中国科学技术大学
University of Science and Technology of China

A Page-Based Storage Framework for Phase Change Memory

Peiquan Jin, Zhangling Wu, Xiaoliang Wang, Xingjun Hao, Lihua Yue

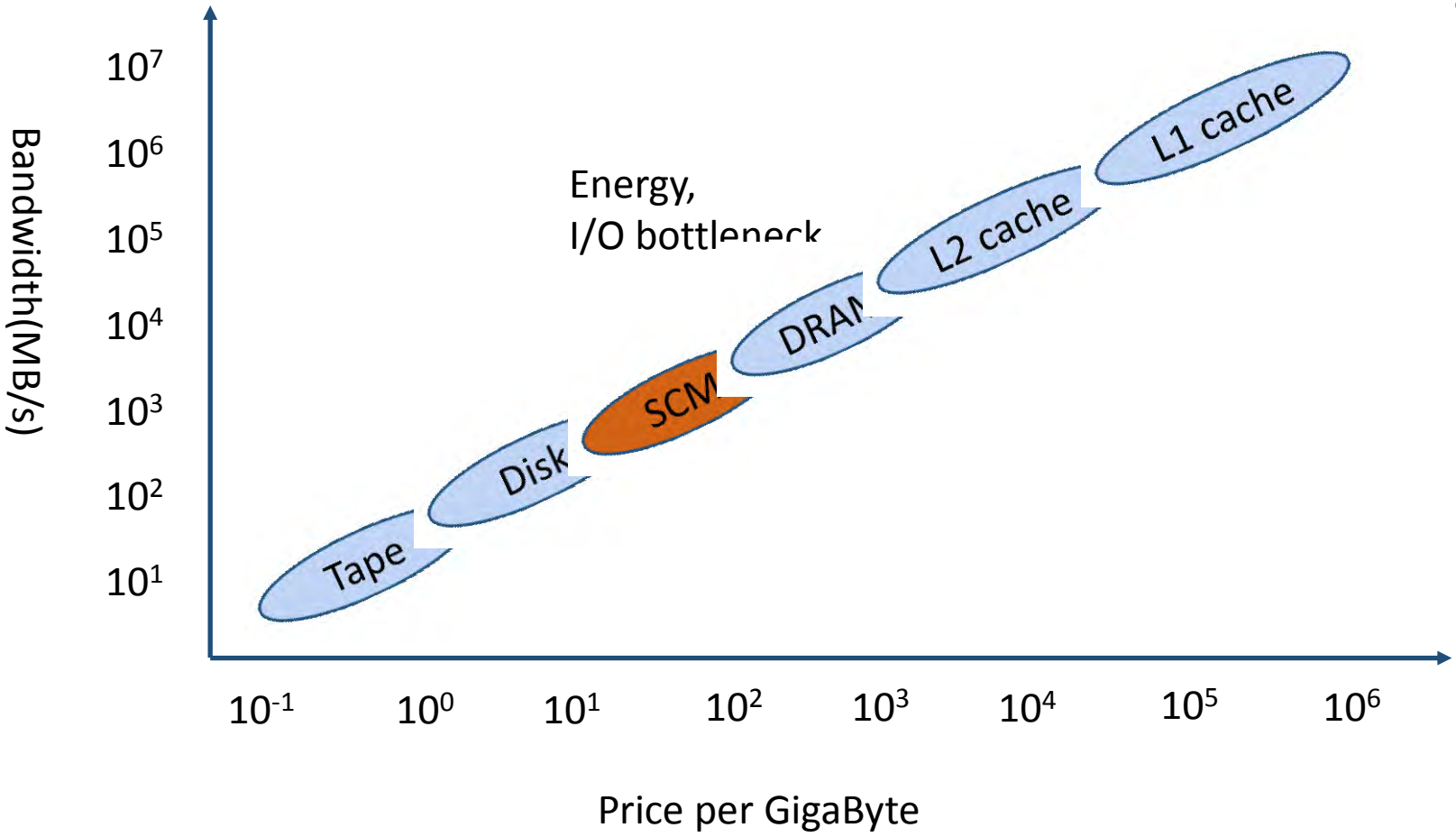
University of Science and Technology of China

2017.5.19

Outline

- Background
- Related work
- Page-Based Storage Framework for PCM
- Experimental results
- Conclusions

Background



- **SCM**: A new class of data storage/memory devices
  - PCM, MRAM, Flash, etc.

Has received much attention from both academia and industries in recent years

Background

PCM VS DRAM VS Flash

Parameters	DRAM	NAND Flash	NOR Flash	PCM
Density	1X	4X	0.25X	2X - 4X
Read latency	60ns	25us	300ns	200 - 300ns
Write speed	≈1Gps	2.4MB/s	0.5MB/s	≈100MB/s
Write endurance	N/A	10^4	10^4	10^6 - 10^8
Nonvolatile	NO	Yes	Yes	Yes

- PCM VS DRAM
  - Advantages
    - Good scalability
    - Non-volatile, no idle power
    - Similar read latency
    - Reliability
  - Disadvantages
    - Slow write speed
    - High read/write energy consumption
    - Endurance problem
- PCM vs Flash
  - Advantage
    - Byte-addressable
    - in-place update
    - Higher write endurance
  - Disadvantage
    - Cost

Related work

PCM write endurance

Wear leveling

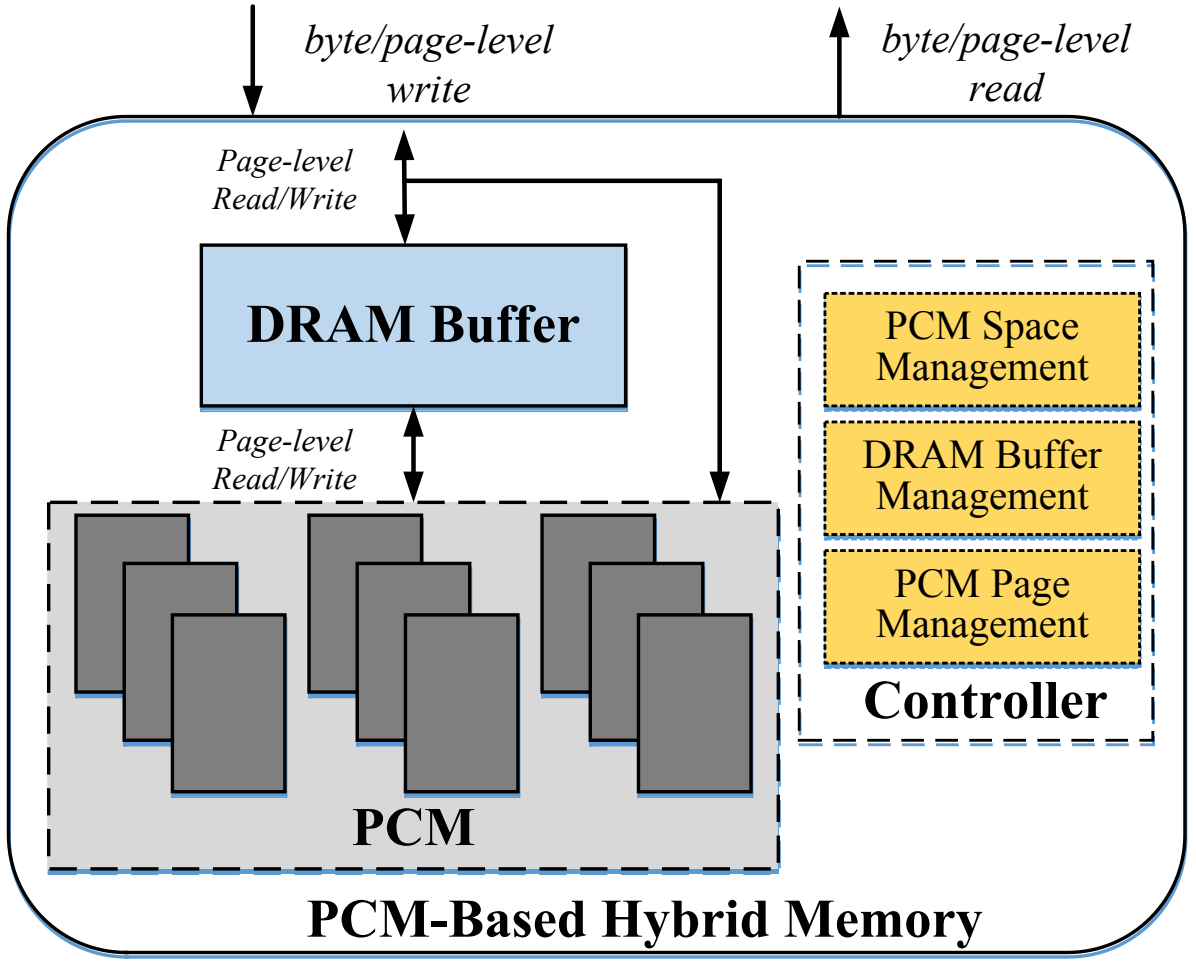
- deterministic algorithms:
  - adaptive data swapping and shifting (park et al. 2014)
  - row shifting and segment swapping (Zhou et al. 2009)
  - bucket-based WL algorithm [Chen et al. 2012]
- Random algorithms (Ferreira et al. 2010)
- PTL (Choi et al. 2013)

Write reduction

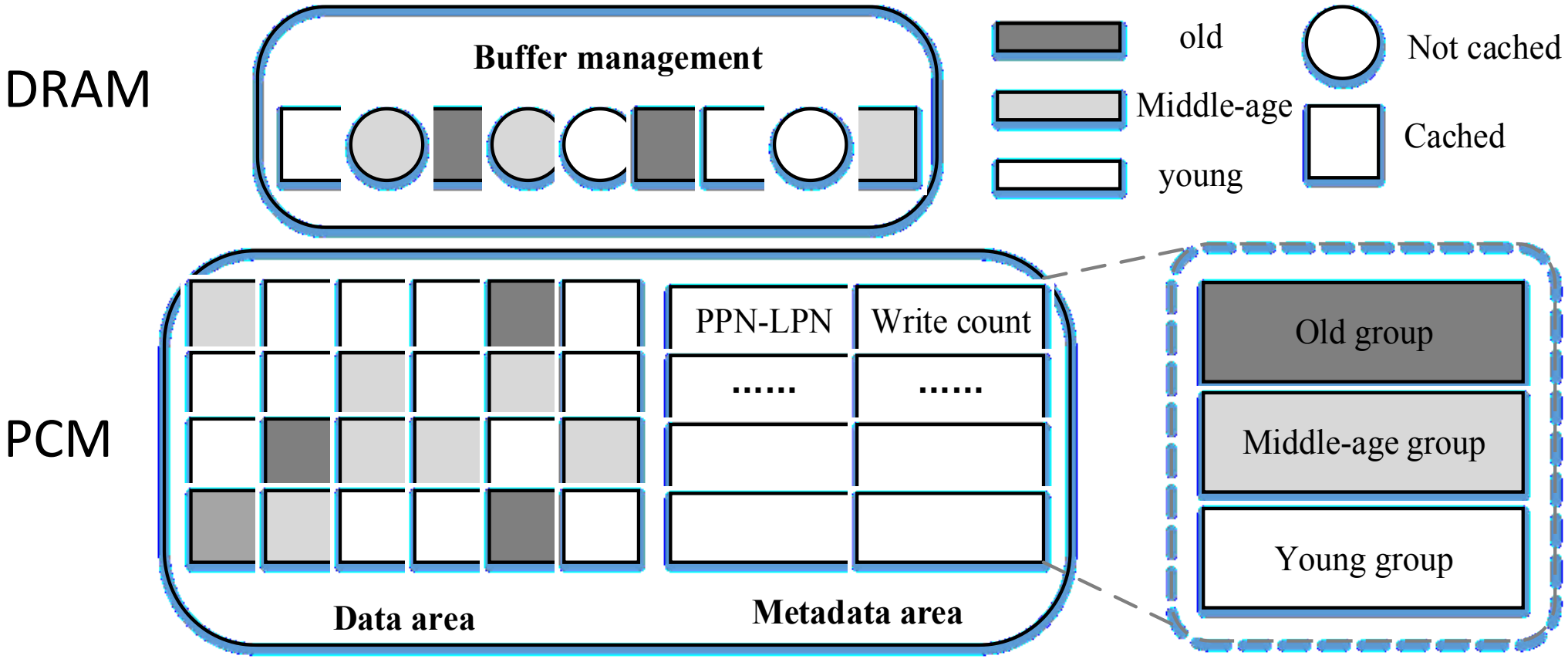
- Hybrid storage architecture (Qureshi et al. 2009(b))
- Buffer management
  - CLOCK-DWF(Lee et al. 2011)
  - Lazy-write organization (Qureshi et al. 2009(b))

Overall Architecture

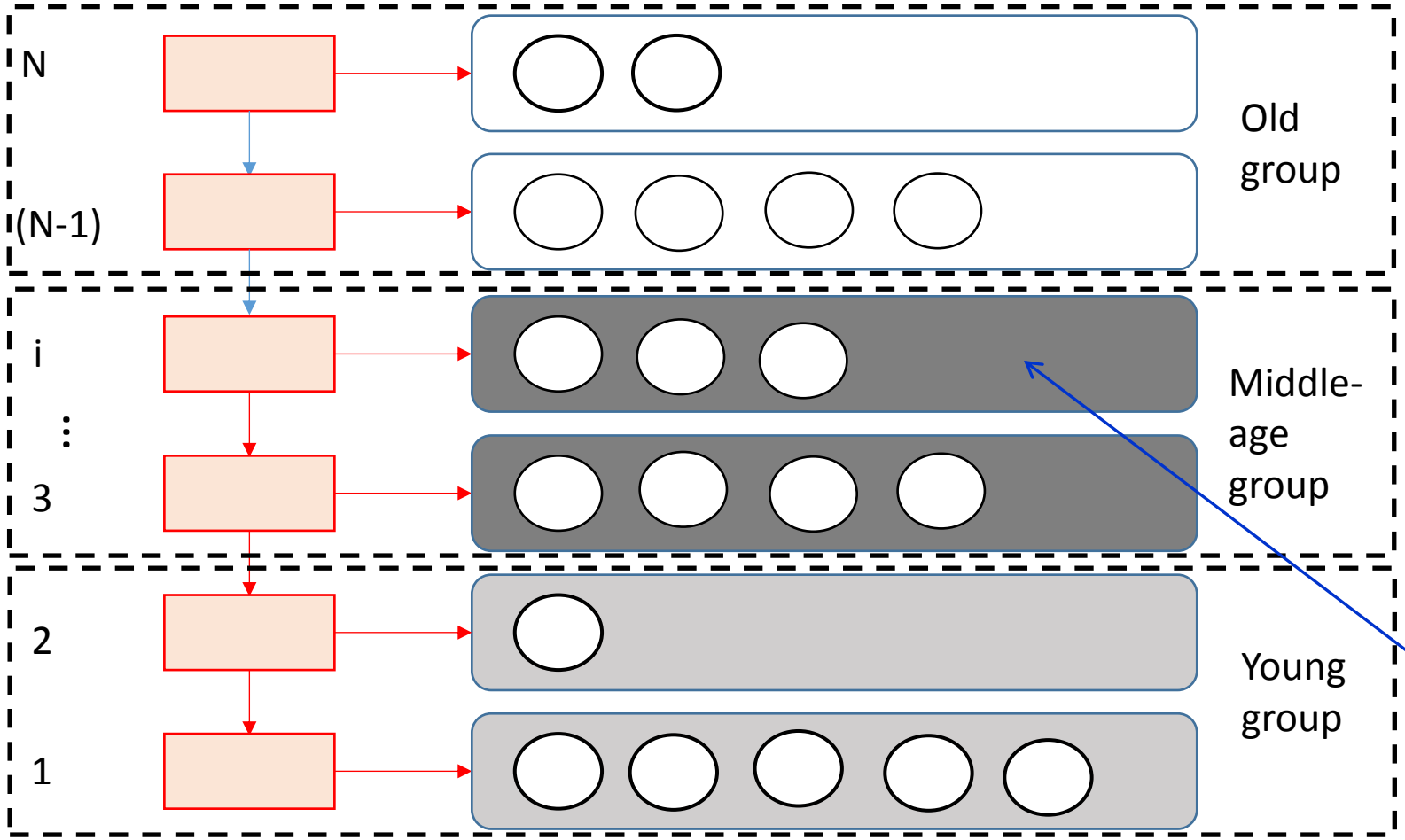
- **Purpose**
  - lengthen the lifetime of PCM
- **Method**
  - DRAM buffer: reduce the write operations to PCM
  - Efficient space management: wear leveling



Memory Structure



Space Management



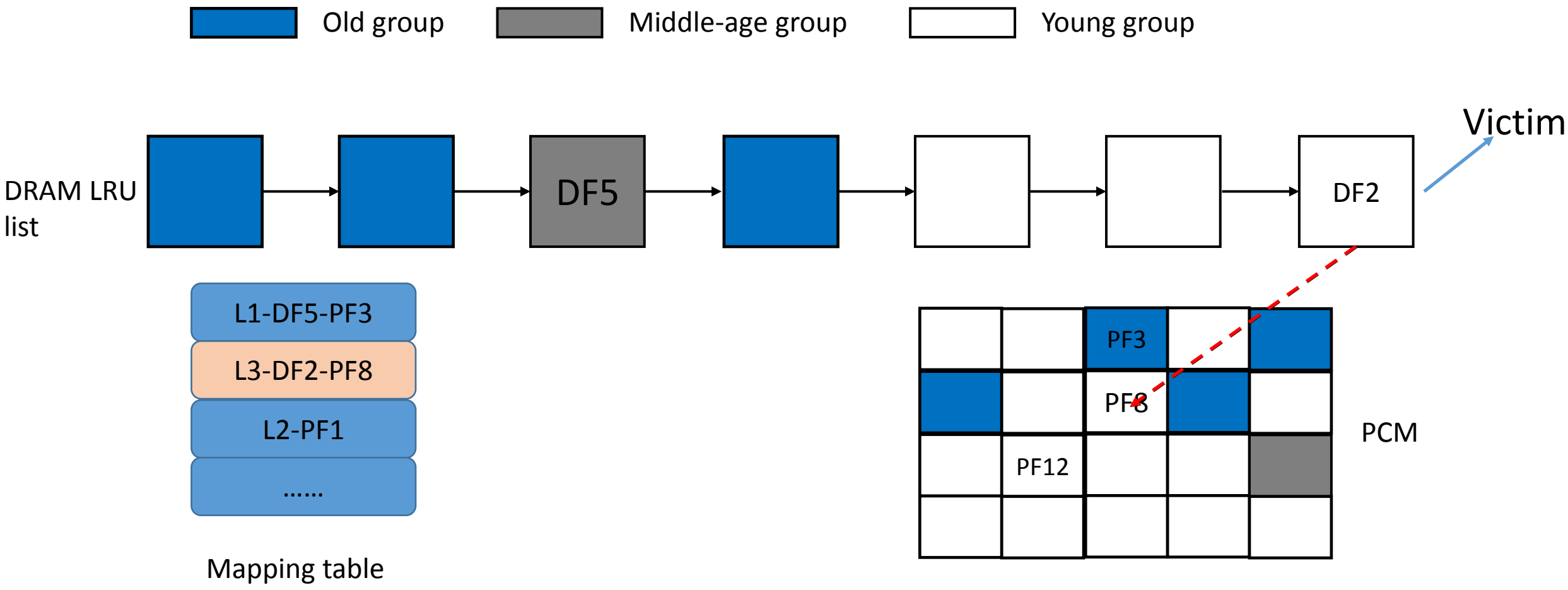
- **Dual Dynamic Bucket Lists:**
  - Free dynamic bucket list for free pages
  - Allocated dynamic bucket list for allocated pages
- *w*: the basic age value of a page.

Updated *n* times
$((i-1)*w \leq n < i*w)$

Group Classification

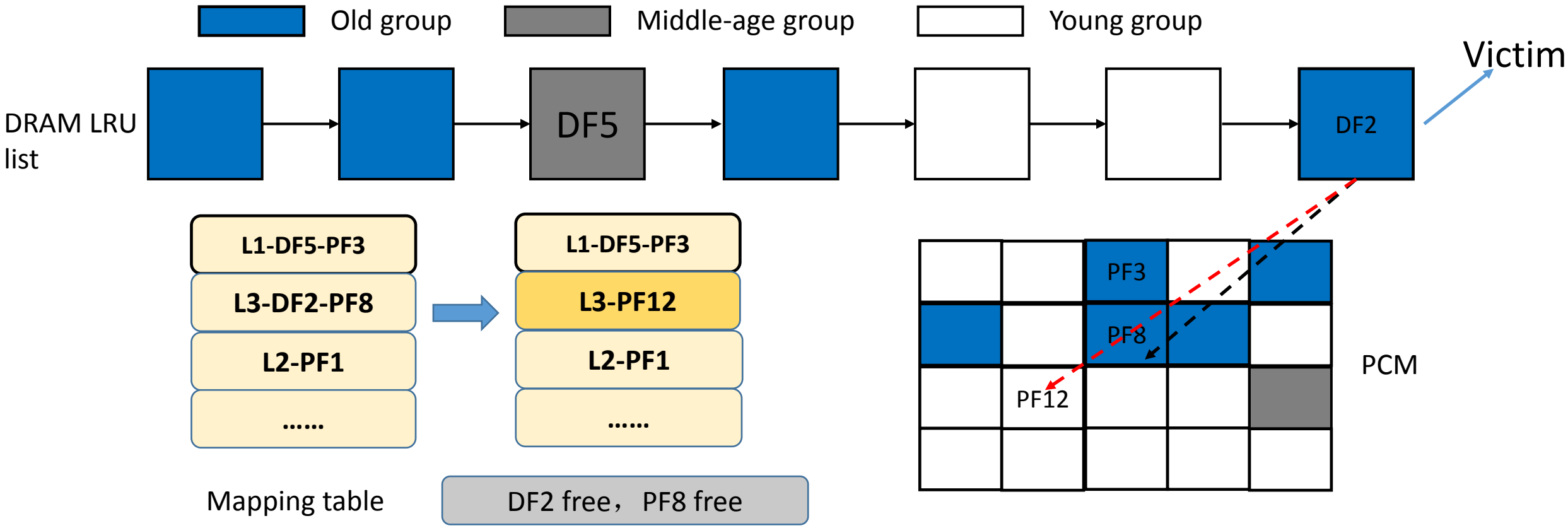
- $\boldsymbol{average\ write\ count} = \frac{total\ write\ count}{total\ pages}$ *(the ideal situation of wear leveling)*
- $\boldsymbol{p_{wear}}$**: write count of page *p***
  - If $|p_{wear} - average\ write\ count| \leq TH$, *p* belongs to middle-age group
  - If $p_{wear} - average\ write\ count > TH$, *p* belongs to old group
  - If $p_{wear} - average\ write\ count < TH$, *p* belongs to young group
- **Page management**
  - Page allocation:
    (1) select the pages which is in the **youngest bucket** to allocate.
    (2) **migrates** the data in the old pages to new places
  - **In-place updating** and **out-of-place updating**

In-place updating

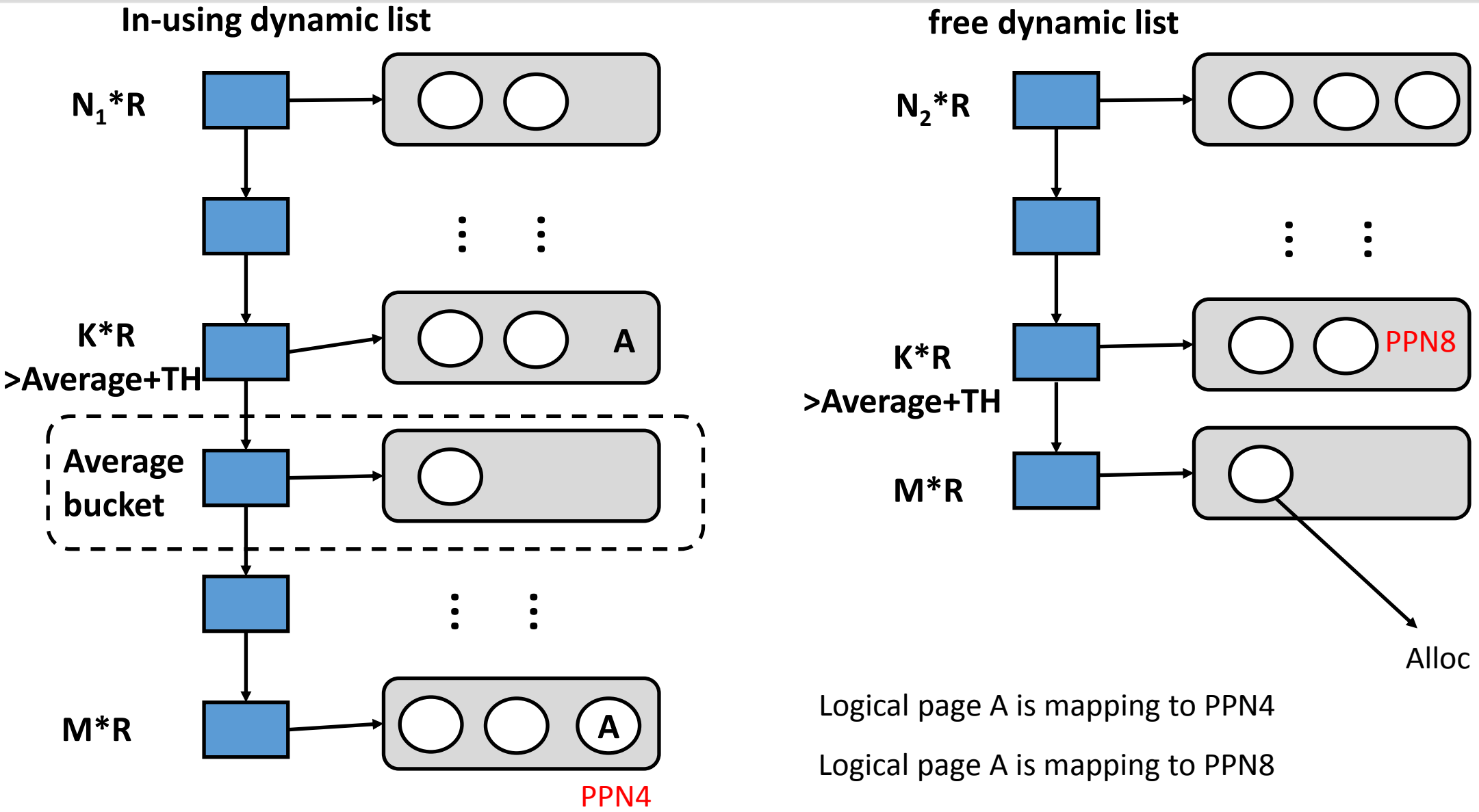


Out-of-place updating

- **New younger free page will be allocated for an old page updated**



Example of page migration

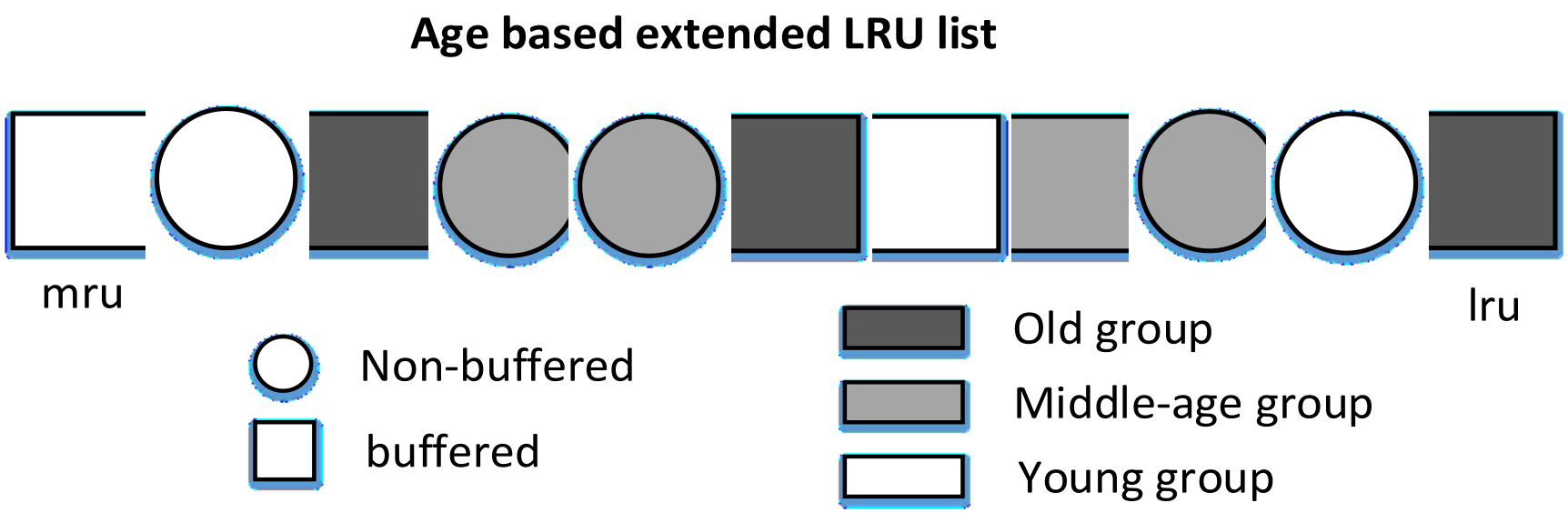


Logical page A is mapping to PPN4

Logical page A is mapping to PPN8

DRAM Buffer Management

- **Basic structure: A-eLRU**

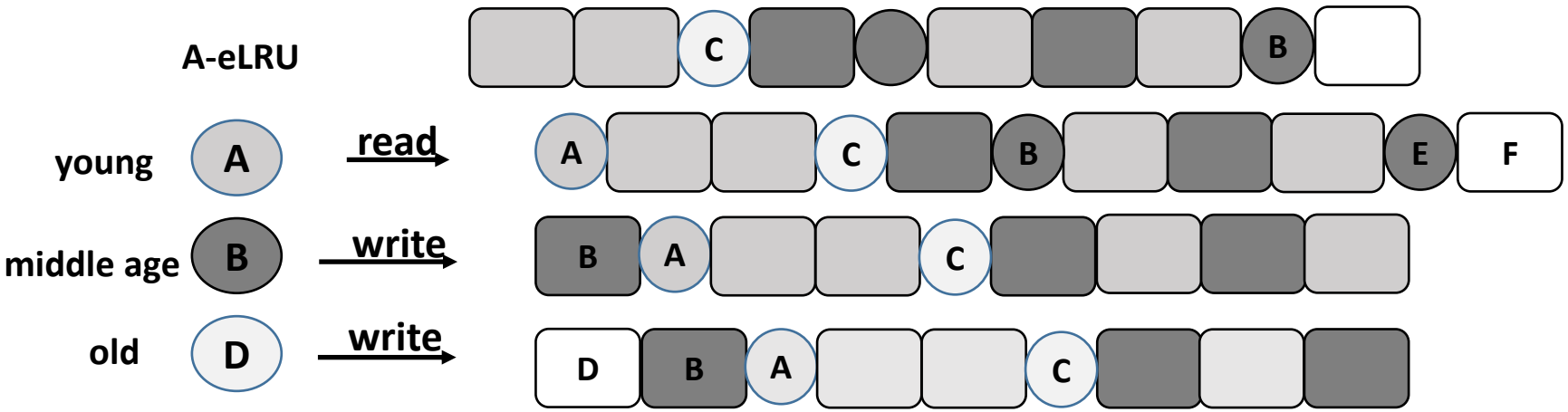


DRAM Buffer Management

- **Buffer strategy: ALC (Age-base Lazy Caching)**
  - Avoid the “wear-out” of old pages
  - Avoid buffering cold data
- **Buffer replacement**
  - Always select the LRU position in the A-eLRU list as the victim

DRAM Buffer Management

- **An example**



Experimental settings

- **Competitive approaches:**
  - **(1) random swapping** [9], which swaps the page to be written with a randomly selected page for every 512 write operations to PCM.
  - **(2) the bucket-based WL algorithm** [8], which uses 500 buckets to maintain allocated and free pages separately; the write count difference of pages in the same buckets is 10.
  - **(3) PTL** [7], in which all pages are updated out-of-place. The original approaches of the three competitors did not use a buffer. To be fair, we implement a buffer for these methods and use the classic LRU as the cache replacement strategy. The page size of both buffer and PCM is set to 4 KB.

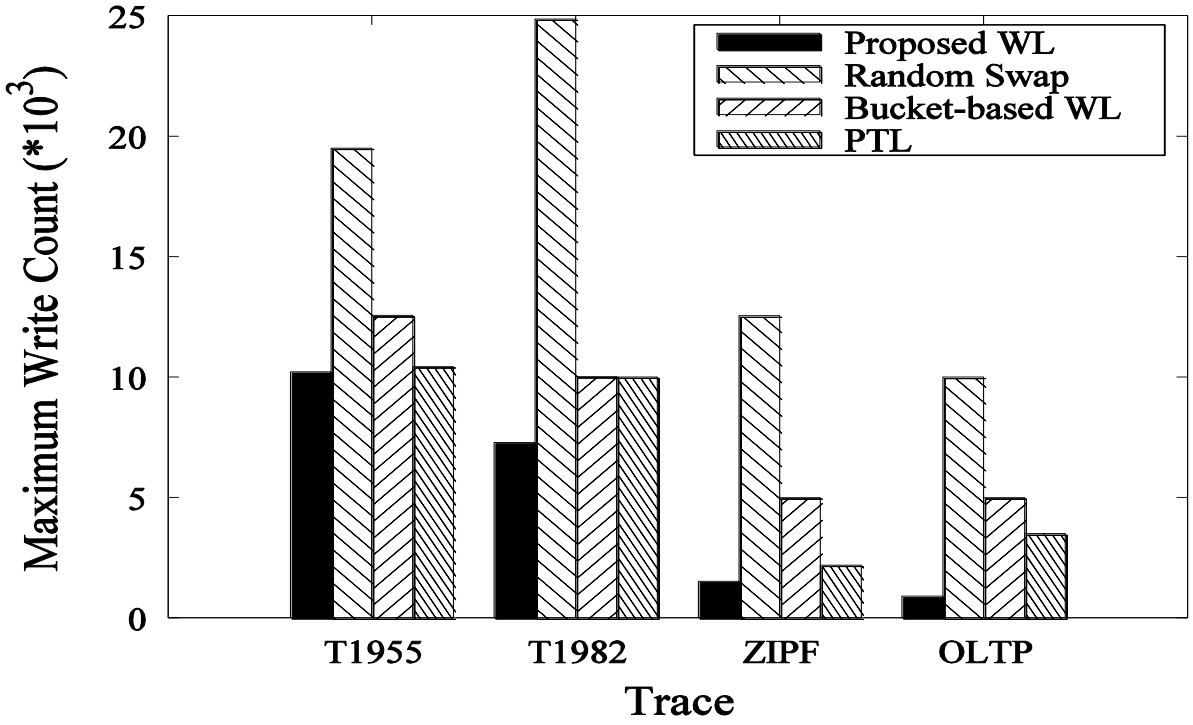
Experimental settings

Trace	Memory Footprint	Read/Write Ratio	Locality	Total Requests
T1982	10,000	10% / 90%	80% / 20%	300,000
T1955	10,000	10% / 90%	50% / 50%	300,000
OLTP	51,880	77% / 23%	~	607,390
ZIPF	47,023	51% / 49%	~	500,000

Parameters	Value	
	Synthetic traces	Real traces
PCM size	12000 pages	52000 pages
DRAM buffer size	1000 pages	
w	10	
TH	30	

Experimental results

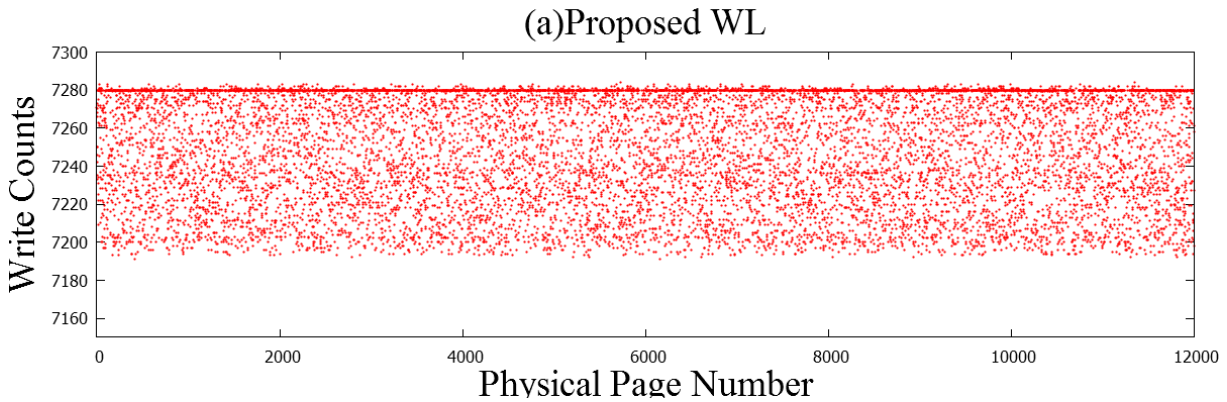
- **Maximum write count**

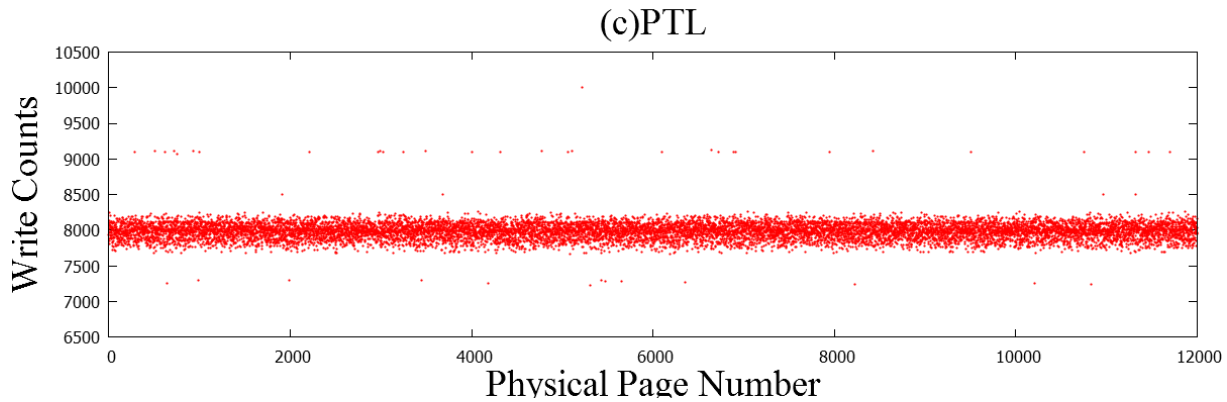


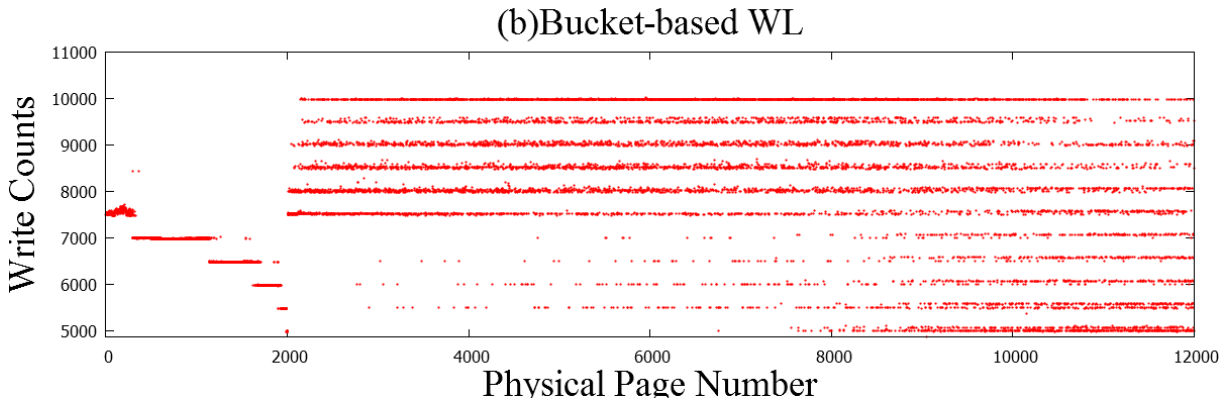
The maximum write count of our proposal is much less than that of competitor algorithms
PCM with the proposed algorithms can tolerate more writes before being worn out

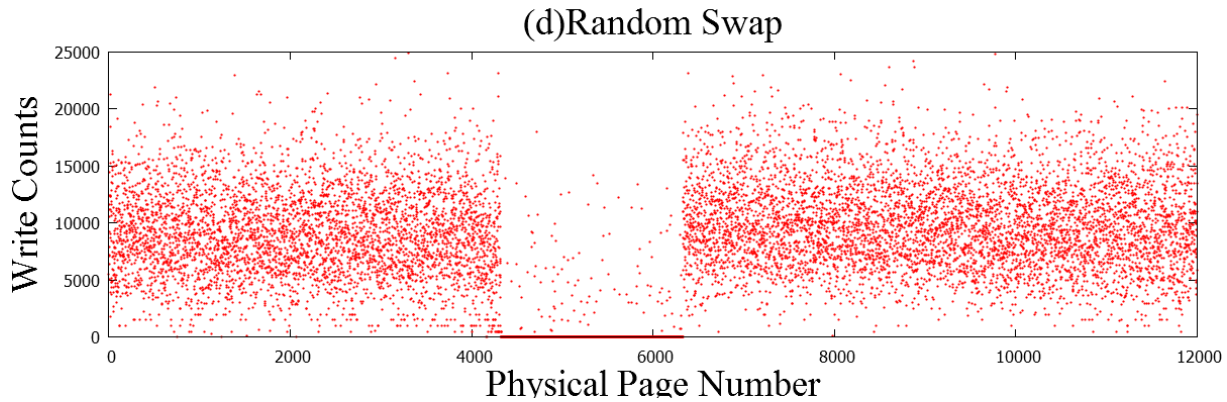
Experimental results

- **Distribution of write count after applying T1982**









our proposal narrows the deviation of write operations. All the pages have been written at least 7190 times and the maximum number of write operations is only 7284.

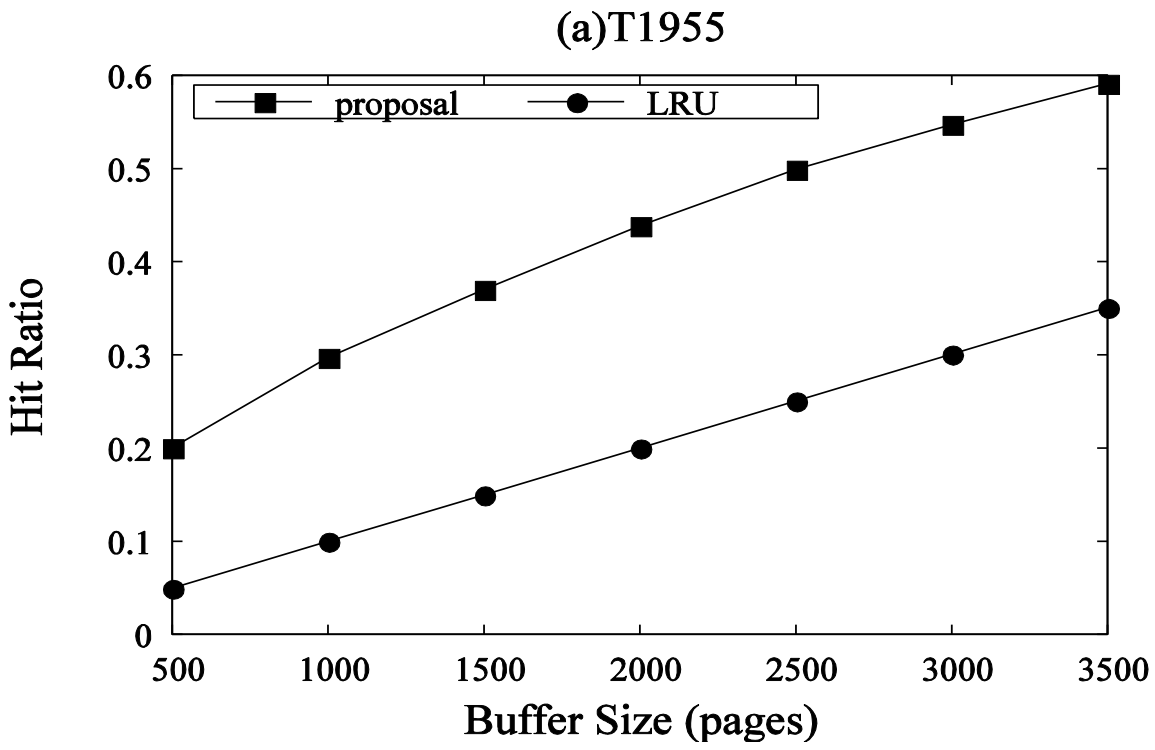
Experimental result s

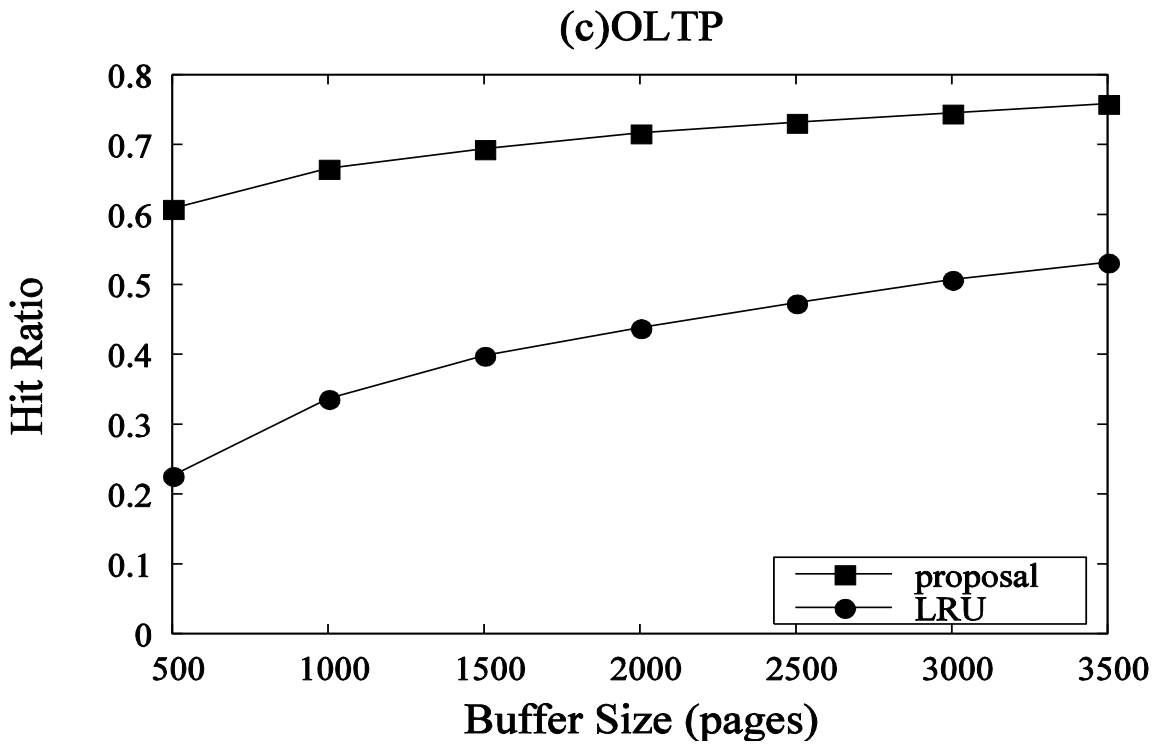
- **Lifetime of PCM**
  - We set the write limitation of a PCM page to be 10000
  - We set the capacity of PCM to be 12000 pages. In the ideal case, 12000 pages can withstand 12000*10000 writes.
  - In particular, it reaches about 99.5% and 96.9% of the ideal write count over T1955 and T1982 respectively.

Policies	Write count of wearing out PCM	
	T1955	T1982
Proposal	119,511,349	116,328,780
PTL	117,628,266	95,740,849
Bucket-based WL	94,416,434	86,691,668
Random swap	62,941,008	26,001,132

Experimental results

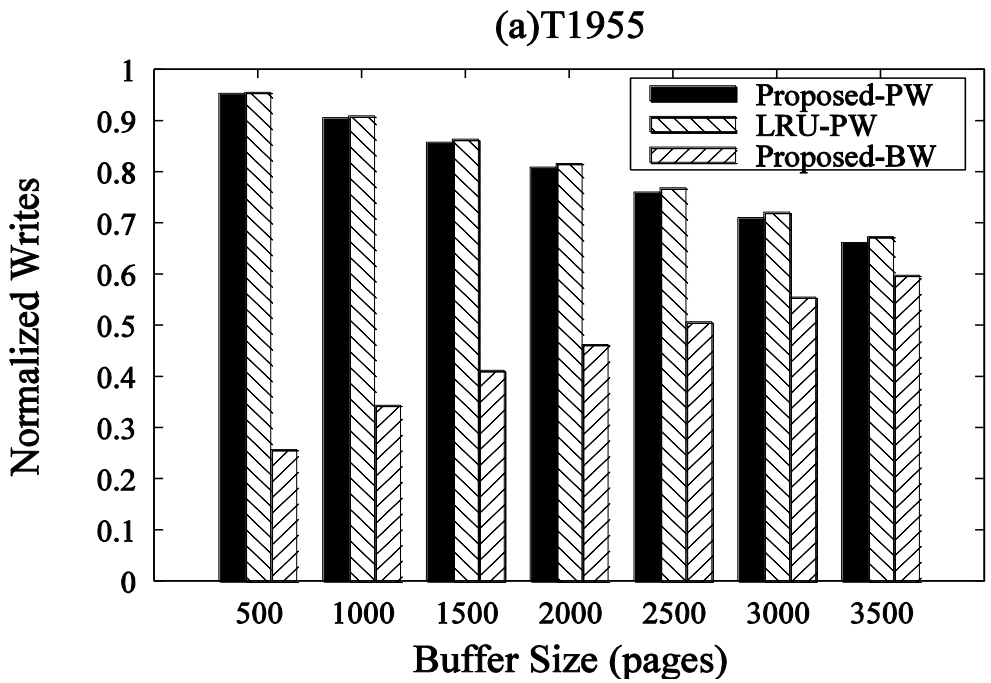
- **Hit ratio of the DRAM buffer**

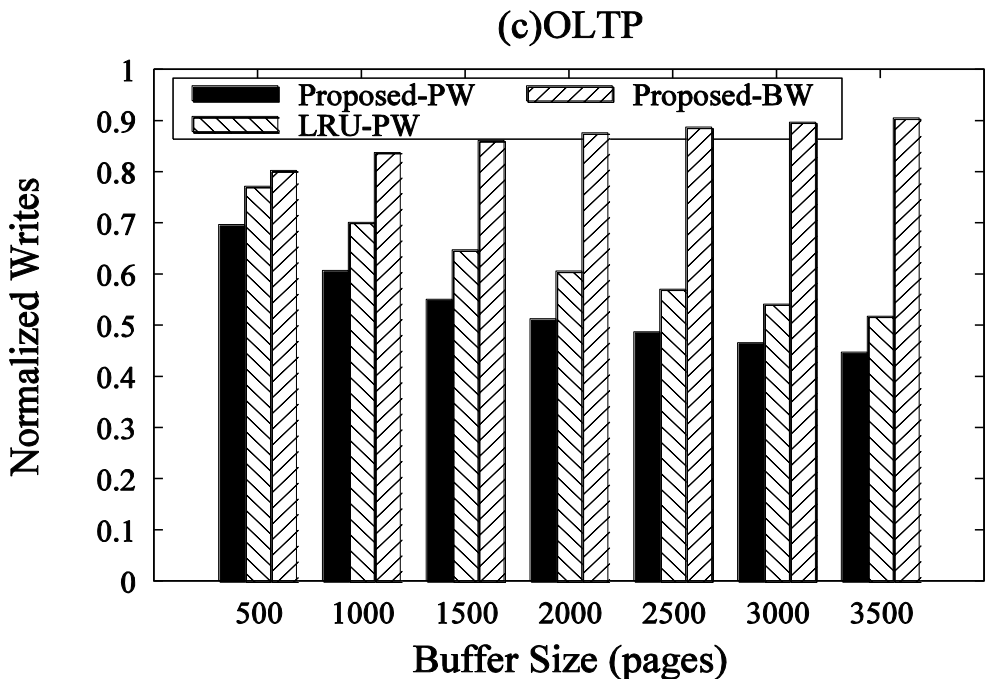




Experimental result s

- **Impact of buffer management on PCM writes**





Conclusions

- We propose a new structure, called ***Dual Dynamic Bucket Lists***, to organize the spaces of the hybrid memory including DRAM and PCM.
- We use a small DRAM buffer for PCM to improve the endurance of PCM. Particularly, we propose the ***A-eLRU*** structure and an ***Age-based Lazy Caching (ALC)*** policy for the management of the buffer.
- We propose a new page allocation algorithm for PCM. It incorporates page migration and page swapping to reduce PCM writes.
- Extensive experiments over both synthetic and real traces show that our proposal outperforms the compared methods in terms of various metrics.

Thank you!