

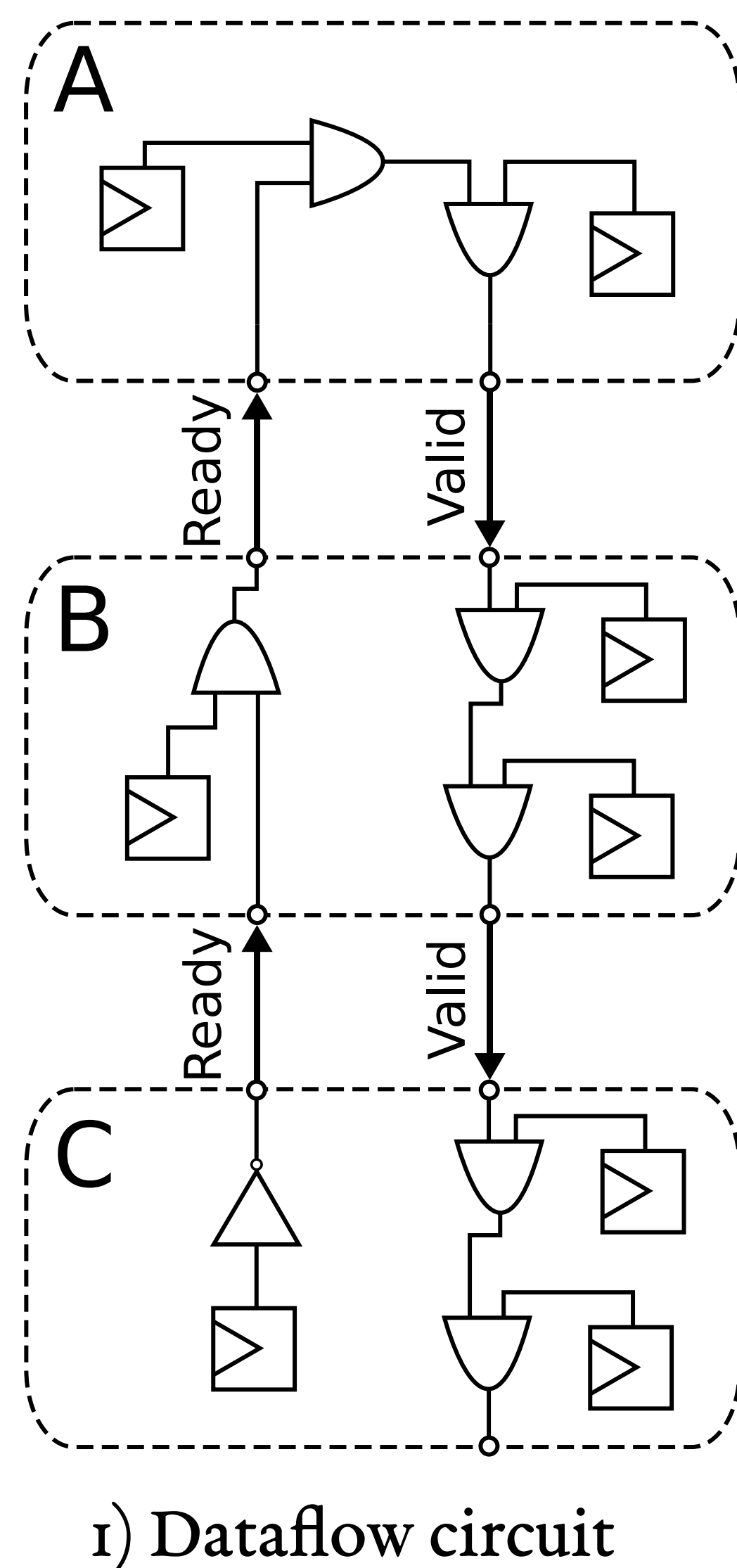
Conquering Timing Unpredictability in High-Level Synthesis

Carmine Rizzi, Advisor: Lana Josipović

ETH Zurich, Zurich, Switzerland

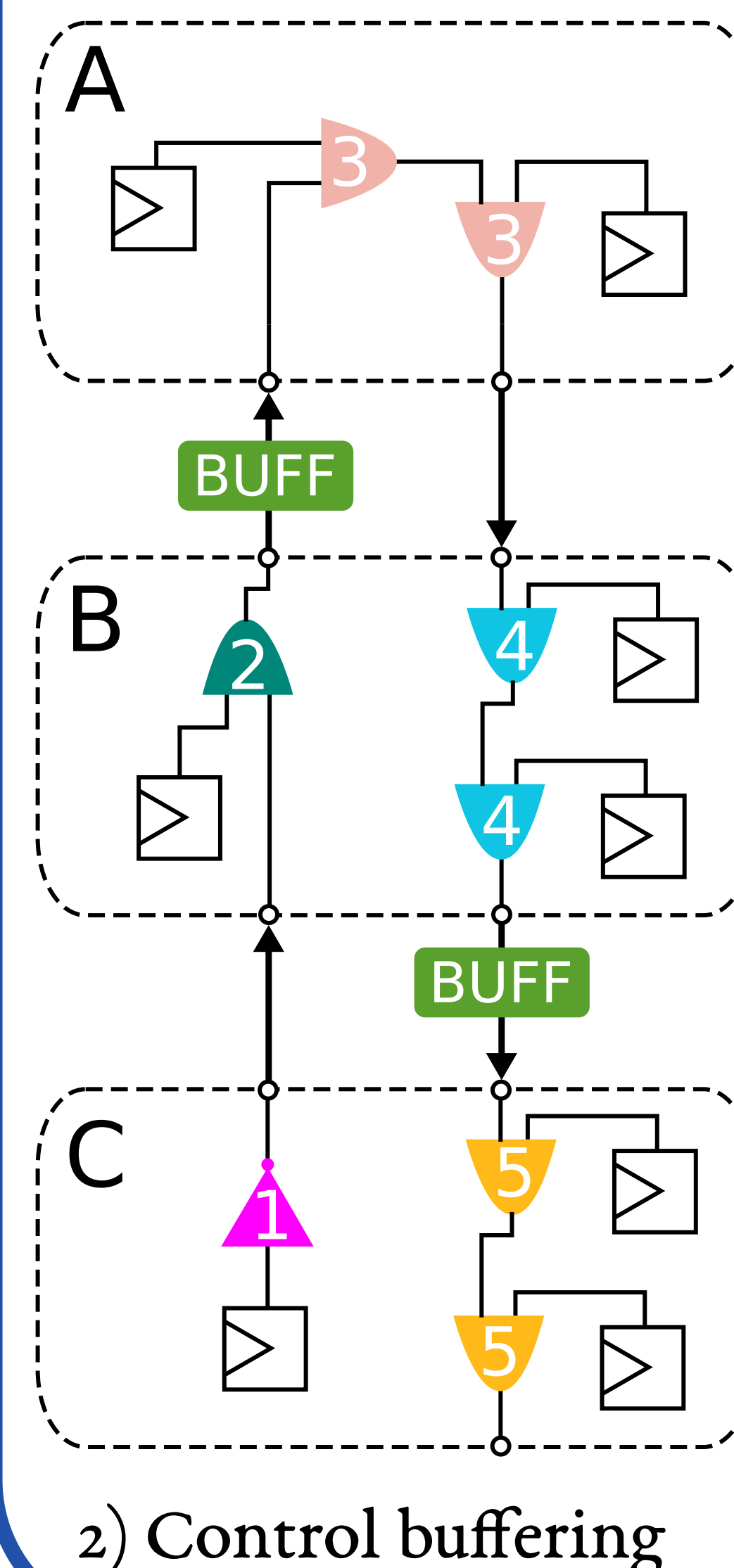
1. Introduction

- *HLS Scheduling* places registers in an untimed circuit based on *pre-characterized* combinational delays.
- In Figure 1, the dataflow circuit units A, B, and C are characterized separately.
- Pre-characterization leads to *frequency gaps* between HLS-produced and engineer-designed RTL circuits.



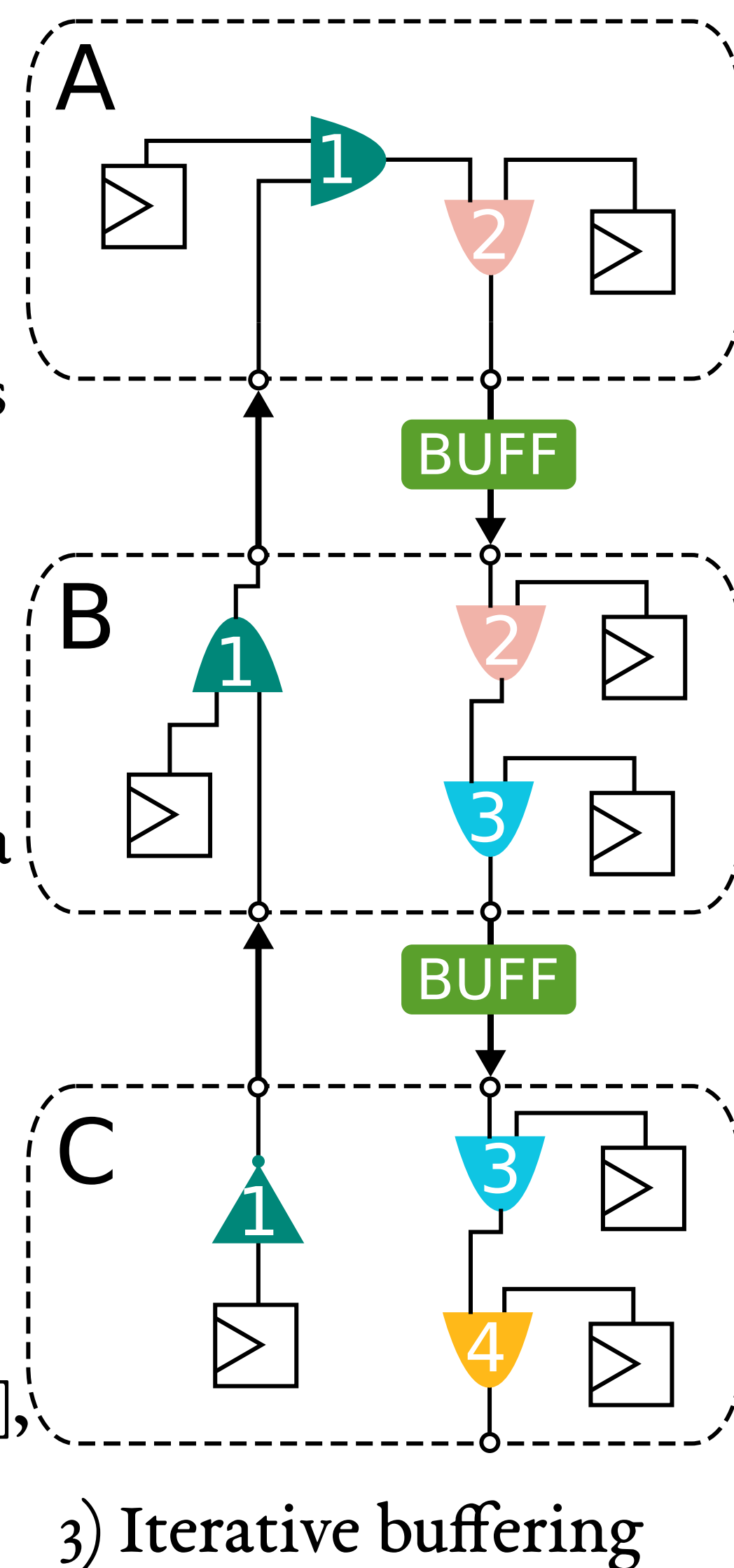
2. Critical Path Is in the Control!

- Prior dataflow pipelining methods focus on datapath delays [1].
- Long combinational paths extend through the *control network*.
- Our model captures datapath and control interactions [2]. Figure 2 shows our scheduling result.
- Our approach reduces the critical path by *up to 38%* over a prior scheduling strategy [1] post-PnR in Vivado 2019.2.



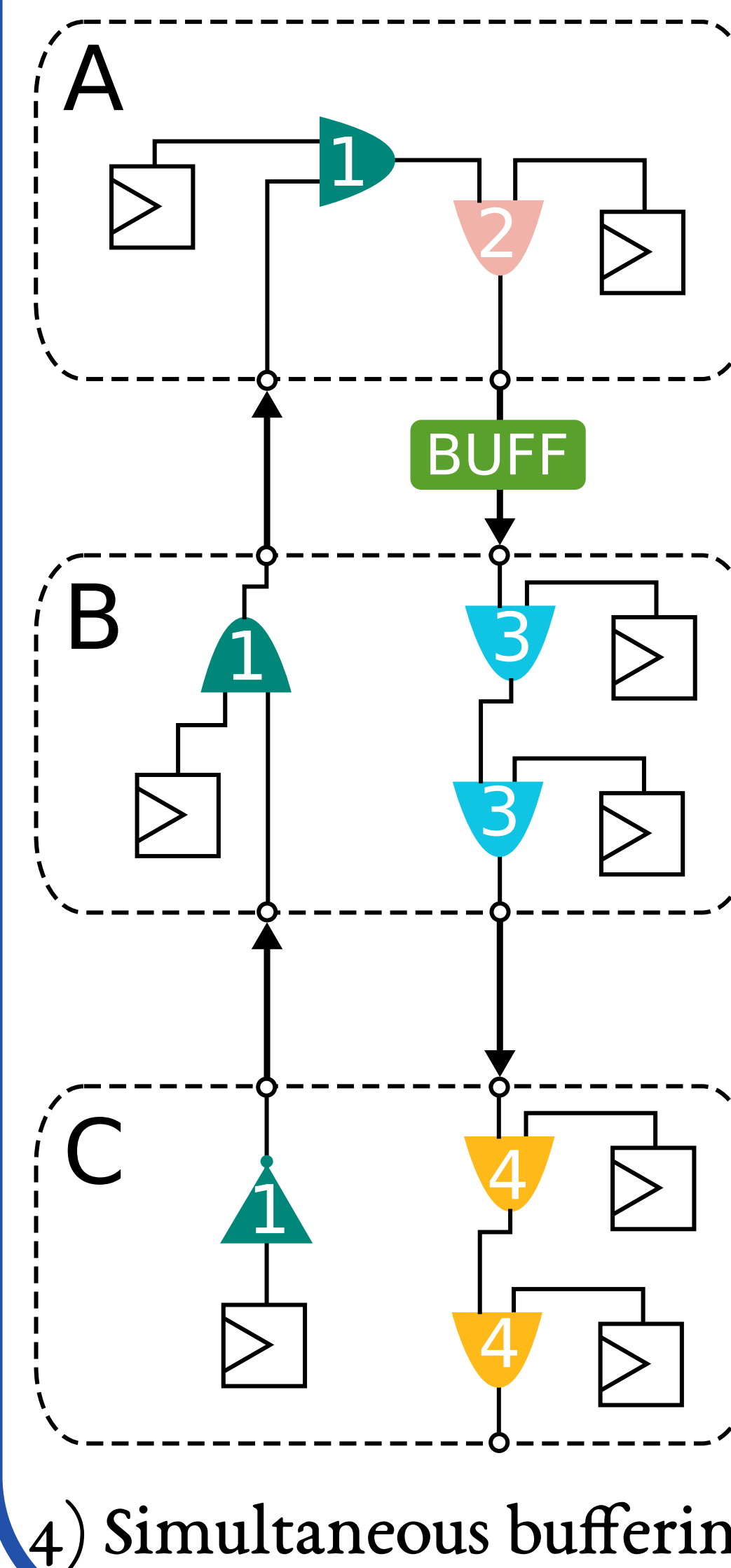
3. Look At the Mapping!

- HLS scheduling overlooks delay changes due to cross-unit logic optimizations.
- Our strategy iteratively extracts and preserves *logic synthesis* results during scheduling [3]. Figure 3 shows the buffering result.
- Our approach lowers execution time and area by *up to 29% and 21%* post-PnR with VTR 8.1.0 compared to a prior scheduler [2].
- If you are curious about efficiently linking a technology mapped circuit to an HLS one [4], come to my talk TS24.3 on Wednesday ;)



4. Simultaneous Is the Way

- The iterative buffering and mapping scheduling technique [3] changes technology mapping results and alter the critical path.
- Our method [5] integrates buffer placement and technology mapping into a unified timing model. Figure 4 shows the result.
- Compared to prior works [2, 3] our approach achieves 25.92% and 13.32% average speedup while using 19.79% and 11.14% fewer FFs post-PnR with VTR 8.1.0.



5. What Is Left to Do?

- The presented methods focus on dataflow circuits. Can we extend it to statically scheduled ones?
- Routing delays from PnR can influence the circuit's performance. How do we account for them during HLS?

References

- [1] Josipović, Sheikhha, Guerrieri, lenne, and Cortadella. "Buffer Placement and Sizing for High-Performance Dataflow Circuits". FPGA'20.
 - [2] Rizzi, Guerrieri, lenne, and Josipović. "A Comprehensive Timing Model for Accurate Frequency Tuning in Dataflow Circuits". FPL'22.
 - [3] Rizzi, Guerrieri, and Josipović. "An Iterative Method for Mapping-Aware Frequency Regulation in Dataflow Circuits". DAC'23.
 - [4] Rizzi, Brunner, Mishchenko, and Josipović. "SimGen: Simulation Pattern Generation for Efficient Equivalence Checking". DATE'25.
 - [5] Wang, Rizzi, and Josipović. "MapBuf: Simultaneous Technology Mapping and Buffer Insertion for HLS Performance Optimization". ICCAD'23.
- Best Paper Award Candidate**

