Hardware-Accelerated Genome Sequencing: A Co-Design Approach

Gagandeep Singh

ETH zürich

AMD
together we advance_

Genome Analysis

NO machine can read the ***entire*** content of a genome

>CCTCCTCAGTGCCACCCAGCCCACTGGCAGCTCCCAAACAGGCTCTTATTAAAACACCCTGTTCCCTGCCCCTTGGAGTGAGGTGTCAAG
GACCTAAACTAAAAAAAAAAAAAGAAAAAGAAAAGAAAAAGAATTTAAAATTTAAGTAATTCTTTGAAAAAAACTAATTTCTAAGCTTCTT
CATGTCAAGGACCTAATGTGCTAAACAGCACTTTTTTGACCATTATTTTGGATCTGAAAGAAATCAAGAATAAATGAAGGACTTGATACATTG
GAAGAGGAGAGTCAAGGACCTACAGAAAAAAAAAAAAAAGAAAAAGAAAAGAAAAAGA**A**TTTAAAATTTAAGTAATTCTTTGAAAAAA
ACTAATTTCTAAGCTTCTT**C**ATGTCAAGGACCTAATGTCTGTGTTGCAGGTCTTCTTGCATTTCCCTGTCAAAAGAAAAAGAATTTAAAATTT
AAGTAATTCTTTGAAAAAAACTAATTTCTAAGCTTCTTCATGTCAAGGACCTAATGTCAGGCCAAGAGTTGCAAAAAAAAAAAAAGAAAAA
GAAAAGAAAAAGAATTTAAAATTTA**A**GTAATTCTTTGAAAAAAACTAATTTCTAAGCTTCTTCATGTCAAGGACCTAATGTAGCCAGAATGG
TTGTGGGATGGGAGCCTCTGTGGACCGACCAGGTAGCTCTCTTTTCCACACTGTAGTCTCAAAGCTTCTTCATGTGGTTTCTCTGAGTGAAA
AAAAAAAAAAGAAAAAGAAAAGAAAAAGAATTTAAAATTTAAGTAATTCTTTGAAAAAAACTAATTTCTAAGCTTT**T**TTCATGTCAAGGACC
TAATGTAGCTATACTGAACGTTATCTAGGGGAAAGATTGAAGGGGAGCTCTAAGGTCAACA……

Mohammed Alser, "Intelligent Genome Analysis", Computer Architecture, ETH Zurich, Fall 2020

Genome Sequencing Pipeline

Obtaining Genomic Sequencing Data

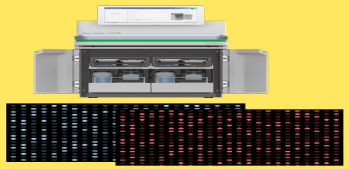
A **DNA Extraction**

DNA Fragmentation

B **Sequencing**

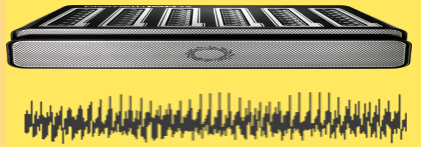
Generating Sequencing Data

Illumina



Multiple images

ONT



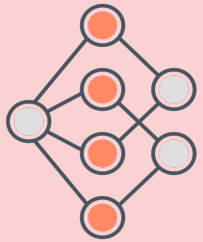
Squiggle

PacBio

30-hour movie

Basecalling

Input signal



CCGTCCCCCCGCAGTAACAT

Read Mapping

A **Indexing**

Reference Genome

Seeds

Seed content	Seed locations
	1, 4, 6
	3, 5, 12
	50, 52
	2, 100

B **Seeding**

Read

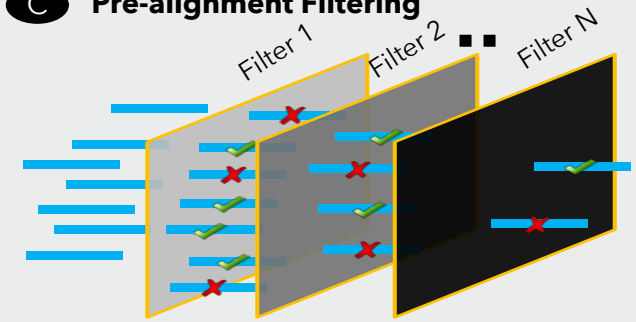
Seeds

Locating common seeds

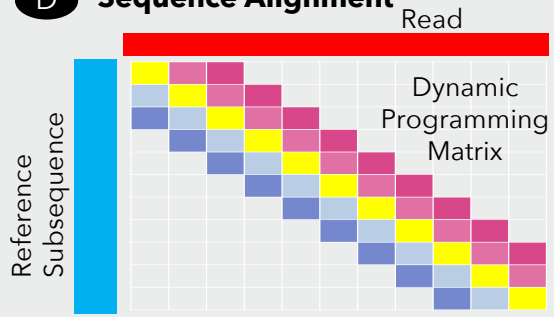
Locations of common seeds

1, 4, 6
3, 5, 12
2, 100

C **Pre-alignment Filtering**



D **Sequence Alignment**

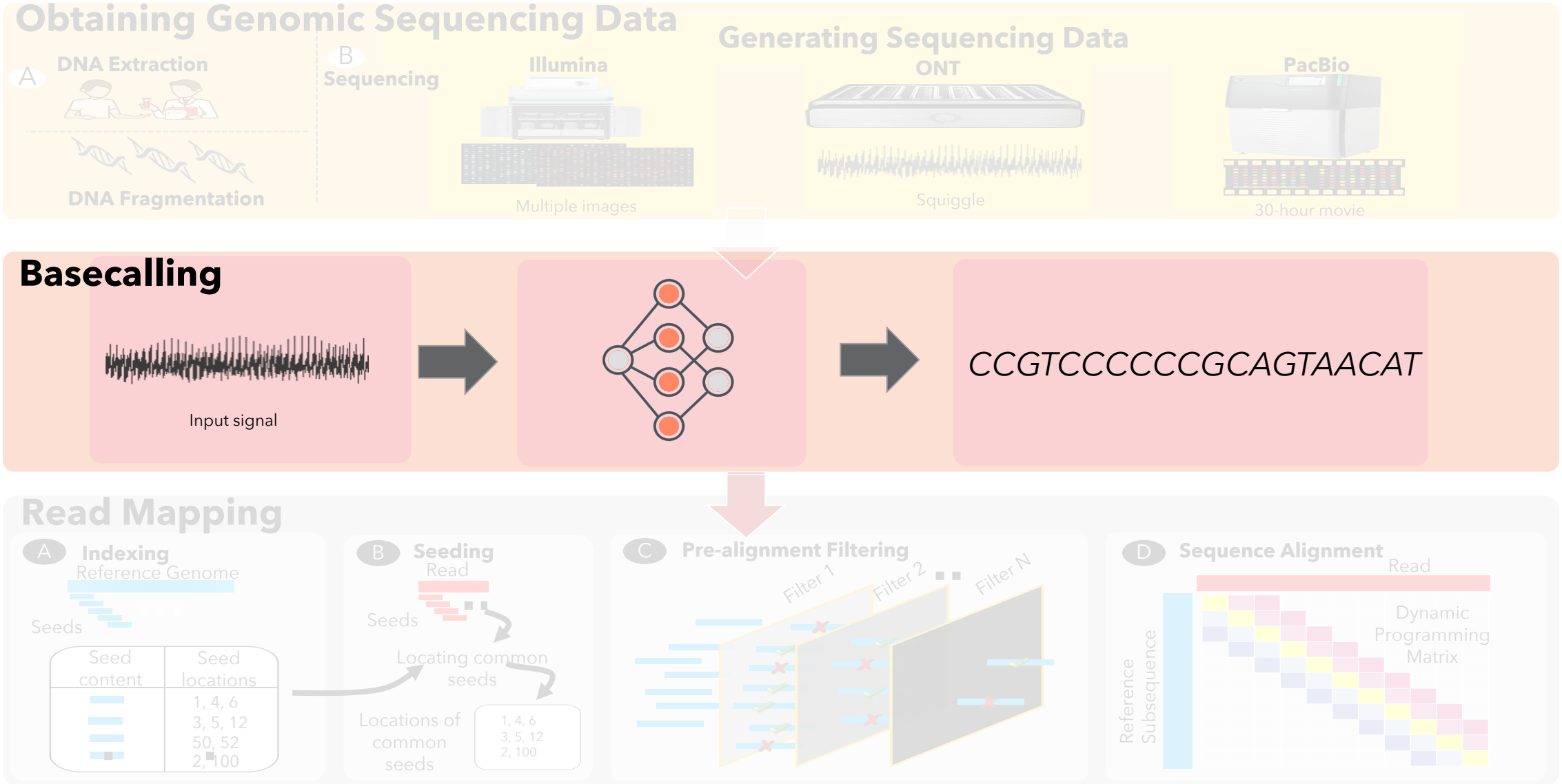


SAFARI

Mohammed Alser et al., "From molecules to genomic variations: Accelerating genome analysis via intelligent algorithms and architectures", Computational and Structural Biotechnology Journal, Volume 20, 2022

AMD
together we advance_

Genome Sequencing Pipeline



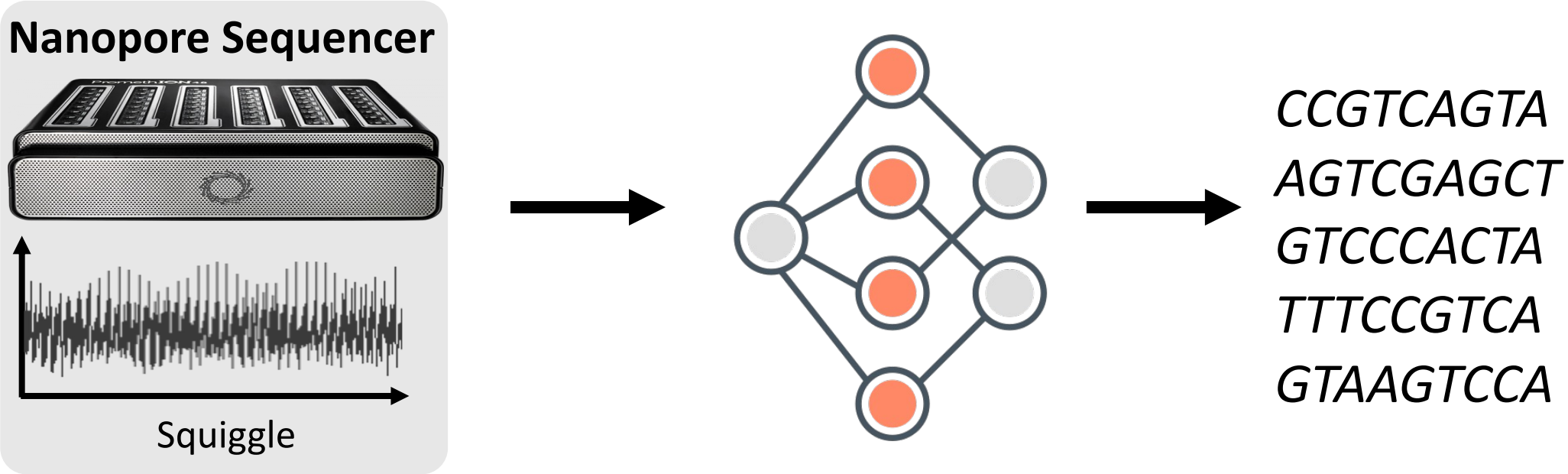
SAFARI

Mohammed Alser et al., "From molecules to genomic variations: Accelerating genome analysis via intelligent algorithms and architectures", Computational and Structural Biotechnology Journal, Volume 20, 2022

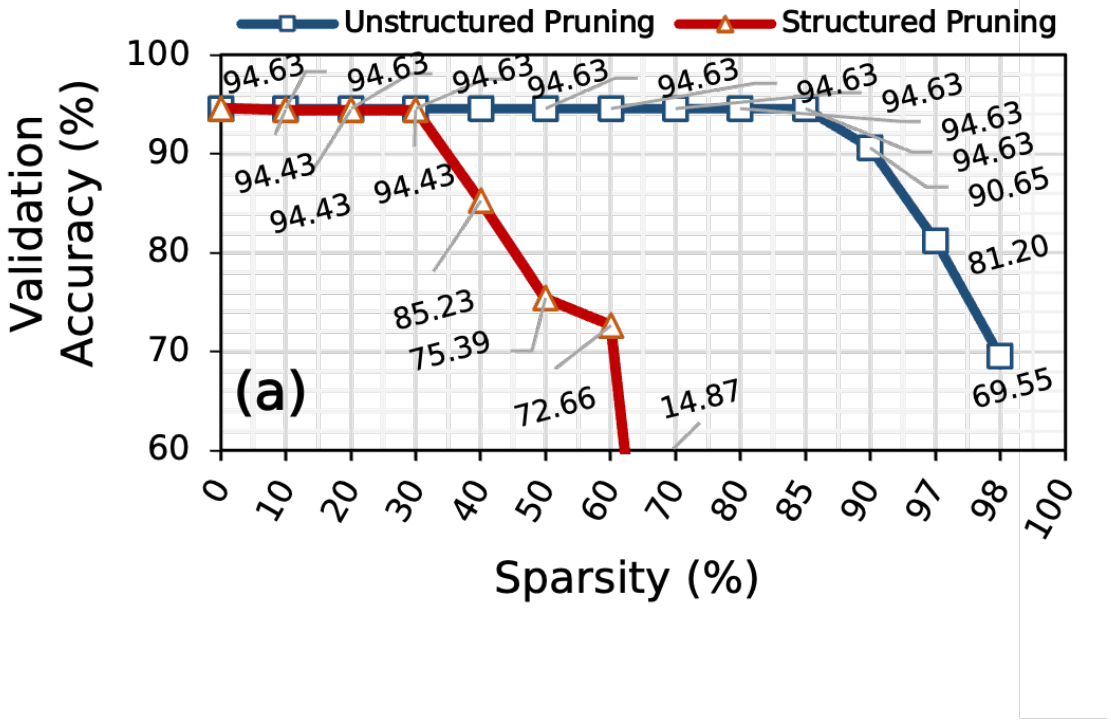
AMD
together we advance_

Basecalling

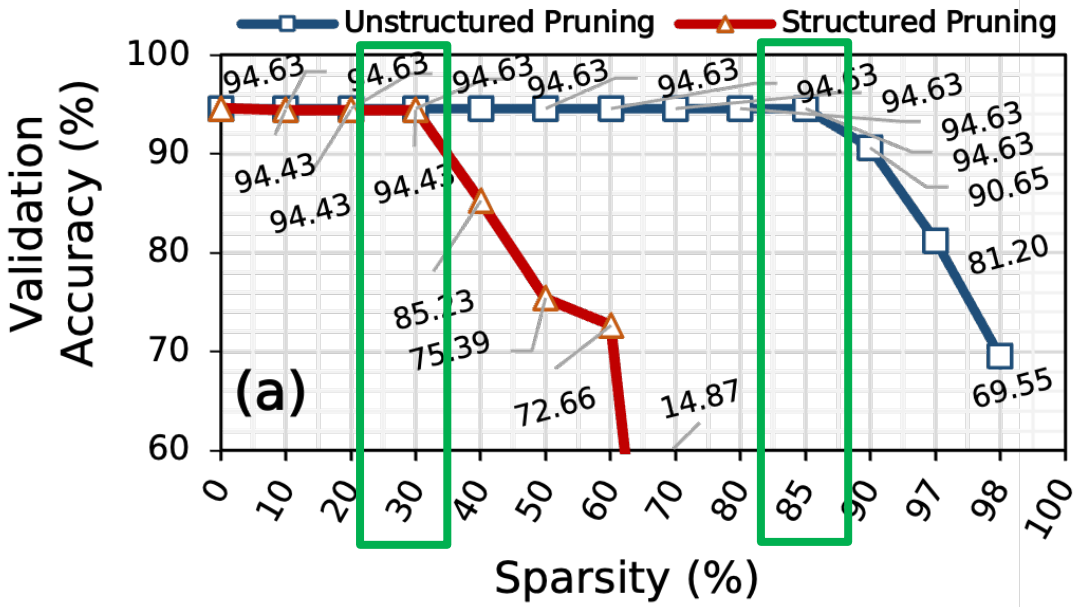
- **Basecalling is the first step in the genomics pipeline** that converts noisy electrical signals to nucleotide bases (i.e., A, C, G, T)
- Modern basecallers **use complex deep learning-based models**

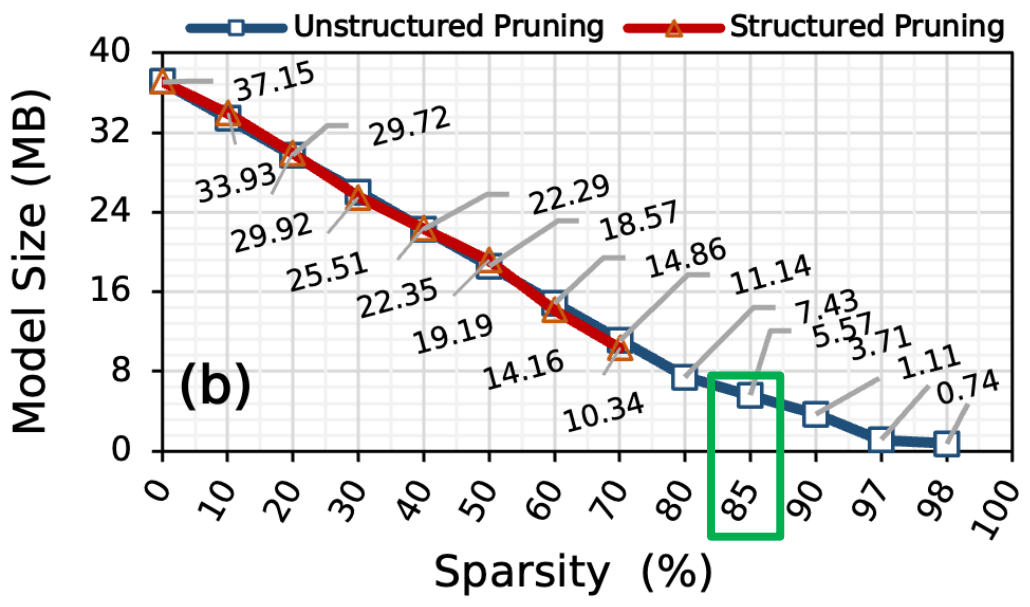


Motivation: Effect of Pruning (1/2)



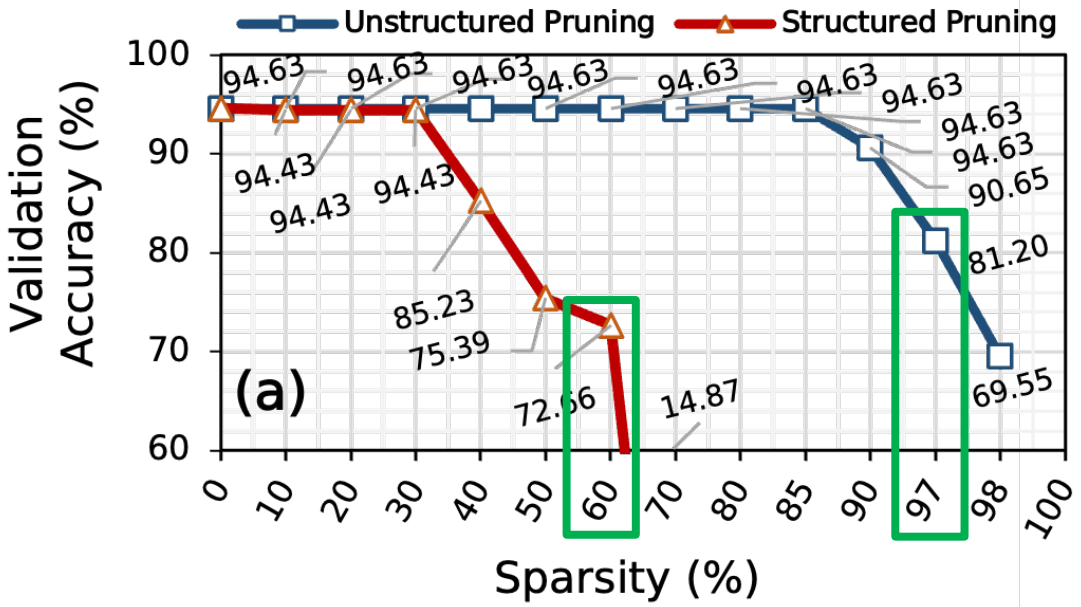
Motivation: Effect of Pruning (1/2)

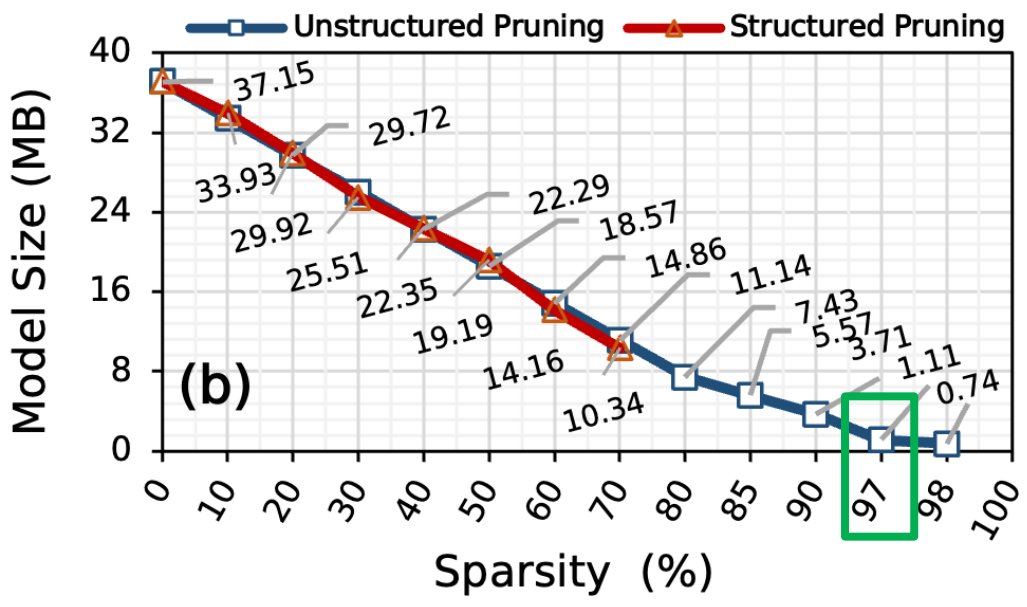




85% of weights can be pruned leading to
6.67x lower model size without any loss in accuracy

Motivation: Effect of Pruning (1/2)

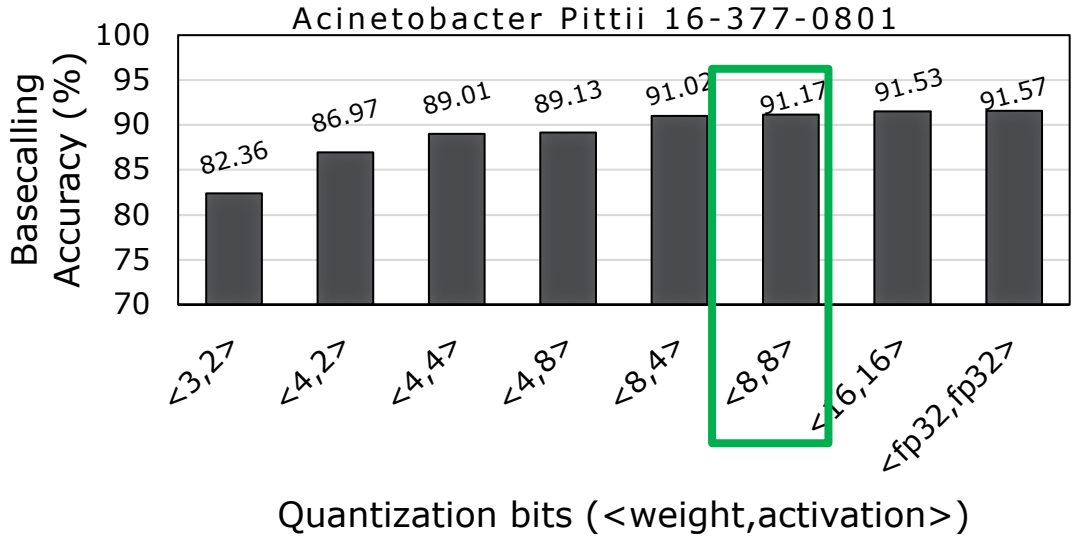


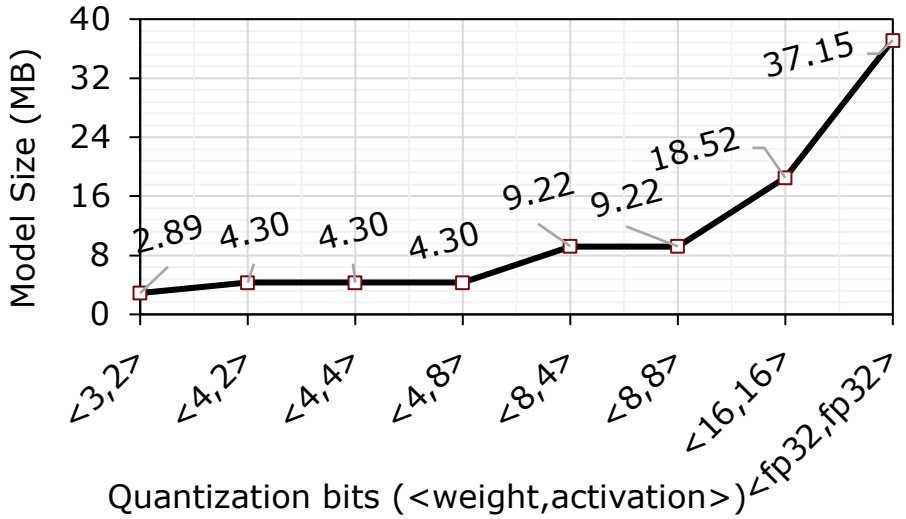


97% of weights can be pruned leading to 33.33x lower model size while providing 81.20% accuracy

Basecallers are often **adapted** from the speech recognition domain **leading to over-parametrized models**

Motivation: Effect of Quantization (2/2)





Provides full accuracy with 4x lower bits for weights and activations

Basecallers use **floating-point precision** to represent **each neural network layer**

Our Goal

Develop a comprehensive framework **for specializing and optimizing** deep learning-based basecallers that provides **high efficiency and performance**

Our Proposal

Framework for Designing Efficient Deep Learning-Based Genomic Basecallers

RUBICON Framework

RUBICON provides **two key mechanisms**

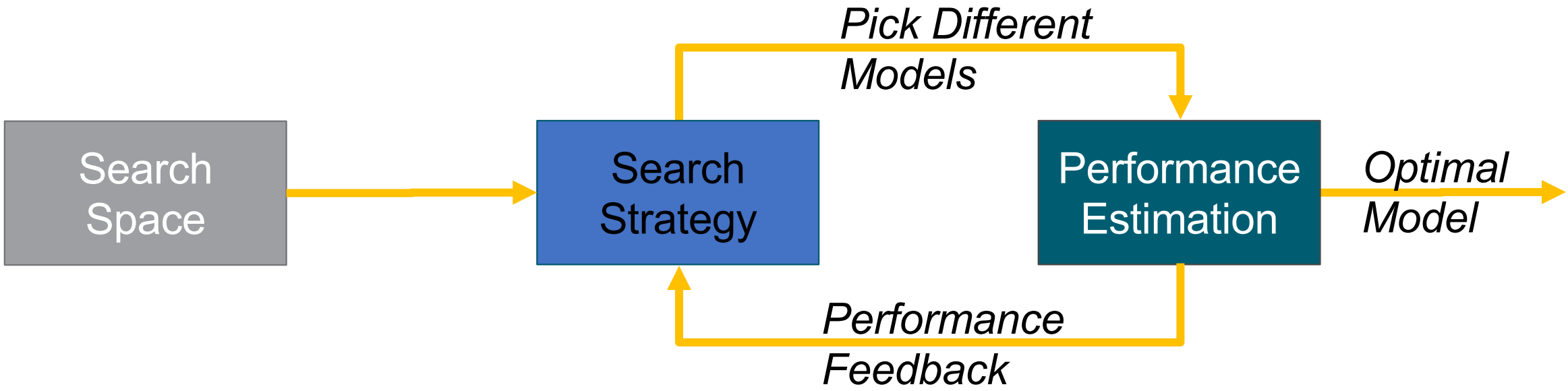
QABAS: Quantization-aware basecalling architecture search

SkipClip: Skip connection removal by teaching

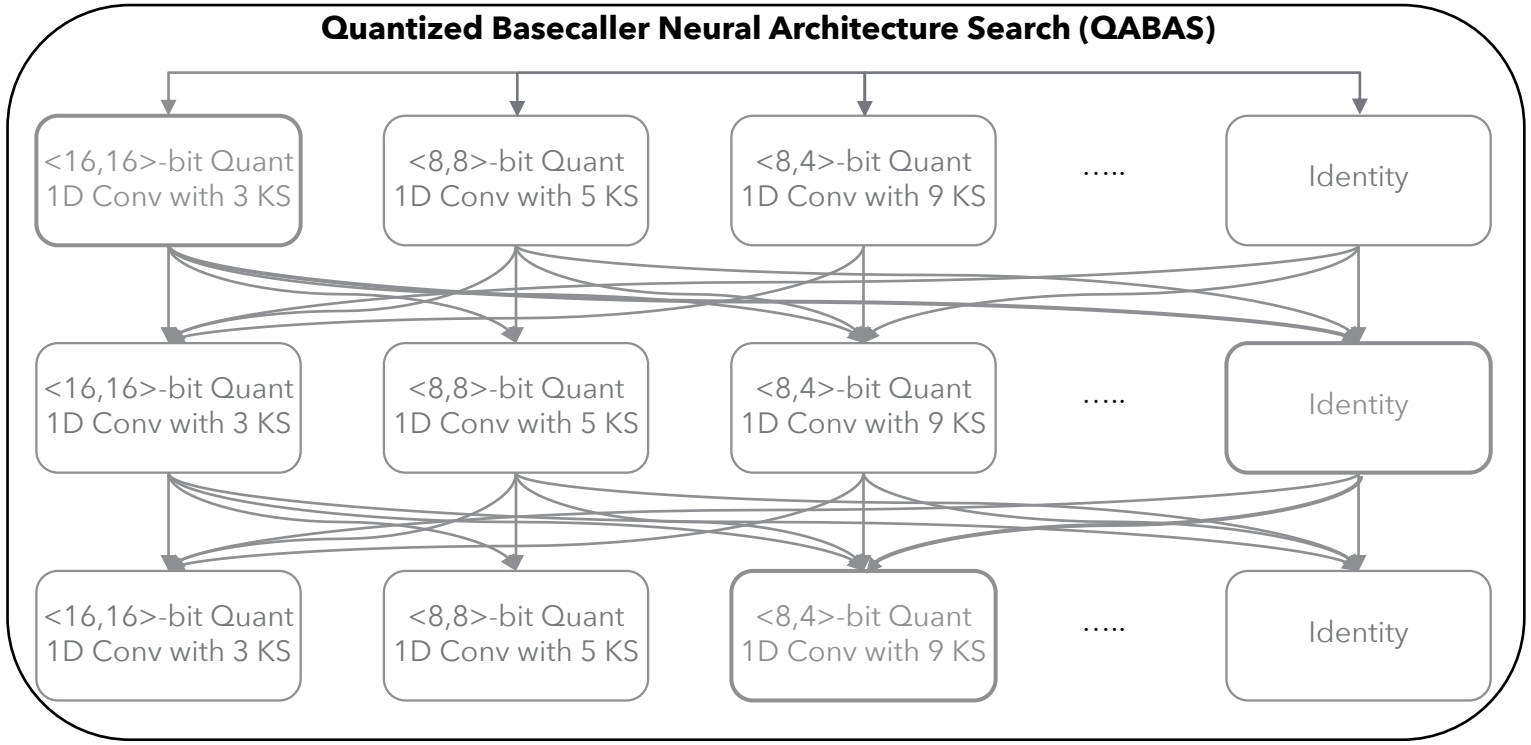
QABAS: Quantization-Aware Basecalling Architecture Search

- QABAS **automates** the process of finding efficient and high-performance hardware-aware genomics basecallers

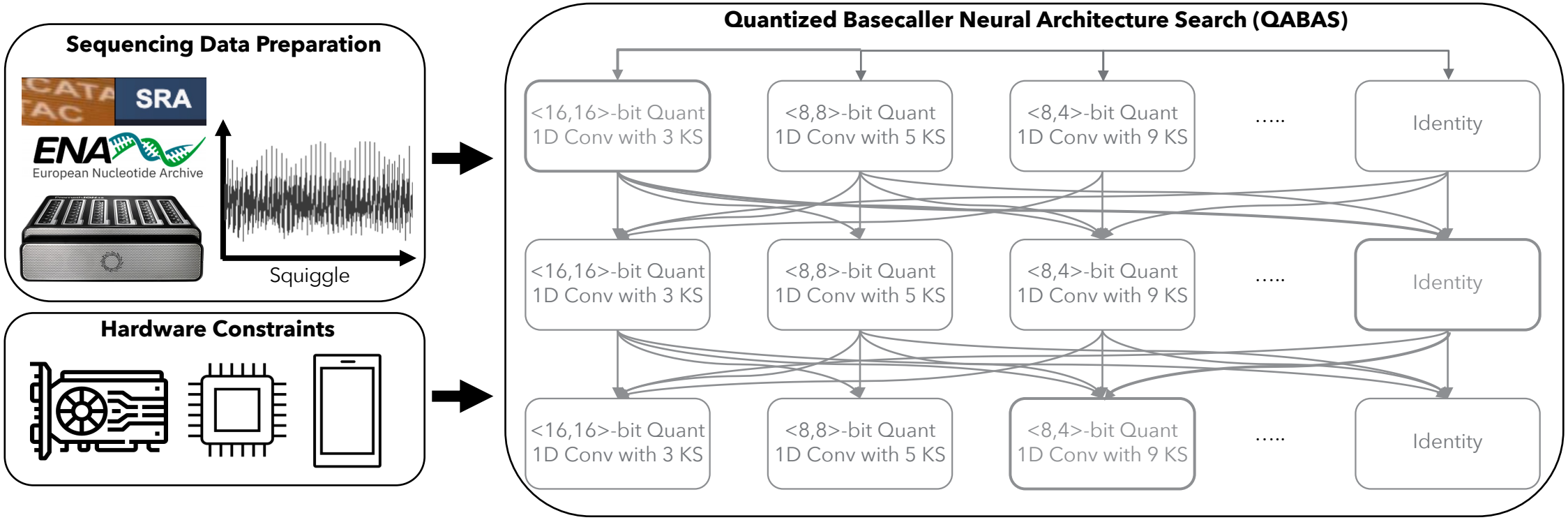
- QABAS **uses neural architecture search (NAS) to evaluate millions of different basecaller architectures**



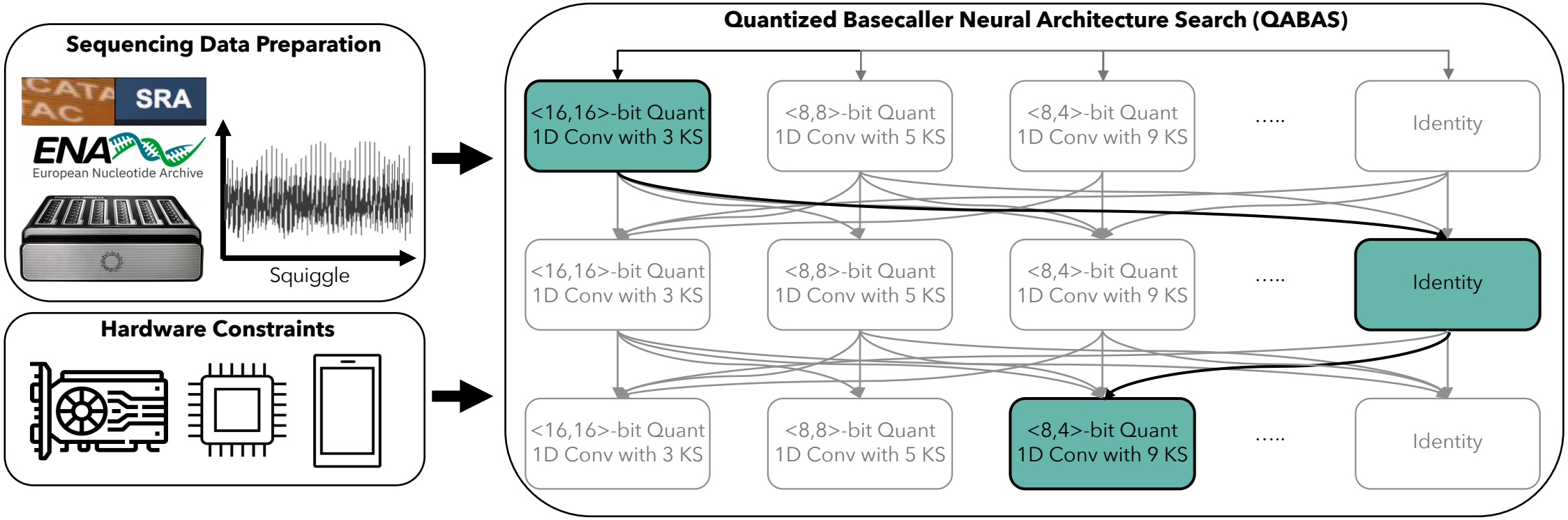
QABAS: Quantization-Aware Basecalling Architecture Search



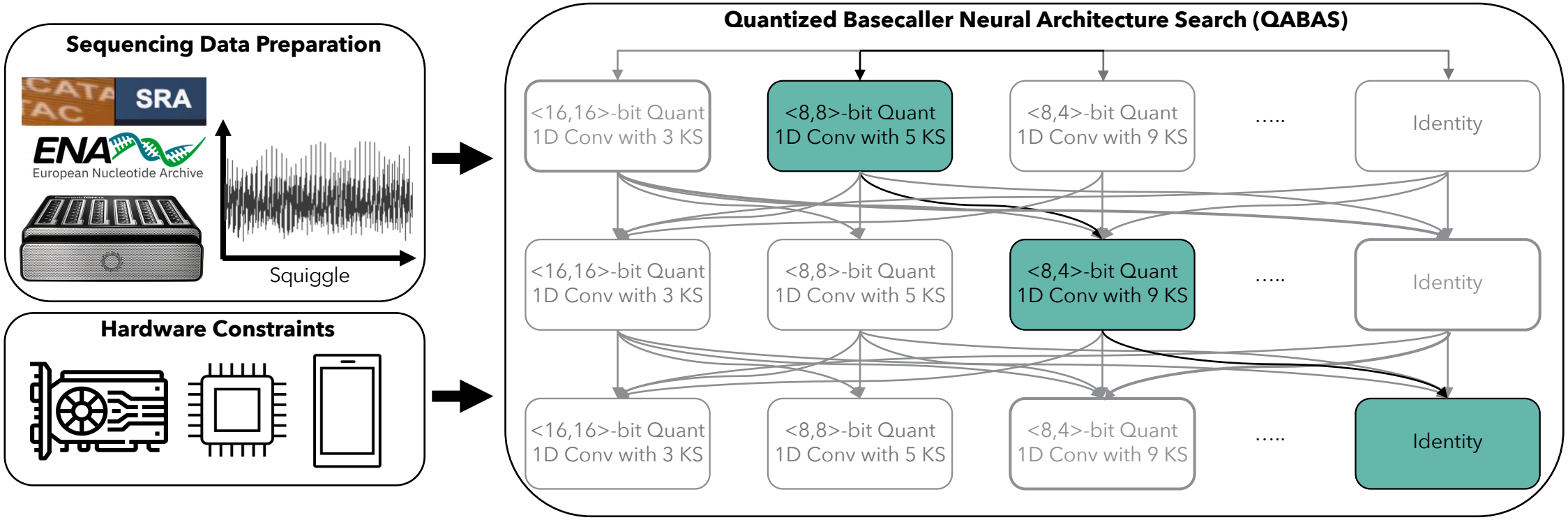
QABAS: Quantization-Aware Basecalling Architecture Search



QABAS: Quantization-Aware Basecalling Architecture Search

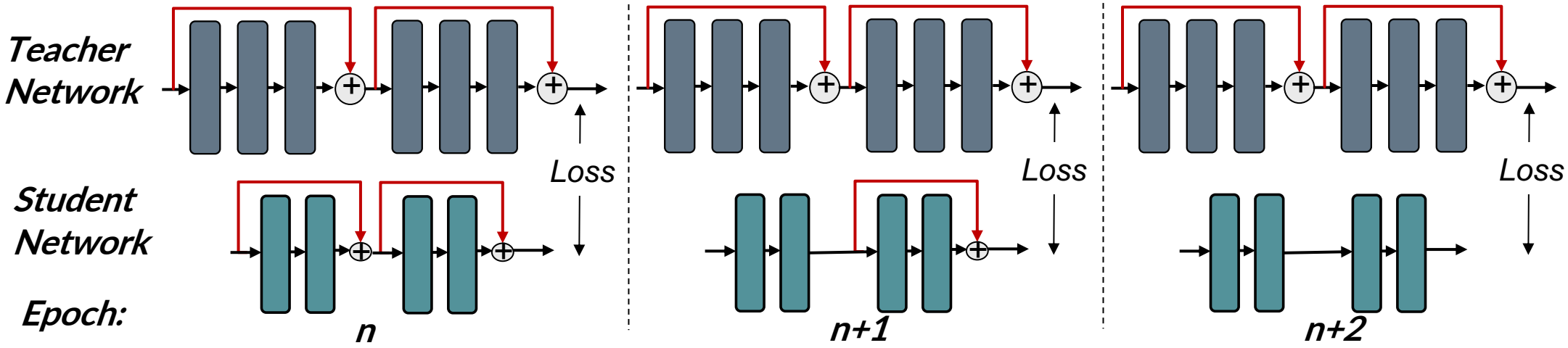


QABAS: Quantization-Aware Basecalling Architecture Search



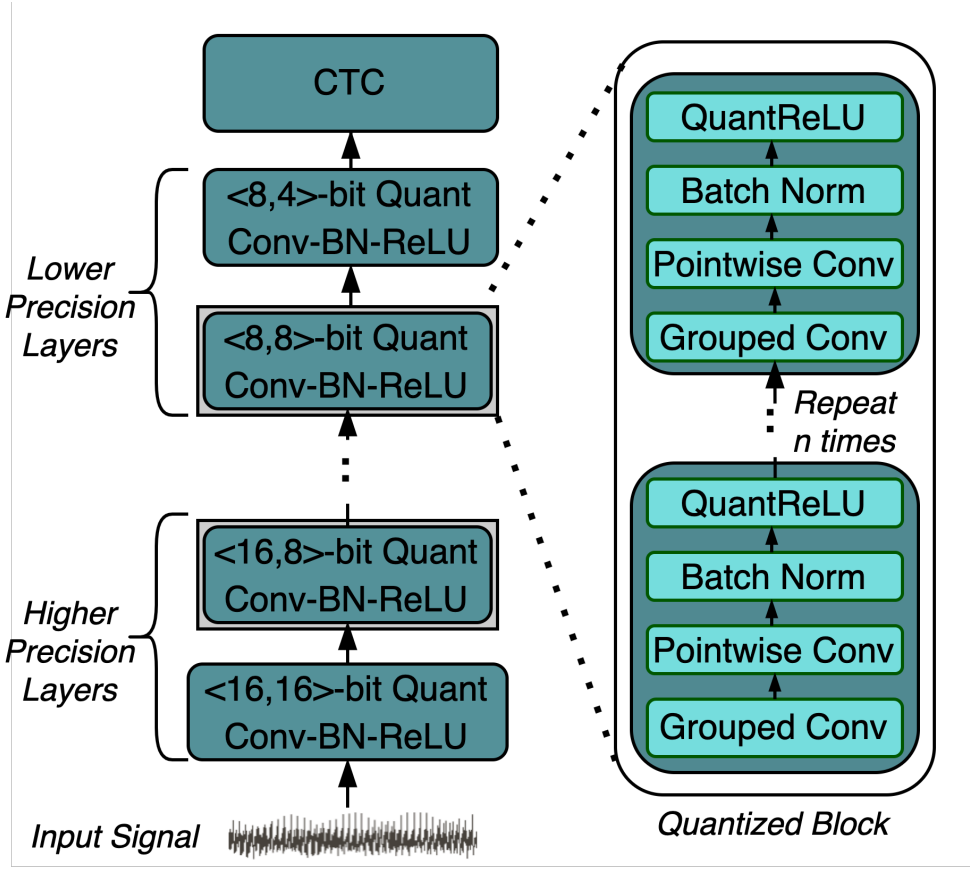
SkipClip: Skip Connection Removal by Teaching

- SkipClip removes all the skip connections present in modern **basecallers to reduce resource and storage requirements without any loss in basecalling accuracy**
- SkipClip **uses knowledge distillation**, where we train a smaller network (student) without skip connections to mimic a pre-trained bigger network (teacher) with skip connections



RUBICALL: A Hardware-Optimized Basecaller

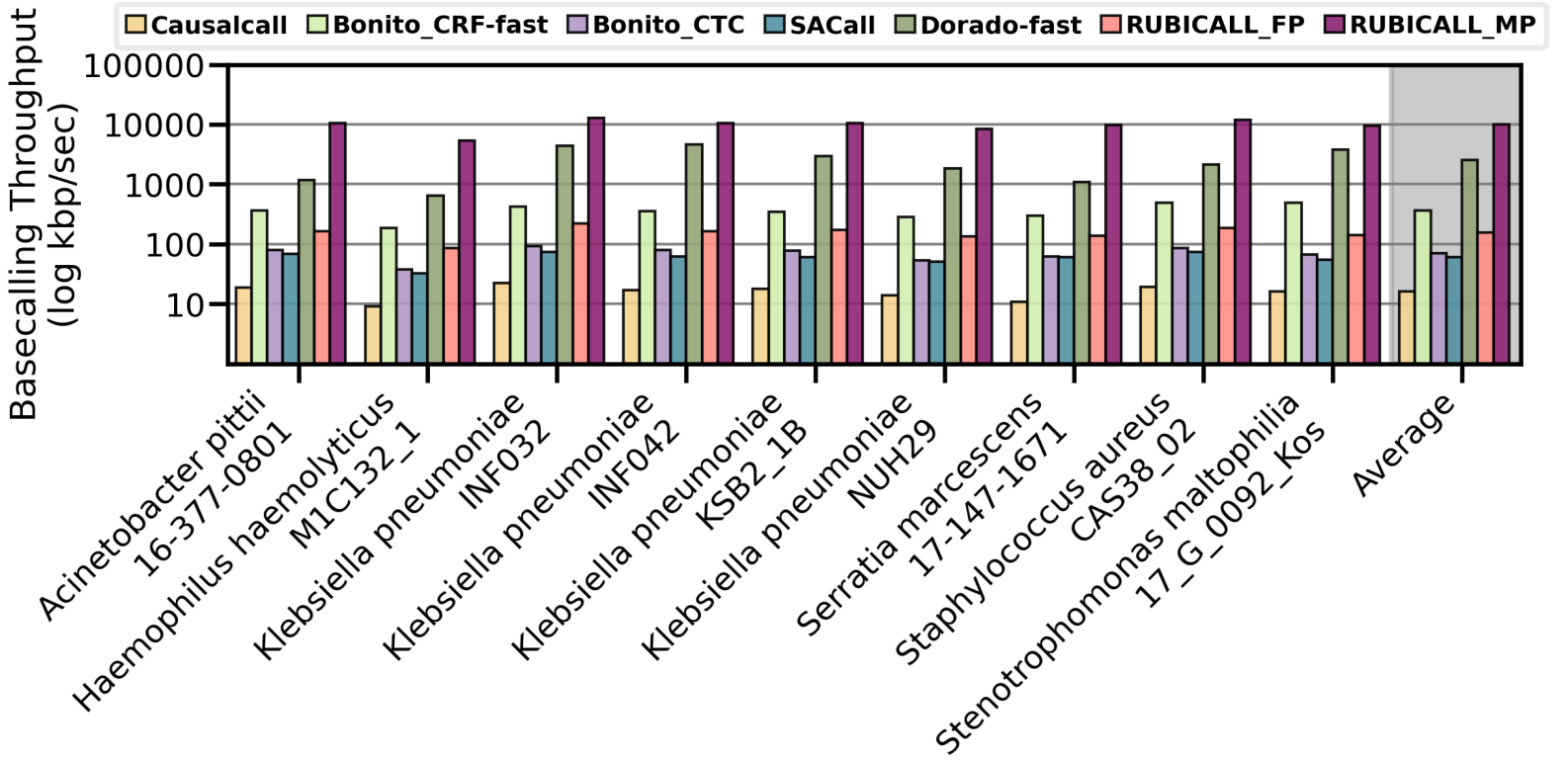
- RUBICALL is **developed using QABAS and SkipClip**
- RUBICALL is uses **mixed-precision computation**



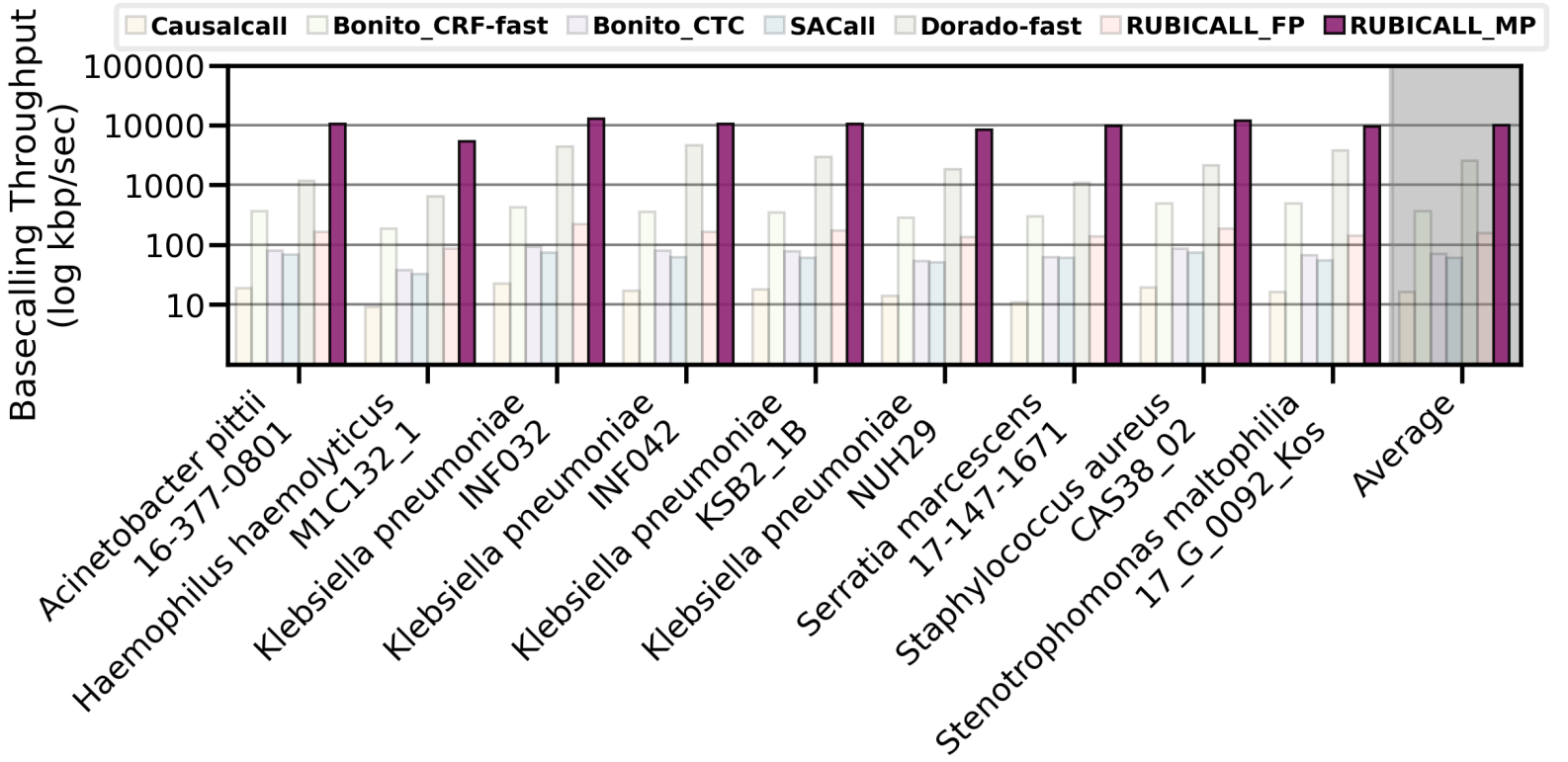
Evaluation Methodology

- Comparison to **five state-of-the-art basecallers**
  - **Bonito-CTC**, an expert-designed convolutional neural network-based basecaller from ONT
  - **Bonito-CRF-fast**, a throughput-optimized recurrent neural network-based basecaller from ONT
  - **Dorado-fast,** a LibTorch version of Bontio-CRF_fast that is optimized for low precision
  - **SACall**, a transformer-based basecaller with attention mechanism
  - **Causalcall**, a state-of-the-art hand-tuned basecaller
- We evaluate two versions of **RUBICALL**
  - **RUBICALL-MP** using mixed-precision computation
  - **RUBICALL-FP** using 32-bit floating-point precision computation

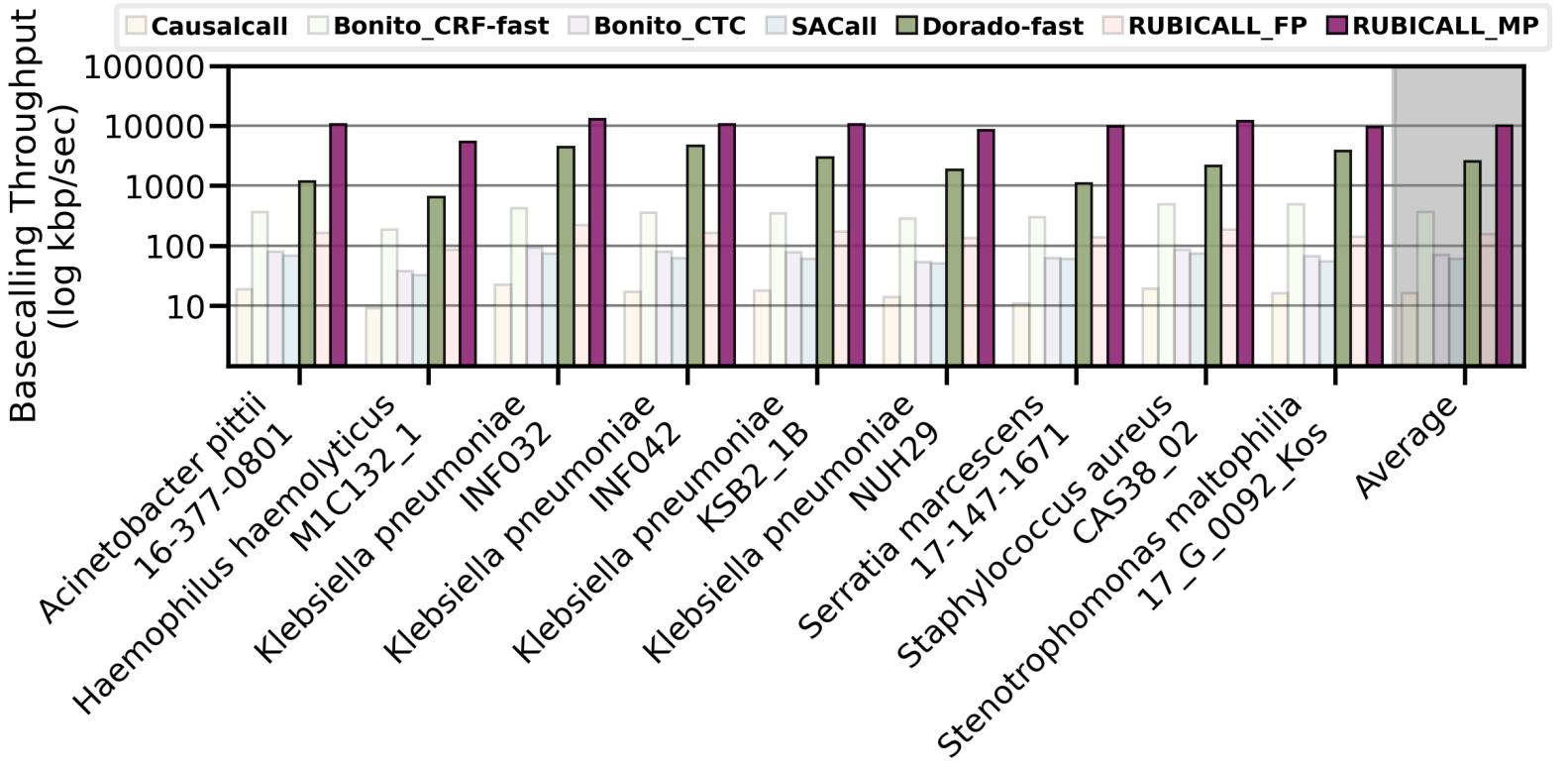
Basecalling Throughput



Basecalling Throughput

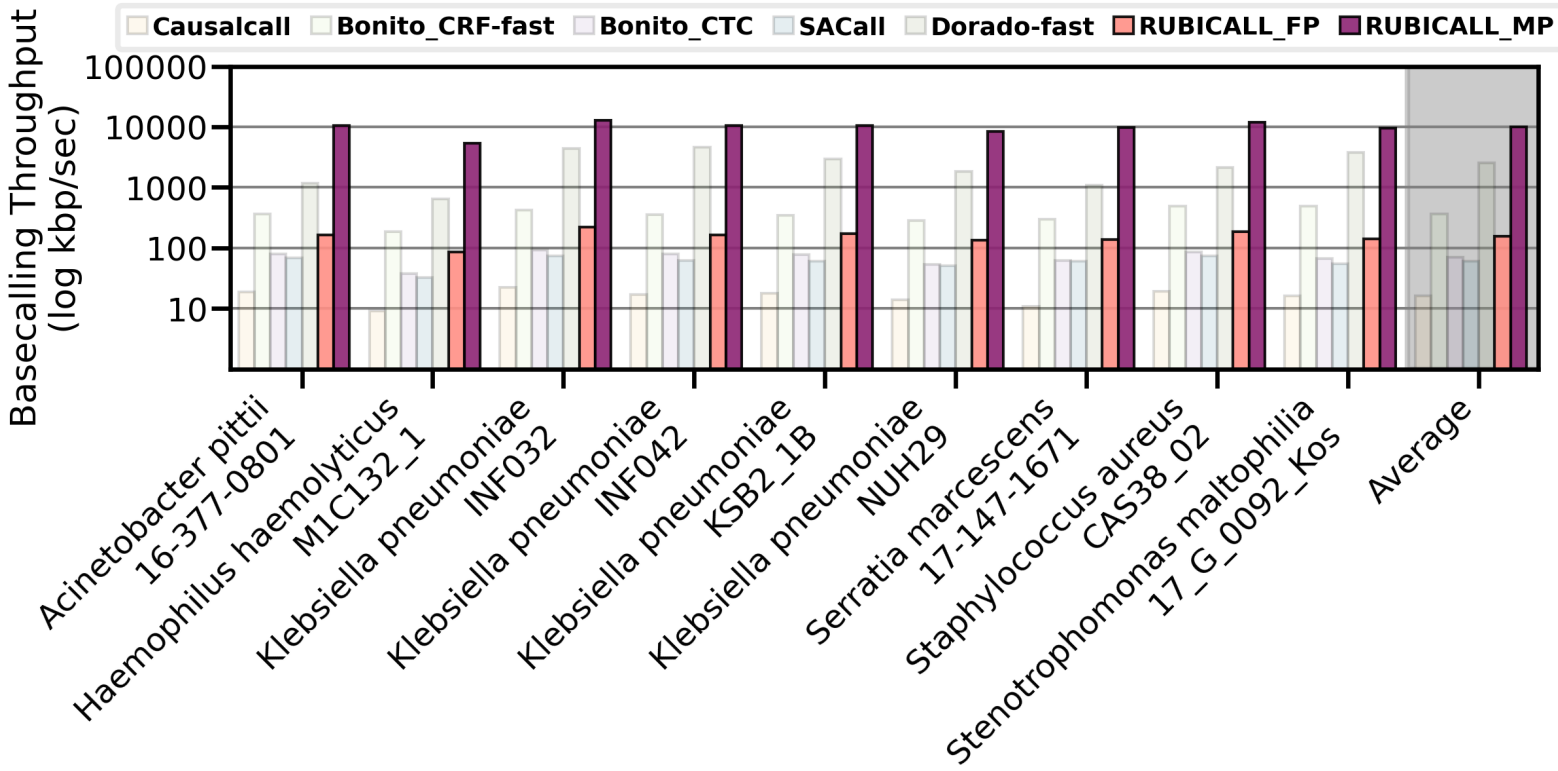


Basecalling Throughput



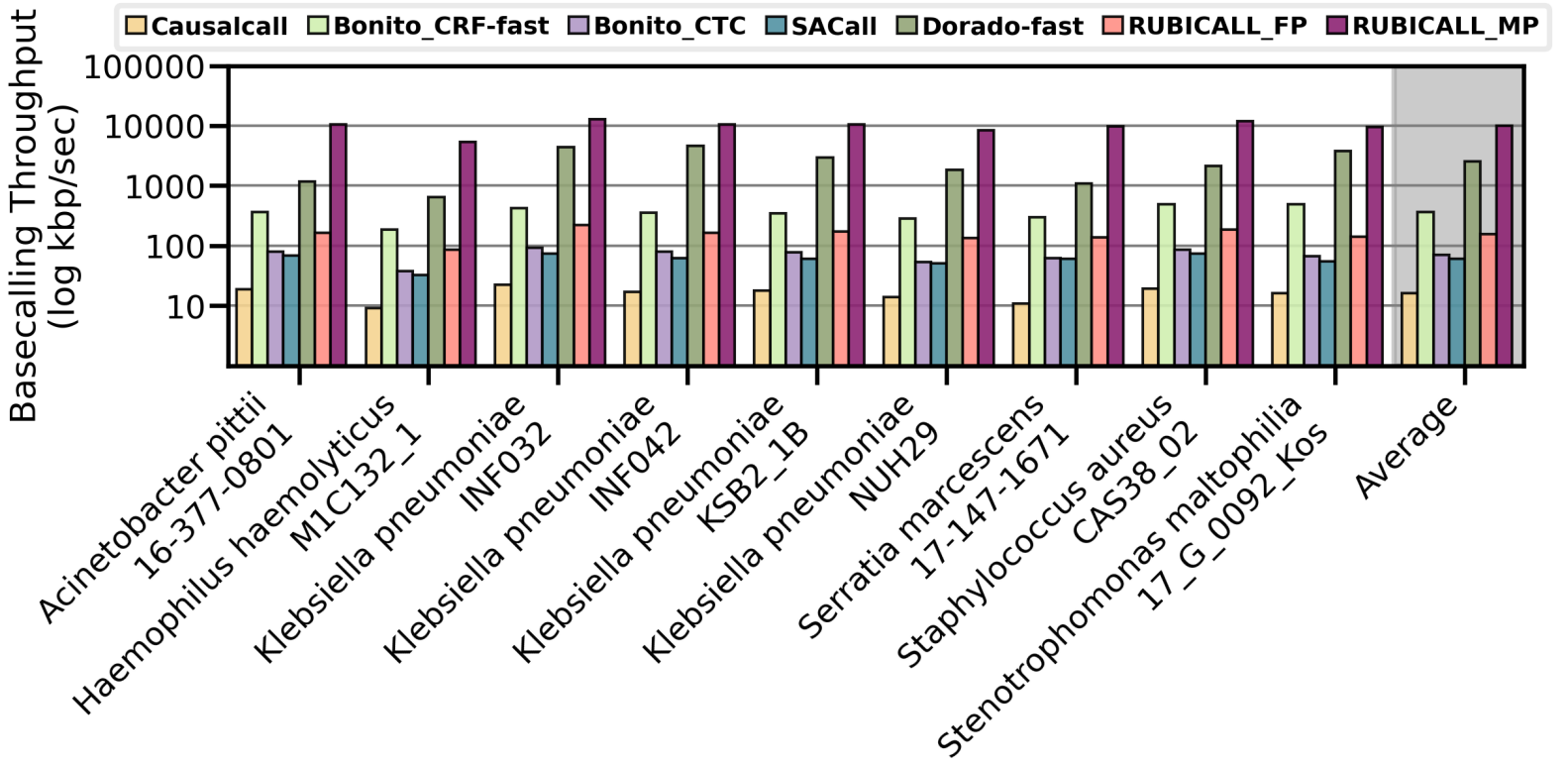
RUBICALL-MP **outperforms** Dorado-fast by **3.96x**

Basecalling Throughput



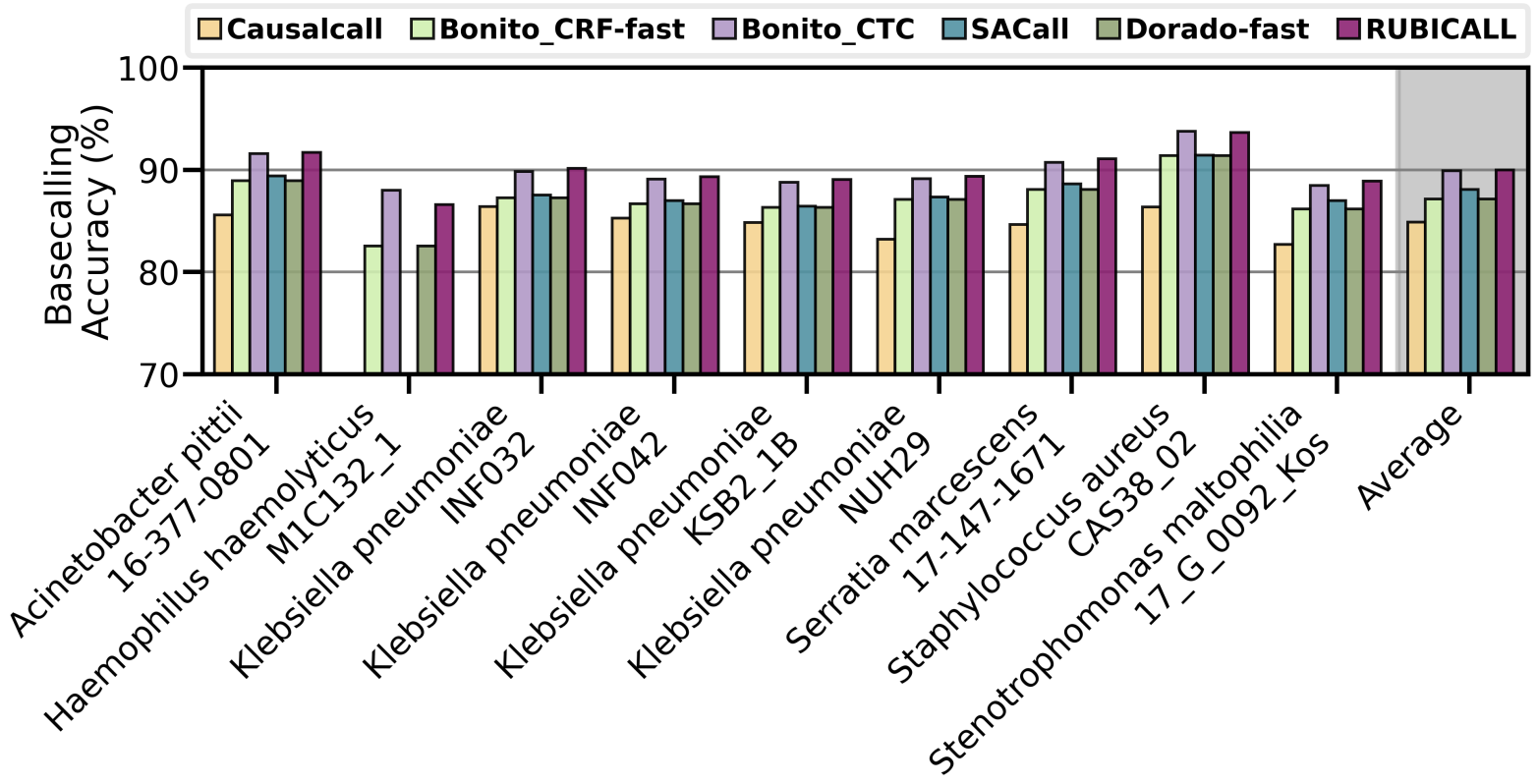
RUBICALL-MP provides **63.61x higher performance** when compared to RUBICALL-FP

Basecalling Throughput

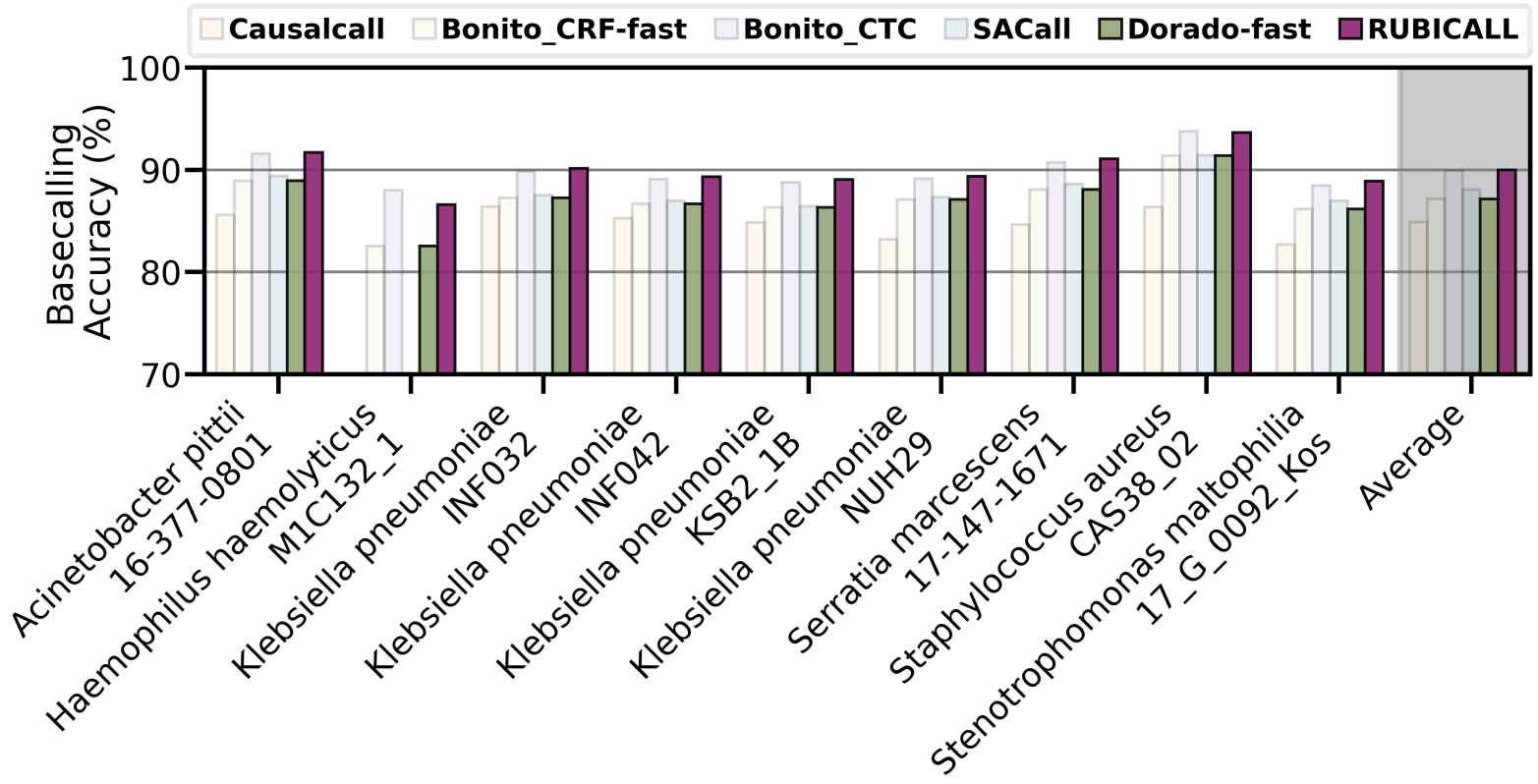


RUBICALL-MP **consistently outperforms** all the evaluated basecallers

Basecalling Accuracy

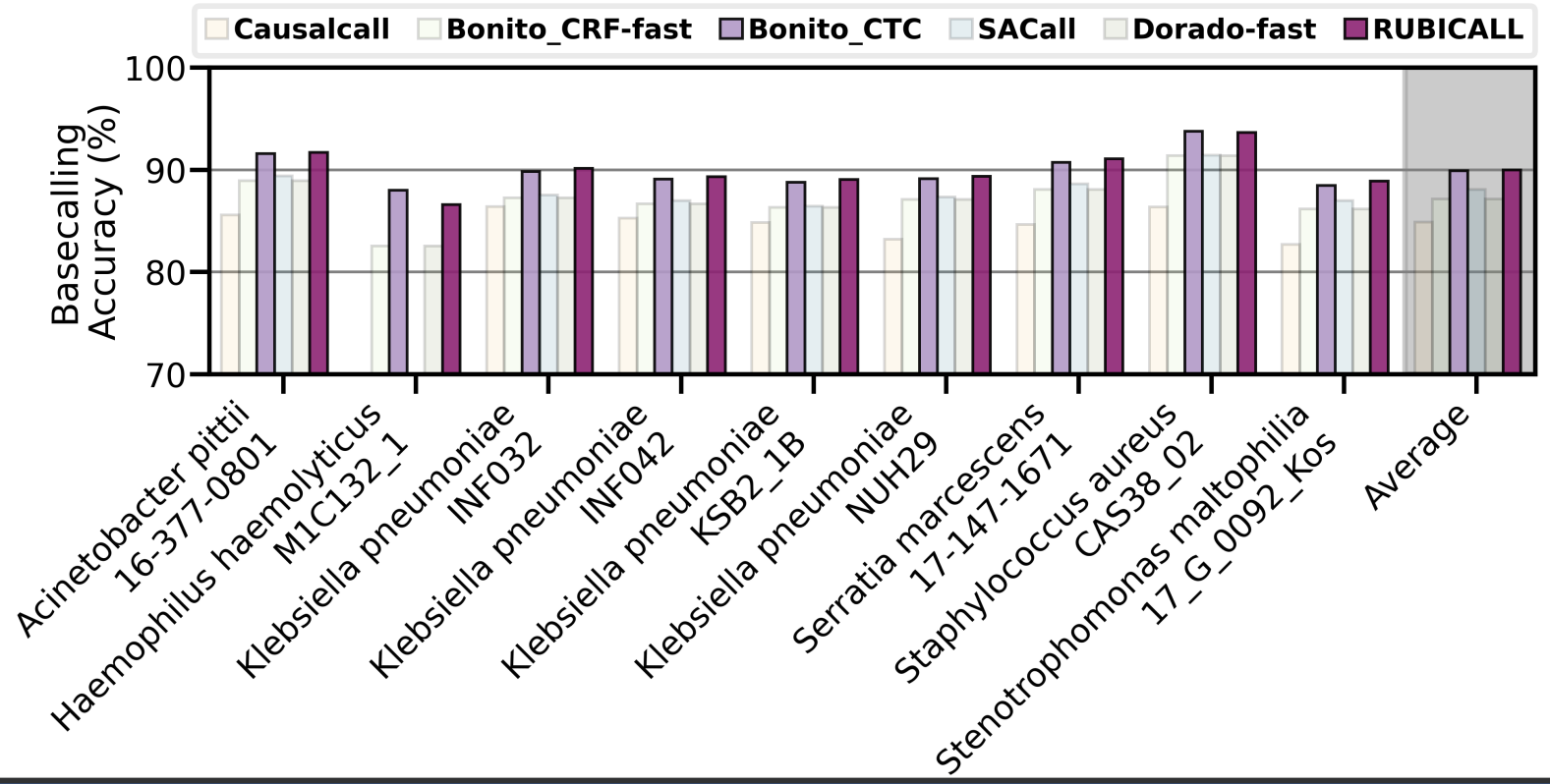


Basecalling Accuracy



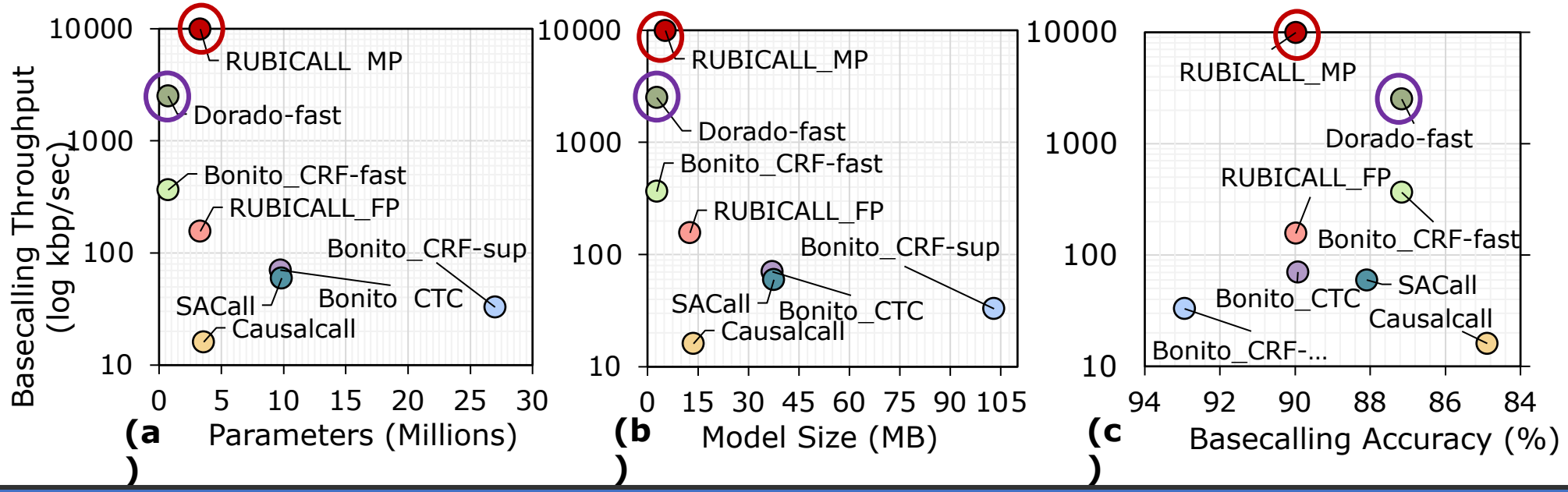
RUBICALL provides **2.97% higher accuracy** than Dorado-fast

Basecalling Accuracy



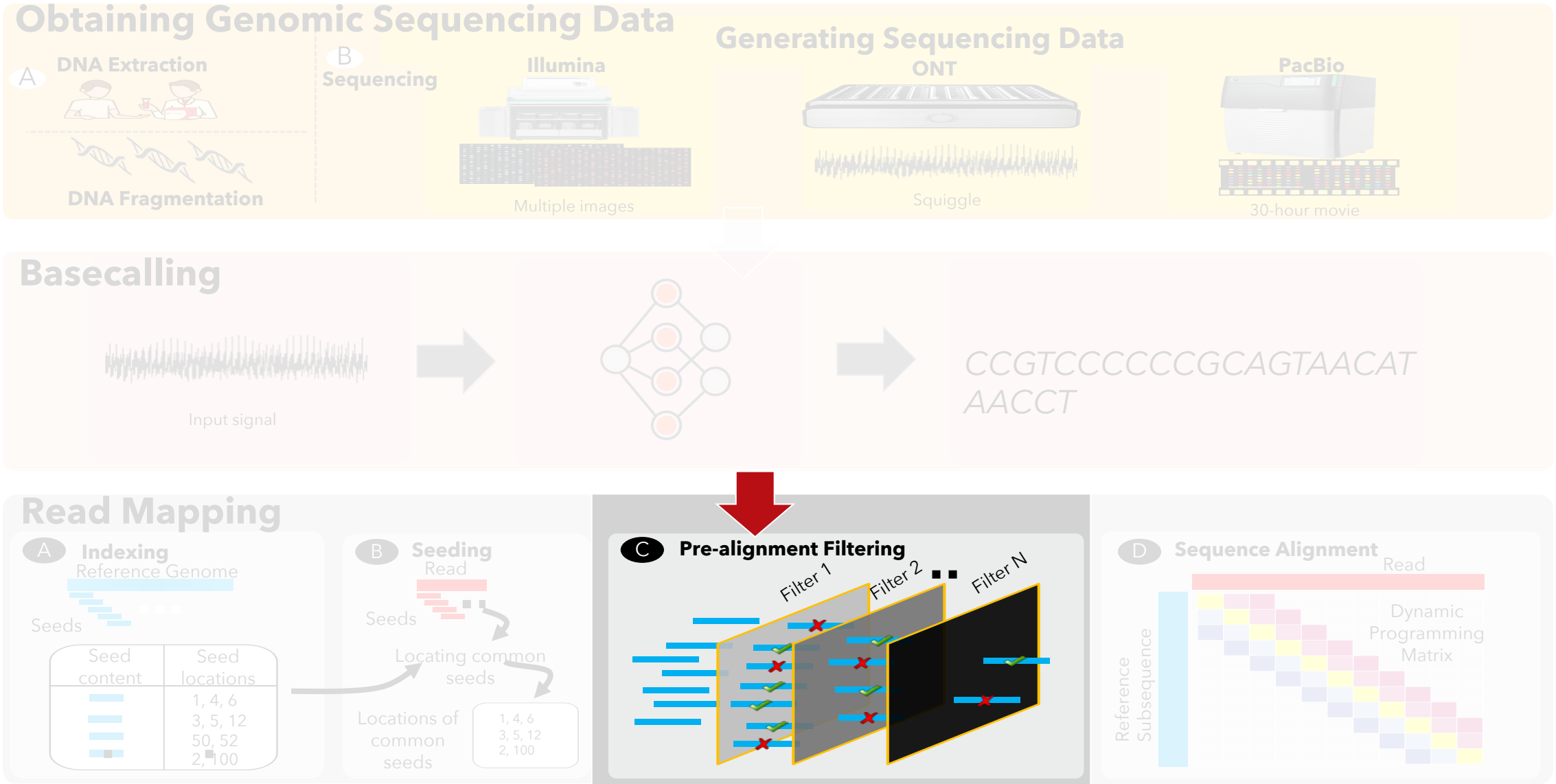
RUBICALL provides **similar accuracy** to an expert-designed basecaller while being **4.17x and 141.15x faster** with RUBICALL-FP and RUBICALL-MP, respectively

Key Results



RUBICALL-MP provides the ability to basecall **accurately, quickly, and efficiently** scale basecalling by providing **reductions in both model size and neural network model parameters**

Genome Sequencing Pipeline



SAFARI

Mohammed Alser et al., "From molecules to genomic variations: Accelerating genome analysis via intelligent algorithms and architectures", Computational and Structural Biotechnology Journal, Volume 20, 2022, Pages 4579-4599

Near-Memory Acceleration

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]

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

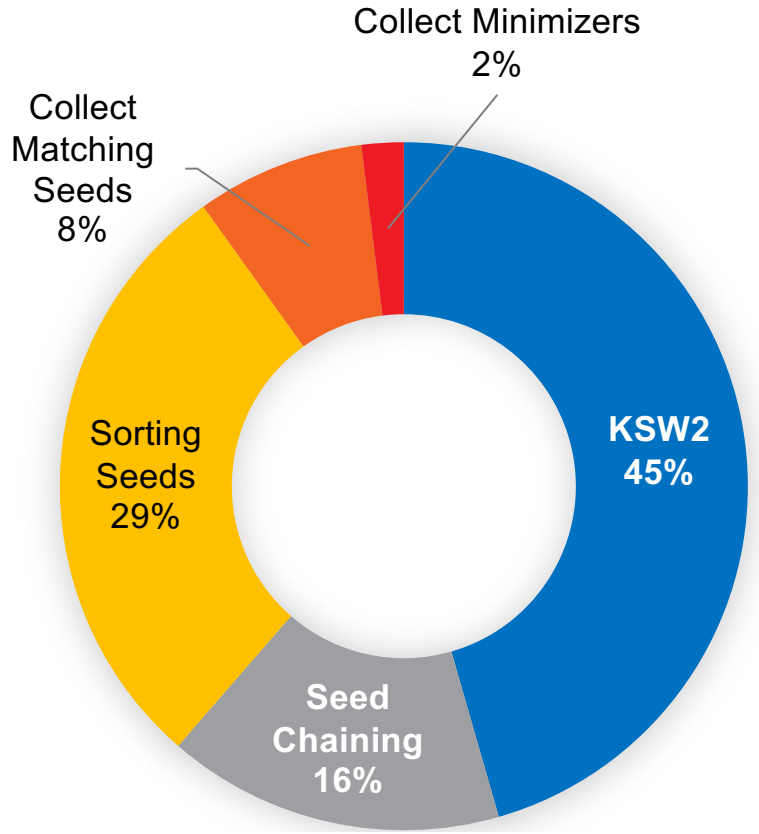
Authors

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

Read Mapping Execution Time

>60%

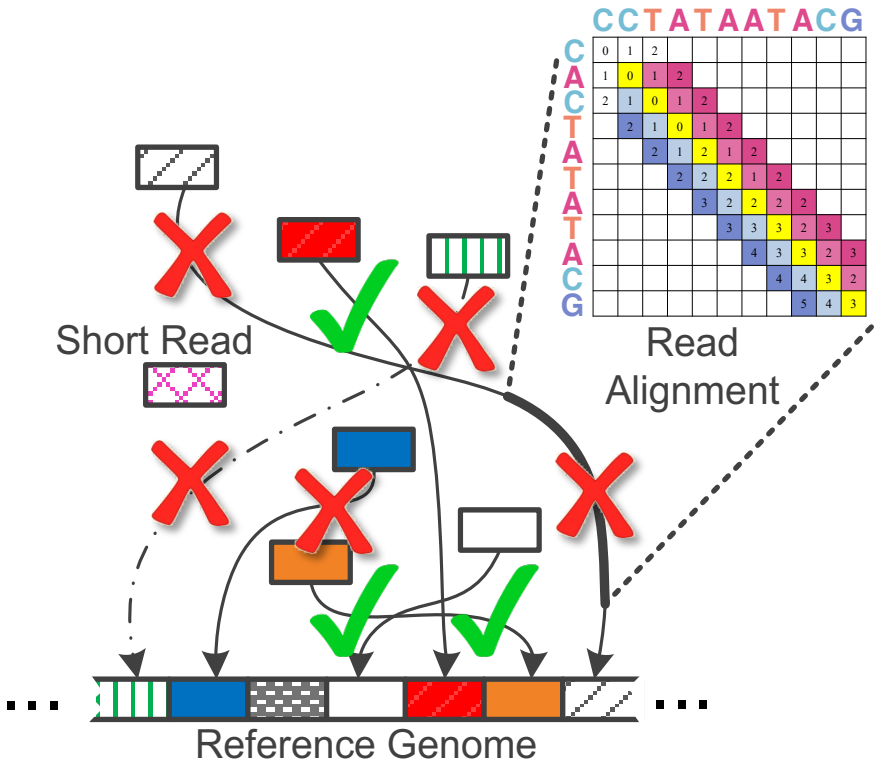
of the read mapper's execution time is spent in sequence alignment



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

Mohammed Alser, "Intelligent Genome Analysis", Computer Architecture, ETH Zurich, Fall 2020

Large Search Space for Mapping Location



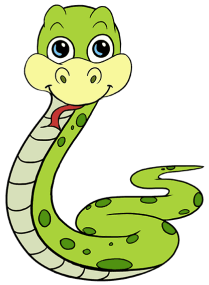
98%

of candidate locations have high dissimilarity with a given read

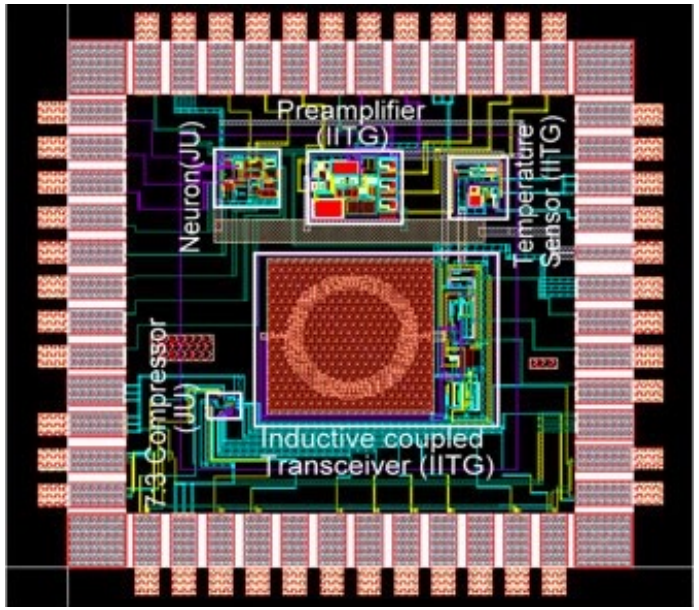
Cheng *et al*, *BMC bioinformatics* (2015)
Xin *et al*, *BMC genomics* (2013)

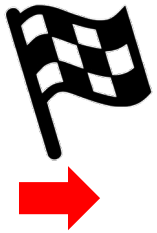
SneakySnake

- **Key idea:**
  - Approximate edit distance calculation is similar to Single Net Routing problem in VLSI chip



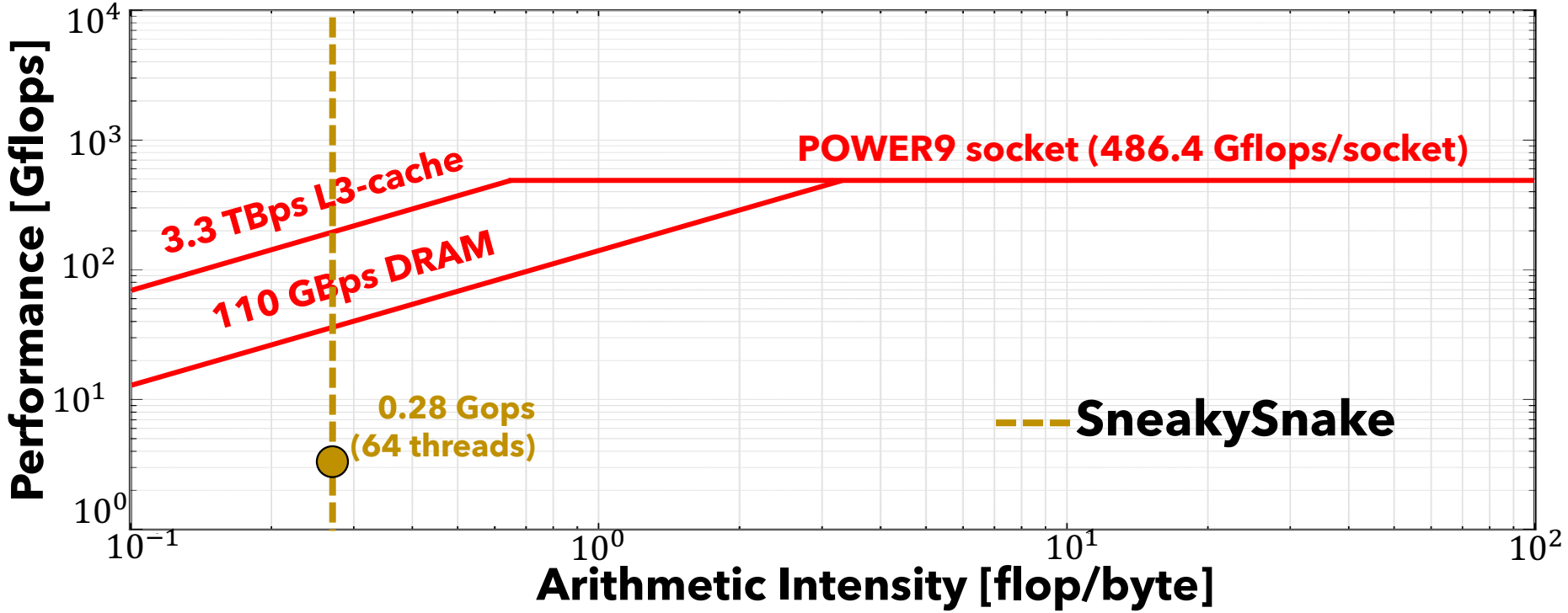
VLSI chip layout





Motivation and Goal

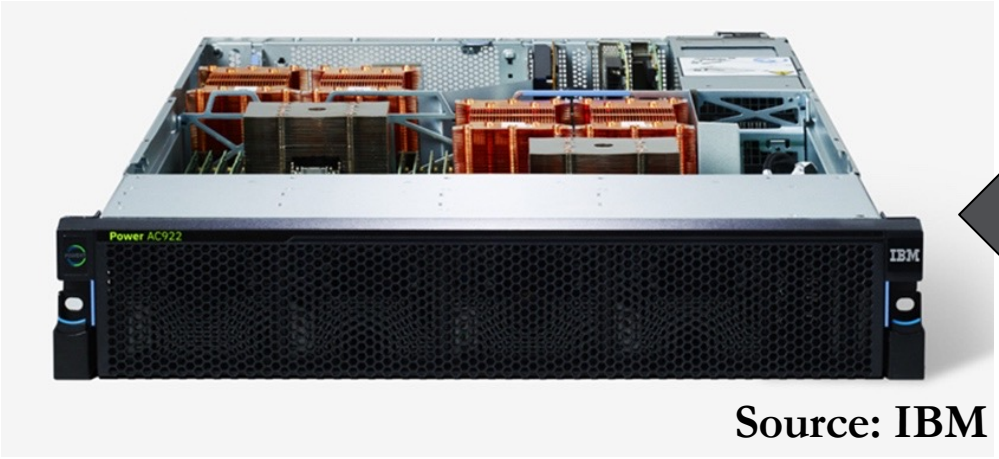
Complex memory access patterns with **limited performance** and **high energy consumption** on **CPU-based system**



Goal:

- **Mitigate** the **performance bottleneck** of modern pre-alignment filtering in an **energy-efficient way**
- Evaluate the use of **near-memory acceleration** using a **FPGA+HBM** connected through an OpenCAPI interface

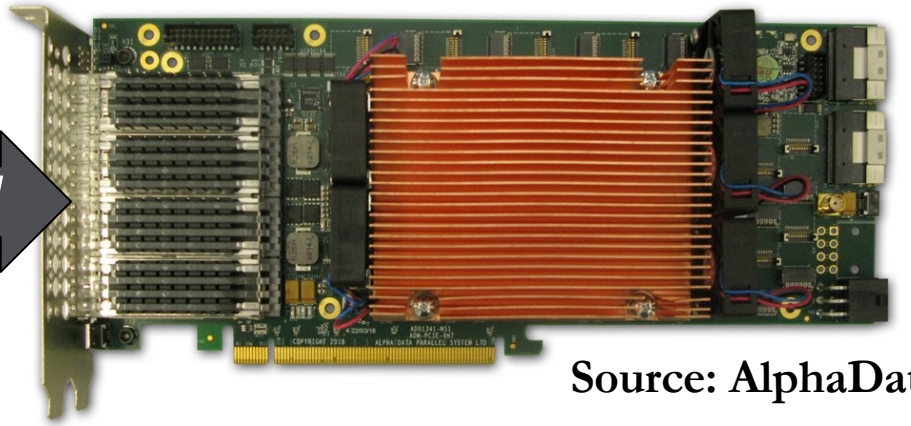
Near-Memory Acceleration



Source: IBM

POWER9 AC922

OCAPI



Source: AlphaData

HBM-based AD9H7 board

We evaluate:

I. Two POWER9+FPGA systems:

1. HBM-based AD9H7 board

Xilinx Virtex Ultrascale+™ XCVU37P-2

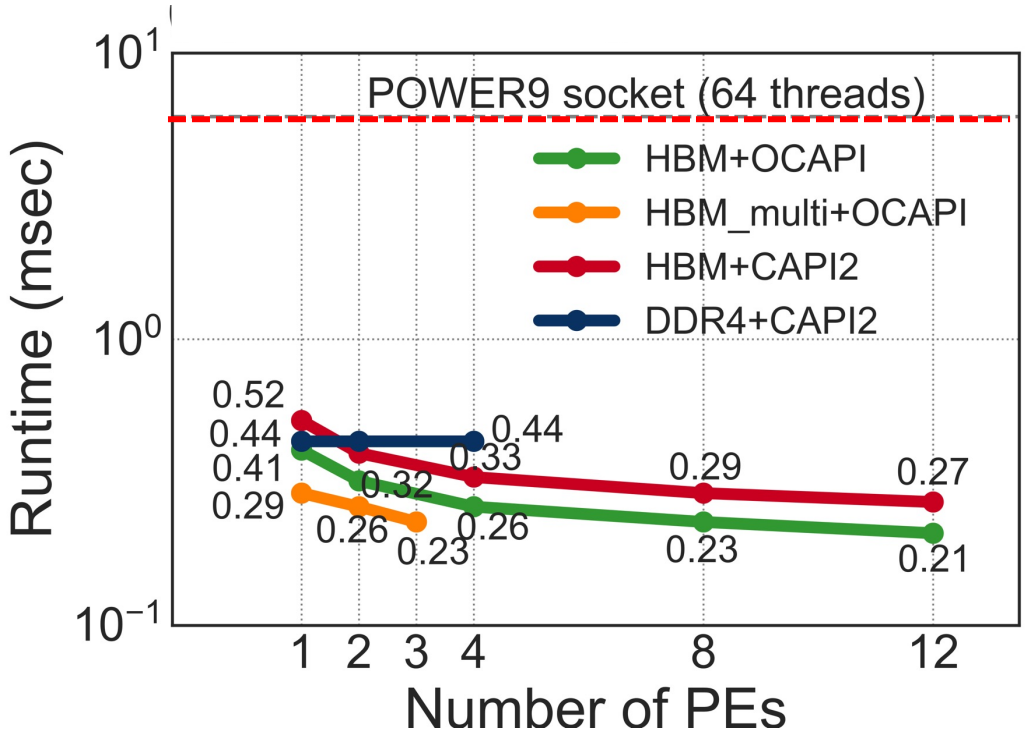
2. DDR4-based AD9V3 board

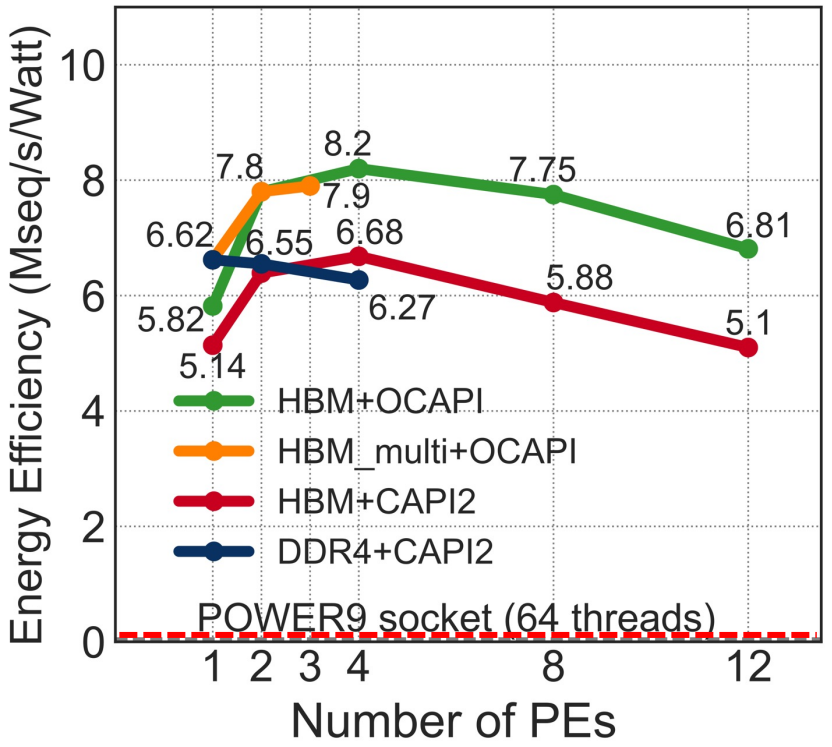
Xilinx Virtex Ultrascale+™ XCVU3P-2

II. Two interconnect technologies: **CAPI2** and **OCAPI**

III. Two processing element (PE) designs: **single channel** and **multiple channel**

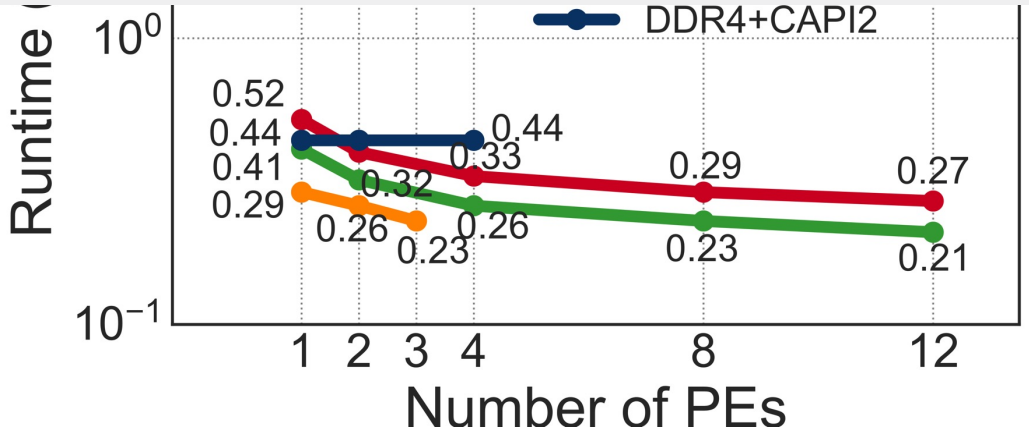
Key Results of Near-Memory Acceleration

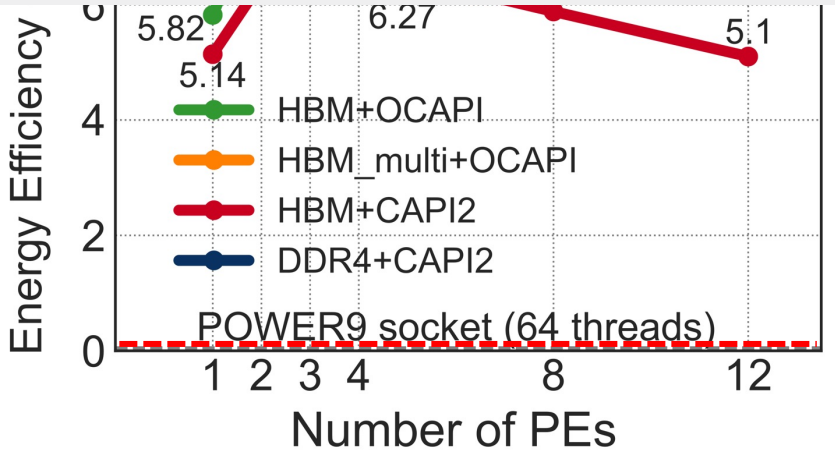




Key Results of Near-Memory Acceleration

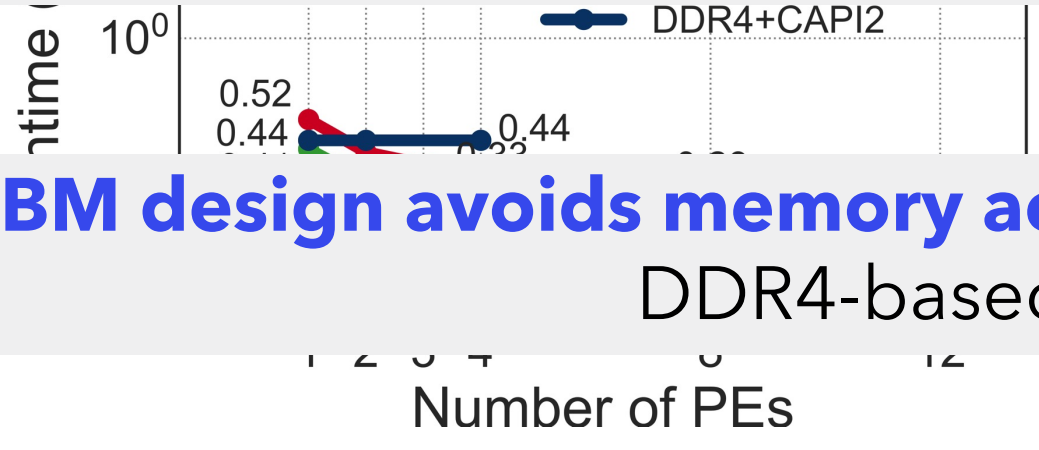
Near-memory acceleration improves **performance** and **energy efficiency** upto 27× and 133×, respectively, over a server-grade CPU-based system

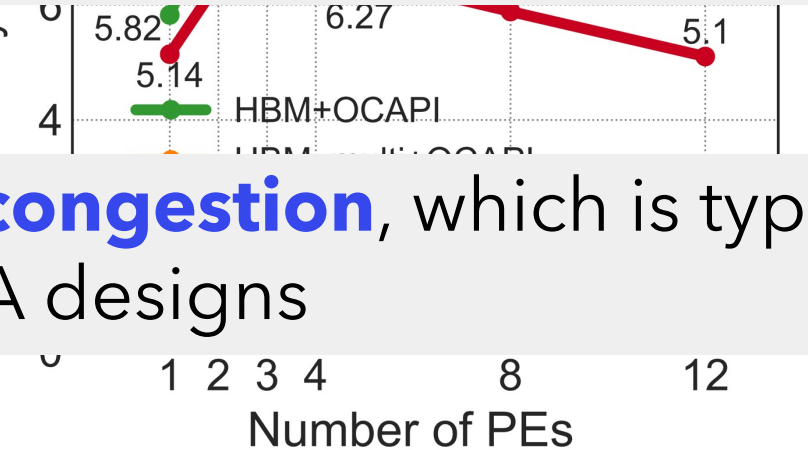




Key Results of Near-Memory Acceleration

Near-memory acceleration improves **performance** and **energy efficiency** upto 27× and 133×, respectively, over a server-grade CPU-based system

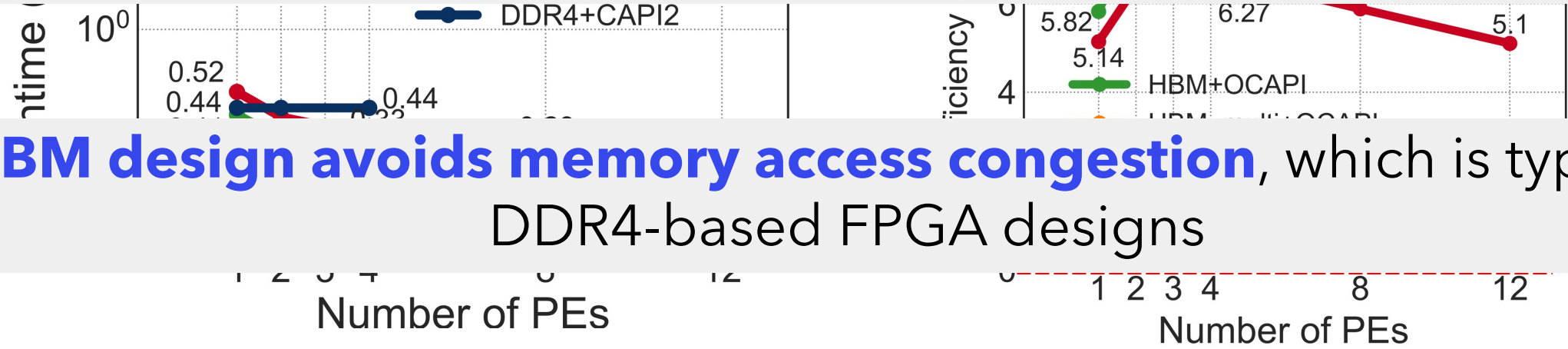




HBM design avoids memory access congestion, which is typical in DDR4-based FPGA designs

Key Results of Near-Memory Acceleration

Near-memory acceleration improves **performance** and **energy efficiency** upto 27× and 133×, respectively, over a server-grade CPU-based system



HBM design avoids memory access congestion, which is typical in DDR4-based FPGA designs

Single channel & multiple channel HBM designs
Open-source: https://github.com/CMU-SAFARI

Near-Memory Acceleration

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]

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

Genome Sequencing Pipeline

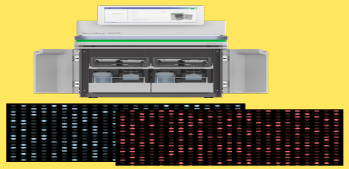
Obtaining Genomic Sequencing Data

A **DNA Extraction**

DNA Fragmentation

B **Sequencing**

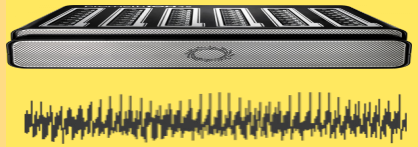
Illumina



Multiple images

Generating Sequencing Data

ONT



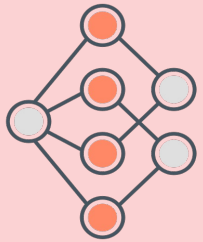
Squiggle

PacBio

30-hour movie

Basecalling

Input signal



CCGTCCCCCCGCAGTAACAT

Read Mapping

A Indexing

Reference Genome

Seeds

Seed content	Seed locations
	1, 4, 6
	3, 5, 12
	50, 52
	2, 100

B Seeding

Read

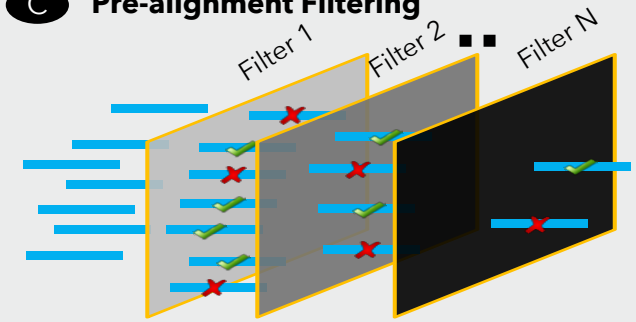
Seeds

Locating common seeds

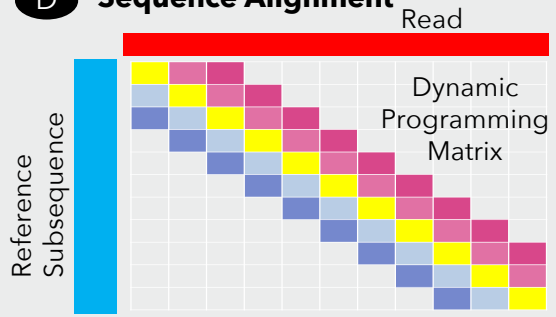
Locations of common seeds

1, 4, 6
3, 5, 12
2, 100

C Pre-alignment Filtering



D Sequence Alignment



Mohammed Alser et al., "From molecules to genomic variations: Accelerating genome analysis via intelligent algorithms and architectures", Computational and Structural Biotechnology Journal, Volume 20, 2022, Pages 4579-4599

Acknowledgements

- AMD:
  - Kristof Denolf
  - Alireza Khodamoradi
- SAFARI Group @ETH Zürich
  - Onur Mutlu
  - Mohammed Alser
  - Juan Gomez-Luna
  - Can Firtina
  - Meryem Banu Cavlak
- Henk Corporaal (TU Eindhoven)


AMD