In-Storage Read-Centric Seed Location Filtering Using 3D-NAND Flash for Genome Sequence Analysis

You-Kai Zheng, **Ming-Liang Wei**, Hsiang-Yun Cheng, Chia-Lin Yang,
Ming-Hsiang Tsai, Chia-Chun Chien, Yuan-Hao Zhong, Po-Hao Tseng, Hsiang-Pang Li

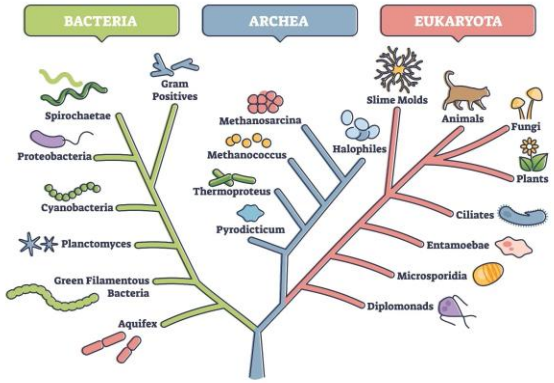
National Taiwan University, Taipei, Taiwan
Academia Sinica, Taipei, Taiwan
Macronix International Co., Ltd., Hsinchu, Taiwan

Outline

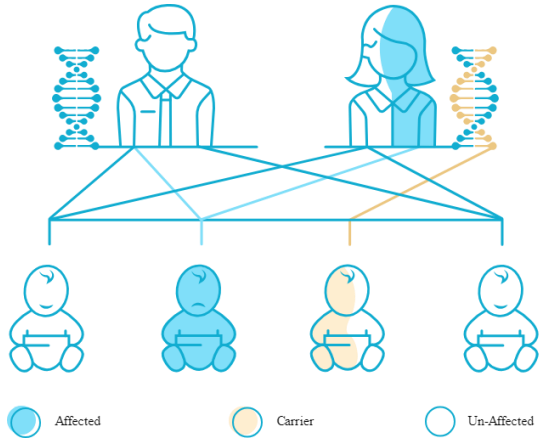
- Background
- Motivation
- Proposed Design
- Evaluation
- Conclusion

Applications of Genomic Sequencing

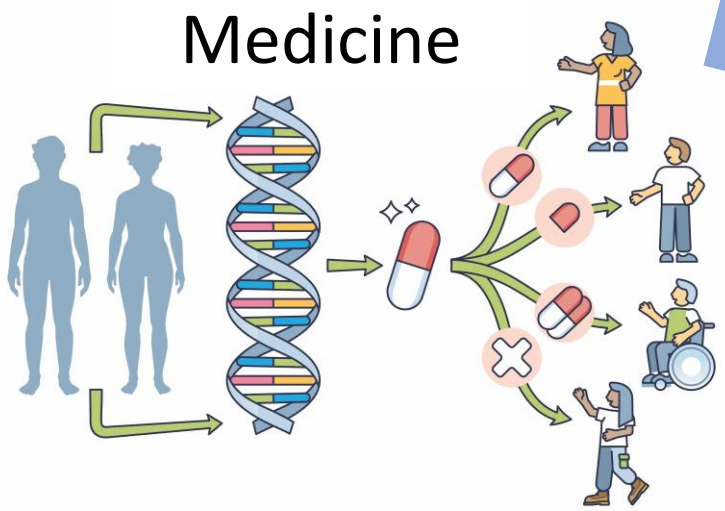
Phylogenetic Tree



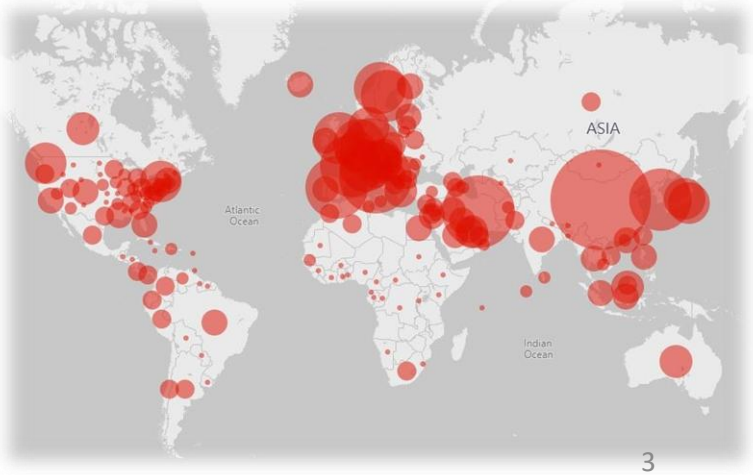
Detection of Genetic Disorders



Personalized Medicine

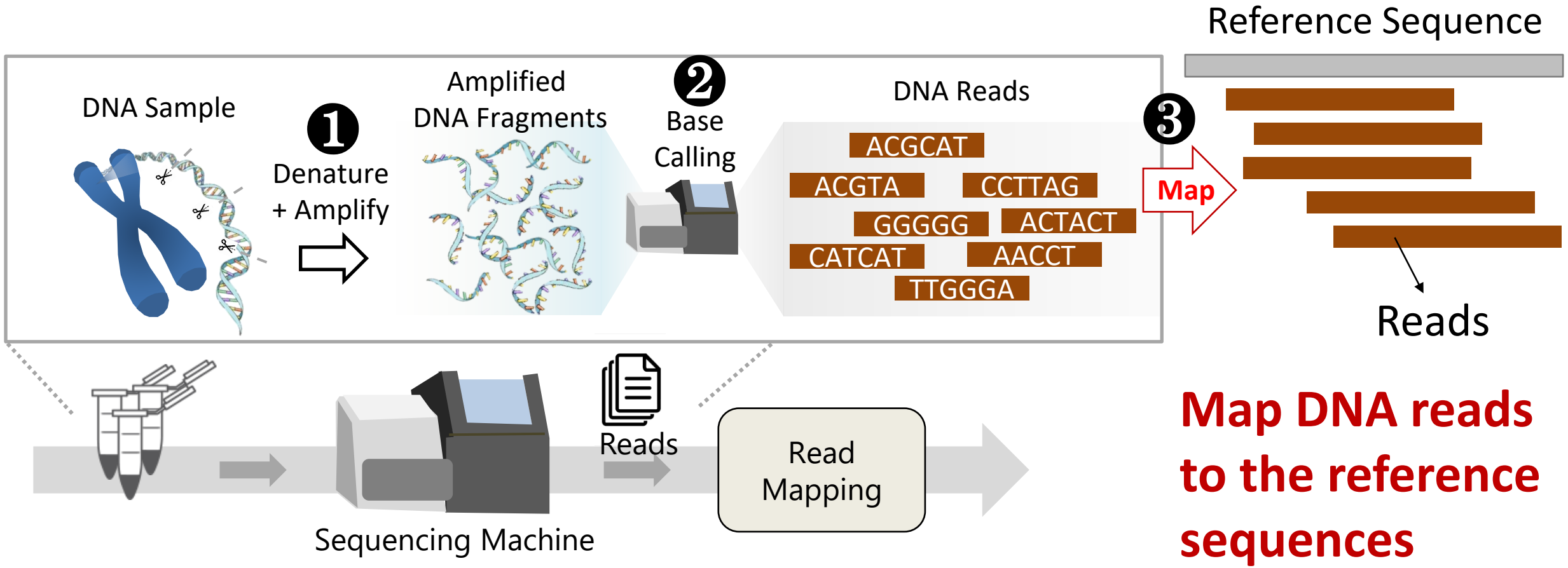


Outbreak Tracking



Background

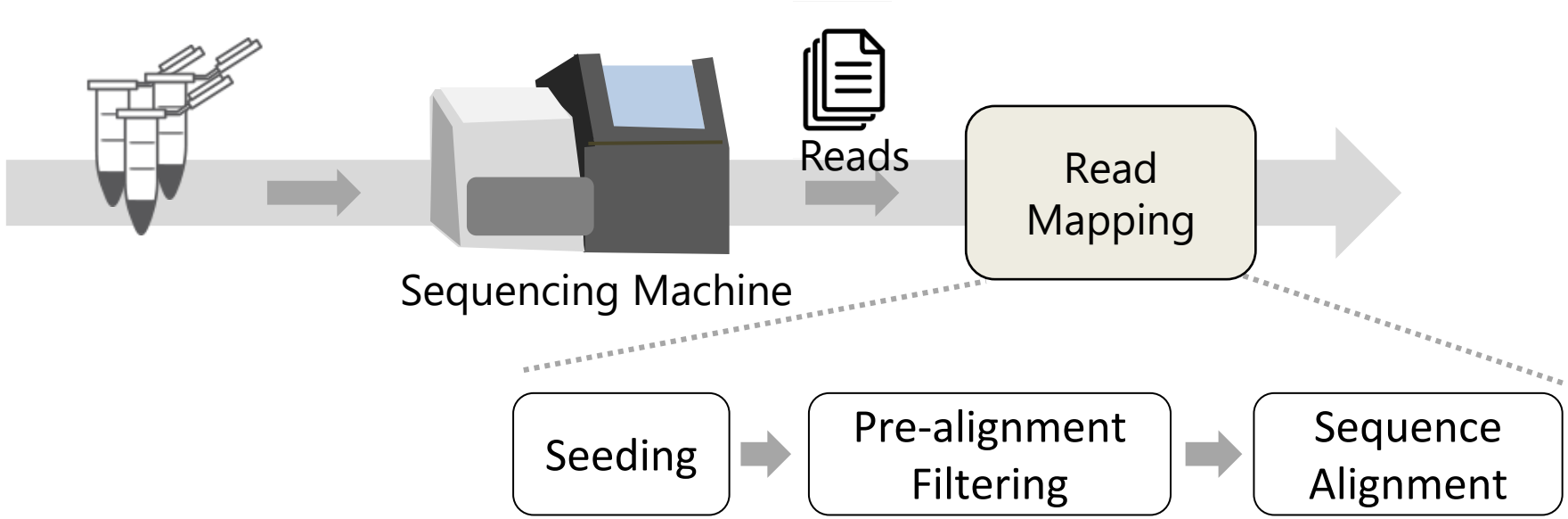
- Genomic Sequence Analysis Pipeline

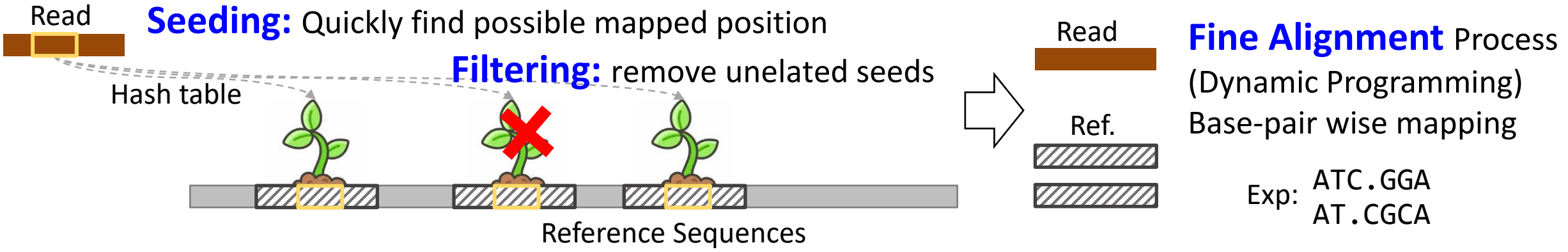


Map DNA reads to the reference sequences

Background

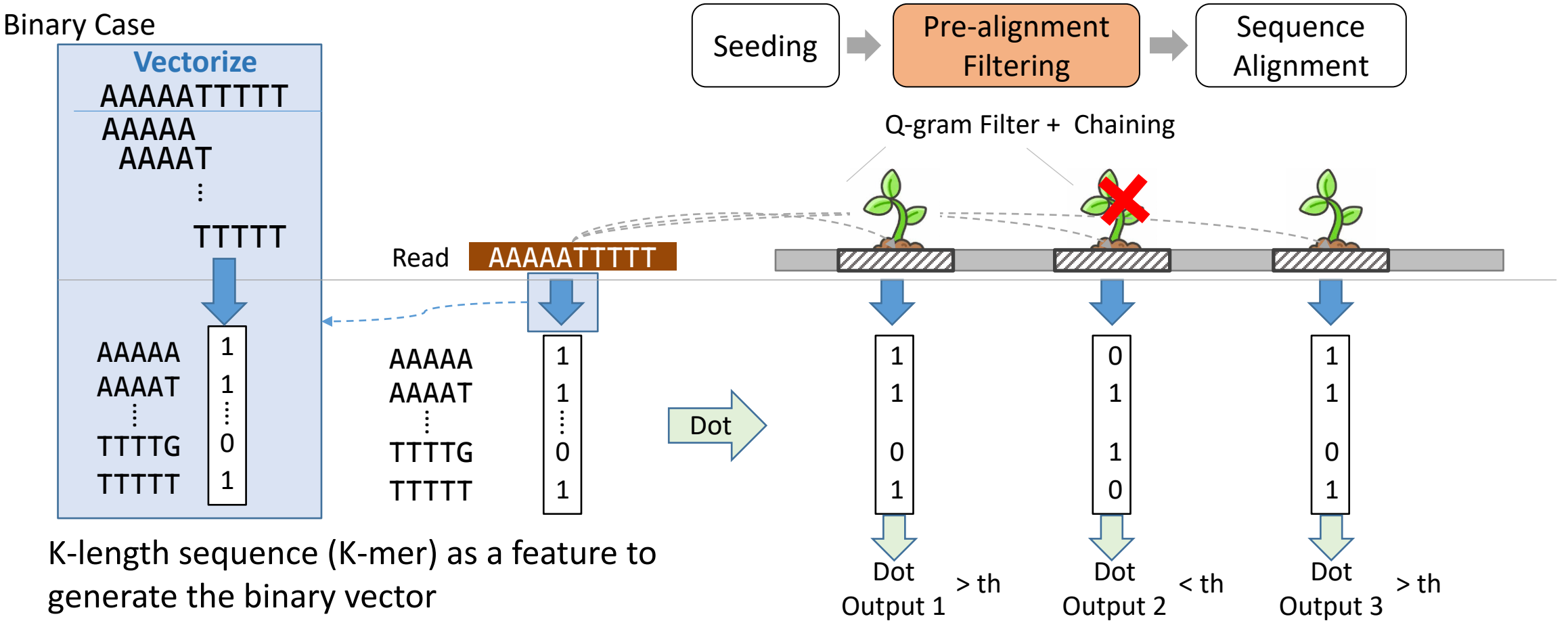
- Read Mapping Process





Background

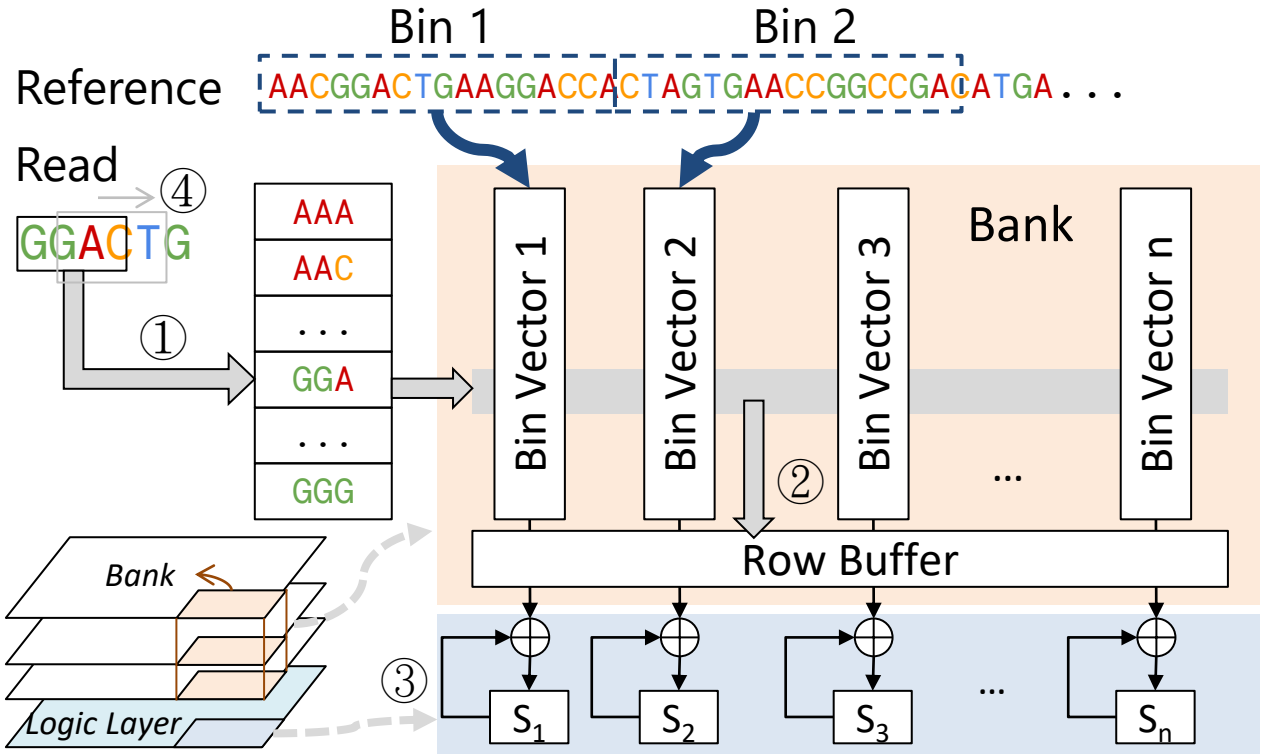
- Prior Q-Gram
  - Convert the sequence into a vector, and check their similarity through inner product.



K-length sequence (K-mer) as a feature to generate the binary vector

Background

- Q-gram Filter in HBM
  - Prior work implements the Q-gram Filter in High Bandwidth-Memory(HBM)
    - Process the inner product directly **at the logic layer of HBM**
    - **Reduce the memory footprint** between core processor and memory

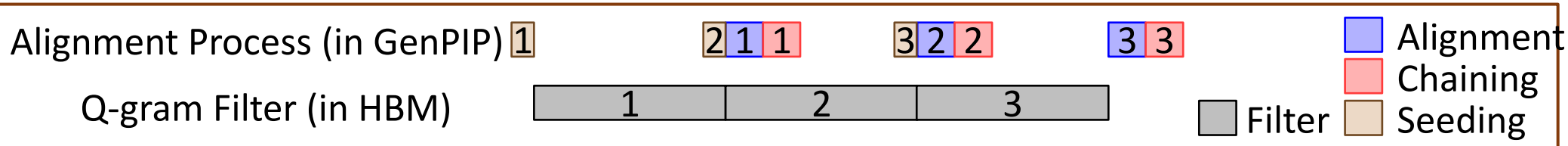


A page read op. **loads dimensions with 1** in the vector **of the read** to the logic layer.

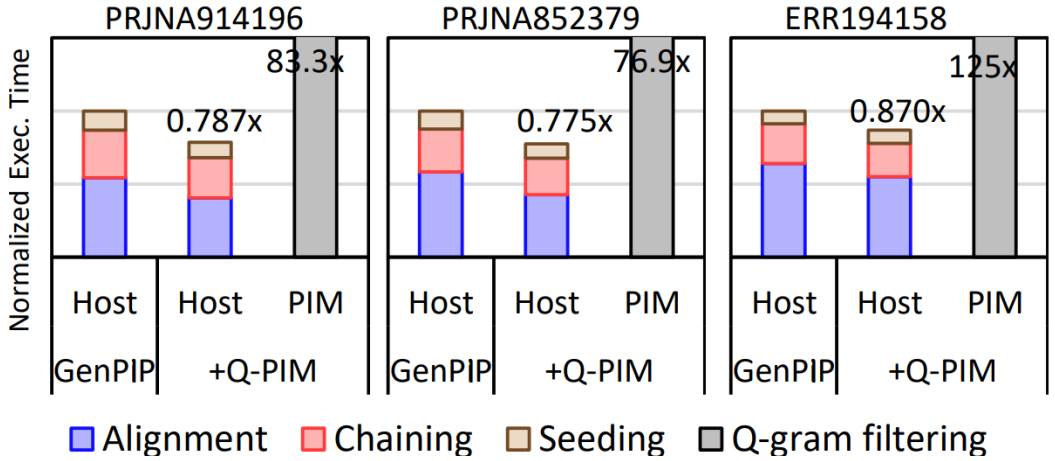
Accumulate them to process the inner product.

Motivation

- Issues of the Q-gram Filter in HBM:

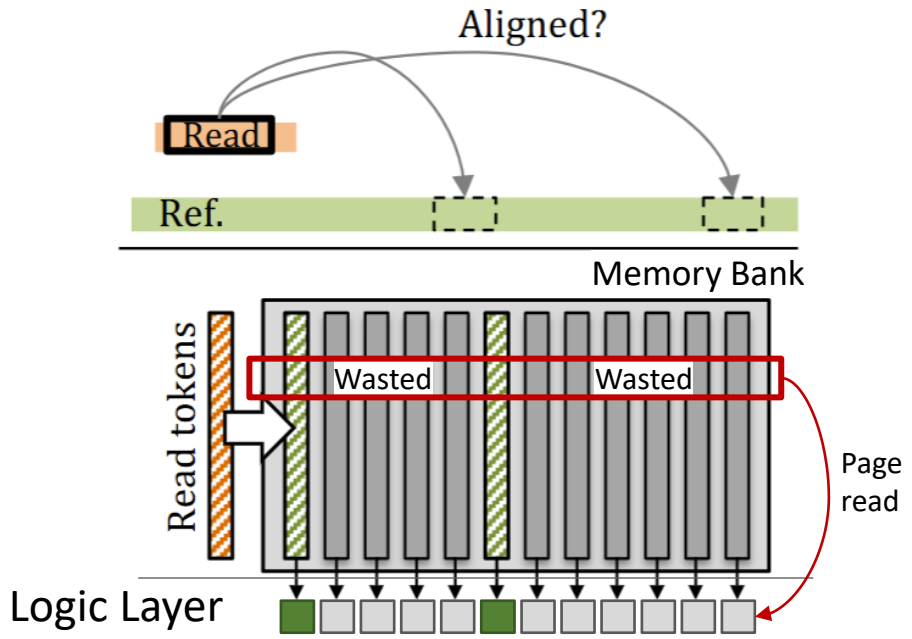


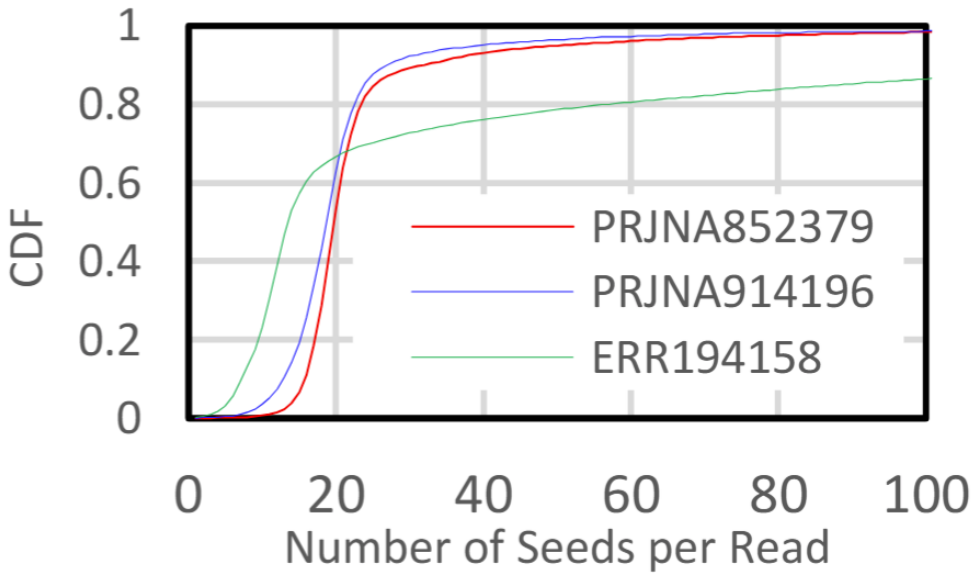
As the alignment process (Chaining, Alignment) is accelerated by ASIC (GenPiP), the **Q-gram Filter becomes the performance bottleneck**.



Issue1 - Underutilization

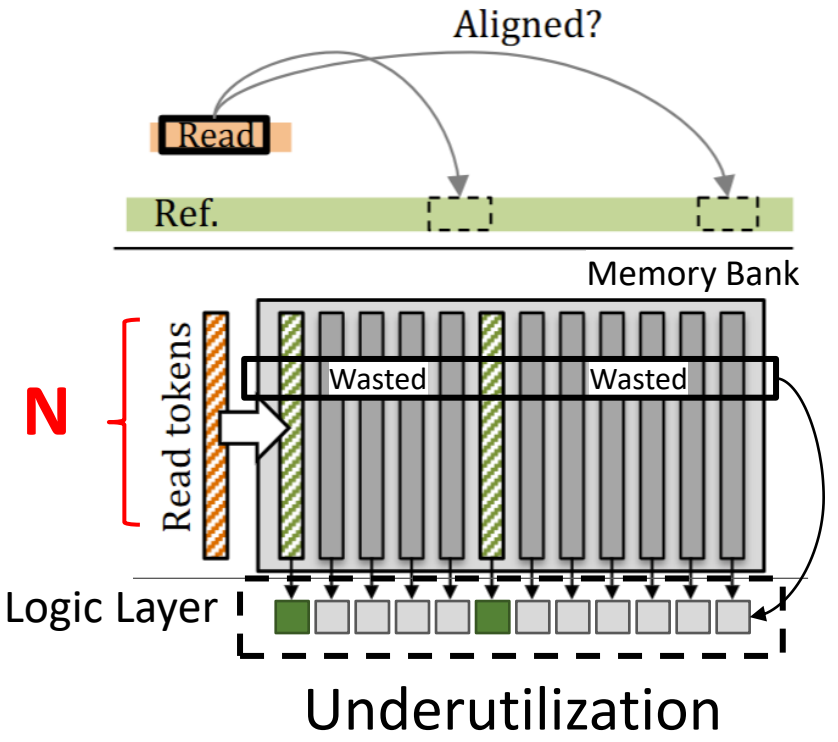
- **Underutilization** of Memory Access Bandwidth
  - Prior Methodology:
    Compare candidate reference position (seed) in the memory block.
  - But **Seed# << memory access granularity (Page Size)**





Issue2-1 Energy Hungry of HBM

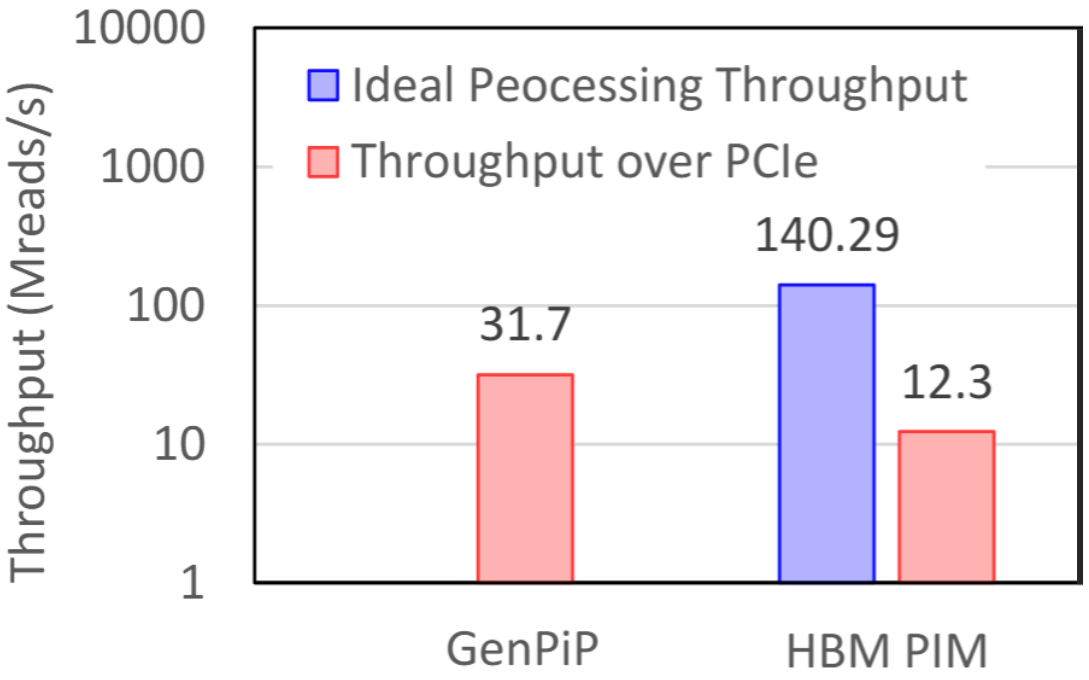
- Energy Hungry for HBM's Solution:
  - Access energy unit: 4pJ/bit
  - Power: **30W** (1TB/s x 32pJ/Byte)
  - Energy : Page size x **N** x 4pJ/bit
    - **N** : page access count :
      The non-zero dimension# of input query
    - **Page size >> Seed#**: underutilization

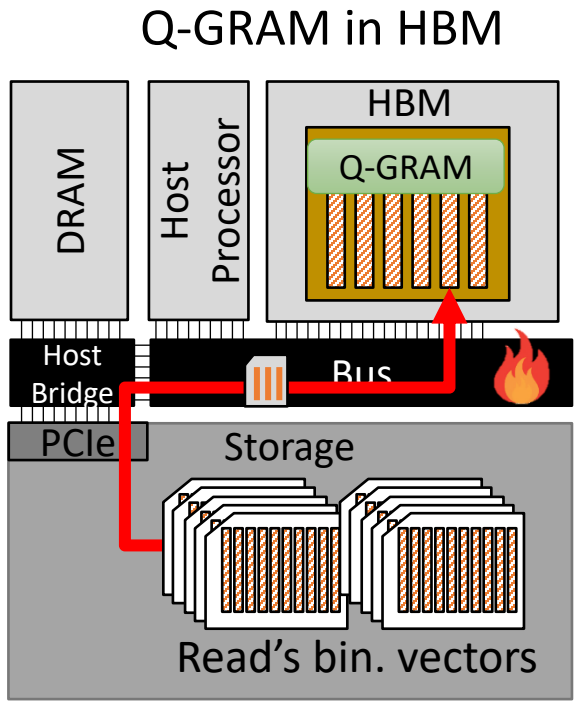


The **underutilization** of read bandwidth and **page access count (N)** lead to the massive **energy consumption**

Issue2-2 Memory Wall

- Memory Wall from Storage
  - The read's sequence needs to load from storage
  - **PCIe bandwidths becomes a bottleneck**



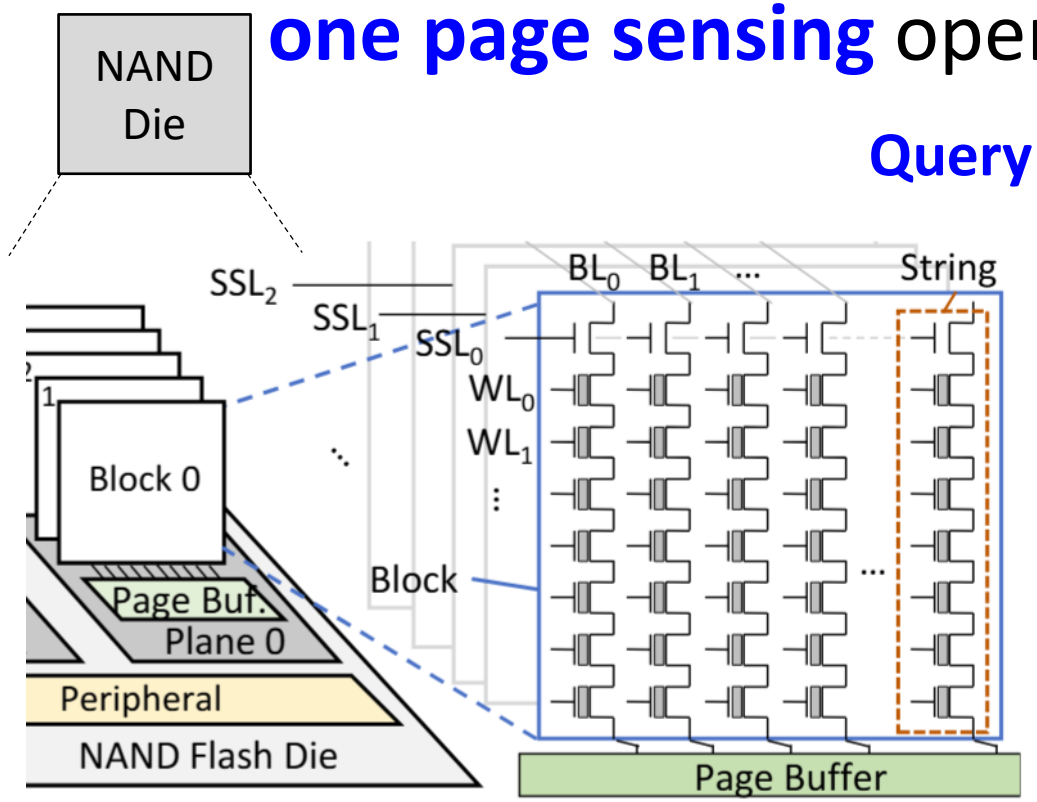


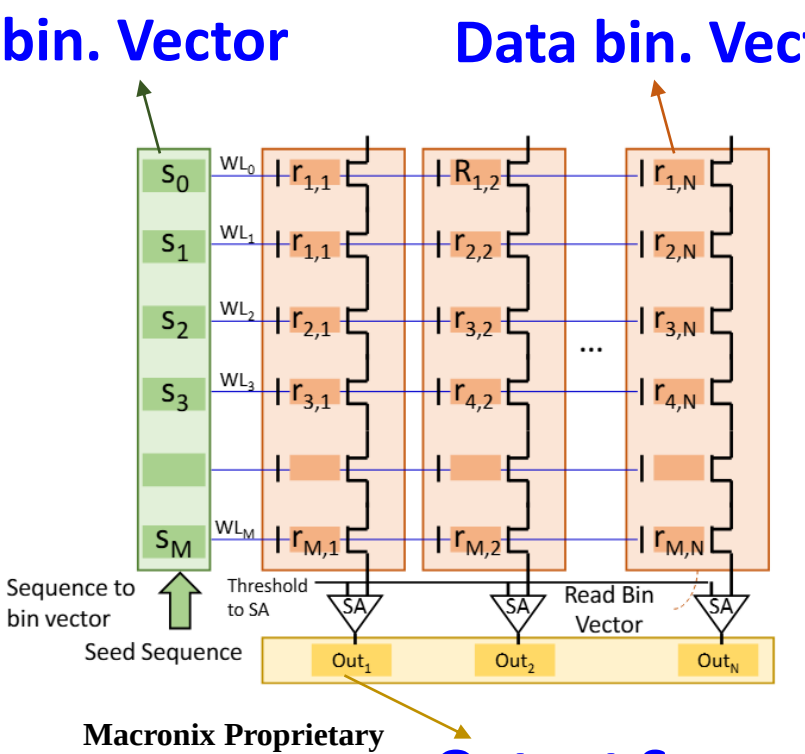
Search Macro Reads Bin Vector

The HBM PIM throughput constraints due to transferring **hundreds of GBs** of read bit-vectors **via the PCIe bus.**

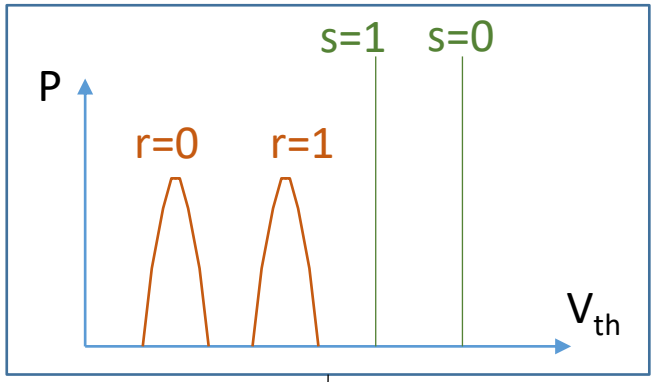
Integrating 3D NAND CIM

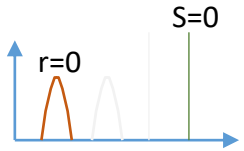
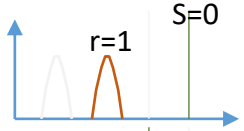
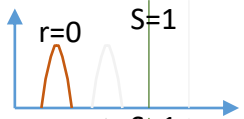
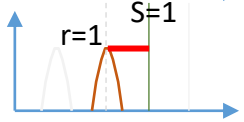
- Opportunity of 3D NAND CIM (Computing-in-Memory)
  - **Reduce data movement**
  - Full-dimension comparison over **one page sensing** operation





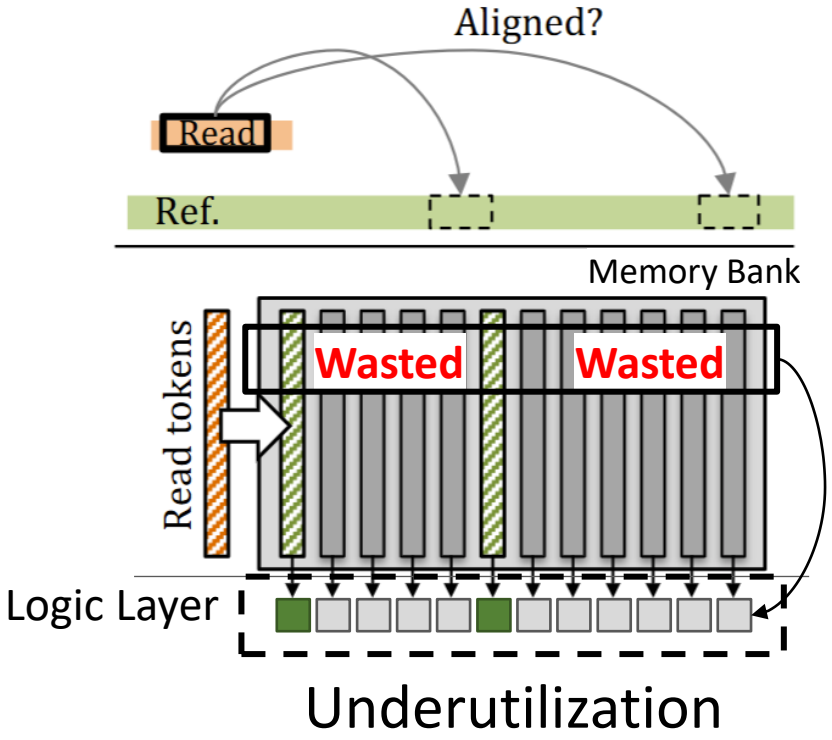
Output Sensed by SA



s	r	Setting	OP.
0	0		Sat.
0	1		Sat.
1	0		Sat.
1	1		Ω Ohm

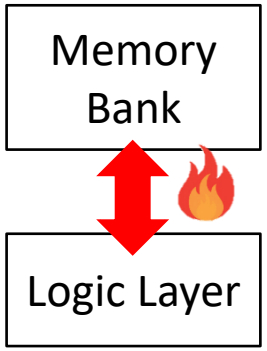
Summary of Issues

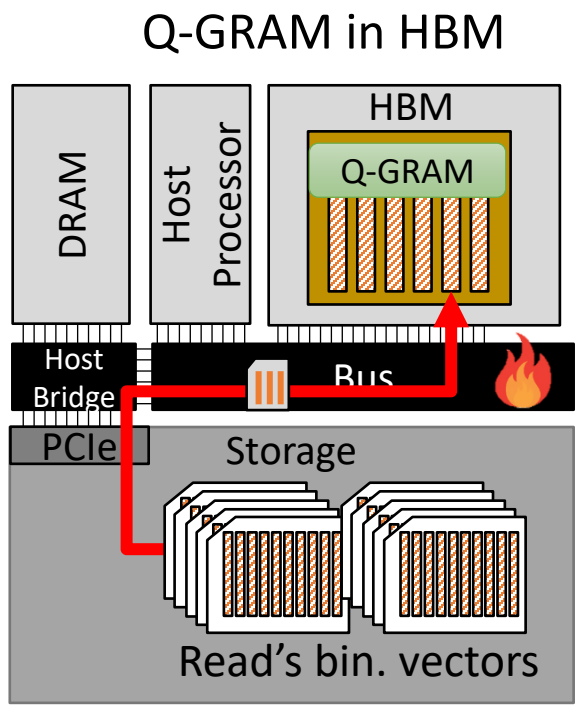
Underutilization of Memory Access Bandwidth



Seed# << Page Size

Energy Hungry of HBM and Memory Wall from PCIe

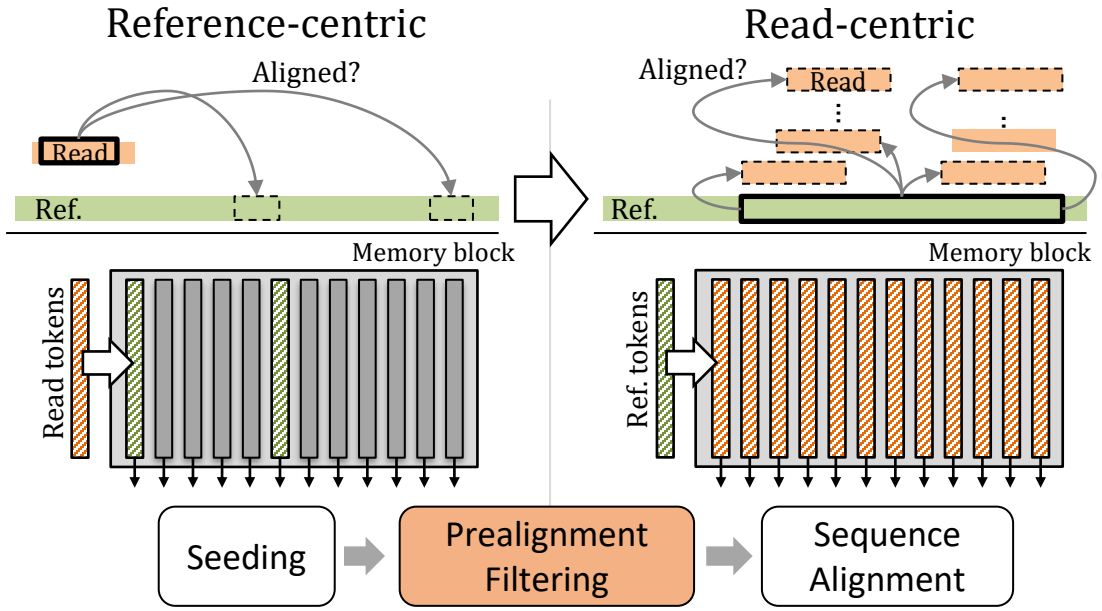




Contributions

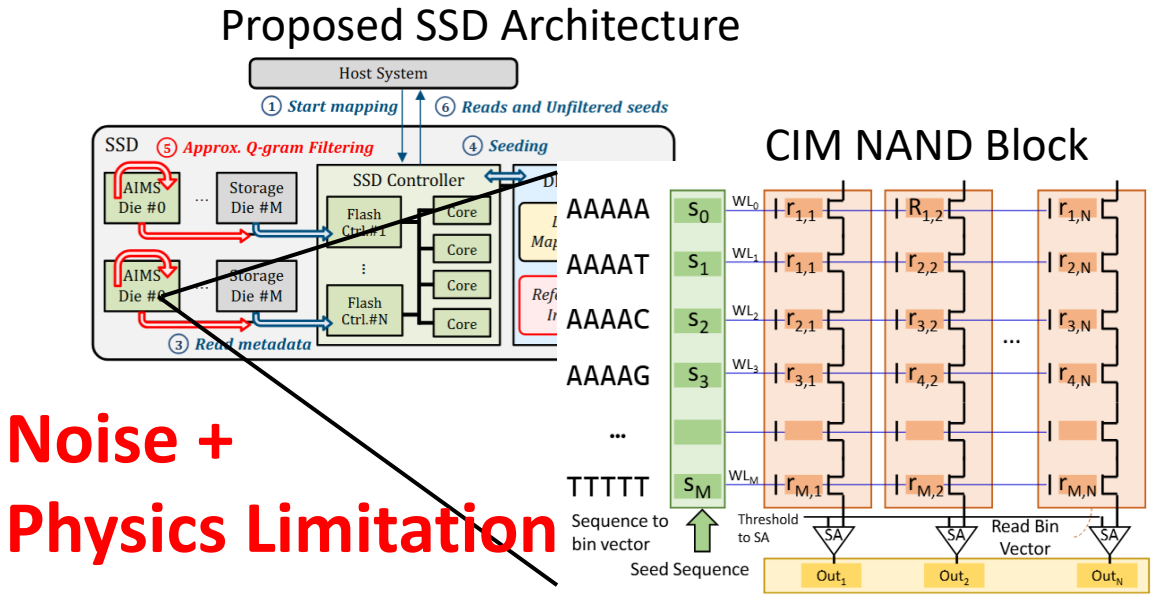
Read-centric Search Methodology

The novel “read-centric” search methodology that **puts the read’s binary vector in the memory block** to fully utilize the memory access bandwidth



Deploy Q-Gram Filter into Storage

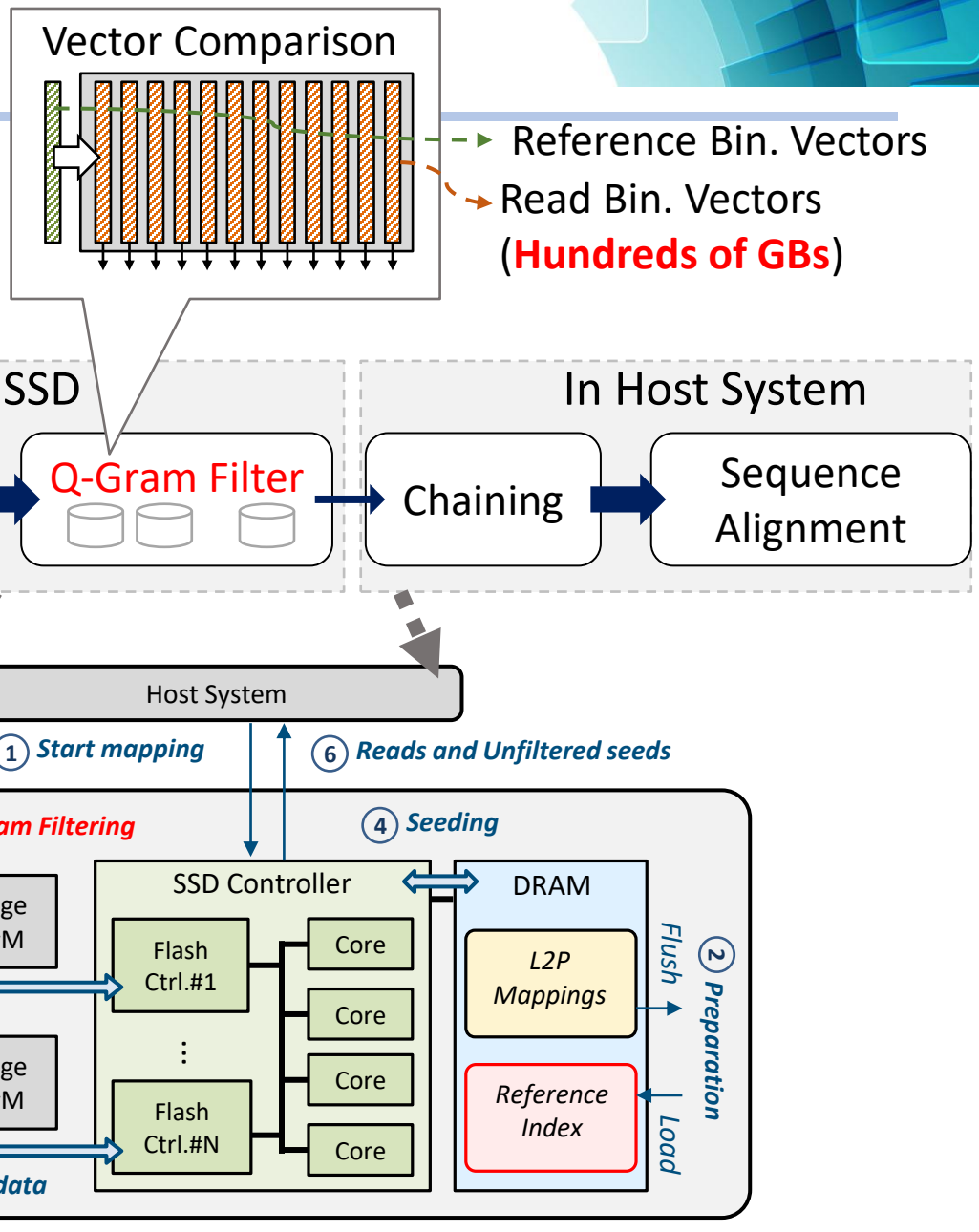
Integrating with **3D NAND CIM** (Computing-in-Memory) and **CIM-aware encoding methodology** enabling the “Q-Gram filter” processed in the storage size.



Noise + Physics Limitation

Overall Architecture

- To Toggle the Data Movement Overhead:
  - Seeding & Q-GRAM Filter in Storage
    - **No movement of read bin. Vectors**
    - Only transmit non-filtered seeds to the host system via PCIe
  - W/ 3D NAND Computing-in-Memory
    - Read bin. vectors are **compared** parallelly **within a NAND read cycle** (20us)



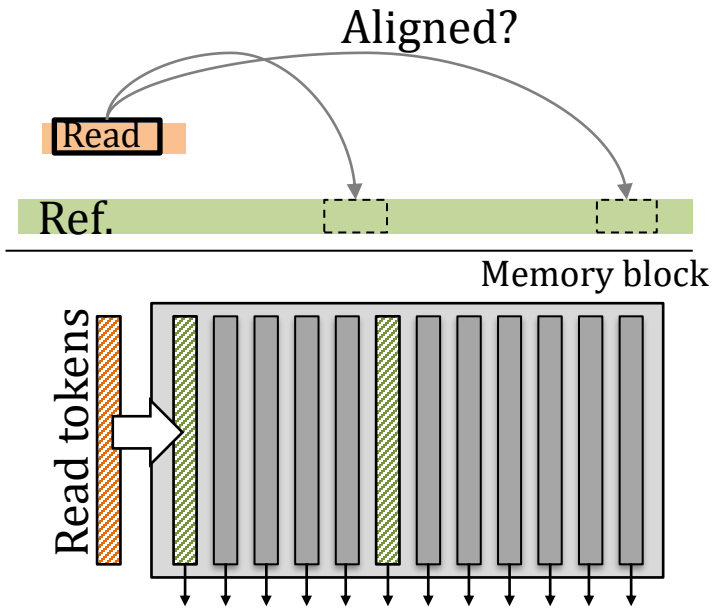
Read-centric Search Methodology

Conventional Search Methodology

Compare one read with its seeds a time

Problem: **Underutilization**

Reference-centric



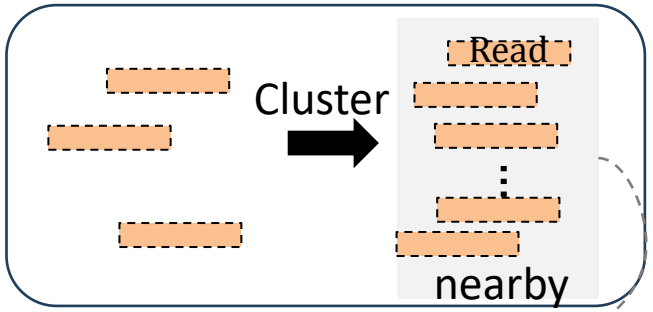
New Methodology

Read-Centric Search Methodology

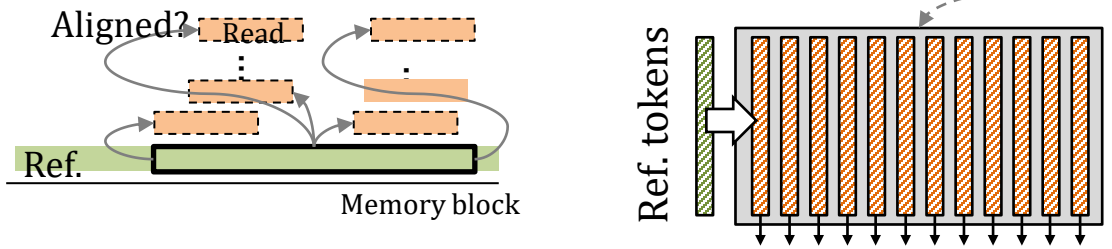
Cluster nearby reads, and store into memory bank/block

Process **cluster-wise Q-gram** filter

To **fully-utilized** the memory access bandwidth



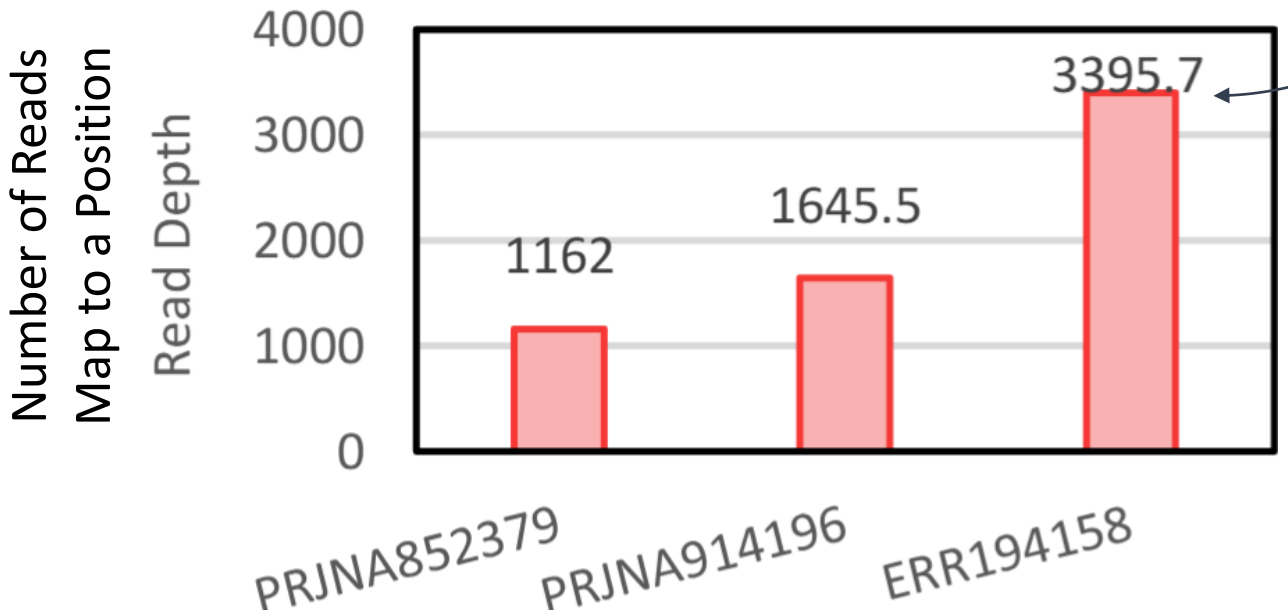
Read-centric

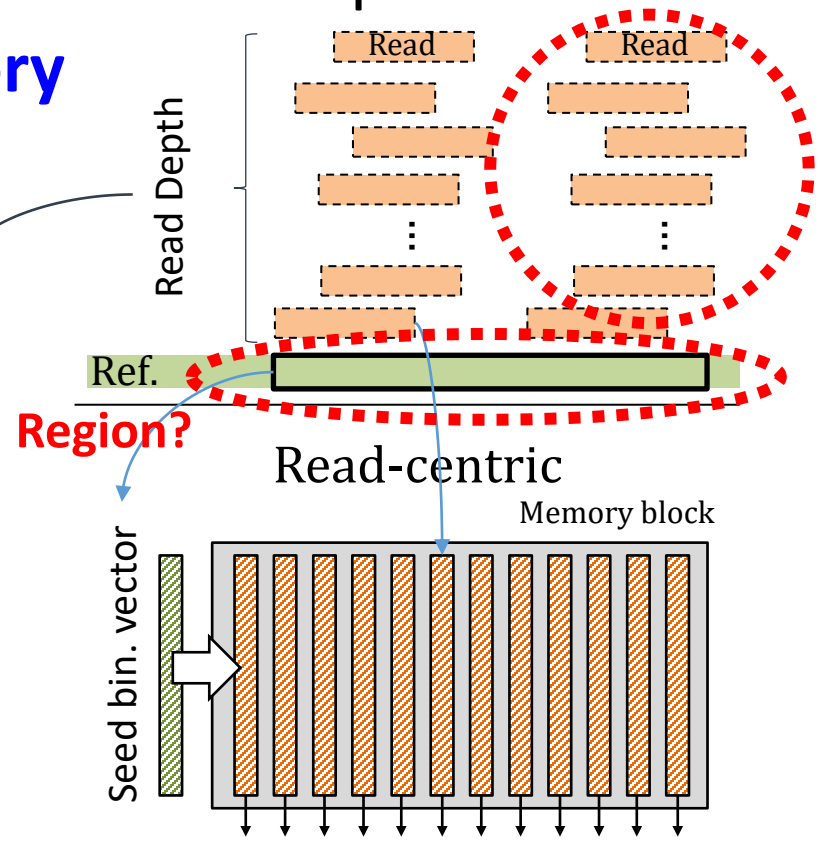


Shared Seed as Input

Read-centric Search Methodology

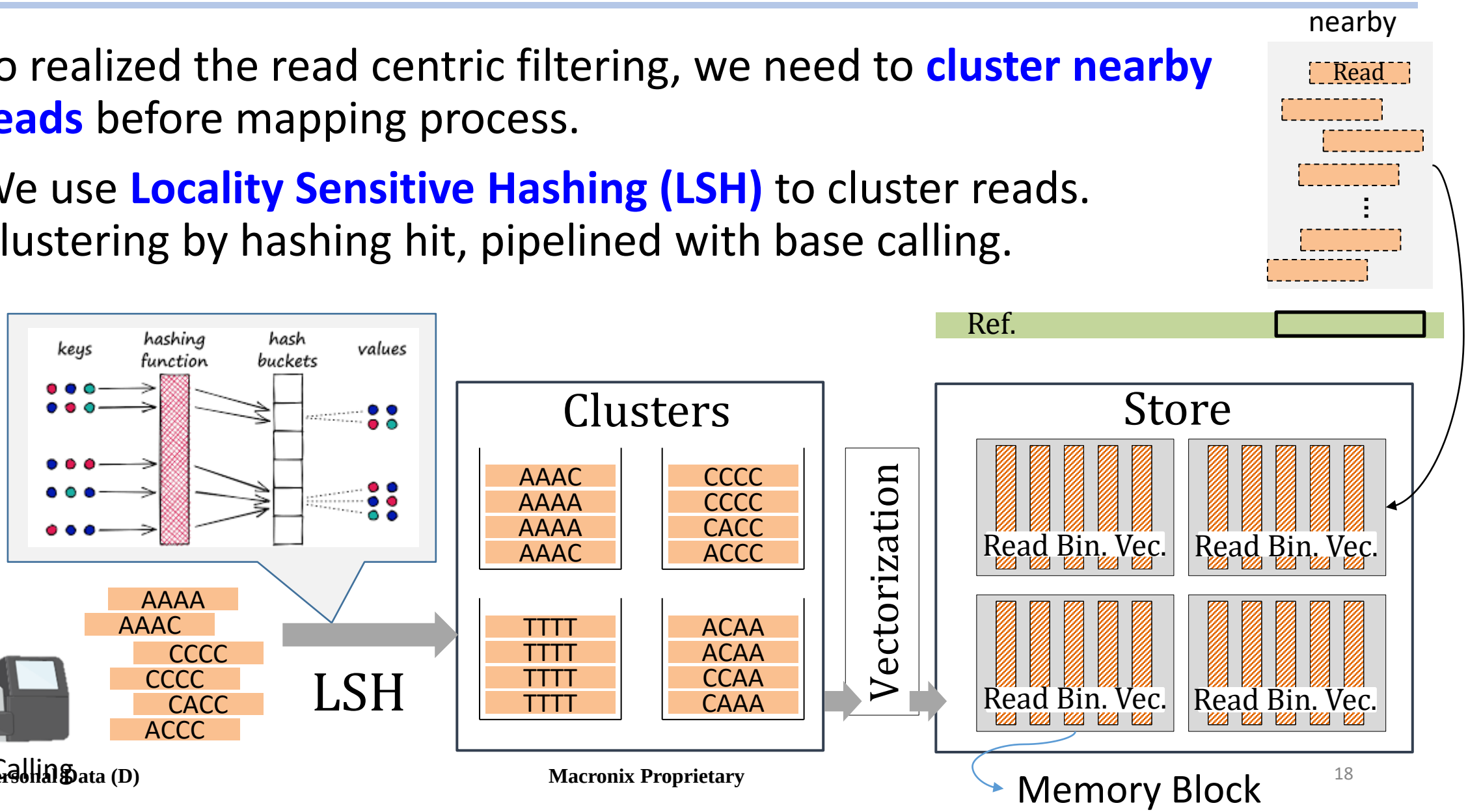
- For Fully Utilized the Memory Access Utilization
  - We observed that **thousands of reads** are **related to few seeds** for application of targe sequencing and pathogenic SNP exploration.
  - Suggest to **put read** binary vector **to memory block**, and seed's binary vector as query





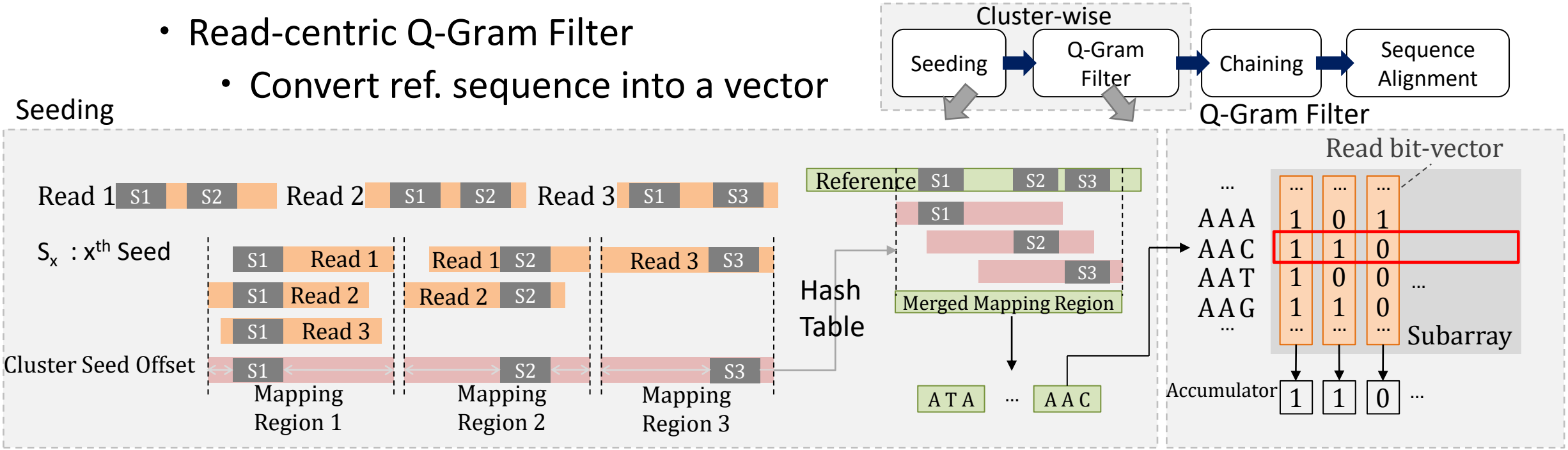
Read-centric Search Methodology

- To realized the read centric filtering, we need to **cluster nearby reads** before mapping process.
- We use **Locality Sensitive Hashing (LSH)** to cluster reads. Clustering by hashing hit, pipelined with base calling.



Read-centric Search Methodology

- Determination of Candidate Mapping Region for a Read Cluster
  - Concept: the **reference sequence covering whole reads** as a query
  - Cluster-wise seeding:
    - Align reads with seeds and determine the mapping region
    - Union the mapping region to query the reference sequence
  - Read-centric Q-Gram Filter
    - Convert ref. sequence into a vector



Deploy to NAND Flash

Ease Error Accumulation

$$\Sigma A \cdot B = \Sigma A - \Sigma A \cdot \overline{B}$$

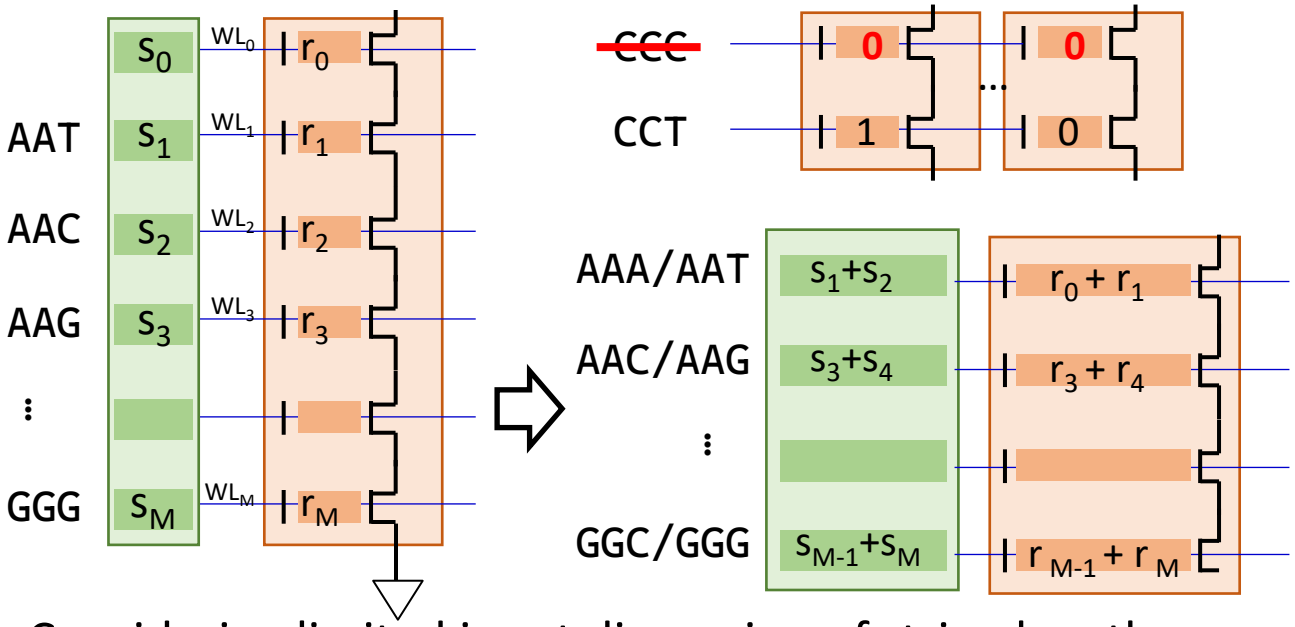
Matched Count >> Mis-matched Count

☹ Noisy while A~B

☺ Distinguishable even A~B

We accumulate the resistance in the analog domain, but the **non-linearity** and cell **noise** is severe **as the number is accumulated**.
Instead, calculate the **mis-match count** in analog domain which is **relative small** due to reference and reads are similar.

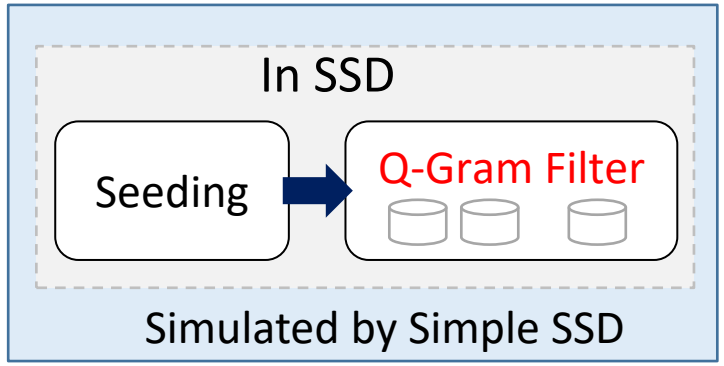
Dimension Reduction

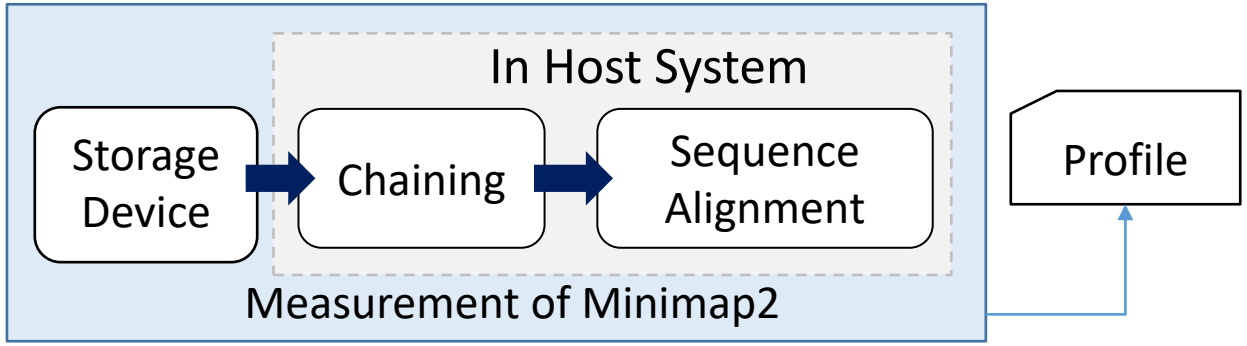


Considering limited input dimension of string length, we **remove unused q-gram** and merge the vector dimension by **fuse q-gram**.
The merging step **slightly** increases **false positive** seeds, but **enable saving massive data movement**.

Evaluation

- Datasets:
  - **Short-read** sequencing dataset: PRJNA914196[2], PRJNA852379 [3], and SRR2052419 [4].
  - Hg38 human reference genome [5].
- Clustering Reads on the Sequencing Machine:
  - Clustering implementation on an Intel® i7-7700 CPU, similar to the i7 CPU in the Illumina NovaSeq 6000.
- Read Mapping Evaluation: (including Storage + Host)
  - Storage System: Simulated by the **SSD simulator** (Simple SSD). Note: Due to the storage system can process the seeding and q-gram filter **is independent from host**, the latency of the storage system is **evaluated individually**.
  - Host System:
    The **proposed storage system is abstract into a PCIe SSD**. The host access the non-filtered read and reference seeds from the storage device.





Evaluation

- Hardware Configuration:
  - 8 Channels/8 Way PCIe gen4 SSD
  - 4KB low latency (20us) NAND
  - 2400MT/s
  - 4 SSDs share a PCIe-bus
- Energy Evaluation
  - Host accelerator power config from GenPIP [6]
  - **HBM** power estimation: **4pJ/bit**. [7]
  - **NAND-IMS** power:
    - **Equal to page read power**
    - Uses an identical sensing scheme: read voltages are applied on WLs, and currents are sensed on all bit-lines.

Parameters of HBM DRAM	
Channels / Channel width	32 / 128 bits
BankGroups / Banks per BankGroup	4 / 4
Subarrays per Bank	64
Columns / Rows per subarray	8192 / 32768
Clock Frequency (1/tCK)	1,000MHz
tRCD-tCAS-tRP (ns)	14-14-14

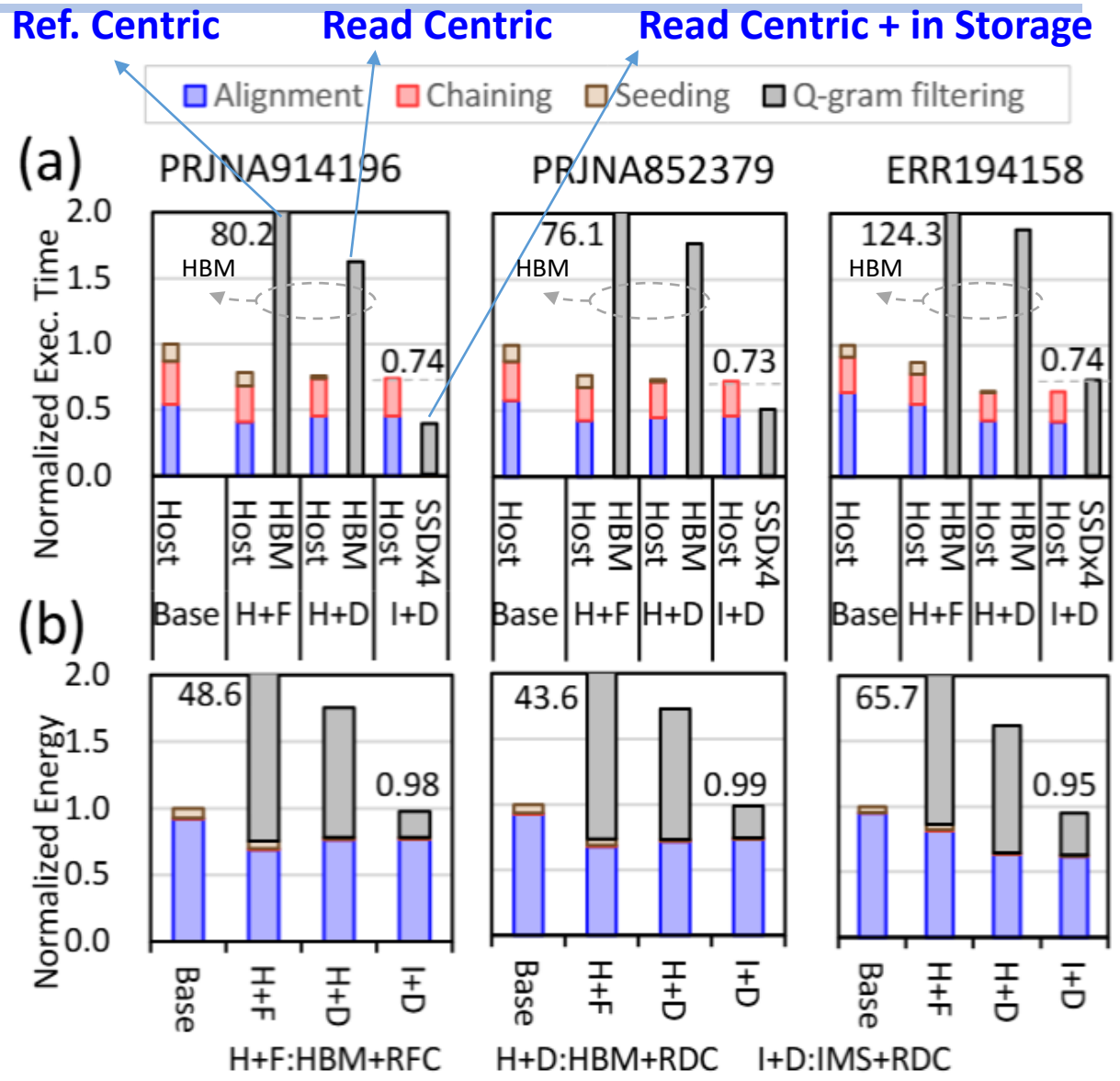
Parameters of SSD	
Memory Cell	MLC
Channels / Dies / Planes / Blocks	8 / 8 / 4 / 1056
BLs / WLs per NAND block	4KB / 256
Read latency	20us
Channel speed / Channel width	2,400MT/s / 8 bits

Component Power	
HBM	30W
GenPIP Chaining unit (with 4MB eDRAM)	1.346W
1024 GenPIP Alignment units	85W
4096 GenPIP Seeding unit	28.2W
256 NAND Chips (4 SSDs)/1 NAND Chip	25.6W/0.1W

[7] Mike O'Connor et al. "Fine-grained DRAM: Energy-efficient DRAM for extreme bandwidth systems". In: Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture. 2017, pp. 41–54.

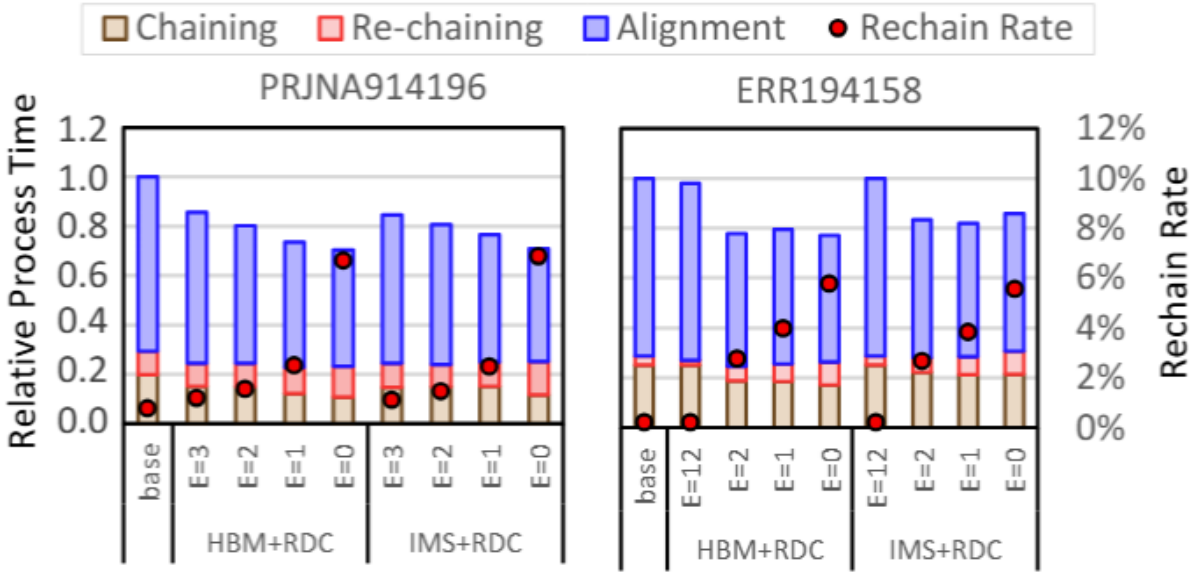
Evaluation

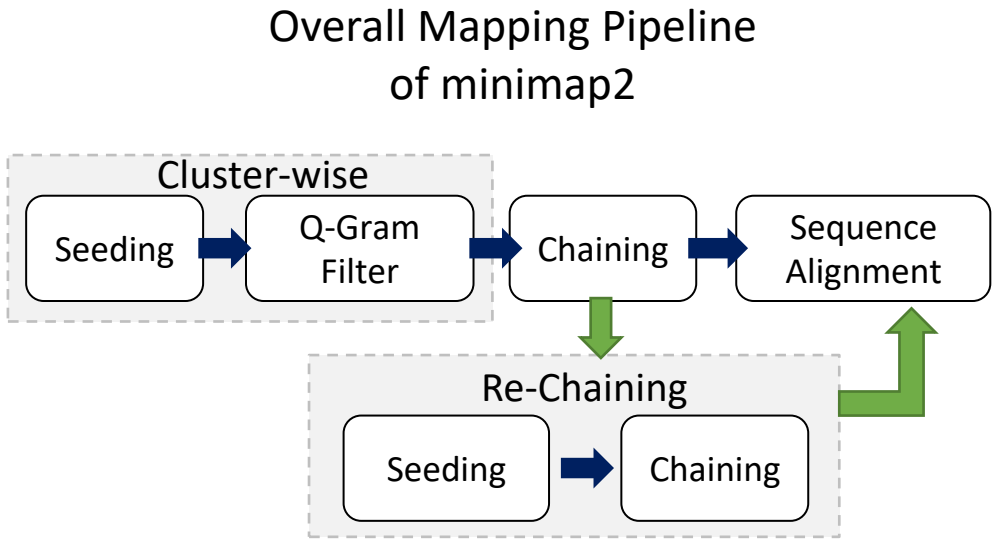
- Latency Evaluation:
  - Q-gram filter reduce 20% of Host's processing time, but
  - **Q-gram filter in HBM** becomes the **bottleneck**
  - Read centric eases filtering time but still bounded by PCIe bandwidth
  - The in storage solution toggles the movement overhead and hide the q-gram filter processing time
  - Compared to prior HBM PIM +reference centric, our **3D NAND IMS + Read Centric** achieves **108.4x, 104.2x, 168.0x speeding up**.
- Power:
  - The **HBM's solution** requires **massive page read** operations for each dimension.
  - Compared to HBM, **the 3D NAND IMS** only takes one page read to implement the q-gram filter **saving 49.6x, 44.0x, and 69.2x energy**.



Evaluation

- Sensitivity of Mismatch-Count (E):
  - We found the **restrict filter condition**, E = 0, 1, **trigger 5% the re-chaining process** in the exist alignment tool to maintain chaining quality, but
  - The **chaining and alignment time is save by 20%**, being more than the additional re-chaining time.
  - As the result, the q-gram filter **saves host system around 80% time with chaining quality is maintained.**





Conclusions

- We address the challenges in mapping process in the genomic analysis pipeline.
- Existing **HBM PIM** search methodologies encounter **read amplification** and **PCIe bottlenecks**.
- Our **read-centric methodology** fully **exploits search parallelism** to enhance performance.
- The proposed **in-storage system** effectively **alleviates the PCIe bottleneck**.
- By converting match counts to mismatch counts, we address non-linearity and noise accumulation issues during processing.
- Additionally, the removal of unused q-grams and dimension merging optimizes the vector size to fit the page strength of a NAND block.
- As a result, the proposed **read-centric + 3D NAND IMS** achieves an average of **23.8x performance gain and 53.3x energy efficiency improvement** compared to state-of-the-art PIM solutions.

References

[1] Po-Hao Tseng et al. “A Hybrid In-Memory-Searching and In-Memory-Computing Architecture for NVMBased AI Accelerator”. In: Symposium on VLSI Technology. IEEE. 2021.

[2] Clara Estela Diaz-Velasquez et al. “Evaluation of genetic alterations in hereditary cancer susceptibility genes in the Ashkenazi Jewish women community of Mexico”. In: Frontiers in genetics 14, 1094260 (2023).

[3] Eric W. Sayers et al. “Database Resources of the National Center for Biotechnology Information”. In: Nucleic Acids Research 50.D1 (Jan. 7, 2022), pp. D20–D26.

[4] Justin M Zook et al. “Extensive sequencing of seven human genomes to characterize benchmark reference materials”. In: Scientific data 3.1, 160025 (2016).

[5] Valerie A. Schneider et al. “Evaluation of GRCh38 and de Novo Haploid Genome Assemblies Demonstrates the Enduring Quality of the Reference Assembly”. In: Genome Research 27.5 (May 2017), pp. 849–864.

[6] Haiyu Mao et al. “Genpip: In-memory acceleration of genome analysis via tight integration of basecalling and read mapping”. In: 55th IEEE/ACM International Symposium on Microarchitecture (MICRO). IEEE. 2022, pp. 710–726

[7] Mike O’Connor et al. “Fine-grained DRAM: Energy-efficient DRAM for extreme bandwidth systems”. In: Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture. 2017, pp. 41–54.