PACT: A Criticality-First Design for Tiered Memory

Hamid Hadian, Jinshu Liu*, Hanchen Xu*, Hansen Idden, Huaicheng Li

* *Equal contribution*

ASPLOS 2026

https://github.com/MoatLab/PACT

The Need for Memory Tiering

- DRAM is a major server **cost**: Azure (**50%**)
- **Memory tiering** allows systems to balance latency and capacity by combining fast but expensive memory (e.g., DRAM) with slower, higher-capacity alternatives (e.g., CXL memory**)**
- **Aggressive promotion** policies that cause excessive, often unnecessary, migrations

Traditional Tiered Memory Designs are Inefficient

Hotness metric alone fails to capture true performance impact of memory accesses and guiding page migrations

PACT: Online, page-granular tiered memory that elevates performance criticality to a first-class design principle

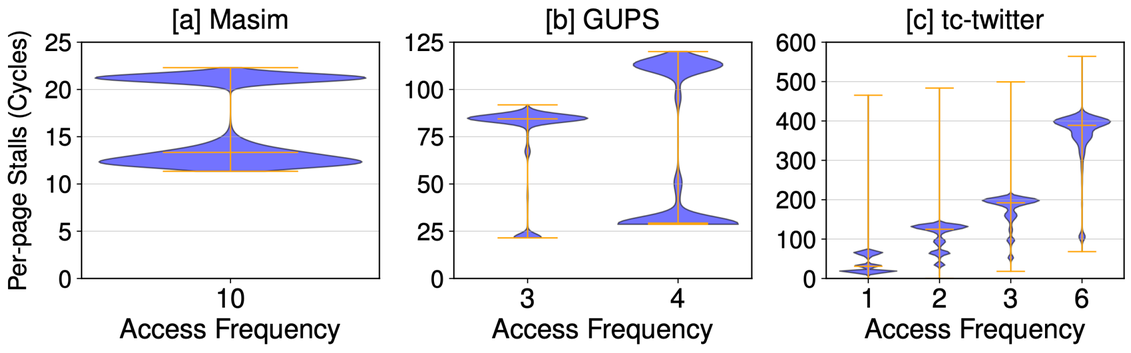
PAC (Per-Page Access Criticality): A Metric That Quantifies Page's Contribution to Performance

Unique challenges: *How do we Quantify per-page criticality? and how do we (re)design memory tiering policies?*

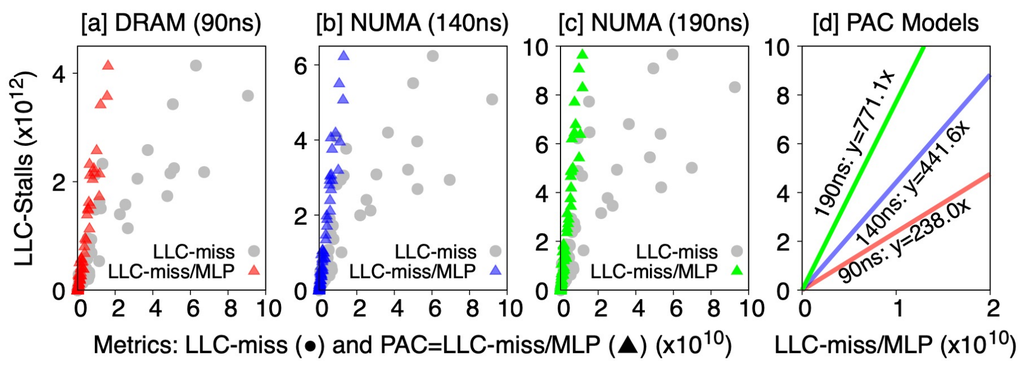
- ❑ Introducing a **fine-grained** metric that directly quantifies the performance impact of page accesses
- ❑ Using a **lightweight** profiling enabling online measurement

Motivation: Hotness-based systems treat all **LLC misses** equally, fail to consider **performance criticality** of page accesses

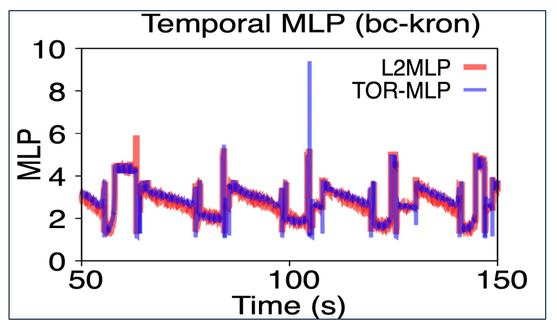
- ❑ Pages with **identical access frequency** can incur vastly **different stall costs**
- ❑ **Substantial variation** in per-page stall cost, often up to **65X**



PAC MLP-based model using 96 workloads across three setups



❑ **TOR-MLP** follows the same temporal pattern of system-wide **L2MLP** (red)



$$MLP_{CXL} = \frac{\text{TOR occupancy}}{\text{Number of active cycles}}$$

❑ **Profiling per-page criticality:**

- Estimating CPU stalls induced by each memory tier
- (**Per-tier MLP**) using **CHA counters**
- Attributing those stalls to the responsible pages (**PEBS**)

$$PAC = Stall_{CXL} \times \frac{Ap}{At} \Rightarrow \frac{\text{Page access count}}{\text{Total access count}}$$

PAC Modeling

PAC modeling : How do we accurately profile per-page access criticality?

- ❑ **MLP-based model:** Including **memory level parallelism** exhibits a strong prediction model for the **LLC-stalls**
- ❑ **Fine-grained metric** reflects per-page access cost in terms of CPU stalls and MLP

$$Stall_{CXL} = K \times \frac{LLC-missesCXL}{MLP_{CXL}}$$

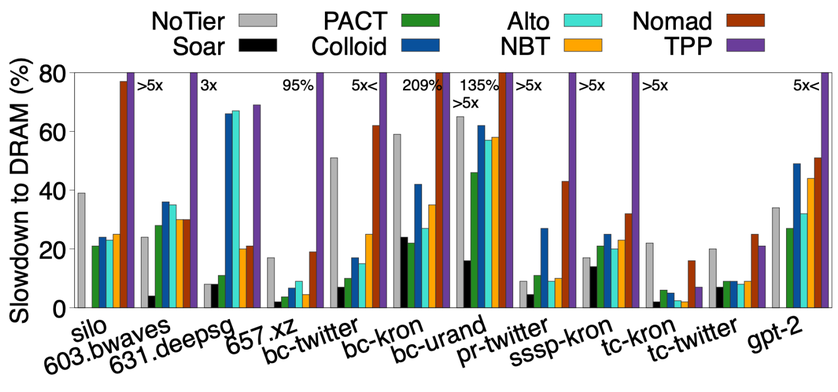
- ❑ Proposed **per-tier estimator** based on CHA queue occupancy

PAC-Centric Migration policies

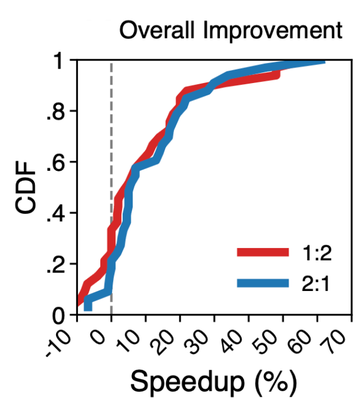
- ❑ **When, what pages, and how often to migrate?** Highly skewed PAC values require a new distribution
- ❑ **Adaptive promotions**: Provides smooth page distribution leveraging the Per-page MLP and adaptive binning in a histogram
- ❑ **Eager demotions**: It **proactively** evicts cold pages from the fast-tier using the kernel's LRU mechanism. It contrasts with traditional **on-demand** demotion strategies

Evaluation

(1). Improvement over second-best hotness-based solution up to **61%** and **50x** less migrations



(2). CDF of PACT performance improvement relative to other tiering solutions for different DRAM/CXL ratio



(3). Page promotion (PACT vs. frequency). The figure contrasts different migration behaviors based on PAC and frequency