Rank Idle Time Prediction Driven
Last-Level Cache Writeback

Zhe Wang, Samira M. Khan, Daniel A. Jiménez

Computer Science Department
University of Texas at San Antonio
Memory Latency is Performance Bottleneck

- **Memory wall**
  - Microprocessor is faster than memory

- **System performance is sensitive to memory read latency**

- **Write-Induced Interference** [Lee et al. 2010]
  - Writes can delay the service of reads, degrade performance
Write-Induced Interference

Service of write requests delay the service of following read requests, thus causing performance degradation.

108 processor cycles
Without write-induced interference, system performance improves 23% on average
Traditional Writeback

- Dirty cache blocks are sent to write buffer when evicted

- The problem
  - Clustering memory traffic: bursty reads with evicted writes
  - Writeback inefficiency: small size of write buffer
Contributions of This Paper

• Propose a technique that services write requests at the point that minimizes the delay caused to the following read requests

• Propose a low-overhead rank idle time predictor to predict long periods of idle time in memory ranks

• Propose a LLC writeback management policy that intelligently writes back bank-level parallelism writes during the long rank idle period
  - Balance the memory bandwidth
  - Isolate the service of memory read and write requests as much as possible
Outline

• Introduction
• Motivation
• Rank Idle Time Prediction Driven Last-Level Cache Writeback Technique.
  - System Structure
  - Rank Idle Time Predictor
• Evaluation
• Conclusion
Reducing Write-Induced Interference

• When to service write requests
  - Memory write requests should be serviced at the time that have minimal interference with read requests

• How to schedule write requests
  - Schedule high locality write requests
  - Large write scheduling space
Related Work: LLC Writeback Technique

• **Eager Writeback** [Lee et al. 2000]
  - Memory scheduling spaced is limited by the write buffer
  - Has no knowledge about how long the rank idle period will be last

• **Virtual Write Queue** [Stuecheli et al. 2010]
  - Requires specific memory address mapping scheme
  - Has no knowledge about how long the rank idle period will be last
Quantifying Rank Idle Time

Ranks are Idle 38% of the time on average
Rank Idle Time Prediction Driven LLC Writeback

Insight: Allow writes to be serviced during long rank idle periods

- Use a predictor to predict long rank idle period once a rank starts to become idle

- Scheduled write requests are generated from LLC and sent to DRAM for service during the predicted long rank idle period
  - Distribute the write requests into long rank idle period
  - Isolate the service of memory read and write requests as much as possible
System Structure

PC of LLC miss → Rank Idle Time Predictor → Rank is Idle

Long rank idle time → Cache Cleaner

DRAM

Write Buffer

Bank-Level Parallelism

LLC

MRU ← LRU

dirty bit
Based on the observation that if an instruction PC leads to long rank idle period, then there is a high probability that the next time this instruction is reached it will also lead to a long rank idle period.
Rank Idle Time Predictor

Rank is idle

T1

m

T2

m

T3

Long rank idle time

PC of Last LLC miss

Rank Idle Cycle Counter

2-Bit Counter

First Level Predictor

PC of Last LLC miss

Long rank idle time (300 CPU cycles)

2-Bit Counter

Second Level Predictor

Cache Cleaner

Rank Idle Time Predictor
Evaluation Methodology

• Simulator
  - MARSSx86 [Patel et al. 2011] + DRAMSim2 [Rosenfeld et al. 2011]

• Execution Core
  - out-of-order, 8-core processor

• Caches
  - 64KB L1 I + D caches, 2-cycle
  - 16MB 16 way set associative LLC, 14-cycle

• DRAM System
  - DDR3 1600MHZ
  - 2 channels, 2/4 rank per channel, 8 banks per rank

• CMP Workloads
  - SPEC CPU2006 benchmarks
  - Six mixes of SPEC CPU2006 benchmarks for 8-core processor
Performance Evaluation

It improves performance of eight benchmarks by at least 10% and delivers an average speedup of 10.5% with two-rank configuration and 10.1% with four-rank configuration.
Prediction Evaluation

False positive rates for the first-level and second-level predictors are 8.5% and 14.7% on average.
Read Latency Evaluation

The technique reduces the read latency on average by 12.7% with two-rank configuration and 14.8% with four-rank configuration.
### Storage Overhead

<table>
<thead>
<tr>
<th></th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>Two-level rank idle time predictor</td>
<td>4KB=2bits * 8096entries*2</td>
</tr>
<tr>
<td>Cache Cleaner</td>
<td>2K bytes</td>
</tr>
<tr>
<td>Total</td>
<td>18KB for 2-rank / 34 KB for 4-rank</td>
</tr>
<tr>
<td>Percentage of LLC Capacity</td>
<td>~0.3%</td>
</tr>
</tbody>
</table>
Conclusion

- Write-induced interference causes significant performance degradation.
- Proposed a rank idle time predictor that predicts the long rank idle time.
- Proposed a LLC writeback management policy that intelligently writes back bank-level parallelism writes during the long rank idle period
  - Balance the memory bandwidth
  - Isolate the service of memory read and write requests as much as possible
Thank You!

Question?