# Supporting Dynamic Translation Granularity for Hybrid Memory Systems

Bokyeong Kim†, **Soojin Hwang***, Sanghoon Cha‡, Chang Hyun Park§, Jongse Park*, and Jaehyuk Huh*

*School of Computing, KAIST

†Samsung Research

‡Samsung Advanced Institute of Technology

§Uppsala University

CASYS
Computer Architecture and Systems Lab

# Contents

- Motivation
- Two-Level Decoupled Address Translation
- Dynamic Frame Size Selection
- Evaluation

KAIST

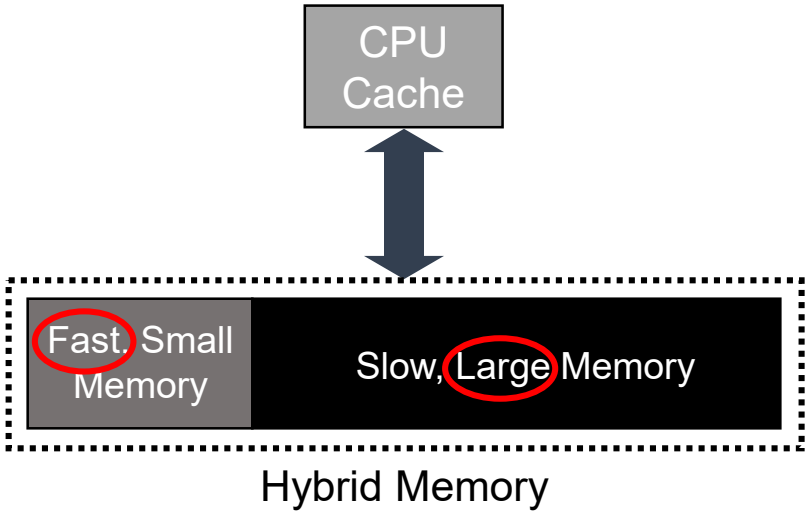
# Motivation

# Why Hybrid Memory?

- Data-centric applications
  - Requires high bandwidth, large capacity
- New memory techniques
  - Advanced performance, but still suffer for cost
- Memory heterogeneity
  - Disaggregated memory

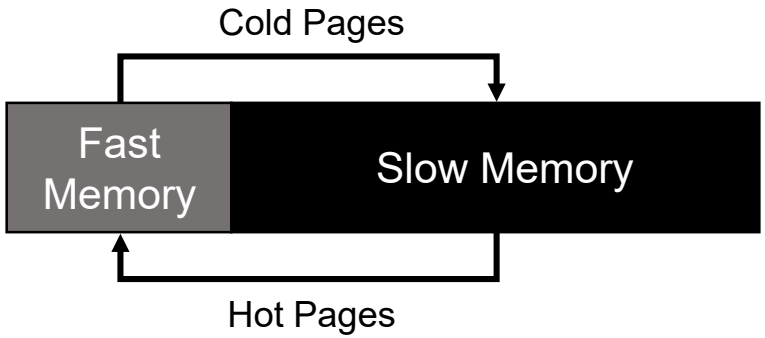
# Hybrid Memory

- Take advantages of various memory devices
  - High bandwidth, large capacity, ...
- Managed by operating system
  - Virtualization: Flexibility on memory management



# Hybrid Memory System

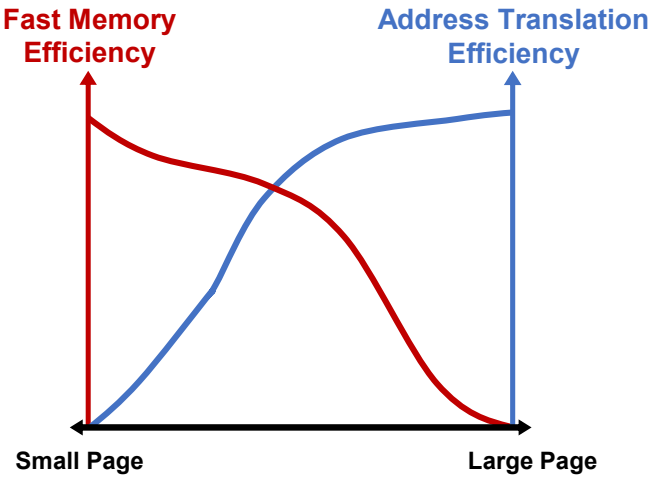
- Hotness-based data placement (migration)
- Design choice: page granularity
  - Fine-grained[1]: Memory management efficiency
  - Huge page[2]: Address translation efficiency




[1] Chou et al., "CAMEO: A Two-Level Memory Organization with Capacity of Main Memory and Flexibility of Hardware-Managed Cache", MICRO'14
[2] Agarwal et al., "Thermostat: Application-transparent Page Management for Two-tiered Main Memory", ASPLOS'17

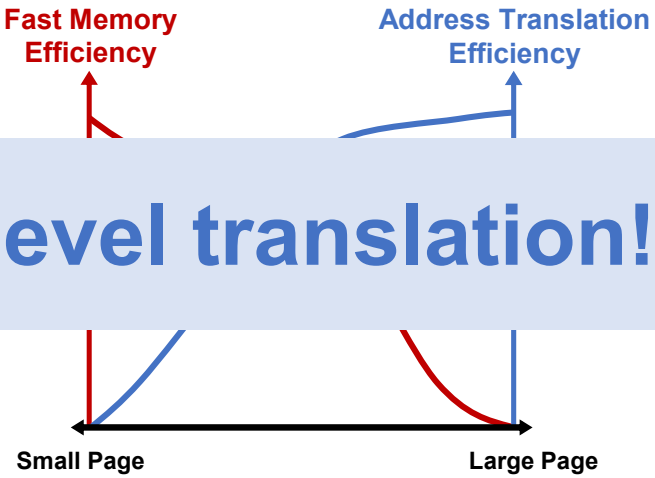
# Virtualized Hybrid Memory

- Conflicting objectives with page sizes
  - Small: Avoids waste of fast memory
  - Large: Reduces translation overhead
- Need reduction of data management cost
  - Nimble hot page detection
  - Efficient migration



# Virtualized Hybrid Memory

- Conflicting objectives with page sizes
  - Small: Avoids waste of fast memory
  - Large: Reduces translation overhead
- Need reduction of data management cost
  - Nimble hot page detection
  - Efficient migration




**HW-SW cooperative two-level translation!**

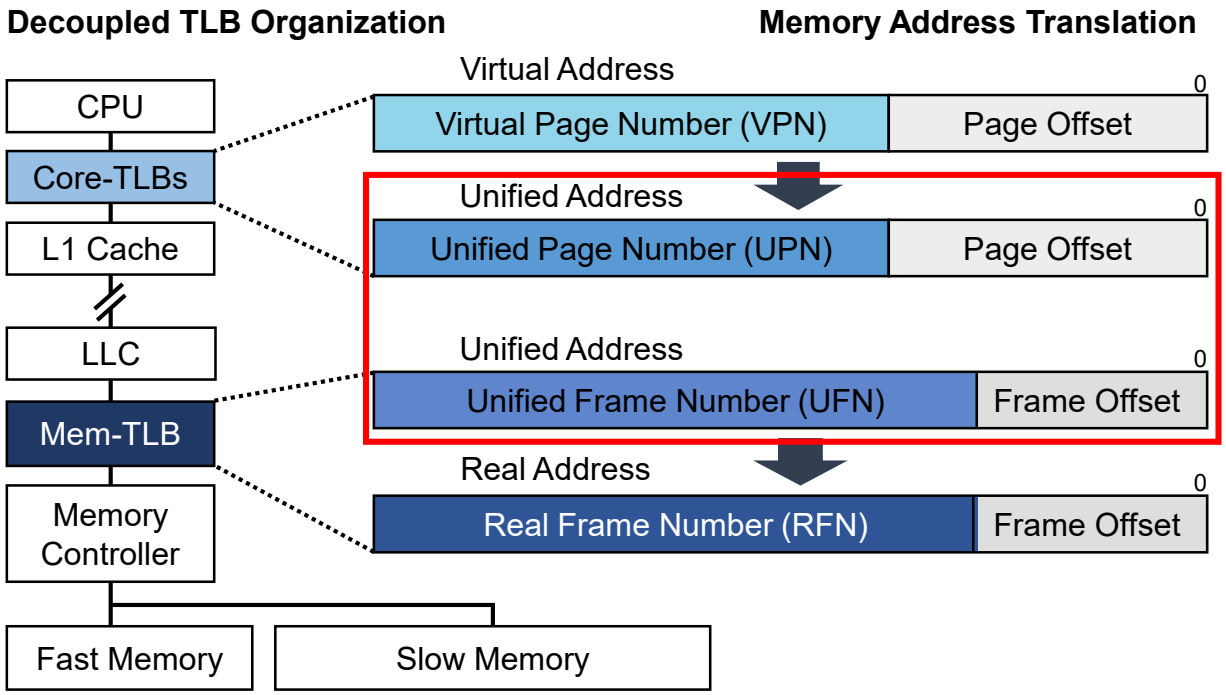
KAIST

# Two-Level Decoupled Address Translation

CASYS
Computer Architecture and Systems Lab

# Decoupled Address Translation

- Adding one more virtualization layer
- Prior use case: compressed memory[1][2]

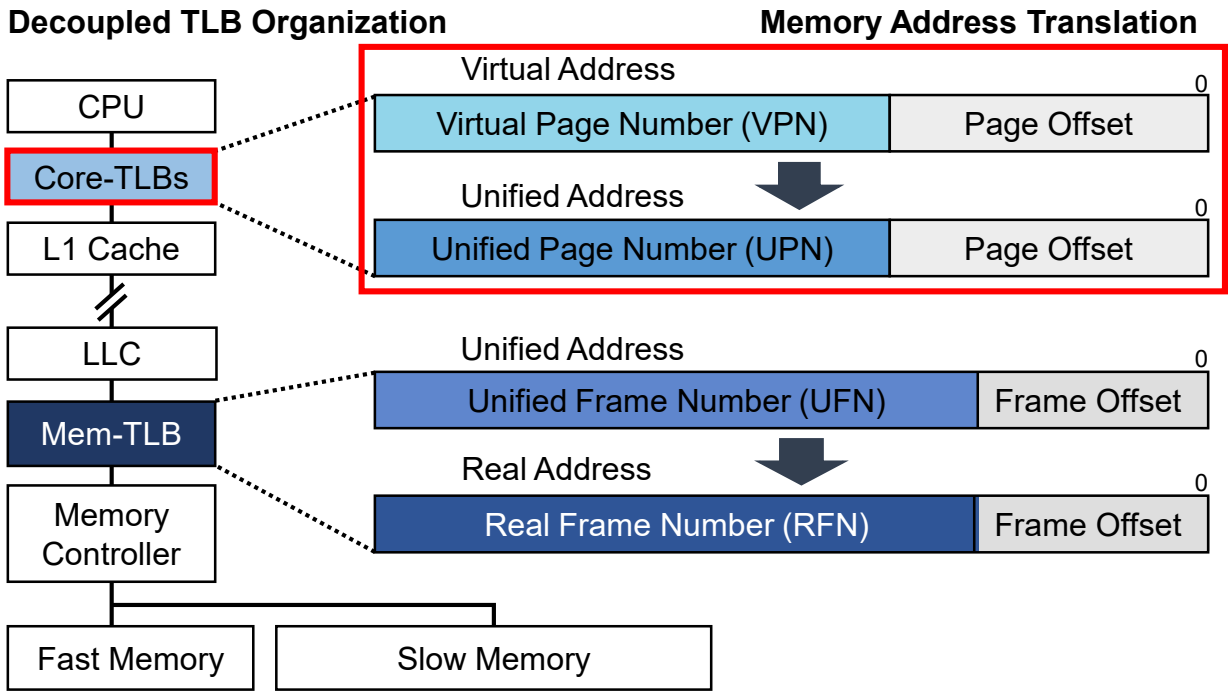




[1] Tremaine et al., "IBM Memory Expansion Technology (MXT)", IBM Journal of Research and Development, 2001
[2] Kim et al., "Transparent Dual Memory Compression Architecture", PACT'17

# Decoupled Address Translation

- Adding one more virtualization layer
- Prior use case: compressed memory[1][2]

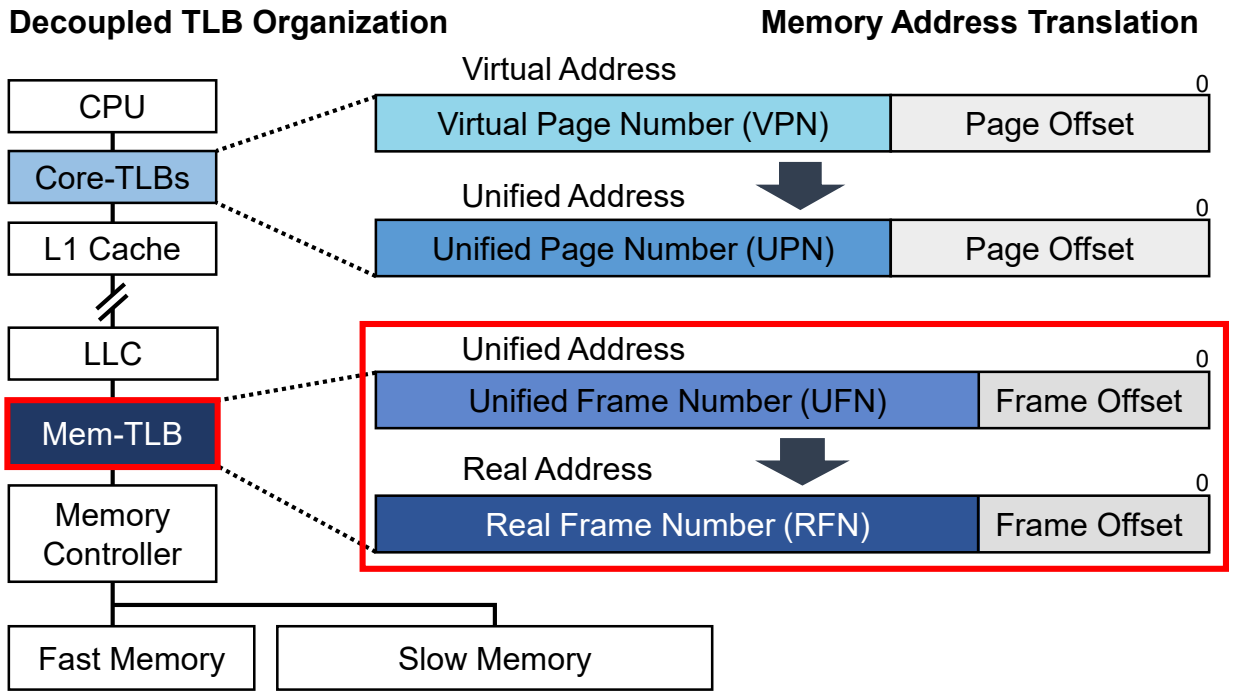





[1] Tremaine et al., "IBM Memory Expansion Technology (MXT)", IBM Journal of Research and Development, 2001
[2] Kim et al., "Transparent Dual Memory Compression Architecture", PACT'17

# Decoupled Address Translation

- Adding one more virtualization layer
- Prior use case: compressed memory[1][2]

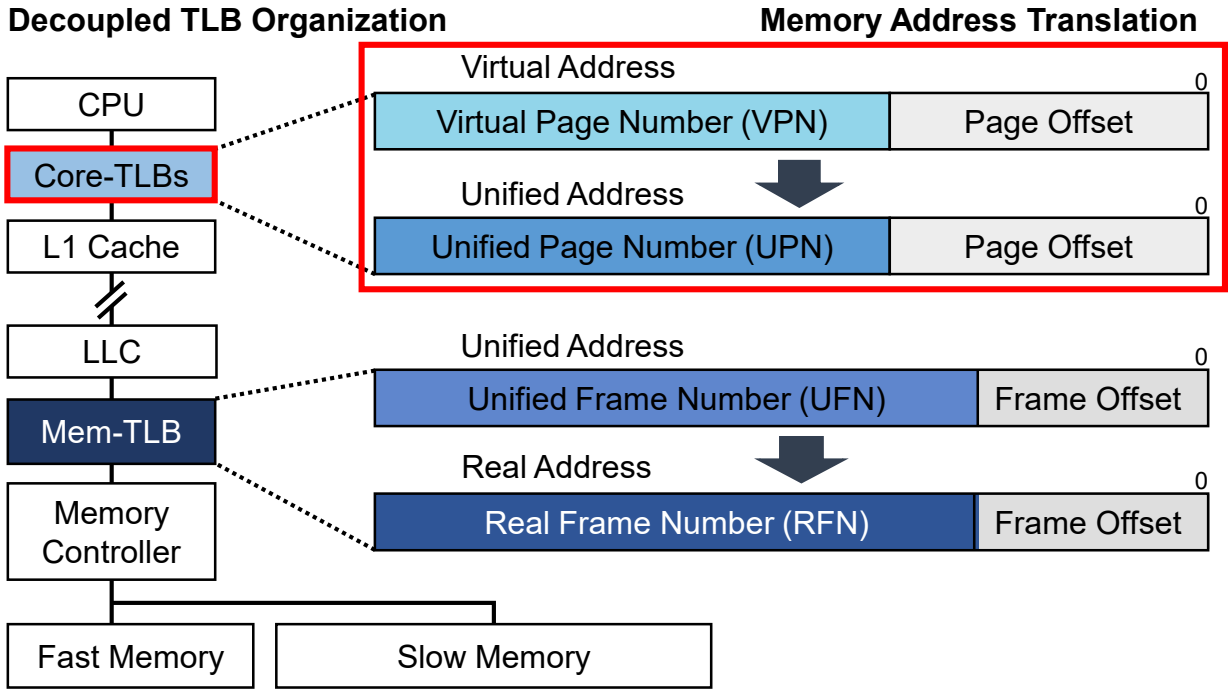





[1] Tremaine et al., "IBM Memory Expansion Technology (MXT)", IBM Journal of Research and Development, 2001
[2] Kim et al., "Transparent Dual Memory Compression Architecture", PACT'17

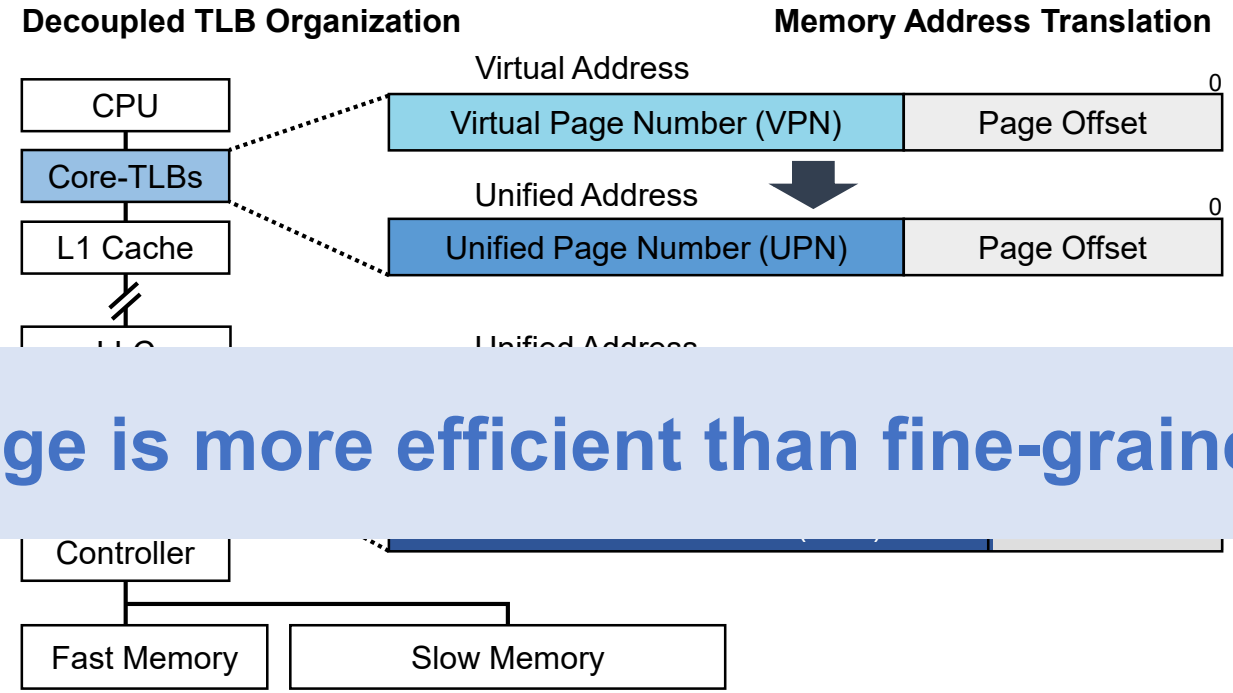
# Design Choice: Granularity

- Core-side translation: Page size
  - Virtualized unit of memory management
  - Reduction of translation costs is important



# Design Choice: Granularity

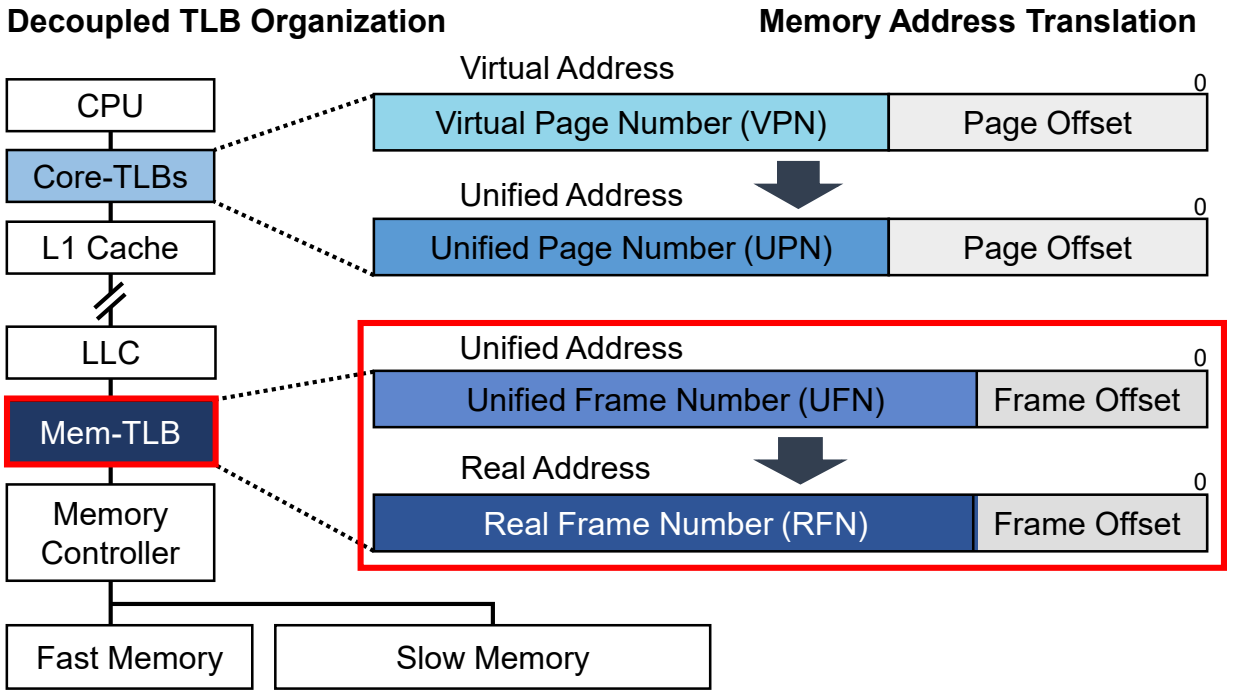
- Core-side translation: Page size
  - Virtualized unit of memory management
  - Reduction of translation costs is important




**Huge page is more efficient than fine-grained page!**

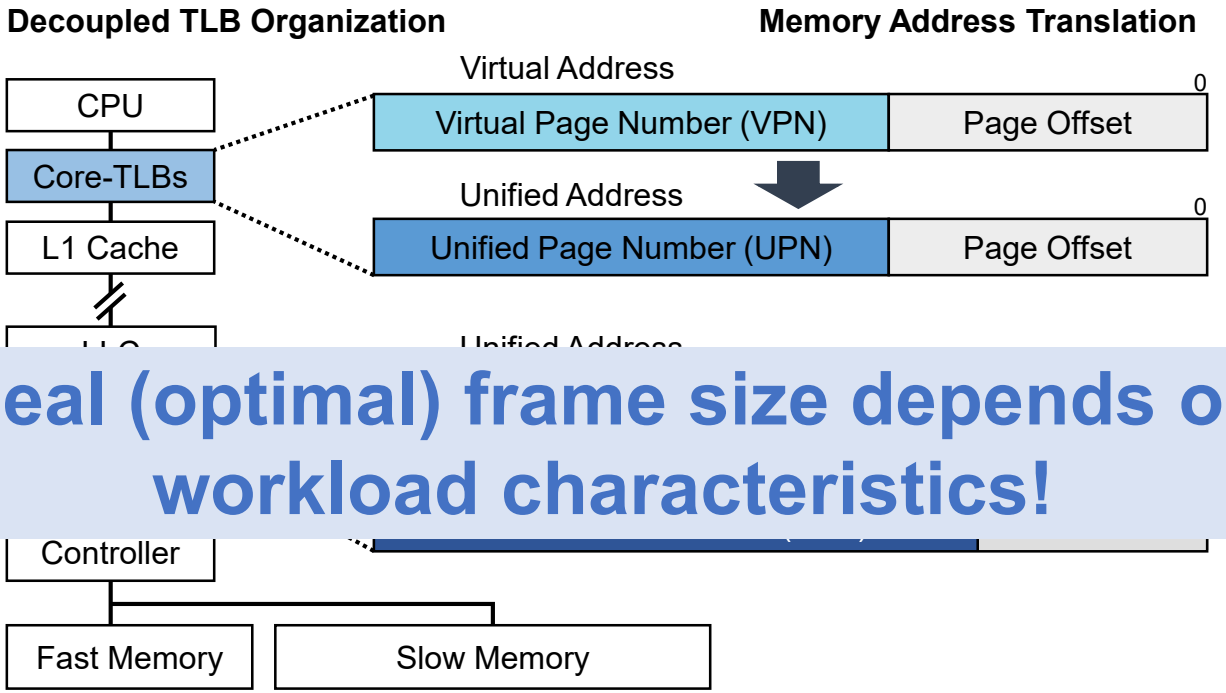
# Design Choice: Granularity

- Memory-side translation: Frame size
  - Real unit of memory management
  - Translation cost can also make impact



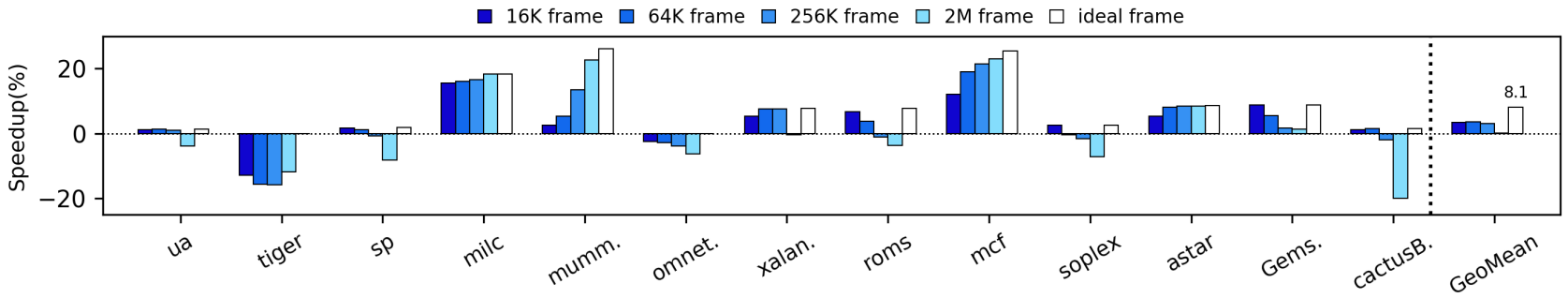
# Design Choice: Granularity

- Memory-side translation: Frame size
  - Real unit of memory management
  - Translation cost can also make impact



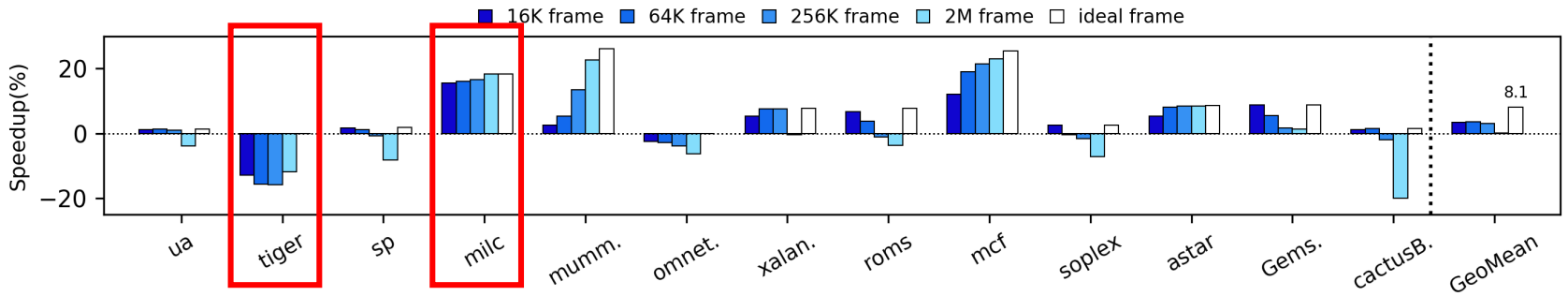

**Ideal (optimal) frame size depends on workload characteristics!**

# Limitation of Fixed Frame Size



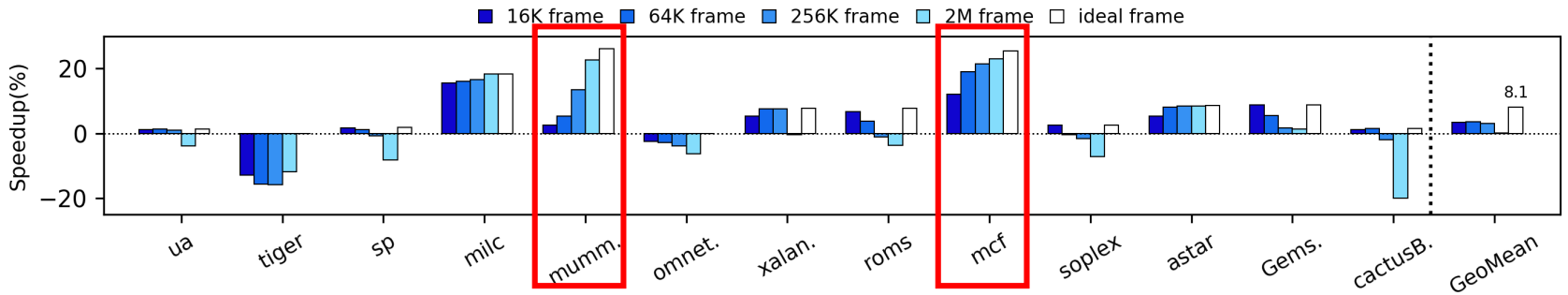
# Limitation of Fixed Frame Size

- Wide range of variation on ideal frame sizes



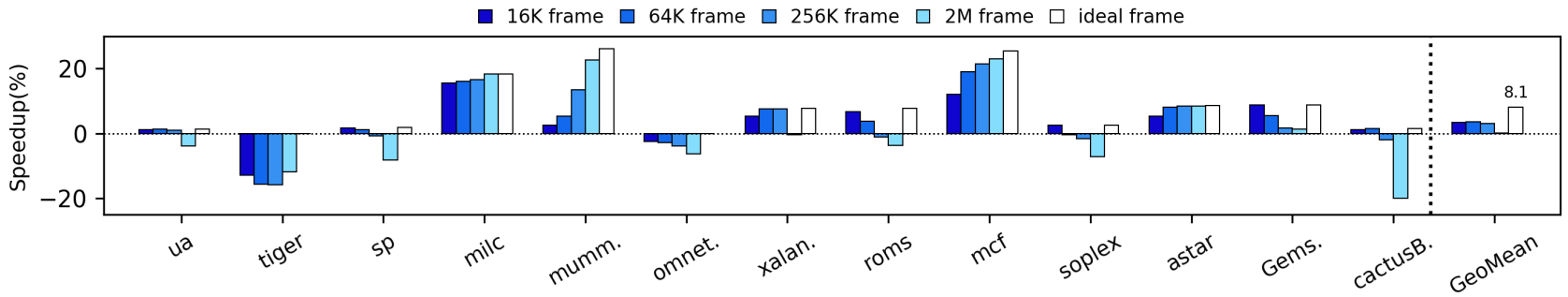
# Limitation of Fixed Frame Size

- Wide range of variation on ideal frame sizes
- Performance gap between ideal and non-ideal



# Limitation of Fixed Frame Size

- Wide range of variation on ideal frame sizes
- Performance gap between ideal and non-ideal



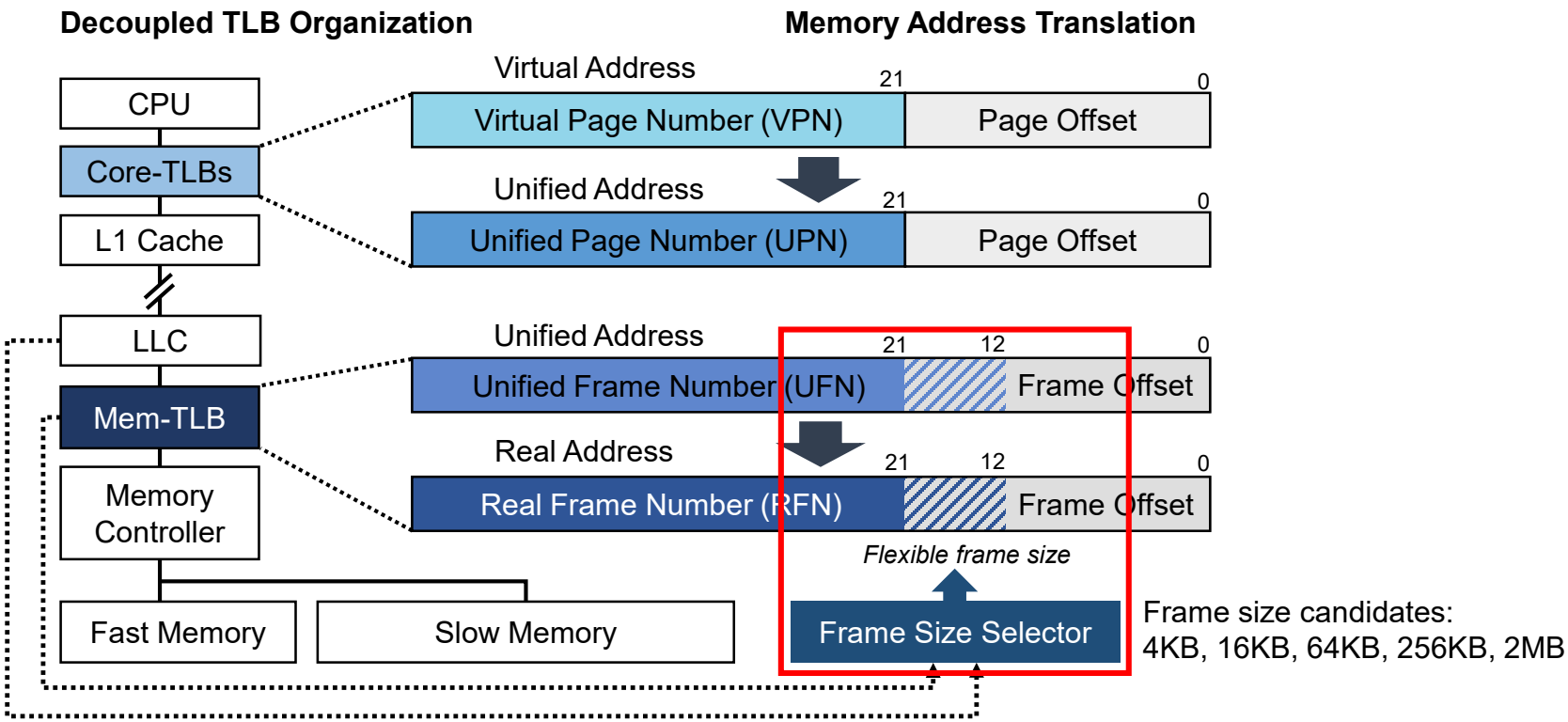

**Need dynamic frame size selection!**

KAIST

# Dynamic Frame Size Selection

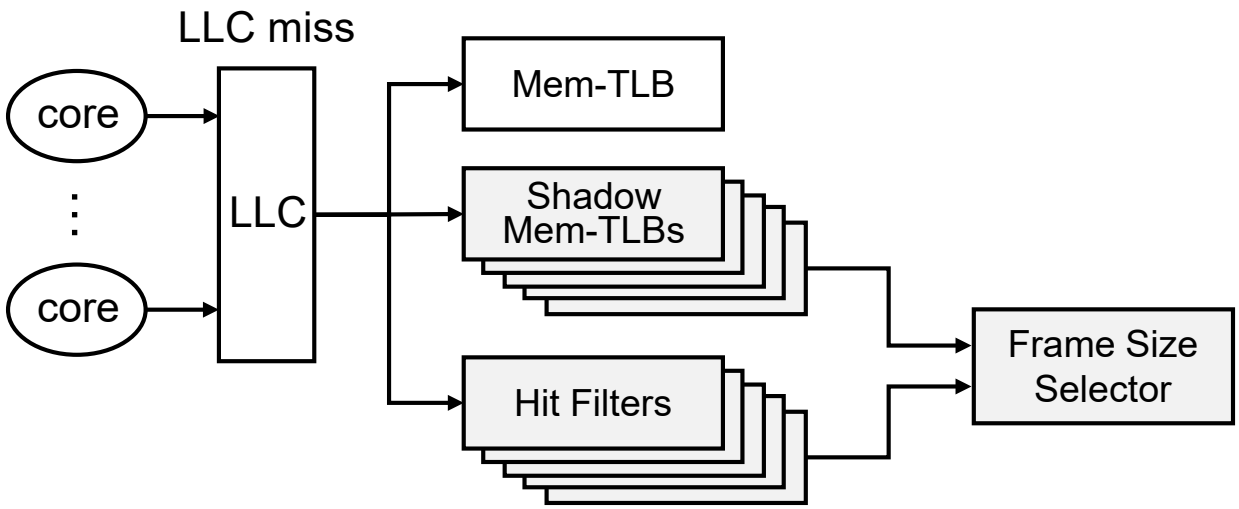
# Overview

- Flexible frame size among 5 candidates



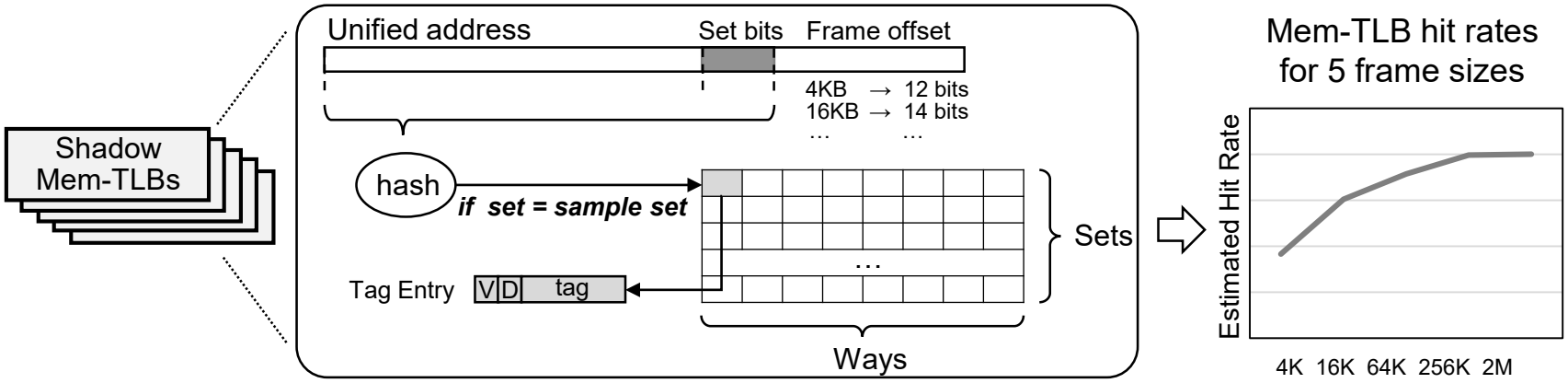
# Architecture

- Shadow mem-TLB
- Hit Filter
- Frame Size Selector



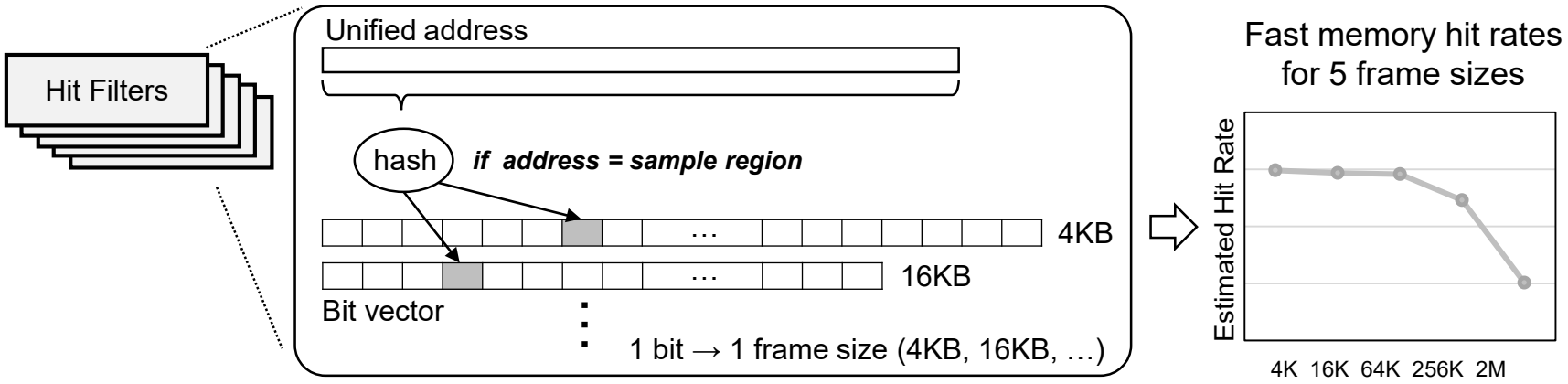
# Architecture: Shadow mem-TLB

- Estimates mem-TLB misses
- Negligible hardware overhead



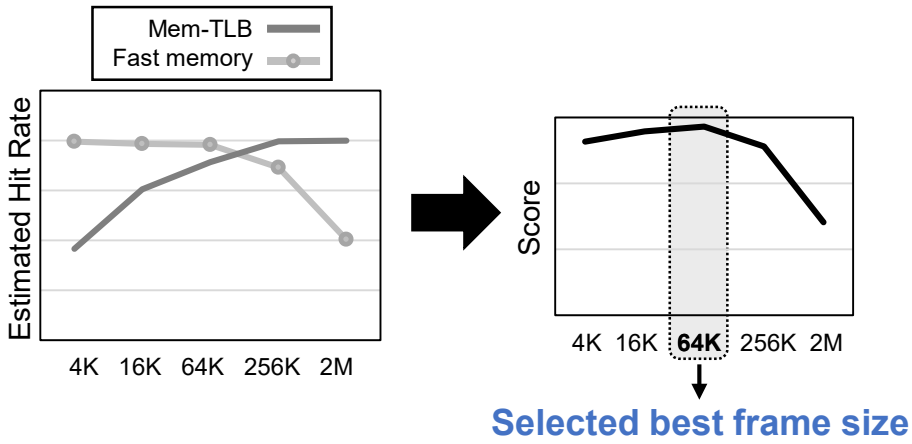
# Architecture: Hit Filter

- Estimates fast memory hit rates
- Frame sampling + bloom filter



# Architecture: Frame Size Selector

- Calculate score from estimated hit rates
  - Weighted sum of estimated hit rates
- Decide optimal frame size after an epoch



KAIST

# Evaluation

# Methodology

- Memory system: DDR4 (fast) – PCM (slow)
- Baseline: Conventional hybrid memory*
- Execution-driven simulation
  - ZSim + DRAMSim2
- 14 Memory-intensive benchmarks

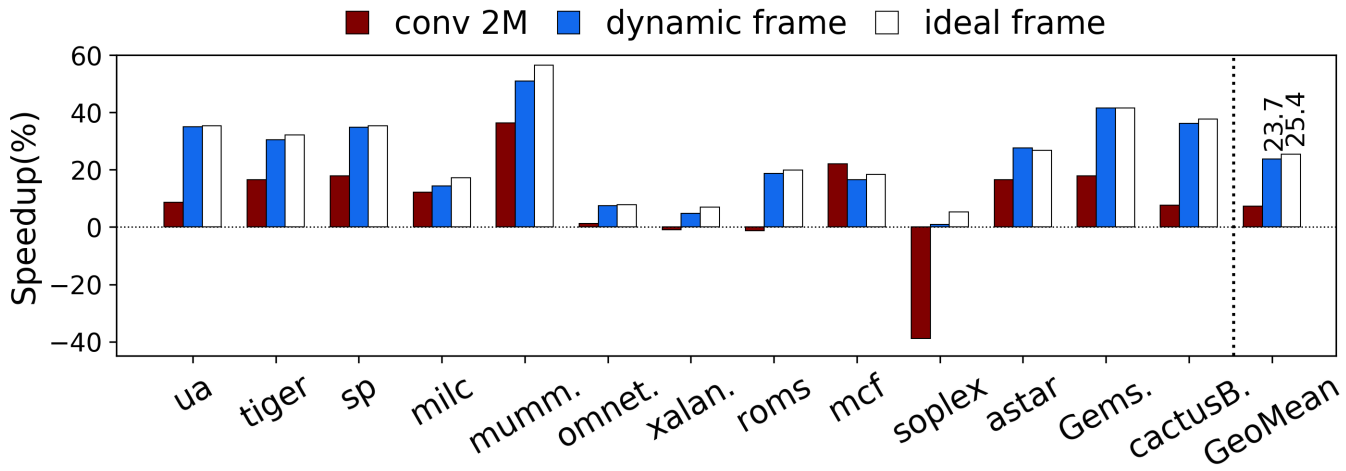
Simulation Parameters

| Component | Configuration |
|---|---|
| core-TLB | 1024/512 entries per core (conv/two-level)<br>4-way SA, miss latency 50 cycles |
| mem-TLB | 4096 entries, 8-way SA, miss latency 200 cycles |
| DRAM | 512MB, 8 channels, DDR4-1600<br>tCAS=11, tRCD=11, tRP=11, tRAS=28 |
| PCM | 4 channels, read/write latency = 150/300ns |

* Meswani et al., “Heterogeneous Memory Architectures: A HW/SW Approach for Mixing Die-stacked and Off-package Memories”, HPCA’15

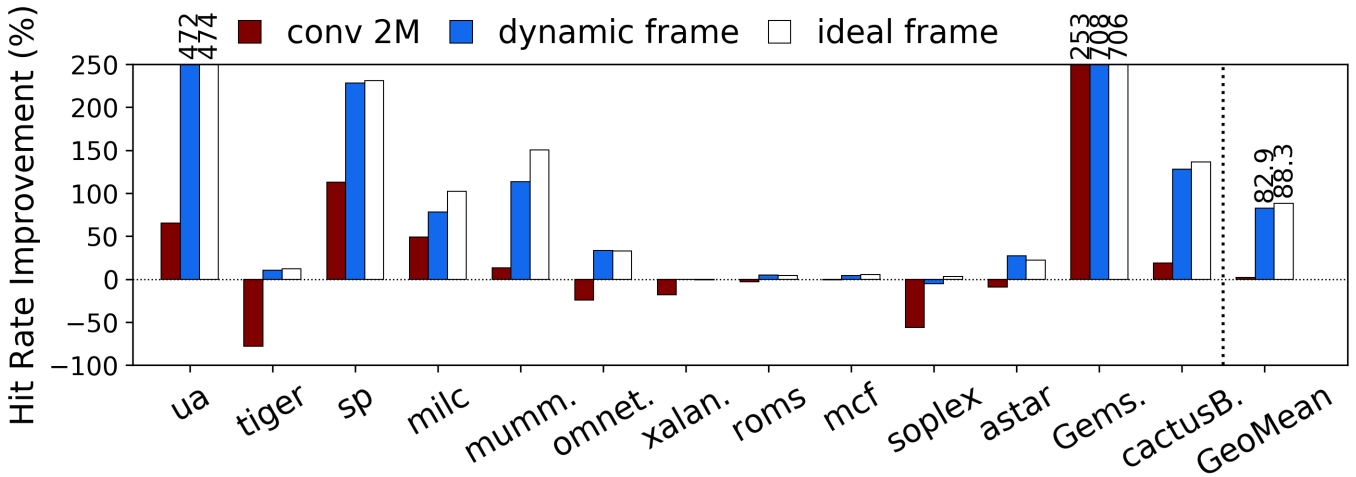
# Performance Evaluation

- Speedup vs. conventional, 4KB page: **+23.7%**
  - vs. conventional, 2MB huge page: **+15.3%**
  - vs. ideal frame size selection: **×0.98**



# Analysis

- Fast memory hit rate improvement
  - **+82.9%** of conventional, 4KB page
  - **×0.94** of ideal frame size selection



# More Results on Paper

- core-TLB MPMI
- mem-TLB MPMI
- Multi-class application performance
- Strict fairness

# Conclusion

- Naive virtualized hybrid memory is inefficient
  - Should handle conflicting objectives of page sizes
- Solution: HW-SW cooperative architecture
  - Two-level **decoupled** address translation
  - **Dynamic** frame size selection
- Shows significant performance improvement
  - vs. conventional: **+23.7% speedup**
  - vs. ideal: **×0.98 speedup**