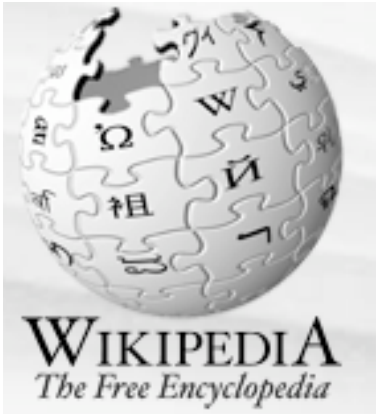
High-throughput Sequence Alignment using Graphics Processing Units

Michael Schatz &
Cole Trapnell

May 21, 2009
UMD NVIDIA CUDA Center Of Excellence Presentation

Searching Wikipedia

- How do you find all pages with your name in the Wikipedia
  - 4M pages x 250 words / page = 1B words to search

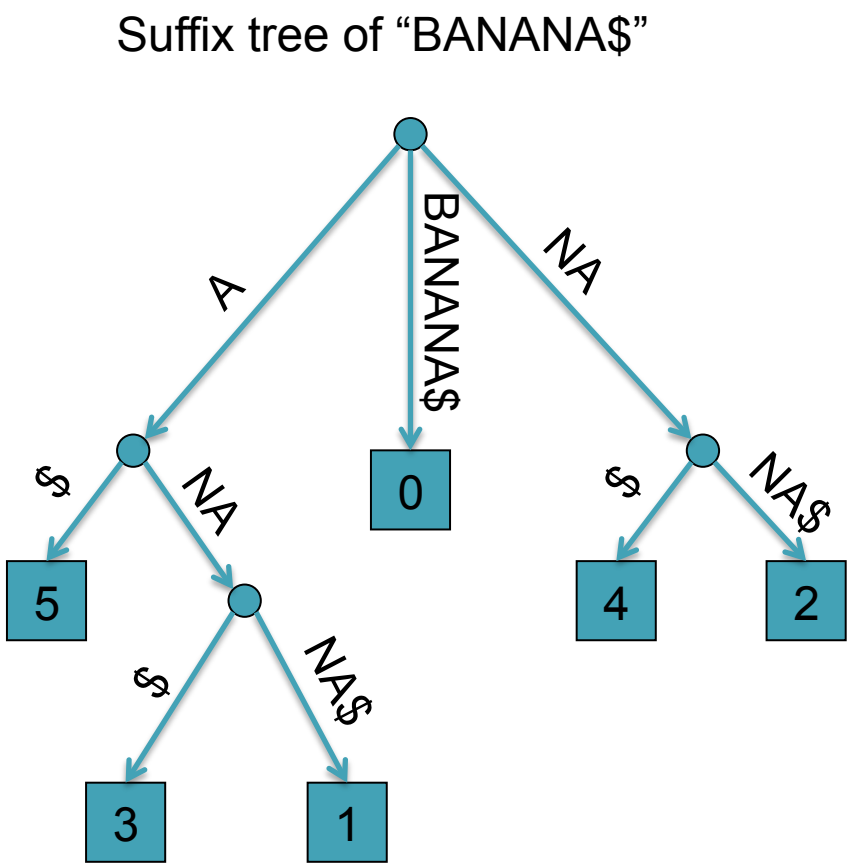


- Sequentially searching every word is too slow, we need an index
  - Is the query Q present, and if so, where?
  - Are there any partial or approximate occurrences of Q?

Michael Schatz

Michel Schatz
Michal Schatz
…
Michael Shatz
Michael Schats
Michael Schatnz

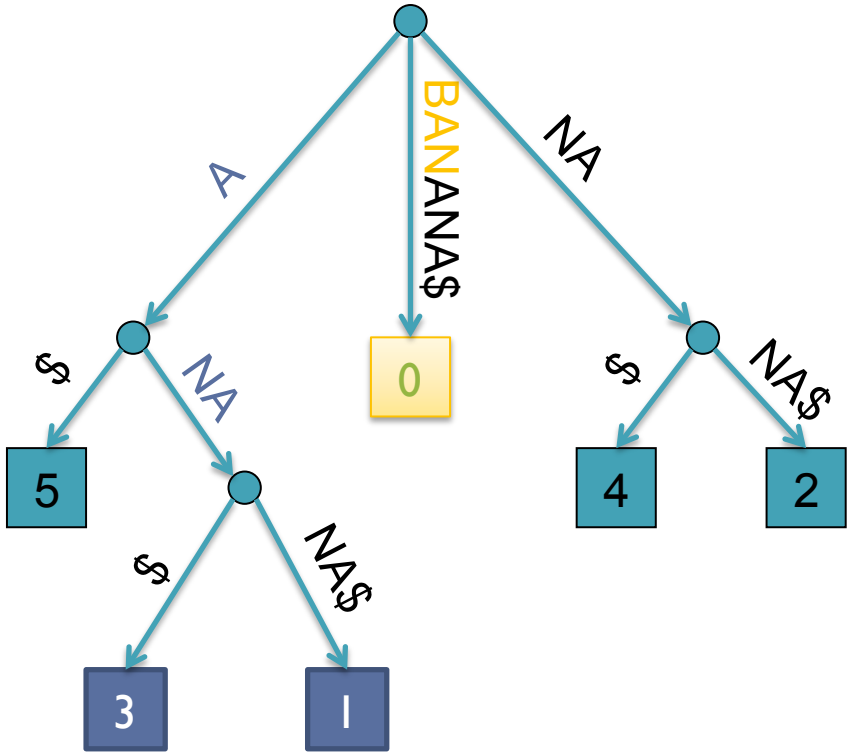
Fast Indexing with Suffix Trees



- Tree of all suffixes of string S
  - Suffix i encoded on path to leaf i
  - Nodes: positions where suffixes diverge
  - Edges: substrings of S
  - Leaves: starting position of suffix
- O(n) Construction
  - Ukkonen’s Algorithm
  - O(|Σ|n) space
  - Exploits inter-suffix relationships and suffix links
- O(q) Substring Matching
  - Walk from root following the characters in the query Q.
  - One leaf for each occurrence of Q
  - Allows variable length searches
  - Use suffix links to quickly match all substrings of the query

Fast Indexing with Suffix Trees

Suffix tree of “BANANA$”



- Tree of all suffixes of string S
  - Suffix i encoded on path to leaf i
  - Nodes: positions where suffixes diverge
  - Edges: substrings of S
  - Leaves: starting position of suffix
- O(n) Construction
  - Ukkonen’s Algorithm
  - O(|Σ|n) space
  - Exploits inter-suffix relationships and suffix links
- O(q) Substring Matching
  - Walk from root following the characters in the query Q.
  - One leaf for each occurrence of Q
  - Allows variable length searches
  - Use suffix links to quickly match all substrings of the query

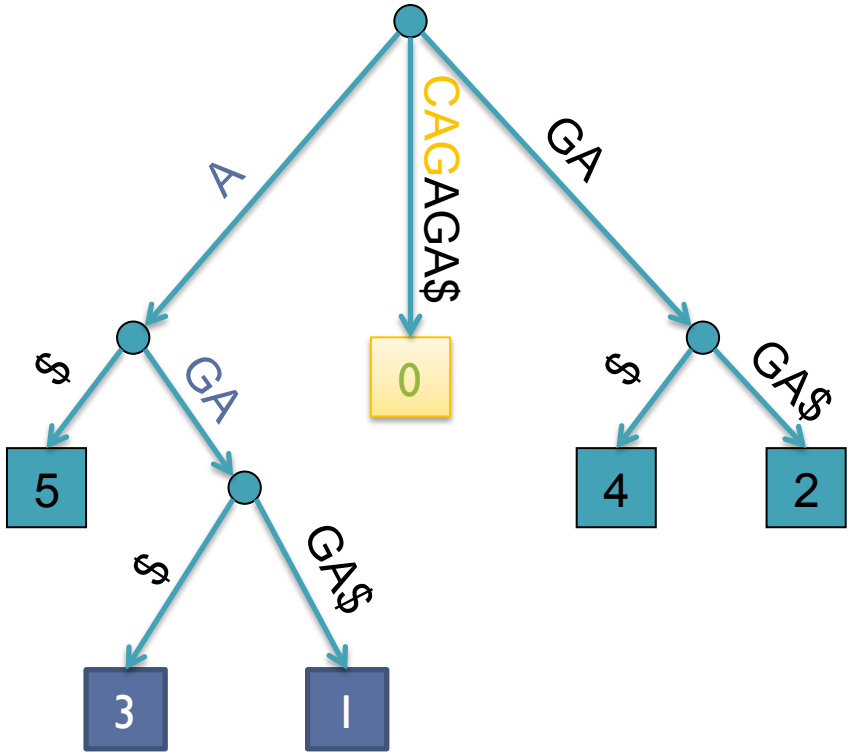
Searching for “BAN” => 0

Searching for “ANA” => 1,3

Searching for “ANN” => Partial match at 1,3

Suffix Trees for DNA Sequences

Suffix tree of "CAGAGA$"

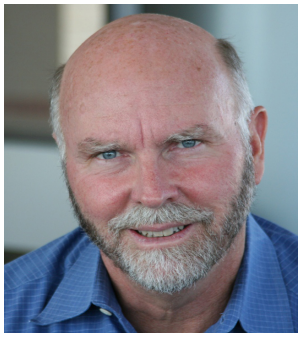


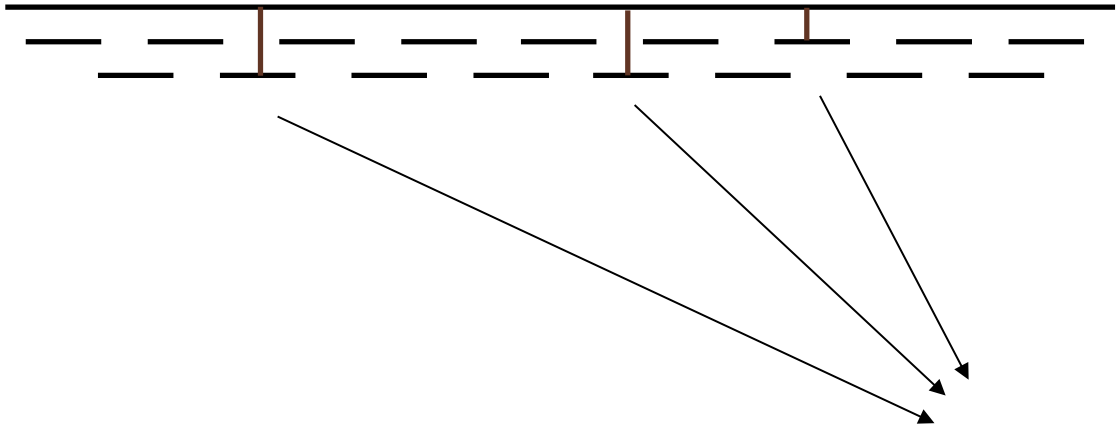
Searching for "CAG" => 0
Searching for "AGA" => 1,3
Searching for "AGG" => Partial match at 1,3

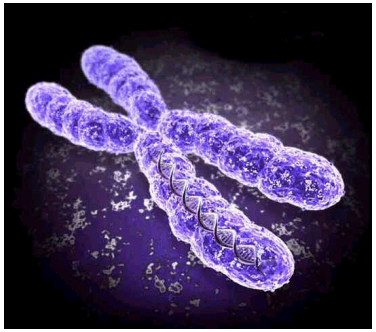
- Genome of an organism encodes the genetic information in long sequence of 4 DNA nucleotides: Σ =ACGT
  - Bacteria: ~5 million bp
  - Humans: ~3 billion bp
- Current DNA sequencing machines can generate 1-2 Gbp of sequence per day
  - Millions of short reads (25-300bp)
- Recent studies of individual human genomes used 3.3 (Wang, et al., 2008) & 4.0 (Bentley, et al., 2008) billion 36bp reads
  - Mapped reads to reference human genome to discover variations between people
  - Many more studies underway

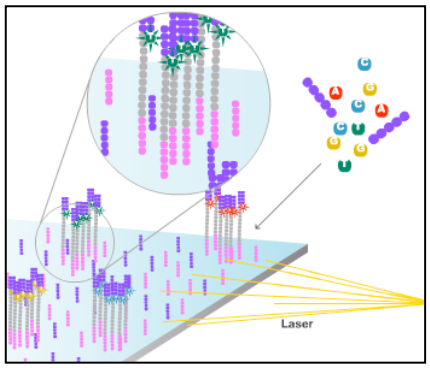
Personal Genomics

- How does your genome compare to Craig's?







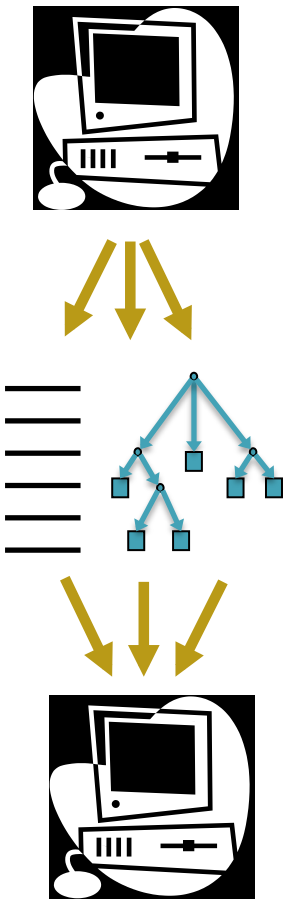


Heart Disease

Cancer

Brilliant Professor

MUMmerGPU 1.0 Overview



1. Load reference & construct suffix tree
2. Load query strings
3. Transfer data to GPU
4. Execute match kernel
   - Many simultaneous matches
5. Fetch results from GPU
6. Post-process & output results

High-throughput sequence alignment using Graphics Processing Units.
Schatz, MC, Trapnell, C, Delcher, AL, Varshney, A. (2007) *BMC Bioinformatics* 8:474.

MUMmerGPU 1.0 Results

Reference	Reference Length (bp)	# queries	Query length mean ± stdev	Min alignment length (l)	Speedup
C. briggsae Sanger sequencing	13,163,117	2,357,666	717.84 ± 159.44	100	3.71
L. monocytogenes 454 pyrosequencing	2,944,528	6,620,471	200.54 ± 60.51	20	3.79
S. suis Illumina/Solexa sequencing	2,007,491	26,592,500	35.96 ± 0.27	20	3.47

- Compare MUMmerGPU versus standard MUMmer
  - End-to-end runtime ~3.5x faster than CPU version
  - GPU matching was 10x faster than CPU version

- Runtime dominated by post-processing matches for printing.
  - Match kernel finds coordinates in suffix tree, explore subtrees to find coordinates in the reference
  - Suffix tree construction, host-device IO were not a significant fraction of the runtime

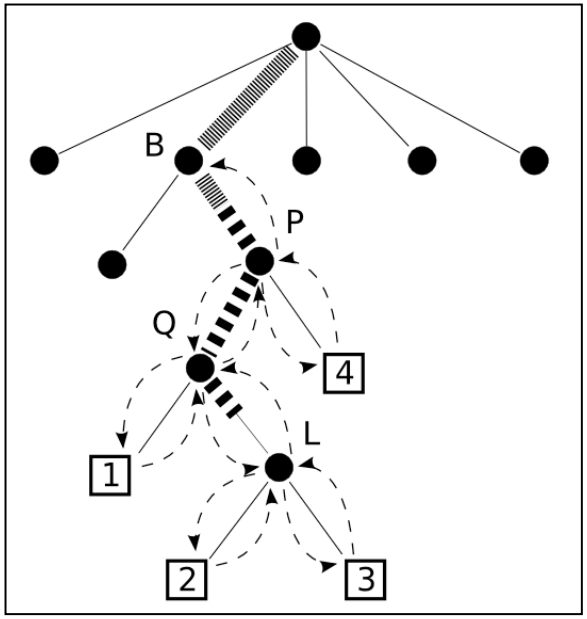
MUMmerGPU 2.0 Highlights

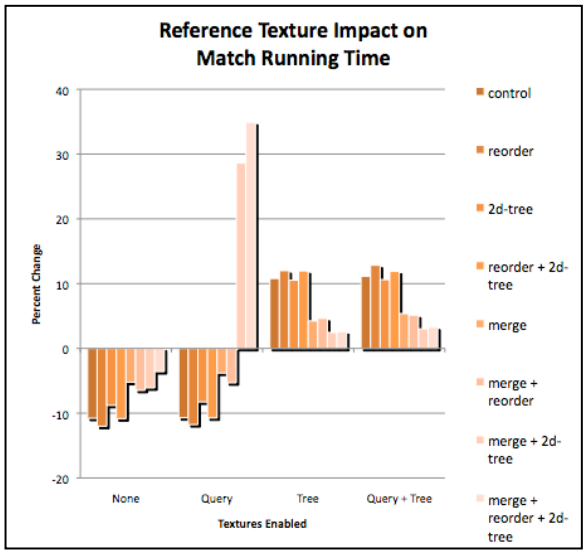
- Rewrite the serial post-match print routines as a parallel GPU kernel
  - Stackless depth first search of the suffix tree
  - Explore a maze by keeping your right hand on the wall at all times

- Kernel Performance Tuning
  - Quantify relative performance of 128 variations from 7 binary options
  - Optimize register use & processor occupancy
  - Use textures to minimizing memory latency, but be careful of cache contention

- Overall effect:
  - Match kernel: up to 25% faster
  - Print kernel: up to 4x faster

* Paper under review, see me for preprint





Grand Challenge of Biology

"NextGen sequencing has completely outrun the ability of good bioinformatics people to keep up with the data and use it well… We need a MASSIVE effort in the development of tools for "normal" biologists to make better use of massive sequence databases."

Jonathan Eisen – JGI Users Meeting – 3/28/09

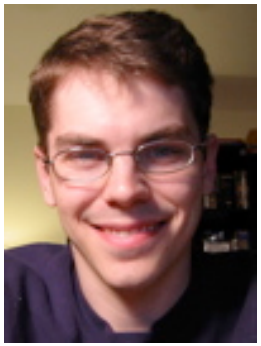
Contributions

- Dramatically accelerate personal genomics on commodity hardware
- Developed novel GPU kernels, and guidelines for data intensive GPGPU programming

More information:

- http://mummergpu.souceforge.net

Acknowledgements



Cole Trapnell

Art Delcher

Amitabh Varshney

Steven Salzberg

CBCB
Center for Bioinformatics & Computational Biology

UMIACS
University of Maryland Institute for Advanced Computer Studies

NVIDIA

Thank You!