CS 795: Distributed Systems & Cloud Computing
Fall 2018

Lec 10: Datacenter management
Yue Cheng

Agenda

- Today's lecture: Datacenter workload management & analysis: An Alibaba case study

- RAMCloud paper discussion

- Course evaluation forms

- Announcement: I will be holding extra office hours 3-5pm Friday (and open for phone call appointment next week)

Characterizing Co-located Datacenter Workloads:

An Alibaba Case Study

The sorry state of server utilization and the impending post-hypervisor era

Alex Benik, Battery Ventures Nov 30, 2013 - 10:30 AM CDT

> *Me: Do you track server and CPU utilization?*
>
> ***Wall Street IT Guru**: Yes*
>
> *Me: So it's a metric you report on with other infrastructure KPIs?*
>
> ***Wall Street IT Guru**: No way, we don't put it in reports. If people knew how low it really is, we'd all get fired.*

- A McKinsey study in 2008 pegging data-center utilization at roughly 6 percent.
- A Gartner report from 2012 putting industry wide utilization rate at 12 percent.
- An Accenture paper sampling a small number on Amazon EC2 machines finding 7percent utilization over the course of a week.
- The charts and quote below from Google, which show three-month average utilization rates for 20,000 server clusters. The typical cluster on the left spent most of its time running between 20-40 percent of capacity, and the highest utilization cluster on the right reaches such heights only because it's doing batch work.

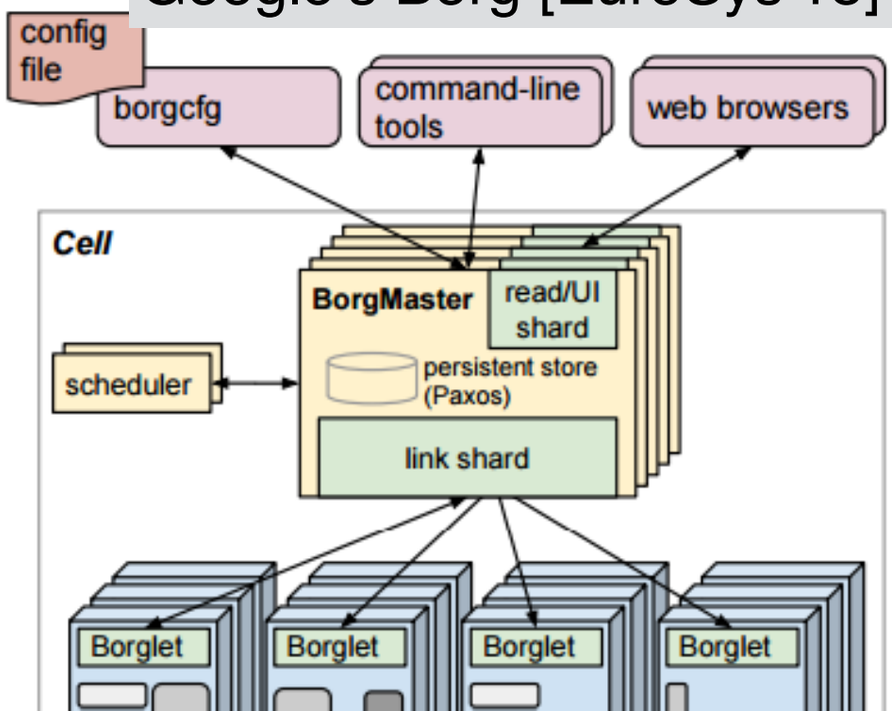
Workload co-location

- Run all workloads on one cluster
  - Latency-sensitive, long-running, **interactive** workloads
  - Transient **batch** job workloads
- Improved utilization and elasticity
  - Fill batch jobs into "**gaps**" in interactive workloads
  - Evict batch jobs if interactive workload demand **spikes**

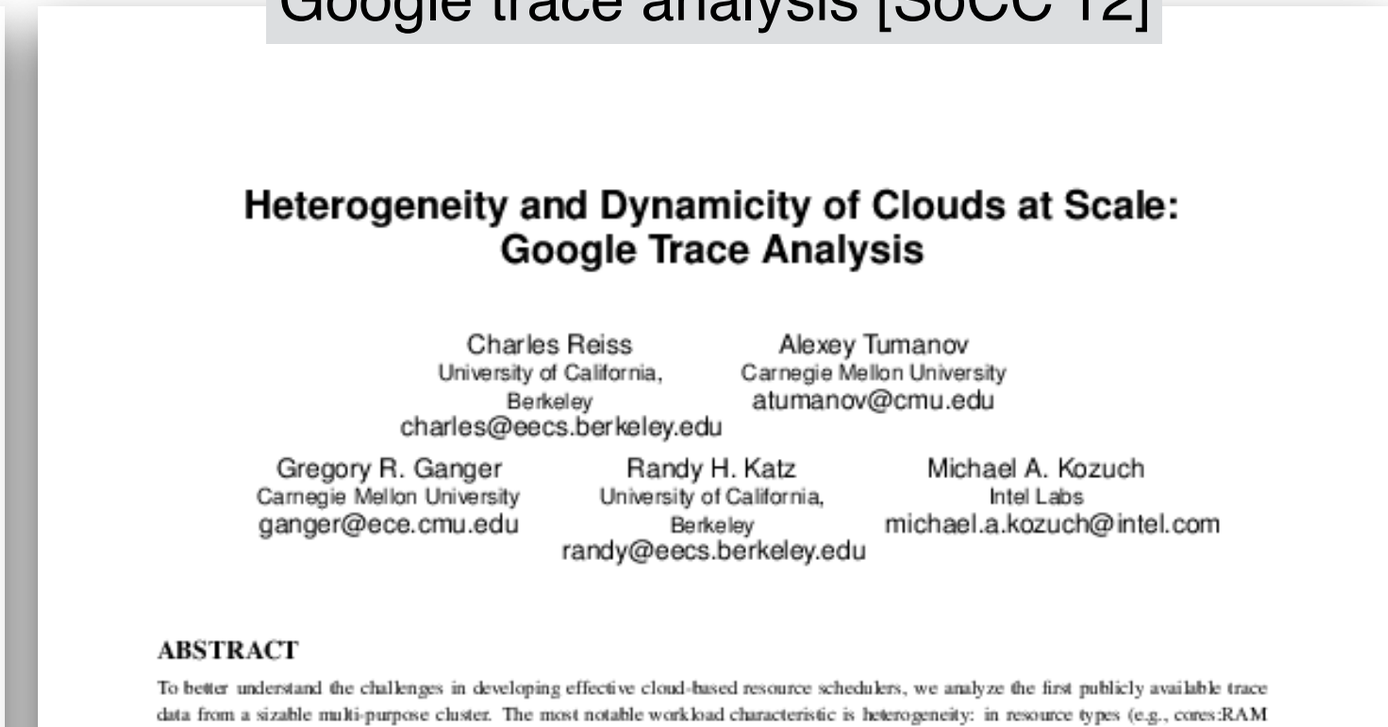
Workload co-location

- Run all workloads on one cluster
  - Latency-sensitive, long-running, **interactive** workloads
  - Transient **batch** job workloads
- Improved utilization and elasticity
  - Fill batch jobs into "**gaps**" in interactive workloads
  - Evict batch jobs if interactive workload demand **spikes**

Google's Borg [EuroSys'15]



Google trace analysis [SoCC'12]



Heterogeneity and Dynamicity of Clouds at Scale: Google Trace Analysis

Charles Reiss
University of California, Berkeley
charles@eecs.berkeley.edu

Alexey Tumanov
Carnegie Mellon University
atumanov@cmu.edu

Gregory R. Ganger
Carnegie Mellon University
ganger@ece.cmu.edu

Randy H. Katz
University of California, Berkeley
randy@eecs.berkeley.edu

Michael A. Kozuch
Intel Labs
michael.a.kozuch@intel.com

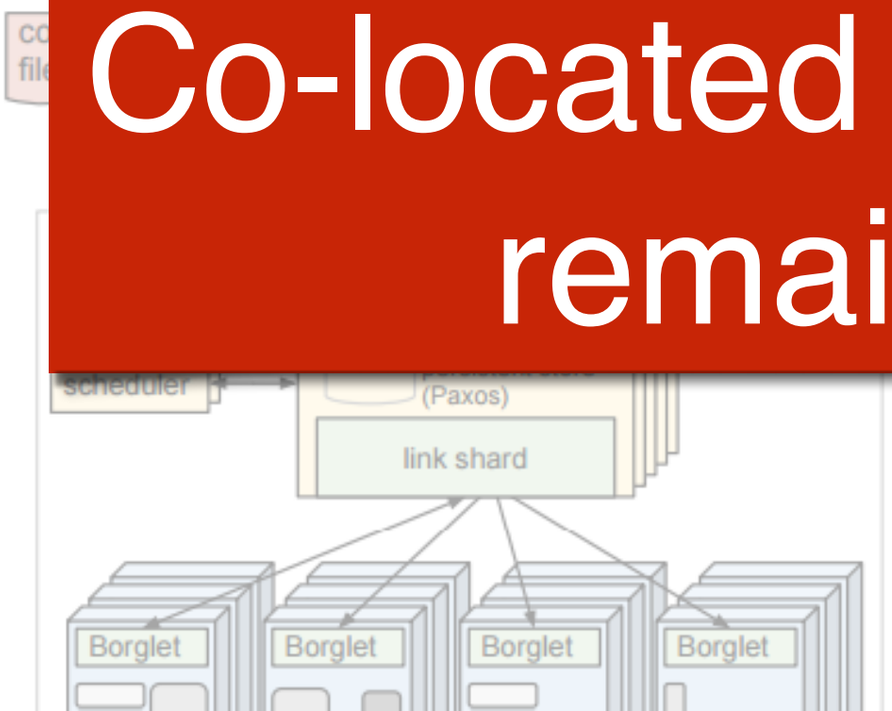
ABSTRACT

To better understand the challenges in developing effective cloud-based resource schedulers, we analyze the first publicly available trace data from a sizable multi-purpose cluster. The most notable workload characteristic is heterogeneity: in resource types (e.g., cores:RAM

Workload co-location

- Run all workloads on one cluster
  - Latency-sensitive, long-running, **interactive** workloads
  - Transient **batch** job workloads
- Improved utilization and elasticity
  - Fill batch jobs into "**gaps**" in interactive workloads
  - Evict batch jobs if interactive workload demand **spikes**

Co-located workload patterns remain a mystery!



Berkeley
charles@eecs.berkeley.edu
atumanov@cmu.edu

Gregory R. Ganger
Carnegie Mellon University
ganger@ece.cmu.edu

Randy H. Katz
University of California, Berkeley
randy@eecs.berkeley.edu

Michael A. Kozuch
Intel Labs
michael.a.kozuch@intel.com

ABSTRACT

To better understand the challenges in developing effective cloud-based resource schedulers, we analyze the first publicly available trace data from a sizable multi-purpose cluster. The most notable workload characteristic is heterogeneity: in resource types (e.g., cores:RAM

The Alibaba trace

11/11 Singles' day shopping festival

11.11 TURNS 10: ALIBABA KICKS OFF 2018 GLOBAL SHOPPING FESTIVAL

ADAM NAJBERG | OCTOBER 19, 2018

Singles' Day: China's $25 billion shopping festival explained

Big discounts for branded clothing, Starbucks coffee, cars, toilet rolls and more

11/11 Singles' day shopping festival

11/11 Singles' day shopping festival

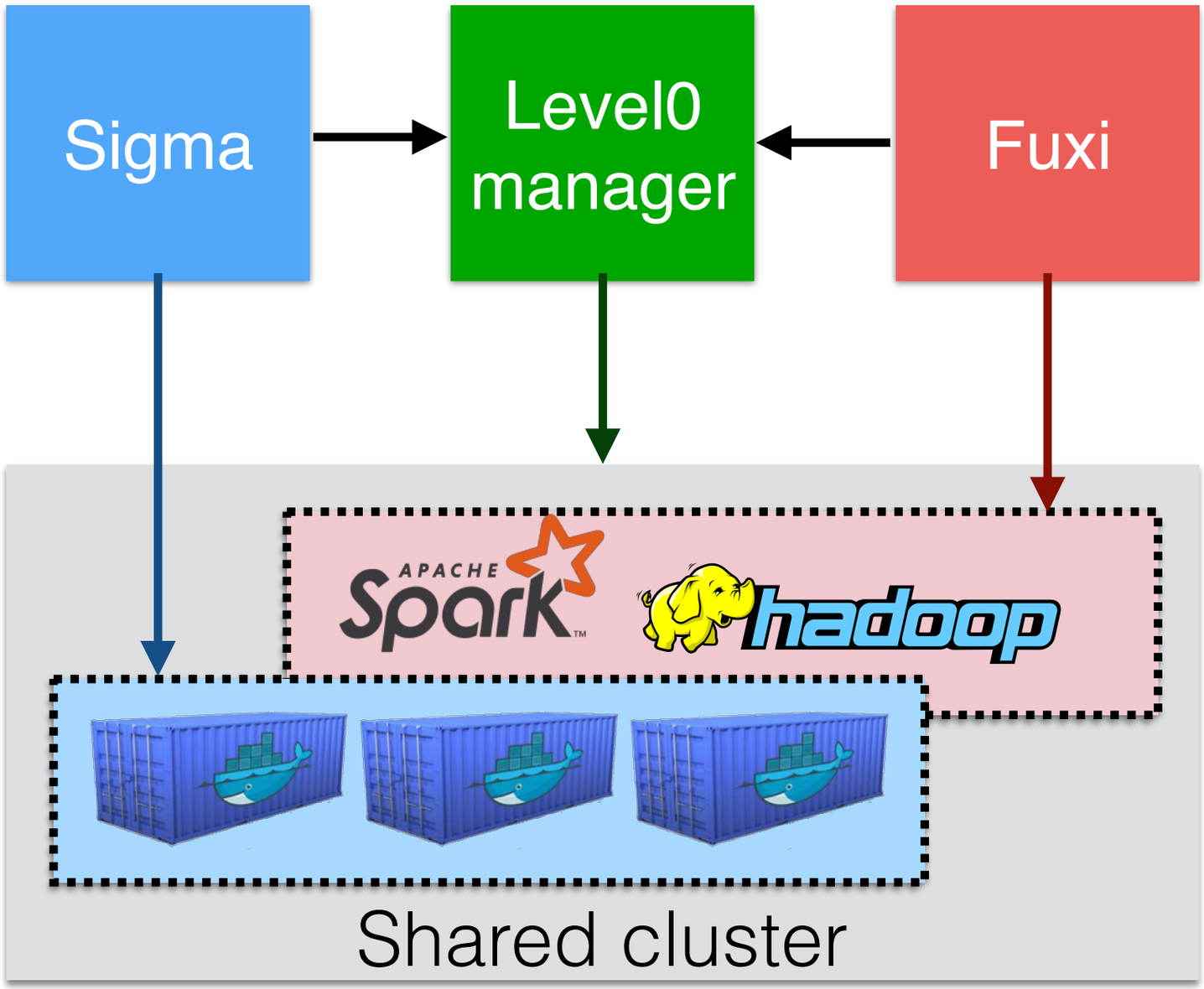
The Alibaba trace

Released Aug 2017

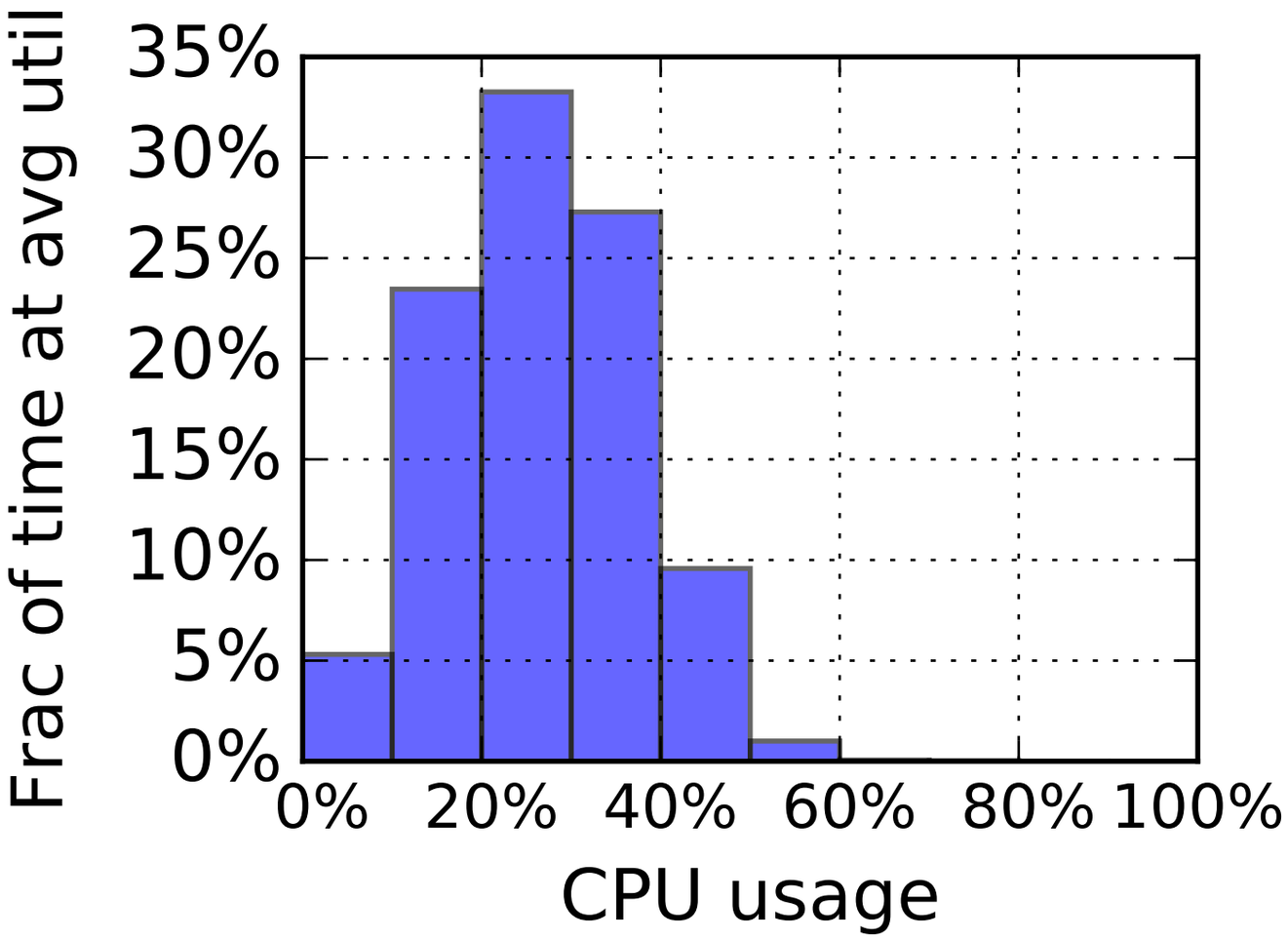
> **“The data is provided to address the challenges Alibaba face in IDCS where online services and batch jobs are co-allocated …**

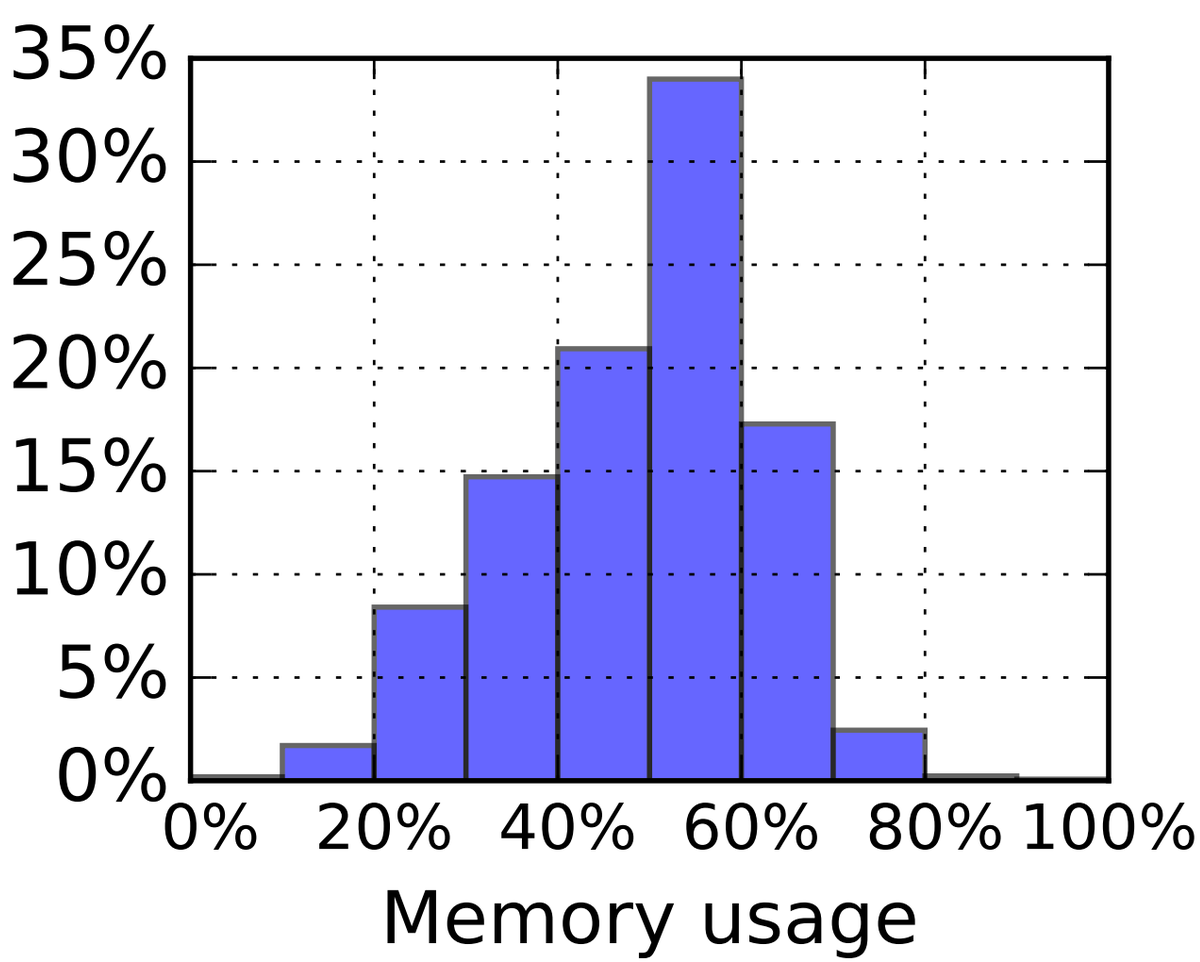
- Two general types of workloads sharing a production cluster of **1.3k machines** for **24 hours**
  - **Containerized** interactive services (e.g., Email)
  - **Batch** jobs (**DAG** of tasks, e.g., MapReduce)
  - Ran on separate clusters before **2015**

Alibaba's cluster management systems



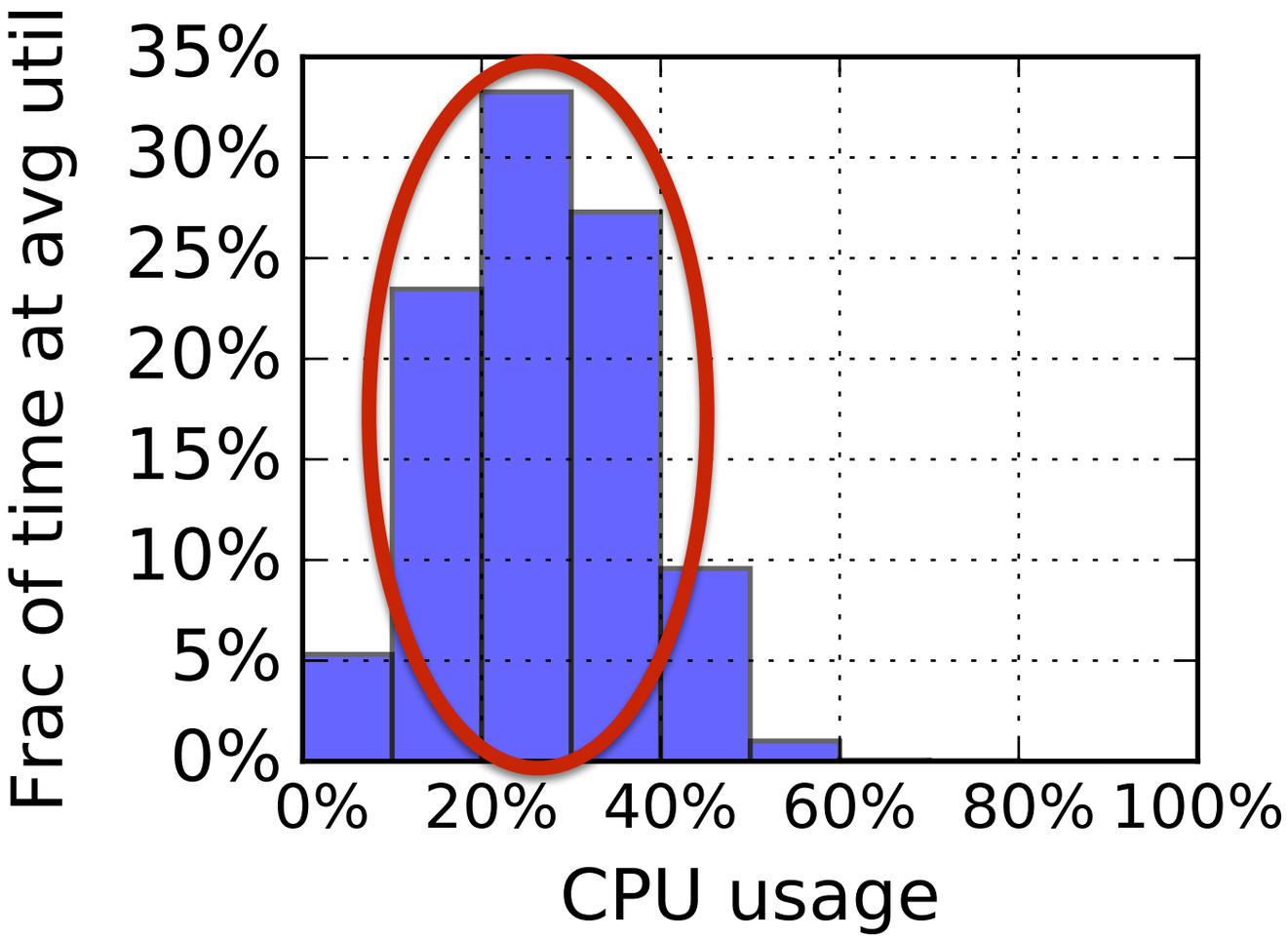
Average machine utilization



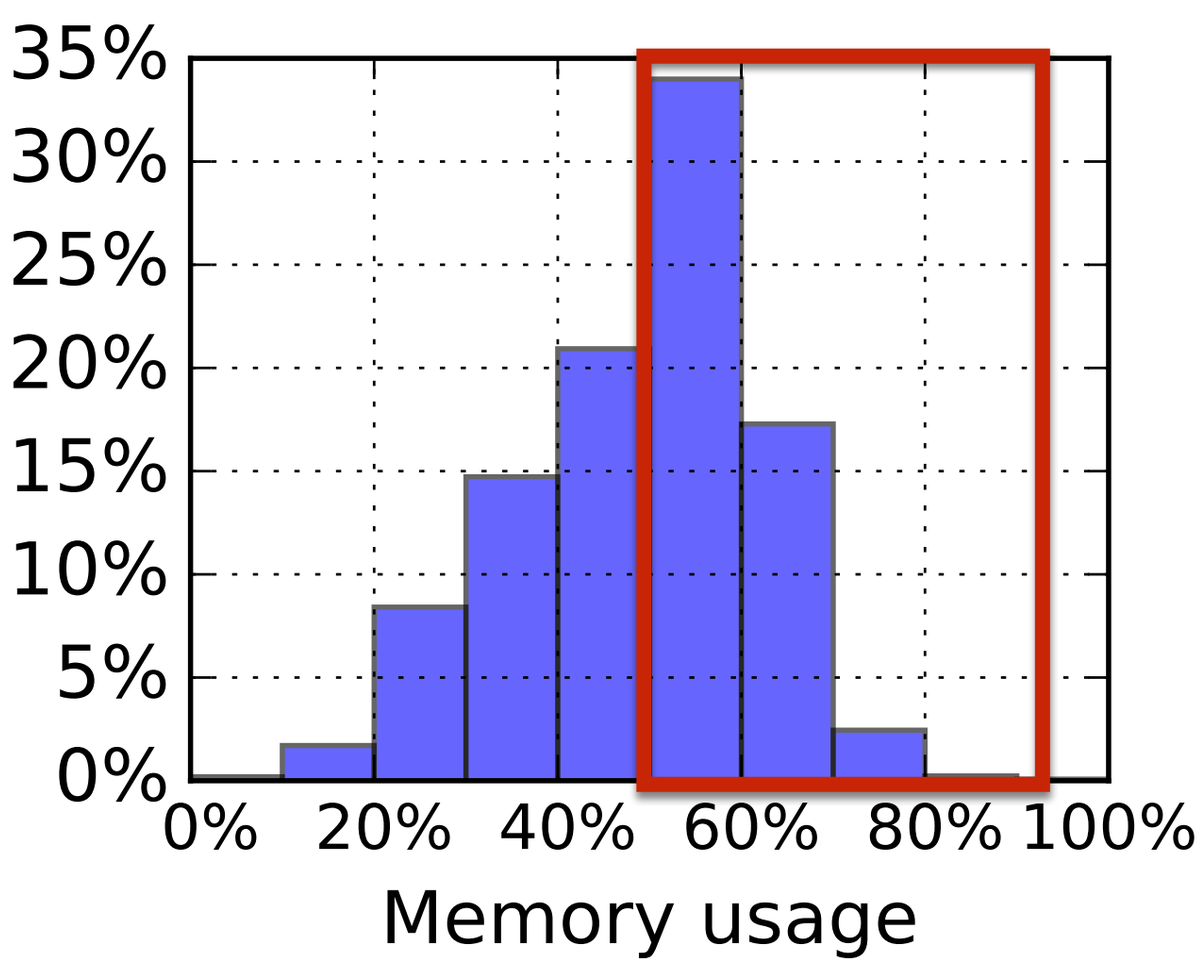


Average machine utilization

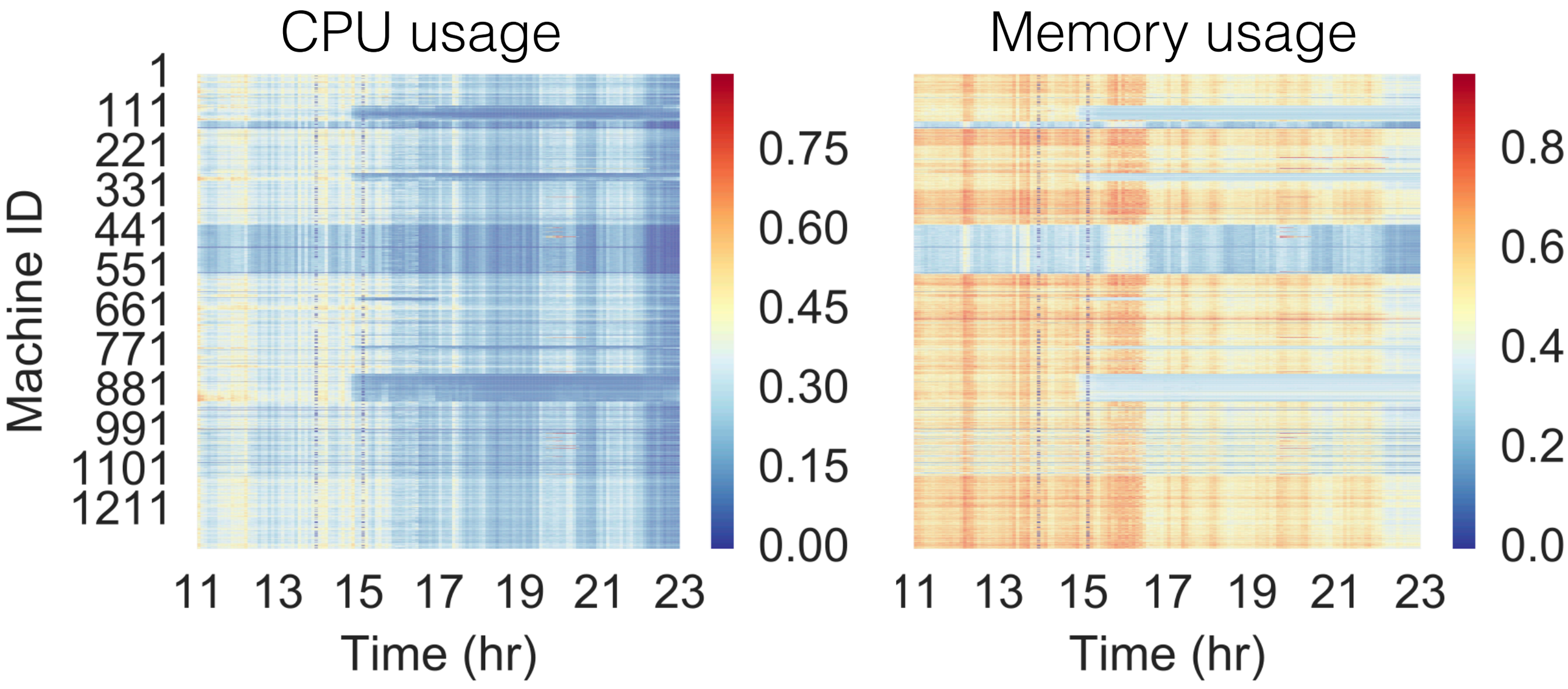
> 80% time running b/w 10-30% CPU usage



> 50% memory usage for over 55% time



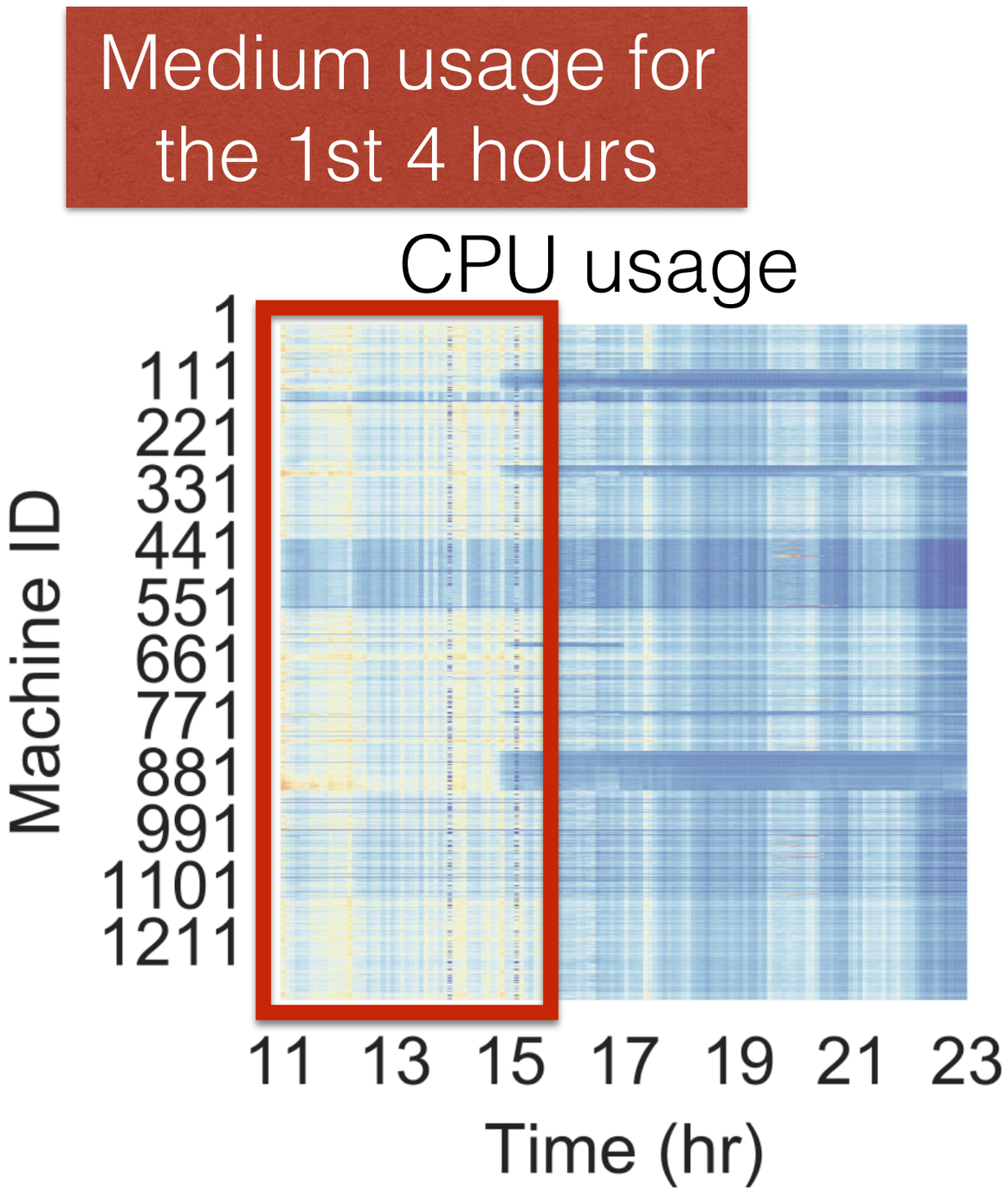
Overall cluster usage heatmap

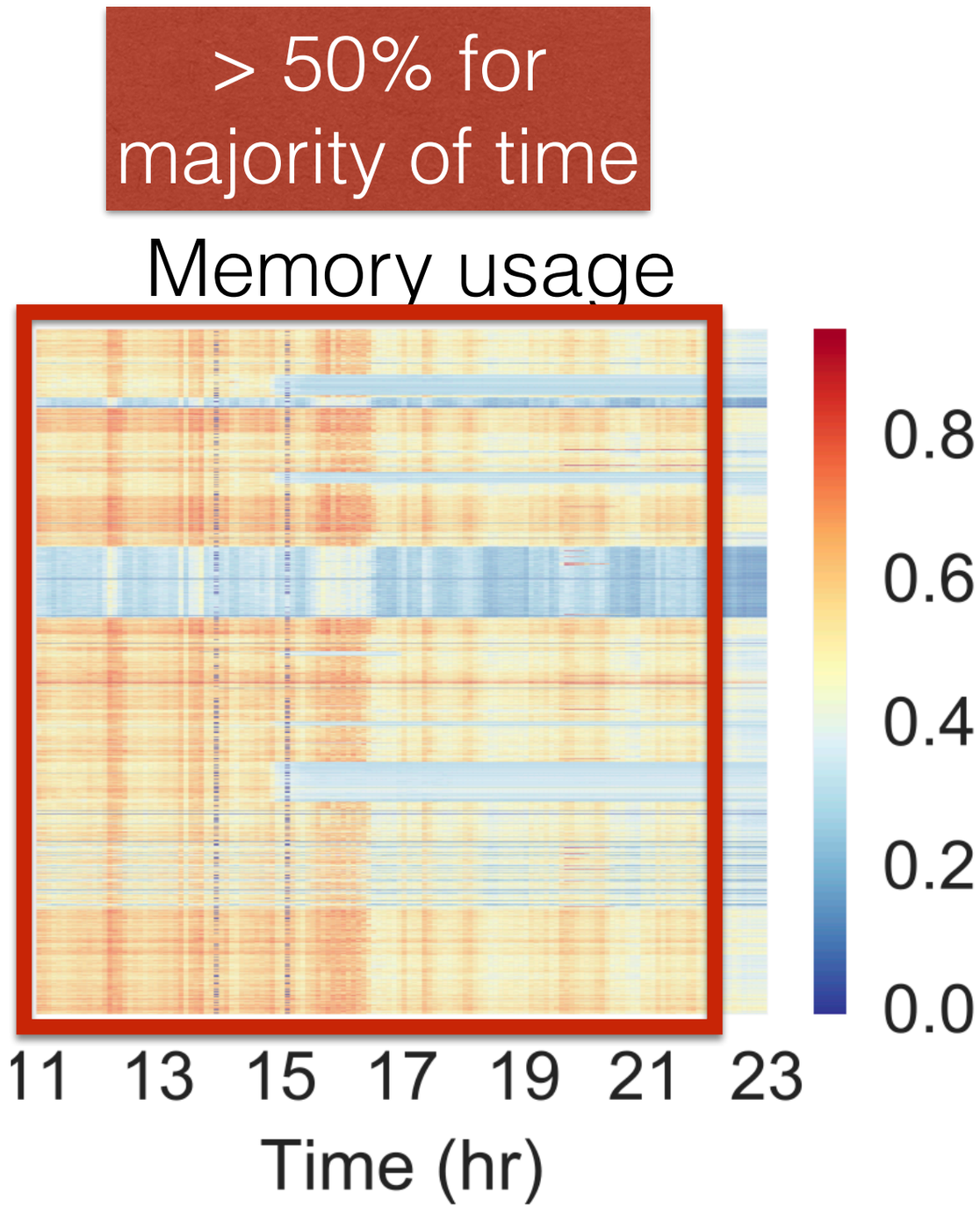


Overall cluster usage heatmap

Medium usage for the 1st 4 hours

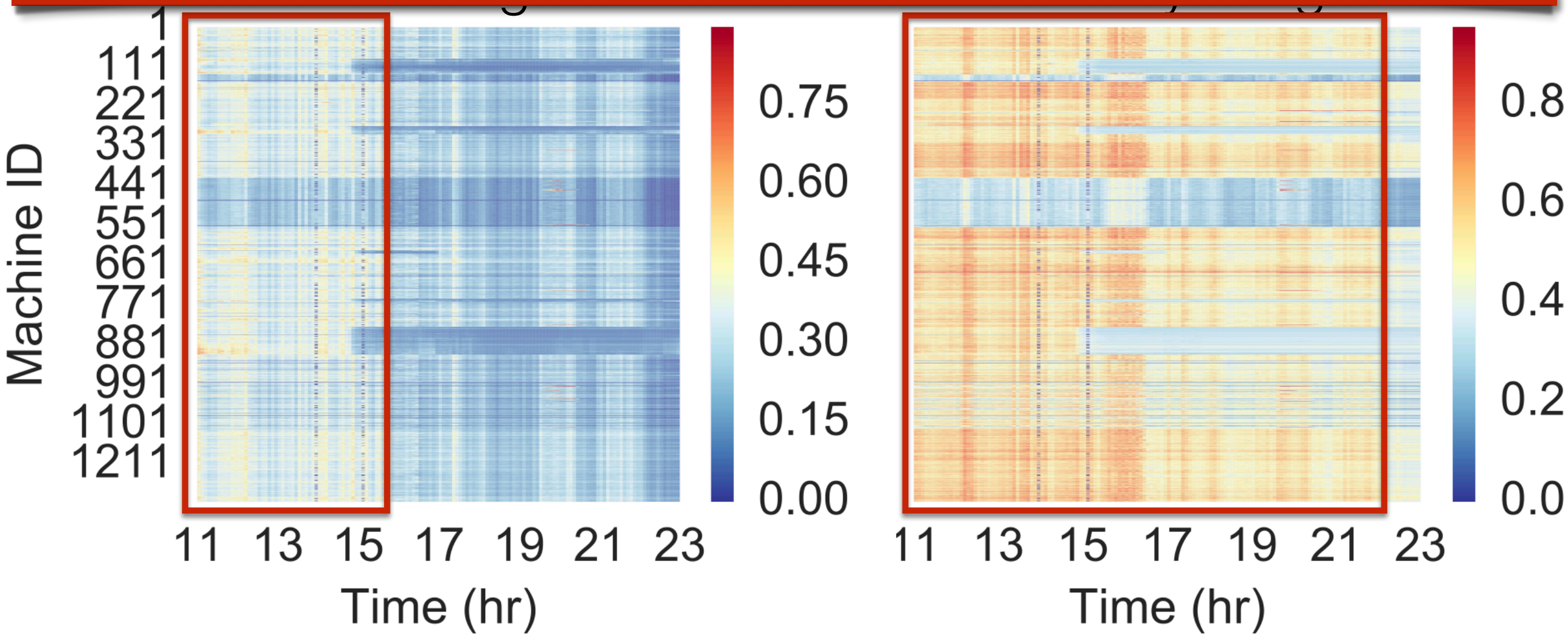
> 50% for majority of time



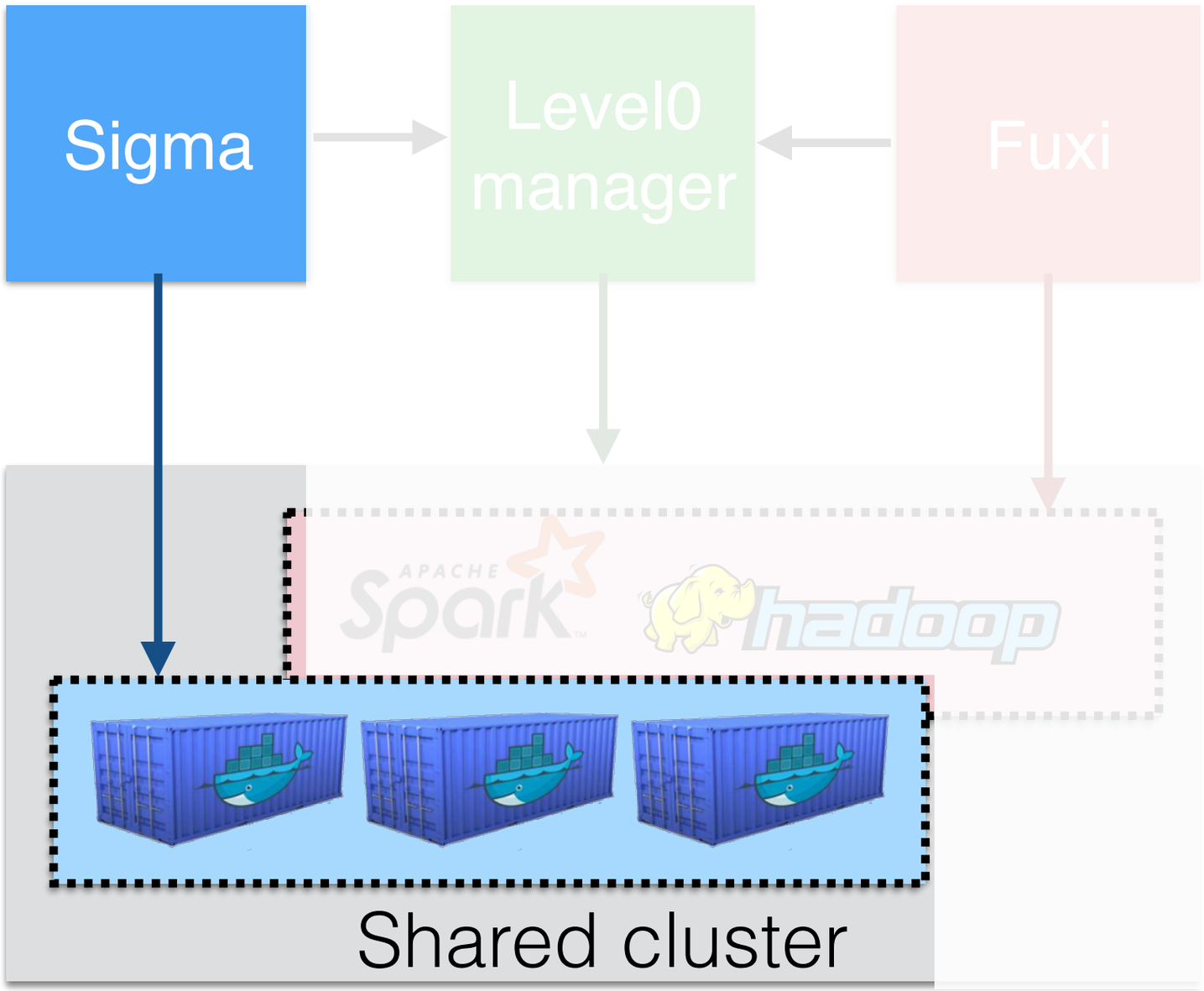


Overall cluster usage heatmap

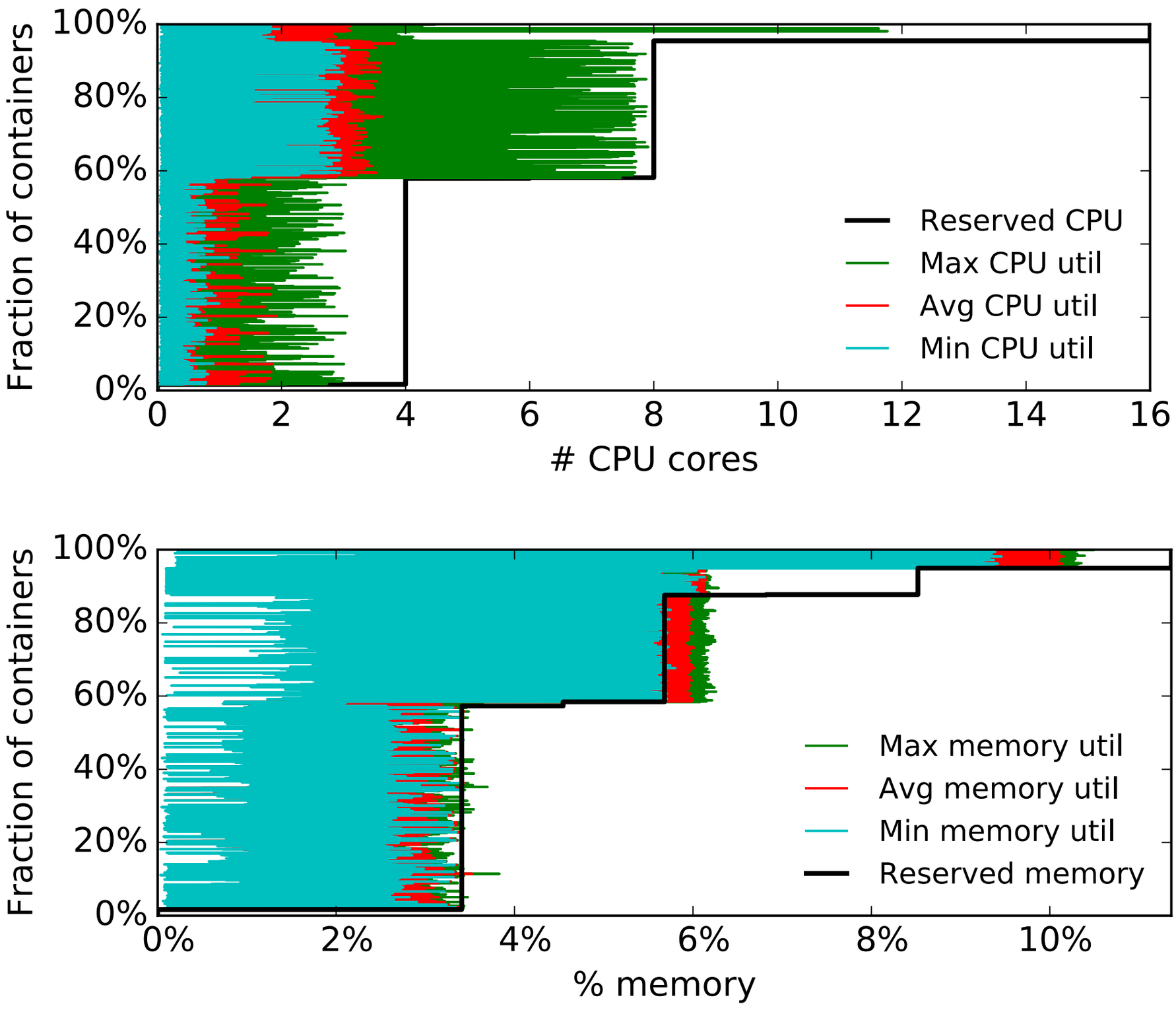
Memory tends to be of higher demands with over half capacity consumed over half the time



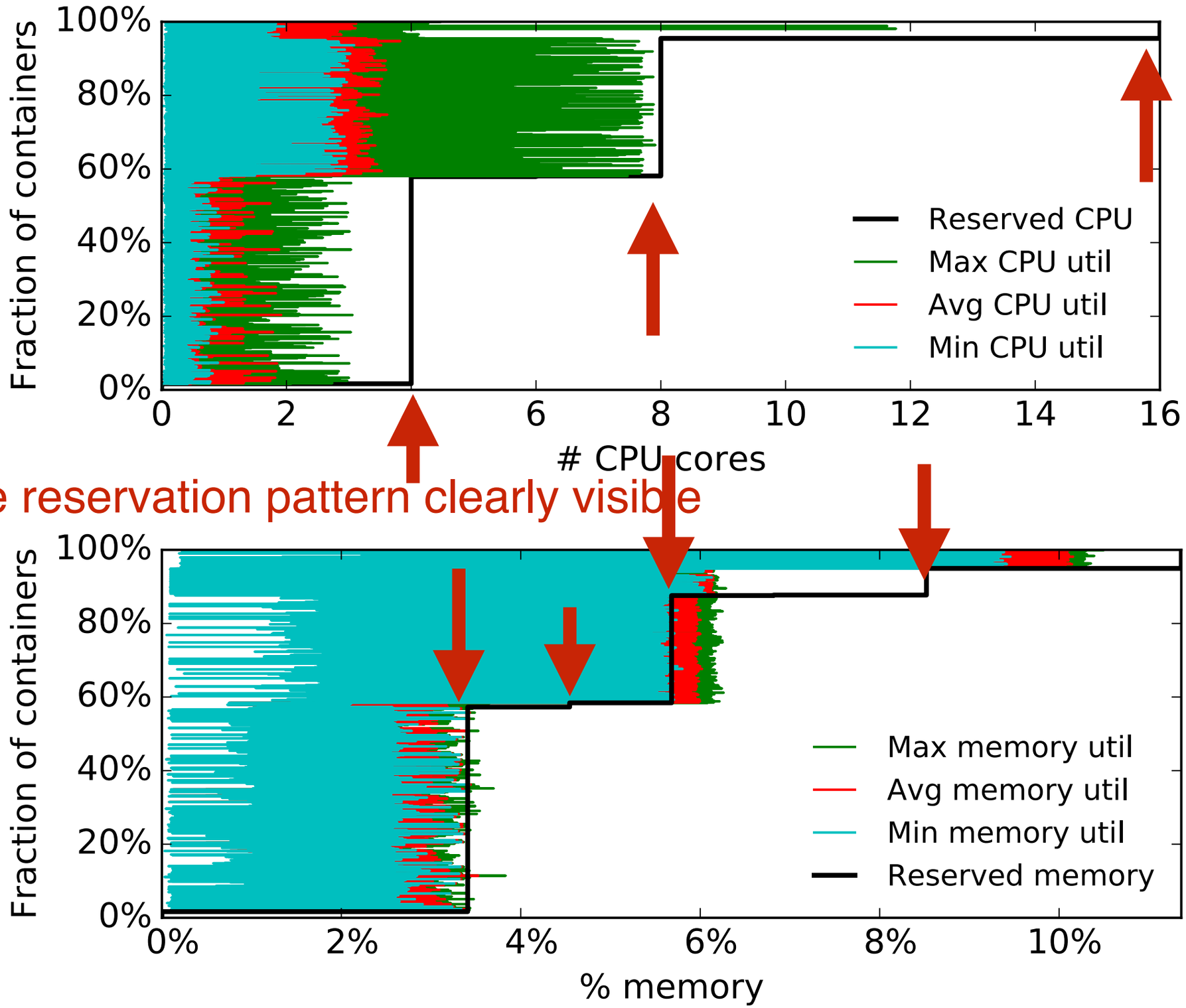
Long-running, containerized workloads



Long-running, containerized workload: Reserved resources vs. actual usage

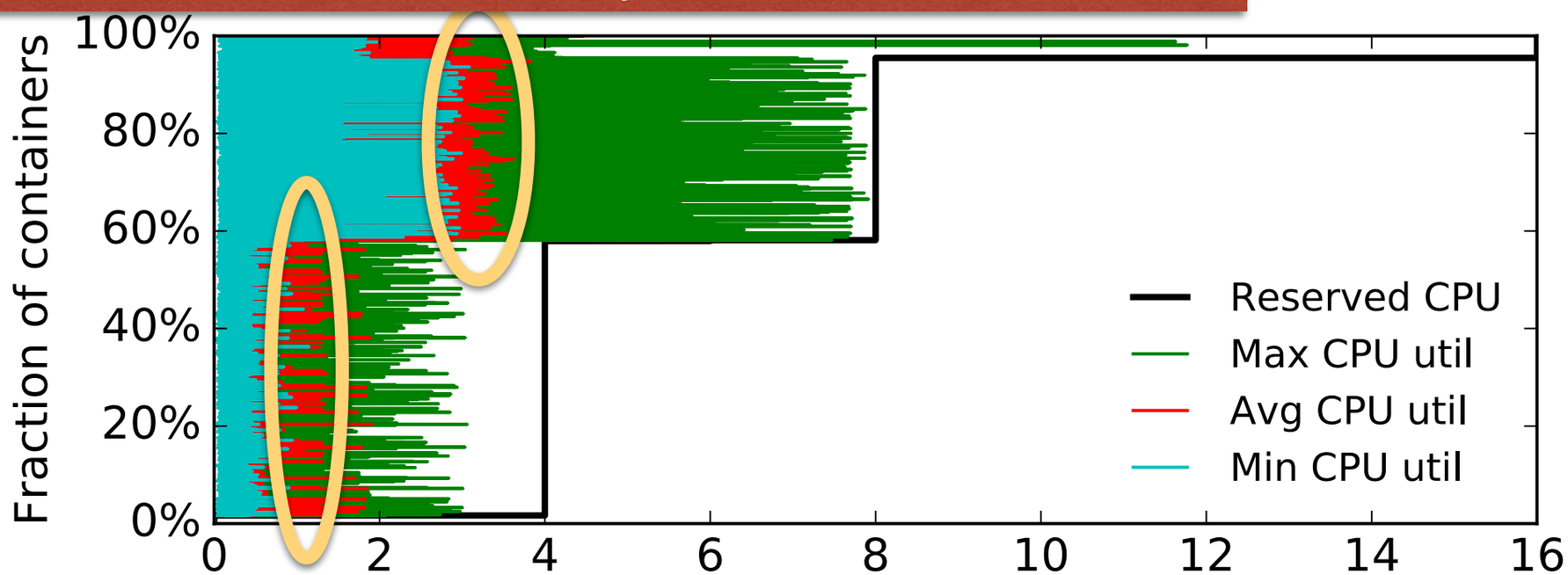


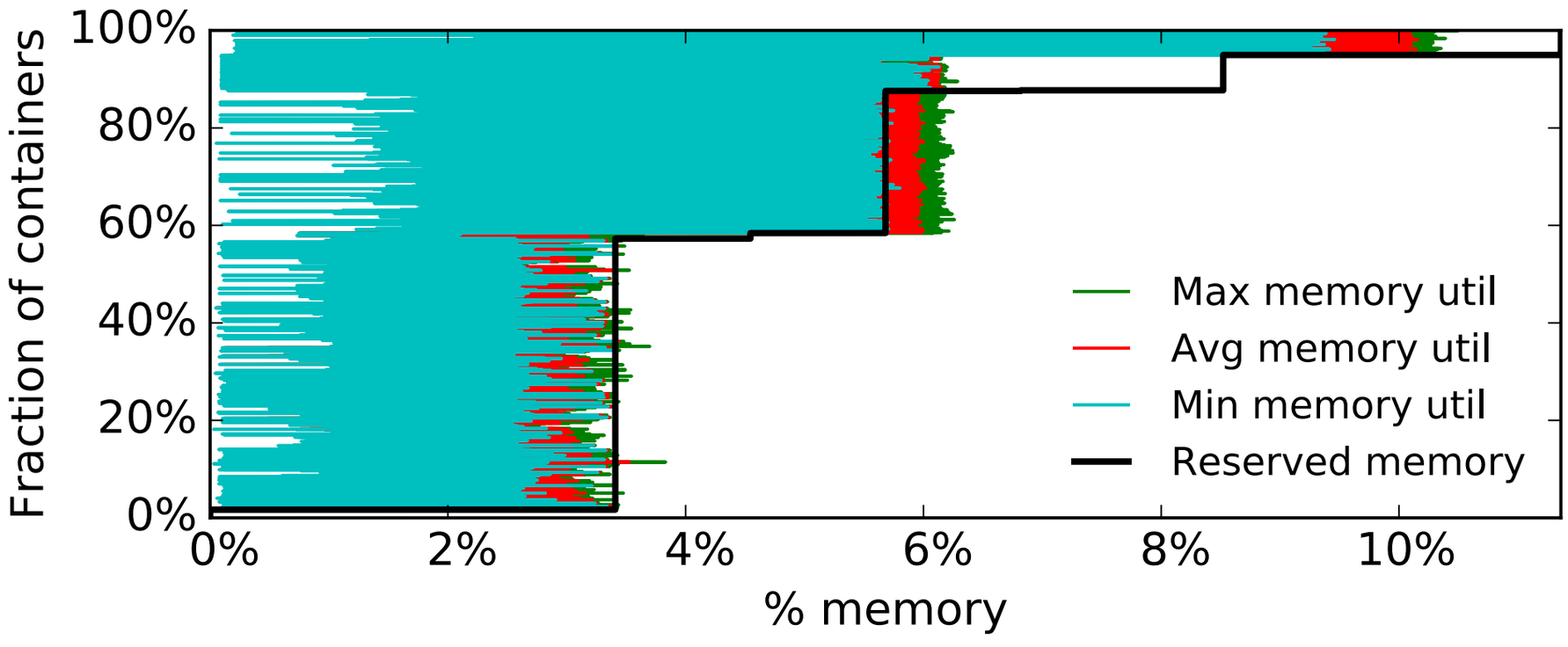
Long-running, containerized workload: Reserved resources vs. actual usage



Long-running, containerized workload: Reserved resources vs. actual usage

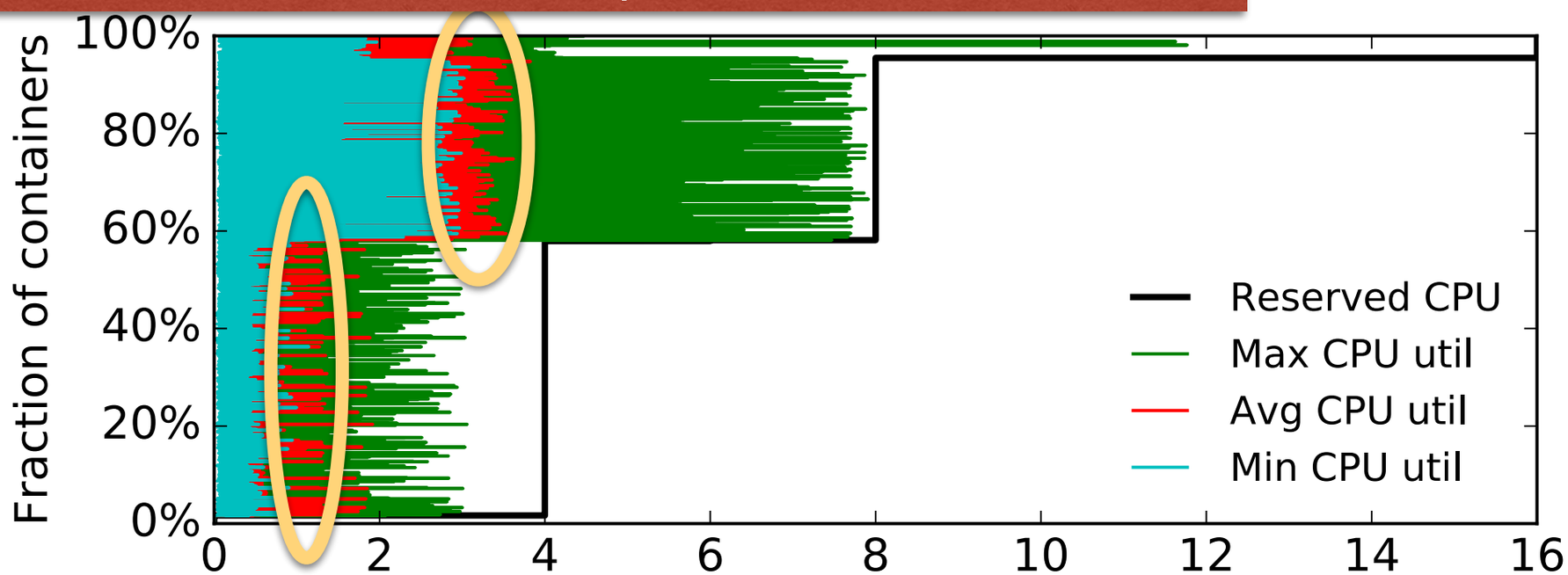
All containers have over-provisioned CPUs



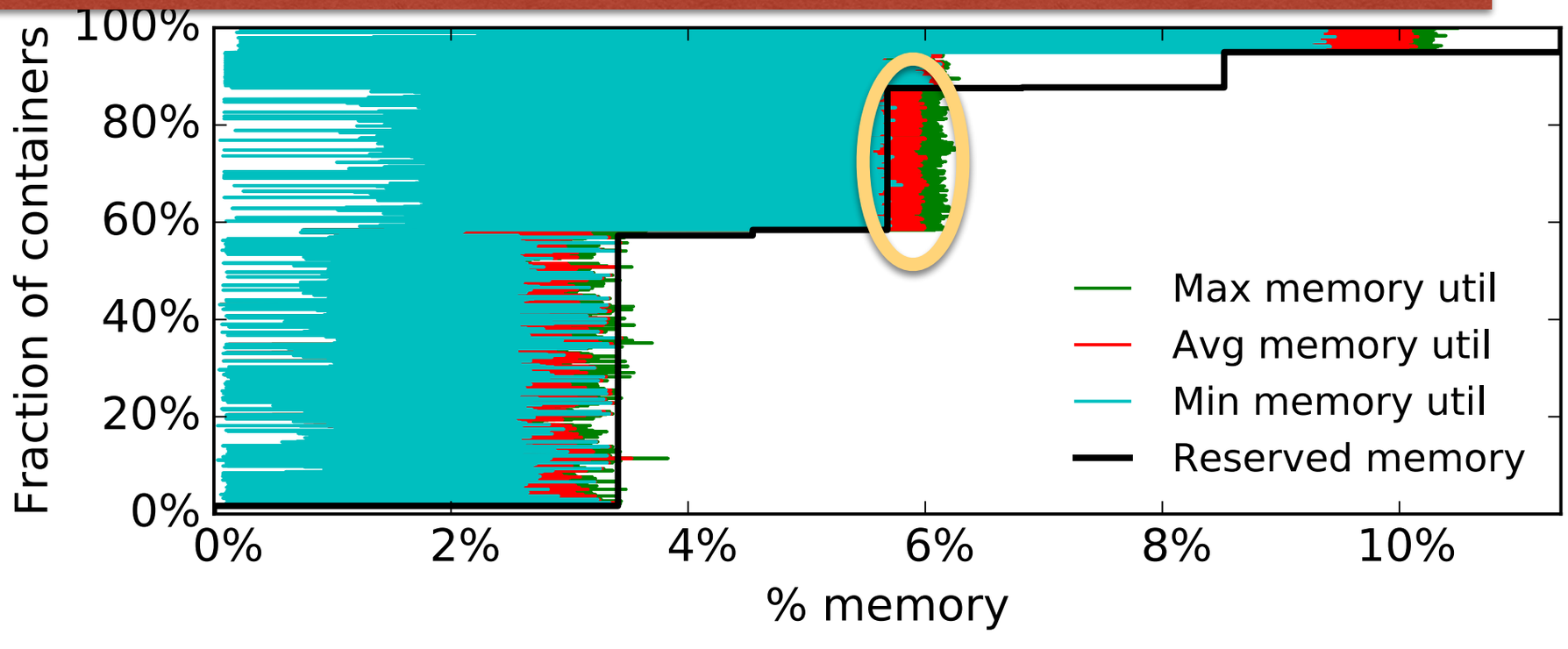


Long-running, containerized workload: Reserved resources vs. actual usage

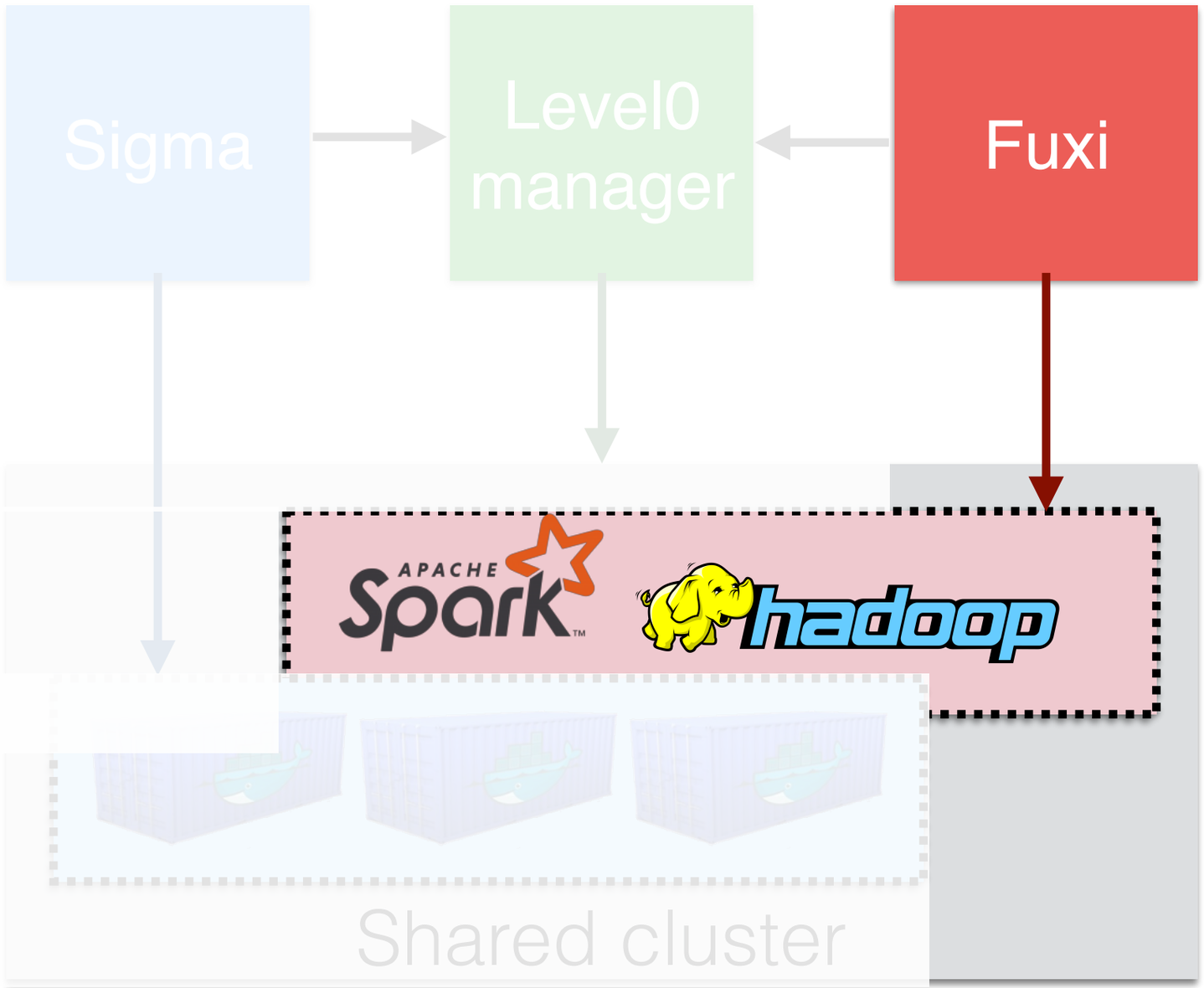
All containers have over-provisioned CPUs



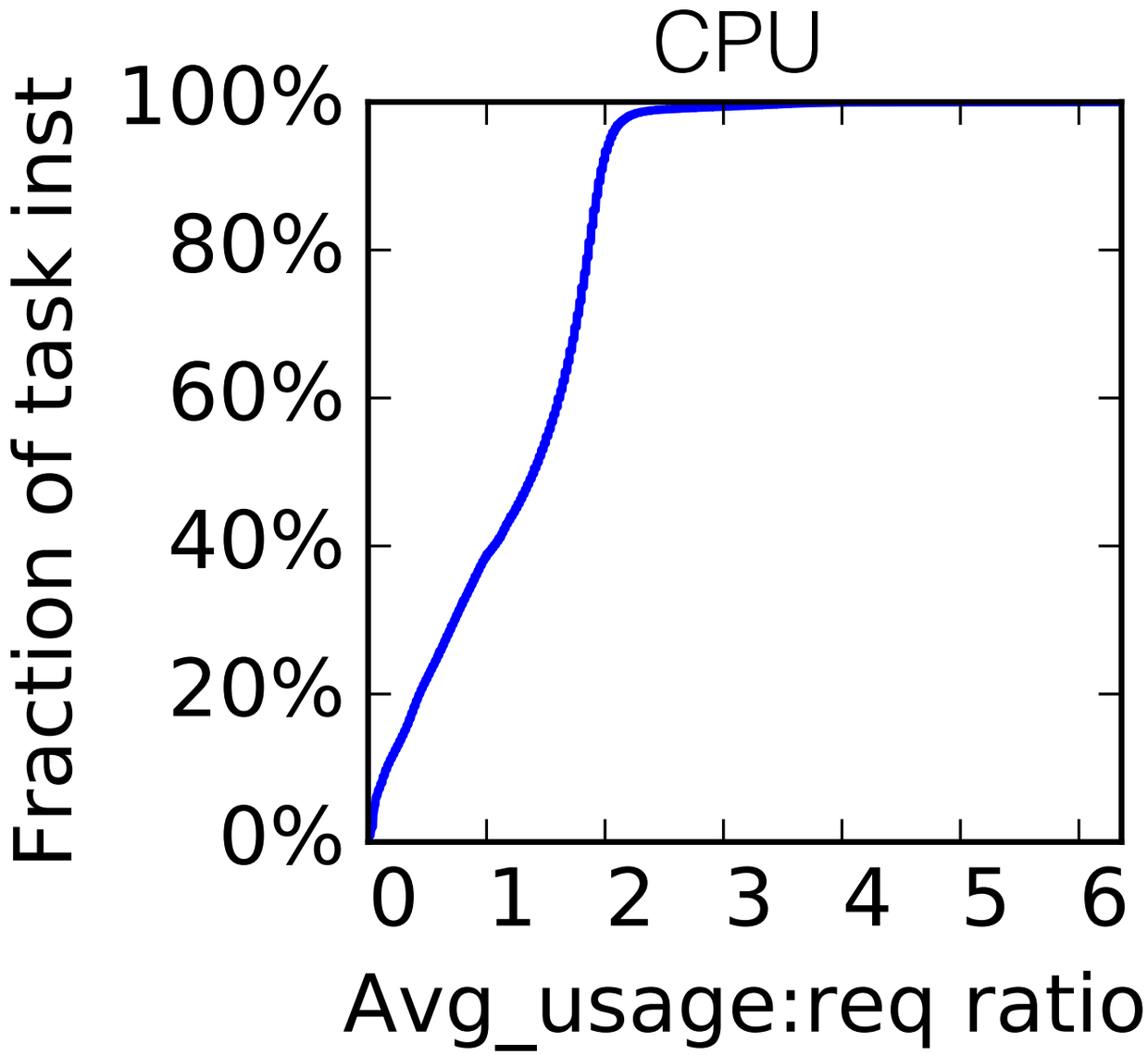
A small fraction of containers overcommit memory

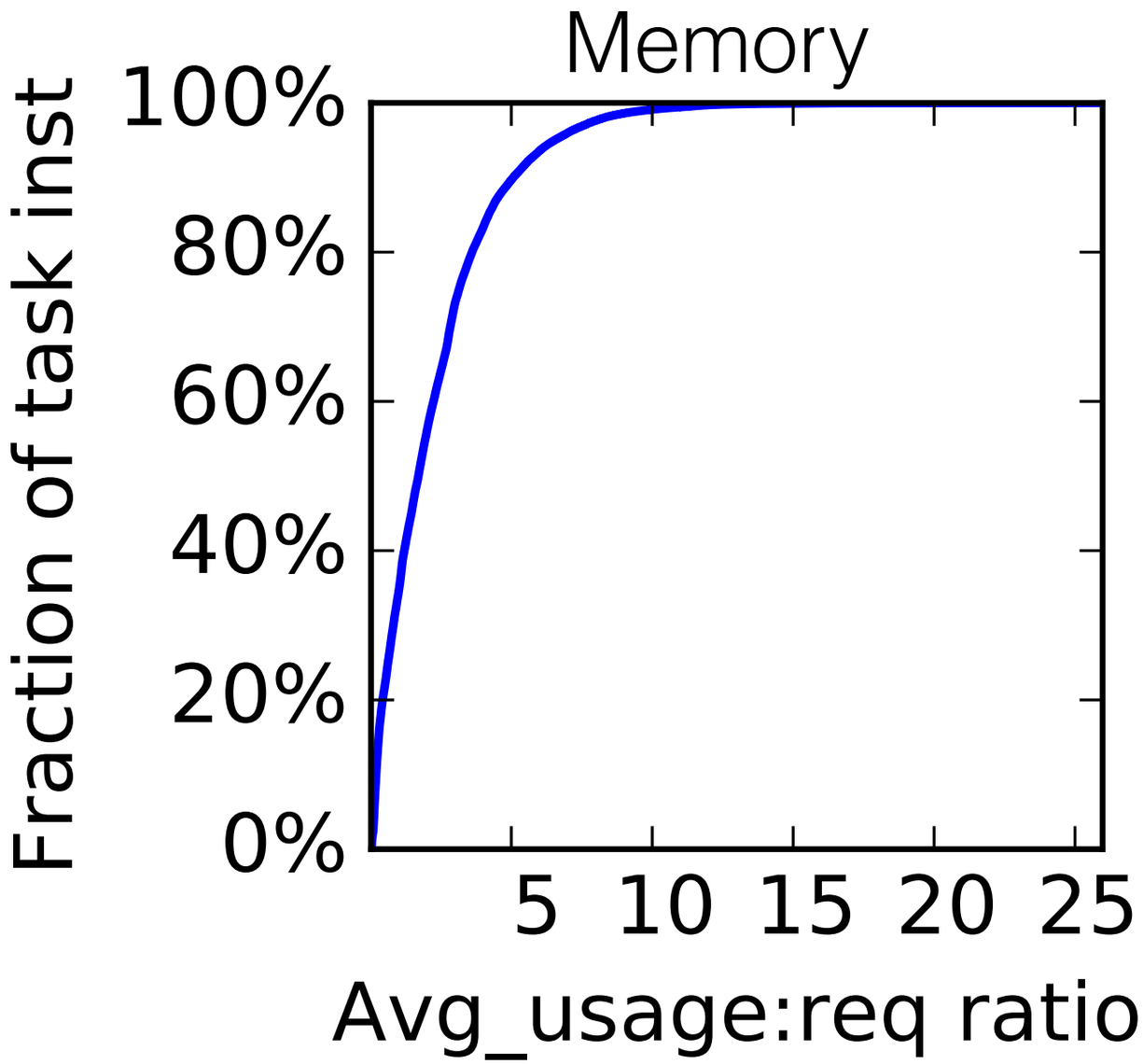


Transient, batch processing workloads

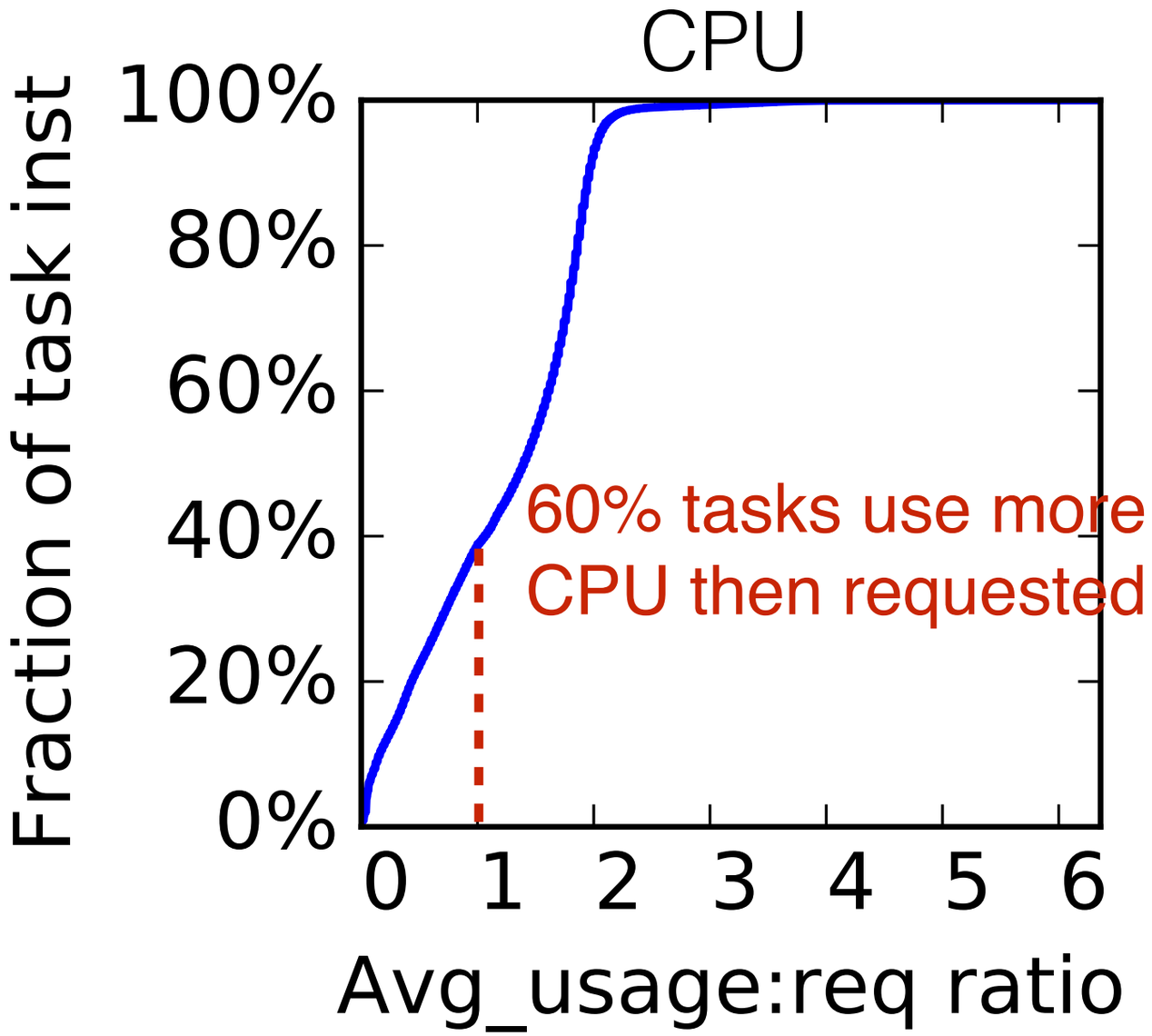


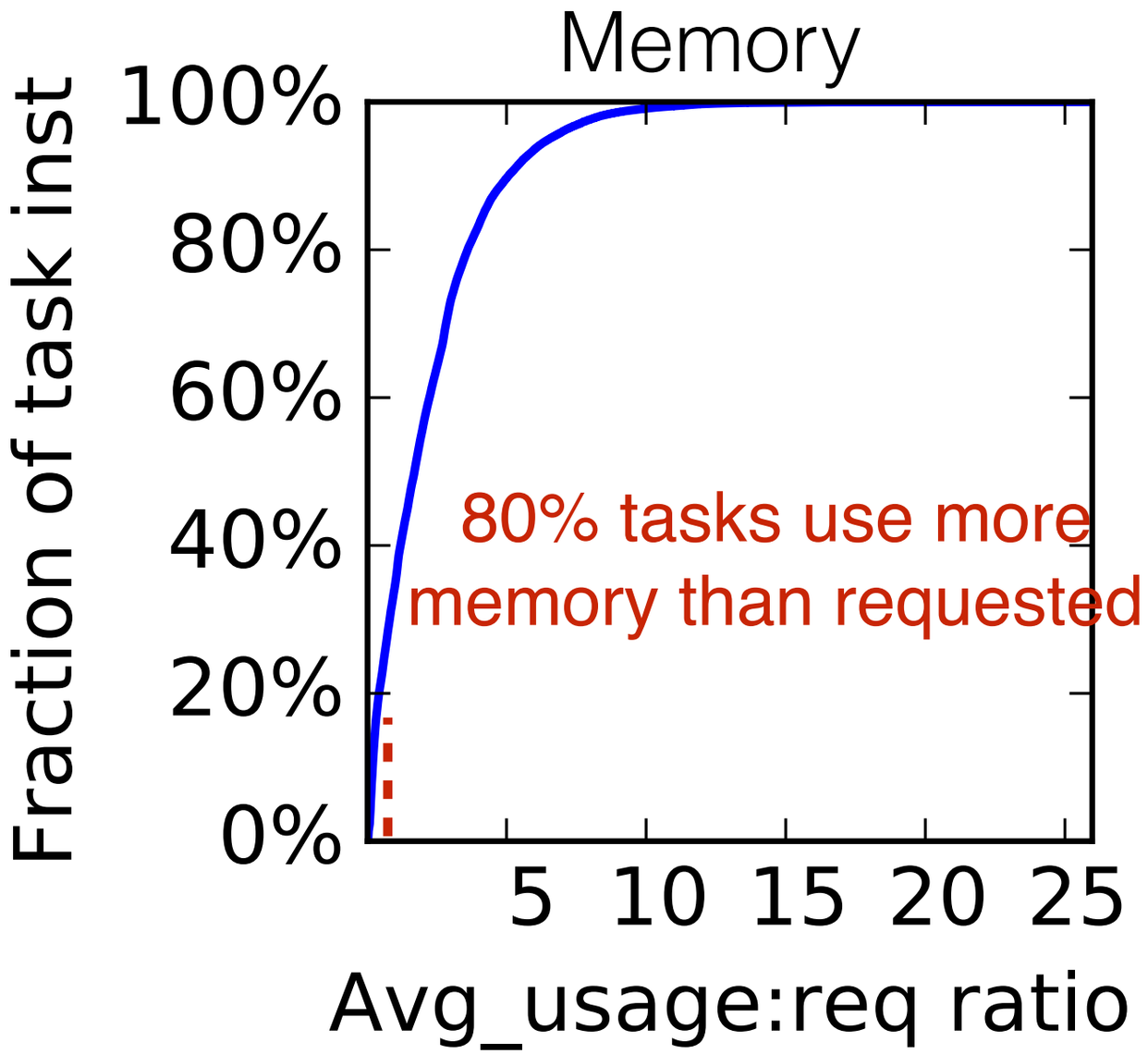
Transient, batch processing workloads: Requested resource vs. average usage





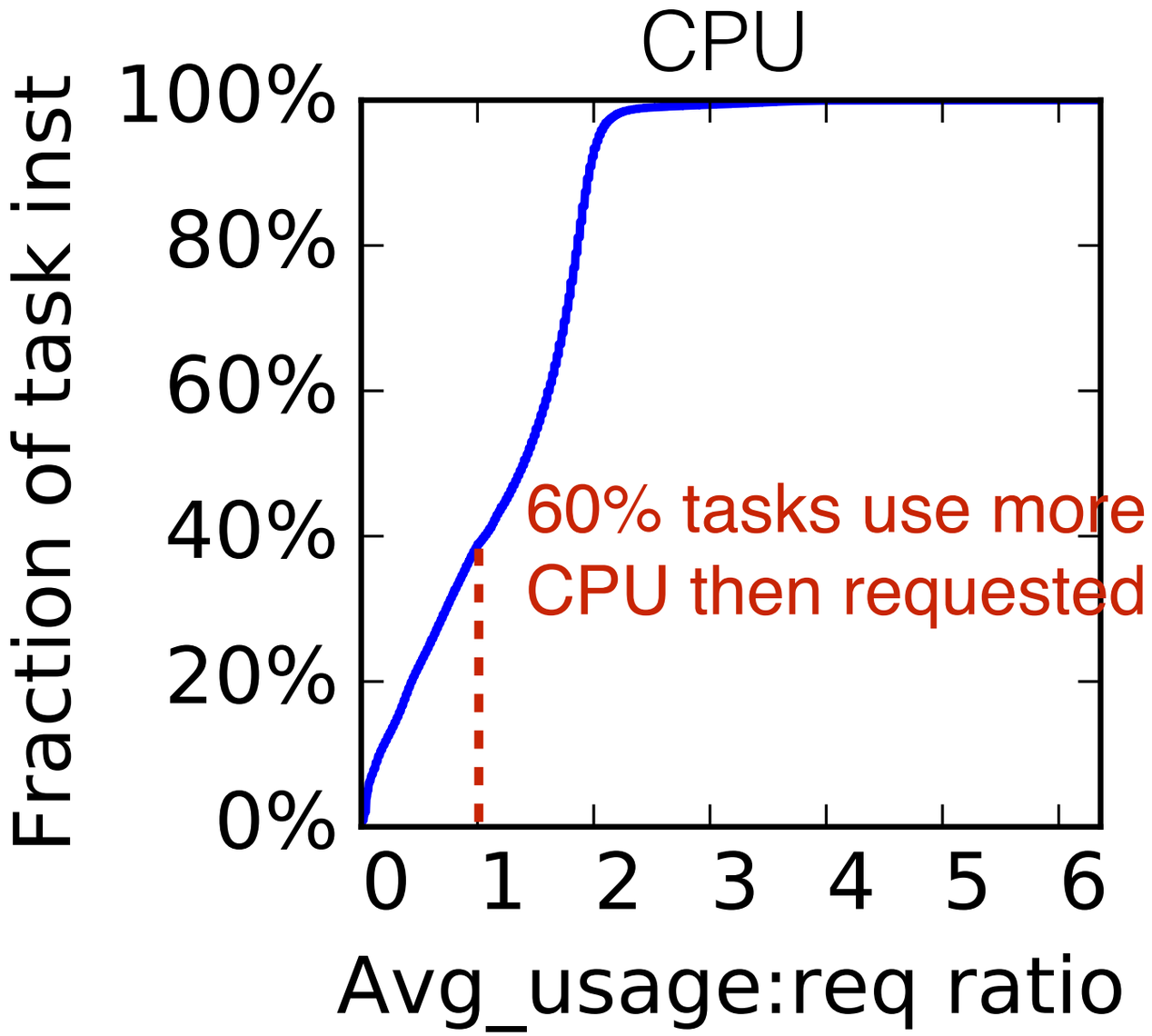
Transient, batch processing workloads: Requested resource vs. average usage

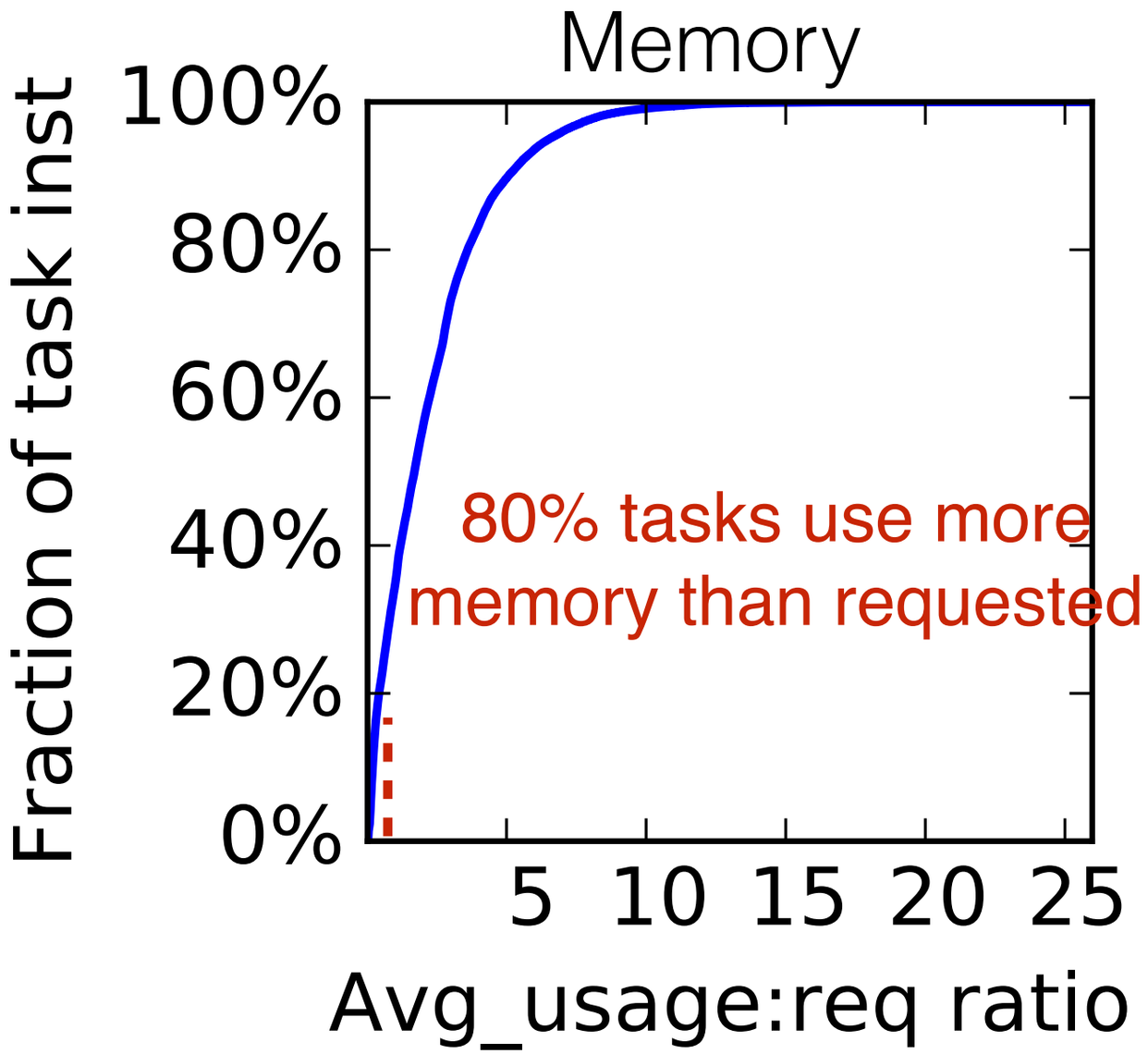




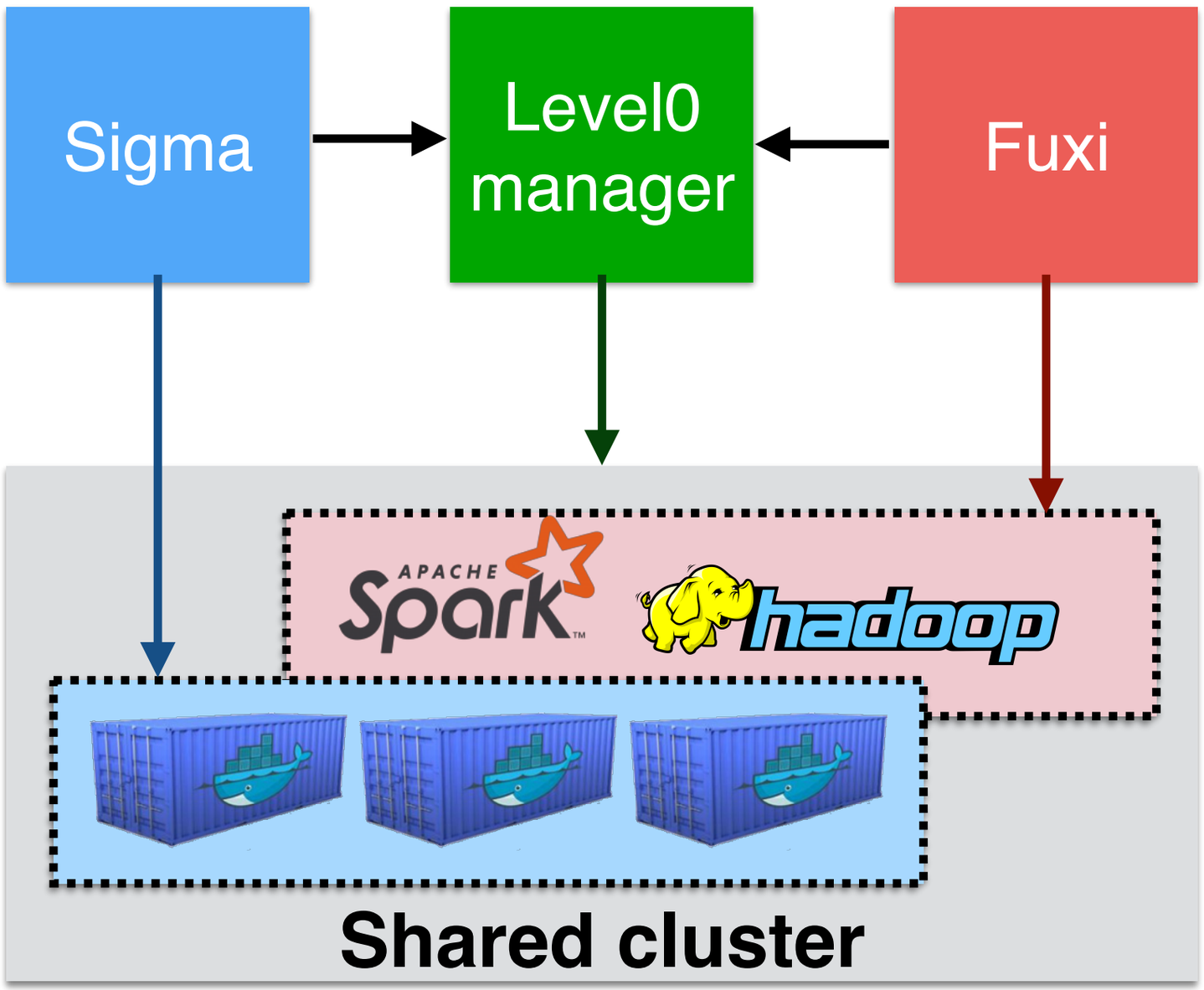
Transient, batch processing workloads: Requested resource vs. average usage

Batch jobs elastically overcommit resources reserved by containerized online services

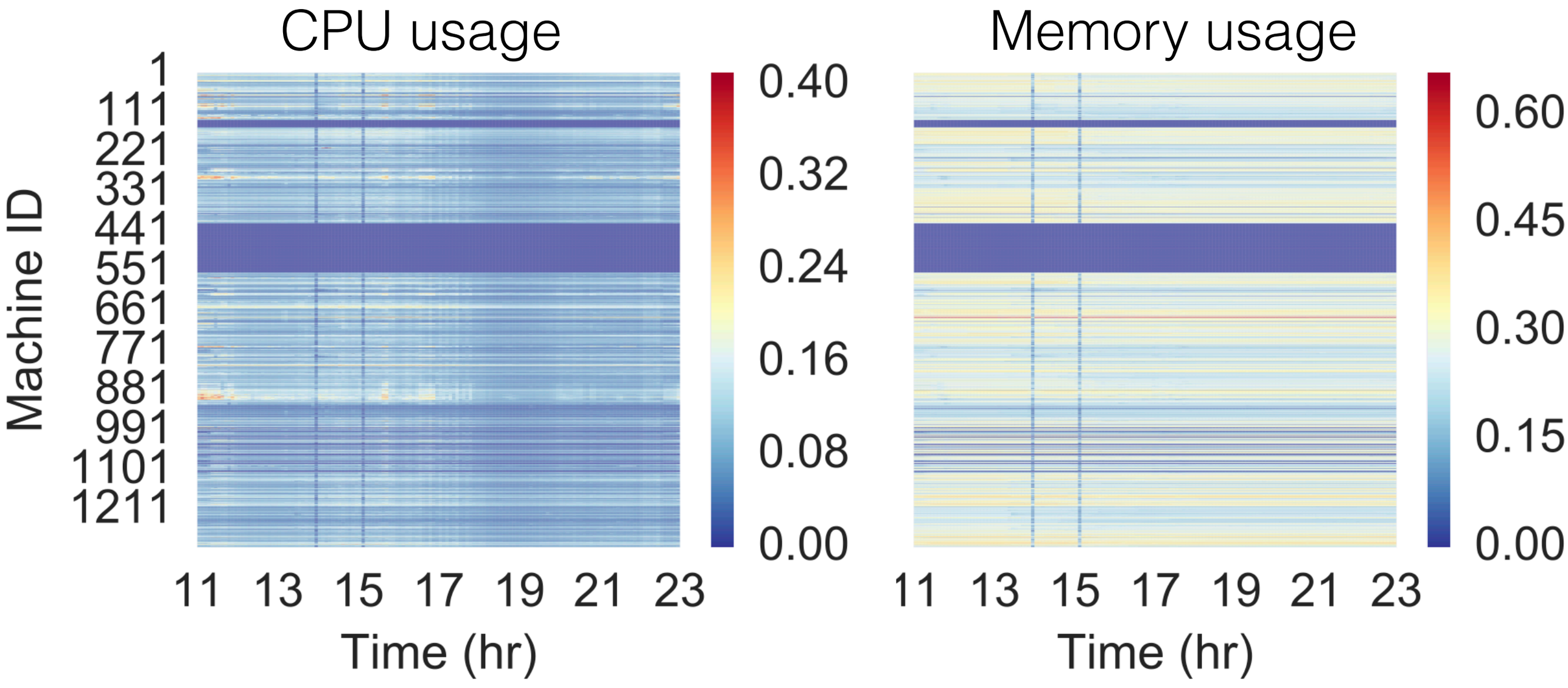




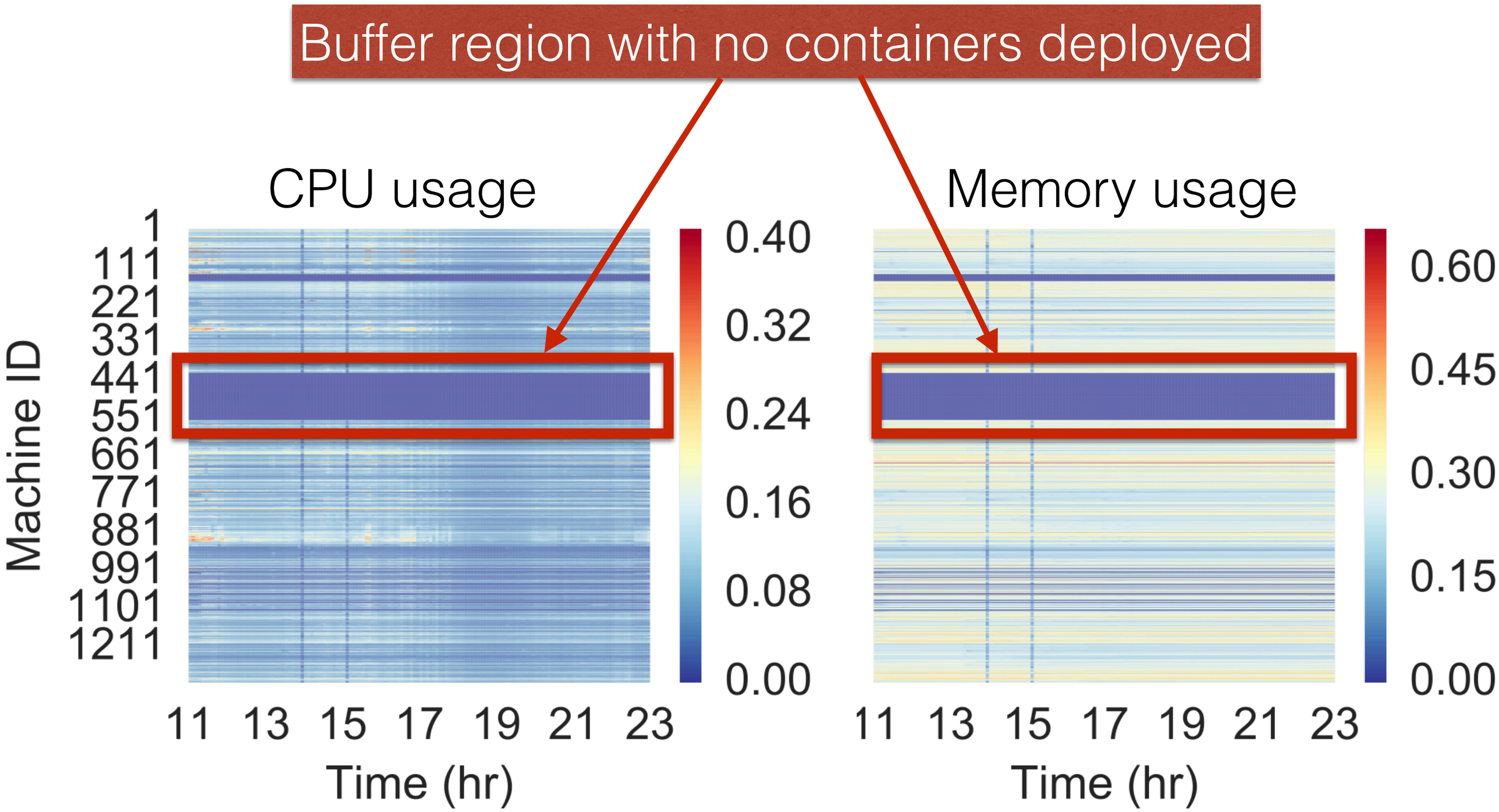
Workload co-location



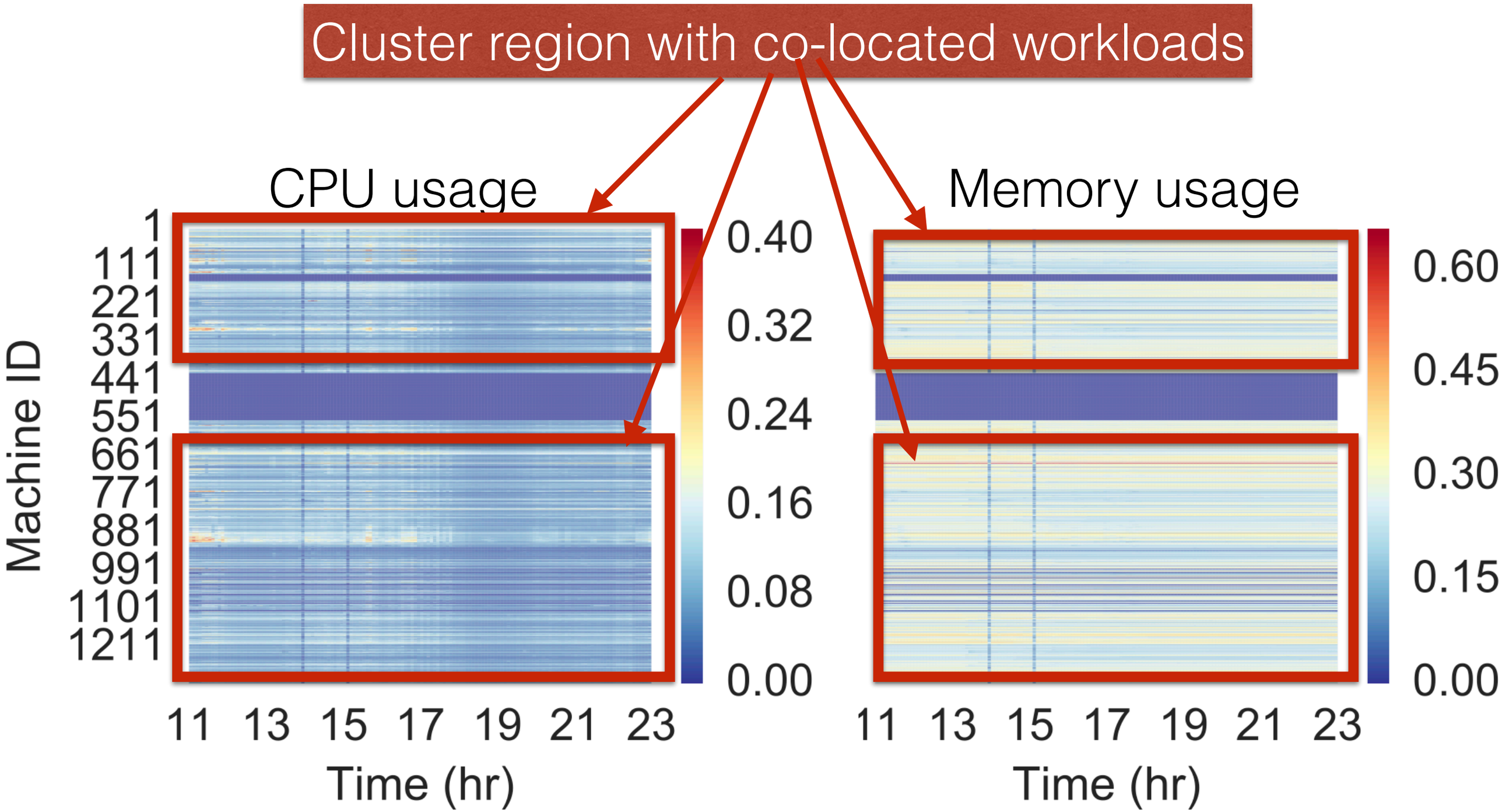
Resource usage heatmap of containerized workloads



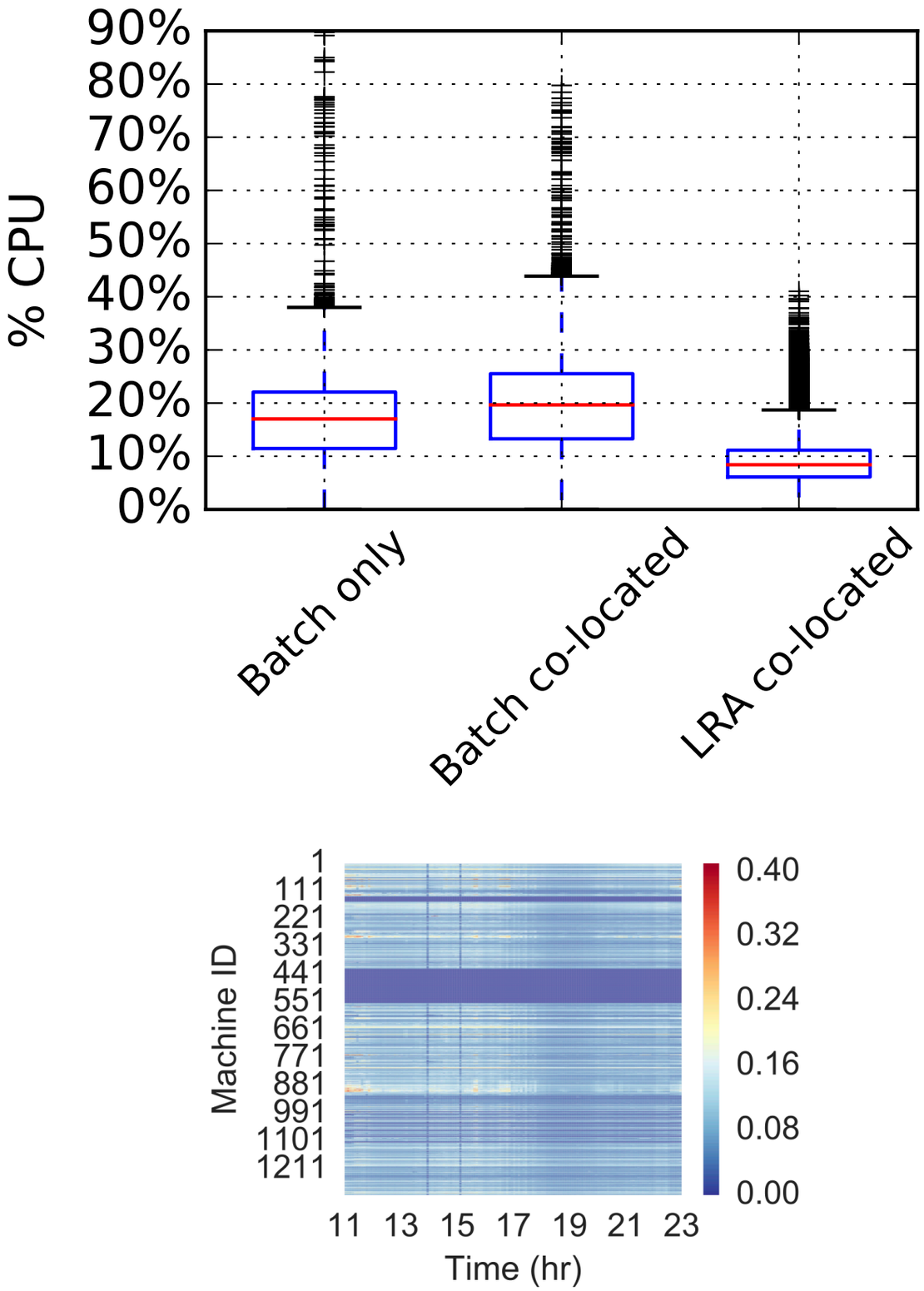
Cluster regions

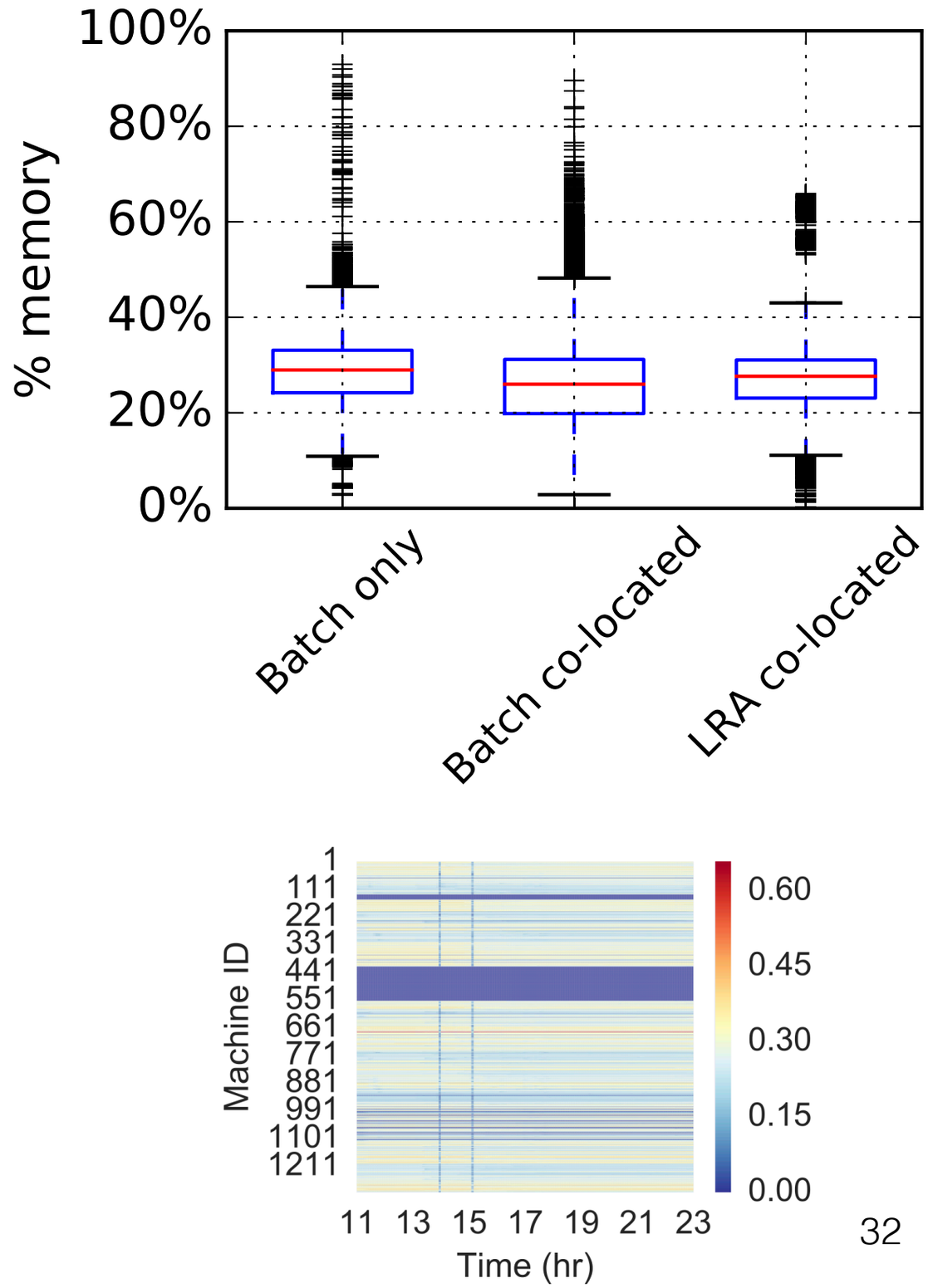


Cluster regions

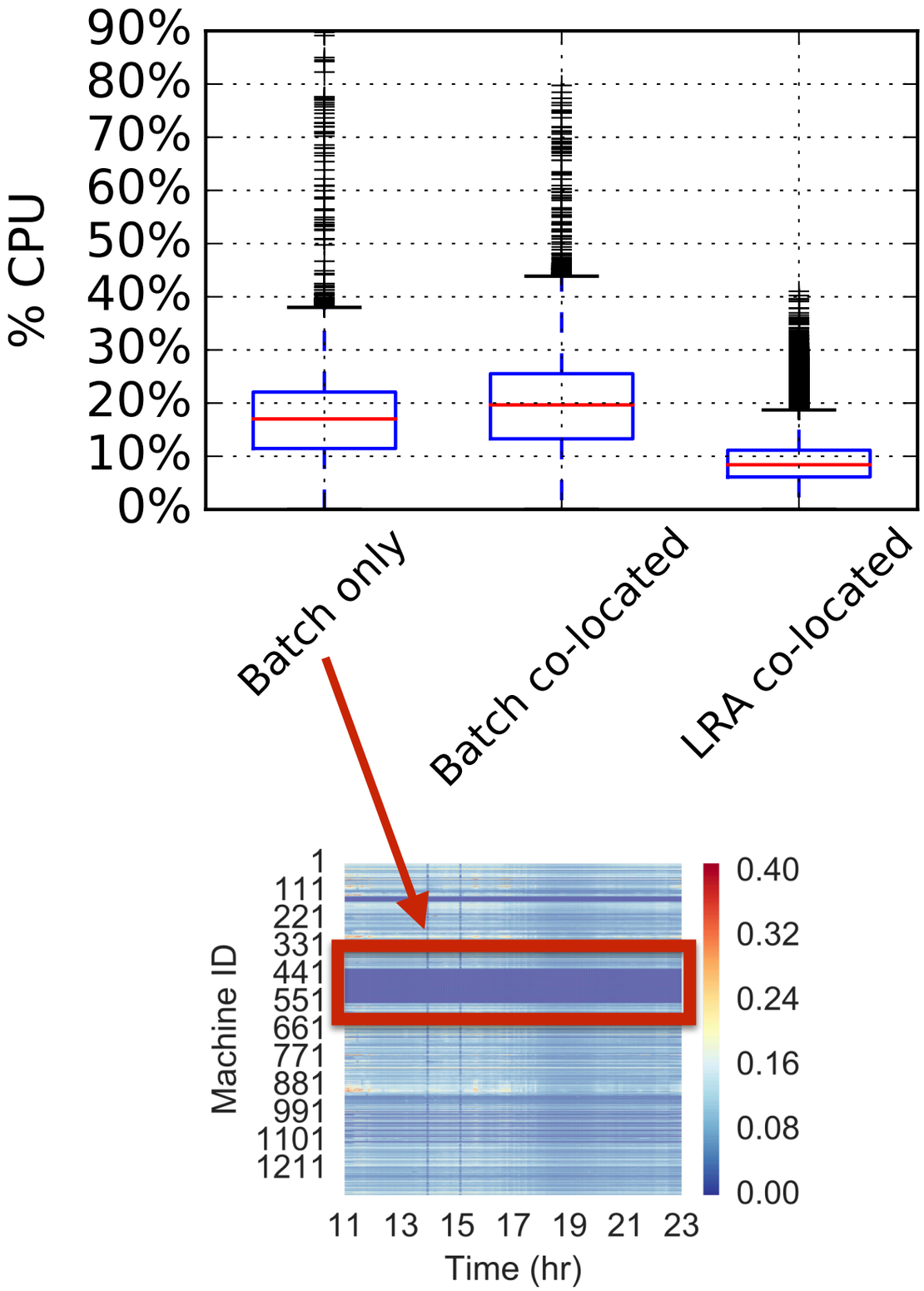


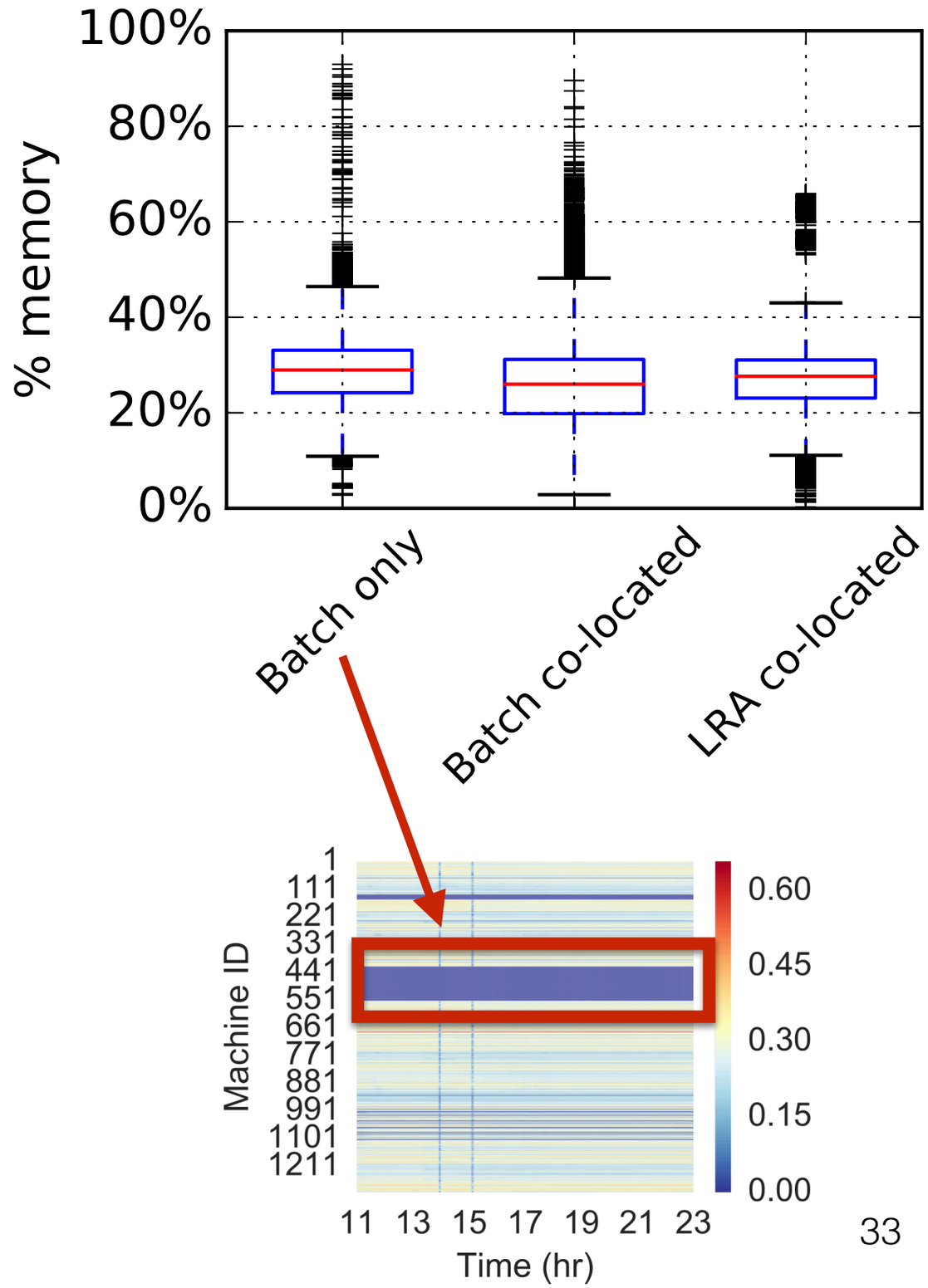
Resource usage at different cluster regions



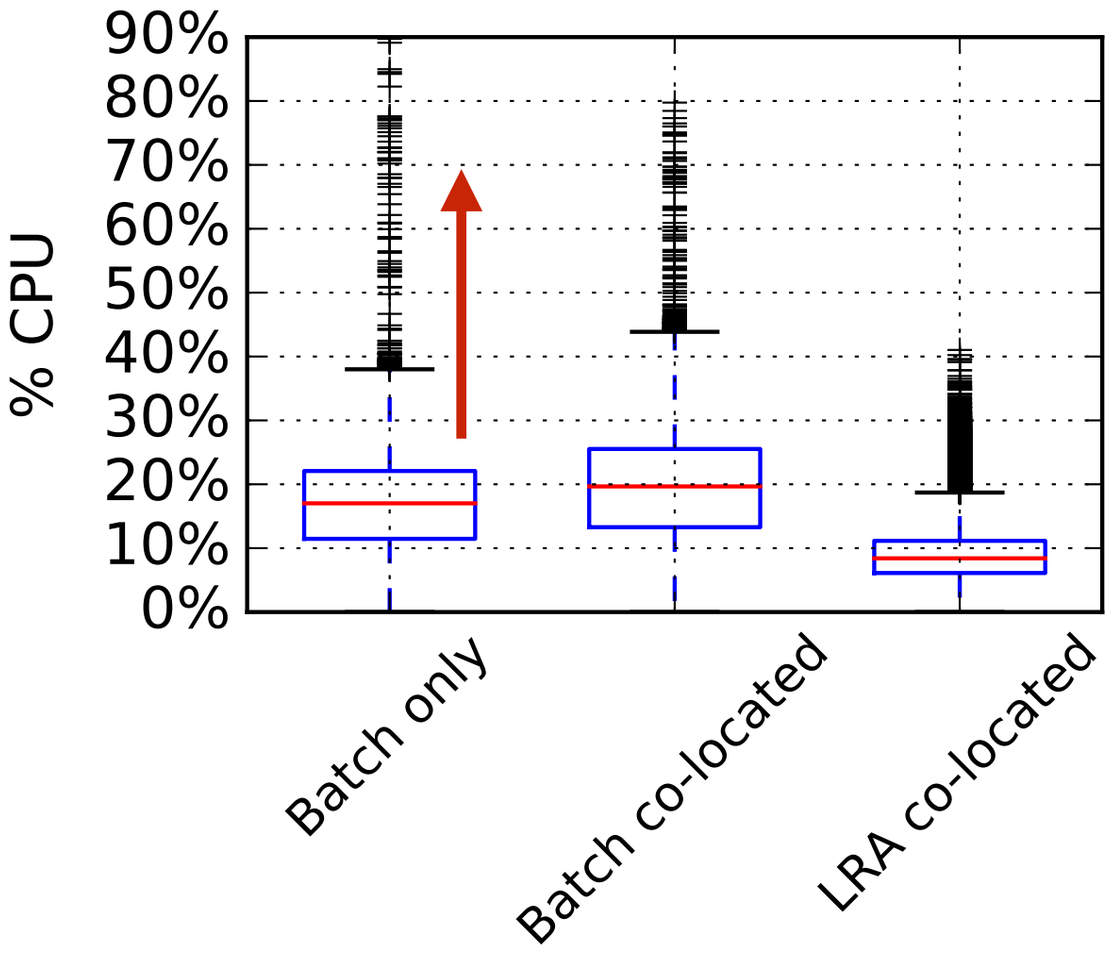


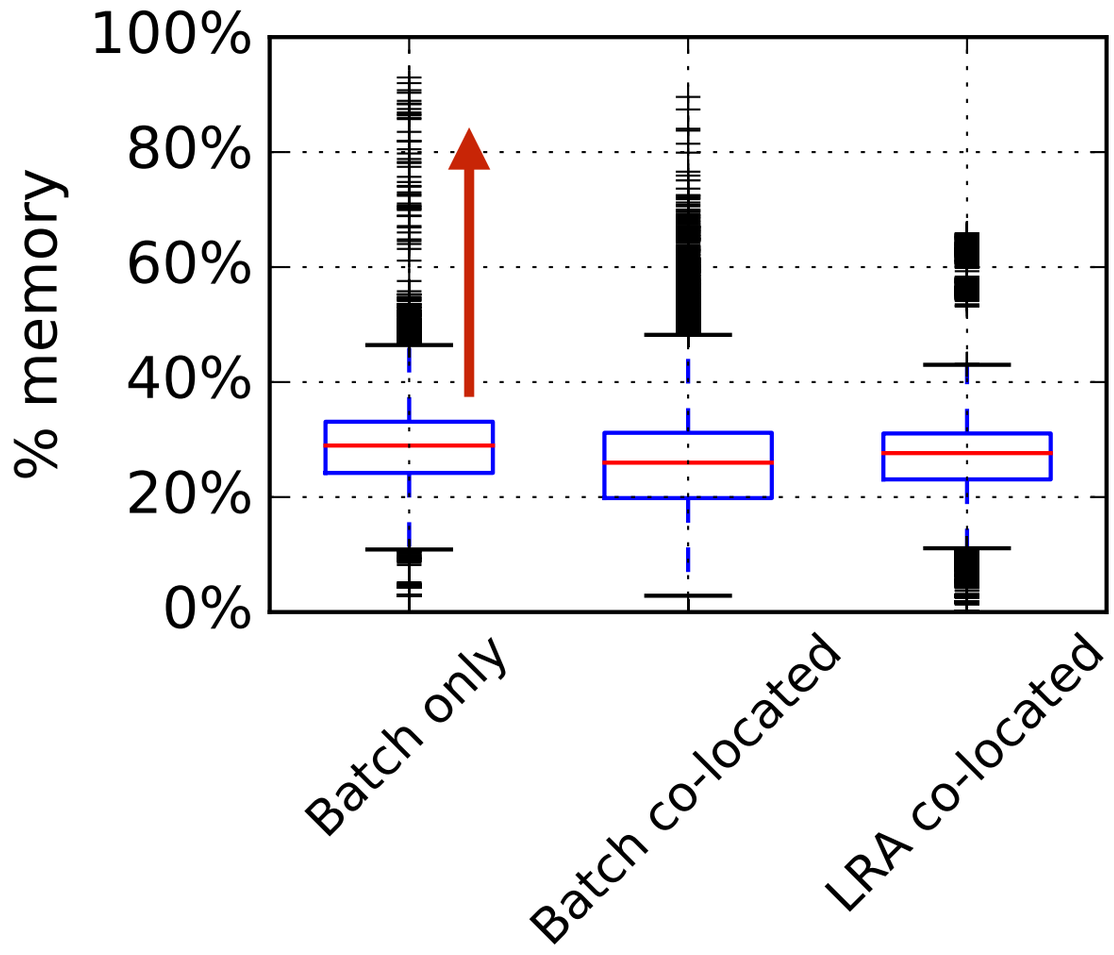
Resource usage at different cluster regions





Resource usage at different cluster regions





The batch only region has potential to improve its resource utilization by accommodating more batch jobs in there

Summary

Alibaba's co-located workloads tend to be more memory-demanding

- Cluster spends over **80% time w/ 10-30% CPU usage**

Workloads show **complementary** patterns

- Long-running containerized online services over-provision resources
- Transient batch processing jobs overcommit unused resources to improve overall resource utilization

Seemingly disjoint scheduling decision making regardless of the co-existence of two workloads

- Need for a more integrated, global Level-0 controller

The RAMCloud project

- An academic project led by Prof John Ousterhout
  - Inventor of LFS (log-structured filesystem)
- Initiated back in 2010/2011
- A bunch of spinoff projects
  - Raft consensus algorithm (ATC 2014)
  - Log-structured memory management (FAST 2014)
  - Better linearizability (SOSP 2015)
  - SLIK: Low-latency KV store indexing (ATC 2016)
  - NanoLog: Nanosecond-scale logging system (ATC 2018)
  - Arachne: Core-aware thread management (OSDI 2018)