1. **Shorten** Page Table to **reduce** accesses

2. **Keep** PTEs in the cache for **quicker** walks

Paper
DOI: 10.1145/3503222.3507718

Contact us
chang.hyun.park@it.uu.se

Every Walk's a Hit: Making Page Walks Single-Access Cache Hits

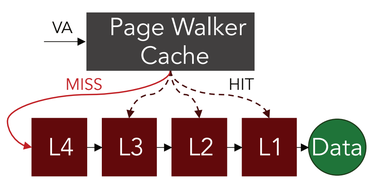
UPPSALA UNIVERSITET

Chang Hyun Park, Ilias Vougioukas, Andreas Sandberg, and David Black-Schaffer

arm Research

1 Background

- TLB reach not scaling with DRAM
- TLB misses resolved with Page table walks
- Page walk caches enable skipping over levels
  - 1.1-2.5 (avg. 1.5) accesses per walk for workloads of up to 8 GB size)

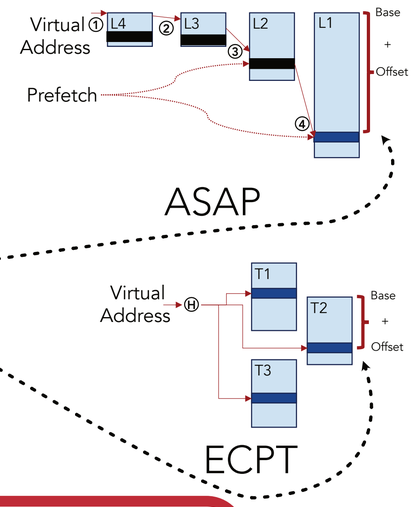


Coverage[1]	#mem access per walk
48 MB	1
4 GB	2
2 TB	3
> 2 TB	4

Larger memory sizes, 5-level page tables → more memory accesses per walk

2 Prior work: require large contiguous memory

- Address Translation with Prefetching[2] (ASAP)
  - Prefetch leaf nodes of page table
- Elastic Cuckoo Page Tables[3] (ECPT)
  - Multi-way hash page tables → concurrent lookups



Prior work **need** large contiguous physical memory allocations

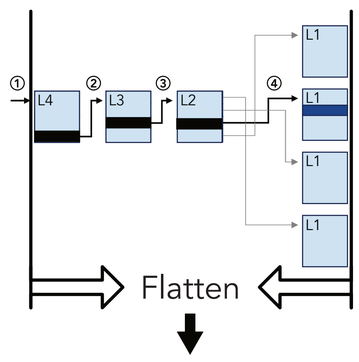
- Page table allocation is a **critical task**
  - Cannot fail, cannot take too long

Difficult to guarantee large contiguous allocations

Our Proposal: Single-access cache hitting walks

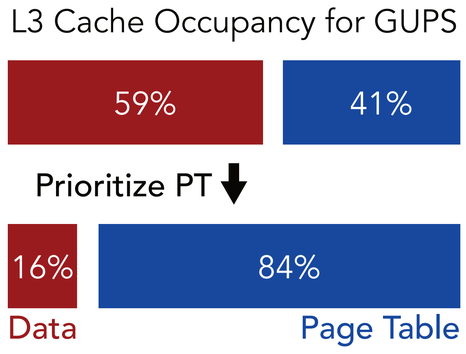
3 Flatten the Page Table to reduce accesses

- Use large pages (2 MB) as page table nodes
  - Leverage existing OS support for large pages
- 2 MB page tables can flatten 2-levels of PT
- Flatten L4+L3 and L2+L1 → 2 memory accesses
  - Page walker cache helps skip L4+L3 → 1 memory access
- Fallback to 4 KB PT when 2 MB not available



Some other possible flattening

4 Prioritize keeping PTE in the cache



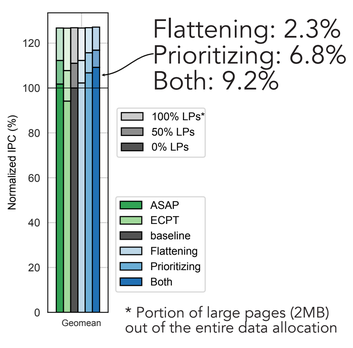
L3 Miss ratio	Data	PT Walk
Normal	99.1%	54.5%
Prioritize PT	99.5%	**14.7%**

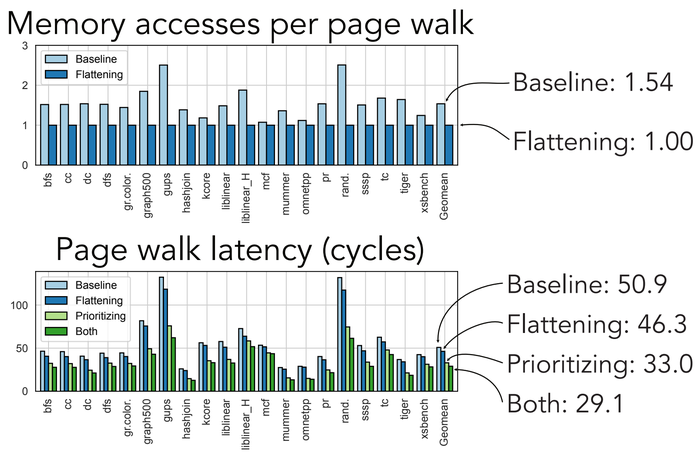
Small increase in data miss,
Massive reduction in PT walk miss

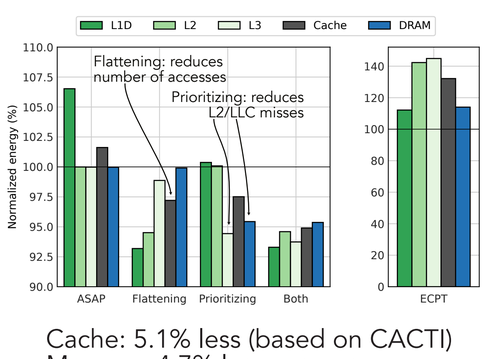
- Workloads with high TLB miss → high data miss
  - Data caching not effective → cache PTE instead
- **Insight:** one PTE represents region of 64 cachelines
  - 64x more likely to hit PTE than individual cachelines
- Simple solution: bias replacement policy
  - Prioritize keeping PTE over data

5 Results

- Gem5 Simulator
- System-call Emulation mode
- Modeled after Intel Skylake uArch







Cache: 5.1% less (based on CACTI)
Memory: 4.7% less accesses

[1] Van Schaik et al., Vrije Universiteit, Technical Report 2017. Coverage for Intel Skylake uArch
[2] Margaritov et al., MICRO '19
[3] Skarlatos et al., ASPLOS '20