Inter-Thread Reuse Distance Based LLC Replacement for Effective Helper Threaded Data Prefeteching on CMPs

Min Cai

School of Computer Science and Technology
Beijing Institute of Technology

June 11, 2011

Outline

Thoughts on HT Induced Cache Pollution

- Helper threaded data prefetching on CMPs
  - is enabled by the presence of shared last level cache (LLC)
  - uses helper thread (HT) to bring data later used by main thread (MT)
  - utilizes accurate and timely MT-HT inter-thread LLC hits
- HT induced cache pollution w.r.t MT performance
  - only happens when HT evicts the data immediately used by MT
  - will not happen when any piece of data that was previously brought by MT but will not be used by MT in future (i.e., with instant intra-MT reuse distance), is evicted by HT, which is typically the case in mst!

Chracterizing Shared LLC Read Misses in mst

- Both MT and HT have a small set of delinquent loads (PCs) that incur most LLC read misses
- Exhibits a thrashing memory access pattern w.r.t. delinquent PCs (thrashing delinquent PCs)
  - once a cache line is brought into LLC by a delinquent PC in MT, it will never be accessed again in the foreseeable future
- Observation 0: in most of the cases, HT prefetches has a good coverage of MT read misses
  - For most LLC read misses from HT, an immediate followup MT reference will access the data and hit in the LLC
  - Then the MT accessed cache line will become DEAD and not accessed agained in the foreseeable future
- Observation 1: Very few HT prefetches are late in that a followup MT reference causes an LLC miss but an LLC MSHR hit

Reuse Distance

- Definition: The stack distance (SD) of an application's memory accesses is the number of UNIQUE memory references between two consecutive accesses to the same cache line
- Property: In a k-way set-associative cache a cache miss with stack distance sd is
  1. sd $<$ k $\Rightarrow$ conflict miss
  2. k$<=$ sd $<$ ∞ $\Rightarrow$ capacity miss
  3. sd $=$ ∞ $\Rightarrow$ cold miss
- Definition: The reuse-distance (RD) of an application's memory accesses is the number of memory references between two consecutive accesses to the same cache line
- Cache allocation ticks (CAT) based reuse-distance of an application's memory accesses is the number of cache replacements between two consecutive accesses to the same cache line
  - CAT time can relate time (events) to space (cache size)

Limitations of Intra-Thread Reuse Distance Prediction Based Cache Replacement

- Recently proposed reuse(rereference/next-use) distance prediction based LLC replacement policies for CMPs have achieved good results for sequential workloads without helper threaded data prefetching enabled
  - Which are good online approximations of Belady's offline optimal cache replacement policy (i.e., evict the cache line that will be accessed in the furtherest future)
- However, the distant intra-thread reuse distance of loads issued from thrashing delinquent PCs
  - Renders reuse distance prediction based LLC replacement useless for programs such as mst

Related Work

- CII miss classification (M. Dubois, etc. "The detection and elimination of useless misses in multiprocessors" In ISCA'1993)
  - Cold Misses
  - Intra-Processor Misses and Inter-Processor Misses
- Novel cache Replacement Policies proposed in recent years
  - MLP awareness (ISCA 06)
  - Utility/marginal gain awareness (Utility based partitioning)
  - Thread awareness (TADIP)
  - Probability (a few..)
  - Reuse distance / rereference distance / next-use distance prediction (ICCD 2007, HiPC 2008, ISCA 2010, ICS 2010 from Zhejiang Univ.)
- Techniques for Reducing Inter-Core Misses
  - Set Pinning (Shekhar Srikantaiah, etc. "Adaptive Set Pinning: Managing Shared Caches in Chip Multiprocessors" in ASPLOS'2008)
  - Block Pinning (Rakesh Kumar, etc. "Adaptive Block Pinning for Multi-core Architectures " in HiPC'2008)

Inter-Thread Reuse Distance

- Fortunately, for the helper threaded data prefetching scheme that we adopted for accelerating mst, a new metric called inter-thread reuse distance can be used to monitor the quality of HT and help synchronize HT with MT
  1. Low HT-MT inter-thread reuse distance is an indicator of good performance of the HT scheme
  2. Medium HT-MT inter-thread reuse distance signals those potentially late HT prefetches that, depend on the LLC replacement policy, may be useful or not for MT
  3. High HT-MT inter-thread reuse distance implies HT and MT is not synchronized caused by either little synchronization or the work in HT and MT is not balanced

Inter-Thread Reuse Distance Based HT Quality Assessment

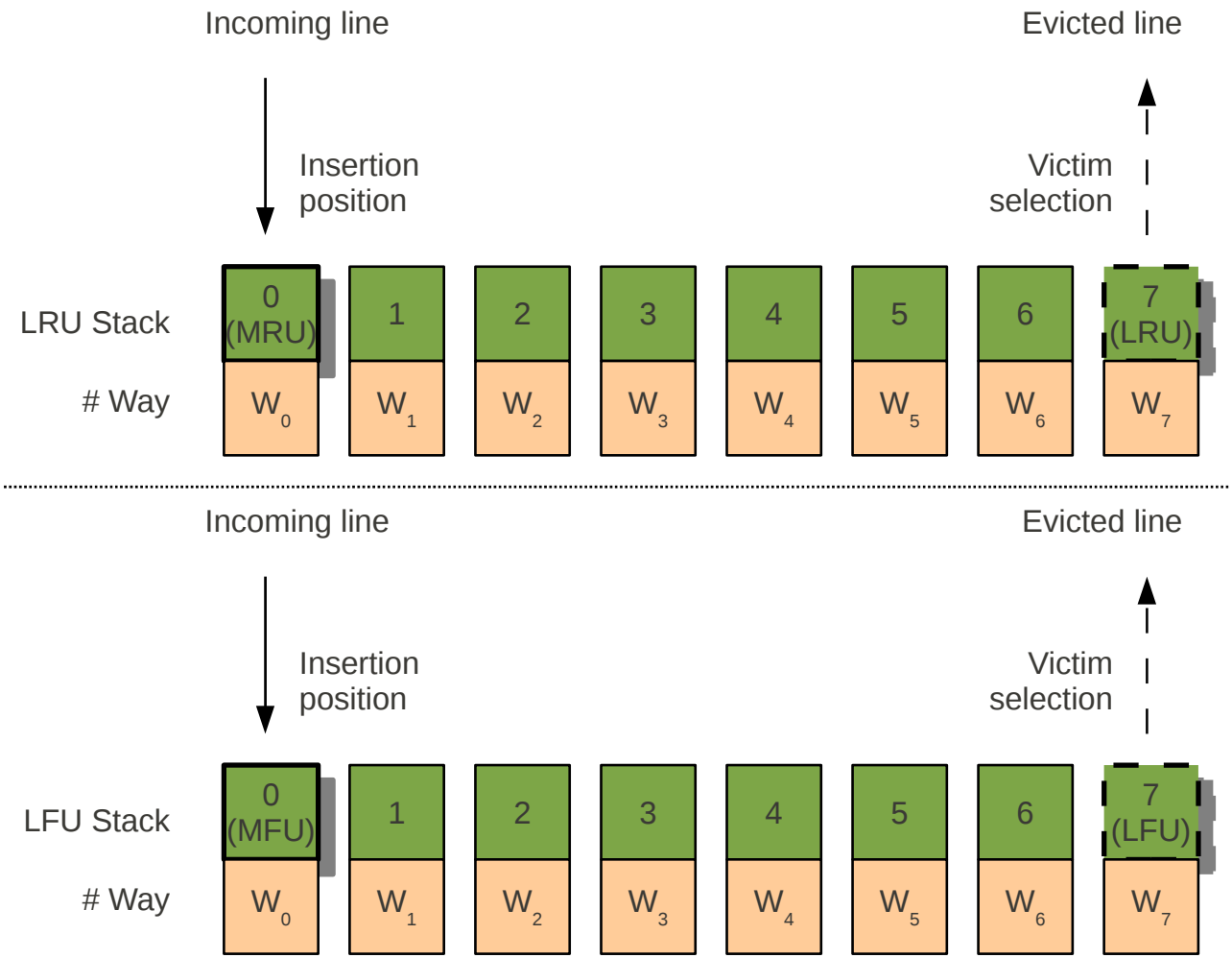
Under the LRU LLC replacement policy, for LLC misses that issued from delinquent PCs in either in MT or HT, the value of HT-MT inter-thread reuse distance is (w.rt. the associativity of the cache)

- HT-MT ITRD $>= 0$
  - ITSD $>$ assoc $\Rightarrow$ replaced HT prefetches (need aggressiveness tuning)
  - ITSD $<=$ assoc
    - Cache hit $\Rightarrow$ useful HT prefetches
    - MSHR hit $\Rightarrow$ late$_1$ HT prefetches (need aggressiveness tuning)
- HT-MT ITRD <0
  - Low $\Rightarrow$ late$_1$ HT prefetches (need aggressiveness tuning)
  - Medium to High $\Rightarrow$ late$_2$ HT prefetches (need synchronization)
  - Distant $\Rightarrow$ MT loads not covered by HT prefetches

Inter-Thread Reuse Distance (ITRD) Prediction

- For each program phase separated by two consecutive synchronizations, in the end of the program phase
  - ITRD = 1/2 * previous_ITRD + 1/2 * ITRD_in_the_current_phase
- A confidence counter with a threshold value of 10 and maximum value of 20
- Main components, similar to the one used in the reuse distance prediction based cache replacement paper
  - Data address indexed FIFO backed sampler
  - PC indexed LRU cache backed predictor (has the same geometry as L2)

Sidebar: Concept of Priority Stack in Cache Replacement



Inter-Thread Reuse Distance Based LLC Replacement for the HT scheme

- Extra hardware: an additional HT bit per L2 line
  - Each cache line has an additional HT bit indicating the line is brought by HT and has not been used by MT yet
  - When a cache line marked by HT is referenced by MT, it is unmarked
- Insertion position of an L2 read miss issued from delinquent PCs and marked as HT, if its predicted HT-MT reuse distance is
  1. high $\Rightarrow$ just bypass the L2 or LRU position
  2. medium $\Rightarrow$ middle position or 1/4 LRU position
  3. low $\Rightarrow$ MRU position
  4. cannot be determined $\Rightarrow$ LRU position
- Insertion position of an L2 read miss issued from delinquent PCs and not marked as HT: LRU position
- Insertion position of an L2 read miss in remaining cases: MRU position

Inter-Thread Reuse Distance Based feedback directed synchronization triggering and aggressiveness adjustment for the HT scheme

- Hardware triggered synchronization of HT with MT
  - Based on the already implemented pseudocall mechanism that pass parameters between the program and the processor in some unused registers by the ADDU MIPS instruction with predefined special parameters
  - A register based polling process is used to let the HT periodically ask the processor if it need to synchronize with MT
  - Tackling the B parameter in K-P-B model
  - Can potentially utilize the synchronization register mecahnisms proposed in existing papers to reduce synchronization overhead
- Dynamic HT aggressivenss adjustment
  - Parameters for adjusting the aggressiveness of the HT are
    - Lookahead (a.k.a. the K parameter in K-P-B model)
    - Prefetching work within one program phase (a.k.a. the P parameter in K-P-B model)

Results Obtained So Far: A Simple Thrashing/HT-Sensitive Augmented LRU LLC Replacement Policy

- 2 X 2 CMP; two level cache hierarchy with MESI coherence between private L1s; 4MB 8-way associative shared L2; fixed DRAM latency = 200 cycles; mst 1000, run to end detailed simulation
- Thrashing/HT-sensitive augmented LRU LLC replacement algorithm: <see demo>
- Speedup of (HT + enhanced LRU) vs. (HT + LRU): 1.011

version	LLC repl	c0t0.llcReadMisses	c1t0.llcReadMisses	cycles
baseline	LRU	1032780	N/A	605676479
HT	LRU	460884	569665	501525524
HT	enhanced LRU	437226	548200	496180137

Conclusion

- Intra-Thread Reuse Distance and its prediction based LLC replacement is not helperful for improving the HT performance of workloads that contain thrashing delinquent PCs
- Inter-Thread Reuse Distance is a good metric for evaluating the effectiveness of the HT scheme
- Inter-Thread Reuse Distance based feedback-directed dynamic HT synchronization triggering and parameter tuning is a viable solution for improving the effectiveness of the HT scheme
- Further work
  - inter-thread reuse distance measurement and prediction
  - inter-thread reuse distance based feedback directed HT scheme