Real-time Analysis of Genomic Sequences from Nanopore Electrical Signals by Fast and Accurate Hash-based Search

Can Firtina
canfirtina@gmail.com
https://cfirtina.com

3 May 2024

Tufts University

SAFARI

ETH zürich

Brief Self Introduction

- **Can Firtina**
  - Senior Ph.D. student in the SAFARI Research Group at ETH Zurich

- **Research interests:** Bioinformatics & Computer Architecture
  - Real-time genome analysis
  - Similarity search in a large space of genomic data
  - Hardware-Algorithm co-design to accelerate genome analysis
  - Genome editing
  - Error correction

- Get to know **our group and our research**
  - **Group website:** https://safari.ethz.ch/
  - **Contact me:** canfirtina@gmail.com
  - **Website:** https://cfirtina.com
  - **Twitter (aka X):** https://twitter.com/FirtinaC

Professor Mutlu

- Onur Mutlu
  - Full Professor @ ETH Zurich ITET (INFK), since September 2015
  - Strecker Professor @ Carnegie Mellon University ECE/CS, 2009-2016, 2016-…
  - PhD from UT-Austin, worked at Google, VMware, Microsoft Research, Intel, AMD
  - https://people.inf.ethz.ch/omutlu/
  - omutlu@gmail.com (Best way to reach)
  - https://people.inf.ethz.ch/omutlu/projects.htm
- Research and Teaching in:
  - Computer architecture, computer systems, hardware security, bioinformatics
  - Memory and storage systems
  - Hardware security, safety, predictability
  - Fault tolerance
  - Hardware/software cooperation
  - Architectures for bioinformatics, health, medicine
  - …

SAFARI Research Group

Computer architecture, HW/SW, systems, bioinformatics, security, memory

SAFARI
SAFARI Research Group
safari.ethz.ch

40+ Researchers

Think BIG, Aim HIGH!

Four Key Current Directions

- Fundamentally Secure/Reliable/Safe Architectures
- Fundamentally Energy-Efficient Architectures
  - Memory-centric (Data-centric) Architectures
- Fundamentally Low-Latency and Predictable Architectures
- Algorithms & Architectures for AI/ML, Genomics, Medicine

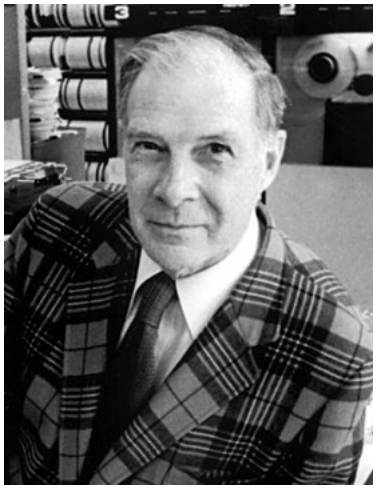
Agenda for Today

- Cutting-edge in Accelerating Genome Analysis
- Enabling Fast and Accurate Real-time Analysis
  - RawHash and RawHash2
- Conclusion

The Goal of Computing: Beyond Numbers

"The purpose of computing is [to gain] insight, not numbers"

Richard Hamming

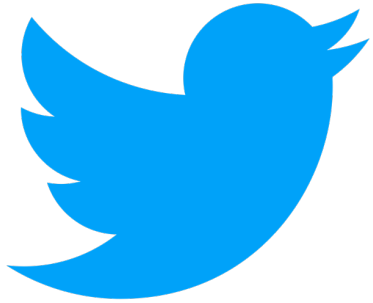


"Numerical Methods for Scientists and Engineers," Richard Hamming, 1962.

We need to gain insights
and observations
much more efficiently
than ever before

Big Data is Everywhere

Astronomy
25 zetta-bytes/year



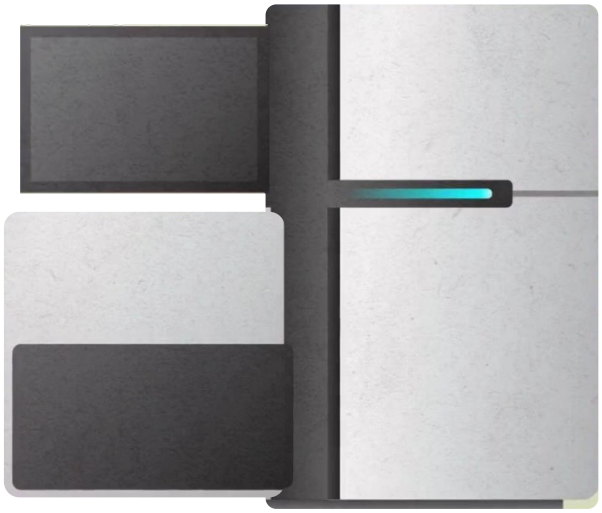
Twitter (now X)
0.5-15 billion tweets/year

YouTube
500-900 million hours/year

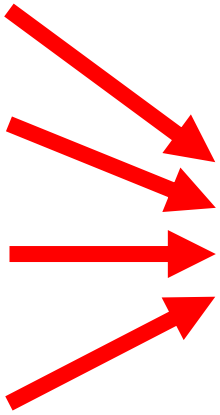
Genomics
1 zetta-bases/year

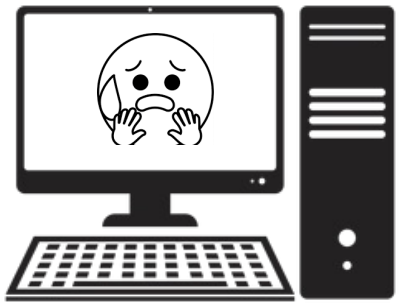
"Big data: astronomical or genomical?", PLoS biology, 2015.

Problems with Data Analysis Today



Special-Purpose Machine
for **Data Generation**





General-Purpose Machine
for **Data Analysis**

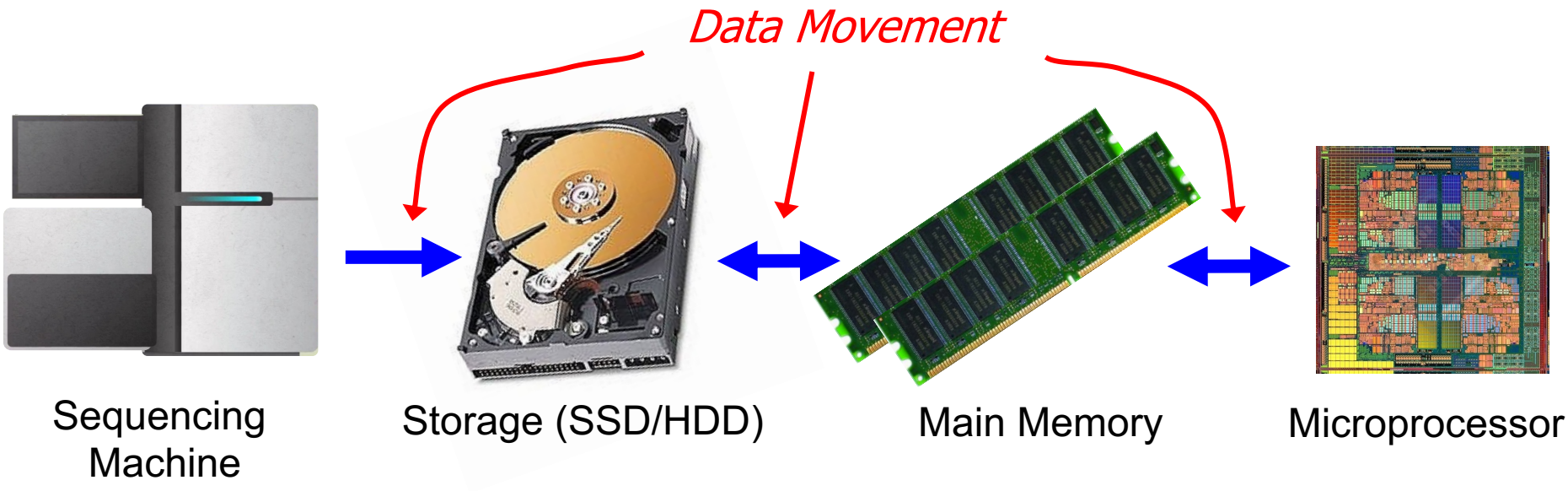
FAST

SLOW

Slow and inefficient processing capability
Large amounts of data movement

Data Movement Dominates Performance

- **Data movement** dominates performance and is a **major** system **energy bottleneck** (accounting for 40%-62%)

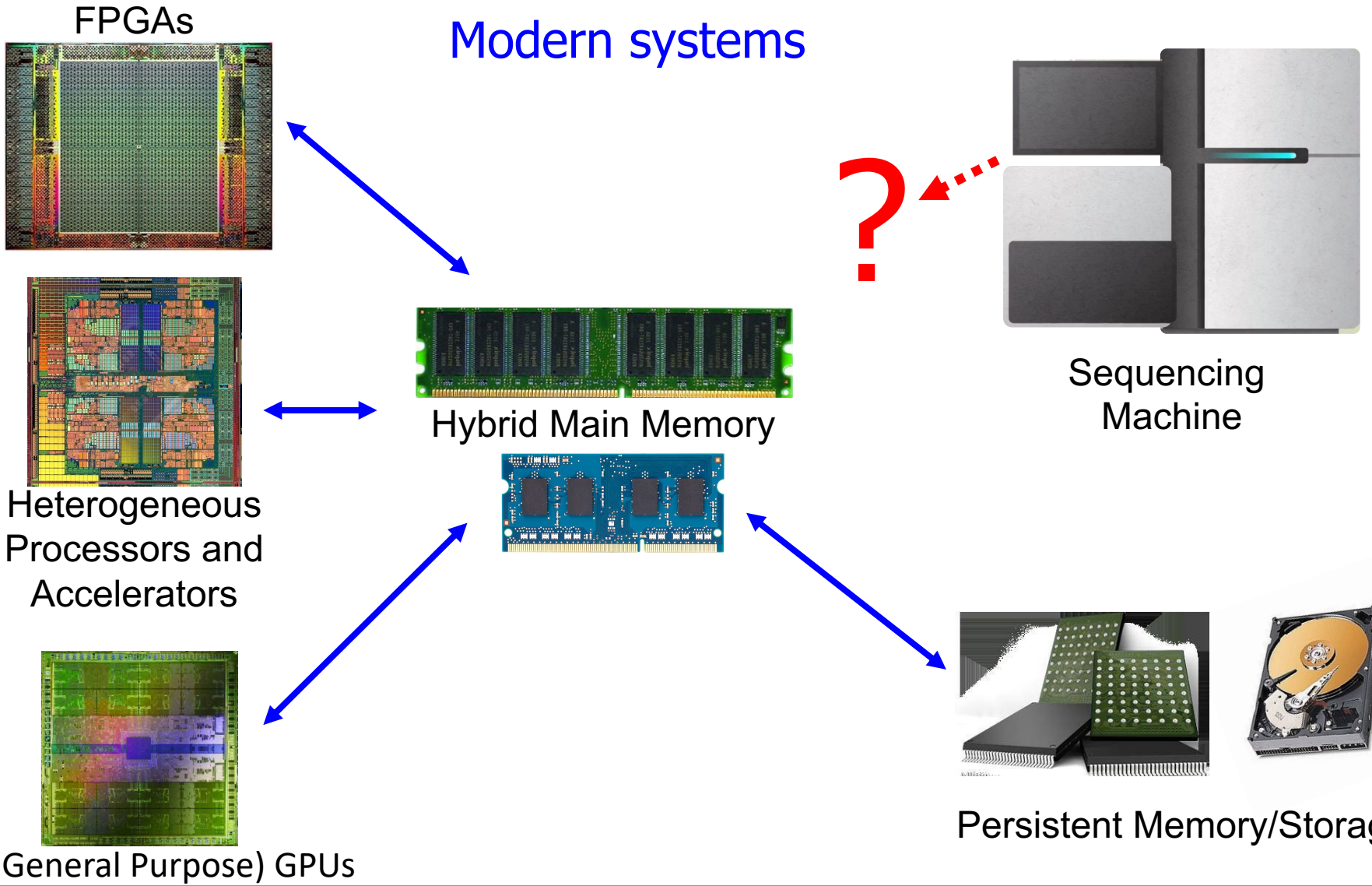


Single memory request consumes >160x-800x more energy compared to performing an addition operation

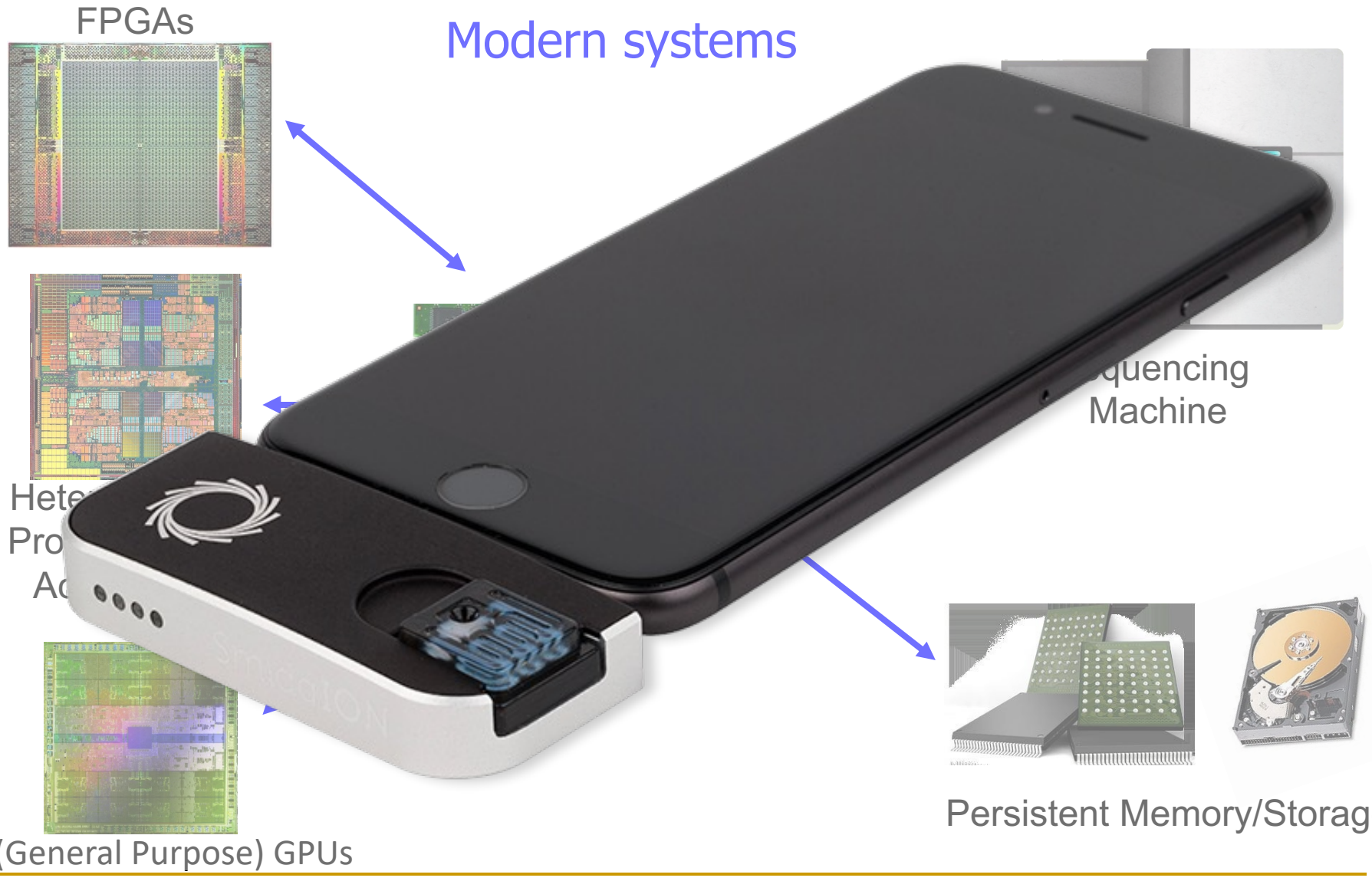
* Boroumand et al., "Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks," ASPLOS 2018
⋆ Kestor et al., "Quantifying the Energy Cost of Data Movement in Scientific Applications," IISWC 2013
☆ Pandiyan and Wu, "Quantifying the energy cost of data movement for emerging smart phone workloads on mobile platforms," IISWC 2014

We need intelligent algorithms and intelligent architectures that handle data well

Pushing Towards New Architectures



Pushing Towards New Architectures



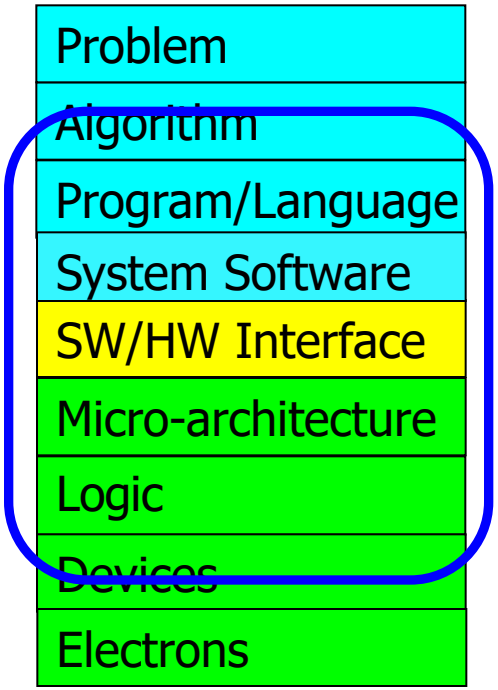
https://nanoporetech.com/products/smidgion

Accelerating Genome Analysis [DAC 2023]

- Onur Mutlu and Can Firtina,
  "Accelerating Genome Analysis via Algorithm-Architecture Co-Design"
  *Invited Special Session Paper in Proceedings of the 60th Design Automation Conference (**DAC**)*, San Francisco, CA, USA, July 2023.
  [Slides (pptx) (pdf)]
  [Talk Video (38 minutes, including Q&A)]
  [Related Invited Paper]
  [arXiv version]

Algorithm-Arch-Device Co-Design is Critical

Computer Architecture (expanded view)



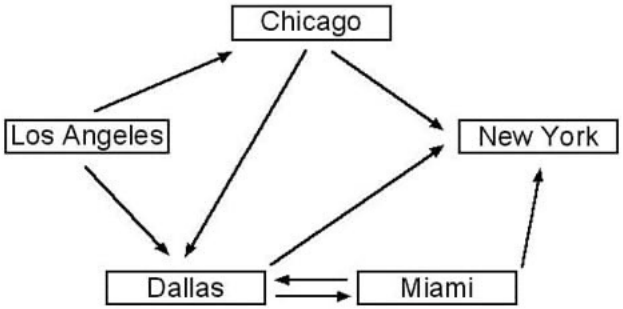
Applications
are only limited
by our imagination

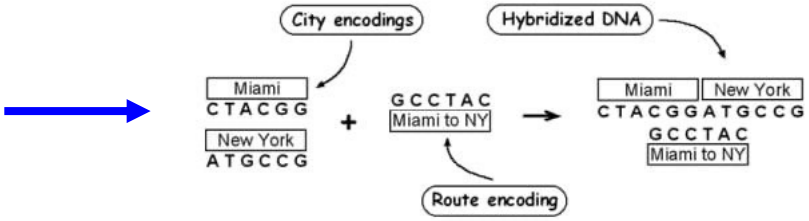
Genome Editing

The Nobel Prize in Chemistry 2020

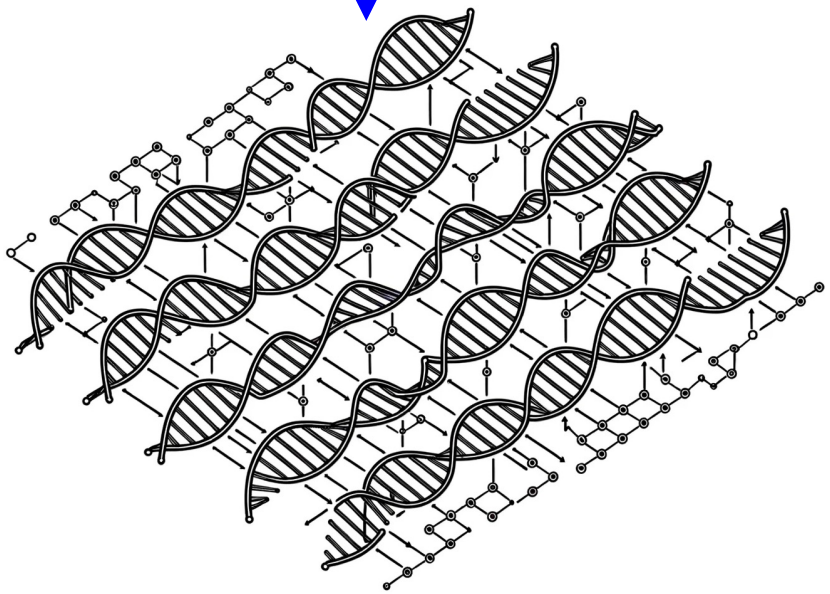
awarded "for the development of a method of genome editing"

DNA Computing





Massive parallelism to solve (hard) problems!



New Genome Sequencing Technologies

Nanopore sequencing technology and tools for genome assembly: computational analysis of the current state, bottlenecks and future directions

Damla Senol Cali ✉, Jeremie S Kim, Saugata Ghose, Can Alkan, Onur Mutlu

Briefings in Bioinformatics, bby017, https://doi.org/10.1093/bib/bby017
Published: 02 April 2018 **Article history** ▾

Oxford Nanopore MinION

Senol Cali+, "**Nanopore Sequencing Technology and Tools for Genome Assembly: Computational Analysis of the Current State, Bottlenecks and Future Directions**," Briefings in Bioinformatics, 2018.
[Open arxiv.org version]

New Frontiers: Raw Signal Analysis [ISMB 2023]

- Can Firtina, Nika Mansouri Ghiasi, Joel Lindegger, Gagandeep Singh, Meryem Banu Cavlak, Haiyu Mao, and Onur Mutlu,
  "RawHash: Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes"
  *Proceedings of the 31st Annual Conference on Intelligent Systems for Molecular Biology (**ISMB**) and the 22nd European Conference on Computational Biology (**ECCB**)*, Jul 2023
  [Bioinformatics Journal version]
  [Slides (pptx) (pdf)]
  [RawHash Source Code]

Bioinformatics, 2023, **39**, i297–i307
https://doi.org/10.1093/bioinformatics/btad272
ISMB/ECCB 2023

OXFORD

RawHash: enabling fast and accurate real-time analysis of raw nanopore signals for large genomes

Can Firtina [1,*], Nika Mansouri Ghiasi [1], Joel Lindegger [1], Gagandeep Singh [1], Meryem Banu Cavlak [1], Haiyu Mao [1], Onur Mutlu [1,*]

[1]Department of Information Technology and Electrical Engineering, ETH Zurich, 8092 Zurich, Switzerland

*Corresponding author. Department of Information Technology and Electrical Engineering, ETH Zurich, Gloriastrasse 35, 8092 Zurich, Switzerland.
E-mail: firtinac@ethz.ch (C.F.), omutlu@ethz.ch (O.M.)

Fast and Accurate Real-Time Genome Analysis

- Can Firtina, Melina Soysal, Joel Lindegger, and Onur Mutlu,
  "RawHash2: Mapping Raw Nanopore Signals Using Hash-Based Seeding and Adaptive Quantization"
  *Preprint on **arXiv***, September 2023.
  [arXiv version]
  [RawHash2 Source Code]

RawHash2: Mapping Raw Nanopore Signals Using Hash-Based Seeding and Adaptive Quantization

Can Firtina Melina Soysal Joël Lindegger Onur Mutlu
ETH Zürich

Fast and Accurate Real-Time Genome Analysis

- Joel Lindegger, Can Firtina, Nika Mansouri Ghiasi, Mohammad Sadrosadati, Mohammed Alser, and Onur Mutlu,
  "RawAlign: Accurate, Fast, and Scalable Raw Nanopore Signal Mapping via Combining Seeding and Alignment"
  Preprint on ***arXiv***, October 2023.
  [arXiv version]
  [RawAlign Source Code]

RawAlign: Accurate, Fast, and Scalable Raw Nanopore Signal Mapping via Combining Seeding and Alignment

Joël Lindegger[§] Can Firtina[§] Nika Mansouri Ghiasi[§]
Mohammad Sadrosadati[§] Mohammed Alser[§] Onur Mutlu[§]

[§]*ETH Zürich*

Machine Learning in Genomics

- M. Banu Cavlak, Gagandeep Singh, Mohammed Alser, Can Firtina, Joel Lindegger, Mohammad Sadrosadati, Nika Mansouri Ghiasi, Can Alkan, and Onur Mutlu, **"TargetCall: Eliminating the Wasted Computation in Basecalling via Pre-Basecalling Filtering"**
  *Proceedings of the 21st Asia Pacific Bioinformatics Conference (**APBC**)*, Changsha, China, April 2023.
  [TargetCall Source Code]
  [arxiv.org Version]
  [Talk Video at BIO-Arch 2023 Workshop]

TargetCall: Eliminating the Wasted Computation in Basecalling via Pre-Basecalling Filtering

Meryem Banu Cavlak[1] Gagandeep Singh[1] Mohammed Alser[1] Can Firtina[1] Joël Lindegger[1]
Mohammad Sadrosadati[1] Nika Mansouri Ghiasi[1] Can Alkan[2] Onur Mutlu[1]
[1]*ETH Zürich* [2]*Bilkent University*

https://arxiv.org/pdf/2301.09200.pdf

Genome Similarity Identification [NARGAB 2023]

- Can Firtina, Jisung Park, Mohammed Alser, Jeremie S. Kim, Damla Senol Cali, Taha Shahroodi, Nika Mansouri Ghiasi, Gagandeep Singh, Konstantinos Kanellopoulos, Can Alkan, and Onur Mutlu,
  "BLEND: A Fast, Memory-Efficient, and Accurate Mechanism to Find Fuzzy Seed Matches in Genome Analysis"
  NAR Genomics and Bioinformatics, March 2023.
  [Online link at NAR Genomics and Bioinformatics Journal]
  [arXiv preprint]
  [bioRxiv preprint]
  [BLEND Source Code]

Volume 5, Issue 1
March 2023

JOURNAL ARTICLE

BLEND: a fast, memory-efficient and accurate mechanism to find fuzzy seed matches in genome analysis

Can Firtina ✉, Jisung Park, Mohammed Alser, Jeremie S Kim, Damla Senol Cali, Taha Shahroodi, Nika Mansouri Ghiasi, Gagandeep Singh, Konstantinos Kanellopoulos, Can Alkan, Onur Mutlu ✉

NAR Genomics and Bioinformatics, Volume 5, Issue 1, March 2023, lqad004,

New Applications: Frequent Database Updates

- Jeremie S. Kim*, Can Firtina*, M. Banu Cavlak, Damla Senol Cali, Nastaran Hajinazar, Mohammed Alser, Can Alkan, and Onur Mutlu,
  "AirLift: A Fast and Comprehensive Technique for Remapping Alignments between Reference Genomes"
  Proceedings of the 21st Asia Pacific Bioinformatics Conference (***APBC***), Changsha, China, April 2023.
  [AirLift Source Code]
  [arxiv.org Version (pdf)]
  [Talk Video at BIO-Arch 2023 Workshop]

METHOD

AirLift: A Fast and Comprehensive Technique for Remapping Alignments between Reference Genomes

Jeremie S. Kim[1†], Can Firtina[1†], Meryem Banu Cavlak[2], Damla Senol Cali[3], Nastaran Hajinazar[1,4], Mohammed Alser[1], Can Alkan[2] and Onur Mutlu[1,2,3*]

 *Equal contribution

Error Correction using ML [Bioinform. 2020]

- Can Firtina, Jeremie S. Kim, Mohammed Alser, Damla Senol Cali, A. Ercument Cicek, Can Alkan, and Onur Mutlu,
  "Apollo: A Sequencing-Technology-Independent, Scalable, and Accurate Assembly Polishing Algorithm"
  Bioinformatics, June 2020.
  [Source Code]
  [Online link at Bioinformatics Journal]

Apollo: a sequencing-technology-independent, scalable and accurate assembly polishing algorithm

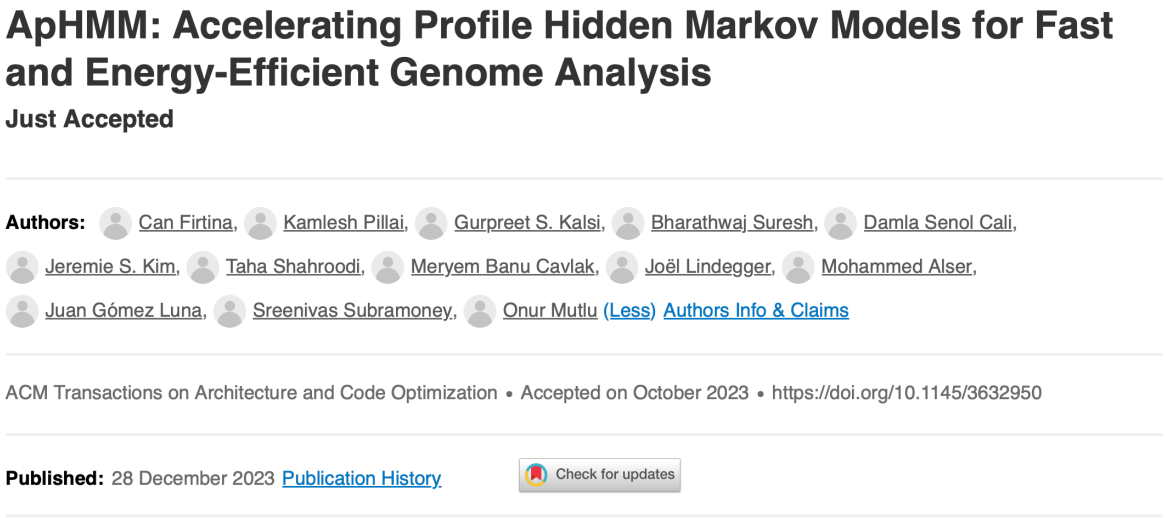
FREE

Can Firtina, Jeremie S Kim, Mohammed Alser, Damla Senol Cali, A Ercument Cicek, Can Alkan ✉, Onur Mutlu ✉

Bioinformatics, Volume 36, Issue 12, 15 June 2020, Pages 3669–3679,
https://doi.org/10.1093/bioinformatics/btaa179
Published: 13 March 2020 **Article history** ▾

Accelerating ML & Genome Graphs [ACM TACO '24]

- Can Firtina, Kamlesh Pillai, Gurpreet S. Kalsi, Bharathwaj Suresh, Damla Senol Cali, Jeremie S. Kim, Taha Shahroodi, Meryem Banu Cavlak, Joël Lindegger, Mohammed Alser, Juan Gómez Luna, Sreenivas Subramoney, and Onur Mutlu,
  "ApHMM: Accelerating Profile Hidden Markov Models for Fast and Energy-Efficient Genome Analysis"
  ACM TACO, Mar 2024.
  [Online link at ACM TACO]
  [arXiv preprint]
  [ApHMM Source Code]



ApHMM: Accelerating Profile Hidden Markov Models for Fast and Energy-Efficient Genome Analysis

Just Accepted

Authors: Can Firtina, Kamlesh Pillai, Gurpreet S. Kalsi, Bharathwaj Suresh, Damla Senol Cali, Jeremie S. Kim, Taha Shahroodi, Meryem Banu Cavlak, Joël Lindegger, Mohammed Alser, Juan Gómez Luna, Sreenivas Subramoney, Onur Mutlu (Less) Authors Info & Claims

ACM Transactions on Architecture and Code Optimization • Accepted on October 2023 • https://doi.org/10.1145/3632950

Published: 28 December 2023 Publication History

Check for updates

Accelerating String Matching [MICRO 2020]

- Damla Senol Cali, Gurpreet S. Kalsi, Zulal Bingol, Can Firtina, Lavanya Subramanian, Jeremie S. Kim, Rachata Ausavarungnirun, Mohammed Alser, Juan Gomez-Luna, Amirali Boroumand, Anant Nori, Allison Scibisz, Sreenivas Subramoney, Can Alkan, Saugata Ghose, and Onur Mutlu, **"GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis"**
  *Proceedings of the 53rd International Symposium on Microarchitecture (**MICRO**)*, Virtual, October 2020.
  [Lightning Talk Video (1.5 minutes)]
  [Lightning Talk Slides (pptx) (pdf)]
  [Talk Video (18 minutes)]
  [Slides (pptx) (pdf)]

GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis

Damla Senol Cali[†⋈] Gurpreet S. Kalsi[⋈] Zülal Bingöl[▽] Can Firtina[◇] Lavanya Subramanian[‡] Jeremie S. Kim[◇†]
Rachata Ausavarungnirun[⊙] Mohammed Alser[◇] Juan Gomez-Luna[◇] Amirali Boroumand[†] Anant Nori[⋈]
Allison Scibisz[†] Sreenivas Subramoney[⋈] Can Alkan[▽] Saugata Ghose[⋆†] Onur Mutlu[◇†▽]

[†]*Carnegie Mellon University* [⋈]*Processor Architecture Research Lab, Intel Labs* [▽]*Bilkent University* [◇]*ETH Zürich*
[‡]*Facebook* [⊙]*King Mongkut's University of Technology North Bangkok* [⋆]*University of Illinois at Urbana–Champaign*

Accelerating Genome Graphs [ISCA 2022]

- Damla Senol Cali, Konstantinos Kanellopoulos, Joel Lindegger, Zulal Bingol, Gurpreet S. Kalsi, Ziyi Zuo, Can Firtina, Meryem Banu Cavlak, Jeremie Kim, Nika MansouriGhiasi, Gagandeep Singh, Juan Gomez-Luna, Nour Almadhoun Alserr, Mohammed Alser, Sreenivas Subramoney, Can Alkan, Saugata Ghose, and Onur Mutlu,
  "SeGraM: A Universal Hardware Accelerator for Genomic Sequence-to-Graph and Sequence-to-Sequence Mapping"
  *Proceedings of the 49th International Symposium on Computer Architecture (**ISCA**)*, New York, June 2022.
  [arXiv version]

SeGraM: A Universal Hardware Accelerator for Genomic Sequence-to-Graph and Sequence-to-Sequence Mapping

Damla Senol Cali[1] Konstantinos Kanellopoulos[2] Joël Lindegger[2] Zülal Bingöl[3]
Gurpreet S. Kalsi[4] Ziyi Zuo[5] Can Firtina[2] Meryem Banu Cavlak[2] Jeremie Kim[2]
Nika Mansouri Ghiasi[2] Gagandeep Singh[2] Juan Gómez-Luna[2] Nour Almadhoun Alserr[2]
Mohammed Alser[2] Sreenivas Subramoney[4] Can Alkan[3] Saugata Ghose[6] Onur Mutlu[2]

[1]Bionano Genomics [2]ETH Zürich [3]Bilkent University [4]Intel Labs
[5]Carnegie Mellon University [6]University of Illinois Urbana-Champaign

In-Storage Genome Filtering [ASPLOS 2022]

- Nika Mansouri Ghiasi, Jisung Park, Harun Mustafa, Jeremie Kim, Ataberk Olgun, Arvid Gollwitzer, Damla Senol Cali, Can Firtina, Haiyu Mao, Nour Almadhoun Alserr, Rachata Ausavarungnirun, Nandita Vijaykumar, Mohammed Alser, and Onur Mutlu,
  "GenStore: A High-Performance and Energy-Efficient In-Storage Computing System for Genome Sequence Analysis"
  *Proceedings of the 27th International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, Virtual, February-March 2022.
  [Lightning Talk Slides (pptx) (pdf)]
  [Lightning Talk Video (90 seconds)]

GenStore: A High-Performance In-Storage Processing System for Genome Sequence Analysis

Nika Mansouri Ghiasi[1] Jisung Park[1] Harun Mustafa[1] Jeremie Kim[1] Ataberk Olgun[1]
Arvid Gollwitzer[1] Damla Senol Cali[2] Can Firtina[1] Haiyu Mao[1] Nour Almadhoun Alserr[1]
Rachata Ausavarungnirun[3] Nandita Vijaykumar[4] Mohammed Alser[1] Onur Mutlu[1]

[1]ETH Zürich [2]Bionano Genomics [3]KMUTNB [4]University of Toronto

Genome Analysis via PIM [MICRO 2022]

- Haiyu Mao, Mohammed Alser, Mohammad Sadrosadati, Can Firtina, Akanksha Baranwal, Damla Senol Cali, Aditya Manglik, Nour Almadhoun Alserr, and Onur Mutlu,
  "GenPIP: In-Memory Acceleration of Genome Analysis via Tight Integration of Basecalling and Read Mapping"
  *Proceedings of the 55th International Symposium on Microarchitecture (**MICRO**)*, Chicago, IL, USA, October 2022.
  [Slides (pptx) (pdf)]
  [Longer Lecture Slides (pptx) (pdf)]
  [Lecture Video (25 minutes)]
  [arXiv version]

GenPIP: In-Memory Acceleration of Genome Analysis via Tight Integration of Basecalling and Read Mapping

Haiyu Mao[1] Mohammed Alser[1] Mohammad Sadrosadati[1] Can Firtina[1] Akanksha Baranwal[1]
Damla Senol Cali[2] Aditya Manglik[1] Nour Almadhoun Alserr[1] Onur Mutlu[1]
[1]*ETH Zürich* [2]*Bionano Genomics*

Basecalling using PIM [MICRO 2023]

- Taha Shahroodi, Gagandeep Singh, Mahdi Zahedi, Haiyu Mao, Joel Lindegger, Can Firtina, Stephan Wong, Onur Mutlu, and Said Hamdioui,
  "Swordfish: A Framework for Evaluating Deep Neural Network-based Basecalling using Computation-In-Memory with Non-Ideal Memristors"
  Proceedings of the 56th International Symposium on Microarchitecture (MICRO), Toronto, ON, Canada, November 2023.
  [Slides (pptx) (pdf)]
  [arXiv version]

Swordfish: A Framework for Evaluating Deep Neural Network-based Basecalling using Computation-In-Memory with Non-Ideal Memristors

Taha Shahroodi[1] Gagandeep Singh[2,3] Mahdi Zahedi[1] Haiyu Mao[3] Joel Lindegger[3] Can Firtina[3]
Stephan Wong[1] Onur Mutlu[3] Said Hamdioui[1]

[1]TU Delft [2]AMD Research [3]ETH Zürich

Agenda for Today

- Cutting-edge in Accelerating Genome Analysis
- Enabling Fast and Accurate Real-time Analysis
  - RawHash and RawHash2
- Conclusion

ISMB/ECCB 2023

RawHash

Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes

Can Firtina

Nika Mansouri Ghiasi Joel Lindegger Gagandeep Singh

Meryem Banu Cavlak Haiyu Mao Onur Mutlu

RawHash RawHash2 Code

SAFARI ETH zürich

Outline

Background

RawHash

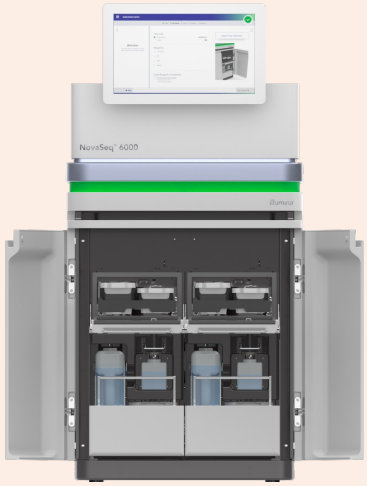
RawHash2

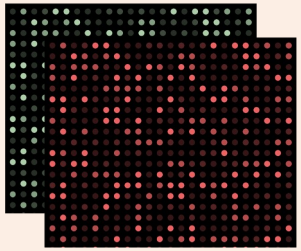
Evaluation

Conclusion

Different Raw Sequencing Data

Illumina

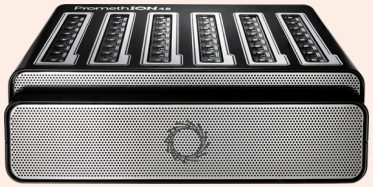


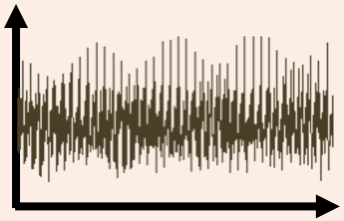


Multiple images

.BCL/.CBCL

Nanopore

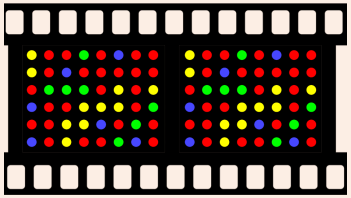




Electrical Signal

.POD5

PacBio



30-hour movie

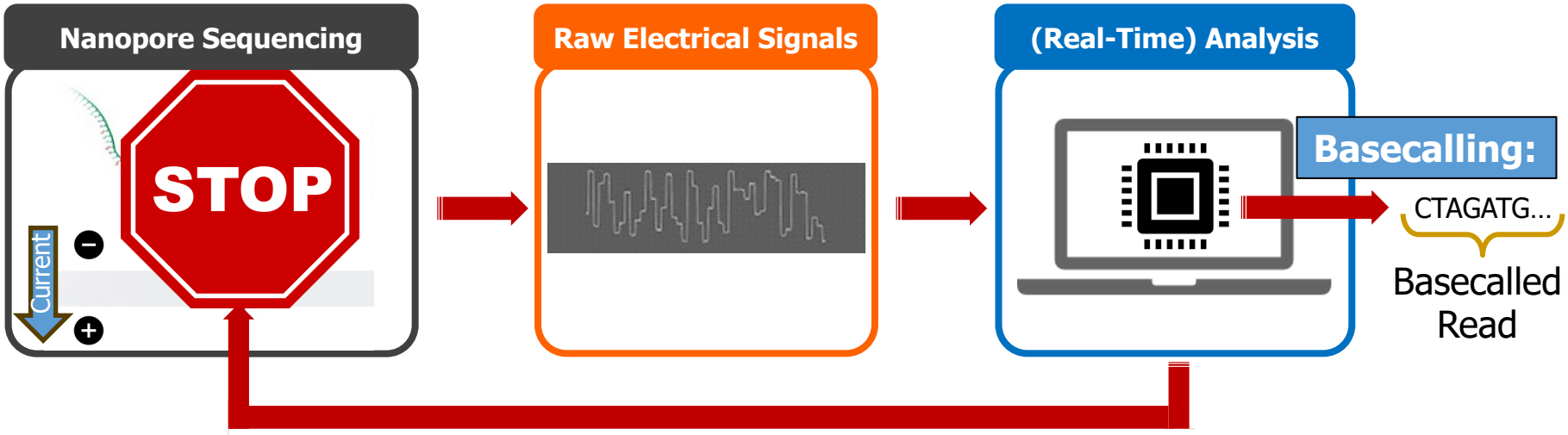
.BAM

Nanopore Sequencing

Nanopore Sequencing: a widely used sequencing technology

- **Long** reads (up to >2 million nucleotides)
- **Portable** sequencing
- Cost-effective
- More unique features: **real-time analysis**

Nanopore Sequencing & Real-time Analysis



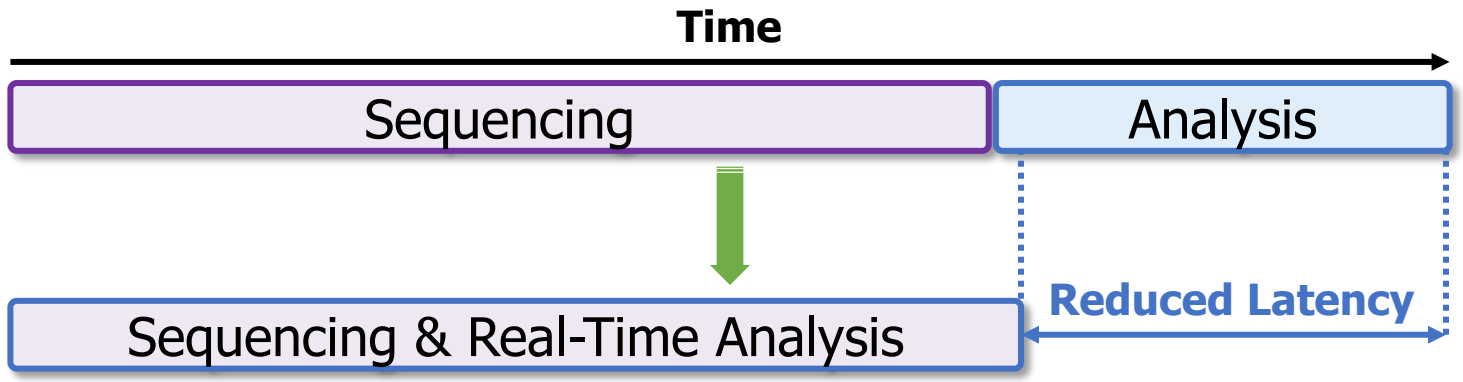
Raw Signals: Ionic current measurements generated as DNA passes through the **nanopore at a certain speed**

(Real-Time) Analysis: Translating to bases or directly analyzing raw signals

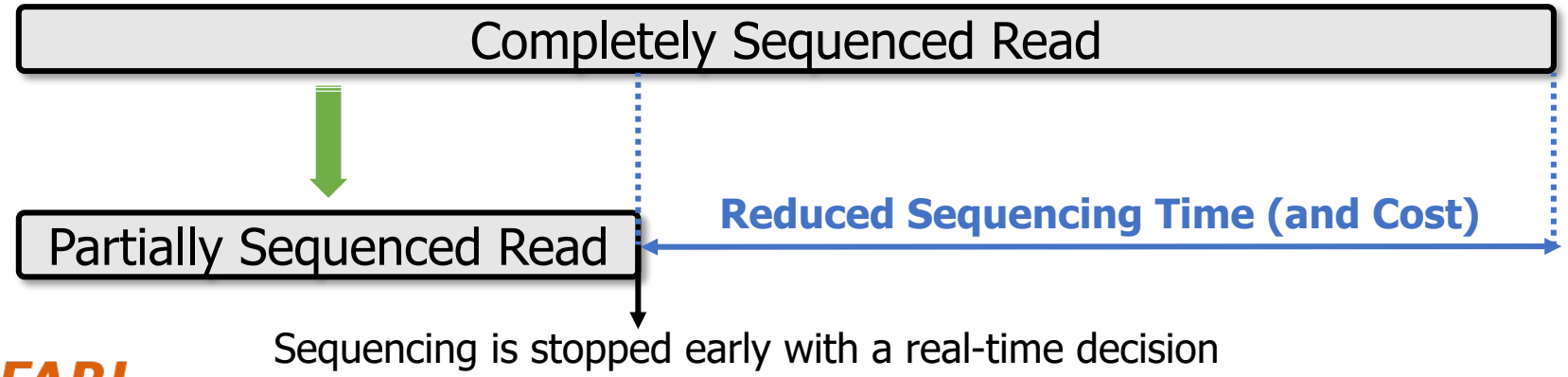
Real-Time Decisions: Stopping sequencing **early** based on real-time analysis

Benefits of Real-Time Analysis

- **Reducing latency** by overlapping the sequencing and analysis steps



- **Reducing sequencing time and cost** by stopping sequencing early



Challenges in Real-Time Analysis

Rapid analysis to match the nanopore sequencer throughput

Timely decisions to stop sequencing as early as possible

Accurate analysis from noisy raw signal data

Power-efficient computation for scalability and portability

Outline

Background

RawHash

RawHash2

Evaluation

Conclusion

Executive Summary

Problem: Real-time analysis of nanopore raw signals is **inaccurate** and **inefficient for large genomes**

Goal: Enable **fast** and **accurate** real-time analysis of raw nanopore signals

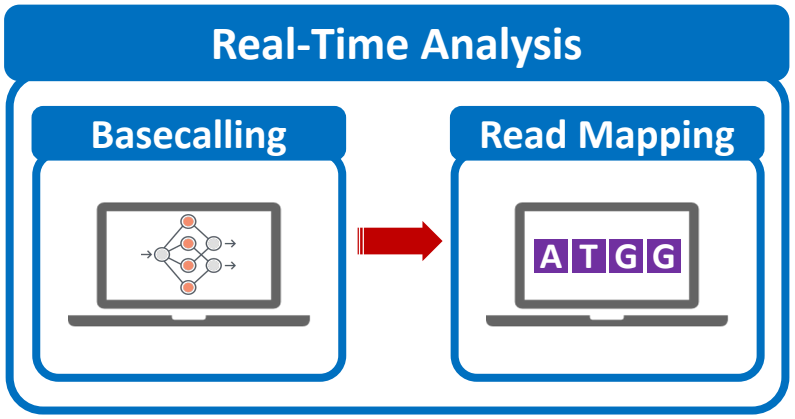
Key Contributions:
1) The **first hash-based mechanism** for mapping raw nanopore signals

2) The novel **Sequence Until** technique can accurately and **dynamically stop the entire sequencing of all reads at once** if further sequencing is not necessary

Key Results: Across 3 use cases and 5 genomes of varying sizes
- **27× 19×, and 4× better average throughput** compared to the state-of-the-art works
- **Most accurate raw signal mapper for all datasets**
- Sequence Until **reduces the sequencing time and cost by 15×**

Existing Solutions

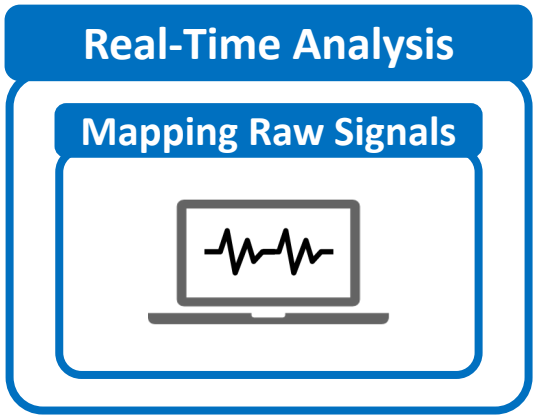
1. Deep neural networks (**DNNs**) for translating **signals** to **bases**



Less noisy analysis from basecalled sequences

Costly and power-hungry computational requirements

2. Mapping **signals** to reference genomes **without** basecalling

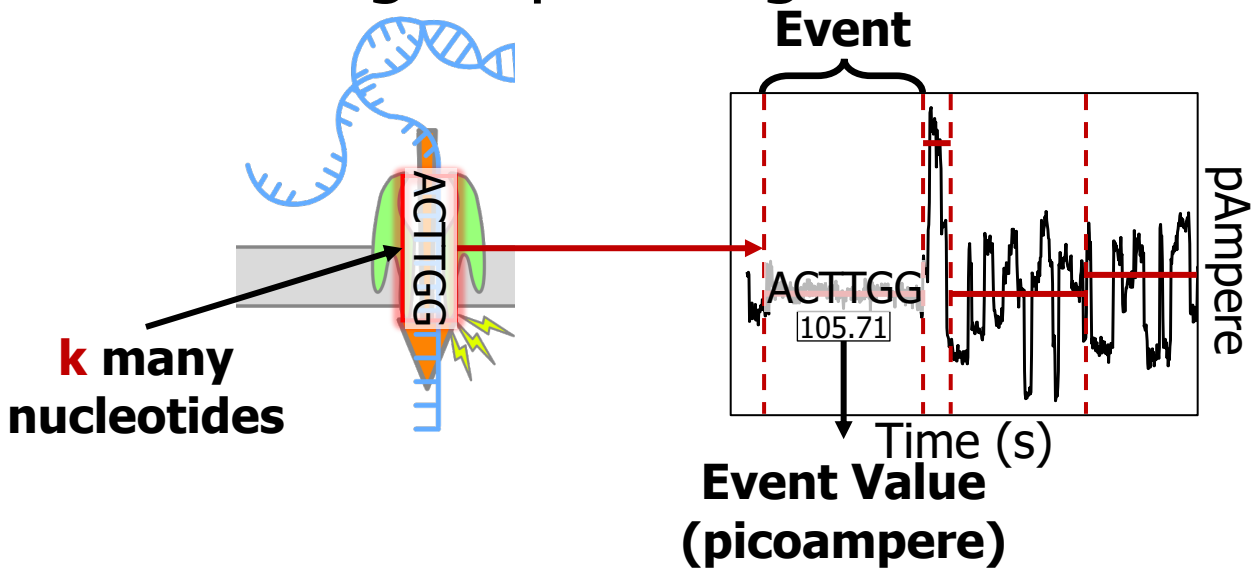


Raw signals contain richer information than bases

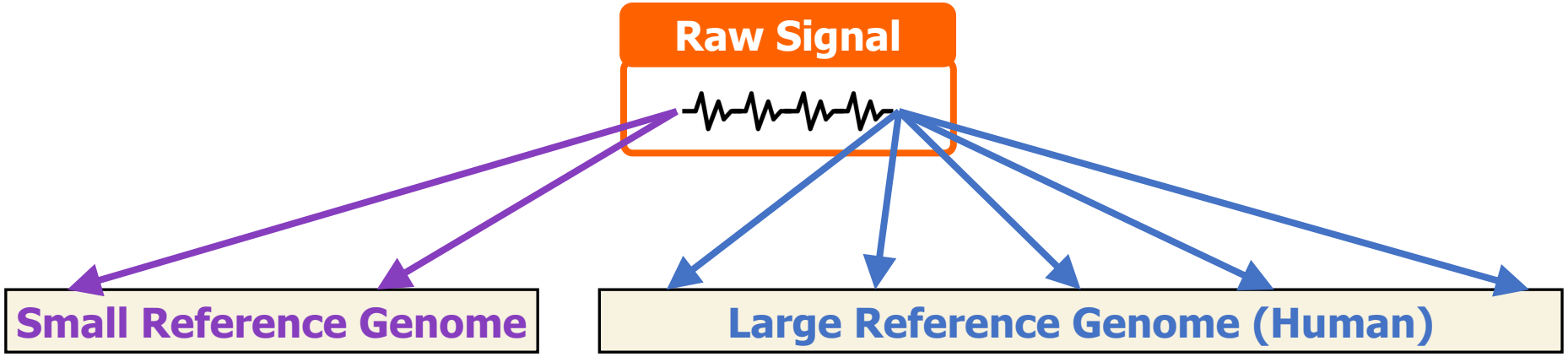
Efficient analysis with better scalability and portability

Enabling Analysis From Electrical Signals

- **K** many nucleotides (**k**-mers) sequenced at a time
- **Event:** A **segment** of the raw signal
  - Corresponds to a **particular k**-mer
  - Abrupt signal changes show sequencing of a new k-mer
  - **Statistical methods** can find these abrupt changes
  - **Event value:** average of signals **within an event**
- **Observation: Identical** k-mers generate **similar** event values during sequencing



The Problem – Mapping Raw Signals



Fewer candidate regions in **small genomes**

Substantially **larger number of regions** to check **per read** as the genome size increases

Accurate mapping

Problem: Probabilistic mechanisms on **many regions → inaccurate mapping**

High throughput

Problem: Distance calculation on **many regions → reduced throughput**

The Problem – Mapping Raw Signals

Existing solutions are
inaccurate or inefficient
for large genomes

Accurate mapping

on **many regions → inaccurate mapping**

High throughput

Problem: Distance calculation
on **many regions → reduced throughput**

Goal

Enable **fast and accurate real-time analysis** of raw nanopore signals **for large genomes**

RawHash

The **first hash-based search mechanism**
to quickly and accurately map raw nanopore signals
to reference genomes

Sequence Until can accurately and **dynamically stop**
the entire sequencing run at once
if further sequencing is unnecessary

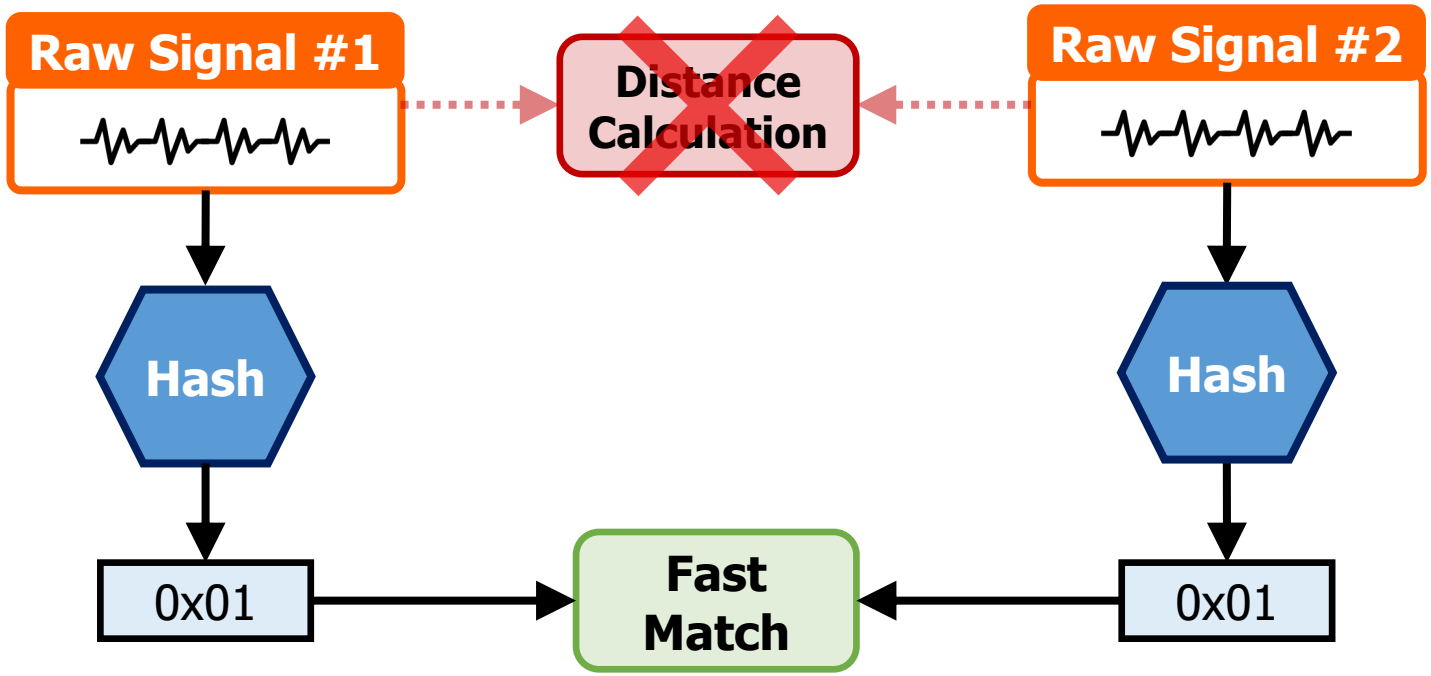
RawHash

The **first hash-based search mechanism**
to quickly and accurately map raw nanopore signals
to reference genomes

Sequence Until can accurately and **dynamically stop**
the entire sequencing run at once
if further sequencing is unnecessary

RawHash – Key Idea

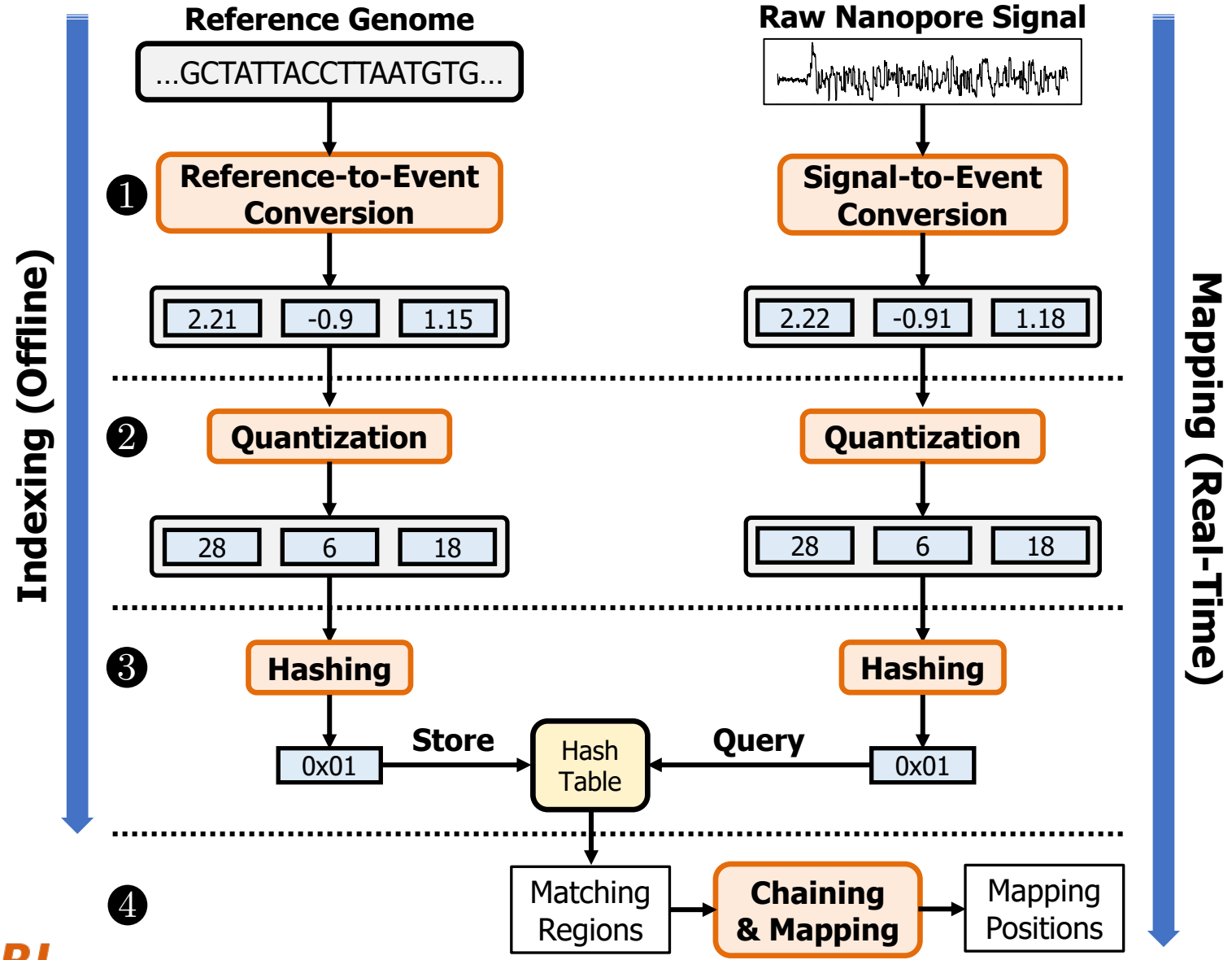
Key Observation: **Identical** nucleotides generate **similar** raw signals



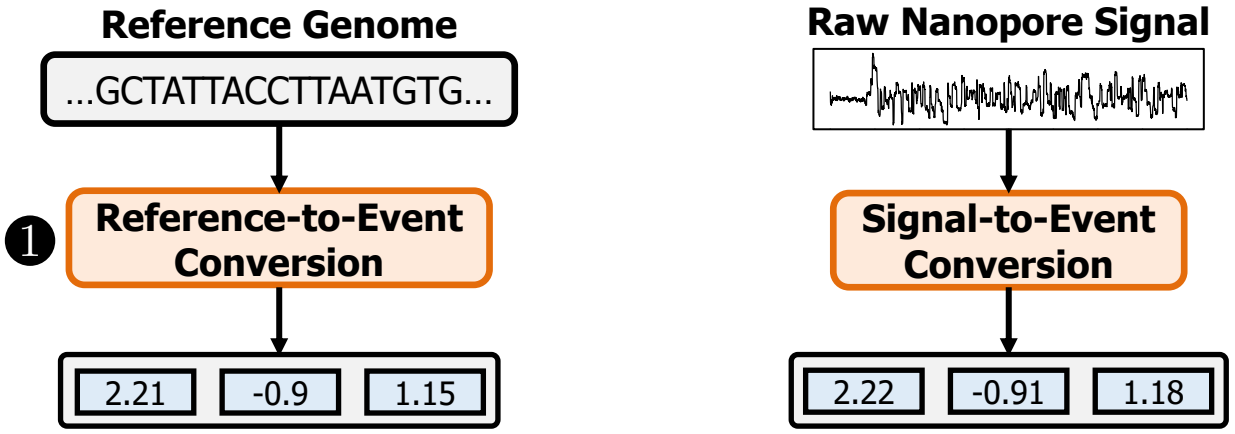
Challenge #1: Generating the **same** hash value for **similar enough** signals

Challenge #2: **Accurately** finding as **few** similar regions as possible

RawHash Overview



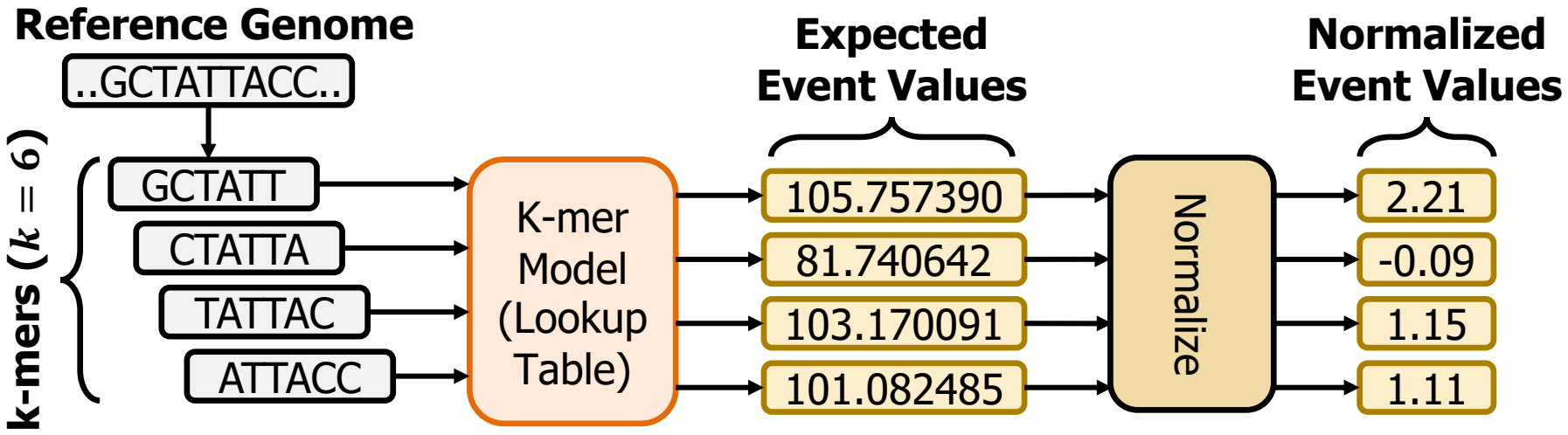
RawHash Overview



Reference-to-Event Conversion

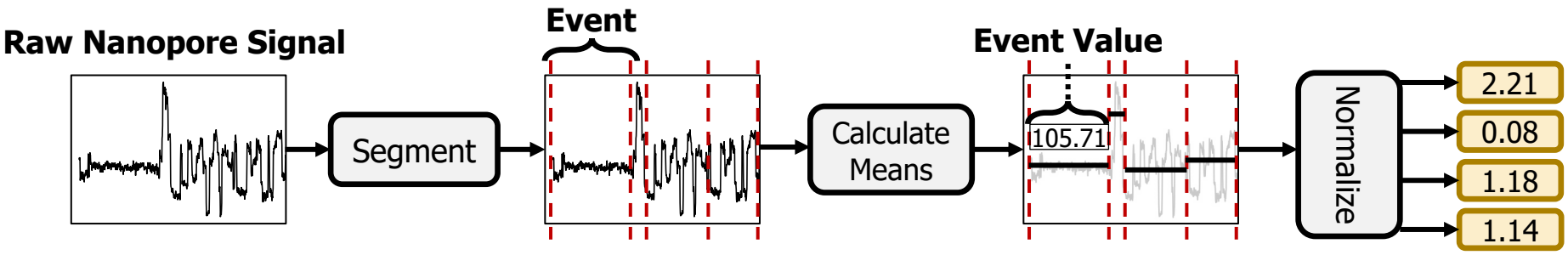
- **K-mer model:** Provides **expected** event values **for each k-mer**
  - Preconstructed based on nanopore sequencer characteristics

- Use the **k-mer model** to convert **all k-mers** of a reference genome to their **expected** event values



Signal-to-Event Conversion

- **Event detection:** Identifies signal regions corresponding to specific k-mers
  - Uses statistical test (**segmentation**) to spot abrupt signal changes



- Consecutive events ➔ consecutive k-mers

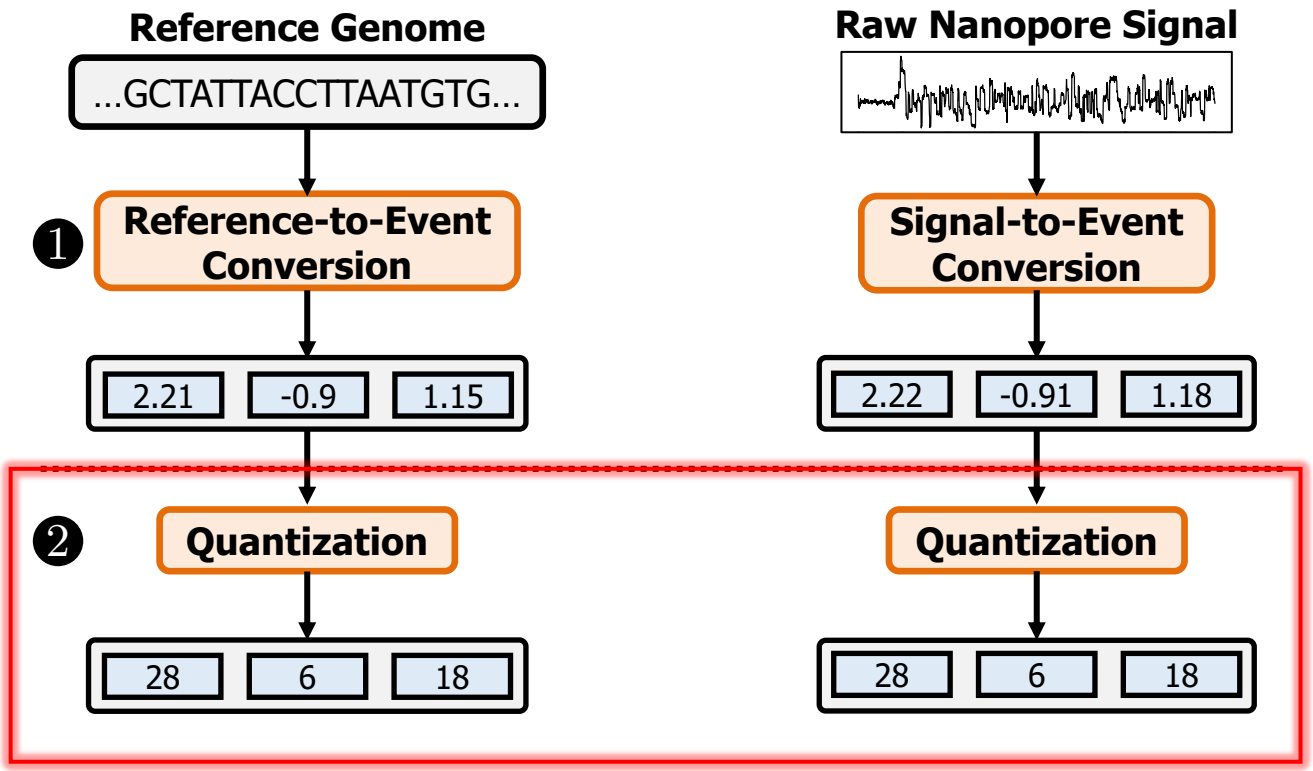
Signal-to-Event Conversion

- **Event detection:** Identifies signal regions corresponding to specific k-mers
  - Uses statistical test (**segmentation**) to spot abrupt signal changes

Can we directly match signals to each other?

- Consecutive events ➔ consecutive k-mers

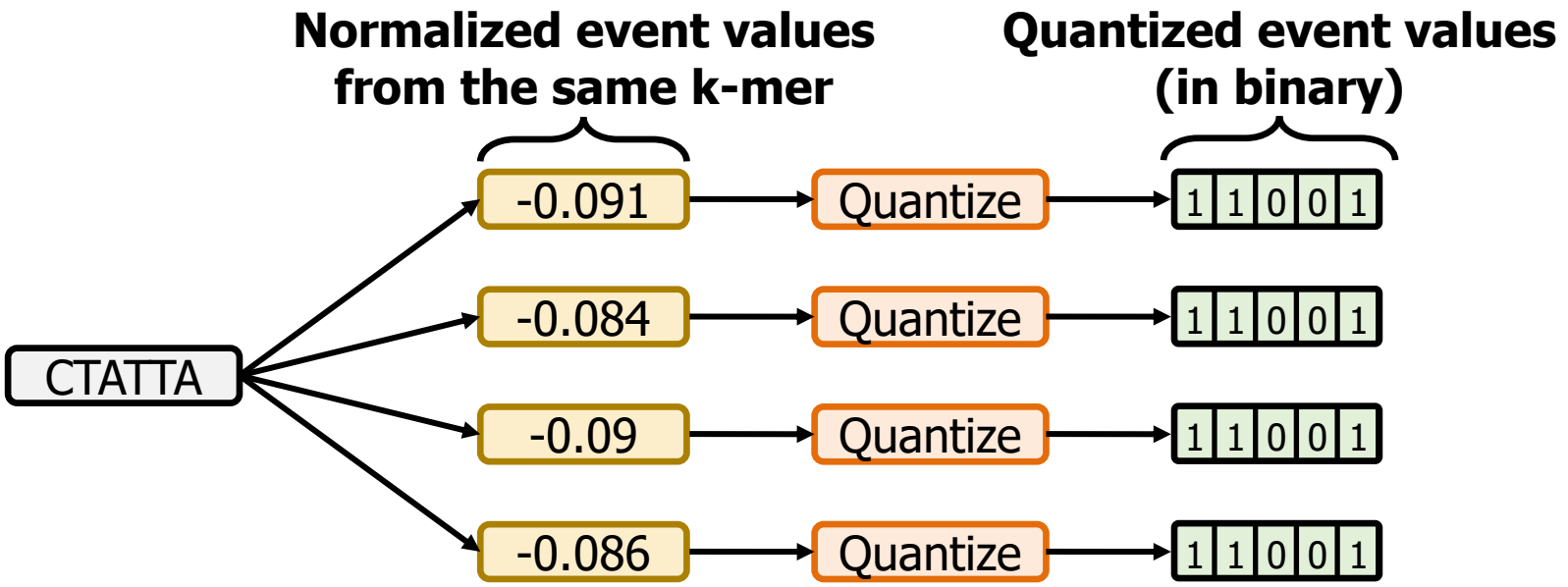
RawHash Overview



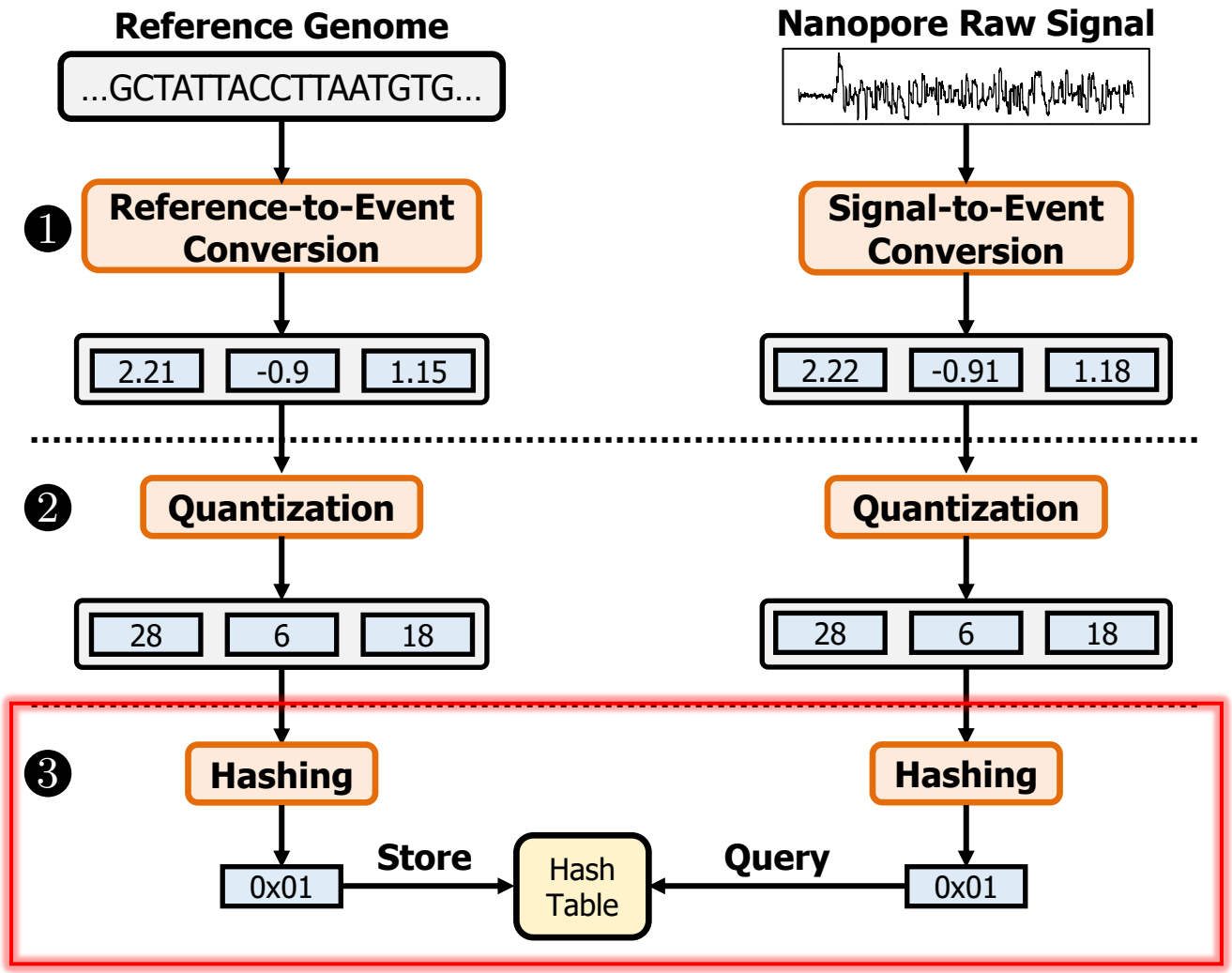
Quantizing the Event Values

- **Observation:** Slight differences in raw signals from identical k-mers
  - **Challenge:** Direct event value matching is not feasible and accurate

- **Key Idea:** Quantize the event values
  - Enables assigning **identical quantized values** to **similar event values**



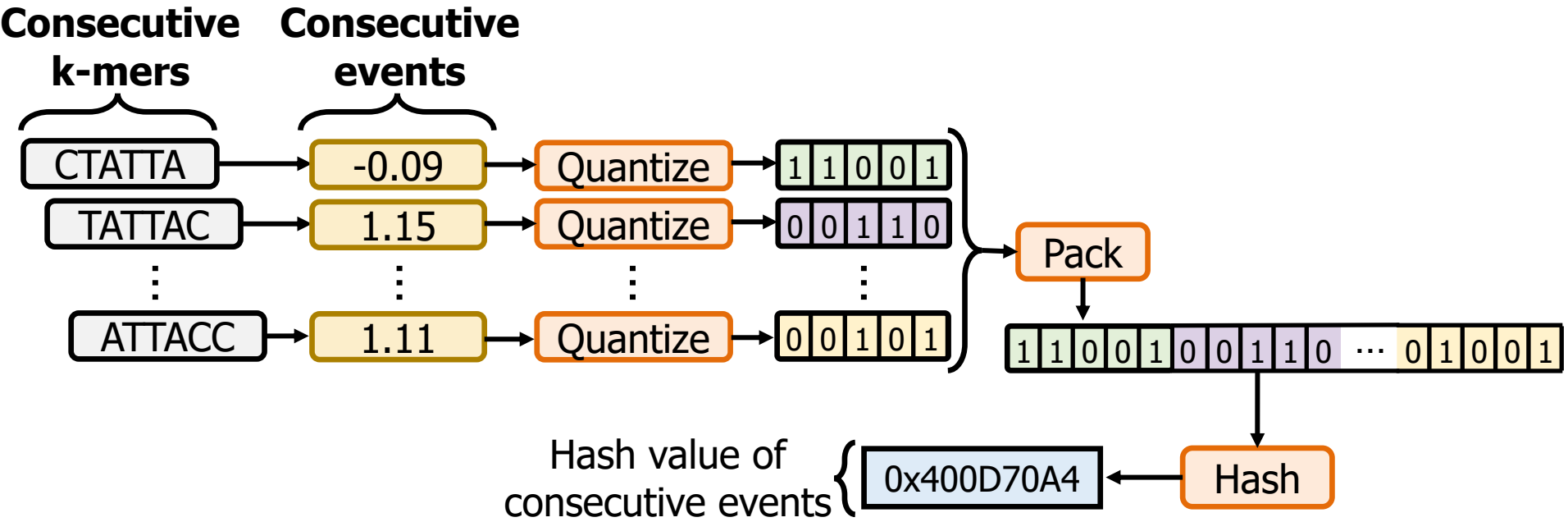
RawHash Overview



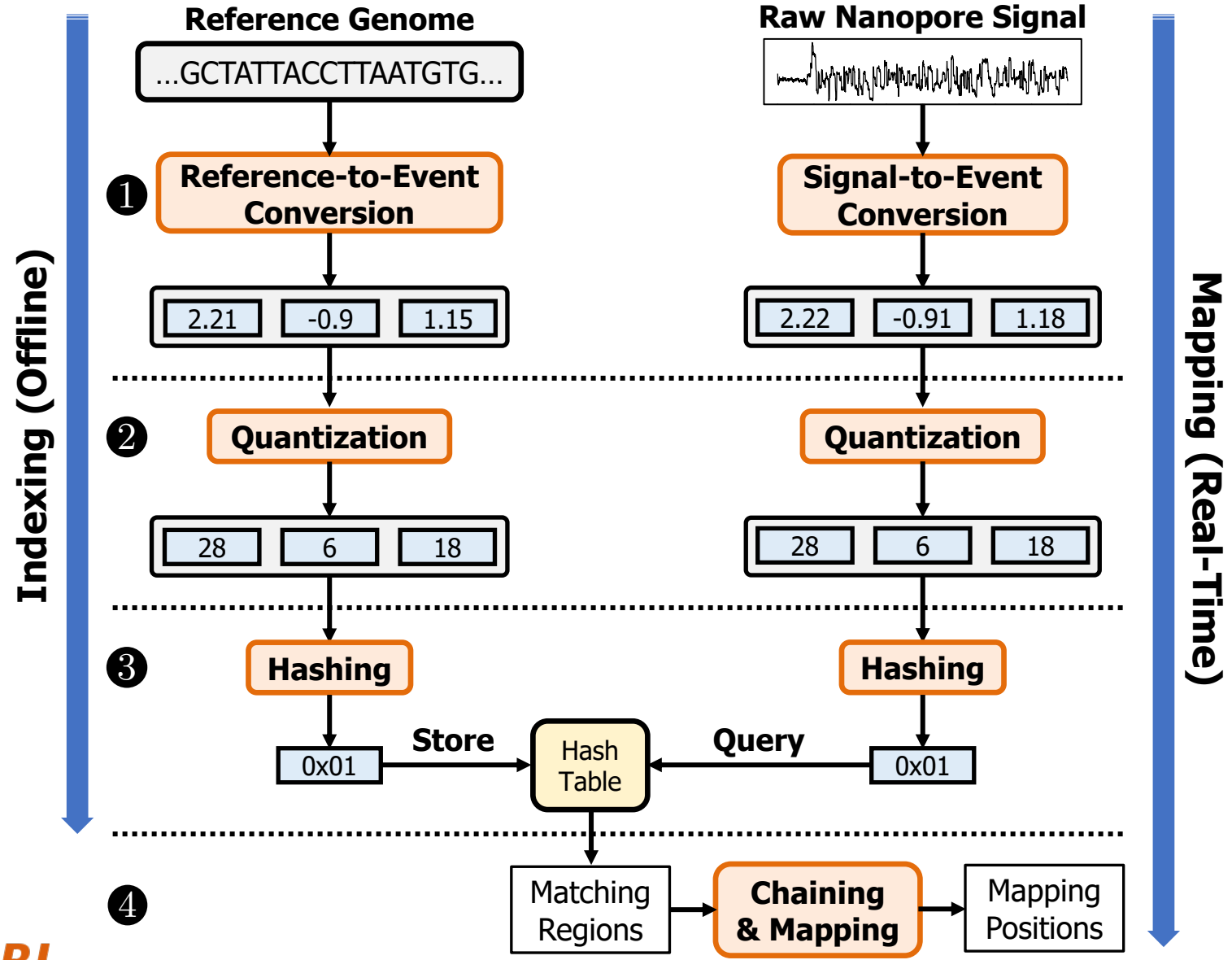
Hashing for Fast Similarity Search

- Each event usually represents a very small k-mer (6 to 9 characters)
  - **Challenge:** Short k-mers are likely to appear in many locations

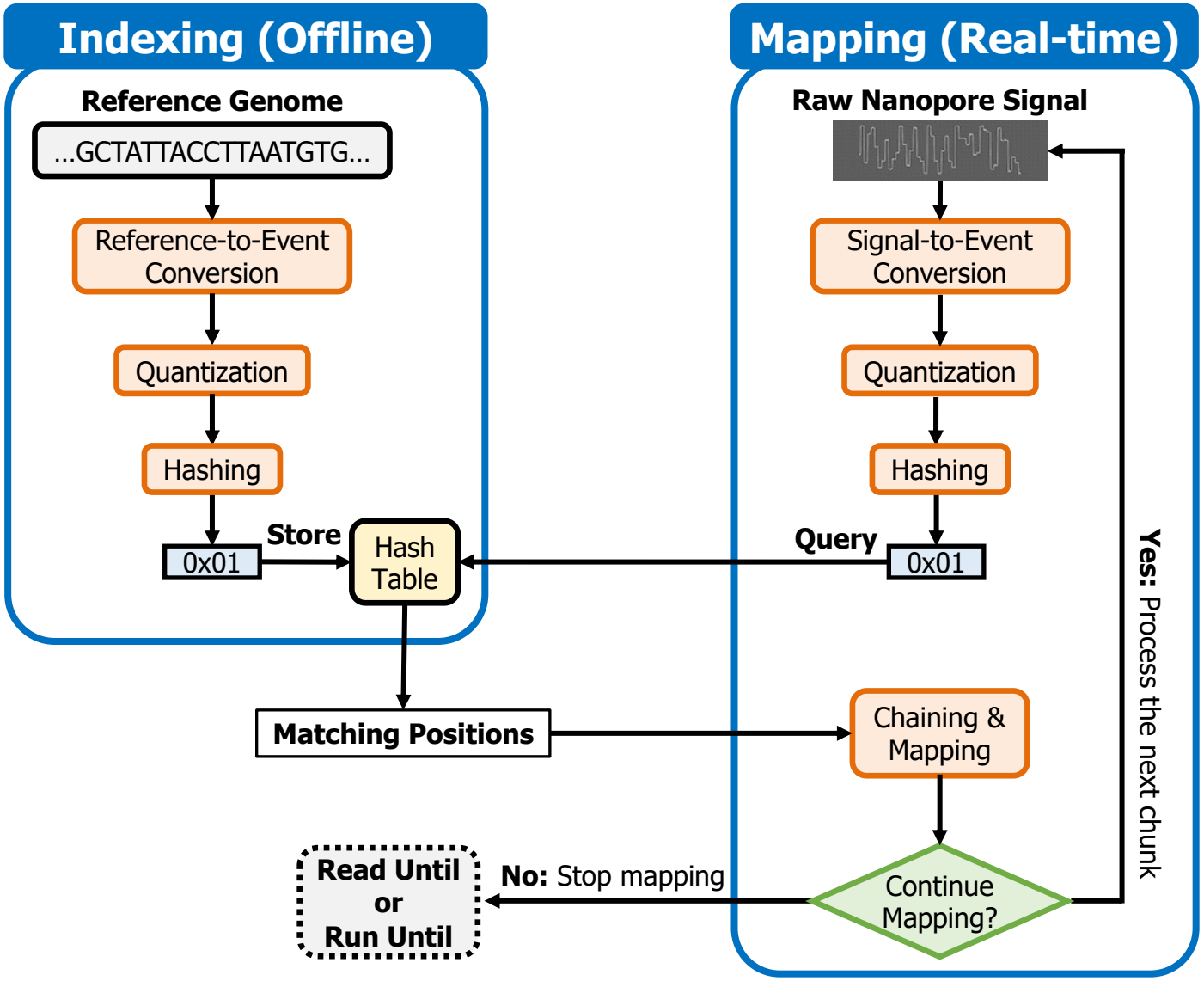
- **Key Idea:** Create longer k-mers from many **consecutive events**
- **Key Benefit:** Directly match hash values to quickly identify similarities



RawHash Overview



Real-Time Mapping using Hash-based Indexing



RawHash

The **first hash-based search mechanism**
to quickly and accurately map raw nanopore signals
to reference genomes

Sequence Until can accurately and **dynamically stop**
the entire sequencing run at once
if further sequencing is unnecessary

RawHash

The **first hash-based search mechanism**
to quickly and accurately map raw nanopore signals
to reference genomes

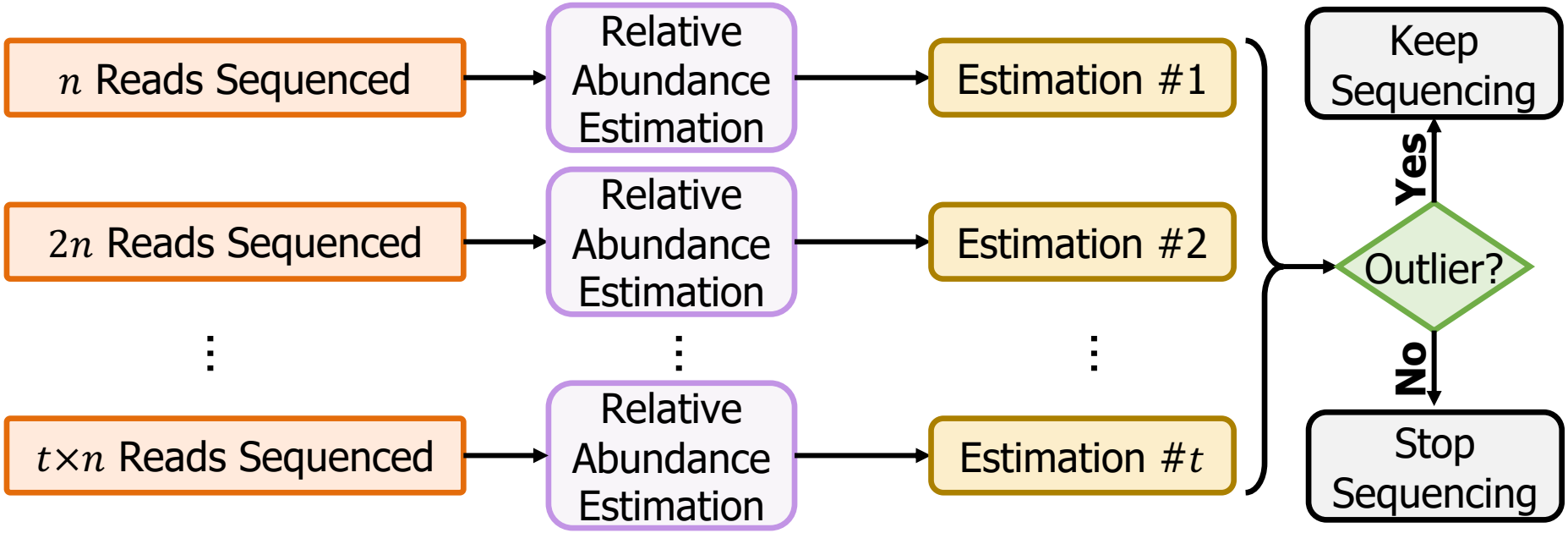
Sequence Until can accurately and **dynamically stop**
the entire sequencing run at once
if further sequencing is unnecessary

The Sequence Until Mechanism

- **Problem:**
  - Unnecessary sequencing waste time, power and money
- **Key Idea:**
  - **Dynamically** decide if further sequencing of the entire sample is necessary to achieve high accuracy
  - Stop sequencing early without sacrificing accuracy
- **Potential Benefits:**
  - Significant **reduction in sequencing time and cost**
- Example real-time genome analysis use case:
  - **Relative abundance estimation**

The Sequence Until Mechanism

- **Key Steps:**
  1. Continuously generate relative abundance estimation after every n reads
  2. Keep the last t estimation results
  3. **Detect outliers** in the results via **cross-correlation** of the recent t results
  4. Absence of outliers indicates **consistent results**
     - Further sequencing **is likely** to generate consistent results ➔ Stop the sequencing



Outline

Background

RawHash

RawHash2

Evaluation

Conclusion

Key Contributions in RawHash2

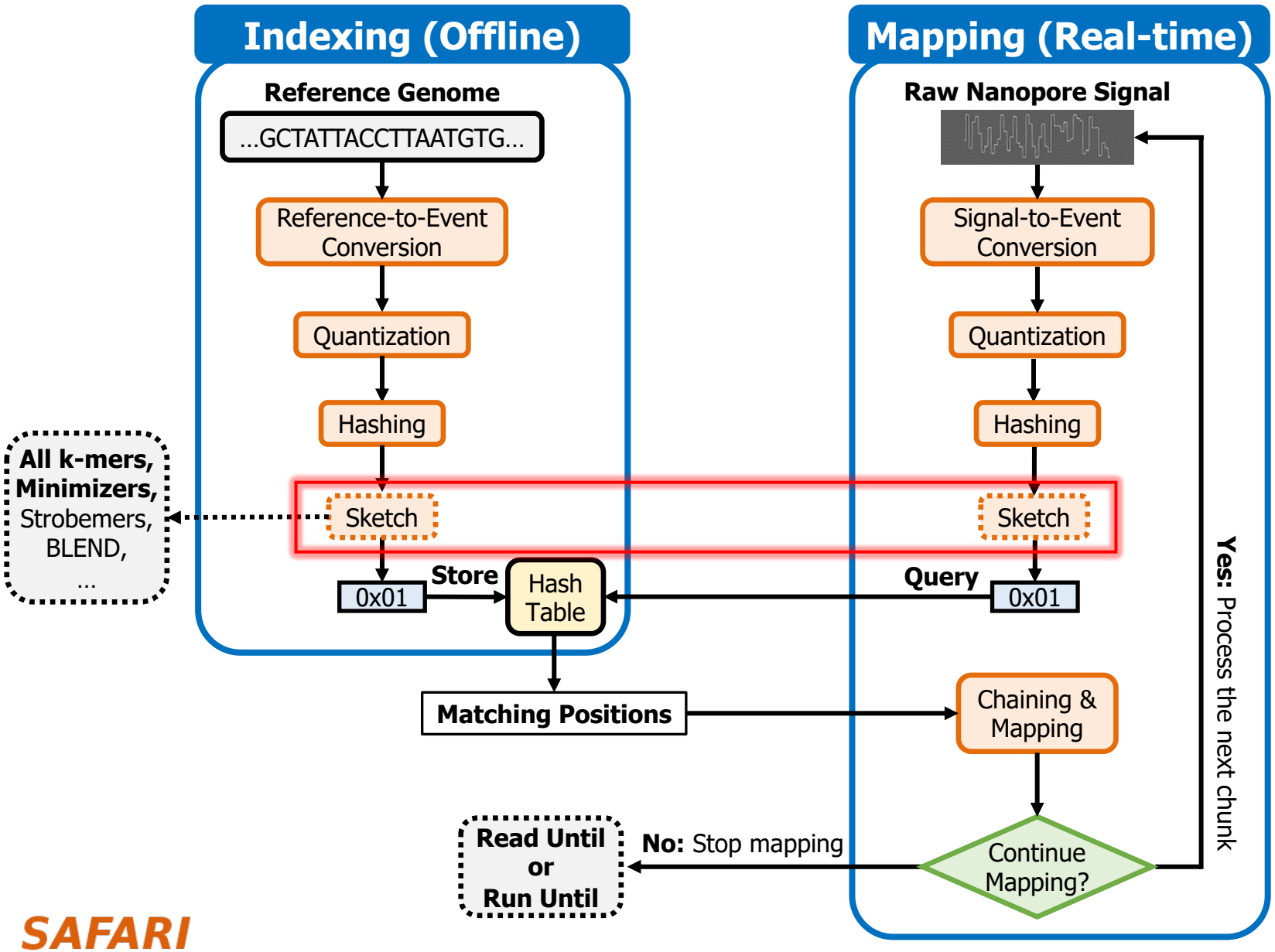
A new adaptive quantization to better fit the expected nanopore signal pattern to achieve **high accuracy**

Improved chaining algorithm with sensitive penalty scores

Weighted decision making for more robust mapping

Frequency filter and **minimizer sketching** to reduce seed matches for faster and space-efficient mapping

Sketching with Hash-based Indexing



Outline

Background

RawHash

RawHash2

Evaluation

Conclusion

Evaluation Methodology

- Two settings for RawHash2:
  - **RawHash2:** All hash values without sampling
  - **RawHash2-Minimizer:** Minimizer sketching

- Compared to **UNCALLED** [Kovaka+, Nat. Biotech. '21], **Sigmap** [Zhang+, ISMB '21] and **RawHash** [Firtina+, ISMB '23]

- **Use cases** for real-time genome analysis:
  1. Read mapping
  2. Relative abundance estimation
  3. Contamination analysis

Evaluation Methodology

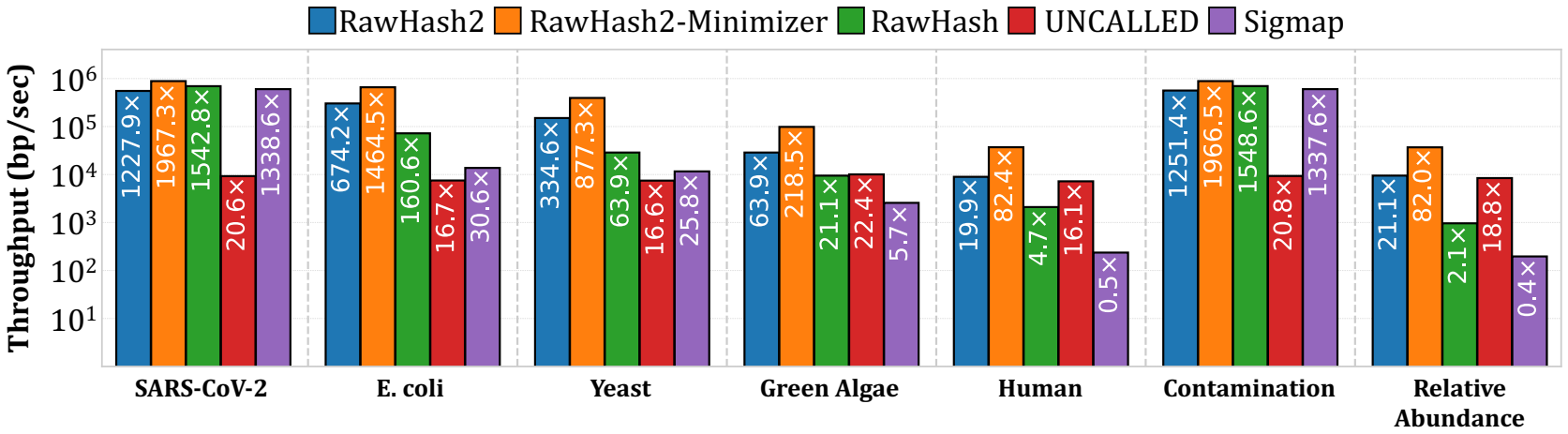
- Evaluation metrics:
  - **Throughput** (bases processed per second per CPU thread)
  - Potential reduction in **sequencing time and cost**
  - **Accuracy**
    - **Baseline:** Mapping basecalled reads using minimap2
    - Precision, recall, and F1 scores
    - Relative abundance estimation distance to ground truth

- **Datasets:**

	Organism	Reads (#)	Bases (#)	Genome Size
Read Mapping				
D1	*SARS-CoV-2*	1,382,016	594M	29,903
D2	*E. coli*	353,317	2,365M	5M
D3	*Yeast*	49,989	380M	12M
D4	*Green Algae*	29,933	609M	111M
D5	*Human HG001*	269,507	1,584M	3,117M
Relative Abundance Estimation				
	D1-D5	2,084,762	5,531M	3,246M
Contamination Analysis				
	D1 and D5	1,651,523	2,178M	29,903

Throughput

- **Real-time analysis requires** faster throughput than sequencer
  - Throughput from a single pore: **~450 bp/sec (data generation speed)**

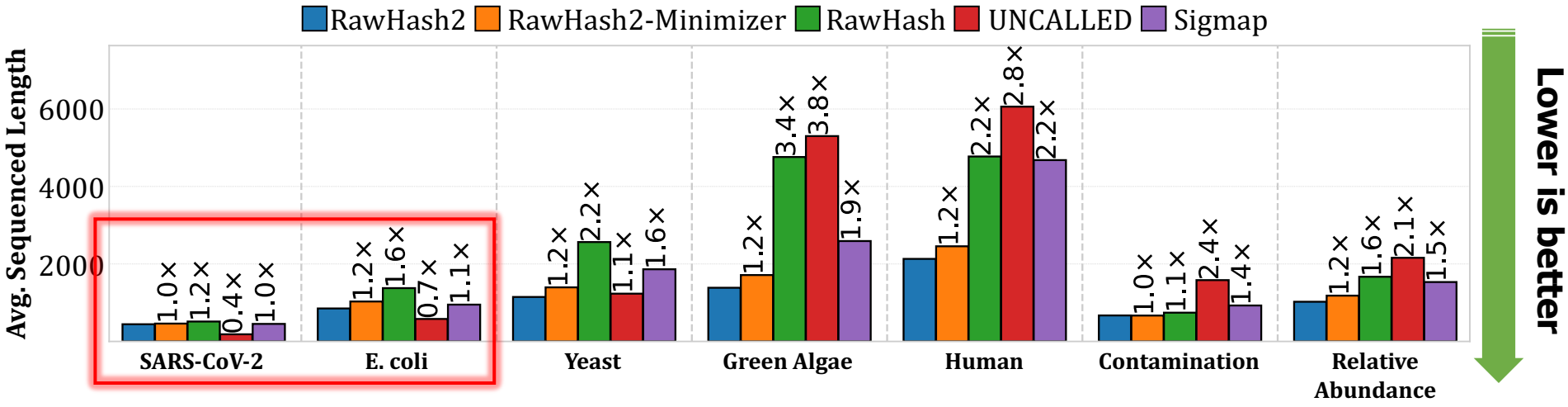


RawHash2: 27×, 19×, and 4× better average throughput compared to **UNCALLED, Sigmap** and **RawHash**, respectively

RawHash2-Minimizer further improves the throughput by **2.5×** compared to **RawHash2**

Average Sequenced Length

- Fewer bases to sequence ➔ Less unnecessary sequencing



RawHash2 **reduces sequencing time and cost**

on average by **1.9**× compared to UNCALLED and RawHash

RawHash2 leads to **sequencing the least amount of bases**

for larger genomes

Accuracy

- Read mapping, contamination, and relative abundance estimation accuracy (baseline: basecalled mapping)

Dataset	Metric	RH2	RH2-Min.	RH	UNCALLED	Sigmap
SARS-CoV-2	F1	**0.9867**	0.9691	0.9252	0.9725	0.7112
E. coli	F1	**0.9748**	0.9631	0.9280	0.9731	0.9670
Yeast	F1	**0.9602**	0.9472	0.9060	0.9407	0.9469
Green Algae	F1	**0.9351**	0.9191	0.8114	0.8277	0.9350
Human	F1	**0.7599**	0.6699	0.5574	0.3197	0.3269
Contamination	Precision	**0.9595**	0.9424	0.8702	0.9378	0.7856
Rel. Abundance	Distance	**0.2678**	0.4243	0.4385	0.6812	0.5430

Best results are **highlighted**.

RawHash2 provides the **most accurate read mapping**

RawHash2-Minimizer provides an **on-par accuracy with RawHash2** while improving **the throughput substantially**

Benefits of Sequence Until

- Running RawHash with and without Sequence Until

	Estimated Relative Abundance Ratios in 50,000 Random Reads					
Tool	***SARS-CoV-2***	***E. coli***	***Yeast***	***Green Algae***	***Human***	**Distance**
RawHash (100%)	0.0270	0.3636	0.3062	0.1951	0.1081	N/A
RawHash + Sequence Until (7%)	0.0283	0.3539	0.3100	0.1946	0.1133	0.0118

Sequence Until enables sequencing **only 7% (~1/15)** of the entire sample **with high accuracy**

UNCALLED and RawHash benefit from **Sequence Until** significantly **by up to 100**× reductions in sequencing

Outline

Background

RawHash

RawHash2

Evaluation

Conclusion

Conclusion

Key Contributions:
1) The **first hash-based mechanism** for mapping raw nanopore signals
2) The novel **Sequence Until** technique can accurately and **dynamically stop the entire sequencing of all reads at once** if further sequencing is not necessary

Key Results: Across 3 use cases and 5 genomes of varying sizes
- **27× 19×, and 4× better average throughput** compared to the state-of-the-art works
- **Most accurate raw signal mapper for all datasets**
- Sequence Until **reduces the sequencing time and cost by 15×**

Many opportunities for analyzing raw nanopore signals in real-time:
- Many hash-based **sketching techniques** can now be used for raw signals
- **Indexing is very cheap:** Many future use cases with the on-the-fly index construction
- We should rethink the algorithms to perform downstream analysis fully using raw signals

ISMB/ECCB 2023

RawHash

Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes

Can Firtina

Nika Mansouri Ghiasi Joel Lindegger Gagandeep Singh

Meryem Banu Cavlak Haiyu Mao Onur Mutlu

RawHash RawHash2 Code

SAFARI ETH zürich

Agenda for Today

- Cutting-edge in Accelerating Genome Analysis
- Enabling Fast and Accurate Real-time Analysis
  - RawHash and RawHash2
- Conclusion

Future is Bright for Raw Signal Analysis

Overlapping Raw Signals in Real-Time

Rawsamble: Overlapping and Assembling Raw Nanopore Signals using a Hash-based Seeding Mechanism

Can Firtina[1] Maximilian Mordig[1,2] Joël Lindegger[1] Harun Mustafa[1,3,4] Sayan Goswami[1]
Stefano Mercogliano[1] Yan Zhu[1,5] Andre Kahles[1,3,4] Onur Mutlu[1]
[1]*ETH Zurich* [2]*Max Planck Institute for Intelligent Systems* [3]*University Hospital Zurich*
[4]*Swiss Institute of Bioinformatics* [5]*University of Toronto*

Real-time *de novo* Assembly Construction

- All-vs-all overlapping using raw signals

	Organism	Shared Overlaps (%)	Unique to Rawsamble (%)	Unique to Minimap2 (%)
D1	*E. coli*	44.57	15.08	40.35
D2	*Yeast*	47.07	35.62	17.32
D3	*Human*	19.73	27.56	52.71

- Building *de novo* assemblies directly from raw signals

Organism	Tool	No. of contigs	Avg. Contig Length	Max. Contig Length
D1	Rawsamble	39	373,594	1,431,572
E. coli	minimap2	4	2,611,044	5,210,938
D2	Rawsamble	431	47,250	256,116
Yeast	minimap2	278	82,757	386,005
D3	Rawsamble	59	16,376	66,163
Human	minimap2	53	10,572	42,654

Opportunities for New Applications

- Improving the basecalling accuracy using the overlapping information between signals
- Full downstream analysis fully using raw nanopore signals
- Cooperating the raw signal analysis and basecalled sequence analysis together
- Many, many more keeping the hardware design in mind

Things Are Happening In Industry

Illumina DRAGEN Bio-IT Platform (2018)

- Processes whole genome at 34x coverage in ~30 minutes with hardware support for data compression

emea.illumina.com/products/by-type/informatics-products/dragen-bio-it-platform.html
emea.illumina.com/company/news-center/press-releases/2018/2349147.html
illumina.com/content/dam/illumina/gcs/assembled-assets/marketing-literature/dragen-bio-it-data-sheet-m-gl-00680/dragen-bio-it-data-sheet-m-gl-00680.pdf

Nova/NextSeq with Analysis Capability

Scale your studies with ease

Process high-throughput data quickly with hardware acceleration

Process high-throughput data quickly with hardware acceleration. With four field-programmable gate arrays (FPGAs) onboard you have the most powerful DRAGEN analysis ever, enabling you to process NovaSeq X System data easily. Perform up to four simultaneous applications per flow cell in a single run.

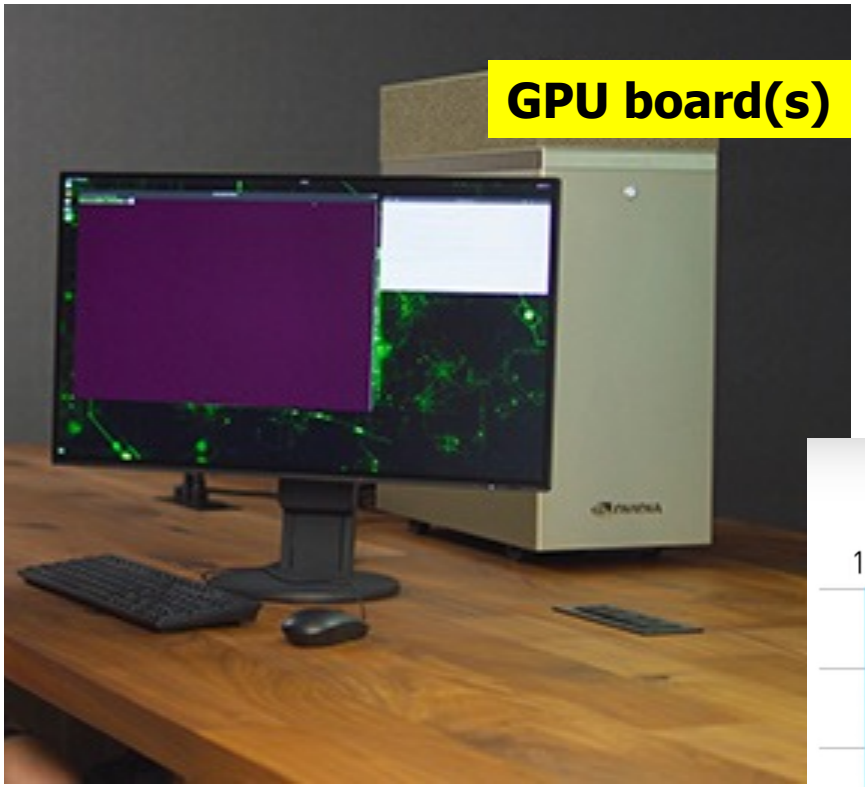
Reduce data footprint, manage and store data easily with lower costs and lower energy consumption, with built-in compression that reduces FASTQ file sizes by up to 80%.2

Stream data directly to Illumina Connected Analytics or BaseSpace Sequence Hub on the cloud for scalable data management, analysis, and aggregation.

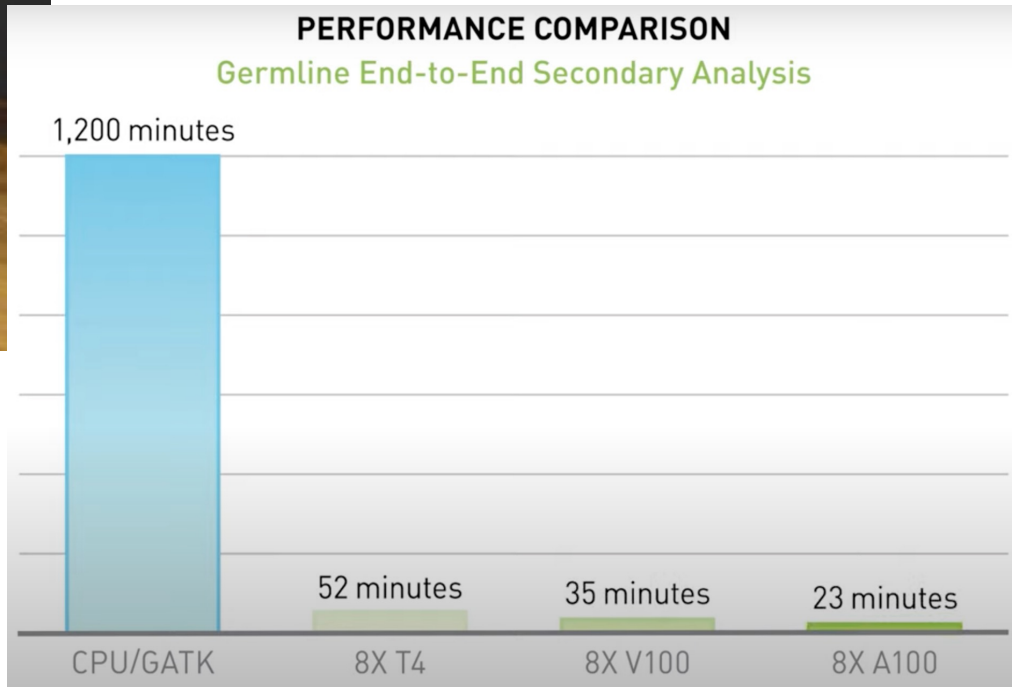
https://www.illumina.com/content/dam/illumina/gcs/assembled-assets/marketing-literature/dragen-bio-it-data-sheet-m-gl-00680/dragen-bio-it-data-sheet-m-gl-00680.pdf

https://emea.illumina.com/systems/sequencing-platforms/novaseq-x-plus/products-services/software.html

NVIDIA Clara Parabricks (2020)

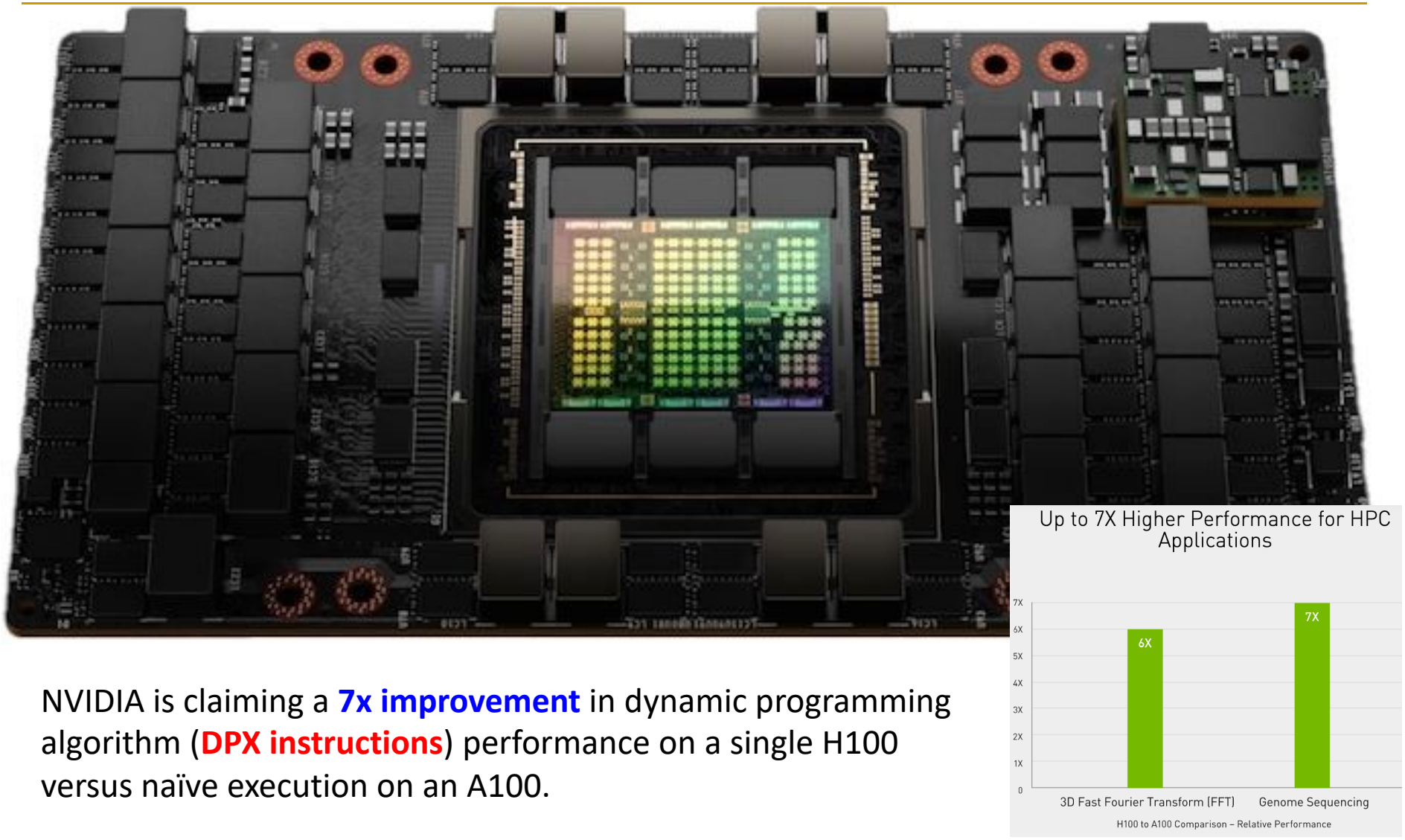


A University of Michigan startup in 2018 joined NVIDIA in 2020



https://developer.nvidia.com/clara-parabricks

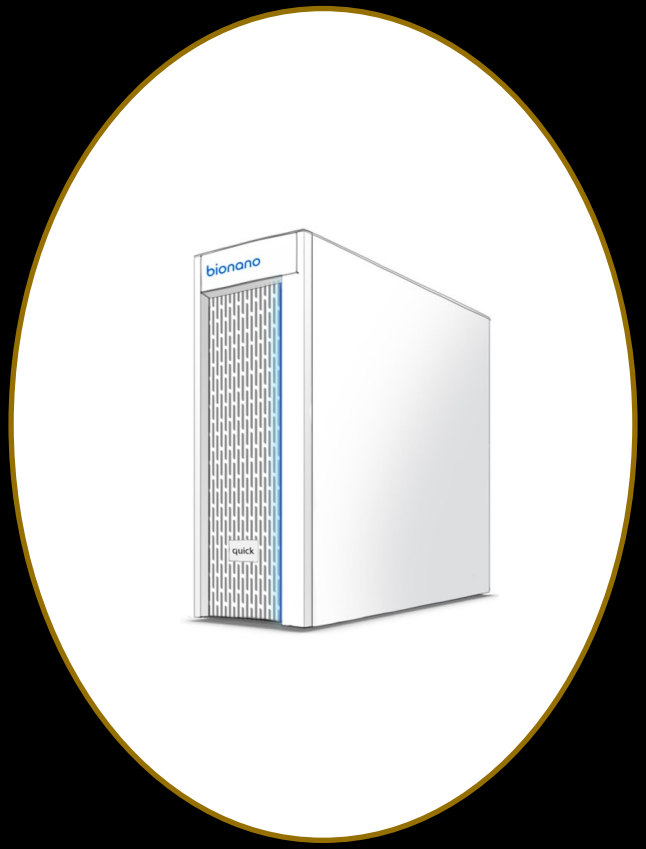
NVIDIA H100 (2022)



NVIDIA is claiming a **7x improvement** in dynamic programming algorithm (**DPX instructions**) performance on a single H100 versus naïve execution on an A100.

https://www.nvidia.com/en-us/data-center/h100/

- **We are accelerating the transformation in how we analyze the human genome!**



Bionano & NVIDIA:

Accelerating Analysis for Fast Time to Results

Technological solution to **support higher throughput**

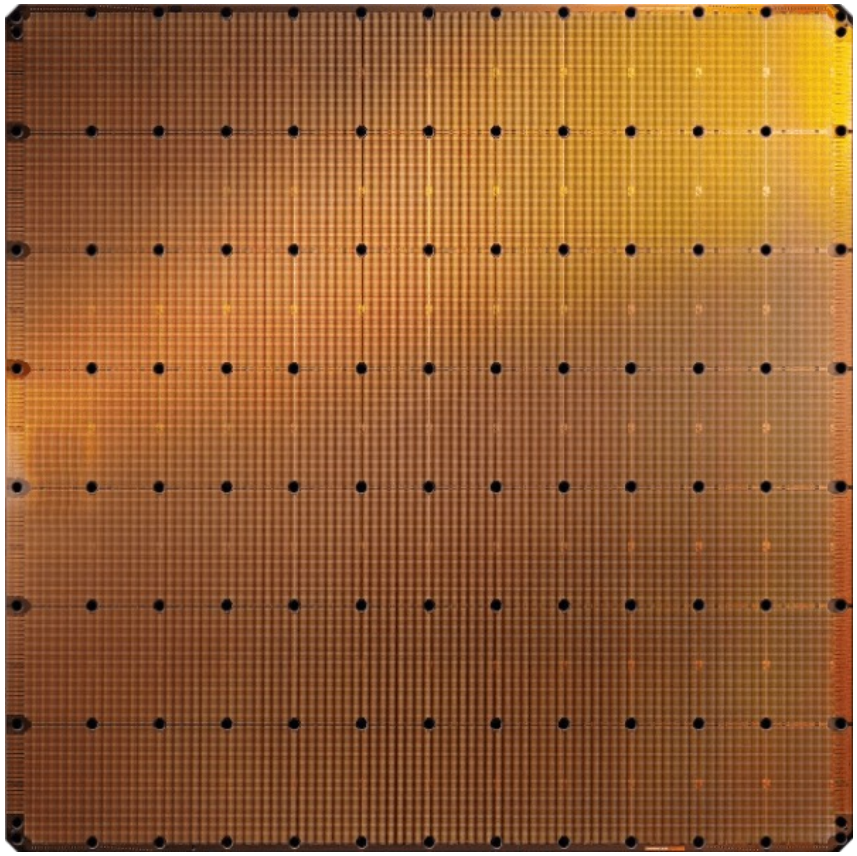
New high-performance algorithms from Bionano

Powered by NVIDIA RTX™ 6000 Ada Generation GPUs

Analysis of highly complex cancer whole genomes in **less than 2 hours**

Workflow tailored for a **small lab and IT footprint**

Cerebras's Wafer Scale Engine (2021)



- The largest ML accelerator chip (2021)
- 850,000 cores

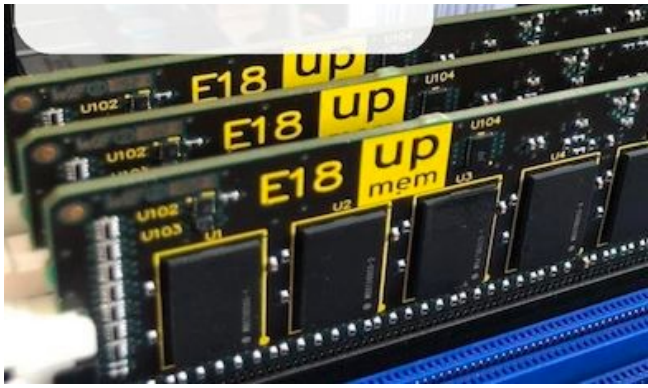
Cerebras WSE-2
2.6 Trillion transistors
46,225 mm^2

Largest GPU
54.2 Billion transistors
826 mm^2
NVIDIA Ampere GA100

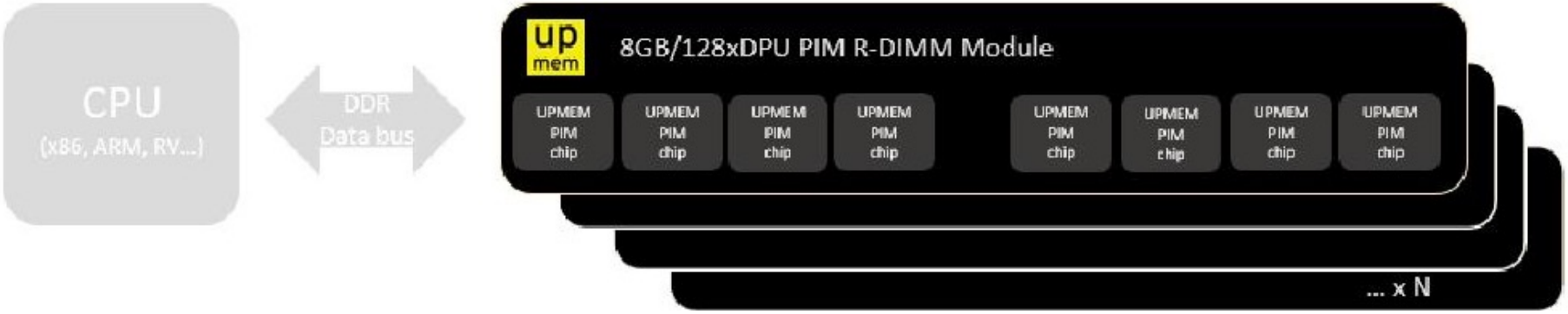
https://www.anandtech.com/show/14758/hot-chips-31-live-blogs-cerebras-wafer-scale-deep-learning
https://www.cerebras.net/cerebras-wafer-scale-engine-why-we-need-big-chips-for-deep-learning/

UPMEM Processing-in-DRAM Engine (2019)

- Processing in DRAM Engine
- Includes **standard DIMM modules**, with a **large number of DPU processors** combined with DRAM chips.



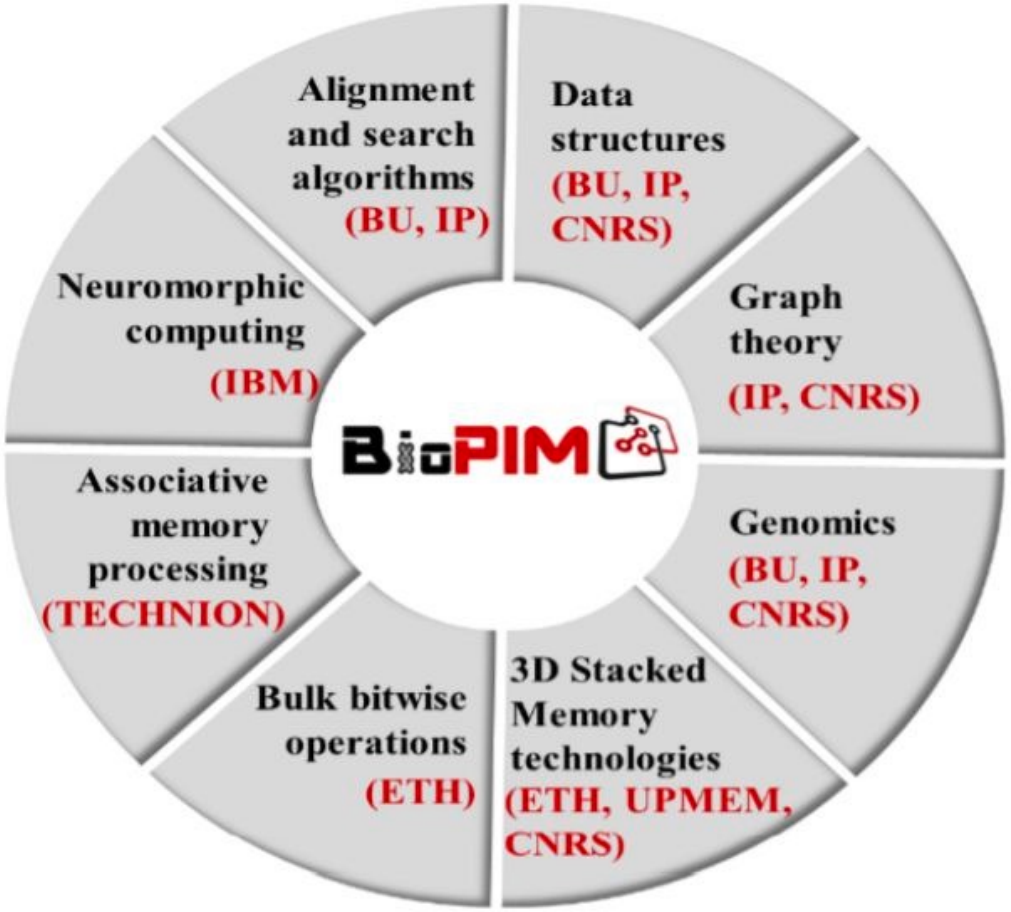
- Replaces **standard** DIMMs
  - DDR4 R-DIMM modules
    - 8GB+128 DPUs (16 PIM chips)
    - Standard 2x-nm DRAM process
  - **Large amounts of** compute & memory bandwidth



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

https://www.upmem.com/video-upmem-presenting-its-true-processing-in-memory-solution-hot-chips-2019/

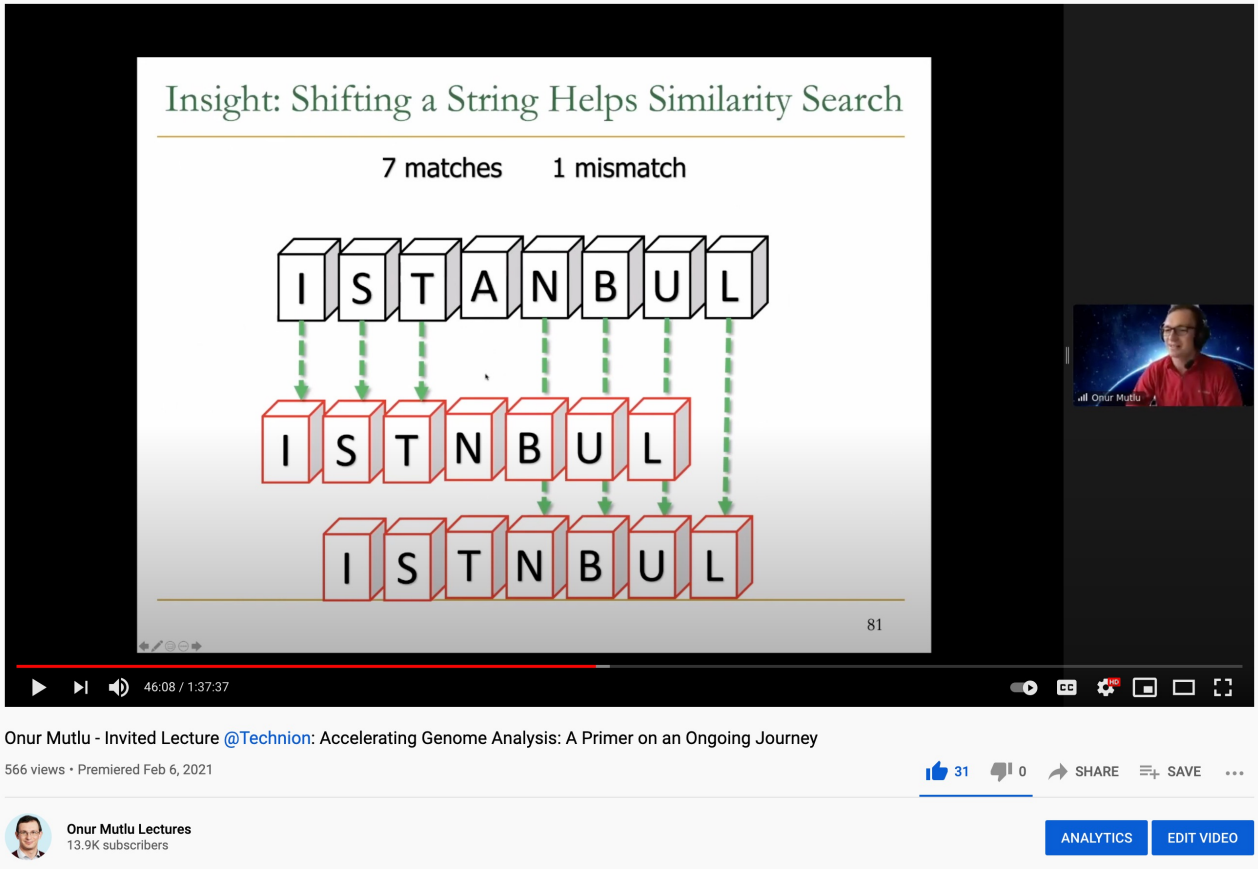
BioPIM (2022)



The vision of BioPIM is the realization of **cheap, ultra-fast and ultra-low energy mobile genomics** that eliminates the current dependence of sequence analysis on large and power-hungry computing clusters/data-centers.

Fast Genome Analysis…

- Onur Mutlu,
  "Accelerating Genome Analysis: A Primer on an Ongoing Journey"
  Invited Lecture at Technion, Virtual, 26 January 2021.
  [Slides (pptx) (pdf)]
  [Talk Video (1 hour 37 minutes, including Q&A)]
  [Related Invited Paper (at IEEE Micro, 2020)]



More on Fast Genome Analysis…

- Onur Mutlu,
  "Accelerating Genome Analysis"
  *Invited Talk at the Barcelona Supercomputing Center (**BSC**)*, Barcelona, Spain, 6 September 2022.
  [Slides (pptx) (pdf)]
  [Talk Video (1 hour 35 minutes, including Q&A)]
  [Related Invited Paper (at IEEE Micro, 2020)]
  [Related Invited Paper (at Computational and Structural Biology Journal, 2022)]

Accelerating Genome Analysis - Onur Mutlu's Invited Talk at the Barcelona Supercomputing Center

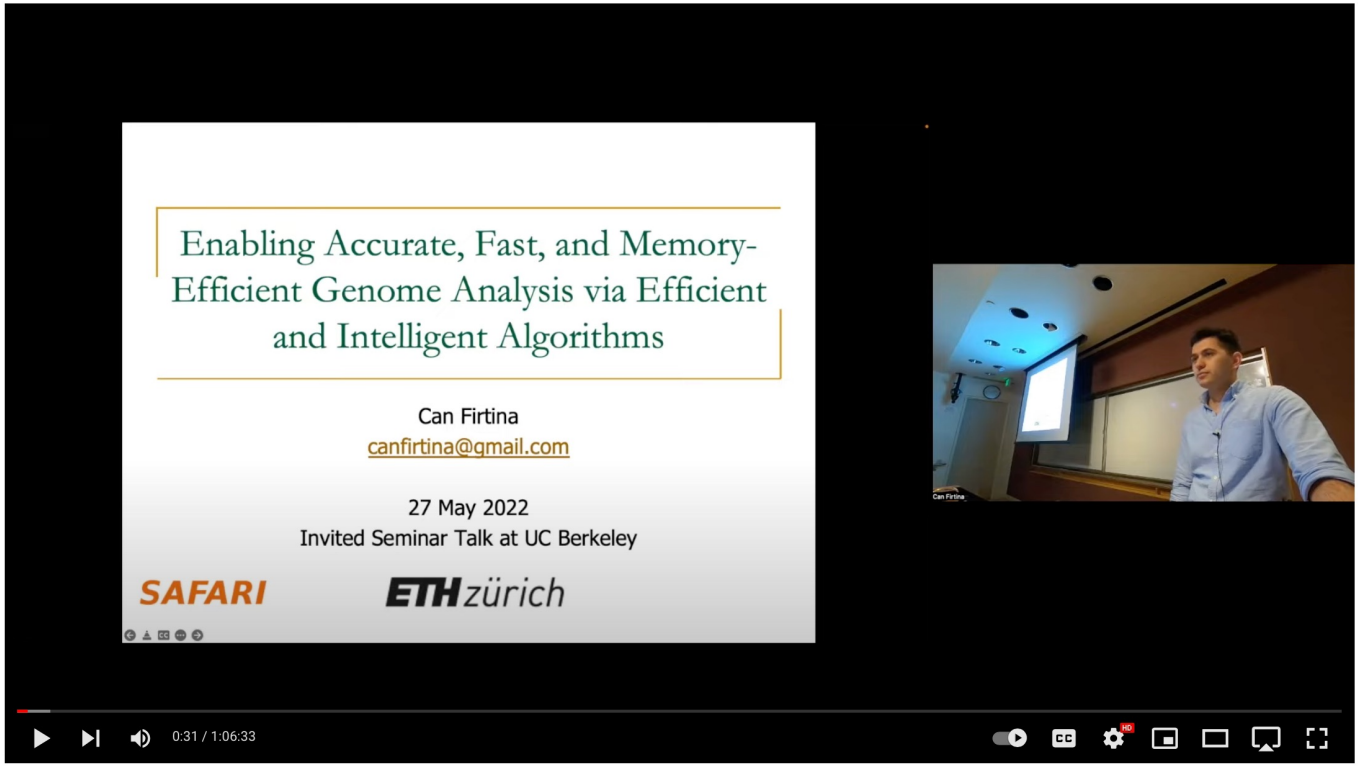
Onur Mutlu Lectures
36.6K subscribers
Analytics | Edit video | Share | Save

413 views Premiered Feb 2, 2023
Invited Talk at the Barcelona Supercomputing Center (BSC)
Accelerating Genome Analysis - A Primer on an Ongoing Journey

Presenter: Professor Onur Mutlu (https://people.inf.ethz.ch/omutlu/)
Date: September 6, 2022
1 hour 35 minutes (including Q&A)

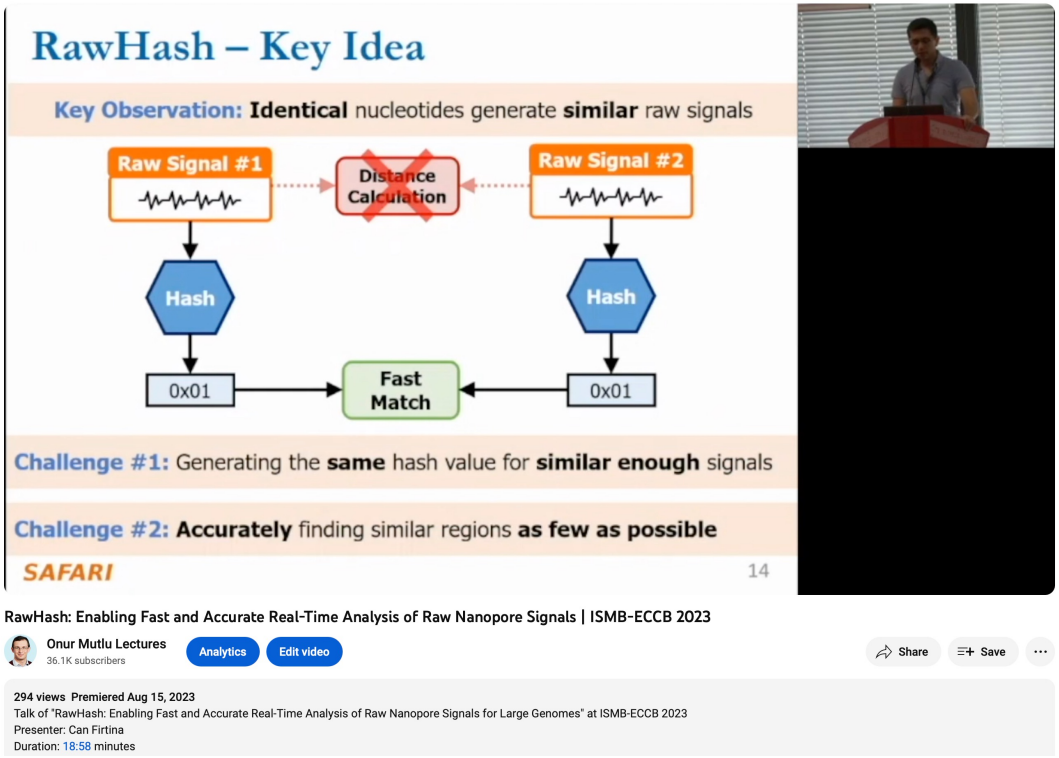
More on Accelerating Genome Analysis

- Can Firtina,
  "Enabling Accurate, Fast, and Memory-Efficient Genome Analysis via Efficient and Intelligent Algorithms"
  Talk at UC Berkeley, Berkeley, CA, United States, May 27, 2022.
  [Slides (pptx) (pdf)]
  [Talk Video (1 hour 6 minutes)]



More on Real-Time Genome Analysis

- Can Firtina,
  "RawHash: Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes"
  Proceedings Talk at ISMB-ECCB, Lyon, France, 25 July 2023.
  [Slides (pptx) (pdf)]
  [Talk Video (18 minutes]



Accelerating Genome Analysis [DAC 2023]

- Onur Mutlu and Can Firtina,
  "Accelerating Genome Analysis via Algorithm-Architecture Co-Design"
  *Invited Special Session Paper in Proceedings of the 60th Design Automation Conference (**DAC**)*, San Francisco, CA, USA, July 2023.
  [Slides (pptx) (pdf)]
  [Talk Video (38 minutes, including Q&A)]
  [Related Invited Paper]
  [arXiv version]

Accelerating Genome Analysis via Algorithm-Architecture Co-Design

Onur Mutlu Can Firtina
ETH Zürich

BIO-Arch Workshop at RECOMB 2023

- April 14, 2023

BIO-Arch: Workshop on Hardware Acceleration of Bioinformatics Workloads

About

BIO-Arch is a new forum for presenting and discussing new ideas in accelerating bioinformatics workloads with the co-design of hardware & software and the use of new computer architectures. Our goal is to discuss new system designs tailored for bioinformatics. BIO-Arch aims to bring together researchers in the bioinformatics, computational biology, and computer architecture communities to strengthen the progress in accelerating bioinformatics analysis (e.g., genome analysis) with efficient system designs that include hardware acceleration and software systems tailored fo new hardware technologies.

Venue

BIO-Arch will be held in The Social Facilities of İstanbul Technical University on **April 14**. Detailed information about how to arrive at the venue location with various transportation options can be found on the RECOMB website.

Our panel discussion will be held in conjunction with the main RECOMB conference. The panel discussion will be held in Marriott Şişli on **April 17 at 17:00**. You can find

https://www.youtube.com/watch?v=2rCsb4-nLmg

Genomics Course (Fall 2023)

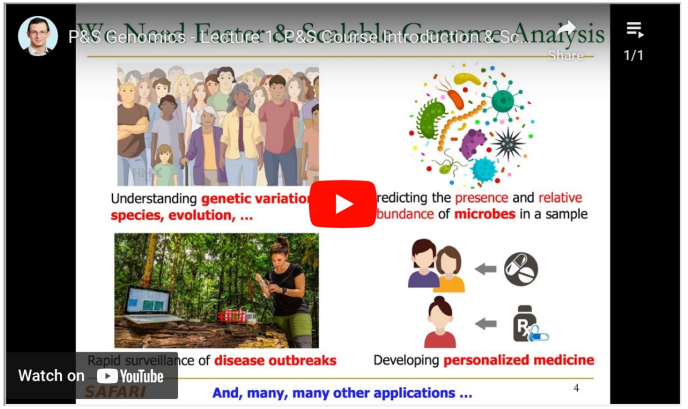
- **Fall 2023 Edition:**
  - https://safari.ethz.ch/projects_and_seminars/fall2023/doku.php?id=bioinformatics
- **Spring 2023 Edition:**
  - https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=bioinformatics

- **Youtube Livestream (Fall 2023):**
  - https://youtube.com/playlist?list=PL5Q2soXY2Zi_O0wyOjiMShG4t2QPZoeE3

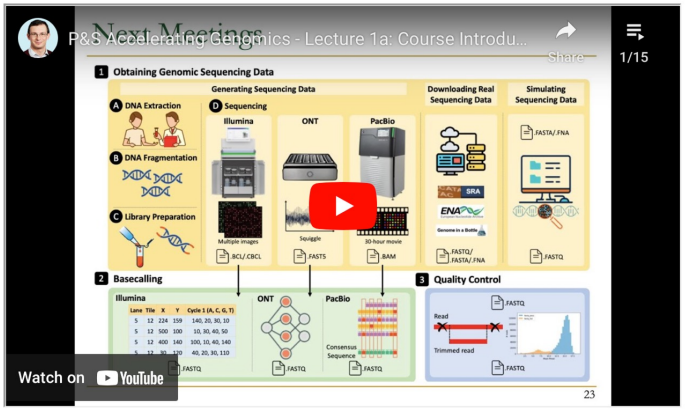
- Project course
  - Taken by Bachelor's students
  - Genomics lectures
  - Hands-on research exploration
  - Many research readings

https://www.youtube.com/onurmutlulectures

Lecture Playlist (Fall 2023):



Complete Lecture Playlist (Spring 2023):



Edit

Fall 2023 Schedule

Week	Date	Livestream	Meeting
W0	05.10 Thu.		L0: Project Introductions and Q&A
W1	11.10 Wed.	YouTube Live	L1: P&S Course Introduction & Scope (PDF) (PPT)
W2	25.10 Wed.		L2: Introduction to Genome Analysis (PDF) (PPT)
W3	01.11 Wed.		L3: From Molecules to Data: An Overview of DNA Sequencing Technologies (PDF) (PPT)
W4	08.11 Wed.		L4a: Fundamentals of Sequence Alignment: Algorithms and Applications (PDF) (PPT)
			L4b: Optimizing Sequence Search: Hashing, Indexing, and Filtering Techniques (PDF) (PPT)

Conclusion

- We covered various recent ideas to
  - Accelerate genome analysis
  - Analyze genomes in ways that were not possible before

- Enabling cost-effective, portable, fast, and accurate genome analysis has many implications
  - What are the new applications to enable with these unique benefits?

- Can we do even better?
  - Understanding and modifying the sequencing process for analyzing other types of biological data

- **Many future opportunities exist**
  - **Especially with new sequencing technologies**
  - **Especially with new applications and use cases**

Real-time Analysis of Genomic Sequences from Nanopore Electrical Signals by Fast and Accurate Hash-based Search

Can Firtina
canfirtina@gmail.com
https://cfirtina.com

3 May 2024

Tufts University

SAFARI

ETH zürich

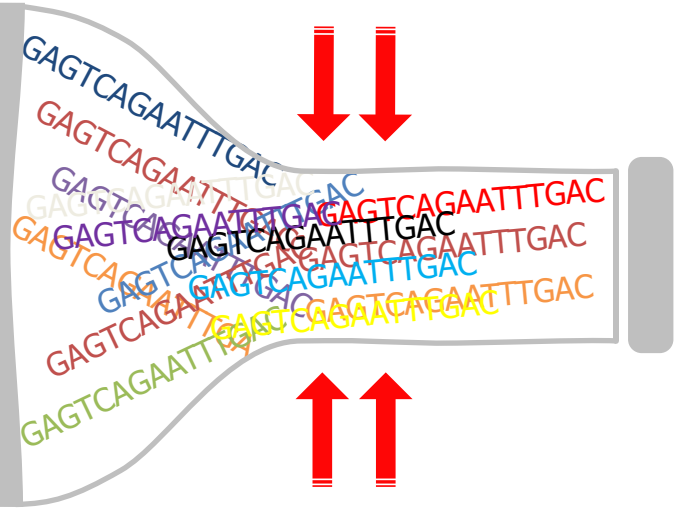
Analysis is Bottlenecked in Read Mapping!!

48 Human whole genomes

at 30× coverage

in about 2 days

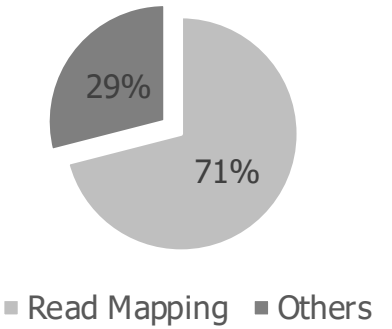
Illumina NovaSeq 6000



1 Human genome

32 CPU hours

on a 48-core processor



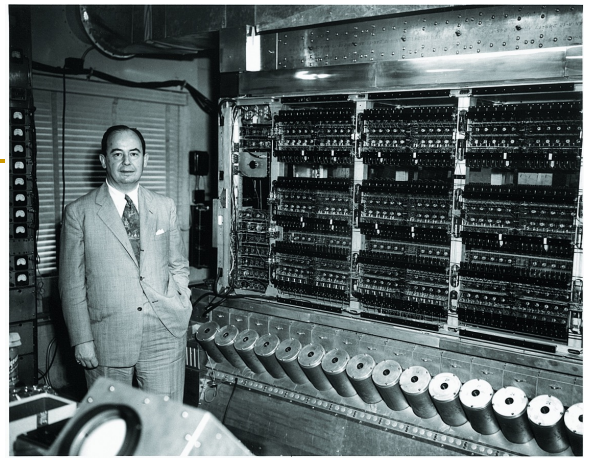
Goyal+, "Ultra-fast next generation human genome sequencing data processing using DRAGENTM bio-IT processor for precision medicine", *Open Journal of Genetics,* 2017.

A Tsunami of Sequencing Data

A Tera-scale increase in sequencing production in the past 25 years		
Genes & Operons	1990	**Kilo** = 1,000
Bacterial genomes	1995	**Mega** = 1,000,000
Human genome	2000	**Giga** = 1,000,000,000
Human microbiome	2005	**Tera** = 1,000,000,000,000
50K Microbiomes	2015	**Peta** = 1,000,000,000,000,000
what is expected for the next 15 years ? (a Giga?)		
200K Microbiomes	2020	**Exa** = 1,000,000,000,000,000,000
1M Microbiomes	2025	**Zetta** = 1,000,000,000,000,000,000,000
Earth Microbiome	2030	**Yotta** = 1,000,000,000,000,000,000,000,000

Source: @kyrpides

Today's Computing Systems

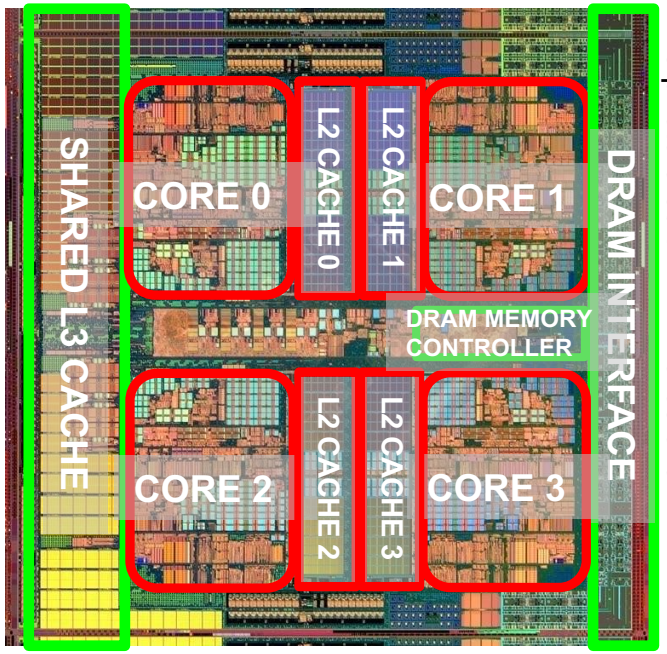


von Neumann model, 1945

where the **CPU** can **access data** stored in an off-chip main memory only through **power-hungry bus**

Storage (SSD/HDD)

Main Memory



*Die photo credit: AMD Barcelona

Microprocessor

Burks, Goldstein, von Neumann, "Preliminary discussion of the logical design of an electronic computing instrument," 1946.

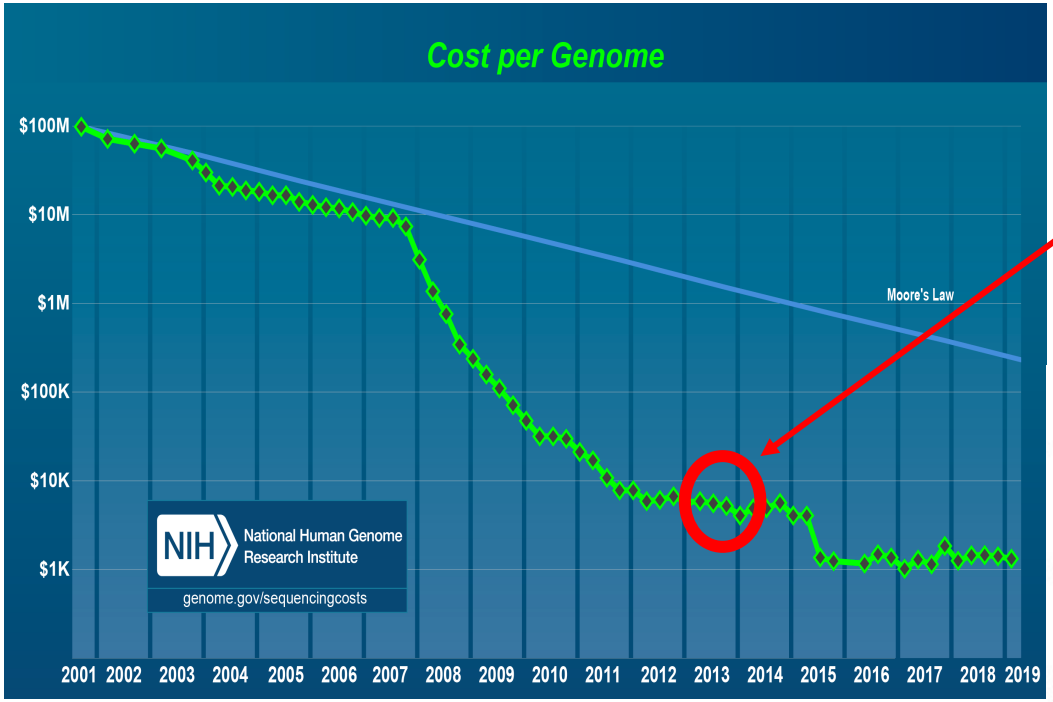
The Problem

Data analysis

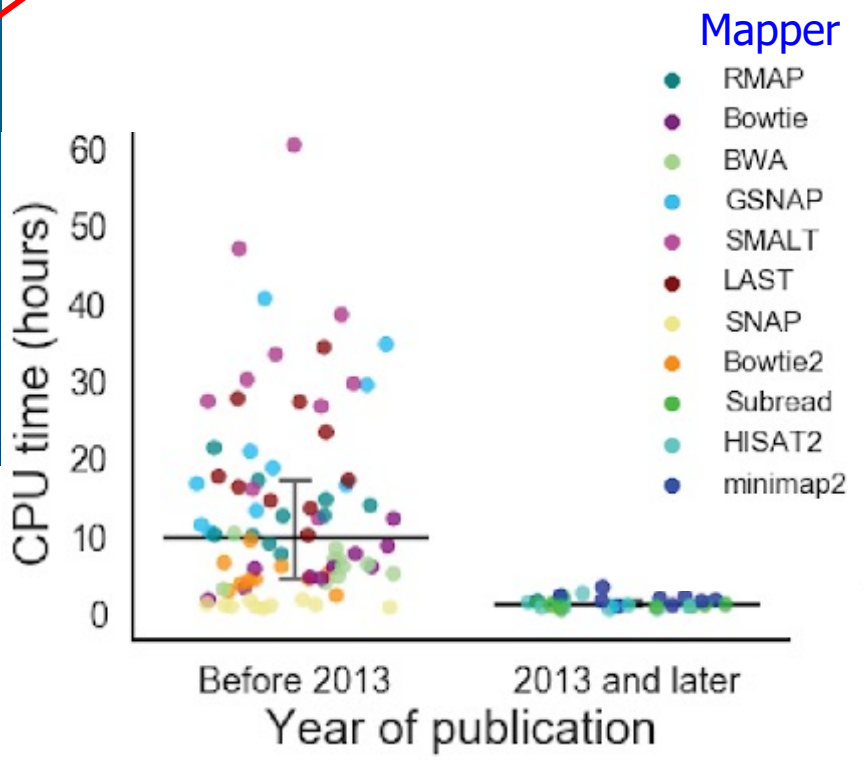
is performed

far away from the data

The Need for Speed



Did we realize the **need** for **faster** genome analysis?



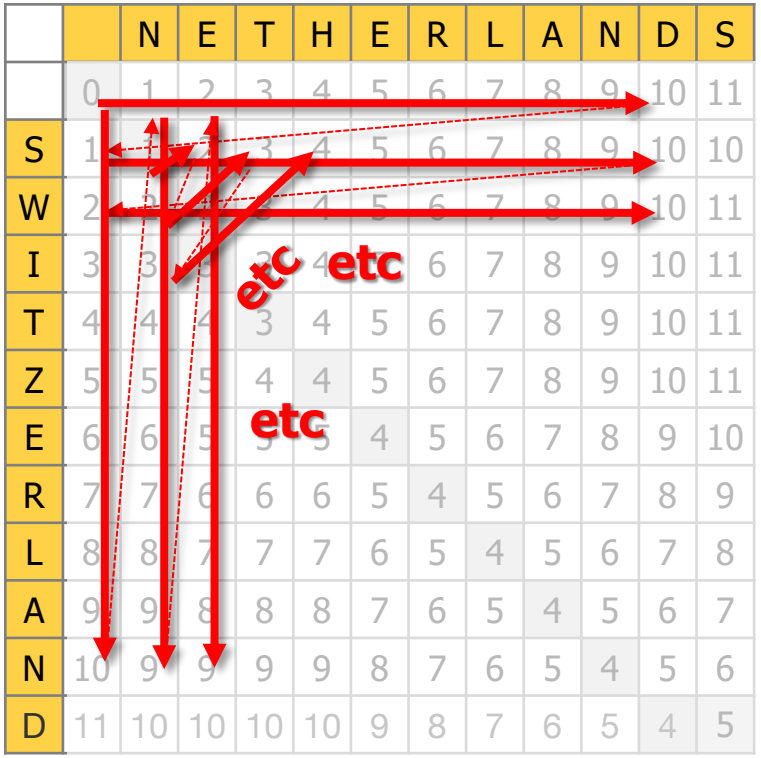
Alser+, "Technology dictates algorithms: Recent developments in read alignment", Genome Biology, 2021

Sequence Alignment in Unavoidable

- Quadratic-time dynamic-programming algorithm **WHY?!**

  Enumerating all possible prefixes

- NETHERLANDS x SWITZERLAND
- NETHERLANDS x S
  NETHERLANDS x SW
  NETHERLANDS x SWI
  NETHERLANDS x SWIT
  NETHERLANDS x SWITZ
  NETHERLANDS x SWITZE
  NETHERLANDS x SWITZER
  NETHERLANDS x SWITZERL
  NETHERLANDS x SWITZERLA
  NETHERLANDS x SWITZERLAN
  NETHERLANDS x SWITZERLAND



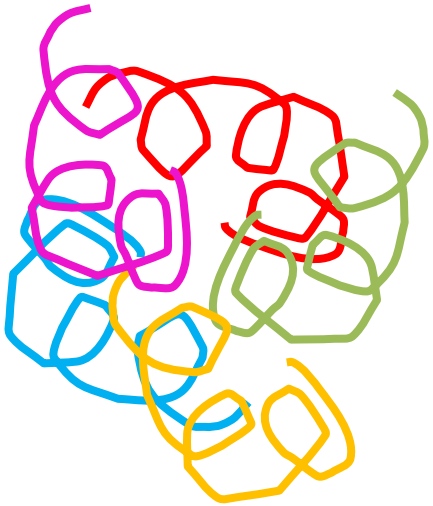
Sequence Alignment in Unavoidable

- **Quadratic-time** dynamic-programming algorithm
  - Enumerating all possible prefixes
- **Data dependencies** limit the computation parallelism
  - Processing row (or column) after another
- **Entire matrix** is computed even though strings can be dissimilar.
  - Number of differences is computed only at the backtraking step.

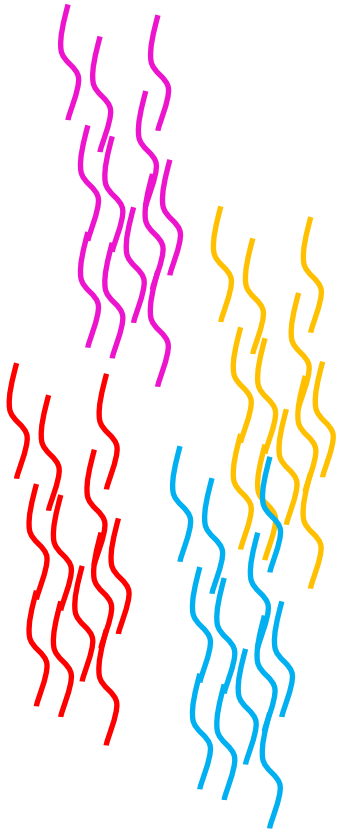
		N	E	T	H	E	R	L	A	N	D	S
	0	1	2	3	4	5	6	7	8	9	10	11
S	1	1	2	3	4	5	6	7	8	9	10	10
W	2	2	2	3	4	5	6	7	8	9	10	11
I	3	3	3	3	4	5	6	7	8	9	10	11
T	4	4	4	3	4	5	6	7	8	9	10	11
Z	5	5	5	4	4	5	6	7	8	9	10	11
E	6	6	5	5	5	4	5	6	7	8	9	10
R	7	7	6	6	6	5	4	5	6	7	8	9
L	8	8	7	7	7	6	5	4	5	6	7	8
A	9	9	8	8	8	7	6	5	4	5	6	7
N	10	9	9	9	9	8	7	6	5	4	5	6
D	11	10	10	10	10	9	8	7	6	5	4	5

Metagenomics Analysis

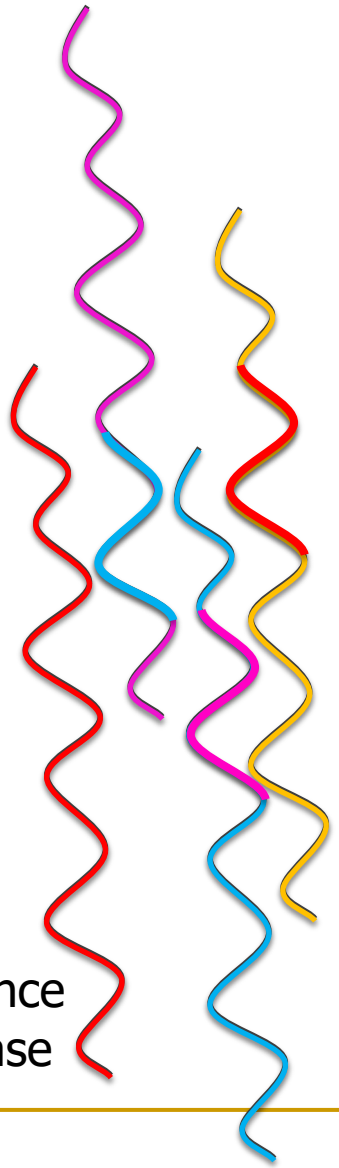
Reads from different unknown donors at sequencing time are mapped to many known reference genomes



genetic material recovered directly from environmental samples

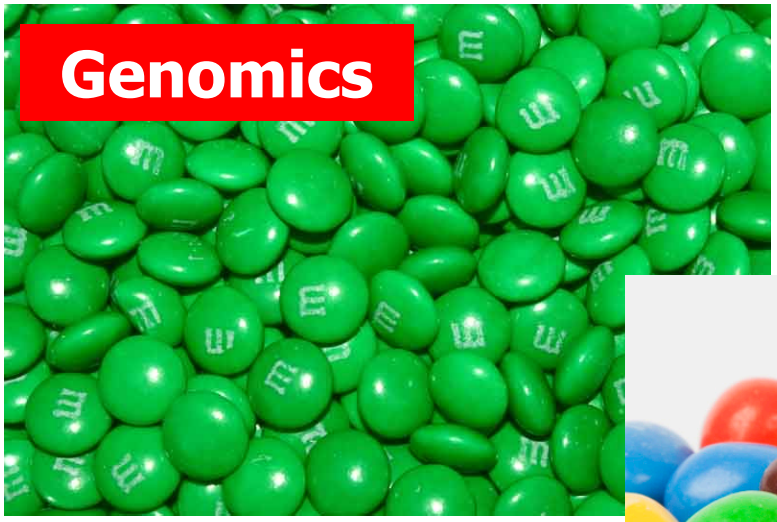


Reads "text format"



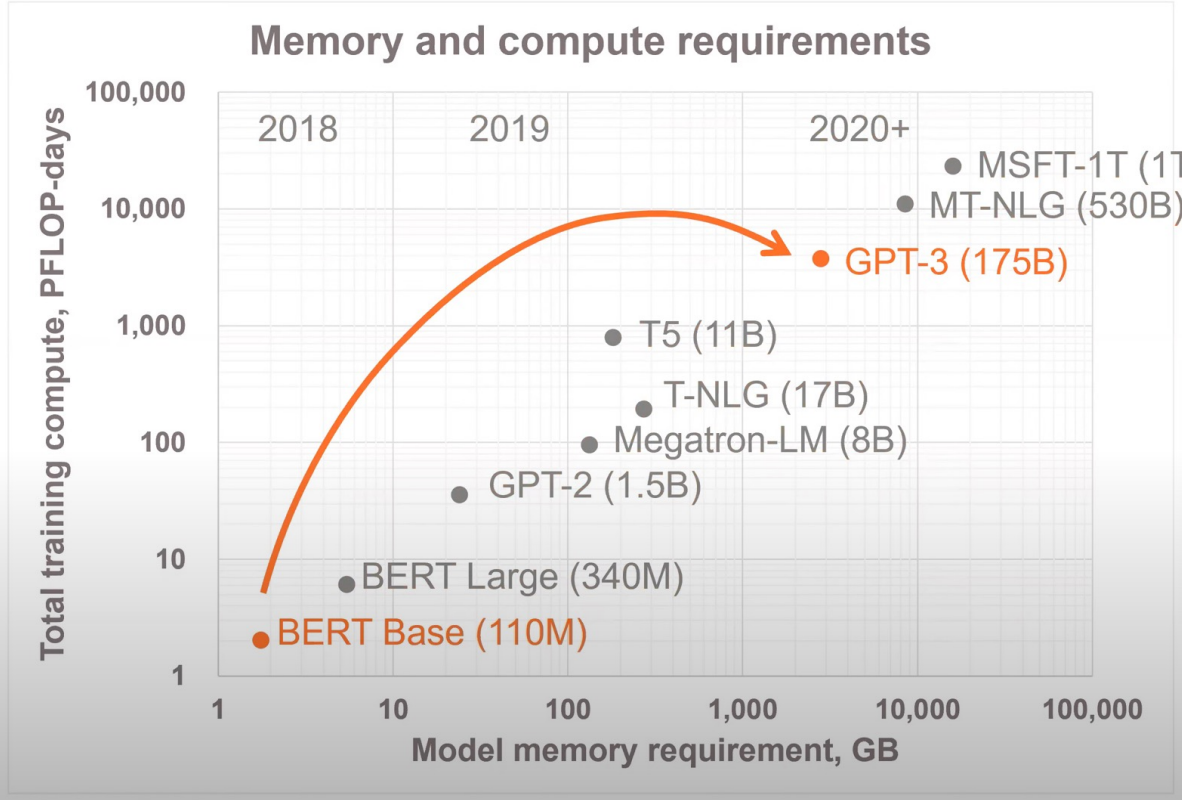
Reference Database

Genomics vs. Metagenomics



Huge Demand for Performance & Efficiency

Exponential Growth of Neural Networks



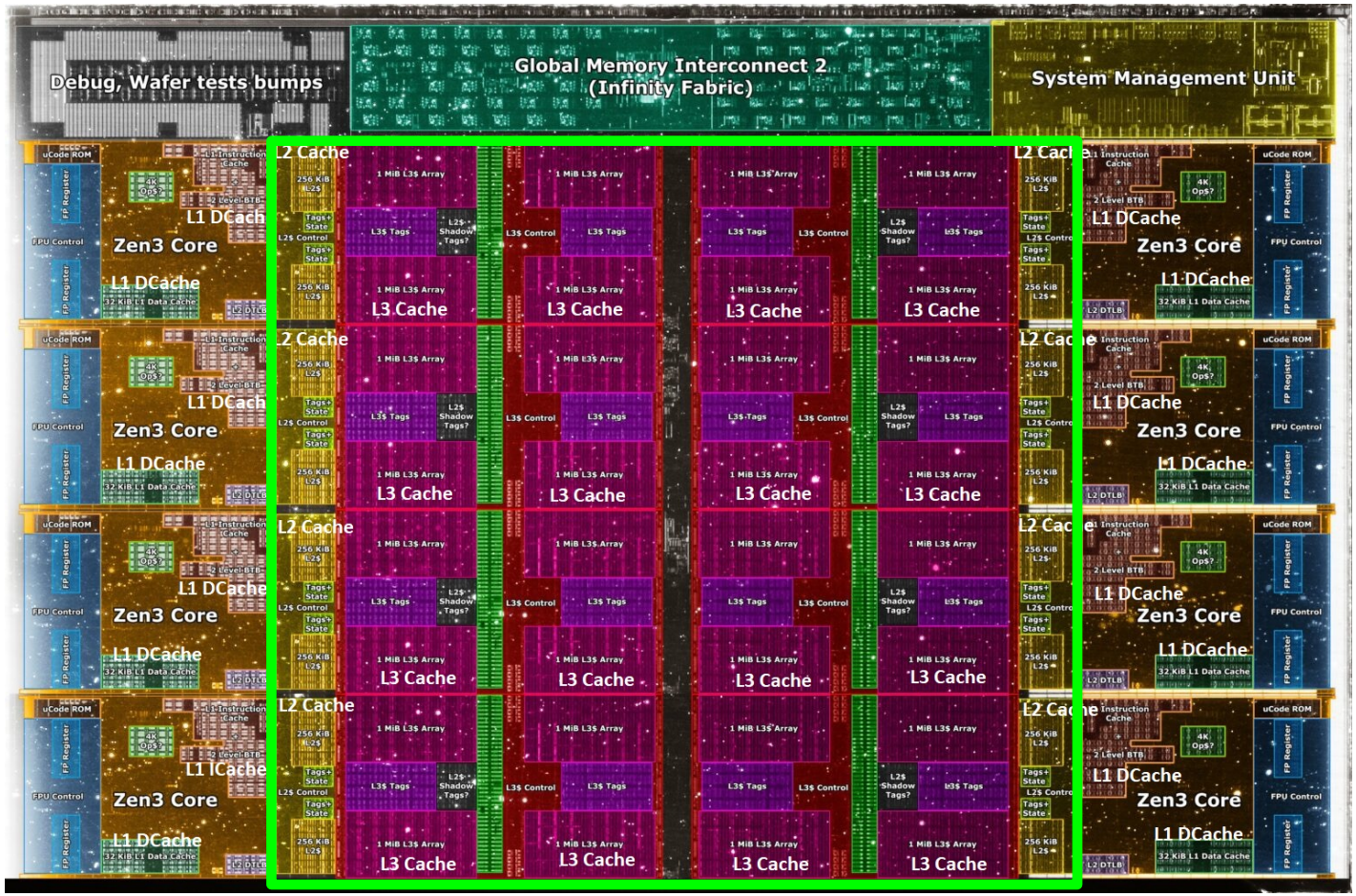
1800x more compute

In just **2 years**

Tomorrow, **multi-trillion** parameter models

Source: https://youtu.be/Bh13Idwcb0Q?t=283

Deeper and Larger Memory Hierarchies



AMD Ryzen 5000, 2020

Core Count:
8 cores/16 threads

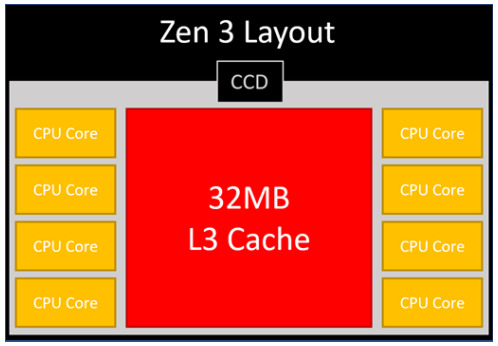
L1 Caches:
32 KB per core

L2 Caches:
512 KB per core

L3 Cache:
32 MB shared

https://wccftech.com/amd-ryzen-5000-zen-3-vermeer-undressed-high-res-die-shots-close-ups-pictured-detailed/

AMD's 3D Last Level Cache (2021)

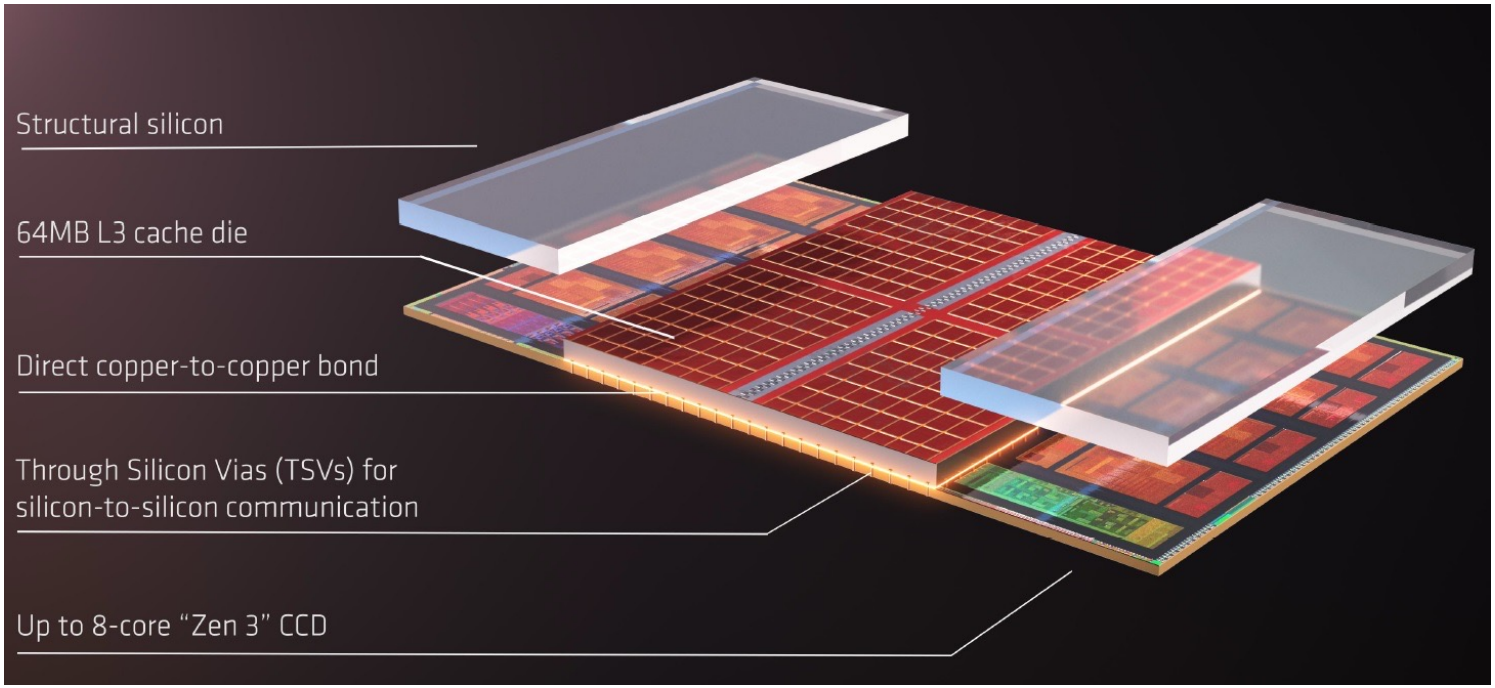


https://community.microcenter.com/discussion/5134/comparing-zen-3-to-zen-2

AMD increases the L3 size of their 8-core Zen 3 processors from 32 MB to 96 MB

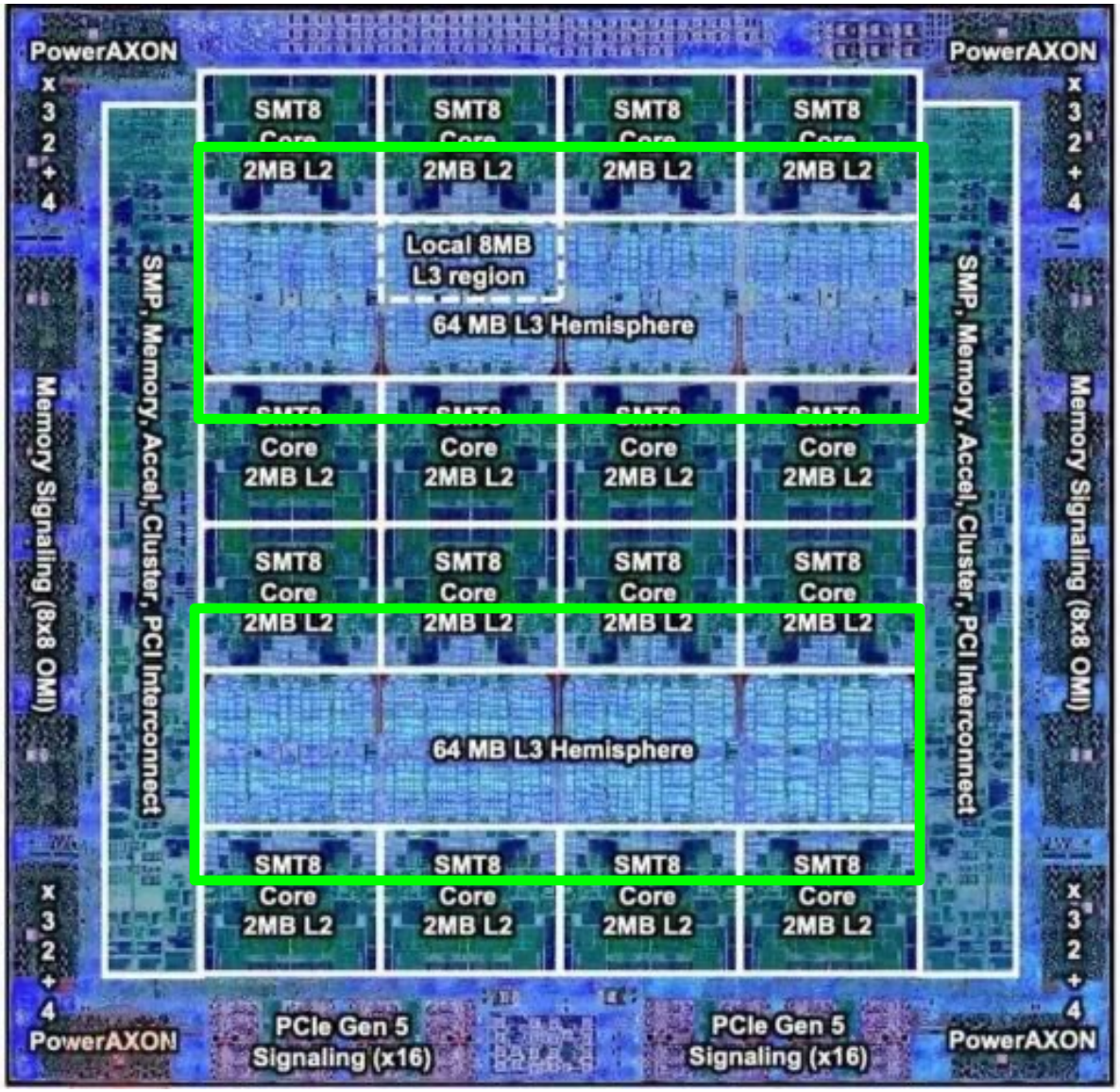
Additional 64 MB L3 cache die stacked on top of the processor die

- Connected using Through Silicon Vias (TSVs)
- Total of 96 MB L3 cache



https://youtu.be/qqAYMx34euU
https://www.tech-critter.com/amd-keynote-computex-2021/

Deeper and Larger Memory Hierarchies



IBM POWER10, 2020

Cores:

15-16 cores,
8 threads/core

L2 Caches:

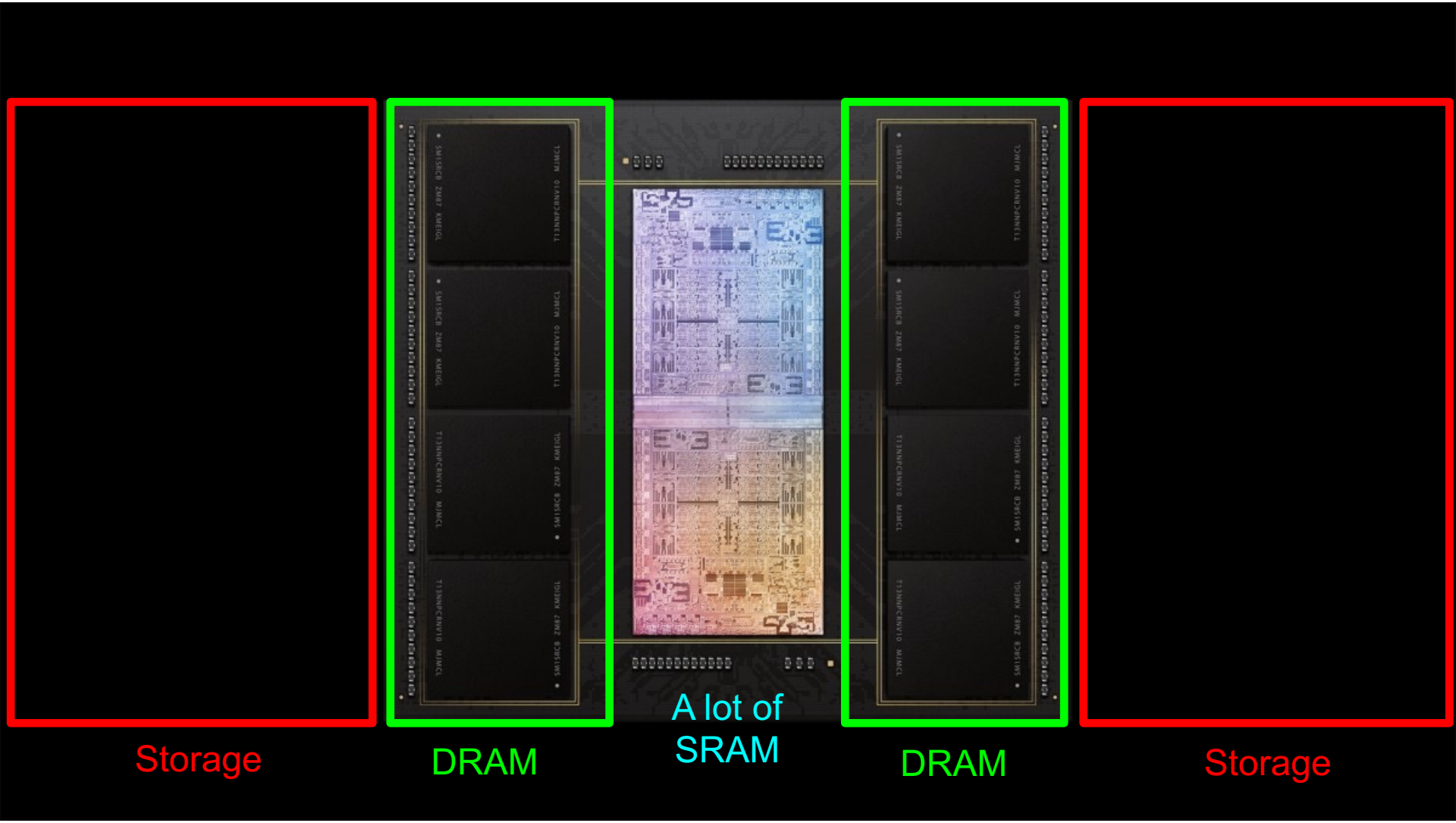
2 MB per core

L3 Cache:

120 MB shared

https://www.it-techblog.de/ibm-power10-prozessor-mehr-speicher-mehr-tempo-mehr-sicherheit/09/2020/

Deeper and Larger Memory Hierarchies



Apple M1 Ultra System (2022)

https://www.gsmarena.com/apple_announces_m1_ultra_with_20core_cpu_and_64core_gpu-news-53481.php

Data Movement Overwhelms Modern Machines

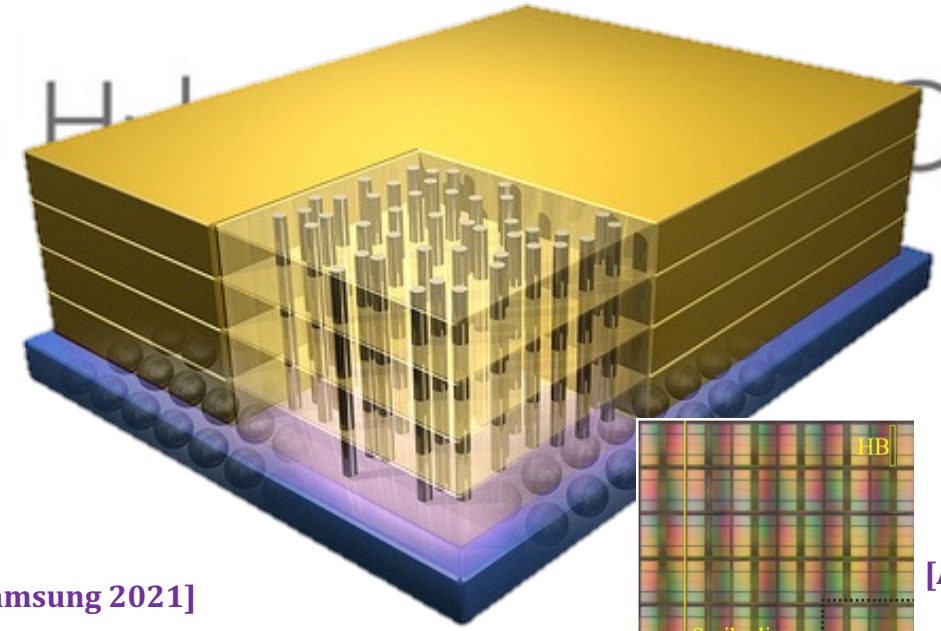
- Amirali Boroumand, Saugata Ghose, Youngsok Kim, Rachata Ausavarungnirun, Eric Shiu, Rahul Thakur, Daehyun Kim, Aki Kuusela, Allan Knies, Parthasarathy Ranganathan, and Onur Mutlu, **"Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks"** *Proceedings of the 23rd International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, Williamsburg, VA, USA, March 2018.

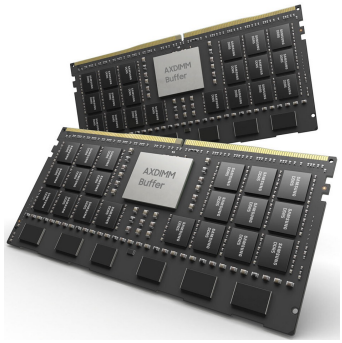
62.7% of the total system energy is spent on data movement

Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

Amirali Boroumand[1] Saugata Ghose[1] Youngsok Kim[2]
Rachata Ausavarungnirun[1] Eric Shiu[3] Rahul Thakur[3] Daehyun Kim[4,3]
Aki Kuusela[3] Allan Knies[3] Parthasarathy Ranganathan[3] Onur Mutlu[5,1]

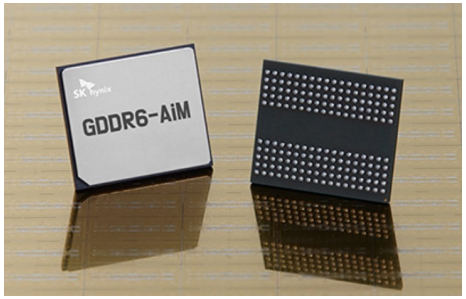
Processing-in-Memory Landscape Today



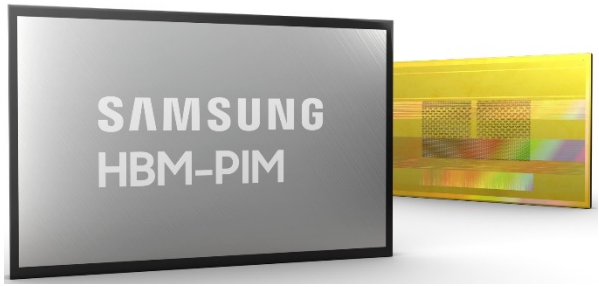


[Samsung 2021]

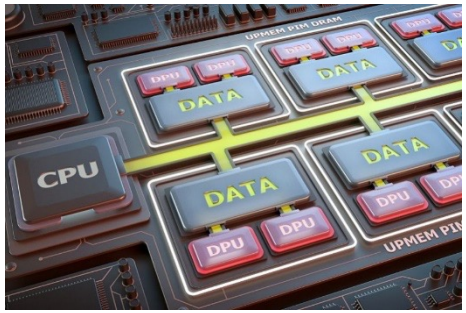
[Alibaba 2022]



[SK Hynix 2022]



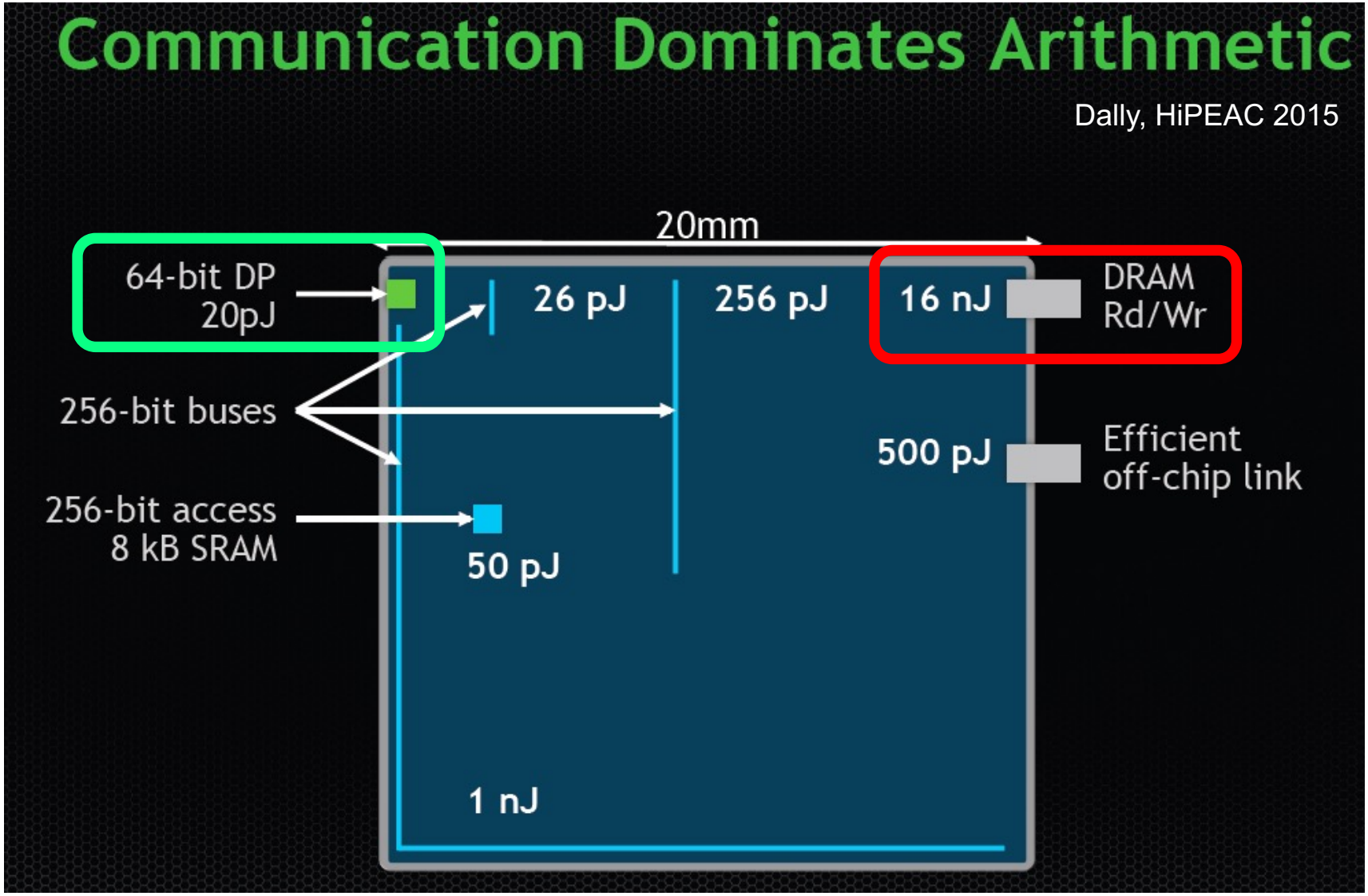
[Samsung 2021]



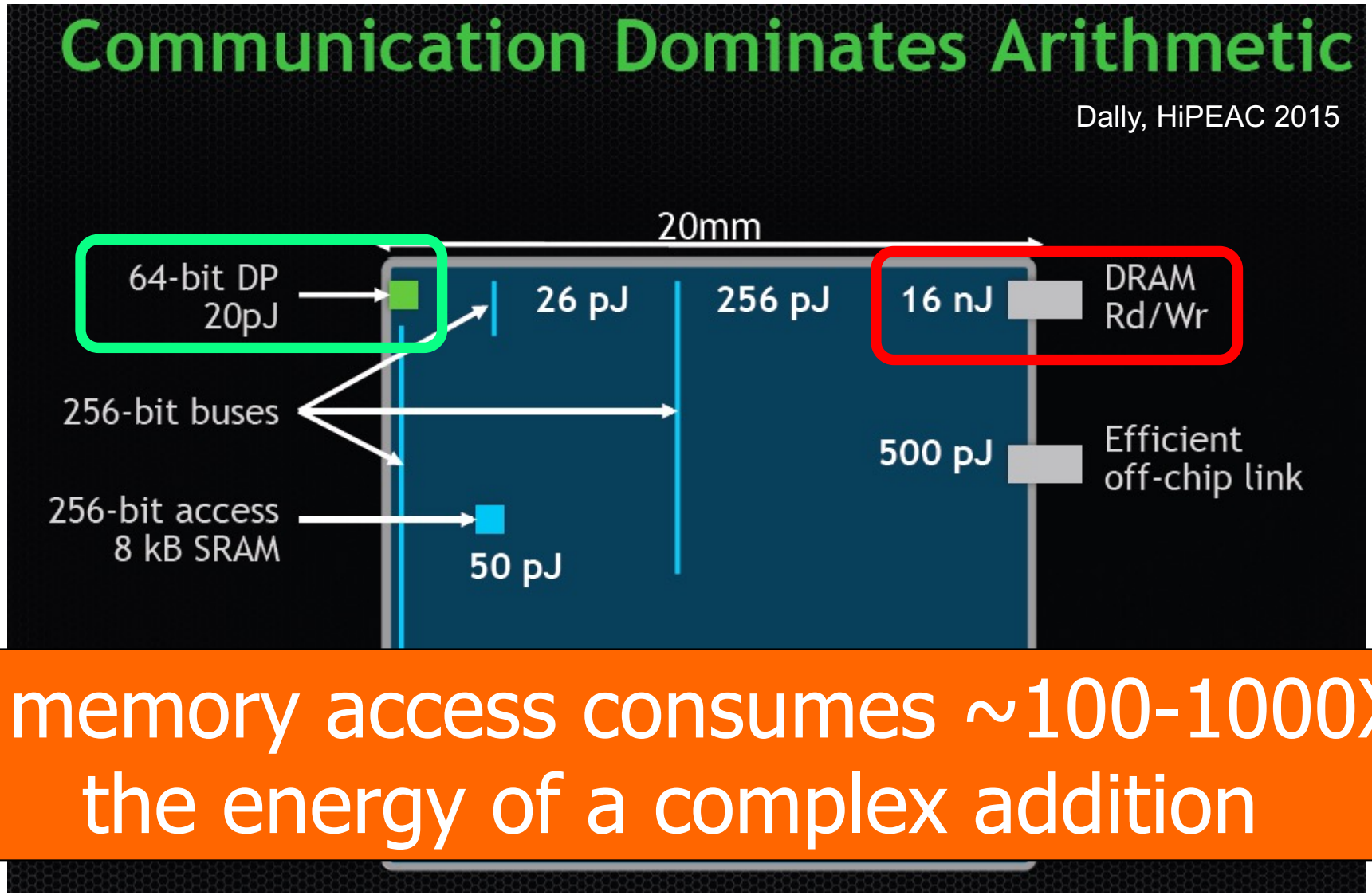
[UPMEM 2019]

And, many other experimental chips and startups

The Energy Perspective

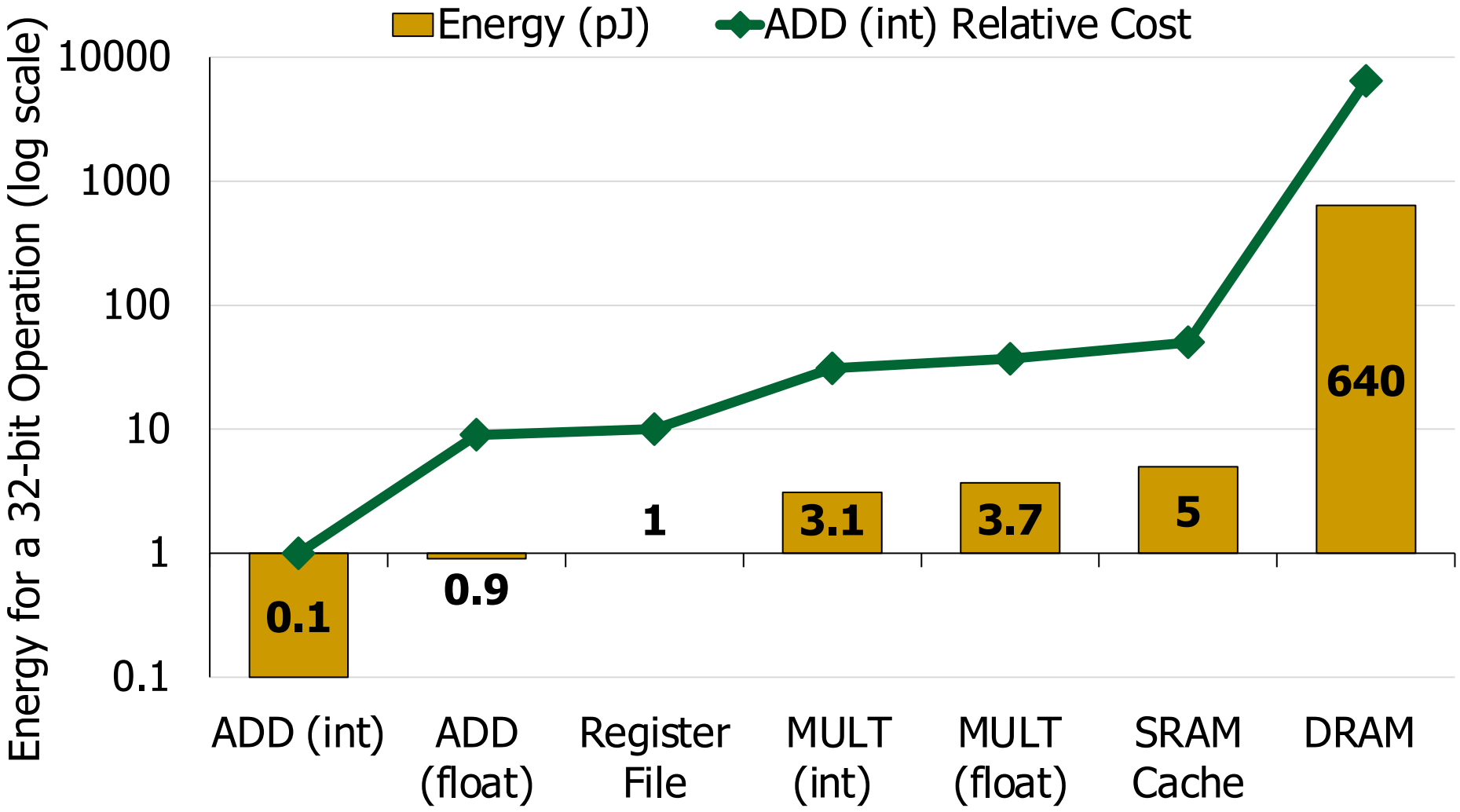


Data Movement vs. Computation Energy



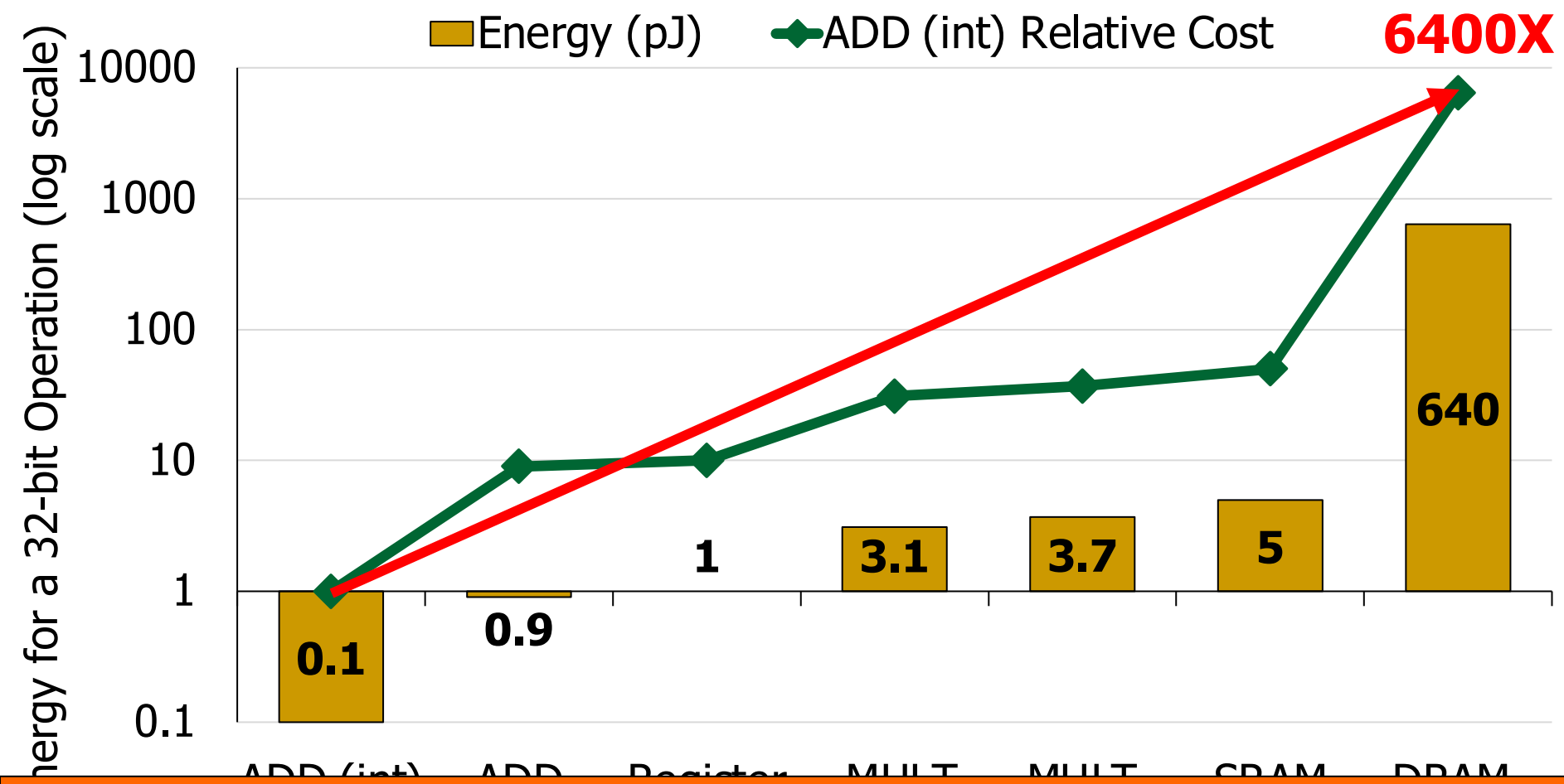
A memory access consumes ~100-1000X the energy of a complex addition

Data Movement vs. Computation Energy



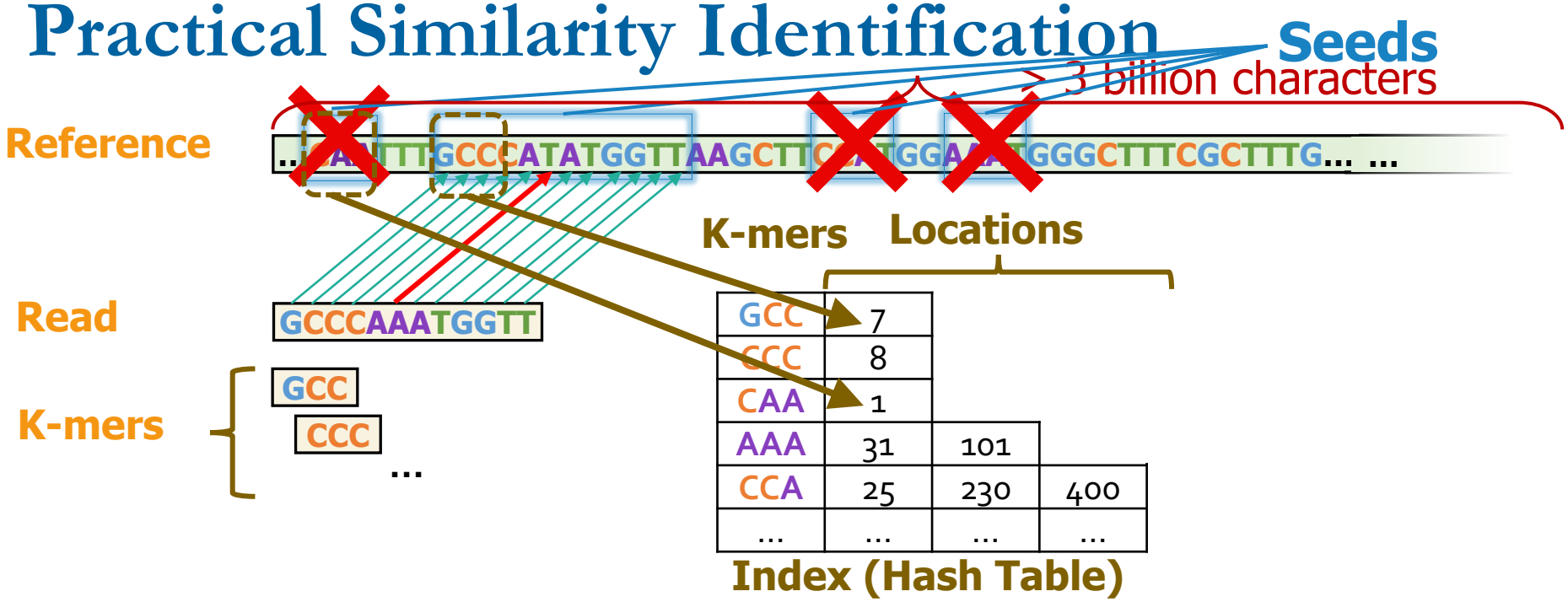
Han+, "EIE: Efficient Inference Engine on Compressed Deep Neural Network," ISCA 2016.

Data Movement vs. Computation Energy



A memory access consumes 6400X the energy of a simple integer addition

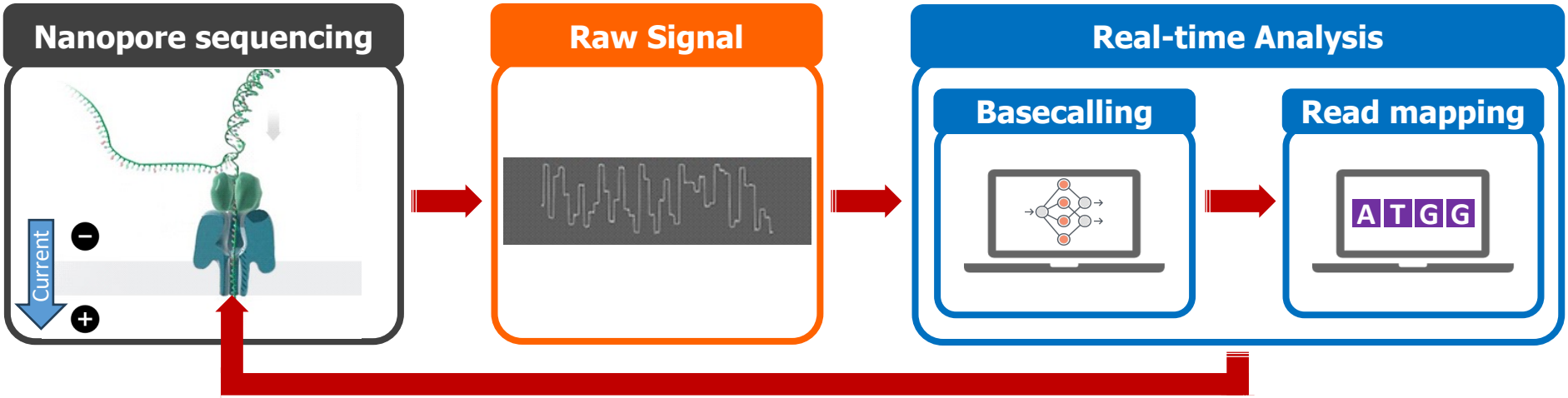
Practical Similarity Identification



Seeding	Determine potential matching regions (seeds) in the reference genome
Seed Filtering (e.g., Chaining)	Prune some seeds in the reference genome
Alignment	Determine the exact differences between the read and the reference genome

Existing Solutions – Real-time Basecalling

Deep neural networks (**DNNs**) for translating **signals** to **bases**

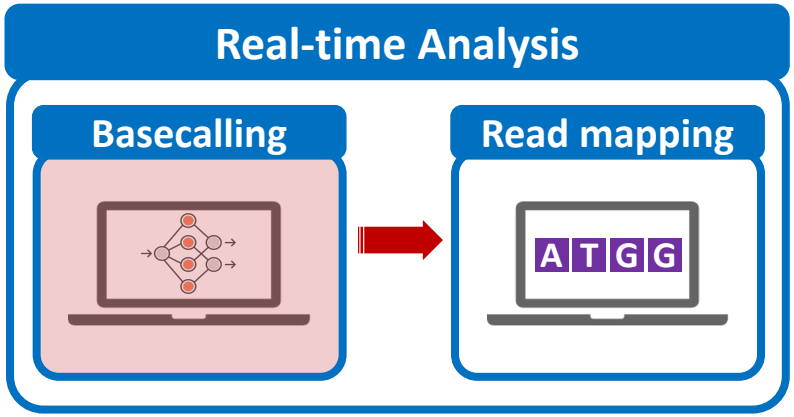


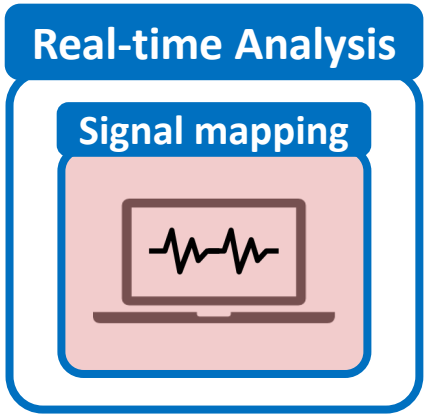
DNNs provide **less noisy analysis** from basecalled sequences

Costly and power-hungry computational requirements

The Problem

The existing solutions are **ineffective for large genomes**





Costly and energy-hungry computations to basecall each read:
Portable sequencing becomes challenging with resource-constrained devices

Larger number of reference regions **cannot be handled accurately or quickly**, rendering existing solutions **ineffective for large genomes**

Applications of Read Until

Depletion: Reads mapping to a particular reference genome is ejected

- Removing contaminated reads from a sample
- Relative abundance estimation
- Controlling low/high-abundance genomes in a sample
- Controlling the sequencing of depth of a genome

Enrichment: Reads **not** mapping to a particular reference genome is ejected

- Purifying the sample to ensure it contains only the selected genomes
- Removing the host genome (e.g., human) in contamination analysis

Applications of Run Until and Sequence Until

Run Until: Stopping the sequencing without informative decision from analysis

- Stopping when reads reach to a particular depth of coverage
- Stopping when the abundance of all genomes reach a particular threshold

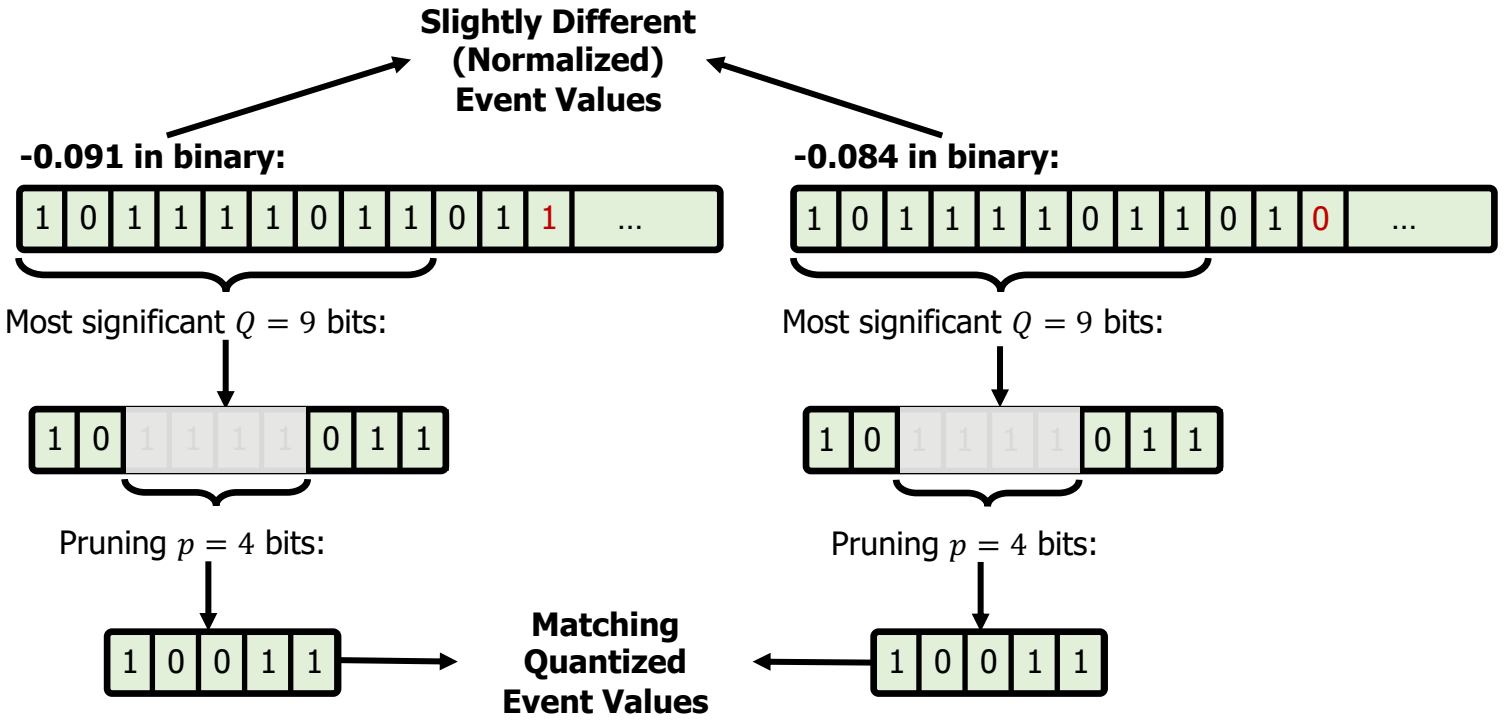
Sequence Until: Stopping the sequencing based on information decision

- Stopping when relative abundance estimations do not change substantially (for high-abundance genomes)
- Stopping when finding that the sample is contaminated with a particular set of genomes
- …

Details: Quantizing the Event Values

- **Observation:** Identical k-mers generate similar raw signals
  - **Challenge:** Their corresponding event values can be slightly different

- **Key Idea:** Quantize the event values
  - To enable assigning the **same quantized value** to the **similar event values**



Breakdown Analysis of the RawHash Steps

	Fraction of entire runtime (%)				
Tool	***SARS-CoV-2***	***E. coli***	***Yeast***	***Green Algae***	***Human***
File I/O	0.00	0.00	0.00	0.00	0.00
Signal-to-Event	21.75	1.86	1.01	0.53	0.02
Sketching	0.74	0.06	0.04	0.03	0.00
Seeding	3.86	4.14	3.52	6.70	5.39
Chaining	73.50	93.92	95.42	92.43	94.46
Seeding + Chaining	77.36	98.06	98.94	99.14	99.86

The entire runtime is **bottlenecked by the chaining step**

Required Computation Resources in Indexing

Tool	*Contamination*	*SARS-CoV-2*	*E. coli*	*Yeast*	*Green Algae*	*Human*	*Relative Abundance*
			CPU Time (sec)				
UNCALLED	8.72	9.00	11.08	18.62	285.88	4,148.10	4,382.38
Sigmap	0.02	0.04	8.66	24.57	449.29	36,765.24	40,926.76
RawHash	0.18	0.13	2.62	4.48	34.18	1,184.42	788.88
			Real time (sec)				
UNCALLED	1.01	1.04	2.67	7.79	280.27	4,190.00	4,471.82
Sigmap	0.13	0.25	9.31	25.86	458.46	37,136.61	41,340.16
RawHash	0.14	0.10	1.70	2.06	15.82	278.69	154.68
			Peak memory (GB)				
UNCALLED	0.07	0.07	0.13	0.31	11.96	48.44	47.81
Sigmap	0.01	0.01	0.40	1.04	8.63	227.77	238.32
RawHash	0.01	0.01	0.35	0.76	5.33	83.09	152.80

The indexing step of RawHash is **orders of magnitude faster** than the indexing steps of UNCALLED and Sigmap, especially **for large genomes**

RawHash requires **larger memory space** than UNCALLED

Required Computation Resources in Mapping

Tool	*Contamination*	*SARS-CoV-2*	*E. coli*	*Yeast*	*Green Algae*	*Human*	*Relative Abundance*
				CPU Time (sec)			
UNCALLED	265,902.26	36,667.26	35,821.14	8,933.52	16,769.09	262,597.83	586,561.54
Sigmap	4,573.18	1,997.84	23,894.70	11,168.96	31,544.55	4,837,058.90	11,027,652.91
RawHash	3,721.62	1,832.56	8,212.17	4,906.70	25,215.23	2,022,521.48	4,738,961.77
				Real time (sec)			
UNCALLED	20,628.57	2,794.76	1,544.68	285.42	2,138.91	8,794.30	19,409.71
Sigmap	6,725.26	3,222.32	2,067.02	1,167.08	2,398.83	158,904.69	361,443.88
RawHash	3,917.49	1,949.53	957.13	215.68	1,804.96	65,411.43	152,280.26
				Peak memory (GB)			
UNCALLED	0.65	0.19	0.52	0.37	0.81	9.46	9.10
Sigmap	111.69	28.26	111.11	14.65	29.18	311.89	489.89
RawHash	4.13	4.20	4.16	4.37	11.75	52.21	55.31

The mapping step of RawHash is **significantly faster than Sigmap** for all genomes, and **faster than UNCALLED for small genomes**

RawHash requires **larger memory space** than UNCALLED