Massively Parallel Mapping of Next Generation Sequence Reads Using GPUs

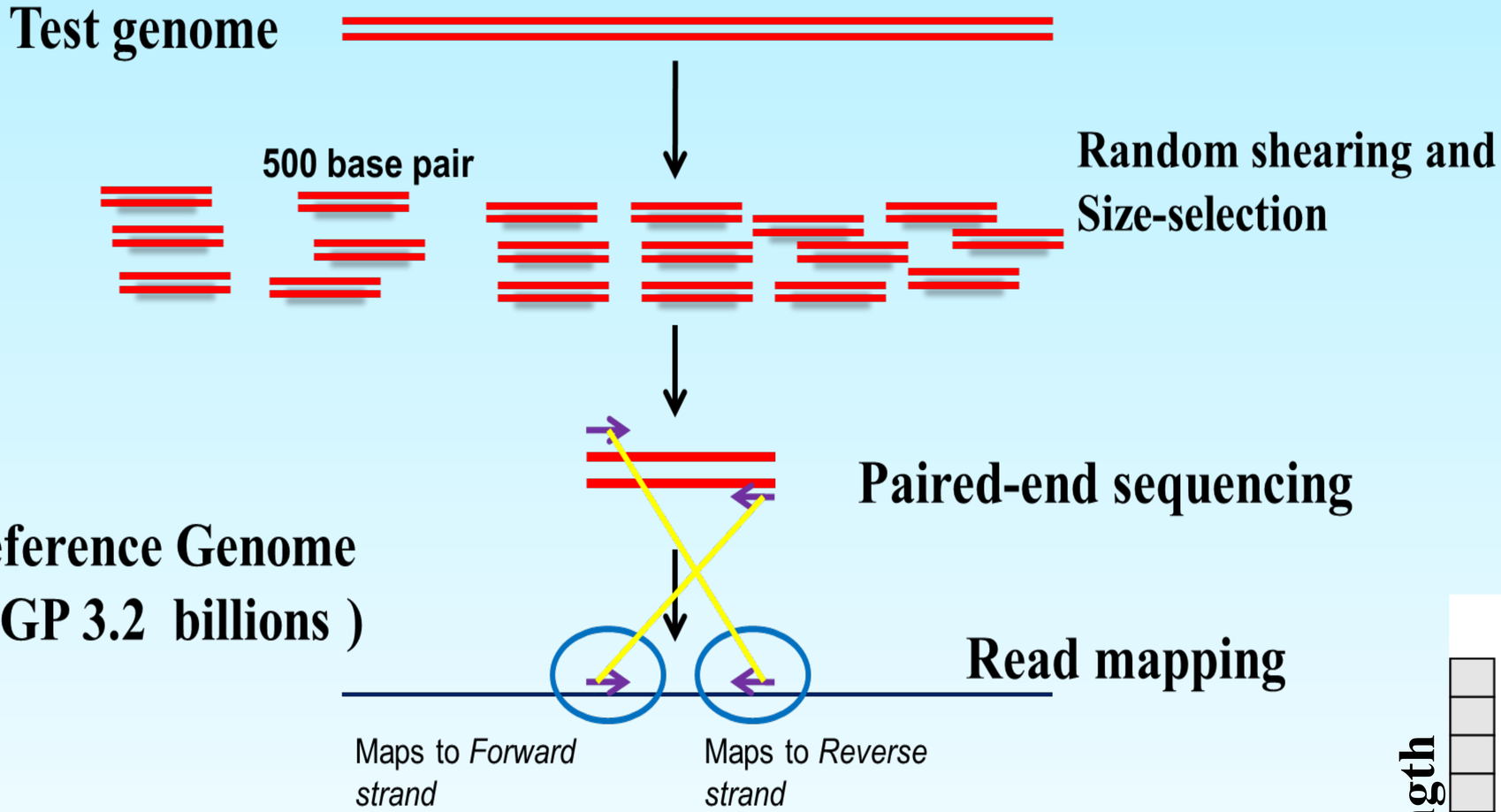
Azita Nouri, Reha Oğuz Selvitopi, Özcan Öztürk, Onur Mutlu, Can Alkan

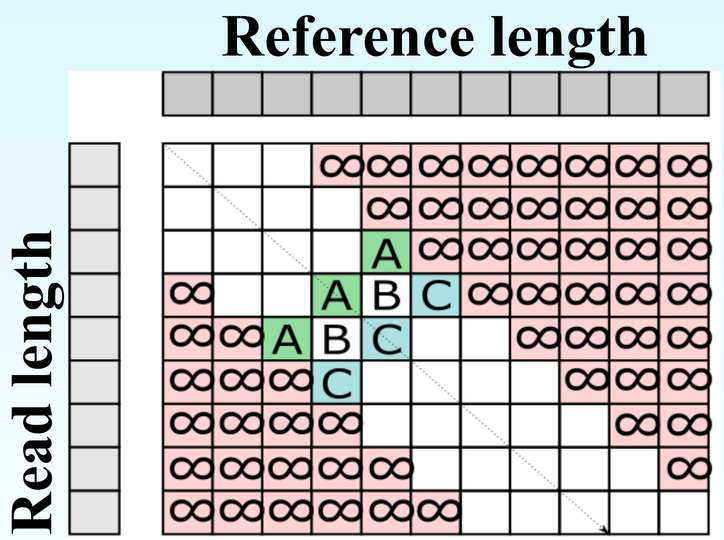
Bilkent University, Computer Engineering Department, Turkey
Carnegie Mellon University, Electrical and Computer Engineering Department, USA

Motivation

- DNA sequence alignment problem is a character-level comparison of DNA sequences obtained from one or more samples against a database of reference genome sequence of the same or a similar species.
- Huge computational burden due to the comparison of >1 billion short (100 characters, or base pairs) “reads” against a very long (3 billion base pairs) reference genome.
- Requires 30-50 CPU days for mapping & alignment
- >1 million whole human genomes by the end of 2017!
- Clinical sequencing in trials in the US
  - Genome sequencing as a routine test at hospitals
  - We need very fast, accurate, and low-cost analysis methods

Whole Genome Sequencing





Edit distance

Aims

- Develop and implement a GPGPU-friendly algorithms to map DNA sequence reads to the reference genome

- Take advantage of the embarrassingly parallel nature of the problem to concurrently align millions of read vs reference pairs

- Implementation using the CUDA (Compute Unified Device Architecture) platform, and testing using the NVIDIA Tesla K20 GPGPU processors.

Alignment GPU Algorithm

Reference length

Read length

		-	A	C	A	C	A	C	T	A	G	T	A
	-	0	0	0	0	0	0						
	A	0	1	0	-1	-1	1	0					
	G	0	0	0	-1	-2	-2	0	-1				
	C	0	-1	1	0	-1	-2	-1	-1	-2			
	A	0	1	0	2	1	0	-1	-2	0	-1		
	C		0	2	1	3	2	1	0	-1	-1	-2	
	A			1	3	2	4	3	2	1	0	-1	-1
	C				2	4	3	5	4	2	1	0	
	A					3	5	4	4	5	3	2	
	T						4	4	5	4	4	4	
	A							3	4	6	5	4	
	G								3	5	7	6	5

Thread 1

Thread 2

Thread 3

Thread 4

Thread 5

Alignment GPU Algorithm

Reference length

Read length

					-	A	C	A	C	A	C	T	A	G	T	A
				-	0	0	0	0	0	0						
				A	0	1	0	-1	-1	1	0					
				G	0	0	0	-1	-2	-2	0	-1				
				C	0	-1	1	0	-1	-2	-1	-1	-2			
				A	0	1	0	2	1	0	-1	-2	0	-1		
				C		0	2	1	3	2	1	0	-1	-1	-2	
				A			1	3	2	4	3	2	1	0	-1	-1
				C				2	4	3	5	4	3	2	1	0
				A					3	5	4	4	5	4	3	2
				T						4	4	5	4	4	5	4
				A							3	4	6	5	4	6
				G								3	5	7	6	5

Thread 5

Thread 4

Thread 1

Thread 2

Thread 3

Contributions

- Map a time-consuming application to massively parallel GPU architectures.
- Move the compute-intense parallel verification step to the GPGPUs.
- Collect the reads in a buffer, then pass to the GPGPU for millions of simultaneous alignments.
- Determine the number of alignments automatically by considering the characteristics of the GPGPU.
- Adjust the number of threads used per alignment dynamically based on the maximum allowed error threshold set by the user.
- Ability to be merged with any existing and future hash-table based read mapping applications.
- Ability to be used for various configurations like different read sizes, reference genome size and error allowance.
- Reduce host to GPU transfer time significantly by placing all relevant data to the GPU global memory in the initialization step.
- Develop dynamic programming backtracking in GPU, bypassing CPU-based post processing all together, except for I/O operations.

Backtracking GPU Algorithm

Reference length

Read length

	-	A	C	A	C	A	C	T	A	G	T	A
-	0	0	0	0	0	0						
A	0	1	0	-1	-1	1	0					
G	0	0		-1	-2	-2	0	-1				
C	0	-1	1		-1	-2	-1	-1	-2			
A	0	1	0	2		0	-1	-2	0	-1		
C		0	2	1	3		1	0	-1	-1	-2	
A			1	3	2	4		2	1	0	-1	-1
C				2	4	3	5		3	2	1	0
A					3	5	4	4		4	3	2
T						4	4	5	4		5	4
A							3	4	6	5		6
G								3	5	7	6	

A - C A C A C T A G T A -

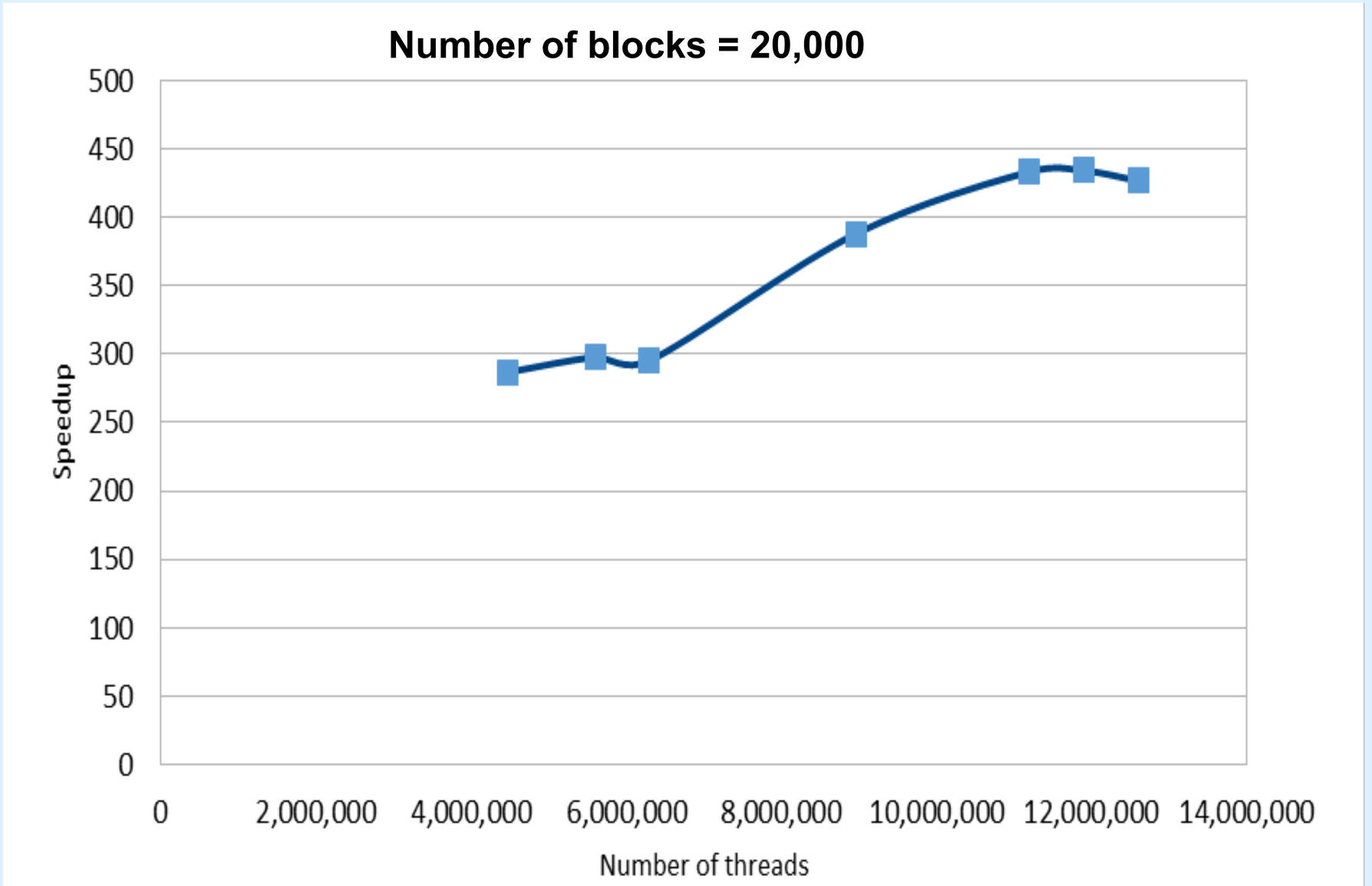
A G C A C A C - A - T A G

-1	-1
1	0

Read length + Reference length

0	0	1	0	0	-1	-1	-2	0	-1	-1	-2	-1	-1	0							
0	0	-1	-1	-2	-2	-2	-1	-1	-2	0	-1	1	0	2	1	3	2	4			
0	0	1	0	0	1	0	-1	1	0	2	1	3	2	4	3	5	4	4	5	4	6	5
0	0	0	-1	1	0	2	1	3	2	4	3	5	4	4	5	4	6	5	7	6		
0	0	1	0	2	1	3	2	4	3	5	4	4	3	4	3	5						

Speedup (tentative results)



Thank You