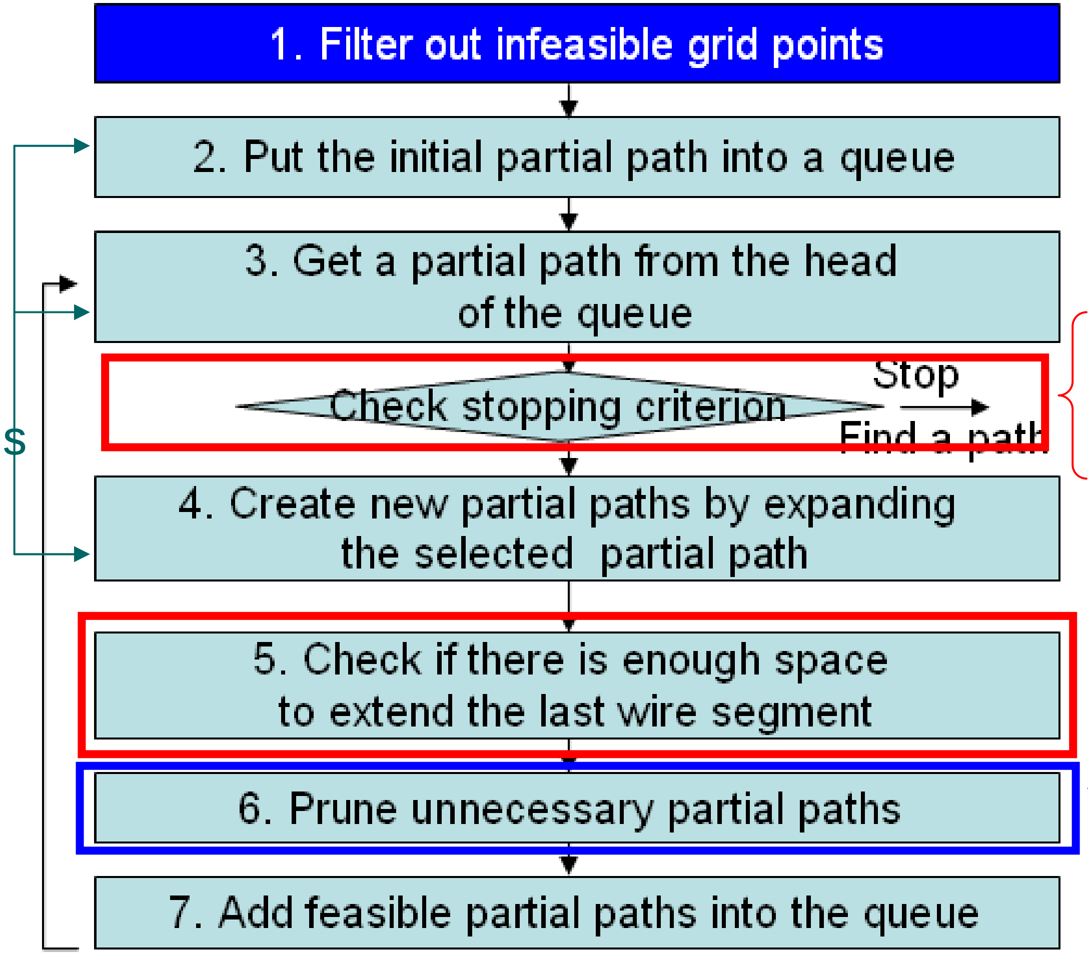


Min-Wire-Length: 3 Pitches

MANA Algorithm

- Concept: search all partial paths from the source and ensure there are enough spacing for wire extension
- Polynomial runtime complexity

MANA Algorithm Flow


1. Filter out infeasible grid points
2. Put the initial partial path into a queue
3. Get a partial path from the head of the queue

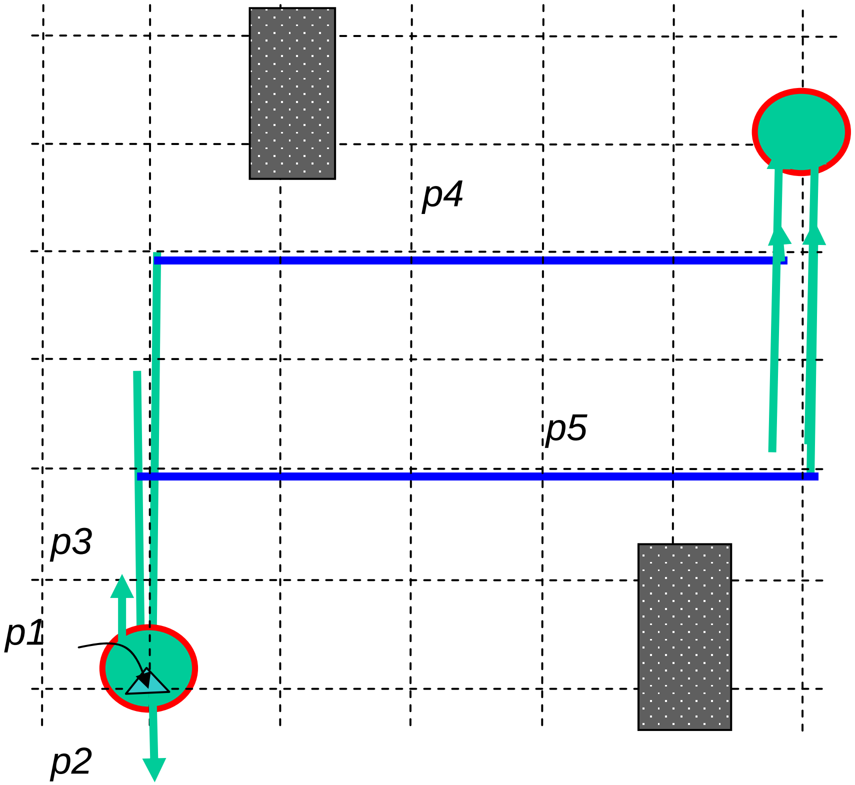
4. Create new partial paths by expanding the selected partial path
5. Check if there is enough space to extend the last wire segment
6. Prune unnecessary partial paths
7. Add feasible partial paths into the queue

Search All possible partial paths

Find a path with length no greater than all min length of partial paths

Polynomial time complexity

An example



Min-Wire-Length: 3 Pitches

min length	
p5	*10*
p4	*9*

Path from *p4*: 12

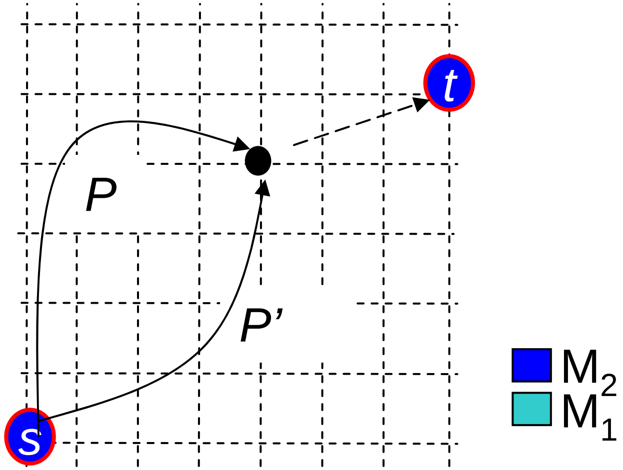
Path from *p5*: 11

Polynomial Runtime Complexity

- Step 6: Prune unnecessary partial paths
  - If any full path extended from a partial path can be replaced by another partial path without increasing the path length, the partial path can be pruned.
  - Two pruning strategies

Two Pruning Strategies 1/2

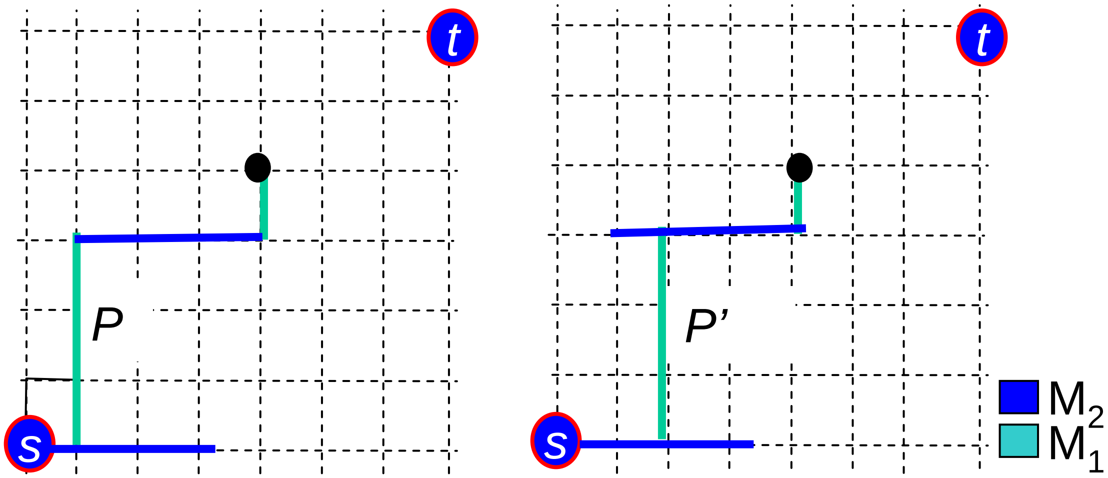
- Strategy 1
  - *P* and *P*’ are two partial paths from the source and connect to a same target point
  - Max(P') $\leq$ Min(P)
    - Prune *P*



Max(): the max length, **Min()**: the min length

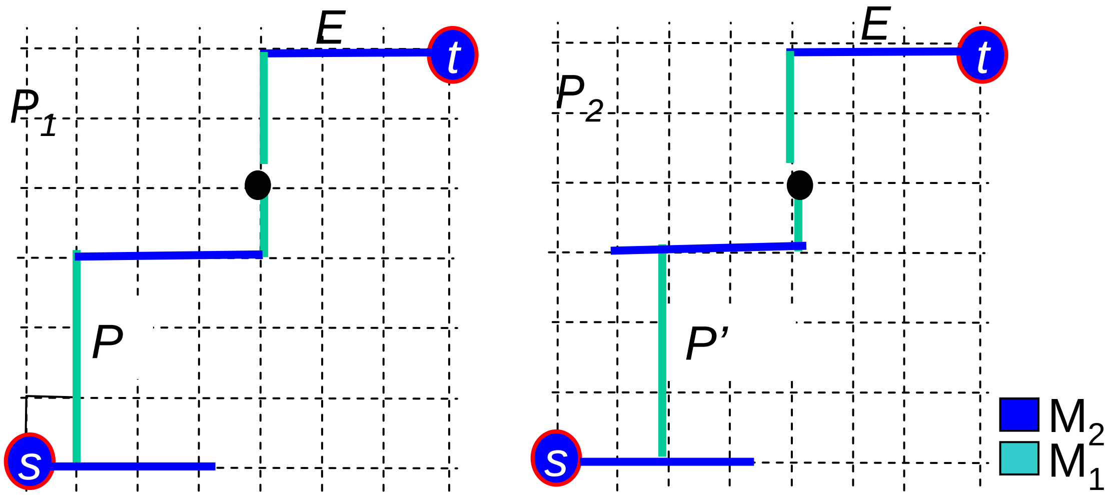
Two Pruning Strategies 2/2

- Strategy 2
  - Max(*P'*) = Max(*P*) and Min(*P'*) = Min(*P*)
    - Prune *P*



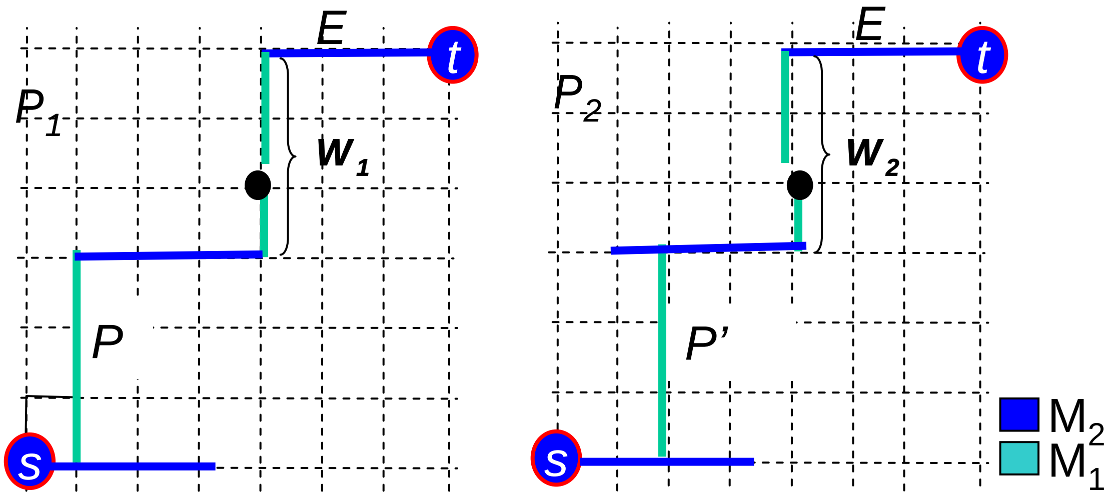
Equal Path Length Theorem 1/3

- **Theorem:** If Max(P)=Max(P') *and* Min(P)= Min(P'), then L(P_1) = L(P_2), where P_1 and P_2 are full paths extended from P and P' respectively with the same extension to the target.



Equal Path Length Theorem 2/3

- W_1 and W_2 contain no extension
  - $L(P_1) = \mathrm{Min}(P) + \mathrm{Min}(E)$
  - $L(P_2) = \mathrm{Min}(P') + \mathrm{Min}(E)$



L(): the length of a path

Equal Path Length Theorem 3/3

- W_1 and W_2 both contain extensions
  - $\mathrm{L}(P_1) = \mathrm{L}(P_2)$
- Only one of W_1 and W_2 contains extensions
  - Non existing case

Algorithm Characteristics

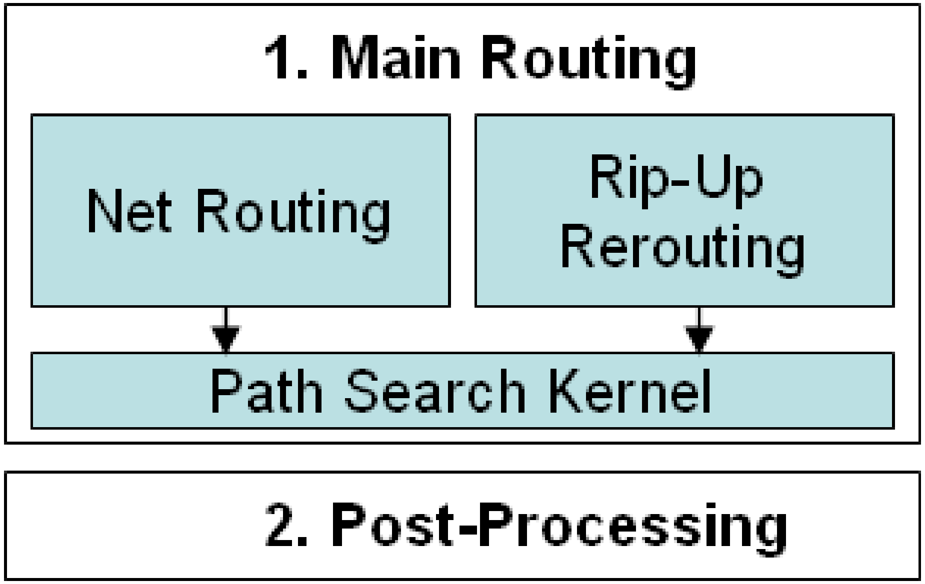
- Prevent major violations
- Special pruning strategies
  - Polynomial time complexity

Outline

- Nanometer wiring rules
- Traditional rule handling methods
- MANA: a MAze algorithm under NAnometer rule
- Experimental results

Architecture

- CDR/SYNOPSYS
- Path Search Kernel
  - A* -> MANA

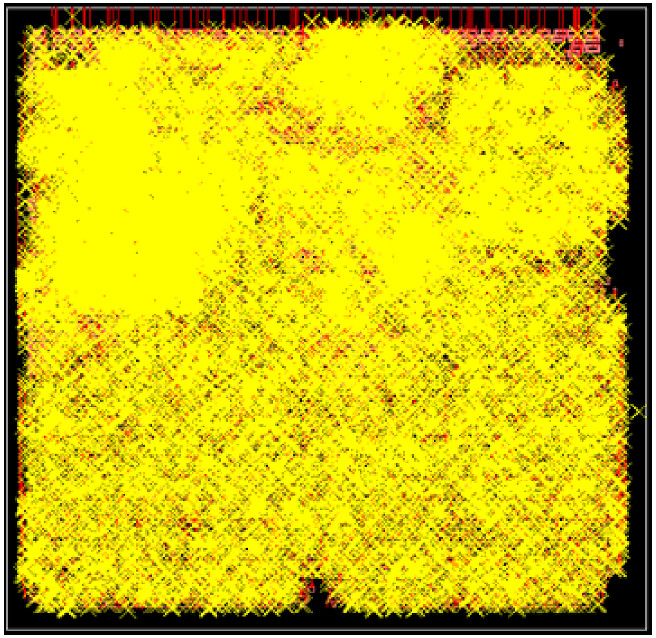


7 Test Cases

Cases	***#inst***	***#net***	***Process***
C1	68,472	12,752	65 nm
C2	1,558	1,775	40 nm
C3	8,971	11,210	40 nm
C4	2,221	407	40 nm
C5	9,984	1,837	40 nm
C6	9,984	1,837	28 nm
C7	11,894	11,210	28 nm

Violation Prevention

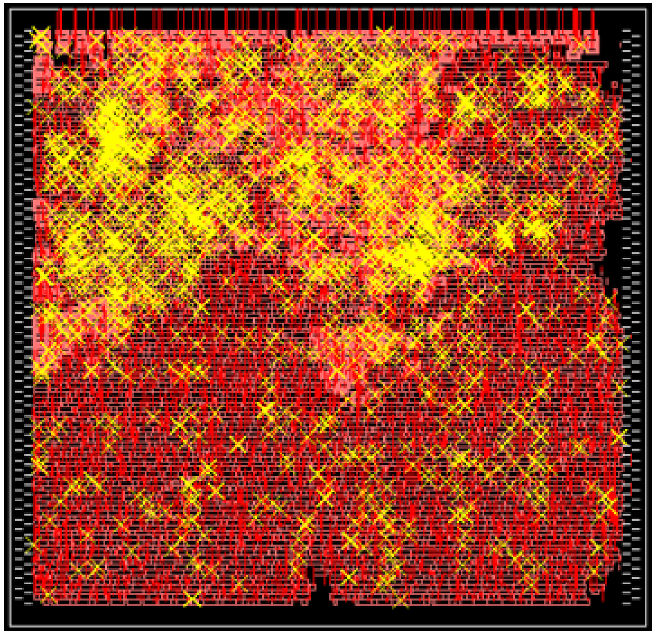
Original Router with A*



32,313 vios

Case: C3

Original Router with MANA



1,816 vios

Experimental Results

- Average runtime
  - > 2 times reduced
- Wiring quality
  - Wirelenth and via# are slightly reduced.

	Original router				**Router with MANA**				**Comparison**		
Case	*time(sec)*	*W.L.(mm)*	*#via*	*#vio*	*time(sec)*	*W.L.(mm)*	*#via*	*#vio*	*t-r*	*w-r*	*v-r*
C1	293	237,084	142,541	0	150	230,178	138,384	0	2.0	1.03	1.03
C2	16	32,943	14,356	45	7	31.983	14,220	**0**	2.3	1.03	1.01
C3	148	111,359	90,504	0	75	108,115	88,742	0	2.0	1.03	1.02
C4	105	9,796	3,373	12	31	9,618	3,357	**0**	3.4	1.02	1.00
C5	492	27,294	17,201	0	101	26,500	17,045	0	4.9	1.03	1.01
C6	188	17,295	15,924	38	170	16,950	15,761	4	1.1	1.02	1.01
C7	209	88,563	74,553	0	183	85,951	74,625	0	1.1	1.03	1.00
			Success	**57%**			**Success**	**86%**	**Avg.** **2.4**	1.03	1.01

Conclusion

- Nanometer wiring rules
  - End-end separation
  - Minimum wire length
- MANA: A Maze routing algorithm under NAnometer rules
  - Equal path length theorem
- Experimental rules
  - 94% violations are prevented
  - > 2 times runtime reduction

Q&A

Reference

- An end-end violation caused by two wire segments of a path