

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Memory System Performance is Messy

Petar Radojkovic

Barcelona Supercomputing Center

It's the Memory, Stupid!

31st of October 2026.

Context: BSC Memory team (Est. 2014)







- Our core business
  - New memory-related technology → How good is it for HPC applications?
- A relatively simple research question
- The main challenge
  - Provide confidence in your results, i.e., your tools: simulators, analytical models, etc.
- Typical workflow
  - Step 1: Select and validate the tools (3-6 months)
  - Step 2: Perform tech evaluations
  - Btw, no need to publish. Even better if you don't.
- Team
  - 1 PhD student (part-time) → 6/7/8 years to finish the PhD
  - 3-4 engineers

Confidence in the prediction tools

- Actual hardware is the gold standard: You can do a lot of evaluations w.r.t. actual hardware.

  "The one who wants finds a way; the one who does not, finds an excuse."

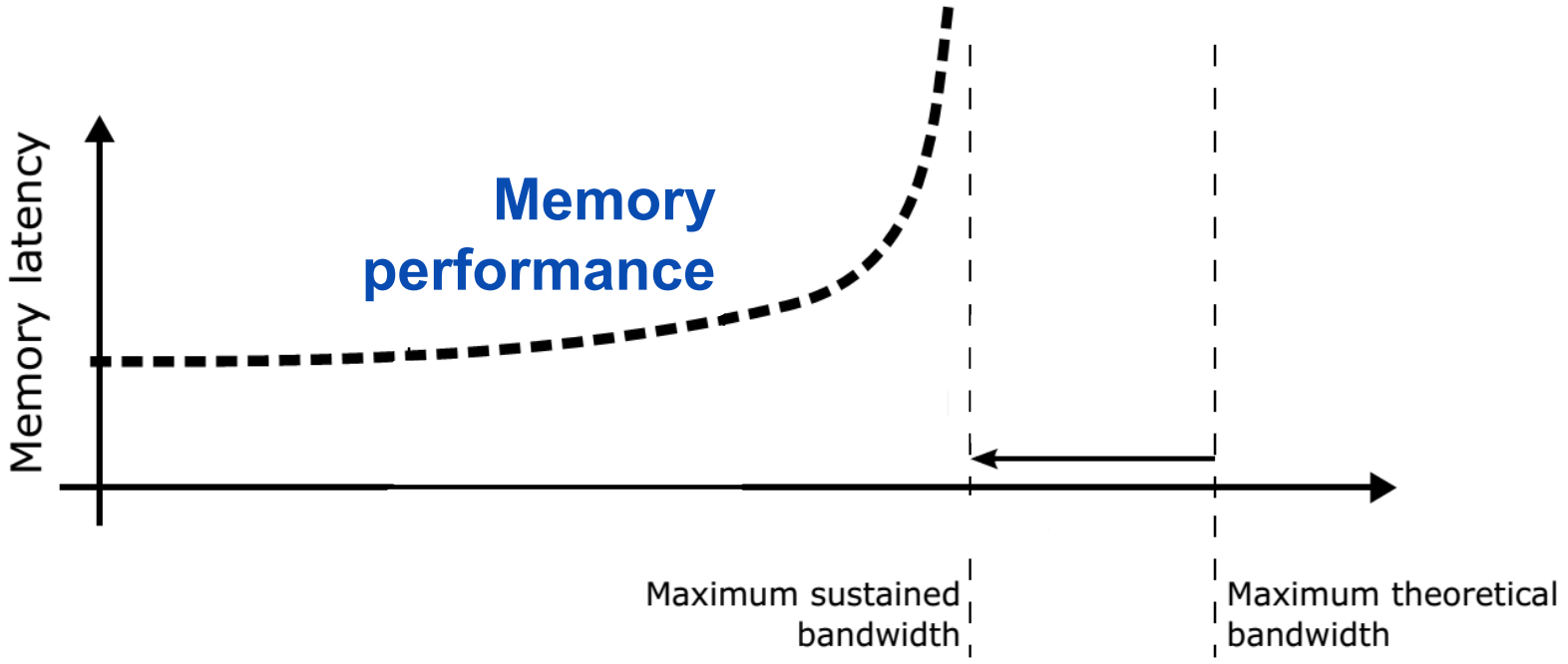
- Example: Intel Sapphire Rapids with 8xDDR5-4800. Dual-socket.
  - Local vs. Remote socket
  - Reduce the memory frequency: DDR5-4800 → xyz
  - Remove some DIMMs
  - Add a CXL memory expander
  - Test in a system with DDR5 + HBM

1. Predict
2. Measure
3. Compare (1) vs. (2)

Quantify the prediction error? 5%, 10%, 20%?
Are trends the same?
What does it even mean to say that "the trends are the same"?

- Then you can start thinking about simulation of a NEW memory system, e.g., STT-MRAM

Memory latency = func(Used memory bandwidth)



B. L. Jacob. The memory system: *You can't avoid it, you can't ignore it, you can't fake it.* Synthesis Lectures on Computer Architecture, 2009.



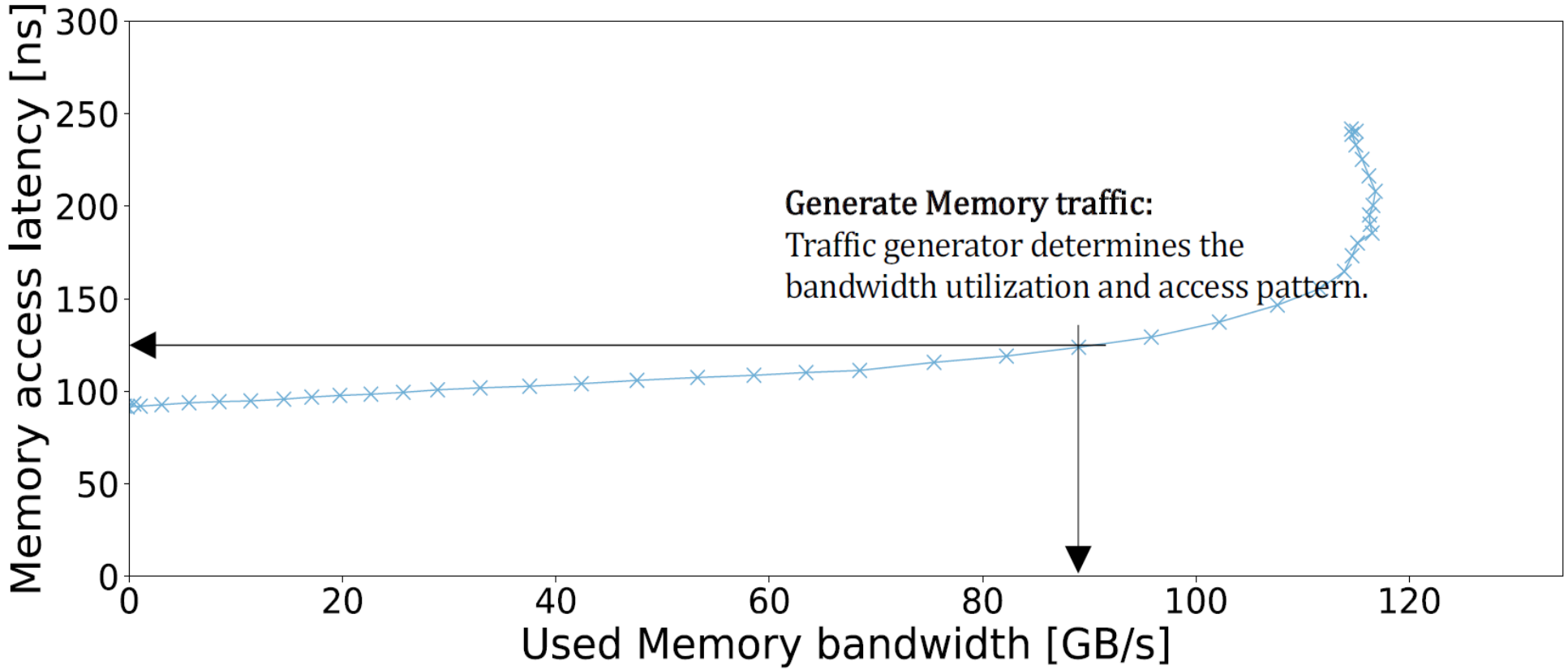
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Memory stress (Mess) benchmark

github.com/bsc-mem/Mess

Let’s make a benchmark that will “plot” the curve



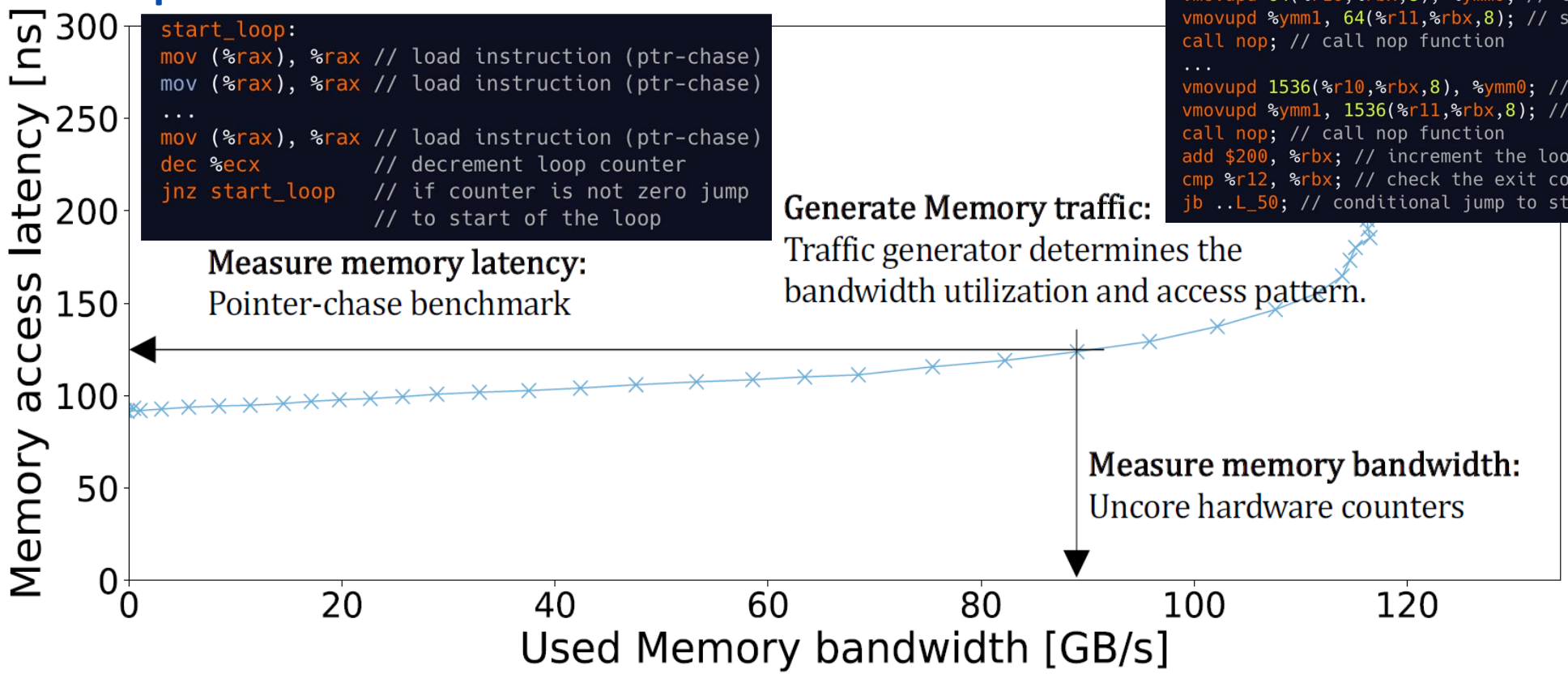
Mess benchmark code “snippet”

Traffic generator:

```
..L_50:
vmovupd (%r10,%rbx,8), %ymm0; // load instruction
vmovupd %ymm1, (%r11,%rbx,8); // store instruction
vmovupd 32(%r10,%rbx,8), %ymm0; // load instruction
vmovupd %ymm1, 32(%r11,%rbx,8); // store instruction
vmovupd 64(%r10,%rbx,8), %ymm0; // load instruction
vmovupd %ymm1, 64(%r11,%rbx,8); // store instruction
call nop; // call nop function
...
vmovupd 1536(%r10,%rbx,8), %ymm0; // load instruction
vmovupd %ymm1, 1536(%r11,%rbx,8); // store instruction
call nop; // call nop function
add $200, %rbx; // increment the loop counter
cmp %r12, %rbx; // check the exit condition
jb ..L_50; // conditional jump to start of the loop
```

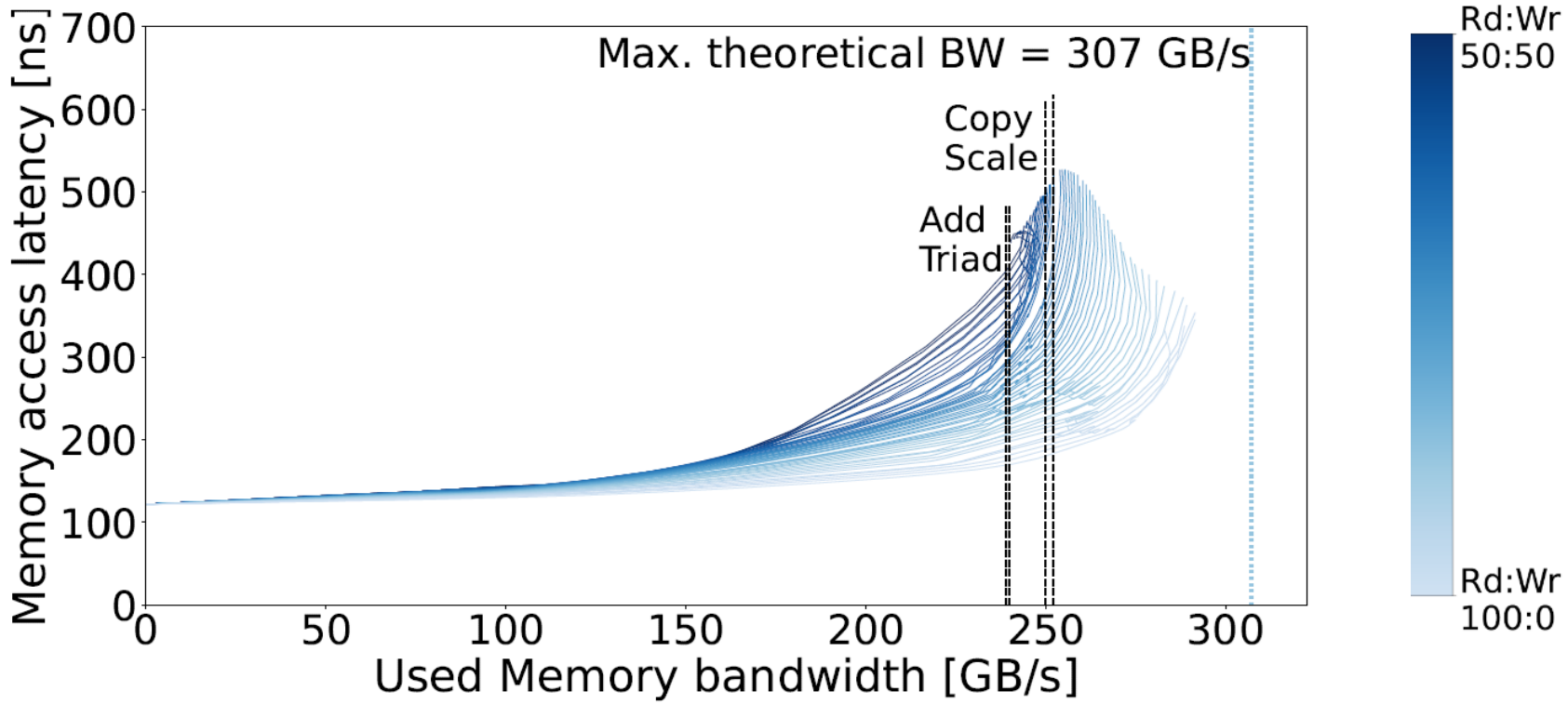
ptr-chase:

```
start_loop:
mov (%rax), %rax // load instruction (ptr-chase)
mov (%rax), %rax // load instruction (ptr-chase)
...
mov (%rax), %rax // load instruction (ptr-chase)
dec %ecx         // decrement loop counter
jnz start_loop   // if counter is not zero jump
                 // to start of the loop
```



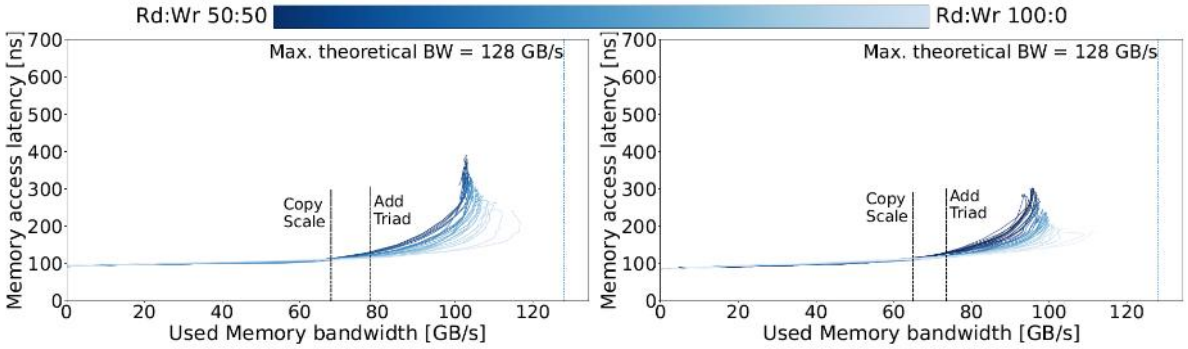
Memory stress (Mess) benchmark

- Connecting the dots



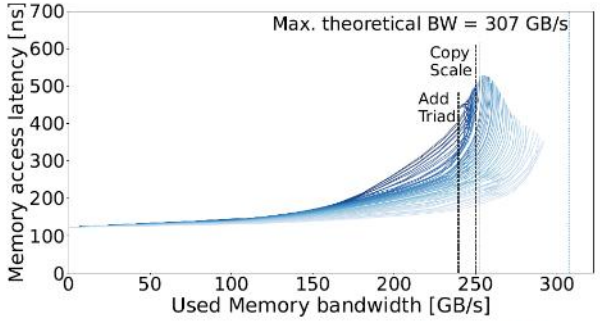
Amazon Graviton3 with 8× DDR5-4800.

Mess benchmark: Actual platforms

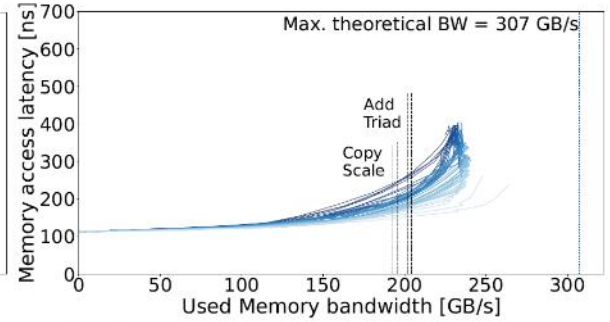


Intel Skylake with 6× DDR4-2666.

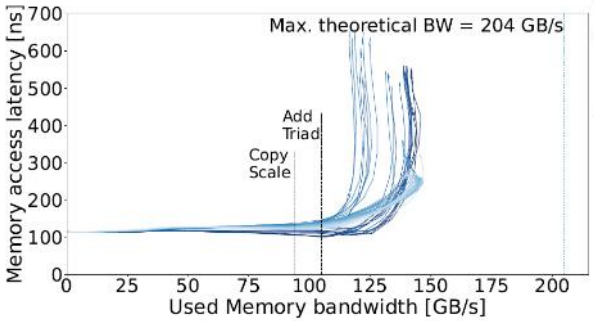
Intel Cascade Lake with 6× DDR4-2666.



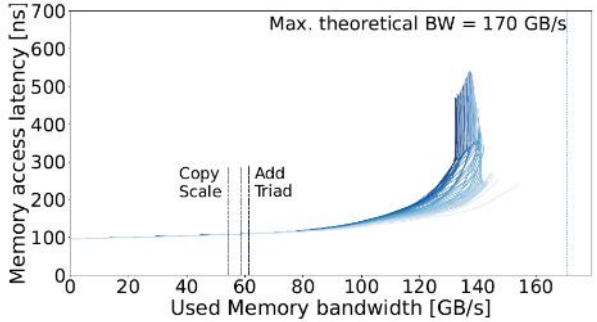
Amazon Graviton3 with 8× DDR5-4800.



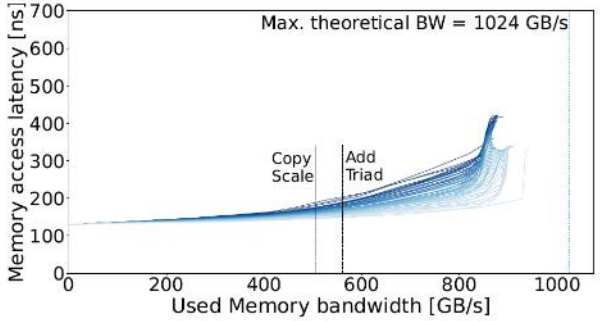
Intel Sapphire Rapids with 8× DDR5-4800.



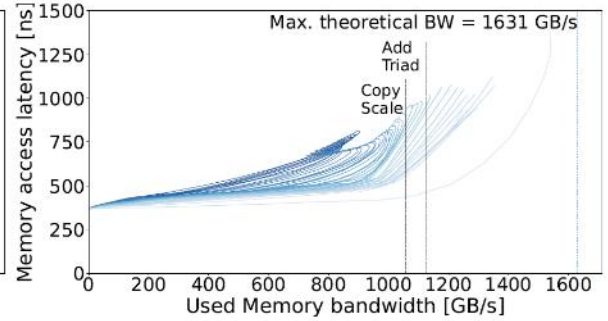
AMD Zen2 with 8× DDR4-3200



IBM Power 9 with 8× DDR4-2666.



Fujitsu A64FX with 4× HBM2.

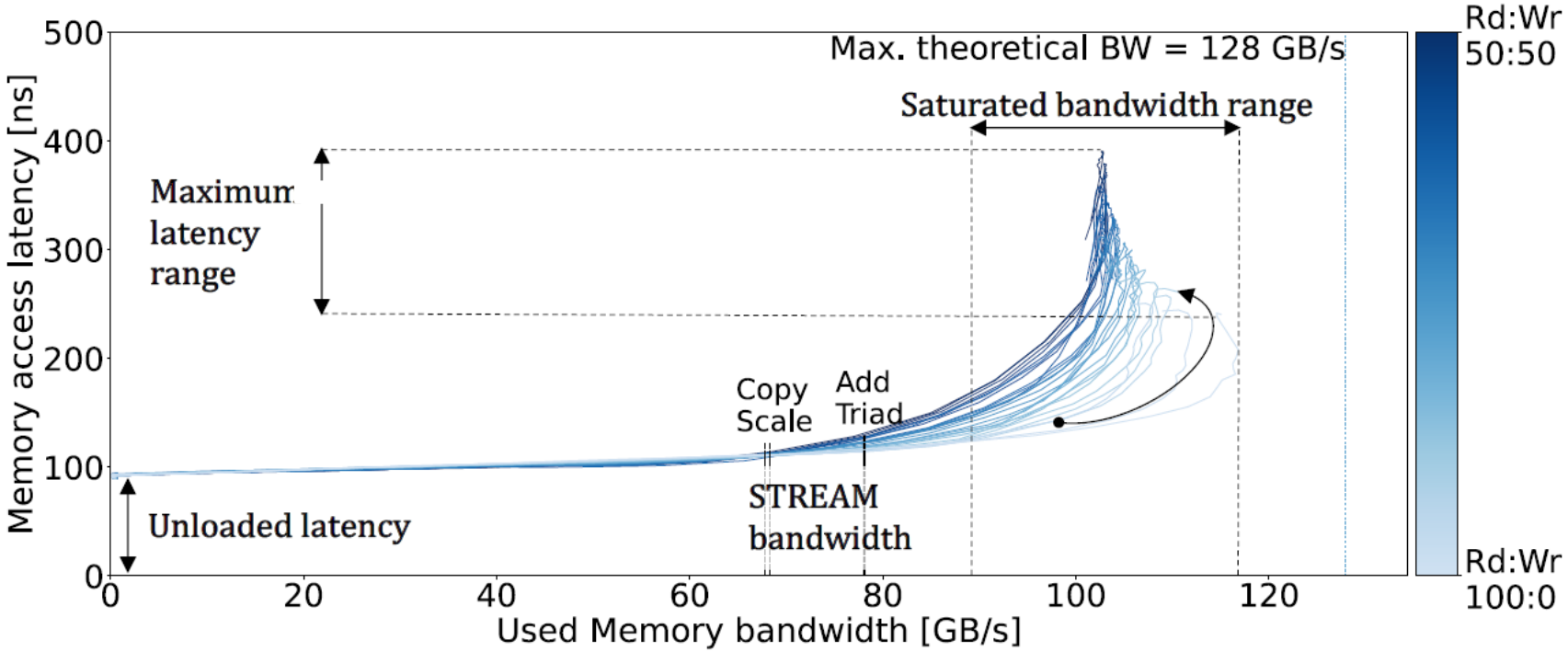


NVIDIA H100 with 4× HBM2E.

- Memory system performance is very, very different

It's the Memory, Stupid!

Mess benchmark



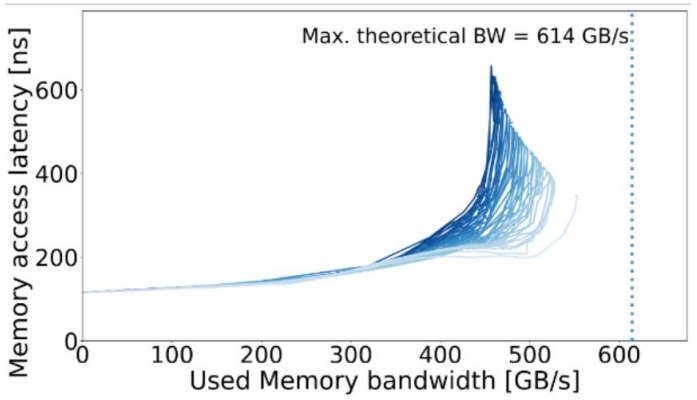
Intel Skylake with 6x DDR4-2666

Mess @ BSC

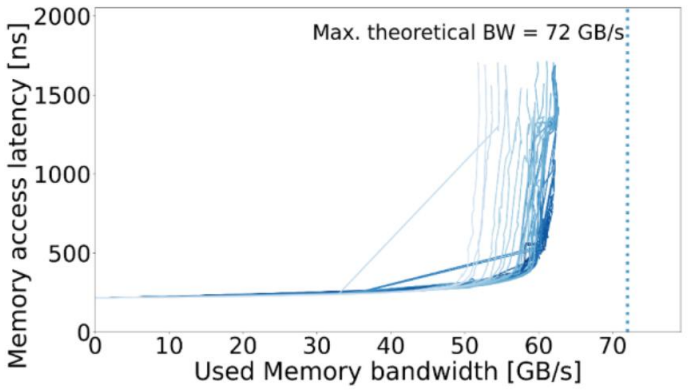
- Memory performance benchmark for any new system
  - HPC server
  - Embedded system
  - Consumer devices
  - Dev Boards
  - …
- Improved usability & Speed
  - Linux, Mac, Windows
  - 8h for a full set of curves: 1000s of data points
    - Smaller subset of curves for validation of simulators

Mess benchmark: Actual platforms

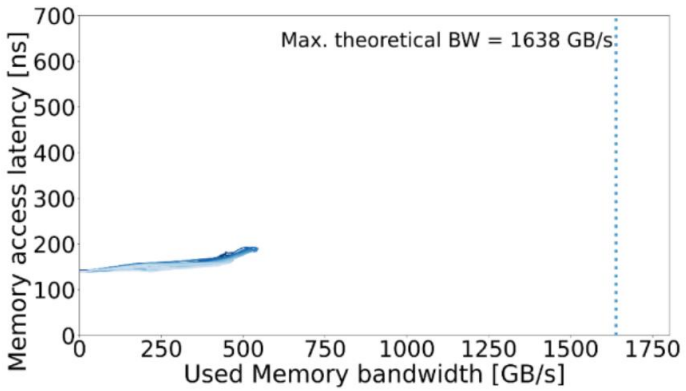
RD:WR 100:0 — RD:WR 50:50



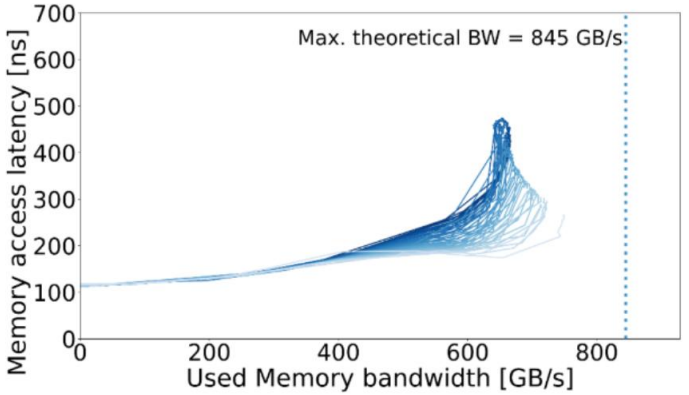
Intel Granite rapids 12x RDIMM 6400



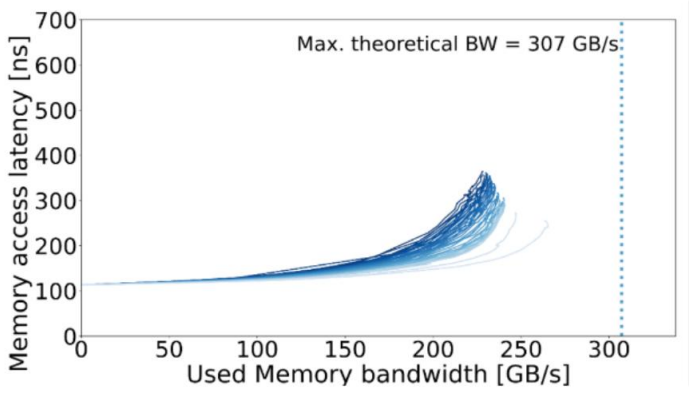
Intel Emerald rapids 2x CZ120



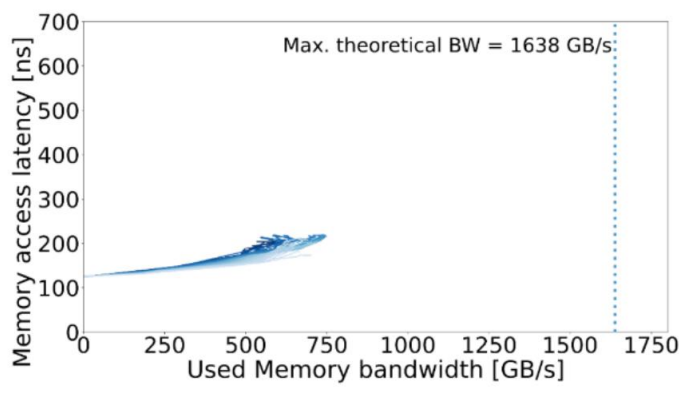
Intel Sapphire rapids 4x HBM2



Intel Granite rapids 12x MRDIMM 8800



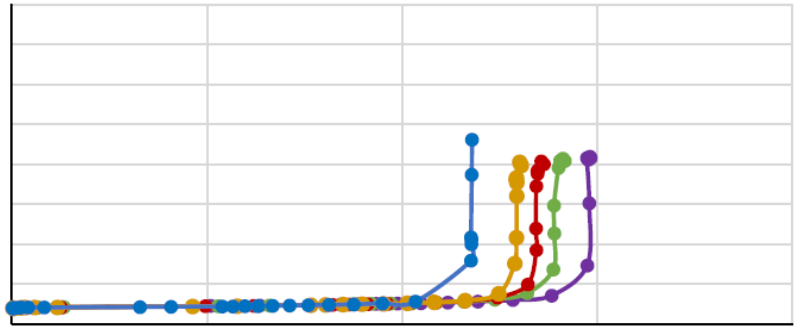
Intel Sapphire rapids 8x 64G DDR5-4800

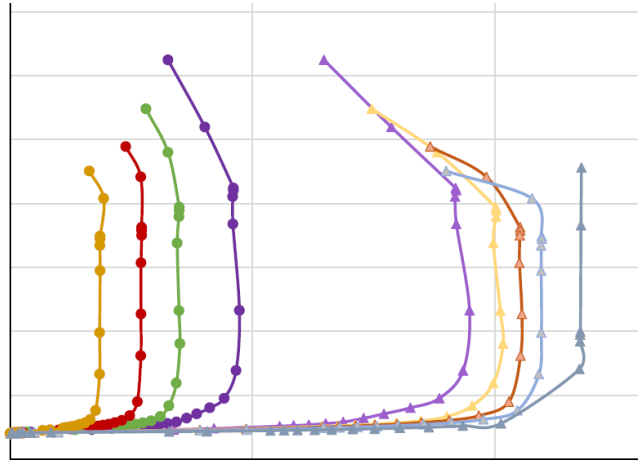


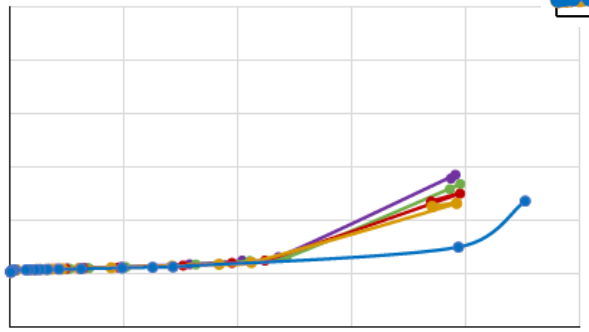
Intel Sapphire rapids 4x HBM2e

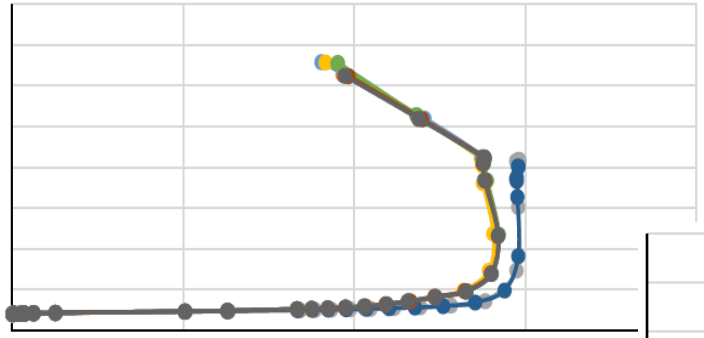
Take care:
Axis ranges vary between the charts

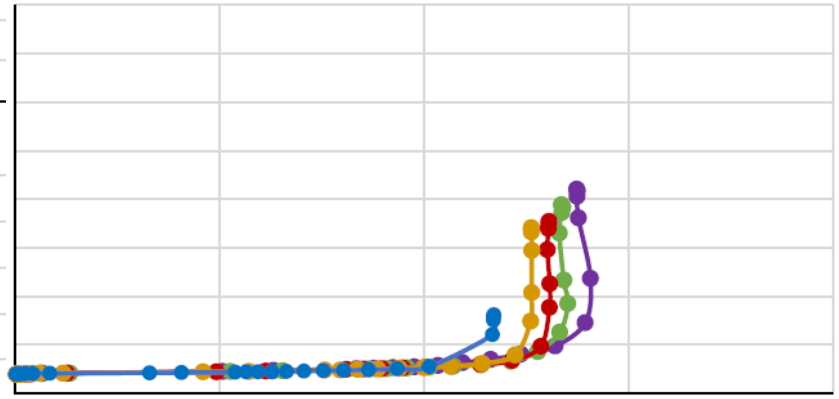
Our partners:
Extensive testing of advanced/future memory devices

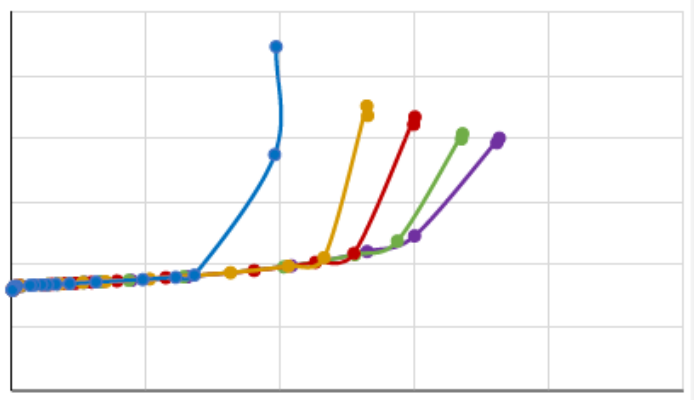


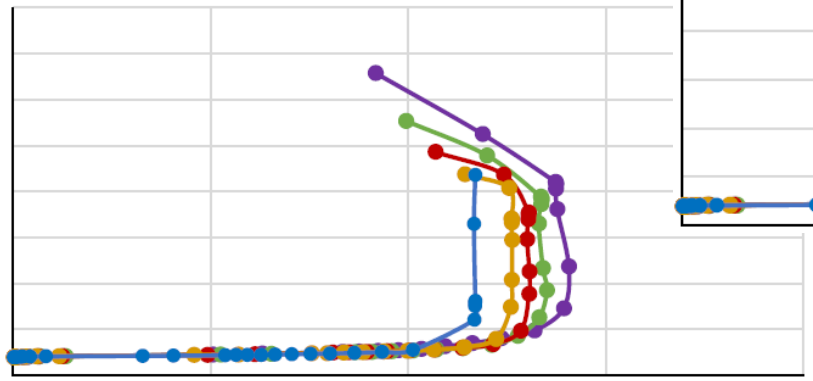














Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Memory System profiling += Application Profiling

Dialogue with HPC users

- *"My application is memory bound …"*

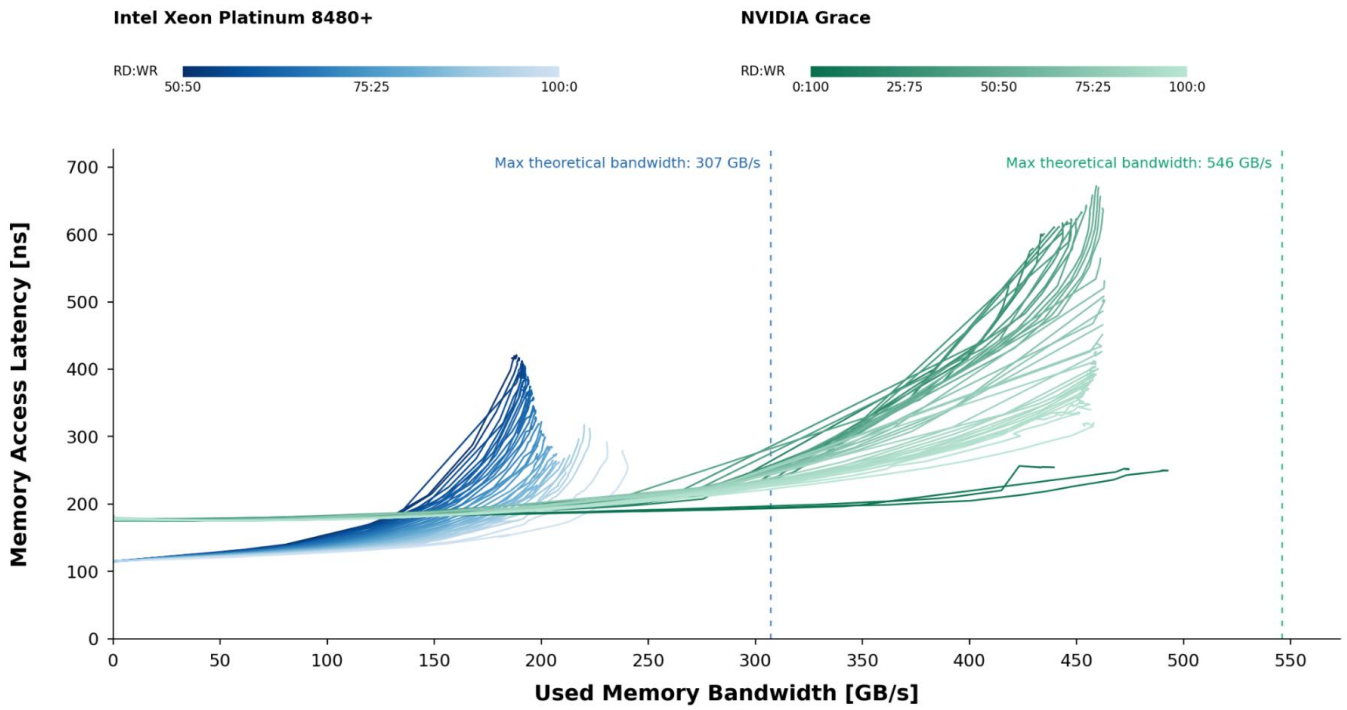
- *"If you give me better memory, my application will run much faster … "*
  - Better memory: Whatever is in the news, e.g., Optane, HBM, CXL expanders
  - Execute the app on the system with this better memory: Oops!
    → I give up!

- Main problems
  - Abstract & Esoteric memory system performance
  - Manufacturers' marketing departments

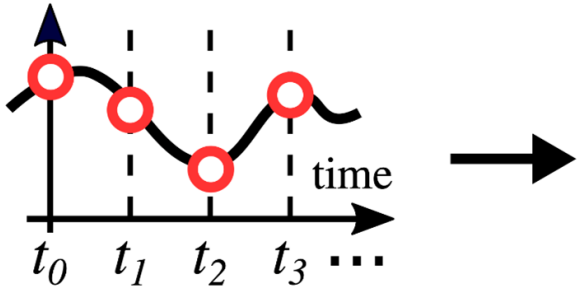


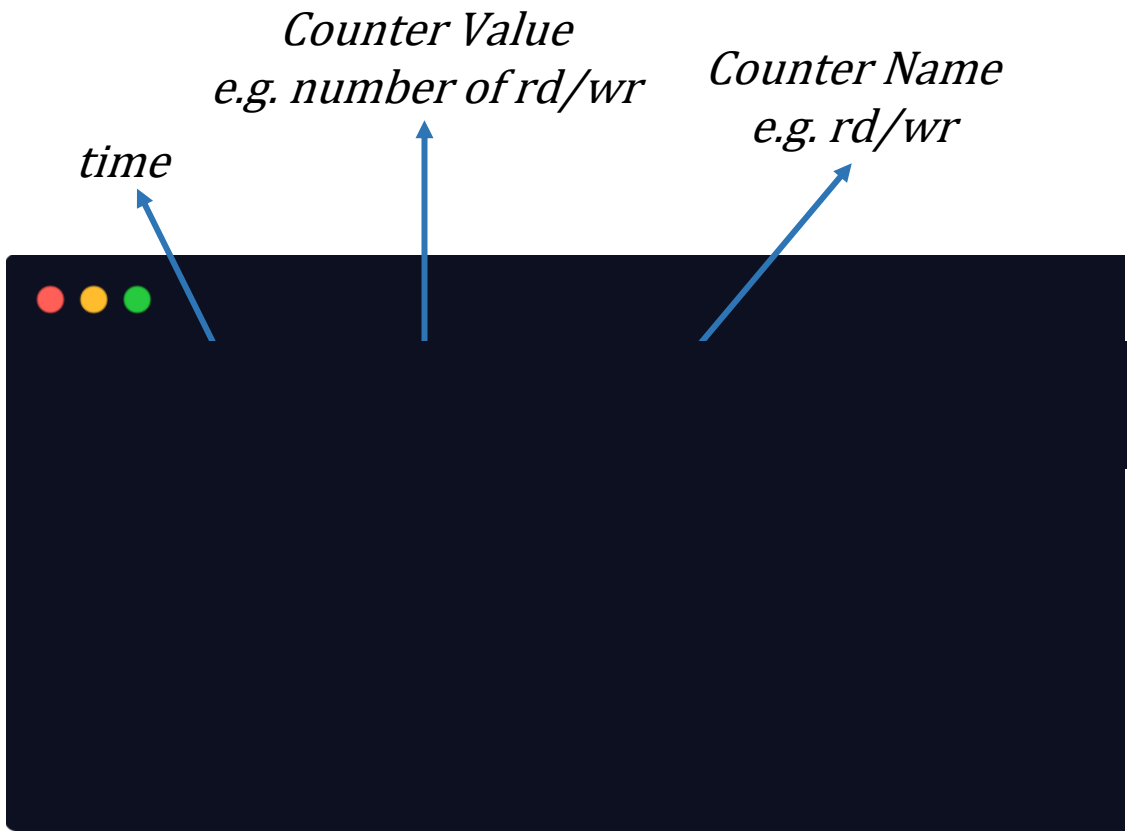
Make memory performance tangible

- MN5-Intel SPR & Nvidia Grace
  - Understand tradeoffs
- Similar story
  - Intel KNL: DDR4 + MCDRAM (HBM)
  - DDR4 + Optane
  - DDR5 + HBM
  - DDR5 + CXL expanders

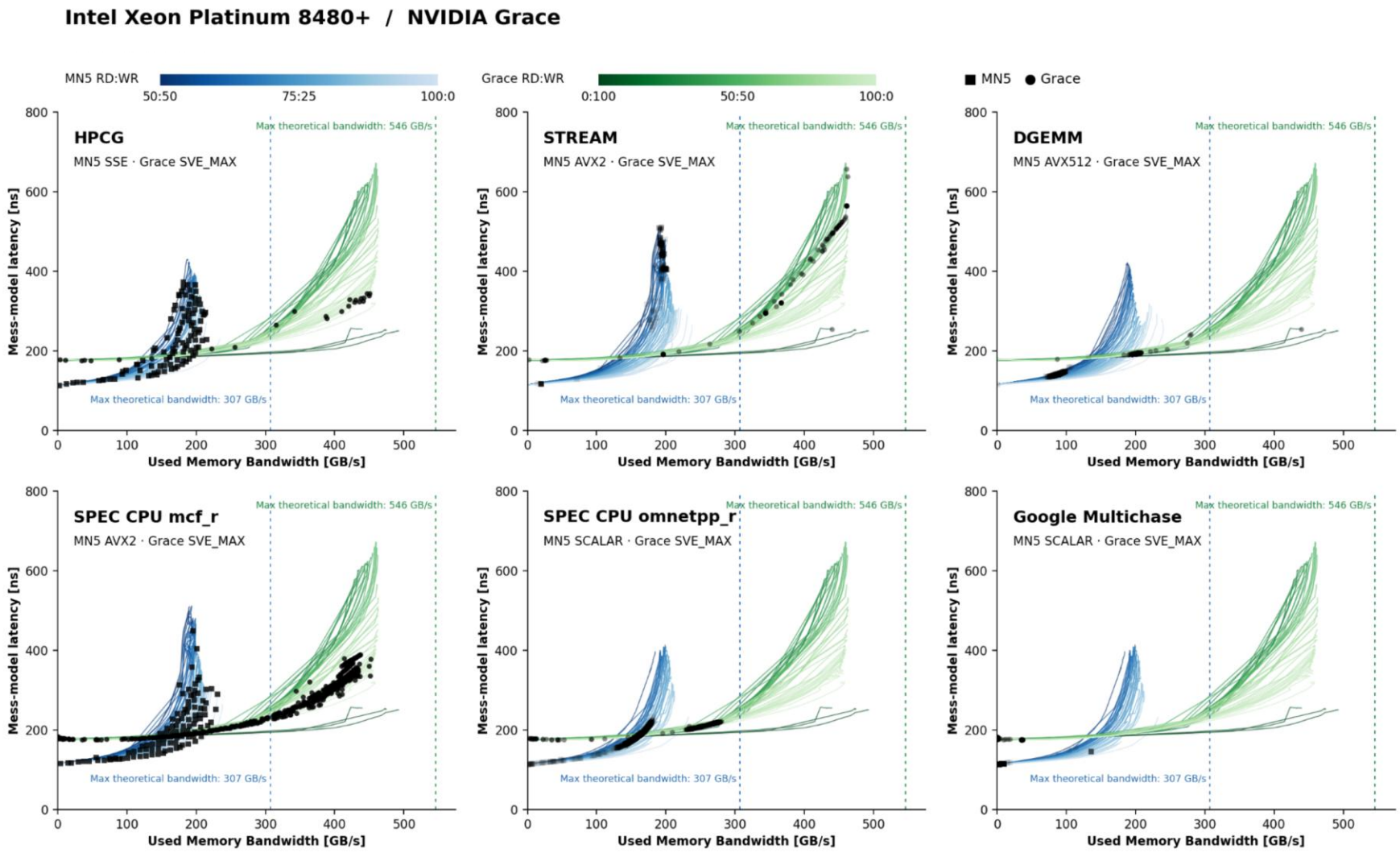


Application profiling





Application profiling: Examples



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

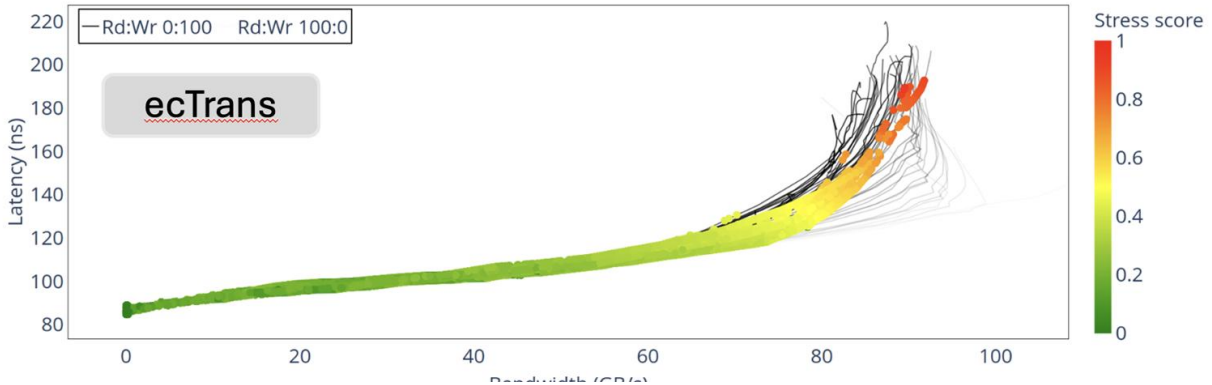
It's the Memory, Stupid!

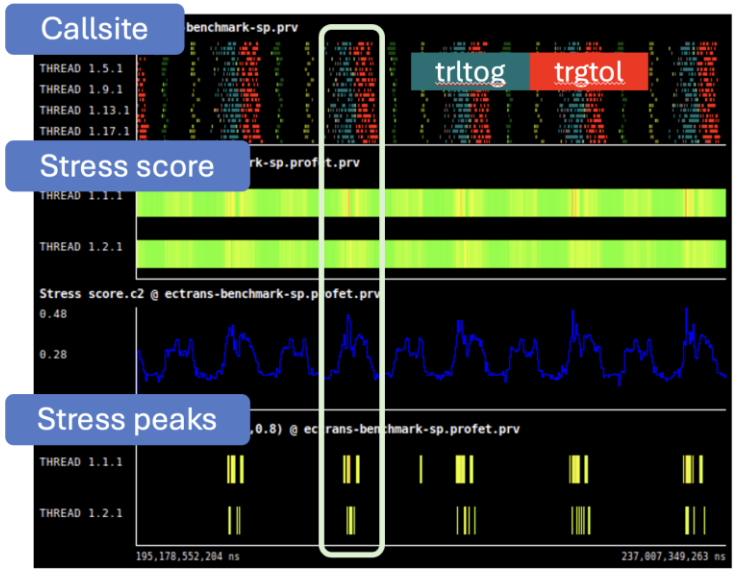
As simple as…

- **Get memory curves** of your favorite server – Mess Benchmark
  - https://github.com/bsc-mem/Mess
  - Or use the already collected curves available at: https://github.com/bsc-mem/Mess-Results
- **Profile your app:** memory read and write traffic
  - Use whatever tool you like: PAPI, perf, Extrae, LIKWID…
- **Visualize** your results
  - Use the Plotter: https://github.com/bsc-mem/Mess/wiki/Plotter-Parser
- You know where your application located on the memory curves

As complex as: HPC production use

- Mess analysis in the Paraver tool
  - Give a memory-related angle to the overall analysis





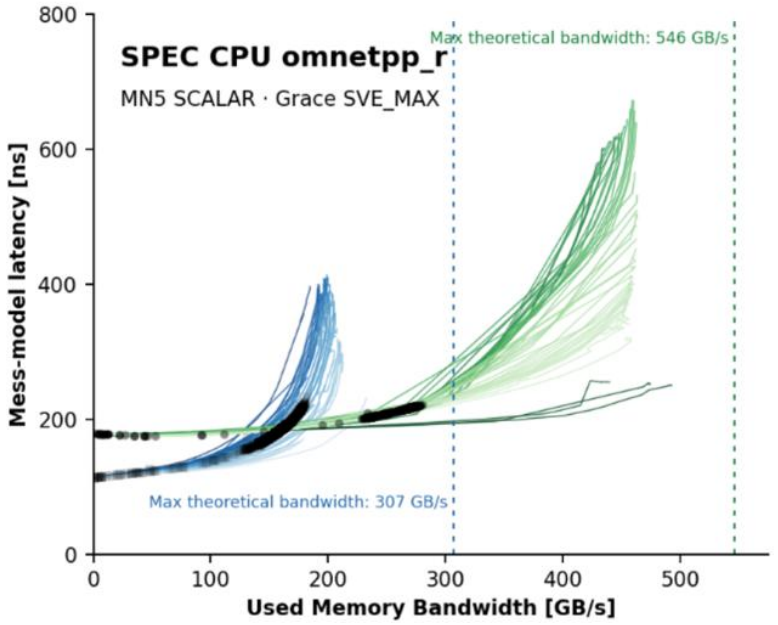
Memory stress @ ectrans-benchmark-sp.profet.prv

	[0.00..0.10)	[0.10..0.20)	[0.20..0.30)	[0.30..0.40)	[0.40..0.50)	[0.50..0.60)	[0.60..0.70)	[0.70..0.80)	[0.80..0.90)	[0.90..1.00]
THREAD 1.1.1	0.46 %	22.92 %	45.57 %	28.06 %	2.50 %	0.34 %	0.03 %	0.01 %	0.11 %	0.00 %
THREAD 1.2.1	0.38 %	20.48 %	55.90 %	22.57 %	0.53 %	0.03 %	0.00 %	0.02 %	0.09 %	0.00 %
Average	0.42 %	21.70 %	50.73 %	25.31 %	1.52 %	0.18 %	0.02 %	0.01 %	0.10 %	0.00 %
Avg/Max	0.92	0.95	0.91	0.90	0.61	0.54	0.51	0.77	0.94	0.60

Memory-system performance for HPC users



- Almost tangible memory-system performance





Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

(Quick) questions?



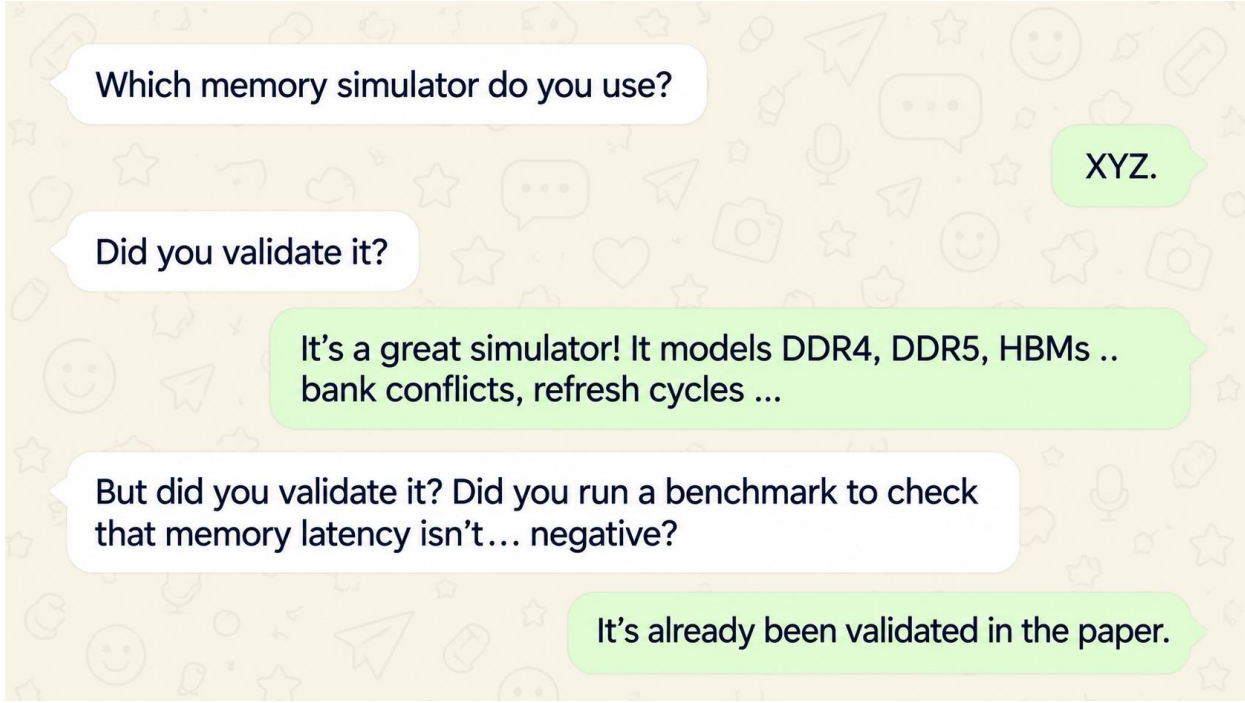
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Memory simulators!

This was interesting from the start …

- We were using ZSim+DRAMSim in various projects with industry
  - Mess-like validation
  - It took us some time to get memory simulation right



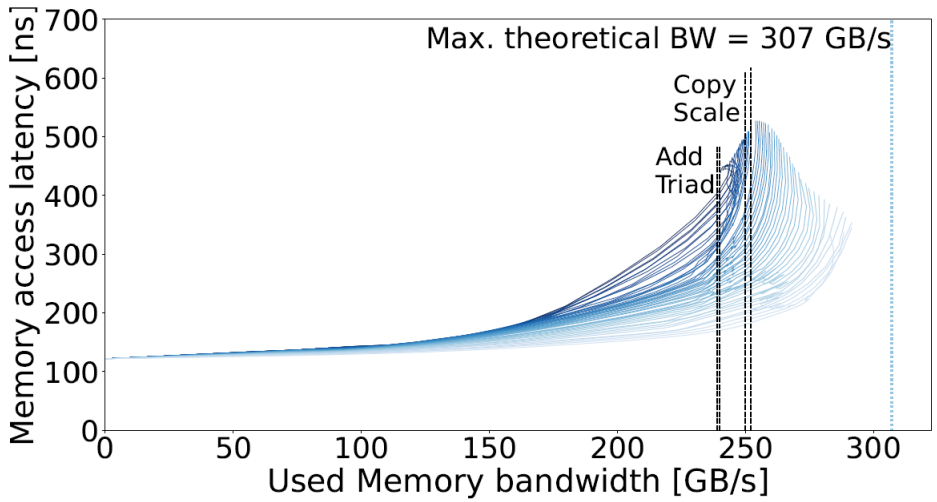
"It has already been validated…"

3.1 Validating the Correctness of Memory Simulator X

To make sure Simulator X memory controller and DRAM device model implementation is correct (i.e., the DRAM commands issued by the controller obey both the timing constraints and the state transition rules), we verify the DRAM command trace against Micron's DDR4 Verilog Model [24] using a similar methodology to prior works [2–4]. To do so, we implement a DRAM command trace recorder as a DRAM controller plugin that can store the issued DRAM commands with the addresses and time stamps using the DDR4 Verilog Model's format. We collect DRAM command traces from eight streaming-access and eight random-access synthetic memory traces and different intensities (i.e., the number of non-memory instructions between memory instructions). We feed the DRAM command trace to the Verilog Model, configured to use the same DRAM organization and timings as we use in Simulator X. We find no timing or state transition violations.

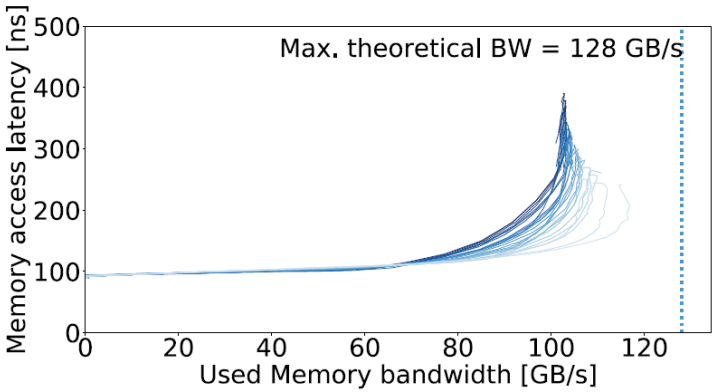
- **Implies nothing for** DDR5, LPx, GDDRx, HBMx, CXL memory expanders

- **Does not imply** ok performance

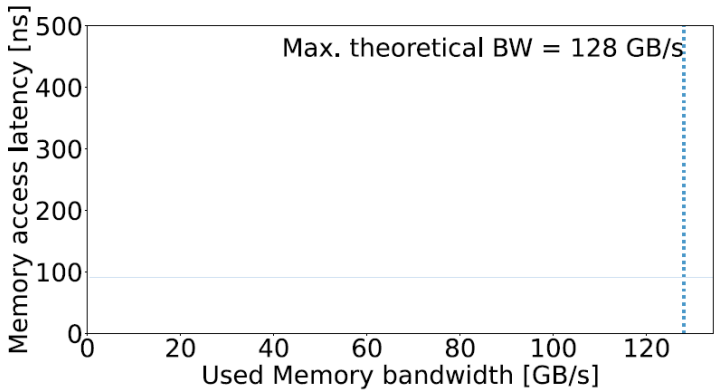


ZSim

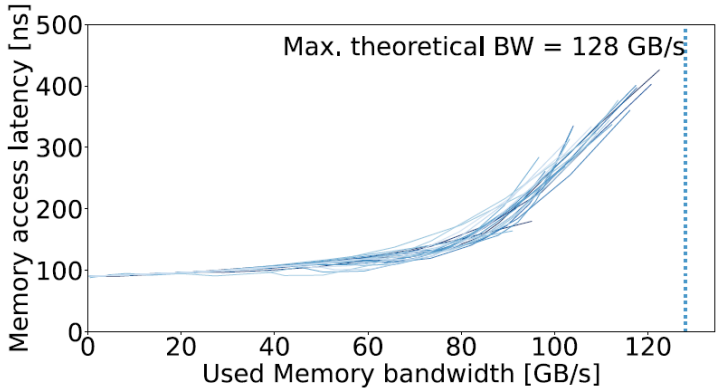




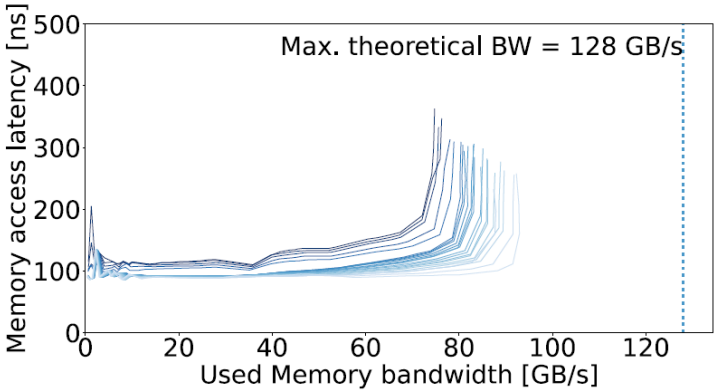
Intel Skylake with 6×DDR4-2666.



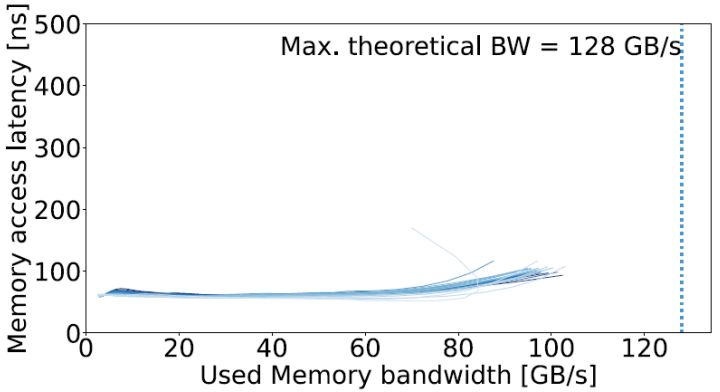
ZSim: Fixed-latency model



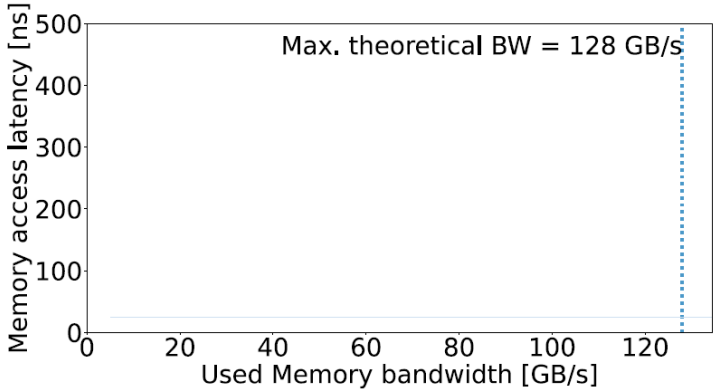
ZSim: M/D/1 Queue model



ZSim: Internal DDR model



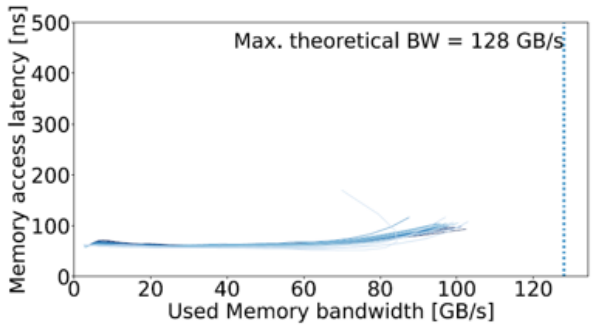
ZSim+DRAMsim3 simulator.



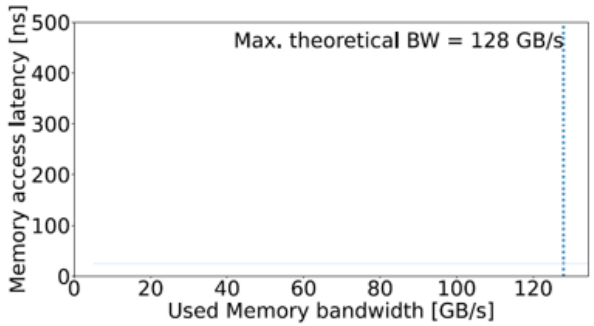
ZSim+Ramulator simulator.

The issue is the simulators' interface

- ZSim + External memory simulators

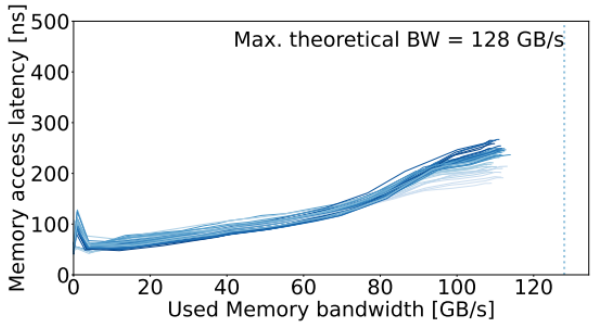


ZSim+DRAMsim3 simulator.

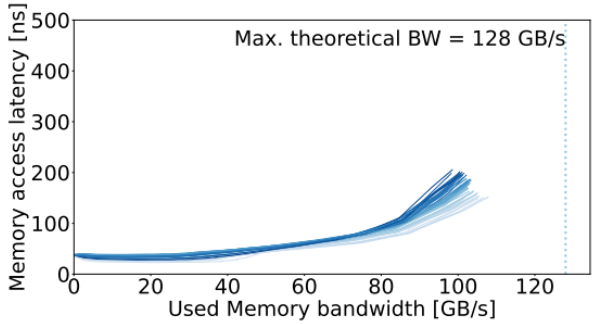


ZSim+Ramulator simulator.

- Trace-driven memory simulators



DRAMsim3: 6×DDR4-2666



Ramulator: 6×DDR4-2666

The fun started even before the paper was published
MEMSYS conference Oct 2024

- What matters the most is the **attitude**:
  - ***Award for the most controversial, truth-telling panel discussion.***
    Bruce Jacob (**DRAMsim**) and Matthias Jung (**DRAMsys**)







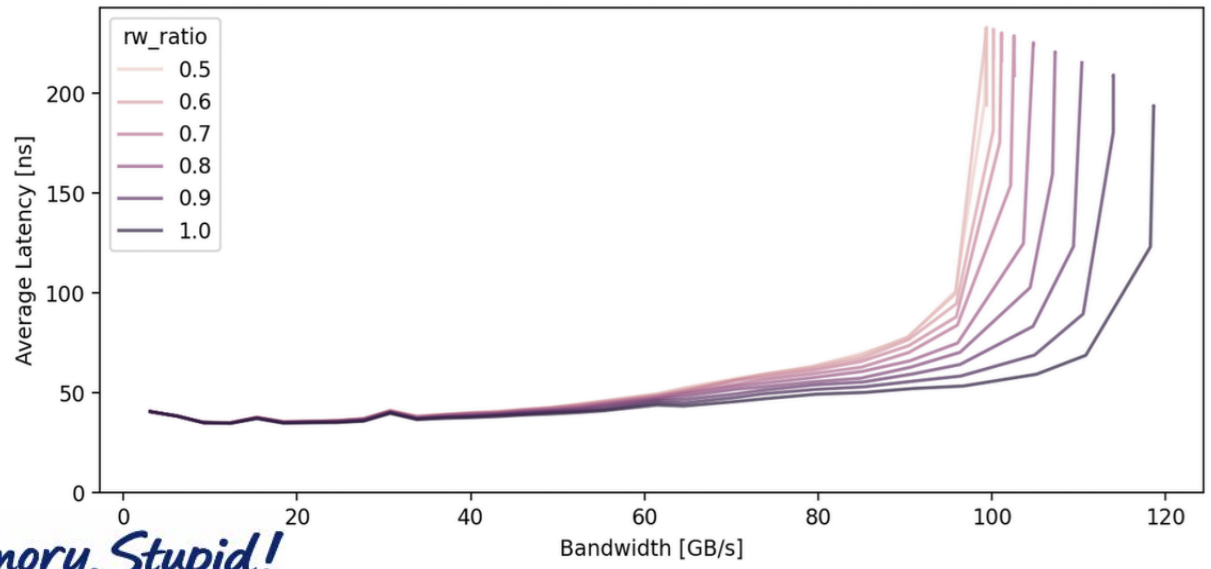
DRAMSys

DRAMSys 4.0

DRAMSys@iese.fraunhofer.de

- Based on the industry standard SystemC TLM-AT
- Widely used in industry ARTERIS IP Rambus
- Fast model, simulation only of important events (TLM)
- Fast and accurate design space exploration
- Open source:
  - DDR3, DDR4, LPDDR4, Wide I/O 1/2, GDDR5, GDDR5X, GDDR6, and HBM2
- Commercial licenses:
  - DDR5, LPDDR5, LPDDR6, HBM3/3e/4, GDDR7
  - Trace Analyzer tool
  - New standard models will be open-sourced when a level of maturity is reached

6xDDR4-2666 (MN4-like)
MemCtrl → Memory → MemCtrl



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

It's the Memory, Stupid!



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

A Mess of Memory System Benchmarking, Simulation and Application Profiling

4th of November 2024.



Why so large simulation errors?

- Honest answer: **We do not know**
- Currently exploring sources of error:
  - Memory simulators design
    - Row buffer statistics
  - Interfaces: CPU sim → Memory sim

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

micron

Large simulation errors: Simulator setup?

- We are (not me!) **skilled users** of the simulators under study
  - If you do not trust me, I have a back-up slide ;-)
- Work presented in the paper: **Typical use-case**
  - Find all info about the simulator installation and setup: papers, git, manuals, blogs, …
  - Unless we had major issues, we asked no support from simulator developers
- **We did not get the most** out of these simulators **yet!**

Large simulation errors: Simulator setup?

- **Ongoing work**
  - Contact simulator developers: DRAMSim3, DRAMSys, gem5 (main branch), Ramulator /2
  - Can we figure out together what are the performance issues?
    - Back-up slide: Preliminary nice story
- **My dream**
  - Be here next year (workshop?) to present the results and experiences

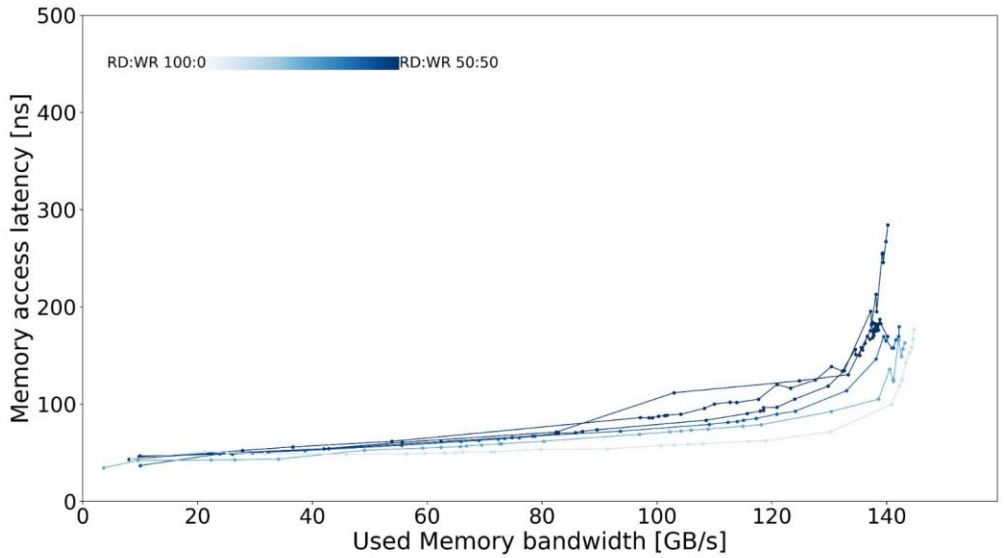
Everything will be okay in the end. If it's not okay, it's not the end.

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

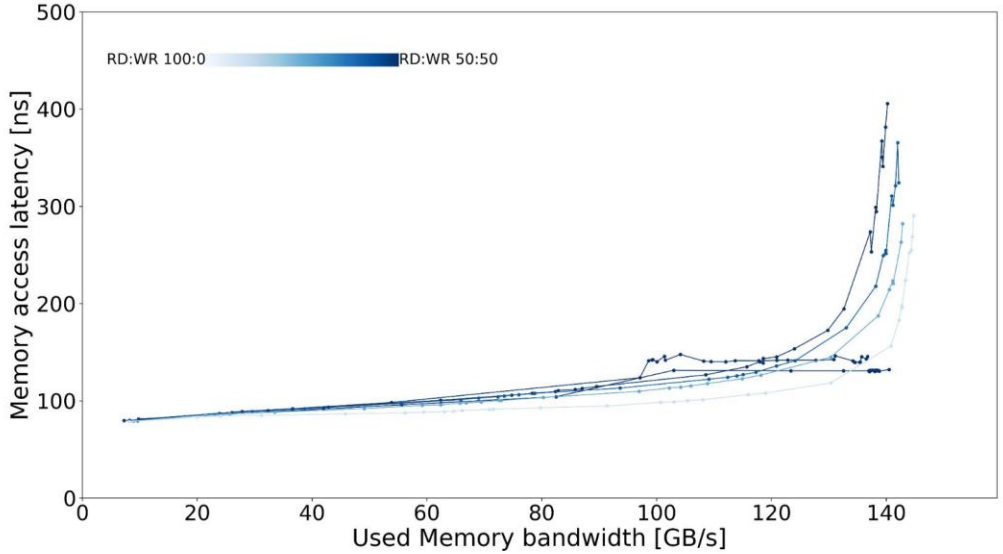
It's the Memory, Stupid!

gem5 + Mess: Work in progress

- BSC gem5 simulation infrastructure
- Internal gem5 model: DDR4-2400
- 8 channels → Max theoretical BW: 153.6 GB/s



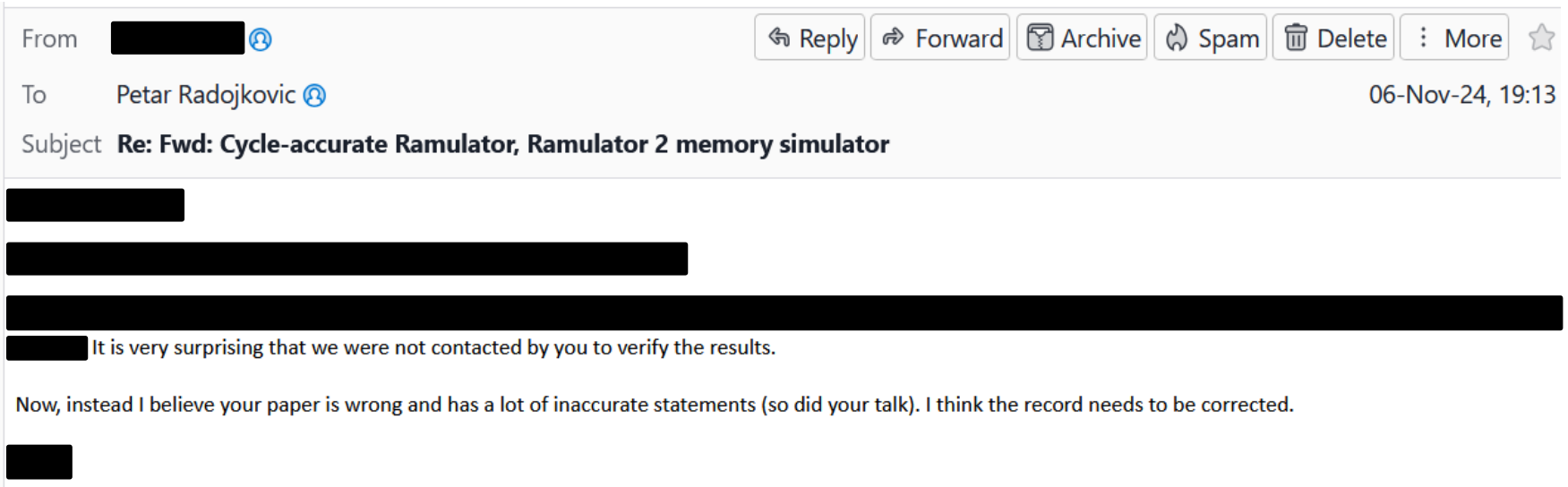
Internal memory counters measurements



Pointer-chase measurements

Ramulator

- Again: **Attitude is what matters**



- When one is faced with an uncomfortable statement, there are two choices:
  Examine the evidence, or attack the mirror.

These statements are limited to restoring a verifiable fact regarding the existence of this email discussion with the aim of preserving a respectful, rigorous, and community-useful environment for scientific discussion.

Out of respect for the confidentiality of communications, we will not publish the full emails or their attachments, recipient lists, email addresses, or screenshots of the thread that would allow the full content of the email exchanges between the two teams to be reconstructed. We have limited ourselves to minimal evidence consisting of communication dates, with the sender and recipient fields and the main messages exchanged redacted, while keeping the remaining information confidential.

In addition, we have also raised this matter through the internal channels made available by the organizations involved.

Months of unpleasant email exchange with the Ramulator crew

- Reported to BSC HR
- Followed the BSC HR recommendation to politely close the exchange
- Informed **Professor Onur Mutlu** about a potential violation of the **ETH Zurich Code of Conduct**

ETH zürich



Our principles:
At ETH Zurich, we treat each other professionally and with mutual respect. We all share responsibility for this: professors, students,
lecturers and researchers, as well as scientific, administrative and technical staff, and academic guests.

Open and fair communication, conflict resolution:
We communicate openly with each other, in a spirit of goodwill. Conflicts are resolved in a fair and objective manner to produce an effective solution. Our culture is built on trust and we welcome constructive criticism.
We support an open leadership culture based on dialogue.

Inappropriate behavior:
... the aim of discrediting ...
Examples include put-downs ...

The ETH code of conduct also describes how to deal with inappropriate behavior:

Dealing with inappropriate behaviour:
ETH Zurich encourages its staff, students and guests not to tolerate inappropriate behaviour and to take firm action against it. If those affected by, or witnessing, such behaviour are in a position to do so, they should make it absolutely clear to the person in question that they find their behaviour inappropriate and unacceptable.

2026 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)



Cleaning up the Mess: Re-Evaluating the Real-System Modeling Accuracy of Ramulator 2.0

F. Nisa Bostancı* Haocong Luo* Ataberk Olgun* Maria Makeenkova
Geraldo F. de Oliveira A. Giray Yağlıkçı Onur Mutlu
SAFARI Research Group
ETH Zürich

$31.00 ©2026 IEEE | DOI: 10.1109/ISPASS69572.2026.00029

A MICRO 2024 best paper runner-up publication (the Mess paper [1]) with all three artifact badges awarded (including "Reproducible") proposes a new benchmark to evaluate real and simulated memory system performance. The publication contends that Ramulator 2.0 [2] and DAMOV [3] (ZSim+Ramulator) (along with other existing memory system simulators) "poorly resemble the actual system performance" and asserts that their simulator is better.

1. Introduction

Cycle-level DRAM simulators enable research focused on improving the performance, efficiency, and robustness (including security, safety, reliability) of DRAM-based memory systems by providing accurate and flexible models for DRAM and memory controller operations. Ramulator 2.0 [2,6] (which builds on Ramulator [7, 8]), is a highly modular and extensible cycle-

The rest of it is history …

memory.bsc.es/ramulator-gate

Memory Systems for HPC and AI

Home About People Projects Publications Tools & Resources News

CONTENTS

00 What is this?
★ TopPicks
01 Memory bandwidth
a. Channel count
b. DDR5 DIMMs: Channel vs. sub-channel
c. Mess and Ramulator views
d. 64-bit DDR5 channels in Ramulator?
e. Bandwidth numbers, dressed in citations
02 Ramulator 2.0 and the magical y-axis
03 Zsim+Ramulator: Oops …
04 Reproducibility has a date. False statements do too.



Ramulator-Gate

When a benchmark exposes an uncomfortable truth, the community has two choices:
Examine the evidence, or attack the mirror.

This article has been accepted for publication in IEEE International Symposium on Workload Characterization (IISWC), 2026 (author version).

Integration, Enhancements and Evaluation of Memory Simulators

*Pouya Esmaili-Dokht[†§], *Arash Yadegari[§‡], Victor Xirau[§], Julian Pavon[§], Hamid Sarbazi-Azad[‡‖], Adrián Cristal[†§], Eduard Ayguadé[†§], Petar Radojković[§]
Universitat Politècnica de Catalunya[†], Barcelona Supercomputing Center[§], Sharif University of Technology[‡], IPM[‖]
{pouya.esmaili, arash.yadegari, victor.xirau, julian.pavon, adrian.cristal, eduard.ayguade, petar.radojkovic}@bsc.es
azad@sharif.edu

[cs.AR] 11 Oct 2026

***Abstract*—Currently, there is no common methodology or set of practices for evaluating memory-system performance in integrated simulation infrastructures. As a result, studies adopt different evaluation approaches, which can lead to conflicting conclusions even for the same simulation infrastructure.**

We propose a validation methodology that evaluates memory-system performance from three complementary views: the memory simulator, the CPU–memory interface and the simulated application. We use these views to detect, understand and correct memory-simulation inaccuracies arising from simulator integration and simplified modeling of microarchitectural features. The methodology reveals that the application's perception of memory performance can become effectively decoupled from the detailed

TABLE I
SIMULATOR USERS FREQUENTLY HAVE TO CHOOSE BETWEEN CYCLE-ACCURATE MEMORY MODELS AND FAST PARALLEL SYSTEM SIMULATORS.

Category	Model	gem5	Sniper	ZSim
Immediate-response	Fixed-latency	✓	✓	✓
	Queuing models (M/D/1)	✓	✓	✓
	Measurement-based (Mess)	✓	✓	✓
Cycle-accurate Delayed-response	DRAMsim	✓	✗	?
	Ramulator	✓	✗	?

What is there for you?

- **Critical thinking**
  - It is published, therefore it is correct!
  - MICRO-2024, ISPASS-2026, IISWC-2026
    - Not all of it can be correct!
- Find the intruder!

- **Free entertainment**



Ramulator-Gate Top Picks

The Mess paper took home a Best Paper Runner-Up. A few Ramulator-Gate statements deserve recognition too.

GOLD
Ramulator 2.0 and the magical y-axis

SILVER
Bandwidth numbers, dressed in citations

BRONZE
Zsim+Ramulator: Oops …
IISWC 2026 study ↗

Ramulator 2.0 and the magical Y-axis

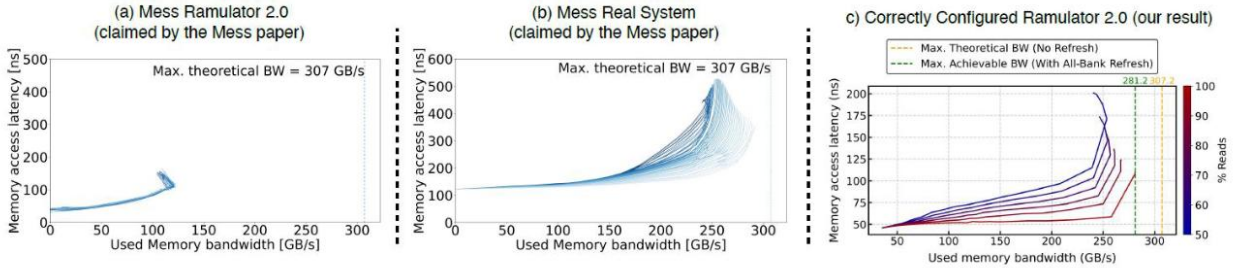
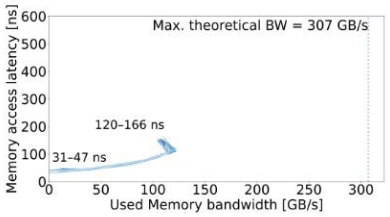


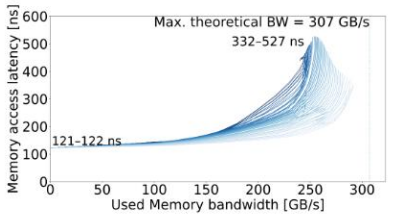
Fig. 1: Mess results for Ramulator 2.0 memory system simulation from Fig. 6 in [1] (a), Mess results for an ARM-based real system from Fig. 4 in [1] (b), our results for Ramulator 2.0 memory system simulation after correcting the errors we found in [1] (c), as explained in detail in Section 2 of this paper

> *"We conclude that Ramulator 2.0, when configured and used correctly, yields reasonable and realistic bandwidth–latency curves that resemble the ones obtained from real systems by [Mess paper] "*

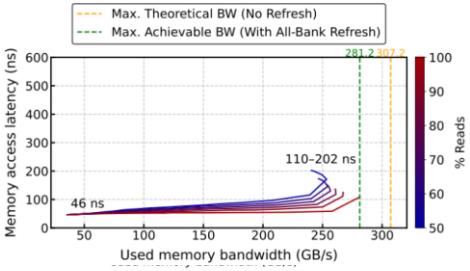
(a) Ramulator 2.0: 8xDDR5-4800 (Mess paper)



(b) Real system: Amazon Graviton 3, 8xDDR5-4800. (Mess paper)



(c) Ramulator 2.0: 16xDDR5-4800. (Ramulator-gate paper)



"The simulated memory latency is unrealistically low"

Aftermath



- ***Did we open something that should have remained closed?***

- I got disappointed with some people and some systems
  - Nothing I haven't heard before

- "It is a pity that we have to invest our time into this … "

We should not forget a lot of good stuff!

- Interest, recognition & adoption
  - Tens of presentations. Various awards.
- A lot of ongoing work with a lot of smart people: from MSc students to companies
- Career impact
- Great support from my superiors, colleagues and the community
- Team cohesion
- **Opportunity to fight for what you believe in!**



"It has already been validated…" – A big blind spot of the community

3.1 Validating the Correctness of Memory Simulator X

To make sure Simulator X memory controller and DRAM device model implementation is correct (i.e., the DRAM commands issued by the controller obey both the timing constraints and the state transition rules), we verify the DRAM command trace against Micron's DDR4 Verilog Model [24] using a similar methodology to prior works [2–4]. To do so, we implement a DRAM command trace recorder as a DRAM controller plugin that can store the issued DRAM commands with the addresses and time stamps using the DDR4 Verilog Model's format. We collect DRAM command traces from eight streaming-access and eight random-access synthetic memory traces and different intensities (i.e., the number of non-memory instructions between memory instructions). We feed the DRAM command trace to the Verilog Model, configured to use the same DRAM organization and timings as we use in Simulator X We find no timing or state transition violations.

- **Implies nothing for** DDR5, LPx, GDDRx, HBMx, CXL memory expanders

- **Does not imply** ok performance

Let´s not replace one blind spot with another

- Blind spot before the Mess paper
  - Memory simulators are validated*
  - Plug & Play everywhere: gem5, ZSim, etc.



- Potential new blind spot
  - Trace-driven / traffic-generator validation of a stand-alone memory simulator
  - Plug & Play everywhere: gem5, ZSim, etc.

"Here you have your bandwidth-latency curves. Now leave me alone!"



- Validation methodology matters, and we do not have it!
  - Could be problematic

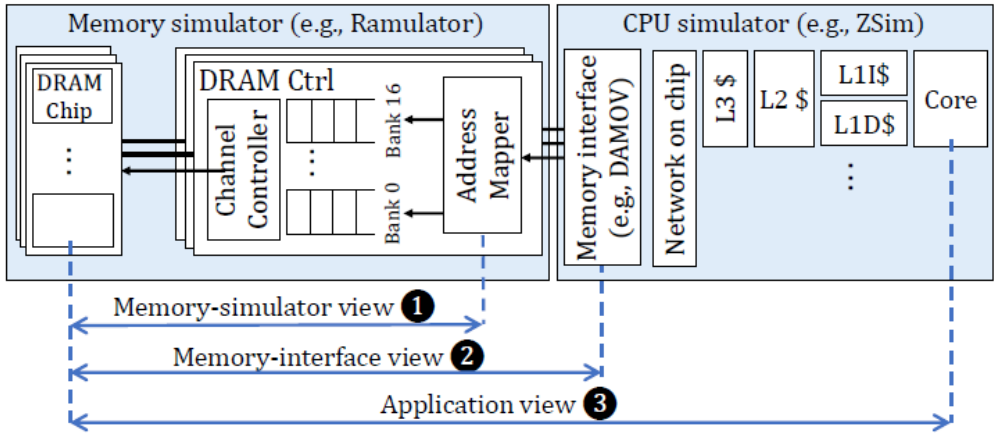
The most controversial: ZSim + Ramulator

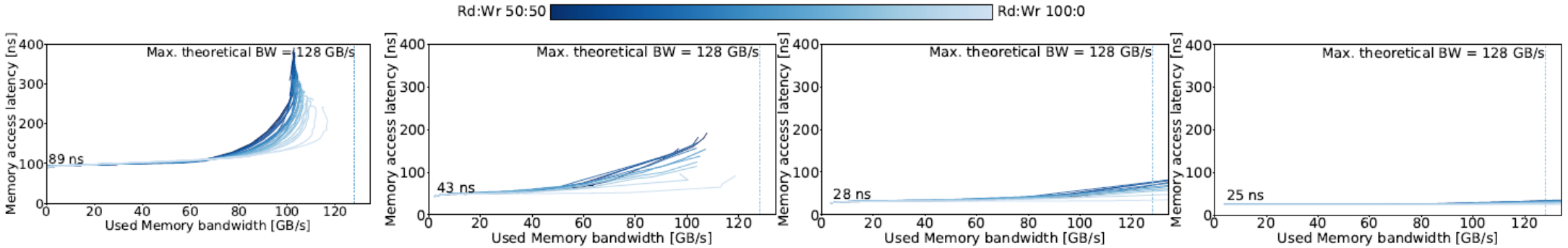
- Two recent studies
  - **The same simulation infrastructure**: ZSim + DAMOV interface + Ramulator
  - **Practically the same measurements**: Memory bandwidth—latency curves [Mess benchmark]
  - **Contradicting conclusions**

- **How is this possible?**

More info:
Esmaili et al., **Integration, Enhancements and Evaluation of Memory Simulators.** IISWC, 2026.

We measured the same thing, but in different places!

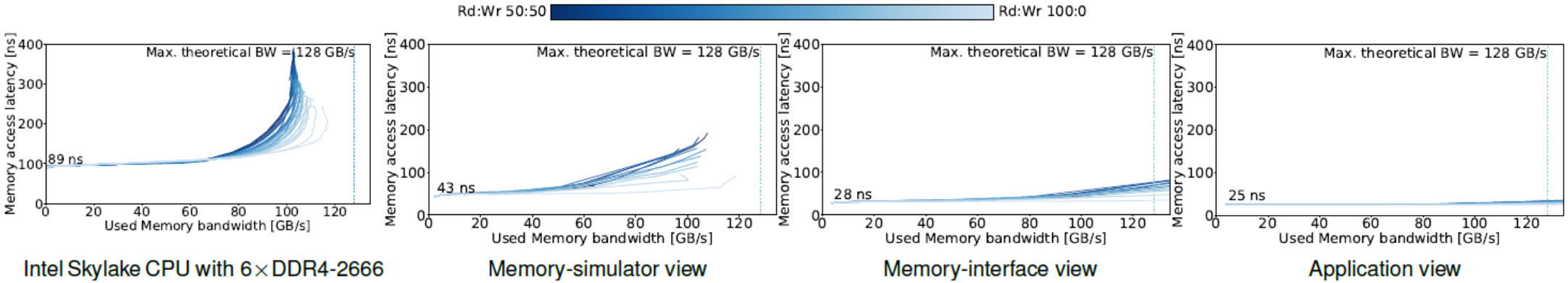




MICRO-2024 study: *

ISPASS-2026 study:

Which approach is the right one?



MICRO-2024 study: *

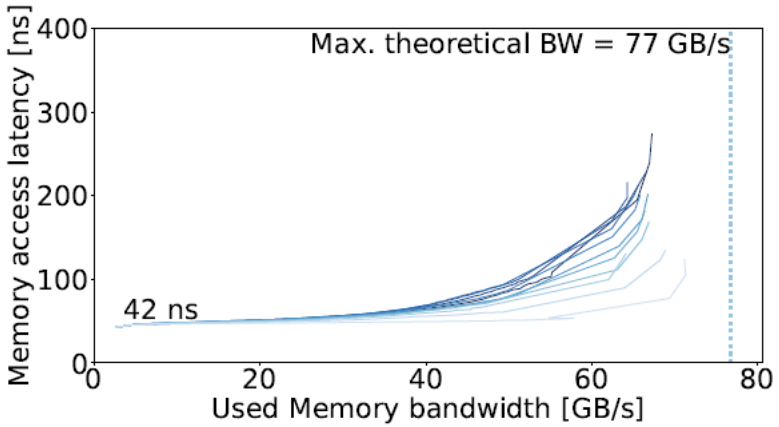
ISPASS-2026 study:

- The methodology and common practices are not defined → **There is no right and wrong!**

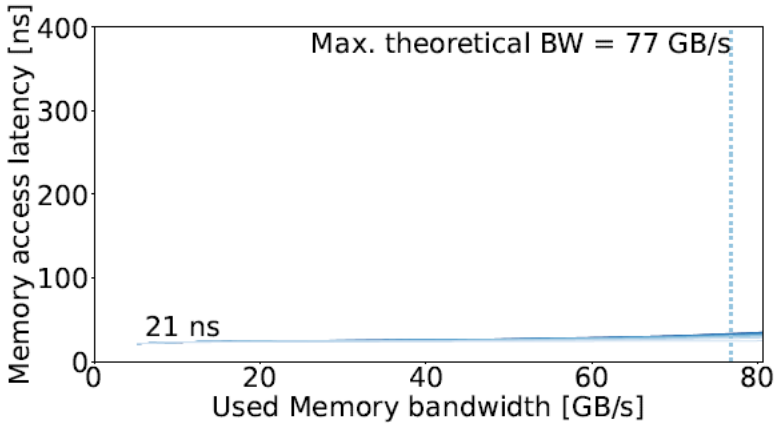
Bullet-proof evidence of the presented ZSim-DAMOV-Ramulator issue

- Let's test the Patient Zero: **Original DAMOV framework**
  - Different CPU model
  - Different memory configuration

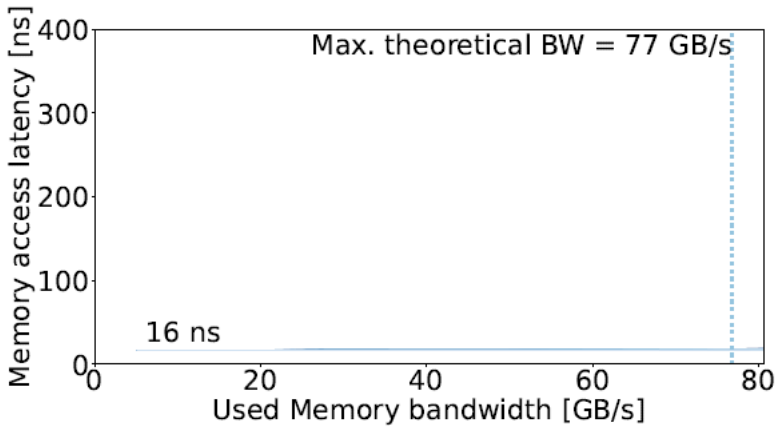
  → The main question: **Does the overall behavior follow our findings?**



Memory-simulator view



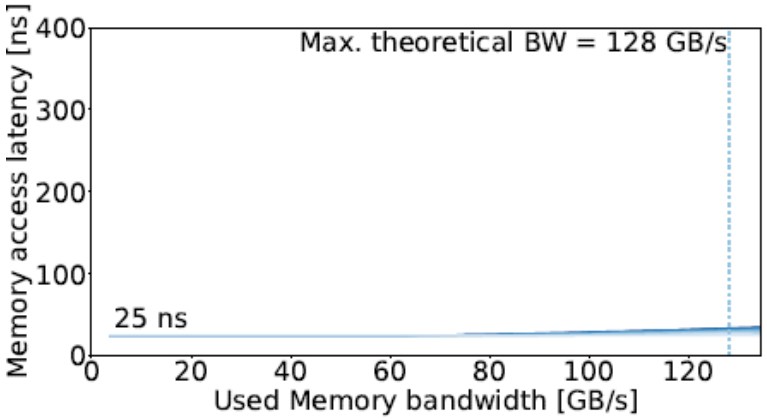
Memory-interface view



Application view

Eight months of engineering later …

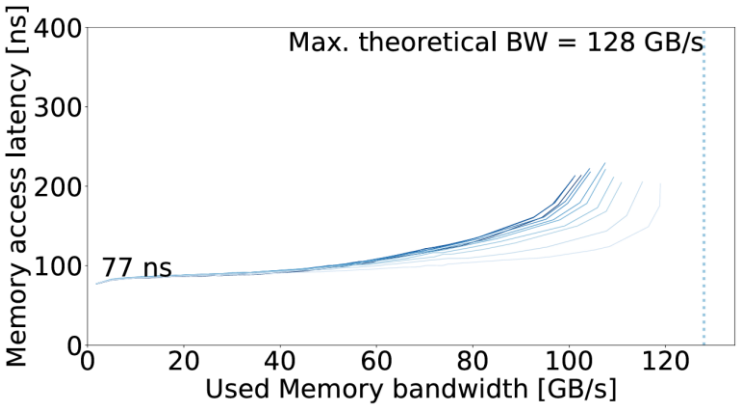
From:



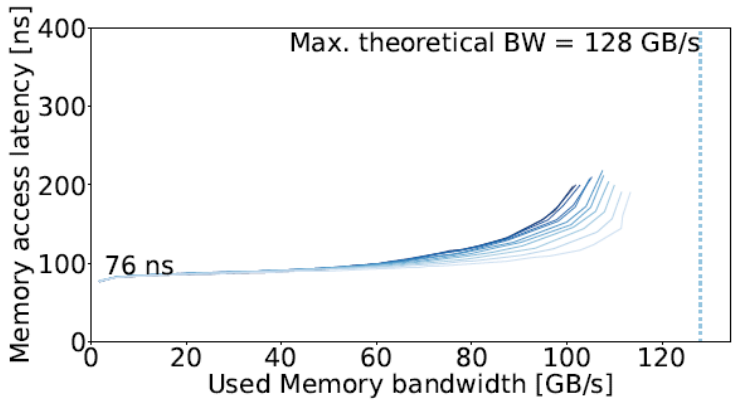
Application view

ZSim+Ramulator

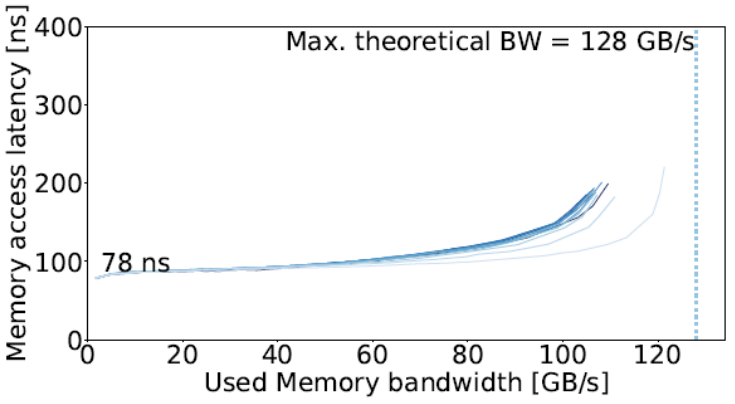
To:



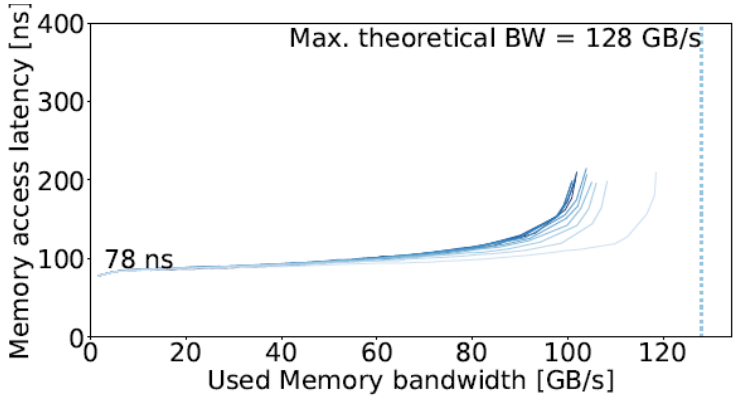
ZSim+Ramulator



ZSim+Ramulator 2



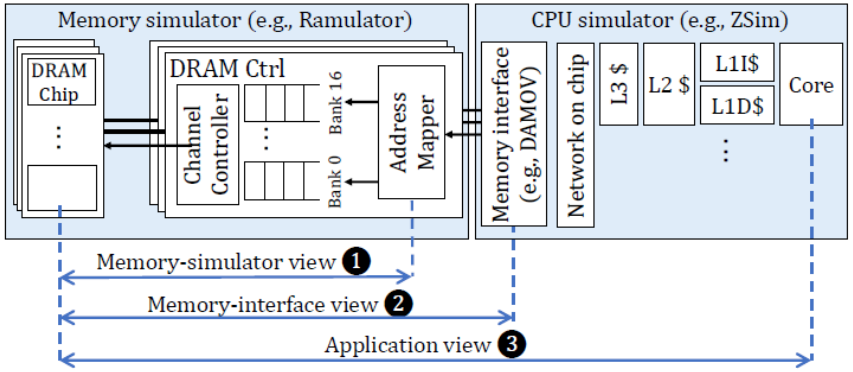
ZSim+DRAMsim3

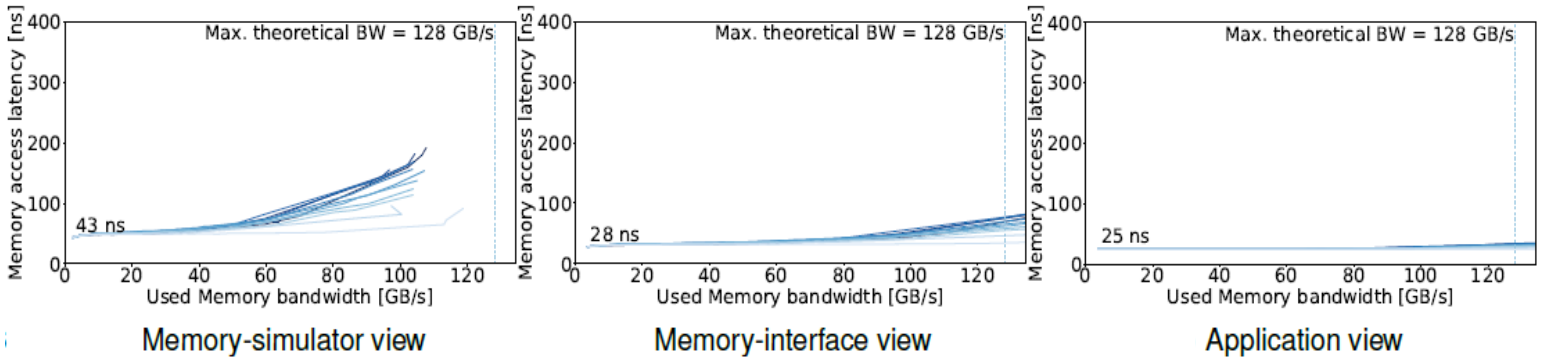


ZSim+DRAMSys 4

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

My desired takeaway from this tutorial





- Can **we agree** on some common practices for simulator validation?
- Our position:
  "The **application view** directly **determines simulated application performance** and is therefore **the relevant measure of memory-simulation accuracy**."
- **What is your position?**

My desired takeaway from this tutorial:
Plug → *[Validate & Correct]* → Play as much as you want

- **Congrats to people who are already doing it!**

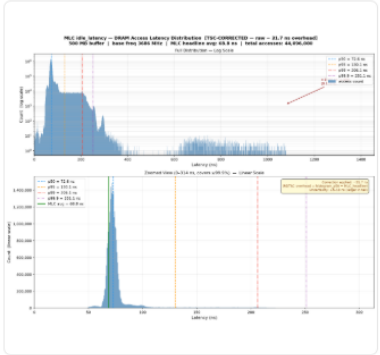
- My attitude changed:
  No hard feelings for the people who don't!
  - We pointed to a potential problem,
    up you to decide what to do about it

- Our "core business" ≠ Your "core business"

- Doing a PhD is complex
  - Effort to validate simulators?
  - Effort to correct all performance issues?
  - Impact on your results and conclusions?
  - Increasing the probability for the paper to be accepted?

[redacted]

this is for mlc --idle_latency (just pointer chase), where intel MLC provides 68-70ns on single socket workstation. Then I plot the histogram and correct the computed P50 - MLC_given value, corrected by 30ns.



[redacted]

when do --idle-latency, at steady state, the pure pointer chase, the understanding is that every access with miss LLC isnt it.

[redacted]

you can plot mlc --idle_latency -f1 (to plot the histogram)

[redacted]

and for ramulator2 calculation, i captured the trace with PIN, removed the initialization, and got the pure pointer chase trace (where a PC will repeat) , where every access is a DRAM access.



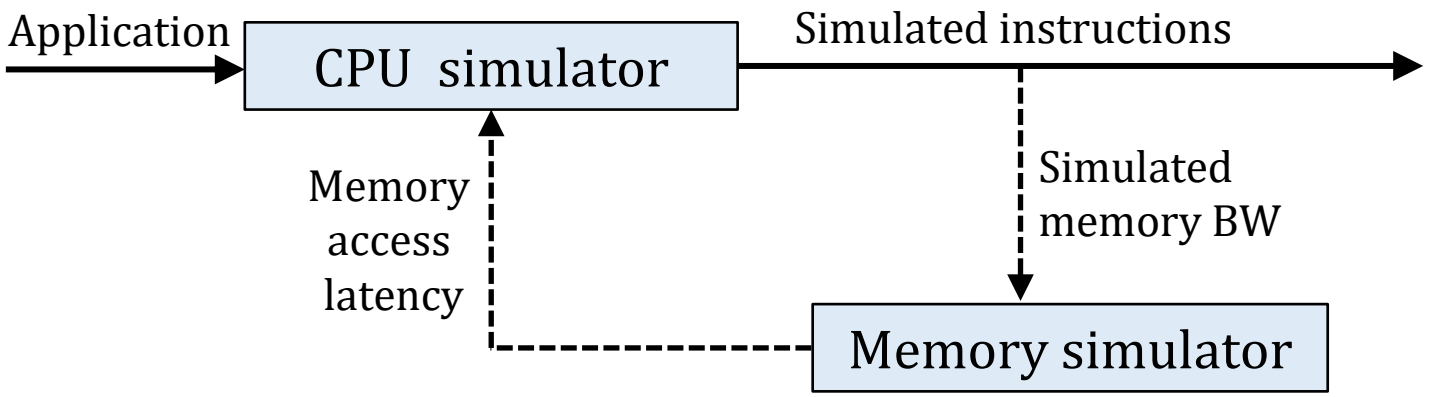
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Mess simulator

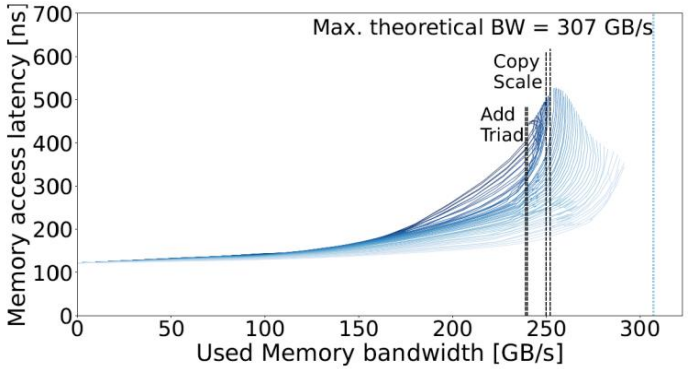
Should we rethink memory simulation?

- Step back: The main purpose of the memory simulator

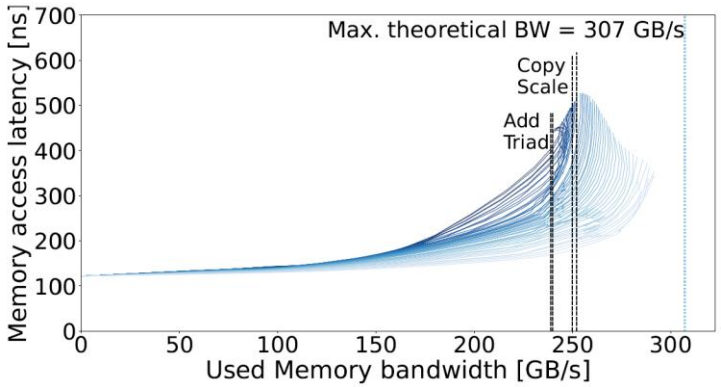


Should we rethink memory simulation?

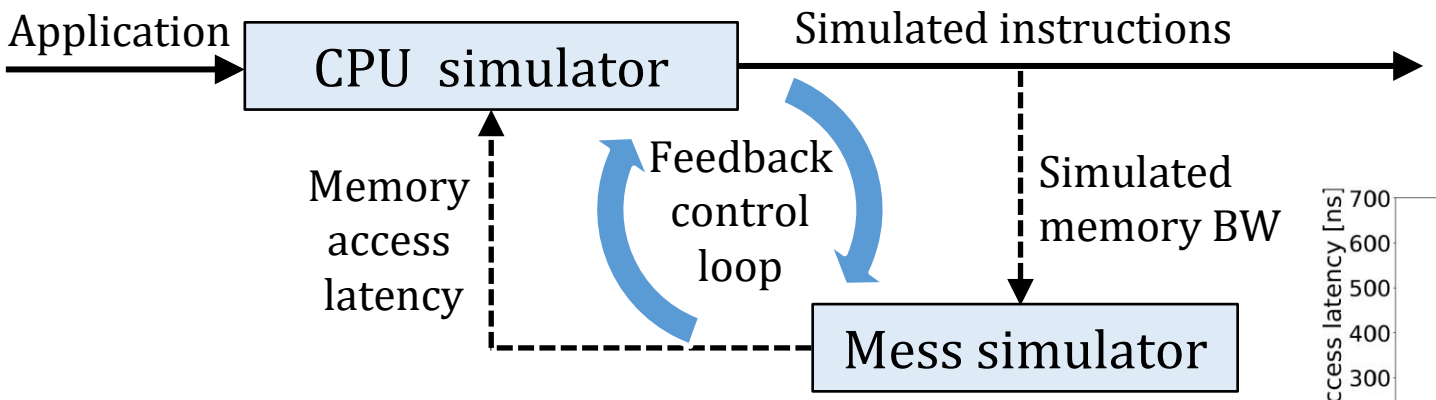
If our objective is to simulate

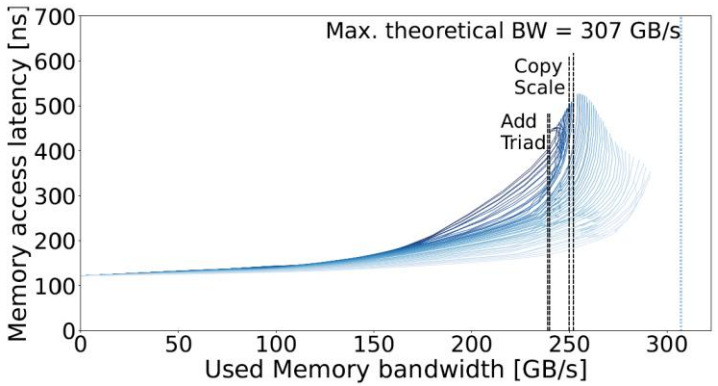


why don’t we use

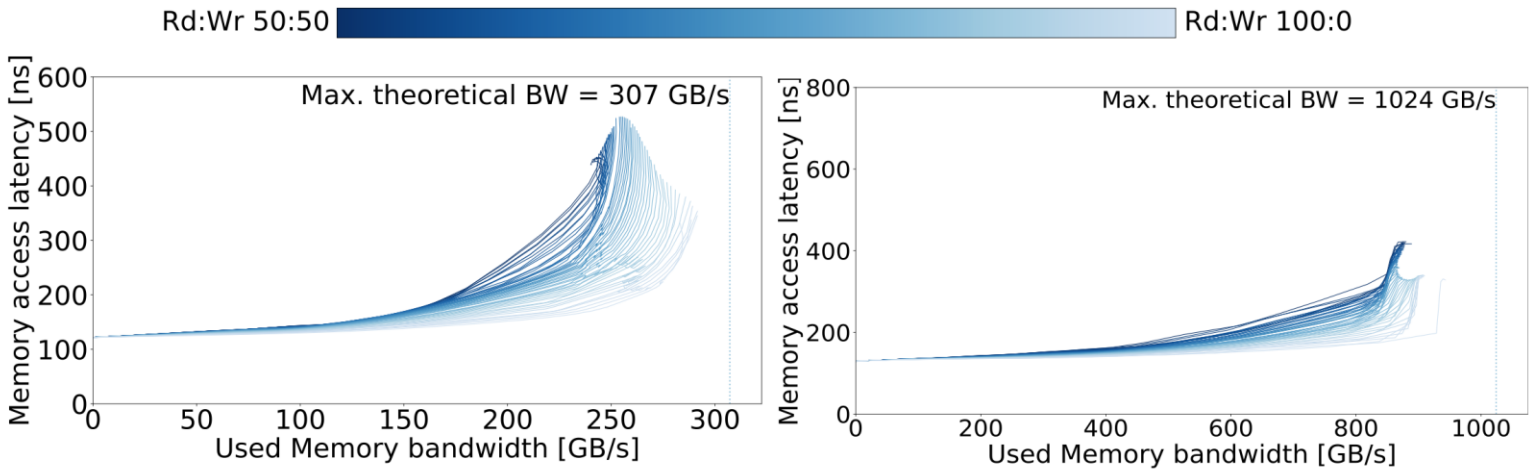


Mess simulator



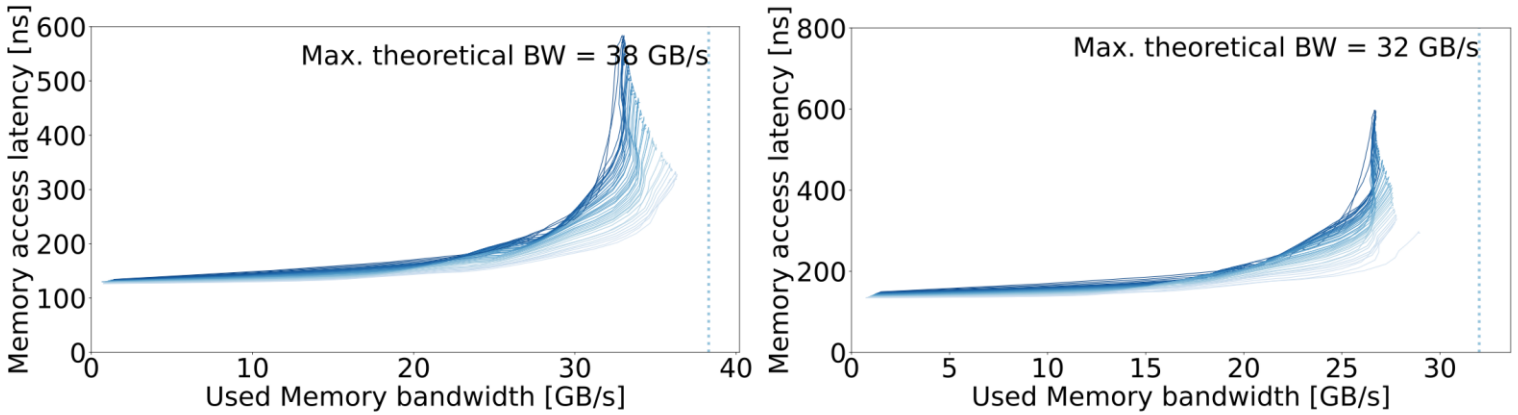


Mess simulator evaluation: gem5



Amazon Graviton3: 8×DDR5-4800

Fujitsu A64FX with 4× HBM2.



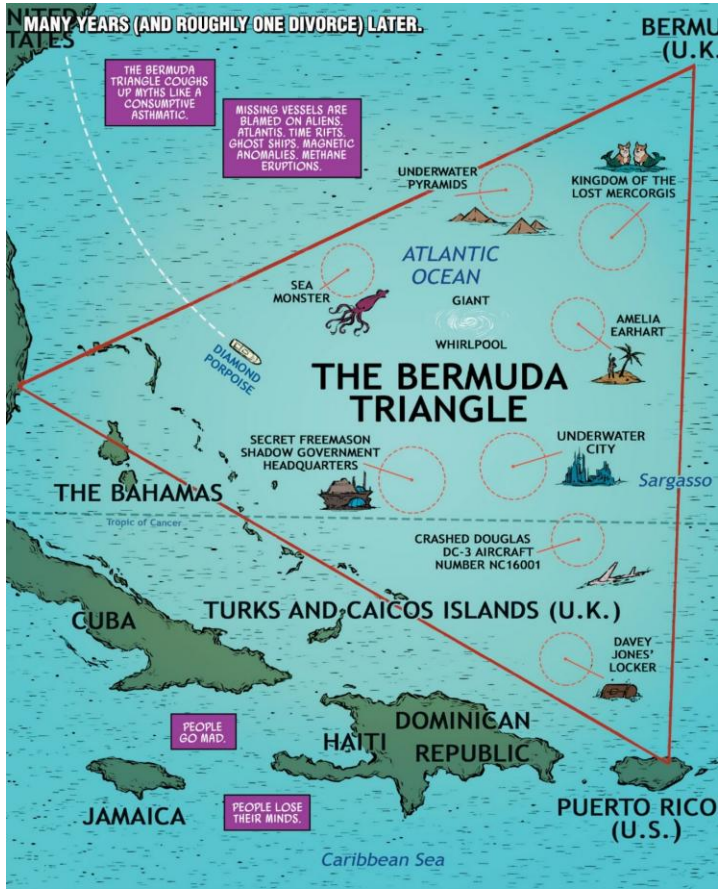
gem5: 16 cores & 1× DDR5-4800

gem5: 16 cores & 1× HBM2 channel

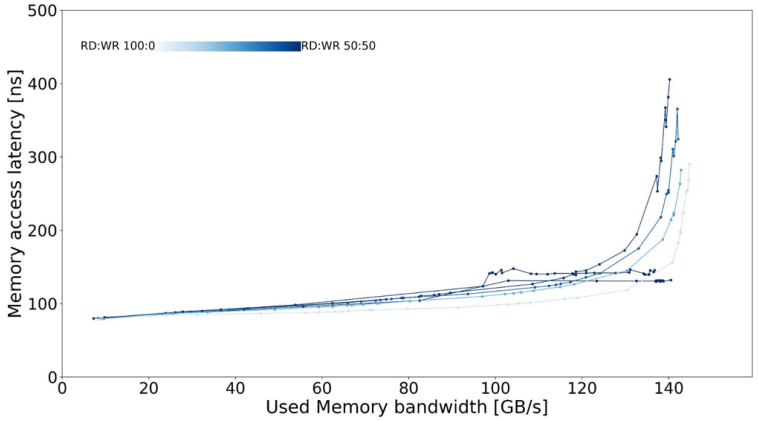
Mess simulator: Low priority

- We should know really well the CPU simulators
  - ZSim warning

- We discovered very good (relatively simple) memory models
  - Queuing theory
  - Regression models
  - Internal models
  - …
  - → **Do we need one more?**



Internal gem5 model: DDR4-2400





Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Happy to see follow-up work in this direction!

PANEM: A Mess-Adjacent Memory Simulator

As seen at IISWC 2026

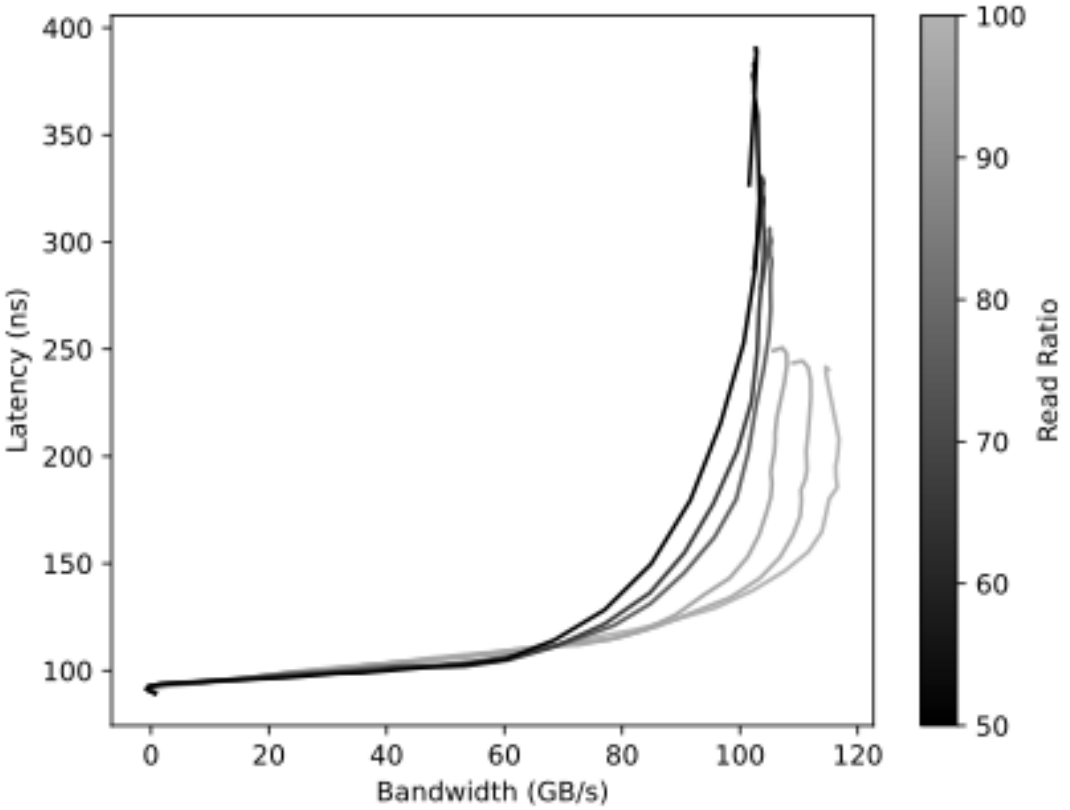
Typical Bandwidth-Latency Based Model:

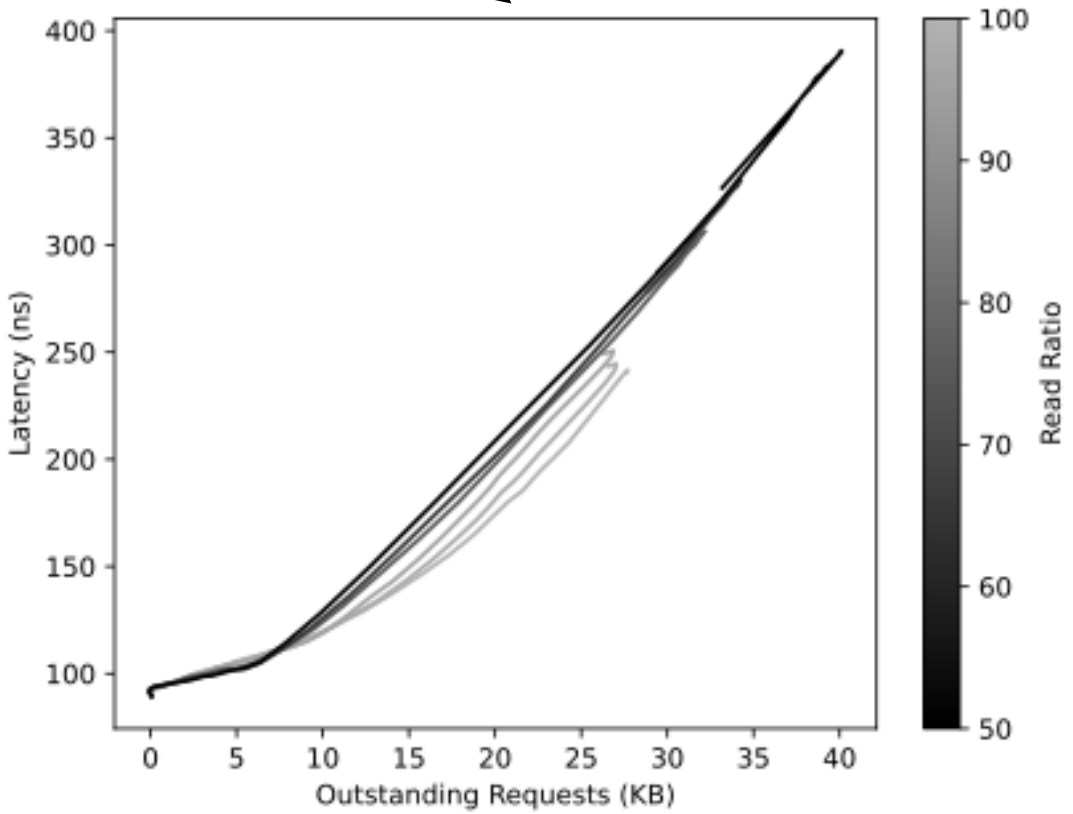
1. Observe Average Bandwidth
2. Select Latency
3. System State (Concurrency) Arises from 1 and 2

PANEM:

1. Observe System State (Concurrency)
2. Select Latency
3. Average Bandwidth Arises from 1 and 2

Little's Law





PANEM: A Mess-Adjacent Memory Simulator

- PANEM is Ampere's most used memory simulator for performance studies and is used in single-core and full mesh models to quickly simulate the impact of a wide variety of memory configurations on cloud workloads. It has impacted feature selection for real-world shipping products you can buy today.
- Demonstration utility, data, and figures available on Zenodo: https://zenodo.org/records/22817989

- Ampere has open sourced its simulation framework and performance models: https://github.com/AmpereComputing/panthera

For more information see:
L. Crowthers and M. Mahesh, "PANEM: A Heuristic Latency Model," Aug. 28, 2026. Accessed: Oct. 20, 2026. [Online]. Available: https://arxiv.org/abs/2609.29629v1

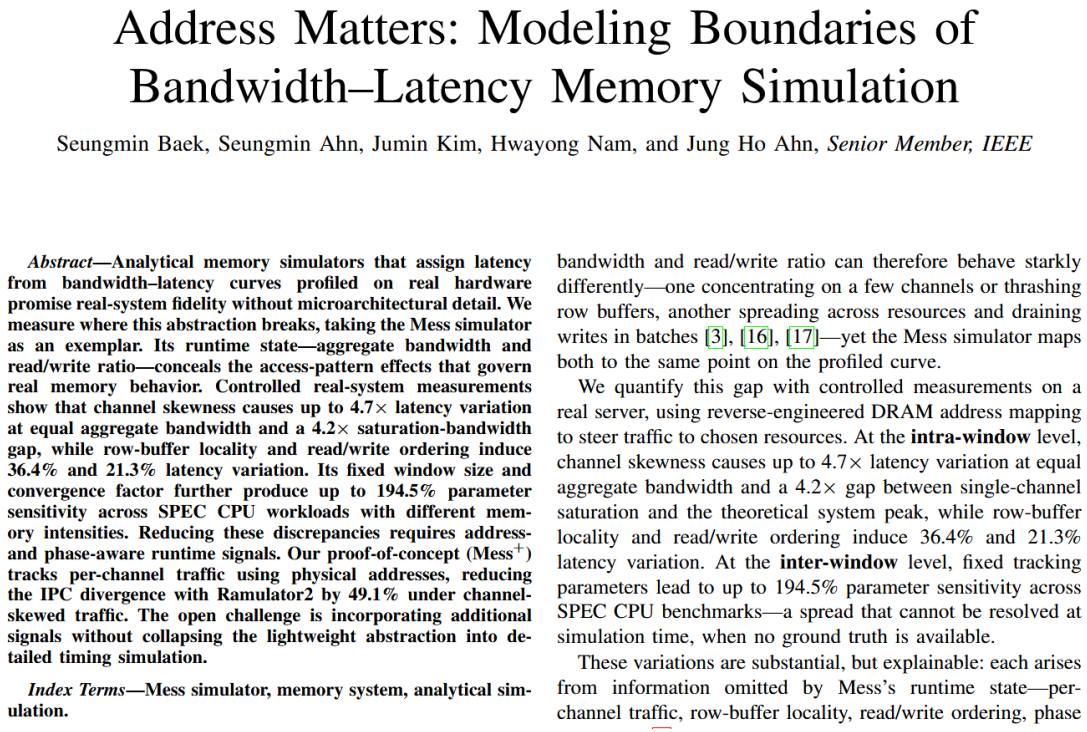
Or contact:
Lucas Crowthers
lucas@amperecomputing.com

Happy to consider any fair criticism

- Are we missing anything important?
  - Model
  - Model description
- How big is the problem in production?
  - Gold standard: Actual systems
- Should we enhance our models?

 # More important:

- **Should we think about an overall methodology for memory-simulator validation?**



Address Matters: Modeling Boundaries of Bandwidth–Latency Memory Simulation

Seungmin Baek, Seungmin Ahn, Jumin Kim, Hwayong Nam, and Jung Ho Ahn, *Senior Member, IEEE*

***Abstract*—Analytical memory simulators that assign latency from bandwidth–latency curves profiled on real hardware promise real-system fidelity without microarchitectural detail. We measure where this abstraction breaks, taking the Mess simulator as an exemplar. Its runtime state—aggregate bandwidth and read/write ratio—conceals the access-pattern effects that govern real memory behavior. Controlled real-system measurements show that channel skewness causes up to 4.7× latency variation at equal aggregate bandwidth and a 4.2× saturation-bandwidth gap, while row-buffer locality and read/write ordering induce 36.4% and 21.3% latency variation. Its fixed window size and convergence factor further produce up to 194.5% parameter sensitivity across SPEC CPU workloads with different memory intensities. Reducing these discrepancies requires address- and phase-aware runtime signals. Our proof-of-concept (Mess^{+}) tracks per-channel traffic using physical addresses, reducing the IPC divergence with Ramulator2 by 49.1% under channel-skewed traffic. The open challenge is incorporating additional signals without collapsing the lightweight abstraction into detailed timing simulation.**

***Index Terms*—Mess simulator, memory system, analytical simulation.**

bandwidth and read/write ratio can therefore behave starkly differently—one concentrating on a few channels or thrashing row buffers, another spreading across resources and draining writes in batches [3], [16], [17]—yet the Mess simulator maps both to the same point on the profiled curve.

We quantify this gap with controlled measurements on a real server, using reverse-engineered DRAM address mapping to steer traffic to chosen resources. At the **intra-window** level, channel skewness causes up to 4.7× latency variation at equal aggregate bandwidth and a 4.2× gap between single-channel saturation and the theoretical system peak, while row-buffer locality and read/write ordering induce 36.4% and 21.3% latency variation. At the **inter-window** level, fixed tracking parameters lead to up to 194.5% parameter sensitivity across SPEC CPU benchmarks—a spread that cannot be resolved at simulation time, when no ground truth is available.

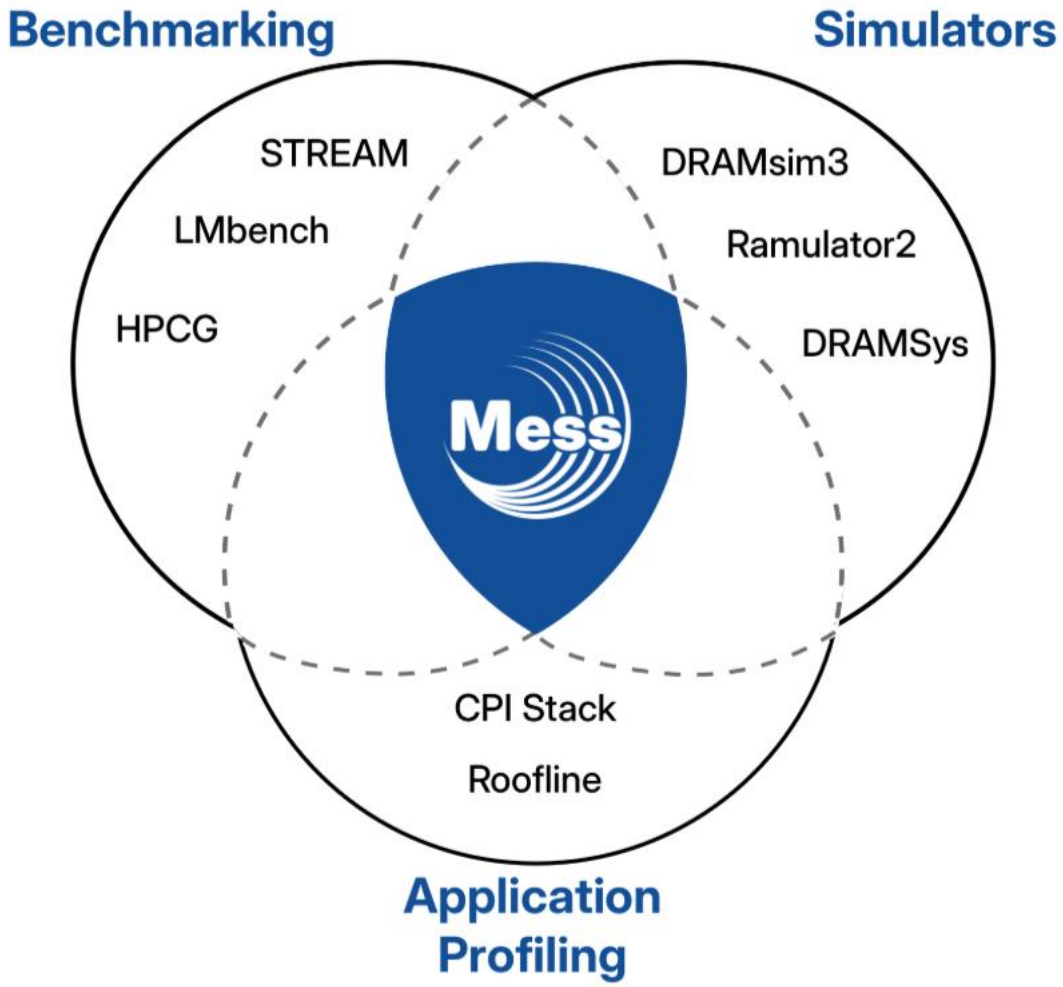
These variations are substantial, but explainable: each arises from information omitted by Mess's runtime state—per-channel traffic, row-buffer locality, read/write ordering, phase



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

My drive



"Have we found some common ground?"

Working with young people is

fun!

-------- Forwarded Message --------

Subject: Getting read for rebuttal...
Date: Wed, 19 Jun 2024 06:28:36 +0200
From: Pouya Esmaili dokht <pouya.esmaili@bsc.es>
To: petar Radojkovic <petar.radojkovic@bsc.es>

5:40 and waiting for the coffee to kick in...

Best Regards,
Pouya



You only live once, but if you do it right, once is enough


Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación
EXCELENCIA
SEVERO
OCHOA
Thank you!
mess@bsc.es