Sampling-Based Garbage Collection Metadata Management for Flash-based Storage

Biplob Debnath, Srinivasan Krishnan, Weijun Xiao

David J. Lilja, David H.C. Du

UNIVERSITY OF MINNESOTA

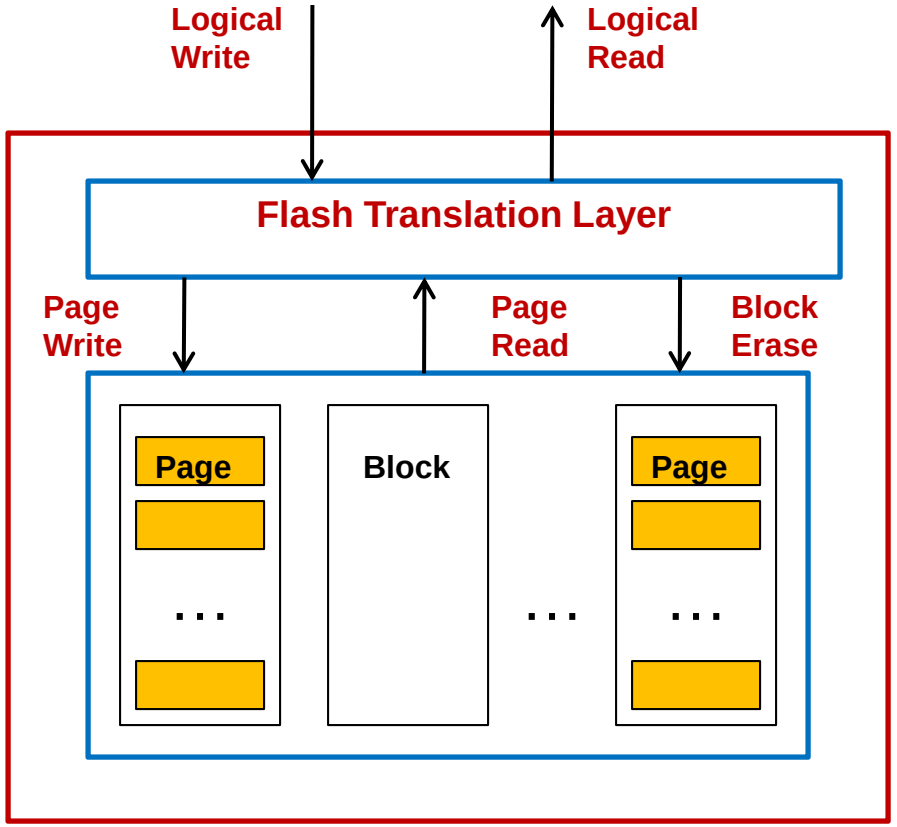
Flash-based Solid State Disk (SSD)

- Solid State Disks
  - Acts like a virtual HDD
  - NAND Flash-based
  - Faster read performance
  - Good sequential write performance

- Read/write in **page** units
- Erase in block units
- Must erase a **block** before write
- Typical **block** = **128K**; **page** = **2K**

- **Read** latency **25** microseconds
- **Write** latency **200** microseconds
- **Erase** latency **1500** microseconds

- Limited number of erases per block



NAND Flash-based SSD

Two Main SRAM Consumers

- Page Addressing Metadata
  - Most of current research focus to reduce SRAM space
- Garbage Collection Metadata
  - **Main focus of this paper**
  - Various garbage collection algorithms exist
  - To implement them
    - Per block metadata needed: utilization, age , erase count
    - Need a in-memory priority queue for faster access

Garbage Collection Metadata

- Need *O(N)* space to implement priority queue
  - *N* = SSD capacity in total no. of blocks
- When SSD capacity scales to bigger size
  - *N* also scales to larger
- In this case, due to RAM scarcity
  - We cannot implement priority queue for all *N* blocks.

Sampling-based Approach

- **Our Goal**
  - Emulate existing garbage collection algorithms in a small amount of SRAM

- **Use sampling-based approximation**
  - Memory requirement is fixed
  - Saves CPU processing time
  - Performance needs to be as good as “no sampling” approach

Illustration (N = 5, M = 2)

Score(j) = min $_j$ (erase_count (j))

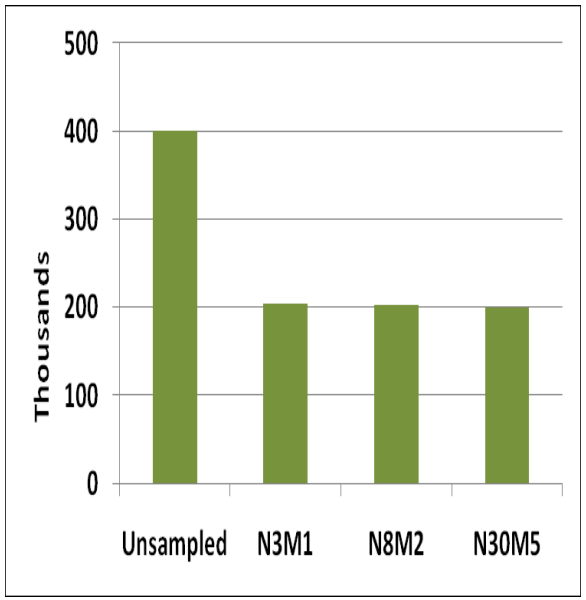
Sampling-based Algorithm

- If (***Eviction Needed***)
  - If (***First Iteration***)
    - Draw *N fresh* samples
  - **Else // subsequent iterations**
    - Draw *N-M fresh samples*
  - Select a victim from these *N* samples
  - Remove *N-M-1 bad* samples
    - which are less likely to be selected to be a victim in the next iteration

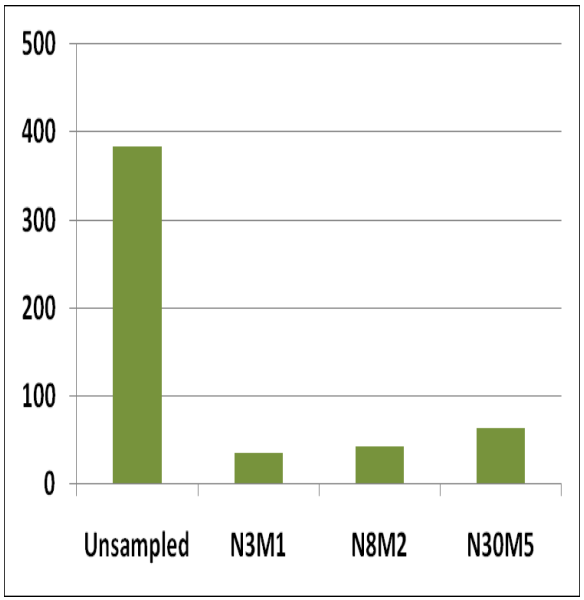
Experimental Setup

- DiskSim SSD Simulator
- Page Mapping Scheme: DFTL
- Traces
  - Financial-1
  - Financial-2
  - Microsoft Cambridge Trace
- Three settings
  - N = 3, M = 1; N = 8, M = 2; N = 30, M = 5

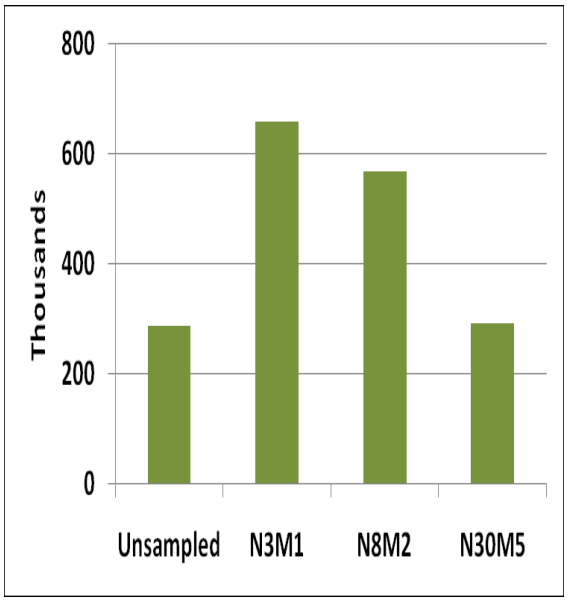
Financial-1



Variance of Erase counts



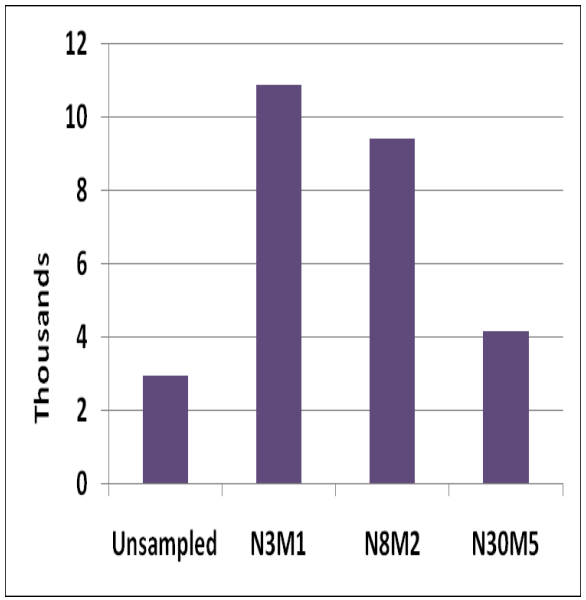
Max Erase count



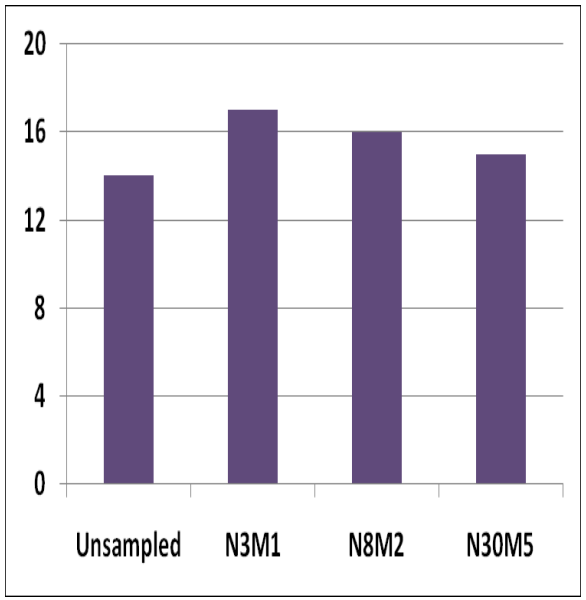
GC Overhead

- Greedy Clean scheme
  - Selects the block with the largest number of invalid pages as a victim

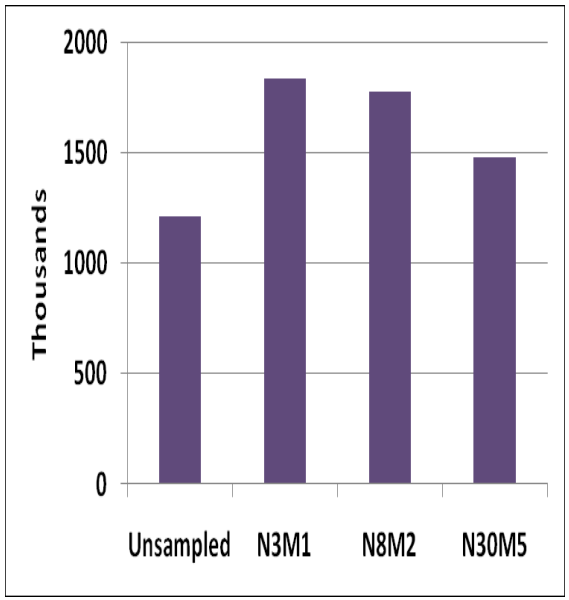
Financial-1



Variance of Erase counts



Max Erase count



GC Overhead

- Greedy Wear scheme
  - Selects the block with the least erase count as a victim

Summary

- Preliminary Results shows that
  - **30** samples ***(N = 30, M = 5)*** are good enough to emulate existing garbage collection algorithms

- Need more experiments and analysis to establish that sampling-based approximation is a good idea

Thank You!

Comments / Questions?

Back-up Slides

Metadata Size on SRAM (1GB)

Total blocks	**8192**
Erase count metadata	**8192*4 = 32 KB**
Other metadata	**8192*4 = 32 KB**
Total metadata size	**64 KB**

For 1 TB SSD Requires, 64 MB !!!!

Sampling overhead

- Depends on *N-M*
  - i.e., the number of new samples drawn in each iteration