Understanding the Performance of Parallel Applications in a Tiered-memory System

Kaiwen Xue <kaiwenx>

Friday, December 15, 2023

Datacenter Status-quo

Applications are memory-intensive. Indications:

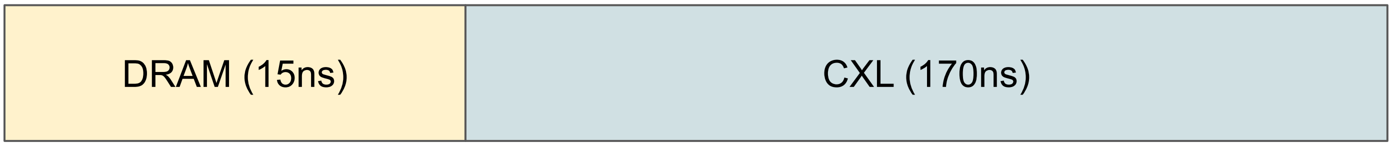
1. High memory cost (up to TB)
2. Larger address translation overhead (due to TLB misses)
3. Parallel applications

Solutions:

1. Tiered-memory (e.g., CXL.mem)
2. Huge pages
3. More cores!

How to Enable Better Solutions?

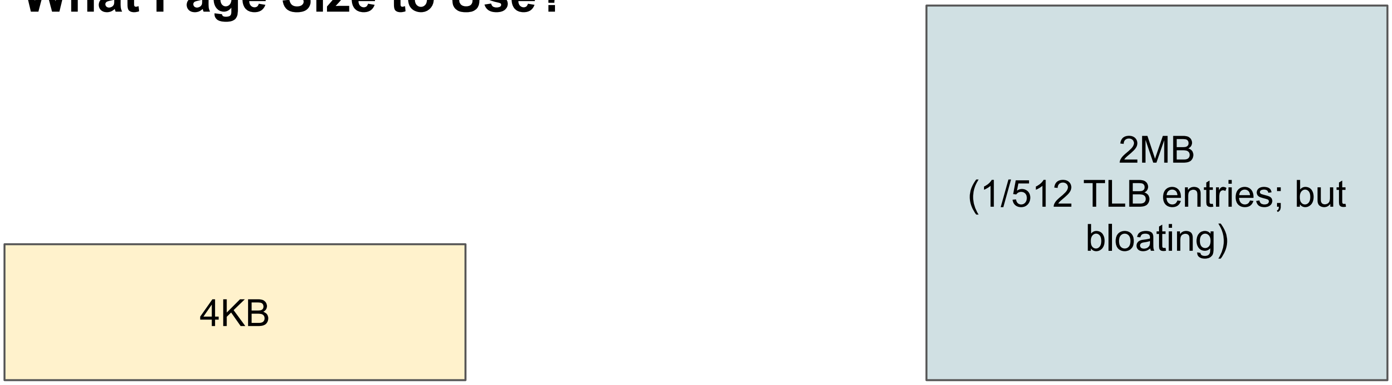
Where to Place?



DRAM (15ns)

CXL (170ns)

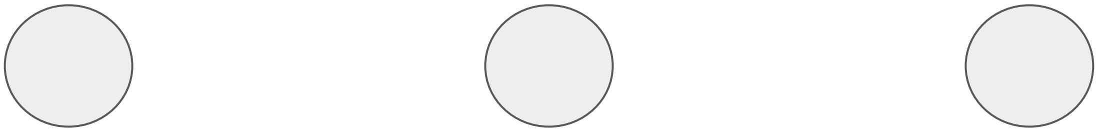
What Page Size to Use?



4KB

2MB
(1/512 TLB entries; but bloating)

What Happens to Multicore?



Question You Can’t Avoid - What is Hot?

Hot pages == Pages that are accessed more

Requirements for tracking hotness:

1. **Tracked in a fine granularity**
   - 64B? 512GB?
2. **Low overhead of tracking**
   - Do you pin a core in order to track them?
3. **Low overhead of decision policy**
   - Do you scan the whole address space (TB)?
4. **Need Per-process hotness to ensure fairness**
   - Place all pages of one process in a slow tier?

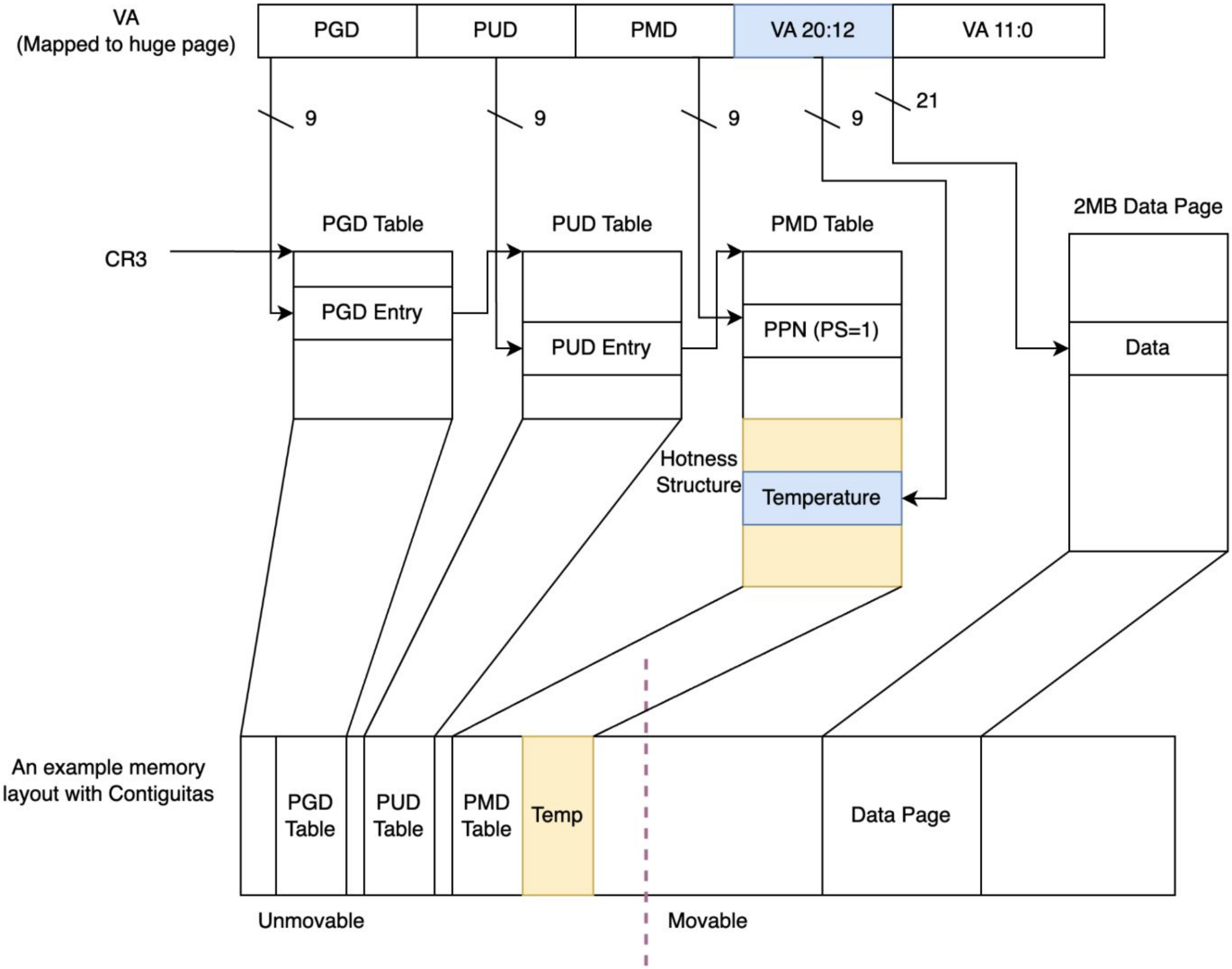
Current Solutions

- TPP (ASPLOS ‘23): Tackles 3, Cannot make 2
- PCC (MICRO ‘23): Tackles 1, 2, 3, Cannot make 4
  - Also cannot scale with memory
- MEMTIS (SOSP ‘23): Tackles 1, Cannot make 2, 4
- Telescope (Arxiv): Tackles 3, Cannot make 1, 2
- HeMem (SOSP ‘21) and MaxMem (Arxiv): Tackles 4, Cannot make 2

How do we make all?

(DEMO: How Simulation Works)

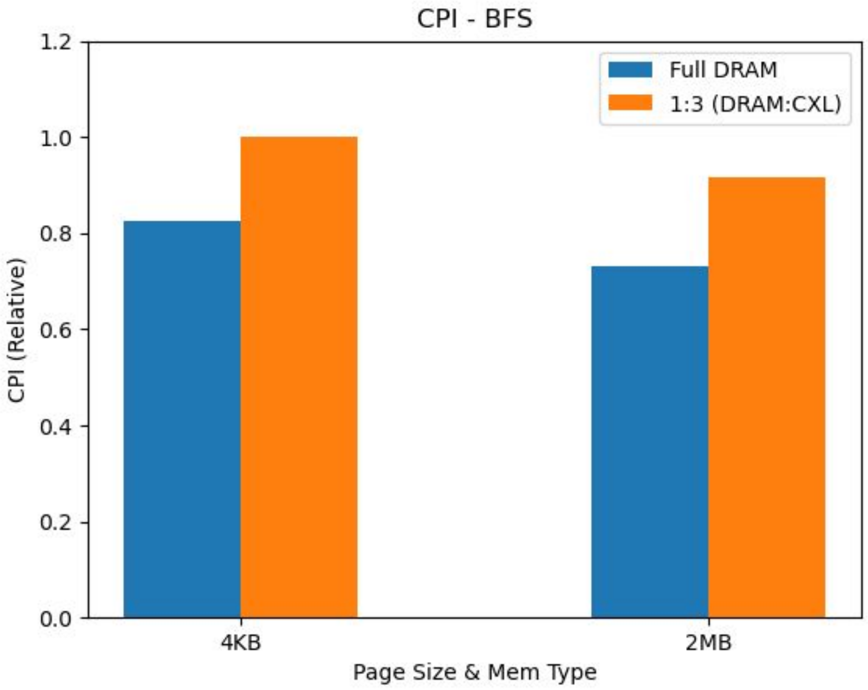
Hotness Walk

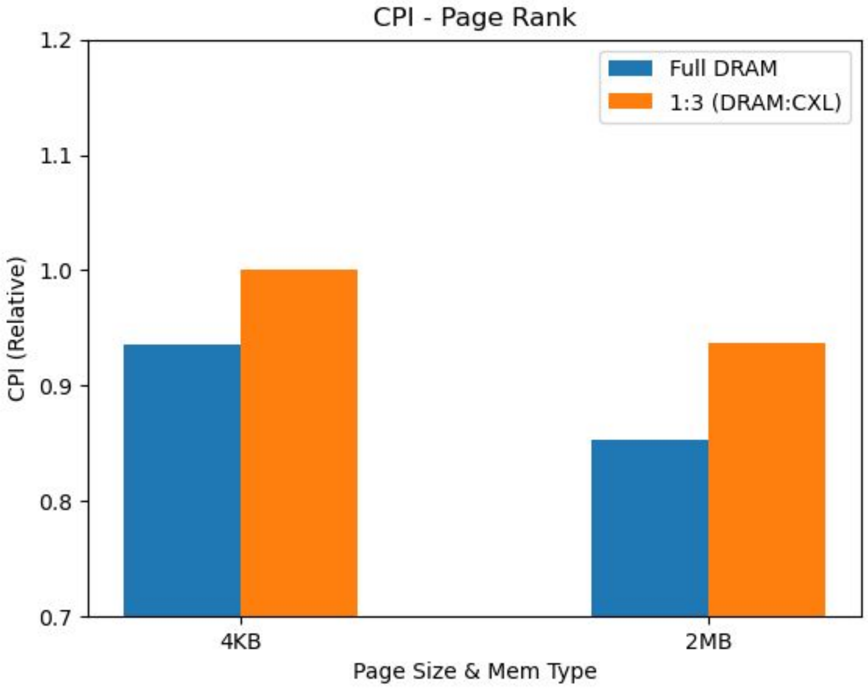


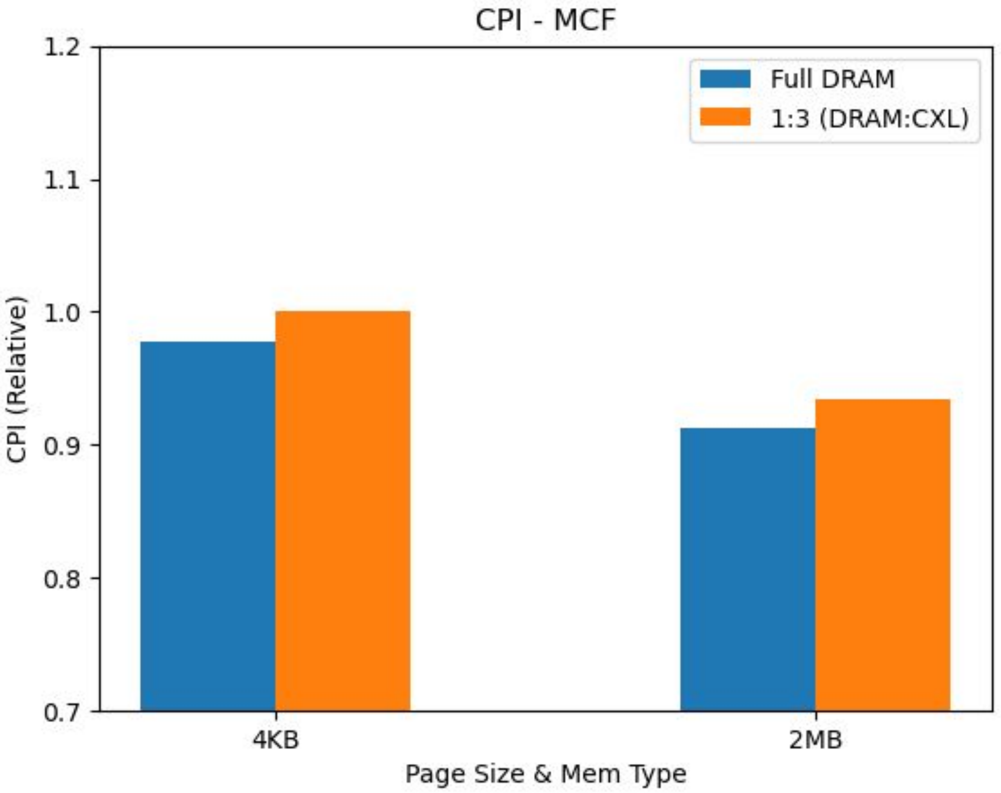
- Tracking in hardware (2)
- MMU: Randomly sampled one more level page walk (2)
- Place the hotness information with the page table (3, 4)
- Granularity can go beyond mapped page size (1)

Purpose of the 418 project: motivate the necessity of this design from parallel applications' perspective.

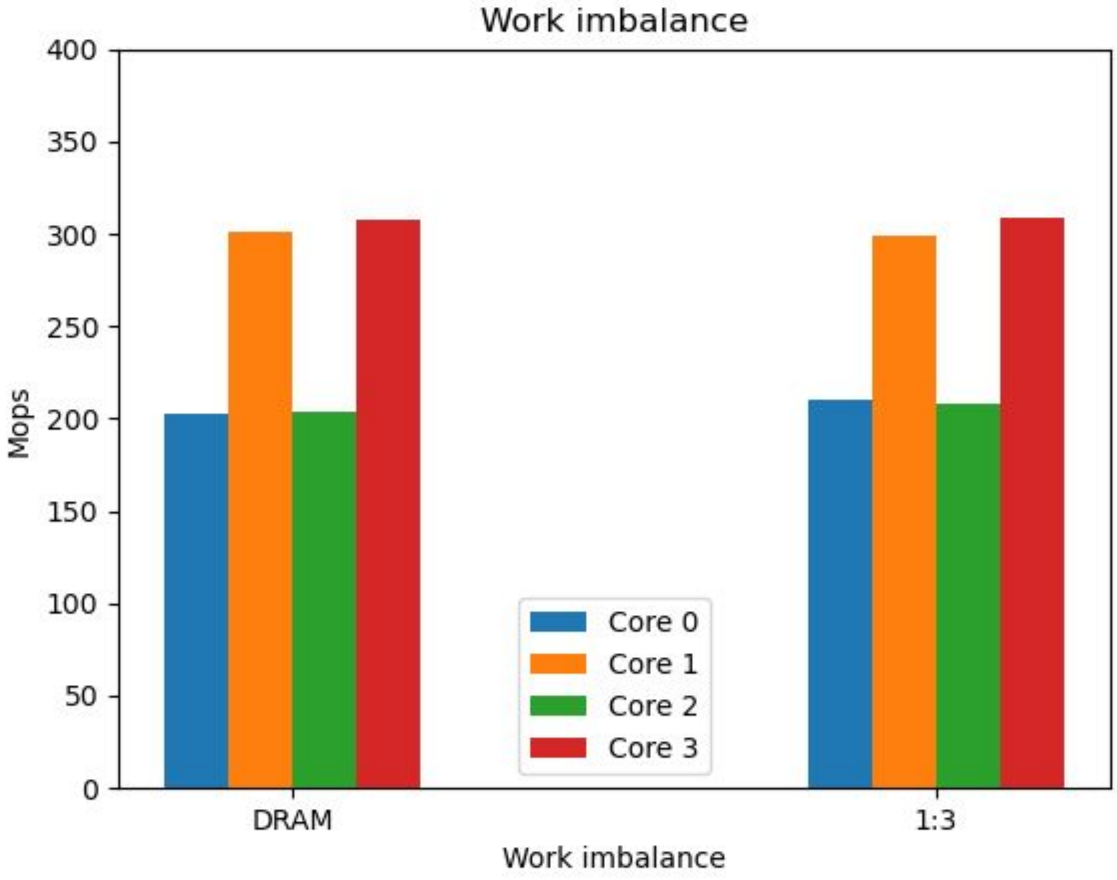
OpenMP-based Application Performance in Tiered-memory Systems


CPI - BFS

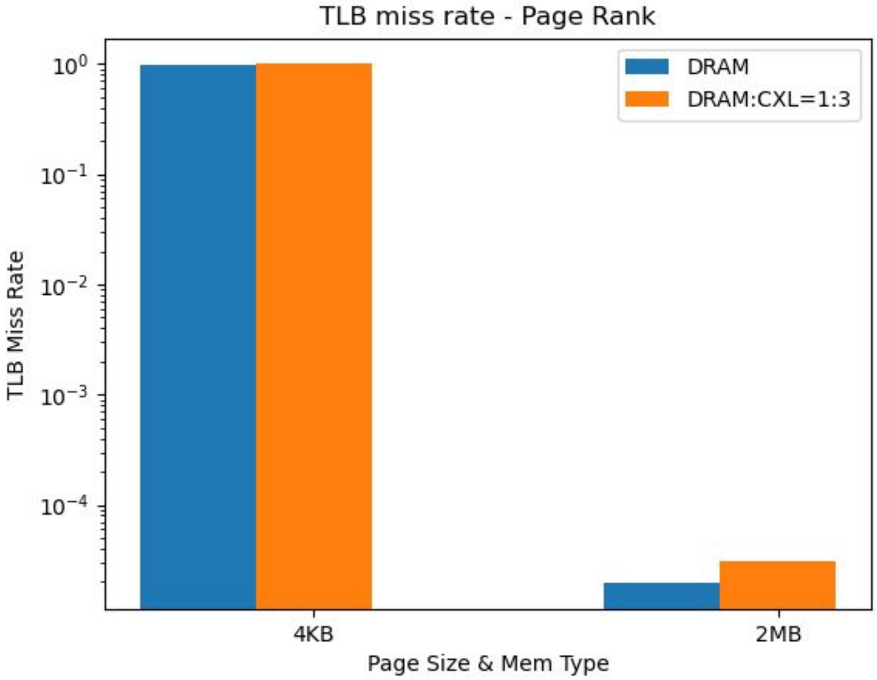

CPI - Page Rank



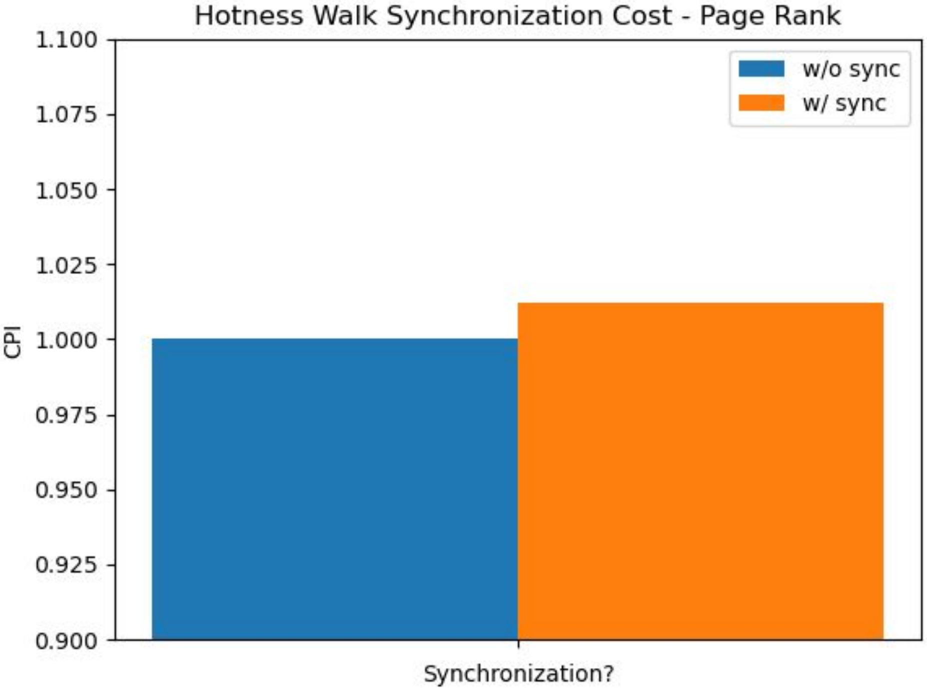
Work Imbalance - Page Rank



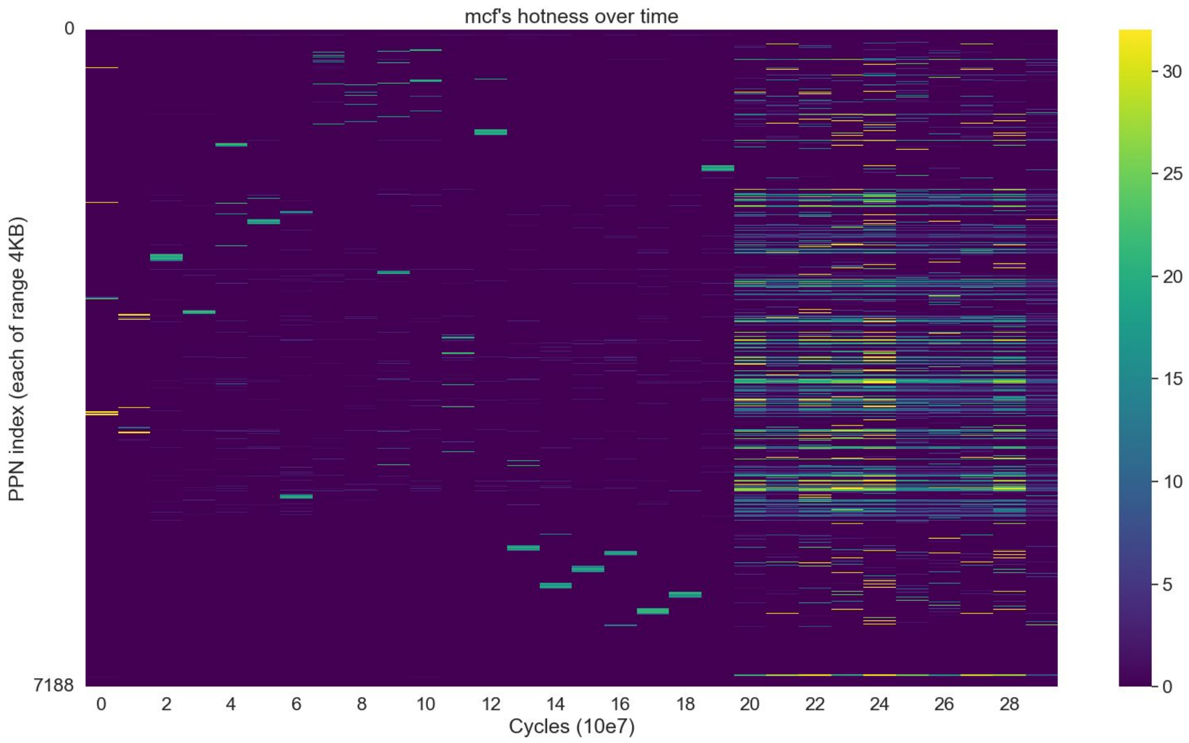
TLB Miss Rate

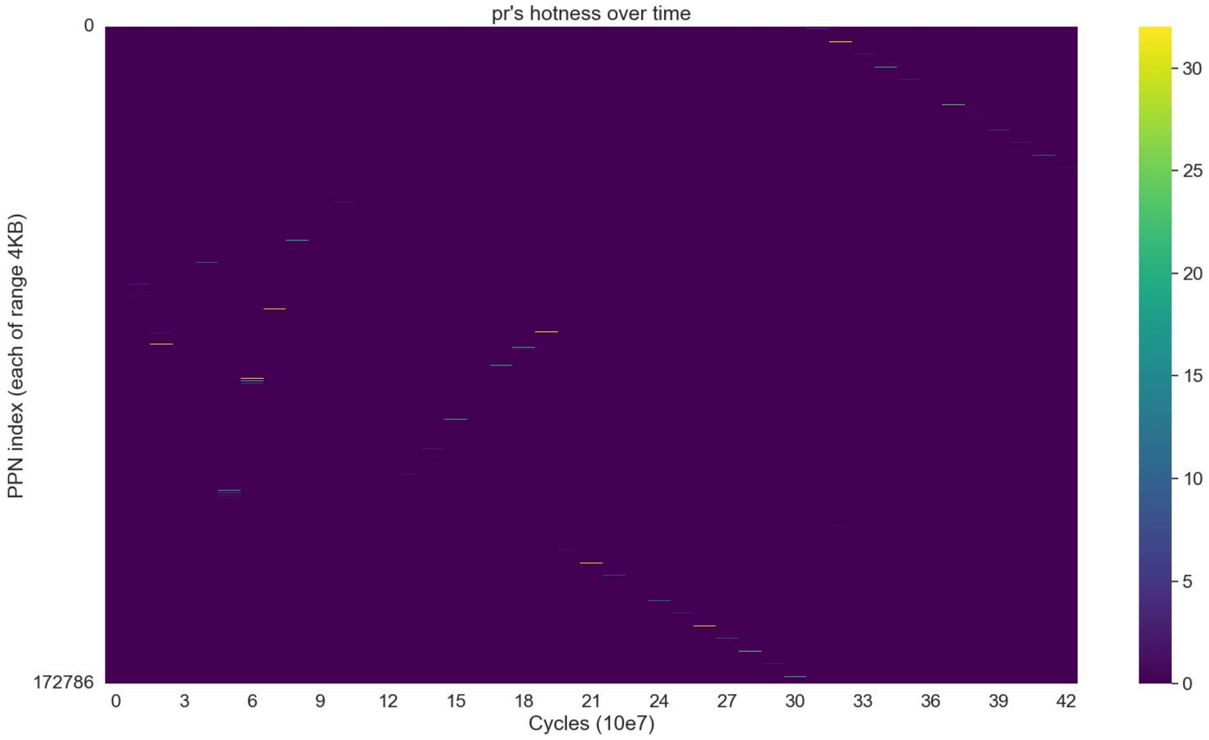


Synchronization Cost of Multi-core Hotness Walk



A Case for Fine Granularity





Conclusion: Hotness changes very fast, and are often imbalance in a region. Hotness should be tracked at a fine granularity and regularly in order to get accurate information

Conclusions & Next Steps

- **IMPORTANT: we found the multi-core simulation is not accurate enough, and removed completely unreliable studies caused by it from the project (more details in the report)**
- We summarized four requirements for efficient virtual memory management in a tiered-memory system
- We proposed the hotness walk design that satisfies all these requirements
- We profiled three parallel applications and studied one in detail on the necessity of our design
- In the future, we will try to fill in the details and implement the design in the architecture and system

References

[1] Amanda Raybuck, Tim Stamler, Wei Zhang, Mattan Erez, and Simon Peter. Hemem: Scalable tiered memory management for big data applications and real nvm. In Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles, SOSP '21, page 392–407, New York, NY, USA, 2021. Association for Computing Machinery.

[2] Taehyung Lee, Sumit Kumar Monga, Changwoo Min, and Young Ik Eom. Memtis: Efficient memory tiering with dynamic page classification and page size determination. In Proceedings of the 29th Symposium on Operating Systems Principles, SOSP '23, page 17–34, New York, NY, USA, 2023. Association for Computing Machinery.

[3] Aninda Manocha, Zi Yan, Esin Tureci, Juan Luis Aragón, David Nellans, and Margaret Martonosi. Architectural support for optimizing huge page selection within the os. In Proceedings of the 56th International Symposium on Microarchitecture (MICRO). IEEE, 2023.

[4] Hasan Al Maruf, Hao Wang, Abhishek Dhanotia, Johannes Weiner, Niket Agarwal, Pallab Bhattacharya, Chris Petersen, Mosharaf Chowdhury, Shobhit Kanaujia, and Prakash Chauhan. Tpp: Transparent page placement for cxl-enabled tiered-memory. In Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3, ASPLOS 2023, page 742–755, New York, NY, USA, 2023. Association for Computing Machinery.

(More in the report)