

Accelerating Genome Analysis

A Primer on an Ongoing Journey

Onur Mutlu

omutlu@gmail.com

<https://people.inf.ethz.ch/omutlu>

8 June 2023

University of Montenegro Faculty of Medicine

SAFARI

ETH zürich

Carnegie Mellon

Overview

- System design for bioinformatics is a critical problem
 - It has large scientific, medical, societal, personal implications
- This talk is about accelerating a key step in bioinformatics: genome sequence analysis
 - In particular, read mapping
- Many bottlenecks exist in accessing and manipulating huge amounts of genomic data during analysis
- Many recent ideas to accelerate read mapping
 - My personal journey since September 2006

Our Dream (circa 2007)

- An embedded device that can perform comprehensive genome analysis in real time (within a minute)
 - Which of these DNAs does this DNA segment match with?
 - What is the likely genetic disposition of this patient to this drug?
 - What disease/condition might this particular DNA/RNA piece associated with?
 - . . .

We Need Faster & Scalable Genome Analysis



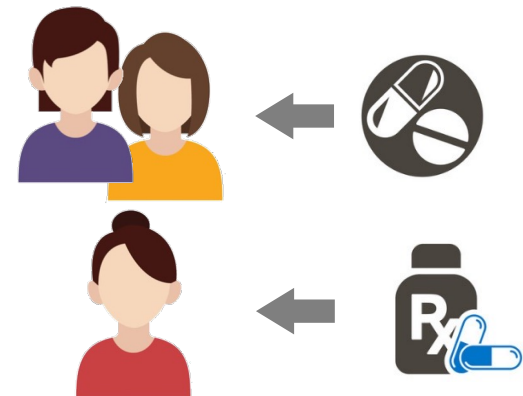
Understanding **genetic variations**,
species, **evolution**, ...



Predicting the **presence** and **relative abundance** of **microbes** in a sample



Rapid surveillance of **disease outbreaks**



Developing **personalized medicine**

A Bright Future for Intelligent Genome Analysis

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, Onur Mutlu
[“Accelerating Genome Analysis: A Primer on an Ongoing Journey”](#) IEEE Micro, August 2020.



MinION from ONT

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Sept.-Oct. 2020, pp. 65-75, vol. 40

DOI Bookmark: [10.1109/MM.2020.3013728](https://doi.org/10.1109/MM.2020.3013728)

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)



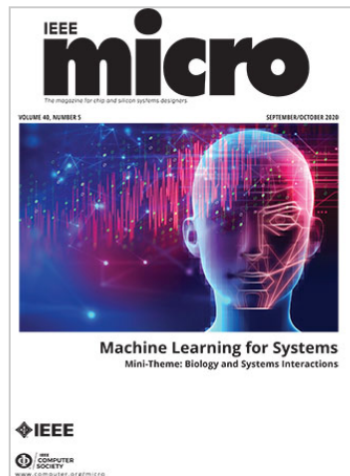
SmidgION from ONT

A Few Overview Readings (I)

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, Onur Mutlu

[“Accelerating Genome Analysis: A Primer on an Ongoing Journey”](#)

IEEE Micro, August 2020.



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2020.05](#)

IEEE Micro

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Sept.-Oct. 2020, pp. 65-75, vol. 40

DOI Bookmark: [10.1109/MM.2020.3013728](https://doi.org/10.1109/MM.2020.3013728)

Authors

[Mohammed Alser](#), ETH Zürich

[Zulal Bingöl](#), Bilkent University

[Damla Senol Cali](#), Carnegie Mellon University

[Jeremie Kim](#), ETH Zurich and Carnegie Mellon University

[Saugata Ghose](#), University of Illinois at Urbana-Champaign and Carnegie Mellon University

[Can Alkan](#), Bilkent University

[Onur Mutlu](#), ETH Zurich, Carnegie Mellon University, and Bilkent University



Previous



Next



Table of Contents



Past Issues

A Few Overview Readings (II)

Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gomez-Luna, Henk Corporaal, Onur Mutlu,

“FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications”

IEEE Micro, 2021.

[[Source Code](#)]



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2021.04](#)

IEEE Micro

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)

Authors

[Gagandeep Singh](#), ETH Zürich, Zürich, Switzerland

[Mohammed Alser](#), ETH Zürich, Zürich, Switzerland

[Damla Senol Cali](#), Carnegie Mellon University, Pittsburgh, PA, USA

[Dionysios Diamantopoulos](#), Zürich Lab, IBM Research Europe, Rüschlikon, Switzerland

[Juan Gomez-Luna](#), ETH Zürich, Zürich, Switzerland

[Henk Corporaal](#), Eindhoven University of Technology, Eindhoven, The Netherlands

[Onur Mutlu](#), ETH Zürich, Zürich, Switzerland

◀	▶
Previous	Next
☰	Table of Contents
📄	Past Issues

A Few Overview Readings (III)

Mohammed Alser, Joel Lindegger, Can Firtina, Nour Almadhoun, Haiyu Mao, Gagandeep Singh, Juan Gomez-Luna, Onur Mutlu

"From Molecules to Genomic Variations: Intelligent Algorithms and Architectures for Intelligent Genome Analysis"

Computational and Structural Biotechnology Journal, 2022

[[Source code](#)]



ELSEVIER



journal homepage: www.elsevier.com/locate/csbj



Review

From molecules to genomic variations: Accelerating genome analysis via intelligent algorithms and architectures



Mohammed Alser*, Joel Lindegger, Can Firtina, Nour Almadhoun, Haiyu Mao, Gagandeep Singh, Juan Gomez-Luna, Onur Mutlu*

ETH Zurich, Gloriastrasse 35, 8092 Zürich, Switzerland

SAFARI

<https://arxiv.org/pdf/2205.07957.pdf>

A Few Overview Readings (IV)

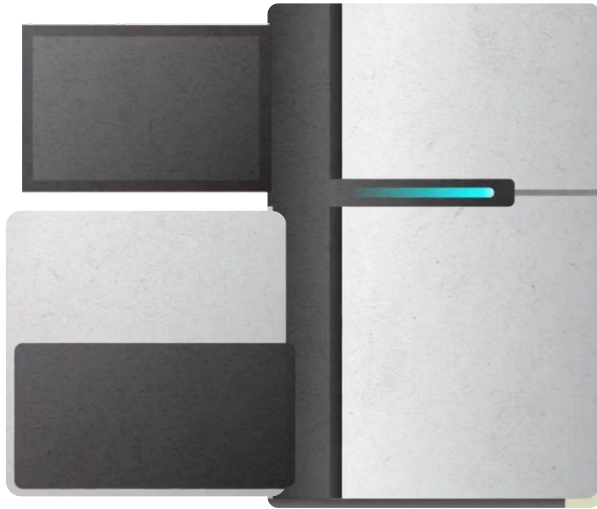
- Onur Mutlu and Can Firtina,
"Accelerating Genome Analysis via Algorithm-Architecture Co-Design"
Invited Special Session Paper in Proceedings of the [60th Design Automation Conference \(DAC\)](#), San Francisco, CA, USA, July 2023.
[\[arXiv version\]](#)

Invited: Accelerating Genome Analysis via Algorithm-Architecture Co-Design

Onur Mutlu Can Firtina

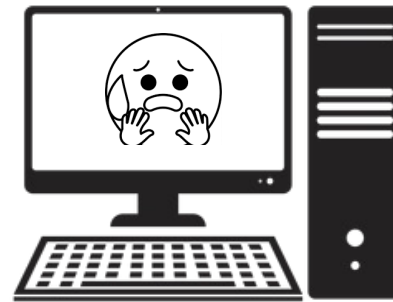
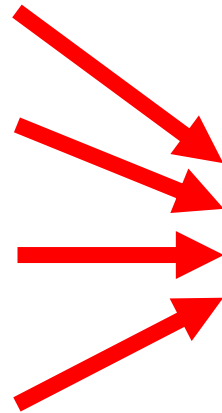
ETH Zürich

Problems with (Genome) Analysis Today



Special-Purpose Machine
for **Data Generation**

FAST



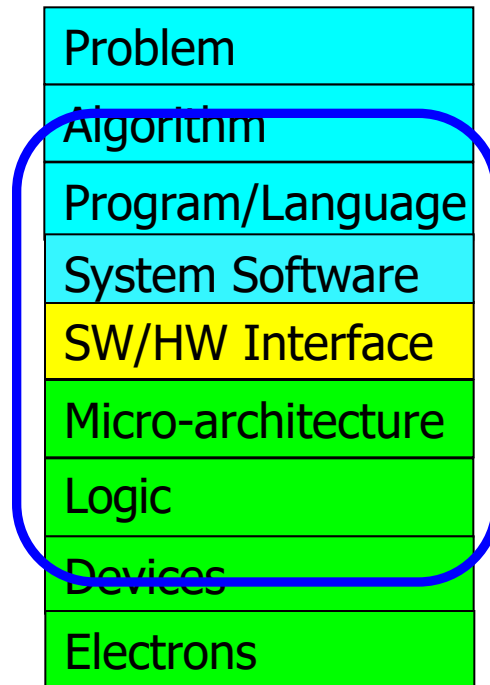
General-Purpose Machine
for **Data Analysis**

SLOW

Slow and inefficient processing capability
Large amounts of data movement

Algorithm-Arch-Device Co-Design is Critical

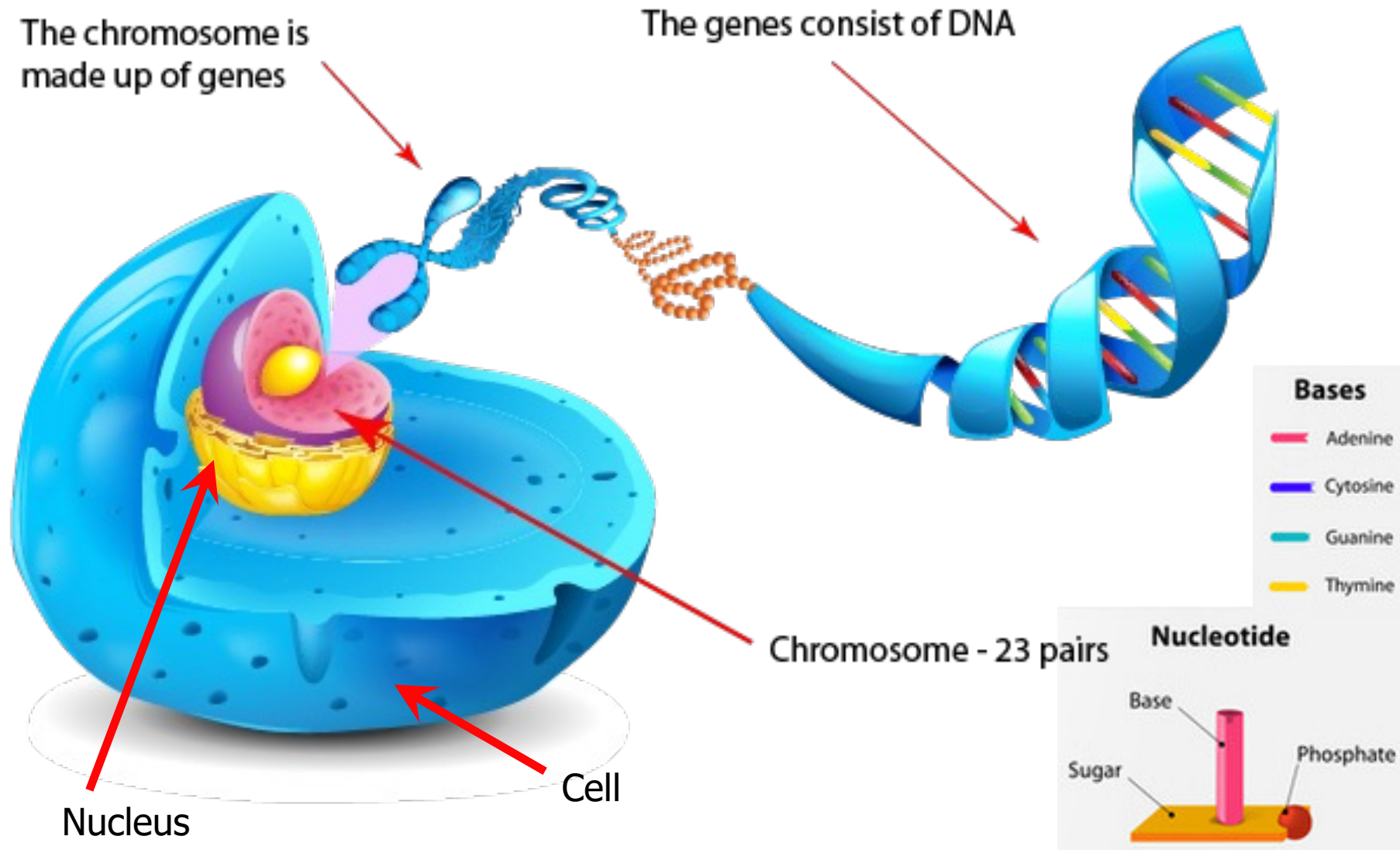
**Computer Architecture
(expanded view)**



Agenda

- The Problem: DNA Read Mapping
 - State-of-the-art Read Mapper Design
- Algorithmic Acceleration
 - Exploiting Structure of the Genome
 - Exploiting SIMD Instructions
- Hardware Acceleration
 - Specialized Architectures
 - Processing in Memory & Storage
- Future Opportunities: New Technologies & Applications

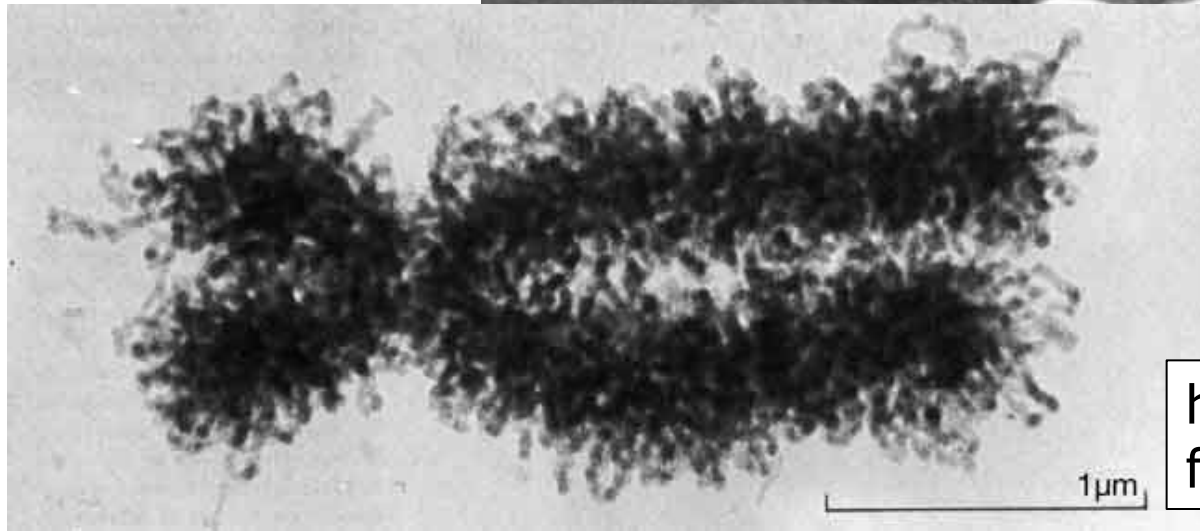
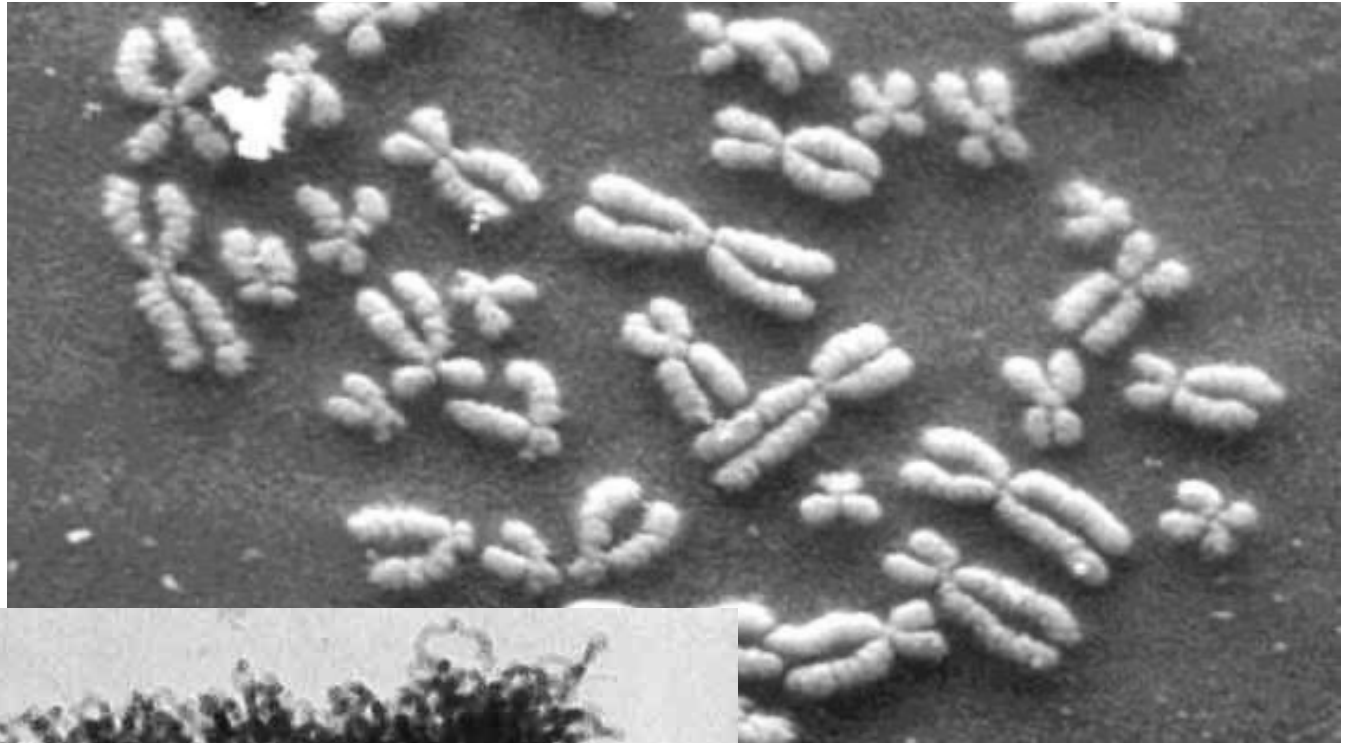
What Is a Genome Made Of?



The diagram illustrates the Central Dogma of Molecular Biology, showing the flow of genetic information from DNA to RNA to Protein. It features three large, colored arrows pointing right: a red arrow for DNA (Genotypes), a brown arrow for RNA, and a green arrow for Protein (Phenotypes). A curved arrow labeled 'Transcription' points from DNA to RNA, and another curved arrow labeled 'Translation' points from RNA to Protein. A third curved arrow labeled 'Replication' points from DNA back to DNA. Small molecular models are shown: a DNA double helix at the top left, an mRNA strand at the bottom of the RNA arrow, and a ribosome at the bottom right. A vertical strip of four human faces is positioned above the Protein arrow, representing the phenotypic expression of the genotype.



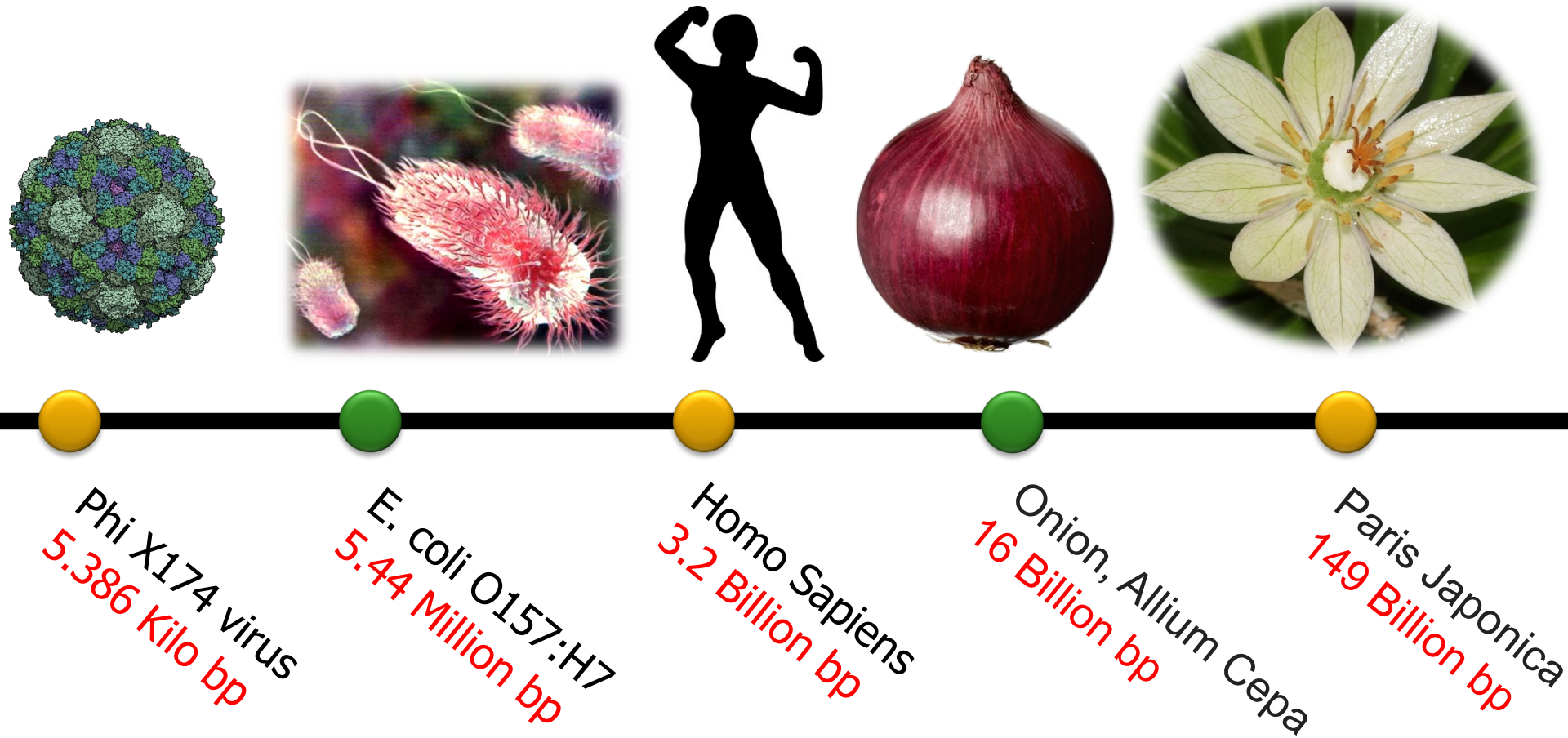
DNA Under Electron Microscope



human chromosome #12
from HeLa's cell

CCTCCTCAGTGCCACCCAGCCCCTGGCAGCTCCCAAACA
GGCTCTTATTAAAACACCCTGTTCCCTGCCCCCTTGGAGTG
AGGTGTCAAGGACCTAAACTAAAAAAAAAAAAAAGAAAA
AGAAAAGAAAAAGAATTTAAAATTTAAGTAATTCTTTGAA
AAAAACTAATTTCTAAGCTTCTTCATGTCAAGGACCTAATG
TGCTAAACAGCACTTTT**TTGACCATTAT**TTTGGATCTGAAA
GAAATCAAGAATAAATGAAGGACTTGATACATTGGAAGA
GGAGAGTCAAGGACCTACAGAAAAAAAAAAAAAAAAAGAAA
AAGAAAAGAAAAAGA**A**TTTAAAATTTAAGTAATTCTTTGA
AAAAAACTAATTTCTAAGCTTCTT**C**ATGTCAAGGACCTAAT
GTCTGTGTTGCAGGTCTTCTTGCATTTCCCTGTCAAAGA
AAAAGAATTTAAAATTTAAGTAATTCTTTGAAAAAAAACTA
ATTTCTAAGCTTCTTCATGTCAAGGACCTAATGTCAGGCC
GGCTCTTATTAAAACACCCTGTTCCCTGCCCCCTTGGAGTG

How Large is a Genome?



DNA Sequencing

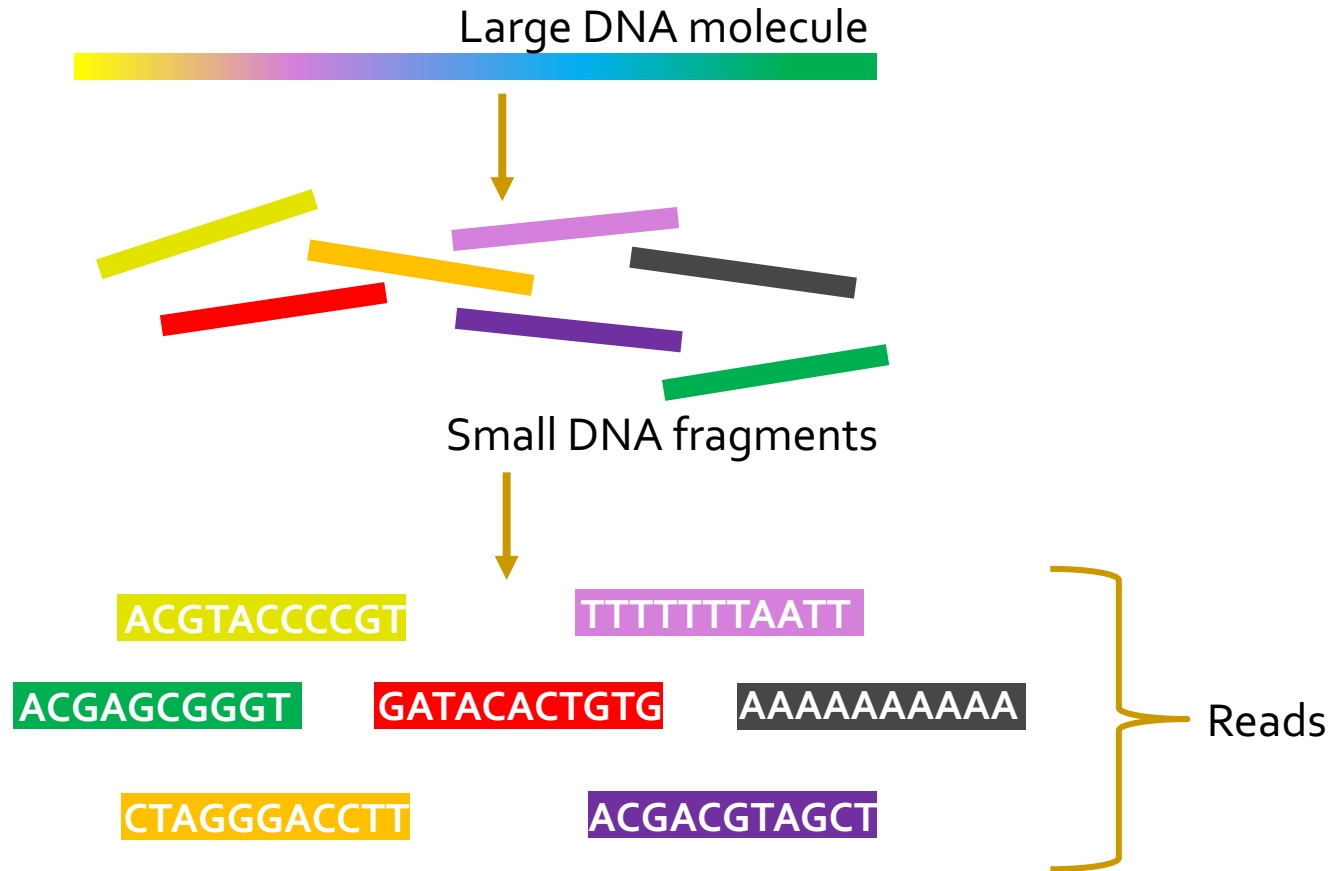
- Goal:

- Find the complete sequence of A, C, G, T's in an organism's DNA

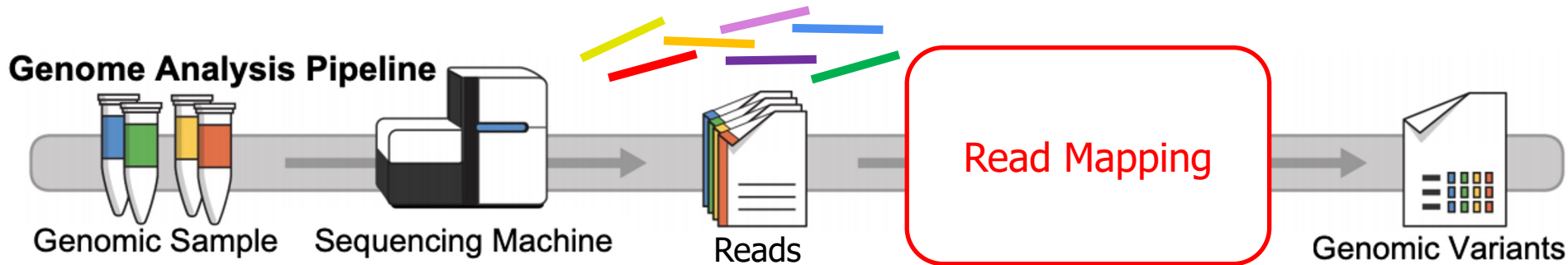
- Challenge:

- There is no machine that takes long DNA as an input, and gives the complete sequence as output
- All sequencing machines chop DNA into pieces and identify relatively small pieces (but not how they fit together)

Genome Sequencing



Genome Sequencing and Analysis



Current sequencing machines provide
small randomized fragments
of the original DNA sequence

Alser+, "[Technology dictates algorithms: Recent developments in read alignment](#)", Genome Biology, 2021

Solving the Puzzle

.FASTA file



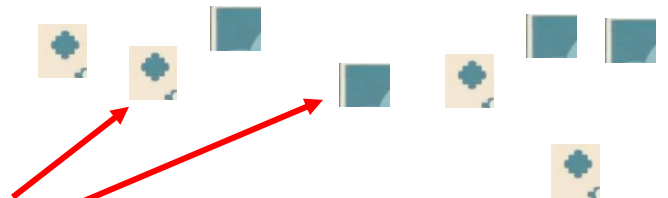
**Reference
genome**



.FASTQ file



**Sequenced
Reads**



<https://www.pacb.com/smrt-science/smrt-sequencing/hifi-reads-for-highly-accurate-long-read-sequencing/>

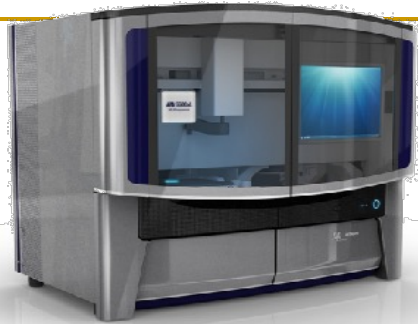
Untangling Yarn Balls & DNA Sequencing



Genome Sequencers



Roche/454



AB SOLiD



Illumina MiSeq



Complete Genomics



Illumina HiSeq2000



Pacific Biosciences RS



Oxford Nanopore MinION



Illumina NovaSeq 6000



Ion Torrent PGM



Ion Torrent Proton



Oxford Nanopore GridION

SAFARI

... and more! All produce data with different properties.

High-Throughput Sequencers



Illumina MiSeq



Pacific
Biosciences
Sequel II

Oxford
Nanopore
PromethION



Oxford Nanopore MinION



Illumina NovaSeq 6000



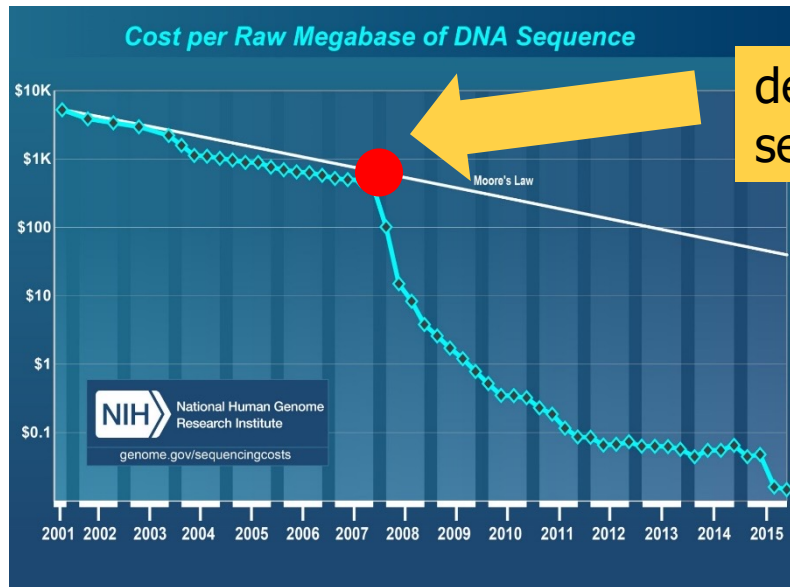
Pacific Biosciences RS II



Oxford
Nanopore
SmidgION

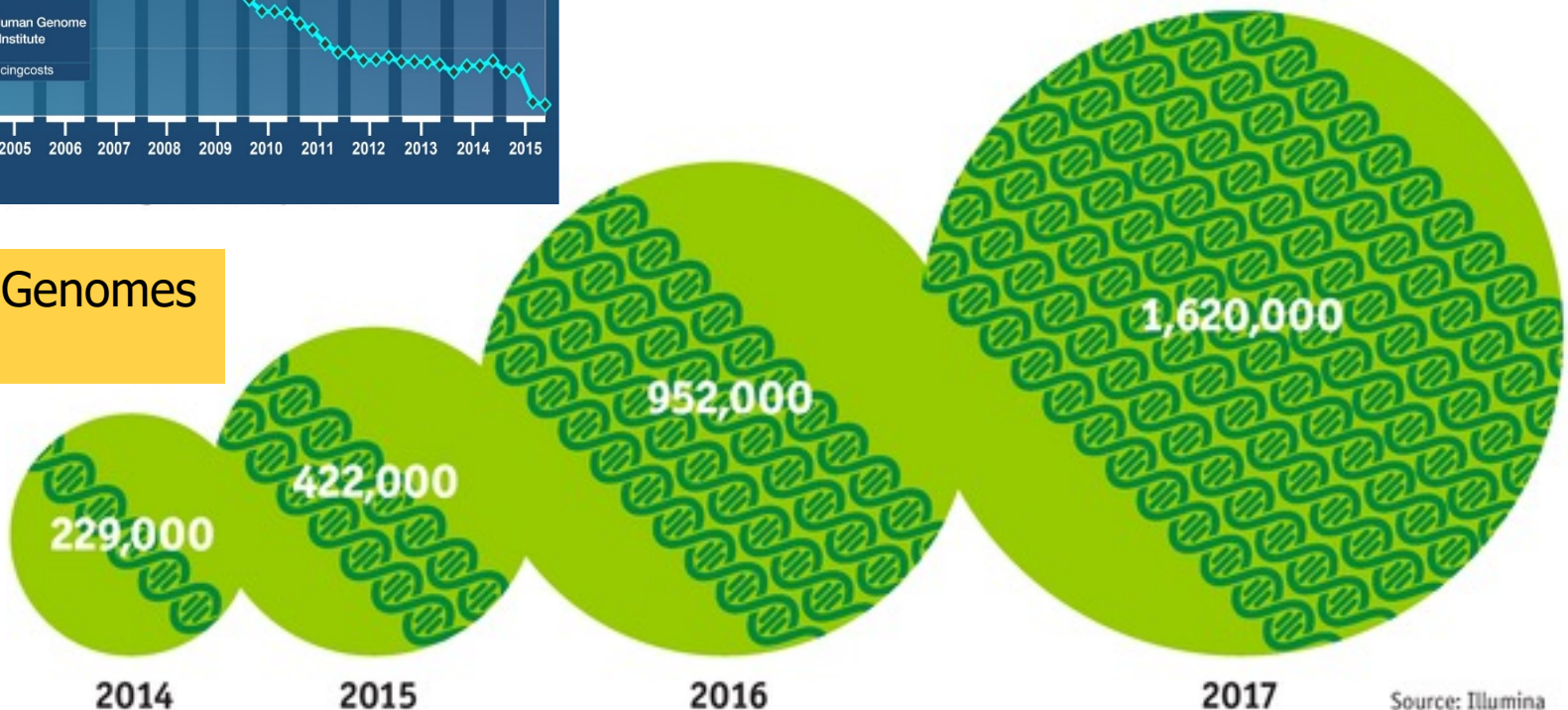
... and more! All produce data with different properties.

The Genomic Era



development of high-throughput sequencing (HTS) technologies

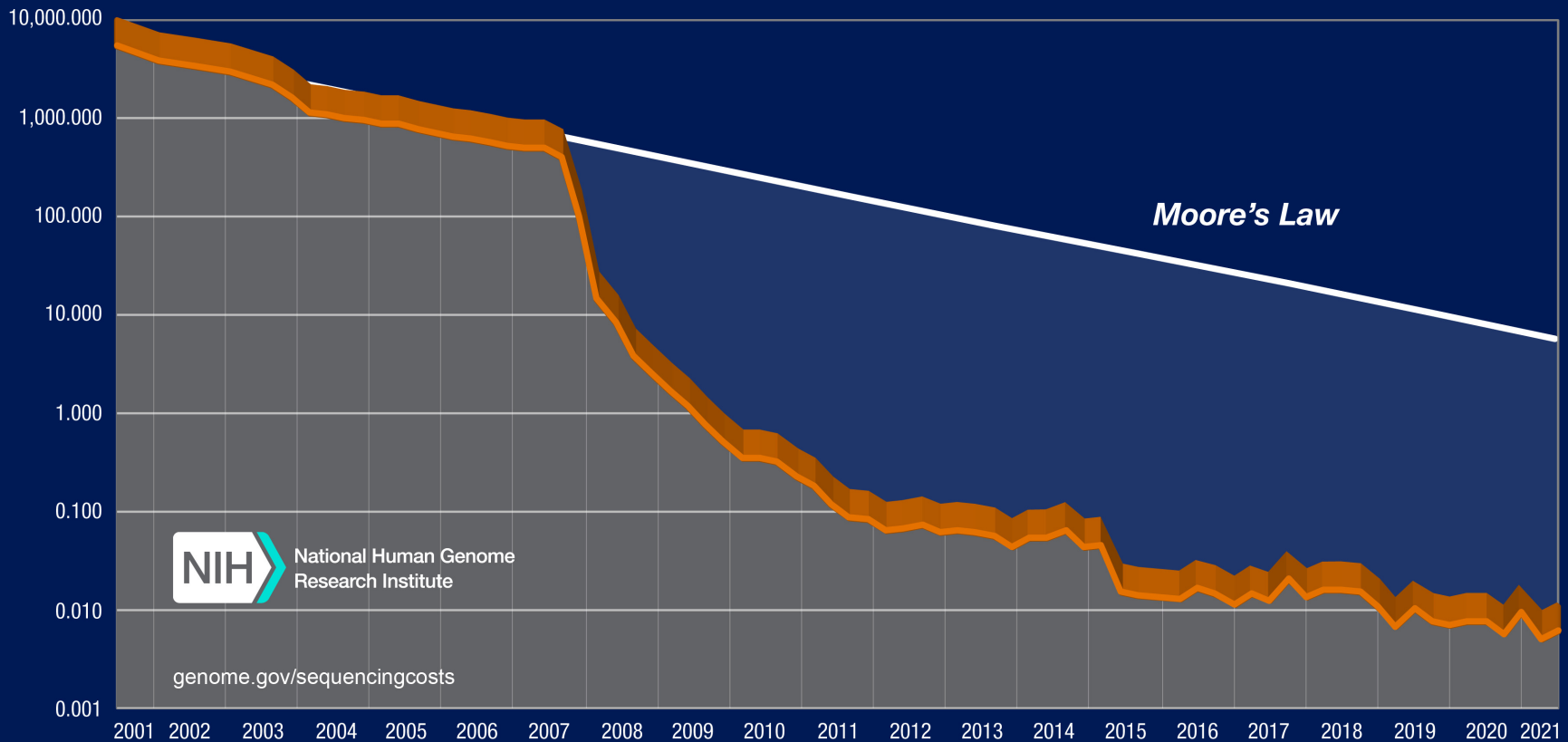
Number of Genomes Sequenced



The Economist

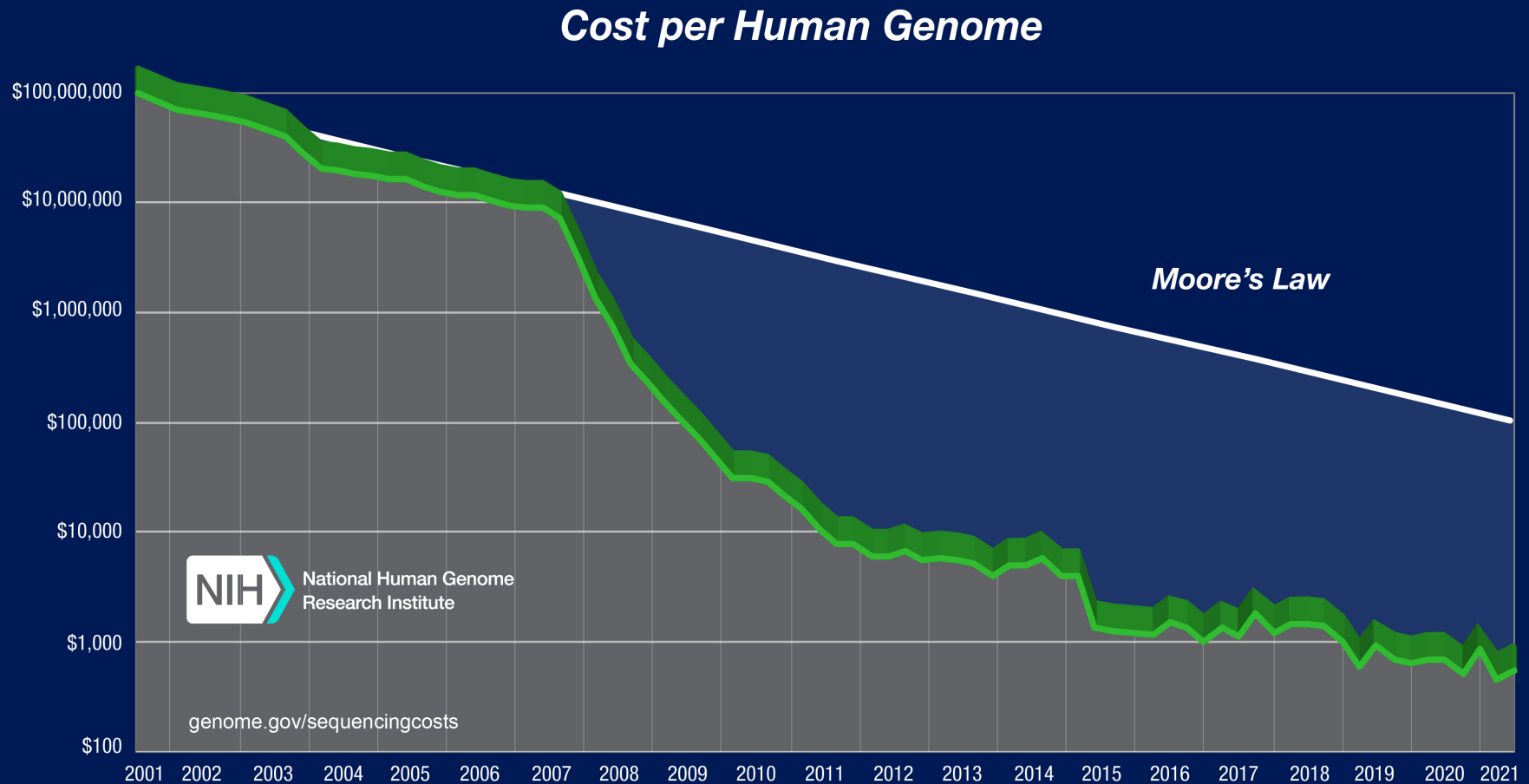
Genome Sequencing Cost Is Reducing

Cost per Raw Megabase of DNA Sequence



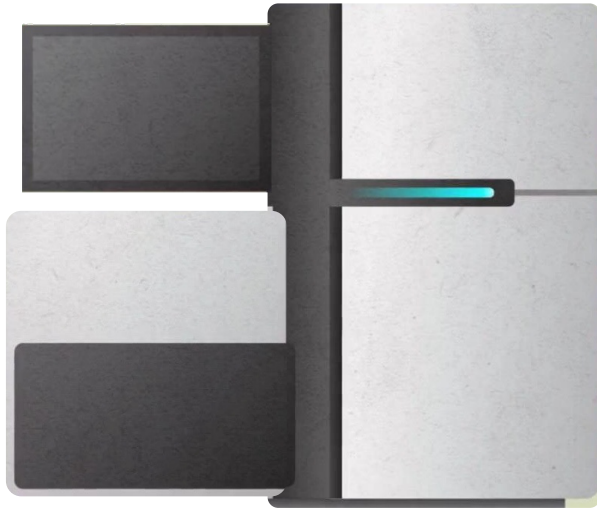
*From NIH (<https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data>)

Genome Sequencing Cost Is Reducing

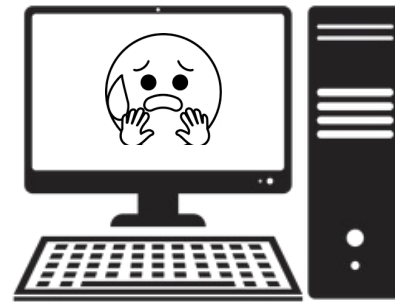
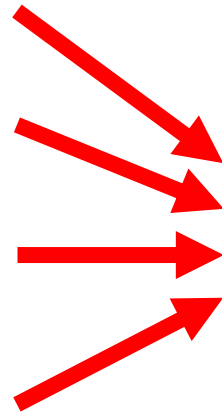


*From NIH (<https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data>)

Problems with (Genome) 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

Newer 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** ▼

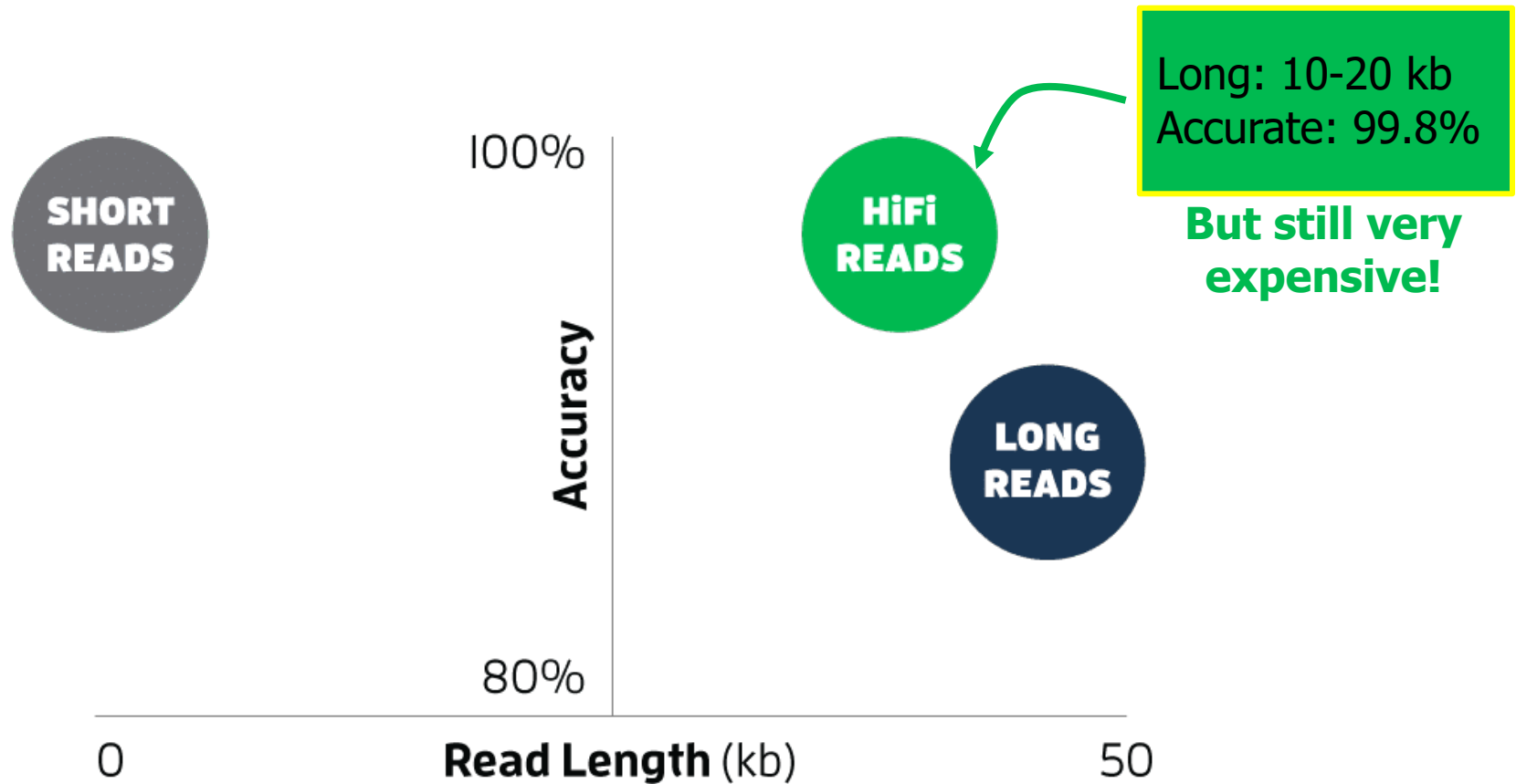


Oxford Nanopore MinION

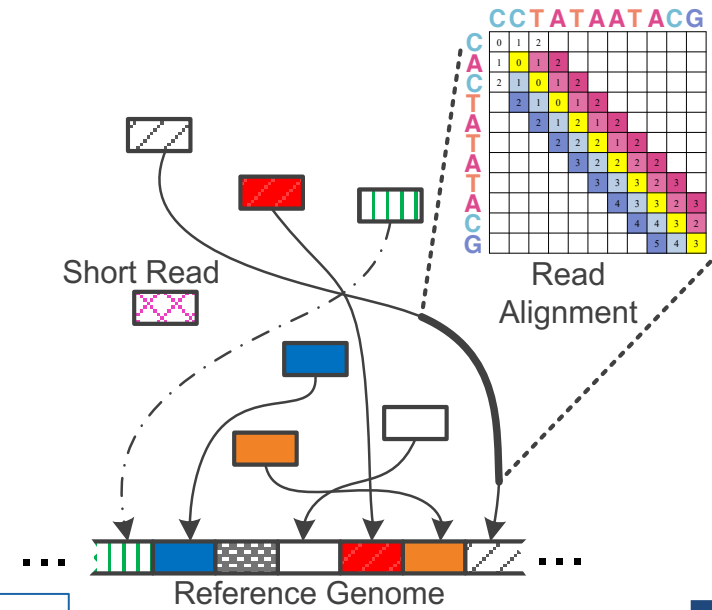
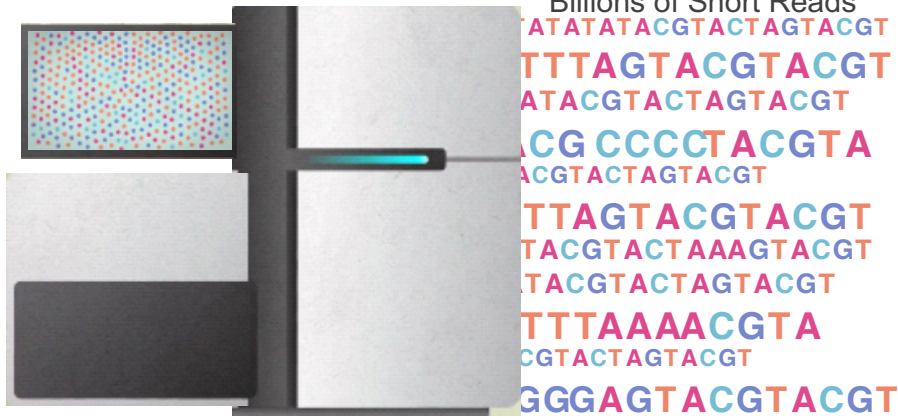
Senol Cali+, “[Nanopore Sequencing Technology and Tools for Genome Assembly: Computational Analysis of the Current State, Bottlenecks and Future Directions](#),” *Briefings in Bioinformatics*, 2018.

[[Open arxiv.org version](#)] [[Slides \(pptx\)](#)] [[pdf](#)] [[Talk Video at AACBB 2019](#)]

Types of Genomic Reads



Wenger+, "[Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome](#)", *Nature Biotechnology*, 2019



1 Sequencing

Genome Analysis

2 Read Mapping

reference: TTTATCGCTTCCATGACGCAG
 read1: ATCGCATCC
 read2: TATCGCATC
 read3: CATCCATGA
 read4: CGCTTCCAT
 read5: CCATGACGC
 read6: TTCCATGAC



3 Variant Calling

4 Scientific Discovery

Read Mapping Techniques in 111 Pages

In-depth analysis of 107 read mappers (1988-2020)

Mohammed Alser, Jeremy Rotman, Dhrithi Deshpande, Kodi Taraszka, Huwenbo Shi, Pelin Icer Baykal, Harry Taegyun Yang, Victor Xue, Sergey Knyazev, Benjamin D. Singer, Brunilda Balliu, David Koslicki, Pavel Skums, Alex Zelikovsky, Can Alkan, Onur Mutlu, Serghei Mangul

"Technology dictates algorithms: Recent developments in read alignment"

Genome Biology, 2021

[[Source code](#)]

Alser et al. *Genome Biology* (2021) 22:249
<https://doi.org/10.1186/s13059-021-02443-7>

Genome Biology

REVIEW

Open Access

Technology dictates algorithms: recent developments in read alignment



Mohammed Alser^{1,2,3†}, Jeremy Rotman^{4†}, Dhrithi Deshpande⁵, Kodi Taraszka⁴, Huwenbo Shi^{6,7}, Pelin Icer Baykal⁸, Harry Taegyun Yang^{4,9}, Victor Xue⁴, Sergey Knyazev⁸, Benjamin D. Singer^{10,11,12}, Brunilda Balliu¹³, David Koslicki^{14,15,16}, Pavel Skums⁸, Alex Zelikovsky^{8,17}, Can Alkan^{2,18}, Onur Mutlu^{1,2,3†} and Serghei Mangul^{5*†} 

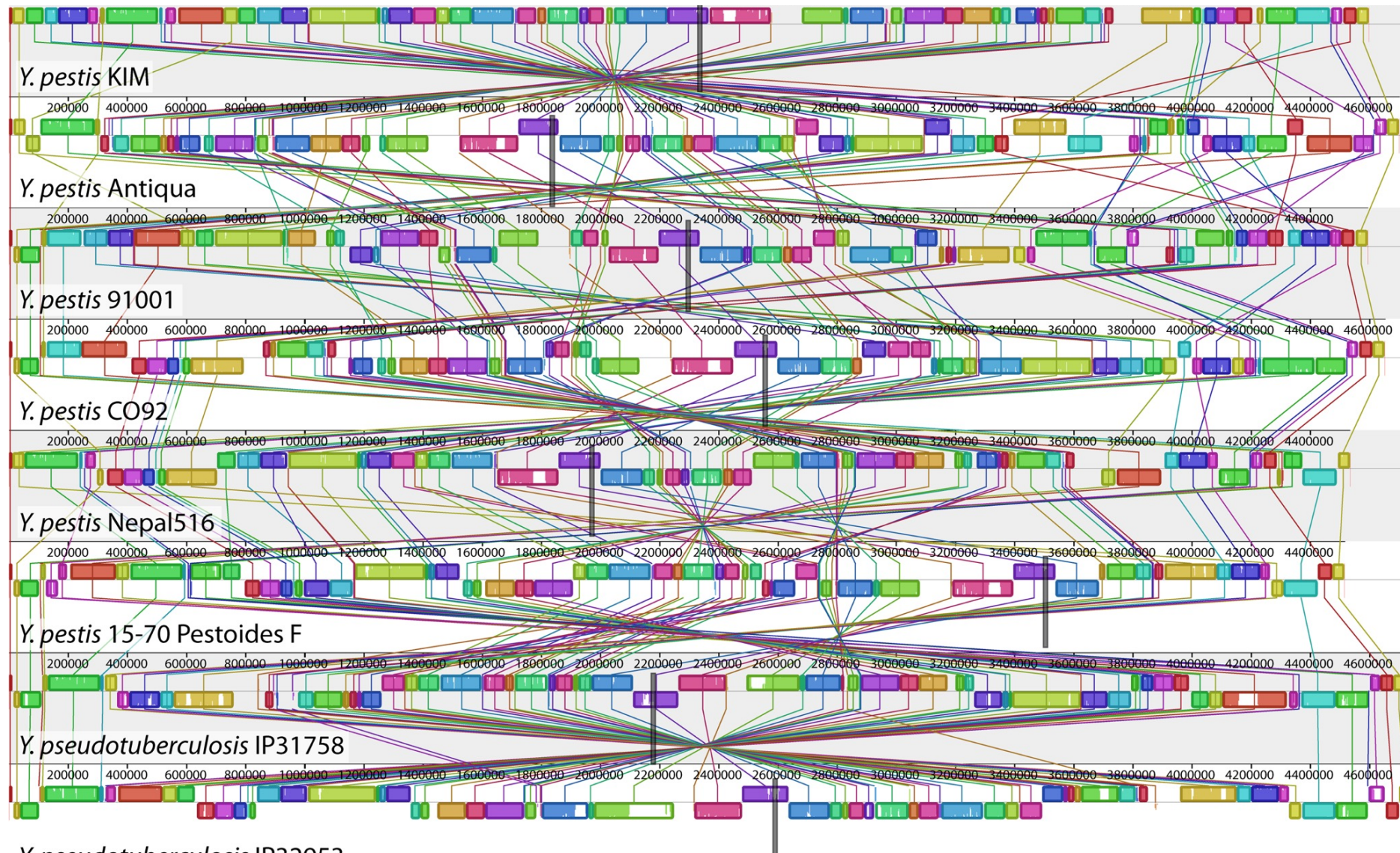
Why Do We Care?

Multiple sequence alignment

PHDHtm			-----MMMMMMMMMMMMMMMMMMMM-----	
16082665	<i>T acid</i>	10	----MASDRKSEGFQSGAGLIRYFEEEEIKGPALDPKLVVYMGIAVAIIVEIAKIFWFP---	(55)
13541150	<i>T volc</i>	10	----MASDRKSEGFQSGAGLIRYFEEEEIKGPALDPKLVVYIGIAVAIMVELAKIFWFP---	(55)
RFAC01077	<i>F acid</i>	13	-MTSMAKDNQNFQSGAGLIRYFNEEEIKGPAIDPKLIITYIGIAMGVIVELAKVFWFPV---	(58)
15791336	<i>H NRC1</i>	10	----MSSGQNSGGLMSSSAGLVRYFDSEDSNALQIDPRSVVAVGAFFGLVLVLLAQFFA----	(53)
RAG22196	<i>A fulg</i>	14	MAKAPKGKAKTPPLMSSSAGIMRYFEE-EKTQIKVSPKTI LAAGIVTGVLI IILNAYYGLWP-	(68)
RPO01000	<i>P abys</i>	9	-----MAKEKTTLPPTGAGLMRFFDE-DTRAIKITPKGAVALTLILIIIFEIILHVVGPRIFG	(56)
RPH01741	<i>P hori</i>	9	-----MAKEKTTLPPTGAGLMRFFDE-DTRAIKITPKGAIALVLILIIIFEIILHVVGPRIFG	(56)
AE000914	<i>M ther</i>	10	----MAKKDKKTLPPSGAGLVRYFEE-ETKGEKLTPEQVVVMSIILAVFCLVLRFSG----	(52)
RMJ09857	<i>M jann</i>	9	-----MSKREESTGLATSAGLIRYMDE-TFSKIRVKPEHVIGVTVAFVIIIEAILTYGRFL---	(53)
15920503	<i>S toko</i>	13	-MPSSKKKKSTVPLASMAGLIRYYEE-ENEKIKISP KLLIIISIMVAGVIVASILIPPP--	(58)
AE006662	<i>S solf</i>	11	-MPSSKKKKSTVPM SMAGLIRYYEE-ENEKVKISP KIVIGASLALTIIIVIVITKLF-----	(55)
RPK02491	<i>P aero</i>	12	--MARRRKYEGLNPFVAAGLIKFS EEGELEKIKLTPRAAVVISLAIIGLLIAINLLLPL--	(58)
RAP00437	<i>A pern</i>	13	-MSVRRRRRERRATPVTAAGLLSFYEE-YEGKIKISPTIVVGAAILVSAVVAABIFLPAVP-	(59)
5803165	<i>H sapi</i>	49	-----SAGTGGMWRFYTE-DSPGLKVGVPVFLVMSLLFIASVFMLHIWGTKYTRS	(96)
13324684	<i>M musc</i>	49	-----SAGTGGMWRFYTE-DSPGLKVGVPVFLVMSLLFIAAVFMLHIWGTKYTRS	(96)
6002114	<i>D mela</i>	53	-----GAGTGGMWRFYTD-DSPGIKVGVPVFLVMSLLFIASVFMLHIWGTKYNRS	(100)
14574310	<i>C eleg</i>	32	-----GGNNGGLWRFYTE-DSTGLKIGVPVFLVMSLVFIASVFVLHIWGTKFTRS	(81)
10697176	<i>Y lipo</i>	41	-----GGSSSTMLKLYTD-ESQGLKVDPVVVMVLSLGFIFSVVALHLLAKVSTK	(91)
6320857	<i>S cere</i>	40	-----GGSSSSILKLYTD-EANGFRVDSLVLFLSVGFIFSVIALHLLTKFTHI	(88)
6320932	<i>S cere</i>	33	-----TNSNNSILKIYSD-EATGLRVDPLVLFLAVGFIFSVVALHVISKVAGK	(82)

Example Question: If I give you a bunch of sequences, tell me where they are the same and where they are different.

Genome Sequence Alignment: Example



Source: By Aaron E. Darling, István Miklós, Mark A. Ragan - Figure 1 from Darling AE, Miklós I, Ragan MA (2008).

"Dynamics of Genome Rearrangement in Bacterial Populations". PLOS Genetics. DOI:10.1371/journal.pgen.1000128., CC BY 2.5, <https://commons.wikimedia.org/w/index.php?curid=30550950>

The Genetic Similarity Between Species



Human ~ Human
99.9%



Human ~ Chimpanzee
96%



Human ~ Cat
90%



Human ~ Cow
80%



Human ~ Banana
50-60%

Finding Variations Associated with Traits

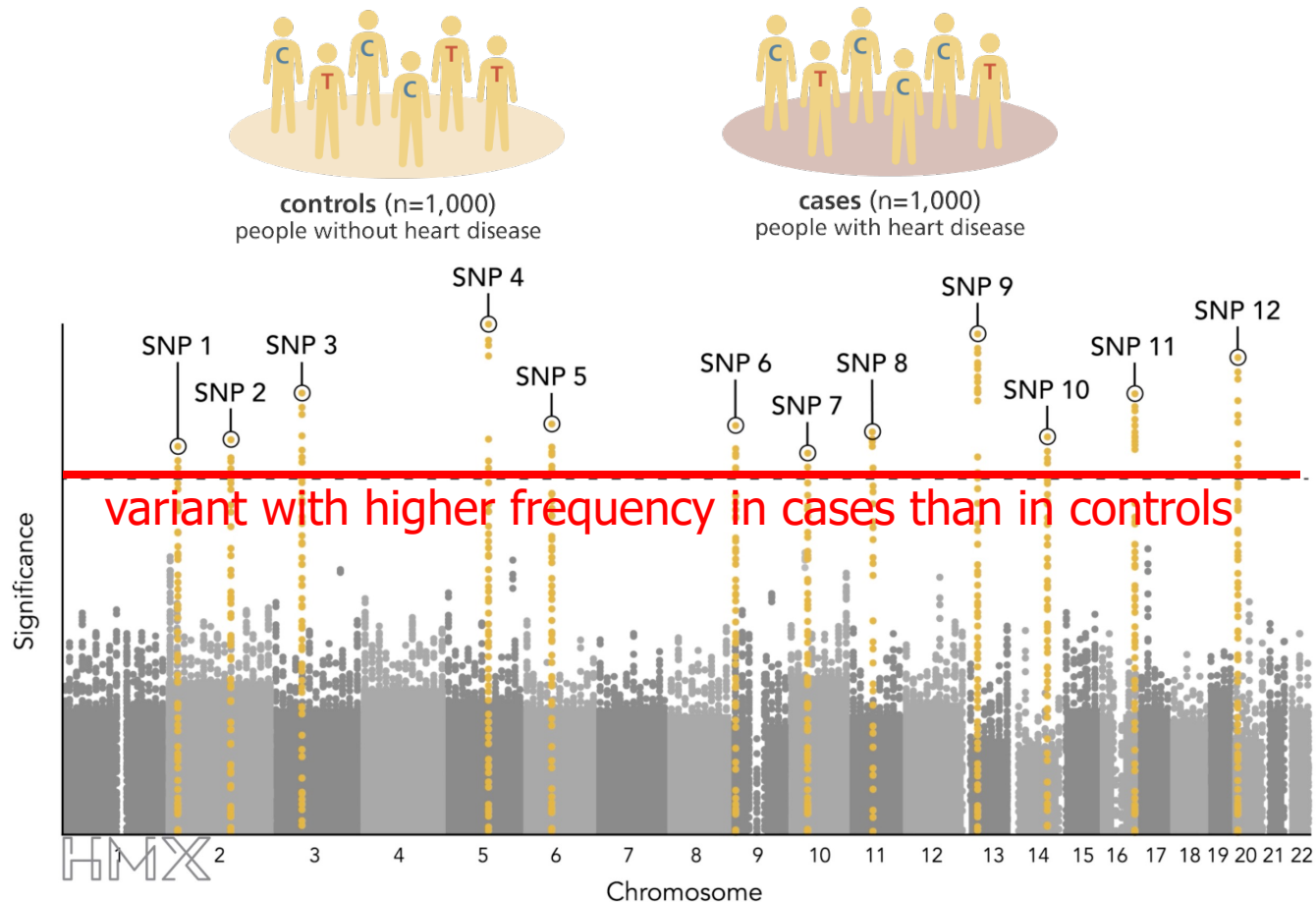
	SNP1	SNP2	Blood Pressure
Individual #1	...ACATG C CGACATTTCATA G GCC...		180
Individual #2	...ACATG C CGACATTTCATA A GCC...		175
Individual #3	...ACATG C CGACATTTCATA G GCC...		170
Individual #4	...ACATG C CGACATTTCATA A GCC...		165
Individual #5	...ACATG C CGACATTTCATA G GCC...		160
Individual #6	...ACATG C CGACATTTCATA G GCC...		145
Individual #7	...ACATG C CGACATTTCATA A GCC...		140
Individual #8	...ACATG C CGACATTTCATA A GCC...		130
Individual #9	...ACATG T CGACATTTCATA G GCC...		120
Individual #10	...ACATG T CGACATTTCATA A GCC...		120
Individual #11	...ACATG T CGACATTTCATA G GCC...		115
Individual #12	...ACATG T CGACATTTCATA A GCC...		110
Individual #13	...ACATG T CGACATTTCATA G GCC...		110
Individual #14	...ACATG T CGACATTTCATA A GCC...		110
Individual #15	...ACATG T CGACATTTCATA G GCC...		105
Individual #16	...ACATG T CGACATTTCATA A GCC...		100



SNP: single nucleotide polymorphism

Genome-Wide Association Studies (GWAS)

- Enables detection of genetic variants associated with phenotypes using two groups of people.



SNPs and Personalized Medicine

openSNP

Q Search

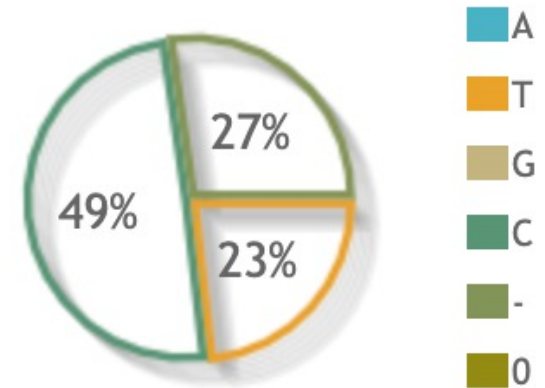
☰

SNP rs12979860

Basic Information

Name	rs12979860
Chromosome	19
Position	39248147
Weight of evidence	926

Allele Frequency



Links to SNPedia

Title	Summary
rs12979860 T/T	~20-25% of such hepatitis c patients respond to treatment
rs12979860 C/C	~80% of such hepatitis c patients respond to treatment
rs12979860 C/T	~20-40% of such hepatitis c patients respond to treatment

Much Larger Structural Variations



AUTISM

Weiss, *N Eng J Med* 2008
Deletion of 593 kb



SCHIZOPHRENIA

McCarthy, *Nat Genet* 2009
Duplication of 593 kb



OBESITY

Walters, *Nature* 2010
Deletion of 593 kb

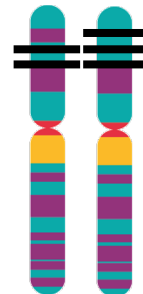


UNDERWEIGHT

Jacquemont, *Nature* 2011
Duplication of 593 kb



Deletion in the short arm
of chromosome 16 (16p11.2)



Duplication in the short arm
of chromosome 16 (16p11.2)

Personalized Medicine for Critically Ill Infants

- rWGS can be performed in 2-day (costly) or 5-day time to interpretation.
- Diagnostic rWGS for infants
 - Avoids morbidity
 - Reduces hospital stay length by 6%-69%
 - Reduces inpatient cost by \$800,000-\$2,000,000.

Article | [Open Access](#) | Published: 04 April 2018

Rapid whole-genome sequencing decreases infant morbidity and cost of hospitalization

Lauge Farnaes, Amber Hildreth, Nathaly M. Sweeney, Michelle M. Clark, S. Chowdhury, Shareef Nahas, Julie A. Cakici, Wendy Benson, Robert H. Kaplan, Richard Kronick, Matthew N. Bainbridge, Jennifer Friedman, Jeffrey J. Gold, Ding, Narayanan Veeraraghavan, David Dimmock & Stephen F. Kingsmore

npj Genomic Medicine 3, Article number: 10 (2018) | [Cite this article](#)

Article | [Open Access](#) | Published: 05 May 2020

Clinical utility of 24-h rapid trio-exome sequencing for critically ill infants

Huijun Wang, Yanyan Qian, Yulan Lu, Qian Qin, Guoping Lu, Guoqiang Cheng, Ping Zhang, Lin Yang, Bingbing Wu & Wenhao Zhou

npj Genomic Medicine 5, Article number: 20 (2020) | [Cite this article](#)

Recommended Reading

nature reviews genetics

Explore our content ▾

Journal information ▾

nature > nature reviews genetics > review articles > article

Review Article | [Published: 15 November 2019](#)

Structural variation in the sequencing era

[Steve S. Ho](#), [Alexander E. Urban](#) & [Ryan E. Mills](#) 

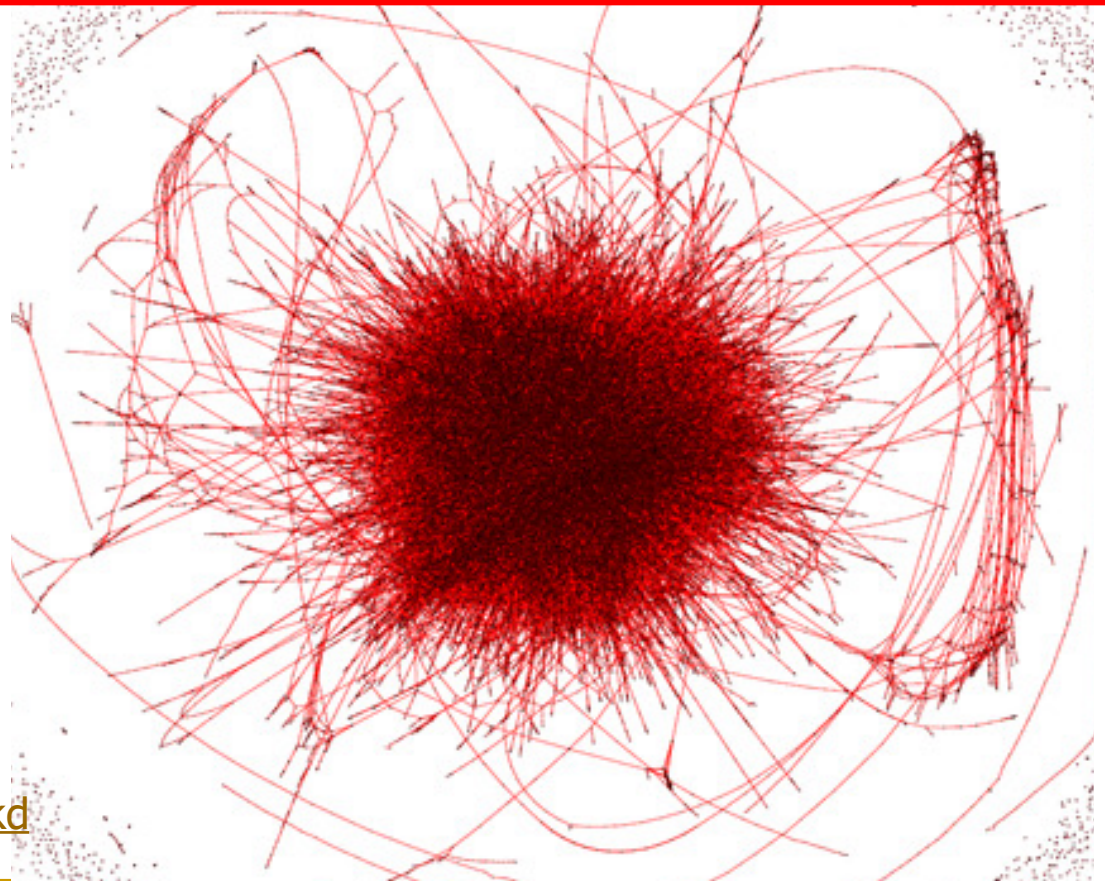
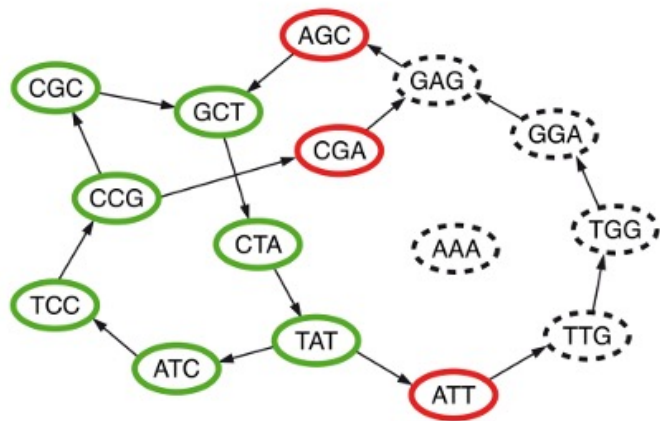
Nature Reviews Genetics **21**, 171–189(2020) | [Cite this article](#)

15k Accesses | **16** Citations | **309** Altmetric | [Metrics](#)

Ho+, "Structural variation in the sequencing era", Nature Reviews Genetics, 2020

Metagenomics, genome assembly, de novo sequencing

Question 2: Given a bunch of short sequences, Can you identify the approximate species cluster for genomically unknown organisms (bacteria)?



uncleaned de Bruijn graph

<http://math.oregonstate.edu/~koslickd/>

Population-Scale Microbiome Profiling



City-Scale Microbiome Profiling

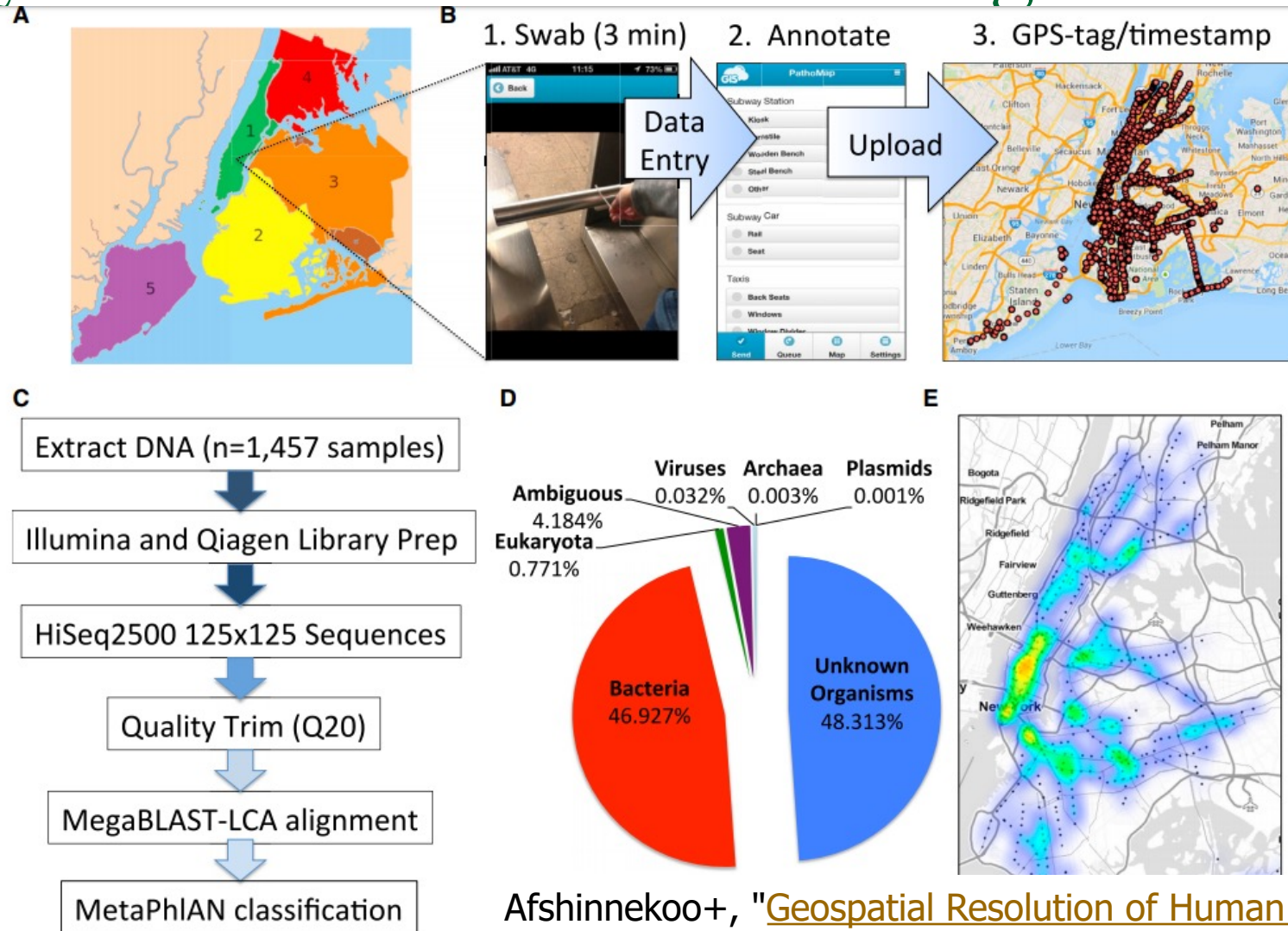


Figure 1. The Metagenome of New York City

(A) The five boroughs of NYC include (1) Manhattan (green)

(B) The collection from the 466 subway stations of NYC across the 24 subway lines involved three main steps: (1) collection with Copan Elution swabs, (2) data entry into the database, and (3) uploading of the data. An image is shown of the current collection database, taken from <http://pathomap.giscloud.com>.

(C) Workflow for sample DNA extraction, library preparation, sequencing, quality trimming of the FASTQ files, and alignment with MegaBLAST and MetaPhlAn to discern taxa present

Afshinnekoo+, "Geospatial Resolution of Human and Bacterial Diversity with City-Scale Metagenomics", Cell Systems, 2015

Global-Scale Microbiome Profiling

Cell Log in Register Su

ARTICLE | ONLINE NOW

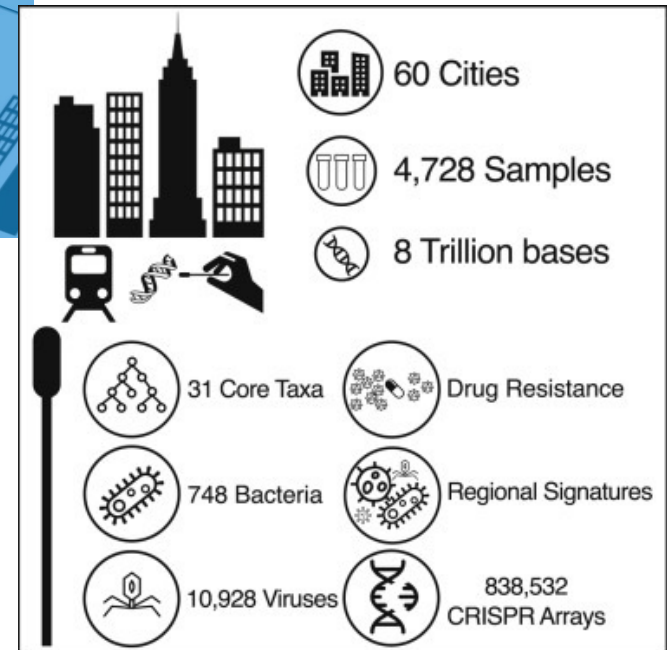
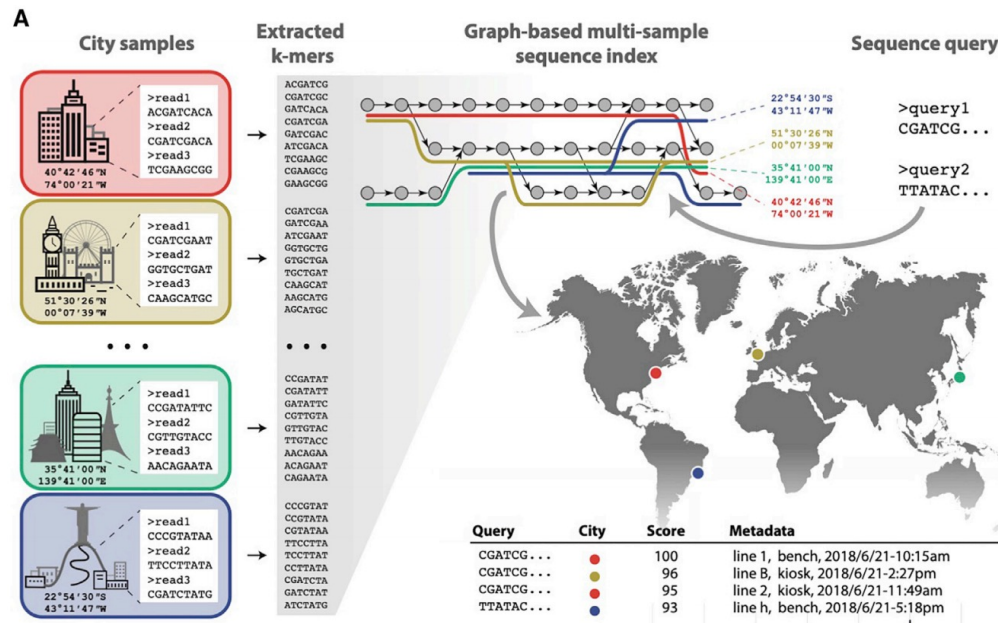
PDF [9 MB] Figures Save

A global metagenomic map of urban microbiomes and antimicrobial resistance

David Danko ⁶⁸ • Daniela Bezdán ⁶⁸ • Evan E. Afshin • ... Sibó Zhu • Christopher E. Mason ⁶⁹

The International MetaSUB Consortium • Show all authors • Show footnotes

Open Access • Published: May 26, 2021 • DOI: <https://doi.org/10.1016/j.cell.2021.05.002>



Danko+, "A global metagenomic map of urban microbiomes and antimicrobial resistance", Cell, 2021

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](#)

Another Question: Example from 2020-...

200 Oxford Nanopore sequencers have left UK for China, to support rapid, near-sample coronavirus sequencing for outbreak surveillance

Fri 31st January 2020

Following extensive support of, and collaboration with, public health professionals in China, Oxford Nanopore has shipped an additional 200 MinION sequencers and related consumables to China. These will be used to support the ongoing surveillance of the current coronavirus outbreak, adding to a large number of the devices already installed in the country.



Each MinION sequencer is approximately the size of a stapler, and can provide rapid sequence information about the coronavirus.



700Kg of Oxford Nanopore sequencers and consumables are on their way for use by Chinese scientists in understanding the current coronavirus outbreak.

Example: Scalable SARS-CoV-2 Testing



THE PREPRINT SERVER FOR HEALTH SCIENCES



Cold
Spring
Harbor
Laboratory



Yale

HOME | ABOUT

Search

[Comments \(1\)](#)

Swab-Seq: A high-throughput platform for massively scaled up SARS-CoV-2 testing

[ID](#) Joshua S. Bloom, [ID](#) Eric M. Jones, [ID](#) Molly Gasperini, [ID](#) Nathan B. Lubock, [ID](#) Laila Sathe, [ID](#) Chetan Munugala, [ID](#) A. Sina Booeshaghi, [ID](#) Oliver F. Brandenburg, [ID](#) Longhua Guo, [ID](#) James Boocock, [ID](#) Scott W. Simpkins, [ID](#) Isabella Lin, [ID](#) Nathan LaPierre, [ID](#) Duke Hong, [ID](#) Yi Zhang, [ID](#) Gabriel Oland, [ID](#) Bianca Judy Choe, [ID](#) Sukantha Chandrasekaran, [ID](#) Evann E. Hilt, [ID](#) Manish J. Butte, [ID](#) Robert Damoiseaux, [ID](#) Aaron R. Cooper, [ID](#) Yi Yin, [ID](#) Lior Pachter, [ID](#) Omai B. Garner, [ID](#) Jonathan Flint, [ID](#) Eleazar Eskin, [ID](#) Chongyuan Luo, [ID](#) Sriram Kosuri, [ID](#) Leonid Kruglyak, [ID](#) Valerie A. Arboleda

doi: <https://doi.org/10.1101/2020.08.04.20167874>

Bloom+, "[Swab-Seq: A high-throughput platform for massively scaled up SARS-CoV-2 testing](#)", *medRxiv*, 2020

Example: Rapid Surveillance of Ebola Outbreak

Figure 1: Deployment of the portable genome surveillance system in Guinea.



Quick+, "Real-time, portable genome sequencing for Ebola surveillance", *Nature*, 2016

We Need Faster & Scalable Genome Analysis



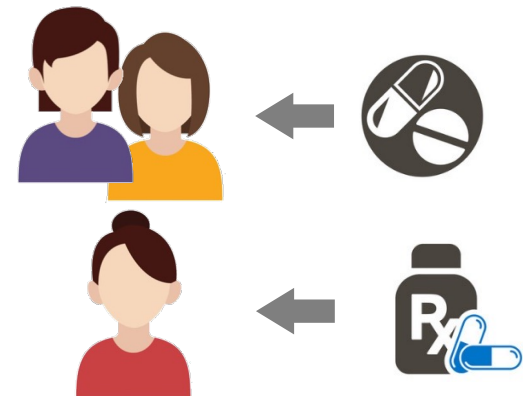
Understanding **genetic variations**,
species, **evolution**, ...



Predicting the **presence** and **relative abundance** of **microbes** in a sample



Rapid surveillance of **disease outbreaks**



Developing **personalized medicine**

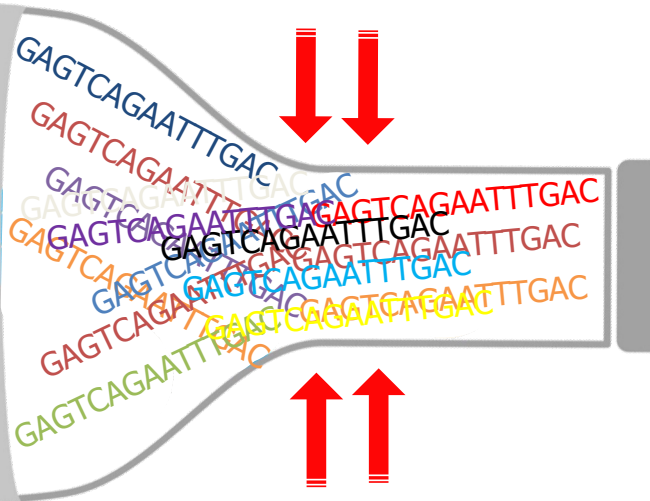
One Problem



The diagram illustrates the process of read alignment. At the bottom, a **Reference Genome** is represented as a horizontal bar with segments of different colors and patterns: green with vertical stripes, solid blue, black with a dot pattern, solid white, red with diagonal stripes, solid orange, and white with a diagonal line pattern. Above the reference genome, several **Short Reads** are shown as colored boxes with patterns: a white box with a diagonal line, a solid red box, a green box with vertical stripes, a blue box, an orange box, and a pink box with a cross-hatch pattern. Arrows indicate the alignment of these reads to specific segments of the reference genome. A thick black arrow points from the top right towards the reference genome, labeled **Read Alignment**. An inset in the top right corner shows a **Read Alignment** matrix for the sequence **CCTATAATACG**. The matrix has columns labeled C, C, T, A, T, A, A, T, A, C, G and rows labeled 0, 1, 2, 1, 0, 1, 2, 2, 1, 0, 1, 2, 2, 1, 2, 3, 2, 2, 3, 2, 3, 4, 3, 3, 2, 3, 4, 4, 3, 2, 5, 4, 3. A path is highlighted in pink, starting from the top left and moving through the matrix.

Read Mapping 2

300 M
bases/min



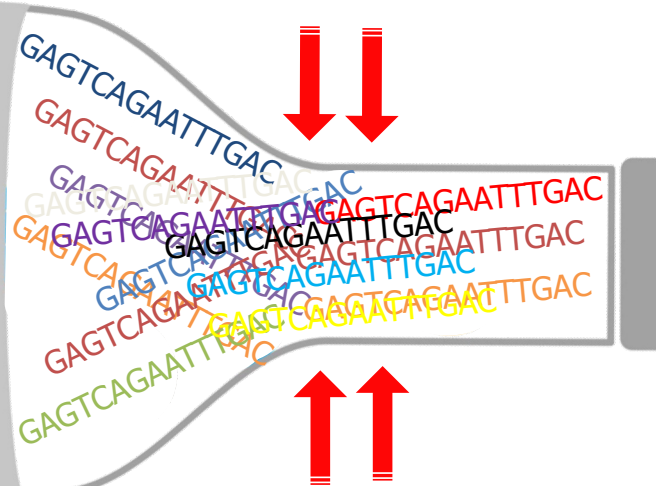
2 M

bases/min
(0.6%)

The Read Mapping Bottleneck

300 Million
bases/minute

Read Sequencing **



2 Million
bases/minute

Read Mapping *

150x slower

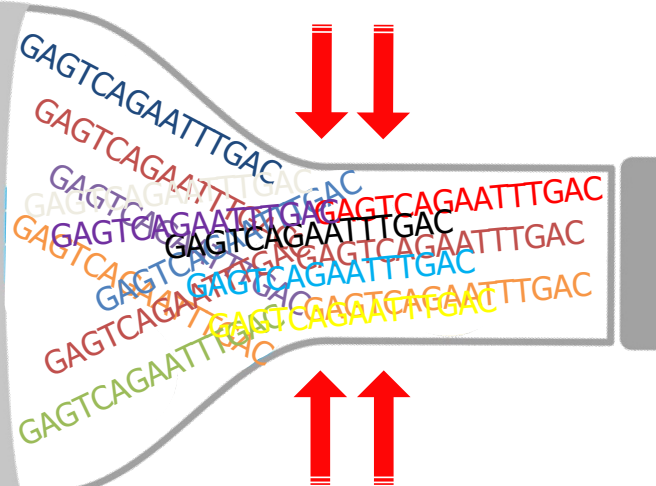
* BWA-MEM

** HiSeqX10, MinION

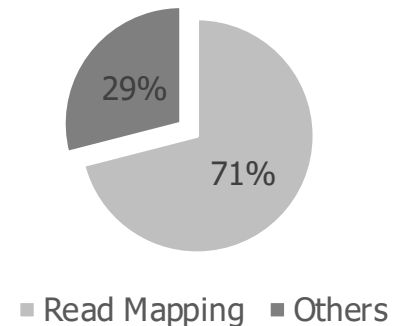
The Read Mapping Bottleneck

48 Human whole
genomes
at 30× coverage
in about 2 days

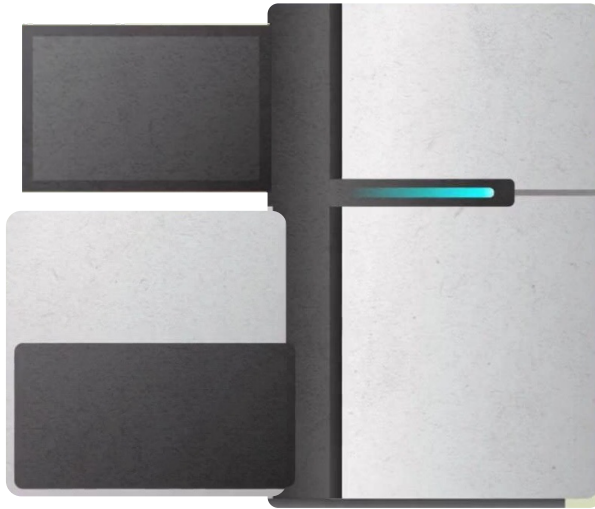
Illumina NovaSeq 6000



1 Human
genome
32 CPU hours
on a 48-core processor

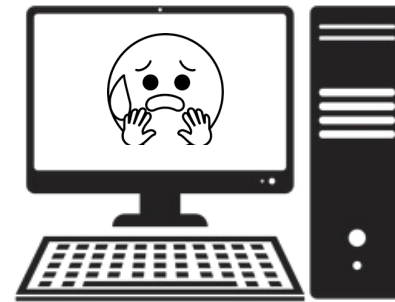
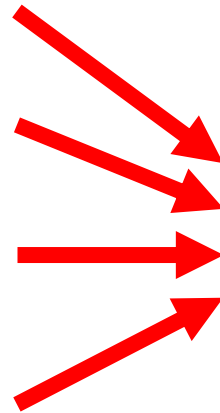


Problems with (Genome) Analysis Today



Special-Purpose Machine
for **Data Generation**

FAST



General-Purpose Machine
for **Data Analysis**

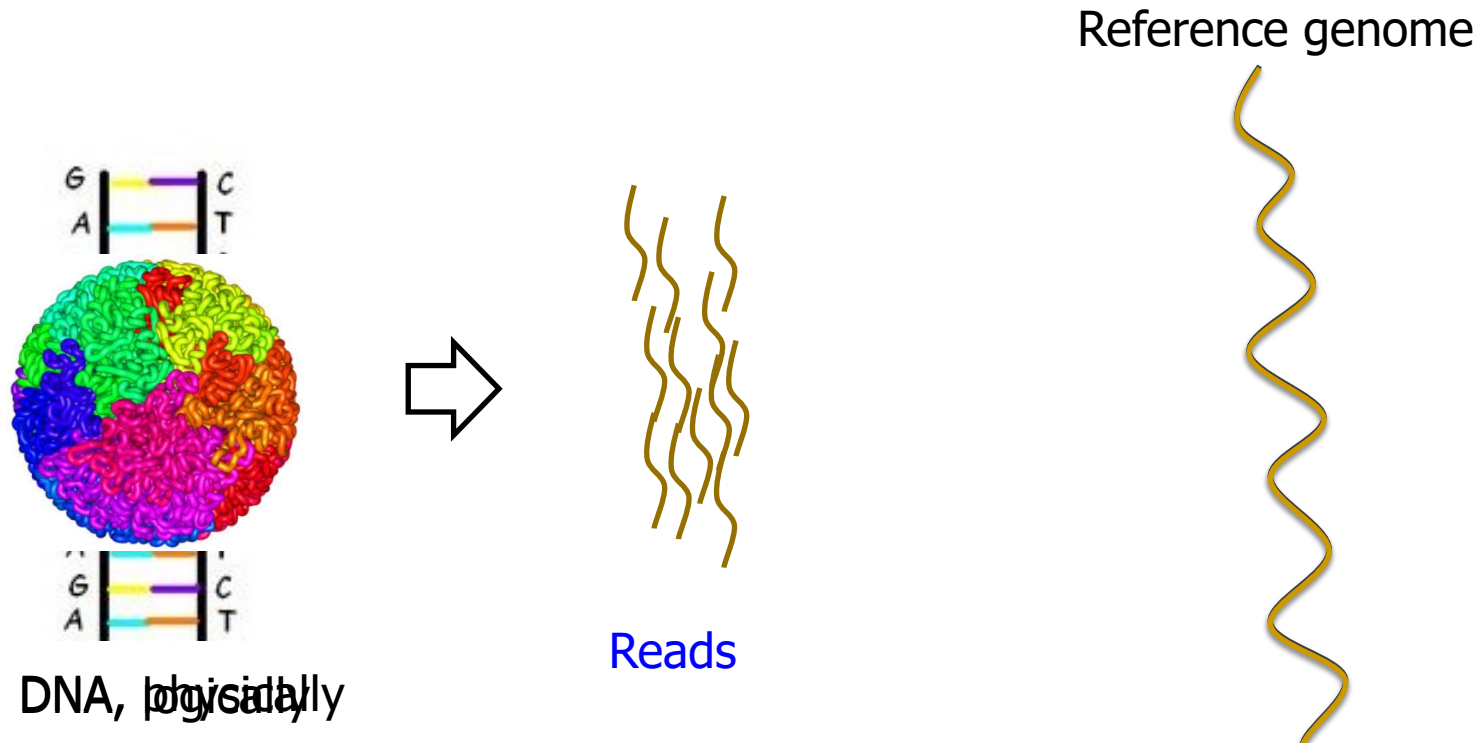
SLOW

Slow and inefficient processing capability
Large amounts of data movement

**Need to construct
the entire genome
from many sequenced reads**

Read Mapping

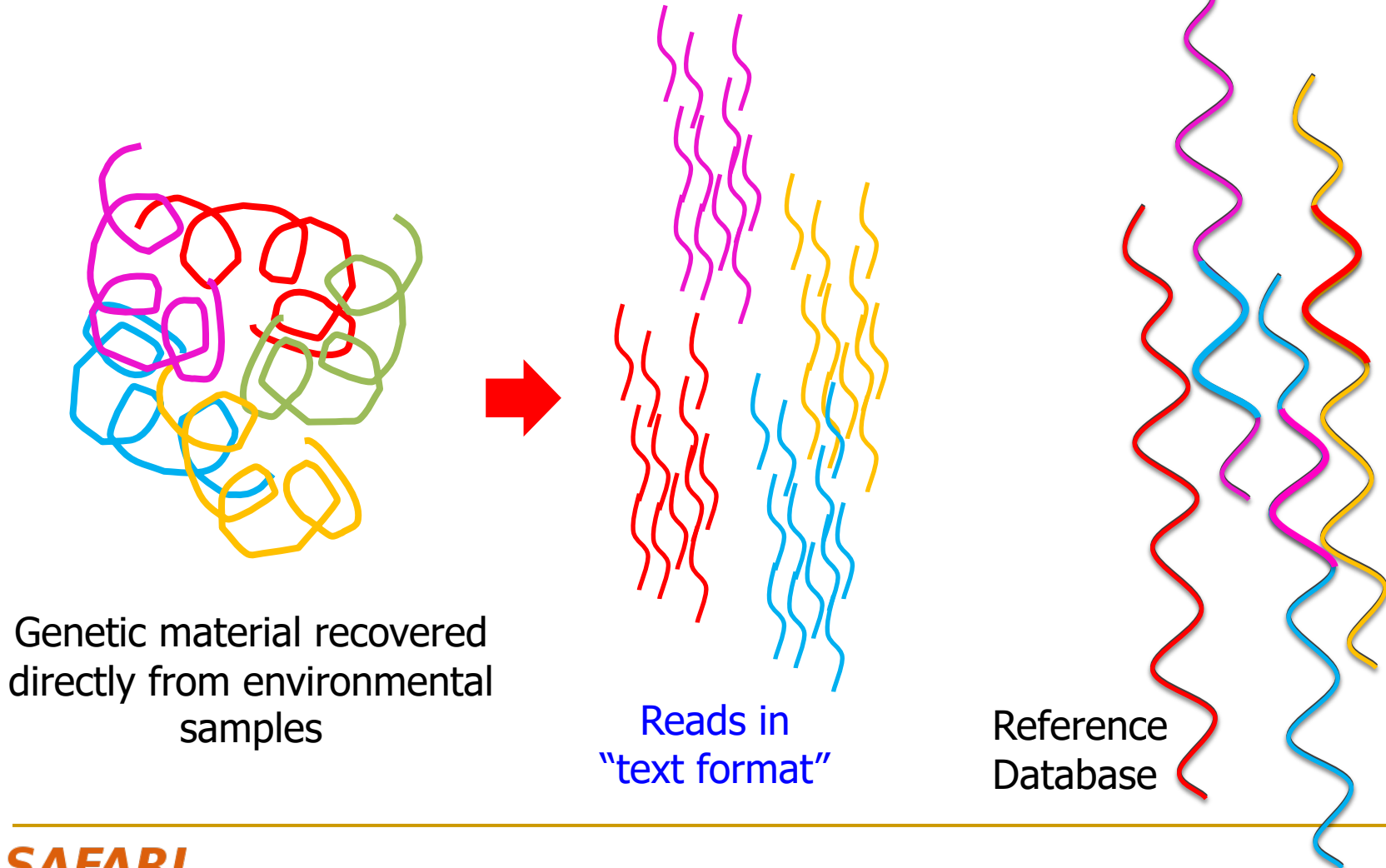
- Map many short DNA fragments (**reads**) to a known reference genome with some differences allowed



Mapping short reads to reference genome is challenging (billions of 50-300 base pair reads)

Read Mapping for Metagenomic Analysis

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



Matching Each Read to Reference Genome

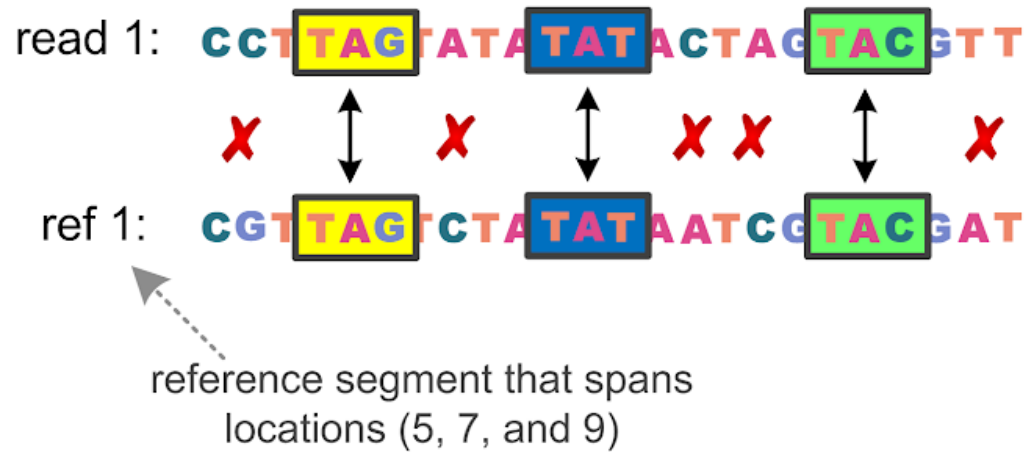
Reference Genome .FASTA file:

```
>NG_008679.1:5001-38170 Homo sapiens paired box 6 (PAX6)
ACCTT[redacted]TCATTGACATTTAAACTCTGGGGCAGGT[redacted]GAACGCGGCTGTCAGATCT
GCCACTTCCCCTGCCGAGCGGCGGTGAGAAGTGTGGGAACCGGCGCTGCCAGGCTCACCTGCCTCCCCGC
CCTCCGCTCCCAGGTAACCGCC[redacted]CCCCGGCCCCGGCTCGGGGCCCGCGGGGCCTCTCCGCTG
CCAGCGACTGCTGTCCCCAAATCAAAGCCCCGCCCAAGTGGCCCCGGGGCTTGATTTTTGCTTTTAAAAG
GAGGCATACAAAGATGGAAGCGAGTTACTGAGGGAGGGATAGGAAGGGGGGTGGAGGAGGGACTTGTCCTT
TCCCGAGTGT[redacted]CAAAAGTAGCA[redacted]CTCCTAA[redacted]TCCAGTCCGGCCCT
GAGCTGGGAGTAGGGGGCGGGAGTCTGCTGCTGCTGTCTGCTAAAGCCACTCGCGACCGCGAAAAATGCA
GGAGGTGGGGACGCACTTTGCATCCAGACCTCCTCTGCATCGCAGTTC[redacted]CGCTTGGGAAAG
TCCGTACCCGCGCCT[redacted]AAAGACACCCTGCCGCGGGTCGGGCGAGGTGCAGCAGAAGTTTCCC
GCGGTTGCAAAGTGCAGATGGCTGGACCGCAACAAAGTCTAGAGATGGGGTTCGTTTCTCAGAAAGACGC
```

Sequenced Reads .FASTQ file:

```
@HWI-EAS209_0006_FC706VJ:5:58:5894:21141#ATCACG/1
T[redacted]AATAAATCT[redacted]TTAGATN[redacted]NNNNNNNNNTAG
+HWI-EAS209_0006_FC706VJ:5:58:5894:21141#ATCACG/1
efcfffffcfeefffcfffffdddf`feed]`_]_Ba_^__[YBBBBBBBBBBRTT
```


Base-by-Base Comparison



Read Alignment/Verification

- **Edit distance** is defined as the minimum number of edits (i.e., insertions, deletions, or substitutions) needed to make the read exactly match (i.e., align with) the reference segment

NETHERLANDS x SWITZERLAND

N	E	-	T	H	E	R	L	A	N	D	S
S	W	I	T	Z	E	R	L	A	N	D	-

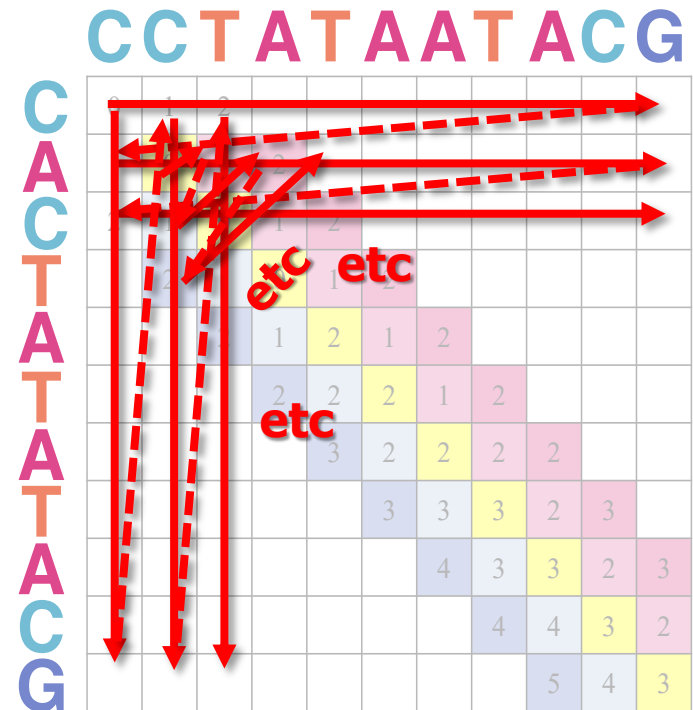
match
deletion
insertion
mismatch

Challenges in Read Mapping

- Need to find many mappings of each read
 - A short read may map to many locations, especially with High-Throughput DNA Sequencing technologies
 - How can we find all mappings efficiently?
- Need to tolerate small variances/errors in each read
 - Each individual is different: Subject's DNA may slightly differ from the reference (Mismatches, insertions, deletions)
 - How can we efficiently map each read with up to e errors present?
- Need to map each read very fast (i.e., performance is important)
 - Human DNA is 3.2 billion base pairs long → Millions to billions of reads (State-of-the-art mappers take weeks to map a human's DNA)
 - How can we design a much higher performance read mapper?

Why Is Read Alignment Slow?

- **Quadratic-time** dynamic-programming algorithm(s)
- **Data dependencies** limit the computation parallelism
- **Entire matrix** computed even though strings may be dissimilar



Read Alignment

Example: Dynamic Programming Table

NETHERLANDS x SWITZERLAND

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

immediate left,
upper left,
upper entries of its own



Example: Dynamic Programming Table

NETHERLANDS x SWITZERLAND

		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

- Matrix-filling is $O(mn)$ time and space.
- Backtrace is $O(m + n)$ time.

Example: Dynamic Programming

- **Quadratic-time** dynamic-programming algorithm

WHY?!

Enumerate all possible prefixes

NETHERLANDS x SWITZERLAND

NETHERLANDS x S

- **[** NETHERLANDS x SW

- C** NETHERLANDS x SWI

NETERLANDS x SWIT

NETHERLANDS x SWITZ

NETHERLANDS x SWITZE

NETHERLANDS x SWITZER

NETHERLANDS x SWITZERL

- **E** NETHERLANDS x SWITZERLA

- €** NETHERLANDS x SWITZERLAN

- C** NETHERLANDS x SWITZERLAND

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

Computational Cost is Mathematically Proven

arXiv.org > cs > arXiv:1412.0348

Search...

Help | Advanced

Computer Science > Computational Complexity

[Submitted on 1 Dec 2014 (v1), last revised 15 Aug 2017 (this version, v4)]

Edit Distance Cannot Be Computed in Strongly Subquadratic Time (unless SETH is false)

Arturs Backurs, Piotr Indyk

The edit distance (a.k.a. the Levenshtein distance) between two strings is defined as the minimum number of insertions, deletions or substitutions of symbols needed to transform one string into another. The problem of computing the edit distance between two strings is a classical computational task, with a well-known algorithm based on dynamic programming. Unfortunately, all known algorithms for this problem run in nearly quadratic time.

In this paper we provide evidence that the near-quadratic running time bounds known for the problem of computing edit distance might be tight. Specifically, we show that, if the edit distance can be computed in time $O(n^{2-\delta})$ for some constant $\delta > 0$, then the satisfiability of conjunctive normal form formulas with N variables and M clauses can be solved in time $M^{O(1)}2^{(1-\epsilon)N}$ for a constant $\epsilon > 0$. The latter result would violate the Strong Exponential Time Hypothesis, which postulates that such algorithms do not exist.

Read Mapping Techniques in 111 Pages

In-depth analysis of 107 read mappers (1988-2020)

Mohammed Alser, Jeremy Rotman, Dhrithi Deshpande, Kodi Taraszka, Huwenbo Shi, Pelin Icer Baykal, Harry Taegyun Yang, Victor Xue, Sergey Knyazev, Benjamin D. Singer, Brunilda Balliu, David Koslicki, Pavel Skums, Alex Zelikovsky, Can Alkan, Onur Mutlu, Serghei Mangul

["Technology dictates algorithms: Recent developments in read alignment"](#)

Genome Biology, 2021

[\[Source code\]](#)

Alser et al. *Genome Biology* (2021) 22:249
<https://doi.org/10.1186/s13059-021-02443-7>

Genome Biology

REVIEW

Open Access

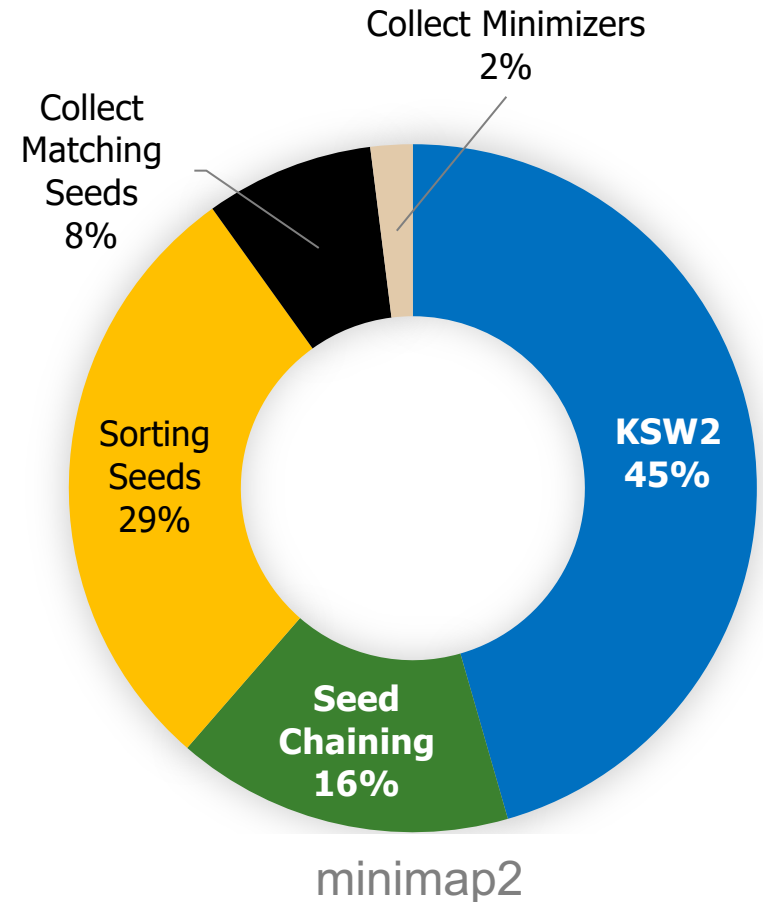
Technology dictates algorithms: recent developments in read alignment



Mohammed Alser^{1,2,3†}, Jeremy Rotman^{4†}, Dhrithi Deshpande⁵, Kodi Taraszka⁴, Huwenbo Shi^{6,7}, Pelin Icer Baykal⁸, Harry Taegyun Yang^{4,9}, Victor Xue⁴, Sergey Knyazev⁸, Benjamin D. Singer^{10,11,12}, Brunilda Balliu¹³, David Koslicki^{14,15,16}, Pavel Skums⁸, Alex Zelikovsky^{8,17}, Can Alkan^{2,18}, Onur Mutlu^{1,2,3†} and Serghei Mangul^{5*†} 

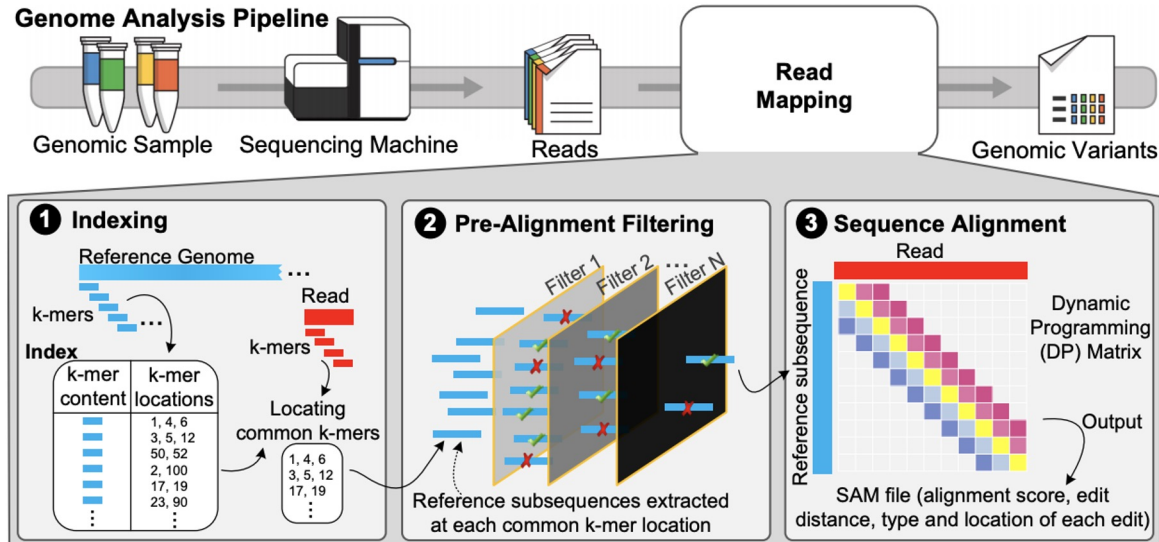
Read Mapping Execution Time (Modern)

>60%
of the read mapper's
execution time is spent
in sequence alignment



ONT FASTQ size: 103MB (151 reads), Mean length: 356,403 bp, std: 173,168 bp, longest length: 817,917 bp

Accelerating Read Mapping



Accelerating Indexing

Reducing the number of seeds

Reducing data movement during indexing

Accelerating Pre-Alignment Filtering

q-gram filtering

Pigeonhole principle

Base counting

Sparse DP

Accelerating Alignment

Accurate alignment accelerators

Heuristic-based alignment accelerators

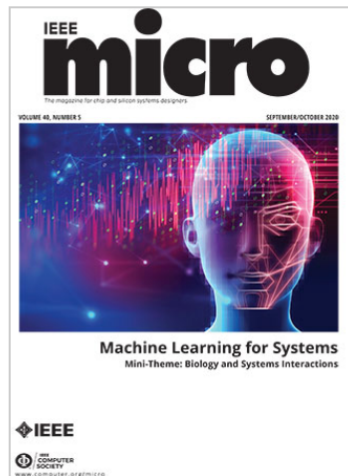
Alser+, "[Accelerating Genome Analysis: A Primer on an Ongoing Journey](#)", IEEE Micro, 2020.

Detailed Analysis of Tackling the Bottleneck

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose,
Can Alkan, Onur Mutlu

[“Accelerating Genome Analysis: A Primer on an Ongoing Journey”](#)

IEEE Micro, August 2020.



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2020.05](#)

IEEE Micro

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Sept.-Oct. 2020, pp. 65-75, vol. 40

DOI Bookmark: [10.1109/MM.2020.3013728](#)

Authors

[Mohammed Alser](#), ETH Zürich

[Zulal Bingöl](#), Bilkent University

[Damla Senol Cali](#), Carnegie Mellon University

[Jeremie Kim](#), ETH Zurich and Carnegie Mellon University

[Saugata Ghose](#), University of Illinois at Urbana-Champaign and Carnegie Mellon University

[Can Alkan](#), Bilkent University

[Onur Mutlu](#), ETH Zurich, Carnegie Mellon University, and Bilkent University

◀ Previous

▶ Next

☰ Table of Contents

📖 Past Issues

Genomics Course (Fall 2022)

Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=bioinformatics

Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=bioinformatics

Youtube Livestream (Fall 2022):

- https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFIQvdxOdizD_EhVAMVQV

Youtube Livestream (Spring 2022):

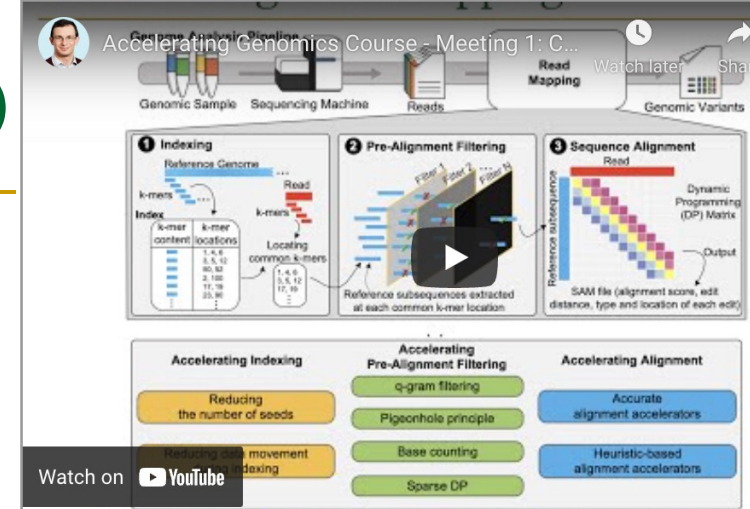
- https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NrPDgOR1yRU_Cxxjw-u18

Project course

- Taken by Bachelor's/Master's students
- Genomics lectures
- Hands-on research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>

SAFARI



Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials
W1	11.3 Fri.	YouTube Live	M1: P&S Accelerating Genomics Course Introduction & Project Proposals (PDF) (PPT)	Required Materials Recommended Materials
W2	18.3 Fri.	YouTube Live	M2: Introduction to Sequencing (PDF) (PPT)	
W3	25.3 Fri.	YouTube Premiere	M3: Read Mapping (PDF) (PPT)	
W4	01.04 Fri.	YouTube Premiere	M4: GateKeeper (PDF) (PPT)	
W5	08.04 Fri.	YouTube Premiere	M5: MAGNET & Shouji (PDF) (PPT)	
W6	15.4 Fri.	YouTube Premiere	M6: SneakySnake (PDF) (PPT)	
W7	29.4 Fri.	YouTube Premiere	M7: GenStore (PDF) (PPT)	
W8	06.05 Fri.	YouTube Premiere	M8: GRIM-Filter (PDF) (PPT)	
W9	13.05 Fri.	YouTube Premiere	M9: Genome Assembly (PDF) (PPT)	
W10	20.05 Fri.	YouTube Live	M10: Genomic Data Sharing Under Differential Privacy (PDF) (PPT)	
W11	10.06 Fri.	YouTube Premiere	M11: Accelerating Genome Sequence Analysis (PDF) (PPT)	

BIO-Arch Workshop at RECOMB 2023

■ April 14, 2023

BIO-Arch: Workshop on Hardware Acceleration of Bioinformatics Workloads

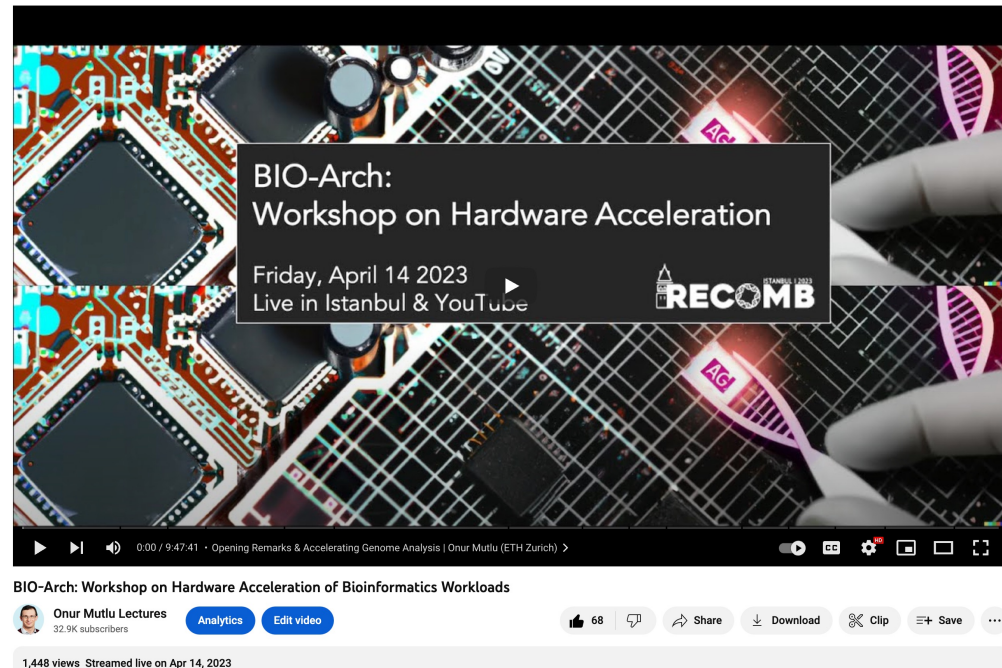
About

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 for new hardware technologies.

Venue

BIO-Arch will be held in [The Social Facilities of Istanbul 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>

<https://safari.ethz.ch/recomb23-arch-workshop/>

Agenda

- The Problem: DNA Read Mapping
 - State-of-the-art Read Mapper Design
- Algorithmic Acceleration
 - Exploiting Structure of the Genome
 - Exploiting SIMD Instructions
- Hardware Acceleration
 - Specialized Architectures
 - Processing in Memory & Storage
- Future Opportunities: New Technologies & Applications

GateKeeper: FPGA-Based Alignment Filtering

- Mohammed Alser, Hasan Hassan, Hongyi Xin, Oguz Ergin, Onur Mutlu, and Can Alkan
"GateKeeper: A New Hardware Architecture for Accelerating Pre-Alignment in DNA Short Read Mapping"
Bioinformatics, [published online, May 31], 2017.
[[Source Code](#)]
[[Online link at Bioinformatics Journal](#)]

GateKeeper: a new hardware architecture for accelerating pre-alignment in DNA short read mapping

Mohammed Alser ✉, Hasan Hassan, Hongyi Xin, Oğuz Ergin, Onur Mutlu ✉, Can Alkan ✉

Bioinformatics, Volume 33, Issue 21, 1 November 2017, Pages 3355–3363,

<https://doi.org/10.1093/bioinformatics/btx342>

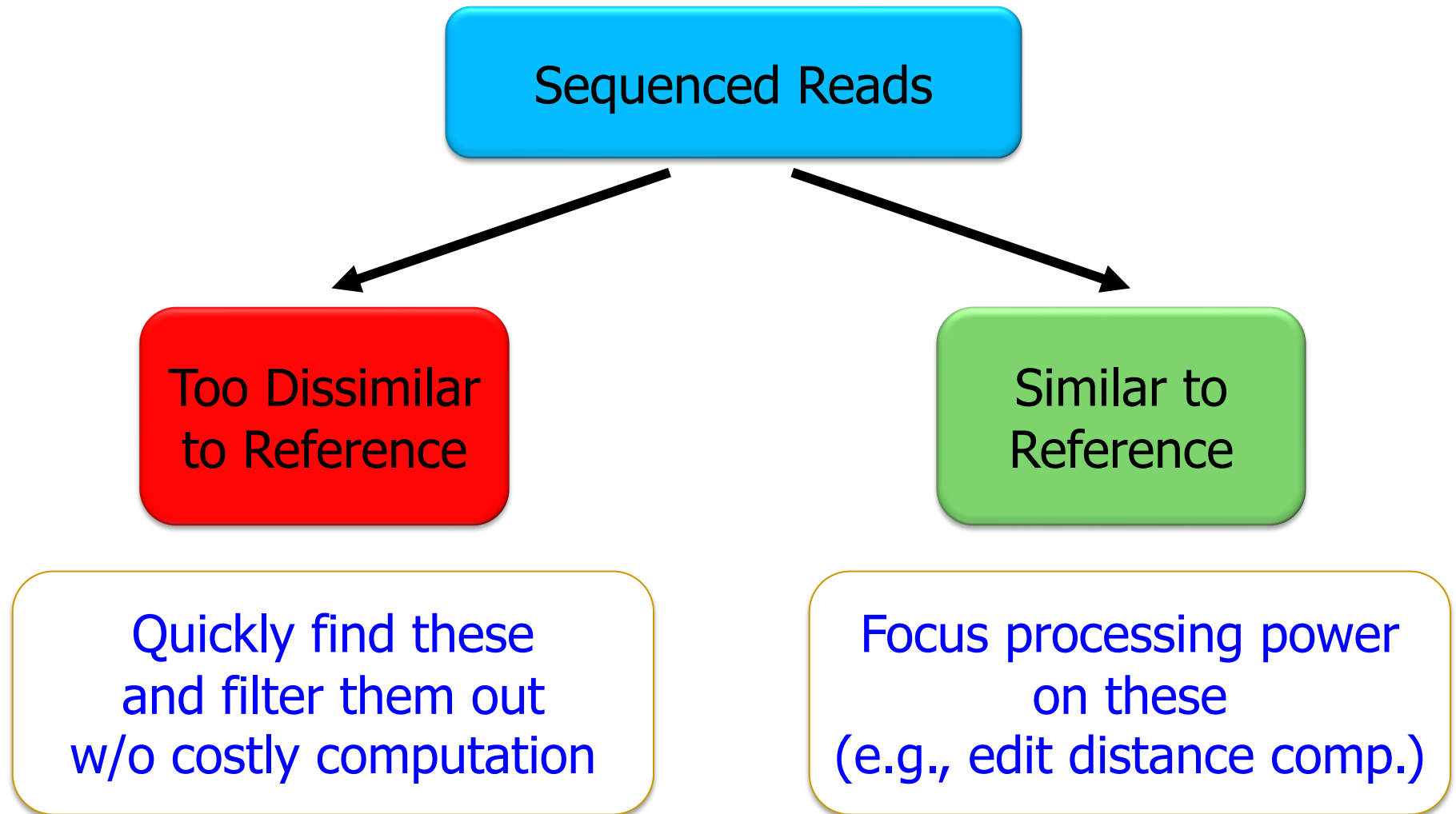
Published: 31 May 2017 **Article history** ▼

Overarching Key Idea

Filter fast before you align

**Minimize costly
edit distance computations**
("approximate string comparisons")

Overarching Key Idea



Our First Filter: Pure Software Approach

- Download the source code and try for yourself
 - [Download link to FastHASH](#)

Xin *et al.* *BMC Genomics* 2013, **14**(Suppl 1):S13
<http://www.biomedcentral.com/1471-2164/14/S1/S13>



PROCEEDINGS

Open Access

Accelerating read mapping with FastHASH

Hongyi Xin¹, Donghyuk Lee¹, Farhad Hormozdiari², Samihan Yedkar¹, Onur Mutlu^{1*}, Can Alkan^{3*}

From The Eleventh Asia Pacific Bioinformatics Conference (APBC 2013)
Vancouver, Canada. 21-24 January 2013

Shifted Hamming Distance: SIMD Acceleration

<https://github.com/CMU-SAFARI/Shifted-Hamming-Distance>

Bioinformatics, 31(10), 2015, 1553–1560

doi: 10.1093/bioinformatics/btu856

Advance Access Publication Date: 10 January 2015

Original Paper

OXFORD

Sequence analysis

Shifted Hamming distance: a fast and accurate SIMD-friendly filter to accelerate alignment verification in read mapping

Hongyi Xin^{1,*}, John Greth², John Emmons², Gennady Pekhimenko¹,
Carl Kingsford³, Can Alkan^{4,*} and Onur Mutlu^{2,*}

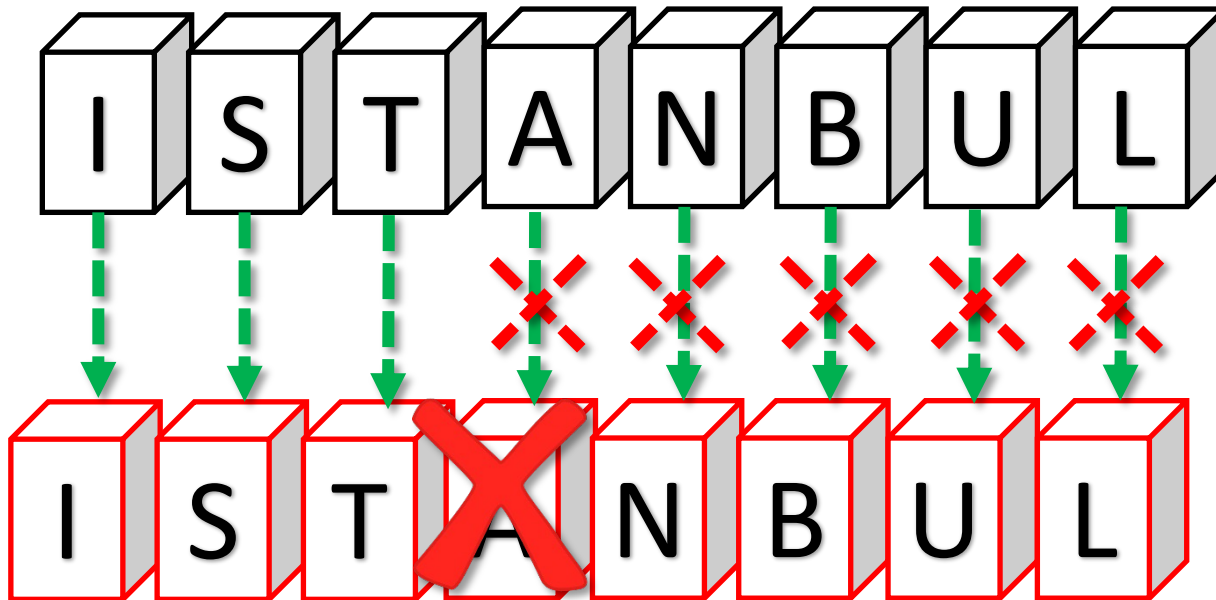
Xin+, "[Shifted Hamming Distance: A Fast and Accurate SIMD-friendly Filter to Accelerate Alignment Verification in Read Mapping](#)", **Bioinformatics 2015.**

Hamming Distance ($\Sigma \oplus$)

3 matches

5 mismatches

Edit = 1 Deletion

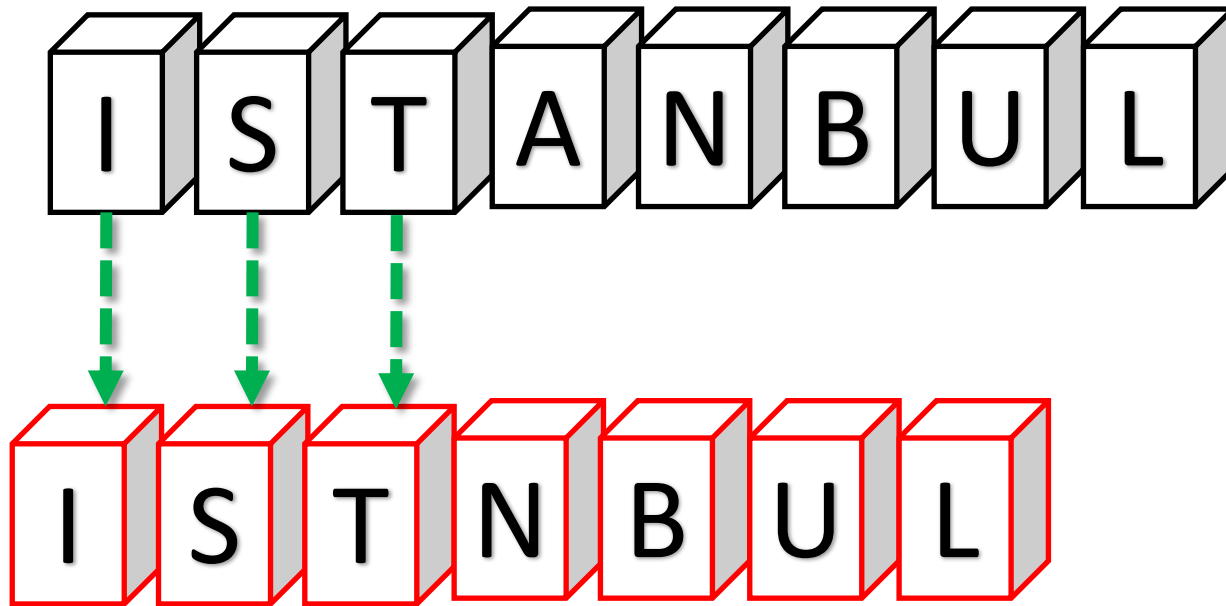


To cancel the effect of a deletion, we need to shift in the *right* direction

Insight: Shifting a String Helps Similarity Search

3 matches

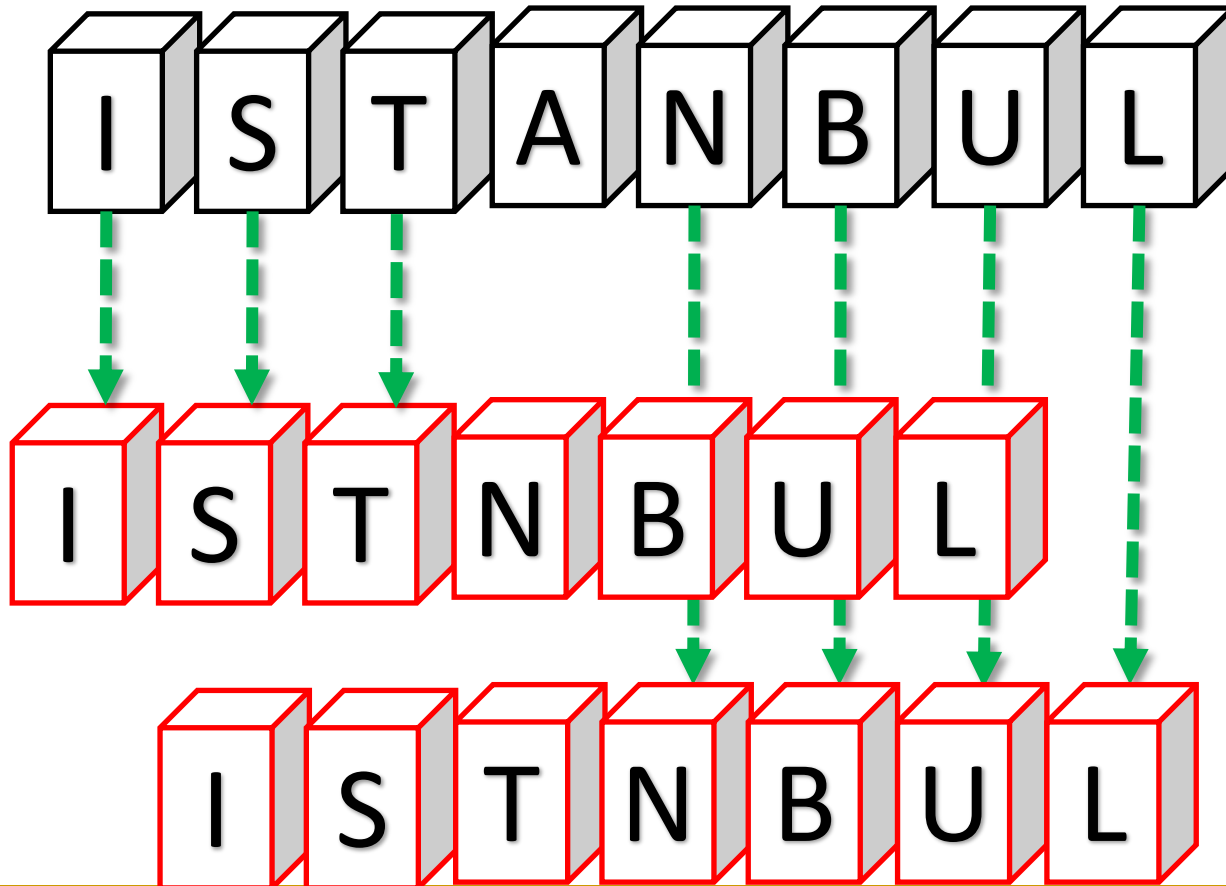
5 mismatches



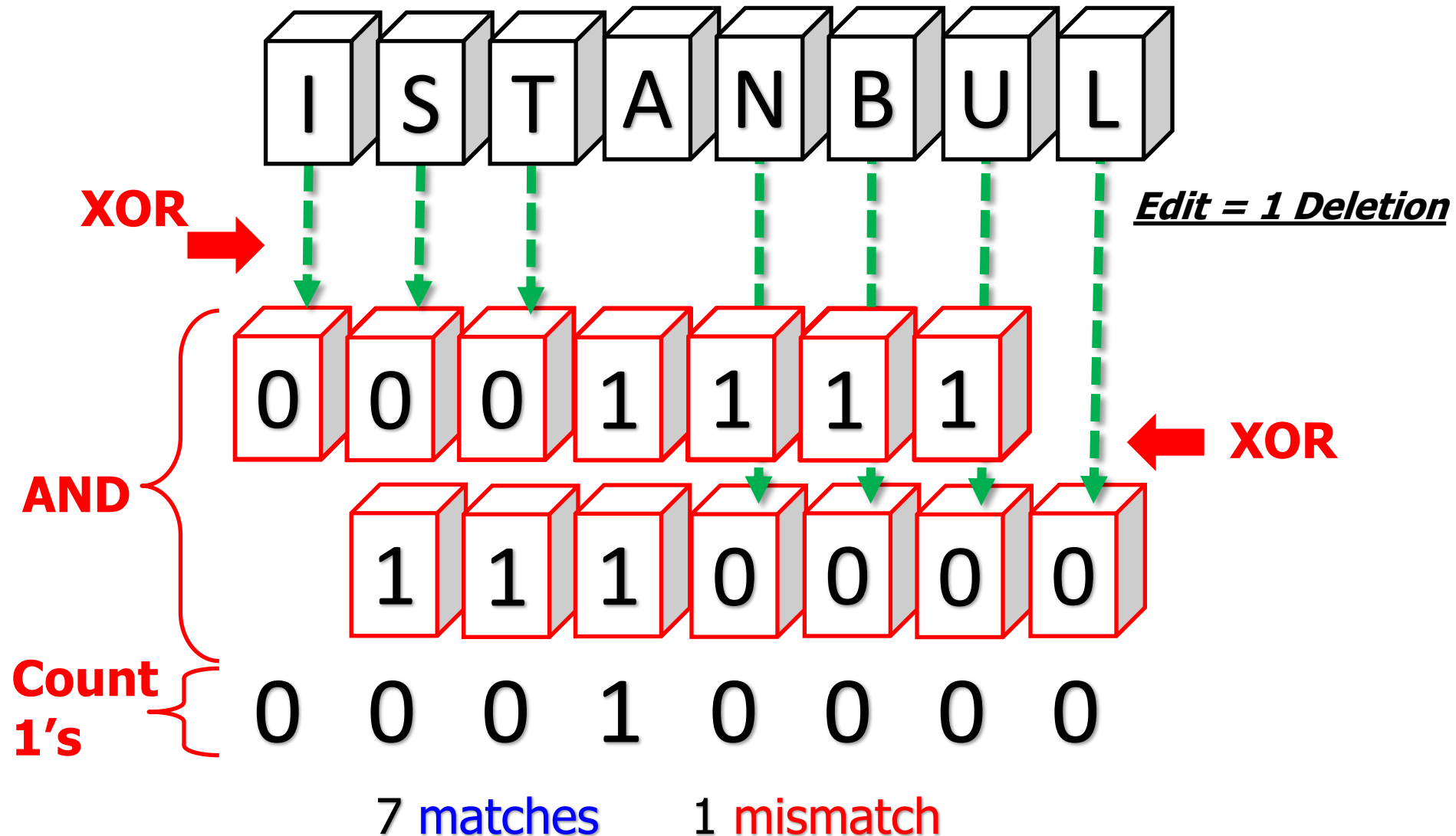
To cancel the effect of the deletion, we need to shift in the *right* direction

Insight: Shifting a String Helps Similarity Search

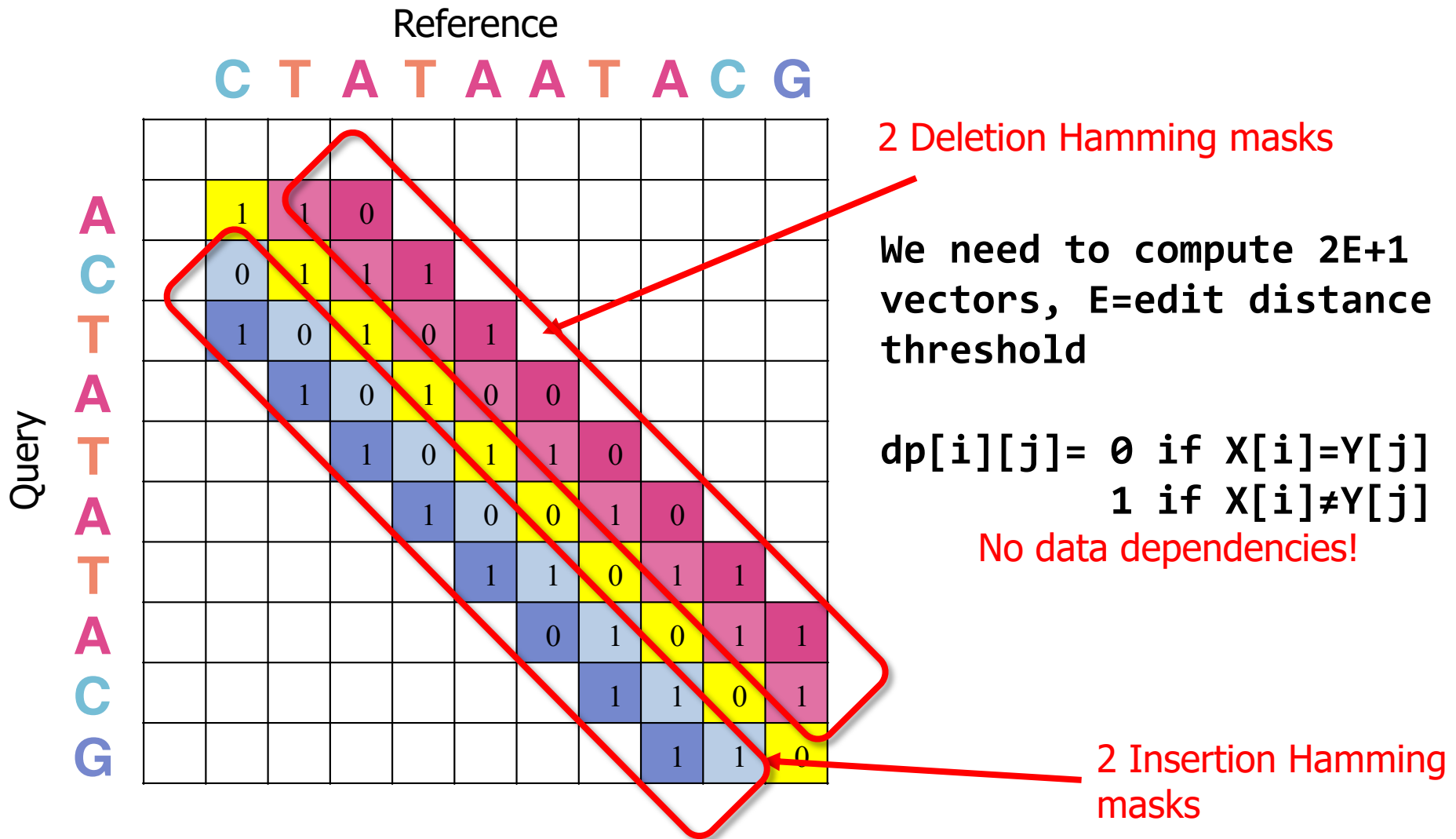
7 matches 1 mismatch



Shifted Hamming Distance

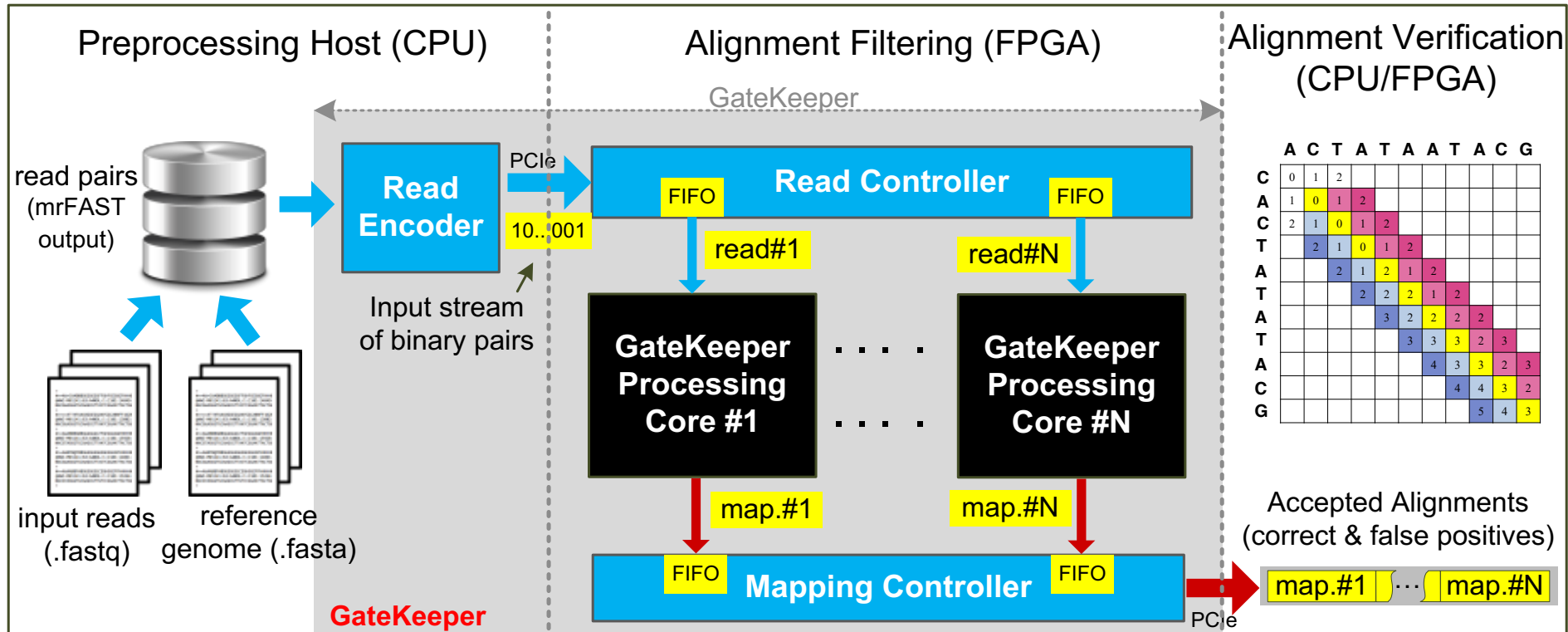


Highly Parallel Matrix Computation

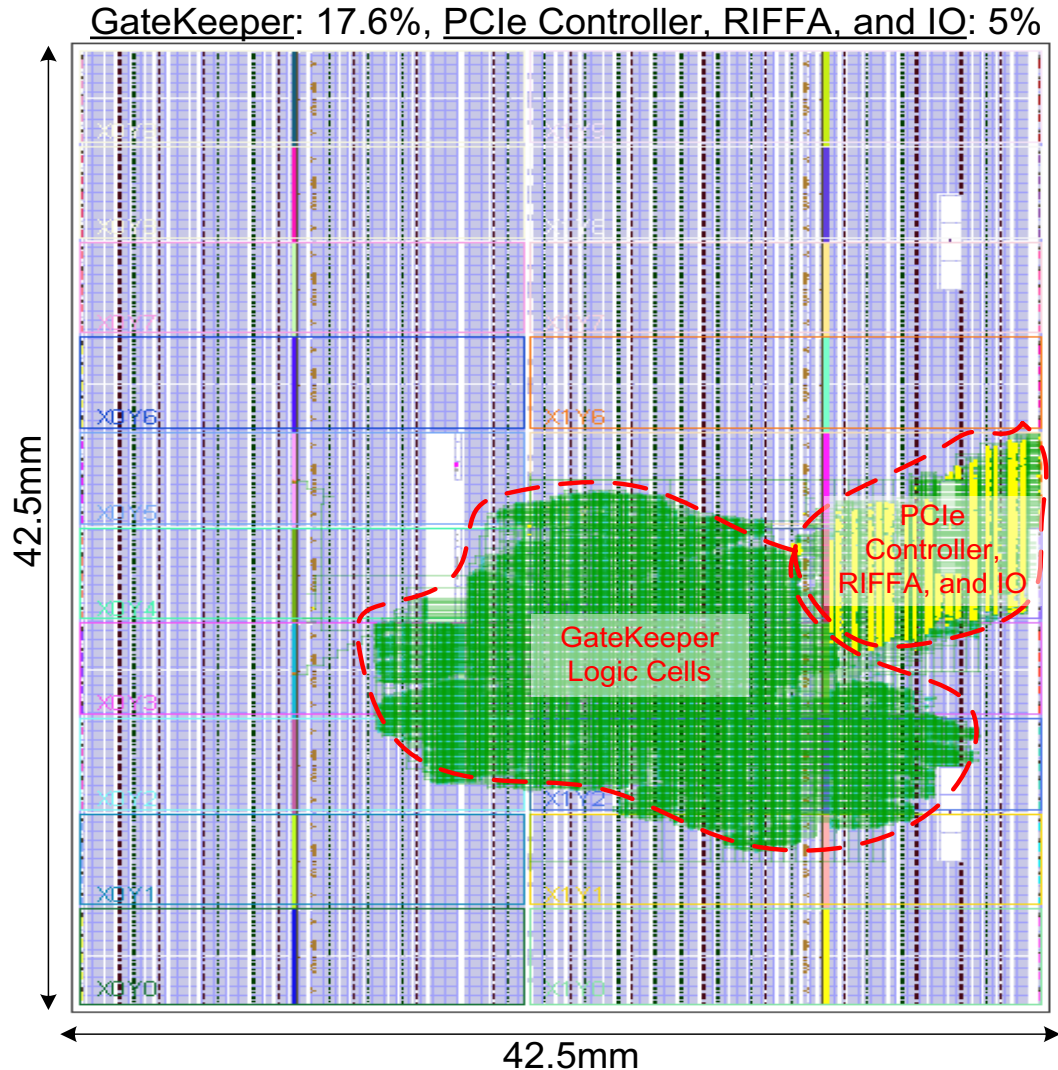


GateKeeper Accelerator Architecture

- **Maximum data throughput** = ~13.3 billion bases/sec
- Can examine **8 (300 bp) or 16 (100 bp) mappings concurrently** at 250 MHz
- **Occupies 50%** (100 bp) to **91%** (300 bp) of the FPGA slice LUTs and registers



FPGA Chip Layout



GateKeeper: Speed & Accuracy Results

90x-130x faster filter

than SHD (Xin et al., 2015) and the Adjacency Filter (Xin et al., 2013)

4x lower false accept rate

than the Adjacency Filter (Xin et al., 2013)

10x speedup in read mapping

with the addition of GateKeeper to the mrFAST mapper (Alkan et al., 2009)

Freely available online

github.com/BilkentCompGen/GateKeeper

GateKeeper Conclusions

- **FPGA-based** pre-alignment **greatly** speeds up read mapping
 - **10x speedup** of a state-of-the-art mapper (mrFAST)
- FPGA-based pre-alignment can be **integrated** with the **sequencer**
 - It can help to hide the complexity and details of the FPGA
 - Enables **real-time filtering** while sequencing
 - Paves the way to **on-device genome analysis**

More on GateKeeper

- Mohammed Alser, Hasan Hassan, Hongyi Xin, Oguz Ergin, Onur Mutlu, and Can Alkan
"GateKeeper: A New Hardware Architecture for Accelerating Pre-Alignment in DNA Short Read Mapping"
Bioinformatics, [published online, May 31], 2017.
[[Source Code](#)]
[[Online link at Bioinformatics Journal](#)]

GateKeeper: a new hardware architecture for accelerating pre-alignment in DNA short read mapping

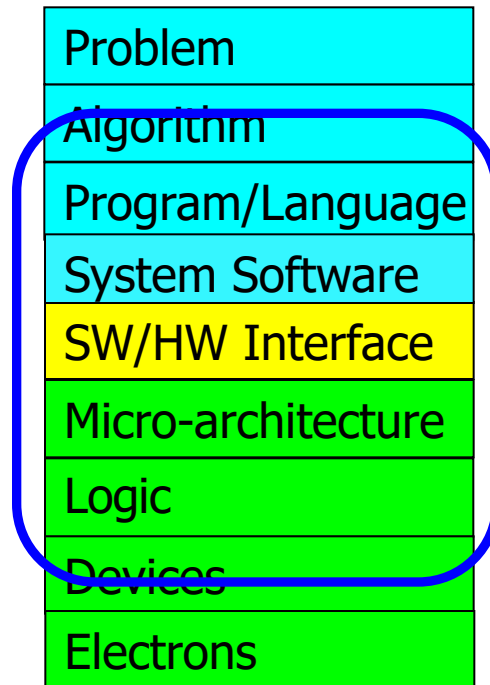
Mohammed Alser ✉, Hasan Hassan, Hongyi Xin, Oğuz Ergin, Onur Mutlu ✉, Can Alkan ✉

Bioinformatics, Volume 33, Issue 21, 1 November 2017, Pages 3355–3363,
<https://doi.org/10.1093/bioinformatics/btx342>

Published: 31 May 2017 **Article history** ▼

Algorithm-Arch-Device Co-Design is Critical

**Computer Architecture
(expanded view)**



Shouji (障子) [Alser+, Bioinformatics 2019]

Mohammed Alser, Hasan Hassan, Akash Kumar, Onur Mutlu, and Can Alkan,
"Shouji: A Fast and Efficient Pre-Alignment Filter for Sequence Alignment"
Bioinformatics, [published online, March 28], 2019.

[\[Source Code\]](#)

[\[Online link at Bioinformatics Journal\]](#)

Bioinformatics, 2019, 1–9

doi: 10.1093/bioinformatics/btz234

Advance Access Publication Date: 28 March 2019

Original Paper

OXFORD

Sequence alignment

Shouji: a fast and efficient pre-alignment filter for sequence alignment

**Mohammed Alser^{1,2,3,*}, Hasan Hassan¹, Akash Kumar², Onur Mutlu^{1,3,*}
and Can Alkan^{3,*}**

¹Computer Science Department, ETH Zürich, Zürich 8092, Switzerland, ²Chair for Processor Design, Center For Advancing Electronics Dresden, Institute of Computer Engineering, Technische Universität Dresden, 01062 Dresden, Germany and ³Computer Engineering Department, Bilkent University, 06800 Ankara, Turkey

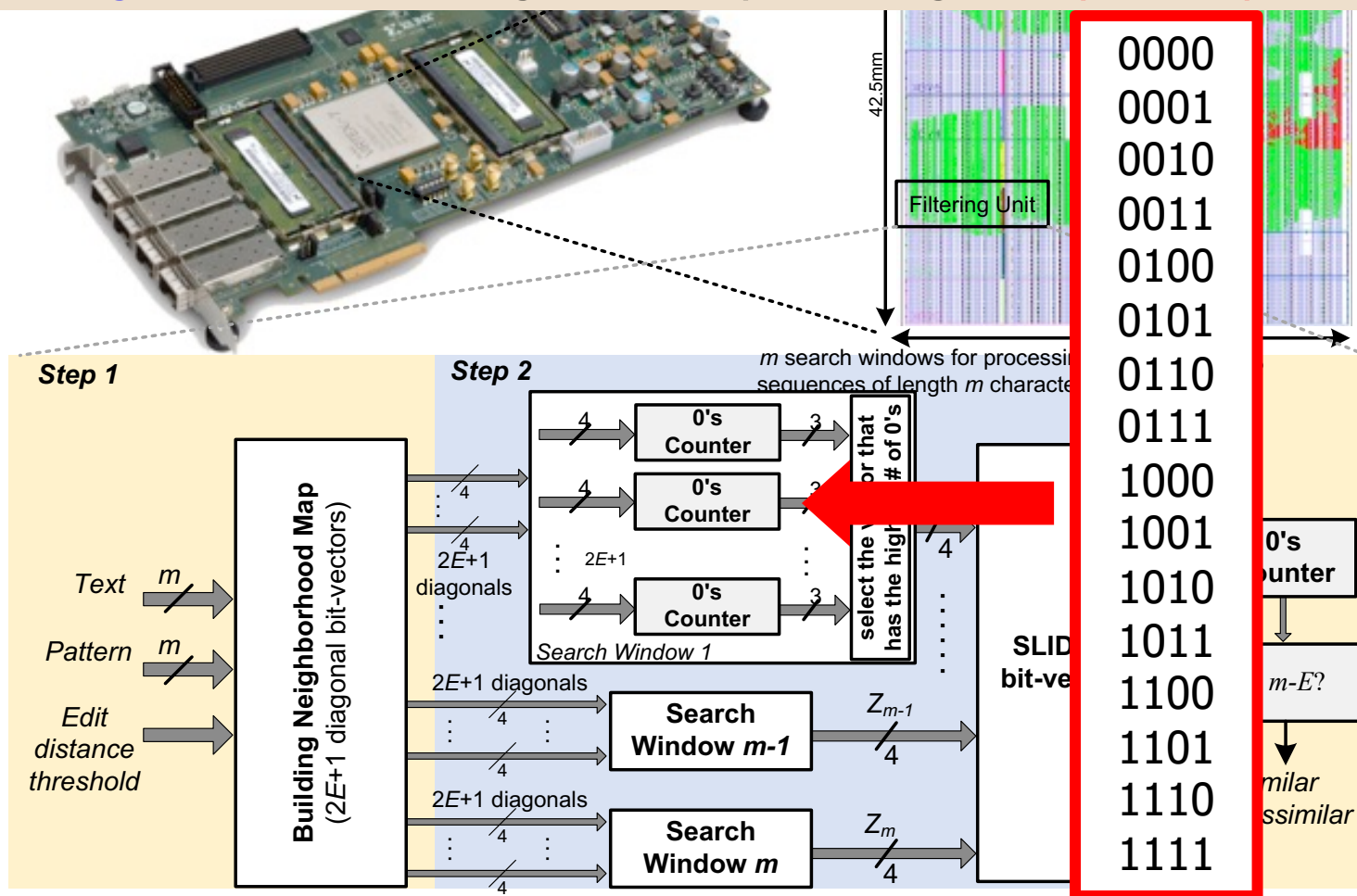
*To whom correspondence should be addressed.

Associate Editor: Inanc Birol

Received on September 13, 2018; revised on February 27, 2019; editorial decision on March 7, 2019; accepted on March 27, 2019

Hardware Implementation

Counting is performed **concurrently** for **all bit-vectors** and **all sliding windows** in a single clock cycle using **multiple 4-input LUTs**



SneakySnake [Alser+, Bioinformatics 2020]

Mohammed Alser, Taha Shahroodi, Juan-Gomez Luna, Can Alkan, and Onur Mutlu,
**"SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment
Filter for CPUs, GPUs, and FPGAs"**

Bioinformatics, to appear in 2020.

[[Source Code](#)]

[[Online link at Bioinformatics Journal](#)]

Bioinformatics

doi.10.1093/bioinformatics/xxxxxx

Advance Access Publication Date: Day Month Year

Manuscript Category

OXFORD

Subject Section

SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment Filter for CPUs, GPUs, and FPGAs

**Mohammed Alser^{1,2,*}, Taha Shahroodi¹, Juan Gómez-Luna^{1,2},
Can Alkan^{4,*}, and Onur Mutlu^{1,2,3,4,*}**

¹Department of Computer Science, ETH Zurich, Zurich 8006, Switzerland

²Department of Information Technology and Electrical Engineering, ETH Zurich, Zurich 8006, Switzerland

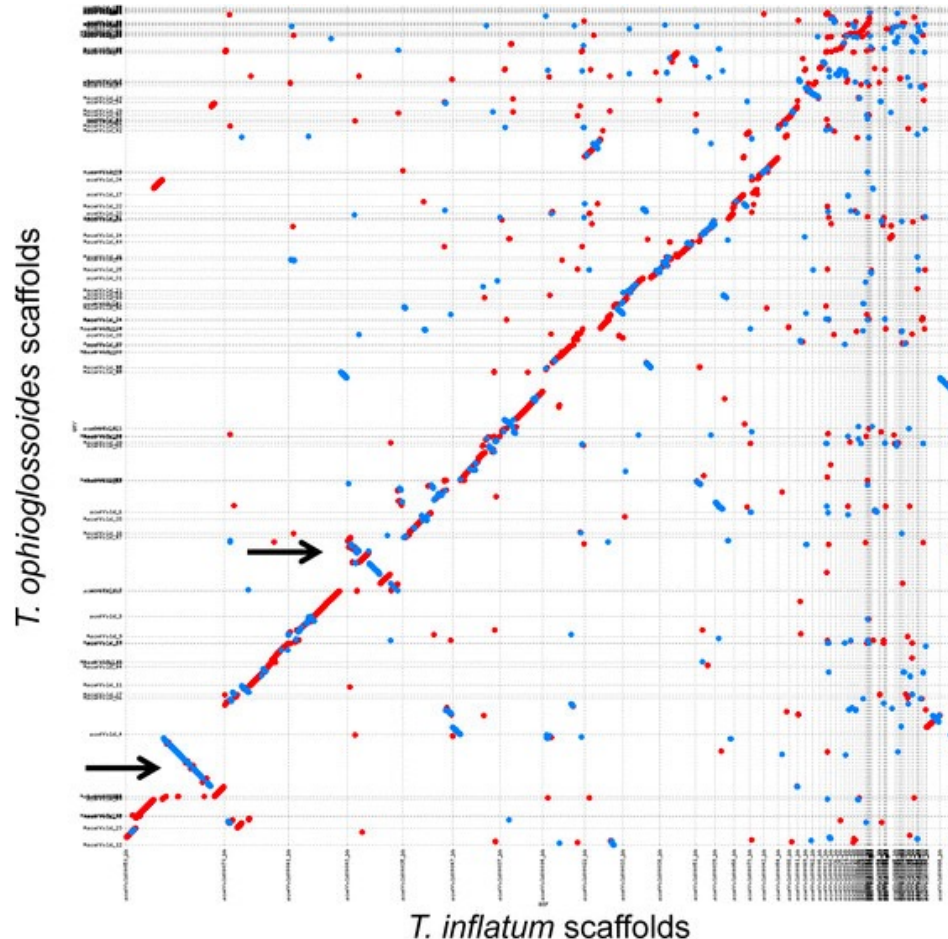
³Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh 15213, PA, USA

⁴Department of Computer Engineering, Bilkent University, Ankara 06800, Turkey

SneakySnake

■ Key observation:

- Correct alignment is a sequence of non-overlapping long matches



Dot plot, dot matrix
(Lipman and Pearson, 1985)

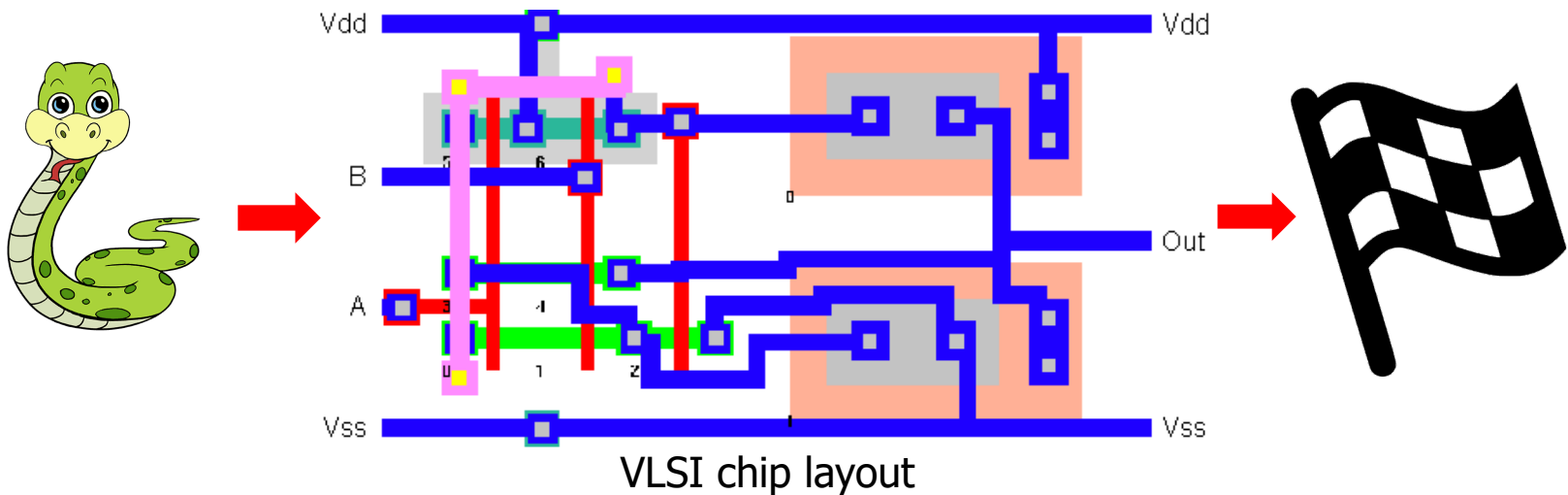
SneakySnake

■ Key observation:

- Correct alignment is a sequence of non-overlapping long matches

■ Key idea:

- Reduce the approximate string matching problem to the **Single Net Routing problem** in VLSI chip layout



SneakySnake

■ Key observation:

- ❑ Correct alignment is a sequence of non-overlapping long matches

■ Key idea:

- ❑ Reduce the approximate string matching problem to the Single Net Routing problem in VLSI chip layout

■ Key result:

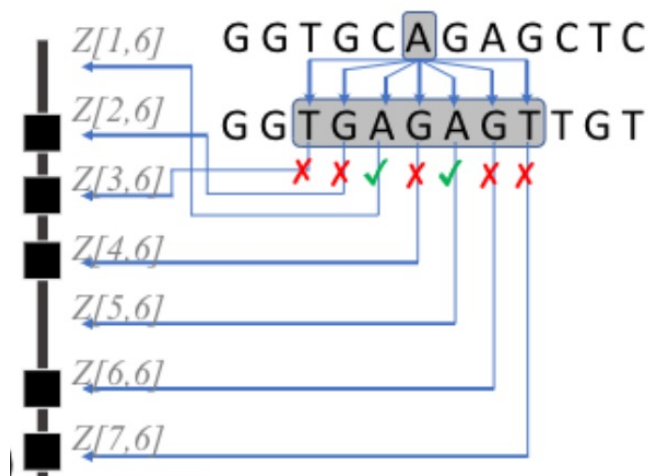
- ❑ SneakySnake is up to four orders of magnitude more accurate than Shouji (Bioinformatics'19) and GateKeeper (Bioinformatics'17)
- ❑ SneakySnake greatly accelerates state-of-the-art CPU sequence aligners, Edlib (Bioinformatics'17) and Parasail (BMC Bioinformatics'16)
 - ❑ by up to 37.7× and 43.9× (>12× on average), on CPUs
 - ❑ by up to 413× and 689× (>400× on average) with *FPGAs/GPUs*

SneakySnake Walkthrough

Building Neighborhood Map

Finding the Optimal Routing Path

Examining the Snake Survival



$$E = 3$$

column	1	2	3	4	5	6	7	8	9	10	11	12
3 rd Upper Diagonal	1	1	1	0	1	1	0	0	0	1	1	1
2 nd Upper Diagonal	1	1	1	0	1	1	1	1	1	1	0	1
1 st Upper Diagonal	1	0	1	1	1	0	0	0	0	1	0	1
Main Diagonal	0	0	0	0	1	1	1	1	1	1	1	1
1 st Lower Diagonal	0	1	1	1	1	0	0	1	1	1	0	1
2 nd Lower Diagonal	1	0	1	0	1	1	1	1	0	1	1	1
3 rd Lower Diagonal	0	1	1	1	1	1	1	1	1	1	1	1

SneakySnake Walkthrough

Building Neighborhood Map

Finding the Optimal Routing Path

Examining the Snake Survival

$$E = 3$$

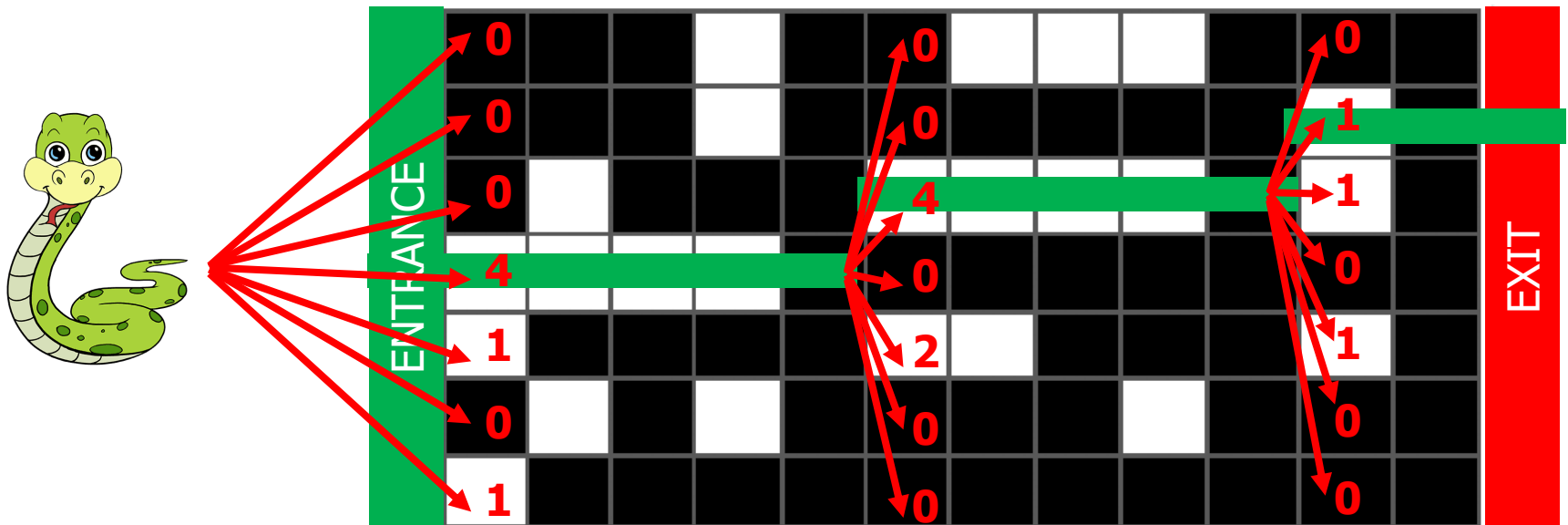
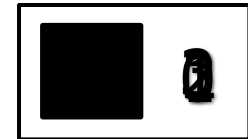
	column	1	2	3	4	5	6	7	8	9	10	11	12	
<i>3rd Upper Diagonal</i>	ENTRANCE	■	■	■	□	■	■	□	□	□	■	■	■	EXIT
<i>2nd Upper Diagonal</i>		■	■	■	□	■	■	■	■	■	■	□	■	
<i>1st Upper Diagonal</i>		■	□	■	■	■	□	□	□	□	■	□	■	
<i>Main Diagonal</i>		□	□	□	□	■	■	■	■	■	■	■	■	
<i>1st Lower Diagonal</i>		□	■	■	■	■	□	□	■	■	■	□	■	
<i>2nd Lower Diagonal</i>		■	□	■	□	■	■	■	■	□	■	■	■	
<i>3rd Lower Diagonal</i>		□	■	■	■	■	■	■	■	■	■	■	■	

SneakySnake Walkthrough

Building Neighborhood Map

Finding the Optimal Routing Path

Examining the Snake Survival



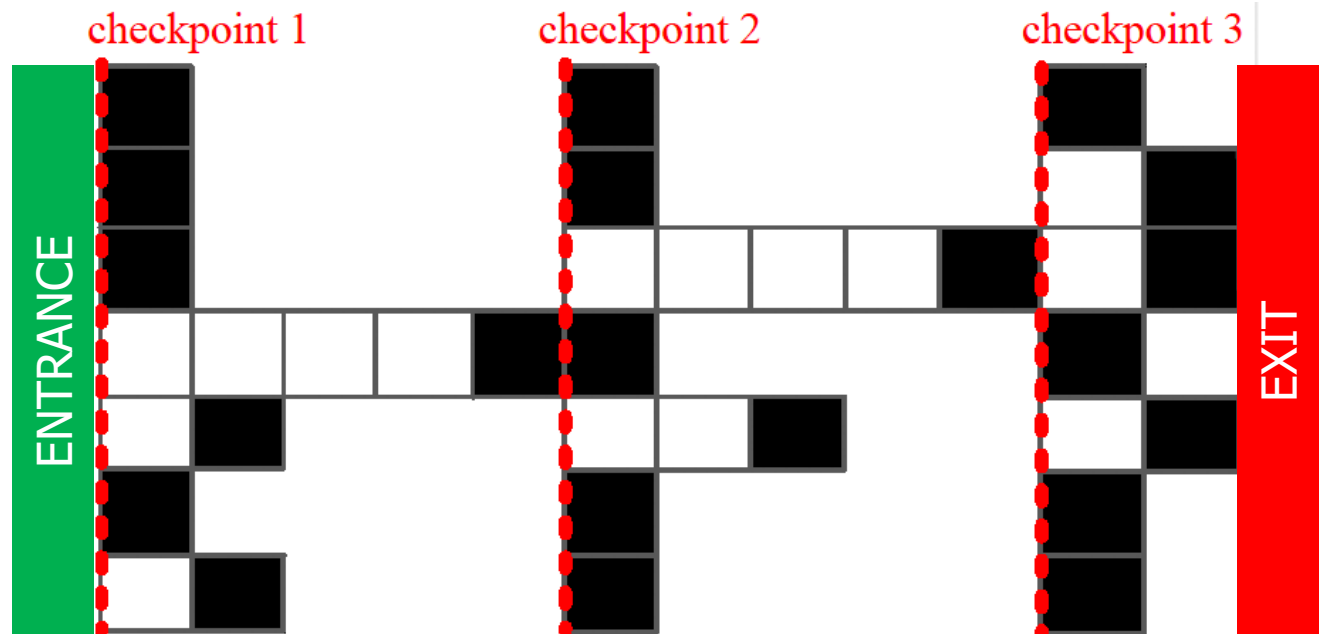
SneakySnake Walkthrough

Building Neighborhood Map

Finding the Routing Travel Path

Examining the Snake Survival

**This is what you actually need to build
and it can be done **on-the-fly!****



FPGA Resource Analysis

- FPGA resource usage for a single filtering unit of GateKeeper, Shouji, and Snake-on-Chip for a sequence length of 100 and under different edit distance thresholds (E).

	E (bp)	Slice LUT	Slice Register	No. of Filtering Units
GateKeeper	2	0.39%	0.01%	16
	5	0.71%	0.01%	16
Shouji	2	0.69%	0.08%	16
	5	1.72%	0.16%	16
Snake-on-Chip	2	0.68%	0.16%	16
	5	1.42%	0.34%	16

Key Results of SneakySnake

- ❑ SneakySnake is up to **four orders of magnitude more accurate** than **Shouji** (Bioinformatics'19) and **GateKeeper** (Bioinformatics'17)
- ❑ Short reads:
 - ❑ SneakySnake **accelerates Edlib** (Bioinformatics'17) and **Parasail** (BMC Bioinformatics'16) by
 - up to **37.7× and 43.9×** (>12× on average), on CPUs
 - up to **413× and 689×** (>400× on average) using **FPGAs/GPUs**
- ❑ Long reads:
 - ❑ SneakySnake **accelerates Parasail** and **KSW2** by **140.1× and 17.1×** on average, respectively, on CPUs

More on SneakySnake [Alser+, Bioinformatics 2020]

Mohammed Alser, Taha Shahroodi, Juan-Gomez Luna, Can Alkan, and Onur Mutlu,
**"SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment
Filter for CPUs, GPUs, and FPGAs"**

Bioinformatics, to appear in 2020.

[Source Code]

[Online link at Bioinformatics Journal]

Bioinformatics

doi.10.1093/bioinformatics/xxxxxx

Advance Access Publication Date: Day Month Year

Manuscript Category

OXFORD

Subject Section

SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment Filter for CPUs, GPUs, and FPGAs

**Mohammed Alser^{1,2,*}, Taha Shahroodi¹, Juan Gómez-Luna^{1,2},
Can Alkan^{4,*}, and Onur Mutlu^{1,2,3,4,*}**

¹Department of Computer Science, ETH Zurich, Zurich 8006, Switzerland

²Department of Information Technology and Electrical Engineering, ETH Zurich, Zurich 8006, Switzerland

³Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh 15213, PA, USA

⁴Department of Computer Engineering, Bilkent University, Ankara 06800, Turkey

GenASM Framework [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.
[[Lighting 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

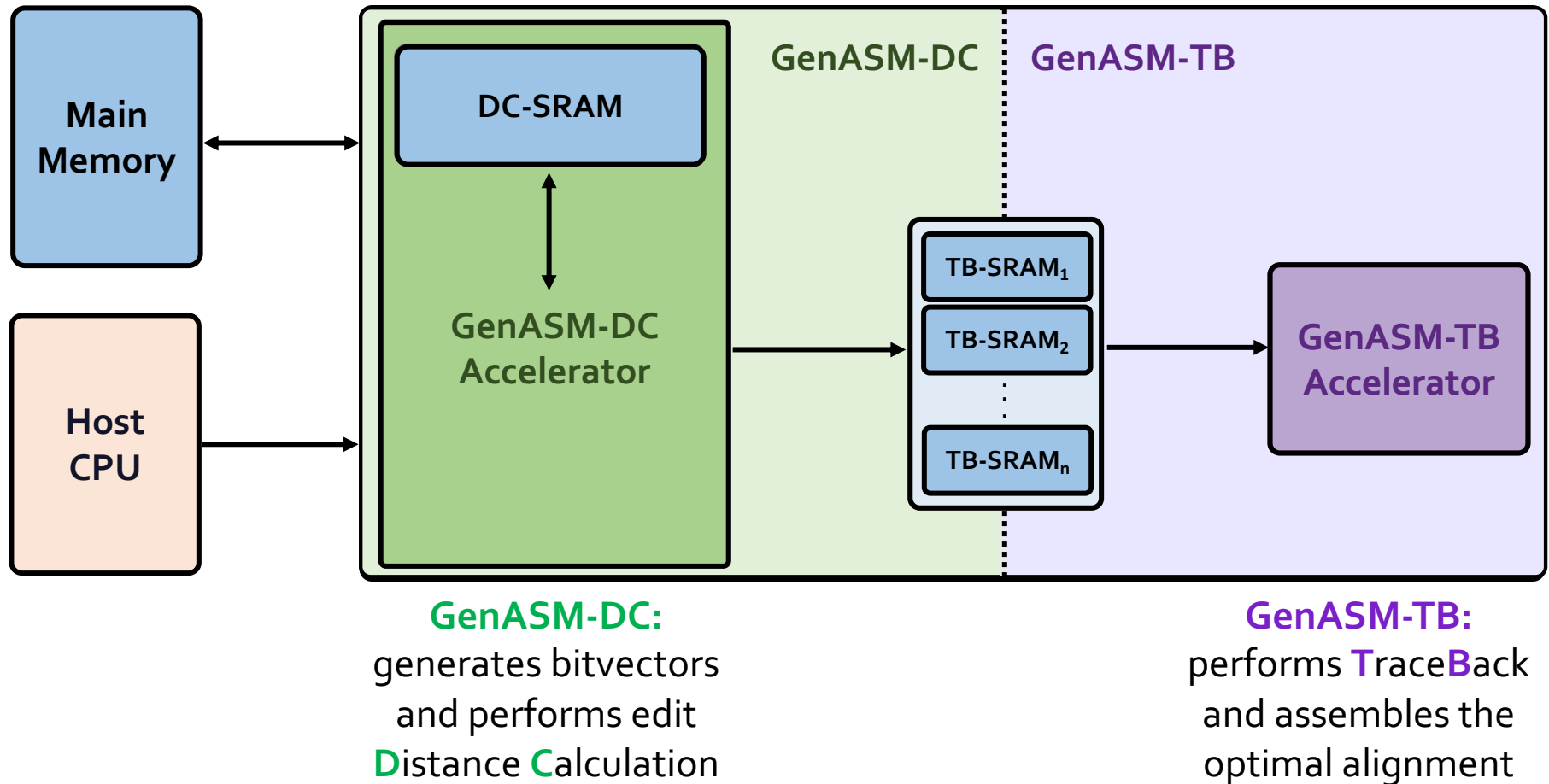
GenASM: ASM Framework for GSA

Our Goal:

Accelerate approximate string matching
by designing a fast and flexible framework,
which can accelerate *multiple steps* of genome sequence analysis

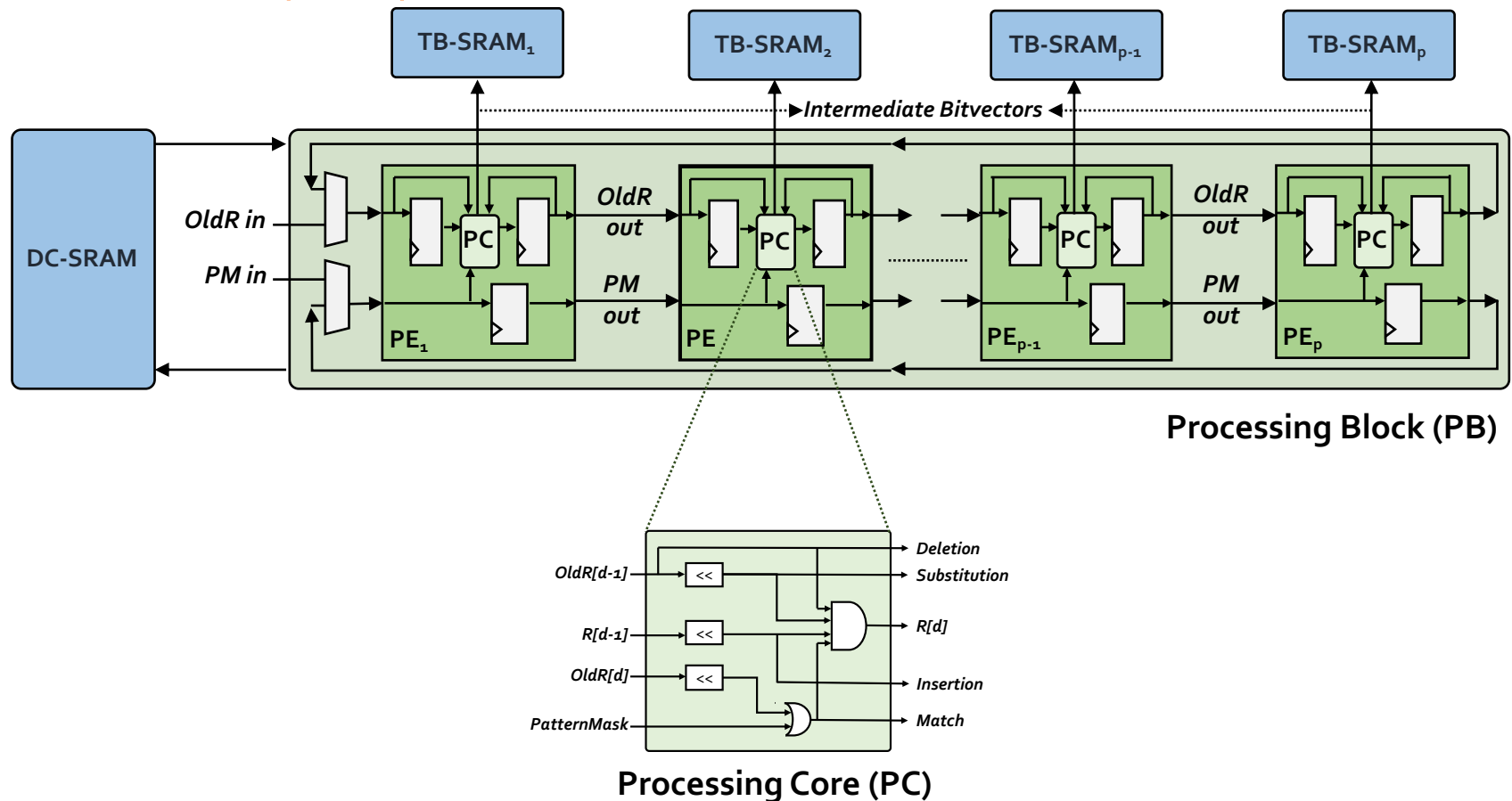
- ❑ **GenASM:** *First* ASM acceleration framework for GSA
 - Based on the *Bitap* algorithm
 - Uses fast and simple bitwise operations to perform ASM
 - Modified and extended ASM algorithm
 - Highly-parallel Bitap with long read support
 - Bitvector-based novel algorithm to perform *traceback*
 - Co-design of our modified scalable and memory-efficient algorithms with low-power and area-efficient hardware accelerators

GenASM: Hardware Design

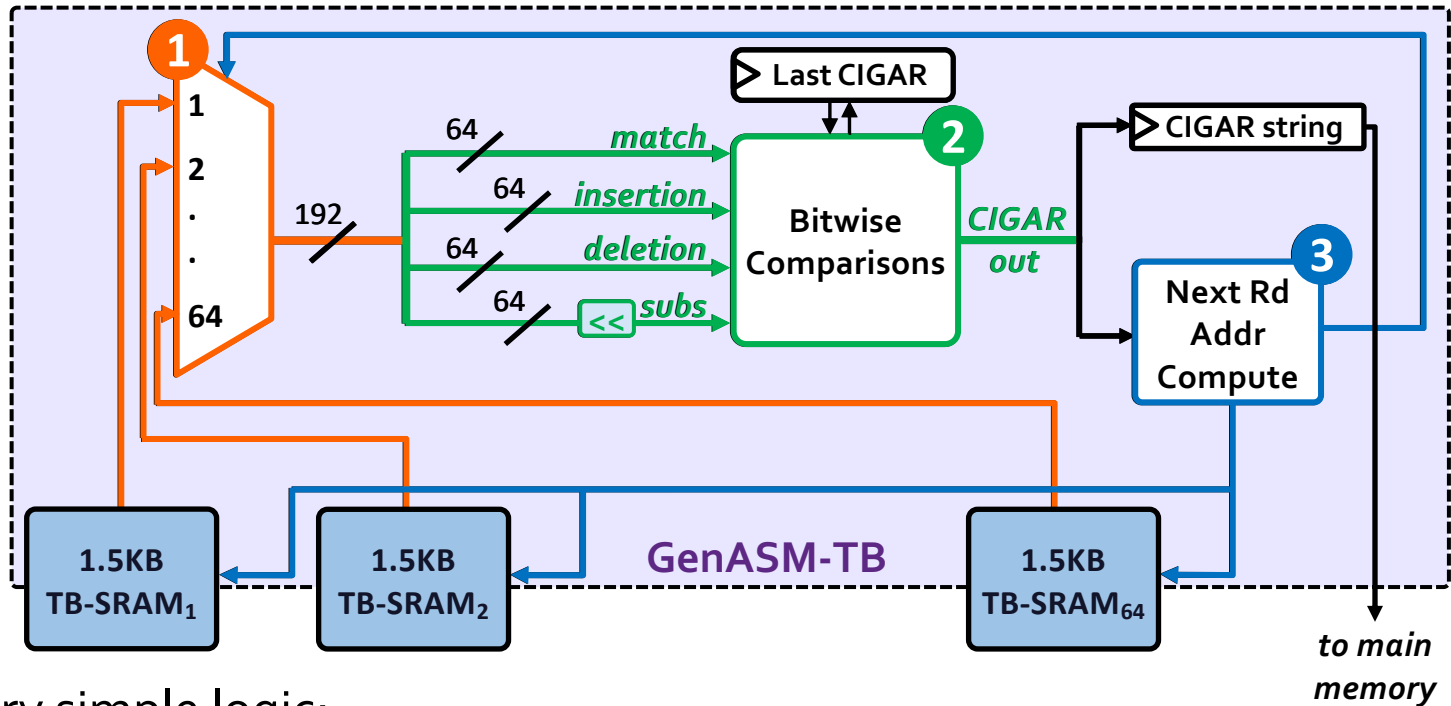


GenASM-DC: Hardware Design

- ❑ **Linear cyclic systolic array based accelerator**
 - Designed to **maximize parallelism** and **minimize memory bandwidth and memory footprint**



GenASM-TB: Hardware Design

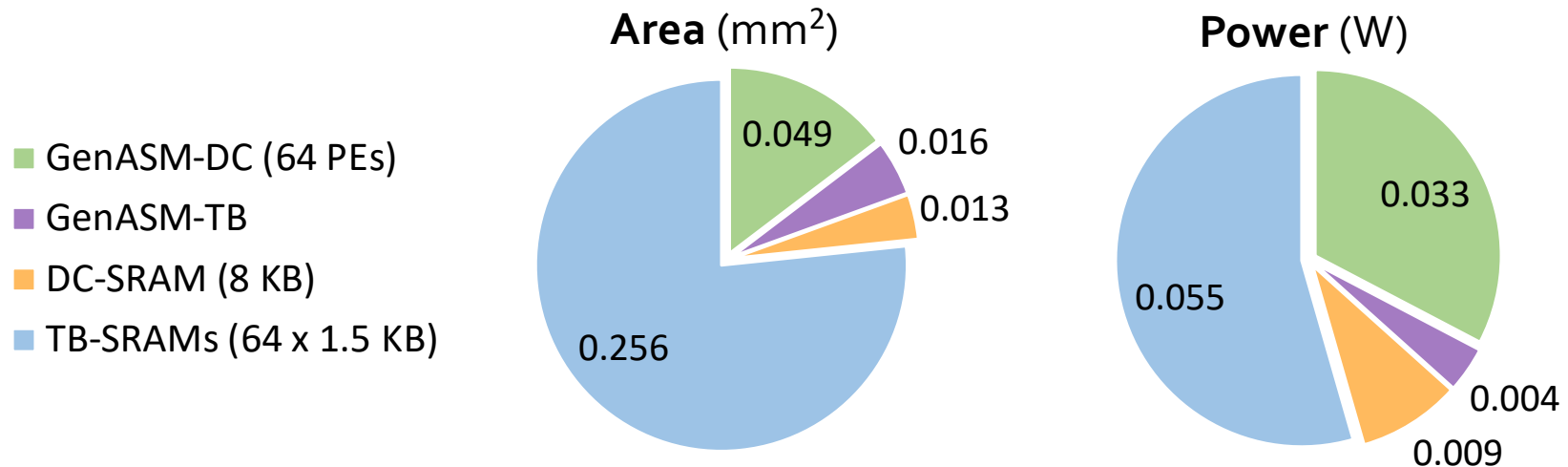


□ Very simple logic:

- ① Reads the bitvectors from one of the TB-SRAMs using the computed address
- ② Performs the required bitwise comparisons to find the traceback output for the current position
- ③ Computes the next TB-SRAM address to read the new set of bitvectors

Key Results – Area and Power

- Based on our **synthesis** of **GenASM-DC** and **GenASM-TB** accelerator datapaths using the Synopsys Design Compiler with a **28nm** LP process:
 - Both GenASM-DC and GenASM-TB operate **@ 1GHz**



Total (1 vault): 0.334 mm²

Total (32 vaults): 10.69 mm²

% of a Xeon CPU core: **1%**

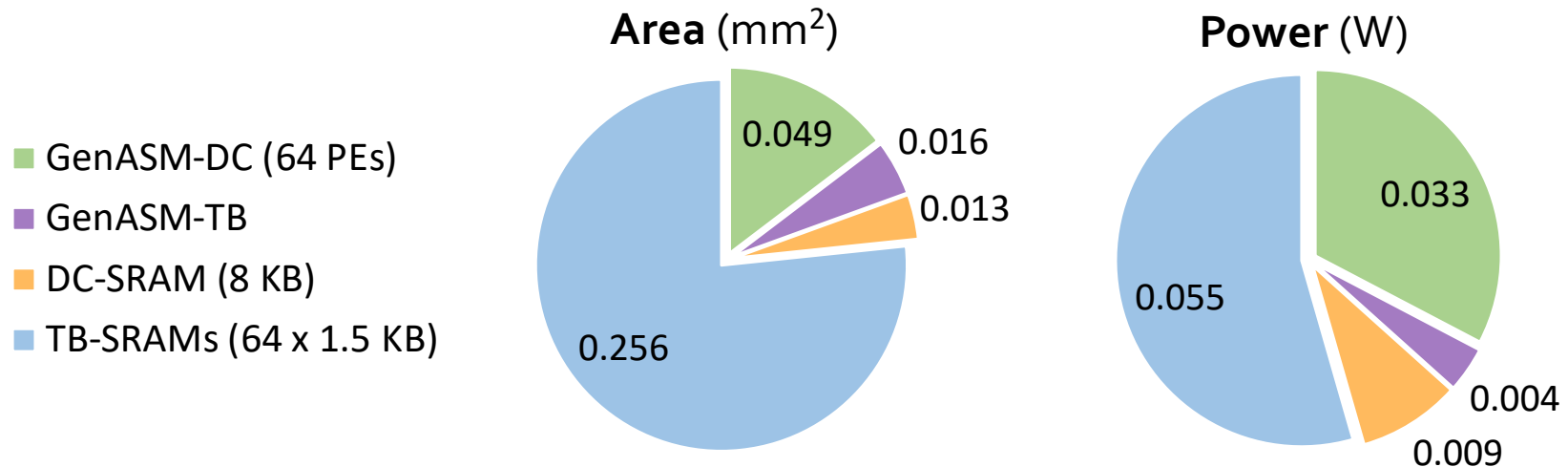
0.101 W

3.23 W

1%

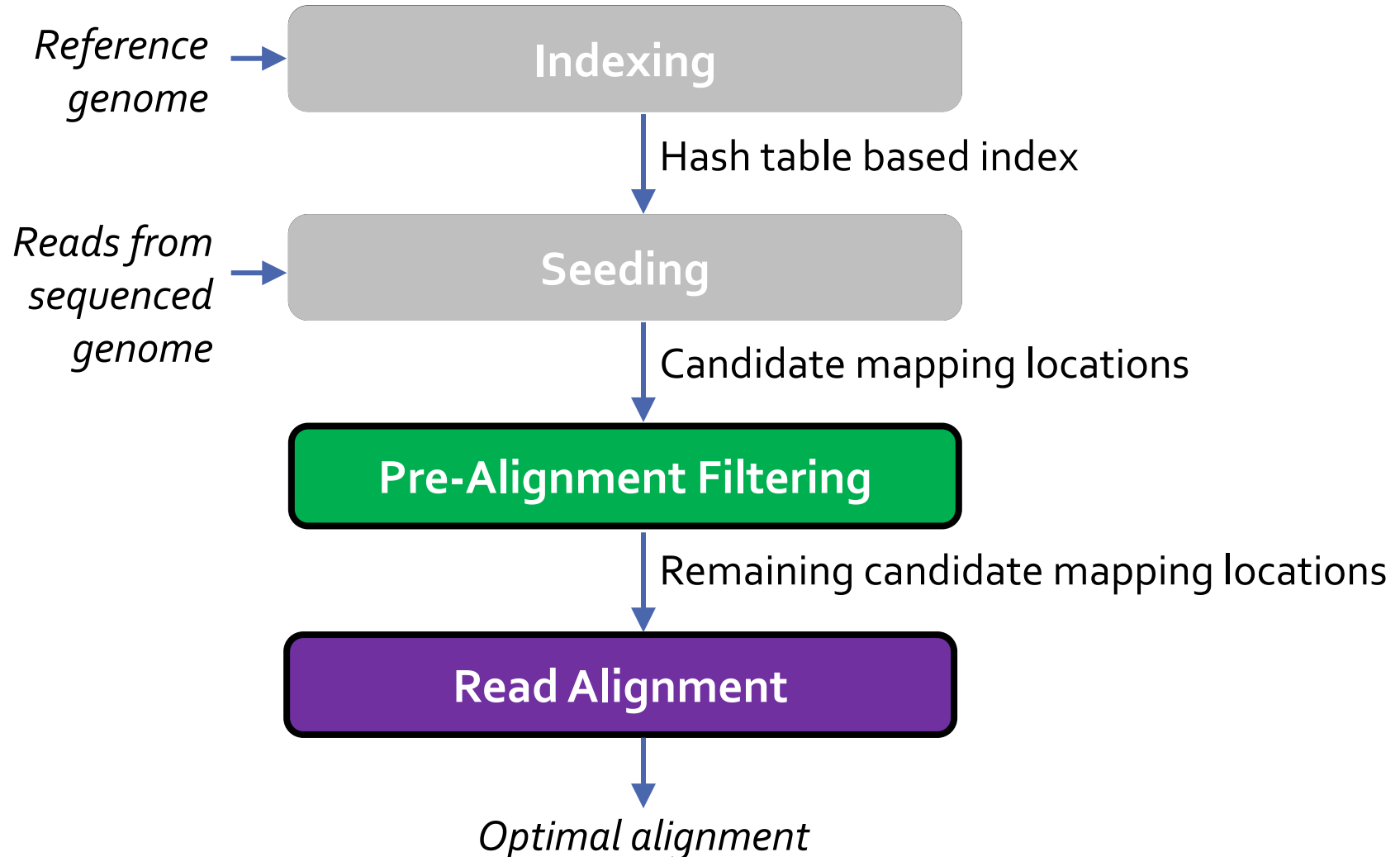
Key Results – Area and Power

- Based on our **synthesis** of **GenASM-DC** and **GenASM-TB** accelerator datapaths using the Synopsys Design Compiler with a **28nm** LP process:
 - Both GenASM-DC and GenASM-TB operate **@ 1GHz**



GenASM has low area and power overheads

Use Cases of GenASM



Use Cases of GenASM (cont'd.)

(1) Read Alignment Step of Read Mapping

- Find the **optimal alignment** of how reads map to candidate reference regions

(2) Pre-Alignment Filtering for Short Reads

- Quickly identify and **filter out the unlikely** candidate reference regions for each read

(3) Edit Distance Calculation

- Measure the **similarity** or **distance** between two sequences
- We also discuss **other possible use cases of GenASM** in our paper:
- Read-to-read overlap finding, hash-table based indexing, whole genome alignment, generic text search

Key Results

(1) Read Alignment

- ❑ **116×** speedup, **37×** less power than **Minimap2** (state-of-the-art **SW**)
- ❑ **111×** speedup, **33×** less power than **BWA-MEM** (state-of-the-art **SW**)
- ❑ **3.9×** better throughput, **2.7×** less power than **Darwin** (state-of-the-art **HW**)
- ❑ **1.9×** better throughput, **82%** less logic power than **GenAx** (state-of-the-art **HW**)

(2) Pre-Alignment Filtering

- ❑ **3.7×** speedup, **1.7×** less power than **Shouji** (state-of-the-art **HW**)

(3) Edit Distance Calculation

- ❑ **22–12501×** speedup, **548–582×** less power than **Edlib** (state-of-the-art **SW**)
- ❑ **9.3–400×** speedup, **67×** less power than **ASAP** (state-of-the-art **HW**)

More on GenASM Framework [MICRO 2020]

- 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, 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

Scrooge: Faster Approximate String Matching

- Joël Lindegger, Damla Senol Cali, Mohammed Alser, Juan Gómez-Luna, Nika Mansouri Ghiasi, and Onur Mutlu,
["Scrooge: A Fast and Memory-Frugal Genomic Sequence Aligner for CPUs, GPUs, and ASICs"](#)
[*Bioinformatics*](#), [published online on] 24 March 2023.
[[Online link at Bioinformatics Journal](#)]
[[arXiv preprint](#)]
[[Scrooge Source Code](#)]

Scrooge: A Fast and Memory-Frugal Genomic Sequence Aligner for CPUs, GPUs, and ASICs

Joël Lindegger[§]
Juan Gómez-Luna[§]

Damla Senol Cali[†]
Nika Mansouri Ghiasi[§]

Mohammed Alser[§]
Onur Mutlu[§]

[§]*ETH Zürich*

[†]*Bionano Genomics*

<https://github.com/cmu-safari/scrooge>

Our Goals

Build a **practical** and **efficient** implementation
of the **GenASM algorithm**
for **multiple computing platforms**

Compete with **state-of-the-art** pairwise sequence
aligners like **Edlib**, **KSW2**, and **BiWFA**

Scrooge

Three **novel algorithmic improvements**
which address **inefficiencies** in the **GenASM algorithm**

Efficient open-source implementations
for **CPUs** and **GPUs**

Key Results

Scrooge consistently **outperforms GenASM**

- **2.1x** speedup over GenASM on **CPU**
- **5.9x** speedup over GenASM on **GPU**
- **3.6x** better area efficiency than GenASM as an **ASIC**

Scrooge consistently **outperforms state-of-the-art CPU** and **GPU** baselines,
including KSW2, Edlib, and BiWFA

SAFARI

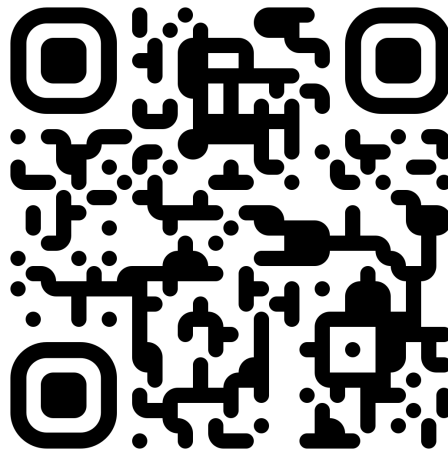
Scrooge on GitHub

☰ README.md ✎

Scrooge: A fast and memory-frugal genomic sequence aligner for CPUs, GPUs and ASICs

Scrooge is a fast pairwise genomic sequence aligner. It efficiently aligns short and long genomic sequence pairs on multiple computing platforms. It is based on the GenASM algorithm ([Senol Cali+, 2020](#)), and adds multiple algorithmic improvements that significantly improve the throughput and resource efficiency for CPUs, GPUs and ASICs. For long reads, the CPU version of Scrooge achieves a 20.1x, 1.7x, and 2.1x speedup over KSW2, Edlib, and a CPU implementation of GenASM, respectively. The GPU version of Scrooge achieves a 4.0x 80.4x, 6.8x, 12.6x and 5.9x speedup over the CPU version of Scrooge, KSW2, Edlib, Darwin-GPU, and a GPU implementation of GenASM, respectively. We estimate an ASIC implementation of Scrooge to use 3.6x less chip area and 2.1x less power than a GenASM ASIC while maintaining the same throughput.

This repository contains Scrooge's CPU and GPU implementations, and several evaluation scripts. We describe Scrooge in our paper [on arXiv](#) and [in Bioinformatics](#).



Scrooge on GitHub

🍴 3 forks

Report repository

Releases

No releases published

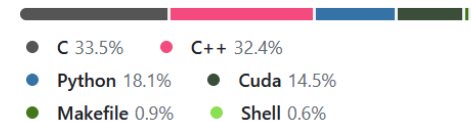
[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Languages



ASIC Results

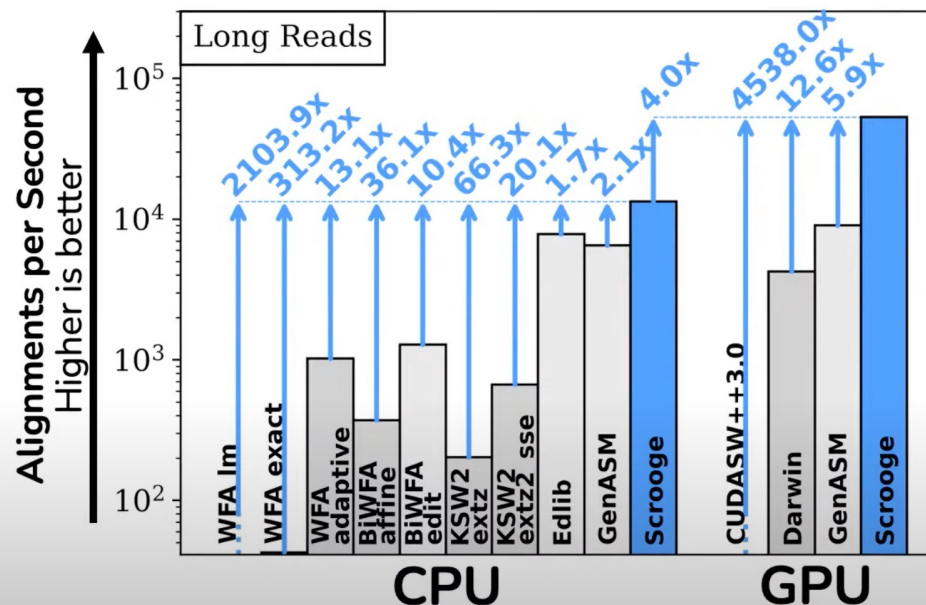
Scrooge introduces
no significant computation overhead
over a GenASM ASIC

Scrooge's on-chip memory is much cheaper than GenASM's
due to the memory footprint and bandwidth reductions
(uses 18x less chip area and 18x less power)

Scrooge uses 3.6x less chip area
and 2.1x less power than a GenASM ASIC

Scrooge Talk Video

Long Read Throughput



For long reads, **Scrooge** outperforms **GenASM**

by 2.1x on CPU and 5.9x on GPU

BIO-Arch: Workshop on Hardware Acceleration of Bioinformatics Workloads



Onur Mutlu Lectures
32.9K subscribers

Analytics

Edit video

68



Share

Download

Clip

Save



1,447 views Streamed live on Apr 14, 2023

BIO-Arch: Workshop on Hardware Acceleration of Bioinformatics Workloads

<https://safari.ethz.ch/recomb23-arch-...>

Accelerating Sequence-to-Graph Mapping

- Damla Senol Cali, Konstantinos Kanellopoulos, Joel Lindegger, Zülal Bingöl, Gurpreet S. Kalsi, Ziyi Zuo, Can Firtina, Meryem Banu Cavlak, Jeremie Kim, Nika Mansouri Ghiasi, 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 Cali¹ Konstantinos Kanellopoulos² Joël Lindegger² Zülal Bingöl³
Gurpreet S. Kalsi⁴ Ziyi Zuo⁵ Can Firtina² Meryem Banu Cavlak² Jeremie Kim²
Nika Mansouri Ghiasi² Gagandeep Singh² Juan Gómez-Luna² Nour Almadhoun Alserr²
Mohammed Alser² Sreenivas Subramoney⁴ Can Alkan³ Saugata Ghose⁶ Onur Mutlu²

¹Bionano Genomics ²ETH Zürich ³Bilkent University ⁴Intel Labs

⁵Carnegie Mellon University ⁶University of Illinois Urbana-Champaign

Genome Sequence Analysis

- Mapping the reads to a reference genome (i.e., *read mapping*) is a *critical step* in genome sequence analysis

Linear Reference: ACG**T**ACGT

Read: ACG**G**

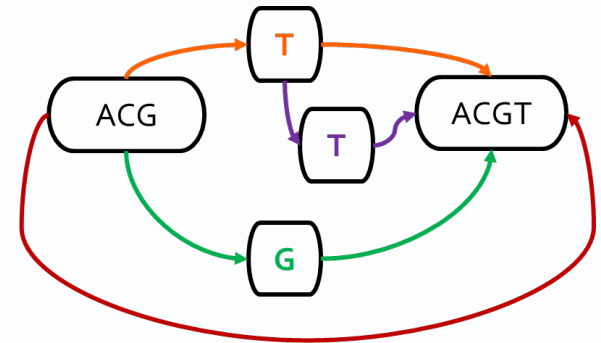
Alternative Sequence: ACG**G**ACGT

Alternative Sequence: ACG**TT**ACGT

Alternative Sequence: ACG–ACGT

Sequence-to-Sequence (S2S) Mapping

Graph-based Reference:



Read: ACG**G**

Sequence-to-Graph (S2G) Mapping

Sequence-to-graph mapping results in **notable quality improvements**.

However, it is a **more difficult** computational problem,
with **no prior hardware design**.

SeGraM: First Graph Mapping Accelerator

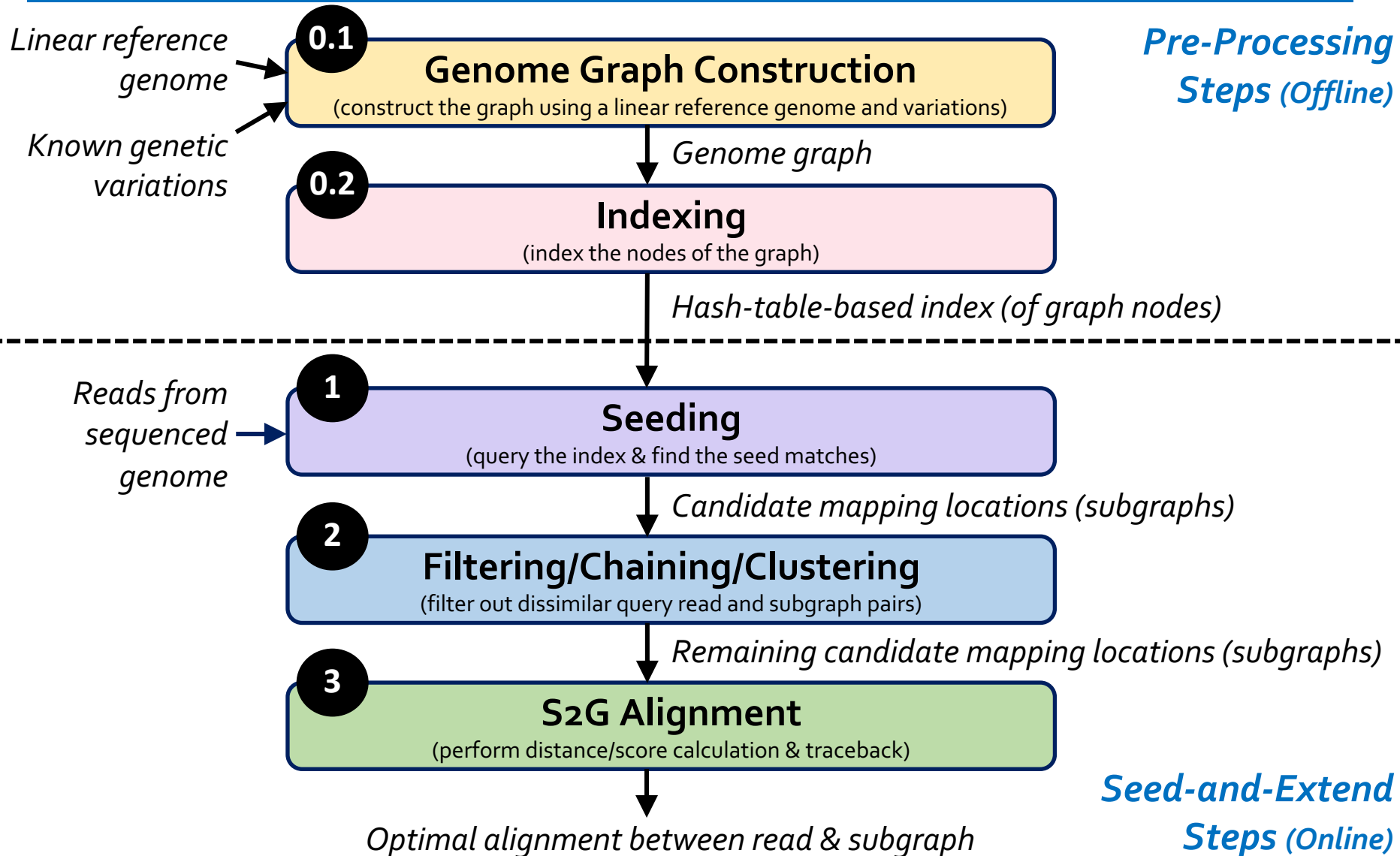
Our Goal:

Specialized, high-performance, scalable, and low-cost algorithm/hardware co-design that alleviates bottlenecks in **multiple steps** of sequence-to-graph mapping

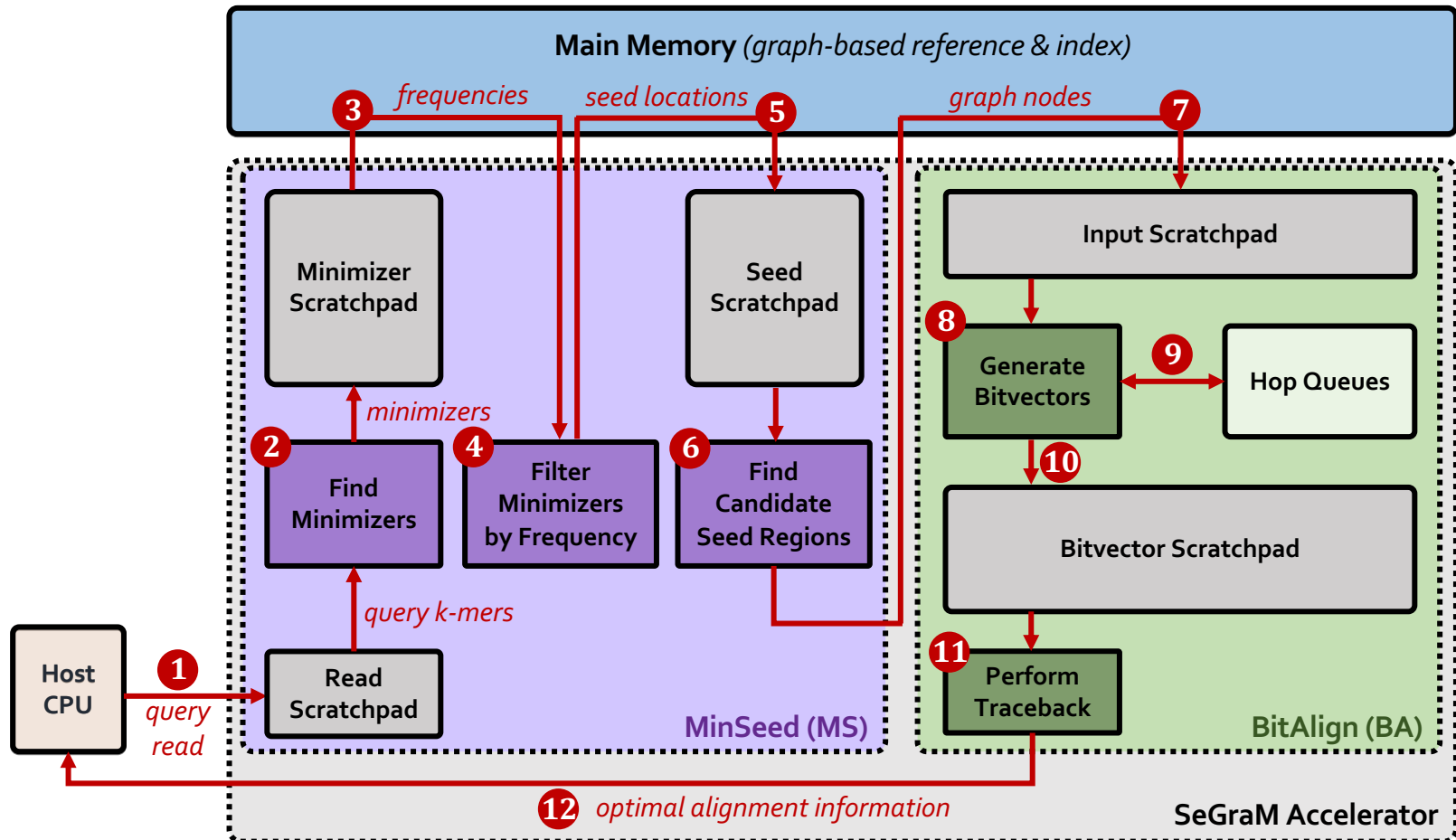
SeGraM: *First universal algorithm/hardware co-designed genomic mapping accelerator* that can effectively and efficiently support:

- ☐ Sequence-to-graph mapping
- ☐ Sequence-to-sequence mapping
- ☐ Both short and long reads

Sequence-to-Graph Mapping Pipeline



SeGraM Hardware Design



MinSeed: first hardware accelerator for Minimizer-based Seeding

BitAlign: first hardware accelerator for (Bitvector-based) sequence-to-graph Alignment

Use Cases & Key Results

(1) Sequence-to-Graph (S2G) Mapping

- ❑ **5.9×/106×** speedup, **4.1×/3.0×** less power than **GraphAligner** for long and short reads, respectively (state-of-the-art **SW**)
- ❑ **3.9×/742×** speedup, **4.4×/3.2×** less power than **vg** for long and short reads, respectively (state-of-the-art **SW**)

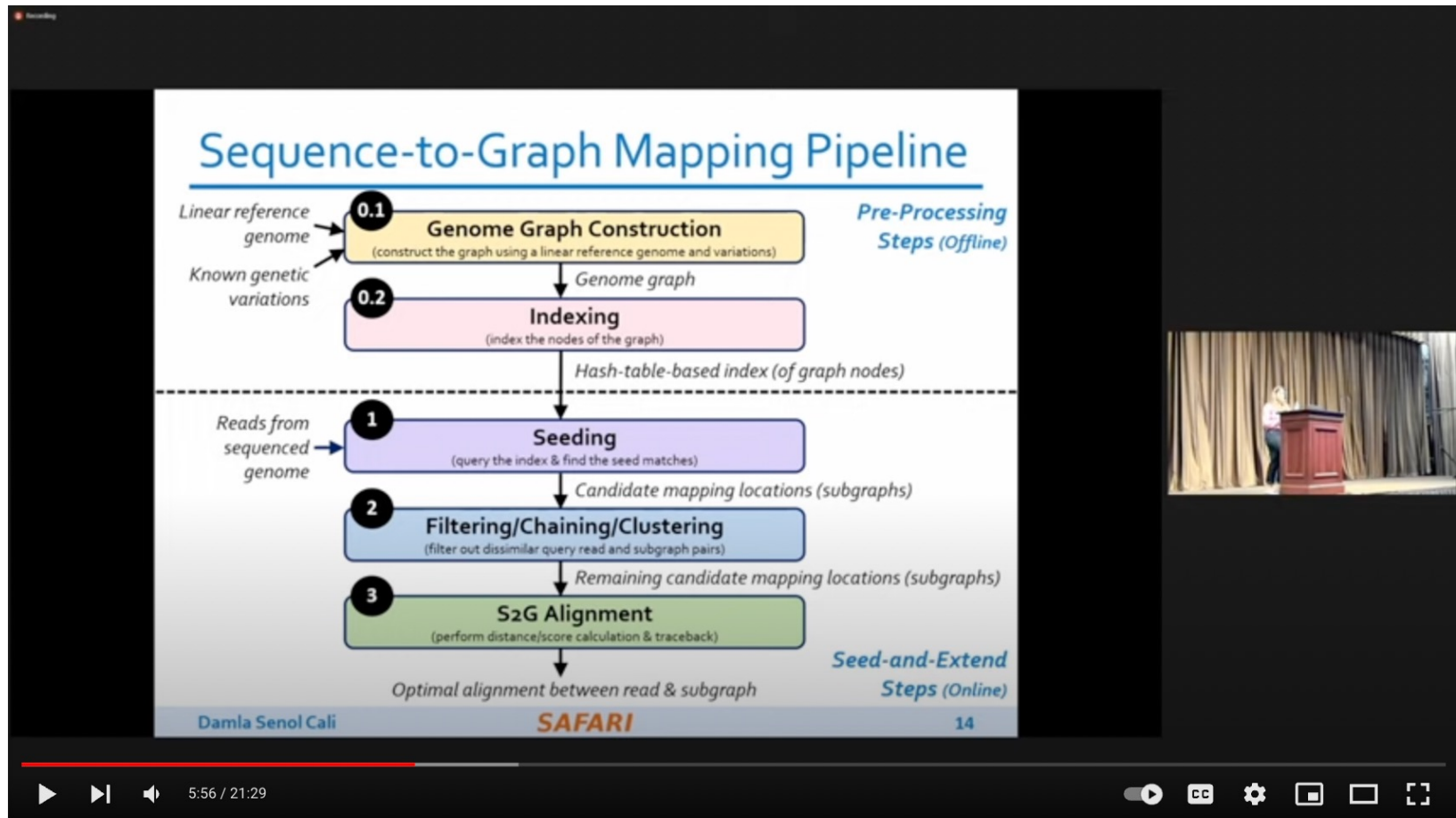
(2) Sequence-to-Graph (S2G) Alignment

- ❑ **41×–539×** speedup over **PaSGAL** with AVX-512 support (state-of-the-art **SW**)

(3) Sequence-to-Sequence (S2S) Alignment

- ❑ **1.2×/4.8×** higher throughput than **GenASM** and **GACT of Darwin** for long reads (state-of-the-art **HW**)
- ❑ **1.3×/2.4×** higher throughput than **GenASM** and **SillaX of GenAX** for short reads (state-of-the-art **HW**)

SeGraM Talk Video



SeGraM: A Universal HW Accelerator for Genomic Sequence-to-Graph Mapping - Damla Senol Cali (ISCA)

136 views • Premiered 21 hours ago

👍 12 🗑 DISLIKE ➦ SHARE ⬇ DOWNLOAD ✂ CLIP ⚙ SAVE ...



Onur Mutlu Lectures
26.9K subscribers

ANALYTICS

EDIT VIDEO

Accelerating Sequence-to-Graph Mapping

- Damla Senol Cali, Konstantinos Kanellopoulos, Joel Lindegger, Zülal Bingöl, Gurpreet S. Kalsi, Ziyi Zuo, Can Firtina, Meryem Banu Cavlak, Jeremie Kim, Nika Mansouri Ghiasi, 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 Cali¹ Konstantinos Kanellopoulos² Joël Lindegger² Zülal Bingöl³
Gurpreet S. Kalsi⁴ Ziyi Zuo⁵ Can Firtina² Meryem Banu Cavlak² Jeremie Kim²
Nika Mansouri Ghiasi² Gagandeep Singh² Juan Gómez-Luna² Nour Almadhoun Alserr²
Mohammed Alser² Sreenivas Subramoney⁴ Can Alkan³ Saugata Ghose⁶ Onur Mutlu²

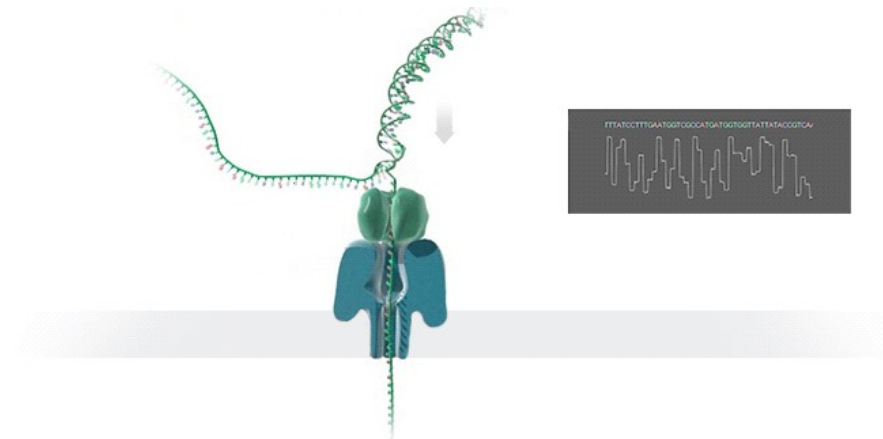
¹Bionano Genomics ²ETH Zürich ³Bilkent University ⁴Intel Labs
⁵Carnegie Mellon University ⁶University of Illinois Urbana-Champaign

Designing & Accelerating Basecallers

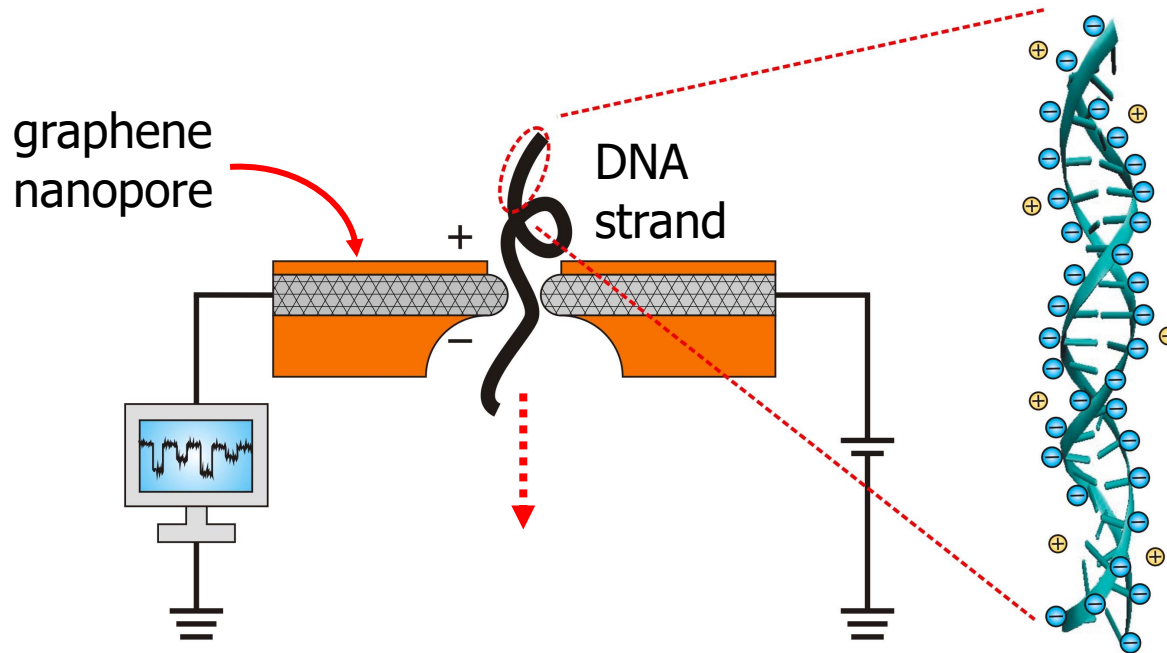
A Framework for Designing Efficient Deep Learning-Based Genomic Basecallers

Gagandeep Singh^a Mohammed Alser^{*a} Alireza Khodamoradi^{*b}
Kristof Denolf^b Can Firtina^a Meryem Banu Cavlak^a
Henk Corporaal^c Onur Mutlu^a
^aETH Zürich ^bAMD ^cEindhoven University of Technology

Nanopore sequencing is a widely-used high-throughput genome sequencing technology that can sequence long fragments of a genome. Nanopore sequencing generates noisy electrical signals that need to be converted into a standard string of DNA nucleotide bases (i.e., A, C, G, T) using a computational step called *basecalling*. The accuracy and speed of basecalling have critical implications for every subsequent step in genome analysis. Currently, basecallers are developed mainly based on deep learning techniques to provide high sequencing accuracy without considering the compute demands of such tools. We observe that state-of-the-art basecallers (i.e., Guppy, Bonito, Fast-Bonito) are slow, inefficient, and memory-hungry



How Does a Nanopore Machine Work?



- **Nanopore** is a nano-scale hole ($<20\text{nm}$).
- In nanopore sequencers, an **ionic current** passes through the nanopores
- When the DNA strand passes through the nanopore, the sequencer measures the **change in current**
- This change is used to identify the bases in the strand with the help of **different electrochemical structures** of the different bases

Accelerating Basecallers: Software Methods

- 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](#)]

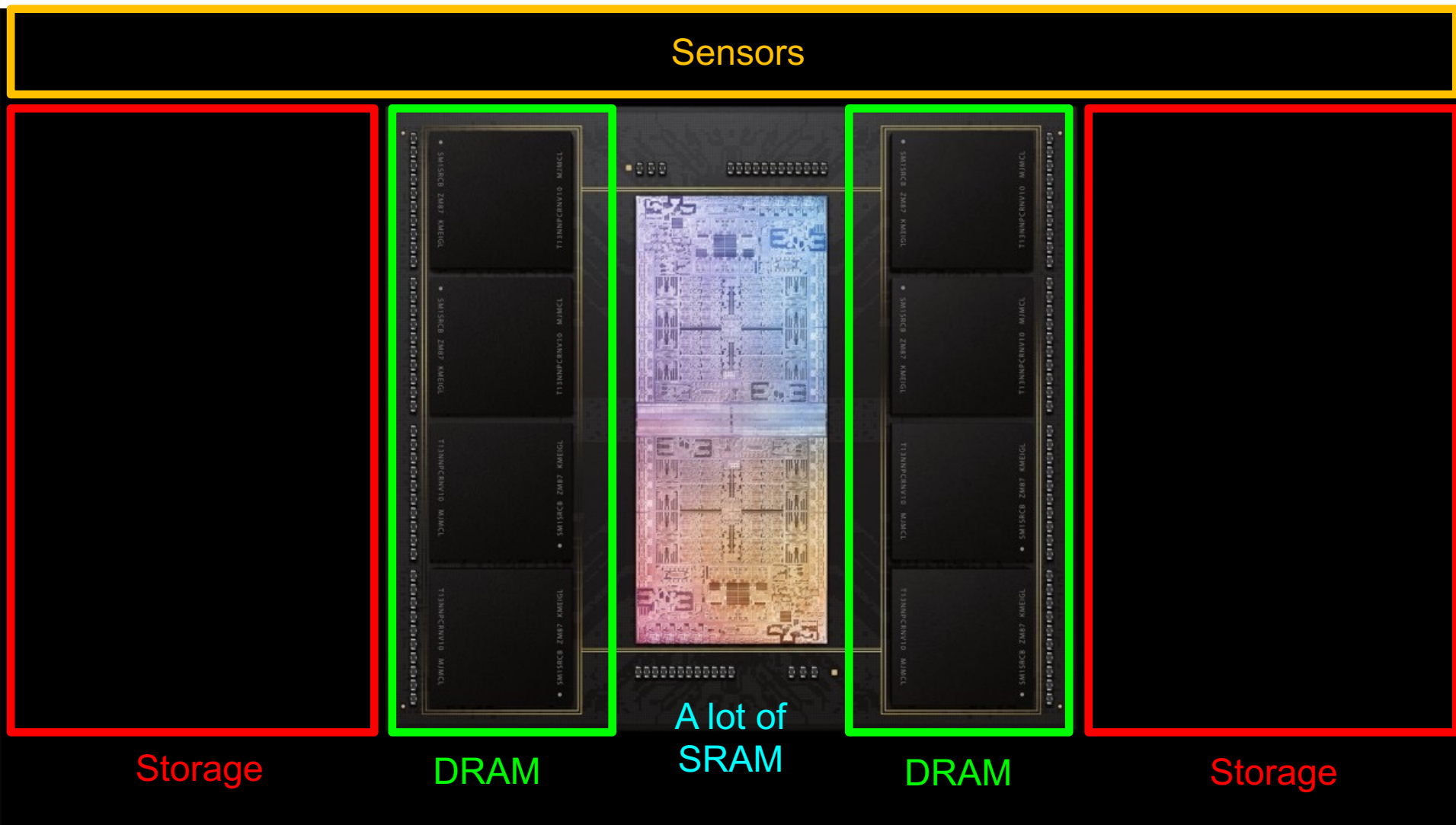
TargetCall: Eliminating the Wasted Computation in Basecalling via Pre-Basecalling Filtering

Meryem Banu Cavlak¹ Gagandeep Singh¹ Mohammed Alser¹ Can Firtina¹ Joël Lindegger¹
Mohammad Sadrosadati¹ Nika Mansouri Ghiasi¹ Can Alkan² Onur Mutlu¹
¹*ETH Zürich* ²*Bilkent University*

Agenda

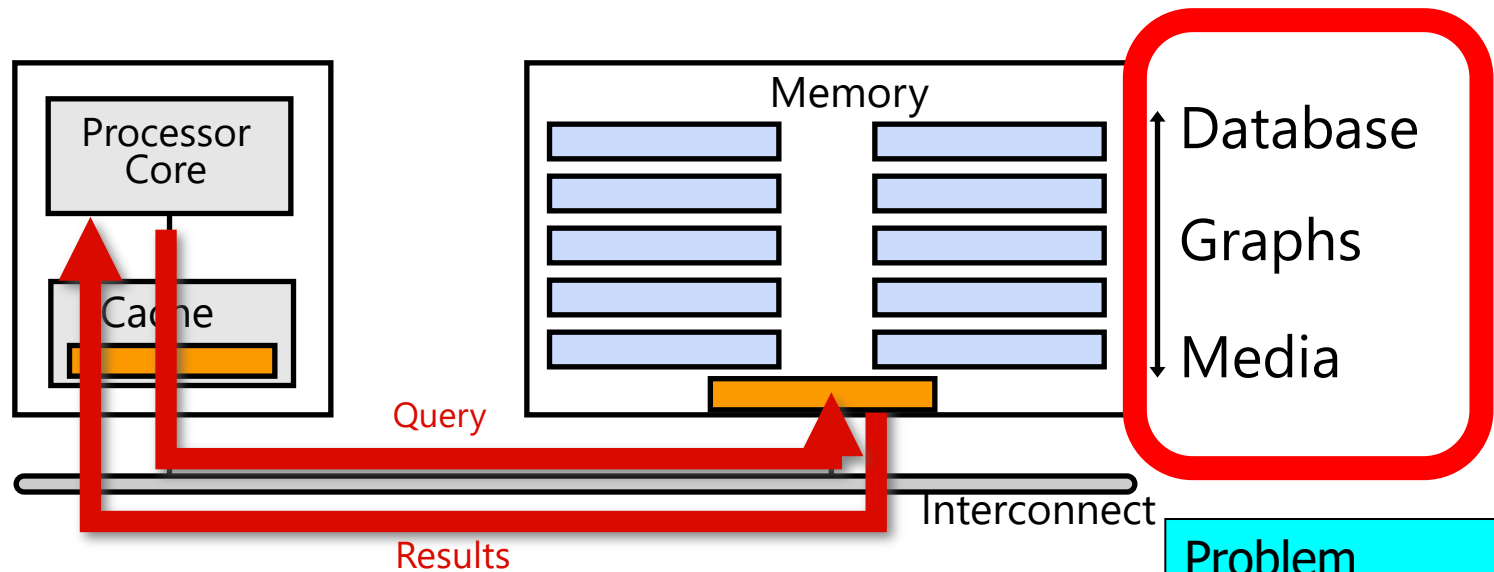
- The Problem: DNA Read Mapping
 - State-of-the-art Read Mapper Design
- Algorithmic Acceleration
 - Exploiting Structure of the Genome
 - Exploiting SIMD Instructions
- Hardware Acceleration
 - Specialized Architectures
 - Processing in Memory & Storage
- Future Opportunities: New Technologies & Applications

Process Data Where It Makes Sense



Apple M1 Ultra System (2022)

Goal: Processing Inside Memory



- Many questions ... How do we design the:
 - ❑ compute-capable memory & controllers?
 - ❑ processors & communication units?
 - ❑ software & hardware interfaces?
 - ❑ system software, compilers, languages?
 - ❑ algorithms & theoretical foundations?

Problem
Algorithm
Program/Language
System Software
SW/HW Interface
Micro-architecture
Logic
Devices
Electrons

Read Mapping & Filtering in Memory

We need to design
mapping & filtering algorithms
that fit processing-in-memory

Near-Memory Pre-Alignment Filtering

Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gomez-Luna, Henk Corporaal, Onur Mutlu,

“FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications”

IEEE Micro, 2021.

[[Source Code](#)]



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2021.04](#)

IEEE Micro

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)

Authors

[Gagandeep Singh](#), ETH Zürich, Zürich, Switzerland

[Mohammed Alser](#), ETH Zürich, Zürich, Switzerland

[Damla Senol Cali](#), Carnegie Mellon University, Pittsburgh, PA, USA

[Dionysios Diamantopoulos](#), Zürich Lab, IBM Research Europe, Rüschlikon, Switzerland

[Juan Gomez-Luna](#), ETH Zürich, Zürich, Switzerland

[Henk Corporaal](#), Eindhoven University of Technology, Eindhoven, The Netherlands

[Onur Mutlu](#), ETH Zürich, Zürich, Switzerland

◀	▶
Previous	Next
☰	Table of Contents
📖	Past Issues

Near-Memory SneakySnake

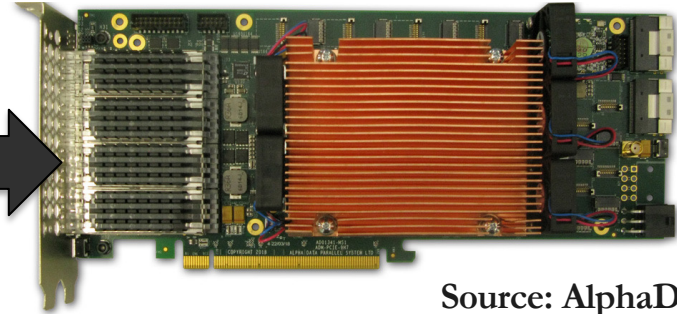
- Problem: Read mapping is heavily bottlenecked by data movement from main memory
- Solution: Perform read mapping near where data resides using specialized logic
- We carefully redesign the accelerator logic of SneakySnake to exploit near-memory computation capability on real FPGA boards that use HBM (high-bandwidth memory)
- **Near-memory SneakySnake** improves **performance** and **energy efficiency** by 27.4× and 133×, respectively, over a 16-core (64-thread) IBM POWER9 CPU

Near-Memory Acceleration using FPGAs



Source: IBM

IBM POWER9 CPU



Source: AlphaData

HBM-based FPGA board

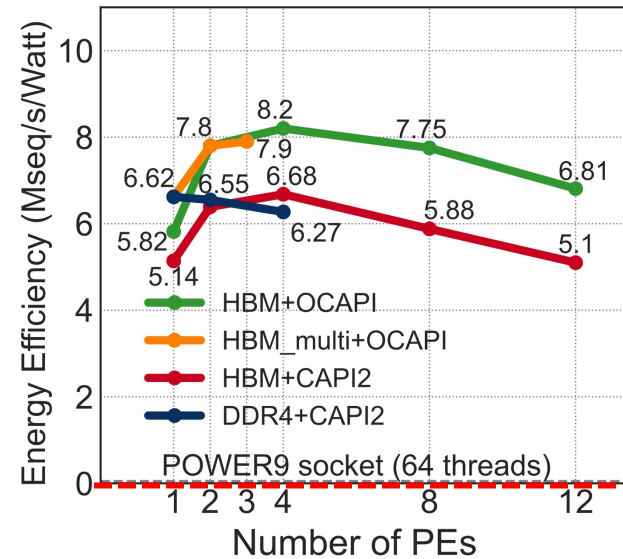
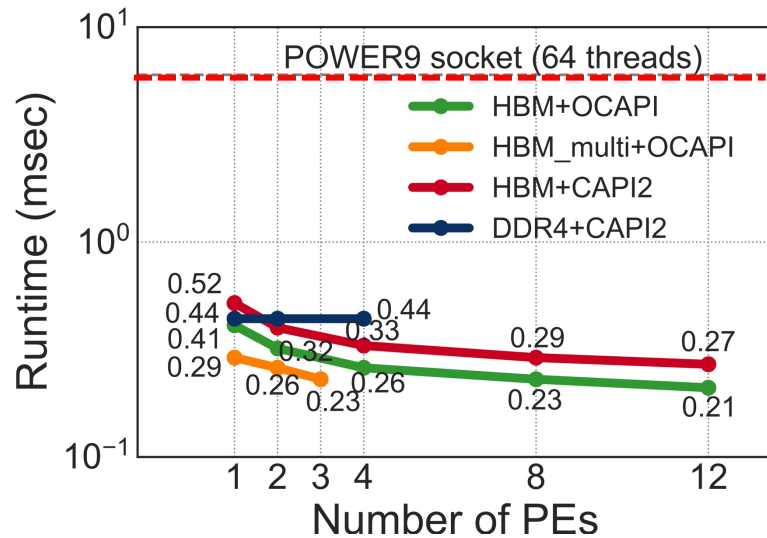
Near-HBM FPGA-based accelerator

Two communication technologies: CAPI2 and OCAPI

Two memory technologies: DDR4 and HBM

Two workloads: Weather Modeling and Genome Analysis

Performance & Energy Greatly Improve



5-27× performance vs. a 16-core (64-thread) IBM POWER9 CPU

12-133× energy efficiency vs. a 16-core (64-thread) IBM POWER9 CPU

HBM alleviates memory bandwidth contention vs. DDR4

More On Near-Memory SneakySnake

Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gomez-Luna, Henk Corporaal, Onur Mutlu,

"FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications"

IEEE Micro, 2021.

[[Source Code](#)]



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2021.04](#)

IEEE Micro

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](#)

Authors

[Gagandeep Singh](#), ETH Zürich, Zürich, Switzerland

[Mohammed Alser](#), ETH Zürich, Zürich, Switzerland

[Damla Senol Cali](#), Carnegie Mellon University, Pittsburgh, PA, USA

[Dionysios Diamantopoulos](#), Zürich Lab, IBM Research Europe, Rüschlikon, Switzerland

[Juan Gomez-Luna](#), ETH Zürich, Zürich, Switzerland

[Henk Corporaal](#), Eindhoven University of Technology, Eindhoven, The Netherlands

[Onur Mutlu](#), ETH Zürich, Zürich, Switzerland

◀	▶
Previous	Next
☰	Table of Contents
📖	Past Issues

Location Filtering in 3D-Stacked PIM

- Jeremie S. Kim, Damla Senol Cali, Hongyi Xin, Donghyuk Lee, Saugata Ghose, Mohammed Alser, Hasan Hassan, Oguz Ergin, Can Alkan, and Onur Mutlu,
["GRIM-Filter: Fast Seed Location Filtering in DNA Read Mapping Using Processing-in-Memory Technologies"](#)
[*BMC Genomics*](#), 2018.
Proceedings of the 16th Asia Pacific Bioinformatics Conference (APBC), Yokohama, Japan, January 2018.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Source Code](#)]
[[arxiv.org Version \(pdf\)](#)]
[[Talk Video at AACBB 2019](#)]

Research | [Open Access](#) | [Published: 09 May 2018](#)

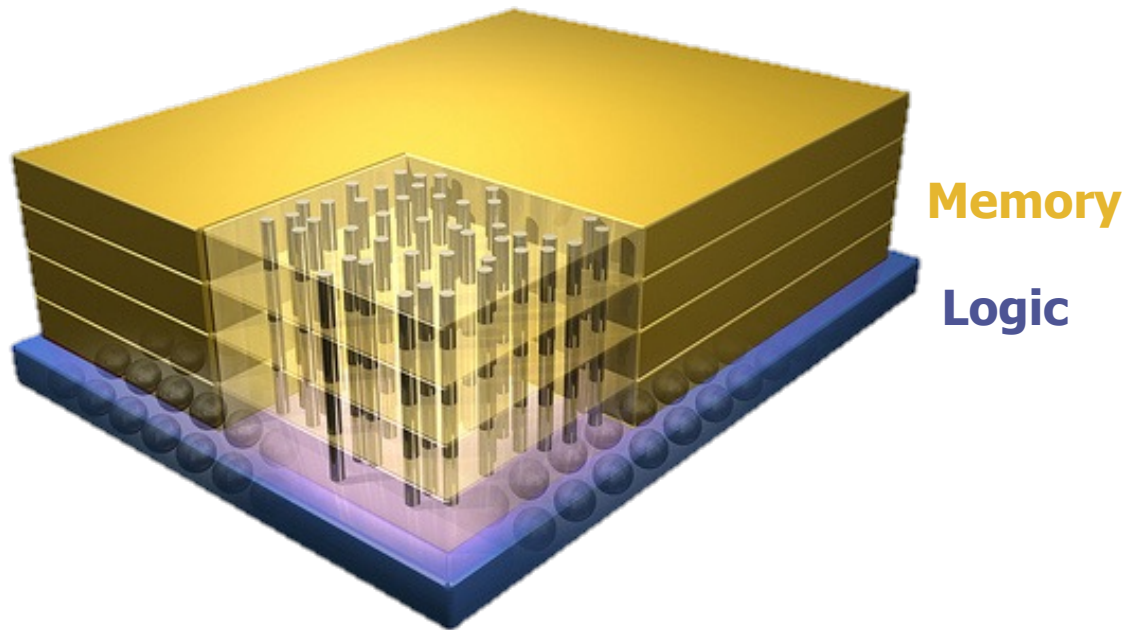
GRIM-Filter: Fast seed location filtering in DNA read mapping using processing-in-memory technologies

[Jeremie S. Kim](#) ✉, [Damla Senol Cali](#), [Hongyi Xin](#), [Donghyuk Lee](#), [Saugata Ghose](#), [Mohammed Alser](#),
[Hasan Hassan](#), [Oguz Ergin](#), [Can Alkan](#) ✉ & [Onur Mutlu](#) ✉

[BMC Genomics](#) **19**, Article number: 89 (2018) | [Cite this article](#)

4340 Accesses | **39** Citations | **9** Altmetric | [Metrics](#)

Opportunity: 3D-Stacked Logic+Memory



Other "True 3D" technologies
under development

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)]

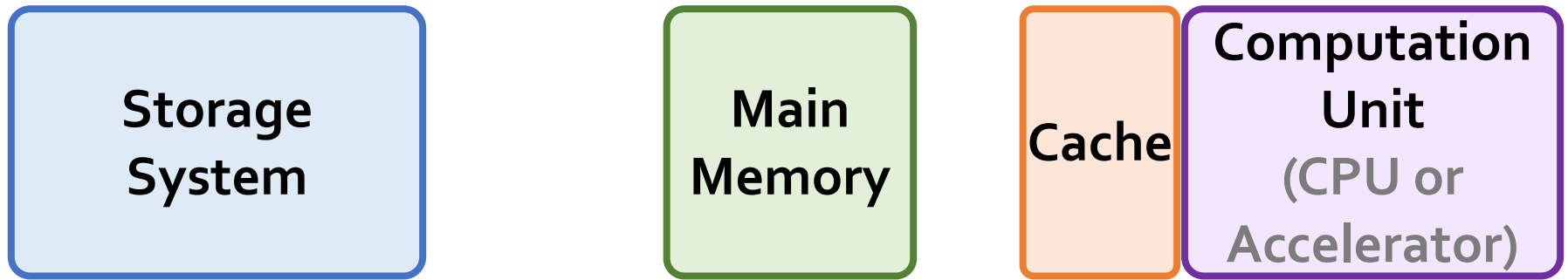
GenStore: A High-Performance In-Storage Processing System for Genome Sequence Analysis

Nika Mansouri Ghiasi¹ Jisung Park¹ Harun Mustafa¹ Jeremie Kim¹ Ataberk Olgun¹
Arvid Gollwitzer¹ Damla Senol Cali² Can Firtina¹ Haiyu Mao¹ Nour Almadhoun Alserr¹
Rachata Ausavarungnirun³ Nandita Vijaykumar⁴ Mohammed Alser¹ Onur Mutlu¹

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

Genome Sequence Analysis

Data Movement from Storage

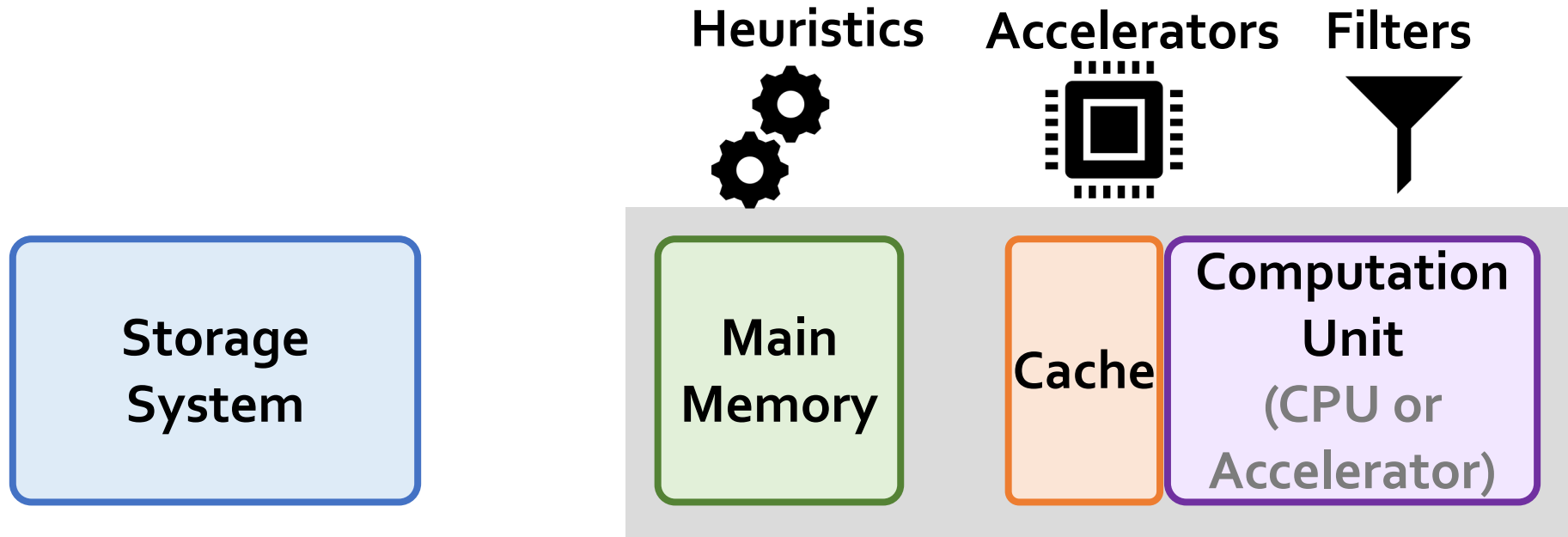


Computation overhead



Data movement overhead

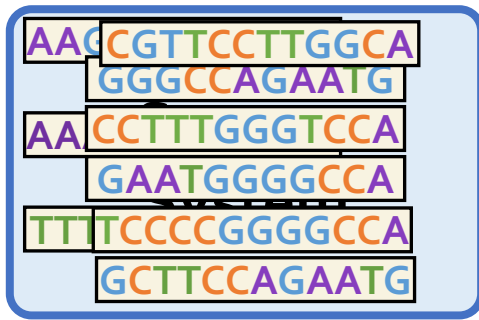
Accelerating Genome Sequence Analysis



Key Idea



*Filter reads that do **not** require alignment inside the storage system*



Filtered Reads

**Main
Memory**

Cache

**Computation
Unit**
(CPU or
Accelerator)

Exactly-matching reads

Do not need expensive approximate string matching during alignment

Non-matching reads

Do not have potential matching locations and can skip alignment

Filtering Opportunities

- Sequencing machines produce one of two kinds of reads
 - Short reads: highly accurate and short
 - Long reads: less accurate and long

Reads that do not require the expensive alignment step:

Exactly-matching reads

Do not need expensive approximate string matching during alignment

- Low sequencing error rates (short reads) combined with
- Low genetic variation

Non-matching reads

Do not have potential matching locations, so they skip alignment

- High sequencing error rates (long reads) or
- High genetic variation (short or long reads)

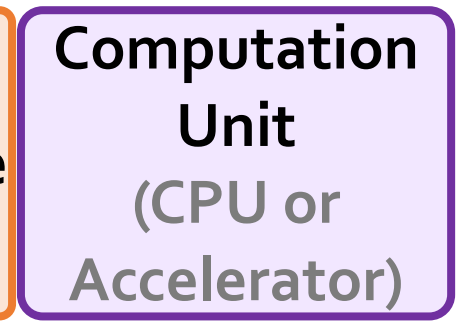
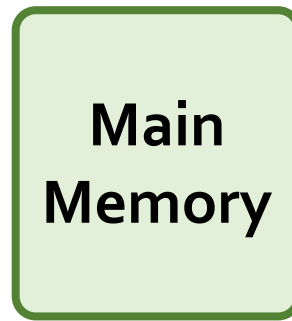
Challenges



*Filter reads that do **not** require alignment
inside the storage system*



Filtered Reads



Read mapping workloads can exhibit different behavior

There are **limited hardware resources**
in the storage system

GenStore



*Filter reads that do **not** require alignment
inside the storage system*

GenStore-Enabled
Storage
System

Main
Memory

Cache

Computation
Unit
(CPU or
Accelerator)



Computation overhead



Data movement overhead

GenStore provides significant speedup (1.4x - 33.6x) and
energy reduction (3.9x - 29.2x) at low cost

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 Ghiasi¹ Jisung Park¹ Harun Mustafa¹ Jeremie Kim¹ Ataberk Olgun¹
Arvid Gollwitzer¹ Damla Senol Cali² Can Firtina¹ Haiyu Mao¹ Nour Almadhoun Alserr¹
Rachata Ausavarungnirun³ Nandita Vijaykumar⁴ Mohammed Alser¹ Onur Mutlu¹

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

PIM Review and Open Problems

A Modern Primer on Processing in Memory

Onur Mutlu^{a,b}, Saugata Ghose^{b,c}, Juan Gómez-Luna^a, Rachata Ausavarungnirun^d

SAFARI Research Group

^a*ETH Zürich*

^b*Carnegie Mellon University*

^c*University of Illinois at Urbana-Champaign*

^d*King Mongkut's University of Technology North Bangkok*

Onur Mutlu, Saugata Ghose, Juan Gomez-Luna, and Rachata Ausavarungnirun,
"A Modern Primer on Processing in Memory"
*Invited Book Chapter in **Emerging Computing: From Devices to Systems - Looking Beyond Moore and Von Neumann**, Springer, to be published in 2021.*

PIM Review and Open Problems (II)

A Workload and Programming Ease Driven Perspective of Processing-in-Memory

Saugata Ghose[†] Amirali Boroumand[†] Jeremie S. Kim^{†§} Juan Gómez-Luna[§] Onur Mutlu^{§†}

[†]*Carnegie Mellon University*

[§]*ETH Zürich*

Saugata Ghose, Amirali Boroumand, Jeremie S. Kim, Juan Gomez-Luna, and Onur Mutlu,

"Processing-in-Memory: A Workload-Driven Perspective"

Invited Article in IBM Journal of Research & Development, Special Issue on Hardware for Artificial Intelligence, to appear in November 2019.

[Preliminary arXiv version]

More on Processing-in-Memory

- Onur Mutlu,

"Memory-Centric Computing Systems"

Invited Tutorial at *66th International Electron Devices Meeting (IEDM)*, Virtual, 12 December 2020.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Executive Summary Slides \(pptx\)](#) ([pdf](#))]

[[Tutorial Video](#) (1 hour 51 minutes)]

[[Executive Summary Video](#) (2 minutes)]

[[Abstract and Bio](#)]

[[Related Keynote Paper from VLSI-DAT 2020](#)]

[[Related Review Paper on Processing in Memory](#)]

<https://www.youtube.com/watch?v=H3sEaINPBOE>

Memory-Centric Computing Systems



Onur Mutlu

omutlu@gmail.com

<https://people.inf.ethz.ch/omutlu>

12 December 2020

IEDM Tutorial

SAFARI

ETH zürich

Carnegie Mellon



0:06 / 1:51:05



IEDM 2020 Tutorial: Memory-Centric Computing Systems, Onur Mutlu, 12 December 2020

1,641 views • Dec 23, 2020

48 0 SHARE SAVE ...



Onur Mutlu Lectures
13.9K subscribers

<https://www.youtube.com/watch?v=H3sEaINPBOE>

ANALYTICS

EDIT VIDEO

<https://www.youtube.com/onurmutlulectures>

Processing in Memory: Faster & Low Energy

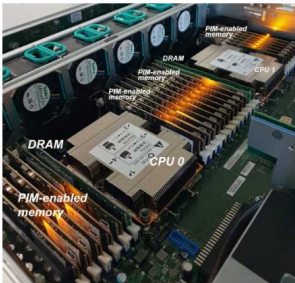
In-Memory-Computing: faster and more energy efficient

10.03.2022 | Sustainability, Industry Projects
By: Anna Julia Schlegel

Big Data applications require high computing performance while consuming as little power as possible. Current computer systems are reaching their limits in both areas. Professor Onur Mutlu is working on alternative systems and has just received the Intel 2021 Outstanding Researcher Award for his work.

You may have heard that Moore's law is coming to an end. This empirical observation states that computers double their performance approximately every 2 years. Alternative approaches to improve the efficiency of computing are therefore in great demand. Prof. Onur Mutlu, whose research interests include hardware/software co-design at ETH Zurich, is pursuing the approach of combining computing and memory. **Processing-in-memory (PIM) computing** makes Big Data applications such as genome analysis both substantially faster and more energy-efficient.

Recently, the Grenoble-based company UPMEM launched the first commercially available PIM architecture. Instead of a processor or CPUs (Central Processing Units), it contains DPUs (DRAM Processing Units), which are memory elements that also process the data. Mutlu and his research group have characterised, analysed, and tested the new system and compared it with a previous state-of-the-art system with CPUs. They have learned that the novel system makes computing up to 23 times faster and five times more energy efficient. The new system is most interesting for data-intensive applications - specific examples include gene analysis or weather forecast models. "Not bad for the first commercial version of a processing-in-memory system," Mutlu says, "compared to a processor-centric CPU system that has been optimised for decades."



The UPMEM Processing-In-Memory-System.
(Source: Onur Mutlu)

Much faster and more energy-efficient

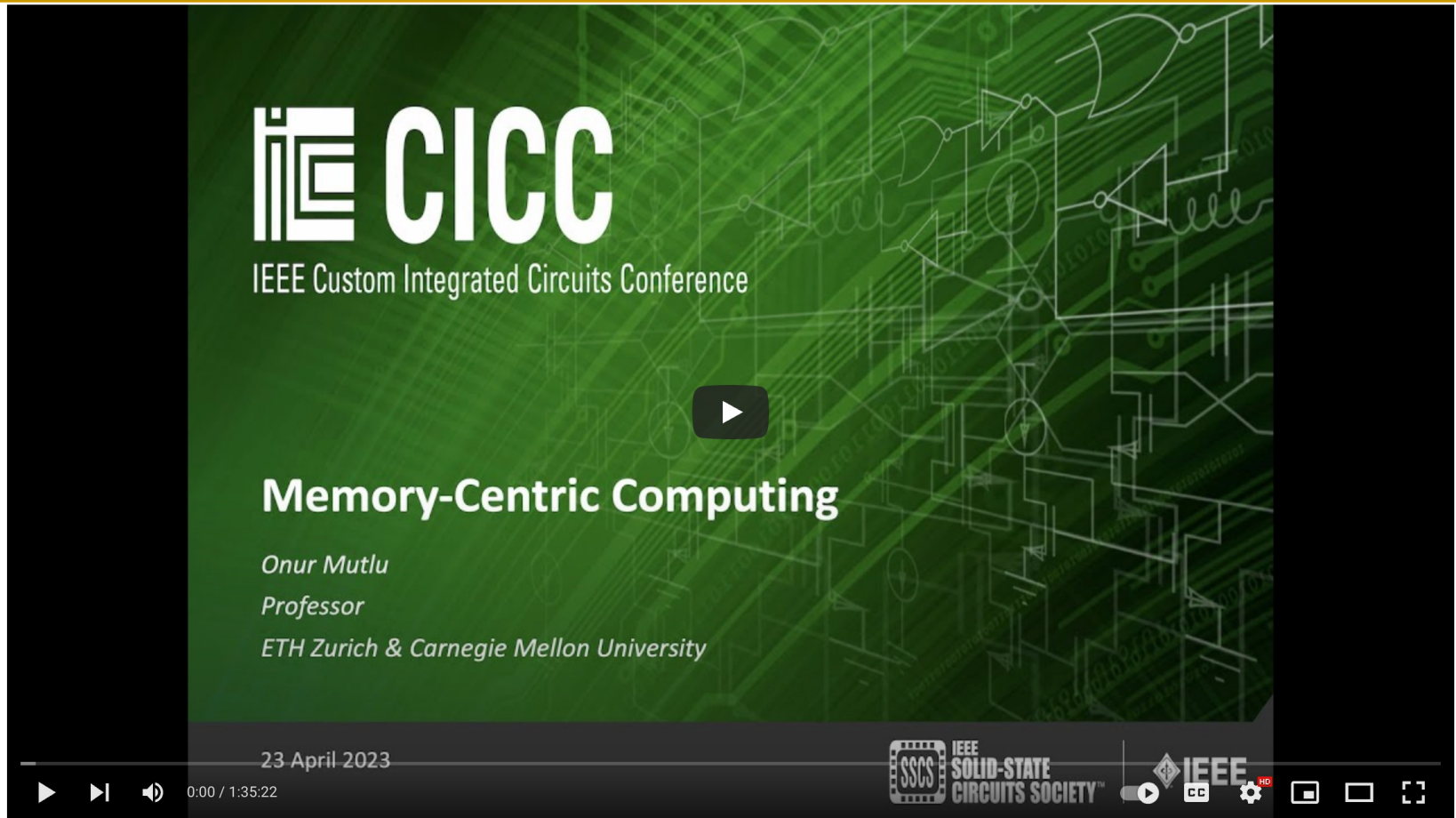
Mutlu and his colleagues have tested the novel system for applications in the fields of data analysis, databases, bioinformatics, image- and video analysis, and neural networks, among others. The PIM-system is best suited for workloads requiring little communication between DPUs (e.g. database and image applications) and primarily simple arithmetic operations (e.g. video analytics or data filtering). "We expect that as these systems evolve, they will become even faster and more energy efficient, and their applications will become even more diverse," Mutlu reckons.

SUSTAINABILITY • INDUSTRY PROJECTS

In-Memory-Computing: faster and more energy efficient

<https://ethz.ch/en/industry/industry/news/data/2022/03/mehr-daten-schneller-und-energiesparender-verarbeiten.html>

Tutorial on Processing in Memory



IEEE CICC Educational Session Talk on Memory-Centric Computing (Prof. Onur Mutlu)



Onur Mutlu Lectures
32.8K subscribers

Analytics

Edit video

20



Share

Download

Clip

Save



411 views Streamed 2 weeks ago

IEEE CICC Educational Session Talk on Memory-Centric Computing
Presenter: Professor Onur Mutlu (<https://people.inf.ethz.ch/omutlu/>)

<https://www.youtube.com/watch?v=4x9nujJtqjM>

<https://www.youtube.com/onurmutlulectures>

- Collaborative EU Horizon Project (2022-2026)
 - Bilkent, IBM Research, UPMEM, CNRS, Institut Pasteur, Bar-Ilan, Technion
 - <http://www.biopim.eu/>
 - **BioPIM** project aims to leverage the emerging processing-in-memory (PIM) technologies to enable powerful edge computing.
 - Synergistic with this task
 - We will focus on co-designing algorithms and data structures commonly used in bioinformatics together with several types of PIM architectures to obtain the highest benefit in cost, energy, and time savings.
 - BioPIM will also impact other fields that employ similar algorithms.
 - Our designs and algorithms will not be limited to cheap hardware, and they will impact computation efficiency on all forms of computing environments including cloud platforms.
 - The targeted breakthrough of **BioPIM** is to invent and leverage in-memory computing architectures to fundamentally improve the performance and energy efficiency of various important bioinformatics algorithms to make mobile genomics a reality

PIM Course (Fall 2022)

■ Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory

■ Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=processing_in_memory

■ Youtube Livestream (Fall 2022):

- <https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQOVD0GOBrnKy>

■ Youtube Livestream (Spring 2022):

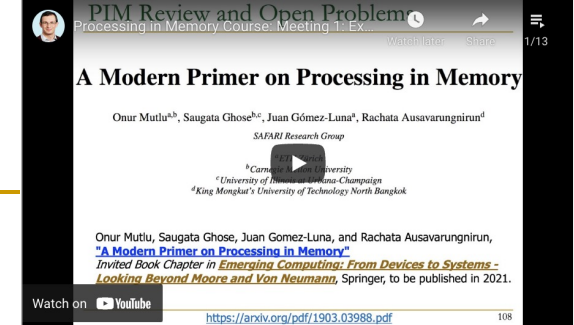
- <https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUK9EsXKhQKRPyX>

■ Project course

- Taken by Bachelor's/Master's students
- Processing-in-Memory lectures
- Hands-on research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>

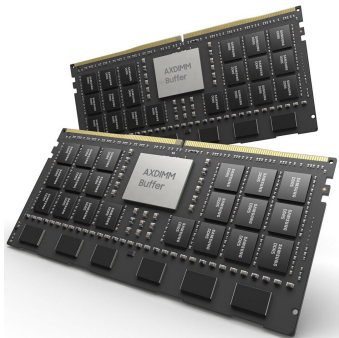
SAFARI



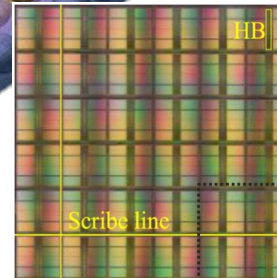
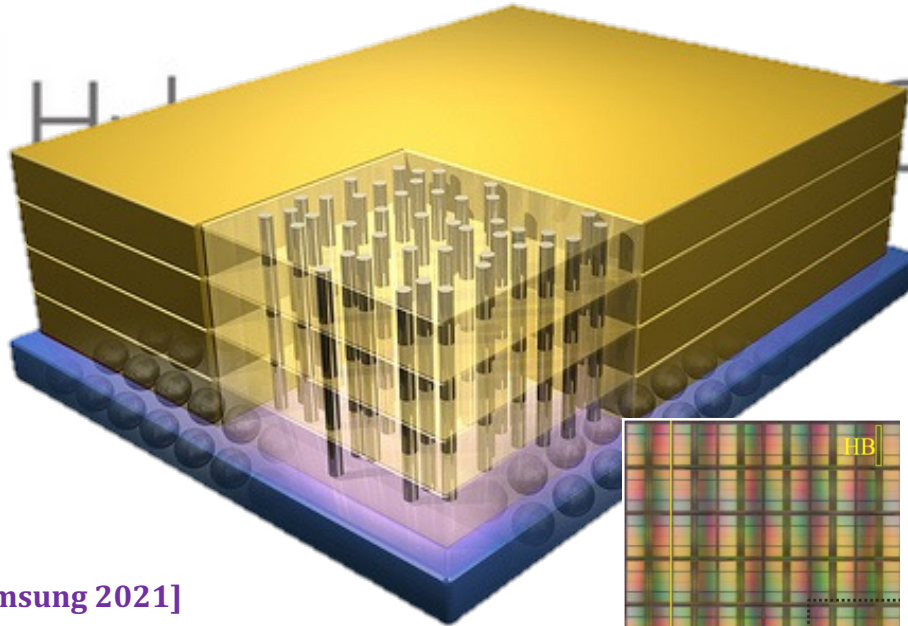
Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	10.03 Thu.	YouTube Live	M1: P&S PIM Course Presentation (PDF) (PPT)	Required Materials Recommended Materials	HW 0 Out
W2	15.03 Tue.		Hands-on Project Proposals		
	17.03 Thu.	YouTube Premiere	M2: Real-world PIM: UPMEM PIM (PDF) (PPT)		
W3	24.03 Thu.	YouTube Live	M3: Real-world PIM: Microbenchmarking of UPMEM PIM (PDF) (PPT)		
W4	31.03 Thu.	YouTube Live	M4: Real-world PIM: Samsung HBM-PIM (PDF) (PPT)		
W5	07.04 Thu.	YouTube Live	M5: How to Evaluate Data Movement Bottlenecks (PDF) (PPT)		
W6	14.04 Thu.	YouTube Live	M6: Real-world PIM: SK Hynix AIM (PDF) (PPT)		
W7	21.04 Thu.	YouTube Premiere	M7: Programming PIM Architectures (PDF) (PPT)		
W8	28.04 Thu.	YouTube Premiere	M8: Benchmarking and Workload Suitability on PIM (PDF) (PPT)		
W9	05.05 Thu.	YouTube Premiere	M9: Real-world PIM: Samsung AxDIMM (PDF) (PPT)		
W10	12.05 Thu.	YouTube Premiere	M10: Real-world PIM: Alibaba HB-PNM (PDF) (PPT)		
W11	19.05 Thu.	YouTube Live	M11: SpMV on a Real PIM Architecture (PDF) (PPT)		
W12	26.05 Thu.	YouTube Live	M12: End-to-End Framework for Processing-using-Memory (PDF) (PPT)		
W13	02.06 Thu.	YouTube Live	M13: Bit-Serial SIMD Processing using DRAM (PDF) (PPT)		
W14	09.06 Thu.	YouTube Live	M14: Analyzing and Mitigating ML Inference Bottlenecks (PDF) (PPT)		
W15	15.06 Thu.	YouTube Live	M15: In-Memory HTAP Databases with HW/SW Co-design (PDF) (PPT)		
W16	23.06 Thu.	YouTube Live	M16: In-Storage Processing for Genome Analysis (PDF) (PPT)		
W17	18.07 Mon.	YouTube Premiere	M17: How to Enable the Adoption of PIM? (PDF) (PPT)		
W18	09.08 Tue.	YouTube Premiere	SS1: ISVLSI 2022 Special Session on PIM (PDF & PPT)		

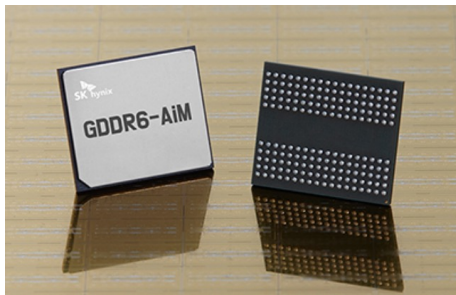
Processing-in-Memory Landscape Today



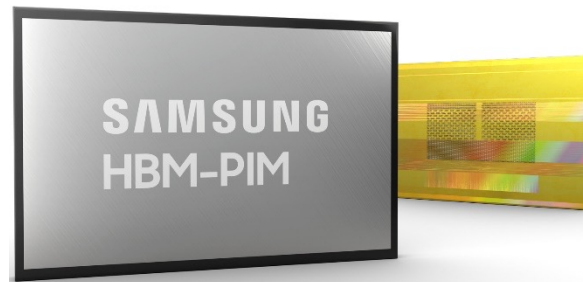
[Samsung 2021]



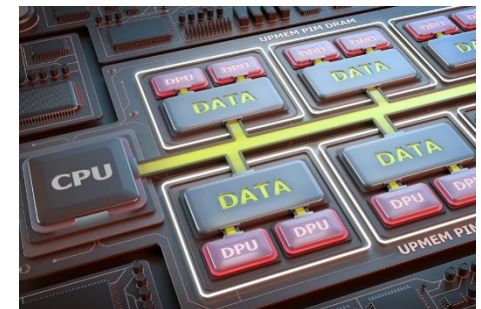
[Alibaba 2022]



[SK Hynix 2022]



[Samsung 2021]

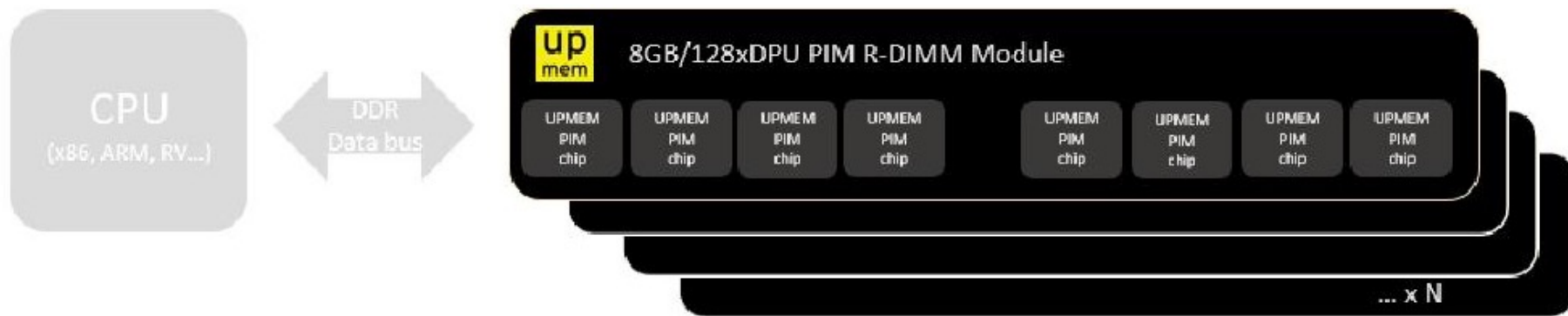


[UPMEM 2019]

This does not include many experimental chips and startups

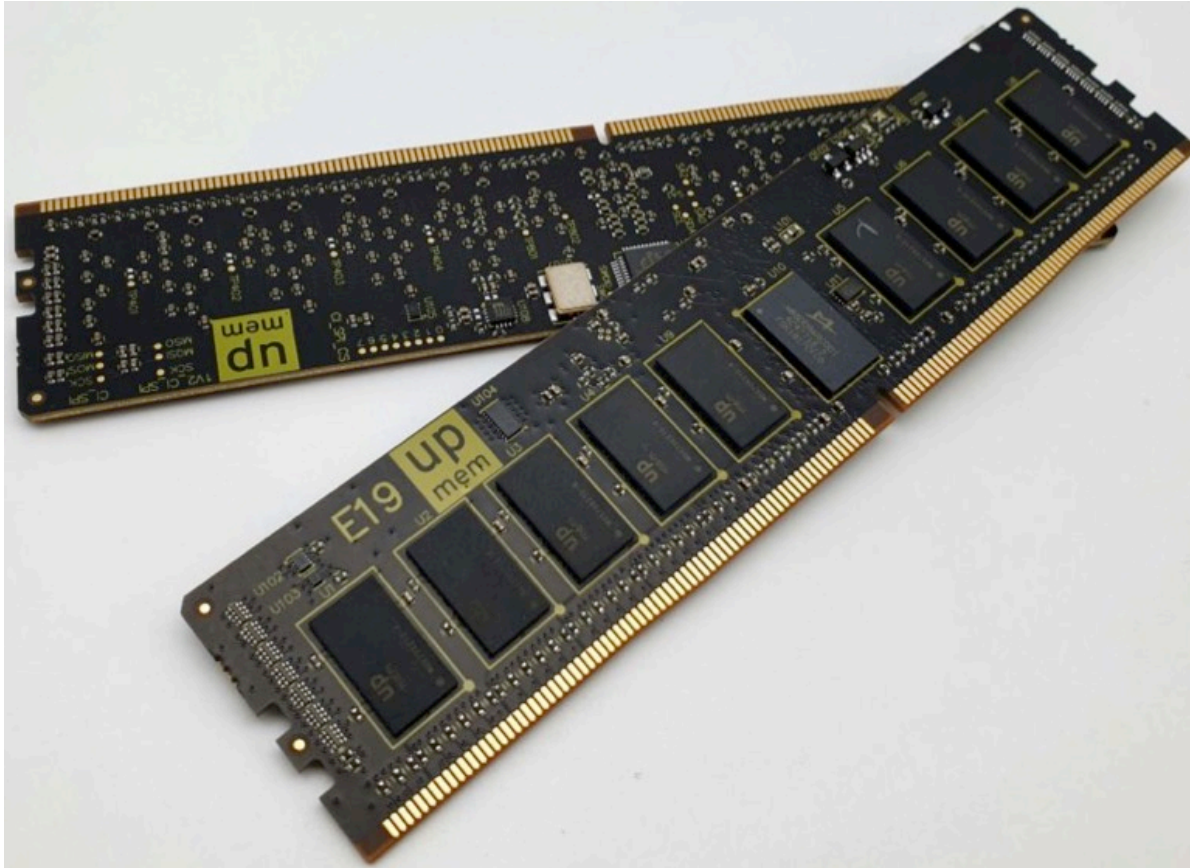
UPMEM Processing-in-DRAM Engine (2019)

- **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

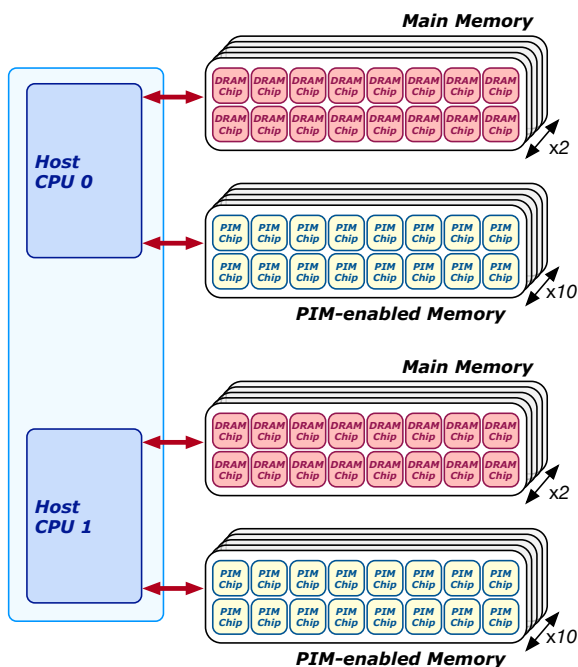


UPMEM Memory Modules

- E19: 8 chips DIMM (1 rank). DPUs @ 267 MHz
- P21: 16 chips DIMM (2 ranks). DPUs @ 350 MHz



2,560-DPU Processing-in-Memory System



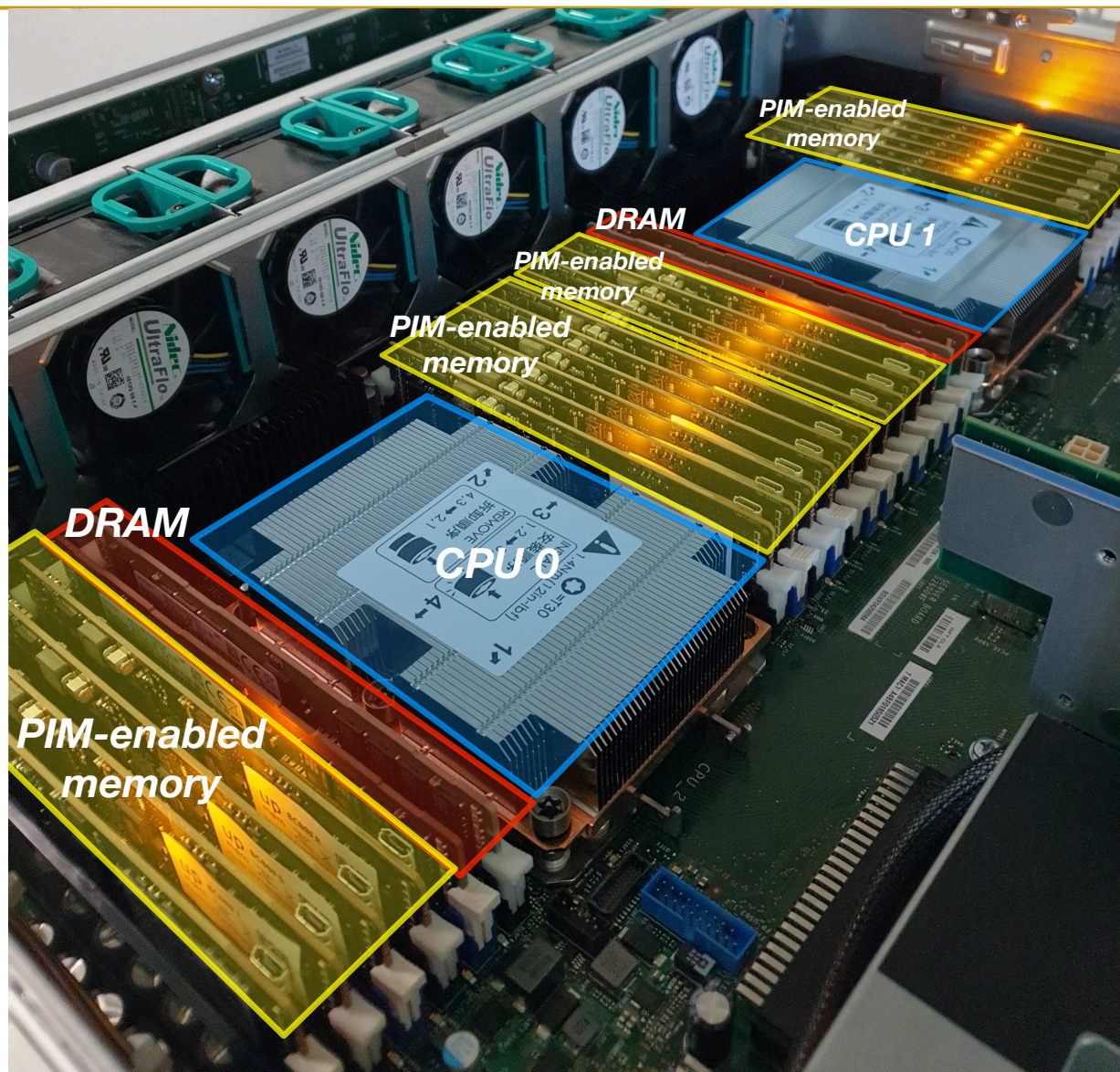
Benchmarking a New Paradigm: An Experimental Analysis of a Real Processing-in-Memory Architecture

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland
IZZAT EL HAJJ, American University of Beirut, Lebanon
IVAN FERNANDEZ, ETH Zürich, Switzerland and University of Malaga, Spain
CHRISTINA GIANNOULA, ETH Zürich, Switzerland and NTUA, Greece
GERALDO F. OLIVEIRA, ETH Zürich, Switzerland
ONUR MUTLU, ETH Zürich, Switzerland

Many modern workloads, such as neural networks, databases, and graph processing, are fundamentally memory-bound. For such workloads, the data movement between main memory and CPU cores imposes a significant overhead in terms of both latency and energy. A major reason is that this communication happens through a narrow bus with high latency and limited bandwidth, and the low data reuse in memory-bound workloads is insufficient to amortize the cost of main memory access. Fundamentally addressing this *data movement bottleneck* requires a paradigm where the memory system assumes an active role in computing by integrating processing capabilities. This paradigm is known as *processing-in-memory (PIM)*.

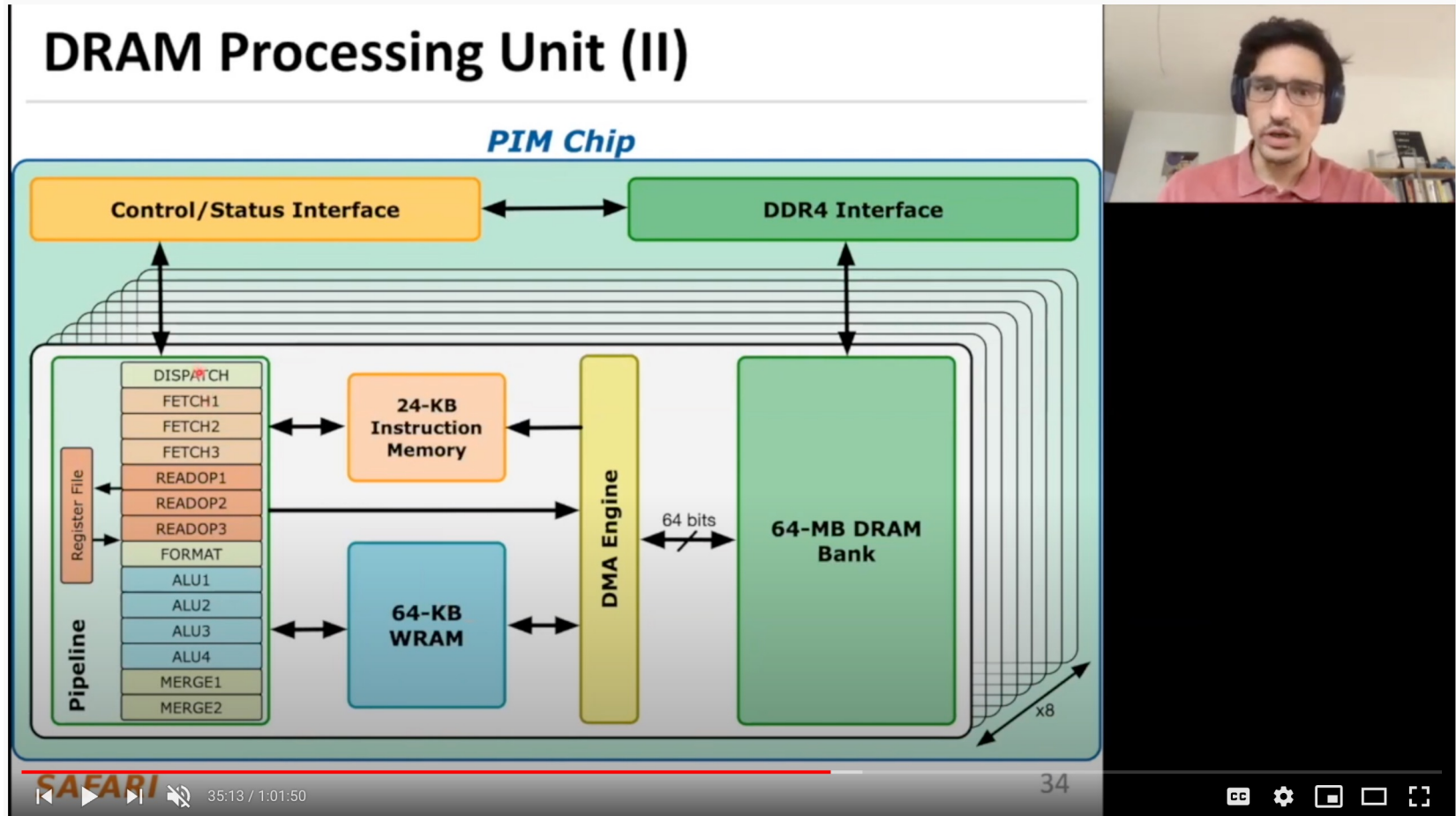
Recent research explores different forms of PIM architectures, motivated by the emergence of new 3D-stacked memory technologies that integrate memory with a logic layer where processing elements can be easily placed. Past works evaluate these architectures in simulation or, at best, with simplified hardware prototypes. In contrast, the UPMEM company has designed and manufactured the first publicly-available real-world PIM architecture. The UPMEM PIM architecture combines traditional DRAM memory arrays with general-purpose in-order cores, called *DRAM Processing Units (DPUs)*, integrated in the same chip.

This paper provides the first comprehensive analysis of the first publicly-available real-world PIM architecture. We make two key contributions. First, we conduct an experimental characterization of the UPMEM-based PIM system using microbenchmarks to assess various architecture limits such as compute throughput and memory bandwidth, yielding new insights. Second, we present *PrIM (Processing-In-Memory benchmarks)*, a benchmark suite of 16 workloads from different application domains (e.g., dense/sparse linear algebra, databases, data analytics, graph processing, neural networks, bioinformatics, image processing), which we identify as memory-bound. We evaluate the performance and scaling characteristics of PrIM benchmarks on the UPMEM PIM architecture, and compare their performance and energy consumption to their state-of-the-art CPU and GPU counterparts. Our extensive evaluation conducted on two real UPMEM-based PIM systems with 640 and 2,560 DPUs provides new insights about suitability of different workloads to the PIM system, programming recommendations for software designers, and suggestions and hints for hardware and architecture designers of future PIM systems.



<https://arxiv.org/pdf/2105.03814.pdf>

More on the UPMEM PIM System



ETH ZÜRICH HAUPTGEBÄUDE

Computer Architecture - Lecture 12d: Real Processing-in-DRAM with UPMEM (ETH Zürich, Fall 2020)

1,120 views • Oct 31, 2020

30 0 SHARE SAVE ...



Onur Mutlu Lectures
16.7K subscribers

ANALYTICS

EDIT VIDEO

<https://www.youtube.com/watch?v=Sscy1Wrr22A&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=26>

SRC TECHCON Presentation

■ Dr. Juan Gomez-Luna

- Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-in-Memory Hardware
- Based on two major works
 - <https://arxiv.org/pdf/2105.03814.pdf>
 - <https://arxiv.org/pdf/2207.07886.pdf>



Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-In-Memory Hardware

Year: 2021, Pages: 1-7

DOI Bookmark: [10.1109/IGSC54211.2021.9651614](https://doi.org/10.1109/IGSC54211.2021.9651614)

Authors

Juan Gómez-Luna, ETH Zürich

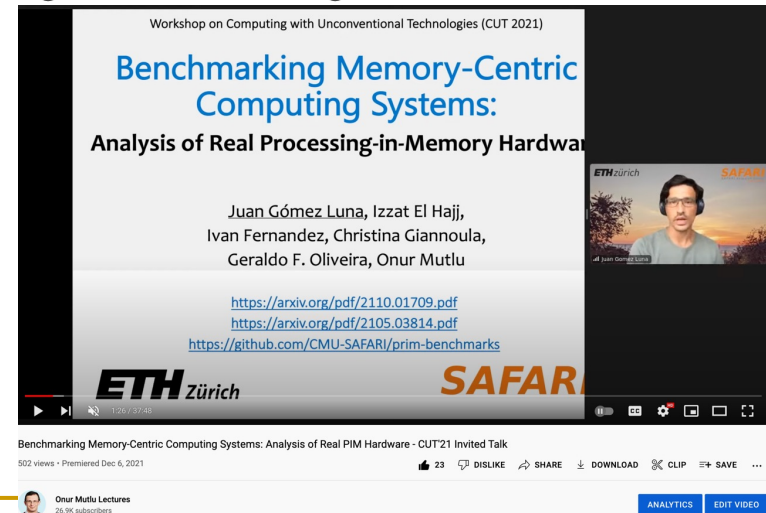
Izzat El Hajj, American University of Beirut

Ivan Fernandez, University of Malaga

Christina Giannoula, National Technical University of Athens

Geraldo F. Oliveira, ETH Zürich

Onur Mutlu, ETH Zürich



UPMEM PIM System Summary & Analysis

- Juan Gomez-Luna, Izzat El Hajj, Ivan Fernandez, Christina Giannoula, Geraldo F. Oliveira, and Onur Mutlu,
"Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-in-Memory Hardware"
*Invited Paper at Workshop on Computing with Unconventional Technologies (**CUT**), Virtual, October 2021.*
[[arXiv version](#)]
[[PrIM Benchmarks Source Code](#)]
[[Slides \(pptx\)](#) ([pdf](#))]
[[Talk Video](#) (37 minutes)]
[[Lightning Talk Video](#) (3 minutes)]

Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-in-Memory Hardware

Juan Gómez-Luna	Izzat El Hajj	Ivan Fernandez	Christina Giannoula	Geraldo F. Oliveira	Onur Mutlu
<i>ETH Zürich</i>	<i>American University of Beirut</i>	<i>University of Malaga</i>	<i>National Technical University of Athens</i>	<i>ETH Zürich</i>	<i>ETH Zürich</i>

ML Training on Real PIM Systems

- Juan Gómez Luna, Yuxin Guo, Sylvan Brocard, Julien Legriel, Remy Cimadomo, Geraldo F. Oliveira, Gagandeep Singh, and Onur Mutlu,
"Evaluating Machine Learning Workloads on Memory-Centric Computing Systems"
Proceedings of the 2023 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Raleigh, North Carolina, USA, April 2023.
[[arXiv version](#), 16 July 2022.]
[[PIM-ML Source Code](#)]
Best paper session.

An Experimental Evaluation of Machine Learning Training on a Real Processing-in-Memory System

Juan Gómez-Luna¹ Yuxin Guo¹ Sylvan Brocard² Julien Legriel²
Remy Cimadomo² Geraldo F. Oliveira¹ Gagandeep Singh¹ Onur Mutlu¹
¹ETH Zürich ²UPMEM

<https://github.com/CMU-SAFARI/pim-ml>

Sequence Alignment on Real PIM Systems

- Safaa Diab, Amir Nassereldine, Mohammed Alser, Juan Gómez Luna, Onur Mutlu, and Izzat El Hajj,
"A Framework for High-throughput Sequence Alignment using Real Processing-in-Memory Systems"
Bioinformatics, [published online on] 27 March 2023.
[[Online link at Bioinformatics Journal](#)]
[[arXiv preprint](#)]
[[AiM Source Code](#)]

A Framework for High-throughput Sequence Alignment using Real Processing-in-Memory Systems

Safaa Diab¹ Amir Nassereldine¹ Mohammed Alser² Juan Gómez Luna²
Onur Mutlu² Izzat El Hajj¹

¹American University of Beirut ²ETH Zürich

<https://github.com/CMU-SAFARI/alignment-in-memory>



Summary

- Sequence alignment on traditional systems is limited by the **memory bandwidth bottleneck**
- **Processing-in-memory (PIM)** overcomes this bottleneck by placing cores near the memory
- Our framework, **Alignment-in-Memory (AIM)**, is a PIM framework that supports multiple alignment algorithms (NW, SWG, GenASM, WFA)
 - Implemented on UPMEM, the first real PIM system
- Results show **substantial speedups over both CPUs (1.8X-28X) and GPUs (1.2X-2.7X)**
- AIM is available at:
 - <https://github.com/CMU-SAFARI/alignment-in-memory>

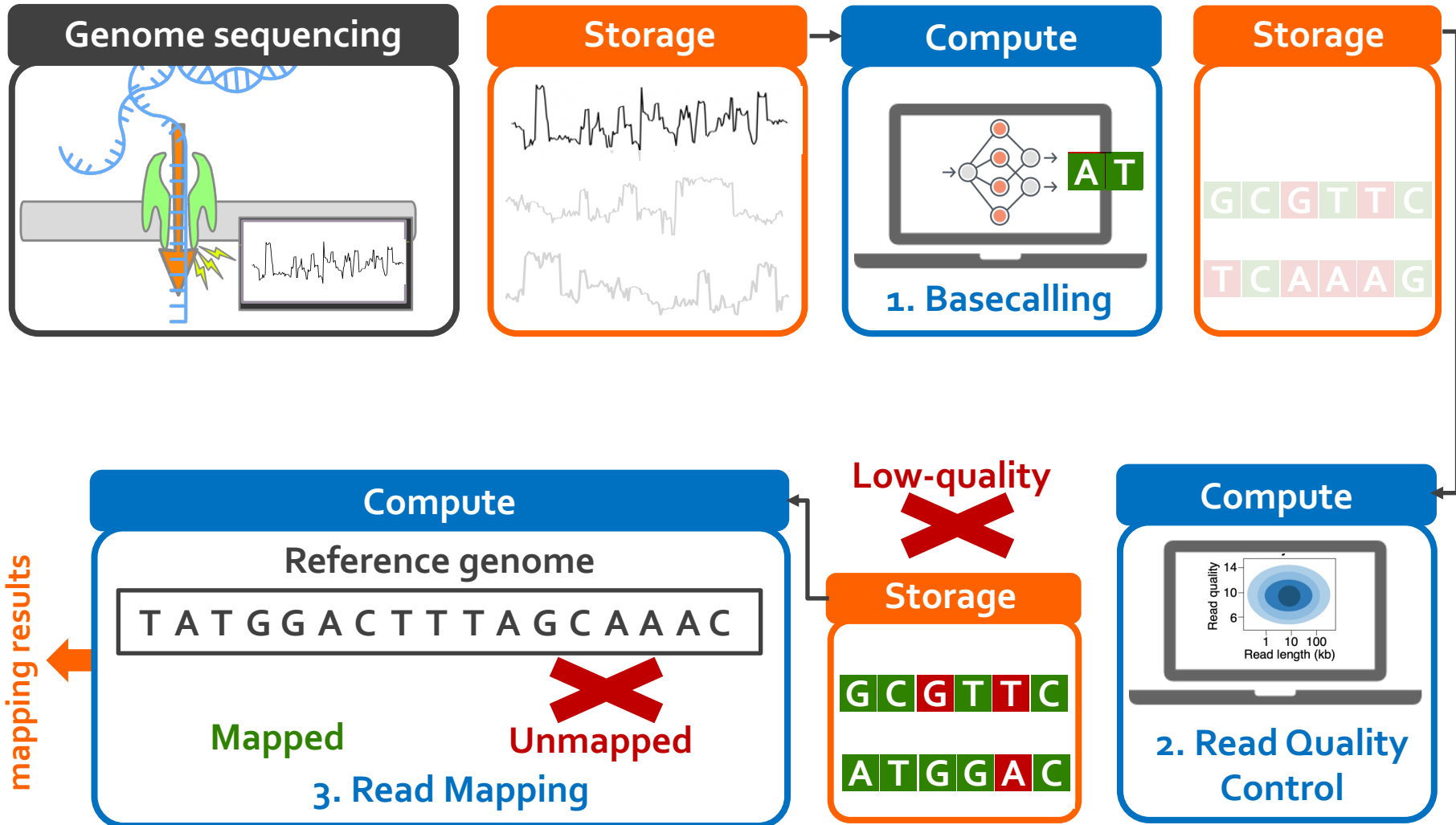
Accelerating Basecalling + Read Mapping via PIM

- 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

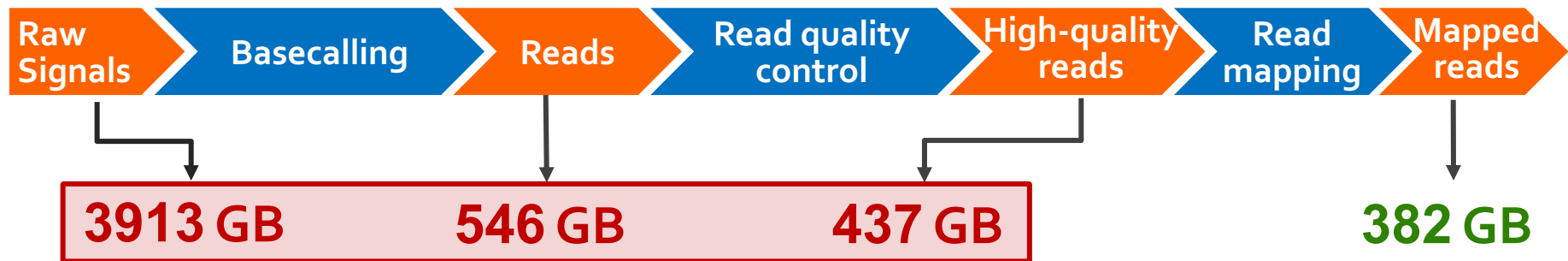
Haiyu Mao¹ Mohammed Alser¹ Mohammad Sadrosadati¹ Can Firtina¹ Akanksha Baranwal¹
Damla Senol Cali² Aditya Manglik¹ Nour Almadhoun Alserr¹ Onur Mutlu¹
¹*ETH Zürich* ²*Bionano Genomics*

Genome Analysis Pipeline



Limitation 1: Large Data Movement

- Using a human dataset in [NC'19] as an example:

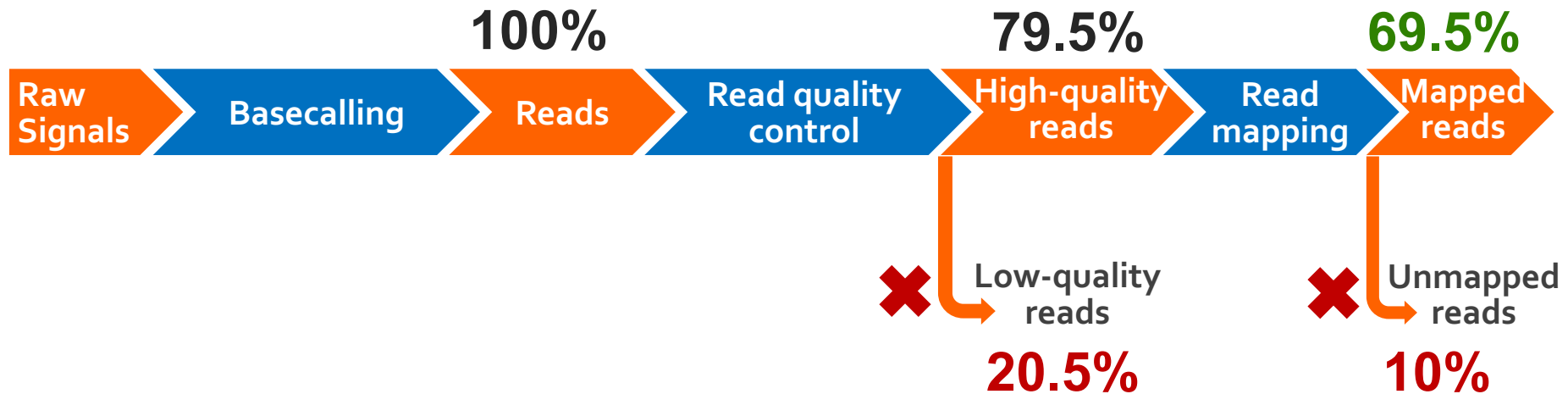


Large data movement between genome analysis steps

[NC'19] Rory Bowden, Robert W Davies, Andreas Heger, Alistair T Pagnamenta, Mariateresa de Cesare, Laura E Oikkinen, Duncan Parkes, Colin Freeman, Fatima Dhalla, Smita Y Patel, et al. Sequencing of human genomes with nanopore technology. Nature Communications, 2019.

Limitation 2: Wasted Computation

□ Using a human dataset in [NC'19] as an example:



A considerable amount of computation on **useless data** due to

- Low-quality reads
- Unmapped reads

[NC'19] Rory Bowden, Robert W Davies, Andreas Heger, Alistair T Pagnamenta, Mariateresa de Cesare, Laura E Oikkonen, Duncan Parkes, Colin Freeman, Fatima Dhalla, Smita Y Patel, et al. Sequencing of human genomes with nanopore technology. Nature Communications, 2019.

Overview: Two Limitations

Multiple steps in genome analysis



Large data movement
between multiple steps



A lot of
wasted computation
done on data that is
later discovered to be
useless

Overview: GenPIP

❑ **GenPIP:** A fast and energy-efficient **in-memory** acceleration system for the Genome analysis PIpline via **tight integration of genome analysis steps**

❑ **GenPIP** has two key techniques

- **Chunk-based pipeline (CP)**

- **Provides fine-grained collaboration** of genome analysis steps

- **Early rejection (ER)**

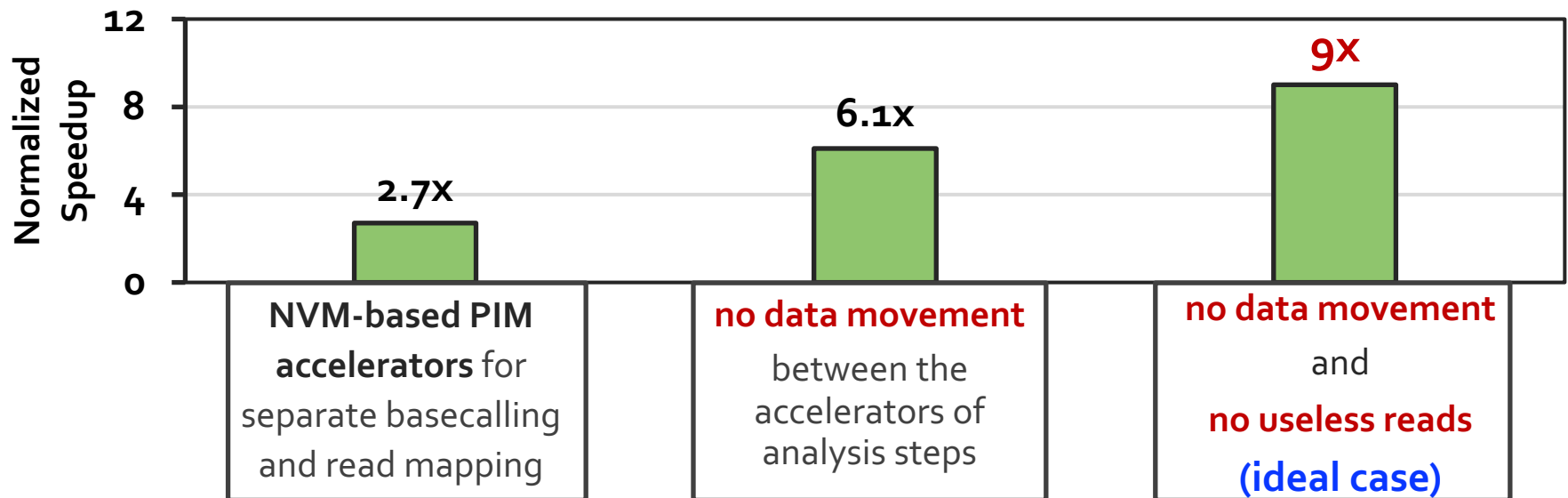
- **Timely stops the execution on useless data** by predicting which reads will not be useful

❑ **GenPIP** outperforms state-of-the-art software & hardware solutions using **CPU**, **GPU**, and **optimistic PIM** by **41.6x**, **8.4x**, and **1.4x**, respectively.

Goal and Opportunities

Goal: Efficiently accelerate the entire genome analysis pipeline while **minimizing data movement and useless computation**

- We perform a study to quantify potential performance benefits
 - Results are normalized to the performance of GPU



Agenda

- The Problem: DNA Read Mapping
 - State-of-the-art Read Mapper Design
- Algorithmic Acceleration
 - Exploiting Structure of the Genome
 - Exploiting SIMD Instructions
- Hardware Acceleration
 - Specialized Architectures
 - Processing in Memory & Storage
- Future Opportunities: New Technologies & Applications

Newer 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](#)] [[Slides \(pptx\)](#)] [[pdf](#)] [[Talk Video at AACBB 2019](#)]

New Applications: Graph Genomes

- Damla Senol Cali, Konstantinos Kanellopoulos, Joel Lindegger, Zulal Bingol, Gurpreet S. Kalsi, Ziyi Zuo, Can Firtina, Meryem Banu Cavlak, Jeremie Kim, Nika Mansouri Ghiasi, 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 Cali¹ Konstantinos Kanellopoulos² Joël Lindegger² Zülal Bingöl³
Gurpreet S. Kalsi⁴ Ziyi Zuo⁵ Can Firtina² Meryem Banu Cavlak² Jeremie Kim²
Nika Mansouri Ghiasi² Gagandeep Singh² Juan Gómez-Luna² Nour Almadhoun Alserr²
Mohammed Alser² Sreenivas Subramoney⁴ Can Alkan³ Saugata Ghose⁶ Onur Mutlu²

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

New Applications: Frequent Reference 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 Cavlak², Damla Senol Cali³, Nastaran Hajinazar^{1,4}, Mohammed Alser¹, Can Alkan² and Onur Mutlu^{1,2,3*}

Mapping Constant Regions Between References

- Jeremie S. Kim, Can Firtina, Meryem Banu Cavlak, Damla Senol Cali, Can Alkan, and Onur Mutlu,
["FastRemap: A Tool for Quickly Remapping Reads between Genome Assemblies"](#)
Bioinformatics, btac554.
[[FastRemap Source Code](#)]

FastRemap: A Tool for Quickly Remapping Reads between Genome Assemblies

Jeremie S. Kim¹

Can Firtina¹

Meryem Banu Cavlak¹

Damla Senol Cali^{2,3}

Can Alkan⁴

Onur Mutlu^{1,2,4}

¹*ETH Zürich*

²*Carnegie Mellon University*

³*Bionano Genomics*

⁴*Bilkent University*

New Frontiers: Raw Signal Analysis

- 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 and the 22nd European Conference on Computational Biology (ISMB/ECCB), Lyon, France, July 2023.
[RawHash Source Code]

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

Can Firtina¹ Nika Mansouri Ghiasi¹ Joel Lindegger¹ Gagandeep Singh¹
Meryem Banu Cavlak¹ Haiyu Mao¹ Onur Mutlu¹
¹*ETH Zurich*

New Frontiers: Raw Signal Analysis

- 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 Cavlak¹ Gagandeep Singh¹ Mohammed Alser¹ Can Firtina¹ Joël Lindegger¹
Mohammad Sadrosadati¹ Nika Mansouri Ghiasi¹ Can Alkan² Onur Mutlu¹
¹*ETH Zürich* ²*Bilkent University*

A Bright Future for Intelligent Genome Analysis

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, Onur Mutlu
["Accelerating Genome Analysis: A Primer on an Ongoing Journey"](#) IEEE Micro, August 2020.



MinION from ONT

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Sept.-Oct. 2020, pp. 65-75, vol. 40

DOI Bookmark: [10.1109/MM.2020.3013728](https://doi.org/10.1109/MM.2020.3013728)

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)



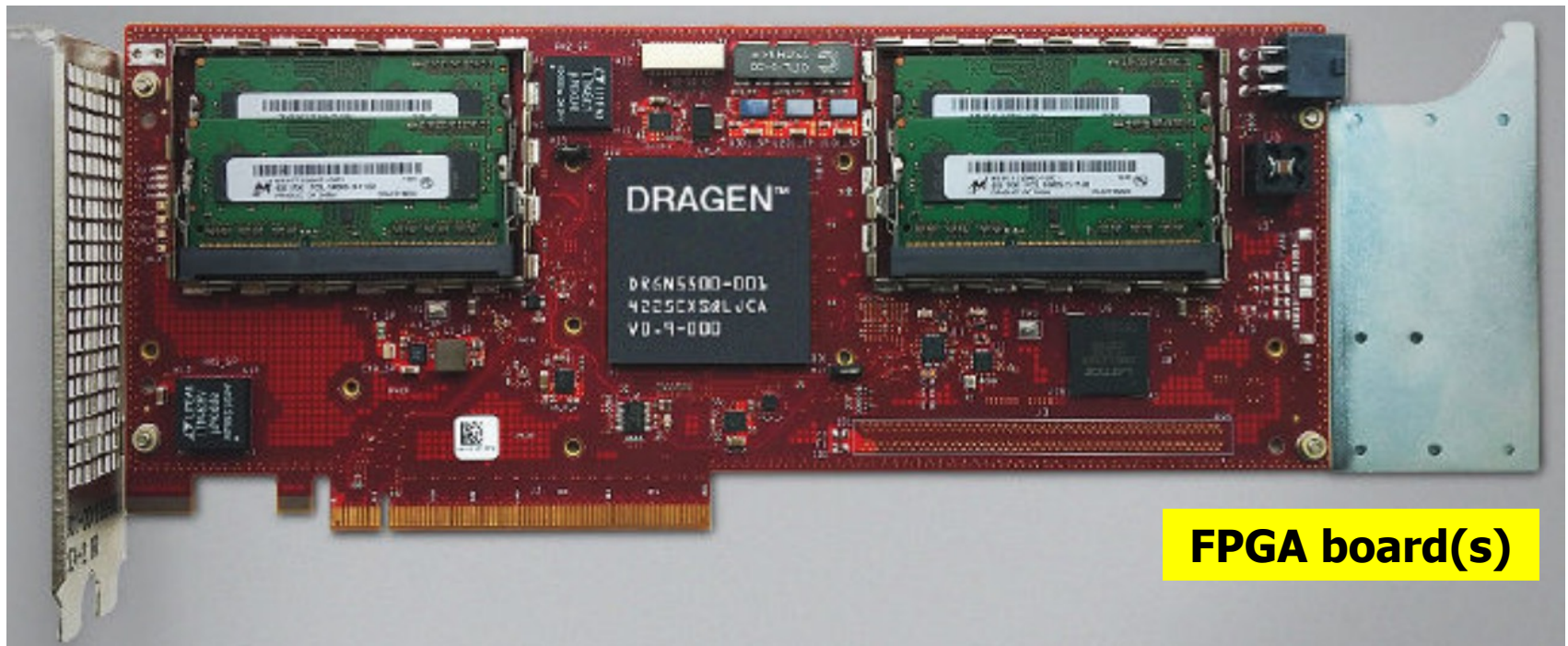
SmidgION from ONT

Conclusion

Things Are Happening In Industry

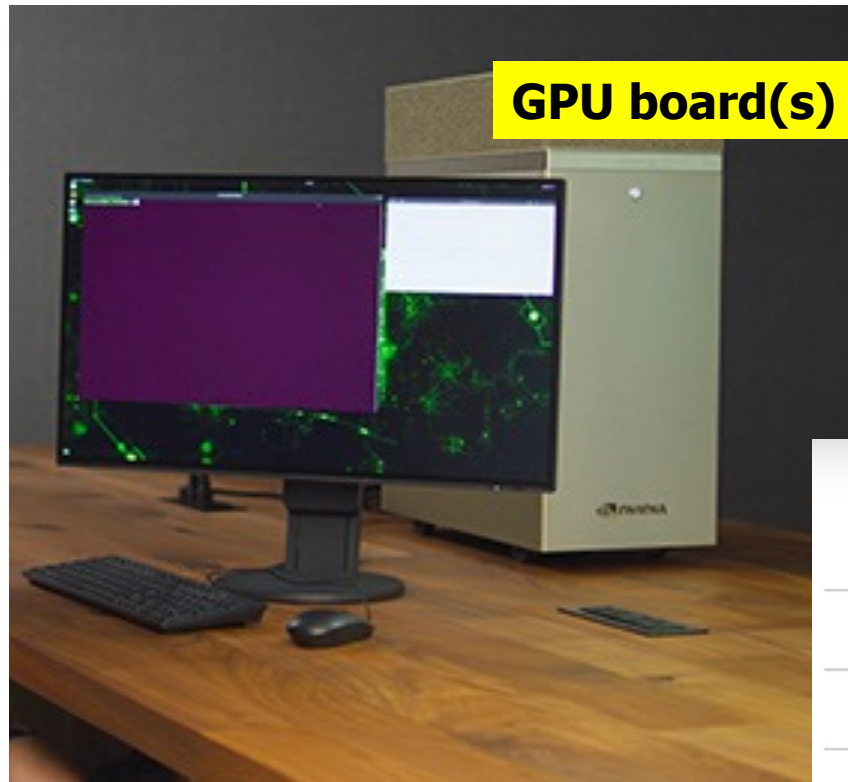
Illumina DRAGEN Bio-IT Platform (2018)

- Processes whole genome at 30x coverage in ~25 minutes with hardware support for data compression

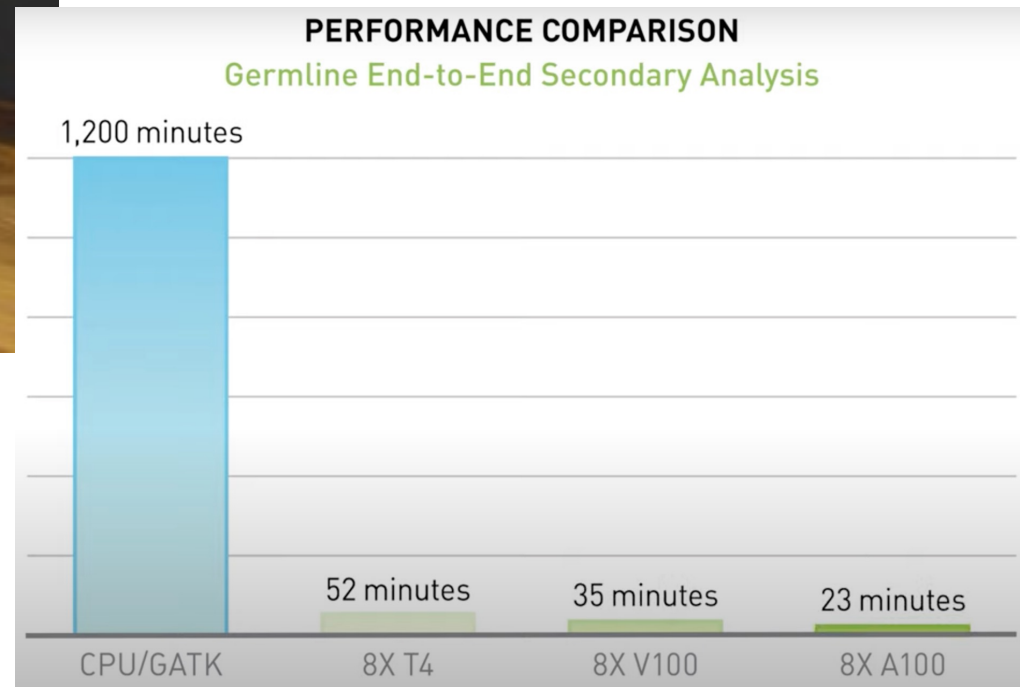


emea.illumina.com/products/by-type/informatics-products/dragen-bio-it-platform.html
emea.illumina.com/company/news-center/press-releases/2018/2349147.html

NVIDIA Clara Parabricks (2020)



A University of Michigan startup in 2018 joined NVIDIA in 2020

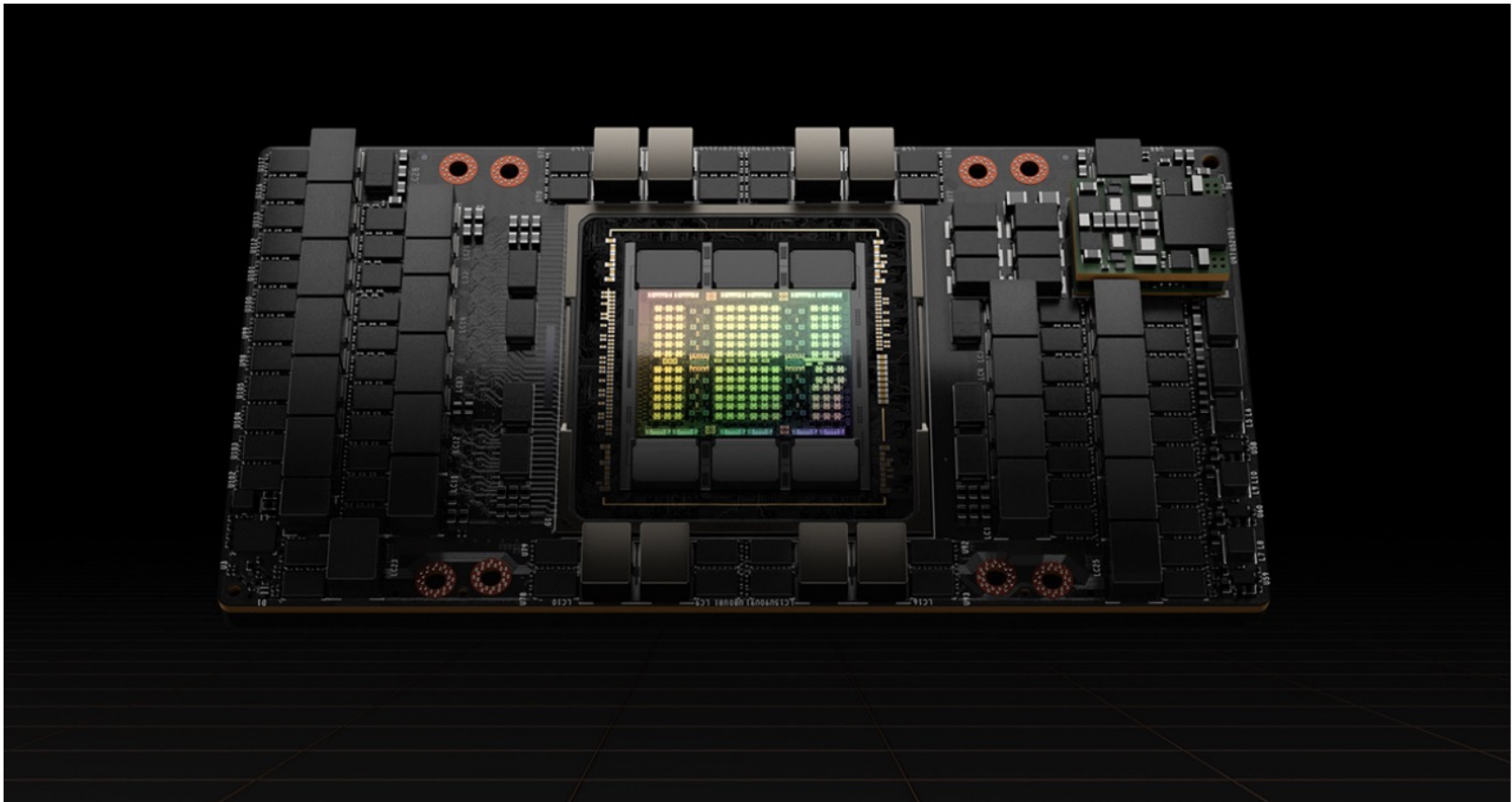


NVIDIA Hopper DPX Instructions (2022)

NVIDIA Hopper GPU Architecture Accelerates Dynamic Programming Up to 40x Using New DPX Instructions

Dynamic programming algorithms are used in healthcare, robotics, quantum computing, data science and more.

March 22, 2022 by **DION HARRIS**



**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**

Recall Our Dream (from 2007)

- An embedded device that can perform comprehensive genome analysis in real time (within a minute)
- Still a long ways to go
 - Energy efficiency
 - Performance (latency)
 - Security & privacy
 - **Huge memory bottleneck**

Conclusion

- **System design for bioinformatics** is a critical problem
 - It has large scientific, medical, societal, personal implications
- This talk is about accelerating **a key step in bioinformatics: genome sequence analysis**
 - In particular, **read mapping**
- We covered various **recent ideas to accelerate read mapping**
 - My personal journey since September 2006
- **Many future opportunities exist**
 - **Especially with new sequencing technologies**
 - **Especially with new applications and use cases**

A Bright Future for Intelligent Genome Analysis

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, Onur Mutlu
[“Accelerating Genome Analysis: A Primer on an Ongoing Journey”](#) IEEE Micro, August 2020.



MinION from ONT

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Sept.-Oct. 2020, pp. 65-75, vol. 40

DOI Bookmark: [10.1109/MM.2020.3013728](https://doi.org/10.1109/MM.2020.3013728)

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

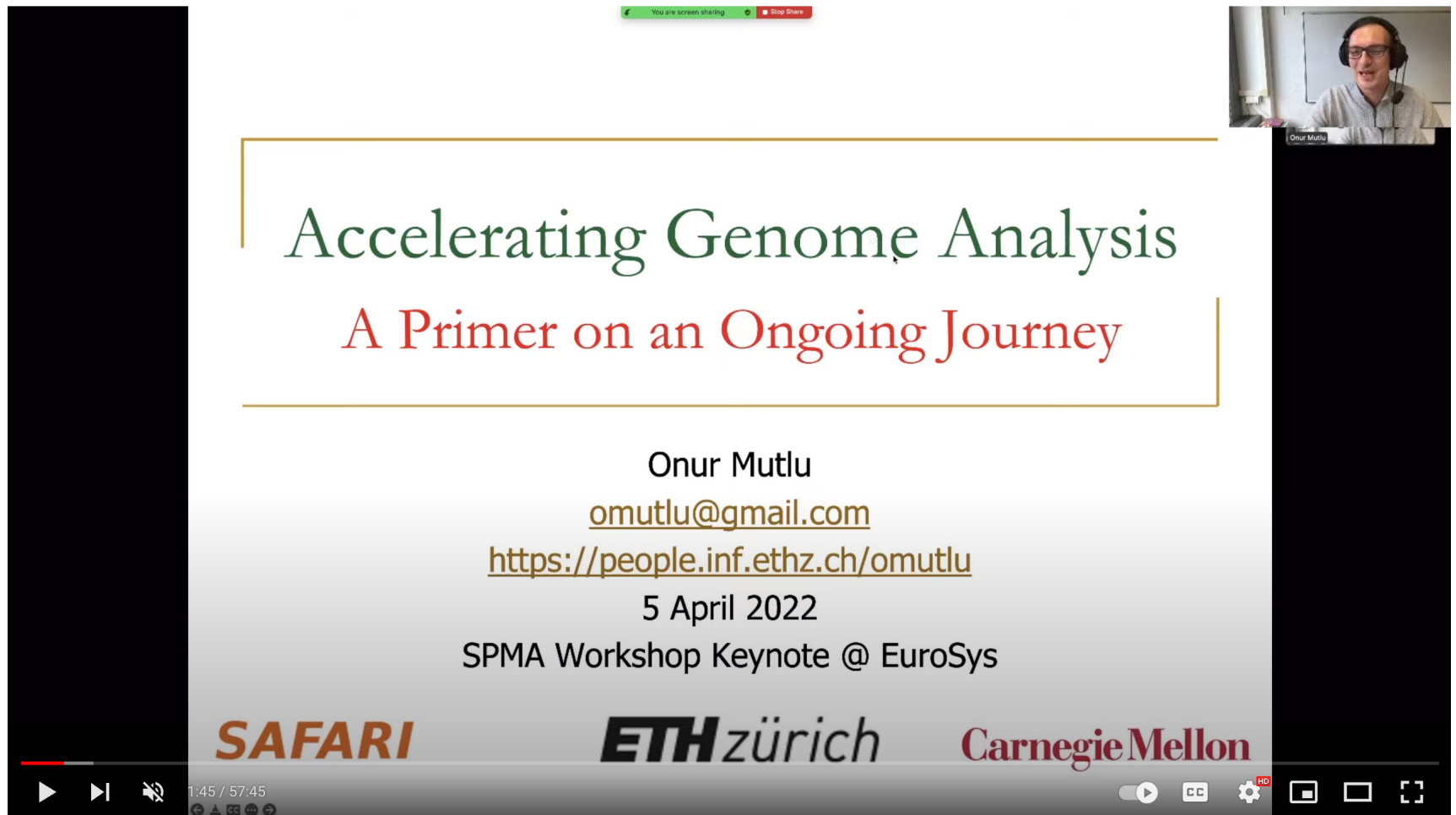
July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)



SmidgION from ONT

A Longer Version of This Talk (I)



Accelerating Genome Analysis

A Primer on an Ongoing Journey

Onur Mutlu
omutlu@gmail.com
<https://people.inf.ethz.ch/omutlu>
5 April 2022
SPMA Workshop Keynote @ EuroSys

SAFARI ETH zürich Carnegie Mellon

Accelerating Genome Analysis - Onur Mutlu (Keynote Talk at Systems for Post-Moore Arch. @ EuroSys)



Onur Mutlu Lectures
28.7K subscribers

Analytics

Edit video

16



Share

Download

Clip

Save



<https://www.youtube.com/watch?v=NCagwf0ivT0>

A Longer Version of This Talk (II)

Read Alignment/Verification

- **Edit distance** is defined as the minimum number of edits (i.e. insertions, deletions, or substitutions) needed to make the read exactly match the reference segment.

NETHERLANDS x SWITZERLAND

N	E	-	T	H	E	R	L	A	N	D	S
S	W	I	T	Z	E	R	L	A	N	D	-

match
deletion
insertion
mismatch



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



Onur Mutlu Lectures
32.3K subscribers



Subscribed ▾



15



Share



Download



385 views 2 months ago

Genomics Course (Fall 2022)

Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=bioinformatics

Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=bioinformatics

Youtube Livestream (Fall 2022):

- https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFIQvdxOdizD_EhVAMVQV

Youtube Livestream (Spring 2022):

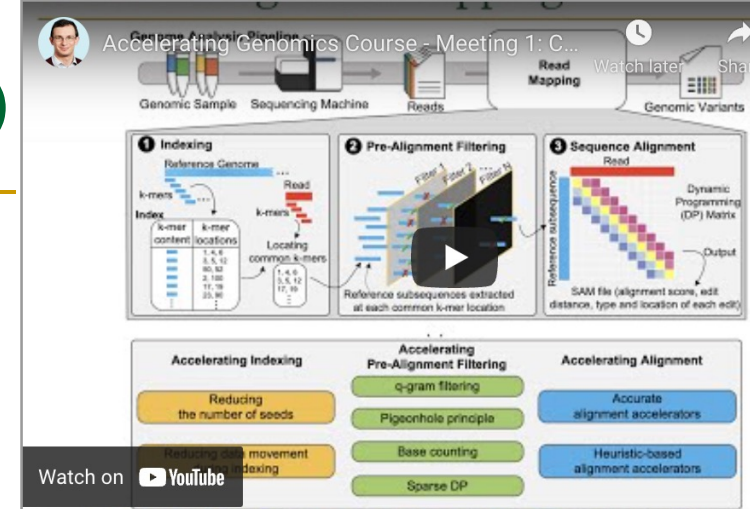
- https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NrPDgOR1yRU_Cxxjw-u18

Project course

- Taken by Bachelor's/Master's students
- Genomics lectures
- Hands-on research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>

SAFARI



Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials
W1	11.3 Fri.	YouTube Live	M1: P&S Accelerating Genomics Course Introduction & Project Proposals (PDF) (PPT)	Required Materials Recommended Materials
W2	18.3 Fri.	YouTube Live	M2: Introduction to Sequencing (PDF) (PPT)	
W3	25.3 Fri.	YouTube Premiere	M3: Read Mapping (PDF) (PPT)	
W4	01.04 Fri.	YouTube Premiere	M4: GateKeeper (PDF) (PPT)	
W5	08.04 Fri.	YouTube Premiere	M5: MAGNET & Shouji (PDF) (PPT)	
W6	15.4 Fri.	YouTube Premiere	M6: SneakySnake (PDF) (PPT)	
W7	29.4 Fri.	YouTube Premiere	M7: GenStore (PDF) (PPT)	
W8	06.05 Fri.	YouTube Premiere	M8: GRIM-Filter (PDF) (PPT)	
W9	13.05 Fri.	YouTube Premiere	M9: Genome Assembly (PDF) (PPT)	
W10	20.05 Fri.	YouTube Live	M10: Genomic Data Sharing Under Differential Privacy (PDF) (PPT)	
W11	10.06 Fri.	YouTube Premiere	M11: Accelerating Genome Sequence Analysis (PDF) (PPT)	

PIM Course (Fall 2022)

■ Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory

■ Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=processing_in_memory

■ Youtube Livestream (Fall 2022):

- <https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQOVD0GOBrnKy>

■ Youtube Livestream (Spring 2022):

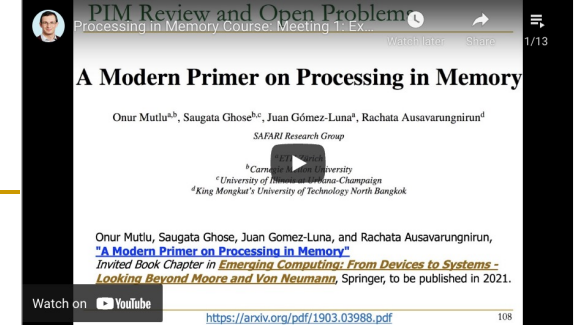
- <https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUK9EsXKhQKRPyX>

■ Project course

- Taken by Bachelor's/Master's students
- Processing-in-Memory lectures
- Hands-on research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>

SAFARI



Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	10.03 Thu.	YouTube Live	M1: P&S PIM Course Presentation (PDF) (PPT)	Required Materials Recommended Materials	HW 0 Out
W2	15.03 Tue. 17.03 Thu.	YouTube Premiere	M2: Real-world PIM: UPMEM PIM (PDF) (PPT)		
W3	24.03 Thu.	YouTube Live	M3: Real-world PIM: Microbenchmarking of UPMEM PIM (PDF) (PPT)		
W4	31.03 Thu.	YouTube Live	M4: Real-world PIM: Samsung HBM-PIM (PDF) (PPT)		
W5	07.04 Thu.	YouTube Live	M5: How to Evaluate Data Movement Bottlenecks (PDF) (PPT)		
W6	14.04 Thu.	YouTube Live	M6: Real-world PIM: SK Hynix AIM (PDF) (PPT)		
W7	21.04 Thu.	YouTube Premiere	M7: Programming PIM Architectures (PDF) (PPT)		
W8	28.04 Thu.	YouTube Premiere	M8: Benchmarking and Workload Suitability on PIM (PDF) (PPT)		
W9	05.05 Thu.	YouTube Premiere	M9: Real-world PIM: Samsung AxDIMM (PDF) (PPT)		
W10	12.05 Thu.	YouTube Premiere	M10: Real-world PIM: Alibaba HB-PNM (PDF) (PPT)		
W11	19.05 Thu.	YouTube Live	M11: SpMV on a Real PIM Architecture (PDF) (PPT)		
W12	26.05 Thu.	YouTube Live	M12: End-to-End Framework for Processing-using-Memory (PDF) (PPT)		
W13	02.06 Thu.	YouTube Live	M13: Bit-Serial SIMD Processing using DRAM (PDF) (PPT)		
W14	09.06 Thu.	YouTube Live	M14: Analyzing and Mitigating ML Inference Bottlenecks (PDF) (PPT)		
W15	15.06 Thu.	YouTube Live	M15: In-Memory HTAP Databases with HW/SW Co-design (PDF) (PPT)		
W16	23.06 Thu.	YouTube Live	M16: In-Storage Processing for Genome Analysis (PDF) (PPT)		
W17	18.07 Mon.	YouTube Premiere	M17: How to Enable the Adoption of PIM? (PDF) (PPT)		
W18	09.08 Tue.	YouTube Premiere	SS1: ISVLSI 2022 Special Session on PIM (PDF & PPT)		

SSD Course (Spring 2023)

Spring 2023 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=modern_ssd

Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=modern_ssd

Youtube Livestream (Spring 2023):

- https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi_8qOM5Icpp8hB2SHtm4z57&pp=iAQB

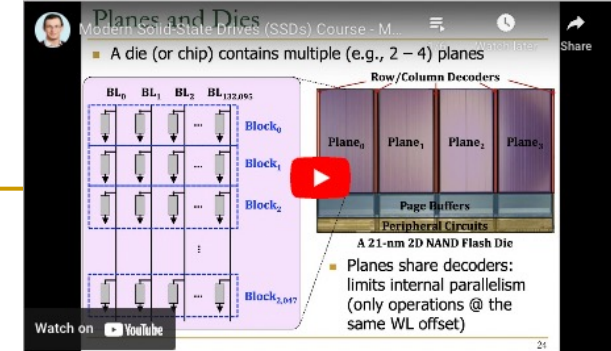
Youtube Livestream (Fall 2022):

- <https://www.youtube.com/watch?v=hqLrd-Uj0aU&list=PL5Q2soXY2Zi9BJhenUq4JI5bwhAMpAp13&pp=iAQB>

Project course

- Taken by Bachelor's/Master's students
- SSD Basics and Advanced Topics
- Hands-on research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>

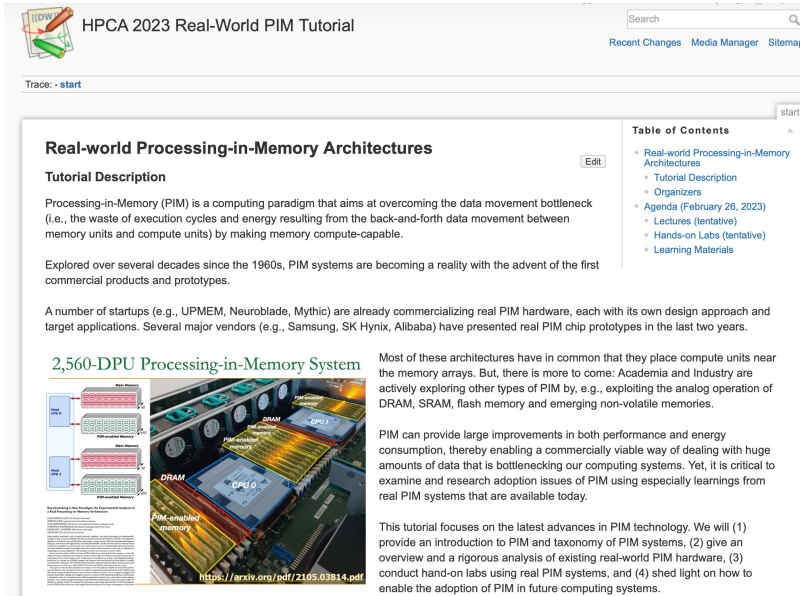


Fall 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	06.10		M1: P&S Course Presentation PDF PPT	Required Recommended	
W2	12.10	YouTube Live	M2: Basics of NAND Flash-Based SSDs PDF PPT	Required Recommended	
W3	19.10	YouTube Live	M3: NAND Flash Read/Write Operations PDF PPT	Required Recommended	
W4	26.10	YouTube Live	M4: Processing inside NAND Flash PDF PPT	Required Recommended	
W5	02.11	YouTube Live	M5: Advanced NAND Flash Commands & Mapping PDF PPT	Required Recommended	
W6	09.11	YouTube Live	M6: Processing inside Storage PDF PPT	Required Recommended	
W7	23.11	YouTube Live	M7: Address Mapping & Garbage Collection PDF PPT	Required Recommended	
W8	30.11	YouTube Live	M8: Introduction to MQSim PDF PPT	Required Recommended	
W9	14.12	YouTube Live	M9: Fine-Grained Mapping and Multi-Plane Operation-Aware Block Management PDF PPT	Required Recommended	
W10	04.01.2023	YouTube Premiere	M10a: NAND Flash Basics PDF PPT	Required Recommended	
			M10b: Reducing Solid-State Drive Read Latency by Optimizing Read-Retry PDF PPT Paper	Required Recommended	
			M10c: Evanescence: Architectural Support for Efficient Data Sanitization in Modern Flash-Based Storage Systems PDF PPT Paper	Required Recommended	
			M10d: DeepSketch: A New Machine Learning-Based Reference Search Technique for Post-Deduplication Delta Compression PDF PPT Paper	Required Recommended	
W11	11.01	YouTube Live	M11: FLIN: Enabling Fairness and Enhancing Performance in Modern NVMe Solid State Drives PDF PPT	Required	
W12	25.01	YouTube Premiere	M12: Flash Memory and Solid-State Drives PDF PPT	Recommended	

Real PIM Tutorial (HPCA 2023)

■ February 26: Lectures + Hands-on labs + Invited Talks



Real-world Processing-in-Memory Architectures

Tutorial Description

Processing-in-Memory (PIM) is a computing paradigm that aims at overcoming the data movement bottleneck (i.e., the waste of execution cycles and energy resulting from the back-and-forth data movement between memory units and compute units) by making memory compute-capable.

Explored over several decades since the 1960s, PIM systems are becoming a reality with the advent of the first commercial products and prototypes.

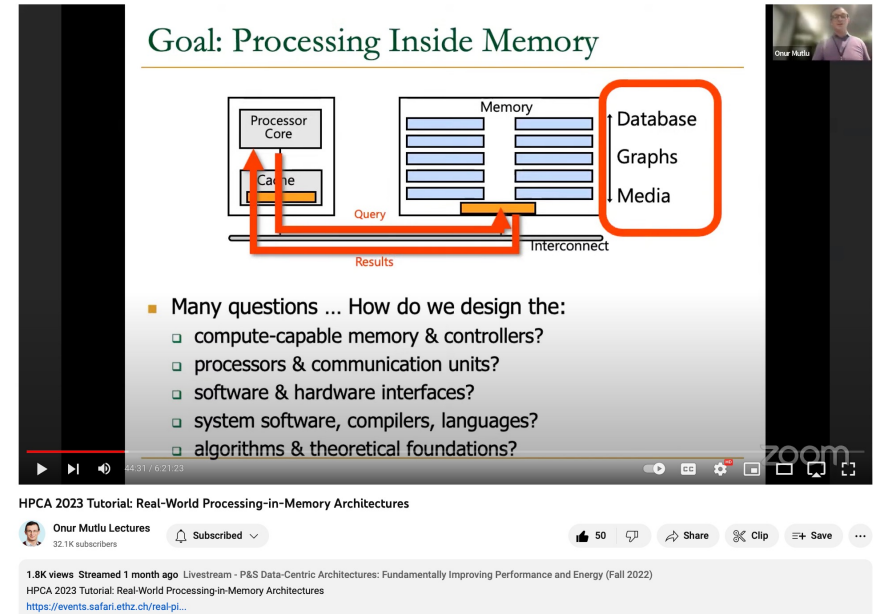
A number of startups (e.g., UPMEM, Neuroblade, Mythic) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Allbaba) have presented real PIM chip prototypes in the last two years.

2,560-DPU Processing-in-Memory System

Most of these architectures have in common that they place compute units near the memory arrays. But, there is more to come: Academia and Industry are actively exploring other types of PIM by, e.g., exploiting the analog operation of DRAM, SRAM, flash memory and emerging non-volatile memories.

PIM can provide large improvements in both performance and energy consumption, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to examine and research adoption issues of PIM using especially learnings from real PIM systems that are available today.

This tutorial focuses on the latest advances in PIM technology. We will (1) provide an introduction to PIM and taxonomy of PIM systems, (2) give an overview and a rigorous analysis of existing real-world PIM hardware, (3) conduct hand-on labs using real PIM systems, and (4) shed light on how to enable the adoption of PIM in future computing systems.



Goal: Processing Inside Memory

Processor Core

Memory

Database

Graphs

Media

Query

Results

Interconnect

■ Many questions ... How do we design the:

- compute-capable memory & controllers?
- processors & communication units?
- software & hardware interfaces?
- system software, compilers, languages?
- algorithms & theoretical foundations?

HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures

Onur Mutlu Lectures

32.1K subscribers

50

Share

Clip

Save

1.8K views · Streamed 1 month ago

Livestream - P&S Data-Centric Architectures: Fundamentally Improving Performance and Energy (Fall 2022)

HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures

<https://events.safari.ethz.ch/real-pi...>

Time	Speaker	Title	Materials
8:00am-8:40am	Prof. Onur Mutlu	Memory-Centric Computing	PDF PPT
8:40am-10:00am	Dr. Juan Gómez Luna	Processing-Near-Memory: Real PNM Architectures Programming General-purpose PIM	PDF PPT
10:20am-11:00am	Dr. Dimin Niu	A 3D Logic-to-DRAM Hybrid Bonding Process-Near-Memory Chip for Recommendation System	
11:00am-11:40am	Dr. Christina Giannoula	SparseP: Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Architectures	PDF PPT
1:30pm-2:10pm	Dr. Juan Gómez Luna	Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components	PDF PPT
2:10pm-2:50pm	Dr. Manuel Le Gallo	Deep Learning Inference Using Computational Phase-Change Memory	
2:50pm-3:30pm	Dr. Juan Gómez Luna	PIM Adoption Issues: How to Enable PIM Adoption?	PDF PPT
3:40pm-5:40pm	Dr. Juan Gómez Luna	Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture	Handout PDF PPT

<https://www.youtube.com/watch?v=f5-nT1tbz5w>

<https://events.safari.ethz.ch/real-pim-tutorial/>

Real PIM Tutorial (ASPLOS 2023)

■ March 26: Lectures + Hands-on labs + Invited talks

ASPLOS 2023 Real-World PIM Tutorial

Trace: • start

Real-world Processing-in-Memory Systems for Modern Workloads

Important note about registration

Tutorial Description

Processing-in-Memory (PIM) is a computing paradigm that aims at overcoming the data movement bottleneck (i.e., the waste of execution cycles and energy resulting from the back-and-forth data movement between memory units and compute units) by making memory compute-capable.

Explored over several decades since the 1960s, PIM systems are becoming a reality with the advent of the first commercial products and prototypes.

A number of startups (e.g., UPMEM, Neuroblade) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Alibaba) have presented real PIM chip prototypes in the last two years. Most of these architectures have in common that they place compute units near the memory arrays. This type of PIM is called processing near memory (PNM).

2,560-DPU Processing-in-Memory System

PIM can provide large improvements in both performance and energy consumption for many modern applications, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to (1) study and understand the characteristics that make a workload suitable for a PIM architecture, (2) assess performance estimates for PIM kernels, and (3) design PIM architectures that are suitable for PIM kernels and workloads.

Tutorial Materials

Time	Speaker	Title	Materials
9:00am-10:20am	Prof. Onur Mutlu	Memory-Centric Computing	PDF PPT
10:40am-12:00pm	Dr. Juan Gómez Luna	Processing-Near-Memory: Real PNM Architectures Programming General-purpose PIM	PDF PPT
1:40pm-2:20pm	Prof. Alexandra (Sasha) Fedorova (UBC)	Processing in Memory in the Wild	PDF PPT
2:20pm-3:20pm	Dr. Juan Gómez Luna & Ataberk Olgun	Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components	PDF PPT PDF PPT
3:40pm-4:10pm	Dr. Juan Gómez Luna	Adoption issues: How to enable PIM? Accelerating Modern Workloads on a General-purpose PIM System	PDF PPT PDF PPT
4:10pm-4:50pm	Dr. Yongkee Kwon & Eddy (Chanwook) Park (SK Hynix)	System Architecture and Software Stack for GDDR6-AiM	PDF PPT
4:50pm-5:00pm	Dr. Juan Gómez Luna	Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture	Handout PDF PPT

ASPLOS 2023 Tutorial

Real-world Processing-in-Memory Systems for Modern Workloads

Accelerating Modern Workloads on a General-purpose PIM System

Dr. Juan Gómez Luna
Professor Onur Mutlu

ETH Zürich SAFARI

Sunday, March 26, 2023

ASPLOS 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads

Onur Mutlu Lectures
32.1K subscribers

Subscribed

33

Share

Clip

Save

...

views Streamed 7 days ago Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

LOS 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads

<https://events.safari.ethz.ch/asplos-2023/>

<https://www.youtube.com/watch?v=oYCaLcT0Kmo>

<https://events.safari.ethz.ch/asplos-pim-tutorial/>

Upcoming Real PIM Tutorial (ISCA 2023)

■ June 18: Lectures + Hands-on labs + Invited talks



ISCA 2023 Real-World PIM Tutorial

Search

[Recent Changes](#) [Media Manager](#) [Sitemap](#)

Trace: • [start](#)

Real-world Processing-in-Memory Systems for Modern Workloads

Tutorial Description

Processing-in-Memory (PIM) is a computing paradigm that aims at overcoming the data movement bottleneck (i.e., the waste of execution cycles and energy resulting from the back-and-forth data movement between memory units and compute units) by making memory compute-capable.

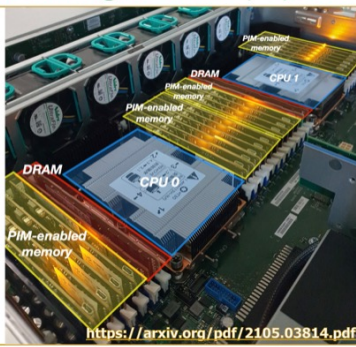
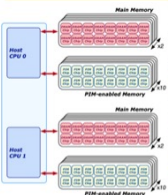
Explored over several decades since the 1960s, PIM systems are becoming a reality with the advent of the first commercial products and prototypes.

A number of startups (e.g., UPMEM, Neuroblade) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Alibaba) have presented real PIM chip prototypes in the last two years. Most of these architectures have in common that they place compute units near the memory arrays. This type of PIM is called processing near memory (PNM).

Table of Contents

- [Real-world Processing-in-Memory Systems for Modern Workloads](#)
- [Tutorial Description](#)
- [Organizers](#)
- [Agenda \(June 18, 2023\)](#)
- [Lectures \(tentative\)](#)
- [Hands-on Labs \(tentative\)](#)
- [Learning Materials](#)

2,560-DPU Processing-in-Memory System



<https://arxiv.org/pdf/2105.03814.pdf>

PIM can provide large improvements in both performance and energy consumption for many modern applications, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to (1) study and understand the characteristics that make a workload suitable for a PIM architecture, (2) propose optimization strategies for PIM kernels, and (3) develop programming frameworks and tools that can lower the learning curve and ease the adoption of PIM.

This tutorial focuses on the latest advances in PIM technology, workload characterization for PIM, and programming and optimizing PIM kernels. We will (1) provide an introduction to PIM and taxonomy of PIM systems, (2) give an overview and a rigorous analysis of existing real-world PIM hardware, (3) conduct hand-on labs about important workloads (machine learning, sparse linear algebra, bioinformatics, etc.) using real PIM systems, and (4) shed light on how to improve future PIM systems for such workloads.

<https://events.safari.ethz.ch/isca-pim-tutorial/>

Acknowledgments

SAFARI

SAFARI Research Group

safari.ethz.ch

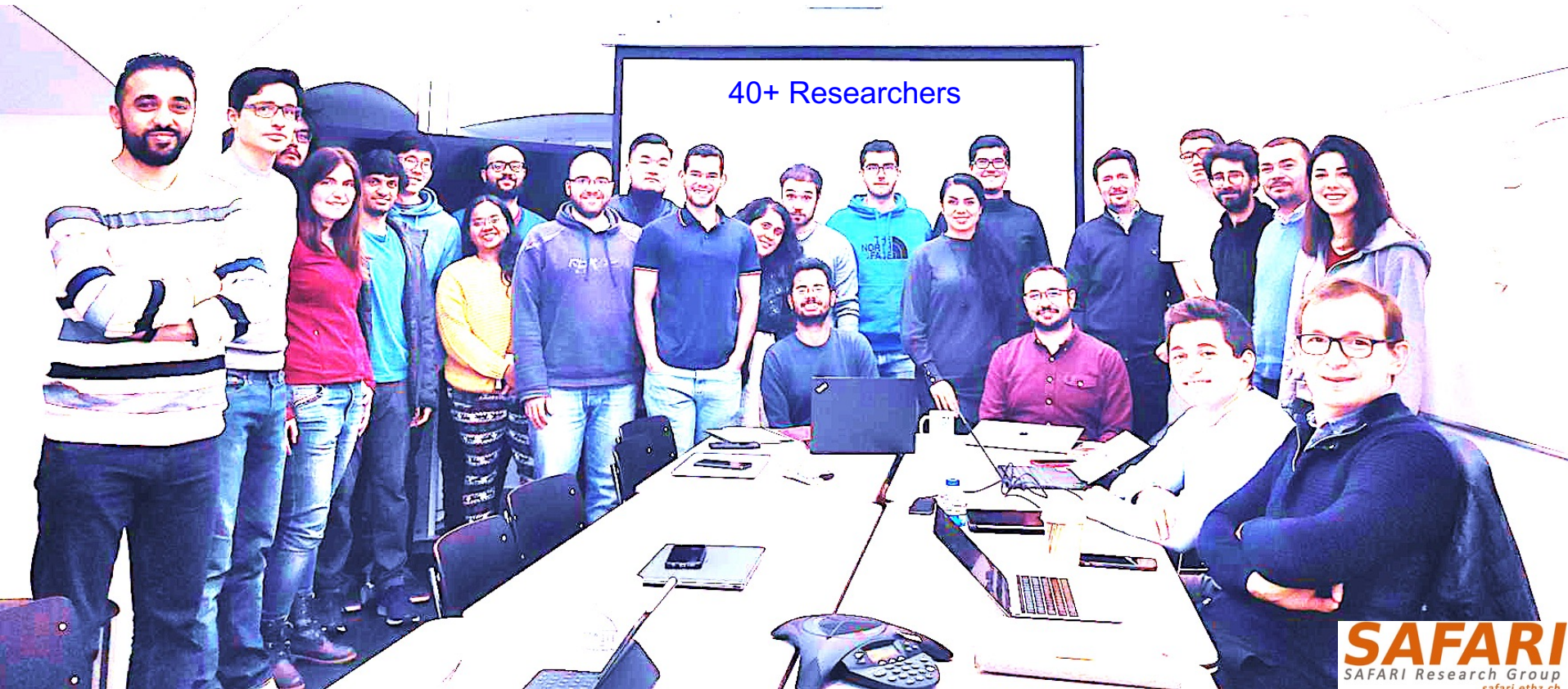
Think BIG, Aim HIGH!

<https://safari.ethz.ch>

Onur Mutlu's SAFARI Research Group

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

<https://safari.ethz.ch/safari-newsletter-january-2021/>



SAFARI
SAFARI Research Group
safari.ethz.ch

Think BIG, Aim HIGH!

SAFARI

<https://safari.ethz.ch>

SAFARI Newsletter December 2021 Edition

- <https://safari.ethz.ch/safari-newsletter-december-2021/>

SAFARI
SAFARI Research Group

Think Big, Aim High

ETH zürich



View in your browser
December 2021



SAFARI Introduction & Research

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



Seminar in Computer Architecture - Lecture 5: Potpourri of Research Topics (Spring 2023)



Onur Mutlu Lectures
32.6K subscribers



719 views Streamed 1 month ago Livestream - Seminar in Computer Architecture - ETH Zürich (Spring 2023)

SAFARI
SAFARI Research Group
safari.ethz.ch

THINK BIG, AIM HIGH!

SAFARI

<https://www.youtube.com/watch?v=mV2OuB2djEs>

Referenced Papers, Talks, Artifacts

- All are available at

<https://people.inf.ethz.ch/omutlu/projects.htm>

<https://www.youtube.com/onurmutlulectures>

<https://github.com/CMU-SAFARI/>

Open Source Tools: SAFARI GitHub



SAFARI Research Group at ETH Zurich and Carnegie Mellon University

Site for source code and tools distribution from SAFARI Research Group at ETH Zurich and Carnegie Mellon University.

 241 followers  ETH Zurich and Carnegie Mellon U...  <https://safari.ethz.ch/>  omutlu@gmail.com

 Overview  Repositories 80  Projects  Packages  People 13

Pinned

 **ramulator** Public

A Fast and Extensible DRAM Simulator, with built-in support for modeling many different DRAM technologies including DDRx, LPDDRx, GDDRx, WIOx, HBMx, and various academic proposals. Described in the...

 C++  415  187

 **MQSim** Public

MQSim is a fast and accurate simulator modeling the performance of modern multi-queue (MQ) SSDs as well as traditional SATA based SSDs. MQSim faithfully models new high-bandwidth protocol implement...

 C++  185  112

 **SparseP** Public

SparseP is the first open-source Sparse Matrix Vector Multiplication (SpMV) software package for real-world Processing-In-Memory (PIM) architectures. SparseP is developed to evaluate and characteri...

 C  55  10

 **prim-benchmarks** Public

PrIM (Processing-In-Memory benchmarks) is the first benchmark suite for a real-world processing-in-memory (PIM) architecture. PrIM is developed to evaluate, analyze, and characterize the first publ...

 C  82  35

 **rowhammer** Public

Source code for testing the Row Hammer error mechanism in DRAM devices. Described in the ISCA 2014 paper by Kim et al. at http://users.ece.cmu.edu/~omutlu/pub/dram-row-hammer_isca14.pdf.

 C  203  40

 **SoftMC** Public

SoftMC is an experimental FPGA-based memory controller design that can be used to develop tests for DDR3 SODIMMs using a C++ based API. The design, the interface, and its capabilities and limitatio...

 Verilog  99  26

<https://github.com/CMU-SAFARI/>

Accelerating Genome Analysis

A Primer on an Ongoing Journey

Onur Mutlu

omutlu@gmail.com

<https://people.inf.ethz.ch/omutlu>

8 June 2023

University of Montenegro Faculty of Medicine

SAFARI

ETH zürich

Carnegie Mellon

Backup Slides for Further Info

Resources & Acknowledgments

Special Research Sessions & Courses

- Special Session at ISVLSI 2022: 9 cutting-edge talks



The image shows a YouTube video player interface. The video title is "In-Memory Processing ISVLSI 2022 Special Session". Below the title, it says "IEEE Computer Society Annual Symposium on VLSI". The video is from the "Onur Mutlu Lectures" channel, which has 26.9K subscribers. The video has 1,286 views and was premiered on Aug 9, 2022. The video player shows a thumbnail of a presentation slide with the text "In-Memory Processing ISVLSI 2022 Special Session" and "IEEE Computer Society Annual Symposium on VLSI". The video is currently at 0:04 / 3:36:35. The video player includes standard controls like play, pause, volume, and full screen. Below the video player, there are buttons for "ANALYTICS" and "EDIT VIDEO".

In-Memory Processing
ISVLSI 2022 Special Session

IEEE Computer Society Annual Symposium on VLSI

ISVLSI 2022

Adonis room
Ailathon resort, Paphos, Cyprus
July 4th, 2022

0:04 / 3:36:35 • Dr. Juan Gómez-Luna, "Introduction to the ISVLSI 2022 Special Session on Processing-in-Memory" >

ISVLSI 2022 Special Session on Processing-in-Memory

1,286 views • Premiered Aug 9, 2022

61 DISLIKE SHARE DOWNLOAD CLIP SAVE ...

Onur Mutlu Lectures
26.9K subscribers

ANALYTICS EDIT VIDEO

Detailed Lectures on Genome Analysis

- **Computer Architecture, Fall 2020, Lecture 3a**
 - **Introduction to Genome Sequence Analysis** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=CrRb32v7SJc&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=5>
- **Computer Architecture, Fall 2020, Lecture 8**
 - **Intelligent Genome Analysis** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=ygmQpdDTL7o&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=14>
- **Computer Architecture, Fall 2020, Lecture 9a**
 - **GenASM: Approx. String Matching Accelerator** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=XoLpzmN-Pas&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=15>
- **Accelerating Genomics Project Course, Fall 2020, Lecture 1**
 - **Accelerating Genomics** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=rgjl8ZyLsAg&list=PL5Q2soXY2Zi9E2bBVAgCqLgwiDRQDTyId>

Comp Arch (Fall'21)

- Lectures/Schedule
- Lecture Buzzwords
- Readings
- HWs
- Labs
- Exams
- Related Courses
- Tutorials

- Computer Architecture FS20: Course Webpage
- Computer Architecture FS20: Lecture Videos
- Digitaltechnik SS21: Course Webpage
- Digitaltechnik SS21: Lecture Videos
- Moodle
- HotCRP
- Verilog Practice Website (HDLBits)

Fall 2021 Edition:

- <https://safari.ethz.ch/architecture/fall2021/doku.php?id=schedule>

Fall 2020 Edition:

- <https://safari.ethz.ch/architecture/fall2020/doku.php?id=schedule>

Youtube Livestream (2021):

- https://www.youtube.com/watch?v=4yfkM_5EFg0&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILkTOF

Youtube Livestream (2020):

- <https://www.youtube.com/watch?v=c3mPdZA-Fmc&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN>

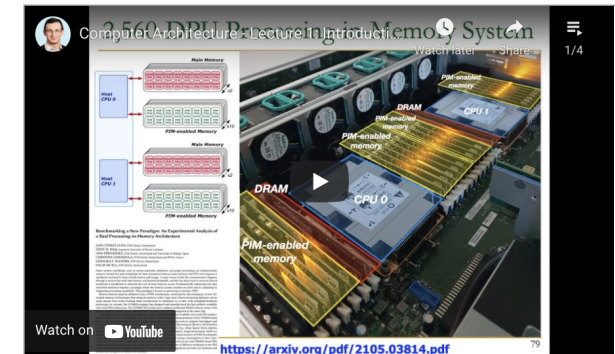
Master's level course

- Taken by Bachelor's/Masters/PhD students
- Cutting-edge research topics + fundamentals in Computer Architecture
- 5 Simulator-based Lab Assignments
- Potential research exploration
- Many research readings

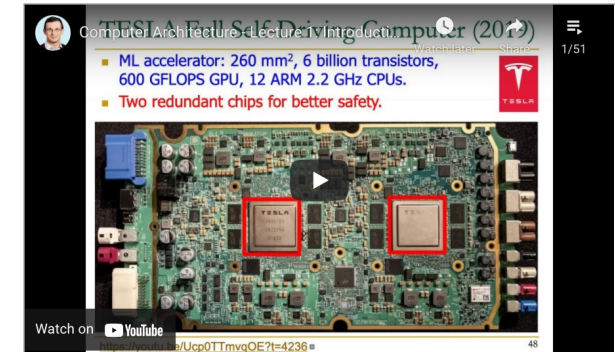
<https://www.youtube.com/onurmutlulectures>

Lecture Video Playlist on YouTube

Livestream Lecture Playlist



Recorded Lecture Playlist



Fall 2021 Lectures & Schedule

Week	Date	Livestream	Lecture	Readings	Lab	HW
W1	30.09 Thu.	YouTube Live	L1: Introduction and Basics PDF PPT	Required Mentioned	Lab 1 Out	HW 0 Out
	01.10 Fri.	YouTube Live	L2: Trends, Tradeoffs and Design Fundamentals PDF PPT	Required Mentioned		
W2	07.10 Thu.	YouTube Live	L3a: Memory Systems: Challenges and Opportunities PDF PPT	Described Suggested		HW 1 Out
			L3b: Course Info & Logistics PDF PPT			
			L3c: Memory Performance Attacks PDF PPT	Described Suggested		
	08.10 Fri.	YouTube Live	L4a: Memory Performance Attacks PDF PPT	Described Suggested	Lab 2 Out	
			L4b: Data Retention and Memory Refresh PDF PPT	Described Suggested		
			L4c: RowHammer PDF PPT	Described Suggested		

DDCA (Spring 2022)

Spring 2022 Edition:

- <https://safari.ethz.ch/digitaltechnik/spring2022/duku.php?id=schedule>

Spring 2021 Edition:

- <https://safari.ethz.ch/digitaltechnik/spring2021/duku.php?id=schedule>

Youtube Livestream (Spring 2022):

- <https://www.youtube.com/watch?v=cpXdE3HwvK0&list=PL5Q2soXY2Zi97Ya5DEUpMpO2bbAoaG7c6>

Youtube Livestream (Spring 2021):

- https://www.youtube.com/watch?v=LbC0EZY8yw4&list=PL5Q2soXY2Zi_uej3aY39YB5pfW4SJ7LIN

Bachelor's course

- 2nd semester at ETH Zurich
- Rigorous introduction into "How Computers Work"
- Digital Design/Logic
- Computer Architecture
- 10 FPGA Lab Assignments

SAFARI

<https://www.youtube.com/onurmutlulectures>



Trace: - schedule

Home

Announcements

Materials

- Lectures/Schedule
- Lecture Buzzwords
- Readings
- Optional HWs
- Labs
- Extra Assignments
- Exams
- Technical Docs

Resources

- Computer Architecture (CMU) SS15: Lecture Videos
- Computer Architecture (CMU) SS15: Course Website
- Digitaltechnik SS18: Lecture Videos
- Digitaltechnik SS18: Course Website
- Digitaltechnik SS19: Lecture Videos
- Digitaltechnik SS19: Course Website
- Digitaltechnik SS20: Lecture Videos
- Digitaltechnik SS20: Course Website
- Moodle

Lecture Video Playlist on YouTube

Livestream Lecture Playlist

Onur Mutlu: Digital Design and Computer Architecture Today

- Computing landscape is very different from 10-20 years ago
- Applications and technology both demand novel architectures

Watch on YouTube

Recorded Lecture Playlist

Digital Design & Computer Architecture: Lect...

How Computers Work

(from the ground up)

Watch on YouTube

Spring 2021 Lectures/Schedule

Week	Date	Livestream	Lecture	Readings	Lab	HW
W1	25.02 Thu.	YouTube Live	L1: Introduction and Basics G20a (PDF) G20b (PPT)	Required Suggested Mentioned		
	26.02 Fri.	YouTube Live	L2a: Tradeoffs, Metrics, Mindset G20a (PDF) G20b (PPT)	Required		
			L2b: Mysteries in Computer Architecture G20a (PDF) G20b (PPT)	Required Mentioned		
W2	04.03 Thu.	YouTube Live	L3a: Mysteries in Computer Architecture II G20a (PDF) G20b (PPT)	Required Suggested Mentioned		

Seminar in Comp Arch (Spring & Fall)

Spring 2022 Edition:

- https://safari.ethz.ch/architecture_seminar/spring2022/doku.php?id=schedule

Fall 2021 Edition:

- https://safari.ethz.ch/architecture_seminar/fall2021/doku.php?id=schedule

Youtube Livestream (Spring 2022):

- https://www.youtube.com/watch?v=rS9UPk509AQ&list=PL5Q2soXY2Zi_hxizriwKmFHgcoe2Q8-m0

Youtube Livestream (Fall 2021):

- https://www.youtube.com/watch?v=4TcP297mdsI&list=PL5Q2soXY2Zi_7UBNmC9B8Yr5JSwTG9yH4

Critical analysis course

- Taken by Bachelor's/Masters/PhD students
- Cutting-edge research topics + fundamentals in Computer Architecture
- 20+ research papers, presentations, analyses



Seminar in Computer Architecture - Spring 2022

Trace: start - schedule

Home

Materials

- Announcements
- Lectures/Schedule
- Lecture Buzzwords
- Readings
- Sections
- Papers
- Synthesis Report
- Homework

Past Course Materials

- Fall 2021
- Spring 2021
- Fall 2020
- Spring 2020
- Fall 2019
- Spring 2019

Resources

Computer Architecture

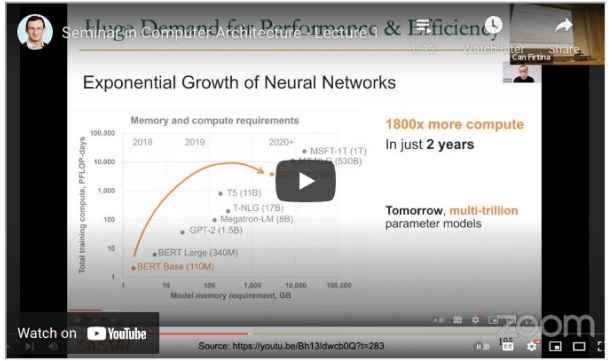
- Fall 2021
- Fall 2021: Lecture Videos
- Fall 2020
- Fall 2020: Lecture Videos
- Fall 2019
- Fall 2019: Lecture Videos
- Fall 2018
- Fall 2018: Lecture Videos

Digital Design and Computer Architecture

- Spring 2021
- Spring 2021: Lecture Videos
- Spring 2020
- Spring 2020: Lecture Videos
- Spring 2019
- Spring 2019: Lecture Videos

Lecture Video Playlist on YouTube

Lecture Playlist



Exponential Growth of Neural Networks

Memory and compute requirements

1800x more compute in just 2 years

Tomorrow, multi-trillion parameter models

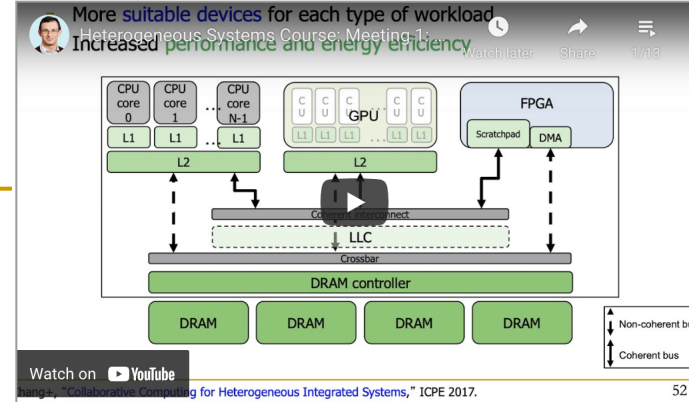
Watch on YouTube

Source: <https://youtu.be/8h13idwcb0Q?t=283>

Spring 2022 Lectures/Schedule

Week	Date	Livestream	Lecture	Readings	Assignments
W1	24.02 Thu.	Yes	L1a: Course Logistics PDF (PPT)	Suggested	
			L1b: Introduction and Basics PDF (PPT)	Suggested	
			L1c: Architectural Design Fundamentals PDF (PPT)	Suggested	
W2	03.03 Thu.	Yes	L2: Memory-Centric Computing PDF (PPT)	Suggested	
W3	10.03 Thu.	Yes	L3: Memory-Centric Computing II PDF (PPT)	Suggested	
W4	17.03 Thu.	Yes	L4: Memory-Centric Computing III PDF (PPT)	Suggested	
W5	24.03 Thu.	Yes	L5: Accelerating Genome Analysis PDF (PPT)	Suggested	
W6	31.03 Thu.	Yes	L6a: Rethinking Virtual Memory I PDF (PPT)	Suggested	
			L6b: Rethinking Virtual Memory II PDF (PPT)	Suggested	
W7	07.04 Thu.	Yes	S1.1: A Logic-in-Memory Computer IEEE Trans. Comput., 1970 PDF (PPT)		
			S1.2: SCA: Scalable Architecture		

Hetero. Systems (Spring'22)



Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=heterogeneous_systems

Youtube Livestream:

- https://www.youtube.com/watch?v=oFO5fTrgFIY&list=PL5Q2soXY2Zi9XrgXR38IM_FTjmY6h7Gzm

Project course

- Taken by Bachelor's/Master's students
- GPU and Parallelism lectures
- Hands-on research exploration
- Many research readings

Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	15.03 Tue.	YouTube Premiere	M1: P&S Course Presentation PDF PPT	Required Materials Recommended Materials	HW 0 Out
W2	22.03 Tue.	YouTube Premiere	M2: SIMD Processing and GPUs PDF PPT		
W3	29.03 Tue.	YouTube Premiere	M3: GPU Software Hierarchy PDF PPT		
W4	05.04 Tue.	YouTube Premiere	M4: GPU Memory Hierarchy PDF PPT		
W5	12.04 Tue.	YouTube Premiere	M5: GPU Performance Considerations PDF PPT		
W6	19.04 Tue.	YouTube Premiere	M6: Parallel Patterns: Reduction PDF PPT		
W7	26.04 Tue.	YouTube Premiere	M7: Parallel Patterns: Histogram PDF PPT		
W8	03.05 Tue.	YouTube Premiere	M8: Parallel Patterns: Convolution PDF PPT		
W9	10.05 Tue.	YouTube Premiere	M9: Parallel Patterns: Prefix Sum (Scan) PDF PPT		
W10	17.05 Tue.	YouTube Premiere	M10: Parallel Patterns: Sparse Matrices PDF PPT		
W11	24.05 Tue.	YouTube Premiere	M11: Parallel Patterns: Graph Search PDF PPT		
W12	01.06 Wed.	YouTube Premiere	M12: Parallel Patterns: Merge Sort PDF PPT		
W13	07.06 Tue.	YouTube Premiere	M13: Dynamic Parallelism PDF PPT		
W14	15.06 Wed.	YouTube Premiere	M14: Collaborative Computing PDF PPT		
W15	24.06 Fri.	YouTube Premiere	M15: GPU Acceleration of Genome Sequence Alignment PDF PPT		
W16	14.07 Thu.	YouTube Premiere	M16: Accelerating Agent-based Simulations PDF ODP		

HW/SW Co-Design (Spring 2022)

■ Spring 2022 Edition:

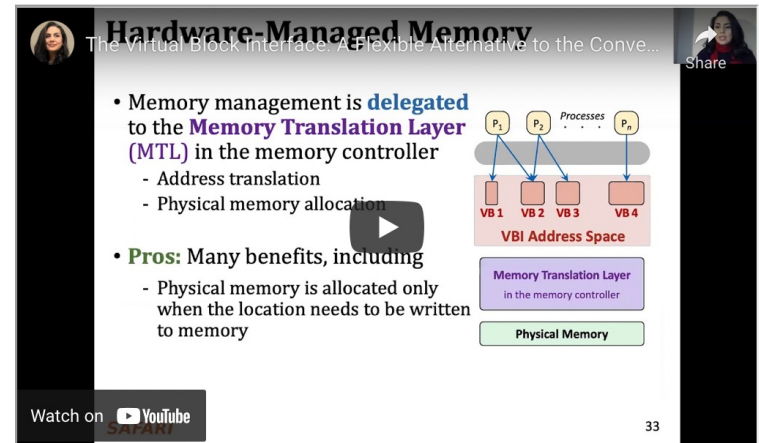
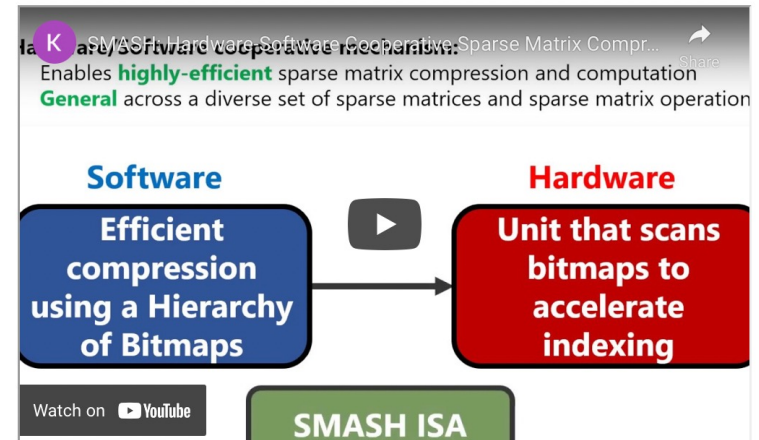
- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=hw_sw_co_design

■ Youtube Livestream:

- <https://youtube.com/playlist?list=PL5Q2soXY2Zi8nH7un3ghD2nutKWWDk-NK>

■ Project course

- Taken by Bachelor's/Master's students
- HW/SW co-design lectures
- Hands-on research exploration
- Many research readings



2022 Meetings/Schedule (Tentative)

Week	Date	Livestream	Meeting	Materials	Assignments
W0	16.03	YouTube Live	Intro to HW/SW Co-Design Zaki (PPTX) Qazi (PDF)	Required	HW 0 Out
W1	23.03		Project selection	Required	
W2	30.03	YouTube Live	Virtual Memory (I) Zaki (PPTX) Qazi (PDF)		
W3	13.04	YouTube Live	Virtual Memory (II) Zaki (PPTX) Qazi (PDF)		

Funding Acknowledgments

- Alibaba, AMD, ASML, [Google](#), Facebook, [Hi-Silicon](#), HP Labs, [Huawei](#), IBM, [Intel](#), [Microsoft](#), Nvidia, Oracle, Qualcomm, Rambus, Samsung, Seagate, [VMware](#), [Xilinx](#)
 - NSF
 - NIH
 - GSRC
 - [SRC](#)
 - CyLab
 - [EFCL](#)
-

Acknowledgments

SAFARI

SAFARI Research Group

safari.ethz.ch

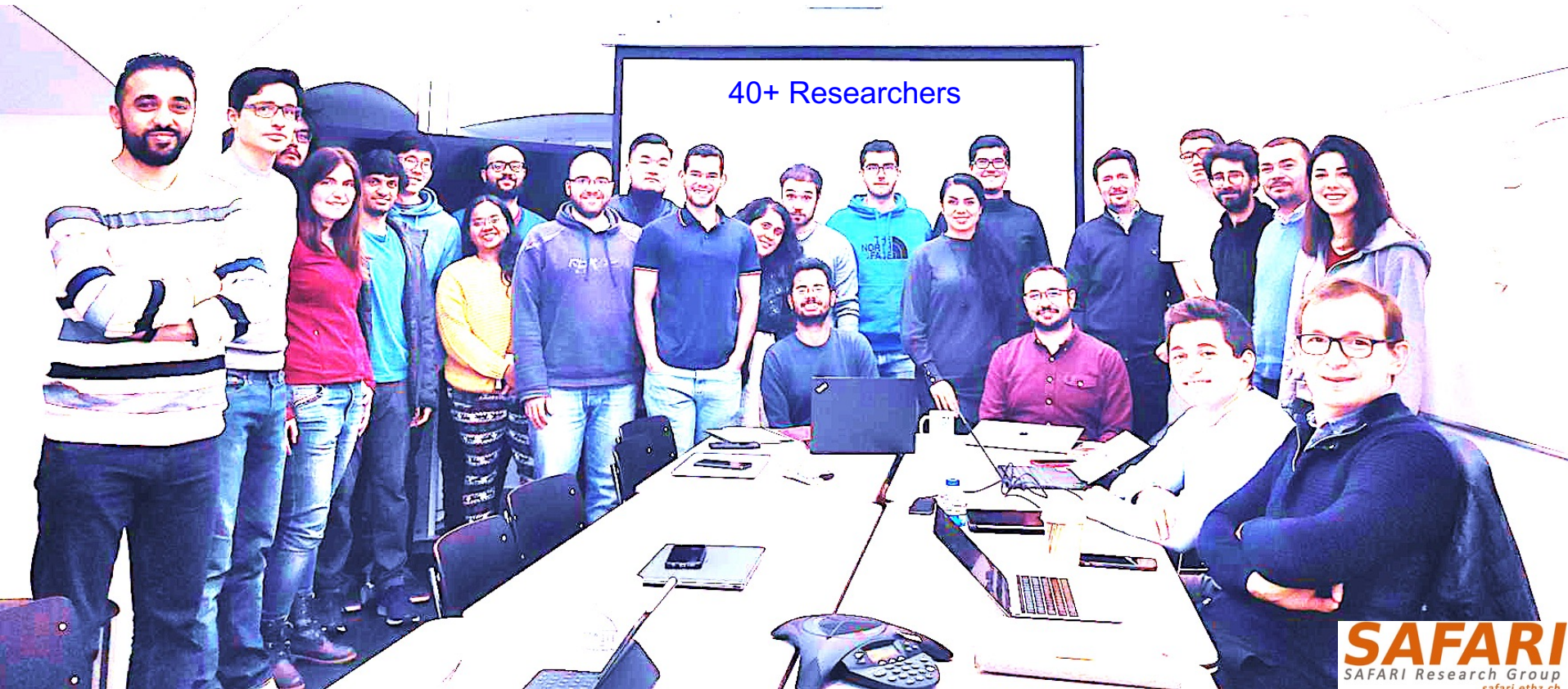
Think BIG, Aim HIGH!

<https://safari.ethz.ch>

Onur Mutlu's SAFARI Research Group

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

<https://safari.ethz.ch/safari-newsletter-january-2021/>



SAFARI
SAFARI Research Group
safari.ethz.ch

Think BIG, Aim HIGH!

SAFARI

<https://safari.ethz.ch>

SAFARI Newsletter April 2020 Edition

- <https://safari.ethz.ch/safari-newsletter-april-2020/>



[View in your browser](#)

Think Big, Aim High



Dear SAFARI friends,

2019 and the first three months of 2020 have been very positive eventful times for SAFARI.

SAFARI Newsletter January 2021 Edition

- <https://safari.ethz.ch/safari-newsletter-january-2021/>



SAFARI
SAFARI Research Group

Newsletter
January 2021

*Think Big, Aim High, and
Have a Wonderful 2021!*



Dear SAFARI friends,

Happy New Year! We are excited to share our group highlights with you in this second edition of the SAFARI newsletter (You can find the first edition from April 2020 [here](#)). 2020 has

SAFARI Newsletter December 2021 Edition

- <https://safari.ethz.ch/safari-newsletter-december-2021/>

SAFARI
SAFARI Research Group

Think Big, Aim High

ETH zürich



View in your browser
December 2021



Referenced Papers, Talks, Artifacts

- All are available at

<https://people.inf.ethz.ch/omutlu/projects.htm>

<https://www.youtube.com/onurmutlulectures>

<https://github.com/CMU-SAFARI/>

Open Source Tools: SAFARI GitHub



SAFARI Research Group at ETH Zurich and Carnegie Mellon University

Site for source code and tools distribution from SAFARI Research Group at ETH Zurich and Carnegie Mellon University.

👤 223 followers 📍 ETH Zurich and Carnegie Mellon U... 🔗 <https://safari.ethz.ch/> ✉ omutlu@gmail.com

[🏠 Overview](#) [📁 Repositories 83](#) [📁 Projects](#) [📦 Packages](#) [👤 Teams 1](#) [👤 People 46](#) [⚙ Settings](#)

Pinned

Customize pins

 **ramulator** Public

A Fast and Extensible DRAM Simulator, with built-in support for modeling many different DRAM technologies including DDRx, LPDDRx, GDDRx, WIOx, HBMx, and various academic proposals. Described in the...

🔴 C++ ⭐ 408 🍴 183

 **prim-benchmarks** Public

PrIM (Processing-In-Memory benchmarks) is the first benchmark suite for a real-world processing-in-memory (PIM) architecture. PrIM is developed to evaluate, analyze, and characterize the first publ...

● C ⭐ 80 🍴 33

 **MQSim** Public

MQSim is a fast and accurate simulator modeling the performance of modern multi-queue (MQ) SSDs as well as traditional SATA based SSDs. MQSim faithfully models new high-bandwidth protocol implement...

🔴 C++ ⭐ 183 🍴 110

 **rowhammer** Public

Source code for testing the Row Hammer error mechanism in DRAM devices. Described in the ISCA 2014 paper by Kim et al. at http://users.ece.cmu.edu/~omutlu/pub/dram-row-hammer_isca14.pdf.

● C ⭐ 200 🍴 40

 **SparseP** Public

SparseP is the first open-source Sparse Matrix Vector Multiplication (SpMV) software package for real-world Processing-In-Memory (PIM) architectures. SparseP is developed to evaluate and characteri...

● C ⭐ 52 🍴 11

 **SoftMC** Public

SoftMC is an experimental FPGA-based memory controller design that can be used to develop tests for DDR3 SODIMMs using a C++ based API. The design, the interface, and its capabilities and limitatio...

🟡 Verilog ⭐ 96 🍴 26

<https://github.com/CMU-SAFARI/>

Some Other Recent Papers

Finding Approximate Seed Matches

- 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"](#)
Preprint in [arXiv](#), 2021.
[[arXiv preprint](#)]
[[BLEND Source Code and Data](#)]

BLEND: A Fast, Memory-Efficient, and Accurate Mechanism to Find Fuzzy Seed Matches

Can Firtina¹ Jisung Park¹ Mohammed Alser¹ Jeremie S. Kim¹ Damla Senol Cali²
Taha Shahroodi³ Nika Mansouri-Ghiasi¹ Gagandeep Singh¹ Konstantinos Kanellopoulos¹
Can Alkan⁴ Onur Mutlu¹
¹*ETH Zurich* ²*Bionano Genomics* ³*TU Delft* ⁴*Bilkent University*

Hardware Acceleration for pHMMs

- Can Firtina, Kamlesh Pillai, Gurpreet S. Kalsi, Bharathwaj Suresh, Damla Senol Cali, Jeremie S. Kim, Taha Shahroodi, Meryem Banu Cavlak, Joel Lindegger, Mohammed Alser, Juan Gómez-Luna, Sreenivas Subramoney, and Onur Mutlu, "[ApHMM: A Profile Hidden Markov Model Acceleration Framework for Genome Analysis](#)"
Preprint in *arXiv*, 2022.
[[Source Code](#)]

ApHMM: A Profile Hidden Markov Model Acceleration Framework for Genome Analysis

Can Firtina¹ Kamlesh Pillai² Gurpreet S. Kalsi² Bharathwaj Suresh² Damla Senol Cali³
Jeremie S. Kim¹ Taha Shahroodi⁴ Meryem Banu Cavlak¹ Joel Lindegger¹ Mohammed Alser¹
Juan Gómez Luna¹ Sreenivas Subramoney² Onur Mutlu¹
¹*ETH Zurich* ²*Intel Labs* ³*Bionano Genomics* ⁴*TU Delft*

Remapping Reads Between References

- Jeremie S. Kim, Can Firtina, Meryem 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"](#)
Preprint in [arXiv](#) and [bioRxiv](#), 2021.
[[bioRxiv preprint](#)]
[[arXiv preprint](#)]
[[AirLift Source Code and Data](#)]

METHOD

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

Jeremie S. Kim^{1†}, Can Firtina^{1†}, Meryem Banu Cavlak², Damla Senol Cali³, Nastaran Hajinazar^{1,4}, Mohammed Alser¹, Can Alkan² and Onur Mutlu^{1,2,3*}

Mapping Constant Regions Between References

- Jeremie S. Kim, Can Firtina, Meryem Banu Cavlak, Damla Senol Cali, Can Alkan, and Onur Mutlu,
["FastRemap: A Tool for Quickly Remapping Reads between Genome Assemblies"](#)
Bioinformatics, btac554.
[[FastRemap Source Code](#)]

FastRemap: A Tool for Quickly Remapping Reads between Genome Assemblies

Jeremie S. Kim¹

Can Firtina¹

Meryem Banu Cavlak¹

Damla Senol Cali^{2,3}

Can Alkan⁴

Onur Mutlu^{1,2,4}

¹*ETH Zürich*

²*Carnegie Mellon University*

³*Bionano Genomics*

⁴*Bilkent University*

Mohammed Alser, Jeremie S. Kim, Nour Almadhoun Alserr, Stefan W. Tell,
Onur Mutlu

[“COVIDHunter: COVID-19 Pandemic Wave Prediction and Mitigation via Seasonality
Aware Modeling”](#)

Frontiers in Public Health 2022

[[Source Code](#)]



ORIGINAL RESEARCH

published: 17 June 2022

doi: 10.3389/fpubh.2022.877621

COVIDHunter: COVID-19 Pandemic Wave Prediction and Mitigation via Seasonality Aware Modeling

Mohammed Alser, Jeremie S. Kim, Nour Almadhoun Alserr, Stefan W. Tell and Onur Mutlu*

Department of Information Technology and Electrical Engineering (D-ITET), ETH Zurich, Zurich, Switzerland

Packaging Omics Methods

Mohammed Alser, Sharon Waymost, Ram Ayyala, Brendan Lawlor, Richard J. Abdill, Neha Rajkumar, Nathan LaPierre, Jaqueline Brito, Andre M. Ribeiro-dos-Santos, Can Firtina, Nour Almadhoun, Varuni Sarwal, Eleazar Eskin, Qiyang Hu, Derek Strong, Byoung-Do (BD)Kim, Malak S. Abedalthagafi, Onur Mutlu, Serghei Mangul

["Packaging, containerization, and virtualization of computational omics methods: Advances, challenges, and opportunities"](#)

arrXiv 2022

Packaging, containerization, and virtualization of computational omics methods: Advances, challenges, and opportunities

Mohammed Alser¹, Sharon Waymost², Ram Ayyala^{3,4}, Brendan Lawlor⁵, Richard J. Abdill⁶, Neha Rajkumar⁷, Nathan LaPierre², Jaqueline Brito⁴, André M. Ribeiro-dos-Santos⁸, Can Firtina¹, Nour Almadhoun¹, Varuni Sarwal², Eleazar Eskin^{2,9,10}, Qiyang Hu¹¹, Derek Strong¹², Byoung-Do (BD) Kim¹², Malak S. Abedalthagafi^{13,14,15*}, Onur Mutlu^{1,*}, Serghei Mangul^{4,*}

Demeter (HD Food Microbiome Profiling)

Taha Shahroodi, Mahdi Zahedi, Can Firtina, Mohammed Alser, Stephan Wong, Onur Mutlu, Said Hamdioui

[“Demeter: A Fast and Energy-Efficient Food Profiler using Hyperdimensional Computing in Memory”](#)

IEEE Access, 2022



RESEARCH ARTICLE

Demeter: A Fast and Energy-Efficient Food Profiler Using Hyperdimensional Computing in Memory

**TAHA SHAHROODI^{ID1}, MAHDI ZAHEDI^{ID1}, CAN FIRTINA², MOHAMMED ALSER^{ID2},
STEPHAN WONG¹, (Senior Member, IEEE), ONUR MUTLU^{ID2}, (Fellow, IEEE),
AND SAID HAMDIOUI^{ID1}, (Senior Member, IEEE)**

¹Q&CE Department, EEMCS Faculty, Delft University of Technology (TU Delft), 2628 CD Delft, The Netherlands

²SAFARI Research Group, D-ITET, ETH Zürich, 8092 Zürich, Switzerland

End of Backup Slides