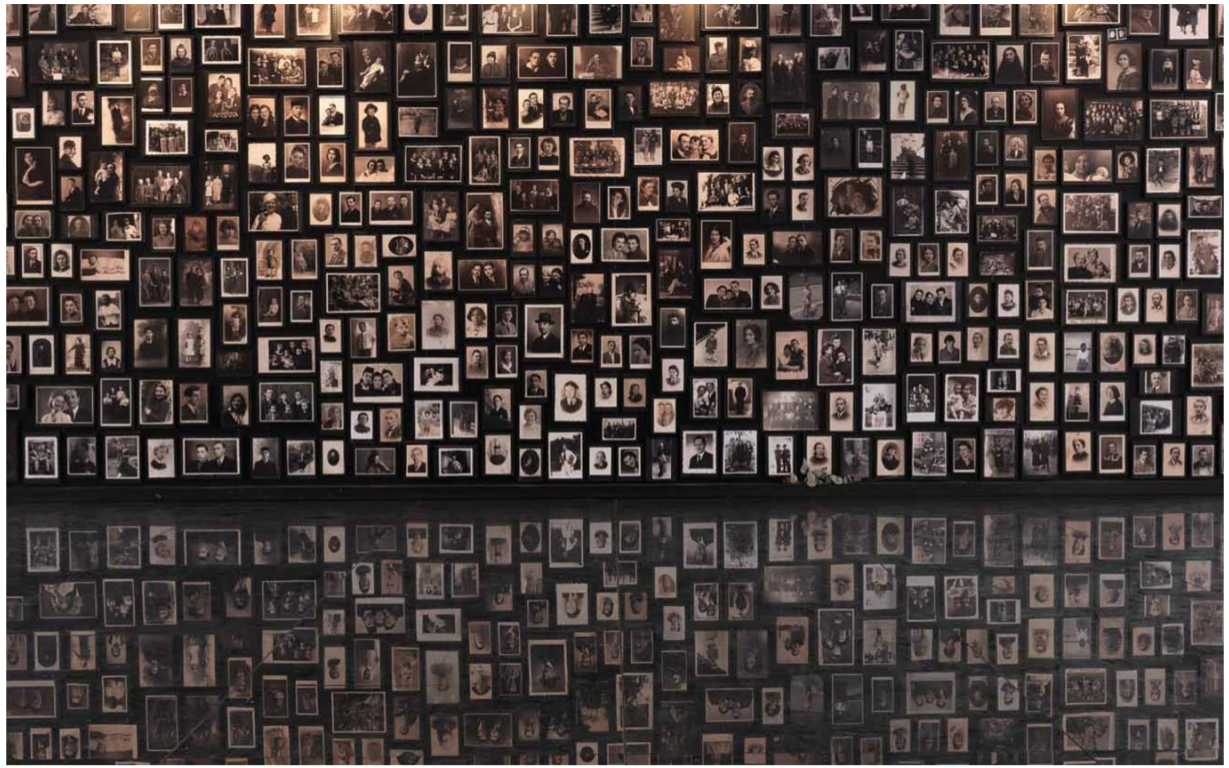
# Measuring CPU Cache Resources in Clouds

Weijia Song

2018-02-28

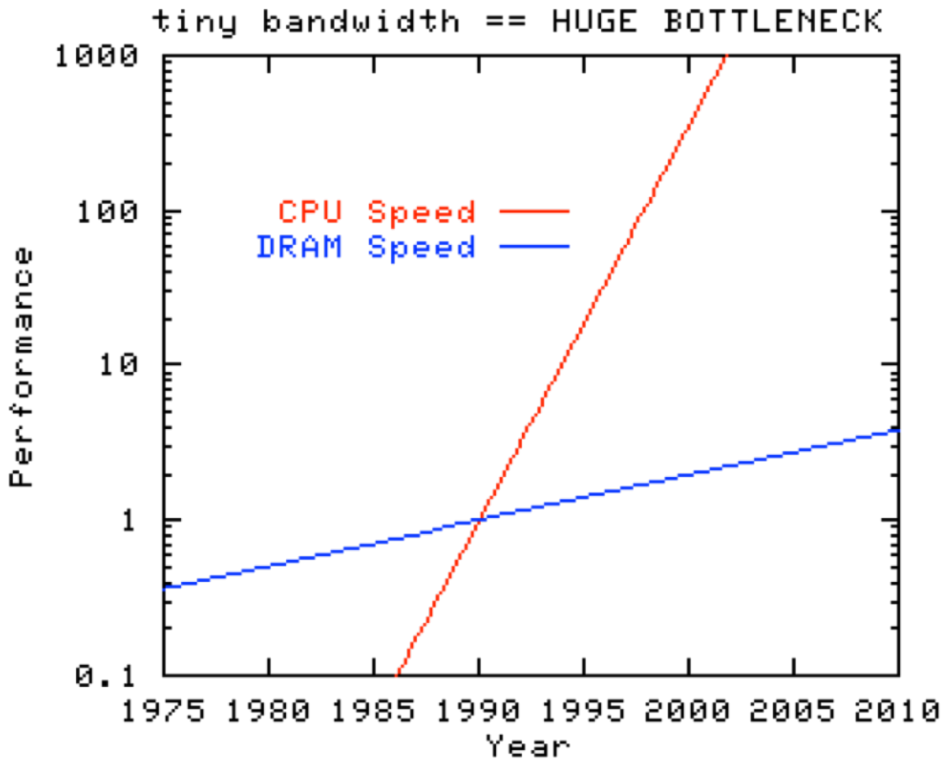
# A Reminder to CPU Cache

- The ***Memory Wall*** problem. (Sally A. McKee, 1995)



# A Reminder to CPU Cache

- The “Memory Wall”




John McCalpin, STREAM

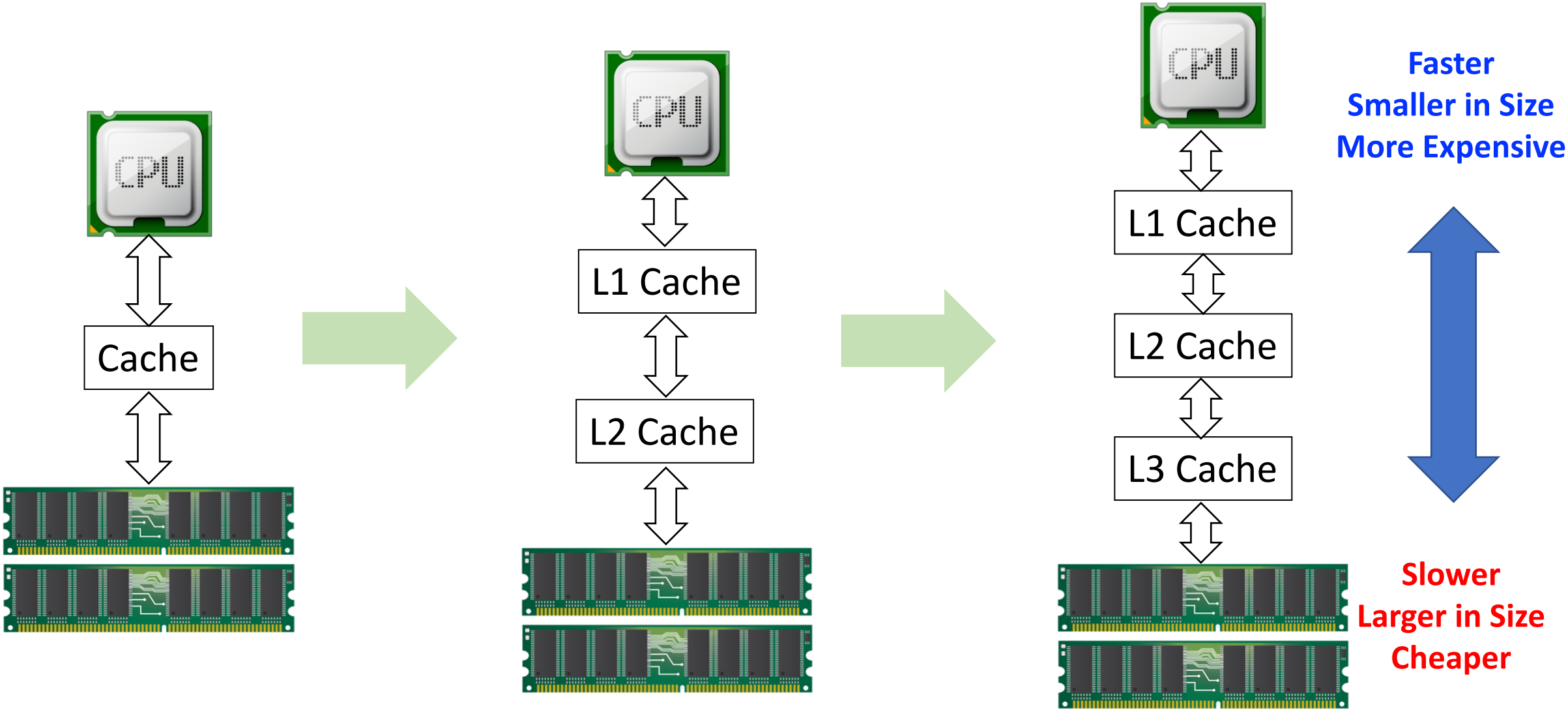
What makes it WORSE?

- Multi-core CPUs
- Parallelism inside the core: MMX, SSE, AVX, FMA…

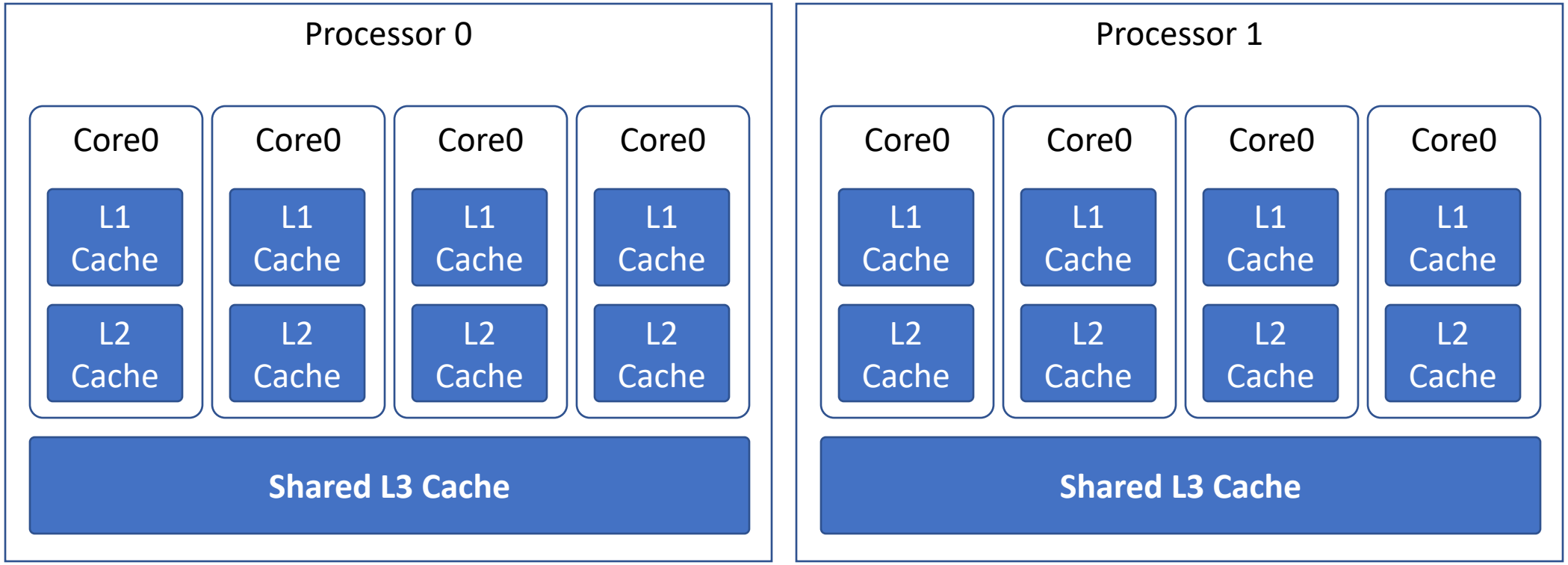

**CPU : ~900 GFlops**
**(single precision floating-point)**

**MEM: 76.8 GB/s**

# A Reminder to CPU Cache

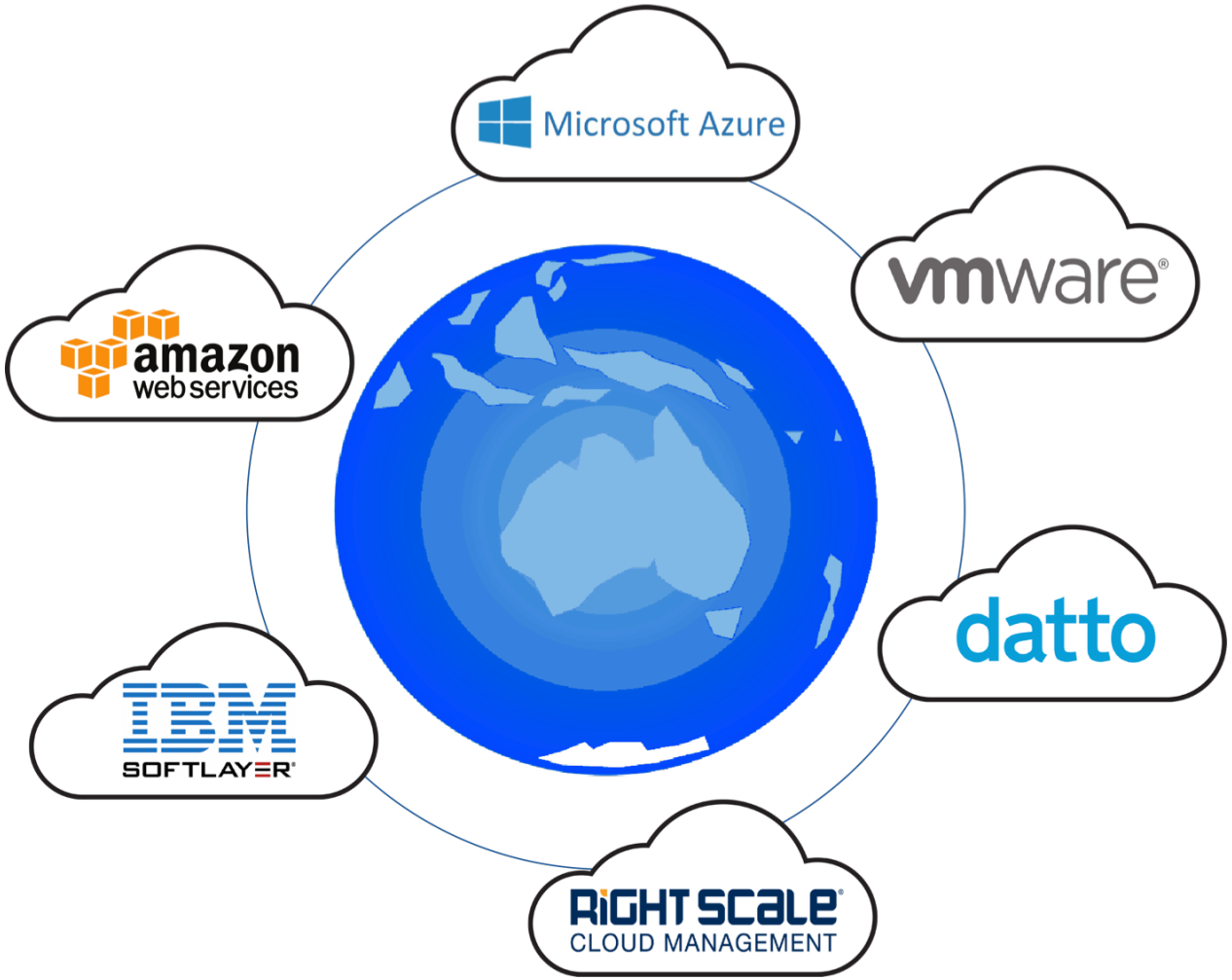


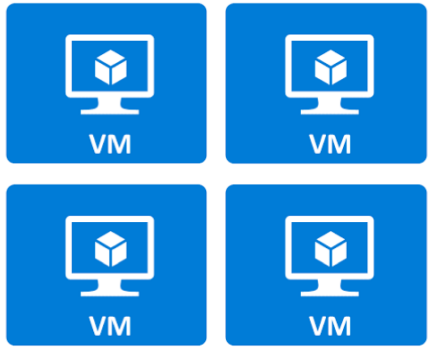
# A Reminder of CPU Cache

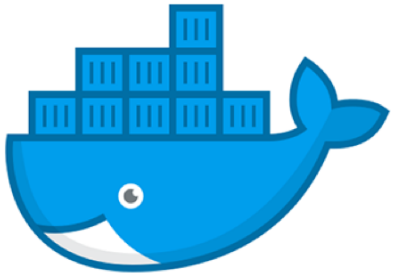



The most popular architecture in the Cloud

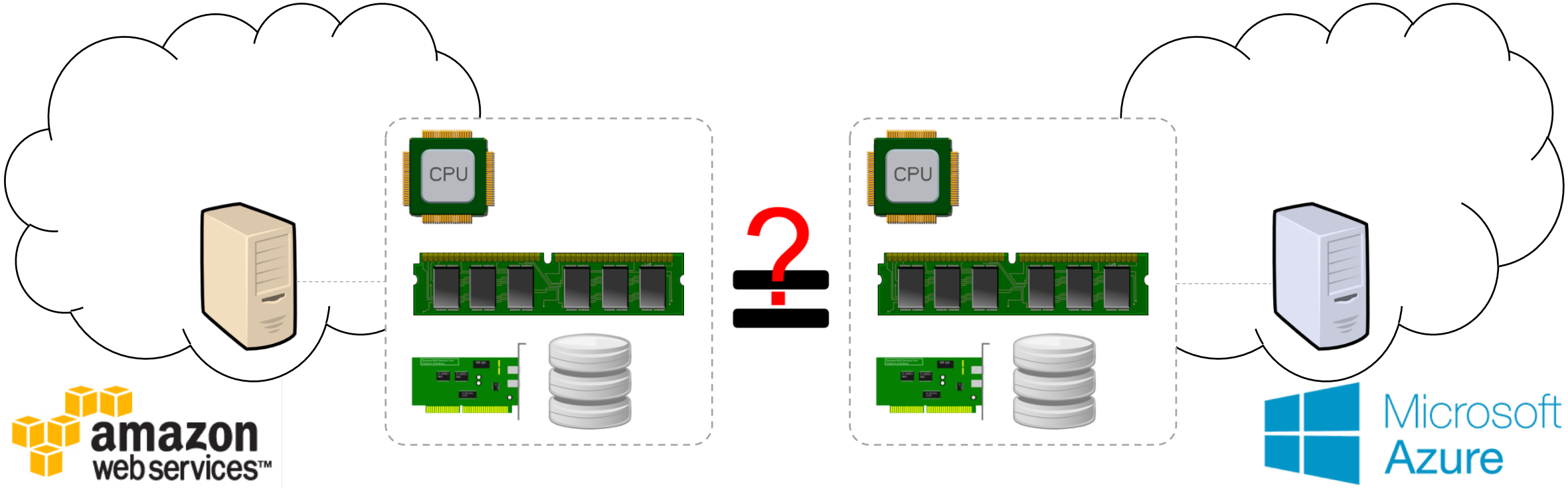
# The Infrastructure-as-a-Service(IaaS) Clouds



# The Problem



# No, Why?

- Different hardware
- Different hypervisors
- Different resource sharing levels
- Isolation: Cache Allocation Technology, Memory Bandwidth Allocation
- Document errors

- However, virtual machine specification presented to user by cloud providers is not sufficiently descriptive, making it hard for application to find the best deployment.

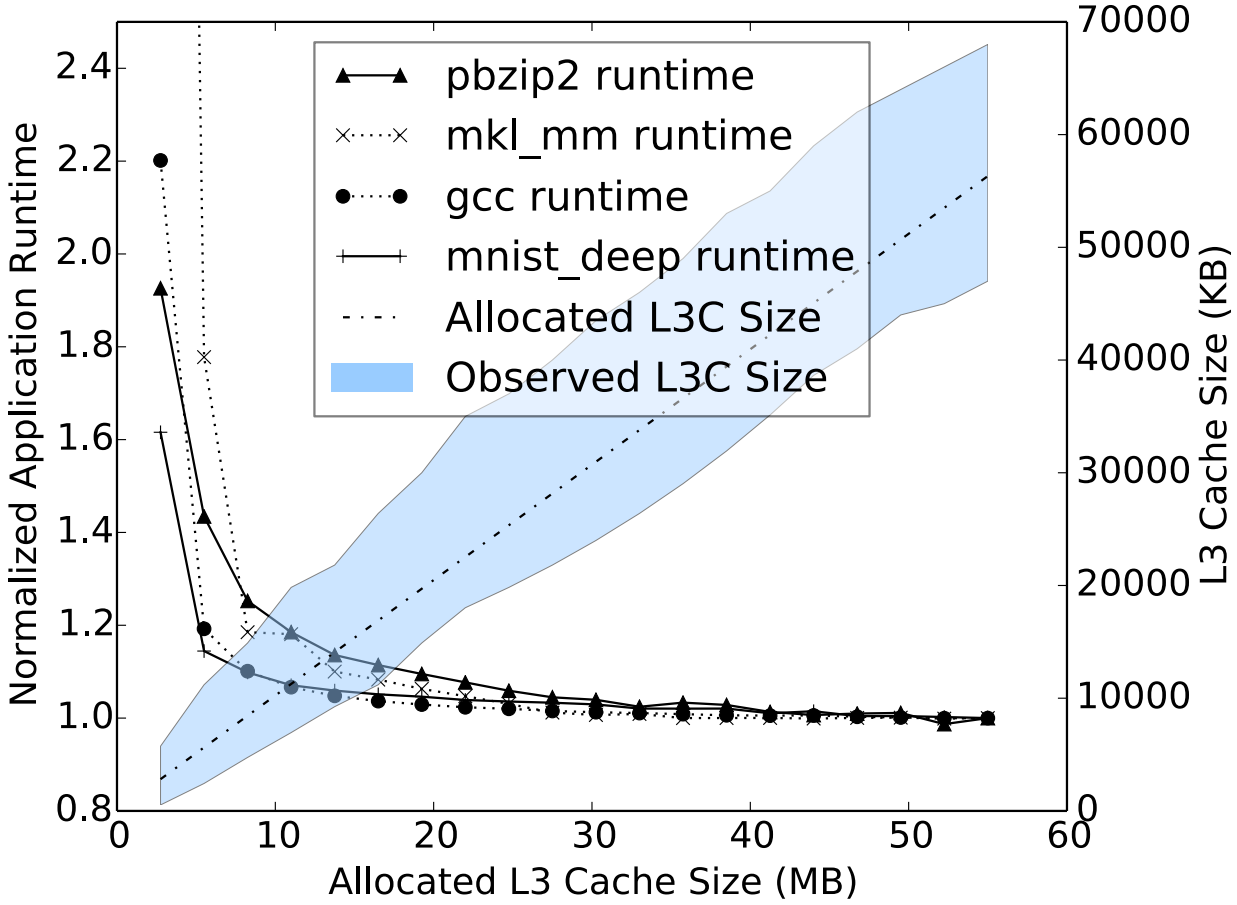
# CloudModel: A tool measuring the resources in the Cloud.

This work focuses on CPU Cache and Memory Resources. It measures

- The true allocated size of each cache level;
- The sequential throughput and latency of each cache level; and
- The sequential throughput and latency of memory.

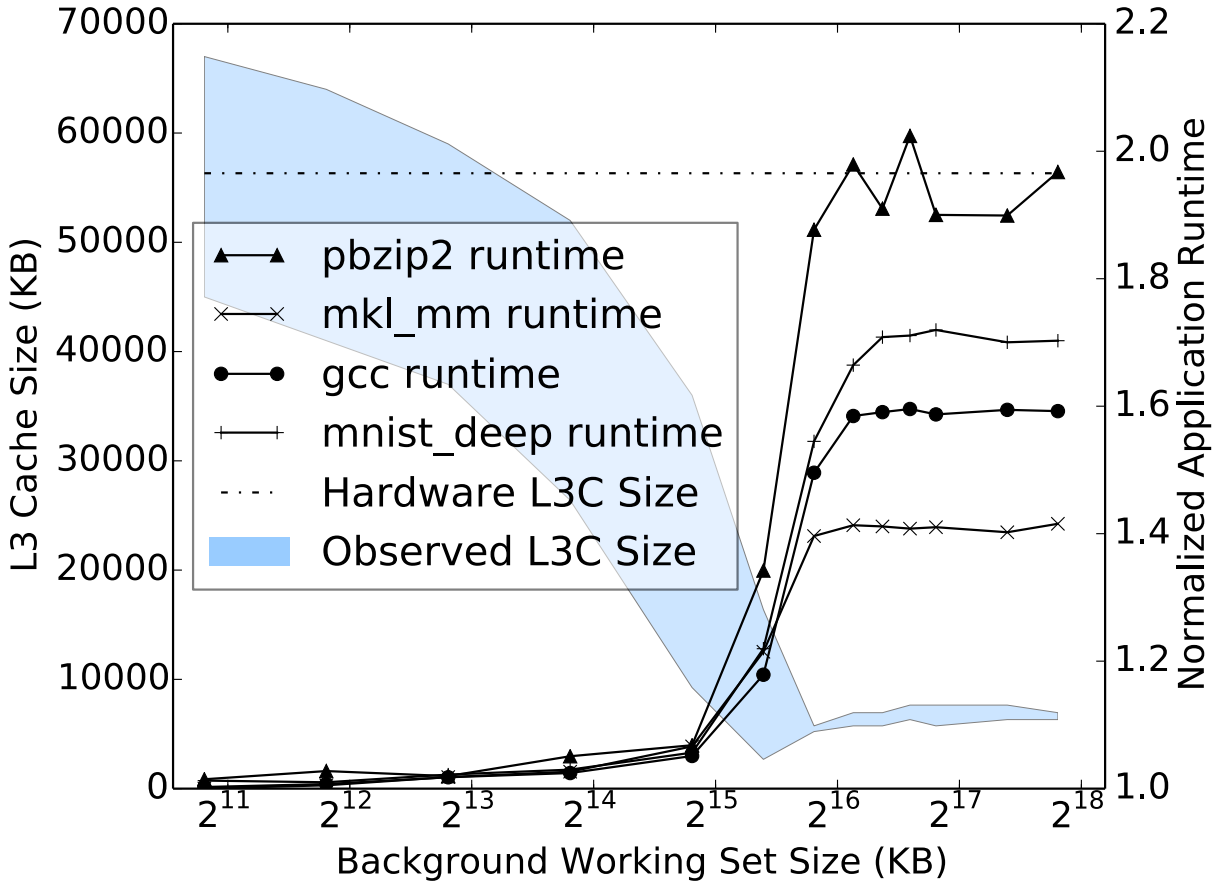
# Cache Allocation vs Application Performance

- Application performance with Cache allocation



# Cache Contention vs Application Performance

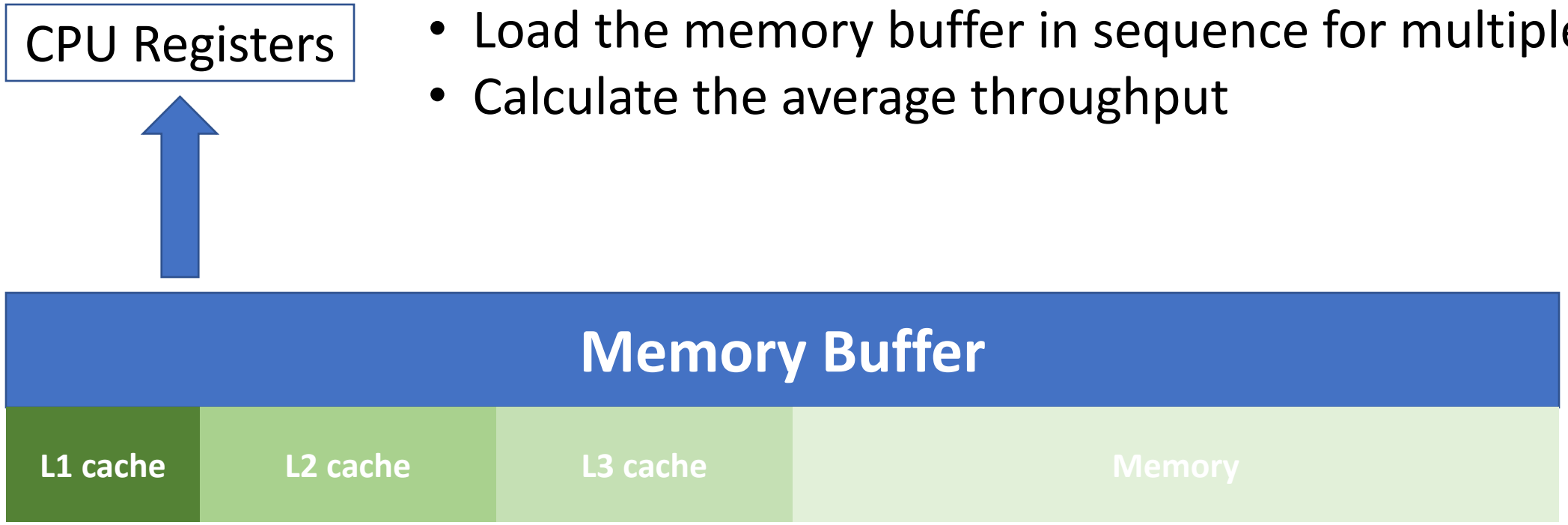
- Application Performance with Cache contention



# Performance Tuning with Cache Size

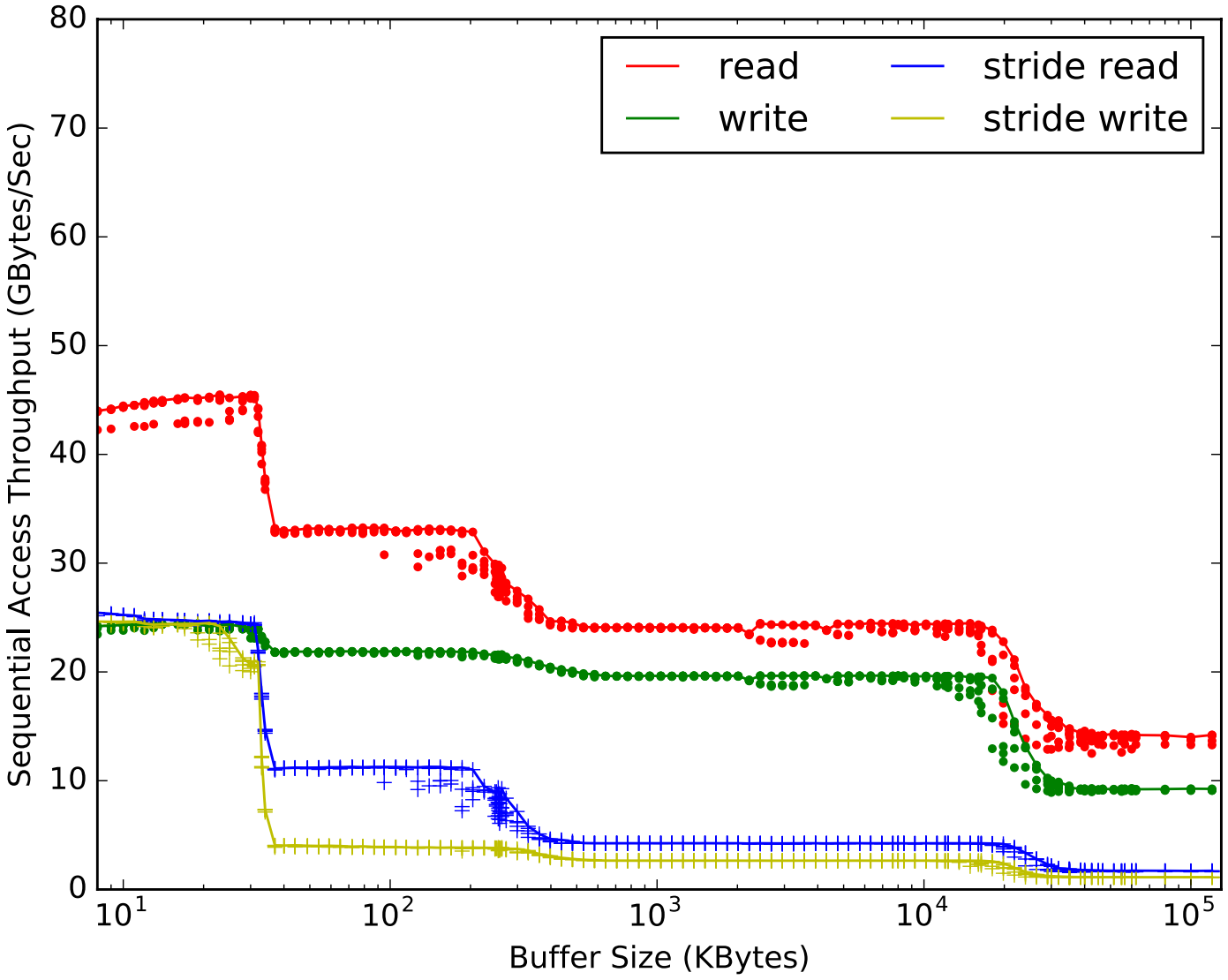
- Intel SSE temporal/non temporal data
- Protean Project

# Measuring Throughput




- Load the memory buffer in sequence for multiple times
- Calculate the average throughput

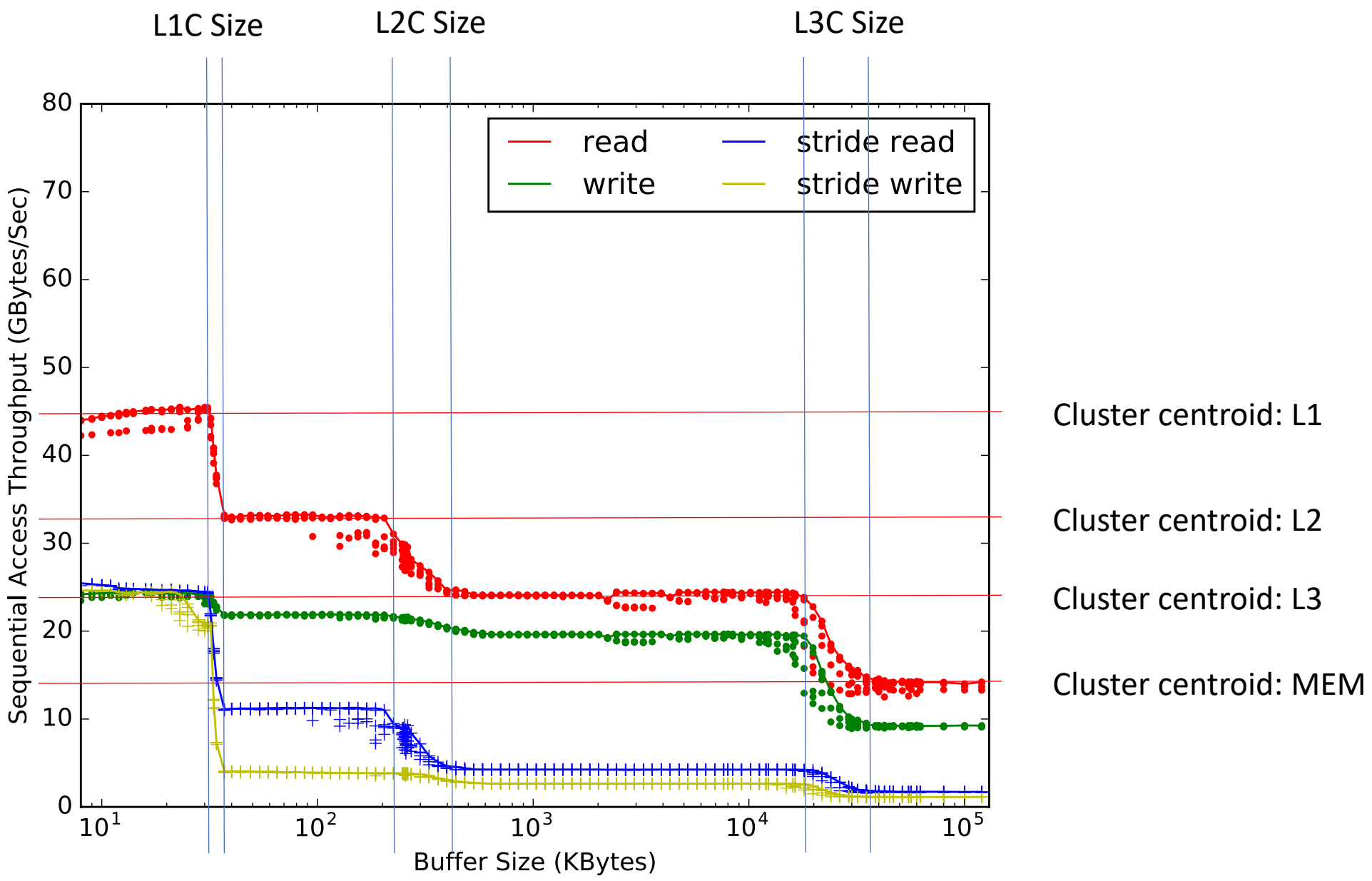
# Measuring Throughput



# Cache Sizes

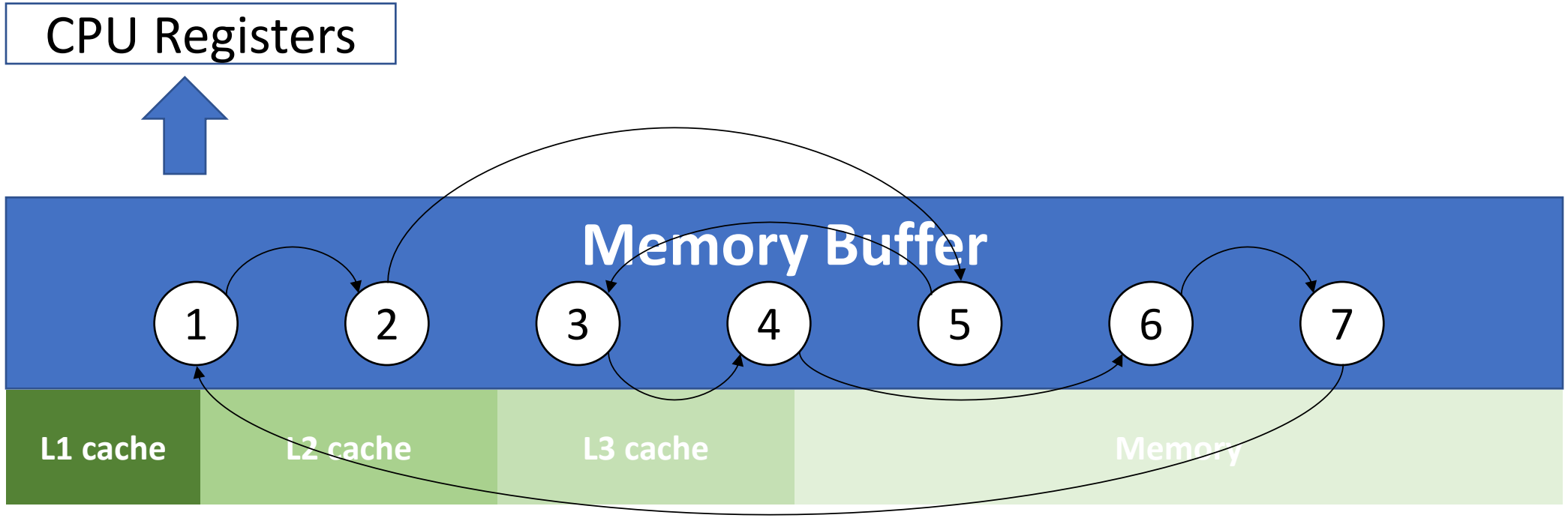
- We give a range instead of a specific value.
- We use the buffer size range in which, the throughput
  - is significantly lower than the upper cache level, and
  - significantly higher than the lower cache level.

  As an estimation of the size of the upper cache level.

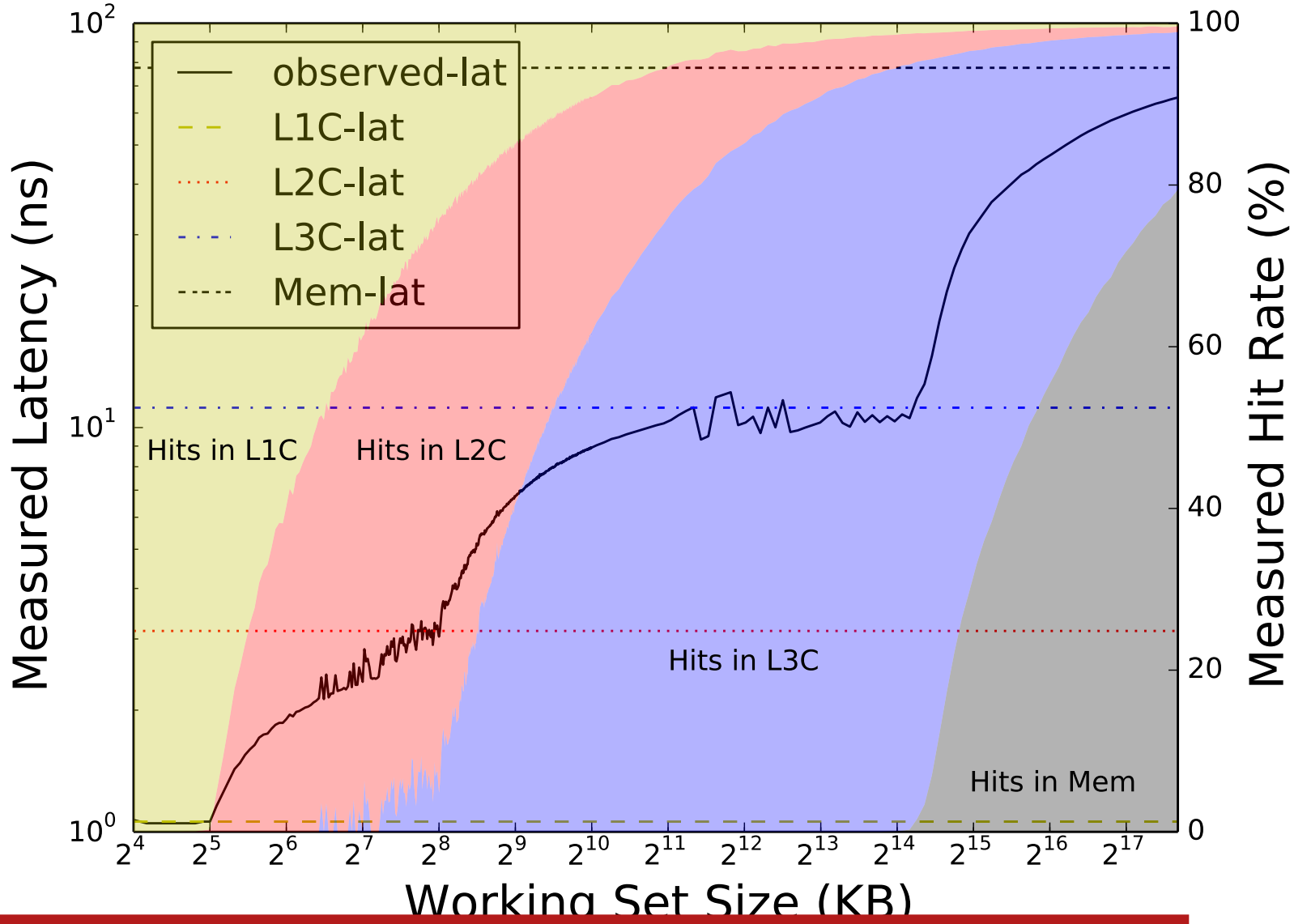

L1C Size
L2C Size
L3C Size
read
write
stride read
stride write
Sequential Access Throughput (GBytes/Sec)
Buffer Size (KBytes)
0
10
20
30
40
50
60
70
80
$10^1$
$10^2$
$10^3$
$10^4$
$10^5$
Cluster centroid: L1
Cluster centroid: L2
Cluster centroid: L3
Cluster centroid: MEM

# Measuring Latency

- We use a randomized cyclic linked list.



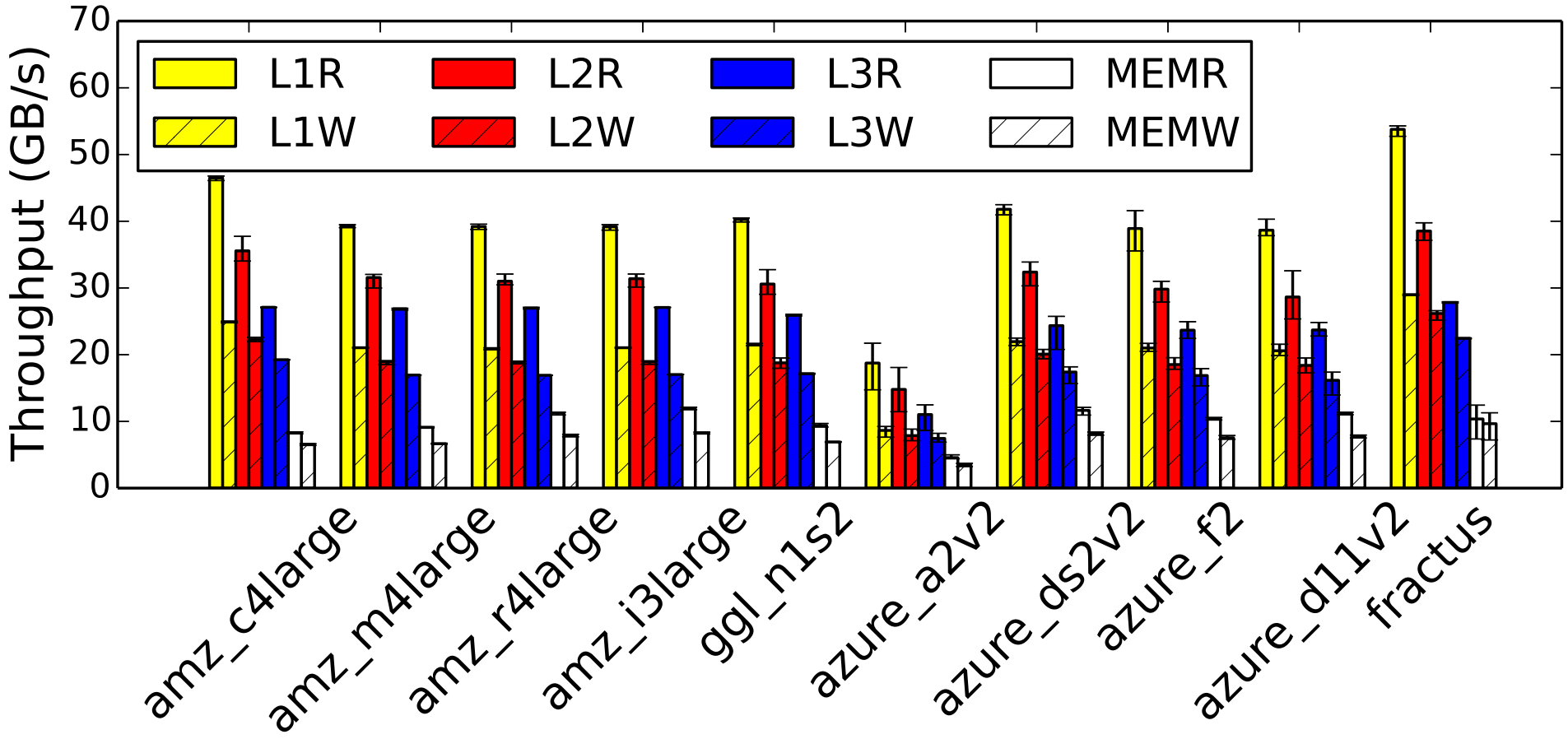
# Measuring Latency



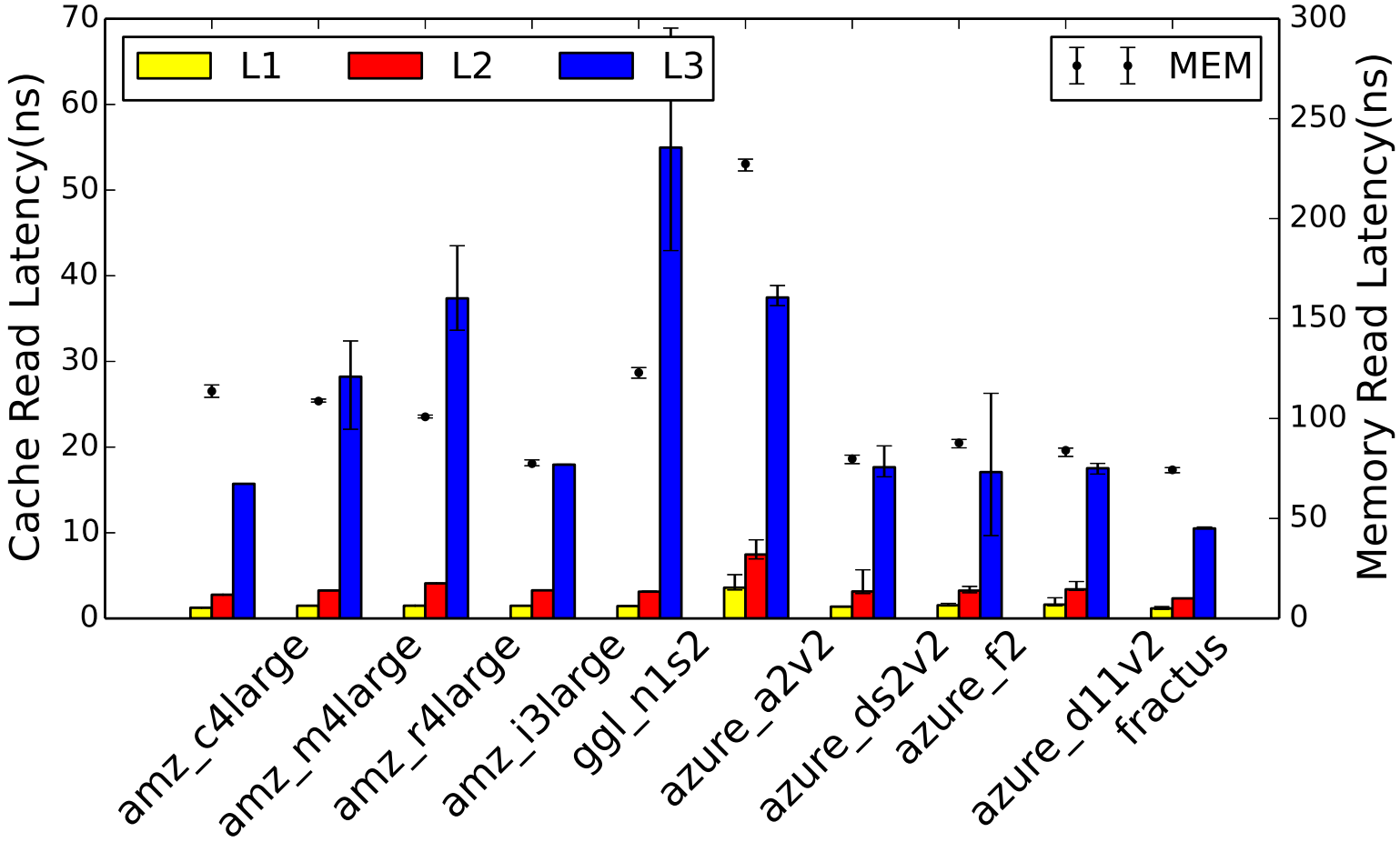
# Measuring Cache and Memory Latencies

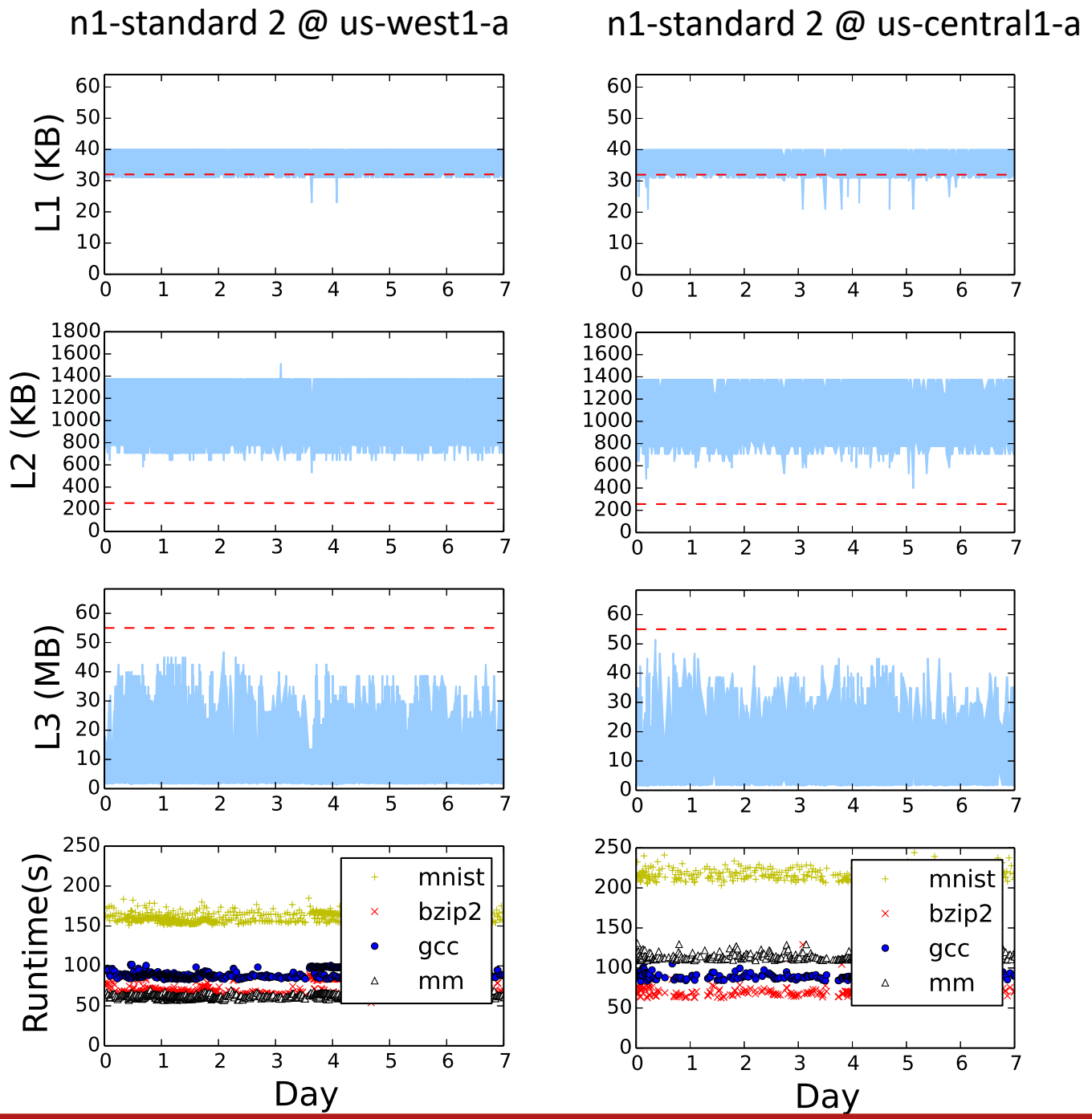
- Observed Latency = L1 latency*L1 hit rate + L2 latency * L2 hit rate + L3 latency * L3 hit rate + Mem latency * Mem hit rate
- Apply multiple linear regression to determine the latency for the cache levels and memory.
- In some clouds, the hit rate may be not available. We use the latency measured with a buffer size smaller but close to corresponding cache size as a quick estimation.

# Throughput Results in Public Cloud



# Latency Results in Public Cloud





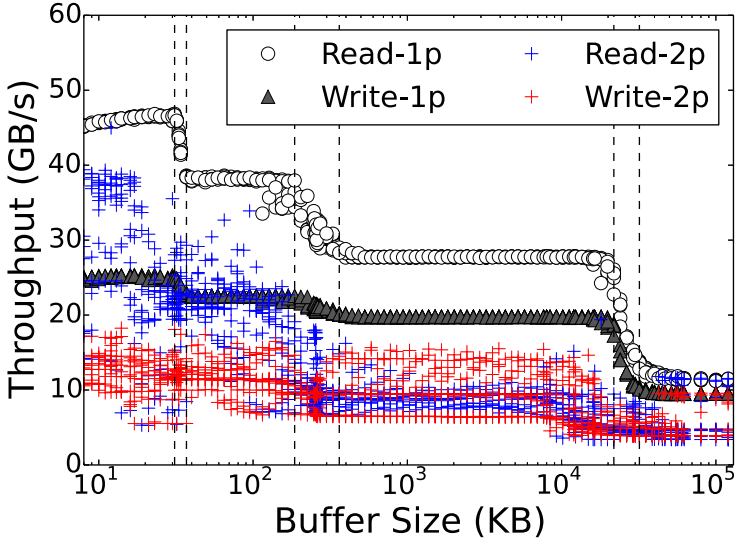

Google Compute Platform:

- Failed to report its cache size
- Inconsistent memory latency

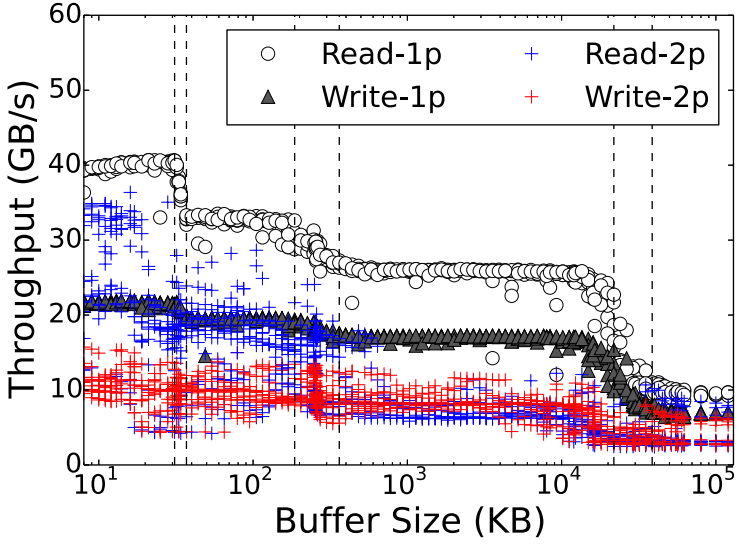
Jetstream

- Failed to report its cache levels

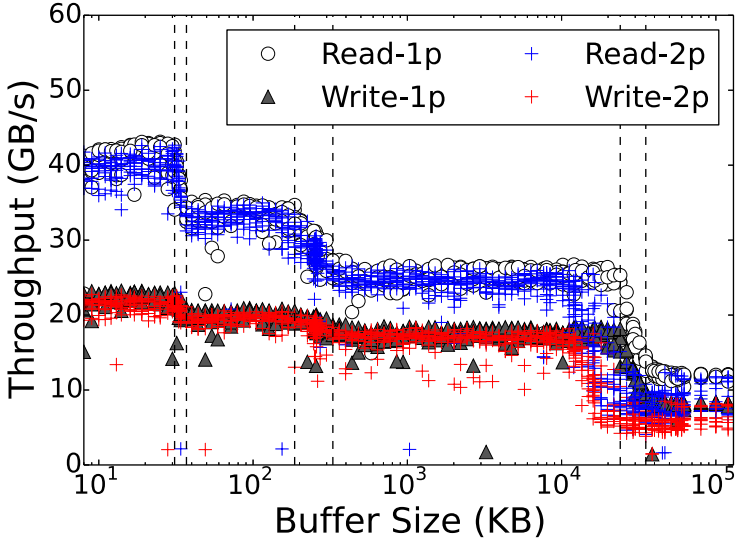
# Hyper-Threading




Amazon c4.large




Google n1-standard-2




Azure Standard DS2 v2

# Conclusion and Future Work

We design and implement a benchmark tool to evaluate the cache and memory resources truly available to a cloud user.

Future work

- Minimize the benchmark overhead with program counters.
- Model application performance with cache size, cache throughput, and latency, memory through and latency.
- Benchmark other resources including network and storage.