MOTIVATION

- Memory is a key shared resource in CMPs
- Contention for memory access leads to:
  - Degradation in single-thread performance
  - Starvation

Containment

Memory

Contention

How to achieve both system throughput and fairness?

INSIGHT: BEST OF BOTH WORLDS

- Prioritize memory non-intensive threads
- Unfairness caused by memory-intensive being prioritized over each other
- Shuffle threads
- Memory-intensive threads have different vulnerability to interference
- Shuffle asymmetrically

HOW TO CLUSTER THREADS?

1. Sort threads by MPKI (misses per kiloinstruction)
2. Memory bandwidth usage \( \alpha T \) divides clusters

\[ T = \text{Total memory bandwidth usage} \\ \alpha < 10\% \text{ of Cluster Threshold} \]

PRIORITIZATION WITHIN INTENSIVE CLUSTER - I

Case Study: Two intensive threads contending
1. random-access
2. streaming

- random-access thread is more easily slowed down
  - Vulnerable to interference
  - Causes less interference to other threads

RESULTS

- FRFCFS
- STFM
- PAR-BS
- ATLAS

How to quantify difference between threads?

- Proposed metric: Niceness

- Bank-level parallelism
- Row-buffer locality

- Vulnerability to interference
- Causes interference

Shuffle priorities in a niceness-aware, asymmetric manner