SNIA DEVELOPER CONFERENCE

SDC 24

BY Developers FOR Developers

September 16-18, 2024
Santa Clara, CA

Optimized Resource Allocation for CXL Tiered-Memory Systems

Heiner Litz & Andrew Quinn
UC Santa Cruz

Center for Research in Systems and Storage (CRSS)

The Center:

- 5 Faculty
- 15 Ph.D. & MS
- 6 Sponsors

Research Topics:

- CXL
- AI Systems
- Sustainability
- Data centers
- Storage Devices
- Operating Systems
- Networking

Output:

- High-impact research

ceph

- Publications

(ASPLOS, MICRO, SoCC, OSDI, ISCA, PLDI, HotOS, FAST)

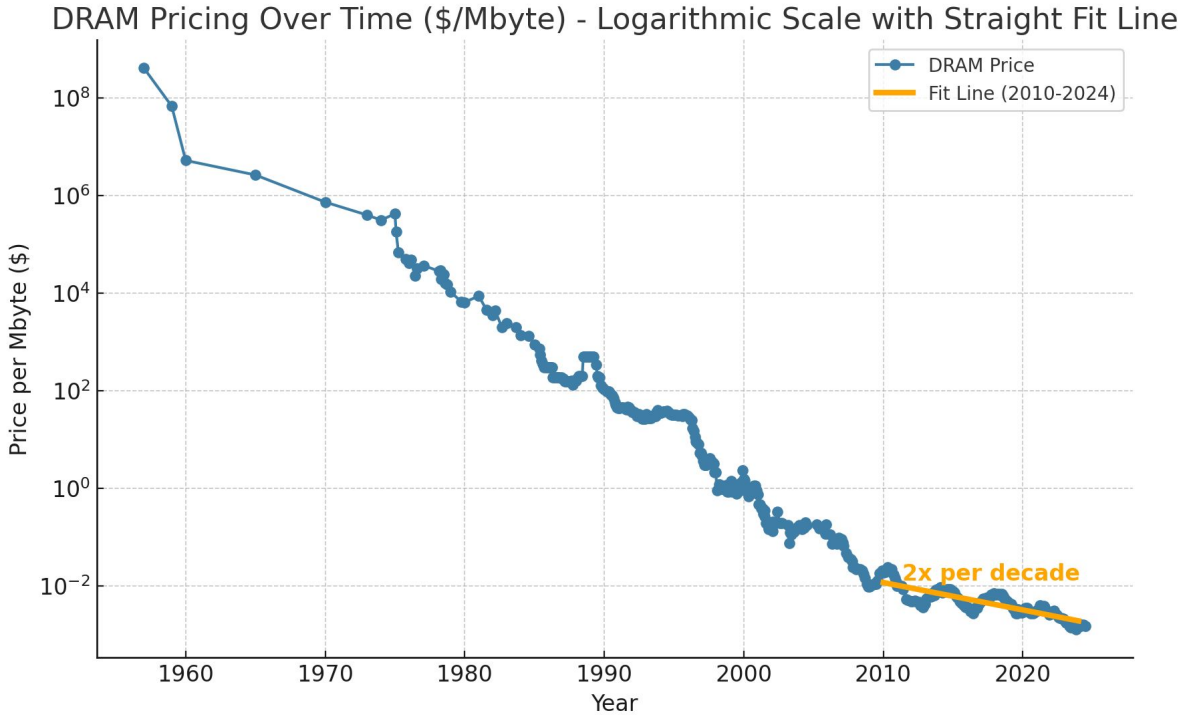
- Excellently-trained graduates

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Today's Problem



Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

CXL: Opportunities & Challenges

- Opportunities
  - Addresses scaling issue by reducing memory cost
  - Open standard enables “small players” to innovate
  - Computational memory enables TCO & perf improvements
- Challenges
  - Performance overhead
  - Heterogeneity increases complexity
  - Requires cross-layer (SW/HW) optimizations

Today's Talk

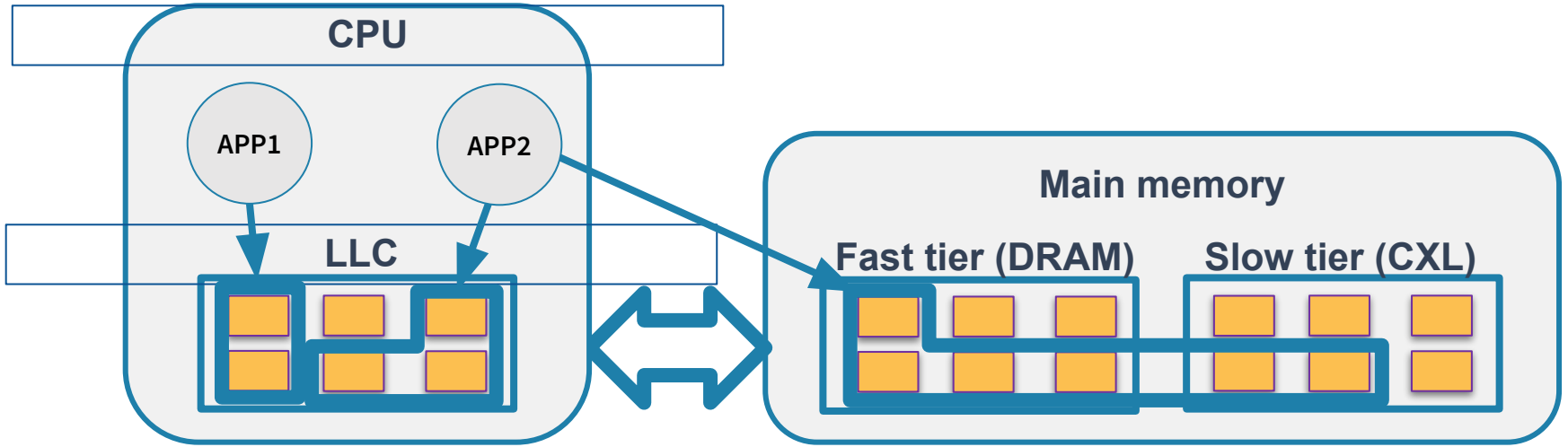
- Part 1: How can we exploit CXL memory to improve TCO (TMC)?
  - Delivered by Dr. Heiner Litz
- Part 2: How can we exploit CXL memory to improve cluster job performance (Bede)?
  - Delivered by Dr. Andrew Quinn

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Memory Tiering



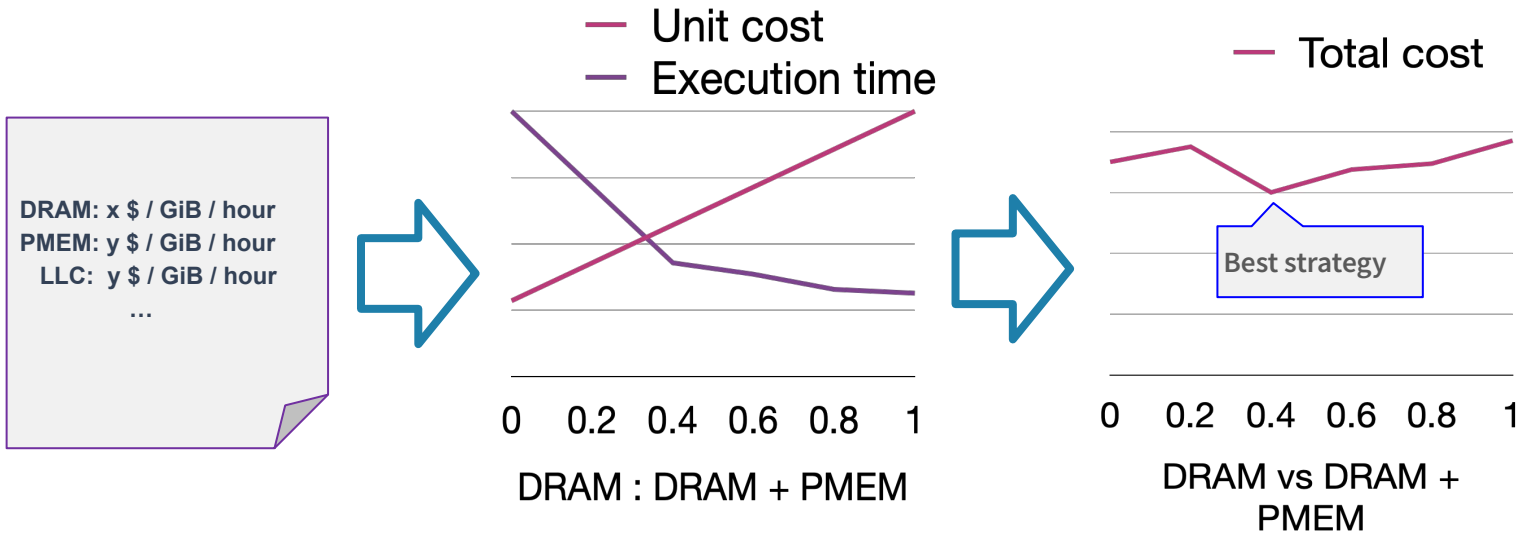
Tiered memory seeks to maintain similar performance at a lower cost

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Determining Optimal Memory Ratio is Hard

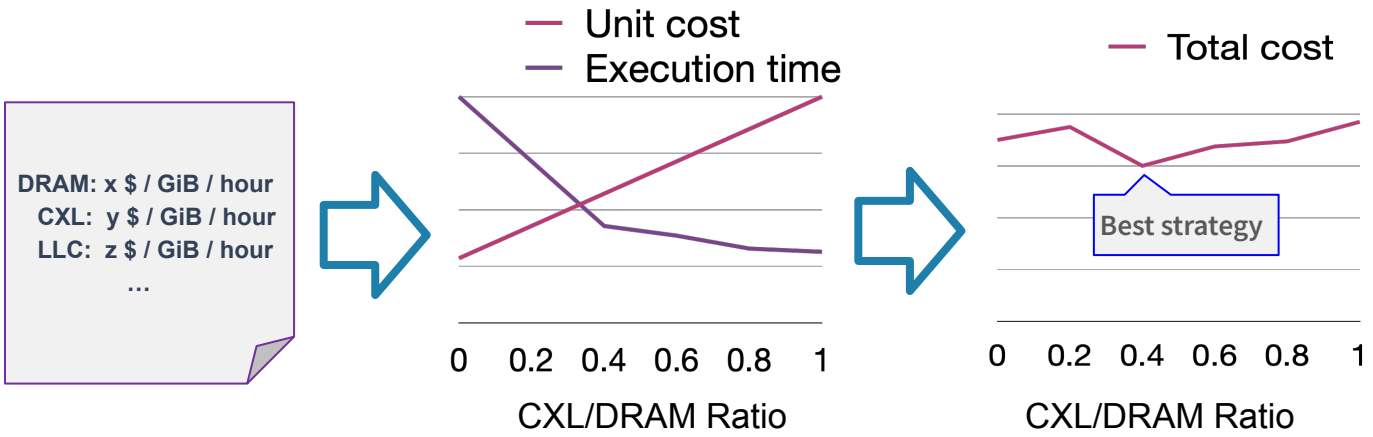


Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Determining Optimal Memory Ratio is Hard



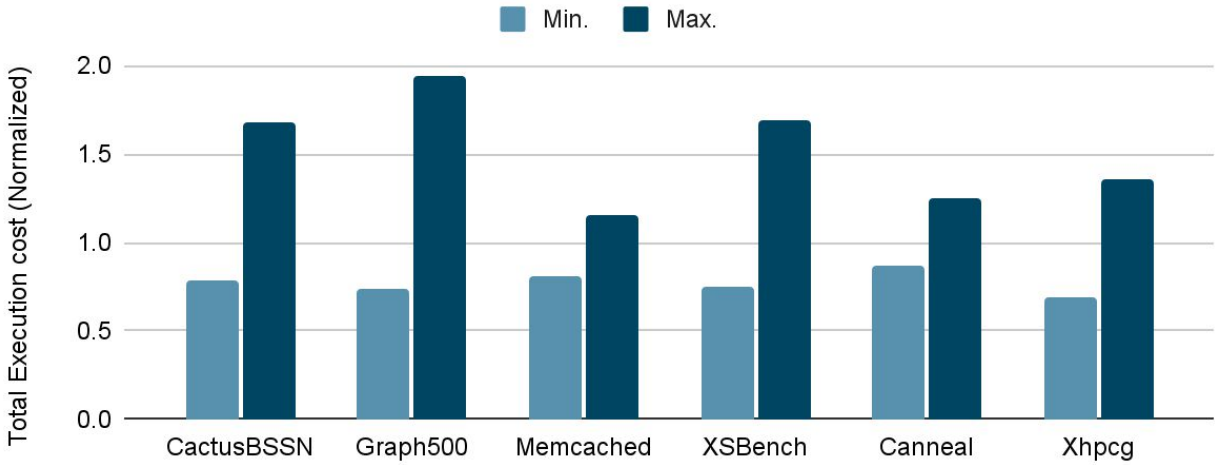
Large search space, scales with memory tiers

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

User: Lowest $ at certain performance level



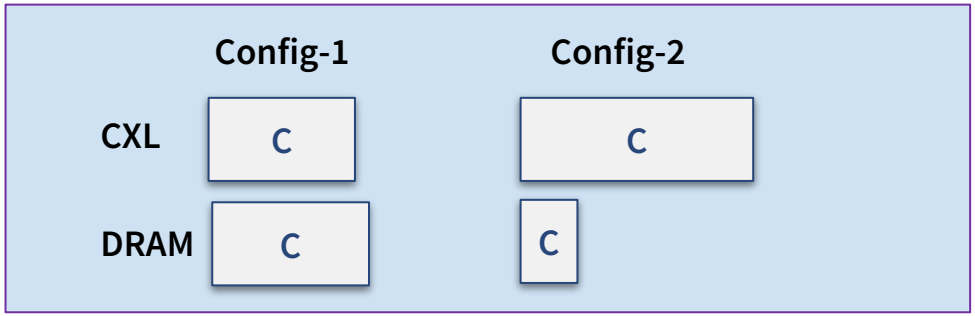
Wrong configurations increases users' cost significantly

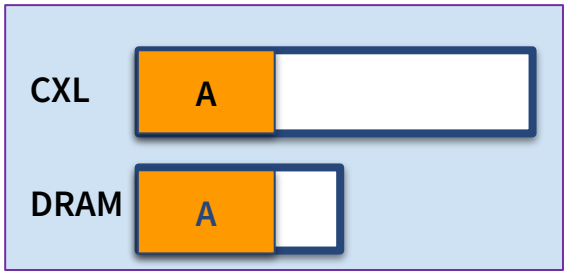
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

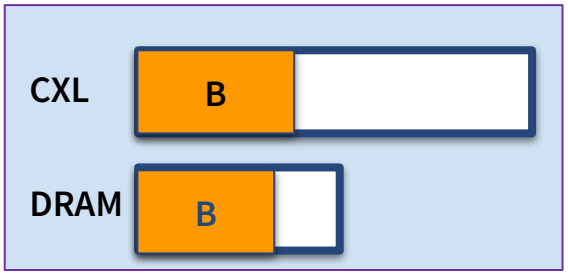
SDC 24

Cloud Provider: Highest Packing Density





Server 1



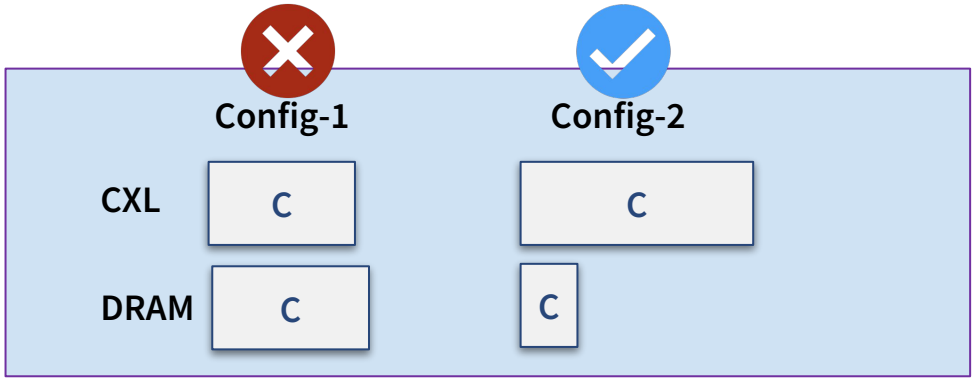
Server 2

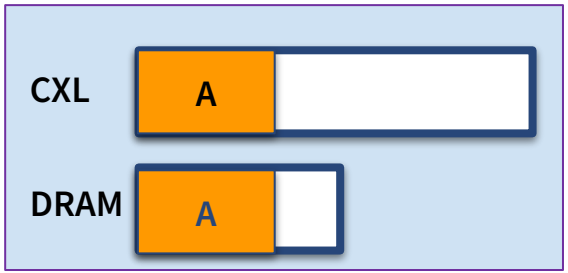
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

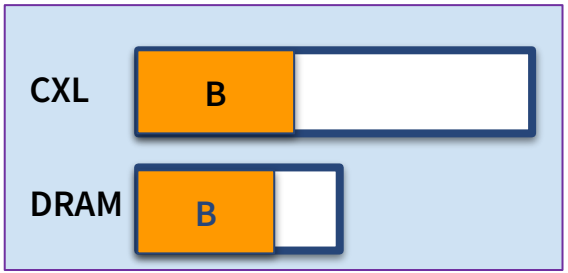
SDC 24

Cloud Provider: Highest Packing Density





Server 1



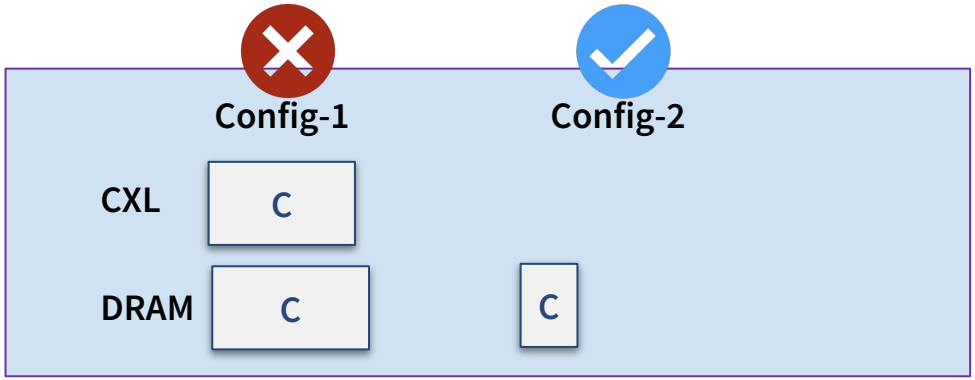
Server 2

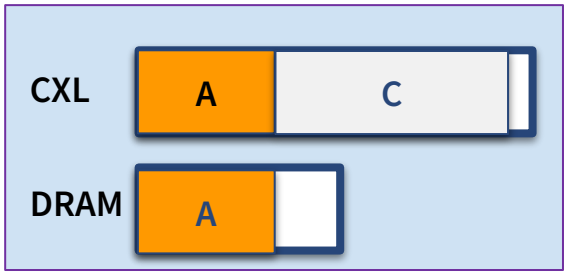
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

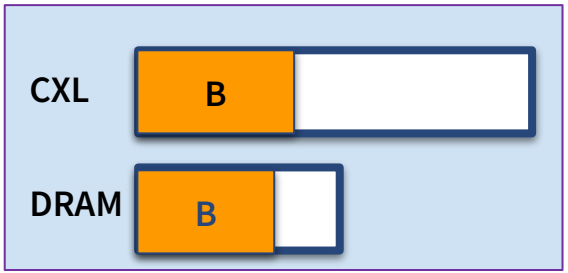
SDC 24

Cloud Provider: Highest Packing Density





Server 1



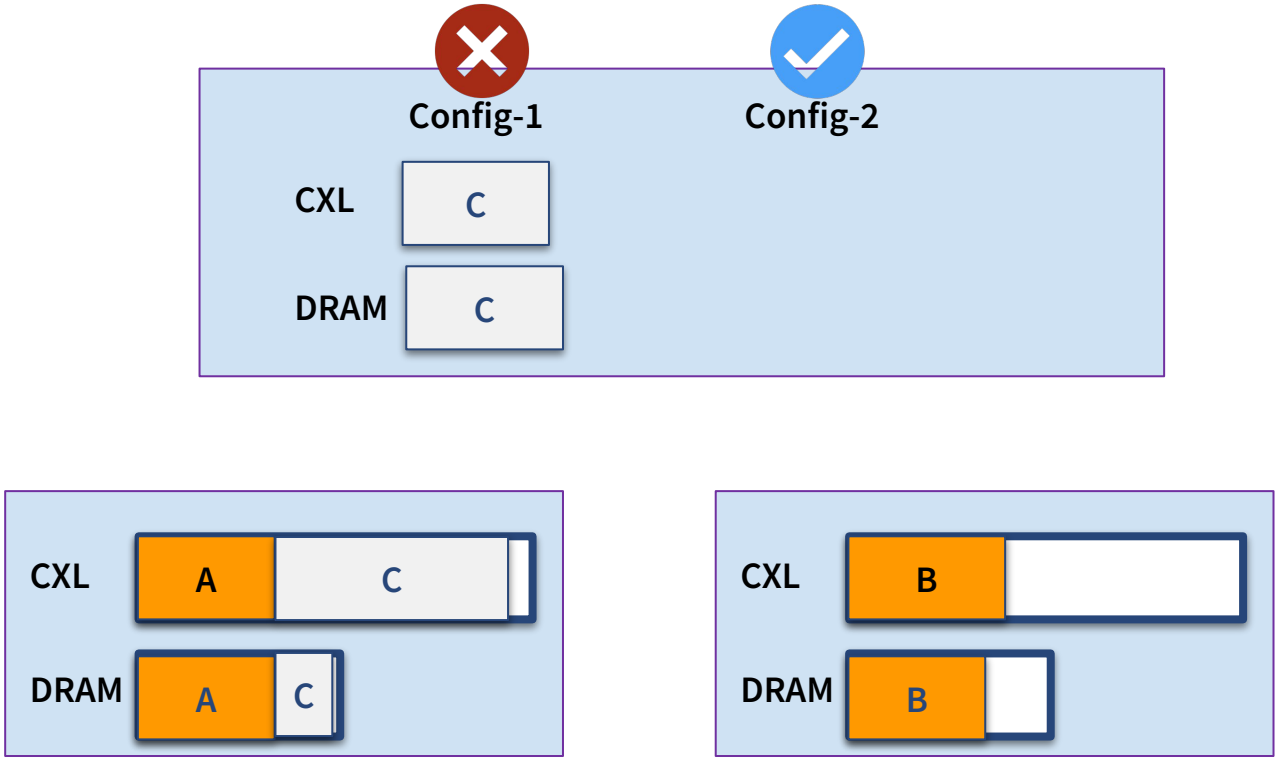
Server 2

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Cloud Provider: Highest Packing Density



Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

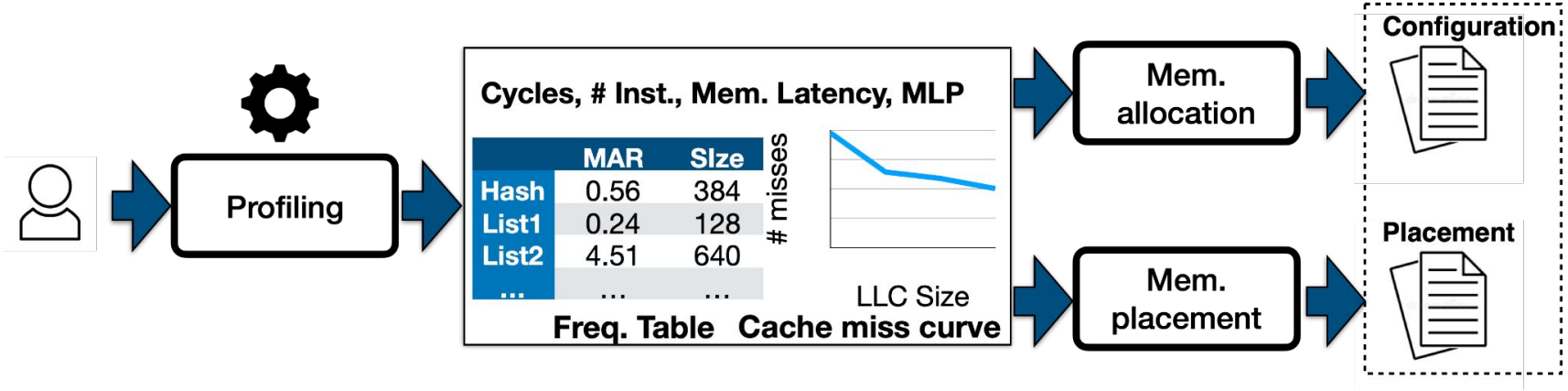
Contributions
Contributions

- Prior Work
  - Blackbox ML-based techniques Bayes[1], Collaborative Filtering [2]
  - Trained on N workloads and M configurations, predict a configuration
- Our Work (TMC)
  - White-box performance model
  - Data-layout hints (what data into CXL/DRAM?)
  - *Why* is a configuration best?
  - Predicts performance of a workload (instead of suggesting a configuration)
  - What-if analysis

[1] Alipourfard et al.: CherryPick: Adaptively Unearthing the Best Cloud Configurations for Big Data Analytics (NSDI'17).

[2] Klimovic et al.: Selecta: Heterogeneous cloud storage configuration for data analytics (ATC'18).

TMC Overview



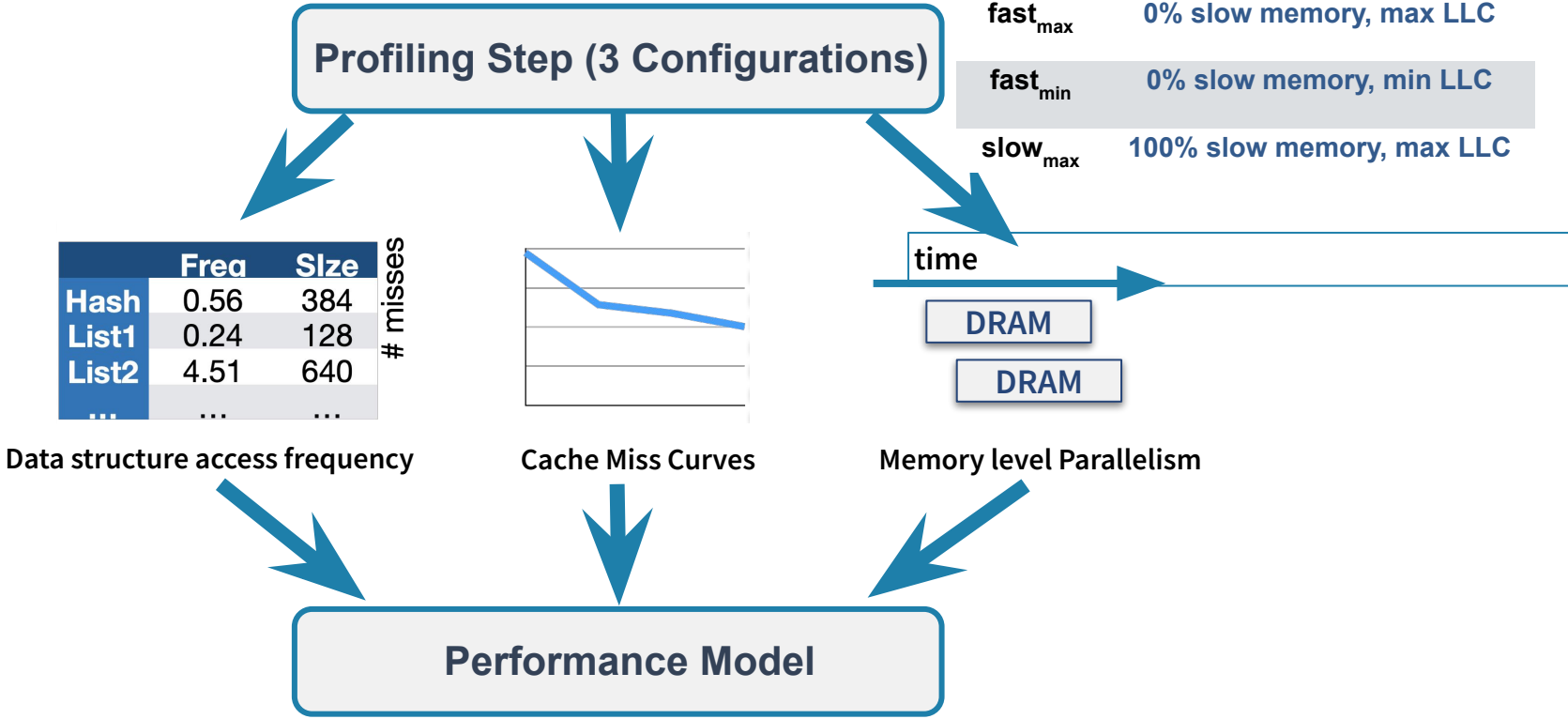
TMC devises a performance model based on the understanding of hardware performance characteristics

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Model Generation



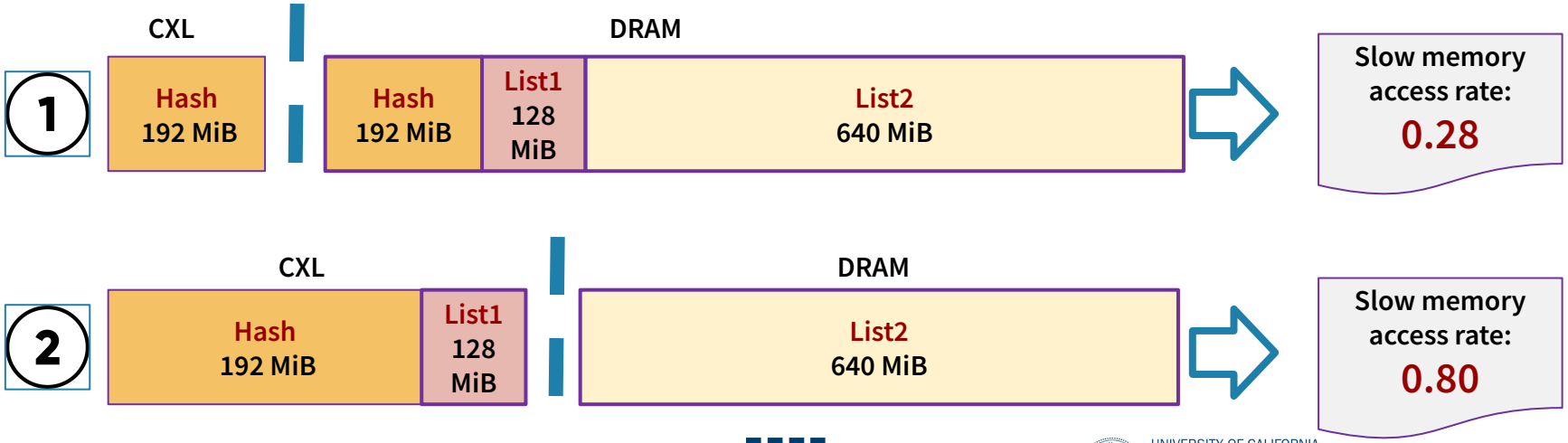
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Model Prediction

	Memory access rate	Size
List1	0.24	128 MiB
List2	4.51	640 MiB
Hash	0.56	384 MIB



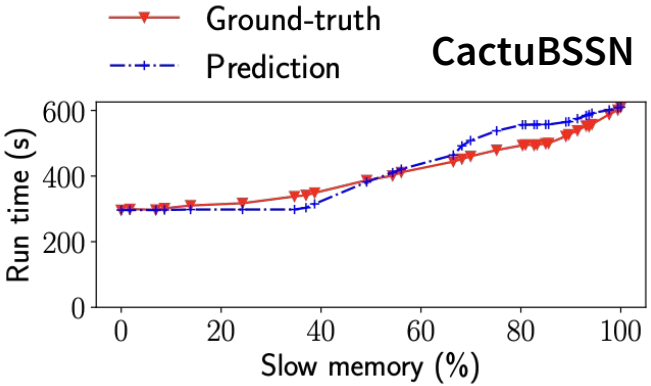
Center for Research in Systems and Storage

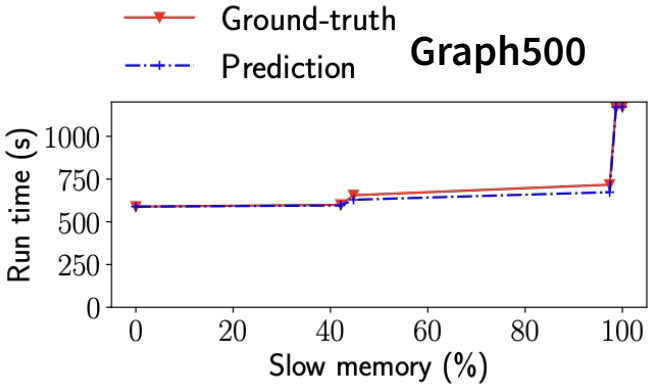
UNIVERSITY OF CALIFORNIA SANTA CRUZ

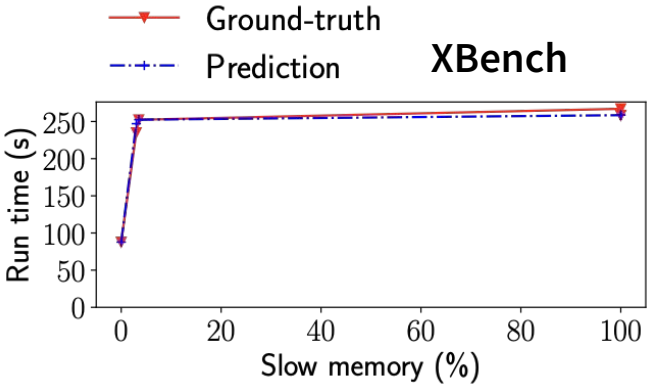
SDC 24

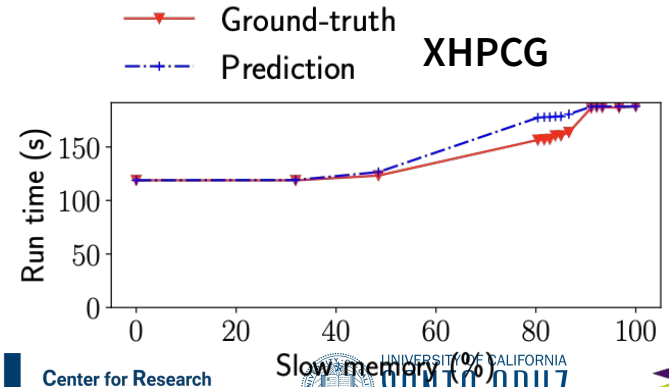
Evaluation

Evaluation Performance Prediction







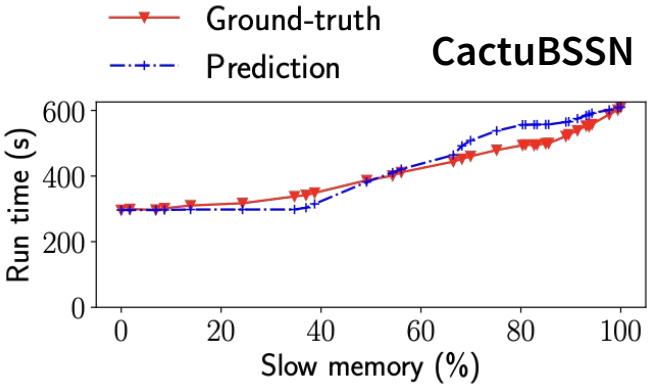


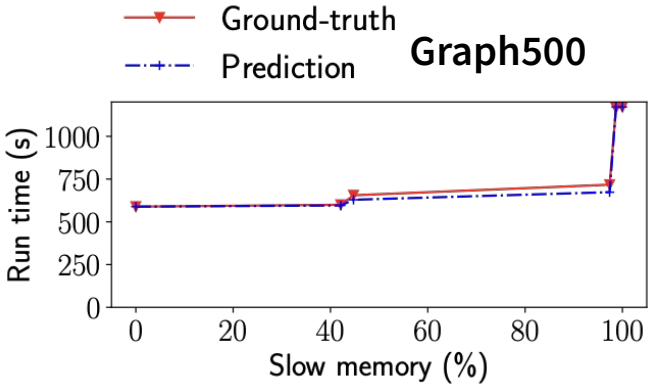
Center for Research in Systems and Storage

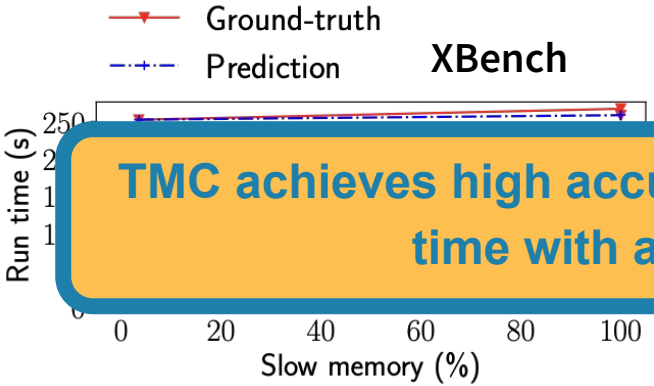
UNIVERSITY OF CALIFORNIA SANTA CRUZ

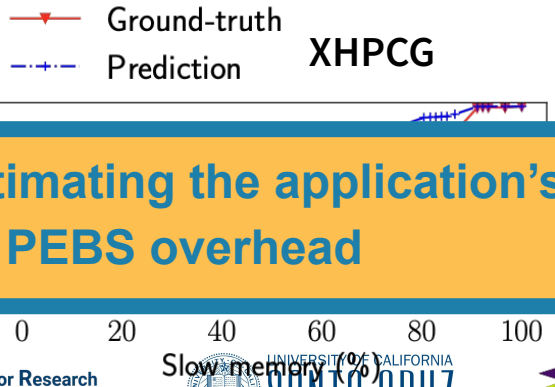
SDC 24

Evaluation Performance Prediction









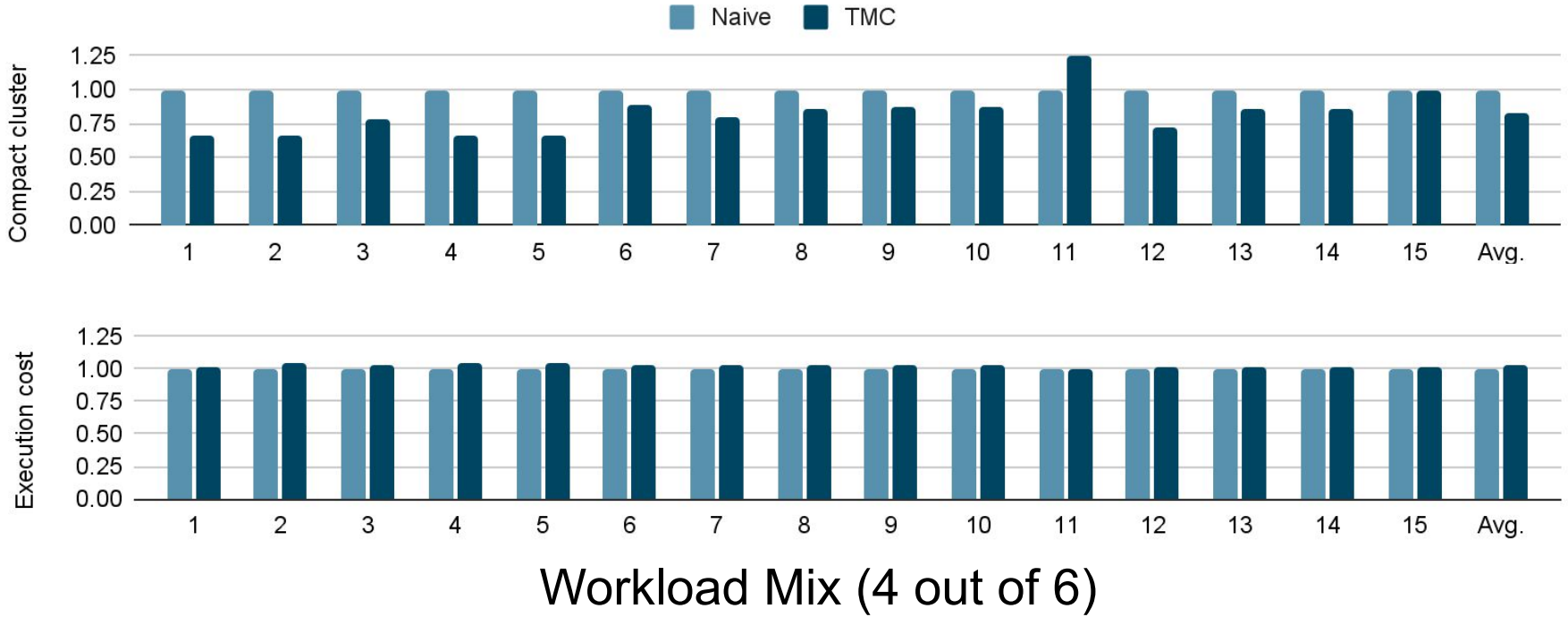
TMC achieves high accuracy in estimating the application's run time with almost zero PEBS overhead

Center for Research in Systems and Storage

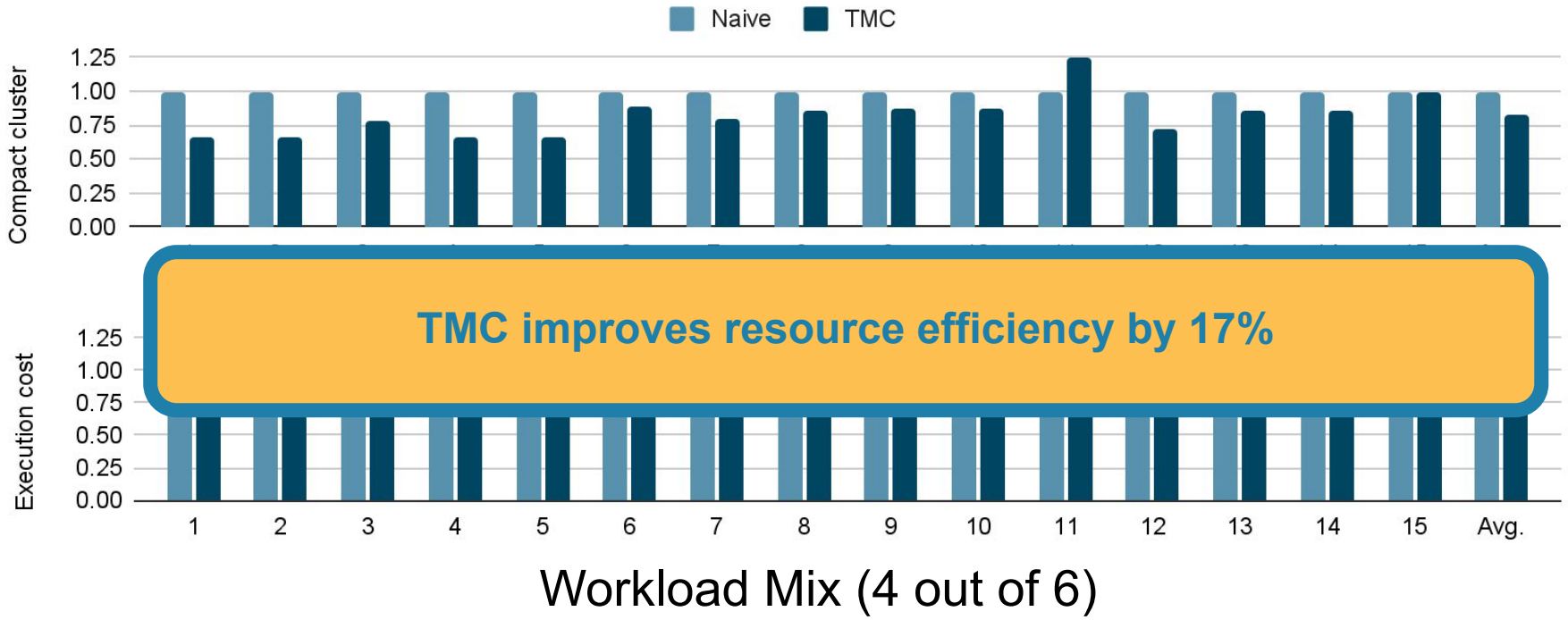
UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

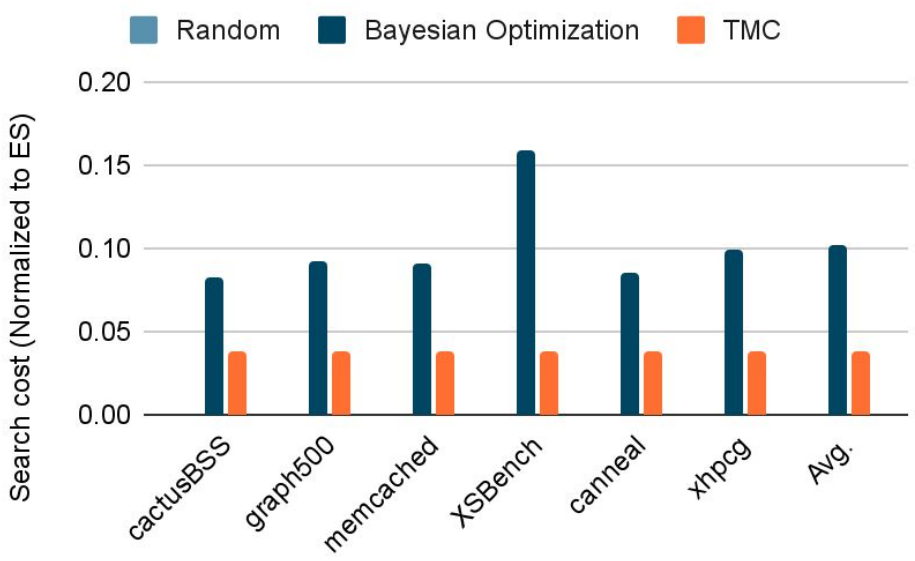
Operator Resource efficiency

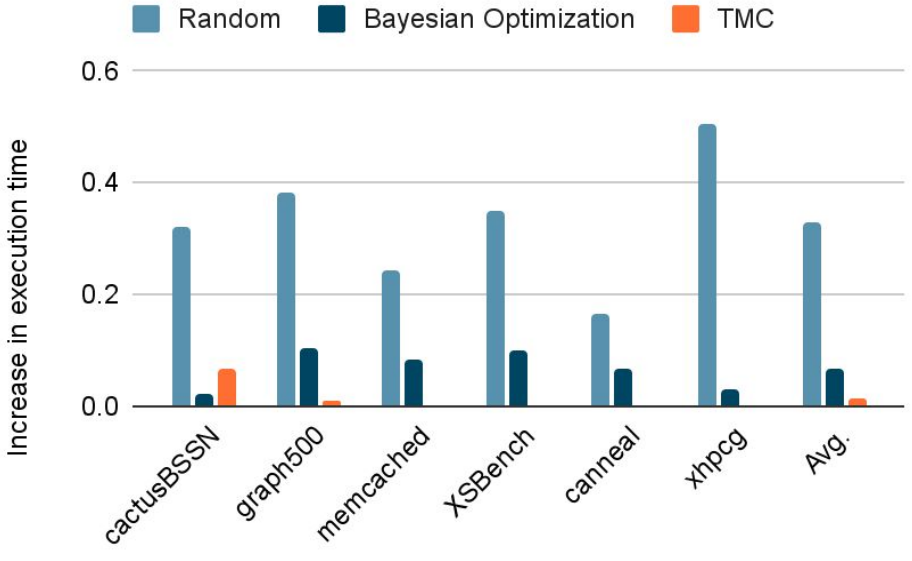


Resource efficiency



Search cost



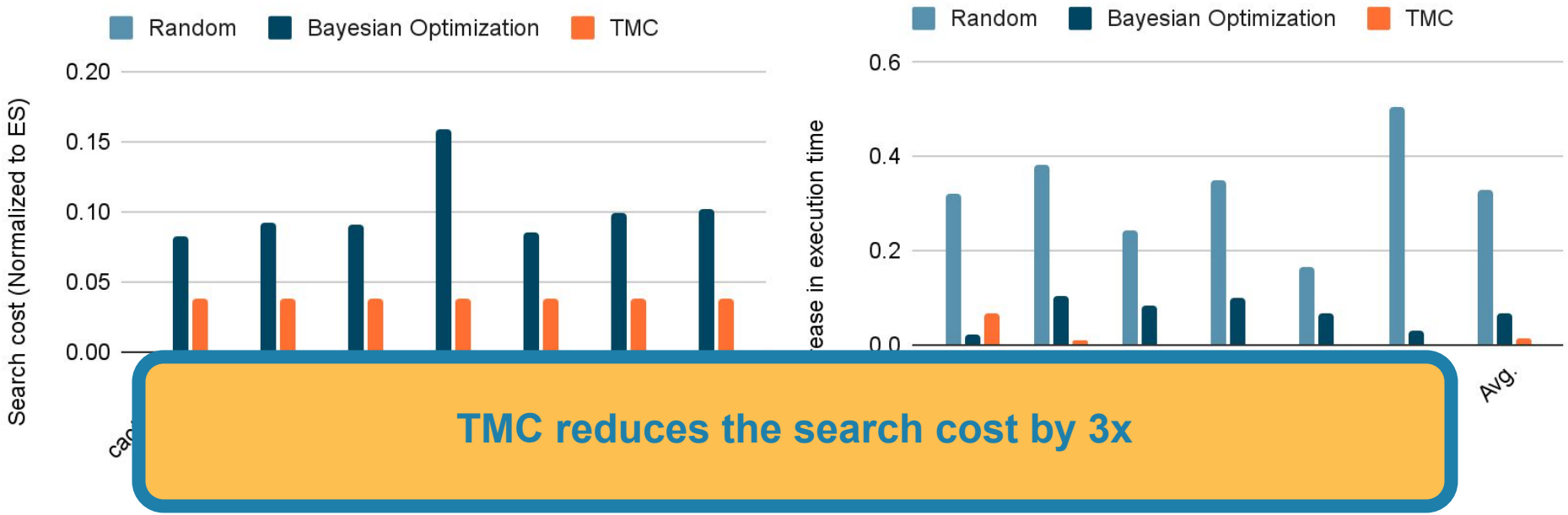


Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Search cost



TMC reduces the search cost by 3x

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Conclusion

- TMC provides a performance model over prior work
  - Selects optimal performance/TCO for the client
  - Optimizes resource allocation for the data center operator
  - Enables what-if analysis
- TMC reduces the search cost by 3x over prior work
- TMC increases resource efficiency by 17%

Scheduling

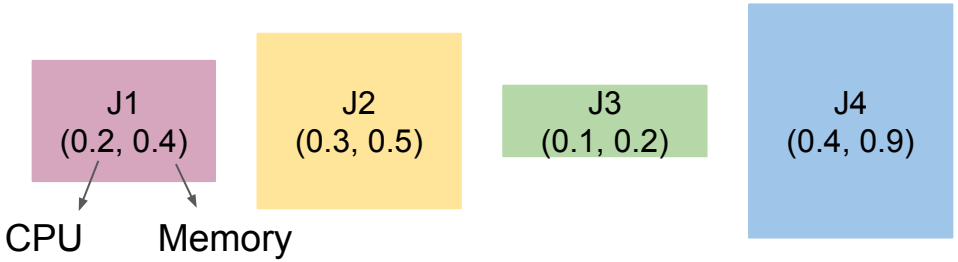
- Job Scheduling is key across many computer systems
  - Cluster management (e.g., Kubernetes, Mesos, Borg)
  - Data Analytics (e.g., Spark, Hadoop)
  - Machine Learning (e.g., PyTorch, TensorFlow)
- Efficient scheduling is crucial for large data centers
  - Even small improvements can save millions at scale

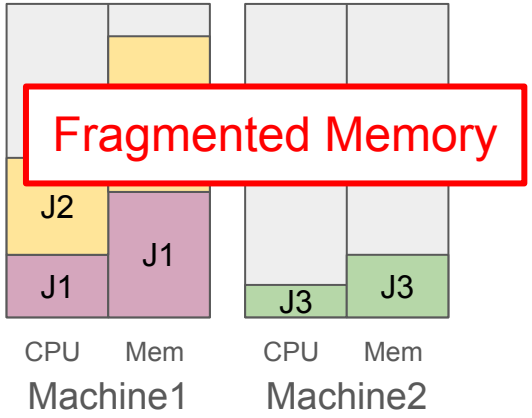
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

Scheduling Example

- Allocate data-center resources for compute jobs
- Jobs require resources, schedule assigns idle resources



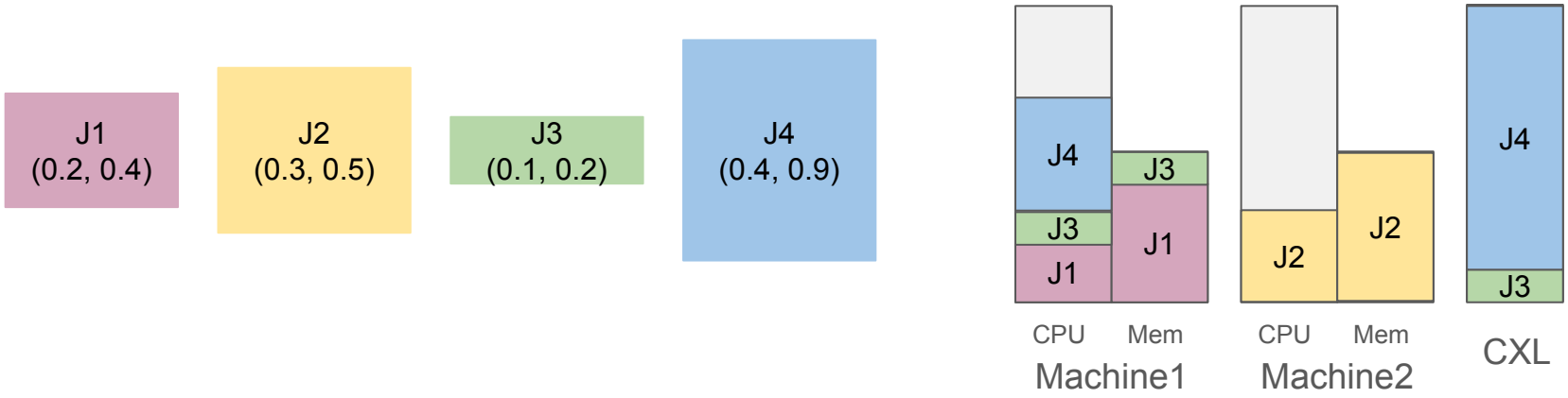


Up to 50% of Jobs face scheduling delays[1]!

[1] Tirmazi et al. Eurosys 20.

CXL to the Rescue?

- Split memory across machines and a CXL memory Pool



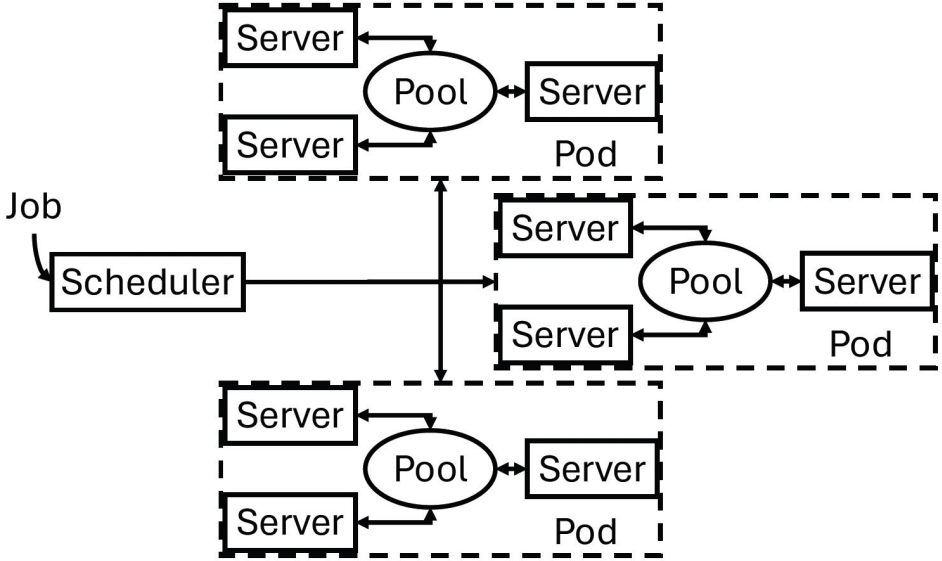
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

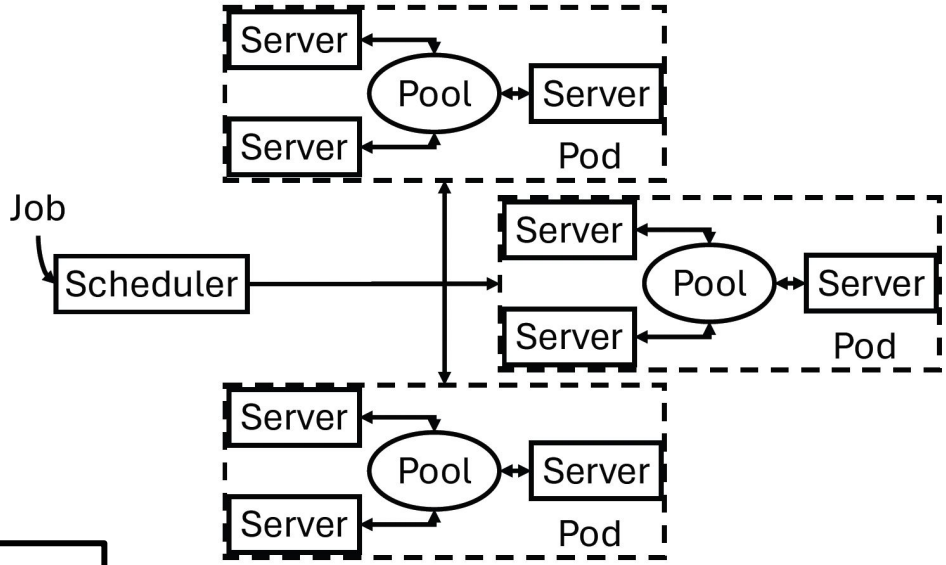
Bede

- Advantages:
  - Less scheduling delay
  - Lower cost
- Disadvantages:
  - Jobs execute more slowly



Bede Research Questions

- Bede Configuration?
  - Built cluster simulator!
- Bede Scheduler?
  - Two new schedulers!



Up to 30x faster than State-of-the-art!

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Cluster Simulator

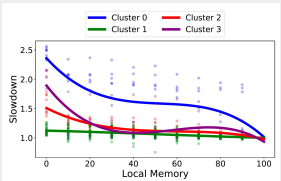
Workload Trace:

Job_id,Start,End,Cores,Memory
1,0,658200,8,16
2,0,2591400,2,4
3,0,2796400,4,32

Configuration:

- Number of servers
- Server shape
- Servers per pool
- Pool size
- Scheduling Policy

Slowdown Models:



Cluster Simulator

Job, machine, local memory, Start, Finish
1,M1,20,0,658200
2,M2,100,1500,2592900
…

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Cluster Simulator–Workloads

- Azure Cluster Traces (2017, 2019)
  - Cortez et al. SOSP 2017

- Google Borg Traces (Clusters B, D)
  - Tirmazi et al. Eurosys 2020

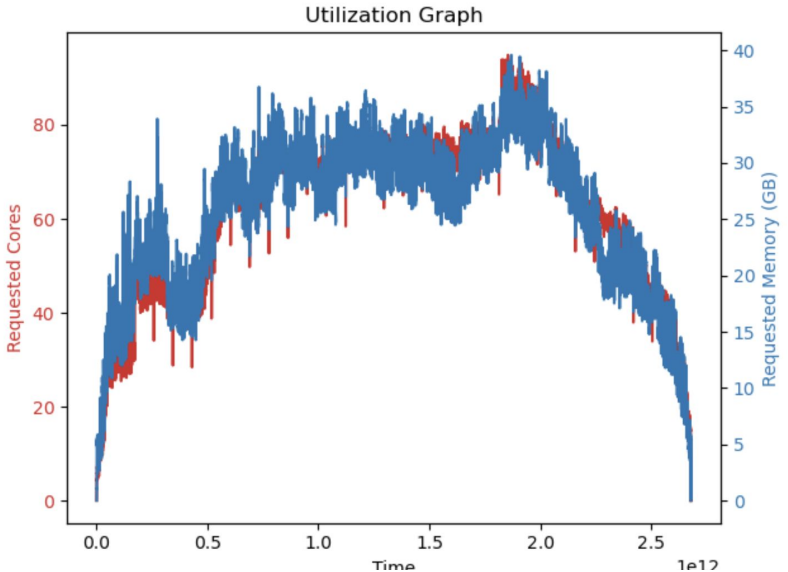
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Cluster Simulator–Configuration Methodology

- Server Shapes
  - 100th percentile of requested CPU
  - 192 cores (large cloud instance)
  - Memory at 50th, 75th, 85th, 95th percentile
- #Servers-per-pool of 2,4,...,32
- Pools of 0,10,...,100% of memory
- SOTA scheduling policies



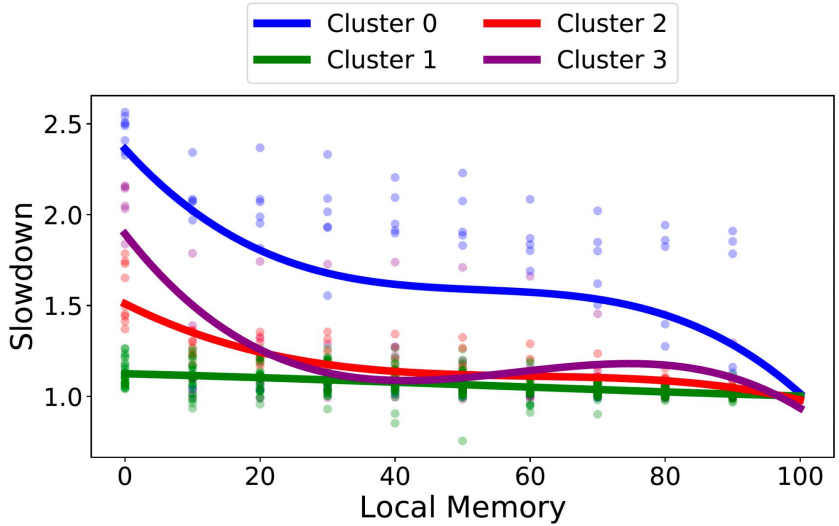
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Cluster Simulator–Slowdown Models

- Methodology:
  - Use Dual-socket NUMA
  - All compute on node 1
  - Vary memory [0–100%] across nodes
- Scale Factor (SF)
  - Account for uncertainty
  - Multiplies NUMA models by constant factor
  - SF 2 means CXL twice as slow as NUMA.



Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Cluster Simulator

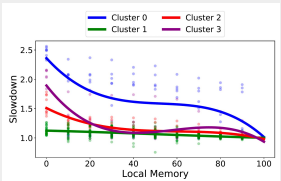
Workload Trace:

Job_id,Start,End,Cores,Memory
1,0,658200,8,16
2,0,2591400,2,4
3,0,2796400,4,32

Configuration:

- Number of servers
- Server shape
- Servers per pool
- Pool size
- **Scheduling Policy**

Slowdown Models:



Cluster Simulator

Job, machine, local memory, Start, Finish
1,M1,20,0,658200
2,M2,100,1500,2592900
…

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Scheduling Policies

- Existing State-of-the-art
  - Generic: FIFO, SJF
  - Far-memory specific: CFM, Pond

- Novel alignment-based policies
  - EVPM-Far: FIFO with alignment
  - T(etris)-Far: SJF with alignment

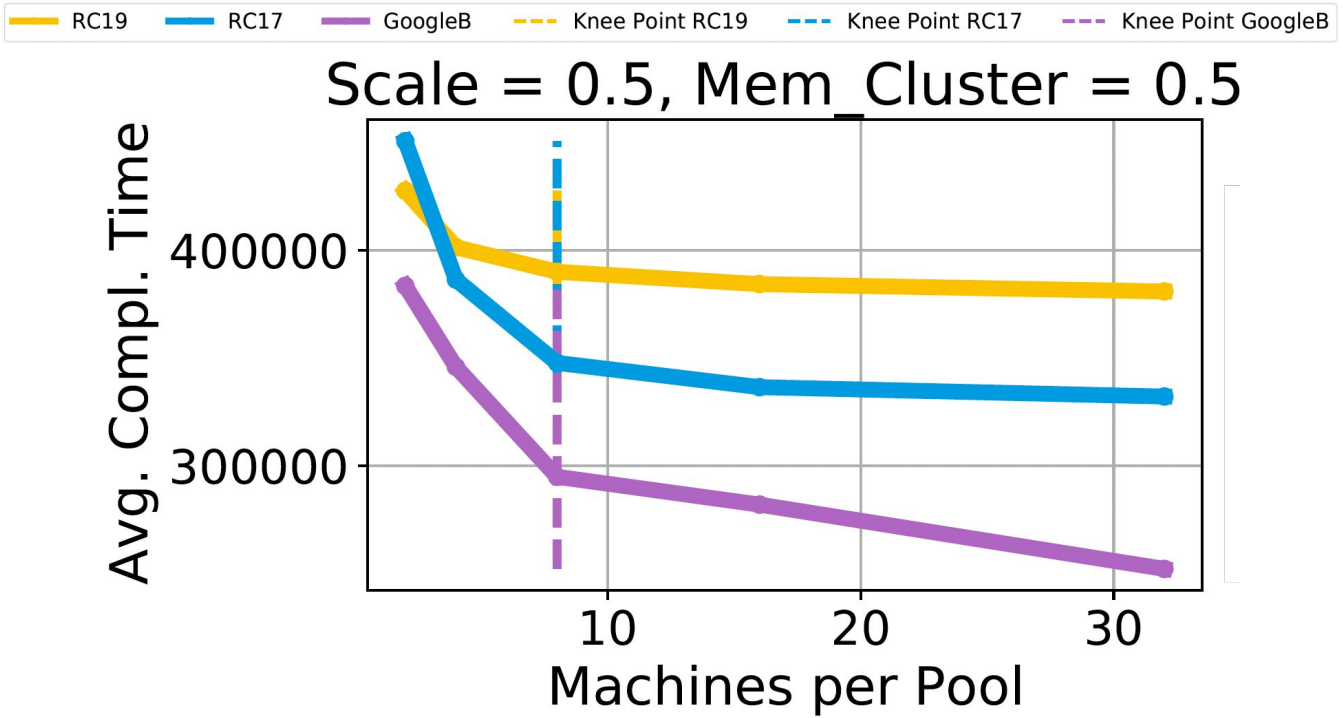
$$L = \min(mem_{Server}, mem_{Job})$$
$$A = \langle core_{Server}, mem_{Server}, mem_{pool} \rangle$$
$$R = \langle core_{Job}, L, 1 - L \rangle$$

$$Alignment = A \cdot R$$

Evaluation

- How many servers should be attached to each pool?
- How should memory be split between servers and pools?
- How does job performance vary by scheduling policy?

Servers Per Pool

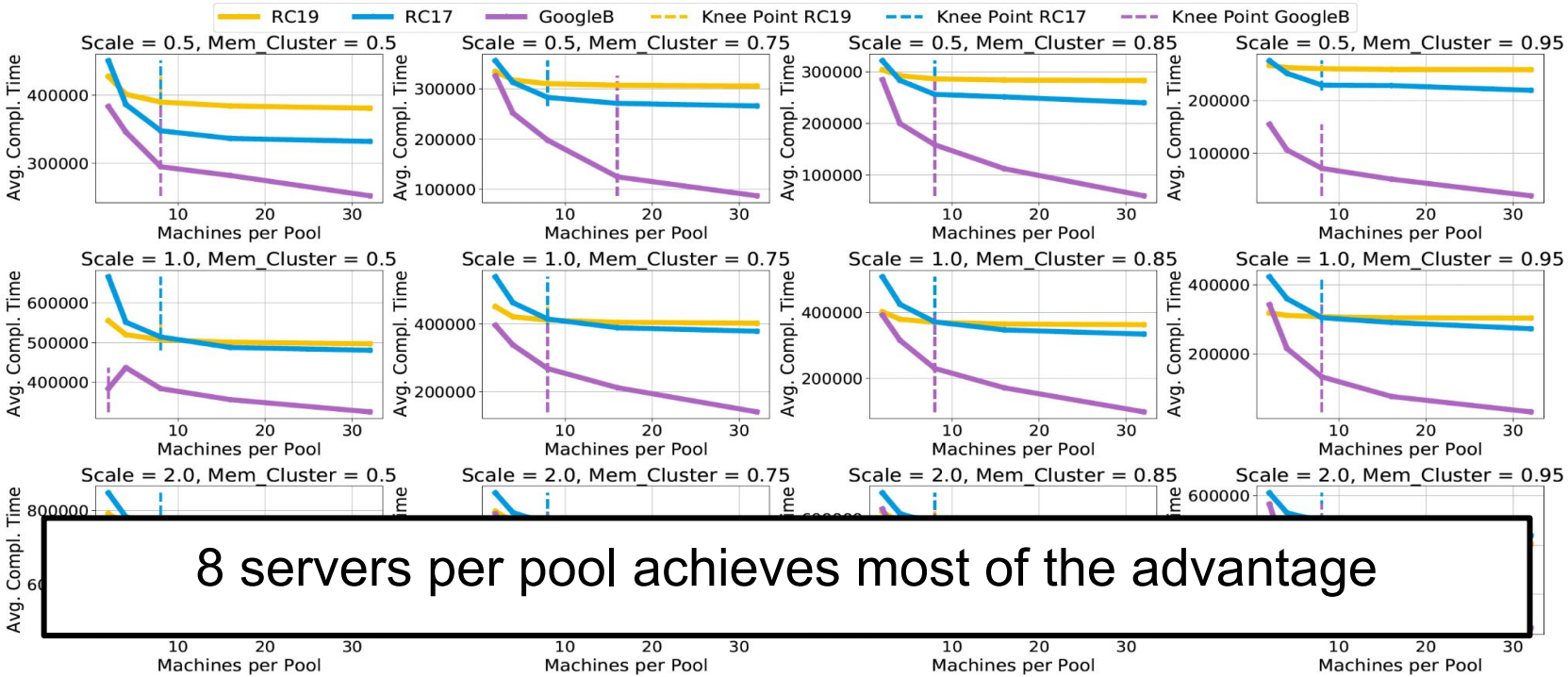


Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Servers Per Pool



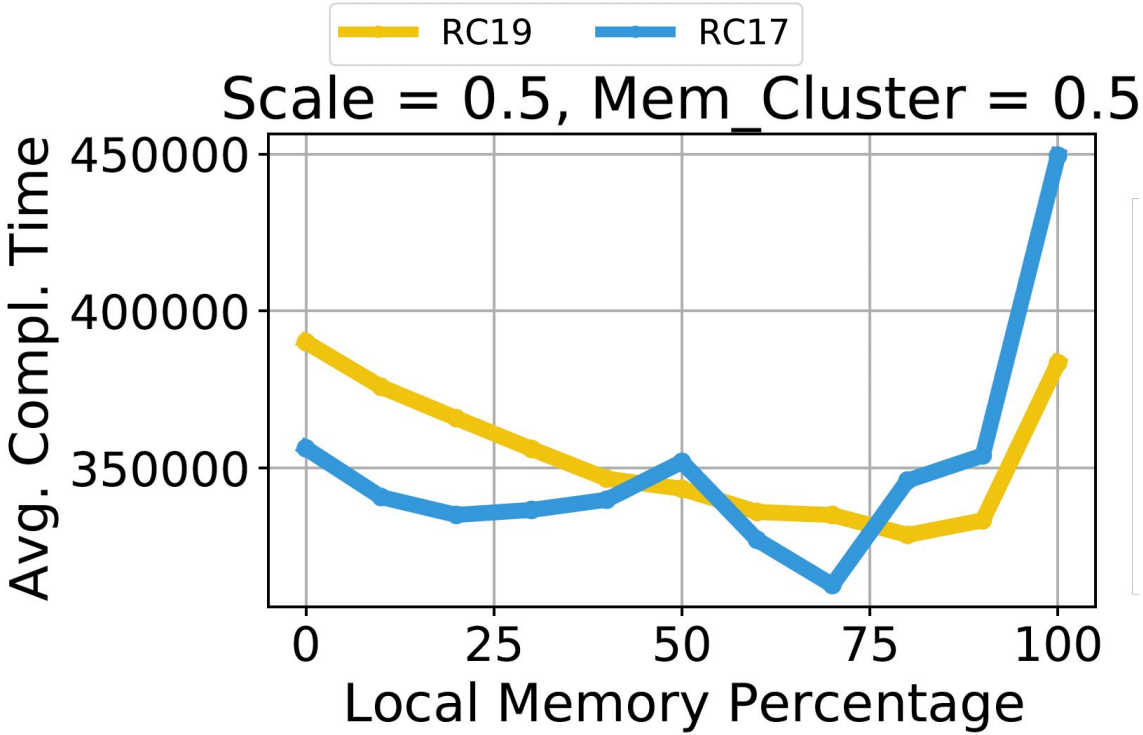
8 servers per pool achieves most of the advantage

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Pool server memory split

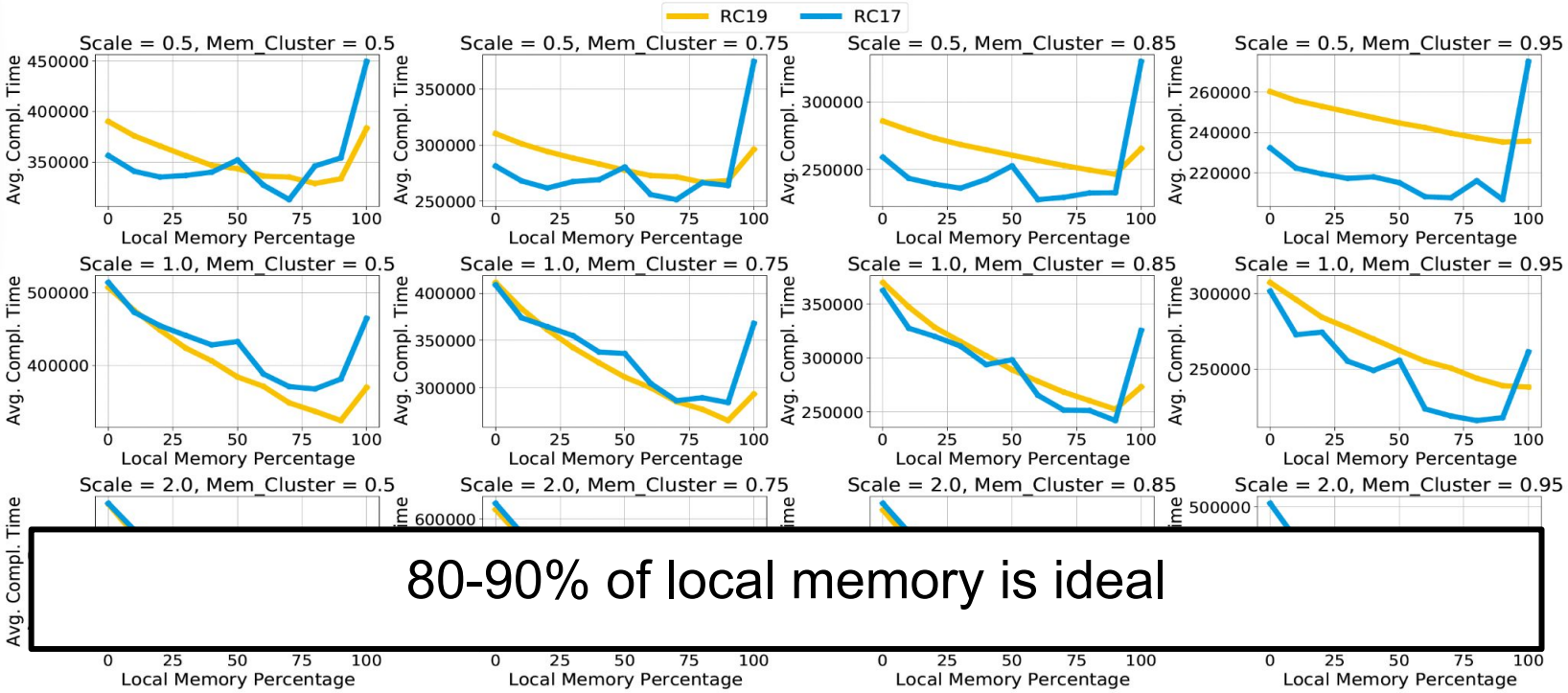


Center for Research in Systems and Storage

SANTA CRUZ

SDC 24

Pool server memory split

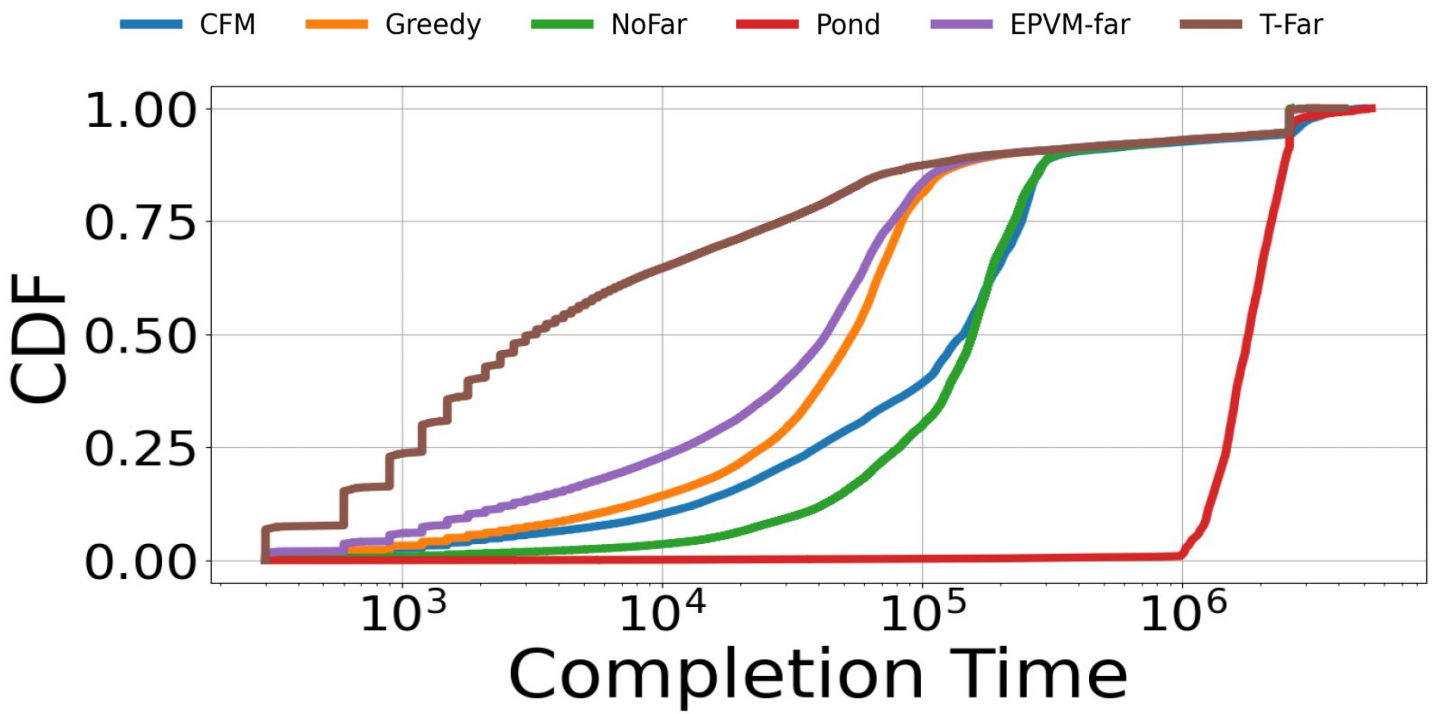


Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Scheduling Policies

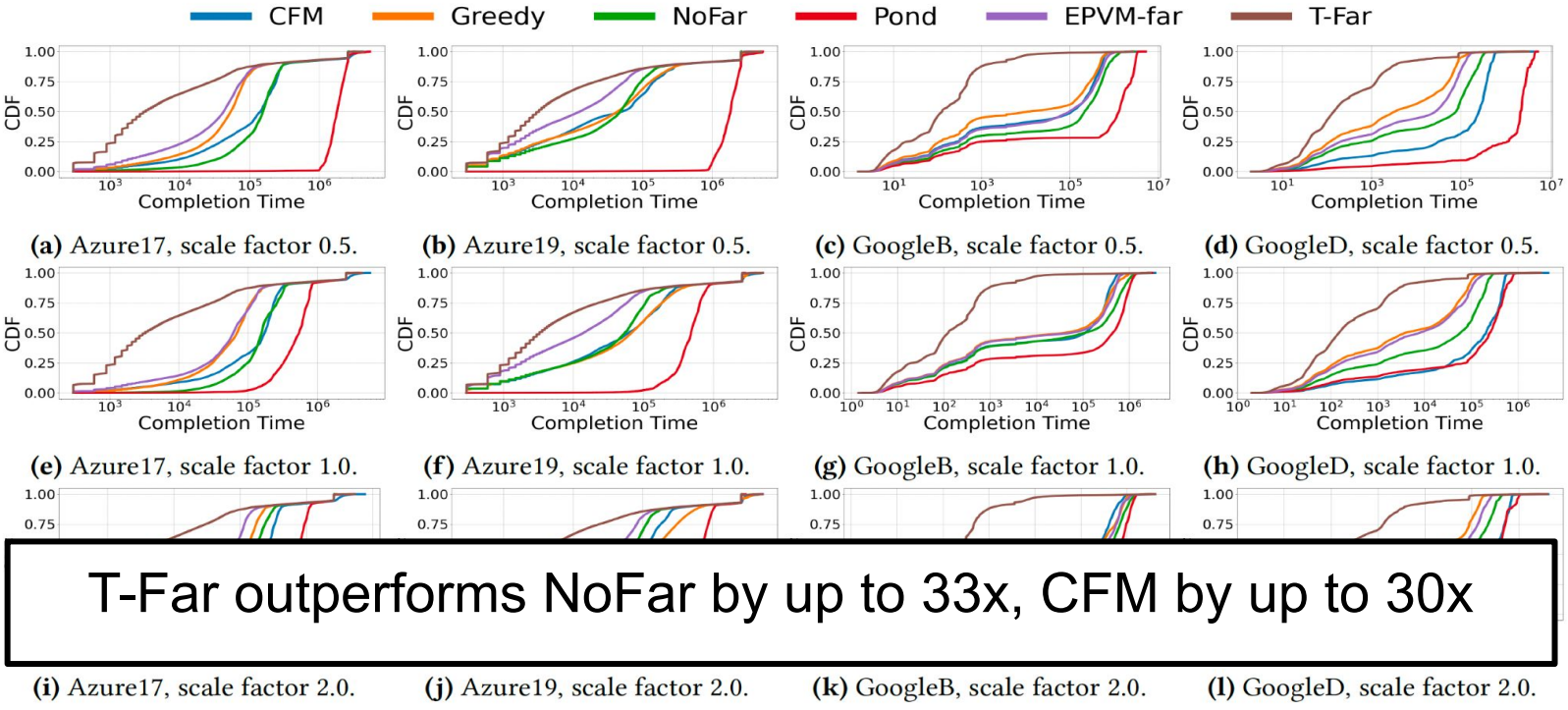


Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Scheduling Policies



T-Far outperforms NoFar by up to 33x, CFM by up to 30x

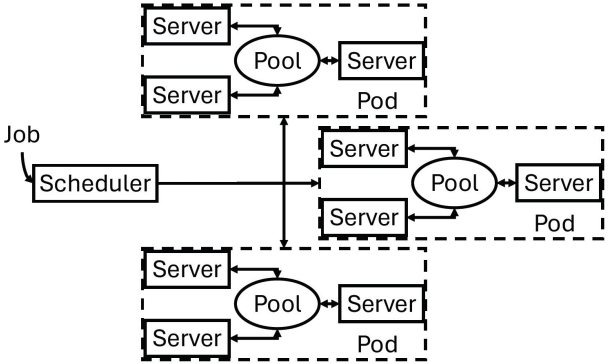
Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Bede Conclusion

- Built simulator to explore configurations
  - Small pools work well
- Two novel scheduling algorithms
  - Up to 30x improvement over state of the art



Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24

Contributions

- CXL is a promising technique to address memory cost
- Not a plug-in replacement, many deployment challenges
- Our work enables modeling of CXL performance & TCO
- Automation can address the complexity challenges of CXL

Contributions

- CXL is a promising technique to address memory cost
- Not a plug-in replacement, many deployment challenges
- Our work enables modeling of CXL performance & TCO
- Automation can address the complexity challenges of CXL

Please reach out if you want to collaborate with us: www.crss.us

Center for Research in Systems and Storage

UNIVERSITY OF CALIFORNIA SANTA CRUZ

SDC 24