Making TLB Replacement Better—CHiRP: Control-Flow History Reuse Prediction

Samira Mirbagher-Ajorpaz
University of California - San Diego

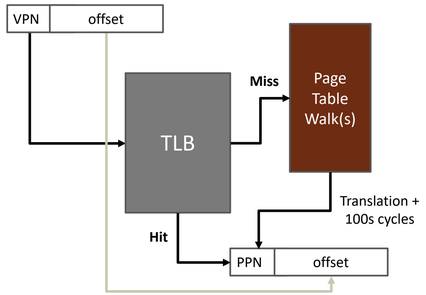
Elba Garza
Texas A&M University
elba@tamu.edu

Gilles Pokam
Intel Labs

Daniel A. Jiménez
Texas A&M University

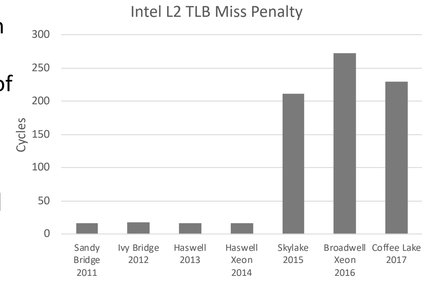
Translation Lookaside Buffers (TLB)

- TLBs lie on critical path to accessing memory
  - **On hit**: Quick virtual-to-physical address translation
  - **On miss**: Costly page table walk, multiple memory accesses (possibly 100's of extra cycles!)
- Limited in size (& reach) due to power, timing, area constraints



- New workloads (e.g. server) put tremendous stress on critical path to memory
- TLB miss overhead: 5.5% to 19% of total execution time[1]
- Improved by L1 and L2 TLBs and MMU caches

Must make best use of small size; keep only useful page translations



Goal: Improve TLB performance without necessitating increased TLB sizes, or extending hierarchies

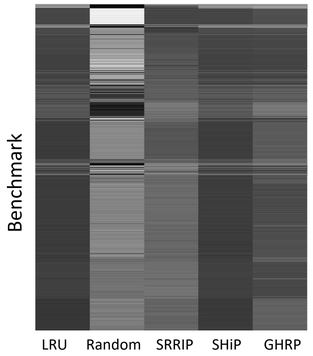
- Least-Recently-Used (LRU) replacement policy is ubiquitous for TLBs
- TLB access patterns similar to caches', albeit at larger granularity
- TLBs are tagged, set-associative SRAM arrays, like caches…

Apply prior predictive replacement policies?

Exploring Existing Replacement Policies

- Consider current TLB replacement policies & cache-based:
  - LRU
  - Random
  - Static Re-reference Interval Prediction (SRRIP[2])
  - Signature-Based Hit Prediction (SHiP[3])
  - Global History Reuse Prediction (GHRP[4])
- Focus on L2 TLB, accounts for most cycles in miss handler[5]

Problem: Not very effective for TLB replacement!



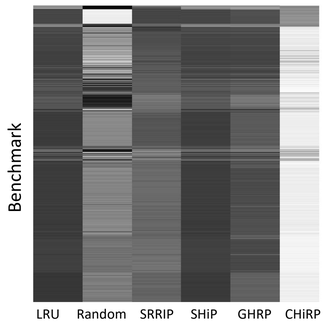
Observations:

1. Replacement algorithm inaccuracies due to conflicts **within** sets, not among sets.
2. Coarse-grained nature of TLBs results in increased aliasing.
3. TLB reuse prediction does not benefit from large global histories. However, branch path history helps.

These observations lead to...

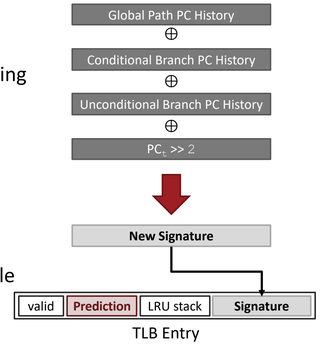
CHiRP: Control-Flow History Reuse Prediction

- Indexes prediction table using a history signature specially designed to correlate with TLB behavior
- Reduces L2 TLB misses by **28.21% over LRU** with 1KB hardware overhead off the critical path
- Yields geometric mean speedup of **4.8% over LRU**



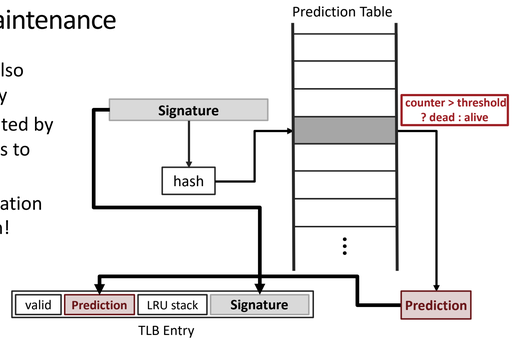
CHiRP Signature Generation

- Signature is constructed by combining three control-flow histories with shifted current-time (t) PC:
  - Global Path PC History
  - Conditional Branch PC History
  - Unconditional Branch PC History
- Stored in associated TLB Entry
- Used to index CHiRP prediction table

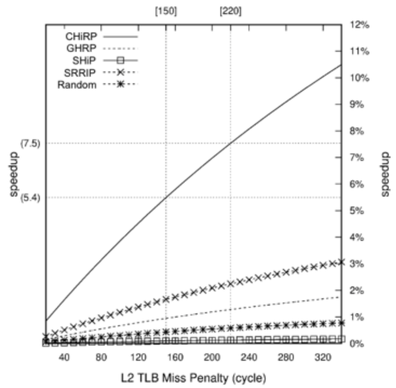


CHiRP Data Maintenance

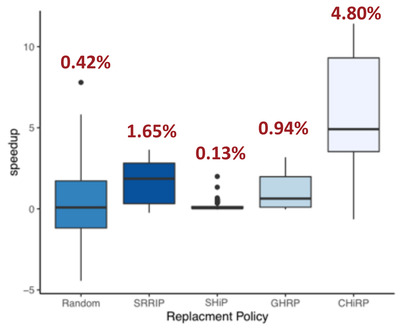
- Reuse Prediction also stored in TLB entry
- Predictions computed by the previous access to table entry
- Prediction computation not on critical path!



Speedup vs. Miss Penalty



Speedup



[1] Peng et al. 2006
[2] Jaleel *et al.* 2010
[3] Wu *et al.* 2012
[4] Mirbagher *et al.* 2018
[5] Basu *et al.* 2013