Intelligent Genome Analysis via Intelligent Algorithms and Architectures

Mohammed Alser, PhD

@mealser

IGGSy 2022
7 July 2022
Plenty of Room at the Indexing

**Misconception:** Indexing step is built only once for each reference genome, why we should care about its performance!

**Fact:** Indexing step affects the performance of all steps of read mapping, as it decides on the number of seeds and their locations.

Reducing the size of the index can speed up read mapping
### Genome Index Properties

- **Seeds** can be Strobemers, Syncmers, BLENDC, LSH, overlapping, non-overlapping, spaced, adjacent, non-adjacent, minimizers, compressed, ...

<table>
<thead>
<tr>
<th>Tool</th>
<th>Version</th>
<th>Index Size</th>
<th>Indexing Parameters</th>
<th>Indexing Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>mrFAST</td>
<td>2.2.5</td>
<td>16.5 GB</td>
<td>default</td>
<td>20.00 min</td>
</tr>
<tr>
<td>minimap2</td>
<td>2.24</td>
<td>7.3 GB</td>
<td>ax map - ont</td>
<td>3.33 min</td>
</tr>
<tr>
<td>BWA-MEM</td>
<td>0.7.17</td>
<td>4.7 GB</td>
<td>default</td>
<td>49.96 min</td>
</tr>
<tr>
<td>BWA-MEM2**</td>
<td>2.2.1</td>
<td>17 GB</td>
<td>default</td>
<td>33.36 min</td>
</tr>
</tbody>
</table>

*Human genome = 3.2 GB (char-encoded) or 1.6 GB (4bit-encoded)

**Its peak memory = 72.3 GB, minimap2 =11.4 GB when building the index.

None of the existing methods try to reduce the size of the reference genome.
Compressed Genomic Analyses

Can we process compressed genomic sequences without decompression?

Image adapted from: https://joinzoe.com/post/genetic-diet-nutrition
Genome-on-Diet Steps

- Compressed Indexing
- Pattern Alignment
- Compressed Seeding
- Location Voting
- Sequence Alignment
Step 1: Compressed Indexing

Reference genome

Diet pattern
1010101010101010101010

Patterned genome

Read

Diet pattern
1010101010

Patterned read

Easy! Isn’t it?
Where to Start Applying the Pattern?

ACCCTAACCCTAACCCTAACCCTAACCCTAA

A_C_T_A_C_T_A_C_T_A_C_T_A_C_T_A
A_C_T_A_C_T_A_C_T_A_C_T_A
C_T_A_C_T_A_C_T_A_C_T_A_C_T_A
T_A_C_T_A_C_T_A_C_T_A_C_T_A
A_C_T_A_C_T_A_C_T_A_C_T_A_C

No Match 😞

CCCTAACCCTAACCCTAACCCTAACCCTAACCCTAA

C_C_A_C_C_A_C_A_C_A_C_A_C_A_C_A_C_A
C_C_A_C_C_A_C_A_C_A_C_A_C_A_C_A_C_A
C_C_A_C_C_A_C_A_C_A_C_A_C_A_C_A_C_A
C_A_C_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A
C_A_C_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A
A_C_C_A_C_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A_C_A

SAFARI
Step 2: Pattern Alignment

Read
Diet pattern 0
1010101010
Patterned read 0

Read
Diet pattern 1
1010101010
Patterned read 1

Counting occurrences in the index
1 occurrence

Alignment index = 1
92 occurrences
Step 3: Compressed Seeding

- Read
- Alignment index
- Diet pattern
- 1010101010
- Patterned read
- Seeds
Step 4: Location Voting

Genome sequence:

- Maximum allowed distance, $D$

- Winning mapping location: $1020 - 3 = 1017$

Read sequence:

- Large insertion
- Large deletion
Step 5: Sequence Alignment

Dynamic programming matrix
Introducing Five Optimization Strategies

- Accelerating Indexing & Seeding with **SIMD Instructions**
- **Sorting** Seed Locations
- **Progressive** Compressed Seeding
- **Rescuing** Mapping Location
- Handling **Exactly-Matching** Short Reads
Genome-on-Diet is 1.6-2.23x, 1.74-2.32x, and 1.56-2.2x faster than minimap2 using Illumina, HiFi, and ONT reads, respectively.
Number of Mapped Reads

Considering the number of mapped reads, Genome-on-Diet is 3.7-6x faster than minimap2
Other Important Results

- **Genome-on-Diet** provides 3.36x, 10.2x, and 17.53x higher number of mapped reads with the **highest mapping quality** (MAPQ=60) than **minimap2**.

- Both **Genome-on-Diet** and **minimap2** agree on the mapping locations in 91%, 84%, and 73% of high-quality reads (MAPQ=60) mapped by both tools when using Illumina, HiFi, and ONT reads.

[GitHub Link](https://github.com/CMU-SAFARI/Genome-on-Diet)
GenStore: A High-Performance In-Storage Processing System for Genome Sequence Analysis

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

\textsuperscript{1}ETH Zürich  \textsuperscript{2}Bionano Genomics  \textsuperscript{3}KMUTNB  \textsuperscript{4}University of Toronto
Data Movement Dominates Performance

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

* Boroumand et al., “Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks,” ASPLOS 2018
* Kestor et al., “Quantifying the Energy Cost of Data Movement in Scientific Applications,” IISWC 2013
* Pandiyen and Wu, “Quantifying the energy cost of data movement for emerging smart phone workloads on mobile platforms,” IISWC 2014
Key Ideas of GenStore (ASPLOS 2022)

**GenStore-EM (exactly-matching reads filter):** In some cases, a large fraction of reads exactly match to subsequences of the reference genome.

**GenStore-NM (non-matching reads filter):** In some cases, a large fraction of reads do not match to subsequences of the reference genome.

**GenStore-EM:** 2.1-6.1× speedup & 3.92x energy saving compared to minimap2.

**GenStore-NM:** 1.4-33.6x speedup & 27.17x energy saving compared to minimap2.
SeGraM: A Universal Hardware Accelerator for Genomic Sequence-to-Graph and Sequence-to-Sequence Mapping

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

\(^1\)Bionano Genomics  \(^2\)ETH Zürich  \(^3\)Bilkent University  \(^4\)Intel Labs  
\(^5\)Carnegie Mellon University  \(^6\)University of Illinois Urbana-Champaign
SeGraM: Universal Genomic Mapping Accelerator

- **First universal genomic mapping accelerator** that can support both sequence-to-graph mapping and sequence-to-sequence mapping, for both short and long reads

- **First algorithm/hardware co-design** for accelerating sequence-to-graph mapping

  - We base SeGraM upon a minimizer-based seeding algorithm, and
  - We propose a novel bitvector-based alignment algorithm to perform approximate string matching between a read and a graph-based reference genome

  - We co-design both algorithms with high-performance, scalable, and efficient hardware accelerators
Sequence-to-Graph Mapping Pipeline

**Pre-Processing Steps (Offline)**

1. **Genome Graph Construction**
   - Linear reference genome
   - Known genetic variations
   - Genome graph

2. **Indexing**
   - Hash-table-based index (of graph nodes)

**Seed-and-Extend Steps (Online)**

1. **Seeding**
   - Candidate mapping locations (subgraphs)

2. **Filtering/Chaining/Clustering**
   - Good candidate mapping locations (subgraphs)

3. **Alignment**
   - Optimal alignment between read & subgraph

SAFARI
SeGraM provides 5.9× and 3.9× throughput improvement over GraphAligner and vg, while reducing the power consumption by 4.1× and 4.4×
Our Contributions

**Near-memory/In-memory Pre-alignment Filtering**
- GRIM-Filter [BMC Genomics’18]
- SneakySnake [IEEE Micro’21]
- GenASM [MICRO 2020]

**In-storage Sequence Alignment**
- GenStore [ASPLOS 2022]

**Near-memory Sequence Alignment**
- GenASM [MICRO 2020]
- SeGraM [ISCA 2022]

**Specialized Pre-alignment Filtering Accelerators (GPU, FPGA)**
- GateKeeper [Bioinformatics’17]
- MAGNET [AACBB’18]
- Shouji [Bioinformatics’19]
- GateKeeper-GPU [arXiv’21]
- SneakySnake [Bioinformatics’20]

Sequencing Machine → Storage (SSD/HDD) → Main Memory → Microprocessor
Our Contributions

Near-memory/In-memory Pre-alignment Filtering

- GenASM [MICRO 2020]
- SeGraM [ISCA 2022]

Specialized Pre-alignment Filtering Accelerators (GPU, FPGA)

Improving **performance** and **energy efficiency** by 1-3 orders of magnitude

- GenASM
- SeGraM
- GateKeeper
- MAGNET
- Shouji
- SneakySnake
- GRIM-Filter
- Near-memory/In-memory Pre-alignment Filtering
- GenASM
- GenStore
- In-storage Sequence Alignment
- SeGraM
- Improving performance and energy efficiency by 1-3 orders of magnitude

Sequencing Machine
Storage (SSD/HDD)
Main Memory
Microprocessor
Intelligent Genome Analysis

https://arxiv.org/abs/2205.07957

Going From Molecules to Genomic Variations to Scientific Discovery: Intelligent Algorithms and Architectures for Intelligent Genome Analysis

Mohammed Alser, Joel Lindegger, Can Firtina, Nour Almadhoun, Haiyu Mao, Gagandeep Singh, Juan Gomez–Luna, Onur Mutlu
Accelerating 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”
Read Mapping 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]
Key Takeaway

Most speedup comes from parallelism enabled by novel architectures and algorithms
SAFARI Research Group

Think BIG, Aim HIGH!

https://safari.ethz.ch
Contributors

Mohammed Alser  
Julien Eudine  
Onur Mutlu

Can Alkan  
Damla Senol Cali  
Nika Mansourighiasi

And many more ...
SAFARI Research Group

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

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

Think BIG, Aim HIGH!

https://safari.ethz.ch
Intelligent Genome Analysis via Intelligent Algorithms and Architectures

Mohammed Alser, PhD

@mealser

IGGSy 2022
7 July 2022