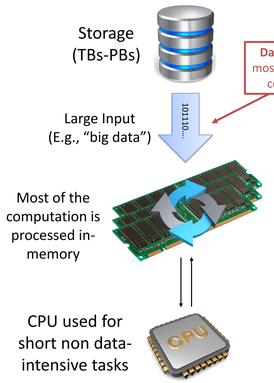
From Processing-in-Memory to Processing-in-Storage (PrinS)

Roman Kaplan

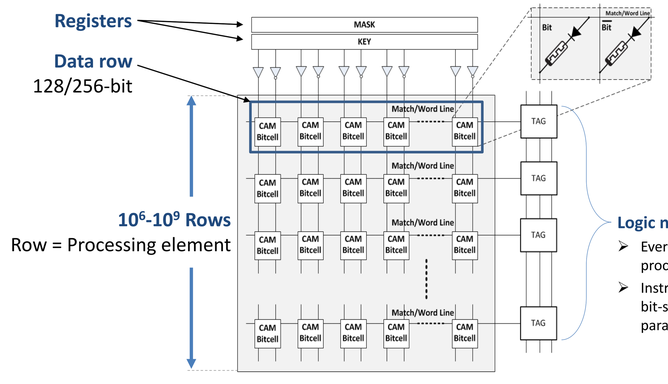
Advisor: Prof. Ran Ginosar In collaboration with: Dr. Leonid Yavits

Processing-in-Memory (PiM): Idea & Problems



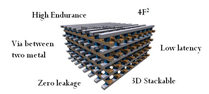
- PiM performs the bulk of computation in main memory
  - ✓ Reducing Data transfers
  - ✓ Saving time & energy
- Data transfers from storage to main memory are still required
- Storage ⇔ main memory is the most energetically expensive link
- Moving large amounts of data takes a lot of time and energy
- To reach 100x lower energy and higher performance we **must bring processing closer to data**

Resistive Content Addressable Memory (ReCAM): A PrinS Device



Resistive Bit-cells

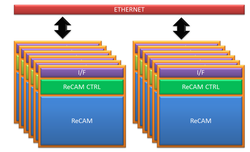
- Based on resistive technology (e.g., memristors)
- High density & non-volatile



Logic next to storage

- Every data row also has a processing unit
- Instructions are performed bit-serially on all PUs in parallel

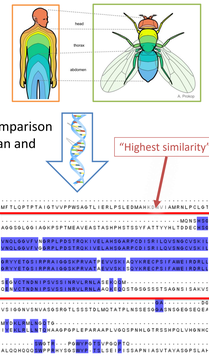
- Entire ReCAM crossbar array can be GBs in size divided to separate ICs
  - ✓ E.g., IC can be 256MB is size
  - ✓ 256MB = 8M rows
  - ✓ Future technology might allow TBs of storage



- *In-situ* processing:
  - ✓ Bit-wise logic
  - ✓ Algebraic operations
  - ✓ Column-shift
  - ✓ Row-wise comparison
  - ✓ CAM operations (explained in 'In-Storage Deduplication' below)
- Bit-cell switching time can allow for a GHz operational frequency

PrinS Application: DNA Local Sequence Alignment

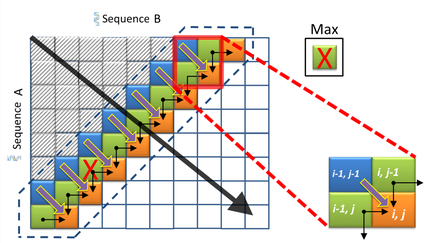
Smith-Waterman: Local Sequence Alignment



- The Smith-Waterman (S-W) algorithm finds regions with "highest similarity" between two sequences (DNA or protein)
  - ✓ Proven to be optimal
- Every "match", "mismatch" & "gap" inside the pair of sequences has a score
- S-W is based on dynamic programming
  - ✓ Fills a $m \times n$ matrix
  - ✓ Has a quadratic computational complexity $O(m \cdot n)$
- Two main steps in the algorithm: (1) scoring and (2) traceback
  - ✓ We implemented the scoring step – more computationally demanding

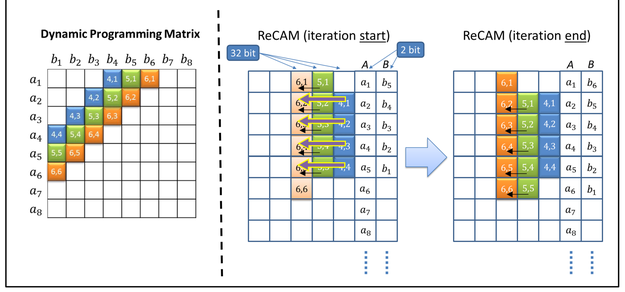
Parallel S-W

- Matrix-fill order is on the main diagonal
- The entire antidiagonal is calculated in parallel
  - ✓ Antidiagonal per group of ReCAM columns
- We search for the maximal score
- Only 3 antidiagonals are stored in each iteration



In-ReCAM S-W

- Every score is a 32-bit integer = 32 ReCAM bitcells
- DNA sequence is stored in 2 columns (2 bits per base-pair)
- Sequence B shifts down at the beginning of an iteration



Simulations

- Cycle-accurate simulator: 8GB of storage @500MHz
- Compared to multi-accelerator state-of-the-art solutions: FPGA, Xeon Phi and GPU

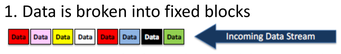
Performance Compared to State-of-the-Art

Accelerator	FPGA	Xeon Phi	GPU	ReCAM
Performance (TCUPS)	6.02	0.23	11.08	**52.68**
# of ICs	128	4	**384**	32
Publication	L.Wienbrandt. **The FPGA-based High-Performance Computer RIVYERA for Applications in Bioinformatics**. In Languages, I/O, Limits: 10th OE, pages 183-192. Springer 2014.	Y. Liu and B. Schmidt. 2014b. **SWAPHI: Smith-Waterman protein database search on Xeon Phi coprocessors**. In IEEE ASAP, 184–185.	Sanders, E. F. O., et al. **"CUDAlign 4.0: Incremental Speculative Traceback for Exact Chromosome-Wide Alignment in GPU Clusters."** IEEE	

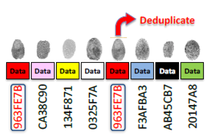
PrinS Application: In-Storage Deduplication

What is Deduplication?

- Deduplication is a technique for storing a single copy of each data block in storage
  - ✓ Can reach 10x reduction in data volume
- How it works:
  1. Data is broken into fixed blocks

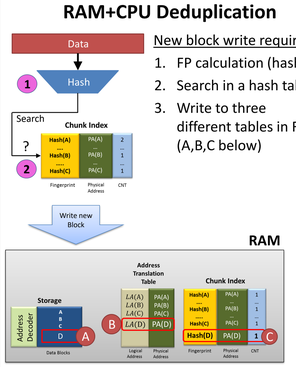


  2. A fingerprint (FP) is calculated for each block
  3. Only pointers are stored for identical blocks



Traditional (RAM+CPU) vs. In-ReCAM Deduplication

RAM+CPU Deduplication

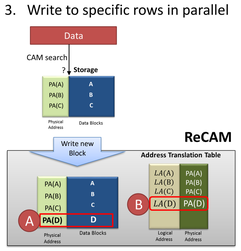


New block write requires:

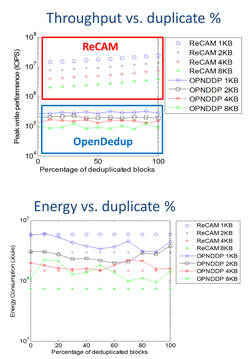
1. FP calculation (hash)
2. Search in a hash table
3. Write to three different tables in RAM (A,B,C below)

ReCAM

- Use CAM operations:
  1. Compare all storage content to a specific word → No need to hash
  2. Write to specific column in parallel
  3. Write to specific rows in parallel



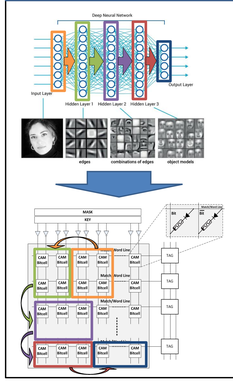
Simulations: ReCAM vs. OpenDedup



- ReCAM was simulated with a cycle-accurate simulator
  - ✓ ReCAM size = 256GB
  - ✓ Frequency = 1GHz
- Opendedup was executed on a high-end server for comparison: 4×8 octa-core CPU, 64GB of RAM and 800GB SSD drive
- **ReCAM has 100× higher throughput than deduplication with RAM+CPU**
- Energy consumption is similar or lower for the common block sizes (4 & 8KB)

Future: Deep Learning

Next Work



- Nowadays Deep Learning for large nets is done in cloud
- No energy figures reported, main concern is performance
- Deep Learning has two parts:
  (1) Feedforward (2) Backpropagation
  Each requires different set of tools

Main Questions

- ❖ Which instruction set?
- ❖ Do we need new modules? (E.g., sum reduction)
- ❖ Will we be able to solve larger problems with PrinS?