High-throughput Pairwise Alignment with the Wavefront Algorithm using Processing-in-Memory

Safaa Diab[1], Amir Nassereldine[1], Mohammed Alser[2], Juan Gómez Luna[2], Onur Mutlu[2], Izzat El Hajj[1]

[1]*American University of Beirut, Lebanon* [2]*ETH Zürich, Switzerland*

HiCOMB 2022
May 30, 2022

Executive Summary

- **Problem:**
  - Genome sequencing analysis is bottlenecked by the data-intensive sequence alignment algorithms used in the read mapping phase.

- **Motivation:**
  - Processing-in-Memory (PIM) alleviates memory bandwidth limitations of existing systems by enabling computation inside the memory without the need to move data.
  - UPMEM developed the first general-purpose real-world PIM architecture.

- **We show that the Wavefront Alignment (WFA) algorithm can achieve substantially higher pairwise read alignment throughput on the PIM system than on a server-grade multi-threaded CPU system.**

Data Movement Bottleneck

- Modern workloads spend a significant portion of execution time and energy moving data between main memory and computing units through high latency and limited bandwidth memory bus.

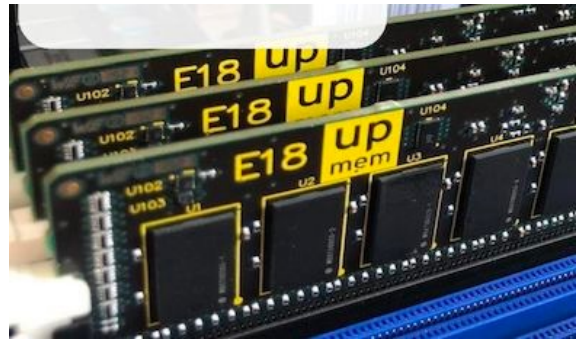
- PIM provides a memory-centric solution that alleviates the limitation factor of memory-bounded (low data-reuse) workloads such as:
  - Genomics
  - Database index search
  - 3D image reconstruction & FFT
  - Compression/Decompression

- Genome analysis utilizes data-intensive sequence alignment algorithms to align billion of read pairs simultaneously.
  - Bottlenecked by the memory bandwidth limitations of existing systems.

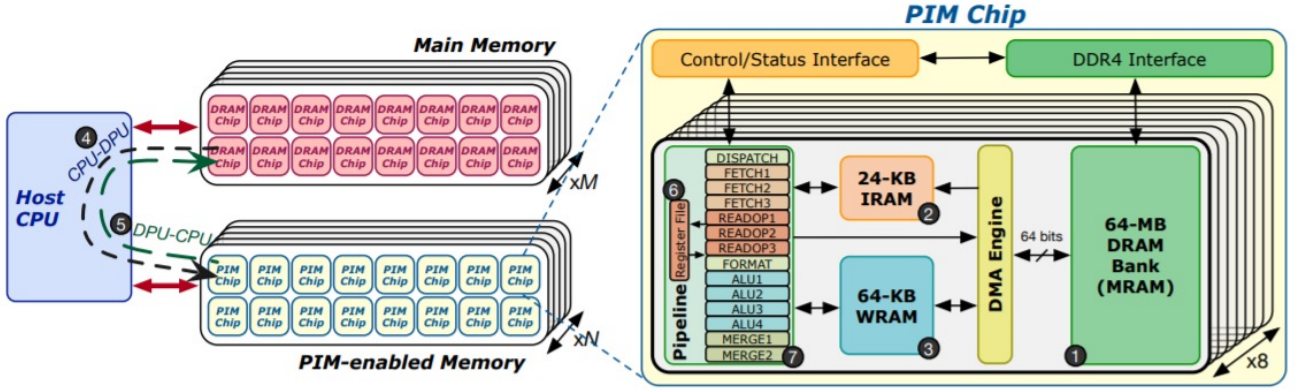
- PIM can accelerate sequencing alignment algorithms by reducing the large number of data transfers required.

UPMEM Processing-in-DRAM Engine (2019)

- **UPMEM DDR4 DIMM modules**: large number of DRAM arrays + general purpose processing cores.
  - Work as a parallel coprocessor connected to the main memory of a host where an x86 platform can have Up to 20 UPMEM DIMMs plugged



- Each DDR4 R-DIMM module consists of 16 PIM enabled chips
- Within each PIM chip there are 8 DRAM Processing Units (DPUs)
- Each DPU works independently and has:
  - 32-bit RISC processor, 24 hardware threads
  - 64MB Main RAM (MRAM) banks
  - 64KB Working RAM (WRAM)
  - 24KB Instruction memory (IRAM)



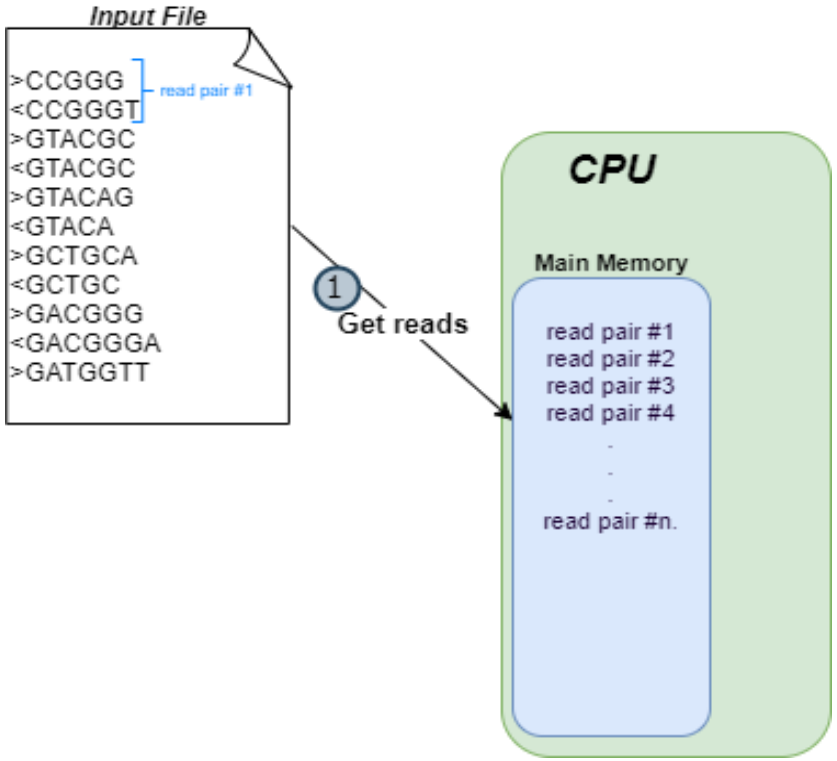
- UPMEM follows the Single Program Multiple Data (SPMD) programming model

https://www.anandtech.com/show/14750/hot-chips-31-analysis-inmemory-processing-by-upmem

*Juan G´omez-Luna, Izzat El Hajj, Ivan Fernandez, Christina Giannoula, Geraldo F. Oliveira, and Onur Mutlu. Benchmarking a new paradigm: An experimental analysis of a real processing-in-memory architecture

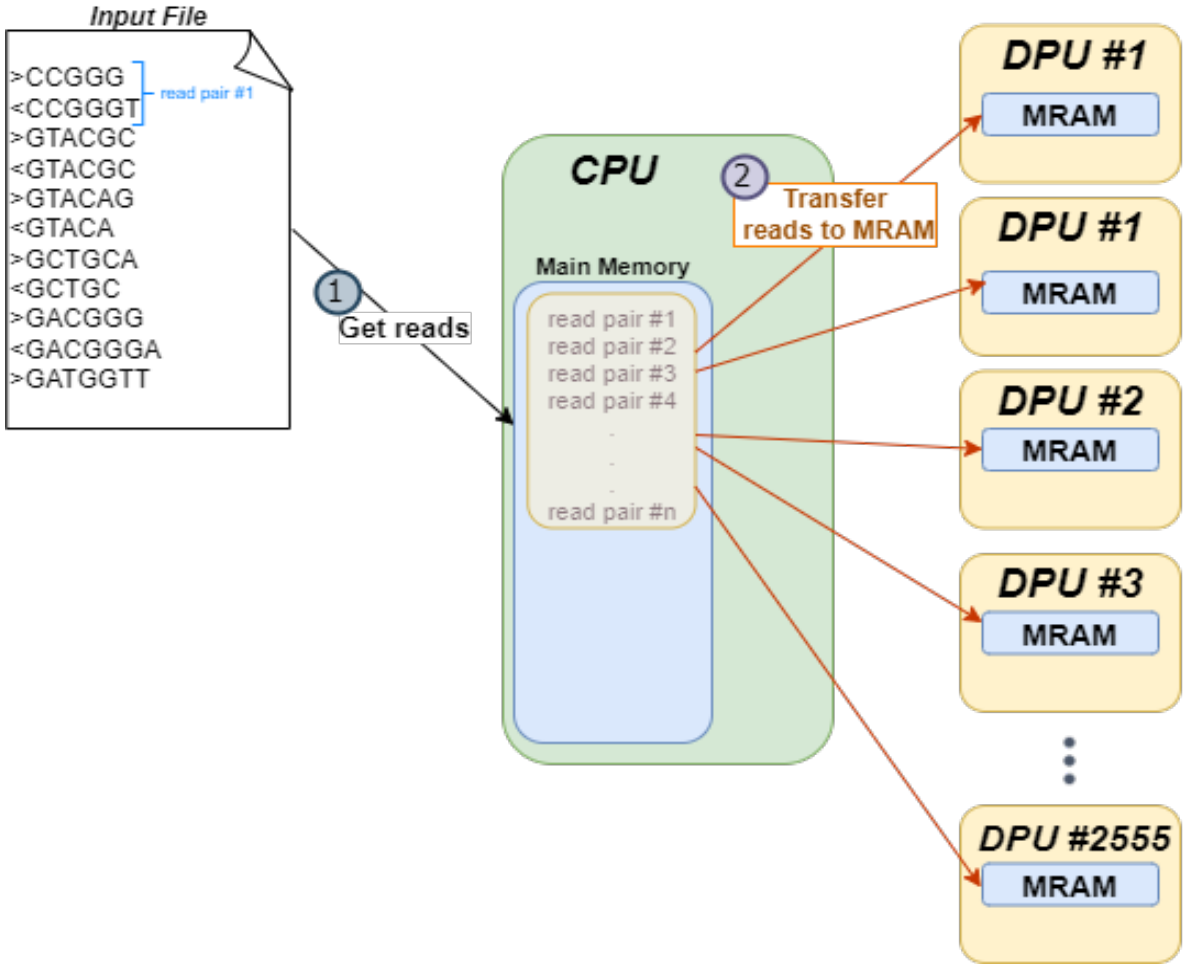
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



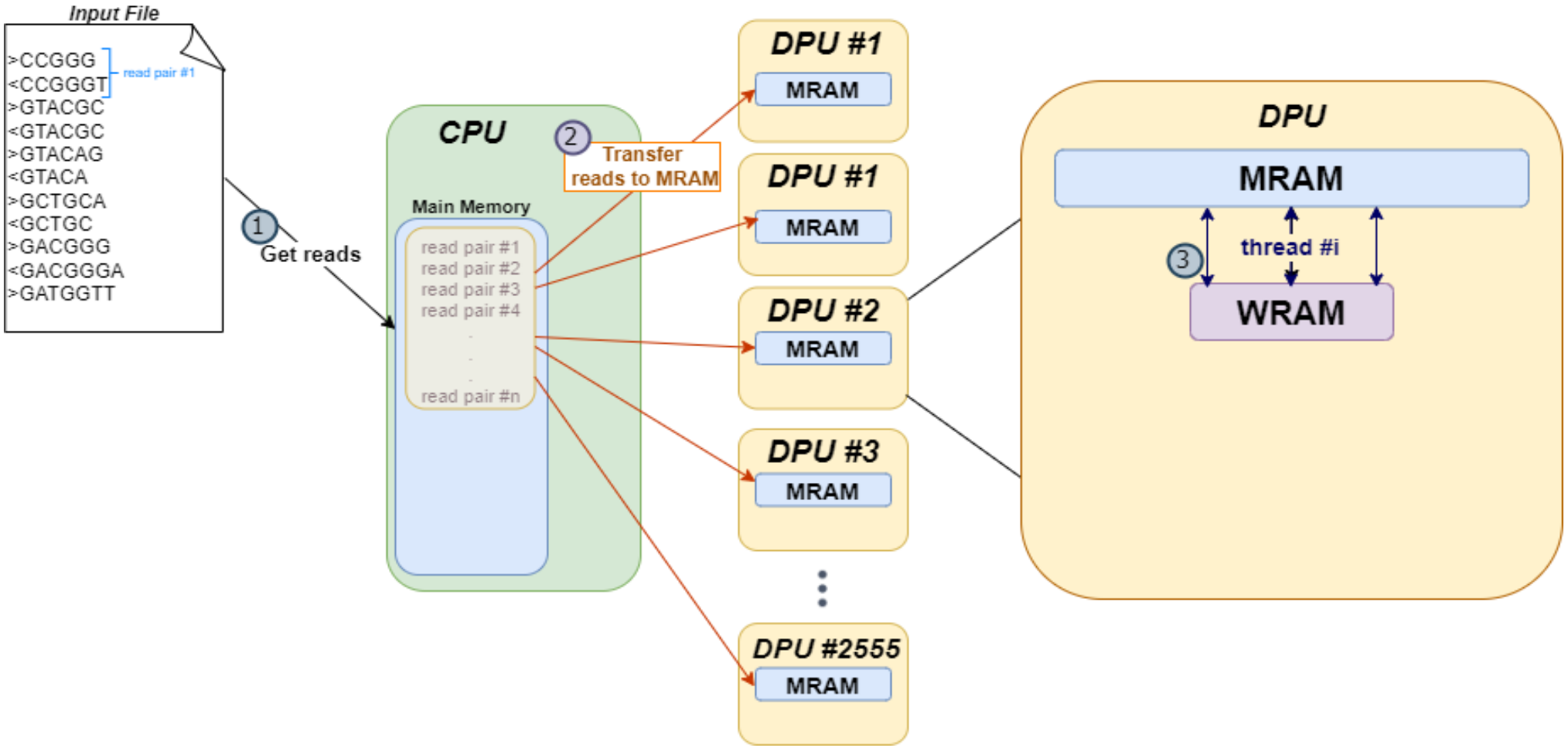
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



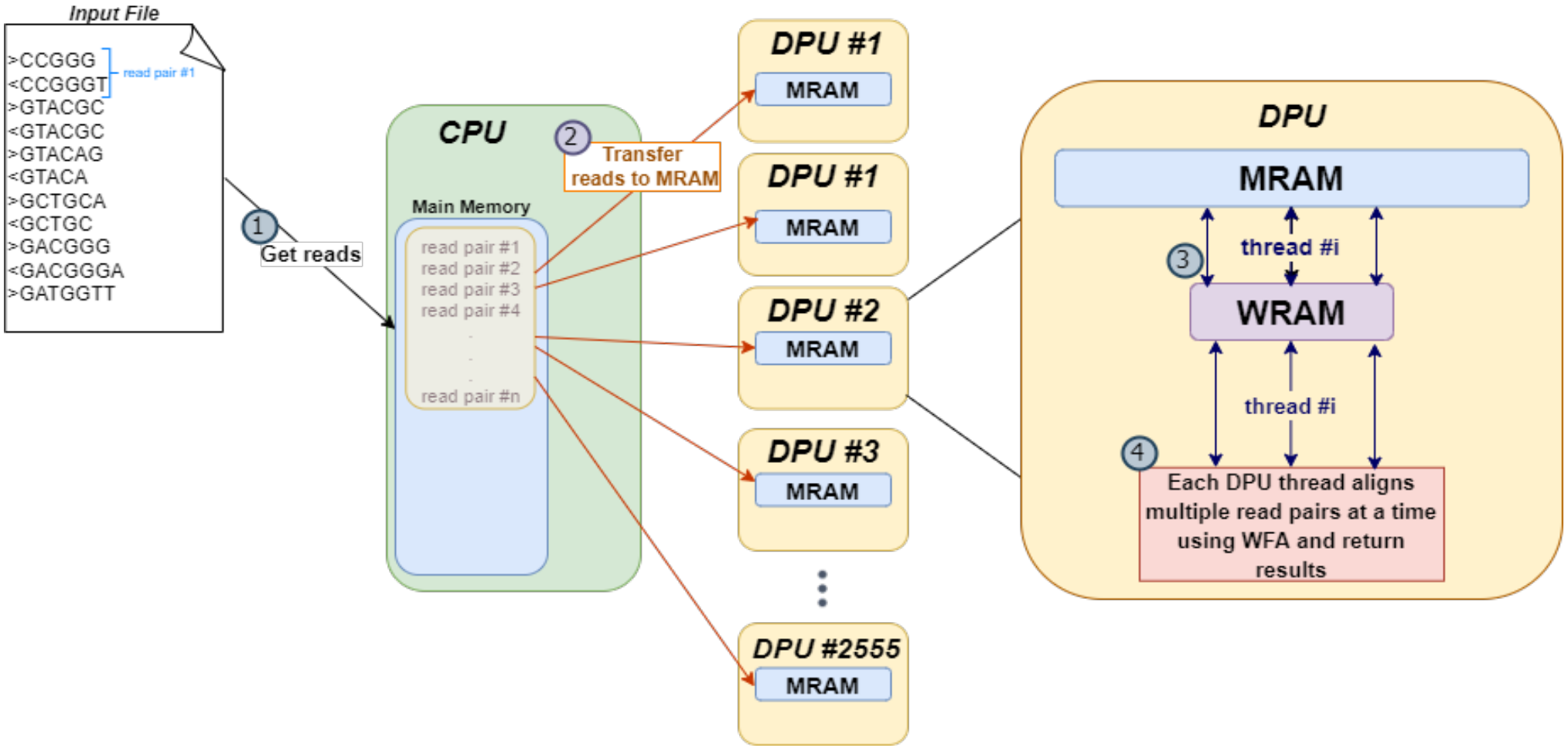
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



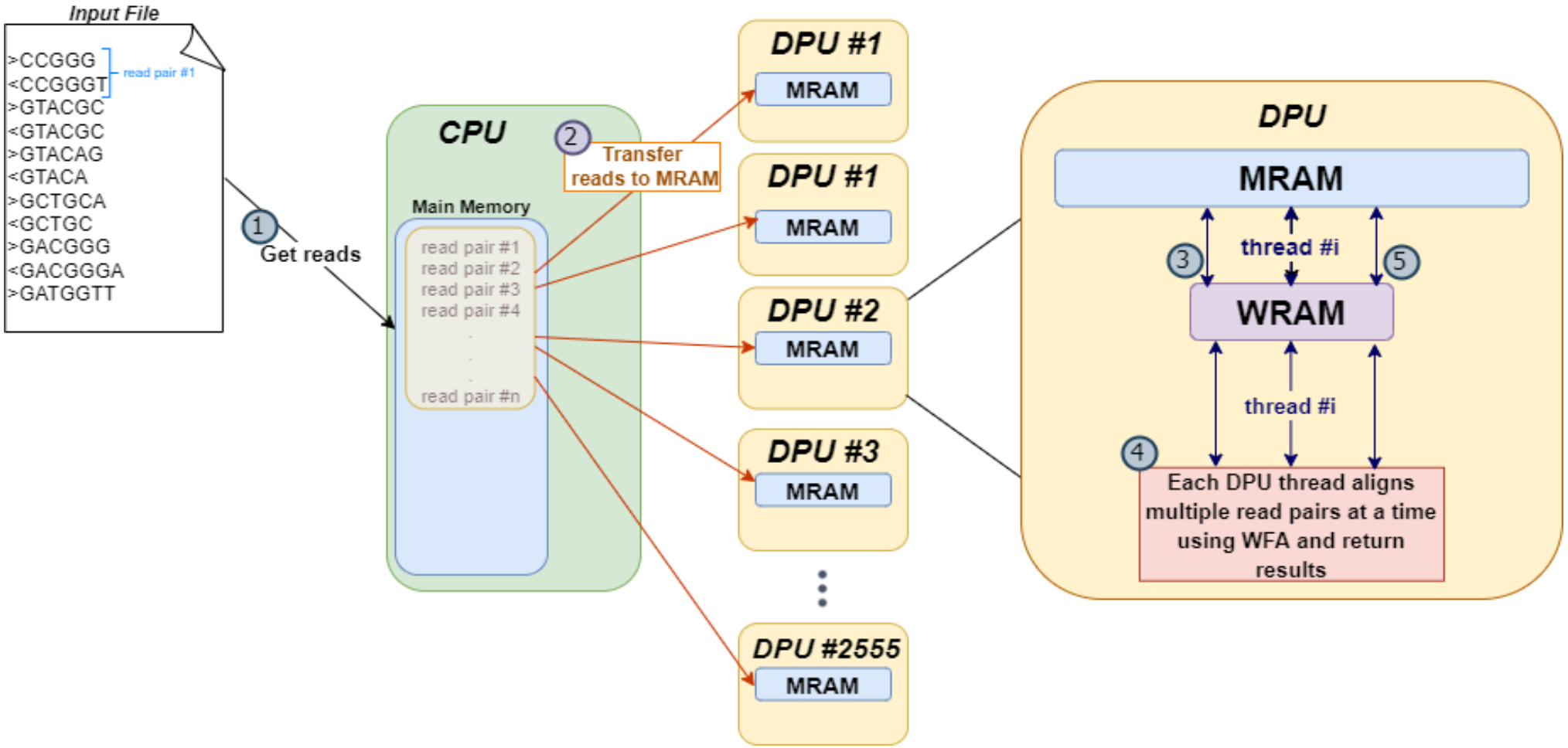
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



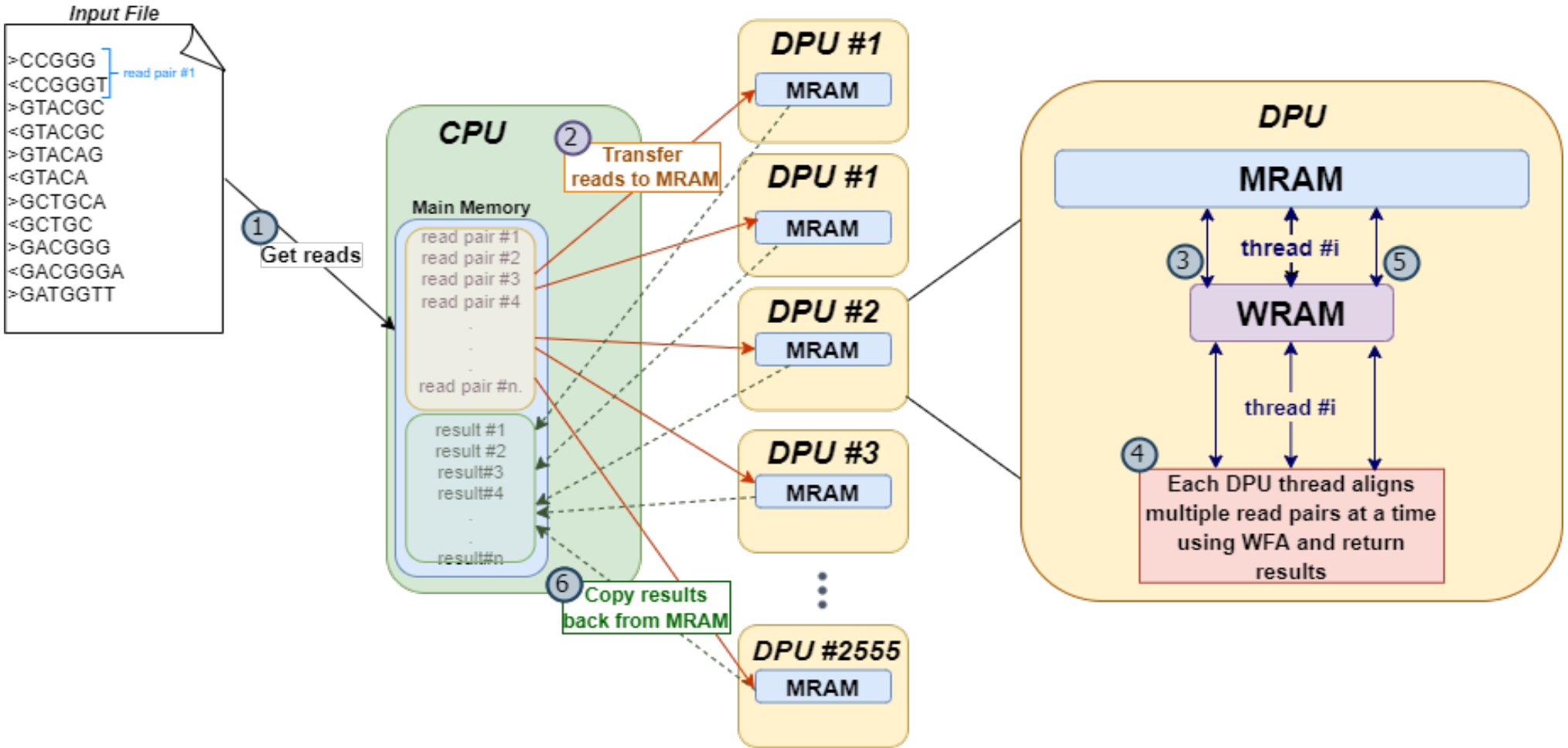
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



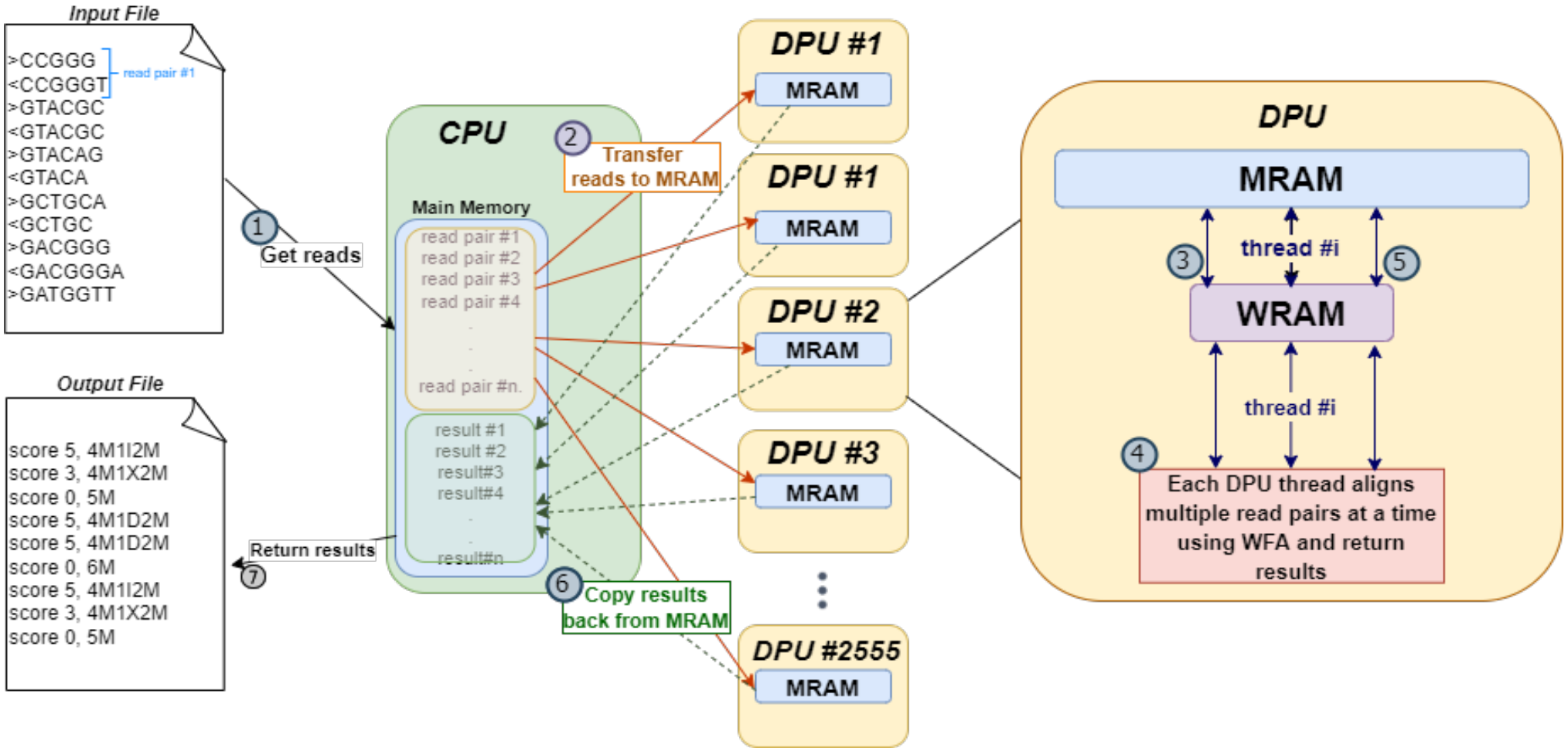
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



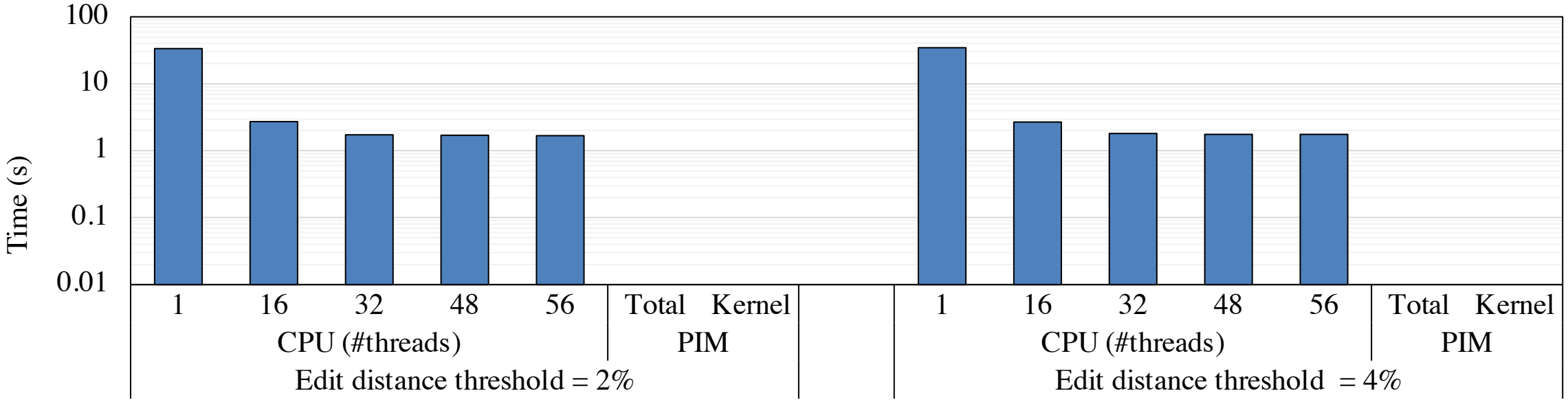
WFA on a PIM System: Implementation

- Implement state-of-the-art alignment algorithm WFA on **UPMEM-PIM architecture**
- Perform high-throughput read pair alignment to detect the peak throughput in which the implementation is efficient



Evaluation Model & Results

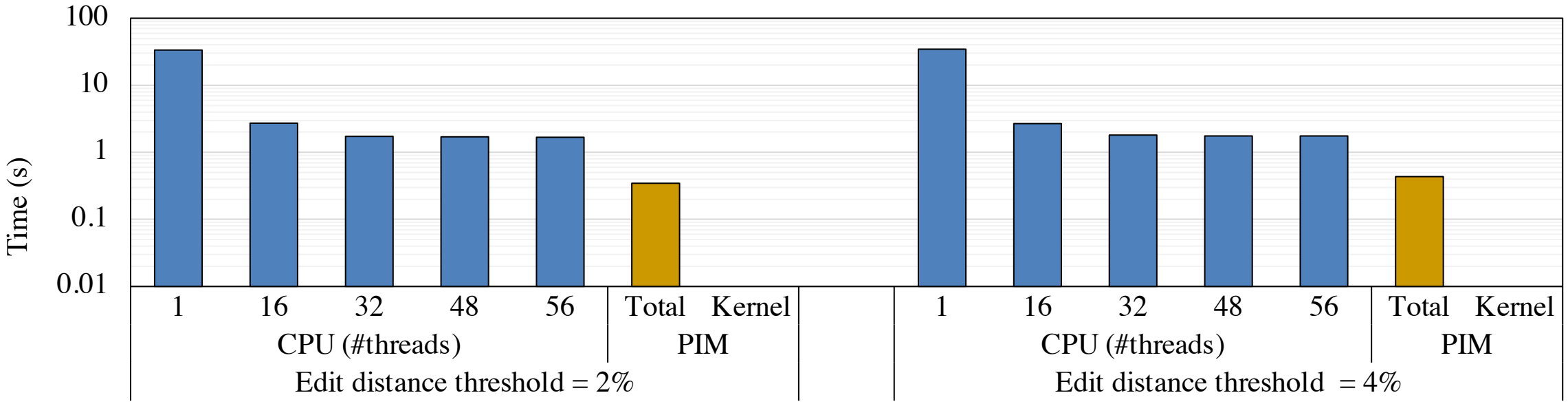
- Read Length: 100bp, Edit distance thresholds: 2% and 4%, Number of read pairs: 5Million
- CPU: Intel® Xeon® Gold 5120 Processor: 56 CPU threads, and 64 GB Memory
- PIM: 2,560 UPMEM DPUs at 425MHz and a total of 150GB MRAM



Observation #1: CPU performance **does not scale** when the number of CPU threads increase, which motivates the use of the PIM system.

Evaluation Model & Results

- Read Length: 100bp, Edit distance thresholds: 2% and 4%, Number of read pairs: 5Million
- CPU: Intel® Xeon® Gold 5120 Processor: 56 CPU threads, and 64 GB Memory
- PIM: 2,560 UPMEM DPUs at 425MHz and a total of 150GB MRAM

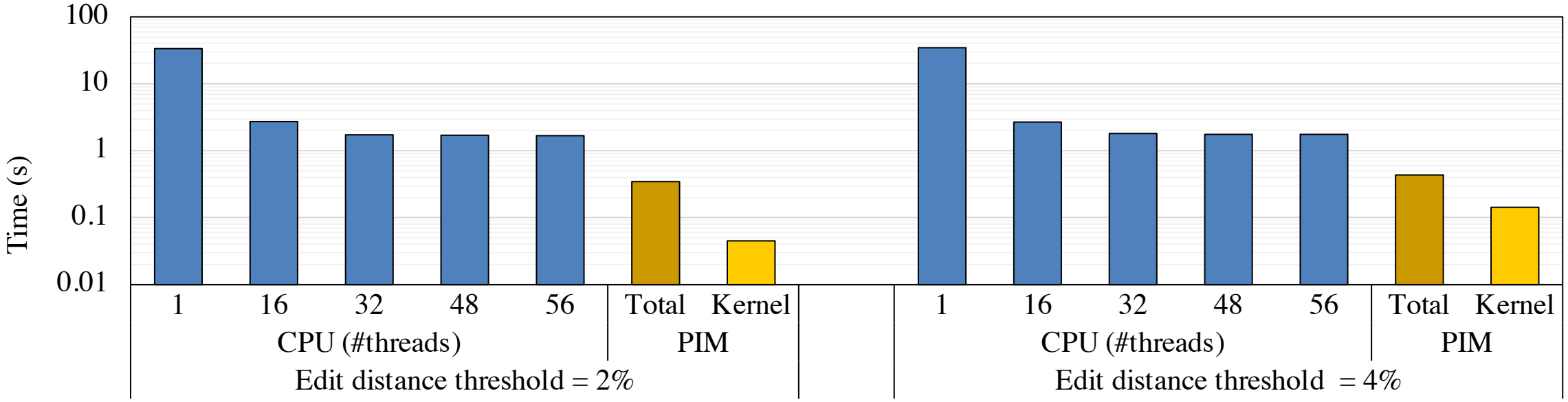


Observation #1: CPU performance **does not scale** when the number of CPU threads increase, which motivates the use of the PIM system.

Observation #2: Our PIM implementation total time achieves **4.87x (ED=2%) and 4.05x (ED=4%) speedup** over 56-thread CPU implementation.

Evaluation Model & Results

- Read Length: 100bp, Edit distance thresholds: 2% and 4%, Number of read pairs: 5Million
- CPU: Intel® Xeon® Gold 5120 Processor: 56 CPU threads, and 64 GB Memory
- PIM: 2,560 UPMEM DPUs at 425MHz and a total of 150GB MRAM



Observation #1: CPU performance **does not scale** when the number of CPU threads increase, which motivates the use of the PIM system.

Observation #2: Our PIM implementation total time achieves **4.87x (ED=2%) and 4.05x (ED=4%) speedup** over 56-thread CPU implementation.

Observation #3: PIM implementation kernel time achieves **37.4x and 12.3x higher speedup** when the CPU-DPU data transfer time is not accounted.

Conclusion

- **Problem:**
  - Read mapping phase of genome sequencing analysis is bottlenecked by the data-intensive sequence alignment algorithms.
- **Motivation:**
  - PIM alleviates memory bandwidth limitations of existing systems.
  - UPMEM-PIM is the first DRAM-processing engine
- **We show that the Wavefront Alignment (WFA) algorithm can achieve substantially higher pairwise read alignment throughput on the PIM system than on a server-grade multi-threaded CPU system.**
- **Future Work:**
  - Run experiments on longer read lengths and higher edit distance thresholds
  - Implement other alignment algorithms on the PIM system

High-throughput Pairwise Alignment with the Wavefront Algorithm using Processing-in-Memory

Safaa Diab[1], Amir Nassereldine[1], Mohammed Alser[2], Juan Gómez Luna[2], Onur Mutlu[2], Izzat El Hajj[1]

[1]*American University of Beirut, Lebanon* [2]*ETH Zürich, Switzerland*

Thank you!