Clustering of Flip-Flops
for Useful-Skew Clock Tree Synthesis

|		
Chuan Yean Tan	•	Purdue University
Rickard Ewetz	•	University of Central Florida
Cheng-Kok Koh	•	Purdue University

PURDUE
UNIVERSITY

Outline

- MBFF Introduction
- Previous MBFF/Clustering Solutions
- Bounded Arrival Time Useful Skew Tree
- Clustering of Flip-Flops Mechanics
- Proposed Methodology
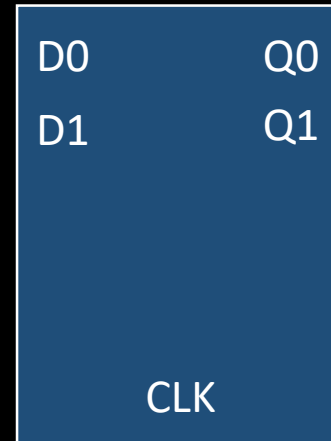- Evaluation & Experimental Results
- Summary

Outline

- **MBFF Introduction**
- Previous MBFF/Clustering Solutions
- Bounded Arrival Time Useful Skew Tree
- Clustering of Flip-Flops Mechanics
- Proposed Methodology
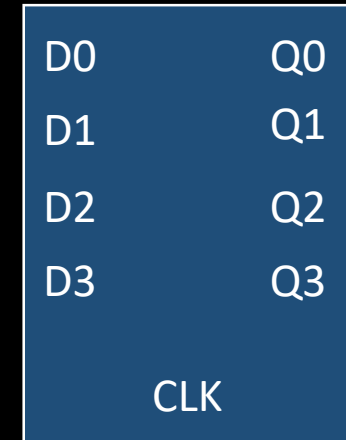- Evaluation & Experimental Results
- Summary

Multi-Bit Flip Flop (MBFF)

D0 Q0		D0 Q0		D0 Q0
		D1 Q1		D1 Q1
				D2 Q2
				D3 Q3
CLK		CLK		CLK
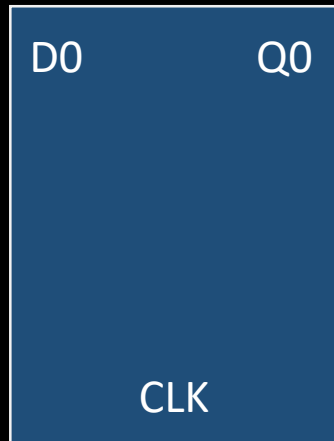
1-Bit Flip-Flop   2-Bit Flip-Flop   4-Bit Flip-Flop

No. of Bits	1-Bit FF	2-Bit FF	4-Bit FF
Power Per Bit	1.00	0.86	0.78
Area Per Bit	1.00	0.96	0.71

Benefits of utilizing MBFF

- Reduce **power consumption**
- Reduce **routing complexity** and **clock tree complexity**, less elements to connect leads to less routing complexity

No. of Bits	1-Bit FF	2-Bit FF	4-Bit FF
Power Per Bit	1.00	0.86	0.78
Area Per Bit	1.00	0.96	0.71

MBFF Power And Area Savings[3]

[3] L.Chen et al. Using MBFF for clock power saving by design compiler. – *Proceedings of Synopsys User Group, 2010*

Outline

- MBFF Introduction
- **Previous MBFF/Clustering Solutions**
- Bounded Arrival Time Useful Skew Tree
- Clustering of Flip-Flops Mechanics
- Proposed Methodology
- Evaluation & Experimental Results
- Summary

Previous Works Summary

Key Methodologies of Previous Works:
- Timing-Driven Intersecting of Feasible Regions[14]
- Maximum Independent Set based on heuristic approach [2]
- x- and y- interval graphs to determine maximum clique [9]
- K-means Algorithm [15,10]

[2] Y.T.Chang et al. Post-placement power optimization with multi-bit flip-flops - ICCAD 2010
[9] I.H.R. Jiang et al. INTEGRA: Fast multi-bit flip-flop cluster for clock power saving. - IEEE Transactions CAD of I.C.S. 2015
[10] A.B.Khang et al. Improve flop tray-based design implementation for power reduction. - ICCAD 2016
[14] S.H.Wang et al. Power-driver flip-flop merging and relocation. - IEEE Transactions CAD of I.C.S 2012
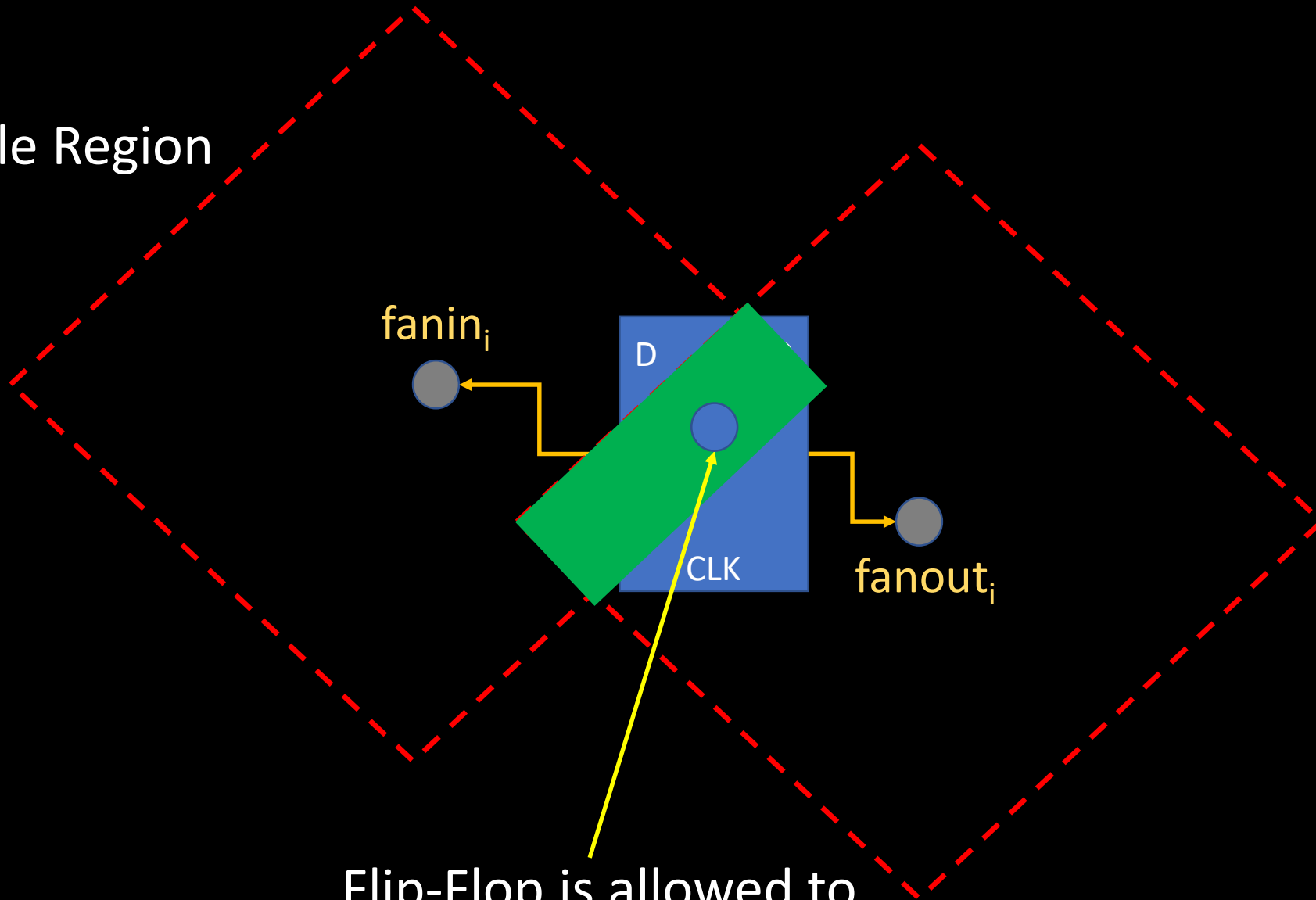[15] G.Wu et al. Flip-Flop clustering by weighted k-means algorithm. - DAC 2016

Feasible Region

Fanout's **slack**

fanin$_i$

D   Q

CLK

fanout$_i$
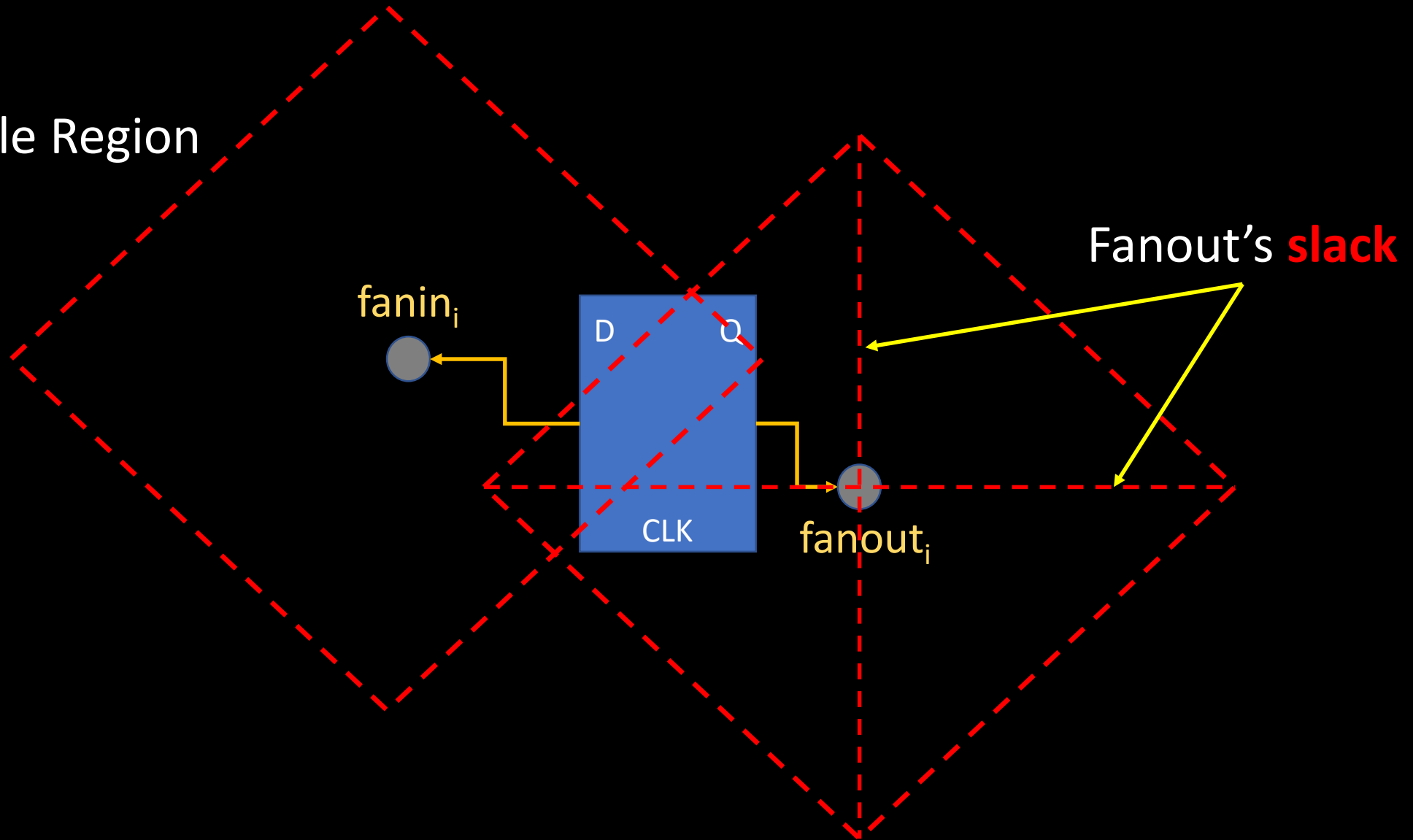
Timing-Driven Clustering of Flip-Flops

8

Feasible Region

fanin$_i$

D

CLK

fanout$_i$

Flip-Flop is allowed to moved with the **region**

9

Analysis of Previous Works

- Previous work cluster flip-flop without consideration of clock tree synthesis (CTS)
- Timing slacks are obtained pre-CTS, **timing does not correlate** as accurately after CTS
- Most modern designs adopt useful-skew tree to reduce area and power consumption
- However, once the UST has been built, updating or displacing FF's placement is **expensive**!

Outline

- MBFF Introduction
- Previous MBFF/Clustering Solutions
- **Bounded Arrival Time Useful Skew Tree**
- Clustering of Flip-Flops Mechanics
- Proposed Methodology
- Evaluation & Experimental Results
- Summary
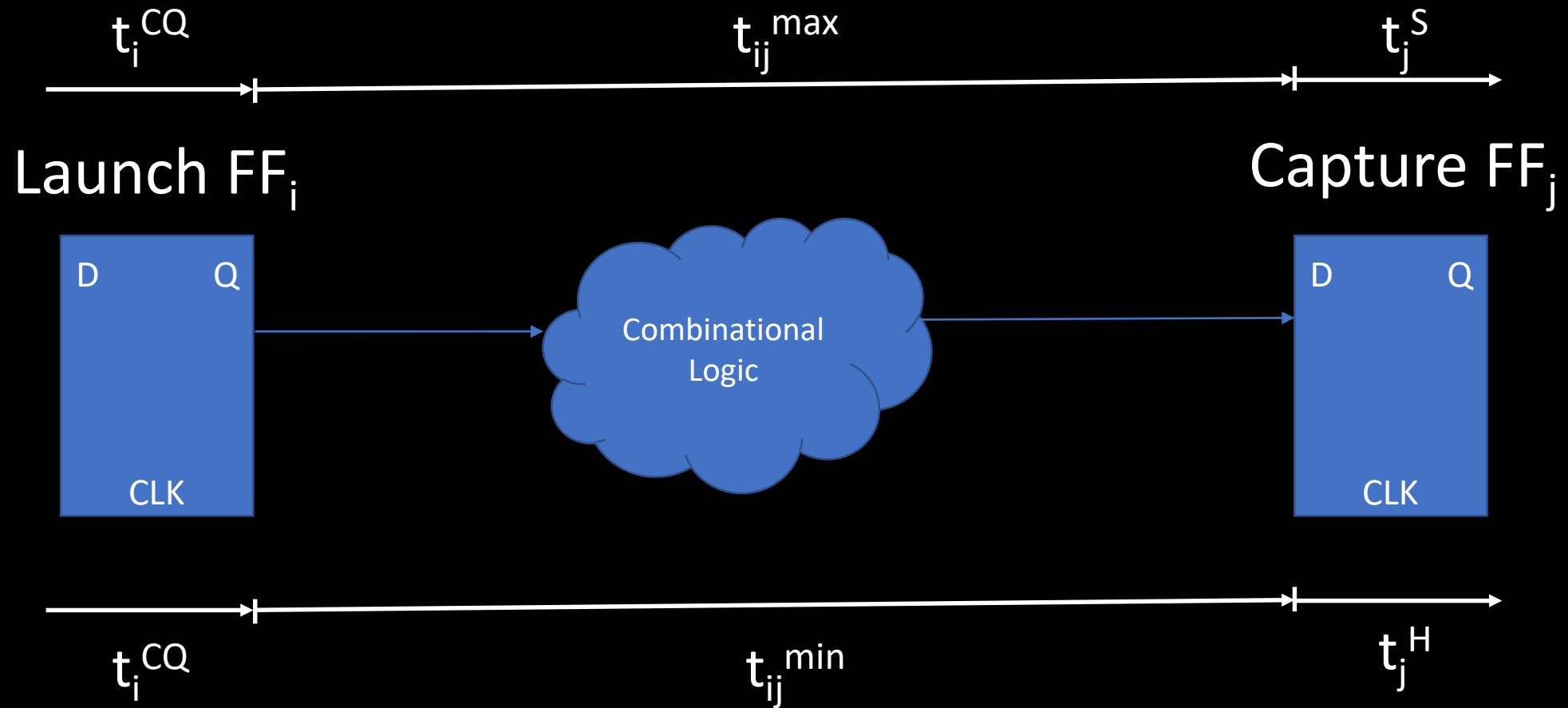
Setup Constraints: $t_i + t_i^{CQ} + t_{ij}^{max} + t_j^S \leq t_j + T$

Hold Constraints: $t_i + t_i^{CQ} + t_{ij}^{min} \geq t_j + t_j^H$

Setup Constraints: $t_i + t_i^{CQ} + t_{ij}^{max} + t_j^S \leq t_j + T$

Hold Constraints: $t_i + t_i^{CQ} + t_{ij}^{min} \geq t_j + t_j^H$

Constraints Formulation:

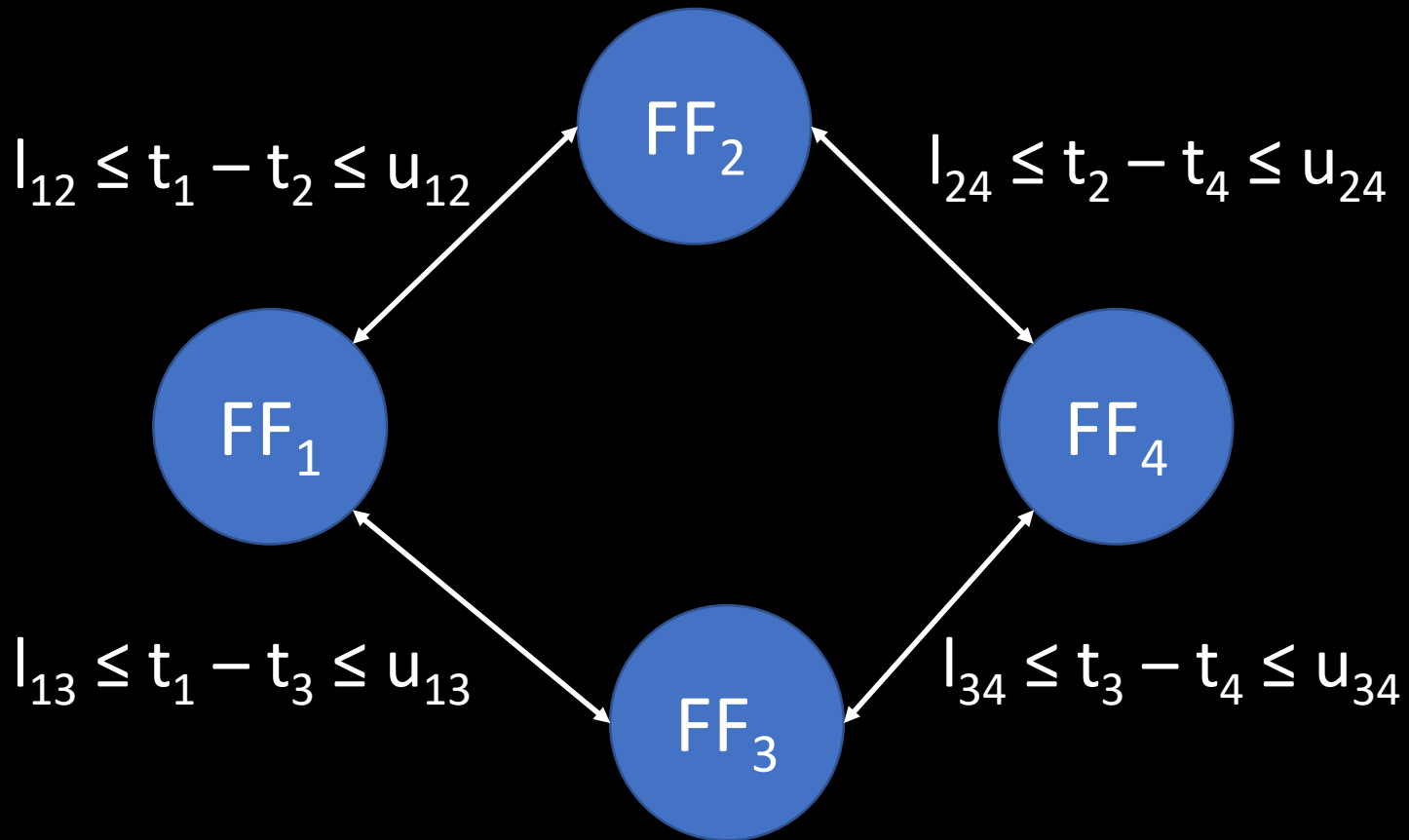$$u_{ij} = T - t_i^{CQ} - t_{ij}^{max} - t_j^S$$

$$l_{ij} = t_j^H - t_i^{CQ} - t_{ij}^{min}$$

Skew Constraints:

$$l_{ij} \leq t_i - t_j \leq u_{ij}$$

$$l_{ij} \leq skew_{ij} \leq u_{ij}$$

Skew Constraint Graph (SCG)

$l_{12} \leq t_1 - t_2 \leq u_{12}$

$l_{24} \leq t_2 - t_4 \leq u_{24}$

$l_{13} \leq t_1 - t_3 \leq u_{13}$

$l_{34} \leq t_3 - t_4 \leq u_{34}$

Skew Constraint Graph

Weighted Edges:

$w_{12} = u_{12}$
$w_{21} = -l_{12}$

From the skew constraints:
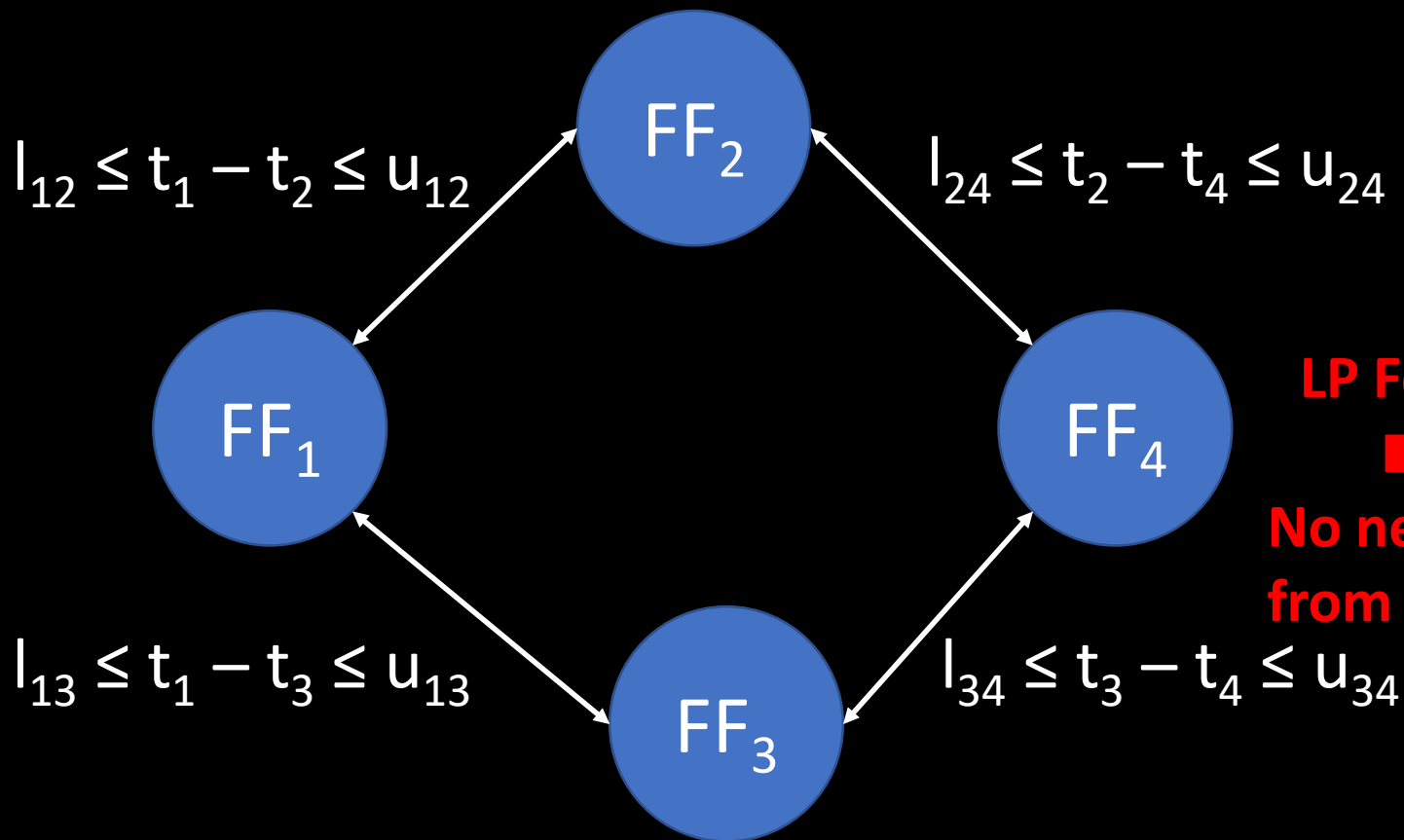
$t_1 - t_2 \leq u_{12}$
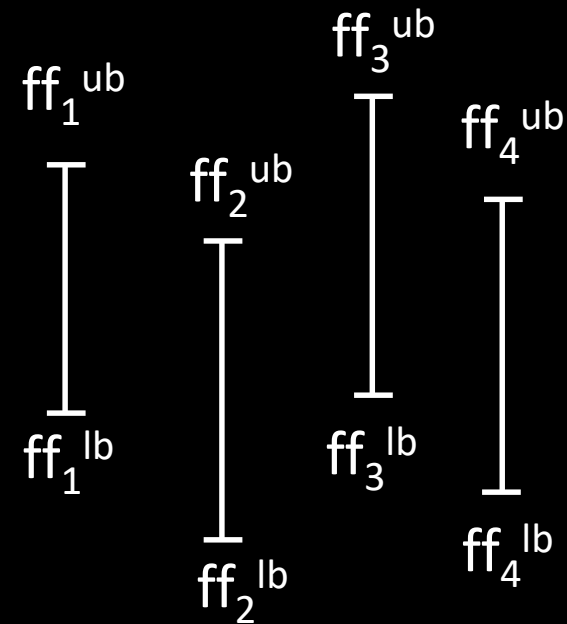$t_2 - t_1 \leq -l_{12}$

Generalize weighted constraint:

$t_i - t_j \leq w_{ij}$
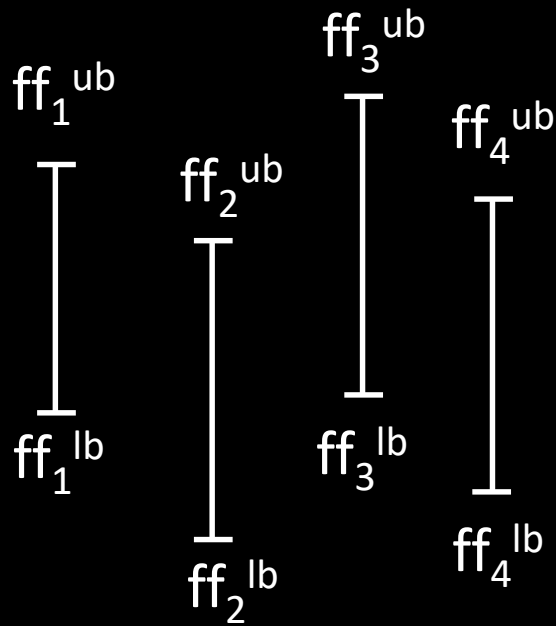
Useful-Skew Tree based on Arrival Time Constraints [5]

$l_{12} \leq t_1 - t_2 \leq u_{12}$

$l_{24} \leq t_2 - t_4 \leq u_{24}$

$l_{13} \leq t_1 - t_3 \leq u_{13}$

$l_{34} \leq t_3 - t_4 \leq u_{34}$

LP Formulation

No negative cycle from SCG

Skew Constraint Graph (SCG)

Arrival Time Ranges

[5] R. Ewetz et al. Clock Tree Construction based on arrival time constraints – ACM 2017

Useful-Skew Tree based on Arrival Time Constraints [5]

Arrival Time Ranges

Clock Tree Construction
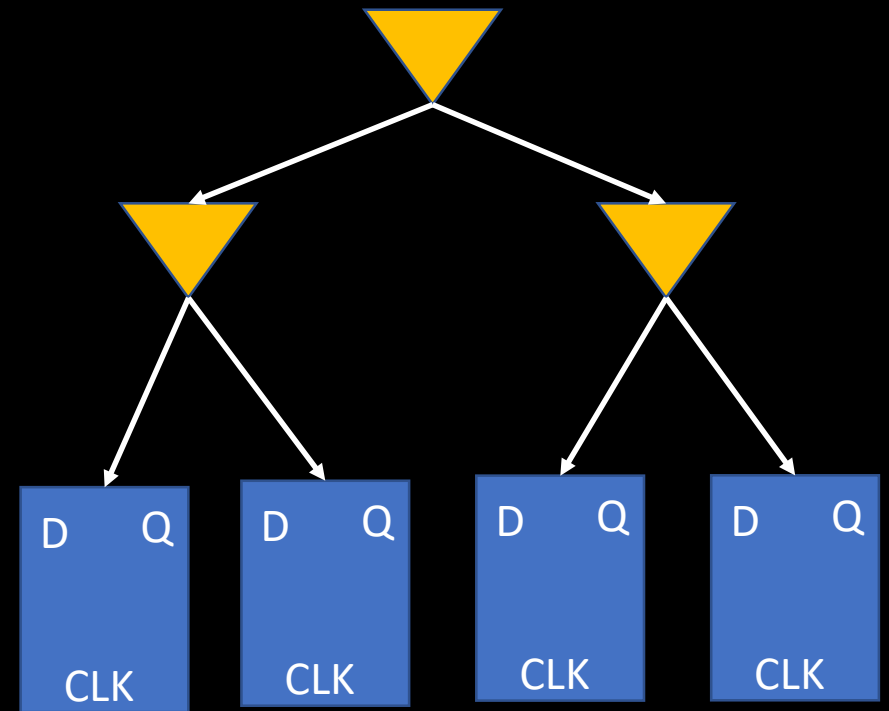
[5] R. Ewetz et al. Clock Tree Construction based on arrival time constraints – ACM 2017
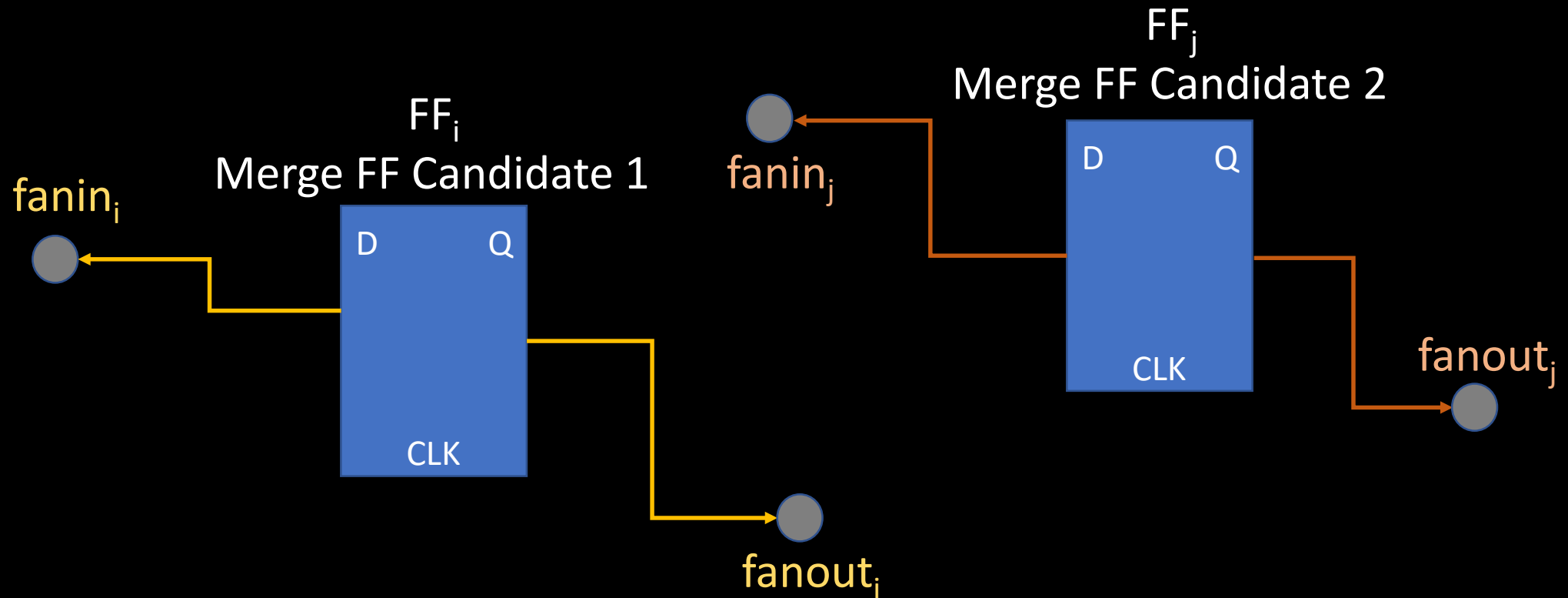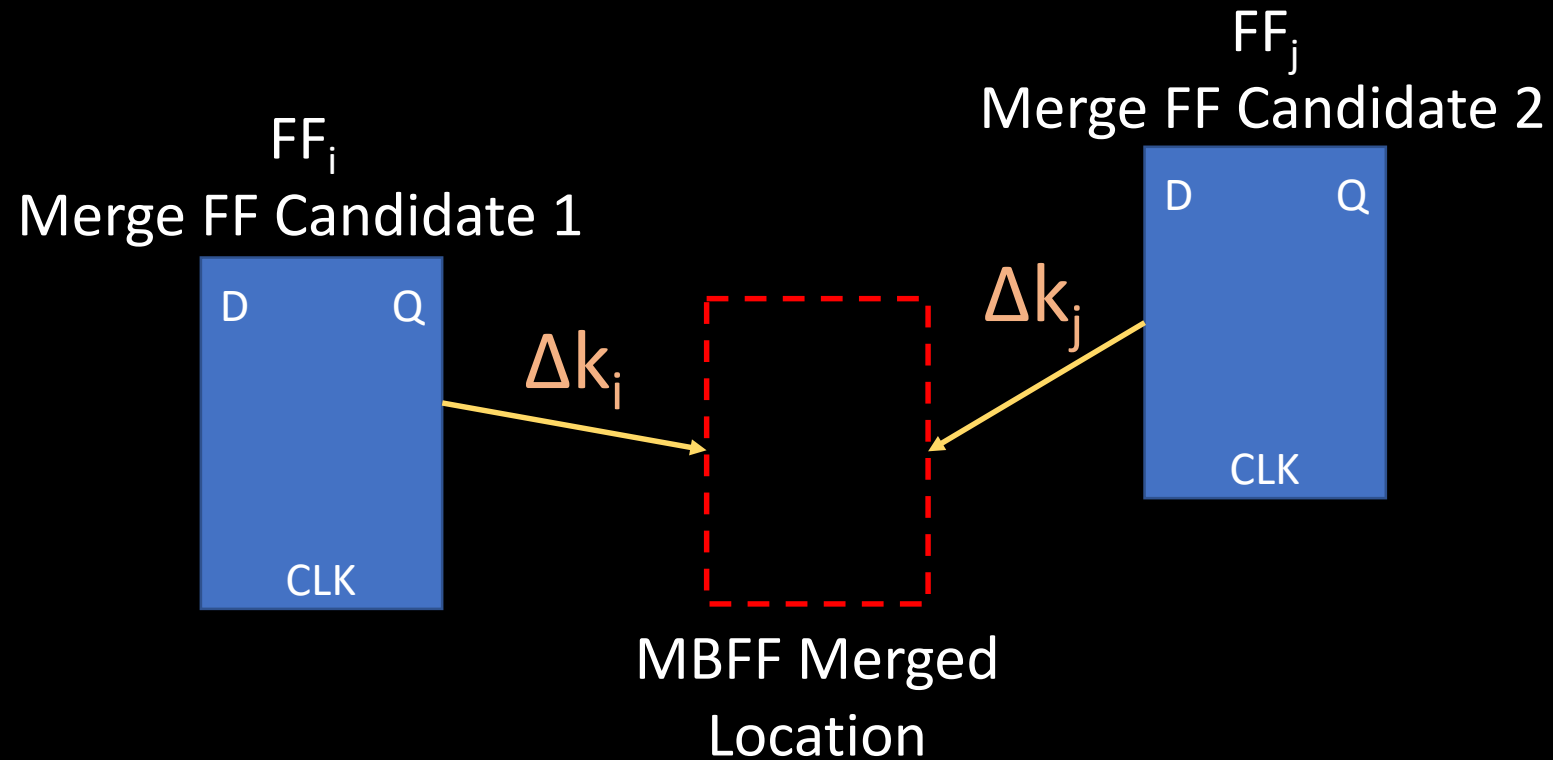
Outline

- MBFF Introduction
- Previous MBFF/Clustering Solutions
- Bounded Arrival Time Useful Skew Tree
- **Clustering of Flip-Flops Mechanics**
- Proposed Methodology
- Evaluation & Experimental Results
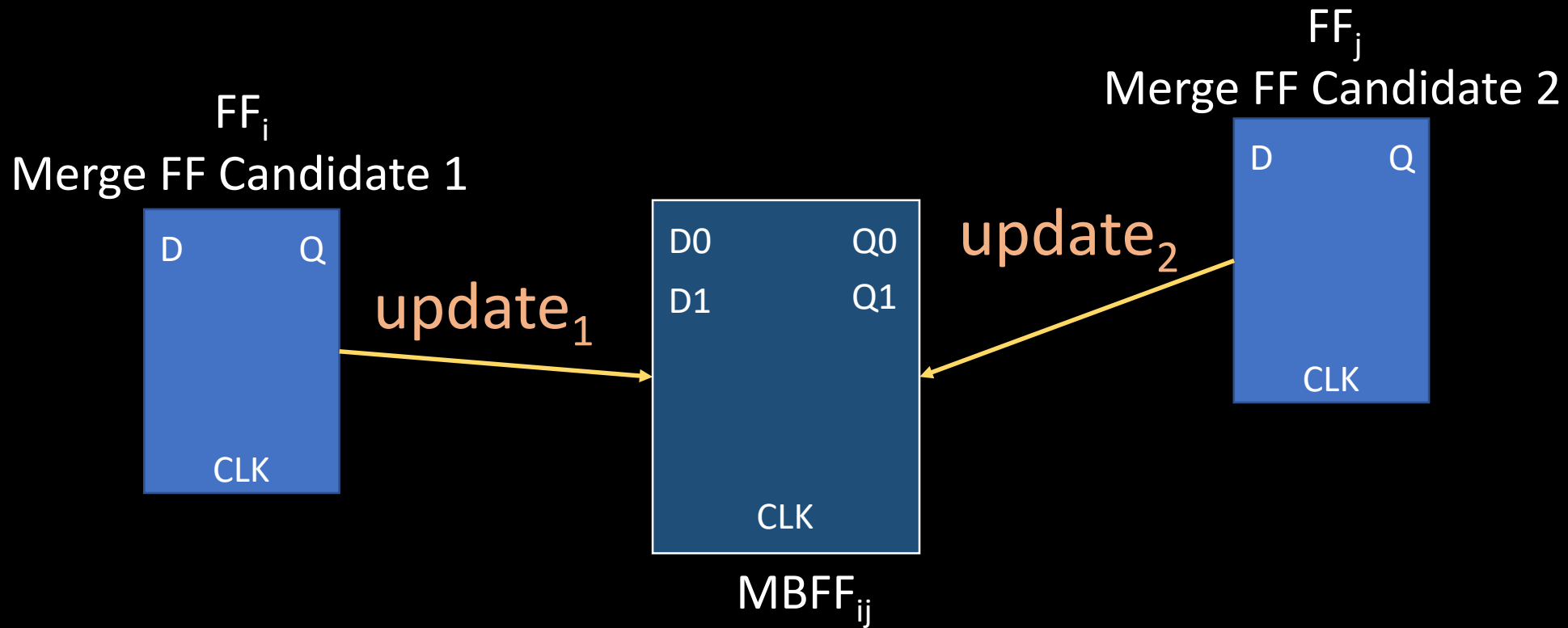- Summary

Clustering of FFs Mechanics

FF$_i$
Merge FF Candidate 1

fanin$_i$

D   Q

CLK

fanout$_i$

FF$_j$
Merge FF Candidate 2

fanin$_j$

D   Q

CLK

fanout$_j$

Displacing FFs

FF$_i$
Merge FF Candidate 1

FF$_j$
Merge FF Candidate 2

D   Q

CLK

D   Q

CLK

Δk_i

Δk_j

MBFF Merged
Location

Worst Case: FF's displacement introduces increase in wirelength

Displacing FFs

FF$_j$
Merge FF Candidate 2

FF$_i$
Merge FF Candidate 1

D   Q

D   Q

CLK

D0   Q0
D1   Q1

CLK

update$_1$

update$_2$

CLK

MBFF$_{ij}$

Moving each FF will **require** update to SCG to avoid "out-dated" skew constraints!!

Displacing FFs

FF$_i$
Merge FF Candidate 1

FF$_j$
Merge FF Candidate 2

D   Q

CLK

update$_1$
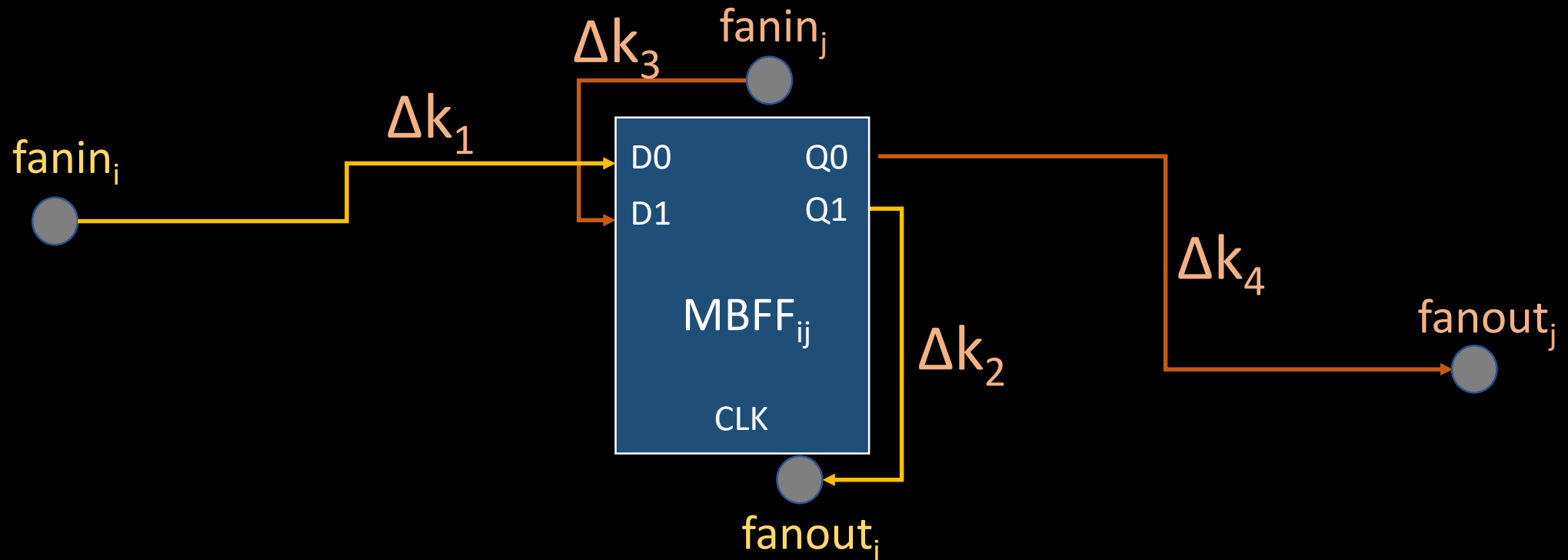
D0   Q0
D1   Q1

CLK

MBFF$_{ij}$

update$_2$

D   Q

CLK
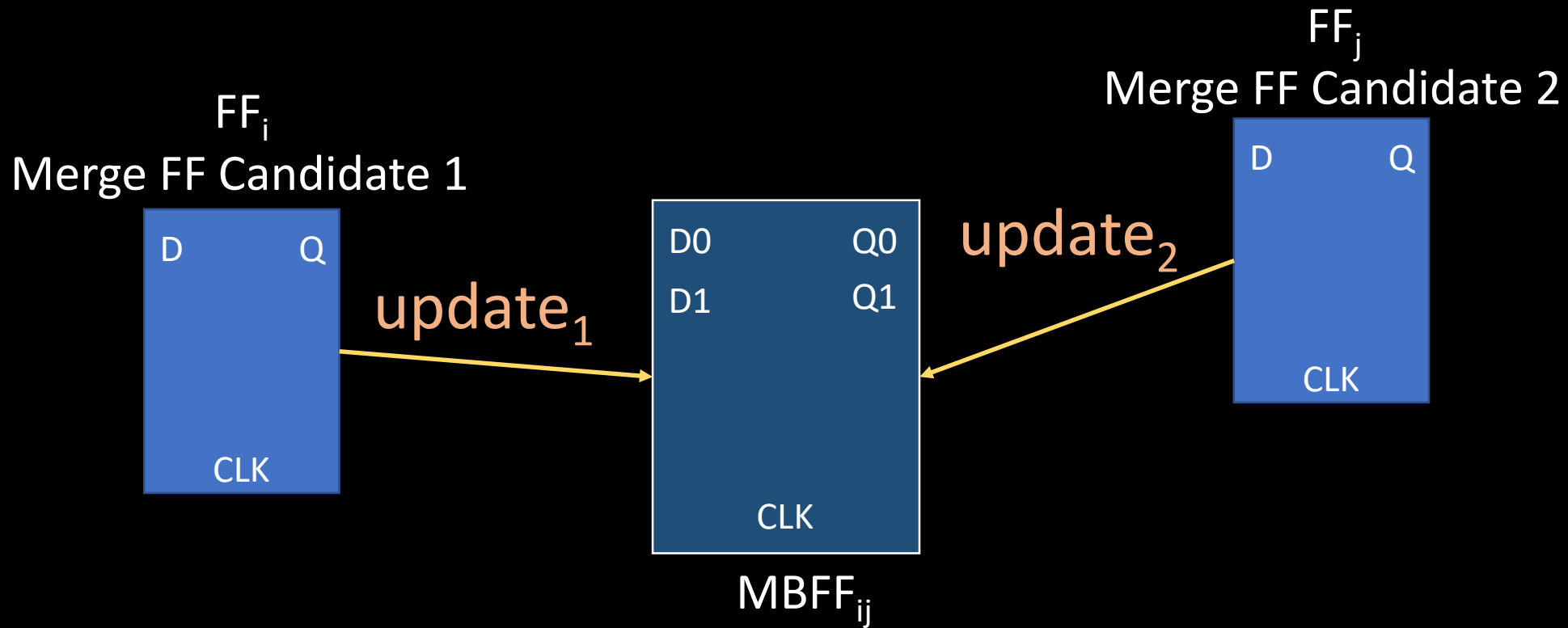
Furthermore, displacing FF$_i$ **does not guarantee**
that all skew constraints will be met after update$_1$

Timing Delays

FF$_j$
Merge FF Candidate 2

FF$_i$
Merge FF Candidate 1

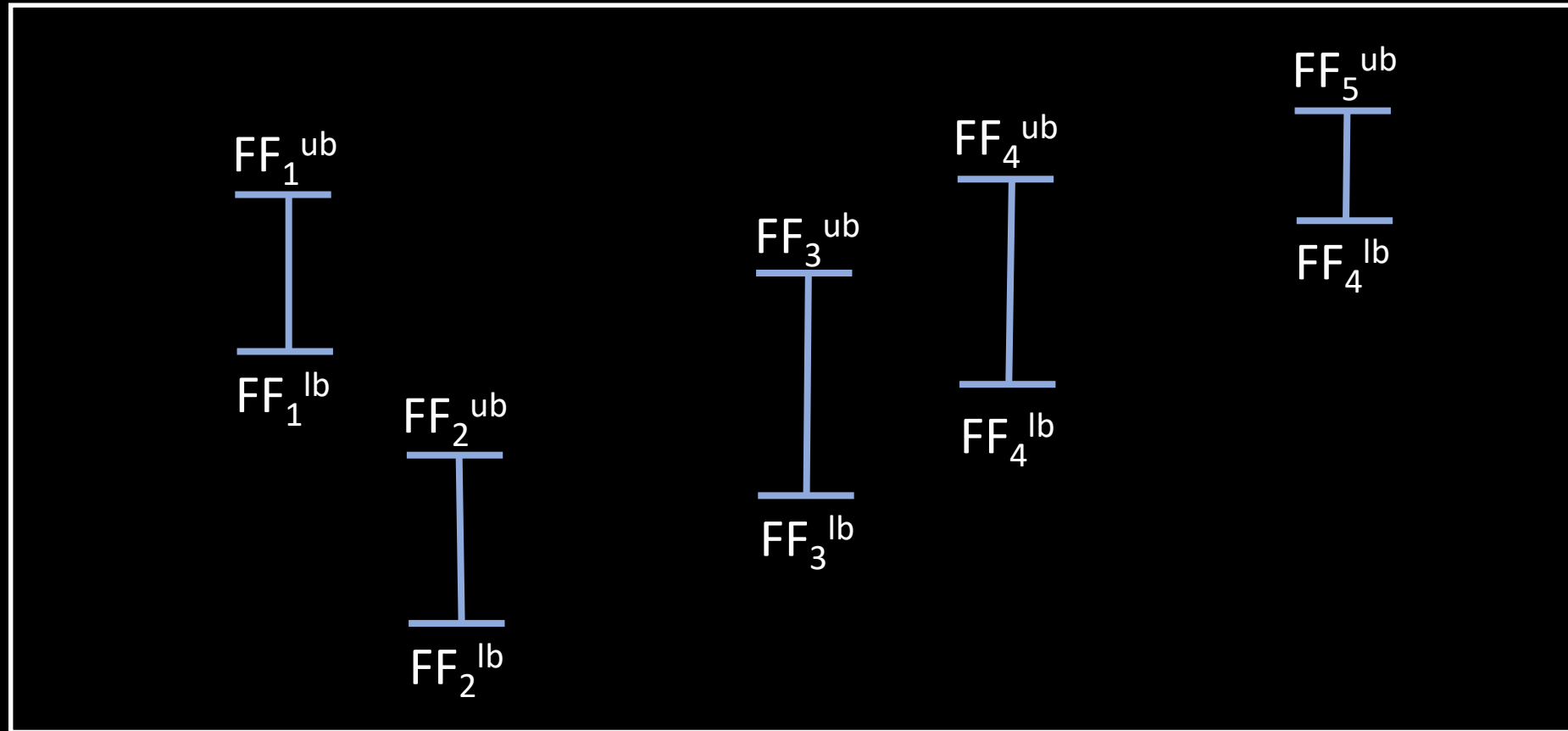Δk_i

Δk_j

Merge Location
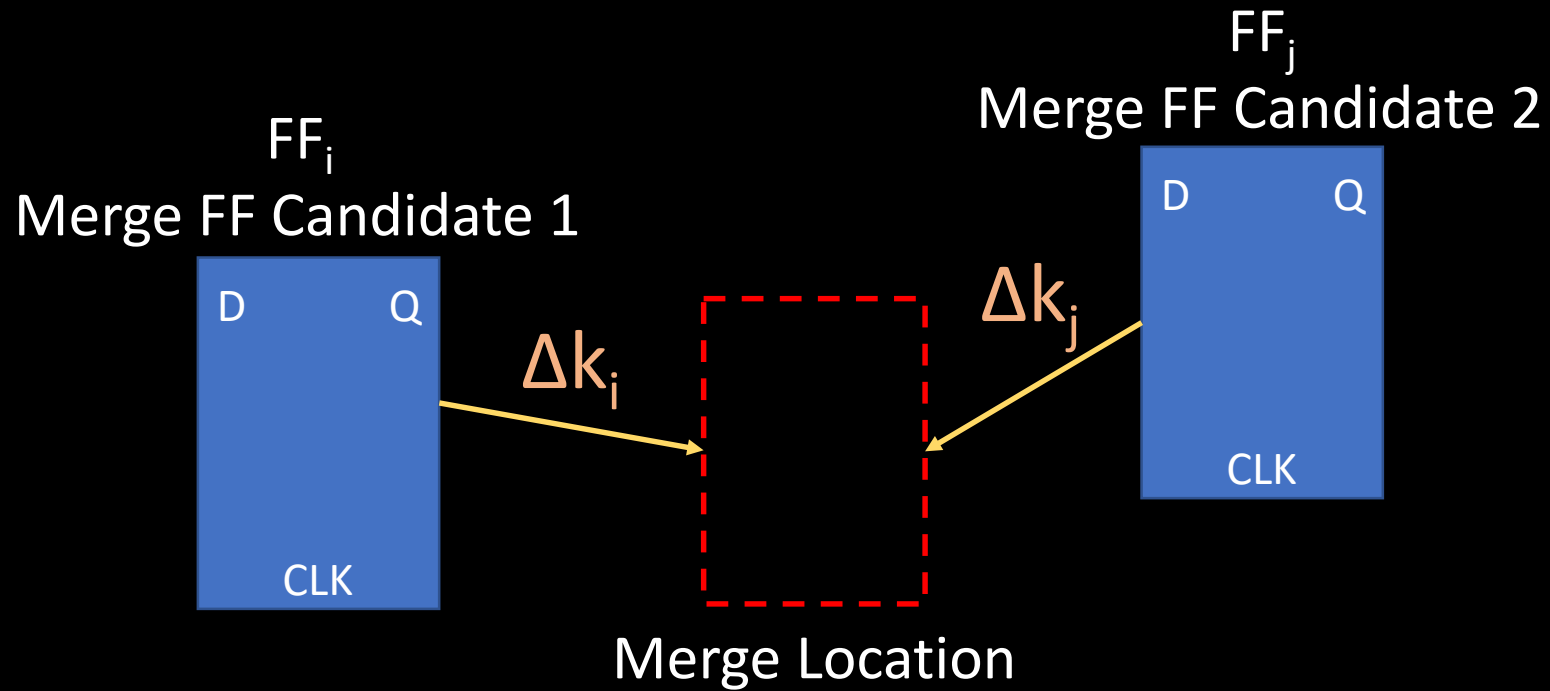
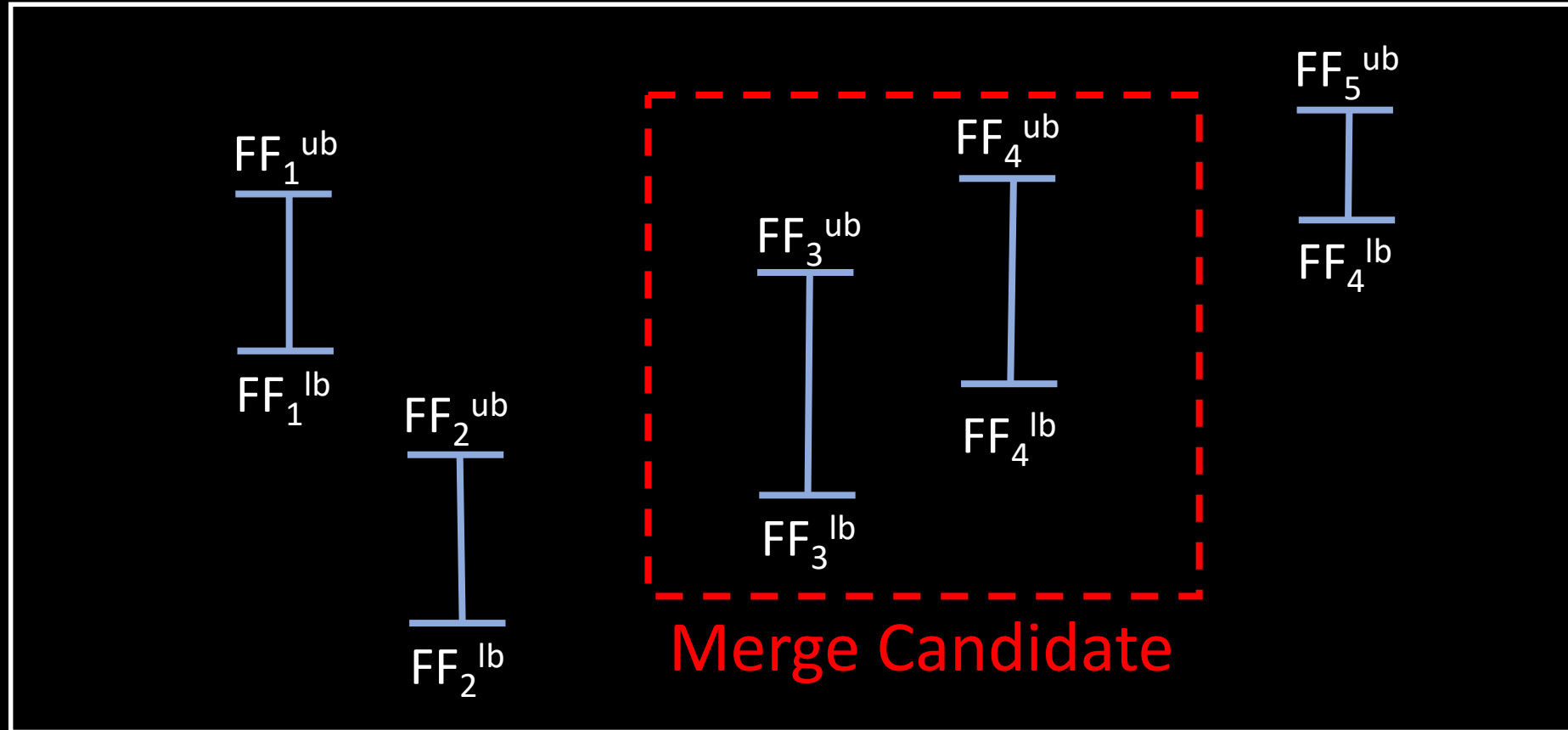Let Δk = timing delay for increase in wirelength

Timing Delays

$\Sigma \Delta k_i$ = total timing delay increased from displacing FFs

Comparing FF's Arrival Time Ranges
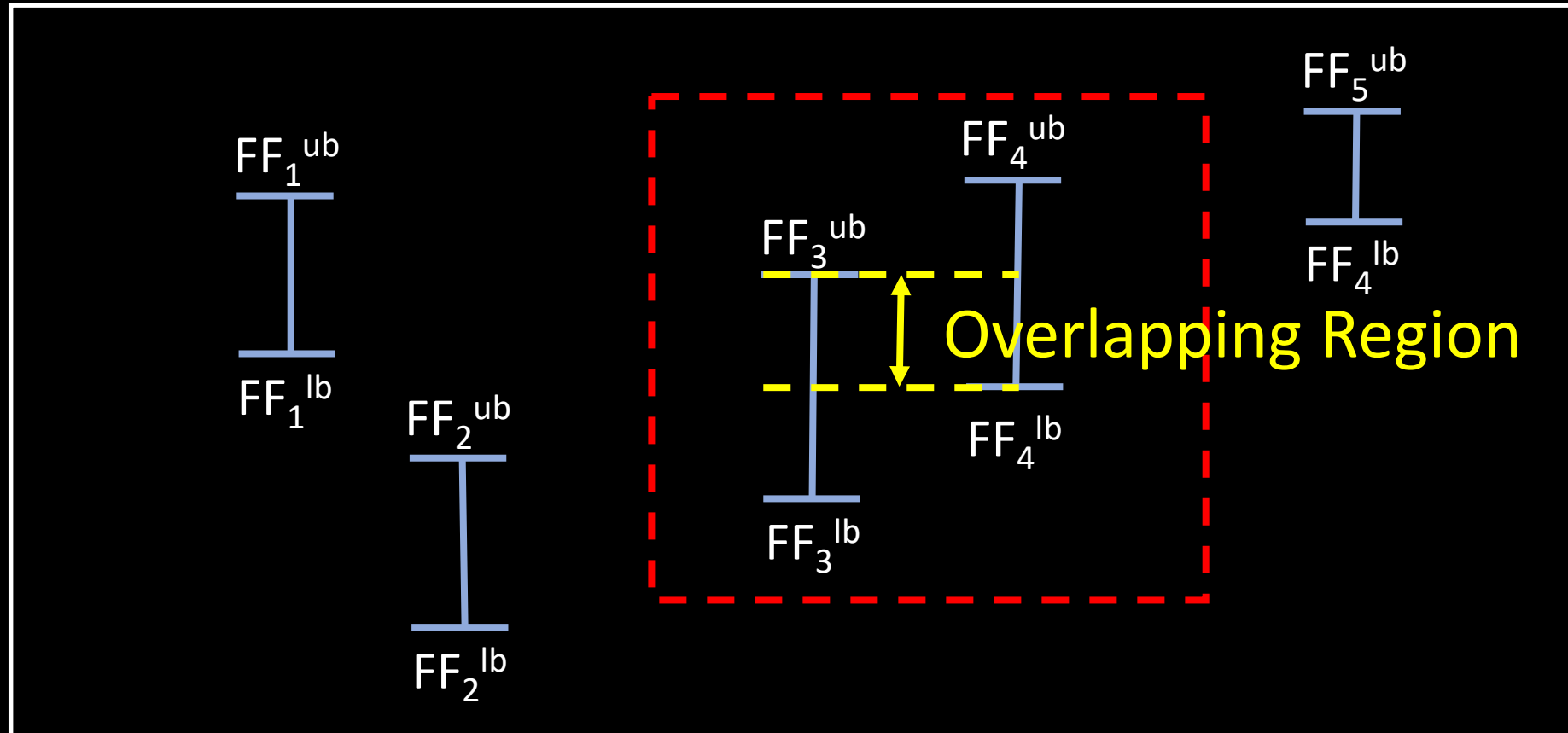

Example of arrival time ranges derived from skew constraints

Merging Flip-Flop Candidates

Overlapping Arrival Time Ranges Between FFs

Updating New Arrival Time Range

FF_1^{ub}

FF_1^{lb}

FF_2^{ub}

FF_2^{lb}

$MBFF_1^{ub}$

$MBFF_1^{lb}$

Resulting Merge FF
Arrival Time Range

FF_5^{ub}

FF_4^{lb}
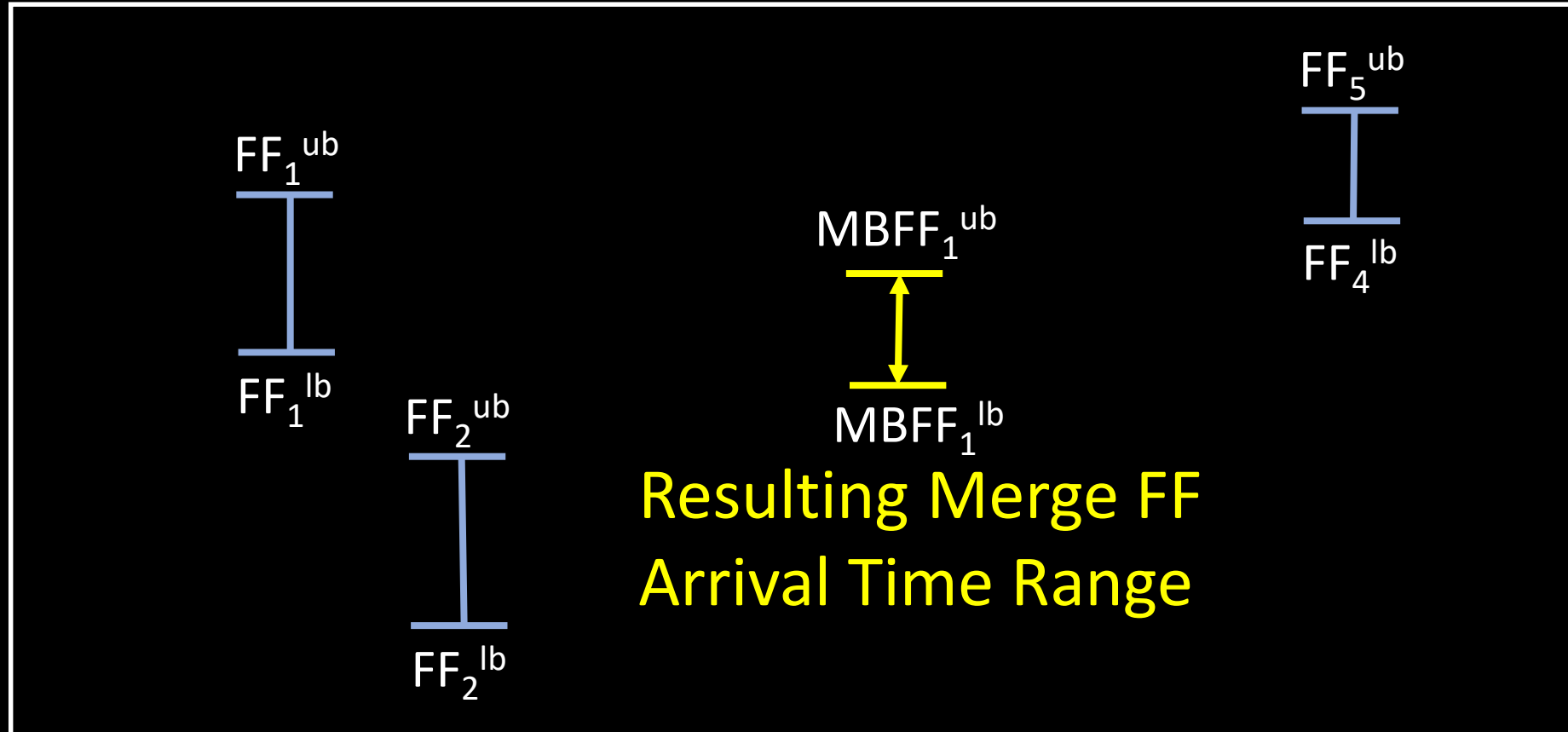
Fast computation to update Arrival Time Ranges
after merging FFs

Outline

- MBFF Introduction
- Bounded Arrival Time Useful Skew Tree
- Previous MBFF/Clustering Solutions
- Clustering of Flip-Flops Mechanics
- **Proposed Methodology**
- Evaluation & Experimental Results
- Summary

Proposed Methodology

Design Netlist

Specify Arrival Time Range

Select a FF Candidate Randomly

if not all FF evaluated

Cluster Flip-Flops Algorithm

Clock Tree Synthesis

Cluster Flip-Flops Algorithm

New FF Candidate

↓

Find Flip-Flop Nearest Neighbor

↓

Arrival Time Ranges Overlaps?

NO

YES ↓

Compute Cluster Location

↓

NO

Scaled Arrival Time Ranages Overlaps?

YES ↓

Cluster Flip-Flops

Cluster Flip-Flops Algorithm

New FF Candidate

↓

Find Flip-Flop Nearest Neighbor

↓

Arrival Time Ranges Overlaps?

NO

YES ↓

Compute Cluster Location

↓

NO

Scaled Arrival Time Ranages Overlaps?

YES ↓

Cluster Flip-Flops

Nearest Neighbor FF Selection Heuristic
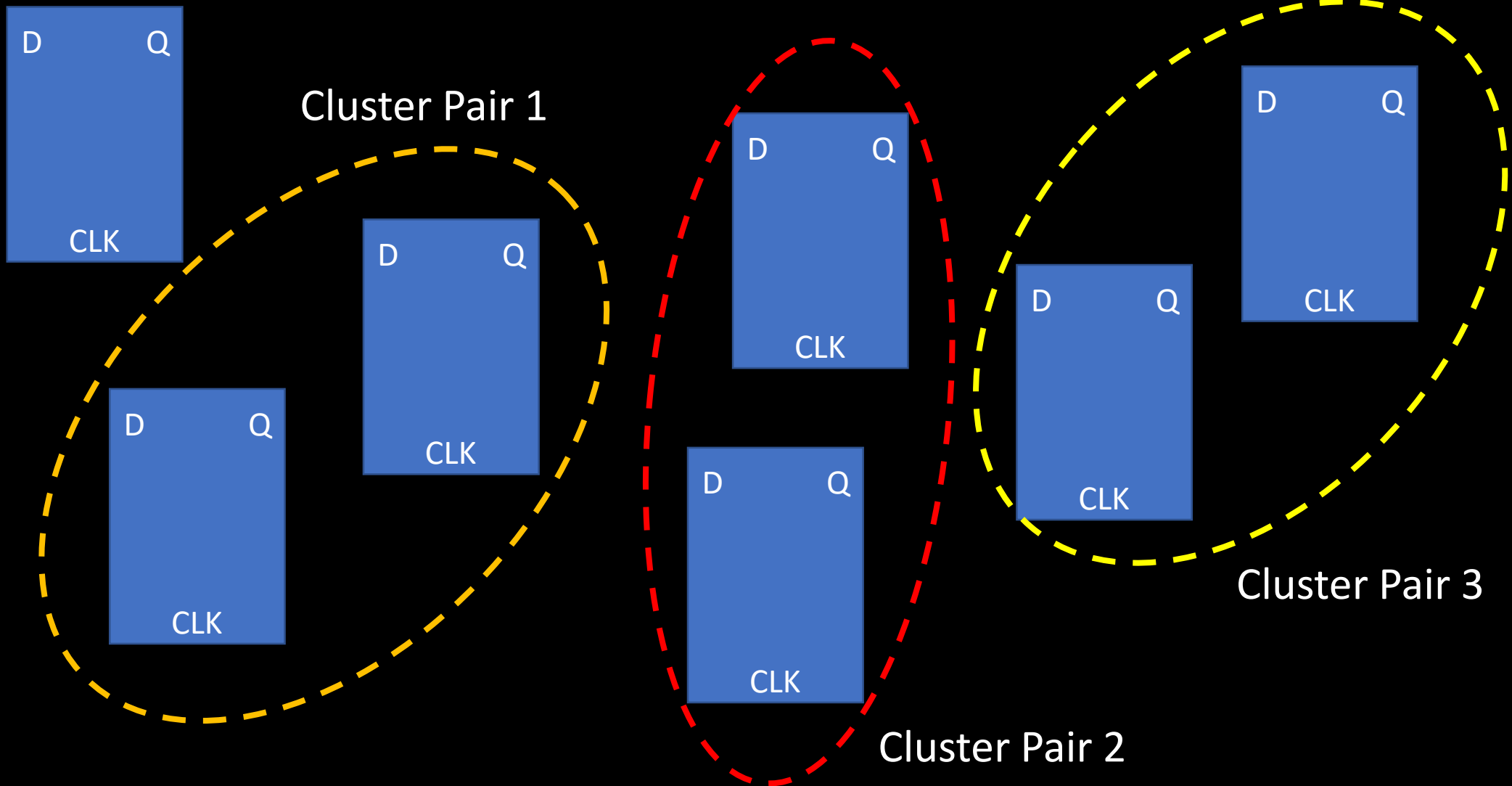
Cluster Pair 1
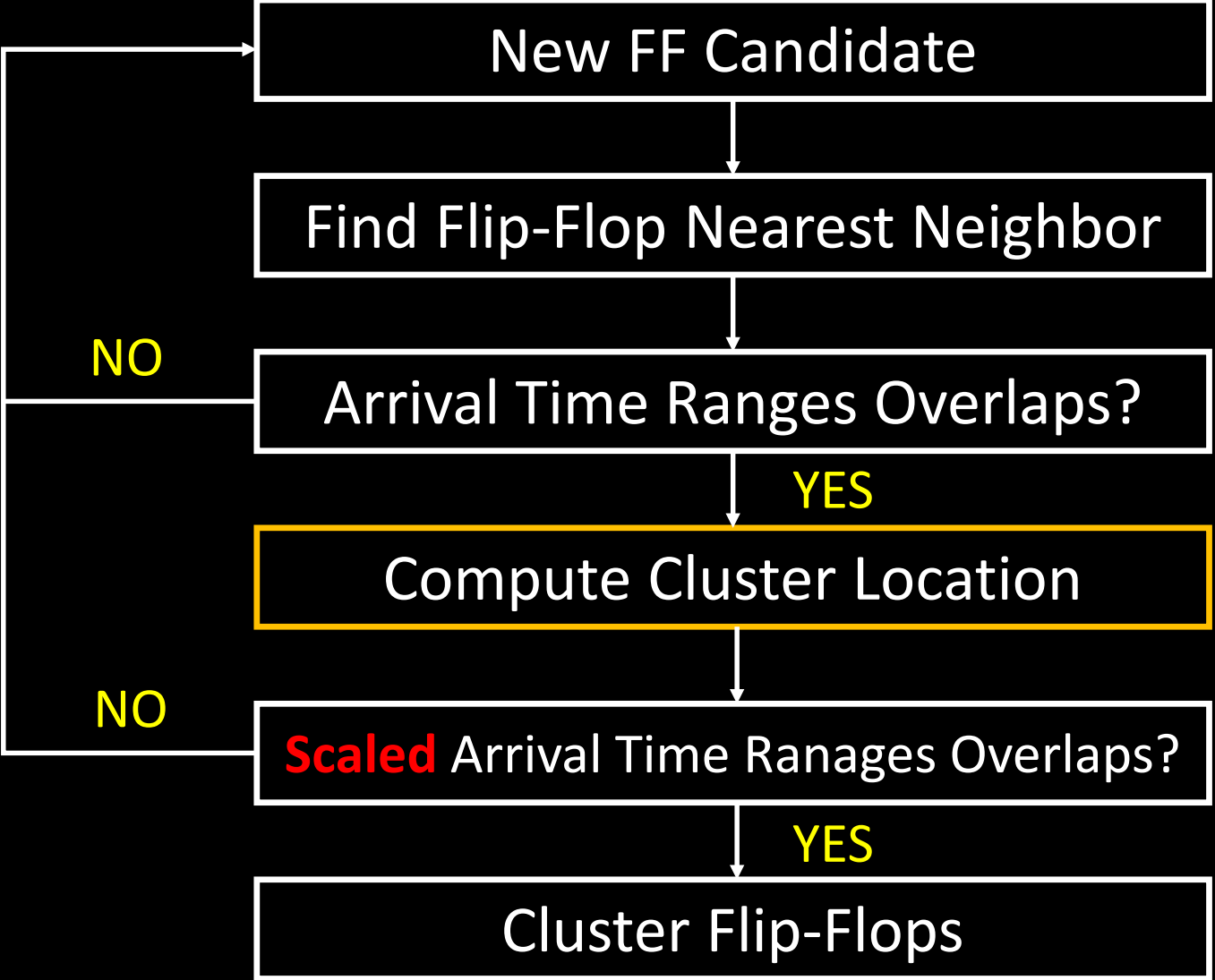
Cluster Pair 2

Cluster Pair 3

Cluster Flip-Flops Algorithm

New FF Candidate

↓

Find Flip-Flop Nearest Neighbor

↓

Arrival Time Ranges Overlaps?   NO →

YES ↓

Compute Cluster Location

↓

Scaled Arrival Time Ranages Overlaps?   NO →

YES ↓

Cluster Flip-Flops

Compute Clustering Location

Merge FF Candidate 1, FF_i
No. of Fanout = 5

Merge FF Candidate 2, FF_j
No. of Fanout = 2

D   Q

CLK

D   Q

CLK

Cluster Location

Cluster Location Algorithm:
Weighted FF candidate based on FF's fanout
Higher fanout ➡ Higher capacitive load

Compute Clustering Location

Merge FF Candidate 1, FF_i
No. of Fanout = 5

Merge FF Candidate 2, FF_j
No. of Fanout = 2

D   Q

D   Q

CLK

CLK

Cluster Location

Displacing FF with **higher capacitive load** will incur **higher delay change**

Cluster Flip-Flops Algorithm

New FF Candidate

↓

Find Flip-Flop Nearest Neighbor

↓

Arrival Time Ranges Overlaps?

NO

YES

↓

Compute Cluster Location

↓

NO

Scaled Arrival Time Ranges Overlaps?

YES

↓
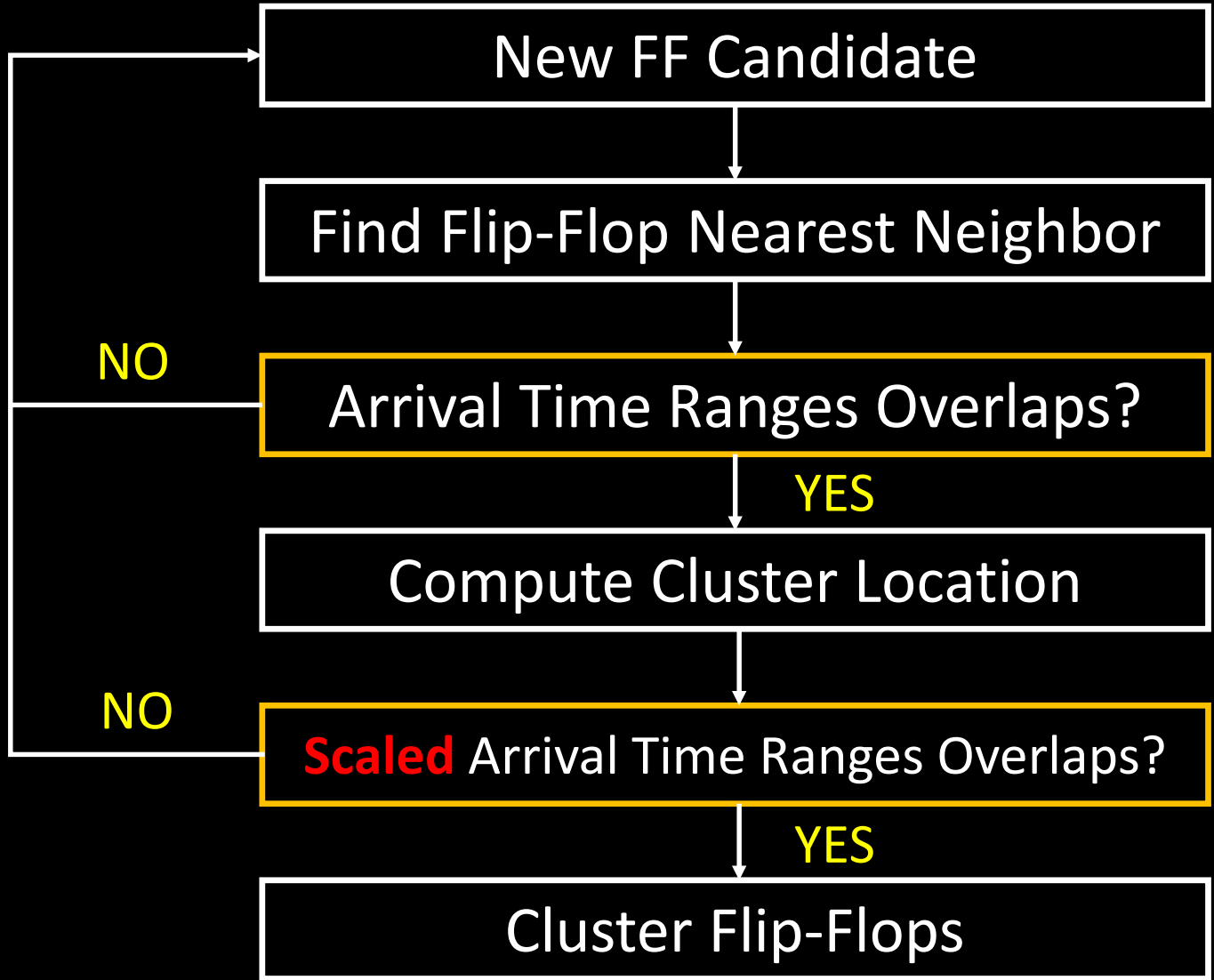
Cluster Flip-Flops

Arrival Time Range Overlaps
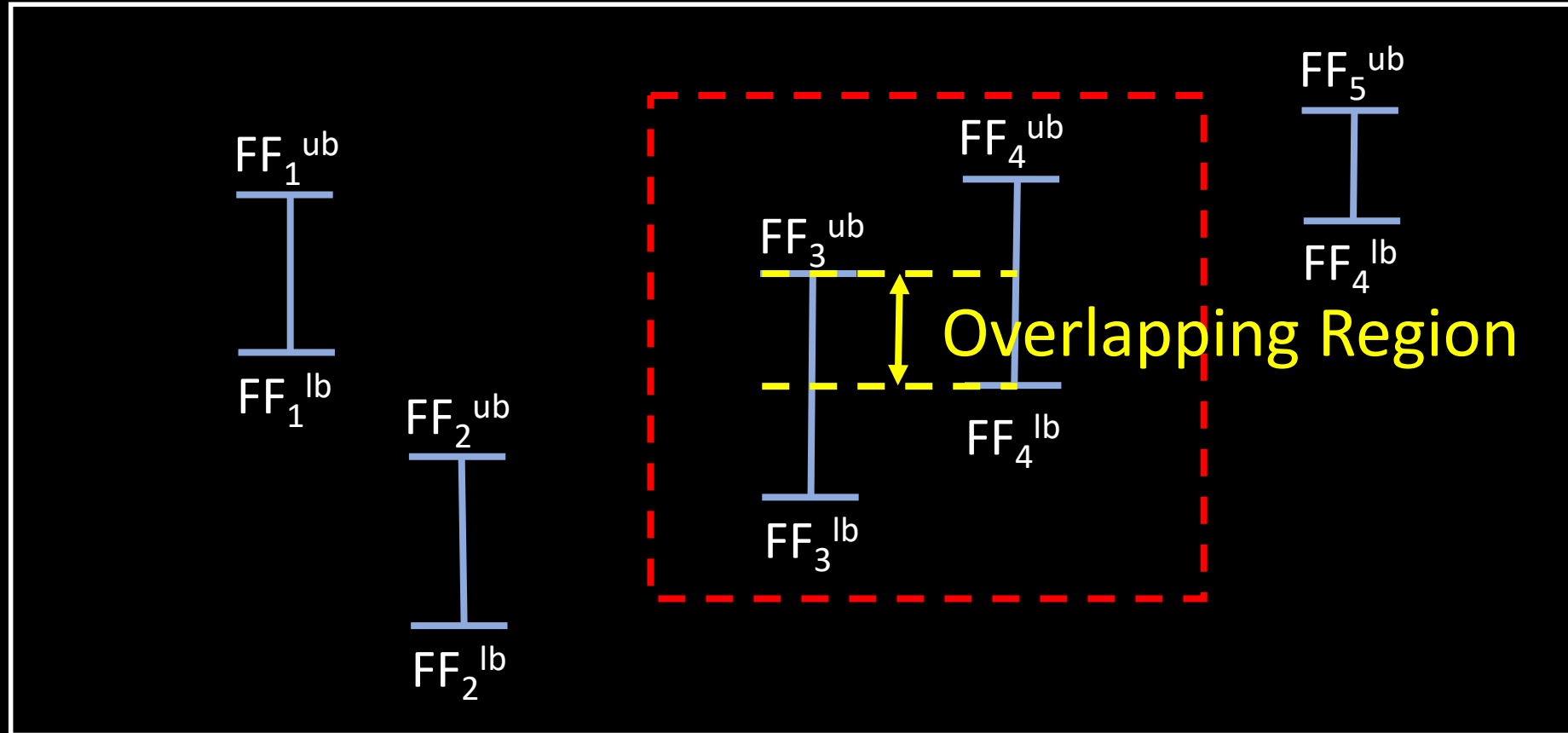
Updating Scaled Arrival Time Ranges

FF_1^{ub}

FF_1^{lb}

FF_2^{ub}

FF_2^{lb}

$MBFF_1^{ub}$

$MBFF_1^{lb}$

FF_5^{ub}

FF_4^{lb}

Scaled from FF's displacement
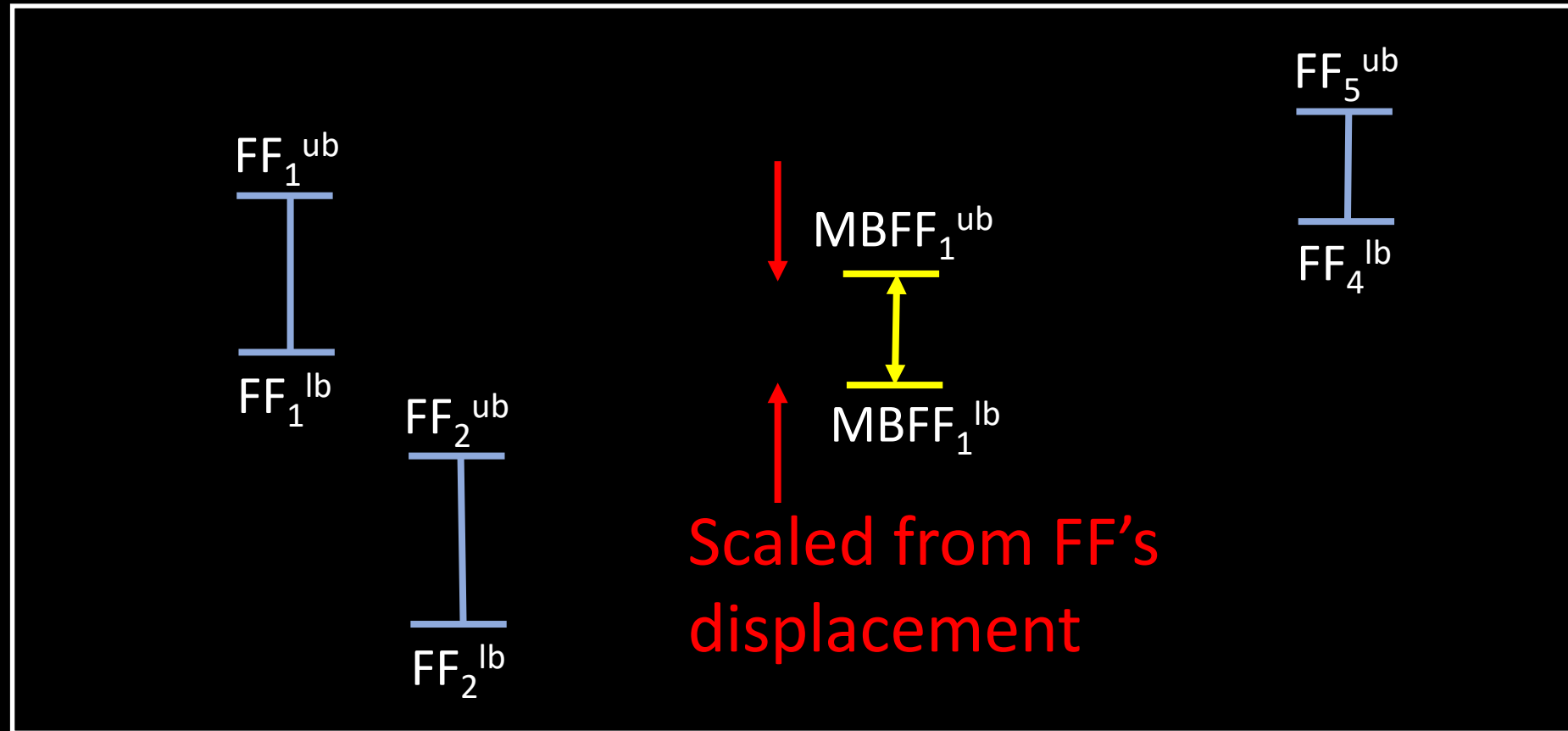
Scaled Arrival Time Range update after FF has been moved

Cluster Flip-Flops Algorithm
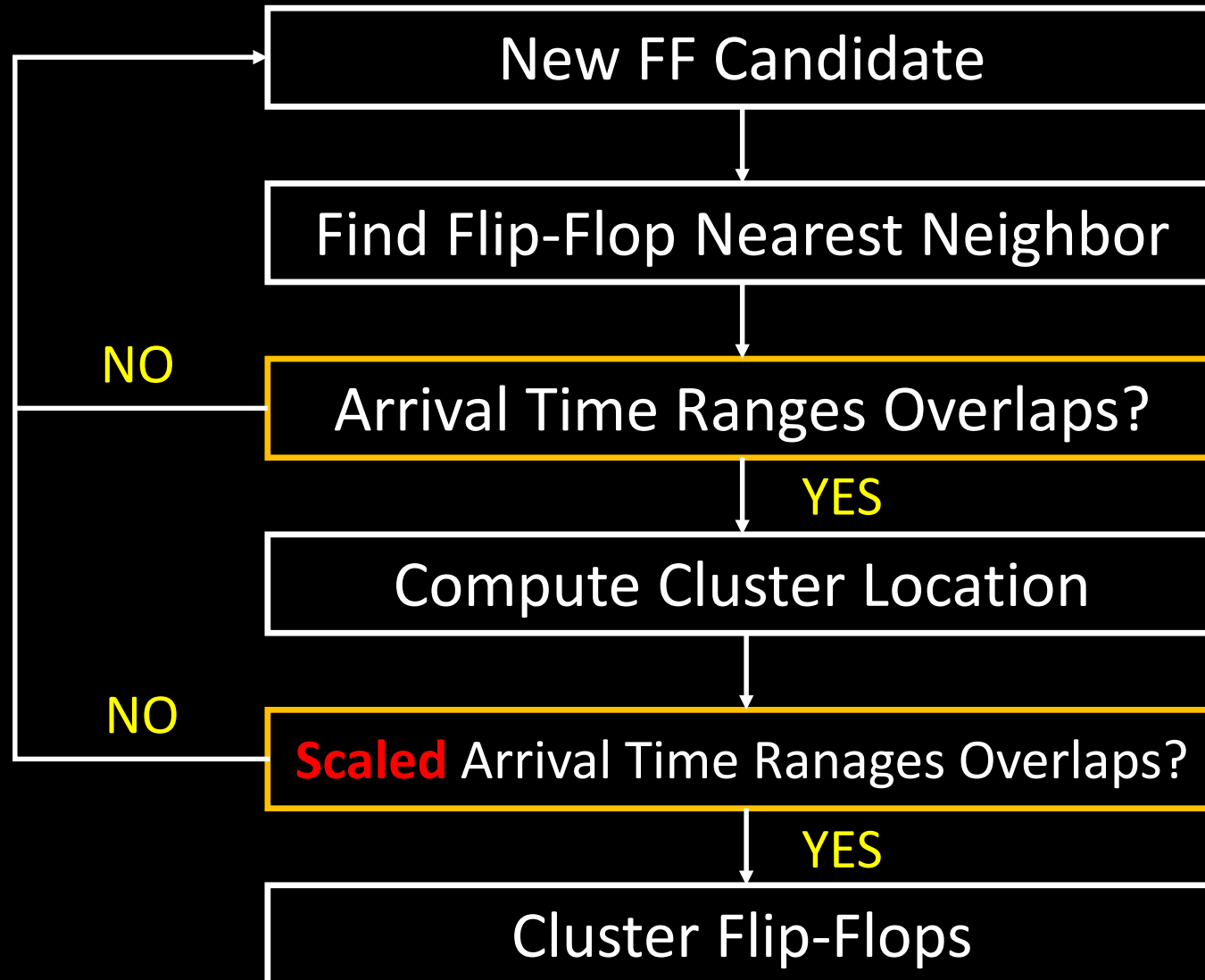
New FF Candidate

↓

Find Flip-Flop Nearest Neighbor

↓

Arrival Time Ranges Overlaps? — NO

↓ YES

Compute Cluster Location

↓

Scaled Arrival Time Ranages Overlaps? — NO

↓ YES

Cluster Flip-Flops

Outline

- MBFF Introduction
- Previous MBFF/Clustering Solutions
- Bounded Arrival Time Useful Skew Tree
- Clustering of Flip-Flops Mechanics
- Proposed Methodology
- **Evaluation & Experimental Results**
- Summary

Evaluation Framework
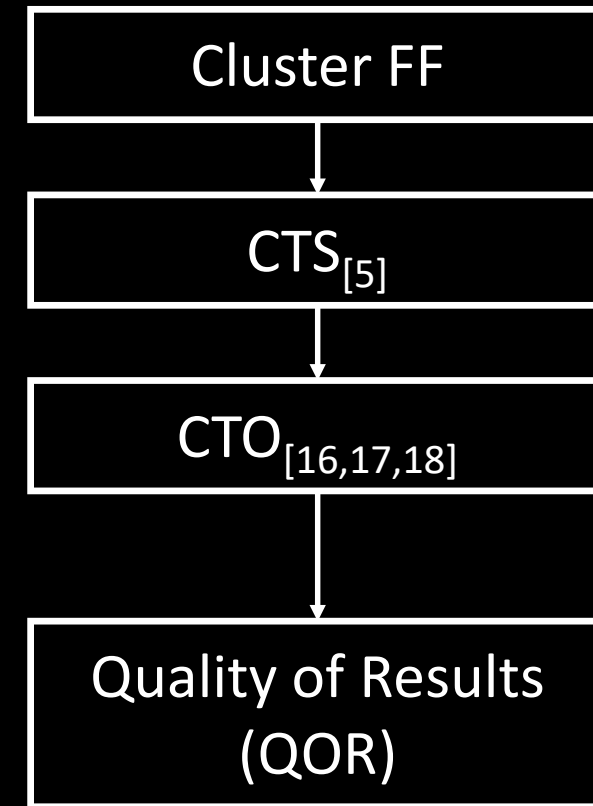
Monte Carlo Framework:

- Process Variation (10%)
- Voltage Variation (15%)
- Temperature Variation (30%)

Tree Structure:

- Zero-Skew Tree (ZST)
- Useful-Skew Tree (UST) [5]

Cluster FF

↓

CTS[5]

↓

CTO[16,17,18]

↓

Quality of Results (QOR)

[5] *R. Ewetz et al.* Clock Tree Construction based on arrival time constraints – *ACM 2017*
[16] *R. Ewetz et al.* MCMM clock tree optimization based on slack redistribution using a reduced slack graph. *ASP-DAC '16*
[17] *V. Ramachandran.* Construction of minimal functional skew clock trees. *ISPD'12*
[18] *S. Roy, et al.* Clock tree resynthesis for multi-corner multi-mode timing closure. *ISPD'14*

Clustering FF Results

Design[1]	No. of FF	Total Capacitance (pF)			
		ZST[2]	Cluster + ZST	UST[3]	Cluster + UST
USB Fast	1752	5.62	5.08	3.53	3.2
DMA	2121	6.94	5.56	4.31	3.72
openMSP430	686	2.66	1.98	1.44	1.23
Ethernet	10544	31.67	26.23	22.54	18.91
AES256	13216	40.47	33.75	25.09	22.72
PCI Bridge	3582	10.55	9.15	7.48	6.52
VGA enchanced	17071	49.89	41.08	34.83	30.03

Benchmark from
IWLS 2005 [8]

8 – 14% decrease in
total capacitance

[8] International Workshop for Logic Synthesis – http://irwls.org/iwls2005/benchmarks.html - 2005

Clustering FF + MBFF Transformation Results

Design[1]	No. of FF	Total Capacitance (pF)	
		UST[3]	Cluster + MBFF + UST
USB Fast	1752	3.53	2.48
DMA	2121	4.31	2.85
openMSP430	686	1.44	1.10
Ethernet	10544	22.54	14.25
AES256	13216	25.09	17.8
PCI Bridge	3582	7.48	5.00
VGA enchanced	17071	34.83	22.92

20% – 34% decrease
in total capacitance

Data & Clock Path Cap Reduction

Assuming **worst case** datapath wirelength increase
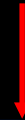
Design[1]	No. of FF	Data Path Wire Cap Reduction (%)	Clock Path Wire Cap Reduction (%)
USB Fast	1752	-4.17	13.25
DMA	2121	-9.86	19.47
openMSP430	686	-7.29	19.81
Ethernet	10544	-6.02	22.25
AES256	13216	-1.8	13.03
PCI Bridge	3582	-4.81	18.05
VGA enchanced	17071	-8.98	19.56

Clock Path WireCap Reduction **>>** Data Path Wire Cap Reduction

Outline

- MBFF Introduction
- Bounded Arrival Time Useful Skew Tree
- Previous MBFF/Clustering Solutions
- Clustering of Flip-Flops Mechanics
- Proposed Methodology
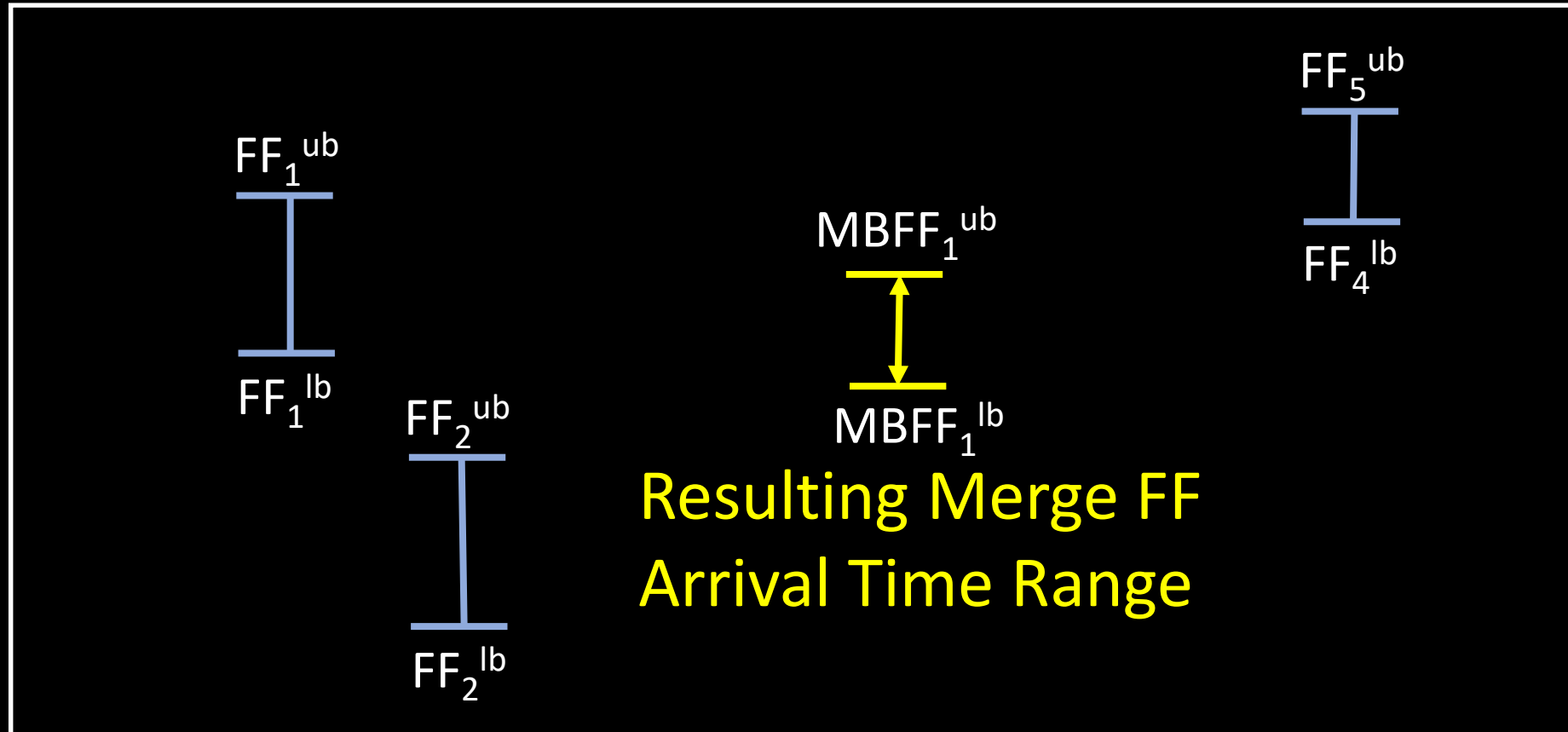- Evaluation & Experimental Results
- **Summary**

Summary

- Displacing FFs requires update to skew constraints to **ensure timing is met**
- Clustering of Flip-Flops based on Bounded Arrival Time Range Clock Tree Synthesis provides **fast update** to the skew constraints while ensuring timing correctness
- Clustering of FFs reduces **routing complexity**
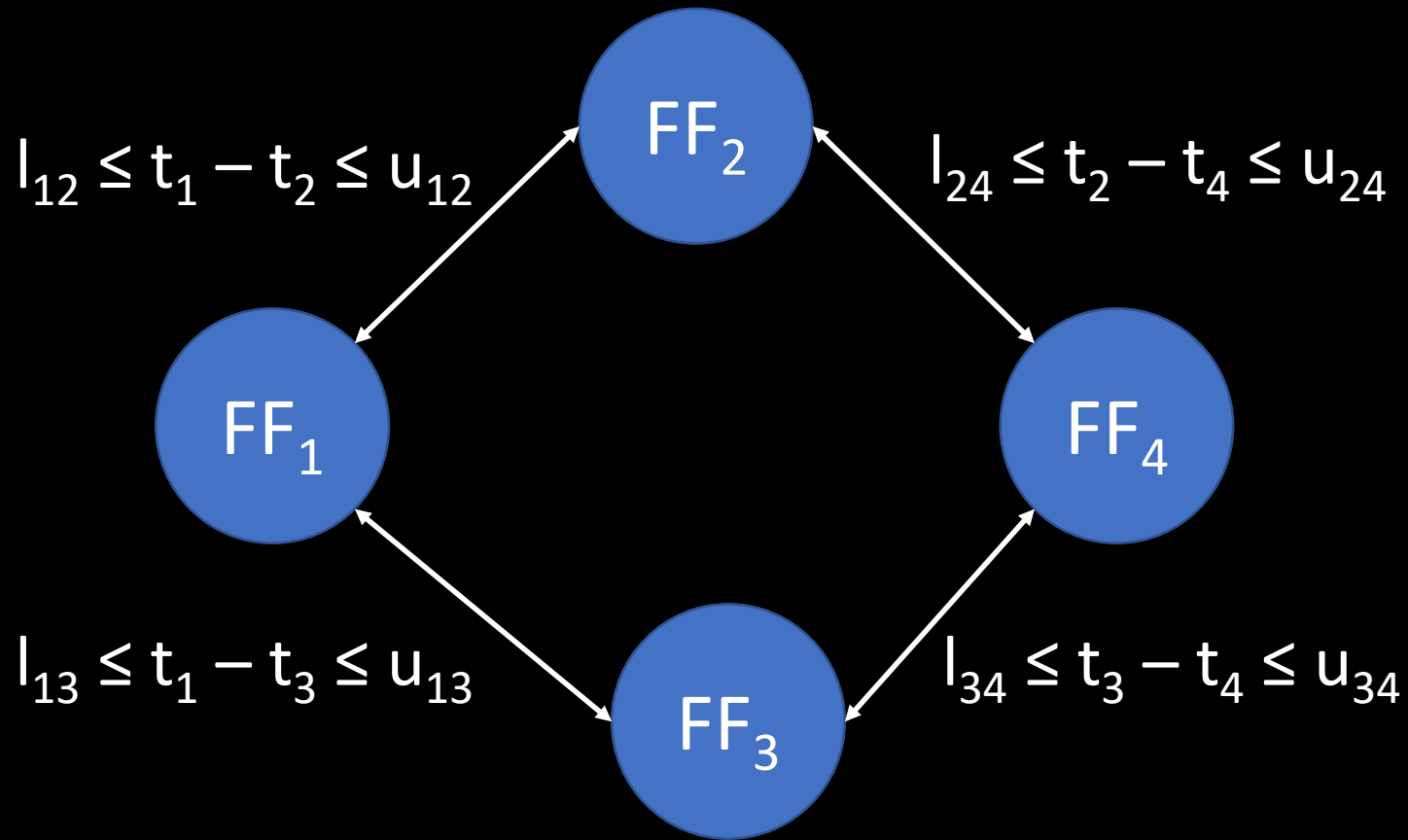- MBFF transformation further **reduces power consumption**

Q&A

Backup Slides

Bounded Arrival Time Ranges

FF_5^{ub}

FF_1^{ub}

$MBFF_1^{ub}$

FF_4^{lb}

FF_1^{lb}

FF_2^{ub}

$MBFF_1^{lb}$

Resulting Merge FF
Arrival Time Range

FF_2^{lb}

Skew Constraint Graph (SCG)

$$l_{12} \leq t_1 - t_2 \leq u_{12}$$

$$l_{24} \leq t_2 - t_4 \leq u_{24}$$

$$l_{13} \leq t_1 - t_3 \leq u_{13}$$

$$l_{34} \leq t_3 - t_4 \leq u_{34}$$

Skew Constraint Graph