
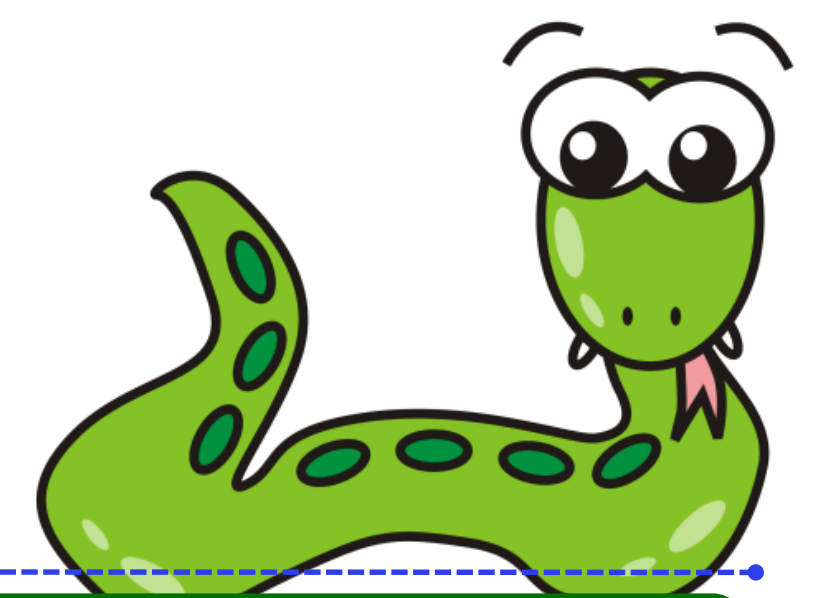


SneakySnake: A New Fast and Highly Accurate Pre-Alignment Filter on CPU and FPGA for Accelerating Sequence Alignment

Mohammed Alser^{1,2} Can Alkan² Onur Mutlu^{1,2,3}
¹ **ETH** zürich ²  Bilkent University ³ **Carnegie Mellon**

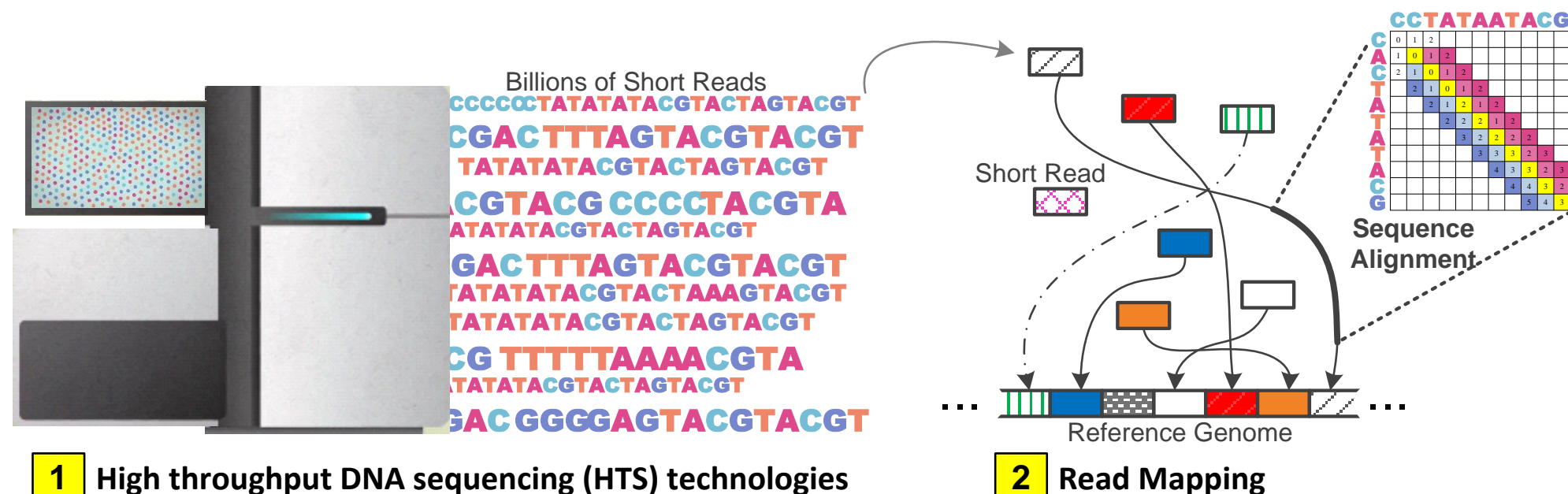


1: Read Mapping

Fact: it remains challenging to sequence the entire DNA molecule as a whole.

As a workaround: high throughput DNA sequencing (HTS) technologies can sequence only segments of the original molecule. This is relatively **quick and cost-effective** but it results in an **excessive number of genomic reads**.

Hence we need **read mapping** to link the reads together and construct back the donor's complete genome by 1) determining the **location** of each read within reference genome and 2) calculating its **optimal sequence alignment**.



2: Problem

Calculating sequence alignment is a major performance bottleneck:

- ✗ Uses **computationally expensive dynamic programming** algorithms.
- ✗ **Bottlenecked by memory bandwidth**, e.g., Illumina NovaSeq 6000 generates 6 Terabases in < 24 hours
- ✗ They are **unavoidable** as they provide accurate information about the quality of the alignment.
- ✗ Majority of **candidate locations** in the reference genome **do not align** with a given read due to **high dissimilarity**.

3: Our Goal

Significantly **reduce** the time spent on calculating the sequence alignment of short sequences using **pre-alignment filtering**.

To this end: We introduce new, fast, and very accurate pre-alignment filters, **SneakySnake** (for CPU) and **Snake-on-Chip** (for FPGA).

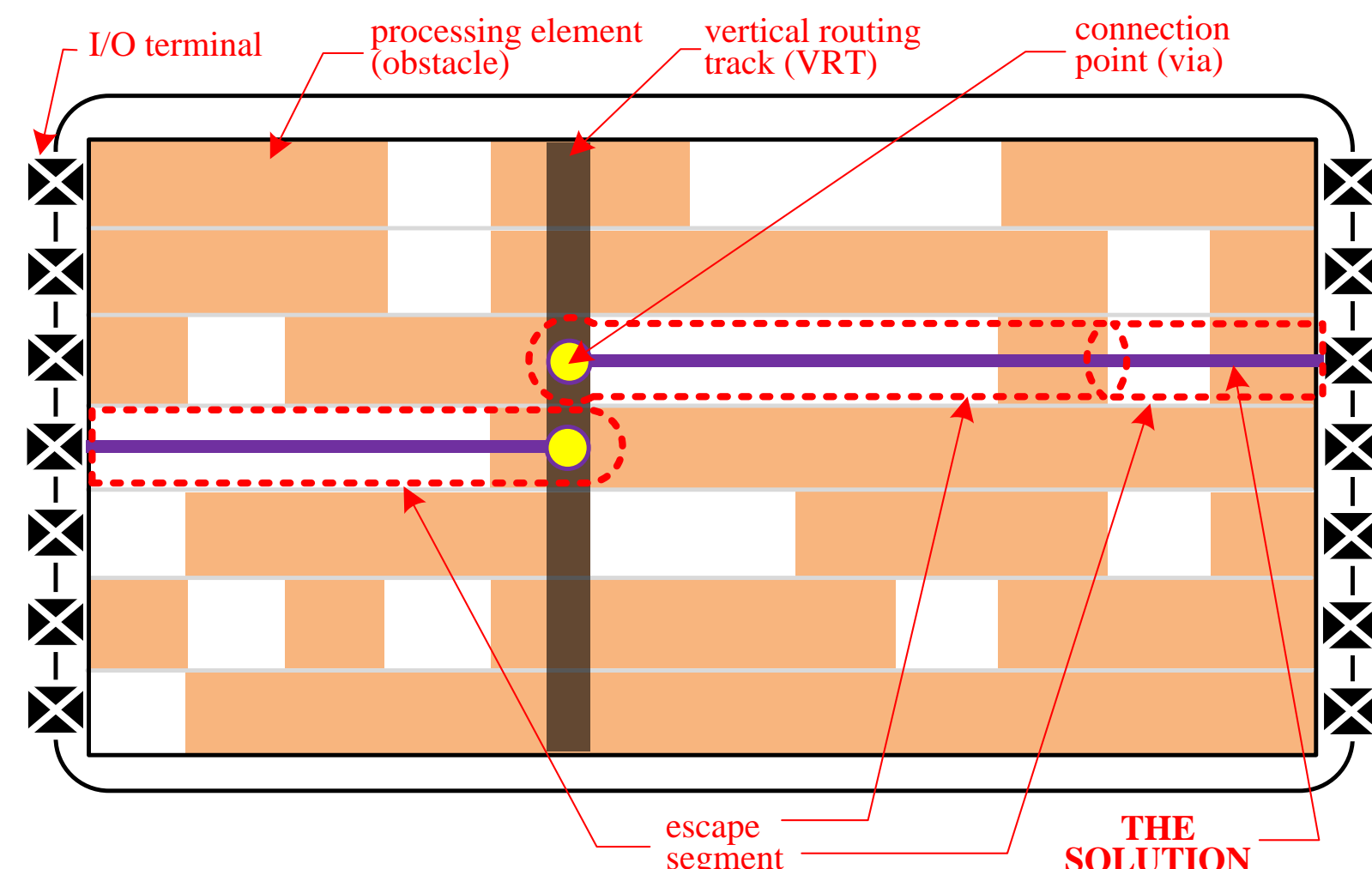
4: Key Ideas

- ✓ Quickly and accurately **filters out highly dissimilar sequence pairs** before applying sequence alignment algorithms.
- ✓ Provides fast and highly accurate filtering by **reducing** the sequence alignment problem to **single net routing (SNR) problem** [Lee+, IEEE-TCAS 1976] in VLSI chip layout.
- ✓ Judiciously leverages the **parallelism-friendly** architecture of modern **FPGAs** to greatly speed up the SneakySnake algorithm.

5: Single Net Routing (SNR) Problem

SNR Problem: finding the **optimal** routing path that:

- Includes the **least** number of **horizontal escape segments**,
 - Passes through the **minimum** number of **obstacles**,
 - Connects two IO terminals** on a special grid layout.
- The **number of obstacles** in the solution to the SNR problem is a **lower bound** on the actual **number of edits** between two genomic sequences.
- Solving the **SNR problem** is **much faster** than solving the **sequence alignment** problem, as calculating the routing path after facing an obstacle is **independent** of the calculated path before this obstacle.



6: SneakySnake Walkthrough

	<i>j</i>	1	2	3	4	5	6	7	8	9	10	11	12
<i>i</i>		G	G	T	G	C	A	G	A	G	C	T	C
1		G	0	0	1	0							
2		G	0	0	1	0	1						
3		T	1	1	0	1	1	1					
4		G	0	0	1	0	1	1	0				
5		A		1	1	1	1	0	1	0			
6		G			1	0	1	1	0	1	0		
7		A				1	1	0	1	0	1	1	
8		G					1	1	0	1	0	1	1
9		T						1	1	1	1	0	1
10		T							1	1	1	0	1
11		G								1	0	1	1
12		T									1	1	0

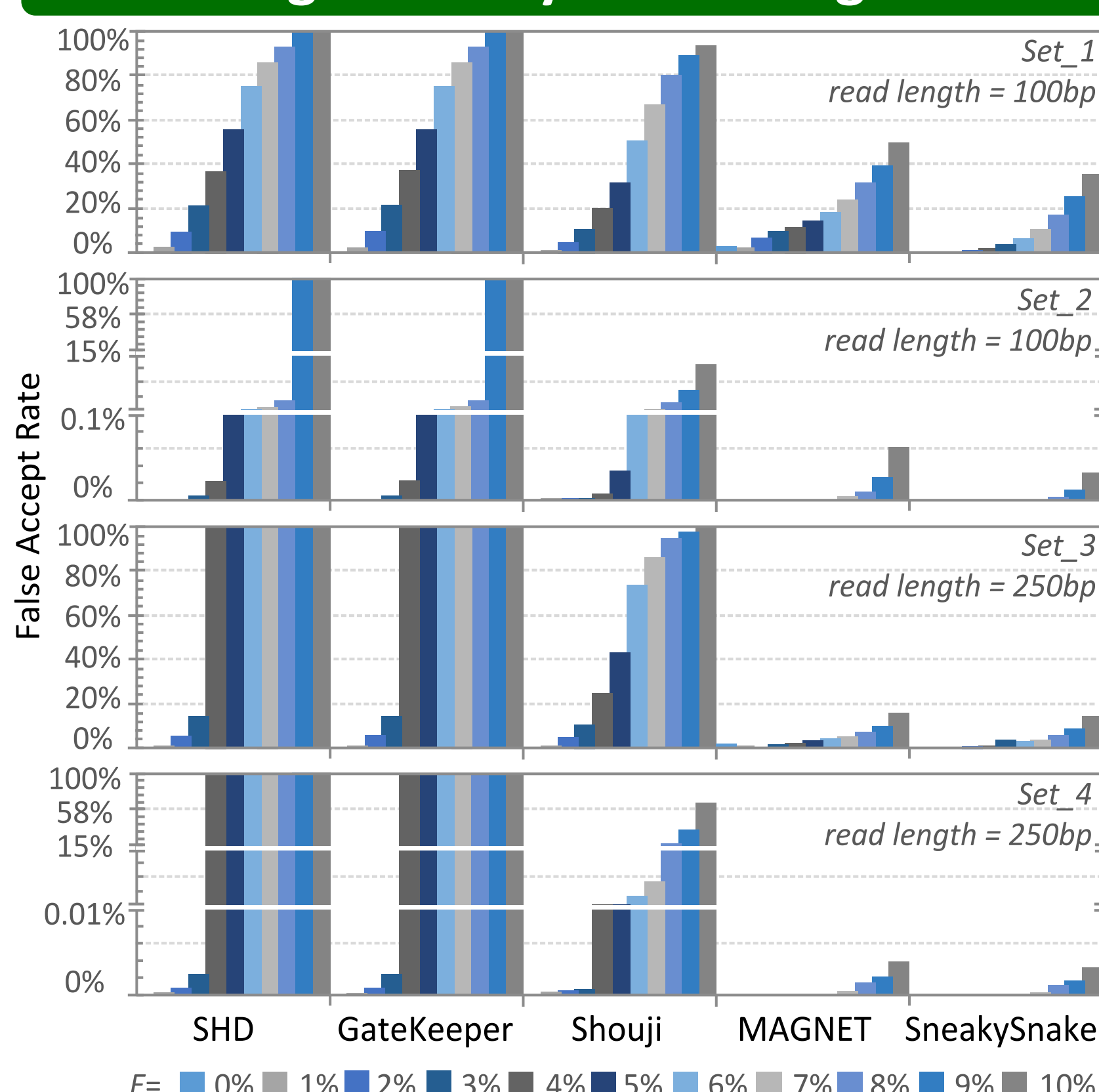
column	1	2	3	4	5	6	7	8	9	10	11	12
3 rd Upper Diagonal	1	1	1	0	1	1	0	0	0	1	1	1
2 nd Upper Diagonal	1	1	1	0	1	1	1	1	1	1	0	1
1 st Upper Diagonal	1	0	1	1	1	0	0	0	0	1	0	1
Main Diagonal	0	0	0	0	1	1	1	1	1	1	1	1
1 st Lower Diagonal	0	1	1	1	1	0	0	1	1	1	0	1
2 nd Lower Diagonal	1	0	1	0	1	1	1	1	0	1	1	1
3 rd Lower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

7: Evaluation & Key Takeaways

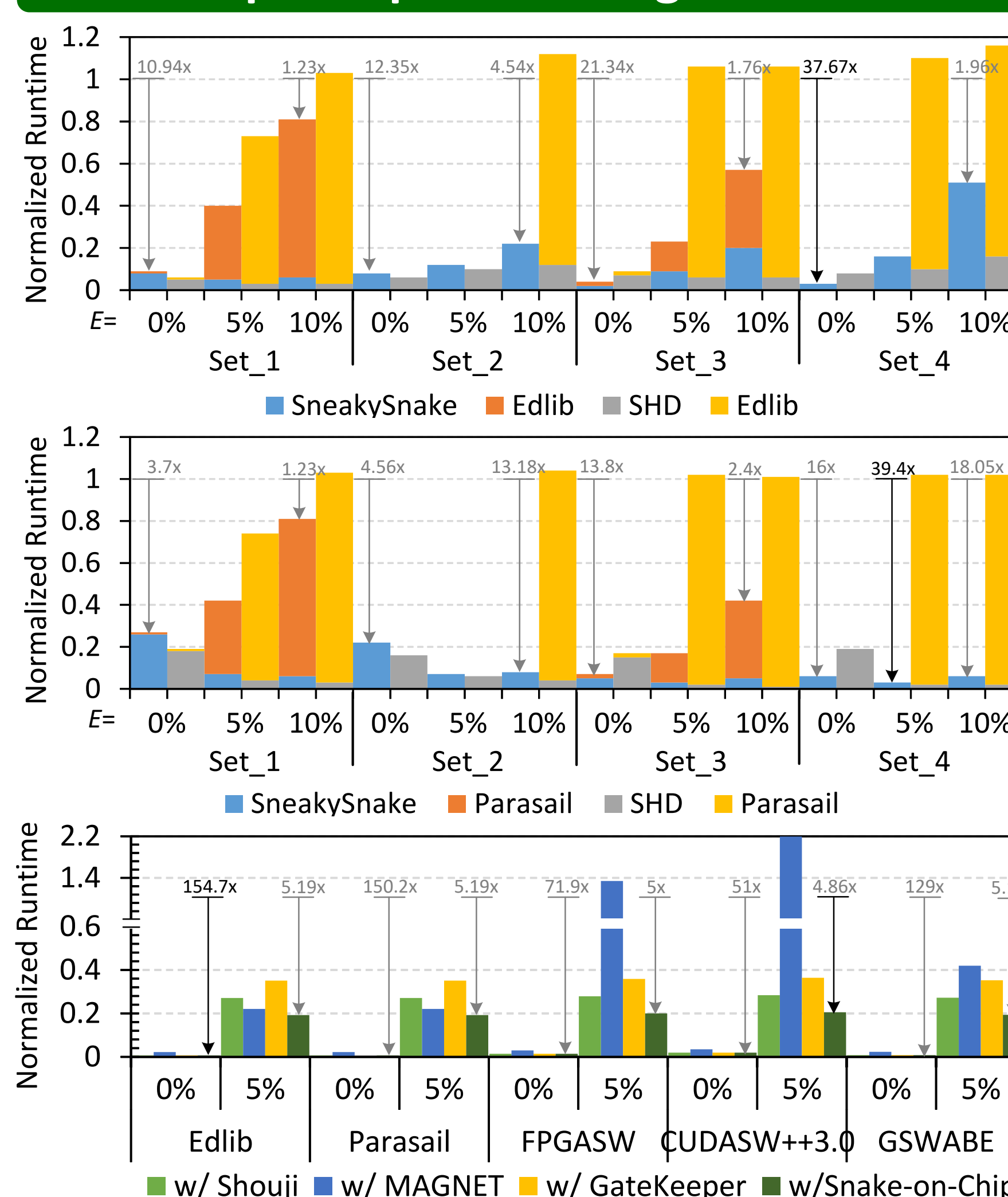
Dataset Description: **Set_1** & **Set_2**: each has 30 million pairs from mapping **ERR240727_1** to the human genome using mrFAST's e= 2, 40, respectively.

Set_3 & **Set_4**: each has 30 million pairs from mapping **SRR826471_1** using mrFAST's e= 8, 100, respectively.

Filtering Accuracy vs. Existing Filters



Speedup vs. Existing Filters



Key Results

- ✓ **< 31412x, 20603x, and 64.1x fewer falsely-accepted sequences** compared to GateKeeper / SHD (using Set_4, E= 10%), Shouji (using Set_4, E= 10%), and MAGNET (using Set_1, E= 1%), respectively.
- ✓ **< 37.6x and 43.9x speedup** with the addition of **SneakySnake** to Edlib [Šošić+, Bioinformatics 2017] (using Set_4, E= 0%) and Parasail [Daily+, Bioinformatics BMC 2016] (using Set_4, E=2%), respectively.
- ✓ **< 154.7x and 150.2x speedup** with the addition of **Snake-on-Chip** to Edlib [Šošić+, Bioinformatics 2017] (E= 0%) and Parasail [Daily+, Bioinformatics BMC 2016] (E=0%), respectively. < 1.4x, 3.4x, and 1.8x more speedup compared to that provided by adding Shouji, MAGNET, and GateKeeper as a pre-alignment filter, respectively.
- ✓ **SneakySnake & Snake-on-Chip**
 - Open-source: <https://github.com/CMU-SAFARI>
 - do **not** replace sequence alignment step.
 - do **not** sacrifice any of the sequence aligner **capabilities** (scoring and backtracking), as they do **not** modify the aligner.