Depth-optimal Buffer and Splitter Insertion and Optimization in AQFP Circuits

Alessandro Tempia Calvino

Giovanni De Micheli

ASP-DAC 2023

Integrated Systems Laboratory, EPFL, Lausanne, Switzerland
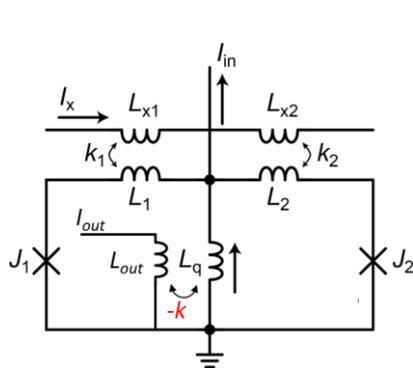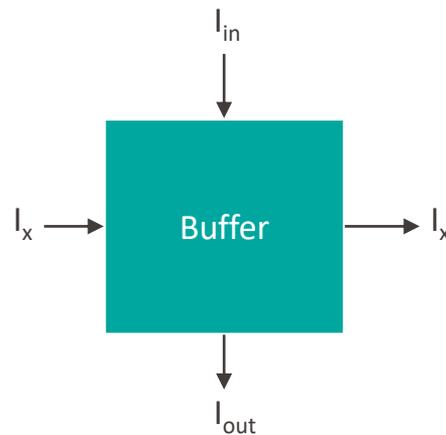
alessandro.tempiacalvino@epfl.ch

17th January 2023

Adiabatic Quantum-Flux Parametron

- Superconducting device [1]

- Operates at few degrees Kelvin ≈4K

- Very **low energy consumption** → 10^{-2} compared to CMOS

Quantum-Flux Parametron (QFP)

[1] M. Hosoya *et al.*, "Quantum flux parametron: a single quantum flux device for Josephson supercomputer," in *IEEE Transactions on Applied Superconductivity*, vol. 1, no. 2, pp. 77-89, June 1991.

Adiabatic Quantum-Flux Parametron

- Driven by AC power (I_x) $\rightarrow$ adiabatic operation
  - Serves also as a **clock** signal
  - Synchronizes the computation
- Every gate is **clocked**
- Clock at ≈5 GHz
- Majority-based logic
  - Free inversion
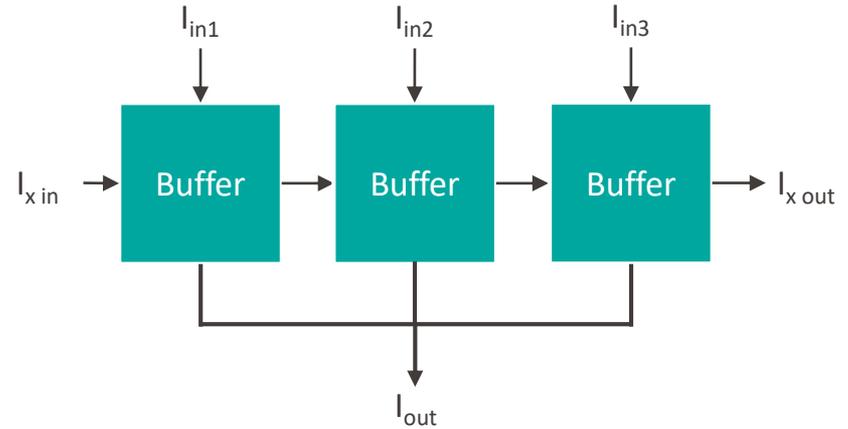
Majority-3 gate

AQFP: technology constraints

- **Fanout branching:**
  - Limited driving capacity of 1 gate
  - Need splitting elements for multiple fanout
  - A technology library contains a 1-to-s_b **clocked splitter** (s_b typically 3 or 4)
- **Path balancing:**
  - Gates are clocked
  - Need operations to be synchronized
  - Insert **clocked buffers** to delay fast paths

AQFP: technology constraints example

Initial graph   Fanout branching   Path balancing
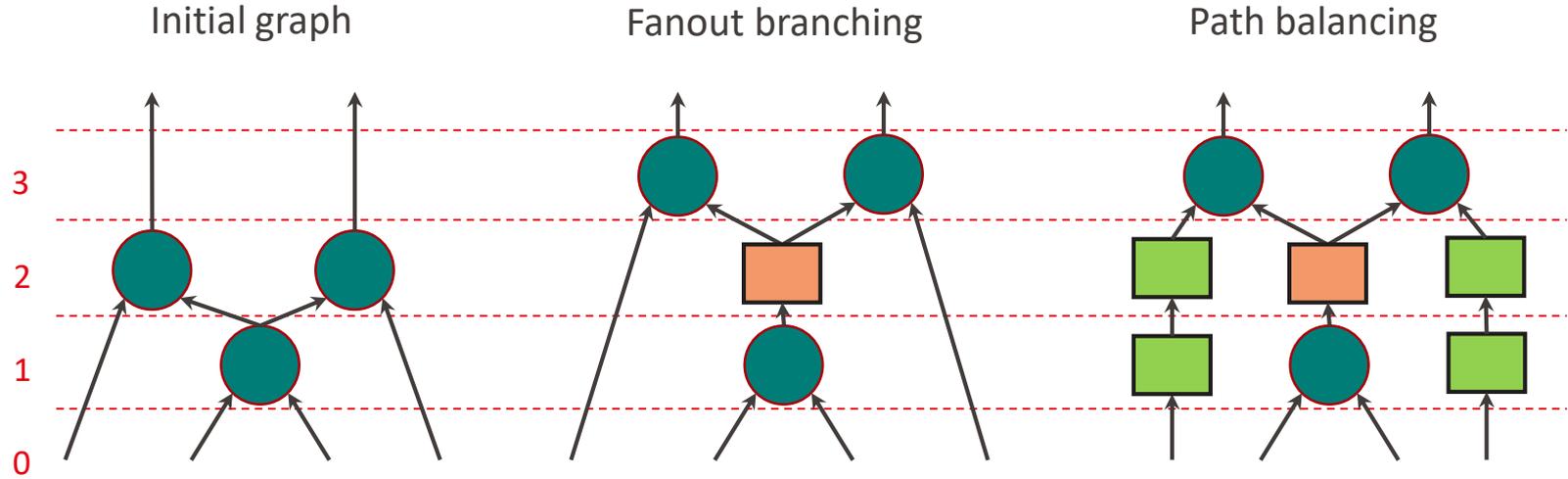
AQFP: EDA challenges

- A **large portion** of the area is occupied by buffers and splitters

- How do we synthesize the logic to reduce imbalances and fanout

- How do we insert buffers and splitters to minimize:
  - Area (NP-hard)
  - Delay

- Area is related to delay → path balancing

AQFP: related works

Main research problems:

1) Logic optimization for AQFP

2) Buffer and splitter insertion and optimization

3) Complete flow for AQFP [2]

[2] Christopher L Ayala, Ro Saito, Tomoyuki Tanaka, Olivia Chen, Naoki Takeuchi, Yuxing He, and Nobuyuki Yoshikawa, "A semi-custom design methodology and environment for implementing superconductor adiabatic quantum-flux-parametron microprocessors", Superconductor Science and Technology 33, 5 (2020).

AQFP: related works

Logic optimization for AQFP:

- Majority-inverter-graph (MIG) optimization [3]

- MIG optimization considering fanout branching and path balancing [4][5]
  - Limit the gate fanout
  - Prefer balanced logic

[3] Luca Amarú, Pierre-Emmanuel Gaillardon, and Giovanni De Micheli, "Majority-Inverter Graph: A New Paradigm for Logic Optimization," IEEE Trans. CAD 2016
[4] Eleonora Testa, Siang-Yun Lee, Heinz Riener, and Giovanni De Micheli, "Algebraic and Boolean Optimization Methods for AQFP Superconducting Circuits," In Proc. ASPDAC 2021
[5] G. Meuli *et al.*, "Majority-based Design Flow for AQFP Superconducting Family," *DATE* 2022

AQFP: related works

A. Tempia Calvino, G. De Micheli

Buffer and splitter insertion and optimization:

- First automated insertion with local optimization [6]

- Optimal insertion for a single wire and local optimization [7]

- Formulation as a scheduling problem [8]

[6] R. Cai, O. Chen, A. Ren, N. Liu, N. Yoshikawa and Y. Wang, "A Buffer and Splitter Insertion Framework for Adiabatic Quantum-Flux-Parametron Superconducting Circuits," ICCD 2019

[7] Chao-Yuan Huang, Yi-Chen Chang, Ming-Jer Tsai, and Tsung-Yi Ho, "An Optimal Algorithm for Splitter and Buffer Insertion in Adiabatic Quantum-Flux-Parametron Circuits," In ICCAD 2021

[8] Siang-Yun Lee, Heinz Riener, and Giovanni De Micheli, "Beyond local optimality of buffer and splitter insertion for AQFP circuits," In Proc. DAC 2022
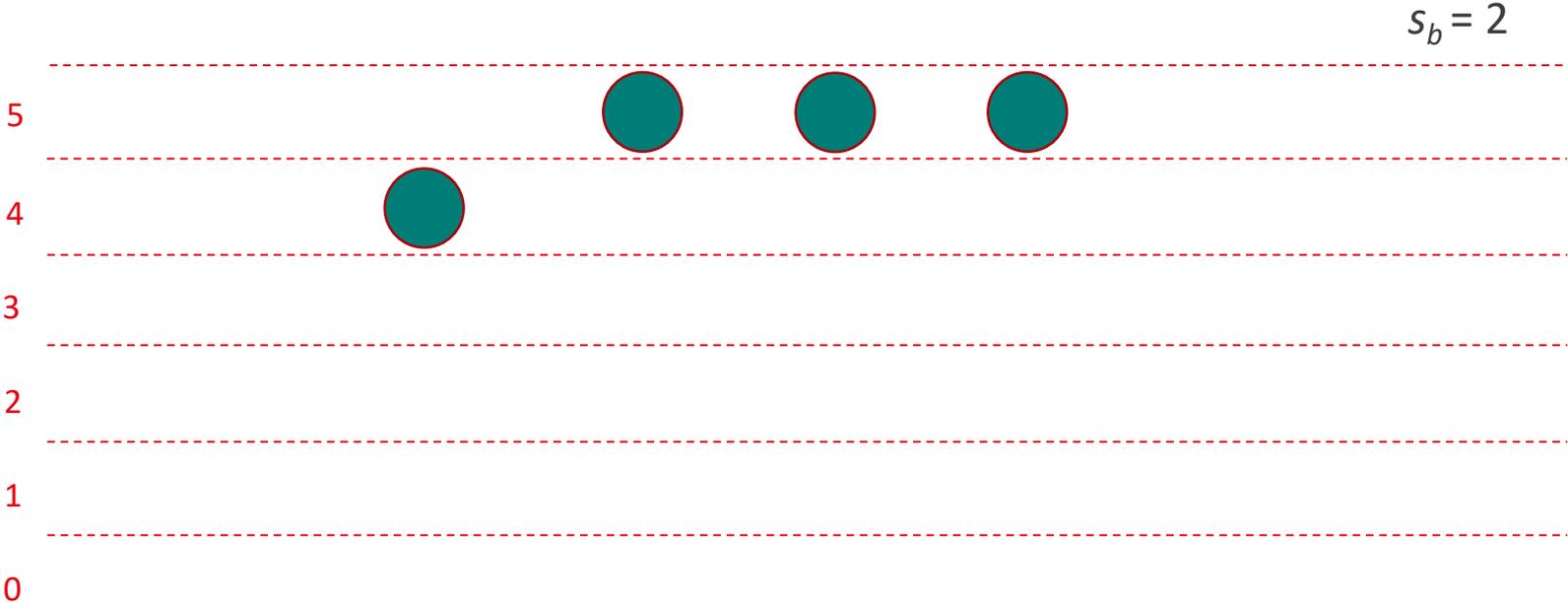
AQFP: contributions

- The first **depth-optimal** algorithm to insert buffers and splitters

- A heuristic based on **minimum-register retiming** for **global** B/S minimization

- An AQFP **mapping and optimization** flow

Depth-optimal buffer/splitter insertion
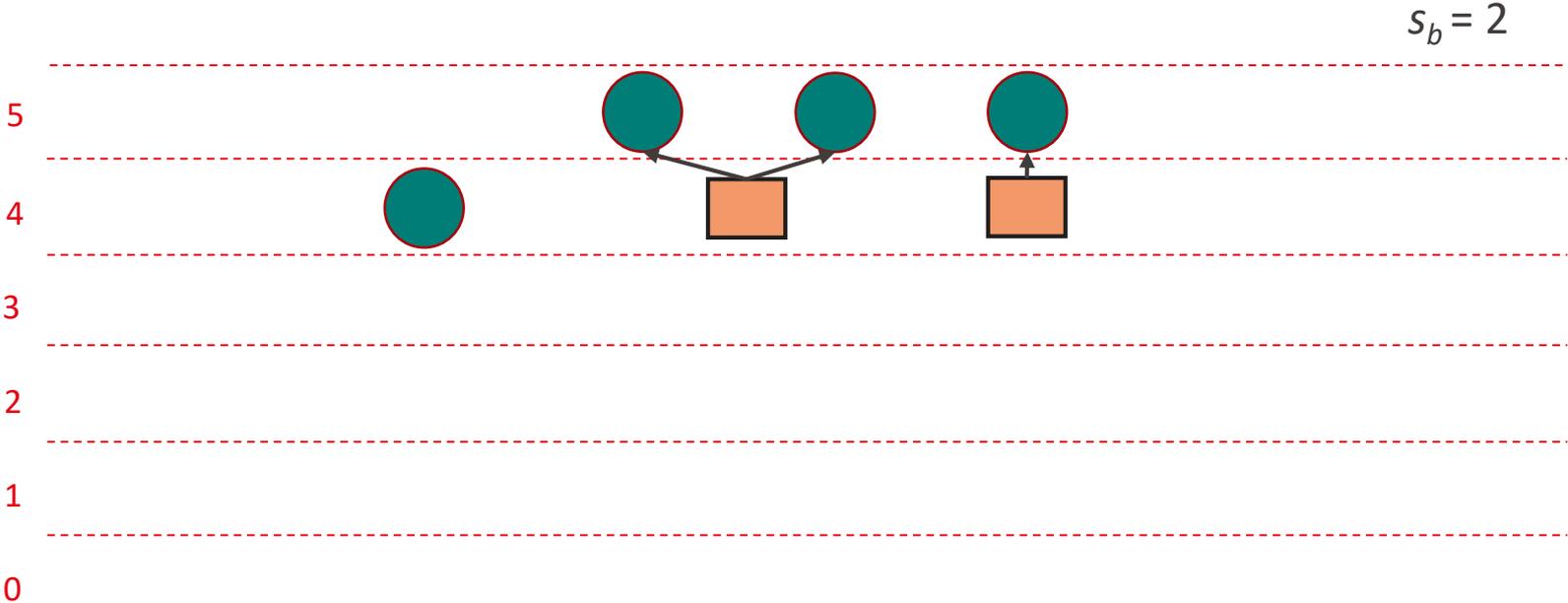
Depth-optimal buffer/splitter insertion

As a scheduling problem → assign nodes to a level

$s_b = 2$

5

4

3

2

1

0

Depth-optimal buffer/splitter insertion

As a scheduling problem → assign nodes to a level

$s_b = 2$

Depth-optimal buffer/splitter insertion

As a scheduling problem → assign nodes to a level
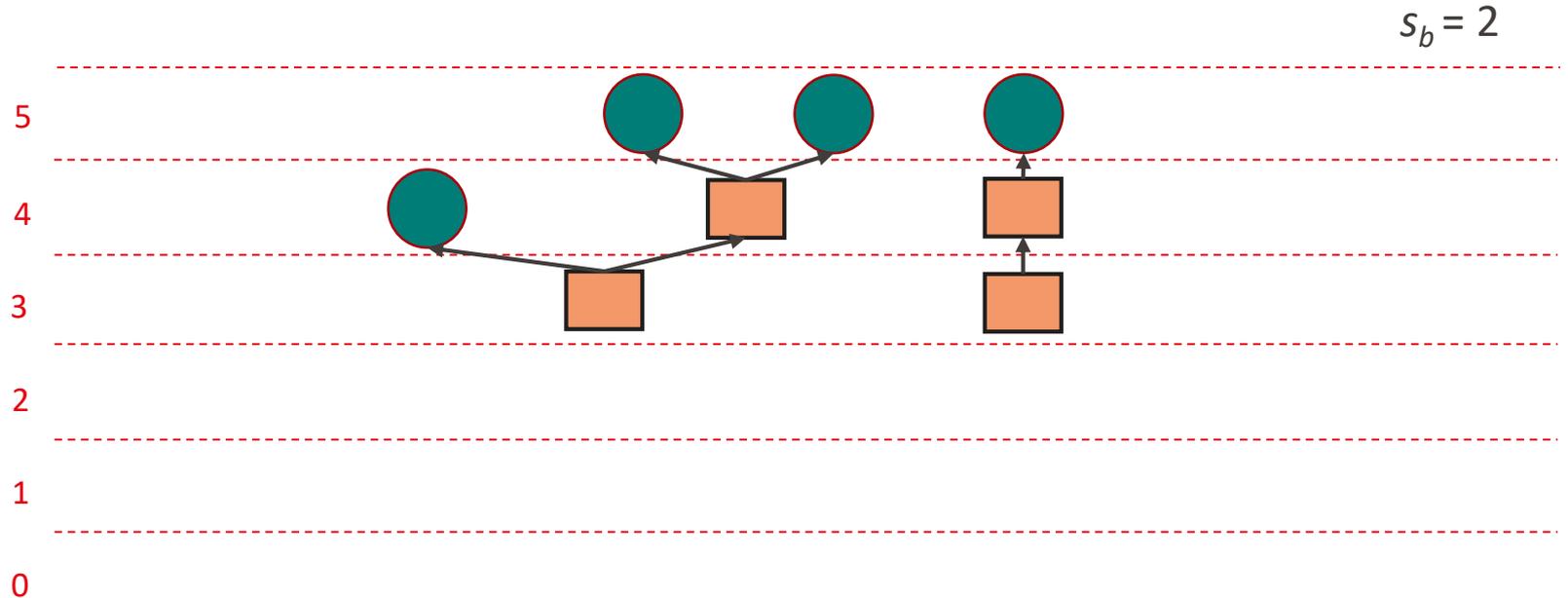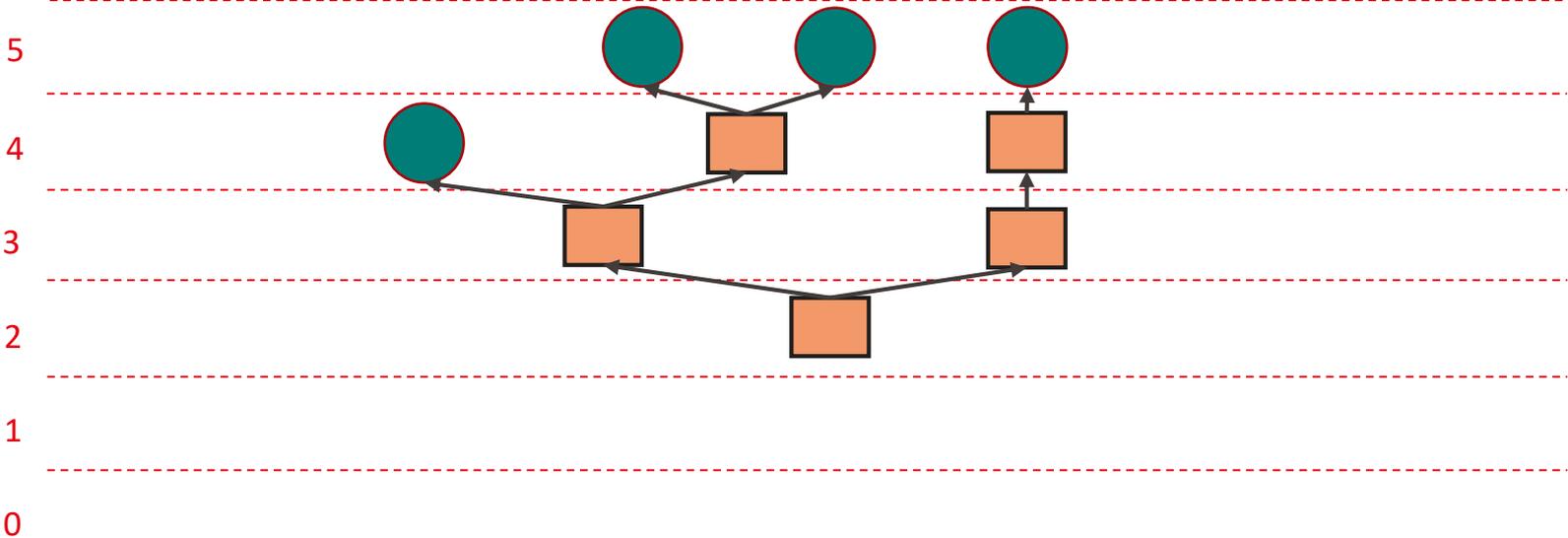
$s_b = 2$

Depth-optimal buffer/splitter insertion

As a scheduling problem → assign nodes to a level

$s_b = 2$

Depth-optimal buffer/splitter insertion

As a scheduling problem → assign nodes to a level

$s_b = 2$

5

4

3

2

1   Assigned to level 1

0
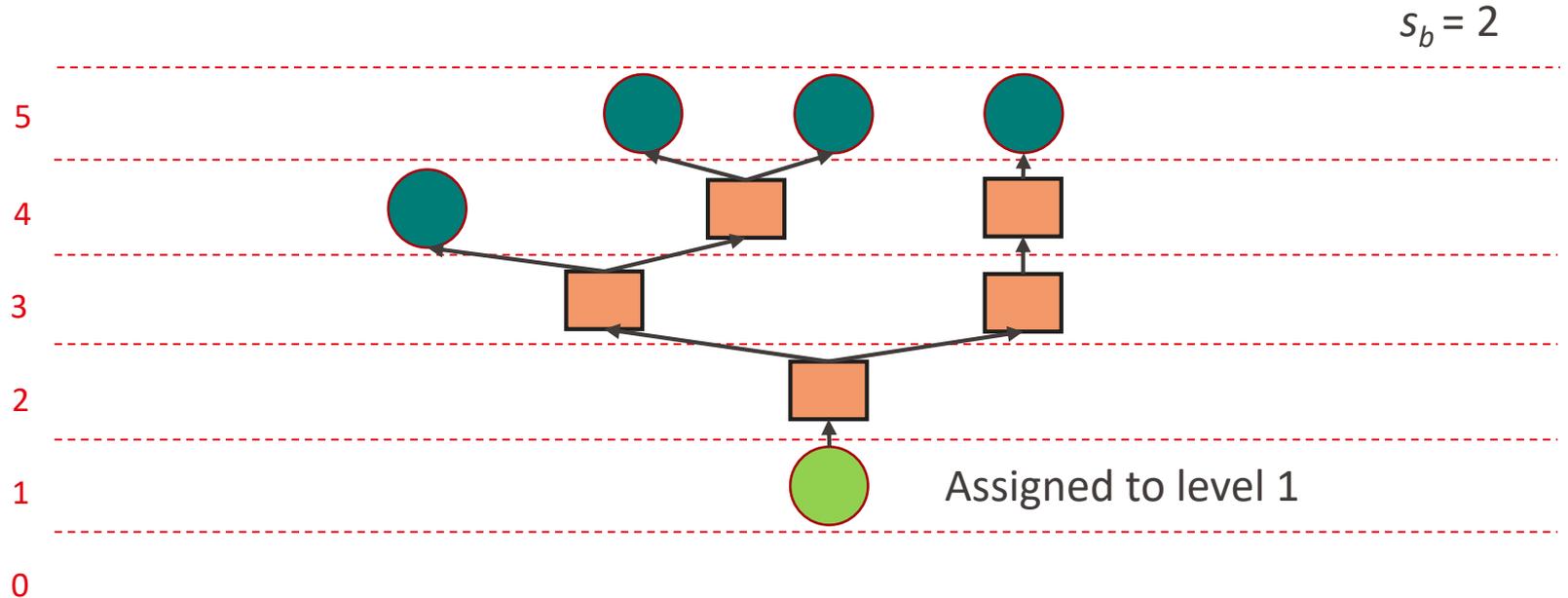
Depth-optimal buffer/splitter insertion

- It generates a **minimum-height splitter tree**

- By applying this algorithm in reverse topological order it generates a **depth-optimal** AQFP circuit → **ALAP** scheduling
  - Assign POs to a level according to an upper bound
  - Construct minimum-height splitter trees in reverse topological order

- Runs in **linear time** w.r.t. the number of gates

Buffer/splitter minimization with retiming

Buffer/splitter minimization with retiming
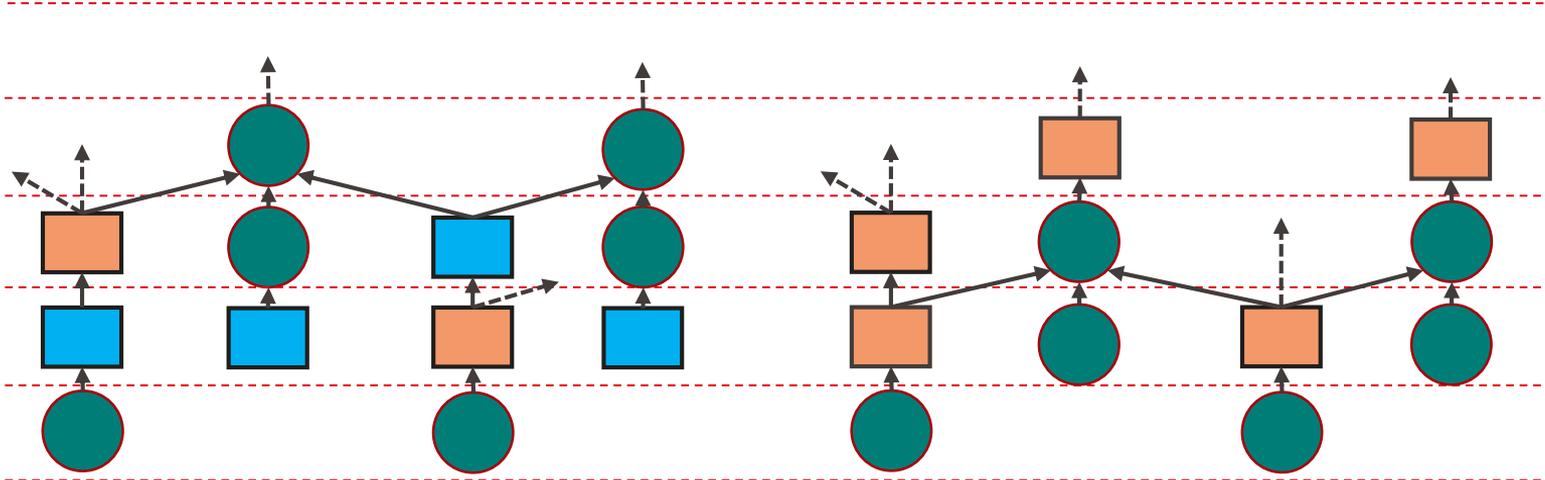
Share buffers and splitters

1 B/S saved

$s_b = 3$

Buffer/splitter minimization with retiming

- Objective of **globally** maximizing the **sharing** of splitters and buffers
  - Previous work on retiming-like approaches is only local on a single wire [2]

- **Minimum-register** retiming [9]
  - Used to minimize globally the number of register
  - Executes in polynomial time
  - Dual to the max-flow/min-cut problem

- Buffers and splitters can be treated like registers **but**:
  - A B/S element that gets relocated transfers its fanout to its fanin gate
  - The relocation is limited by the fanout constraints

[2] Christopher L Ayala, Ro Saito, Tomoyuki Tanaka, Olivia Chen, Naoki Takeuchi, Yuxing He, and Nobuyuki Yoshikawa, "A semi-custom design methodology and environment for implementing superconductor adiabatic quantum-flux-parametron microprocessors", Superconductor Science and Technology 33, 5 (2020).
[9] Charles E. Leiserson and James B. Saxe, "Retiming Synchronous Circuitry," Algorithmica 6, 1991, 5–35

Buffer/splitter minimization with retiming

1. Select a subset of B/S elements to be retimed
   - Their relocation does not violate the fanout constraints
2. Each B/S element is a source and sink of a flow
3. Compute the minimum cut
4. Reposition the B/S elements to the minimum cut
5. Repeat until no more improvement

Mapping to AQFP

Mapping to AQFP

1. Start from a majority-inverter graph (MIG) [3]
   - Abstracts the technology
2. Depth-optimal B/S insertion
3. B/S retiming
4. Loop:
   1. Chunk movement [8] → chunk of nodes are moved up or down
   2. B/S retiming

[3] Luca Amarú, Pierre-Emmanuel Gaillardon, and Giovanni De Micheli, "Majority-Inverter Graph: A New Paradigm for Logic Optimization," IEEE Trans. CAD 2016
[8] Siang-Yun Lee, Heinz Riener, and Giovanni De Micheli, "Beyond local optimality of buffer and splitter insertion for AQFP circuits," In Proc. DAC 2022
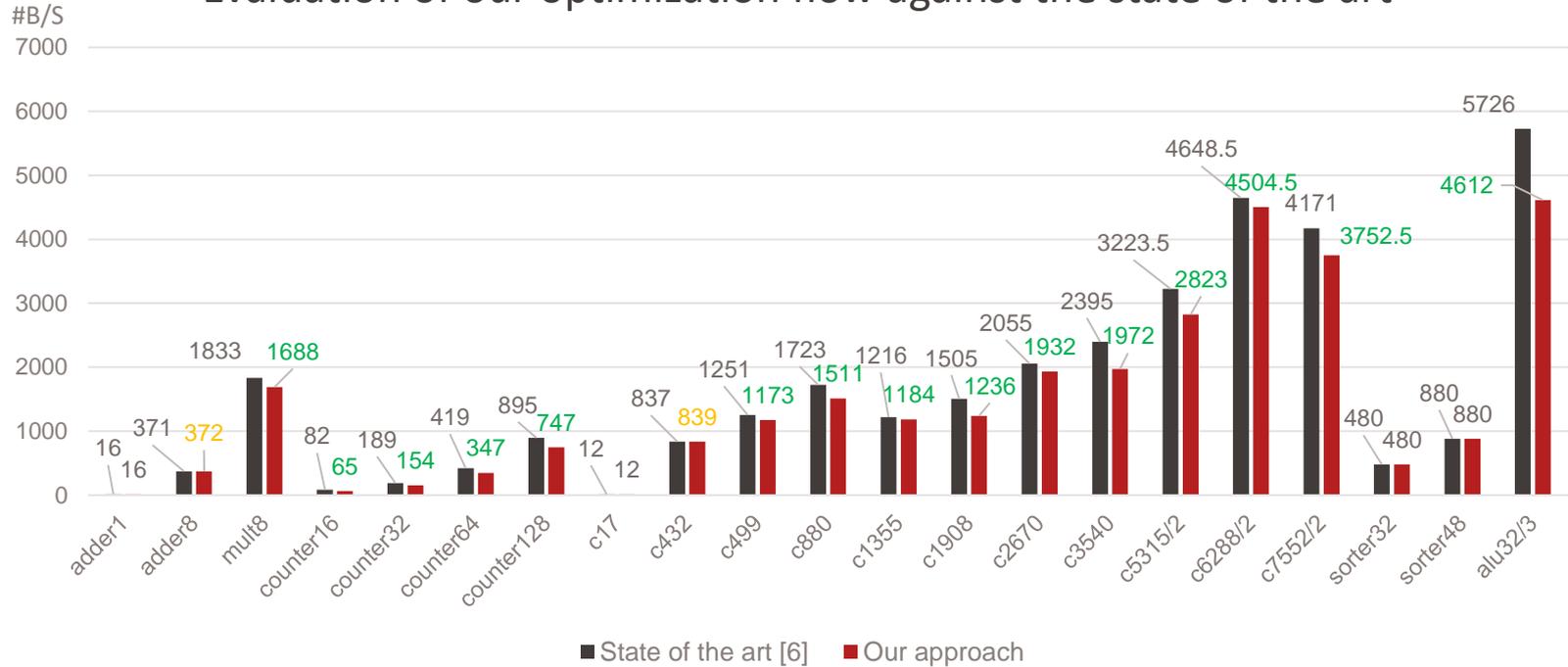
Open Source Implementation

- The code has been implemented in C++ in the library **Mockturtle**
  https://github.com/lsils/mockturtle

- Tools for mapping and optimizing MIG

- Tools for AQFP optimization and verification

Experiments

Evaluation of our optimization flow against the state of the art

A. Tempia Calvino, G. De Micheli

Depth-optimal Buffer and Splitter Insertion and Optimization in AQFP Circuits

■ State of the art [6]   ■ Our approach

Up to **20.7% reduction** in the number of buffers and splitters

55x speedup

Optimal depth

[7] Chao-Yuan Huang, Yi-Chen Chang, Ming-Jer Tsai, and Tsung-Yi Ho, "An Optimal Algorithm for Splitter and Buffer Insertion in Adiabatic Quantum-Flux-Parametron Circuits," In ICCAD 2021

Mapping to AQFP: EPFL benchmarks

Bench.	Baseline		Depth-optimal B/S insertion				B/S optimization				
	Size	Depth	#B/S	#JJs	JJ depth	Time (s)	#B/S	ΔB/S(%)	#JJs	ΔJJs(%)	Time (s)
adder	384	129	49788	101880	258	0.00	49535	0.51	101374	0.50	0.39
bar	3016	12	2001	22098	20	0.01	2001	0.00	22098	0.00	0.10
div	57300	2217	1881255	4106310	4371	0.87	-	-	-	-	>300
hyp	136109	8762	9035578	18887810	17246	2.78	-	-	-	-	>300
log2	24456	200	129547	405830	379	0.10	86705	33.07	320146	21.11	64.18
max	2413	150	69892	154262	160	0.01	68462	2.05	151402	1.85	1.38
multiplier	19710	133	102005	322270	264	0.08	63414	37.83	245088	23.95	43.50
sin	4303	110	18905	63628	188	0.01	14886	21.26	55590	12.63	4.12
sqrt	23238	3366	1791005	3721438	6628	0.49	1343705	24.97	2826838	24.04	284.10
square	12180	126	89516	252112	251	0.03	63630	28.92	200340	20.54	18.30
arbiter	7000	59	27566	97132	63	0.01	25721	6.69	93442	3.80	1.28
mem_ctrl	42758	73	216927	690402	114	0.27	215202	0.80	686952	0.50	10.55
voter	7860	47	19263	85686	86	0.01	15736	18.31	78632	8.23	0.92

Conclusions

- Contributions:
  - The first **depth-optimal** algorithm to insert buffers and splitters
  - **Global** B/S optimization algorithm based on minimum register retiming
  - An AQFP **mapping and optimization** flow
- Improvements:
  - Up to **20.7% reduction** in the number of buffers and splitters
  - Depth optimality
- Our method scales to large benchmarks
  - Optimization up to 1M buffers and splitters in < 300s

Depth-optimal Buffer and Splitter Insertion and Optimization in AQFP Circuits

Alessandro Tempia Calvino

Giovanni De Micheli

ASP-DAC 2023

Integrated Systems Laboratory, EPFL, Lausanne, Switzerland

alessandro.tempiacalvino@epfl.ch

17th January 2023