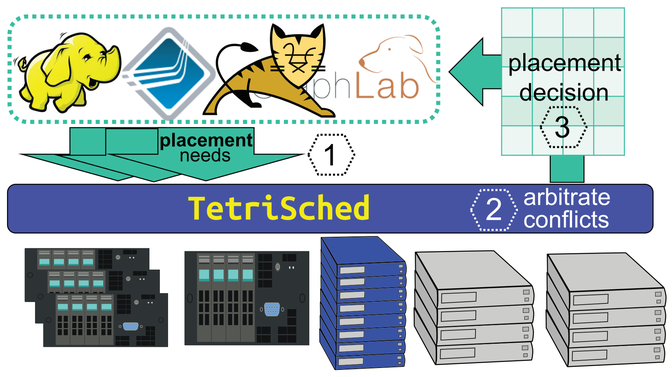
TetriSched: Space-Time Scheduling for Heterogeneous Datacenters

Alexey Tumanov, Timothy Zhu, Michael A. Kozuch*, Mor Harchol-Balter, Greg Ganger (CMU, *Intel)

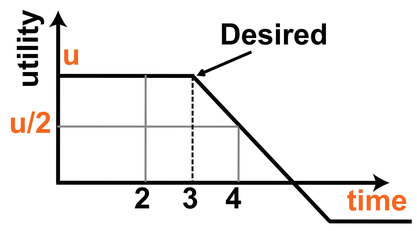
PROBLEM STATEMENT

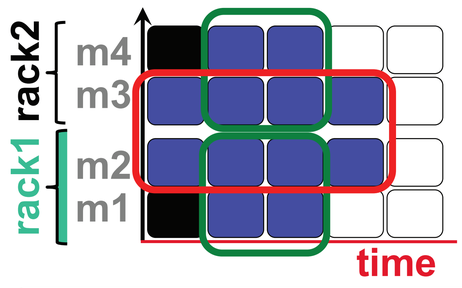


- Datacenters – increasingly heterogeneous
- Datacenter workloads – increasingly diverse
- User objectives – differ, conflict, change
- Cluster schedulers – map work to resources

UTILITY FUNCTIONS

- User-defined utility functions
  - Completion time
  - Availability
  - Queuing delay
- Scheduler-facing utility expressions
  - "n Choose k" building blocks





OR

max

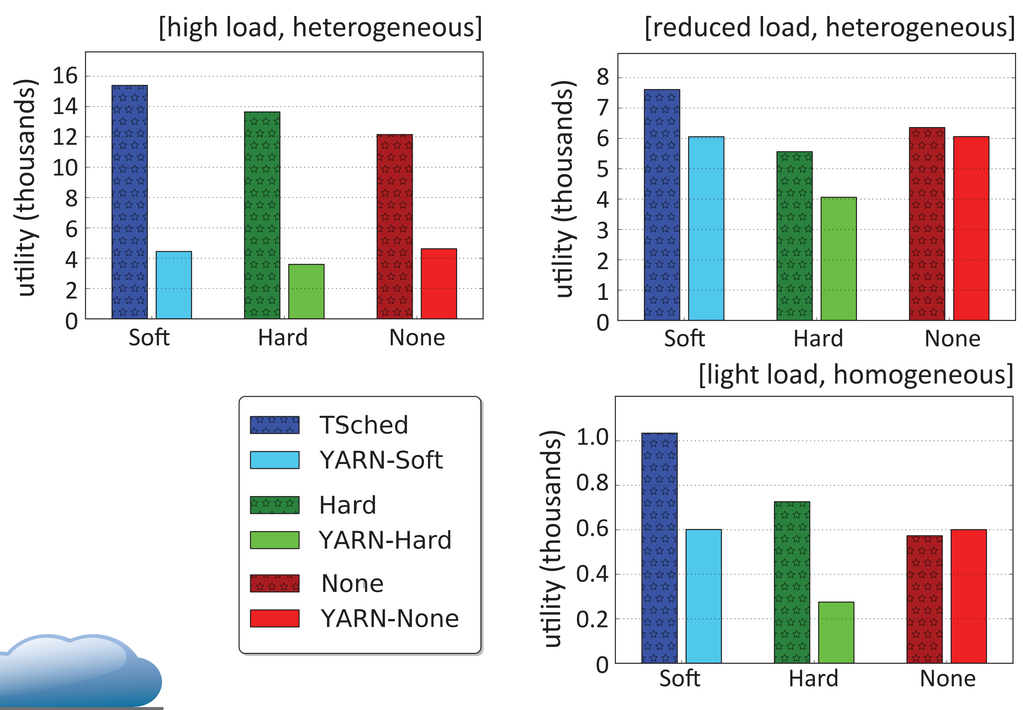
rack1 nCk ($m_i \in$ rack1, k=2, s=1, d=2, u)

rack2 nCk ($m_i \in$ rack2, k=2, s=1, d=2, u)

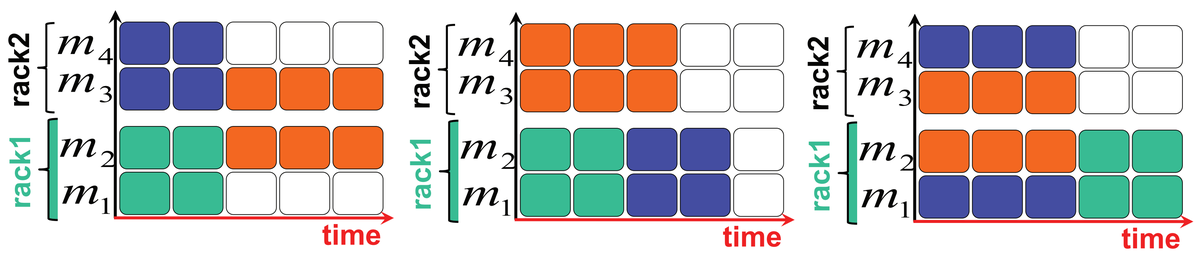
anywhere nCk ($\cup m_i$, k=2, s=0, d=4, u/2)

REAL SYSTEM EXPERIMENTS

- TetriSched: outperforms YARN in all cases
- Hard & None ≥ Yarn-Hard & Yarn-None

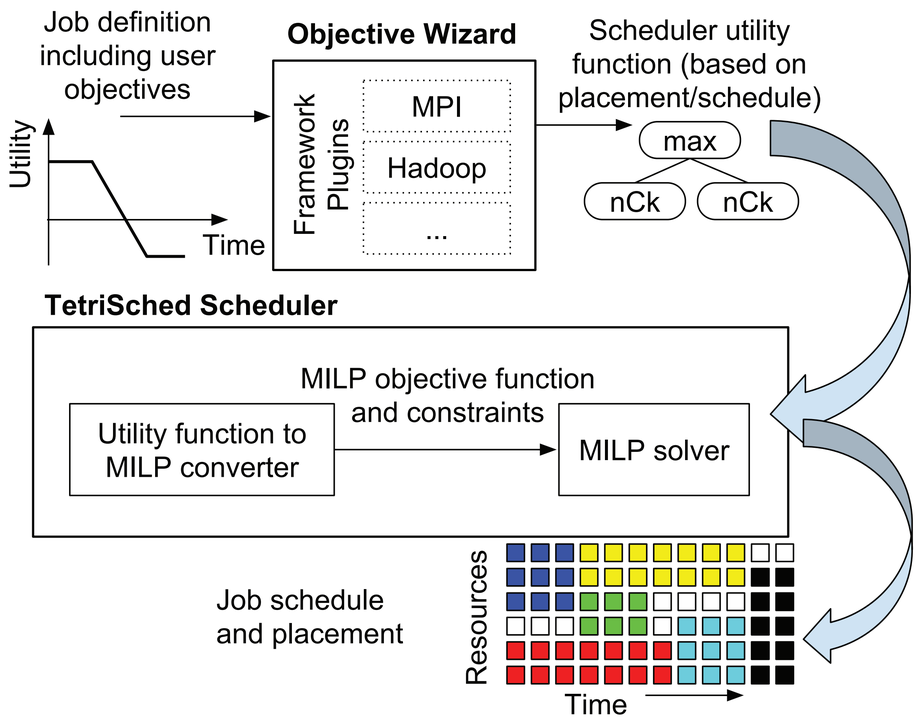


FLEXIBLE SPACE-TIME PLACEMENT



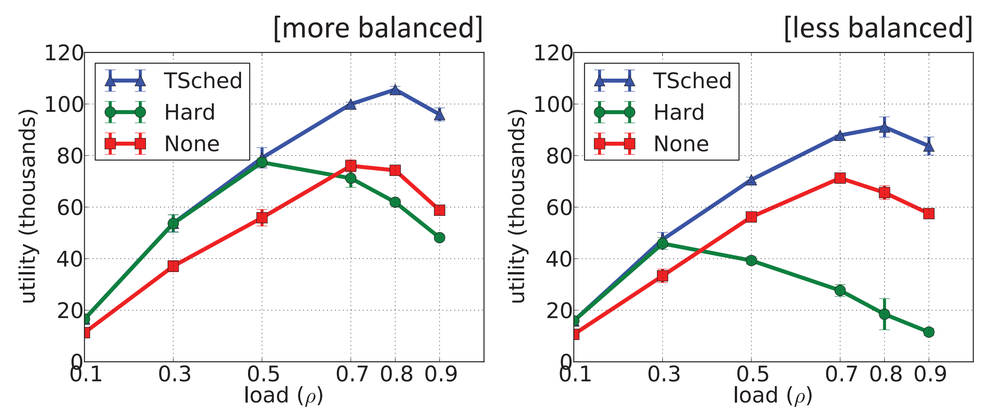
- GPU: run 2 tasks on GPU nodes (rack1) if possible
- MPI: colocate 2 tasks on the same rack and complete ASAP
- Availability: place 2 tasks, each on a different rack

TETRISCHED SYSTEM MODEL

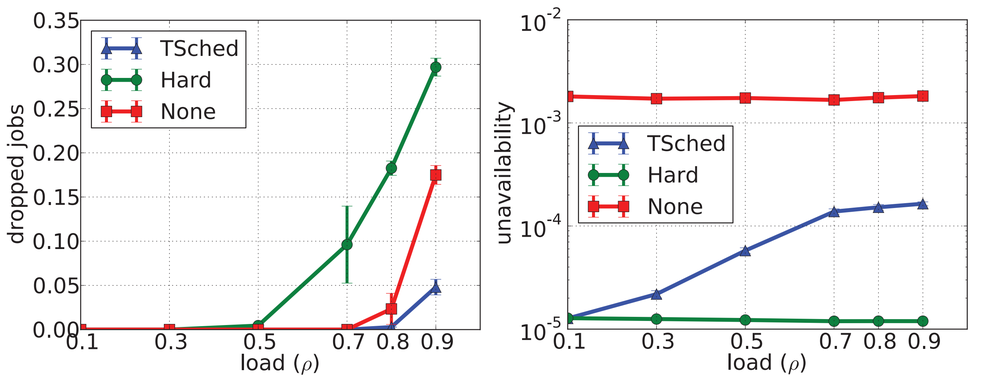


SIMULATION RESULTS

- Flexible placement maximizes utility



- TetriSched exploits tradeoffs under the hood
  - More jobs meet completion time SLO



Intel Science & Technology Center for Cloud Computing
NSERC CRSNG
Carnegie Mellon University
Georgia Tech
intel
PRINCETON UNIVERSITY
UC Berkeley
UNIVERSITY of WASHINGTON