



Fine-Grained Decay-Based Capacity Management For Shared CMP Caches

Carole-Jean Wu and Margaret Martonosi, Princeton University

Design Technologies for
Concurrent Systems, 1.1.2.5



GSRC Annual
Symposium

Sep 3-4, 2009

Abstract

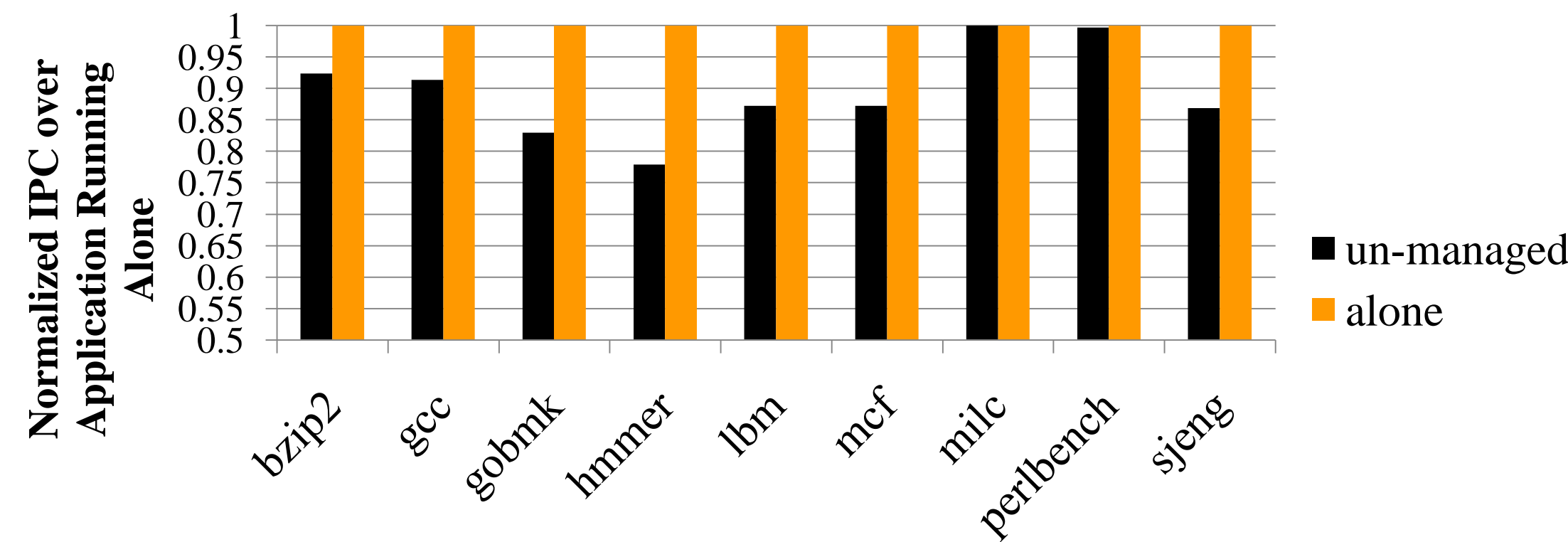
In chip multiprocessors (CMPs), several high-performance cores typically compete for capacity in a shared last-level cache. This causes degraded and unpredictable memory hierarchy performance for multiprogrammed and parallel workloads. In response, recent schemes apportion cache bandwidth and capacity in ways that offer better performance predictability and better control of how the process's OS priority level influences its memory hierarchy performance. These schemes focus, however, primarily on relatively coarse-grained management.

This work explores capacity management approaches that are both temporally and spatially more fine-grained than prior work. We propose a capacity management mechanism based on timekeeping techniques that track the time interval since the last access to cached data. This Adaptive Decay-Based Replacement (ADR) scheme maintains aggregate cache occupancies that reflect the priority and footprint of each application. We evaluate our ideas using a full-system simulator and multiprogrammed workloads of both sequential and parallel applications. To the best of our knowledge, this is the first detailed study of shared cache capacity management considering thread behaviors in parallel applications. ADR outperforms a baseline system by as much as 2.25X and by an average of 1.33X. ADR's fine-grained temporal control is particularly important for parallel applications, which are expected to be increasingly prevalent in years to come. Overall, ADR is promising both as a capacity management scheme on its own, as well as in concert with other previously-proposed bandwidth management schemes.



Motivation

- Core counts scaling up \rightarrow shared cache becomes highly contested.
- Common-used LRU is oblivious to process priority or application memory sensitivity.

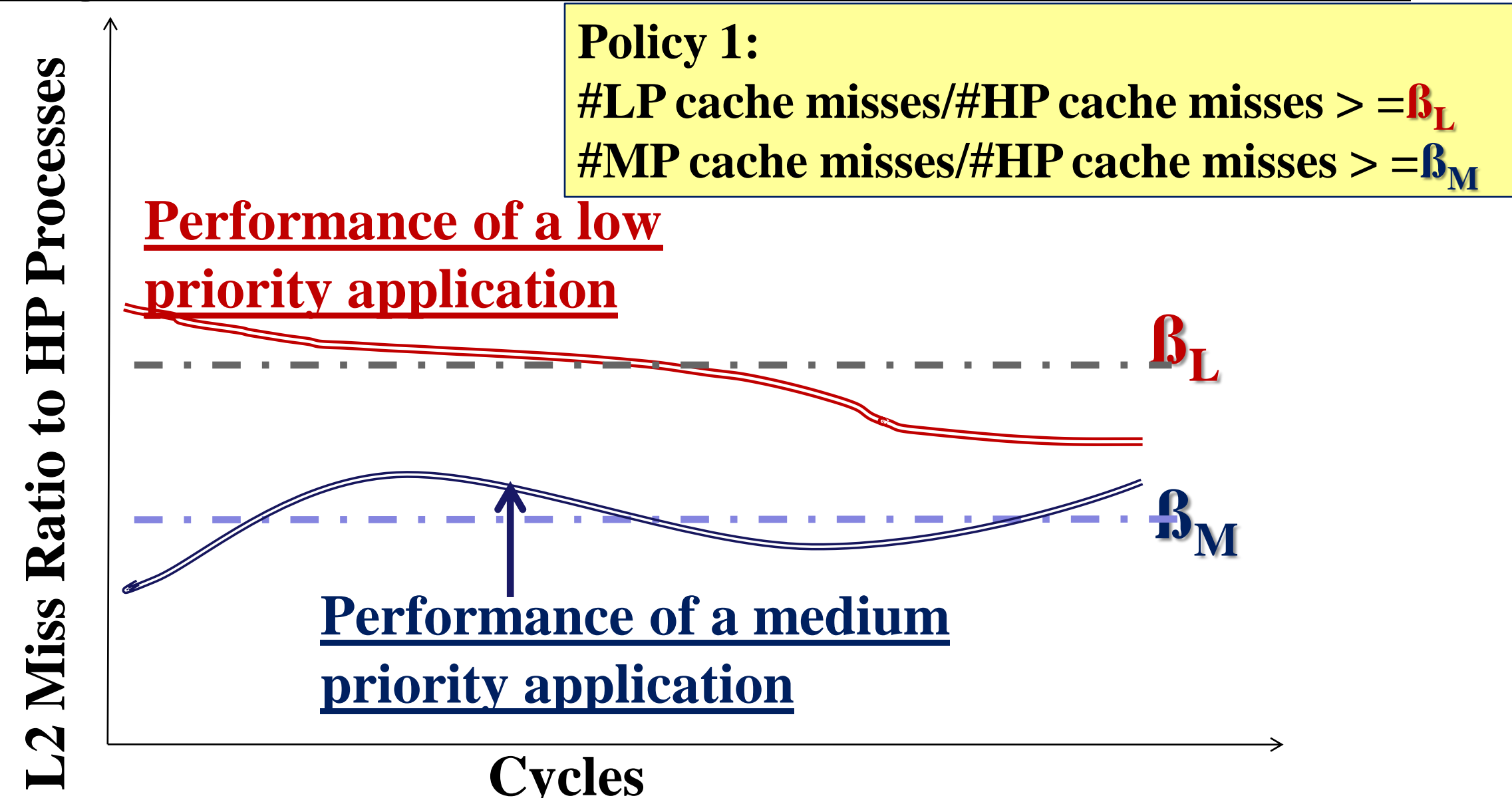


Multiple heterogeneous applications running simultaneously results in

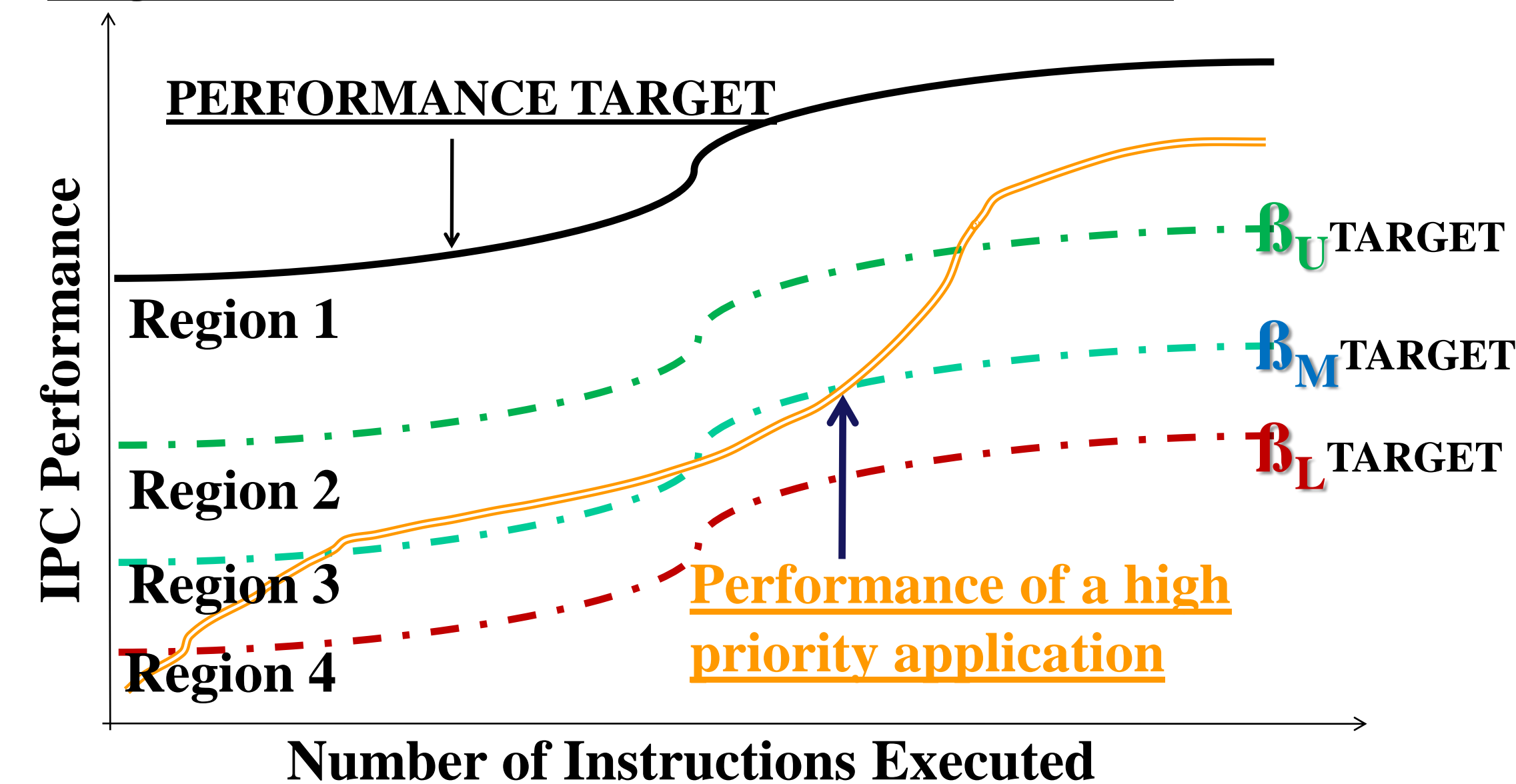
- Performance degradation,
- Performance unpredictability,
- Lack of performance isolation and quality of service.

ADR Algorithm

Region Classification with a Cache Resource Metric



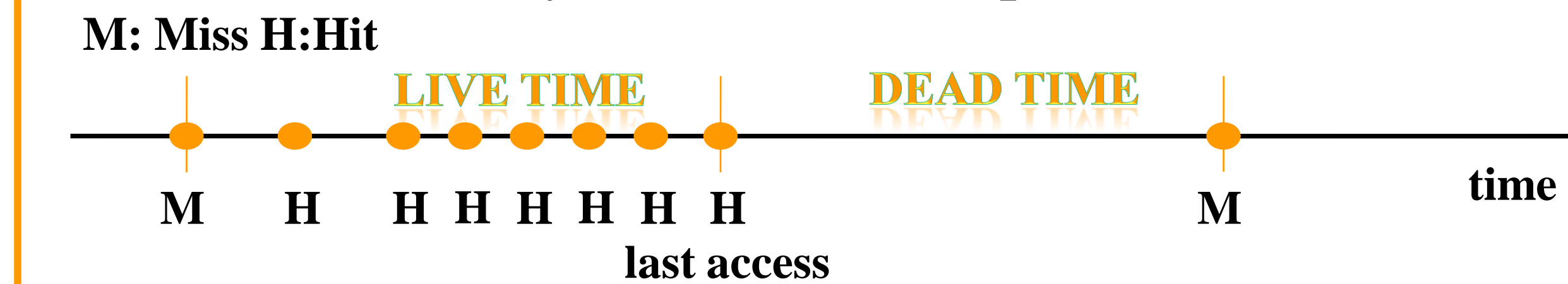
Region Classification with an IPC Metric



Time Keeping Techniques and Cache Replacement

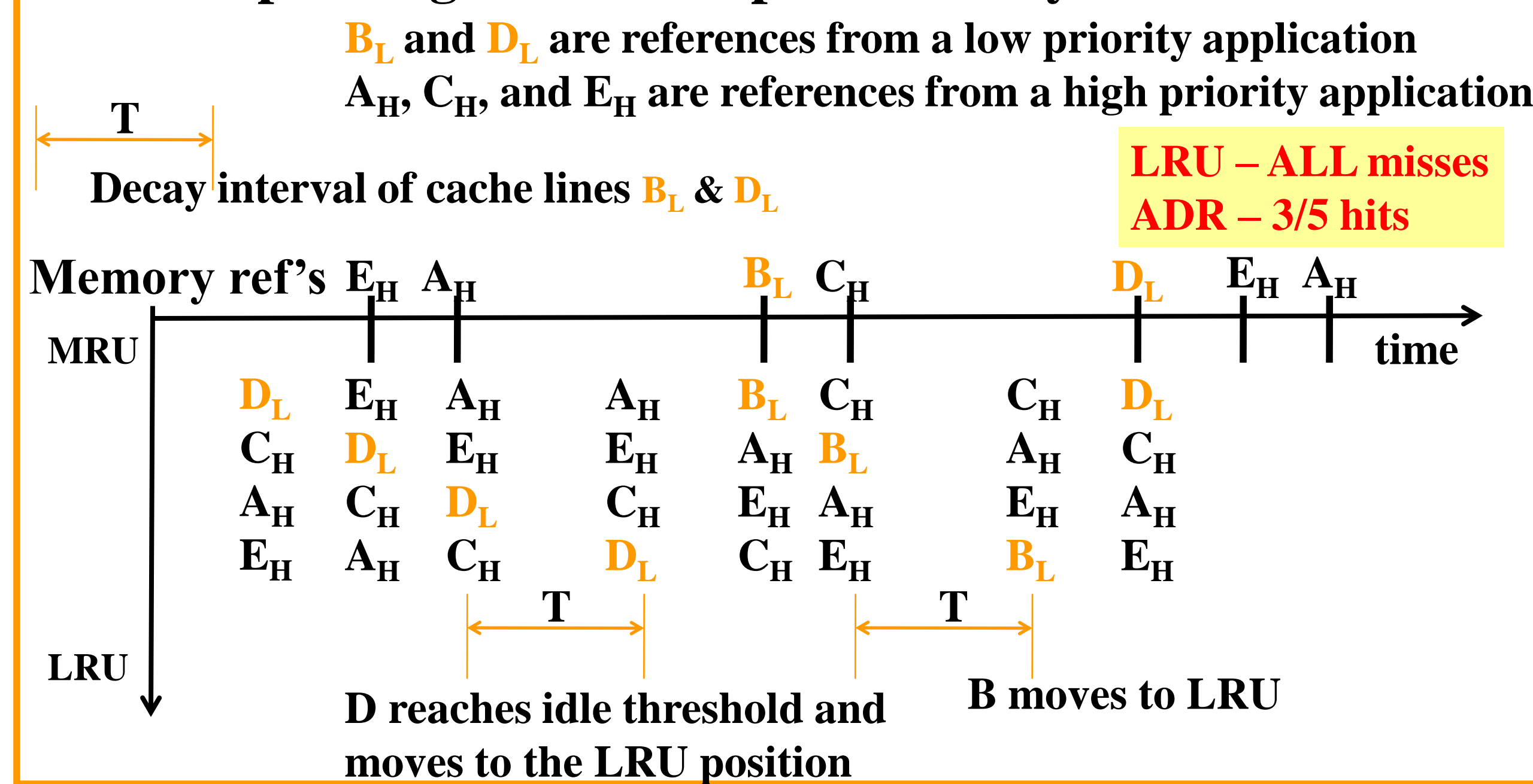
Cache decay

- Cache lines "decay" after a certain period of inactive time



Shared cache capacity management

- Decay counters reach 0 \rightarrow cache lines becomes the immediate candidate for replacement (even if not LRU)
 - Set decay intervals on per-process basis
 - Long decay intervals for high priority processes
 - Short decay intervals for low priority processes
- Vary decay intervals adaptively: **Adaptive Decay-based Replacement (ADR)**
- ADR employs the aspect of priority-based replacement while responding to data temporal locality!



Experimental Setup

Simulation Framework

- Full-system simulation (GEMS): 16 in-order cores, each has 32KB private L1 cache, and all share the 4MB L2 cache.

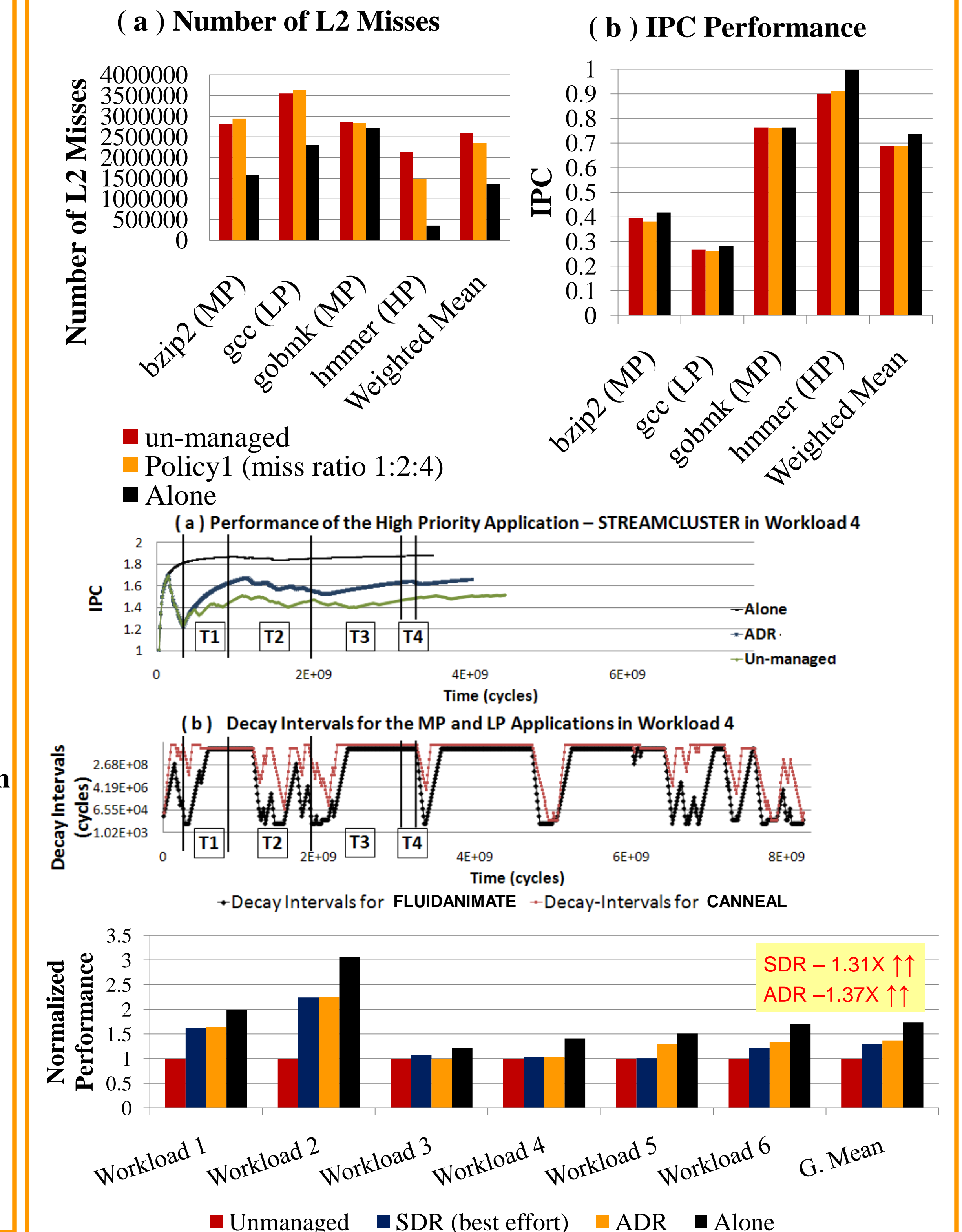
Workload Construction

- Workloads 1 & 2 – general, sequential
- Workloads 3 & 4 – general, parallel
- Workloads 5 & 6 – high contention, sequential

Acknowledgement

The authors acknowledge the support of the Gigascale Systems Research Center, one of five research centers funded under the Focus Center Research Program, a Semiconductor Research Corporation program. In addition, this work was supported in part by National Science Foundation grant #CNS-0627650.

Evaluation



Conclusion

- ADR outperforms a baseline system by **2.25X** and **1.37X** on average for all workloads.
- ADR speeds up critical threads in **parallel applications** by **8%** resulting in better overall application performance because of ADR's fine granularity control of
 - space, shared cache, and
 - time, data temporal locality.
- ADR is promising both as a stand alone capacity management scheme, as well as in concert with other resource management schemes.