Tao CAI, Pengfei GAO, Dejiao NIU, Yueming MA, Tianle LEI, Jianfei DAI, 2023.
NEHASH: high-concurrency extendible hashing for non-volatile memory.
Frontiers of Information Technology & Electronic Engineering, 24(5):703-715.
https://doi.org/10.1631/FITEE.2200462

NEHASH: high-concurrency extendible hashing for non-volatile memory

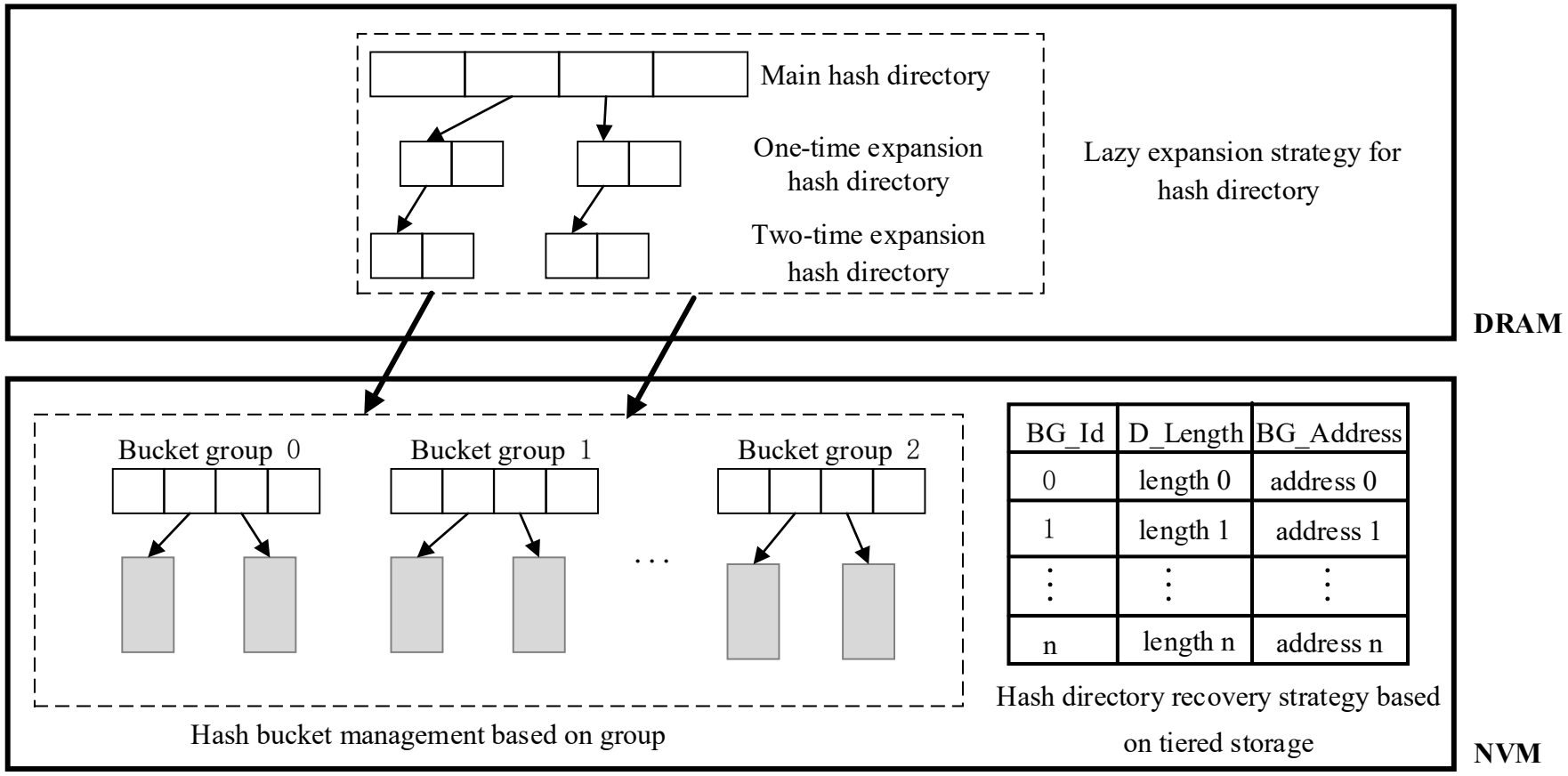
Key words: Extendible hashing; Non-volatile memory (NVM); High concurrency

Corresponding authors: Tao CAI, Pengfei GAO
E-mail: caitao@ujs.edu.cn; 1306943800@qq.com
ORCID: https://orcid.org/0000-0003-1423-2710;
https://orcid.org/0009-0004-8535-4437

Motivation

- During the operation of extendible hashing, the data in a hash bucket need to be dynamically expanded. At this time, there will be a need to lock the entire hash directory, and then expand the size of the hash directory to twice the original, and the hash bucket also needs to be locked and split. This will cause the entire extendible hashing to be no longer accessible by other processes, so that all read and write access is suspended.
- In extendible hashing, a single hash bucket is used to store hash keys under each hash directory, and the corresponding hash bucket needs to be locked when writing the hash key, so that the hash bucket cannot be accessed by other processes. At the same time, the hash bucket adopts unordered management of hash keys. Although it can reduce the overhead such as sorting, the time overhead of the search is large, making the size of the hash bucket an important factor affecting extendible hashing performance.
- The increase in the number of hash keys in extendible hashing makes it difficult to store all hash keys in dynamic random access memory (DRAM), which makes the input/output (I/O) performance of external devices an important factor affecting extendible hashing performance.

System structure

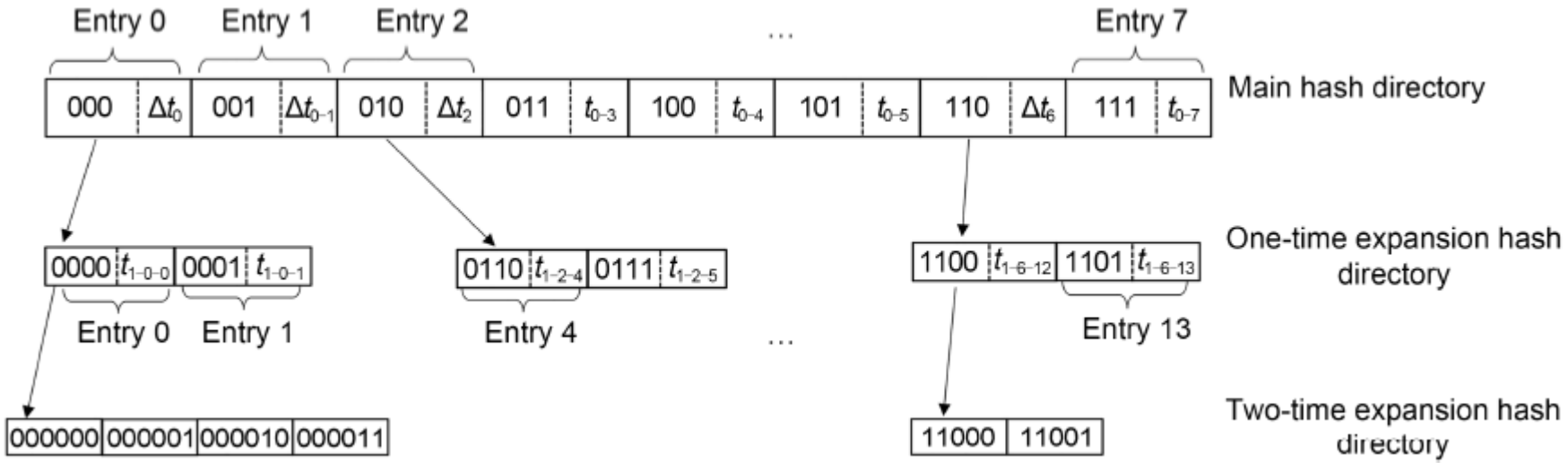


System structure

- System structure
  - Lazy expansion strategy for hash directory
  - Hash bucket management based on group
  - Hash directory recovery strategy based on tiered storage

Lazy expansion strategy for hash directory

The structure of lazy expansion strategy for hash directory



Lazy expansion algorithm

$$\begin{cases} \Delta t_i \geq \Delta t_{i-j}, \ \text{Spread\_rate} = 0, \\ \Delta t_i < \Delta t_{i-j}, \ \text{Spread\_rate} = 1. \end{cases}$$

Rule 1: If the value of Spread_rate is 1, the extended two-time expansion hash directory contains four sets of hash directory entries in the form of HL;

Rule 2: If the value of Spread_rate is 0, the extended two-time expansion hash directory contains two sets of hash directory entries in the form of HL.

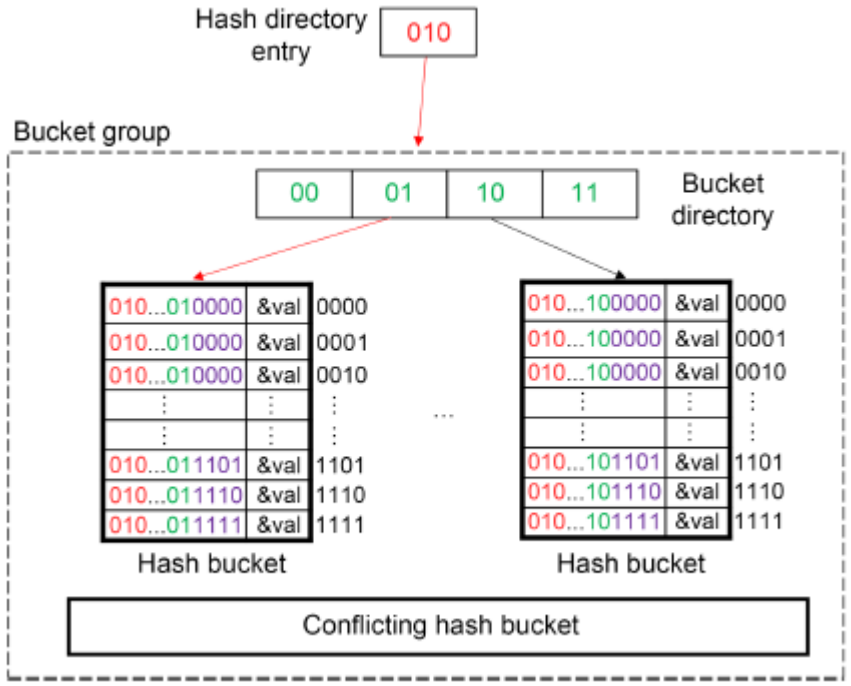
Advantages

Reduce the chance of hash directory expansion

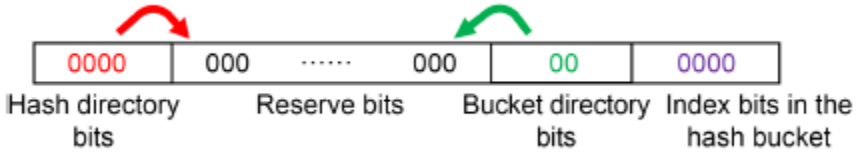
Improve concurrency

Hash bucket management based on group

The structure of bucket group



The role of bits in hash key



Hash directory bits: Index hash directory, can be expanded to the right bits and can occupy reserve bits

Bucket directory bits: Index bucket directories in a bucket group, can be expanded to the left bits and can occupy reserve bits

Index bits in the hash bucket: Four fixed bits, can be quickly located within the hash bucket

Advantages

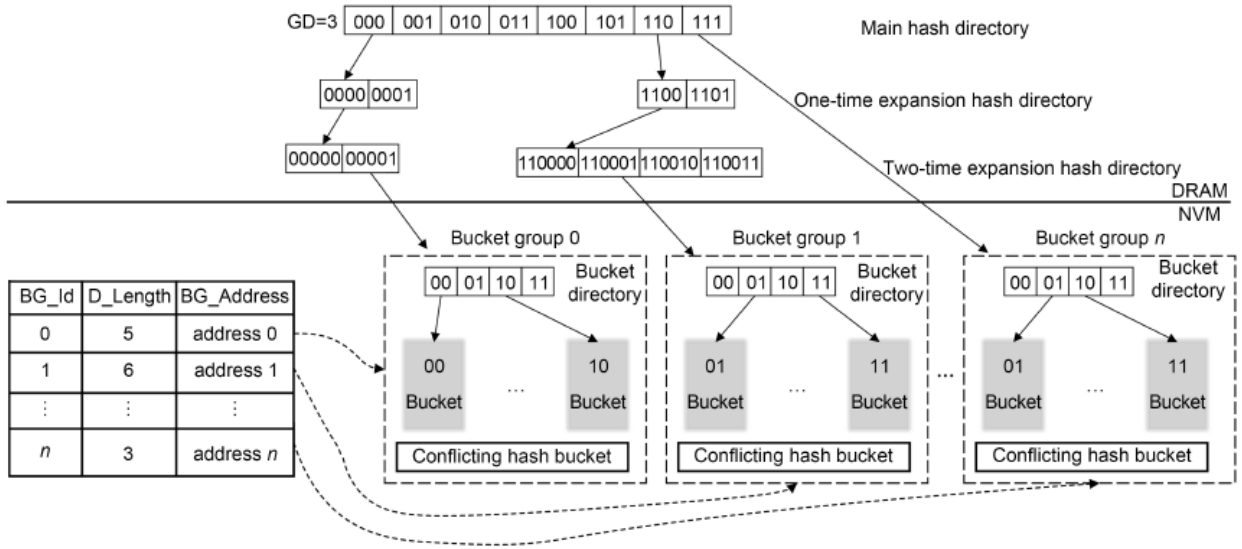
Reduce the granularity of the lock

Improve the access speed of the hash bucket

Improve the efficiency of hash bucket management

Hash directory recovery strategy based on tiered storage

The structure of hash directory recovery strategy



Advantages

Effectively take advantage of DRAM and NVM

Improve access to hash directory

Quickly recovery hash directories

Recovery strategy

Step 1: Access the Rec_BG_TABLE stored in the NVM storage device and read all Recovery_BG;

Step 2: Compare the size of D_Length in all Recovery_BG, find the maximum value and record it as D_L_Max;

Step 3: Build the main hash directory with $2^{D\_L\_Max}$ as the length;

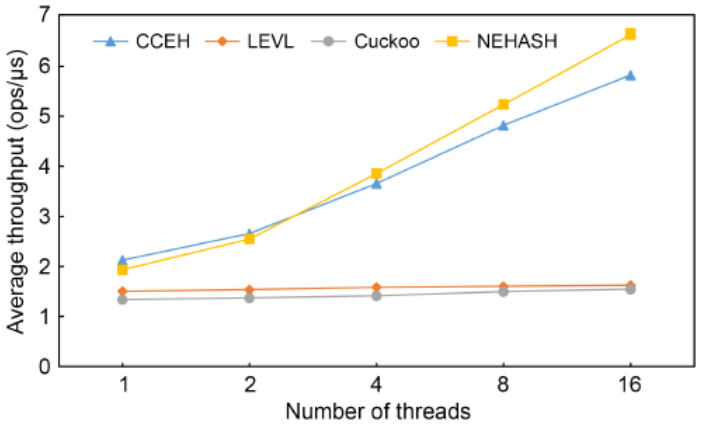
Step 4: Compare the D_Length value of each Recovery_BG in Rec_BG_TABLE in turn; if it is equal to D_L_Max, go to step 5; otherwise, go to step 6;

Step 5: Access the corresponding bucket group according to the BG_Address of Recovery_BG, and connect hash directory by D_Length;

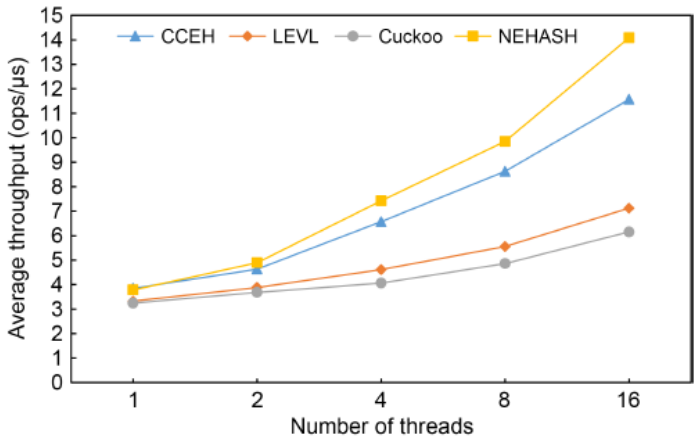
Step 6: Split the bucket group pointed to by BG_Address, construct the corresponding bucket group according to D_L_Max, and connect the corresponding bucket group.

Experimental results

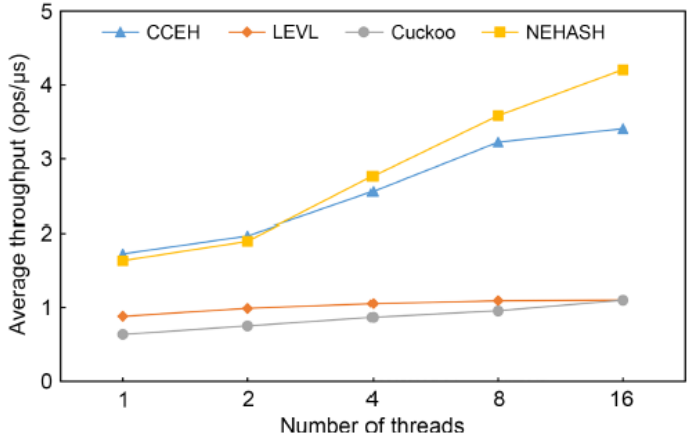
Concurrent I/O performance



50% read and 50% write



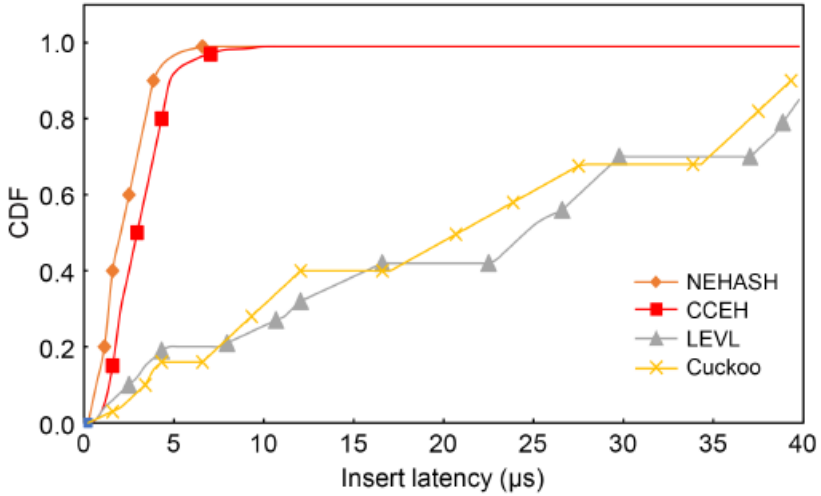
95% read and 5% write

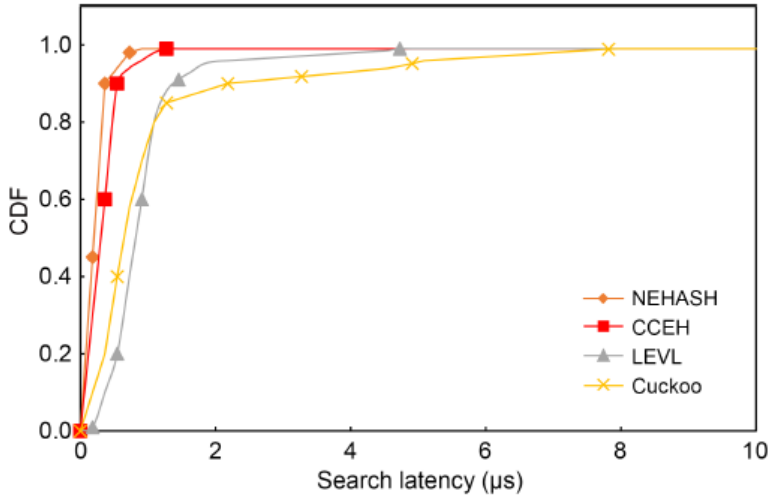


5% read and 95% write

NEHASH has higher concurrent I/O performance

Tail latency





NEHASH has lower I/O latency

Conclusions

- A multilevel hash directory based on lazy expansion is designed to improve the concurrency and efficiency of extendible hashing.
- A hash bucket management algorithm based on groups is presented to improve the efficiency of hash key management by reducing the size of the hash bucket, thereby improving the performance of extendible hash.
- A hierarchical storage strategy of extendible hashing for NVM is given to take advantage of DRAM and NVM.

Tao CAI, born in 1976. PhD, Associate professor. The School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His main research interests include new storage system and non-volatile memory.

Pengfei GAO, born in 1997. Master candidate. The School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His main research interests include non-volatile memory.

Dejiao NIU, born in 1978. PhD, Associate professor. The School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. Her main research interests include new storage system and neural network.

Yueming MA, born in 1997. Master candidate. The School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His main research interests include non-volatile memory.

Tianle LEI, born in 1999. Master candidate. The School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His main research interests include non-volatile memory.

Jianfei DAI, born in 1999. Master candidate. The School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His main research interests include non-volatile memory.