Profiling an Architectural Simulator

Using FireSim to Profile gem5!

Johnson Umeike, **Neel Patel**, Alex Manley, Amin Mamandipoor, Heechul Yun, Mohammad Alian

Electrical Engineering and Computer Science

THE UNIVERSITY OF KANSAS

Executive Summary

- **Motivation**: software-based architectural simulation is abundant and slow!
- **Observation**: gem5 runs up to 1.7~3.7x faster on a MacBook Pro w/ M1 vs. Dell server w/ Xeon Gold!
- **Goal**: Extensively profile gem5 using hardware performance counters
- **Initial insights**: gem5 is frontend bound and sensitive to L1 size
- **Used FireSim** to compare gem5 performance w/ various L1 size
- **Results**: gem5 performance increases by up to 58% when increasing L1 cache size from 16 to 64KB

Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Introduction

- Software-based architectural simulation is widespread and slow!
  - ✓gem5, Zsim, Sniper,
  - ✓E.g., gem5 simulates at 0.1 to 1MIPS

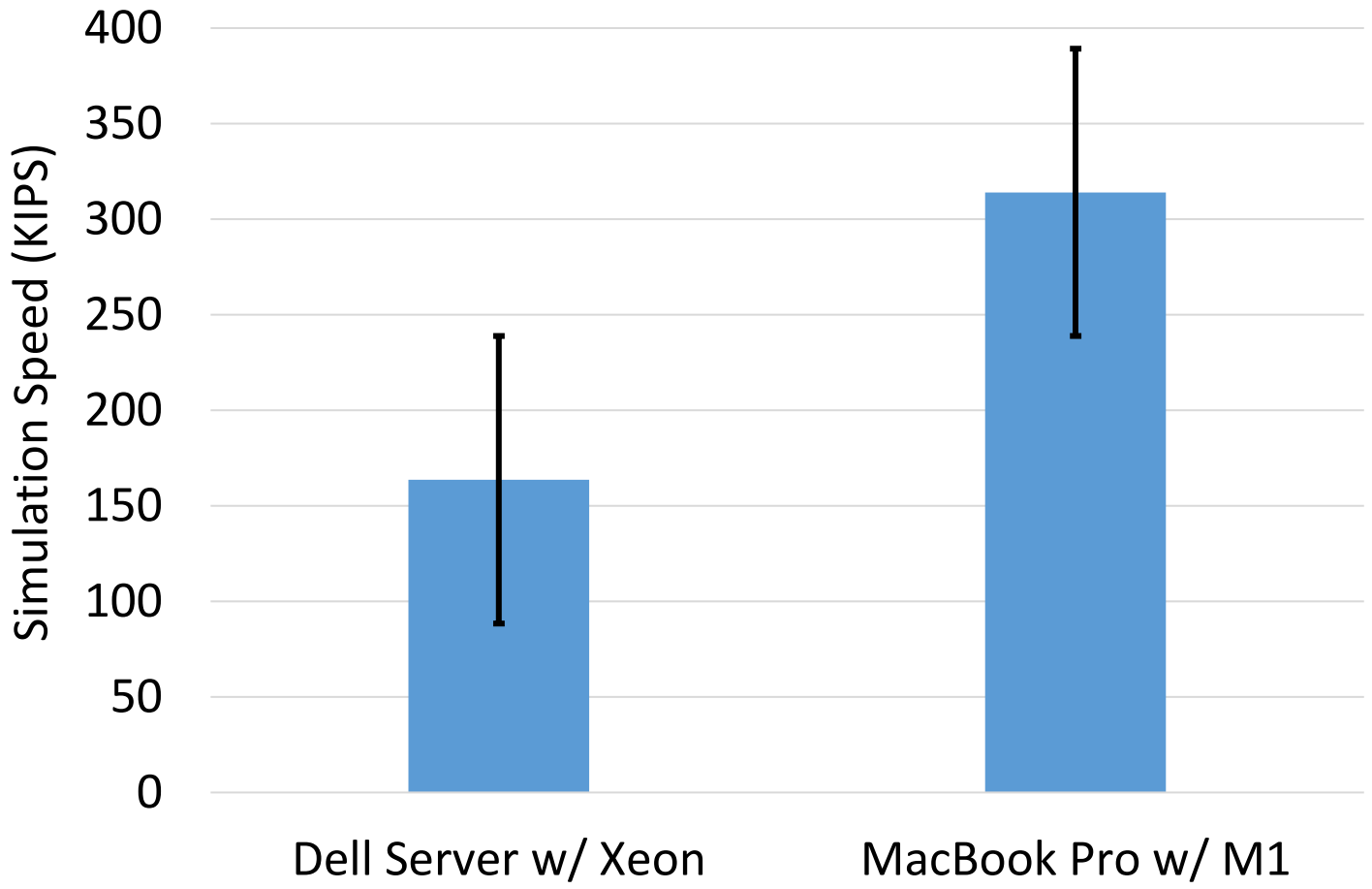
- Many efforts on reducing simulation time:
  - ✓Sampling, checkpointing [SMARTS'03, SimPoint'03, LoopPoint'22]
  - ✓FPGA acceleration [FAME'10, FireSim'18]
  - ✓Parallel/distributed simulation [ZSim'13, dist-gem5'17]
  - ✓Trading off simulation accuracy for speed [SlackSim'09]

Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Observation

- gem5 simulation speed on different platforms



Research Questions?

- Why does gem5 run faster on M1?
- Where are the bottlenecks in running gem5 simulation?
- How much faster can gem5 run by tuning micro-architecture and system configurations?

Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Methodology

- Use gem5 as the application
  - ✓Simulate systems with different CPU types running different PARSEC workloads
- Use hardware performance counters to profile gem5

Parameters	Dell Server w/ Xeon Gold 6242 R	Apple MacBook w/ M1 Chip
Max Frequency	3.1 GHz (4.1 Turbo Boost)	3.2 GHz (Performance Cores)
iCache, dCache	32KB, 32KB	192KB, 128KB
L2, L3	20MB, 35.75MB	12MB, 8MB
DRAM, Total BW	96GB DDR4-2933, 141 GB/sec	8GB LPDDR4X-4266, 68GB/sec

Methodology

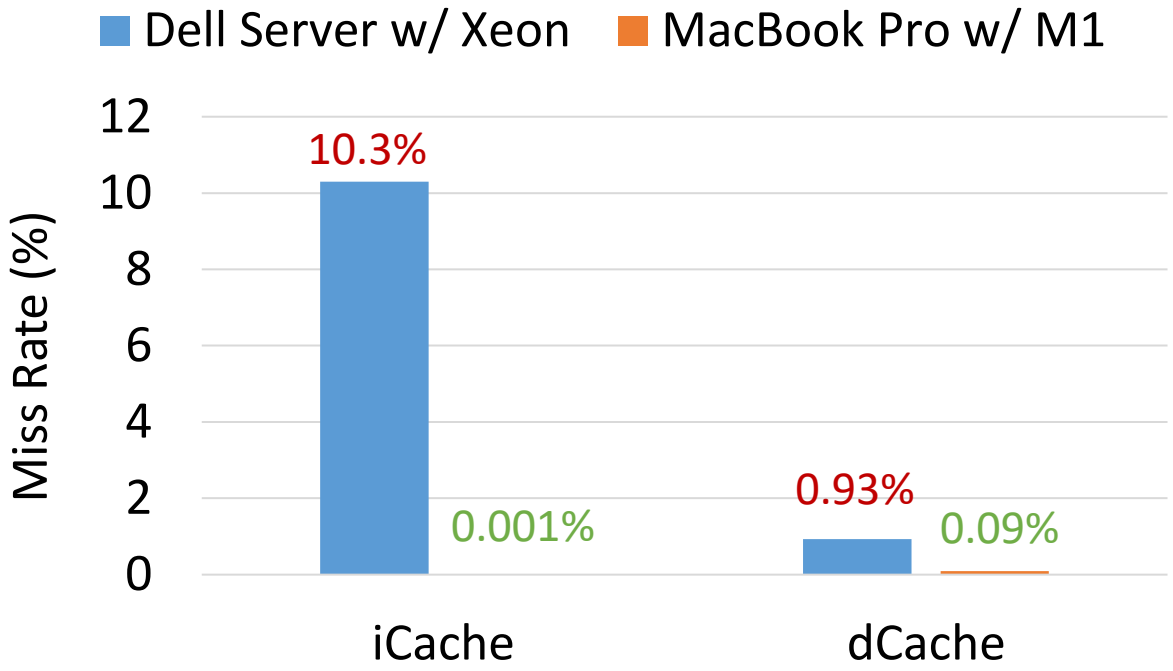
- Use gem5 as the application
  - ✓Simulate systems with different CPU types running different PARSEC workloads
- Use hardware performance counters to profile gem5

Parameters	Dell Server w/ Xeon Gold 6242 R	Apple MacBook w/ M1 Chip
Max Frequency	3.1 GHz (4.1 Turbo Boost)	3.2 GHz (Performance Cores)
iCache, dCache	32KB, 32KB	192KB, 128KB
L2, L3	20MB, 35.75MB	12MB, 8MB
DRAM, Total BW	96GB DDR4-2933, 141 GB/sec	8GB LPDDR4X-4266, 68GB/sec

M1 has 6x larger iCache and 4x larger dCache

Profiling L1 Miss Rate

- M1 has much lower L1 miss rate compared to Xeon



- More profiling results in our upcoming ISPASS paper
  - ✓ J. Umeike, N. Patel, A. Manley, A. Mamandipoor, H. Yun, and M. Alian, "Profiling gem5 Simulator," ISPASS'2023

Hypothesis

- **gem5 simulation speed is very sensitive to L1 size**

- How to prove this hypothesis?
  - ✓Run gem5 on gem5??
    - Too slow!

- FireSim is the solution!

Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Running gem5 on FireSim

1. Set up the AWS FireSim environment
2. Build the gem5 binary for RISCV ISA
3. Prepare gem5 workload and transfer it to the instance
4. Create FireSim workload using FireMarshal
5. Build our target design
6. Modify parameters, tests, and results

Running gem5 on FireSim

1. Set up the AWS FireSim environment
2. Build the gem5 binary for RISCV ISA
3. Prepare gem5 workload and transfer it to the instance
4. Create FireSim workload using FireMarshal
5. Build our target design
6. Modify parameters, tests, and results

Check our website to reproduce steps 1 - 5

Running gem5 on FireSim

1. Set up the AWS FireSim environment
2. Build the gem5 binary for RISCV ISA
3. Prepare gem5 workload and transfer it to the instance
4. Create FireSim workload using FireMarshal
5. Build our target design
6. Modify parameters, tests, and results

Check our website to reproduce steps 1 - 5

Simulated System on FireSim

- quad-core Rocket Chip with an 16KB 4-way set associative icache & dcache, and a 512KB L2 cache base config

Parameter	Value
Design	Rocket Chip
Number of Cores	4
L1 Instruction Cache Size	16KB
L1 Data Cache Size	16KB
L2 Cache Size	512KB
Cache Block Size	64B
Number of Sets	64

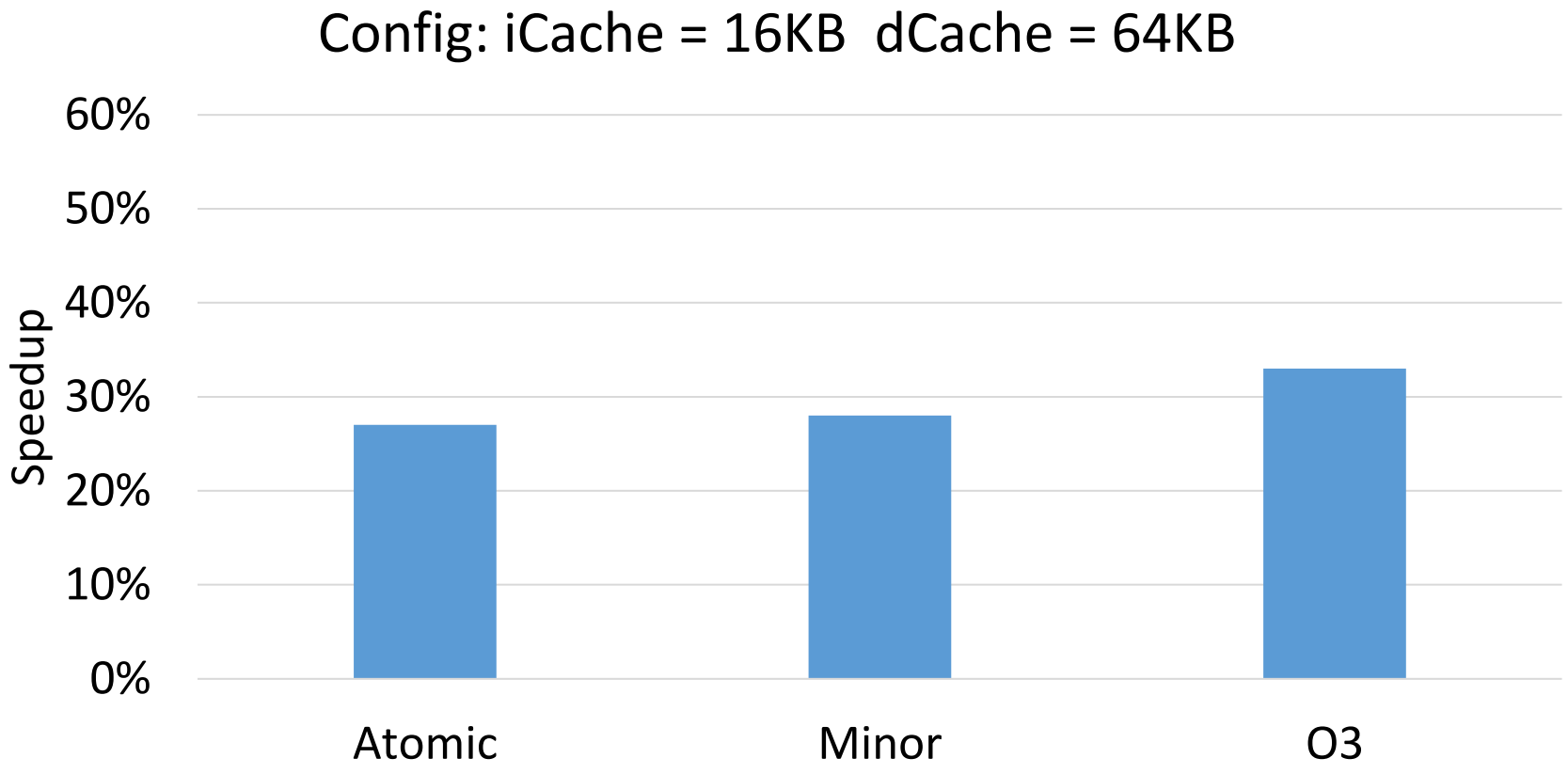
Simulated System on FireSim

- We test the following L1 & L2 cache configurations

Configs	iCache	dCache	L2
1	4-way 16KB	16KB, 4 ways	8-way 512KB
2	4-way 16KB	16-way 64KB	8-way 512KB
3	16-way 64KB	4-way 16KB	8-way 512KB
4	8-way 32KB	8-way 32KB	8-way 512KB
5	4-way 16KB	16-way 64KB	16-way 1024KB
6	4-way 16KB	16-way 64KB	32-way 2048KB
7	16-way 64KB	16-way 64KB	8-way 512KB

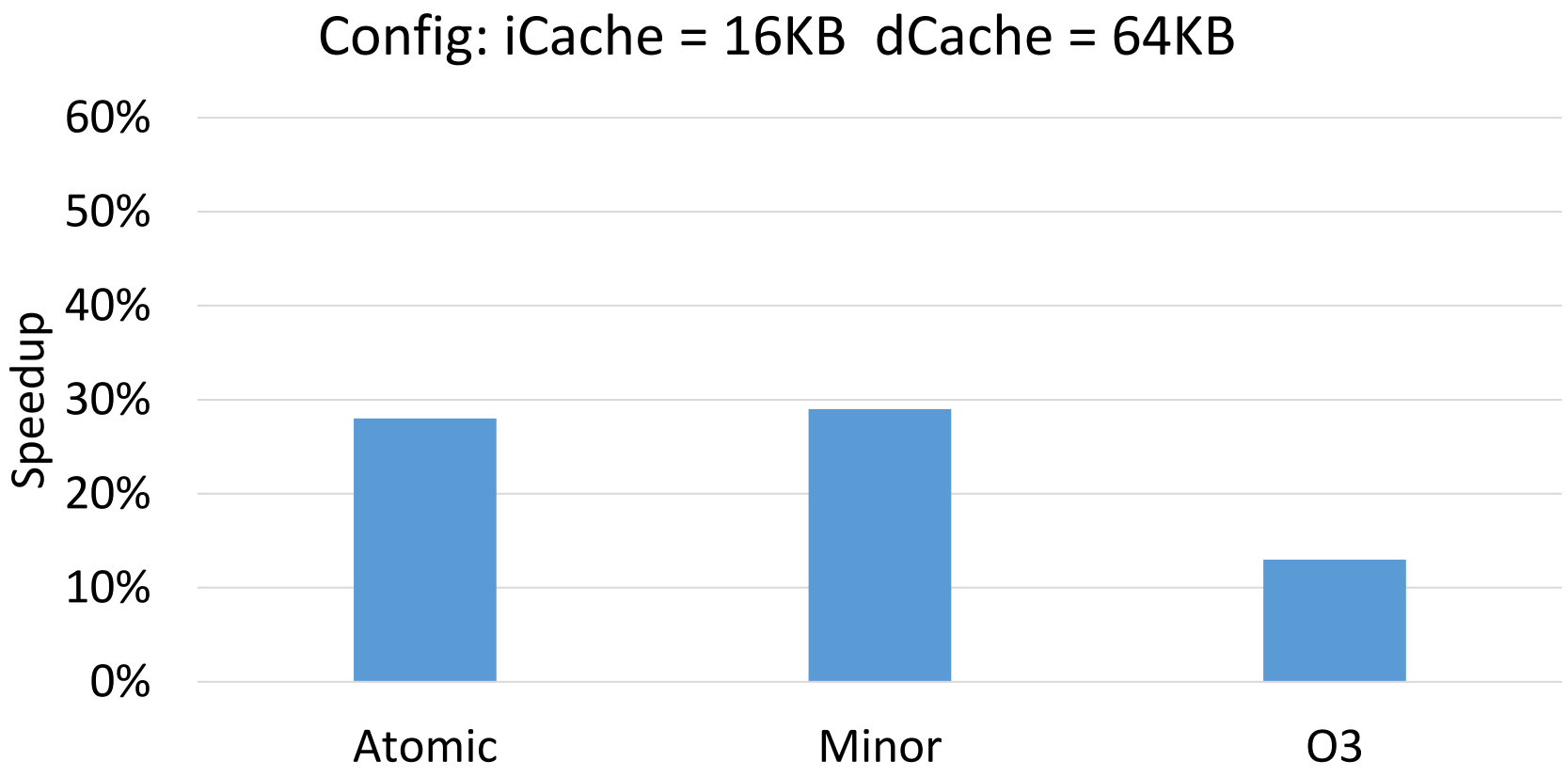
dCache Size x4

- Baseline: iCache and dCache set to 16KB



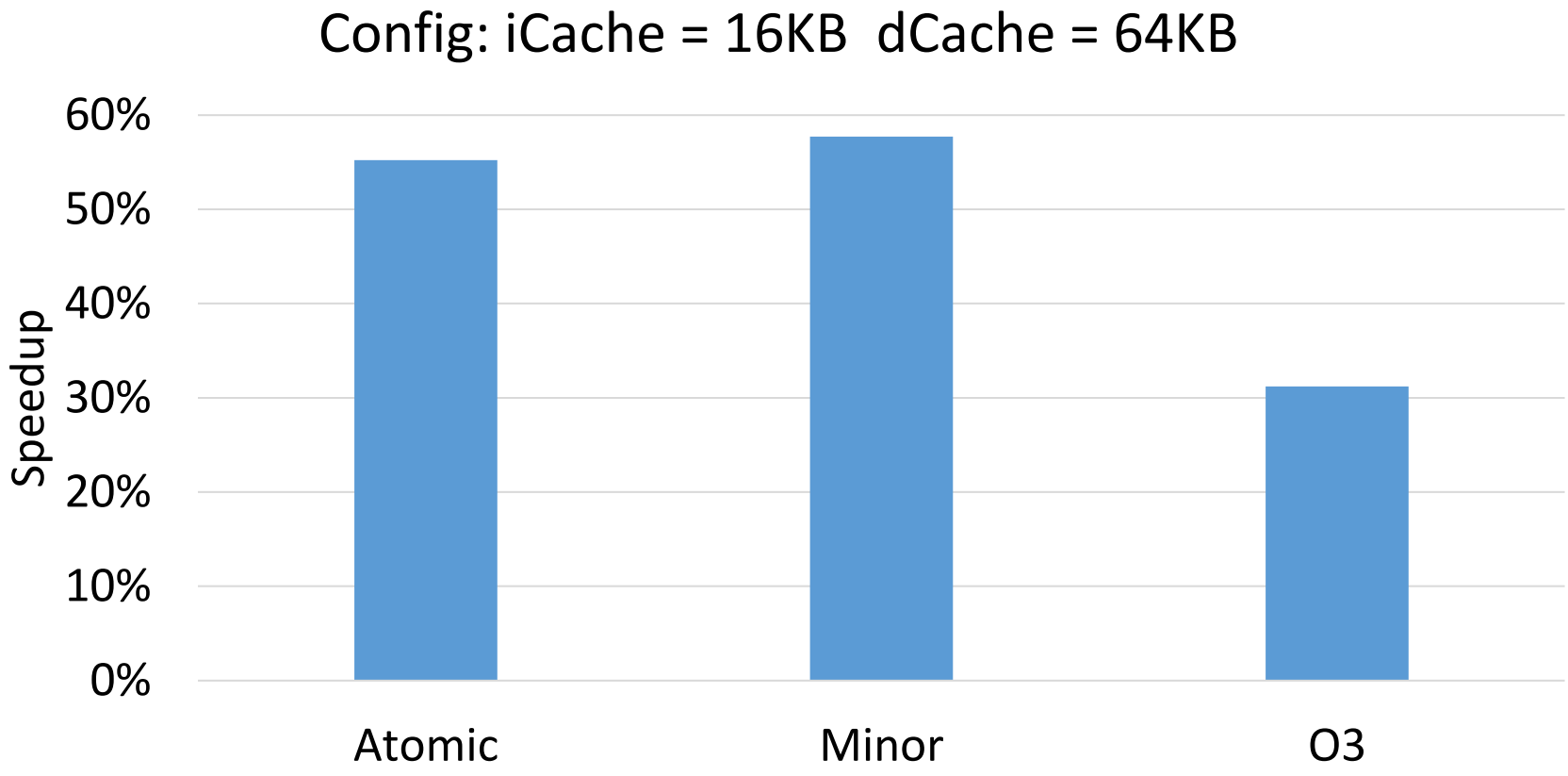
iCache Size x4

- Baseline: iCache and dCache set to 16KB



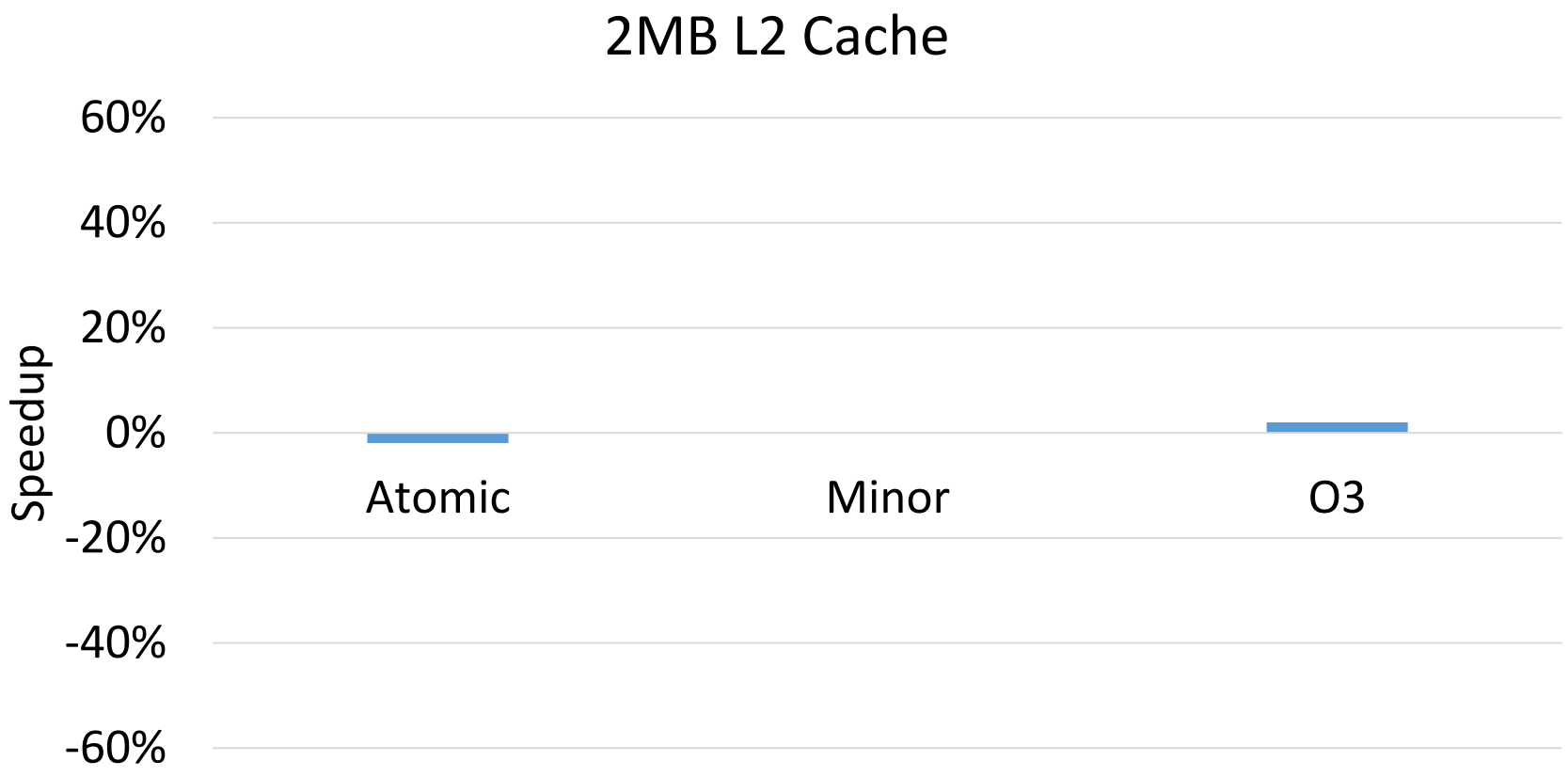
iCache Size x4 & dCache Size x4

- Baseline: iCache and dCache set to 16KB



L2 Size x4

- Baseline: iCache and dCache set to 16KB



Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Challenges and Comments for Using FireSim

- Preparing complex benchmarks on FireSim
  - ✓It took us one week
- FireSim is relatively slow and thus expensive
  - ✓E.g., if the gem5 simulation runs in 2.34 sec on real hardware, running it on FireSim results in (~118x) slowdown
  - ✓Different usage model compared w/ traditional software simulators
- Invalid configurations passed compilation but failed at the AGFI build stage
  - ✓Results in losing time and money
- FireSim is relatively expensive
  - ✓Each failed FPGA image build costs $6 - $52!

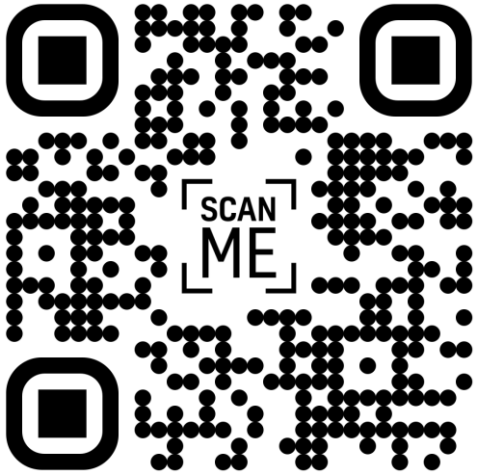
Outline

- Introduction
- Observation and Research Question
  - ✓gem5 runs much faster on M1 platforms!
- Initial Results
  - ✓gem5 is front-end bound
  - ✓M1 has larger L1 cache size
- Using FireSim
  - ✓Sensitivity of gem5 to L1 cache size
- Challenges of Using FireSim
- Conclusion

Summary

- gem5 is frontend bound because of its large instruction footprint
- Using FireSim we showed
  - ✓The size of the iCache and dCache significantly increase gem5's performance
  - ✓L2 size does not impact gem5 performance
- FireSim enabled us to study the sensitivity of gem5 performance to L1 cache size
- Future work: specialized cores for simulation?
  - ✓J. Umeike, N. Patel, A. Manley, A. Mamandipoor, H. Yun, and M. Alian, "Profiling gem5 Simulator," ISPASS'2023

Thank You!



Acknowledgment
This work was supported in part by NSF and ACE, one of the seven centers in JUMP 2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA

Electrical Engineering and Computer Science

KU THE UNIVERSITY OF KANSAS