CSR: Small: Effective Sampling-Based Miss Ratio Curves – Theory and Practices

Zhenlin Wang, Michigan Technological University

- Develop a new cache model based on reuse time distribution and a novel concept of average eviction time (AET) to construct miss ratio curves effectively [1]. Focus on in-depth study of the hypotheses behind the model and develop theoretical foundation for sampling.
- Conduct a systematic comparison of recent cache models with respect to their assumptions of access distribution: the reuse distance-based models such as SHARDS and Counter Stacks, and the reuse time-based models: the footprint theory, Statstack, and AET [2].
- Study theory and practice for hardware cache partitioning. Develop effective online miss ratio curve approaches that exploit the recent Intel Cache Allocation Technology (CAT) [3].

**Objectives**

- Investigate theory and practice for hi-page sets and exploit hi-page pages to reduce TLB pressure. The two classes of pages, regular or hi-page, introduce a new challenge to the AET model with nonuniform miss penalties and block granularities. We propose a study on composability of AET-based MRCs and examine the impact of different page sizes and mis penalties [4].
- Research theory and practice for key-value memory cache management and its interaction with hypervisor-level dynamic memory management [5].

**References and Publications**


**Miss Ratio Curve (MRC) using Average Eviction Time (AET)**

**AET Process:**
1. Find reuse time distribution through sampling
   - For all t, find P(t), the probability for an access with a reuse time greater than t
2. AET Model: Given P(t), find average eviction time of cache size c
3. Miss ratio curve: given cache size c, find miss ratio

**AET shows comparative prediction accuracy when compared to SHARDS [6], using the MSR traces by Microsoft Research Cambridge**

- SHARDS is a reuse distance-based model
  - Applies spatial sampling to reduce costs
  - Uses approximation w.r.t. sampling rate to estimate reuse distances
- AET is a reuse time-based model
  - Explores a variety of sampling
  - Needs to model the relationship between reuse time and reuse distance

**AET model is composable:**
- With reuse-time distribution of each individual program or trace, AET can model the co-run MRC.

The graph on the right shows the MRC of four MSR traces and the co-run MRC.

**Summary:**
AET shows advantages in both time and space complexities

**Cache Allocation Through Partial Sharing (CAPS)**

**AET-Based Working Set Size (WSS) Prediction for Virtual Machines**

**Apply AET to predict WSS of a virtual machine:**
- To track a page access, set a reserved bit to trap into the hypervisor
- Load balance between virtual machines
  - Sampling (random or spatial)
  - Hot set: only trap/model cold pages

**Sampling rate control:**
- Empirically, one over a million soft page faults yields acceptable overhead
- Need to track a sufficient number of cold pages
- Balance between hot set size (HSS) and sampling rate

**Without hot set, the AET model shows high accuracy with low sampling rate. The result on the right shows 1% sampling rate delivers the same MRC as 100% (a micro-benchmark on Pin). In [1], we show 10^6 sampling rate is acceptable**

**Trapping short reuse times is prohibitive. Hot set filters short reuse times as shown on the right. The RTH of mp in SPEC 2006 beyond the 512-page hot set is accurate with the counts reduced to 1% (sampling rate) of the actual.**

**The two figures compare the predicted WSSs along the execution of mp in SPEC 2006 beyond the 512-page hot set is accurate with the counts reduced to 1% (sampling rate) of the actual.**

**Challenges and ongoing work:**
- Disjoint partitioning (non-overlap) performs better than full sharing by avoiding cache contention
- Partial sharing explores the potential of CAT better by allowing partition overlapping
- The improvement space is significant, such as 11% in average throughput across 75 workloads
- Diverse performance target yields different partitioning schemes
- Partial sharing explores the potential of CAT better by allowing partition overlapping
- The improvement space is significant, such as 11% in average throughput across 75 workloads

**Case study on one workload mix:**
- Different performance target yields different partitioning schemes
- Disjoint partitioning performs better than full sharing by avoiding cache contention

**Simulated annealing algorithm:**
- Probabilistic algorithm to search a large space
- Applies random walk and a probability, AT, to accept a worse state to avoid getting stuck in a local optimum
- Uses MRCs and an iterative cache occupancy model to predict target performance
- Controls search using temperature

**Average performance across 75 workloads:**
- Disjoint partitioning (non-overlap) performs better than full sharing by avoiding cache contention
- Partial sharing explores the potential of CAT better by allowing partition overlapping
- The improvement space is significant, such as 11% in average throughput

**Collaborators**
- Peking University: Yingwei Luo, Xiaolin Wang, Xiumeng Hu, Lan Zhou, Zhigang Wang, Fan Hou, Taowei Luo, Zihui Huang, Yascheng Xiang
- Michigan Tech: Nilifar Onder, Daniel Byrne, Wei Kuang
- University of Rochester: Chen Ding