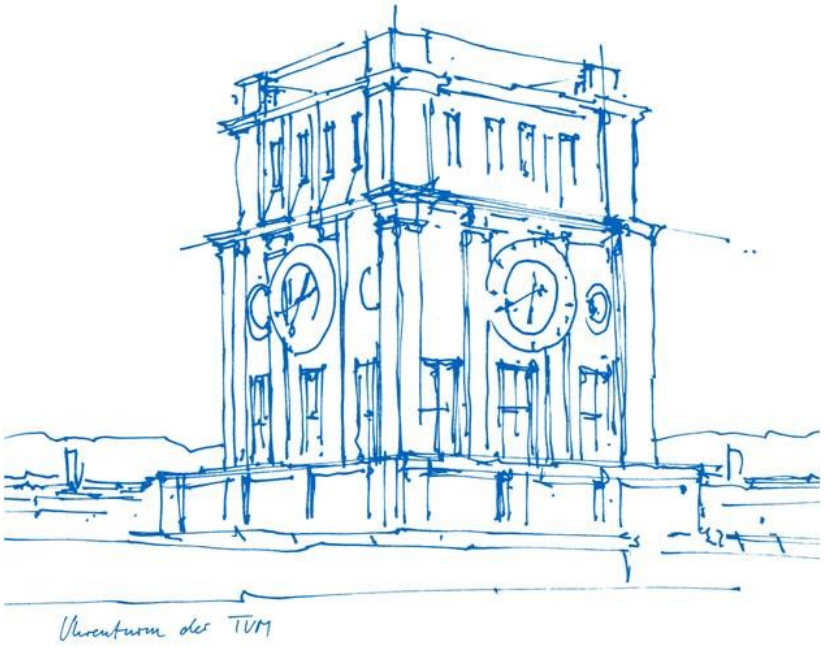
Chair of Communication Networks
Departement of Electrical and Computer Engineering
Technical University of Munich

AHAB: Data-Driven Virtual Cluster Hunting

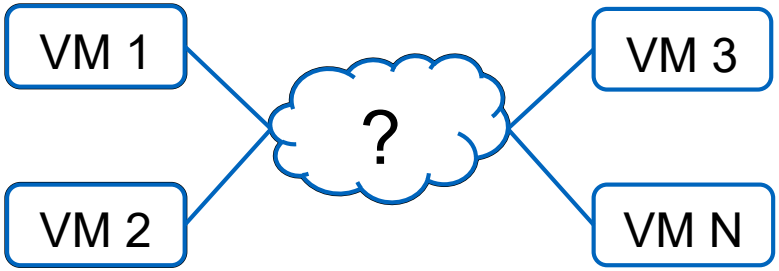
Johannes Zerwas* Patrick Kalmbach* Carlo Fuerst°
Arne Ludwig° Andreas Blenk* Wolfgang Kellerer*
Stefan Schmid^

**Technical University of Munich, Germany*
°*Technical University of Berlin, Germany*
^*University of Vienna, Austria*

IFIP Networking 2018, Zurich, Switzerland



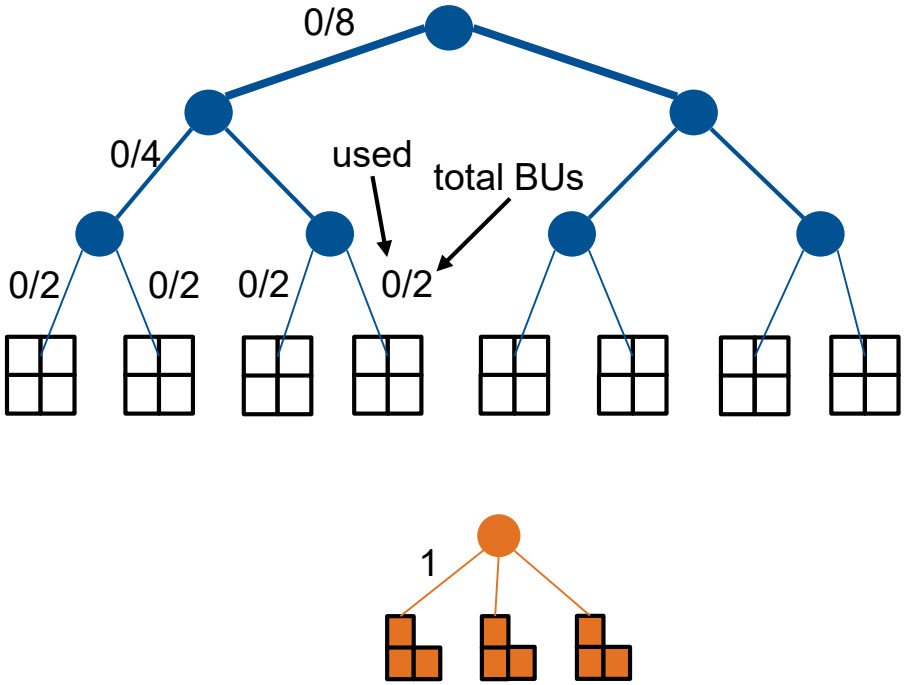
Context



- Increased use data-intensive applications in shared data centers
- Many provider-tenant interfaces neglect network as a resource
- Problems:
  - Unpredictable application performance
  - Limited applicability of cloud
  - Inefficiencies in production data centers

- Solution: Network-aware abstraction - Virtual Cluster (ACM SIGCOMM 2011)

Background: Virtual Cluster Abstraction



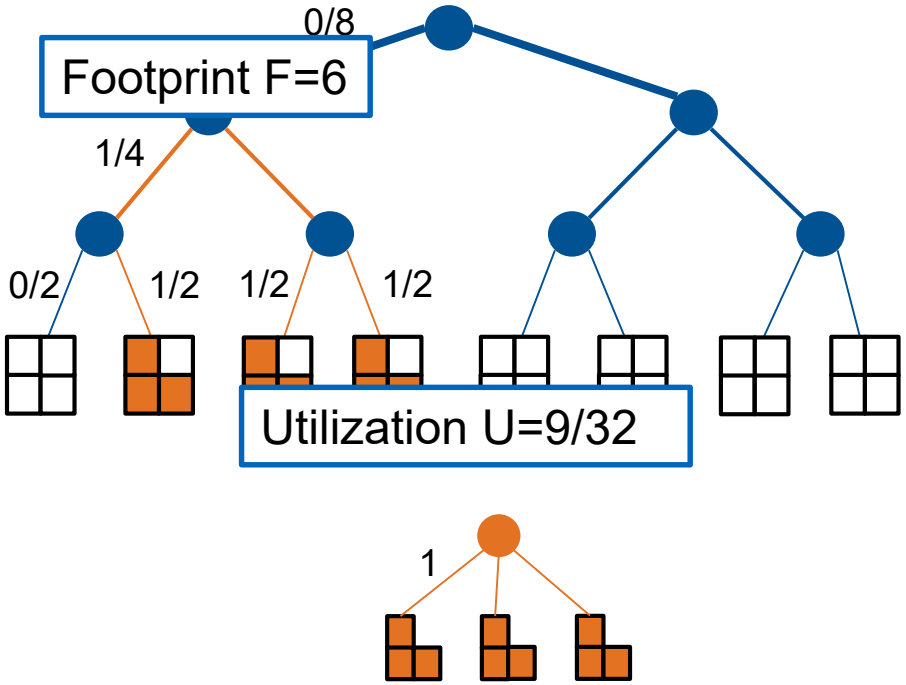
Physical Cluster

- Compute Units (CUs)
- Bandwidth Units (BUs)
- Tree-like topology (abstracted from Fat-Tree)

Virtual Cluster (VC)

- Number of VMs (N)
- Size of VMs (S)
- Bandwidth (B)
- Lifetime given resource fulfillment

Background: Virtual Cluster Abstraction



Physical Cluster

- Compute Units (CUs)
- Bandwidth Units (BUs)
- Tree-like topology (abstracted from Fat-Tree)

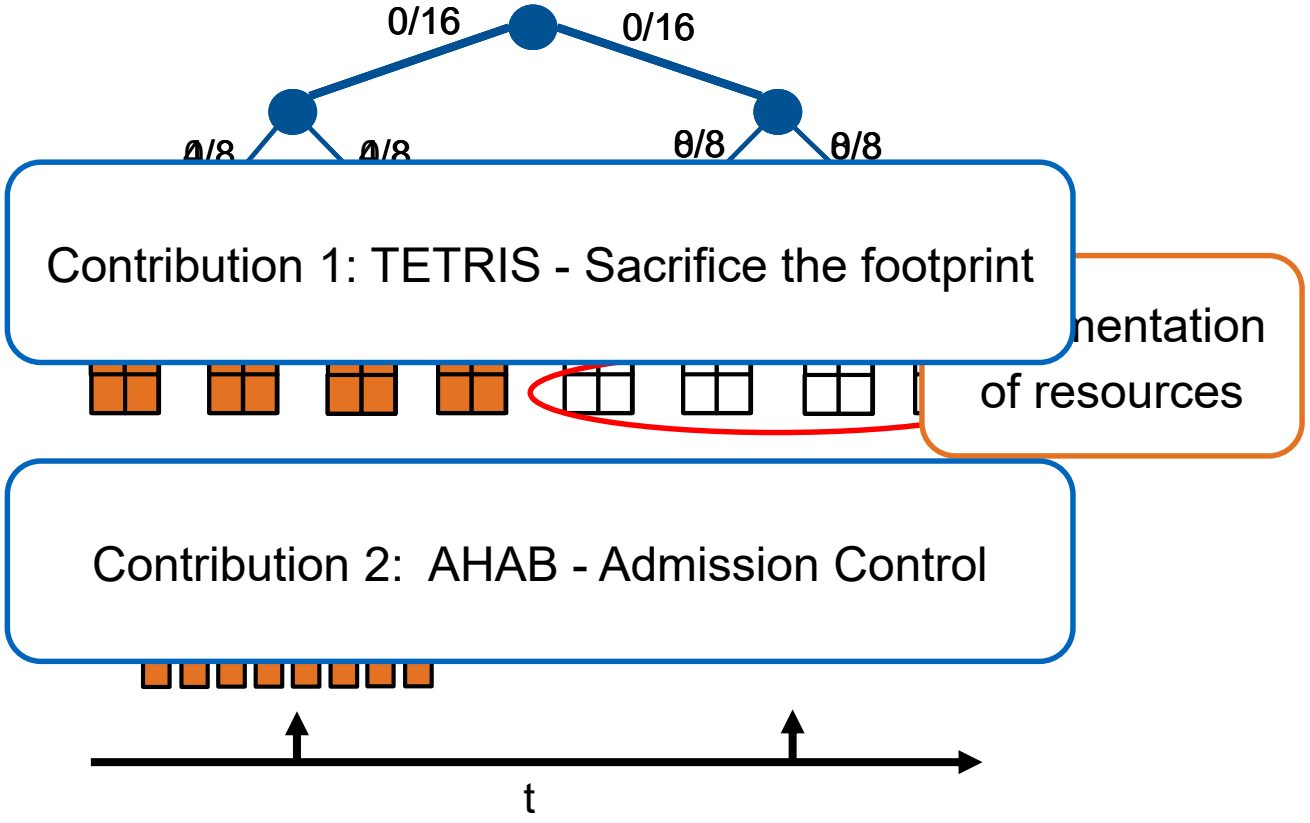
Virtual Cluster (VC)

- Number of VMs (N)
- Size of VMs (S)
- Bandwidth (B)
- Lifetime given resource fulfillment

Problem: Resource Fragmentation

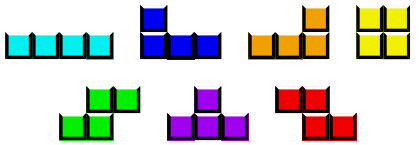
Existing allocation algorithms focus on single request:

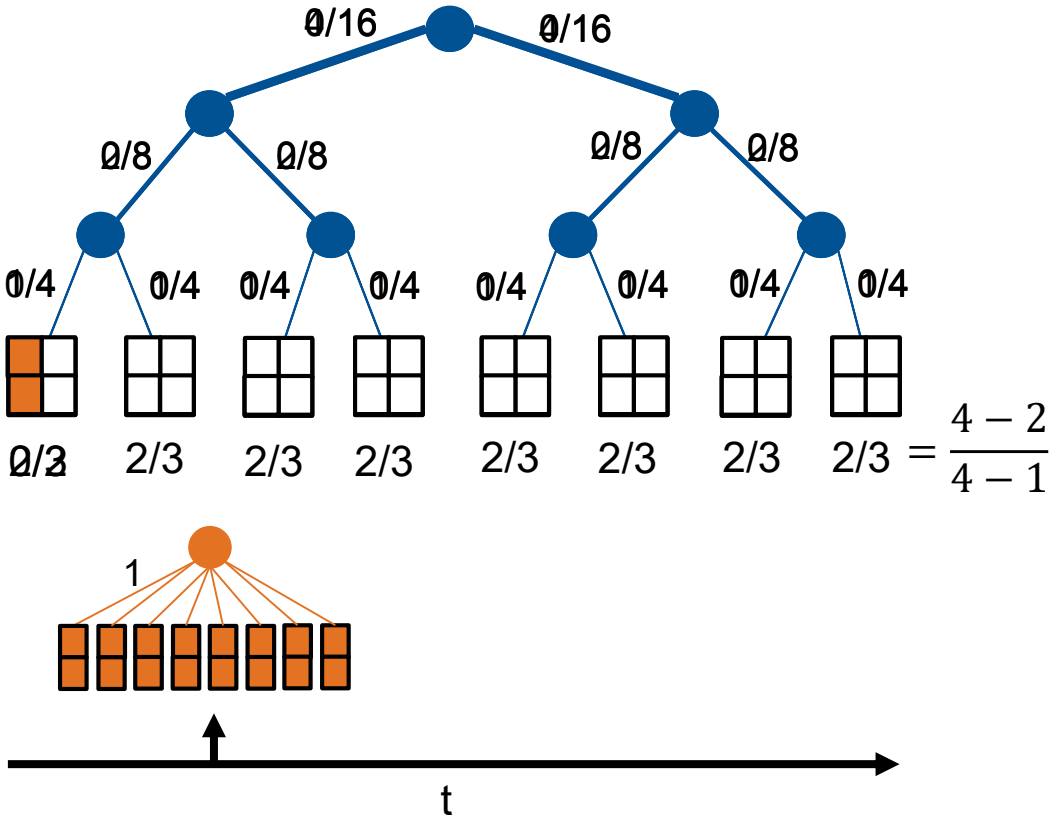
- Oktopus (ACM SIGCOMM 2011)
- Kraken (IEEE/ACM TON 2018)



TETRIS: Sacrifice Footprint for Fragmentation

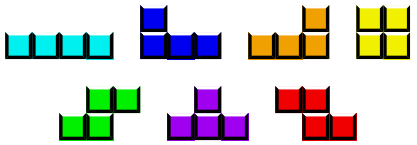
Choose hosts with max. ratio of residual resources

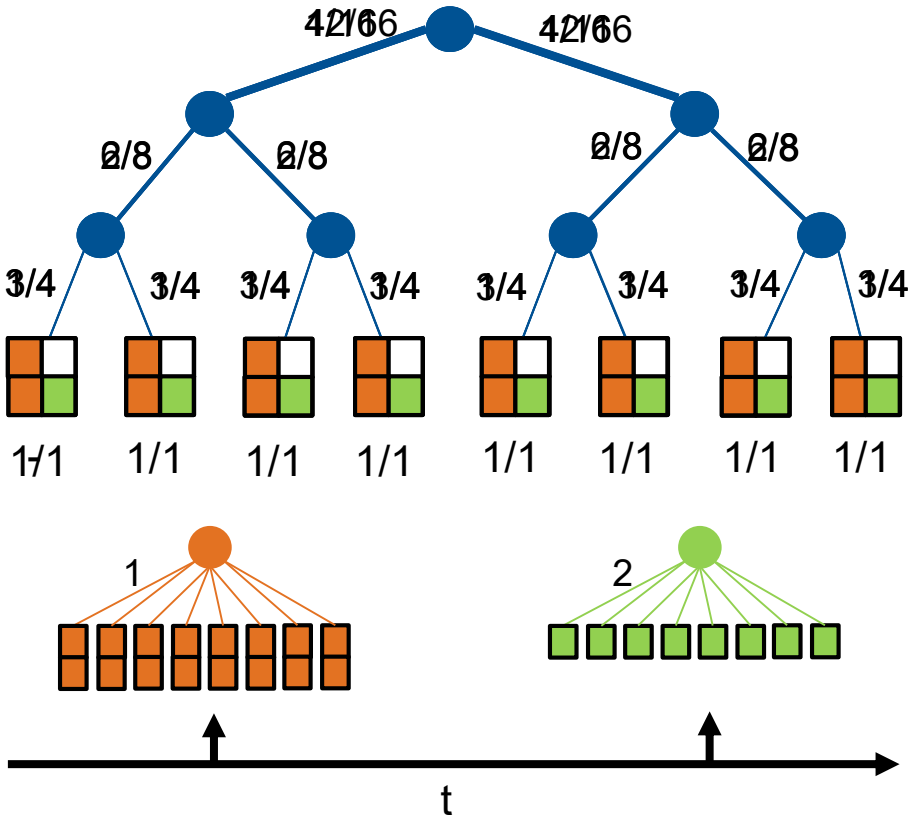




TETRIS: Sacrifice Footprint for Fragmentation

Choose hosts with max. ratio of residual resources



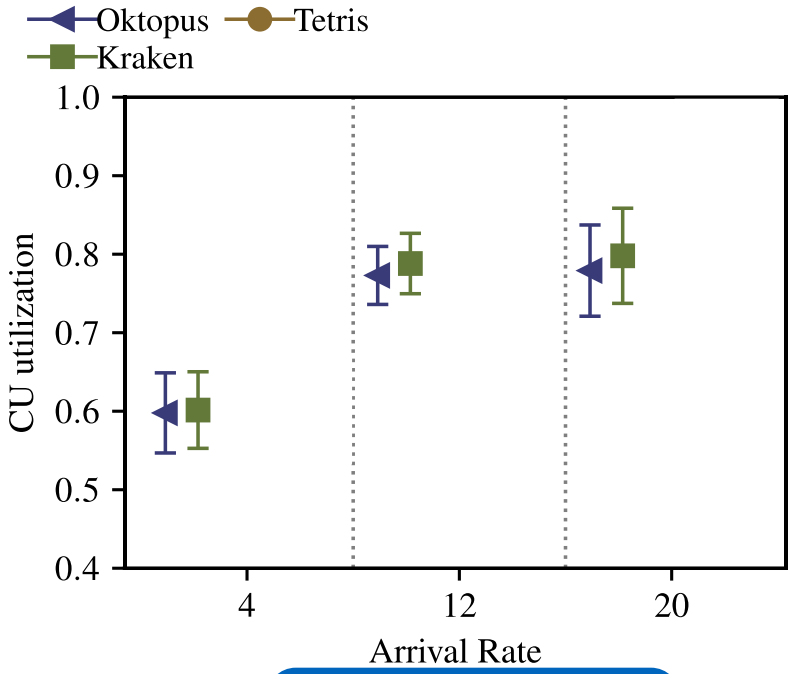


Resources still usable

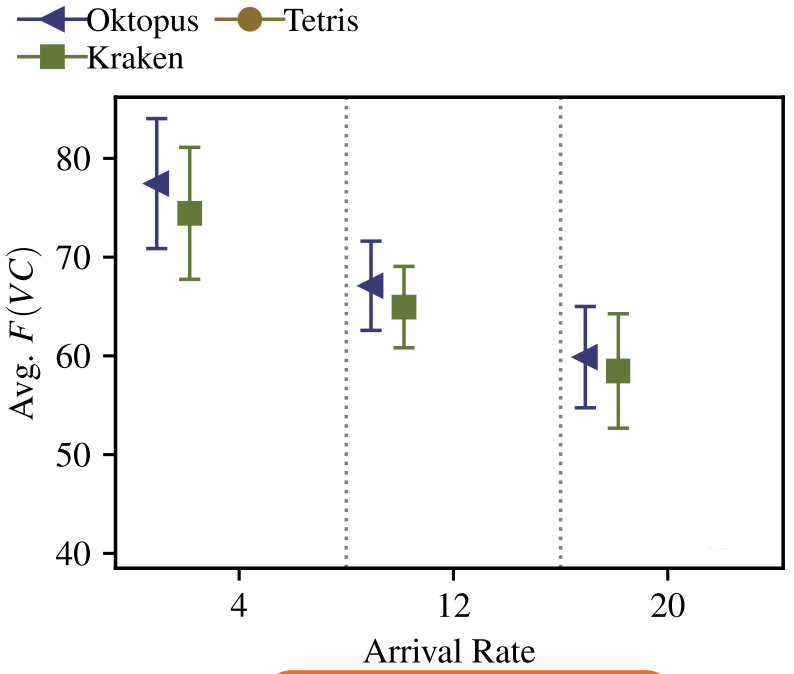
Algorithm Evaluation

- Baseline: OKTOPUS (ACM SIGCOMM 2011), KRAKEN (IEEE/ACM TON 2018)
- Physical Cluster: Fat-Tree with k=12, 8CUs and 8BUs
- Performance metrics: CU Utilization, avg. VC Footprint
- Virtual Cluster Requests:
  - 1000 / run with varying arrival rates
  - Num. VMs, size VMs, BW similar to traces from Google & Microsoft

TETRIS Evaluation

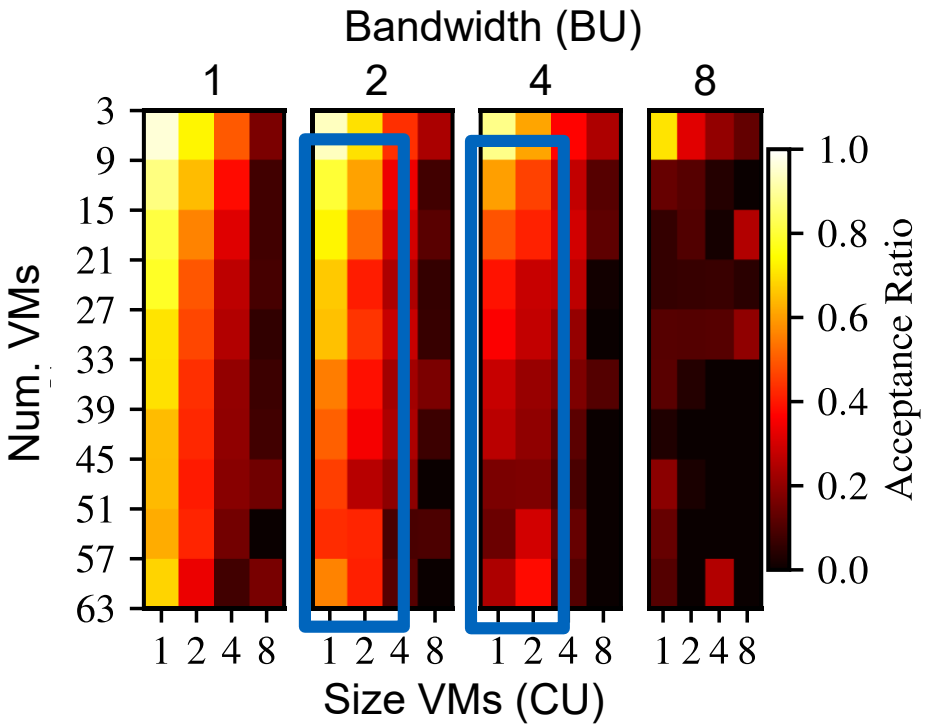


+5% utilization



+10% footprint

TETRIS Evaluation



Add Admission Control

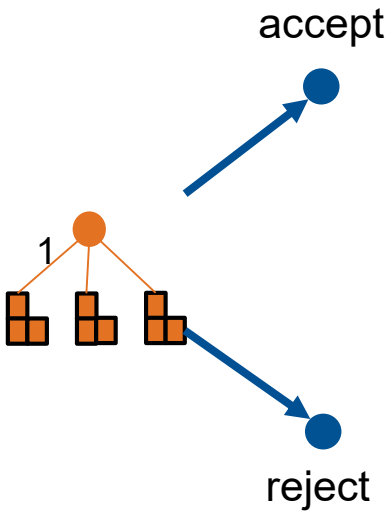
AHAB: The Case for Data-Driven Admission Control

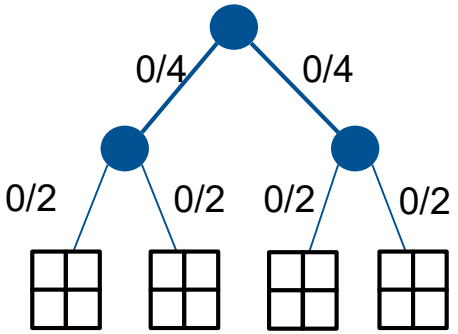
Leverage Knowledge

Monte Carlo Tree Search

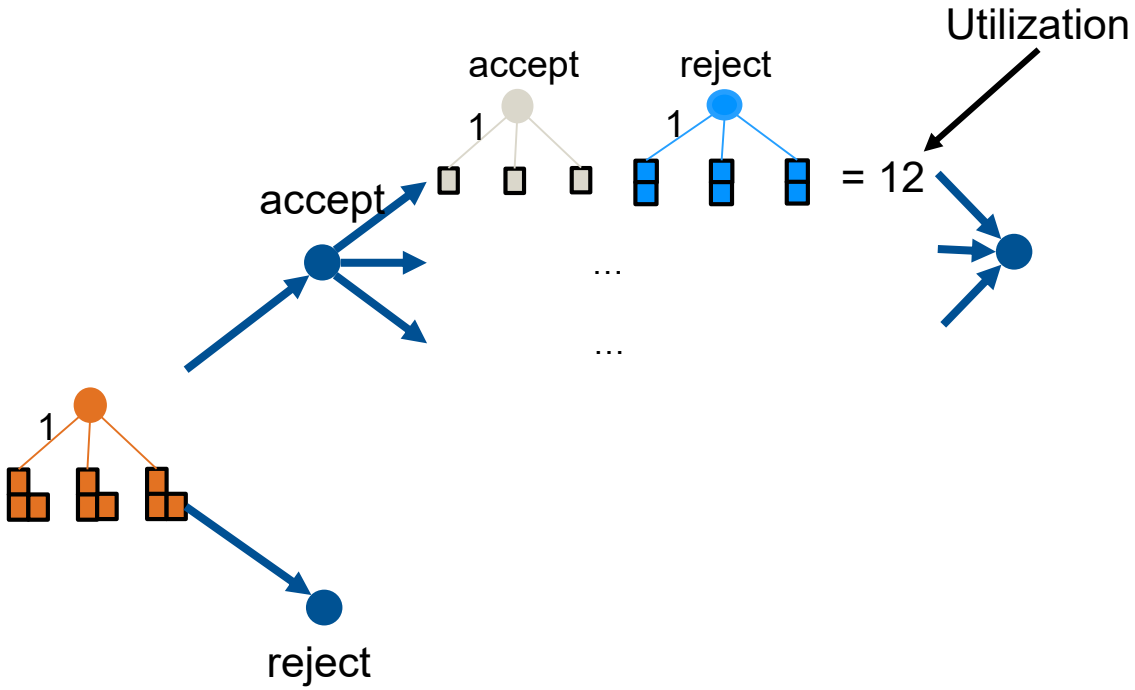
Data-Driven Decision

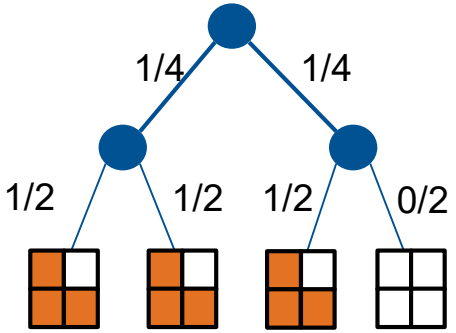
AHAB: The Case for Data-Driven Admission Control



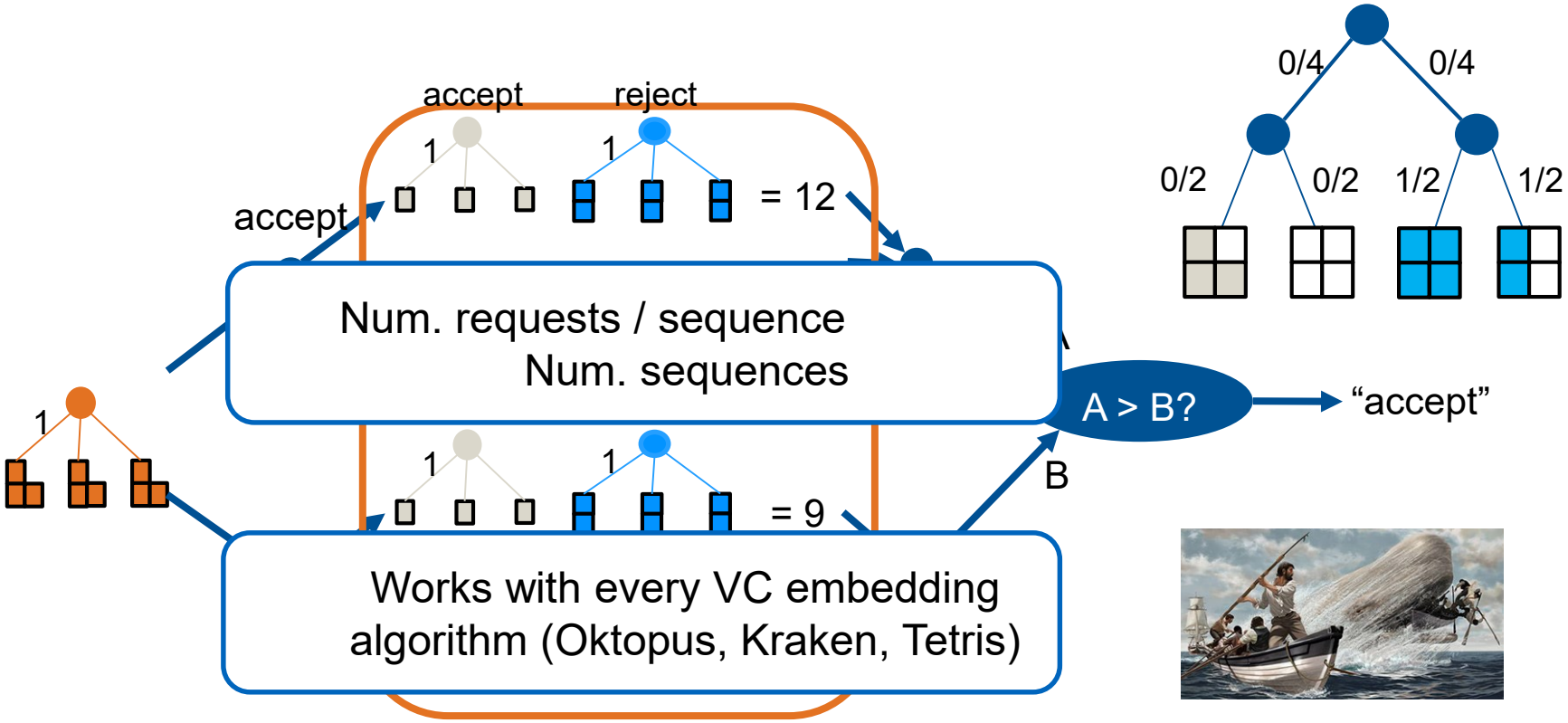


AHAB: The Case for Data-Driven Admission Control

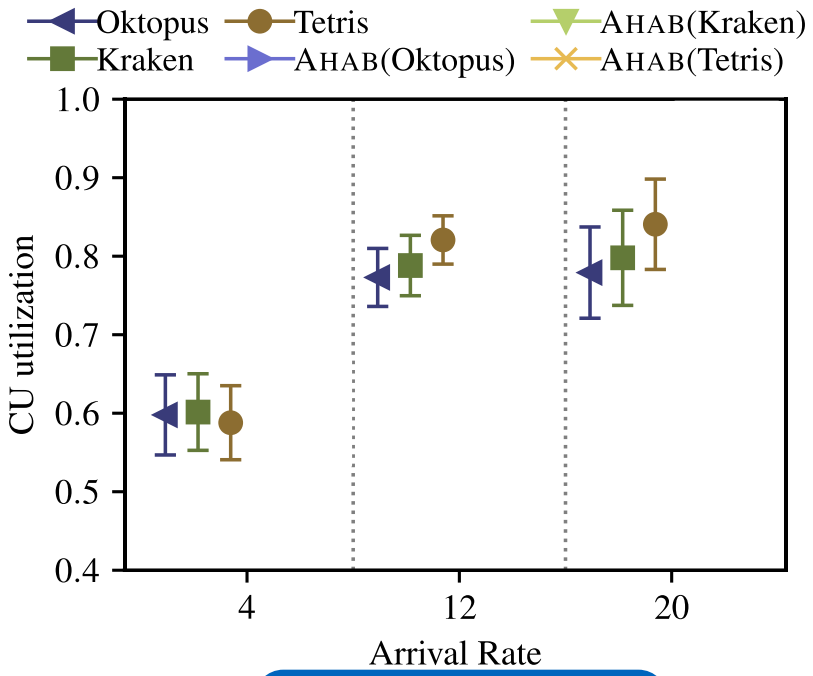




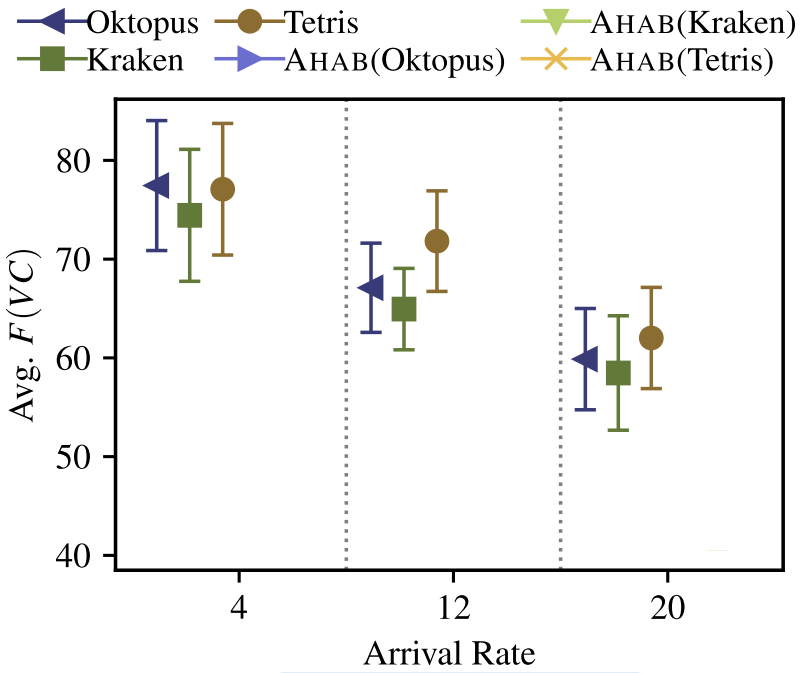
AHAB: The Case for Data-Driven Admission Control



AHAB improves utilization

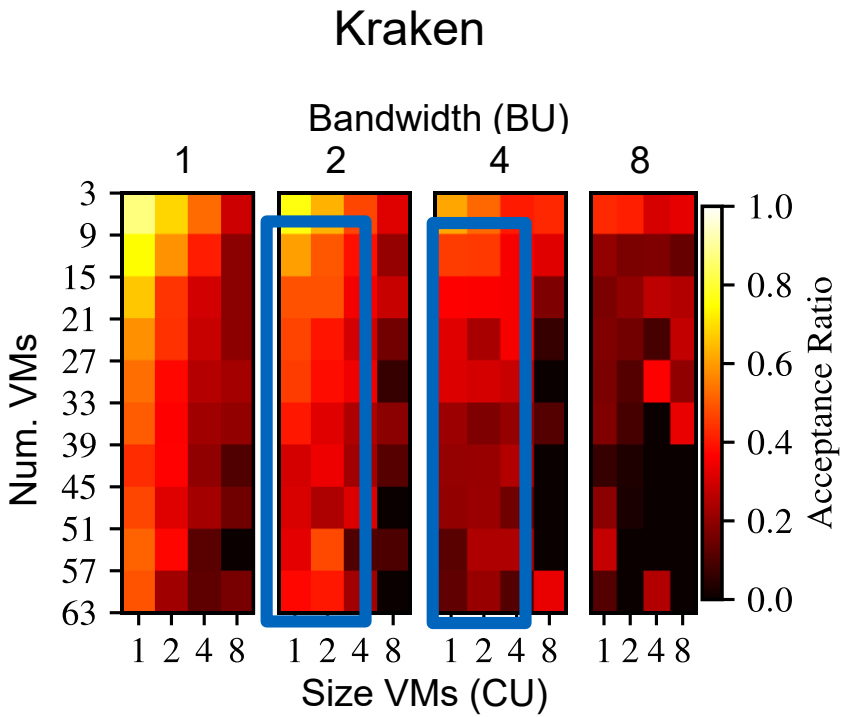


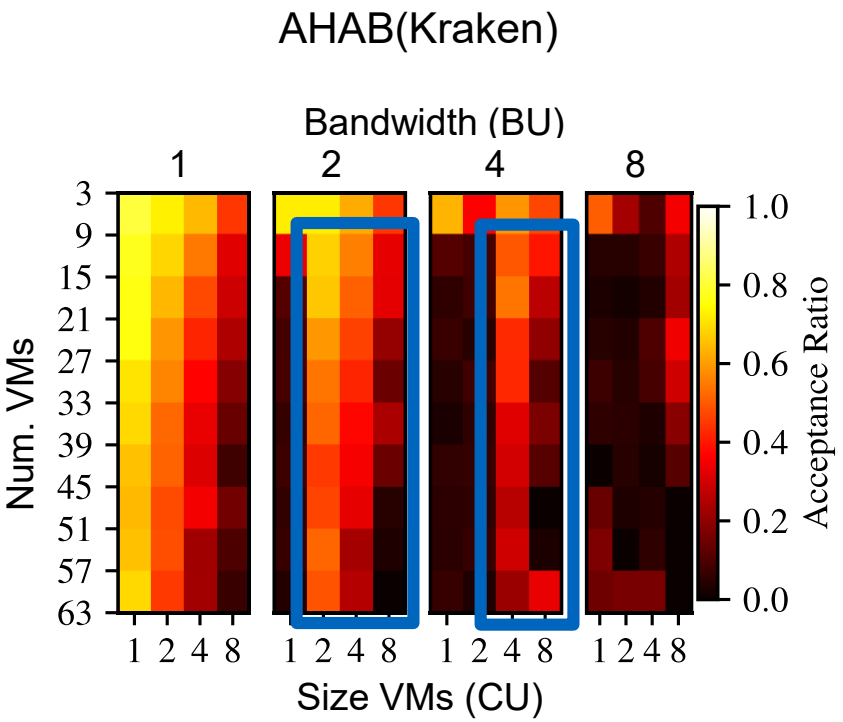
+10% utilization



-25% footprint

Why is AHAB better?

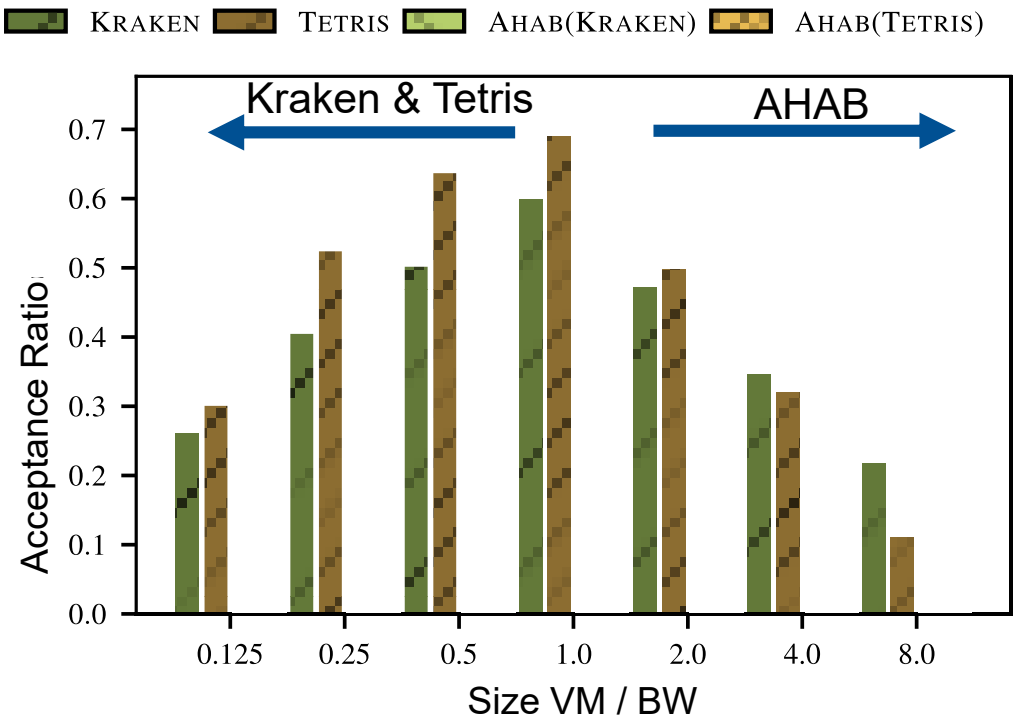




Small VMs
Large BW

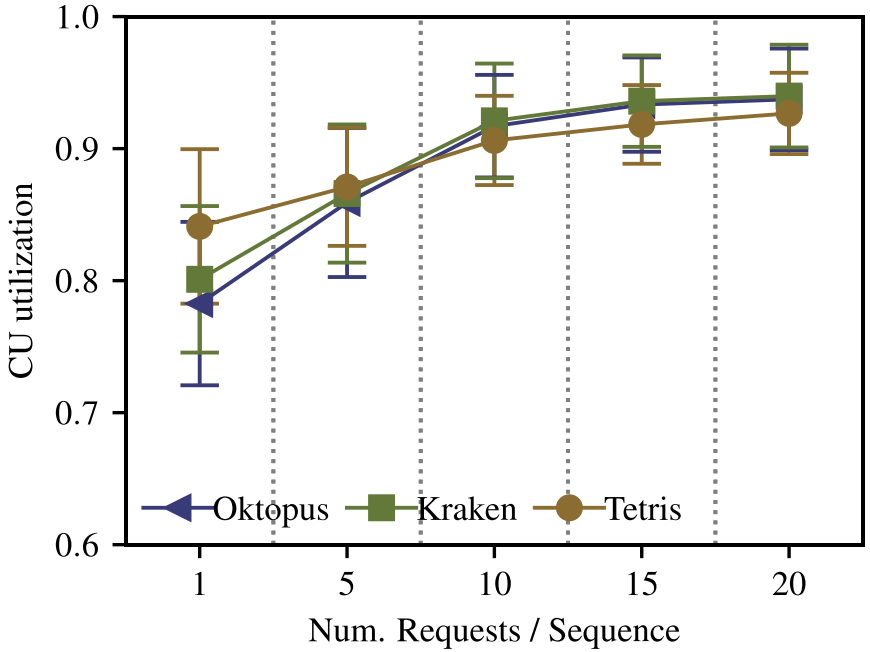
Large VMs
Small BW

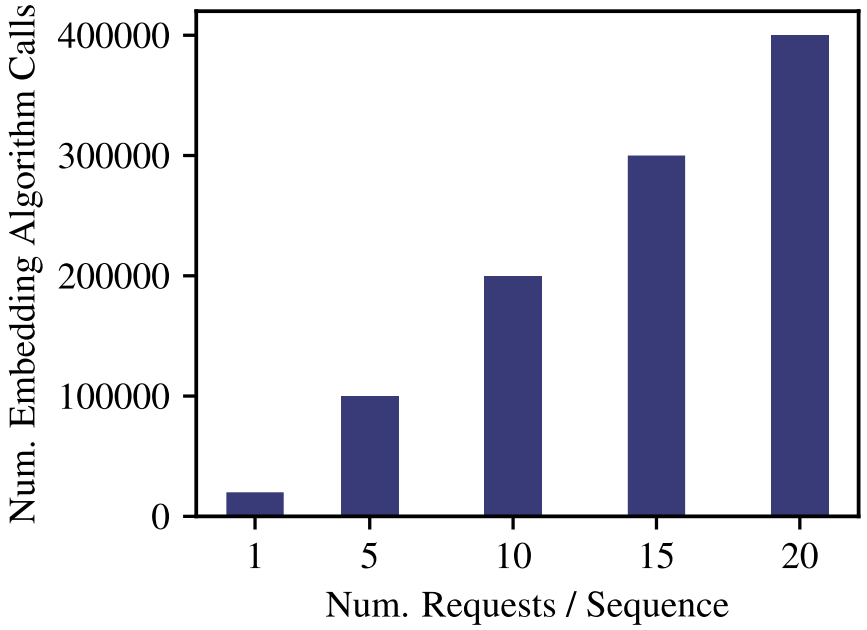
Why is AHAB better?



AHAB accepts more valuable requests

Optimization Opportunities





Trade-Off: Utilization - Computations

Use ML for speed-up

Summary

- TETRIS sacrifices footprint increase utilization
- AHAB employs a data-driven approach for Admission Control
- AHAB evaluates the impact of a single request on future requests
- AHAB's approach applies also to other use-cases
- Future Work: Use ML to predict AHAB's decisions

Thank you!

Questions?