TPP : Transparent Page Placement for CXL-Enabled Tiered-Memory

Hasan Al Maruf, Hao Wang, Abhishek Dhanotia, Johannes Weiner, Niket Agarwal, Pallab Bhattacharya, Chris Petersen, Mosharaf Chowdhury, Shobhit Kanaujia, Prakash Chauhan

University of Michigan

SymbioticLab

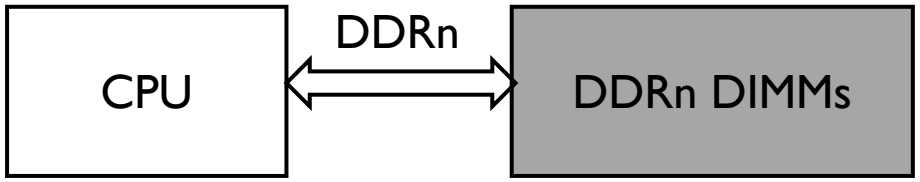
Meta

ASPLOS 2023, Vancouver, Canada, 29 March 2023

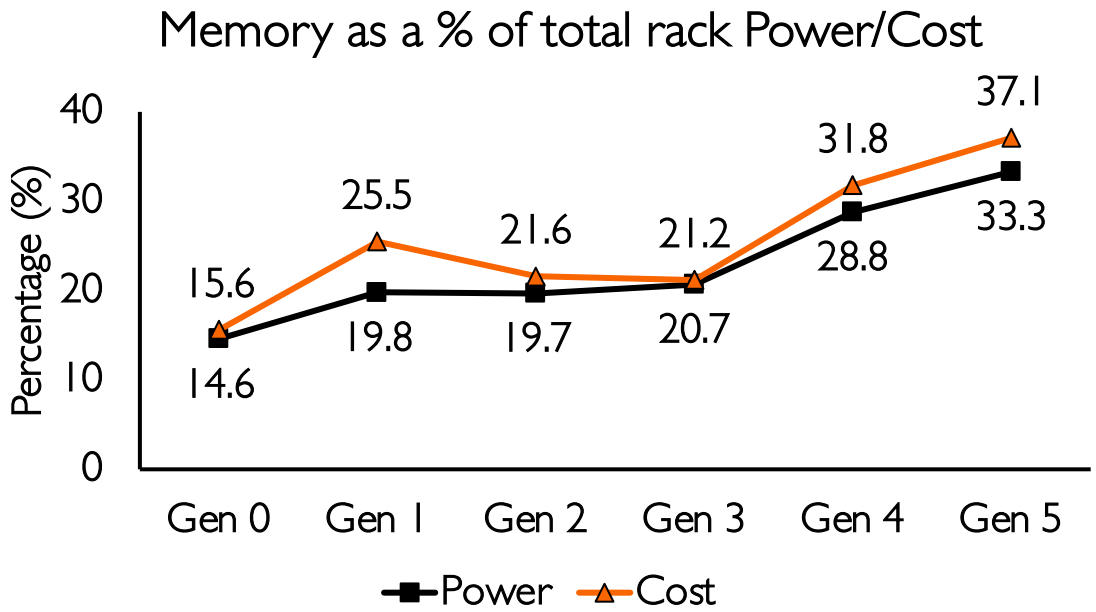
Memory Today Tightly Coupled to CPU

Memory is homogeneous

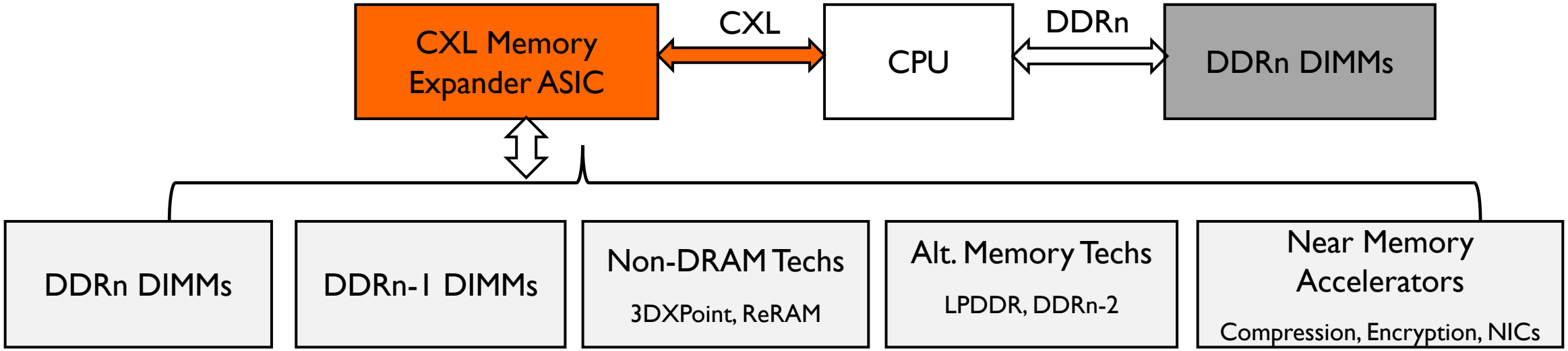
- same type, latency, capacity, bandwidth etc.



Rack-level memory power and cost increases with new hardware generations



CXL-based Heterogeneous Memory



Flexible CPU and memory bus

- different memory capacity to bandwidth ratio
- combine different generation of DIMMs
- use cheaper and low power memory alternatives
- utilize near memory accelerators

CXL-Memory Characteristics

Byte addressable in same physical address space

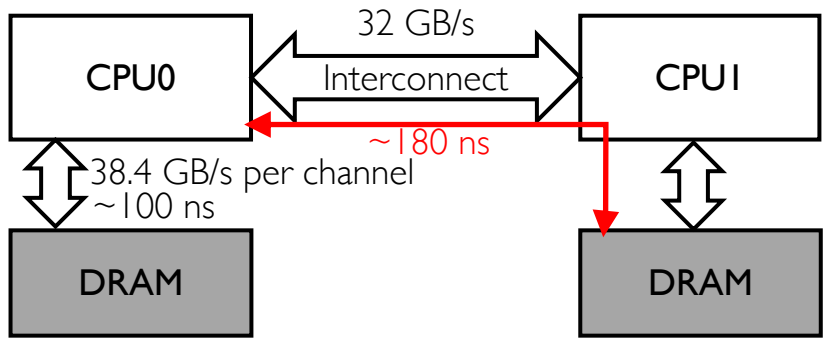
- transparent allocation with cache-line granular access

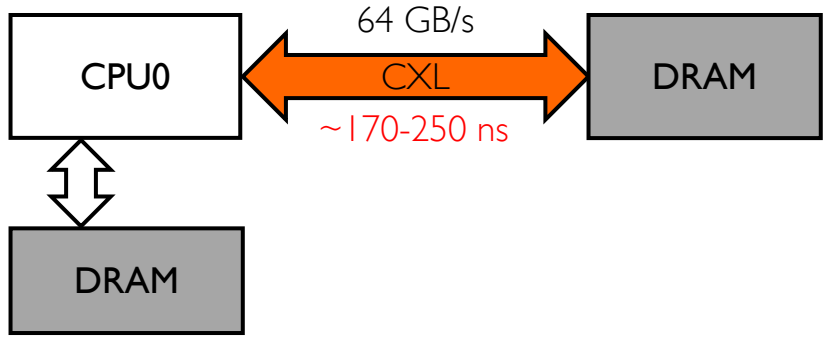
Memory bandwidth is like DDR channels

- NUMA BW is better than a dual socket system

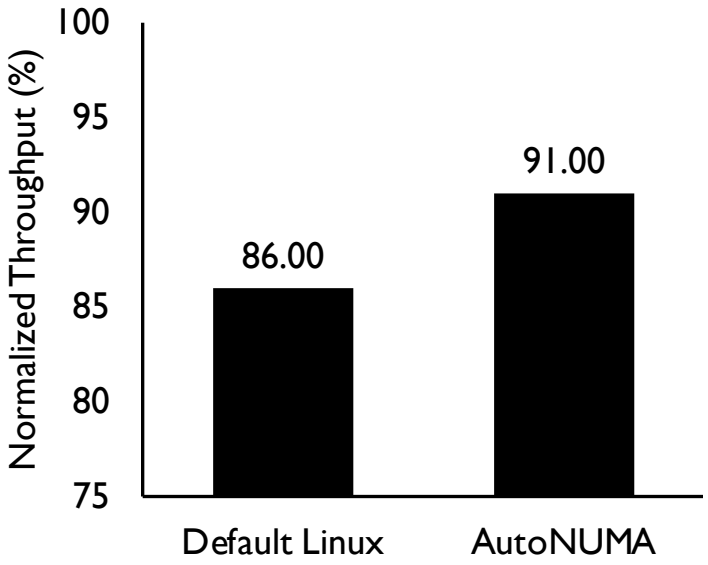
Close to NUMA latency on dual socket systems

- adds ~100ns latency over normal DRAM access



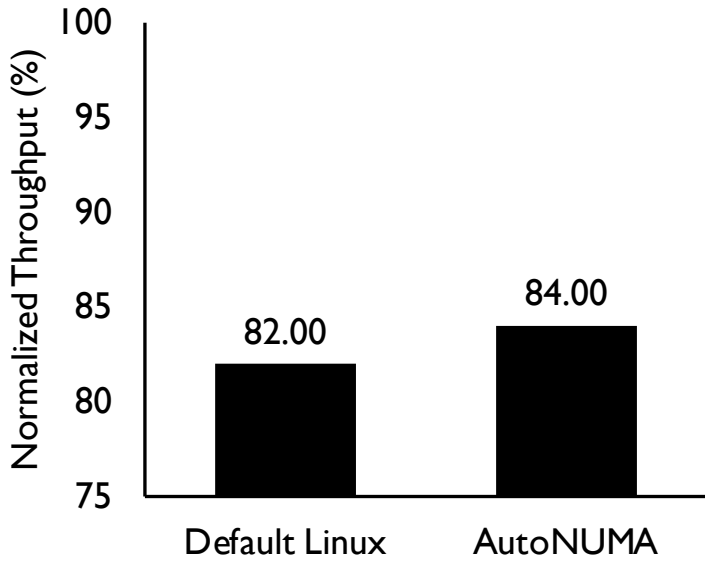


Performance Drops with Large CXL-Memory



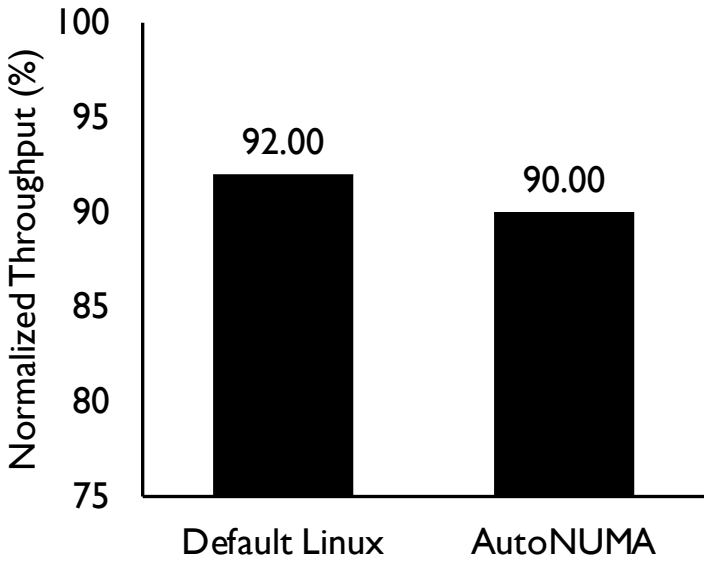
Cache Application

14%



Web Application

18%



Data Warehouse

10%

Transparent Management of **Tiered-Memory**

1. **Effective placement of hot pages**
   - *faster page allocation*
   - *apt hot page detection*
   - *lightweight page movement*
   - *sensitivity towards different page types*

2. **Workload characterization**
   - *page temperature and re-access time*
   - *application's expected behavior*

Transparent Page Placement (TPP) for Heterogeneous Tiered-Memory System

source code available at https://lwn.net/Articles/876993/

Effective memory management for tiered-memory system

- lightweight demotion to slow memory tier
- efficient hot page promotion to fast memory tier
- optimized page allocation path to reduce latency
- workload aware page allocation policy

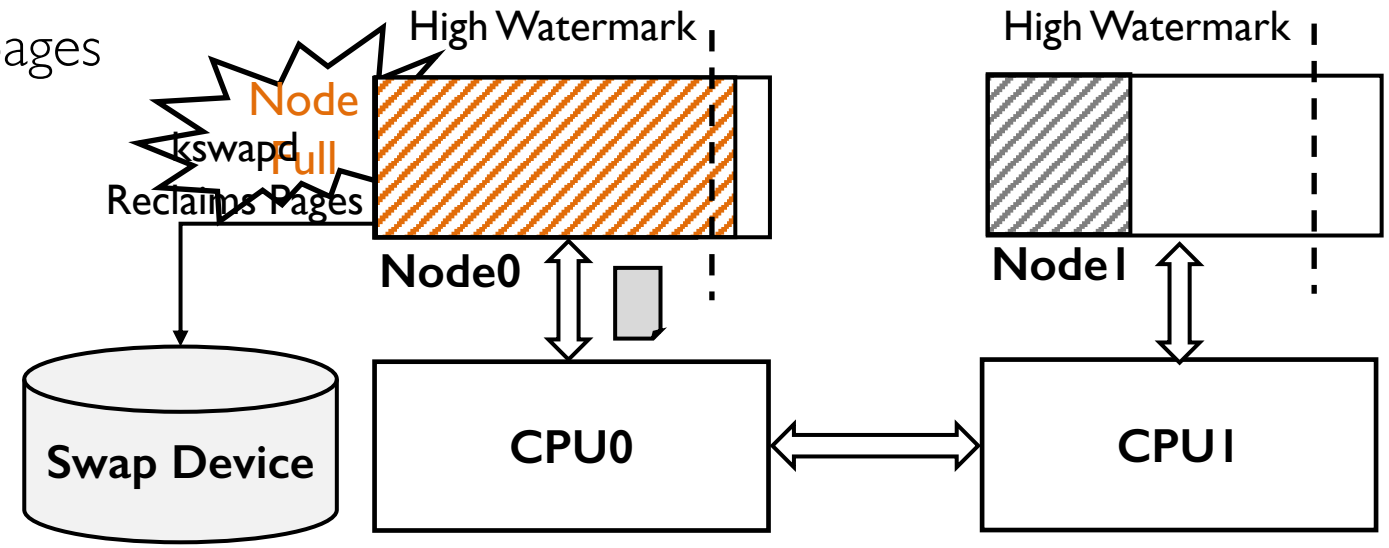
Without modifying any

- applications, or
- hardware

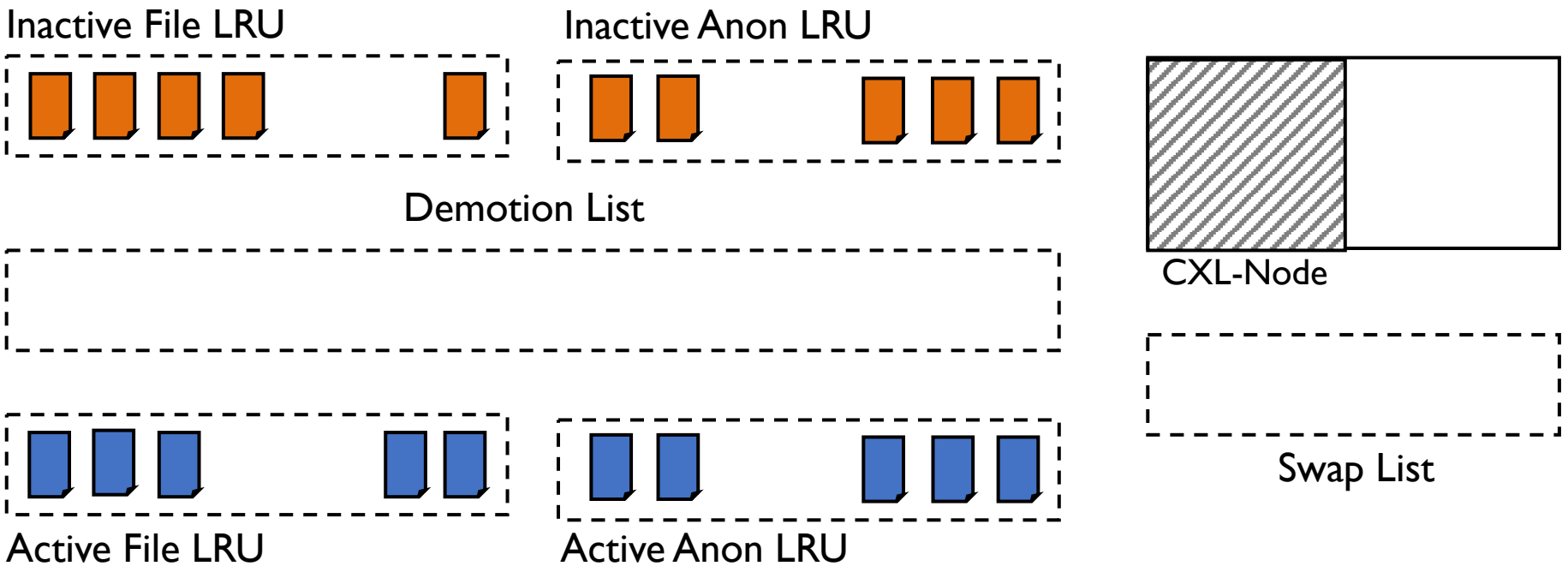
Page Placement in Default Linux

Every node maintains a watermark to determine load

- reclamation triggers when number of free pages goes below the watermark
- new pages get allocated to remote node
- reclamation stops when free pages goes above the watermark
- new allocations again happen on local node



Demotion in TPP - Migrate to Slow Tiers



Maintains a separate demotion page list

- scans inactive pages fist
- if not enough, move to active pages

Tries to migrate scanned pages to slow memory tier

- failed pages follows default reclamation path

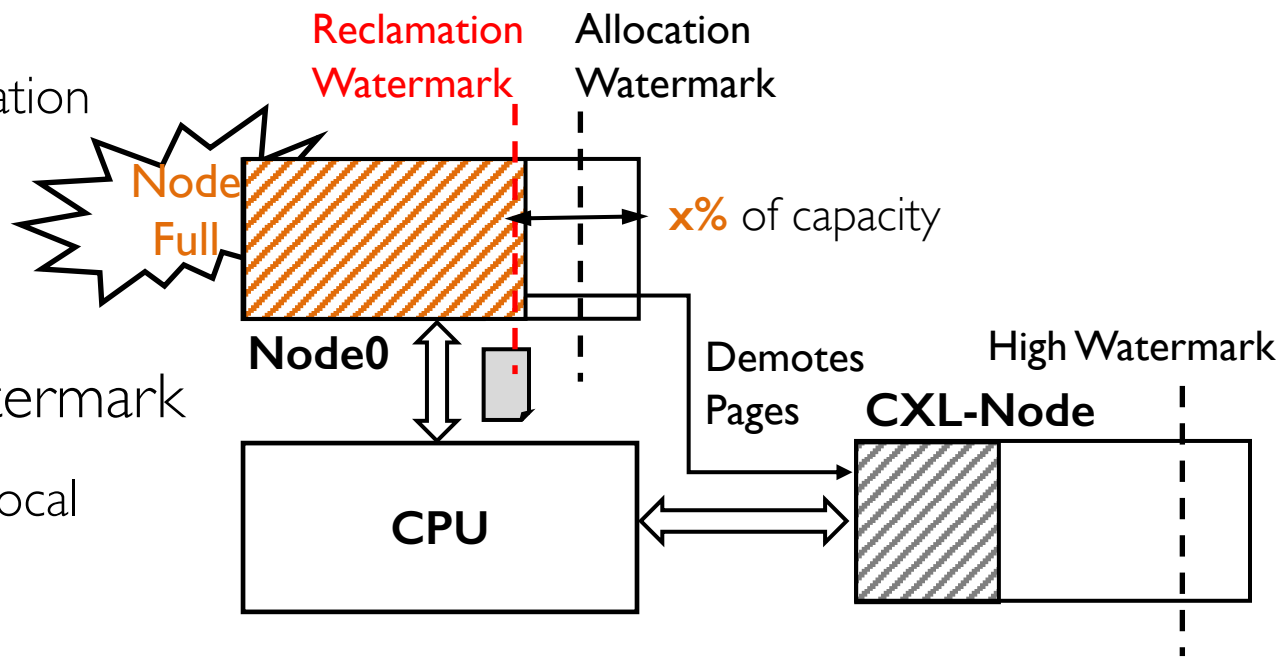
Optimized Allocation Path in TPP

Decouples page allocation and reclamation logic

- reclamation triggers when x% memory is left
- allocation happens on local node as long as allocation watermark is satisfied

User-space interface to control reclamation watermark

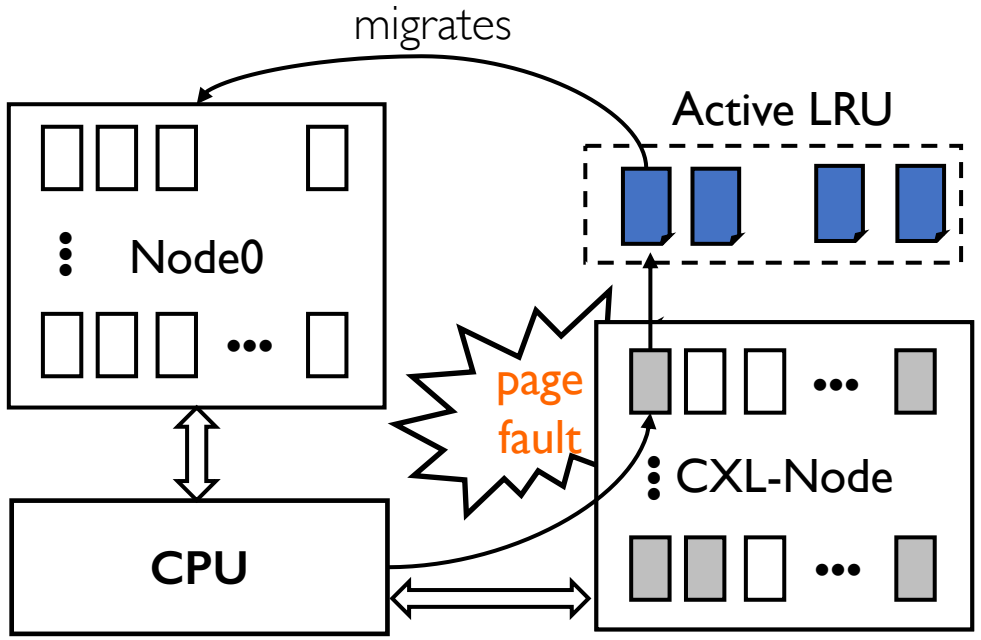
- *vm.demote_scale_factor* (by default, set to 2% of local node's capacity)



Effective Promotion of Trapped Hot Pages

Samples only CXL-node

- promoting local node pages is meaningless



Considers page activeness during promotion

- NUMA hint may come from infrequently accessed page
- such pages become demote candidate after being promoted
- include active LRU heuristics in promotion
- move inactive hinted page to active LRU and wait for next fault
- anon and file promotion rate varies on respective LRU activities

User Interface

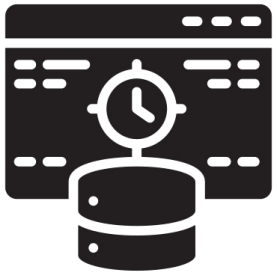
TPP appears as a new **AutoNUMA** mode

- *echo x > /proc/sys/kernel/numa_balancing*
  - 0x0: NUMA_BALANCING_DISABLED
  - 0x1: NUMA_BALANCING_NORMAL
  - **0x2: NUMA_BALANCING_MEMORY_TIERING**

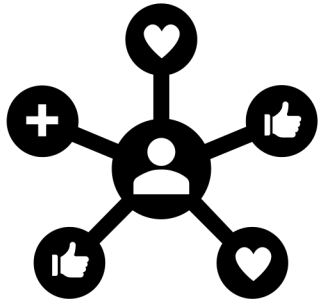
- If there is a single CPU-attached memory node, automatically falls back to NUMA_BALANCING_MEMORY_TIERING mode

Evaluation

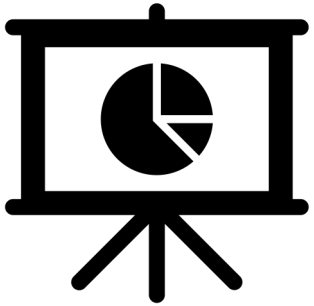
Deploy and evaluate on **Meta cluster** *in production w/* CXL-Memory expander ASIC



Caching applications



Social media application

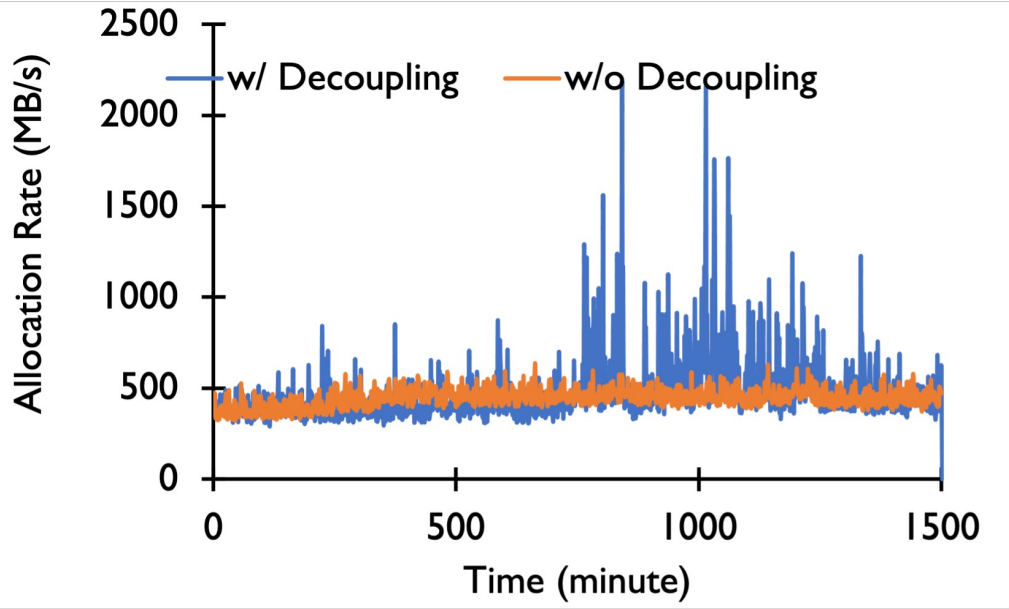


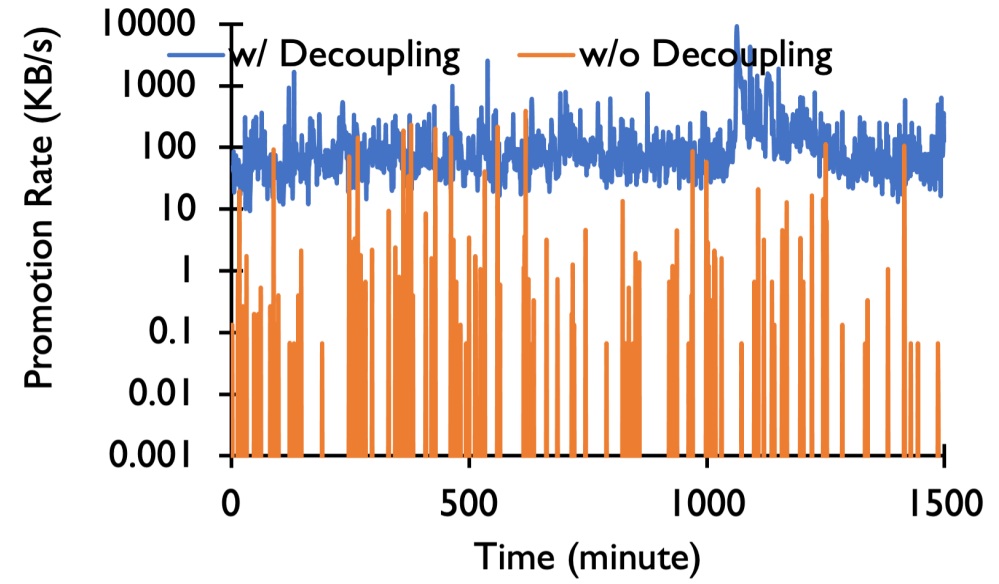
Data warehouse & analytics

Better Allocation and Promotion with TPP

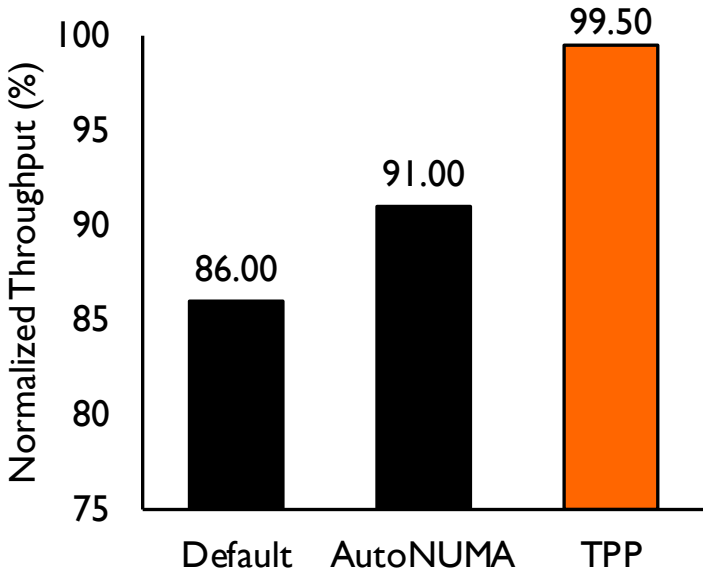
Decoupling allocation and reclamation logic helps handle bursts more effectively

- **1.6x** better allocation rate at 95th percentile
- promotion can be **30x** faster than default Linux



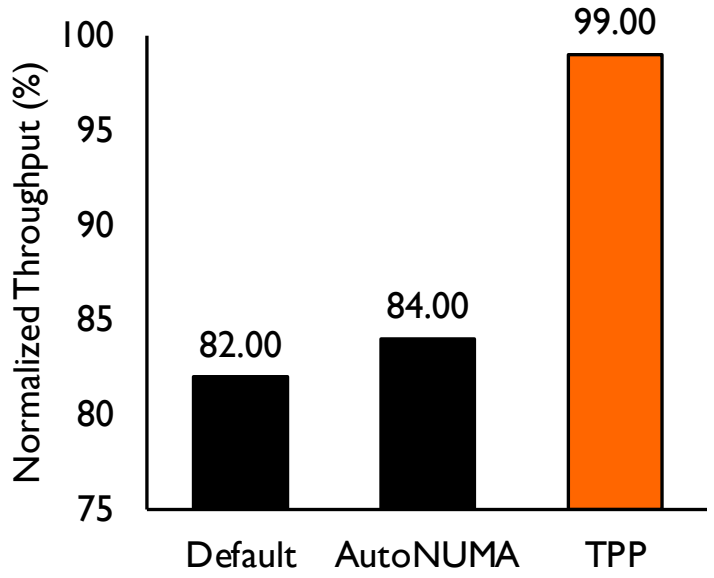


Performs Great w/ 80% CXL-Memory



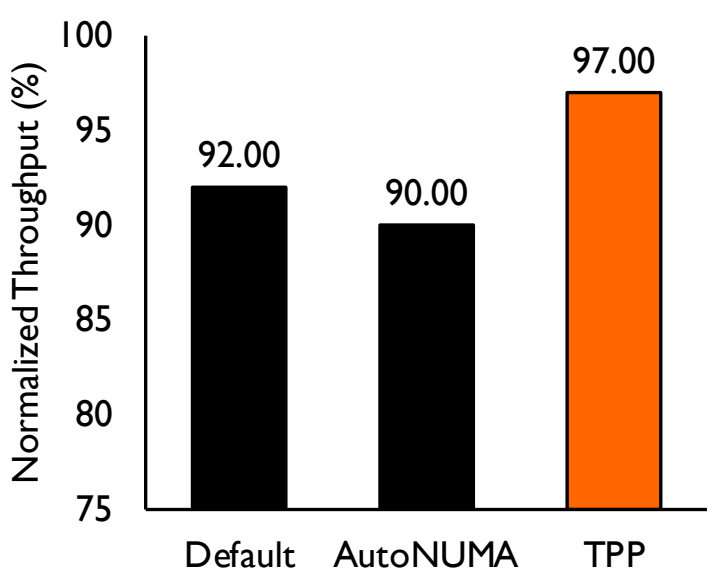
Cache Application

13%



Web Application

17%



Data Warehouse

7%

Transparent Page Placement for Tiered-Memory System

source code available at https://lwn.net/Articles/876993/

Effective memory management for tiered-memory system

- lightweight demotion
- **30x** faster hot page promotion
- **1.6x** optimized page allocation
- workload aware page allocation policy

Without modifying any

- applications, or
- hardware

Thank You!

for any queries, contact at
hasanal@umich.edu