國立交通大學電子工程學系

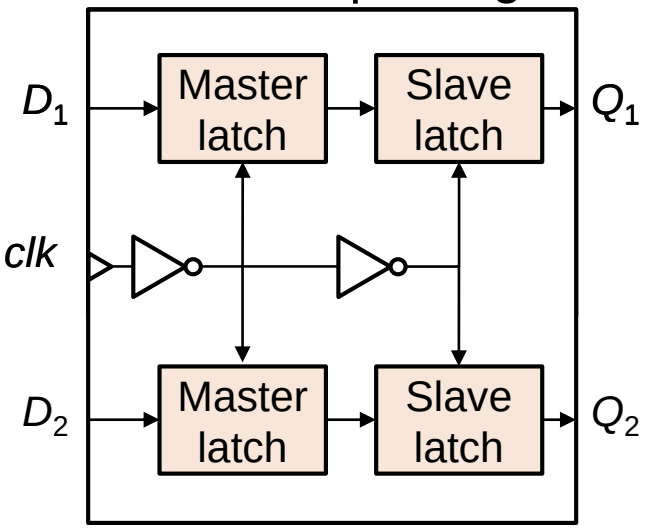
FF-Bond: Multi-bit Flip-flop Bonding at Placement

CHANG-CHENG TSAI
YIYU SHI
GUOJIE LUO
IRIS HUI-RU JIANG

Outline

Multi-Bit Flip-Flops (MBFFs)

- **Clock power is critical for modern IC designs**
  - $Dynamic\ power \propto CV^2 f$
- **MBFFs present a smaller load on the clock network**
- **Replace FFs with MBFFs**
  - Effectively reduces both clock network power and MBFF power
  - Prefer large FFs (high bit number), avoid orphans (single-bit FF)
  - Avoid impacting timing critical paths



Bit number	Normalized power per bit	Normalized area per bit
1	1.00	1.00
2	0.86	0.96
4	0.78	0.71

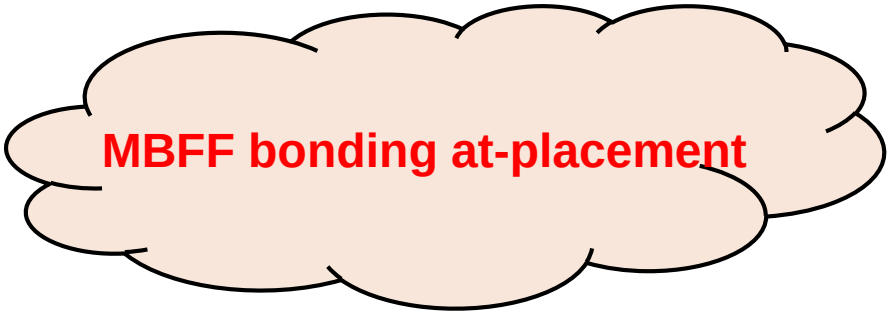
Power efficient

Prior Work

- **Relocating flip-flops benefits clock network synthesis**
  - [Cheon+, DAC05], [Papa+, Micro11], [Lee/Markov, TCAD12]

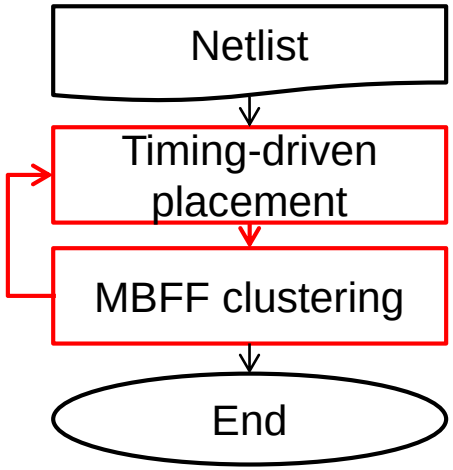
- **Replacing flip-flops with MBFFs saves clock power**
  - [Yan/Chen, ICGCS10], [Chang+, ICCAD10], [Wang+, ISPD11] [Jiang+, ISPD11], [Liu+, DATE12]
  - Focus on **post-placement** MBFF clustering

- **Pre-placement**
  - Lack physical information
- **Post-placement**
  - Cells are immovable
  - Limited clustering flexibility and quality



One Possible Solution …

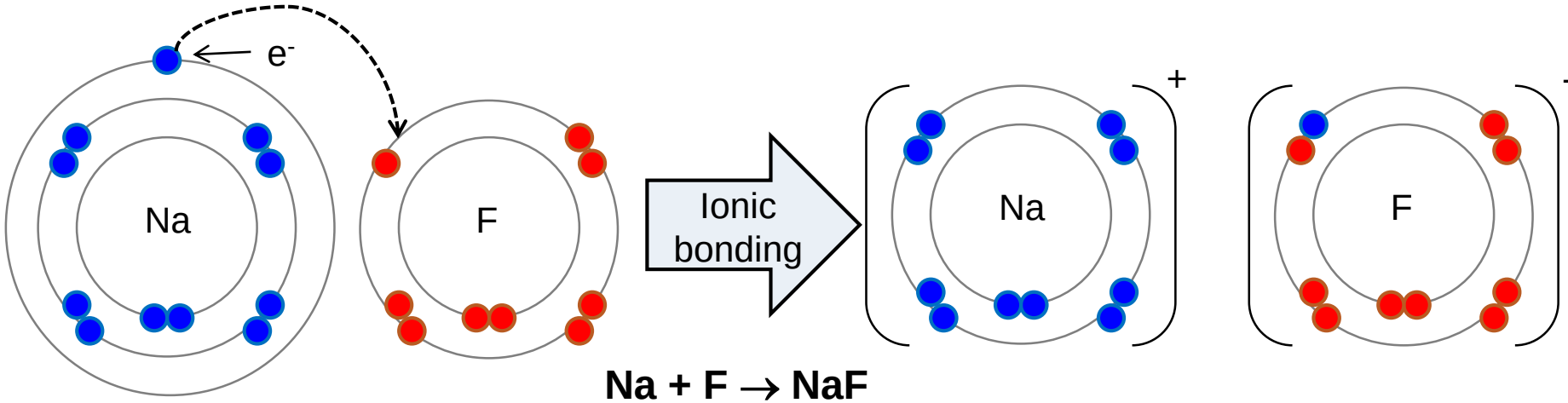
- **Directly integrate placement & post-placement MBFF clustering**



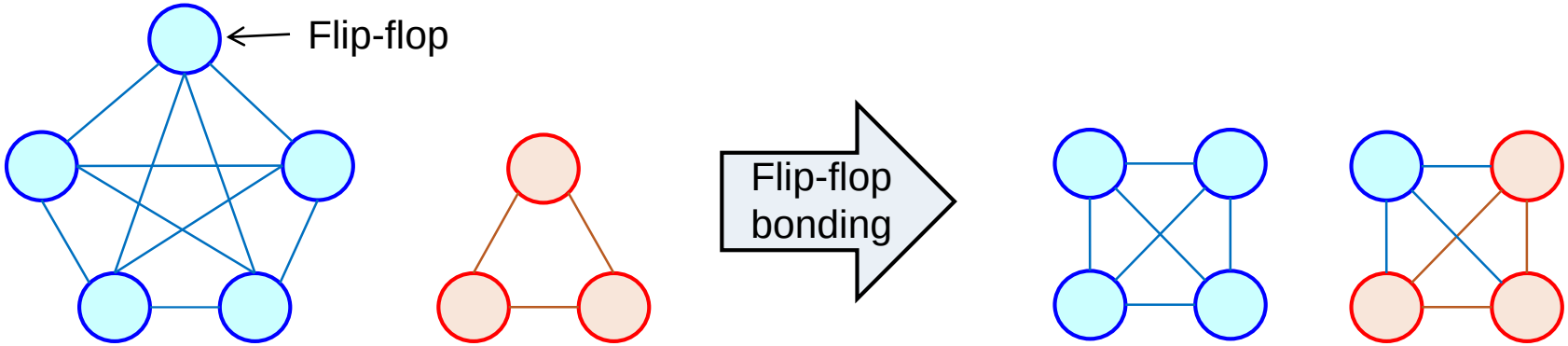
- **The movement of flip-flops**
  - Is constrained by the placement at the current iteration
  - May oscillate among iterations

Ionic Bonding and Flip-flop Bonding

- **Goal: Guide flip-flops towards merging friendly locations**



Na + F → NaF



Mergeable flip-flop sets

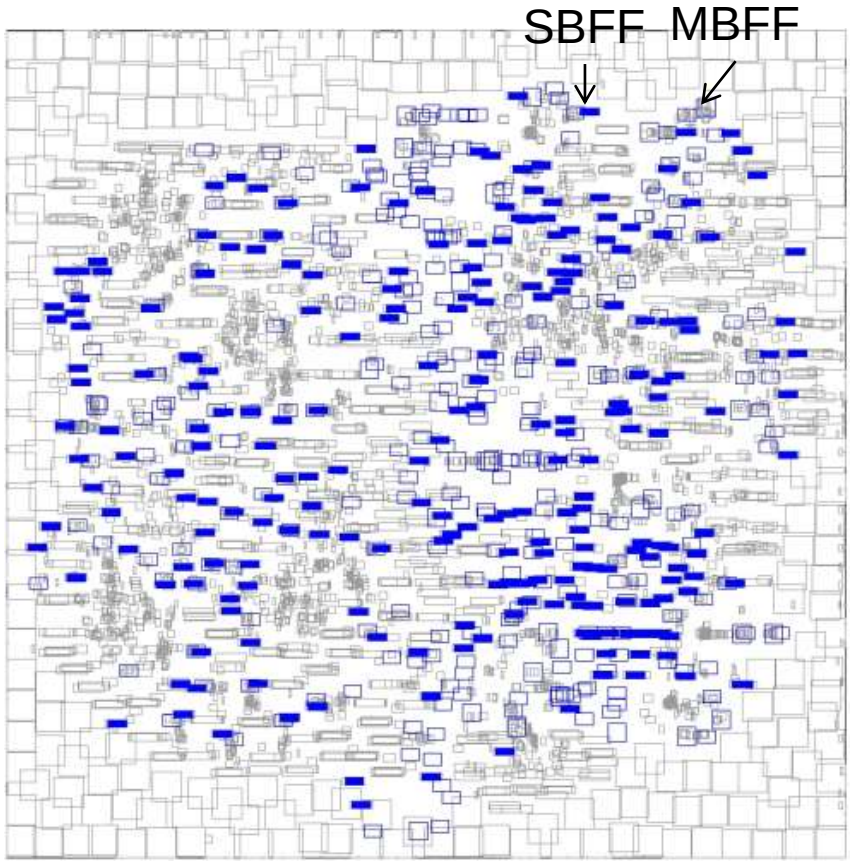
Example: MBFF library: 1-bit, 2-bit, 4-bit

http://en.wikipedia.org/wiki/Ionic_bond

Post-Placement vs. At-Placement - s38417

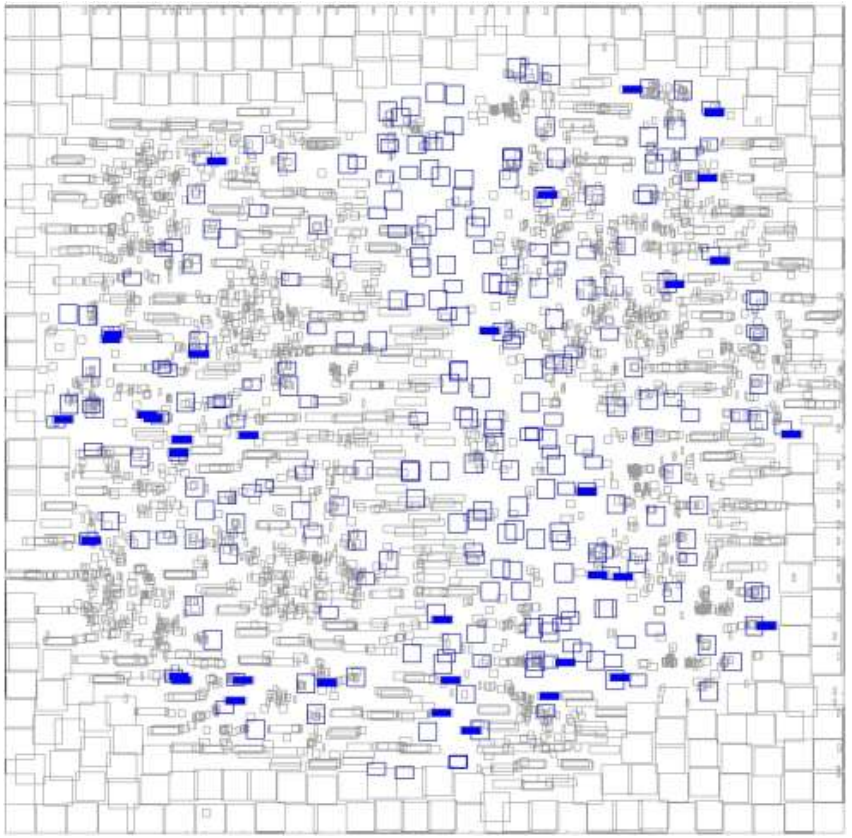
Post-placement clustering

- **# MBFFs 4-/2-/1-bit**
  - 35/252/237



At-placement bonding

- **# MBFFs 4-/2-/1-bit**
  - 159/105/35

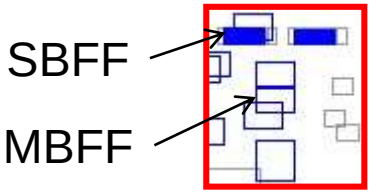


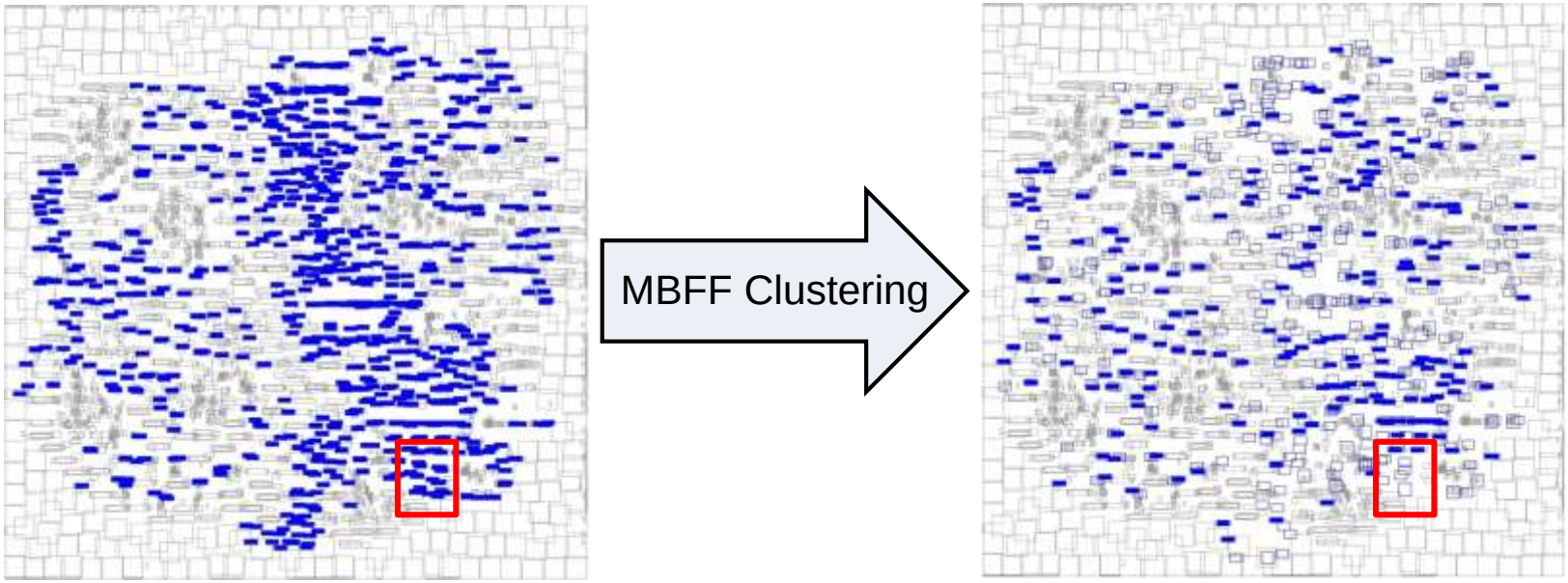
Outline

- Introduction
- **Preliminaries**
- Problem formulation
- Algorithm - FF-Bond
- Experimental results
- Conclusion

Post-Placement MBFF Clustering

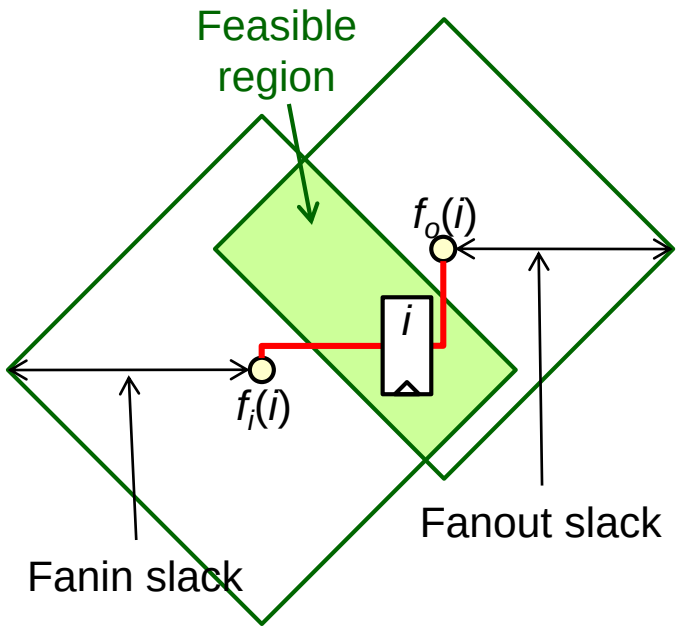
- **Given**
  - A placed design
  - MBFF library
  - Timing slacks of flip-flops
- **Replace FFs with MBFFs**
  - Minimize flip-flop power
  - Satisfy timing constraints

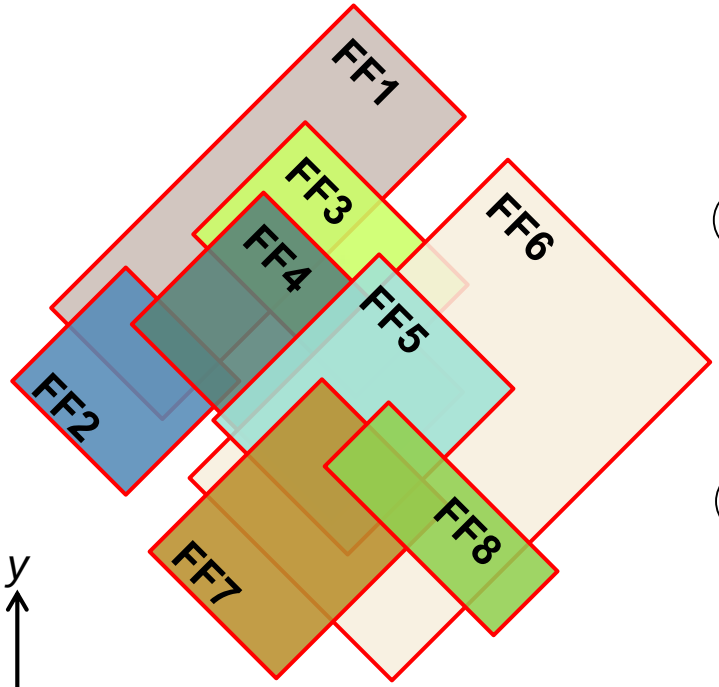


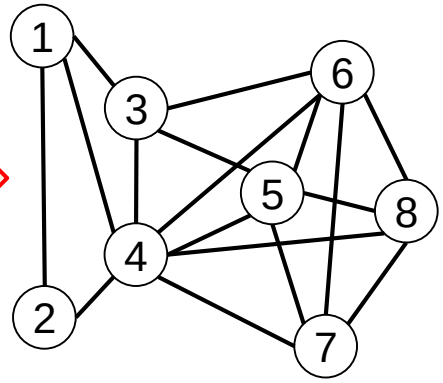


Intersection Graph

- **Define the feasible region of a flip-flop according to its slack**
- **Model the overlap of feasible regions by an intersection graph**
  - A proper-sized clique corresponds to an MBFF



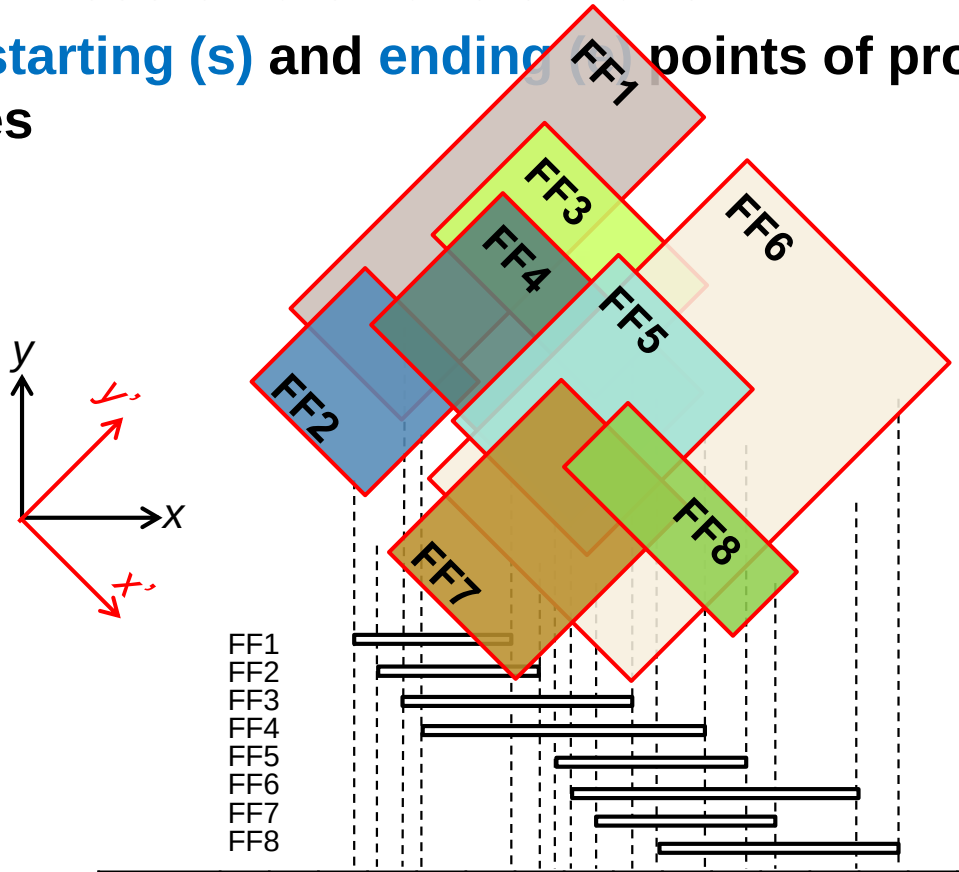




Intersection graph

INTEGRA (INTErval GRAph)

- **Perform coordinate transformation**
- **Sort starting (s) and ending (e) points of projection in the *x'* and *y'* axes**

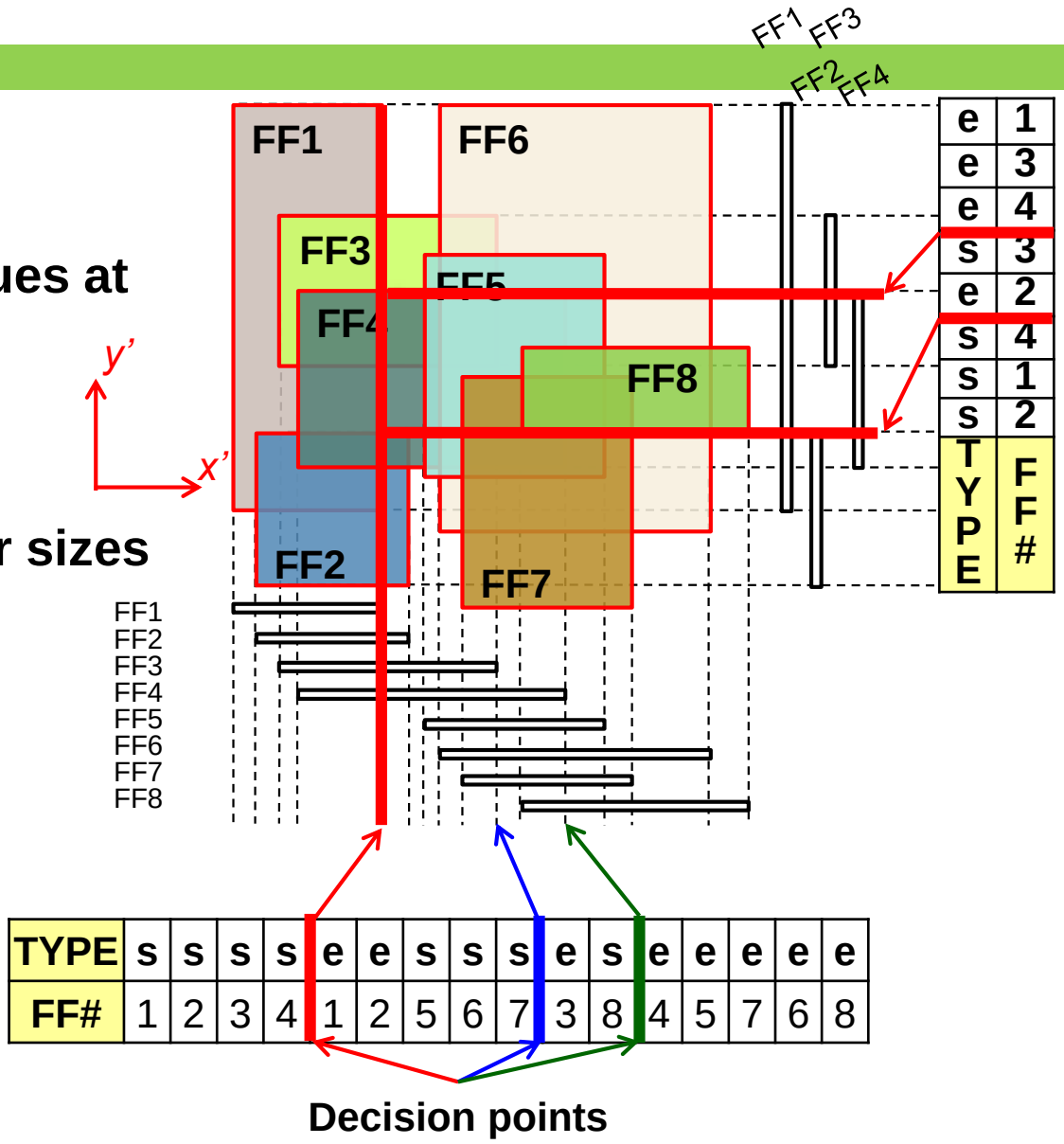


TYPE	s	s	s	s	e	e	s	s	s	e	s	e	e	e	e	e
FF#	1	2	3	4	1	2	5	6	7	3	8	4	5	7	6	8

Jiang *et al.* "INTEGRA: Fast multibit flip-flop clustering for clock power saving," TCAD12, ISPD11.

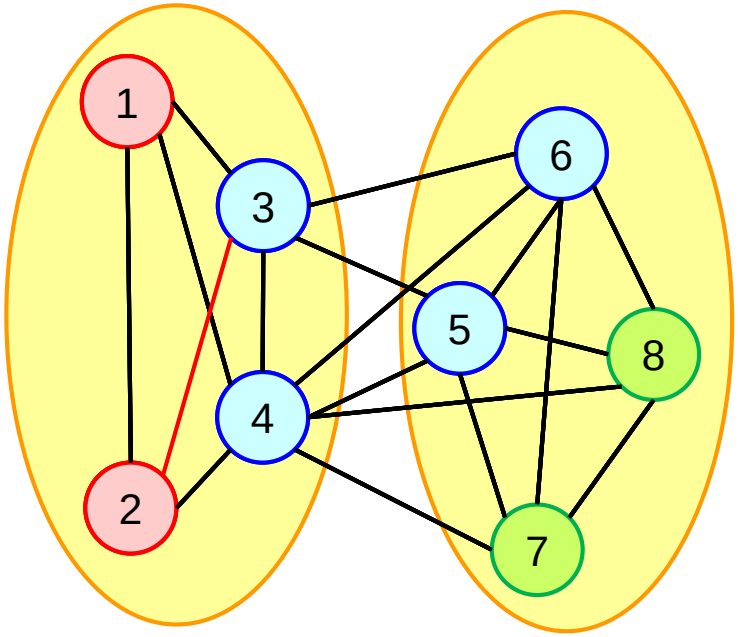
INTEGRA

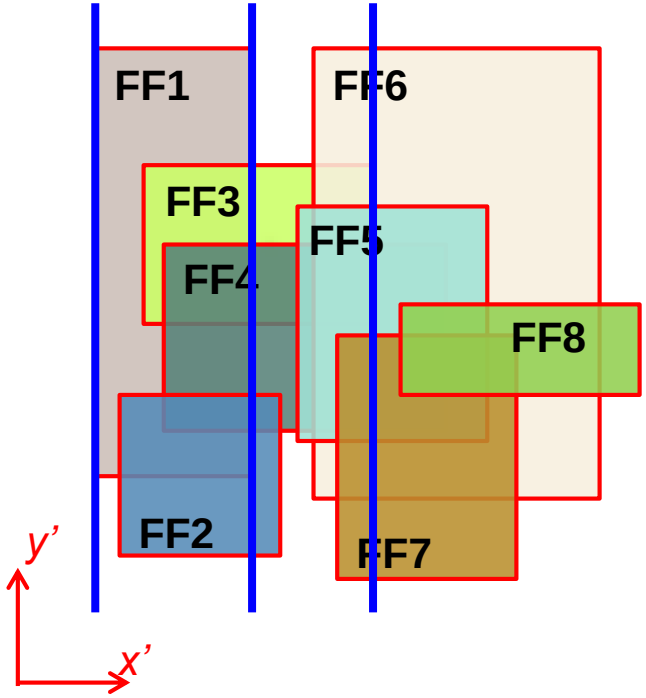
- **Find decision points**
  - ‘se’ in *x’* axis
- **Retrieve maximal cliques at decision points**
  - Check *x’* and *y’* axes
  - {1, 2, 4} or {1, 3, 4}
- **Form MBFFs of proper sizes**
  - e.g., {1, 2}



Example: INTEGRA

- **Example: MBFF library: 1-bit, 2-bit, 4-bit**





Guide FFs towards merging friendly locations

Outline

- Introduction
- Preliminaries
- **Problem formulation**
- Algorithm - FF-Bond
- Experimental results
- Conclusion

The MBFF Bonding at Placement Problem

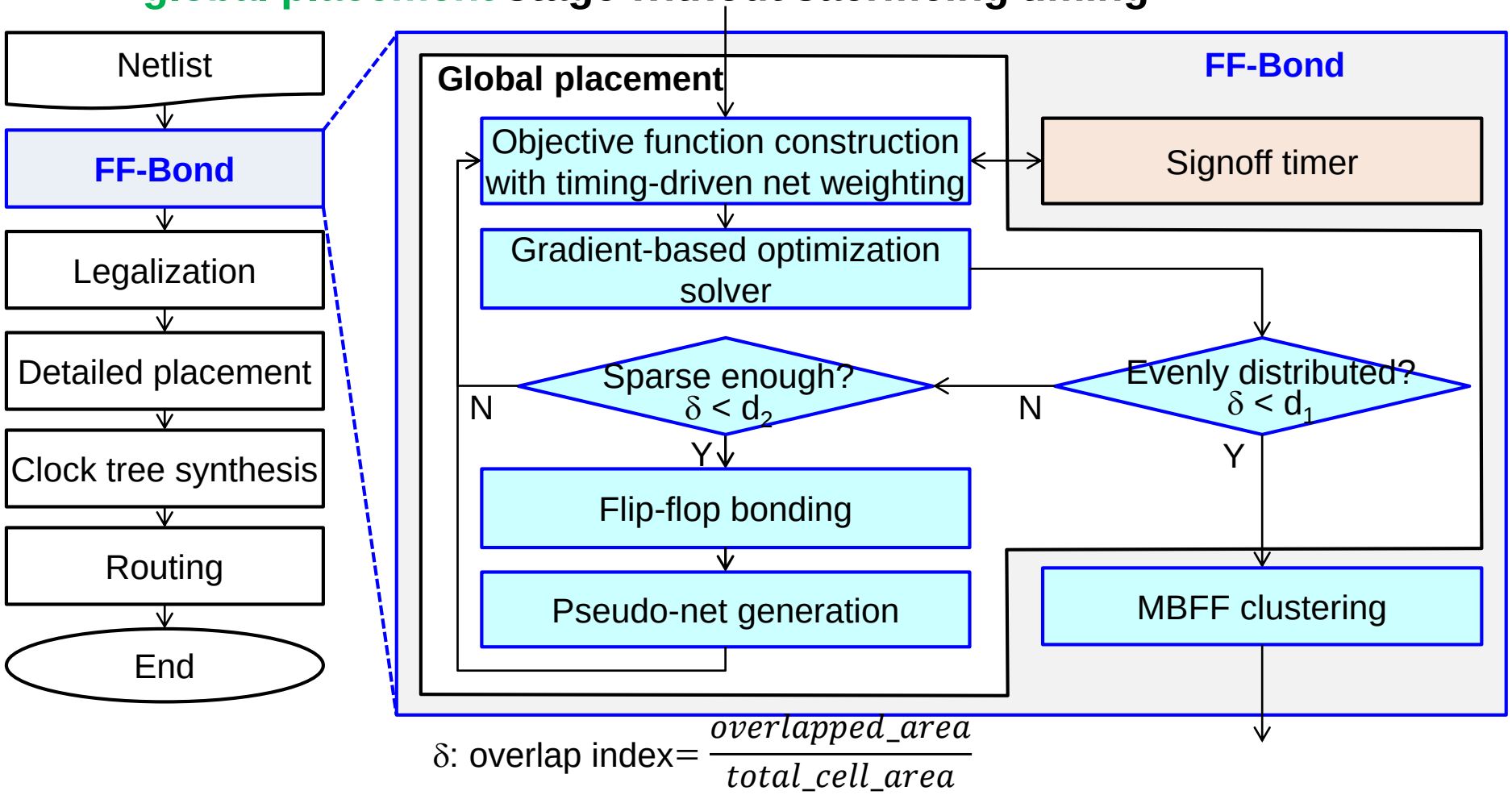
- **Given**
  - Gate-level netlist
  - MBFF library
  - Timing constraints

- **Find a placement and replace FFs with MBFFs**
  - Minimize flip-flop power
  - Satisfy timing constraints

Outline

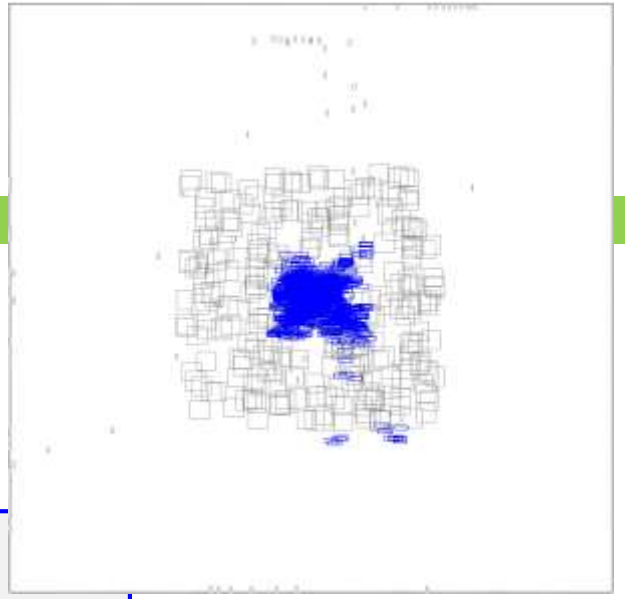
The Overview of FF-Bond

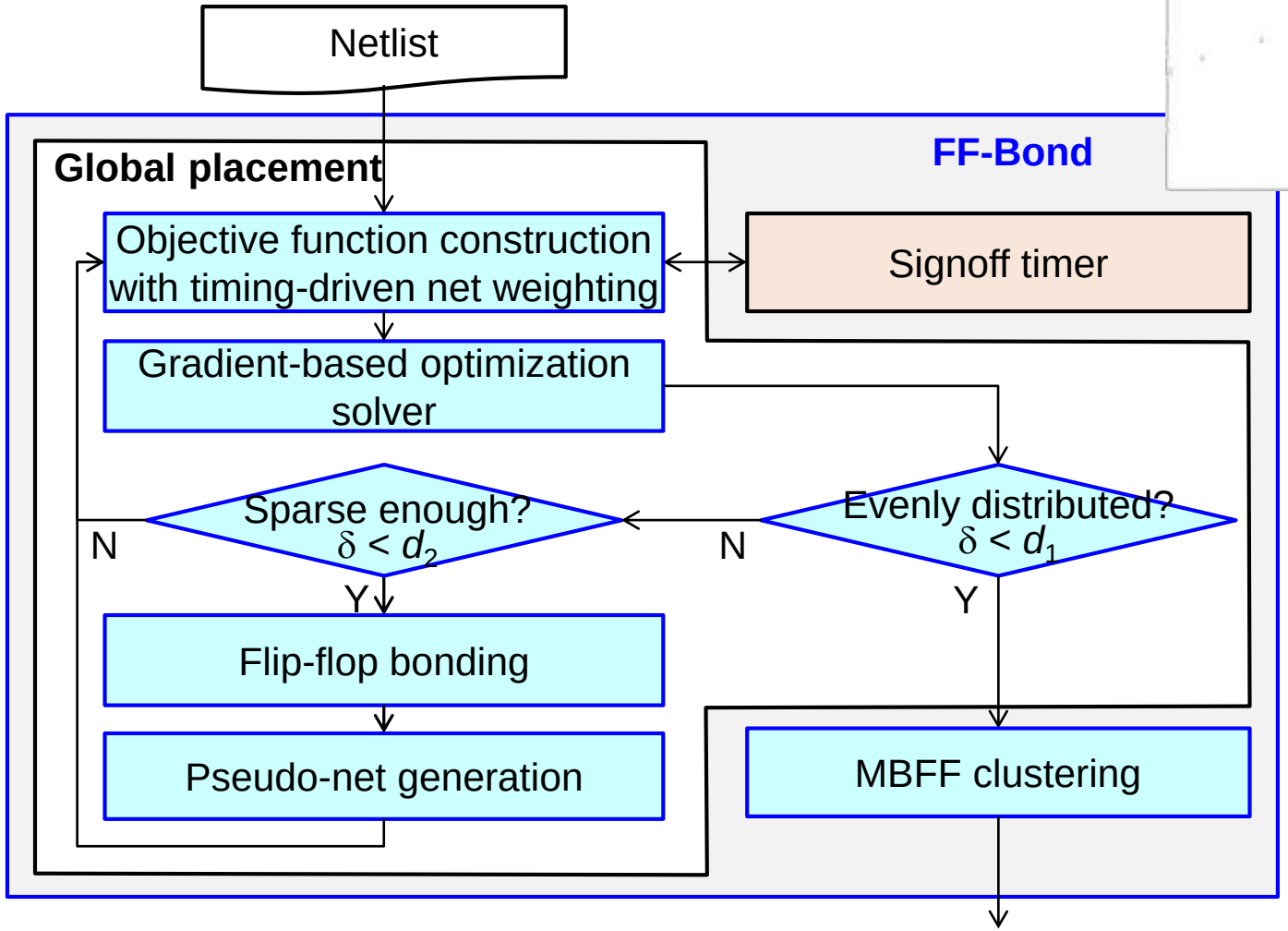
- **Guide flip-flops towards merging friendly locations at the global placement stage without sacrificing timing**



δ: overlap index= $\frac{overlapped\_area}{total\_cell\_area}$

Example: s38584

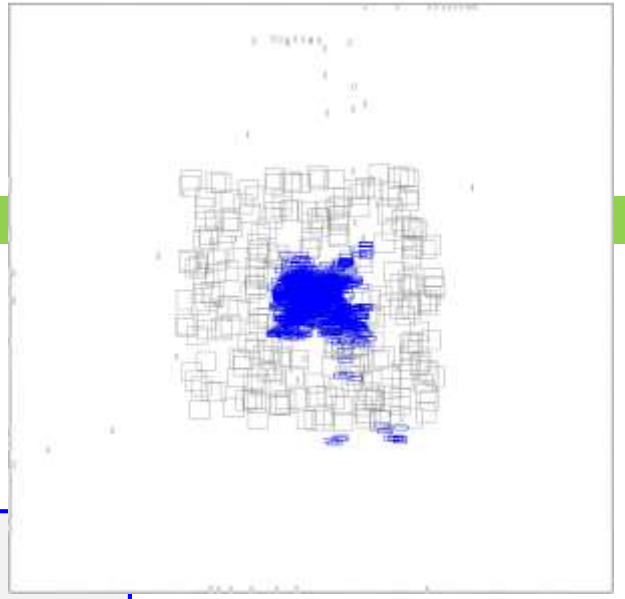




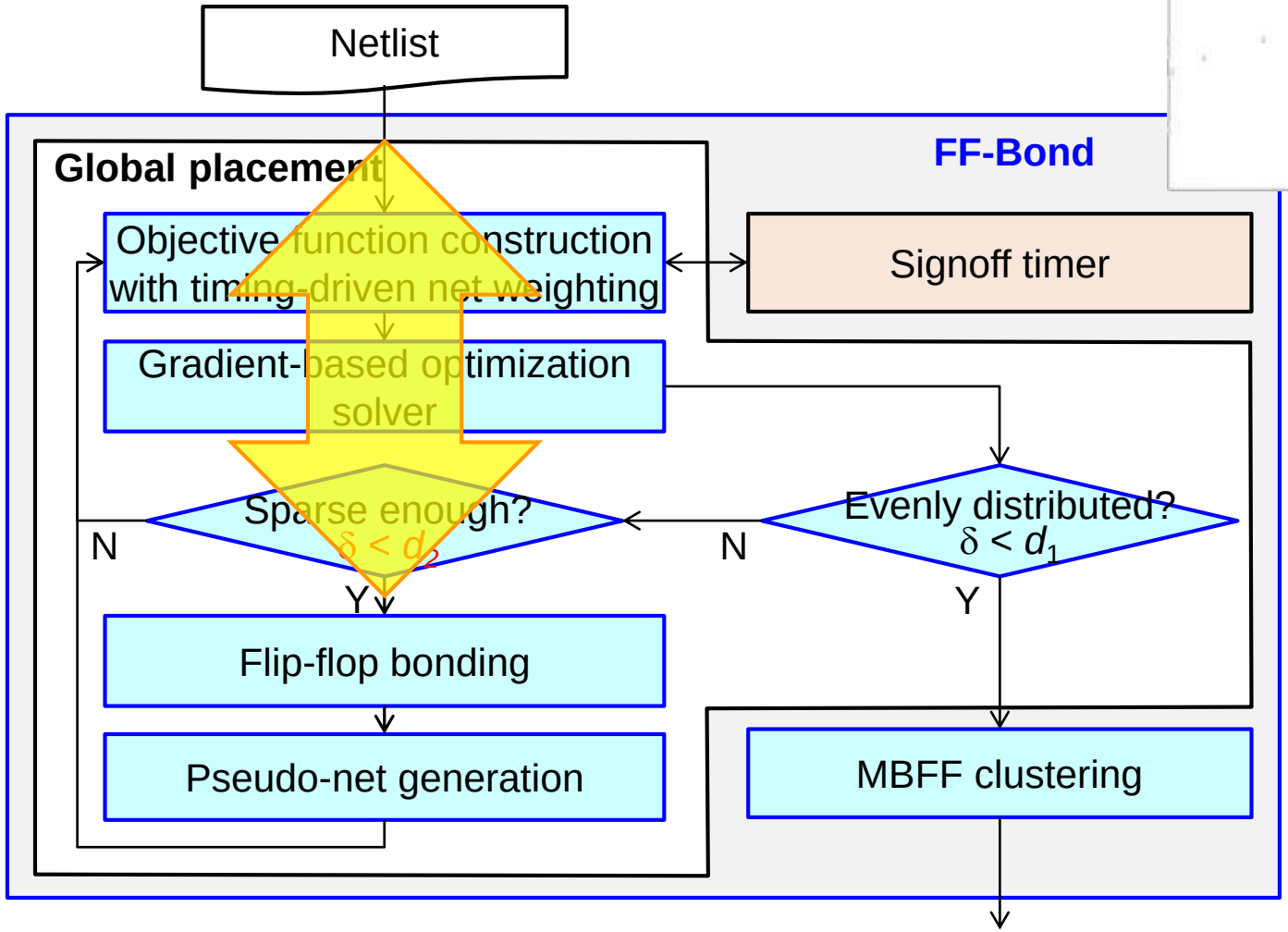
δ: overlap index

$$= \frac{overlapped\_area}{total\_cell\_area}$$

Example: s38584



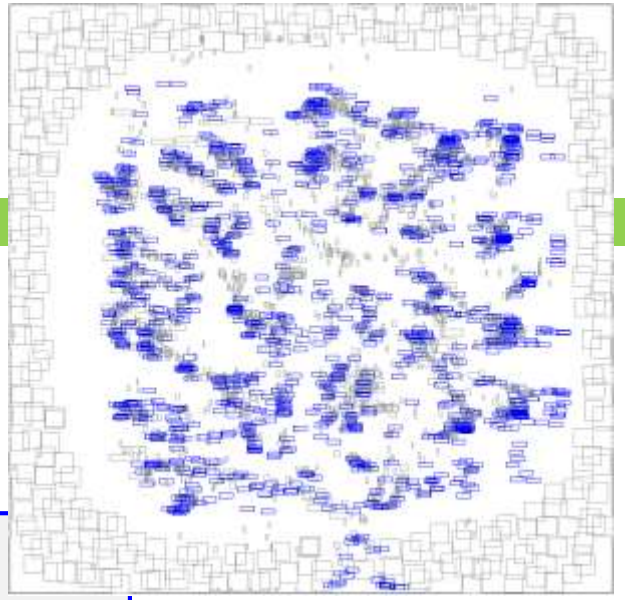
- **Spread cells until sparse enough**

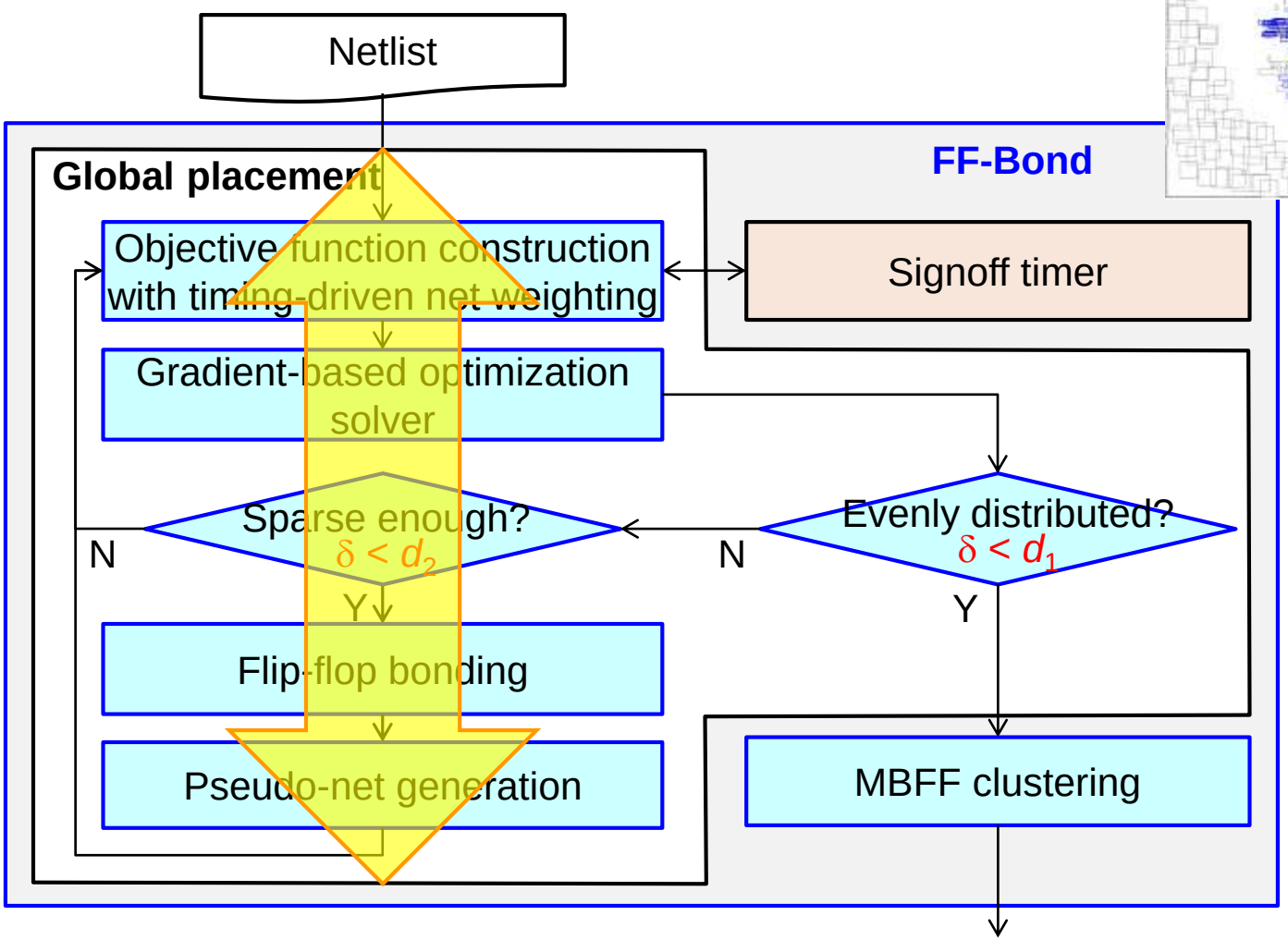


δ: overlap index $= \frac{overlapped\_area}{total\_cell\_area}$

Example: s38584

- **Apply flip-flop bonding**





δ: overlap index

$$= \frac{overlapped\_area}{total\_cell\_area}$$

Timing-Driven Placement

- **Pure wirelength-driven placement + slack-based net-weighting**
- **Pure wirelength-driven analytical placement: mPL**
  - $\textbf{min} \quad W(x,y) = \sum_{e \in E} \left(\max_{v_i, v_j \in e, i<j} |x_i - x_j| + \max_{v_i, v_j \in e, i<j} |y_i - y_j| \right)$

    $\textbf{s.t.} \quad D_{ij} = K, \qquad 1 \le i \le m, 1 \le j \le n$

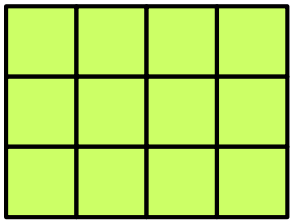
  - Smooth the objective function and the constraints
    - Log-sum-exp approximation

$$\widehat{W}(x,y) = \eta \sum_{e \in E} \left(\log \sum_{v_k \in e} \exp\left(\frac{x_k}{\eta}\right) + \log \sum_{v_k \in e} \exp\left(\frac{-x_k}{\eta}\right) + \log \sum_{v_k \in e} \exp\left(\frac{y_k}{\eta}\right) + \log \sum_{v_k \in e} \exp\left(\frac{-y_k}{\eta}\right) \right)$$

    - Inverse Laplace transformation

$\textbf{min} \quad \widehat{W}(x,y)$

$\textbf{s.t.} \quad \psi_{ij} = \bar{K}, \qquad 1 \le i \le m, 1 \le j \le n$



- **Slack-based net weighting**
  - $net\ weight = \left(1 - \frac{slack}{T_{clk}}\right)^{\alpha}, \ \alpha > 1$
  - slack = 0 for the 1st iteration (pure wirelength-driven placement)

T. Chen *et al*. “Multilevel generalized force-directed method for circuit placement,” ISPD05.

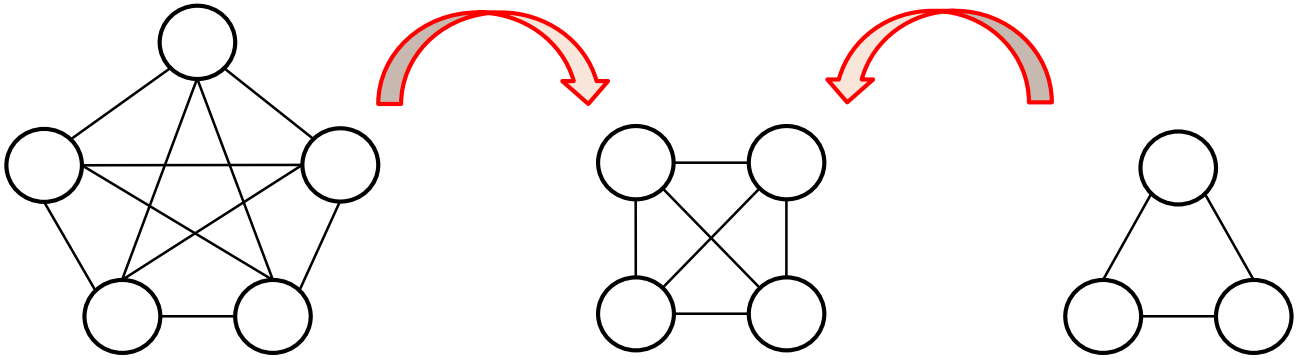
Flip-Flop Bonding

- **Bond flip-flops into perfect-sized cliques**
  - Perfect: most power efficient

- **Example:**

Bit number	Normalized power per bit	Normalized area per bit
1	1.00	1.00
2	0.86	0.96
4	0.78	0.71

Power efficient →



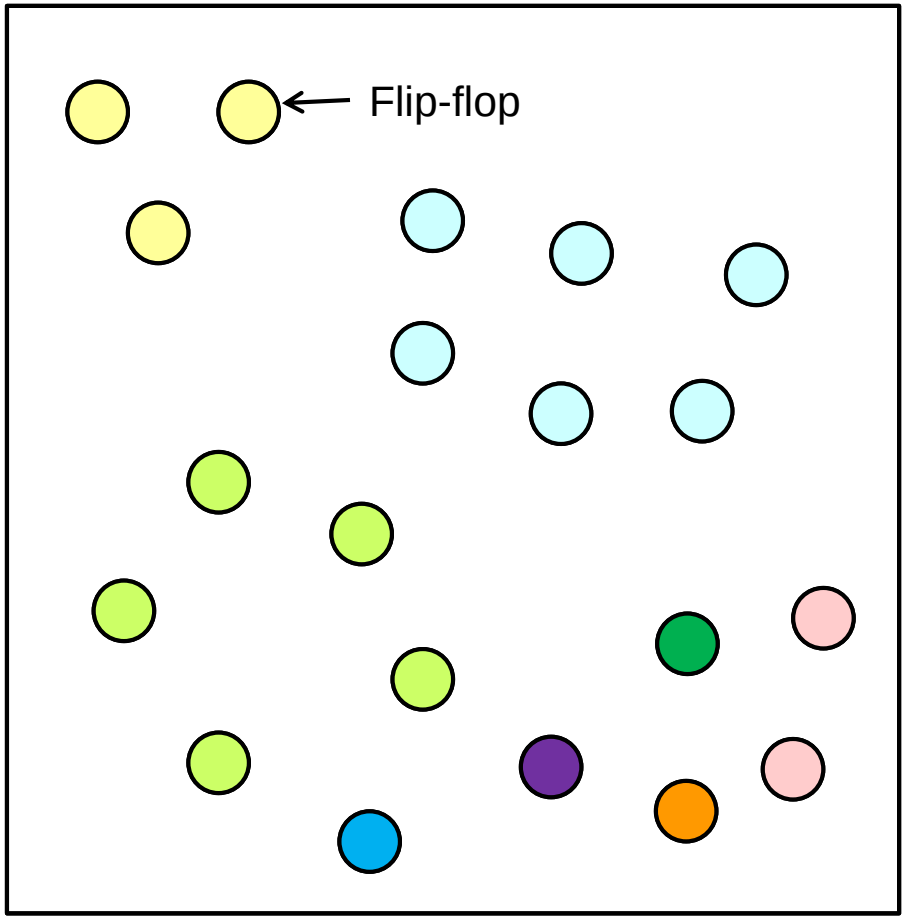
Flip-Flop Bonding

- **Bond flip-flops into perfect-sized cliques**

- **Priority of processing maximal cliques:**
  - Perfect > undersize > oversize
- **Perfect size: preserved**
- **Undersize/oversize: try to form a target-sized clique by selecting the nearest flip-flops in a specified search region**
  - The target size: the flip-flop configuration that is larger than, nearest to, and more power efficient than the investigated clique size

- **Adjacency inside the search region**
  - Physical & timing: $|x_c - x_i| + |y_c - y_i| - \varepsilon_i \times [s_{fi}(i) + s_{fo}(i)]$

Example: Flip-Flop Bonding (1/3)

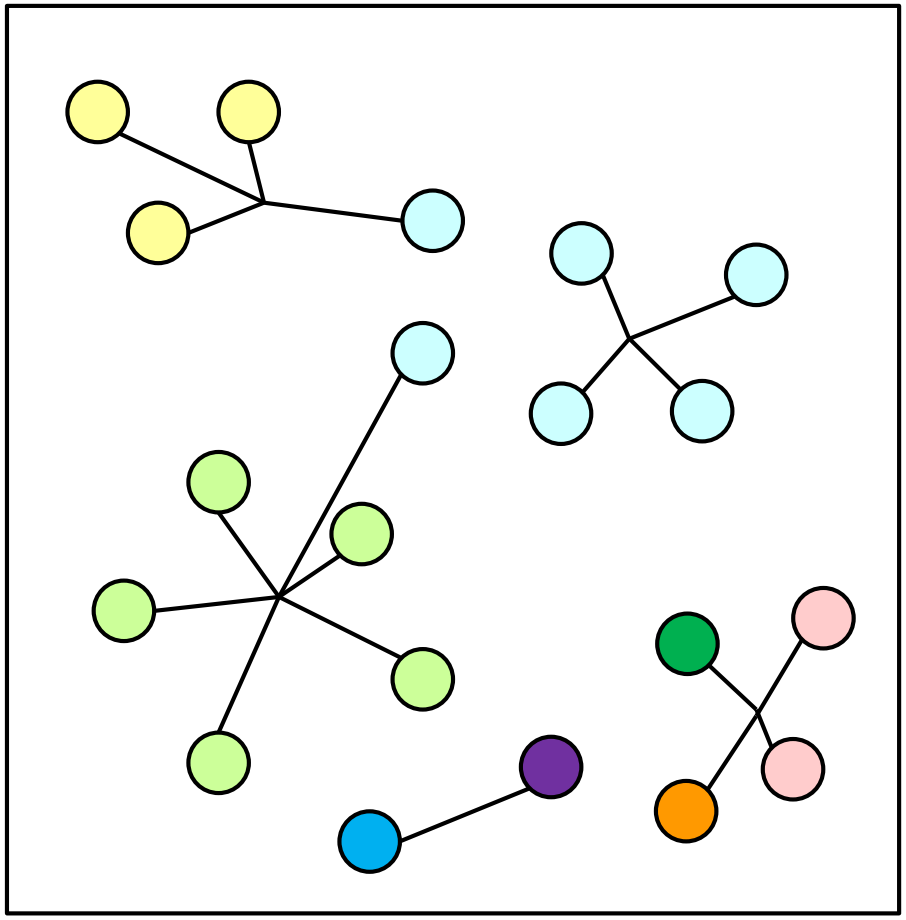


- **Extract maximal cliques**

Bit number	Normalized power per bit	Normalized area per bit
1	1.00	1.00
2	0.86	0.96
4	0.78	0.71

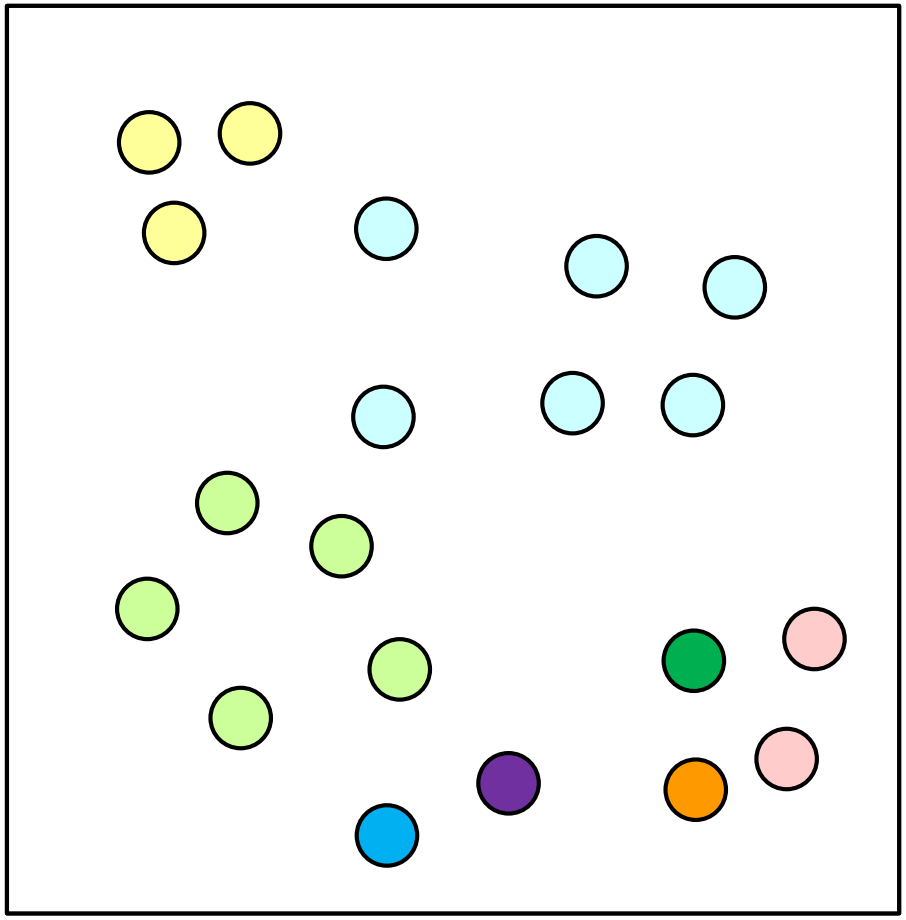
Power efficient

Example: Flip-Flop Bonding (2/3)



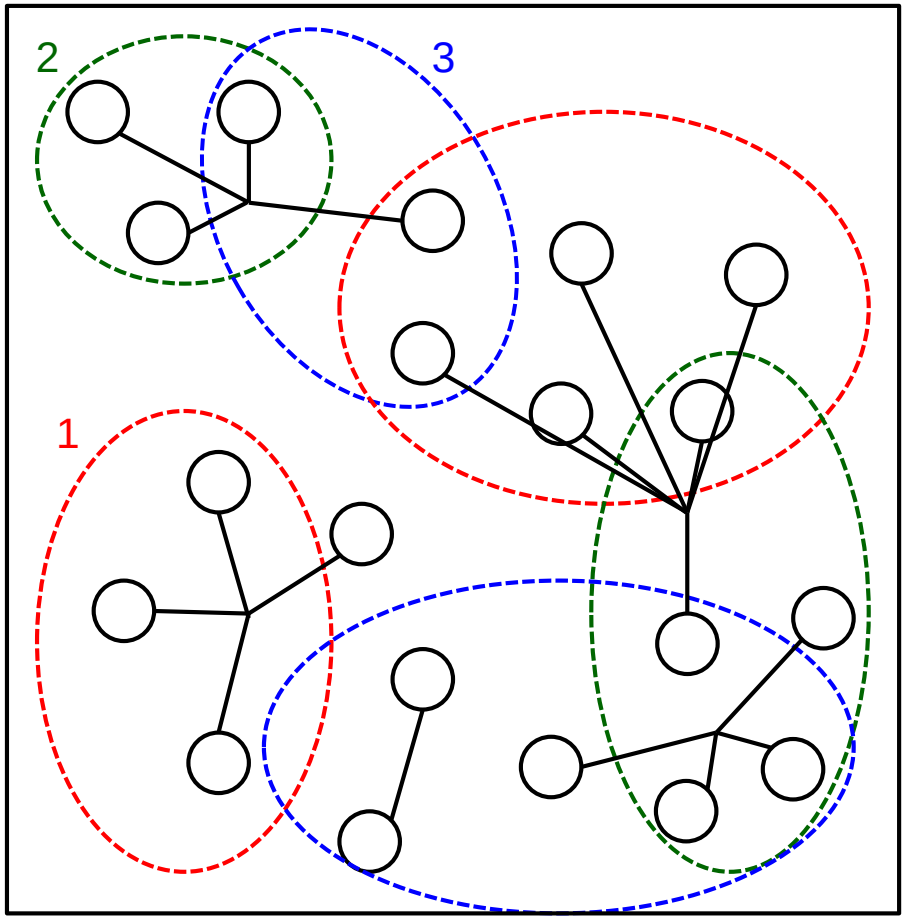
- **Bonding strategy**
  - Choose an undersized clique with priority 3>2>1
  - Select nearest flip-flops to form target-sized cliques
    - 3→4
    - 2→4
    - 1→2
  - Choose an oversized clique
  - Select nearest flip-flops to form target-sized cliques
    - Even→4X
    - Odd→even

Example: Flip-Flop Bonding (3/3)



- **Direct flip-flops to each other**
  - Set a flip-flop bonding anchor

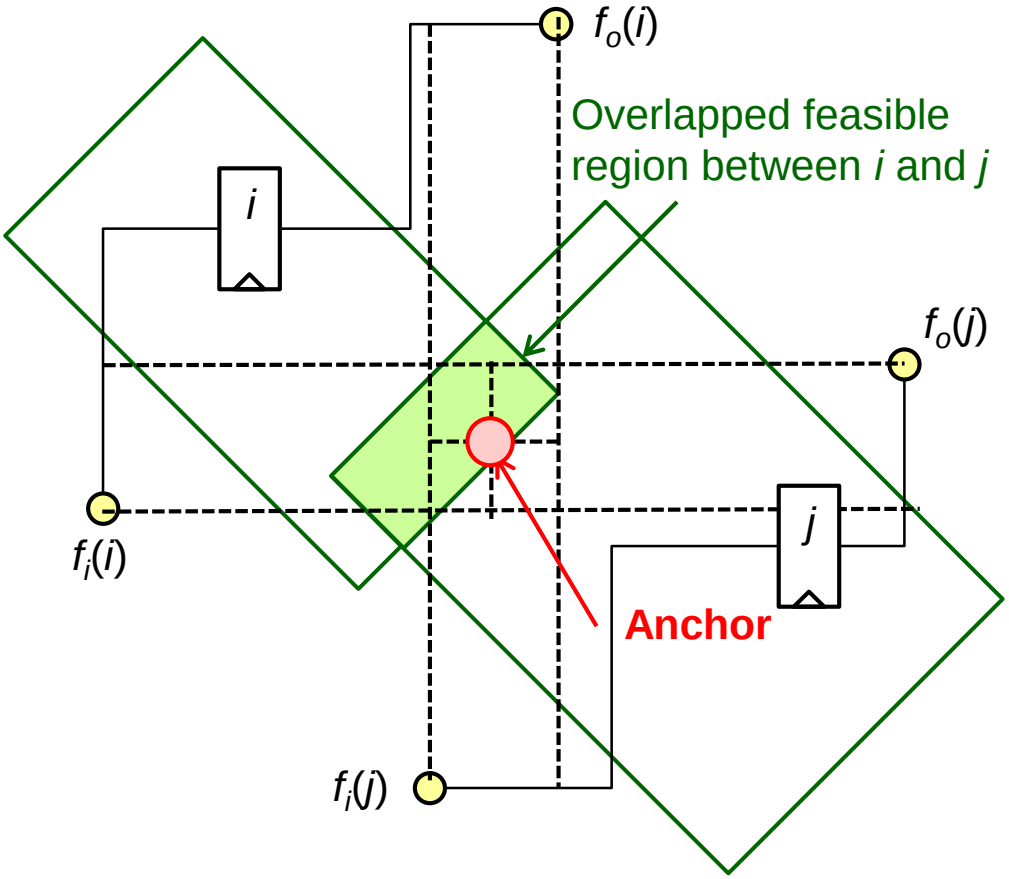
Overlapping Maximal Cliques



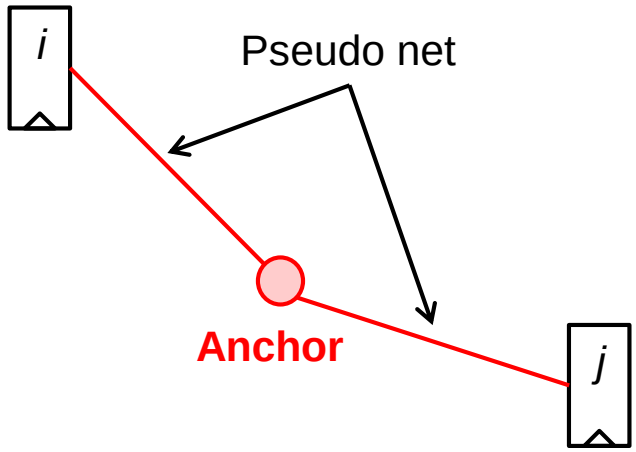
- **Bonding strategy**
  - Priority: undersized > oversized
- **Undersized**
  - 3 > 2 > 1
  - 3→4
  - 2→4
  - 1→2
- **Oversized**
  - Even→4X
  - Odd→even
- **Independency**
  - (#shared flip-flops)/cliquesize

Pseudo Net Generation

Set an anchor



Pseudo-net



High pseudo-net weight

Outline

- Introduction
- Preliminaries
- Problem formulation
- Algorithm - FF-Bond
- **Experimental results**
- Conclusion

Experimental Setting

- **Programming language: C++**
- **Platform: Intel Xeon 2.4GHz CPU, 16GB memory, Linux OS**

- **Technology: UMC 55nm**
- **Faraday MBFF library**

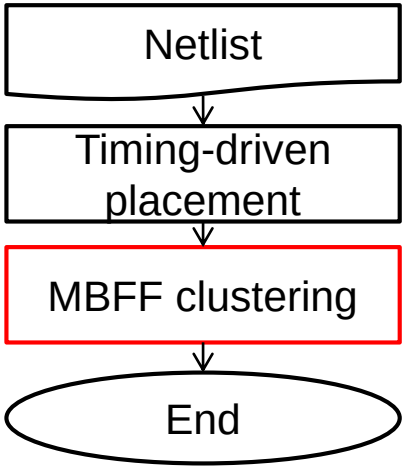
Bit number	Normalized power per bit	Normalized area per bit
1	1.00	1.00
2	0.86	0.96
4	0.78	0.71

- **Signoff timer: PrimeTime**

Comparison with Two Representative Flows

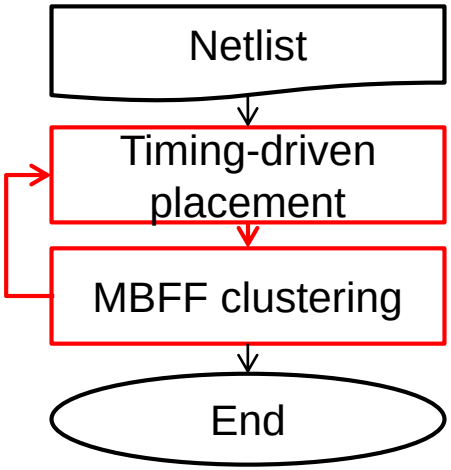
PMC

- **Post-placement MBFF clustering**
  - INTEGRA



IMC

- **Interleaving placement & post-placement MBFF clustering**
  - Timing-driven mPL + INTEGRA



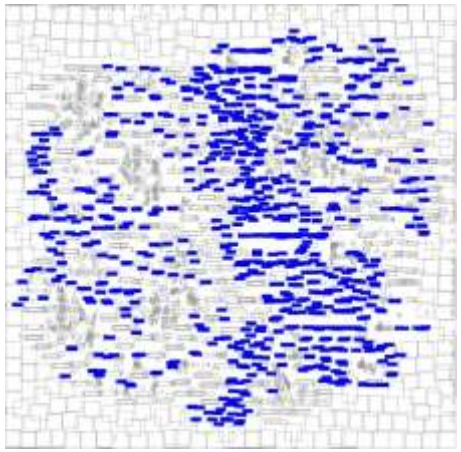
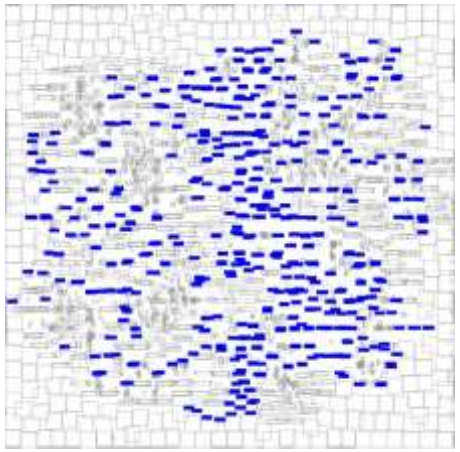
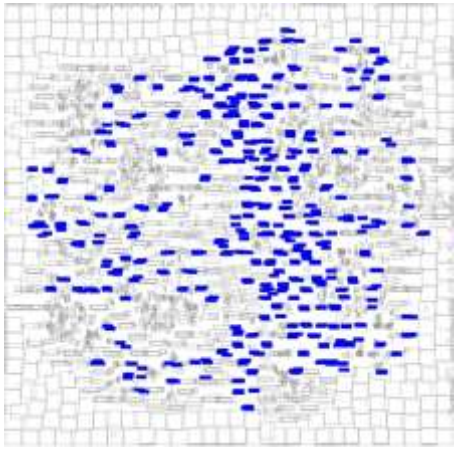
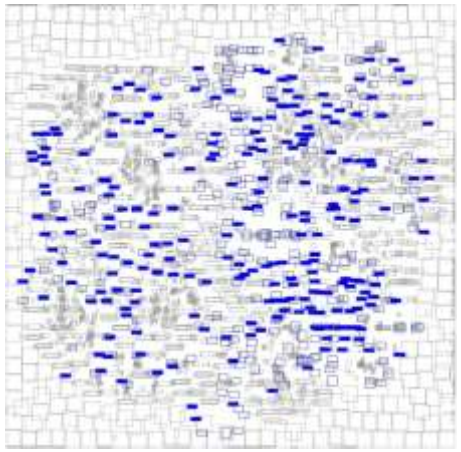
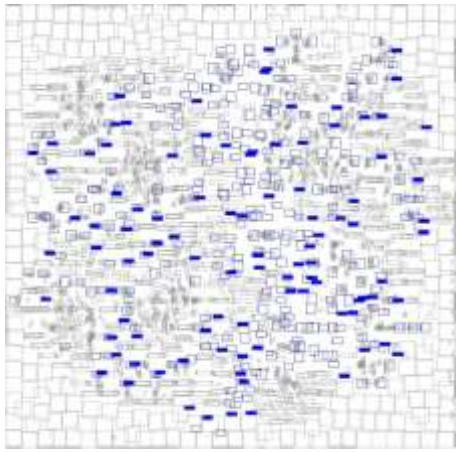
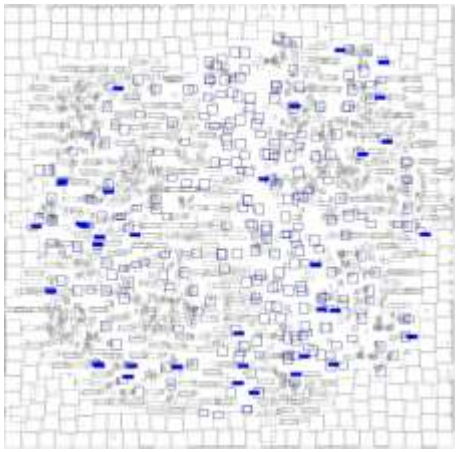
Flip-flop Power Comparison

- **Power ratio**
  - FF-Bond < IMC < PMC
  - Lower bound of power ratio: 0.78 (All are 4-bit MBFFs)
- **Main constituents of formed FFs**
  - FF-Bond: 4-bit FFs, PMC: 1-bit FFs, IMC: 2-bit FFs
- **Wirelength**
  - PMC < IMC < FF-Bond
  - Timing is satisfied, the induced power increment < 1%

		PMC			IMC			FF-Bond		
Circuit Name	**#Flip-flops**	**#MBFFs 4-/2-/1-bit**	**FF power ratio**	**HPWL**	**#MBFFs 4-/2-/1-bit**	**FF power ratio**	**HPWL**	**#MBFFs 4-/2-/1-bit**	**FF power ratio**	**HPWL**
s13207	212	8/57/66	0.892	4.569E+06	23/51/18	0.837	4.975E+06	35/31/10	0.814	5.344E+06
s15850	128	10/29/30	0.868	2.117E+06	18/23/10	0.826	2.869E+06	23/15/6	0.809	2.903E+06
s38417	881	35/252/237	0.885	4.599E+07	105/179/103	0.838	4.789E+07	159/105/35	0.808	5.406E+07
s38584	1069	46/291/303	0.886	5.992E+07	96/282/121	0.847	6.213E+07	203/116/25	0.803	6.926E+07
b17	1068	53/264/328	0.887	1.346E+08	137/201/118	0.834	1.363E+08	196/124/36	0.806	1.470E+08
b19	4384	378/886/1100	0.868	7.187E+08	593/742/528	0.834	7.267E+08	851/425/130	0.802	8.023E+08
Avg. ratio	**-**	**0.21/0.91/1.00**	**0.881**	**0.85**	**1.20/2.05/1.00**	**0.836**	**0.92**	**5.33/3.33/1.00**	**0.807**	**1.00**

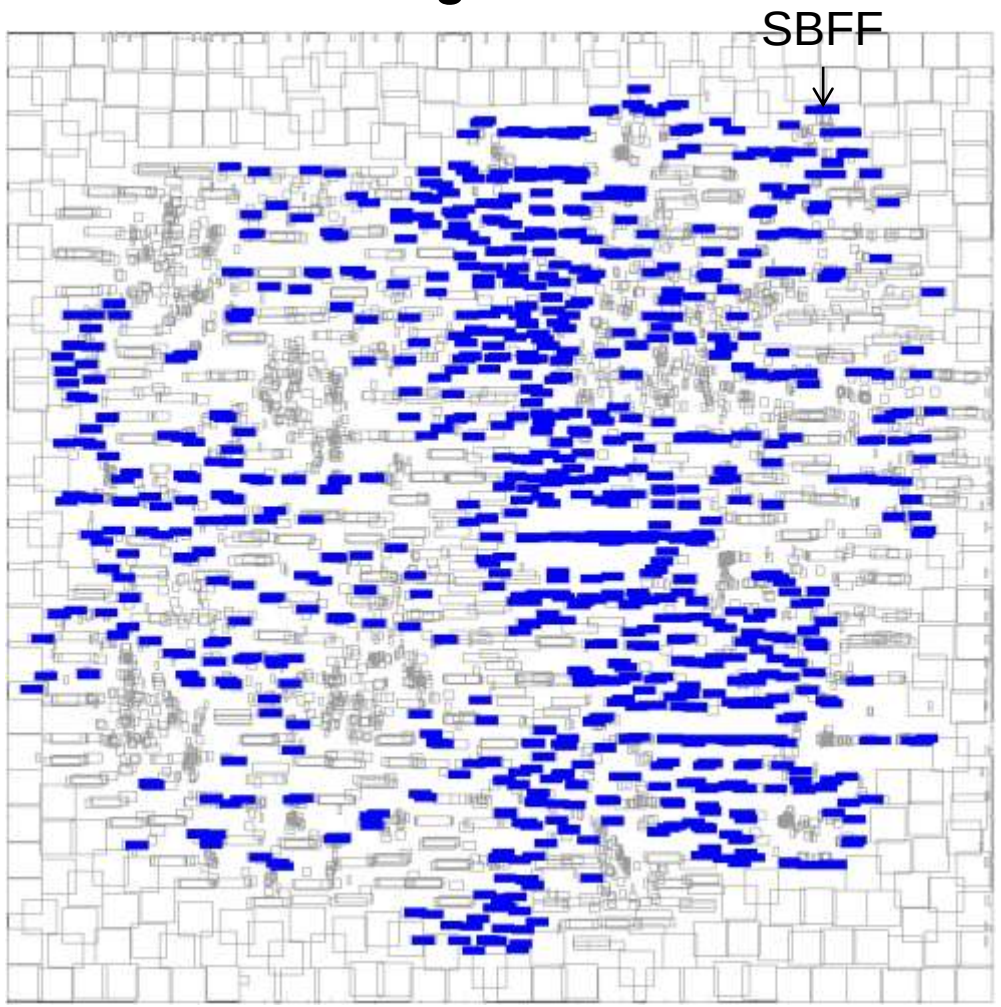
d_1 = 0.1, d_2 = 0.5, α = 1.2, the search region is bounded by 20% of chip dimension
the net weight of a pseudo net is 5X the default value for a positive slack net.

Experimental Results – s38417

s38417	PMC	IMC	FF-Bond
Global placement result (before MBFF merging)			
MBFF merging result (before legalization)			
#MBFFs (4-/2-/1-bit)	35/252/237	105/179/103	159/105/35

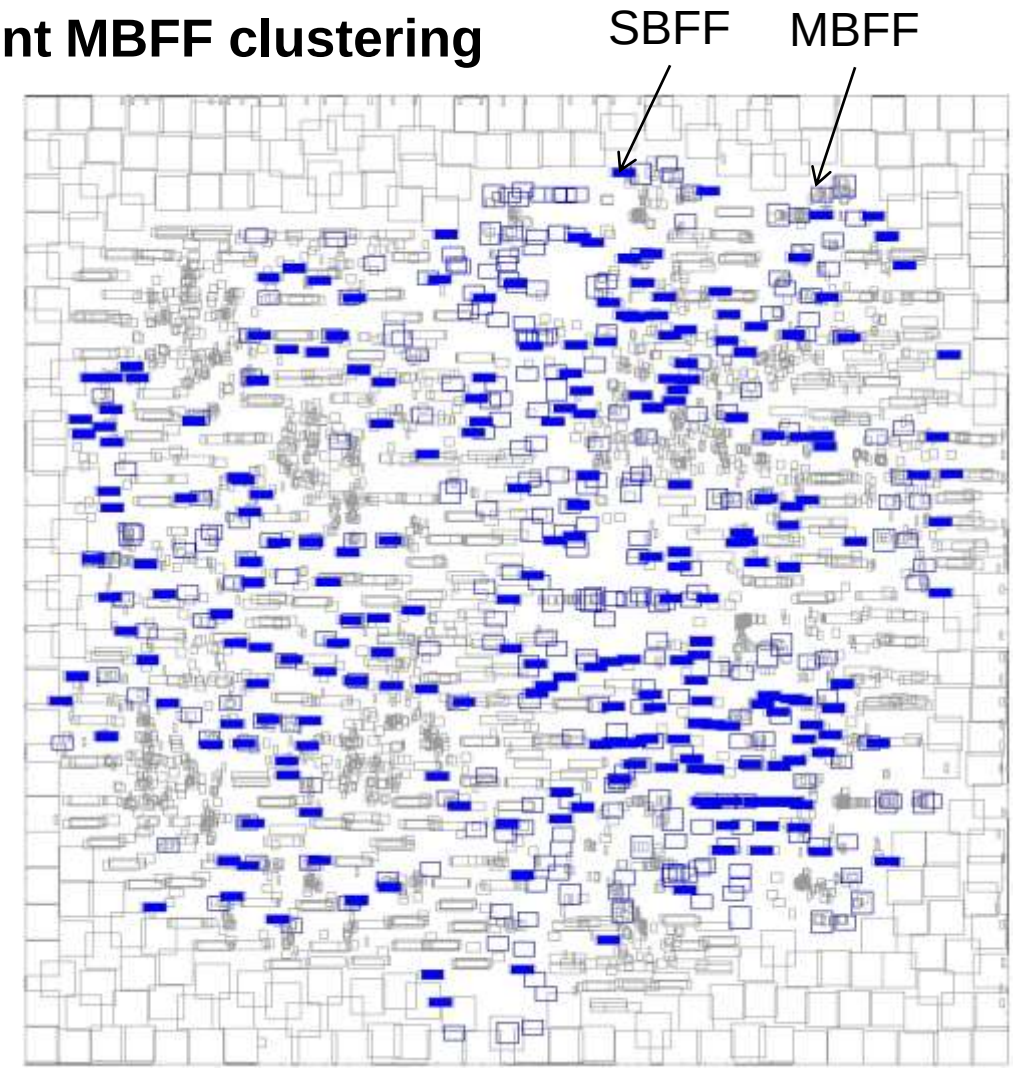
Experimental Results – s38417

- **PMC: Post-placement MBFF clustering**
  - Global placement



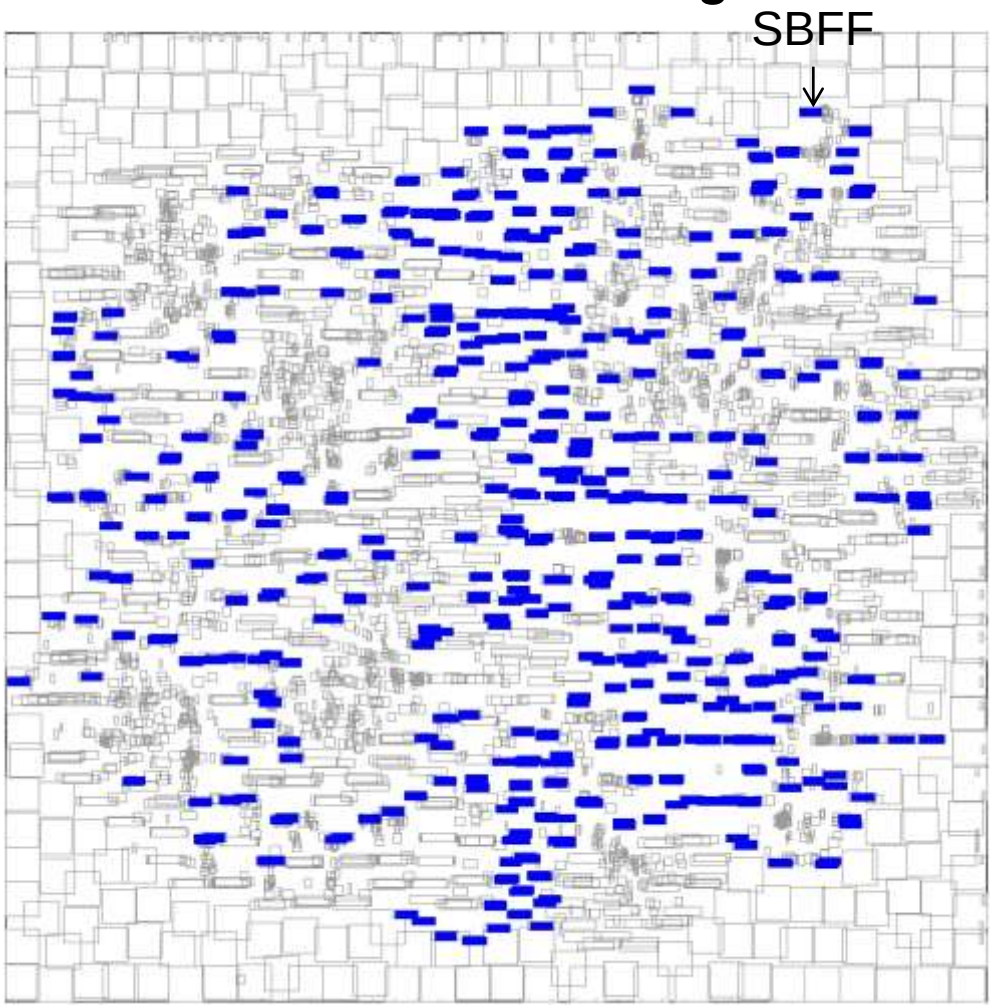
Experimental Results – s38417

- **PMC: Post-placement MBFF clustering**
  - MBFF merging
  - 4-/2-/1-bit MBFFs
    - 35/252/237



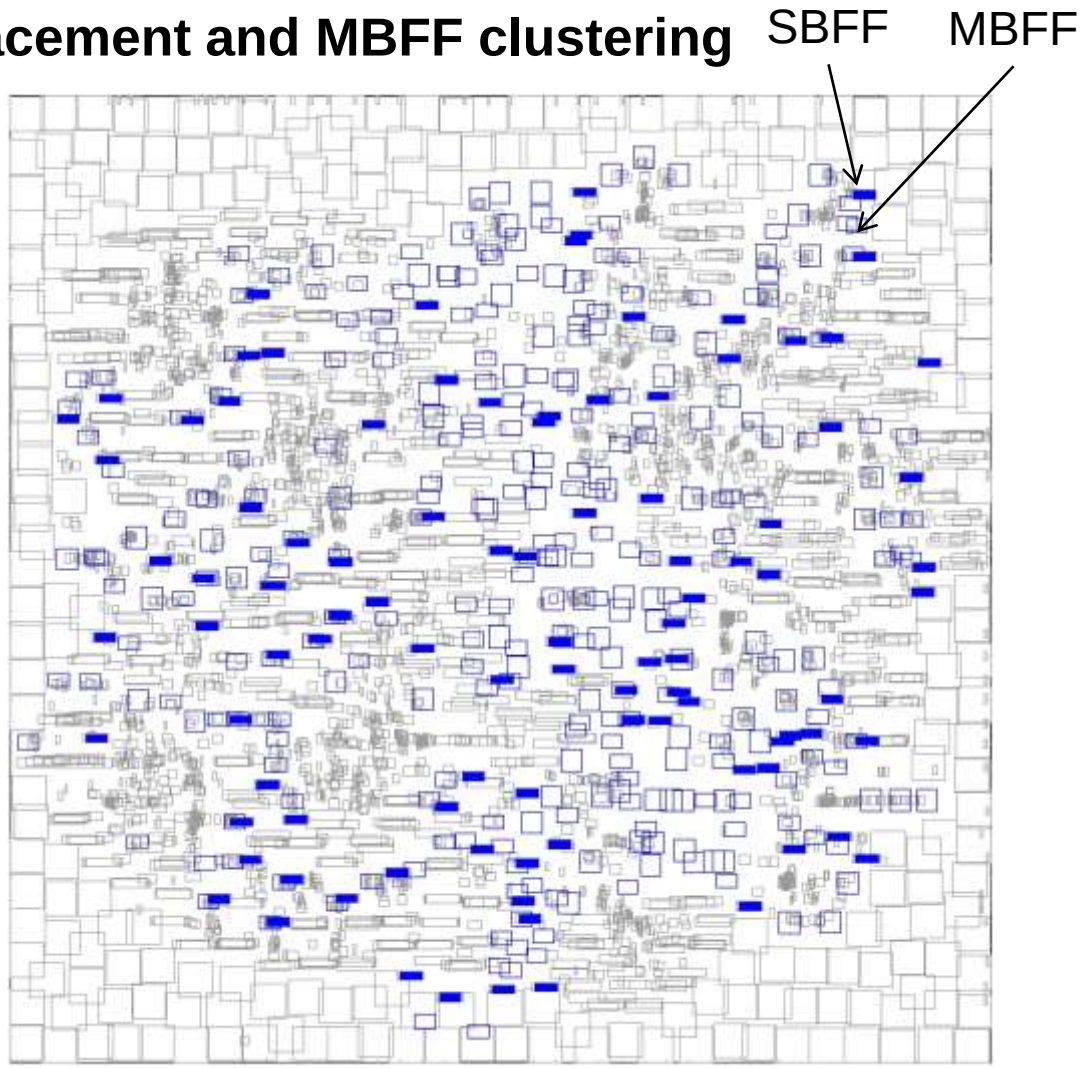
Experimental Results – s38417

- **IMC: Interleaving placement and MBFF clustering**
  - Global placement



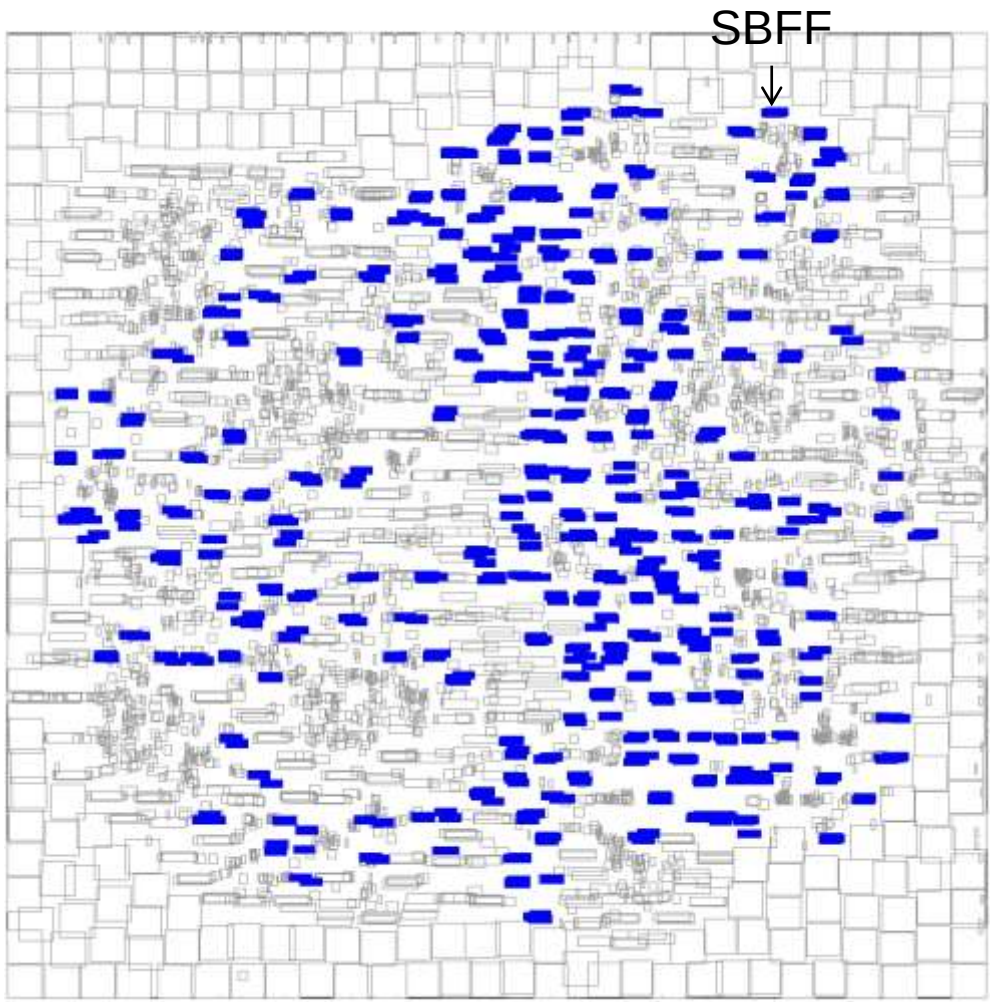
Experimental Results – s38417

- **IMC: Interleaving placement and MBFF clustering**
  - MBFF merging
  - 4-/2-/1-bit MBFFs
    - 105/179/103



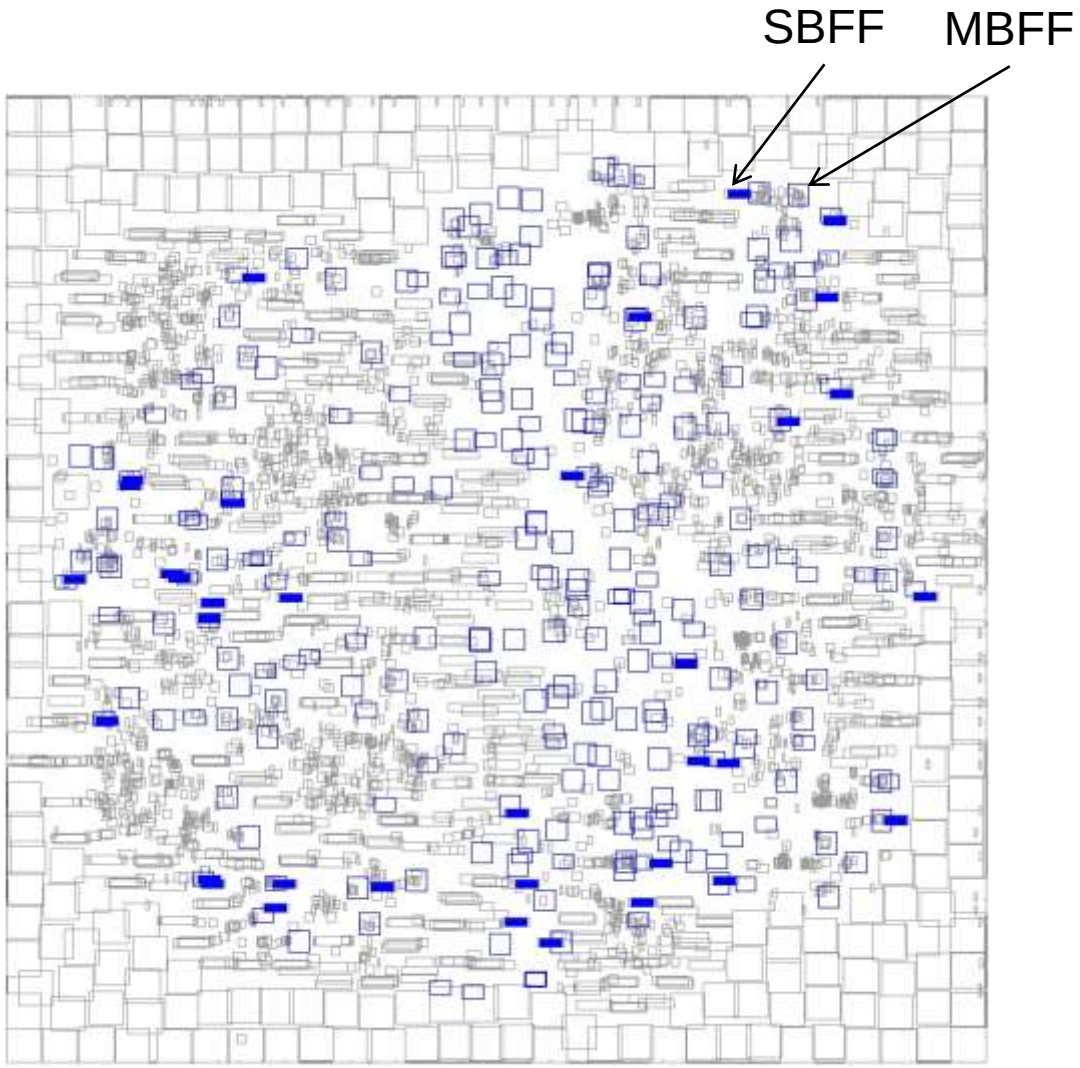
Experimental Results – s38417

- **FF-Bond**
  - Global placement



Experimental Results – s38417

- **FF-Bond**
  - MBFF merging
  - 4-/2-/1-bit MBFFs
    - 159/105/35



Clock Power Comparison

- **Clock power (flip-flops + clock network)**
  - FF-Bond < IMC < PMC < Without MBFF
  - FF-Bond totally saves 27% clock power on average
  - Compared with PMC, FF-Bond further reduces 14% clock power

	Without MBFF clustering				PMC				IMC				FF-Bond			
Circuit Name	**Total Cap. (pF)**	**Sinks**	**Buffer**	**Wire**	**Total Cap. (pF)**	**Sinks**	**Buffer**	**Wire**	**Total Cap. (pF)**	**Sinks**	**Buffer**	**Wire**	**Total Cap. (pF)**	**Sinks**	**Buffer**	**Wire**
s13207	1.333	48.5%	36.4%	15.1%	1.223	46.8%	39.7%	13.5%	1.094	49.6%	38.8%	11.7%	1.056	49.8%	40.2%	10.0%
s15850	0.901	43.3%	47.1%	9.6%	0.837	40.9%	50.6%	8.5%	0.806	39.6%	52.6%	7.8%	0.799	39.5%	53.1%	7.4%
s38417	5.051	53.2%	31.6%	15.2%	4.113	57.9%	26.8%	15.3%	3.884	58.2%	27.4%	14.5%	3.711	58.5%	28.6%	12.9%
s38584	6.100	53.5%	28.9%	17.6%	5.352	54.3%	29.9%	15.8%	4.576	60.6%	24.1%	15.3%	4.870	53.9%	32.8%	13.3%
b17	6.273	51.9%	28.1%	20.0%	5.574	51.8%	28.7%	19.5%	5.241	51.9%	30.5%	17.6%	4.513	58.2%	26.3%	15.5%
b19	25.611	52.2%	26.8%	21.0%	22.081	52.4%	28.1%	19.5%	19.410	57.4%	23.9%	18.7%	18.277	58.7%	24.5%	16.8%
Avg. Ratio	**1.00**	-	-	-	**0.87**	-	-	-	**0.77**	-	-	-	**0.73**	-	-	-

Slack Comparison: PMC vs. FF-Bond

- **The slacks of the two flows are quite similar**

		PMC		FF-Bond	
Circuit Name	**Clock period (ns)**	**Worst slack (ns)**	**Average slack (ns)**	**Worst slack (ns)**	**Average slack (ns)**
s13207	1.5	0.042	0.580	0.041	0.579
s15850	1.8	0.158	0.336	0.154	0.334
s38417	2.0	0.164	1.049	0.163	1.047
s38584	2.0	0.217	0.871	0.209	0.869
b17	3.0	0.122	1.272	0.128	1.269
b19	2.7	0.112	1.278	0.109	1.273

‘Worst slack’ represents the worst timing slack over all endpoints
‘Average slack’ means the average

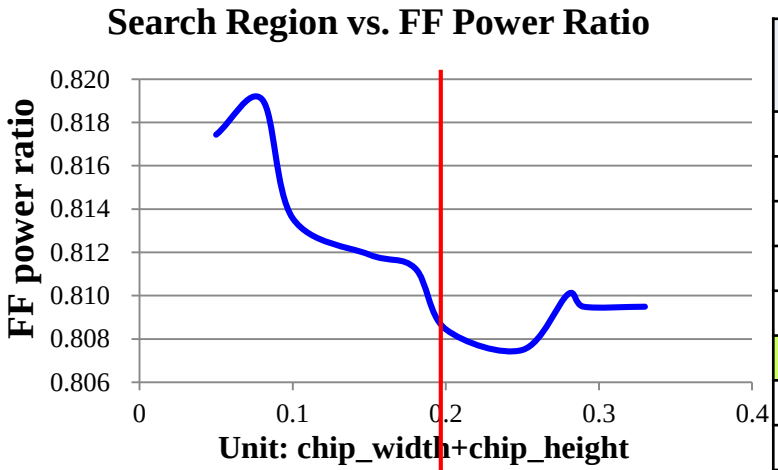
FF-Bond: d_2 Analysis

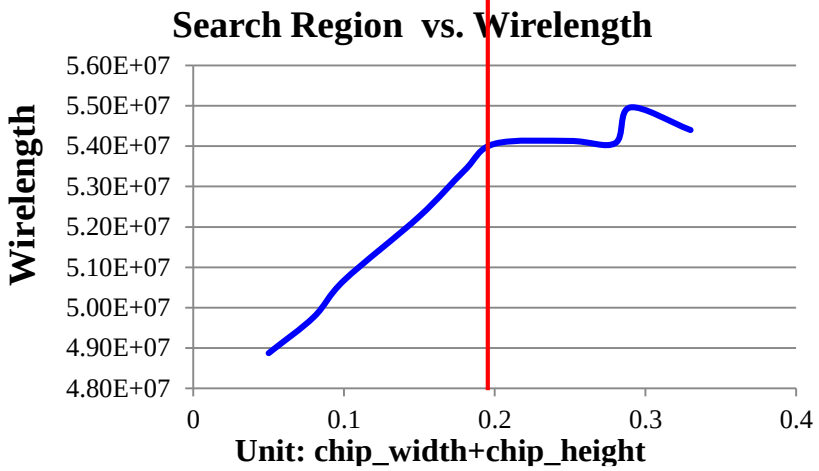
- **d_2: the sparsity criterion to apply flip-flop bonding**
  - Smaller d_2 : flip-flop bonding starts at later iterations but does not guide FFs well because of the low flexibility of moving FFs
    - Power↑, WL↓
  - Larger d_2 : flip-flop bonding starts at earlier iterations but incurs longer wirelength because distant flip-flops are attracted
    - Power↑(slightly), WL↑

		d_2=0.3			d_2=0.5			d_2=0.7		
Circuit Name	**#Flip-flops**	**#MBFFs 4-/2-/1-bit**	**FF power ratio**	**HPWL**	**#MBFFs 4-/2-/1-bit**	**FF power ratio**	**HPWL**	**#MBFFs 4-/2-/1-bit**	**FF power ratio**	**HPWL**
s13207	212	27/41/22	0.834	5.243E+06	35/31/10	0.814	5.344E+06	38/25/10	0.809	5.518E+06
s15850	128	20/20/8	0.819	2.620E+06	23/15/6	0.809	2.903E+06	22/16/8	0.814	2.895E+06
s38417	881	171/85/27	0.802	5.258E+07	159/105/35	0.808	5.406E+07	164/89/47	0.808	5.569E+07
s38584	1069	194/135/23	0.805	6.861E+07	203/116/25	0.803	6.926E+07	192/134/33	0.807	6.944E+07
b17	1068	186/135/23	0.809	1.460E+08	196/124/36	0.806	1.470E+08	202/116/28	0.803	1.485E+08
b19	4384	847/427/142	0.803	7.927E+08	851/425/130	0.802	8.023E+08	851/431/118	0.802	8.037E+08
Avg. ratio	**-**	**4.99/3.44/1.00**	**0.812**	**0.97**	**5.33/3.33/1.00**	**0.807**	**1.00**	**5.04/3.04/1.00**	**0.807**	**1.01**

Search Region Analysis

- **Search region ↑, power ratio ↓, wirelength ↑**





Search region (Unit: chip_width+chip_height)	FF power ratio	HPWL	Worst slack (ns)
0.05	0.817	4.89E+07	0.163
0.08	0.819	4.98E+07	0.163
0.10	0.814	5.07E+07	0.164
0.15	0.812	5.23E+07	0.163
0.18	0.811	5.34E+07	0.164
0.20	0.808	5.41E+07	0.163
0.25	0.807	5.41E+07	0.163
0.28	0.810	5.41E+07	0.163

s38417

Outline

- Introduction
- Preliminaries
- Problem formulation
- Algorithm - FF-Bond
- Experimental results
- **Conclusion**

Conclusions

- **MBFFs reduce clock load effectively**
  - Prior work: post-placement MBFF clustering
- **Proposed MBFF bonding at-placement – FF-Bond**
  - Directed flip-flops towards merging friendly locations
  - FF-Bond can save 27% clock power on average
  - Compared with post-placement MBFF clustering, FF-Bond can further reduce 14% clock power
- **Future work: MBFF bonding with routability consideration**

Thank You!

Contact info:
Iris Hui-Ru Jiang
huiru.jiang@gmail.com