Metadata Management for Large-Scale Hybrid Memory Architectures

Shunsuke Tsukada, Masayuki Sato, Kazuhiko Komatsu, Hiroaki Kobayashi
Tohoku University

Introduction

Requirements for main memory

- More bandwidth and shorter latency for high performance applications
- More memory capacity for HPC and big data applications

Both high performance and a large capacity are required for main memory.

Hybrid memory architectures (HMAs)

Combine two types of memories to realize both high performance and large capacity

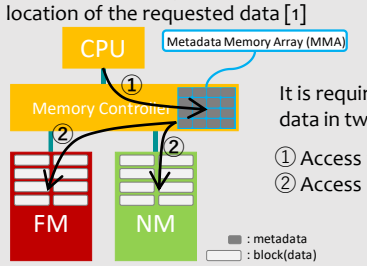
◇**Far memory (FM)**
Large capacity but **low** performance to store **infrequently**-used data

◇**Near memory (NM)**
High performance but **small** capacity to store **frequently**-used data

Management of data between NM and FM is essential to keep many frequently-used data in NM.

Metadata

Records the management of information such as location of the requested data [1]



It is required to access data in two steps.

① Access metadata
② Access FM/NM

[1] Chia Chen Chou, et al. Cameo: A two-level memory organization with capacity of main memory and flexibility of hardware-managed cache. In 2014 47th Annual IEEE/ACM International Symposium on Microarchitecture, pp. 1–12, 2014.

Problem : A Significant Increase in Capacity for Metadata on an HMA

Estimation of the size of the MMA in the future

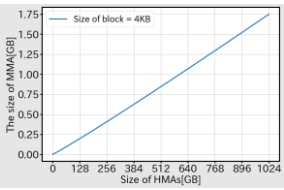
$$(\textit{The size of MMA}) = (N_{metadata}) \times (S_{metadata})[Bytes]$$

The number of metadata

$$N_{metadata} = \frac{(\textit{Size of HMAs})}{(\textit{Size of block})}$$

The size of metadata

$$S_{metadata} = 2 \times log_2 (N_{metadata})[Bits]$$
$$= 2 \times \frac{1}{8} log_2 (N_{metadata})[Bytes]$$



- **The size of the MMA increases as the size of the HMA increases.**

It is not practical to store metadata only on a chip in the future.

Qualitative Analysis of Metadata Access Pattern

In-memory metadata with metadata memory array

How to solve the problem

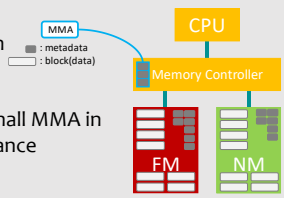
- Manage metadata by distributing them to the FM/NM and the MMA

Approach

- Stores only necessary metadata in a small MMA in a memory controller to avoid performance degradation caused by MMA miss

Related research

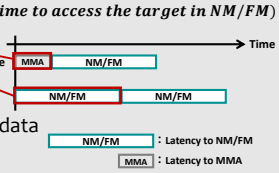
- Stores only metadata of frequently accessed data without metadata prediction [2]



Behavior of metadata accesses

Equation of the time to access data

$$(\textit{Time to access data}) = (\textit{Time to refer metadata}) + (\textit{Time to access the target in NM/FM})$$



MMA hit : Accesses metadata in MMA
MMA miss : Accesses metadata in NM/FM
Increases the access latency to the data

To avoid increasing the access latency, it is important to save necessary metadata in the MMA in advance.

To find necessary metadata, it is required to analyze metadata access patterns.

[2] Evangelos Vasilakis, et al. Hybrid2: Combining caching and migration in hybrid memory systems. In 2020 IEEE International Symposium on High Performance Computer Architecture (HPCA), pp. 649–662. IEEE, 2020.

Quantitative Analysis of Block Access Patterns

Purpose

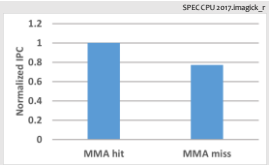
To analyze block access patterns for efficient control of metadata

Experimental environment

Obtain an address trace based on block address using simulators (Gem5 and NVMain) and analyze the accesses to metadata

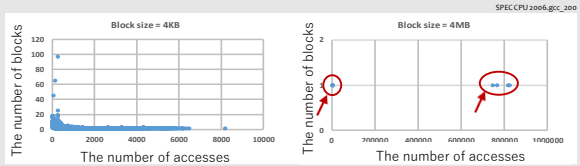
- [Benchmark] SPEC CPU 2006, 2017
- [Cache size] L1I:64KB, L1D:64KB, L2:512KB, L3:2MB
- [Latency] M:2cycles, NM:32cycles, FM:68cycles
- If data is stored in NM/FM, NM/FM is accessed to refer metadata.

Experimental results and discussions



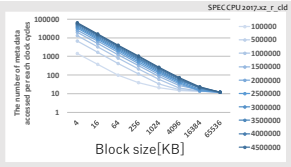
- ***The MMA miss decreases IPC by accessing metadata in NM and/or FM.***

It is important for performance to predict necessary metadata and store the predicted metadata in the MMA in advance.

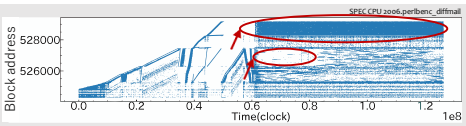


- ***When the block size is large, the number of blocks is limited.***

By using the large block size, blocks can be categorized by the number of accesses.

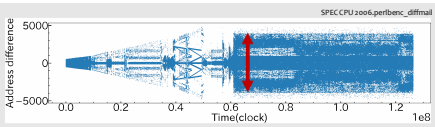


By decreasing time interval to count the accesses and/or increasing the size of block, the number of metadata decreases.



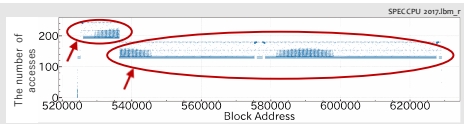
- ***Frequently accessed data or its neighbors continue to be frequently accessed and infrequently accessed data will not be accessed.***

By using access frequency, It is possible to find high priority metadata.



- ***The address between currently accessed data and next accessed data is within a certain range.***

By using the past access pattern, it is possible to predict the future access range.



- ***Neighborhood of data have the similar number of accesses.***

By using the block address, It is possible to predict the number of accesses.

Conclusions & Future Work

- Large-scale HMAs need more metadata, and the capacity for metadata should be mitigated.
- This research analyzes block access pattern to realize the in-memory metadata management with metadata caching.
- The experimental results show the behavior of metadata accesses to predict metadata in the HMAs.
- In the future, it is planned to develop the metadata prediction mechanism by using the metrics and implement it.

Acknowledgements

This research was partially supported by MEXT Next Generation High-Performance Computing Infrastructures and Applications R&D Program, entitled "R&D of A Quantum-Annealing-Assisted Next Generation HPC Infrastructure and its Applications," by Grants-in-Aid for Scientific Research(A) #19H01095, and by Grants-in-Aid for Early-Career Scientists #19K20232.