Tetrisched: Space-Time Scheduling for Heterogeneous Datacenters
Alexey Tumanov*, Timothy Zhu*, Michael A. Kozuch**, Mor Harchol-Balter*, Greg Ganger*
*Carnegie Mellon University **Intel Labs

PROBLEM STATEMENT

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

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

- Job definition including user objectives
- Objective Wizard
- Utility function, based on placement/schedule
- Tetrisched Scheduler
- MILP objective function and constraints
- MILP solver

UTILITY FUNCTIONS

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

SIMULATION RESULTS

- Flexible placement maximizes utility
- Tetrisched exploits tradeoffs under the hood
- More jobs meet completion time SLO
- Plan-ahead helps (up to 3x), esp. when bursty

REFERENCES