

Sequence Alignment with GPU: Performance and Design Challenges

Gregory M. Striemer and Ali Akoglu

{gmstrie, akoglu}@ece.arizona.edu

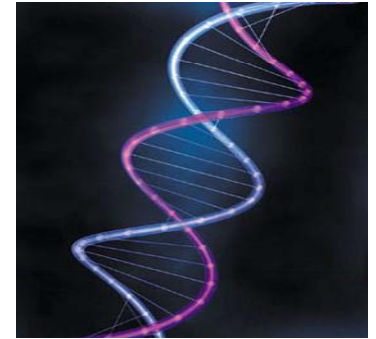
Reconfigurable Computing Lab

Department of Electrical and Computer Engineering

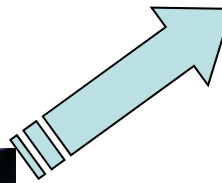
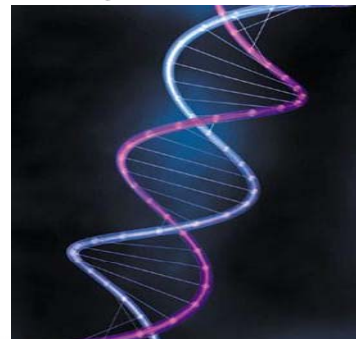
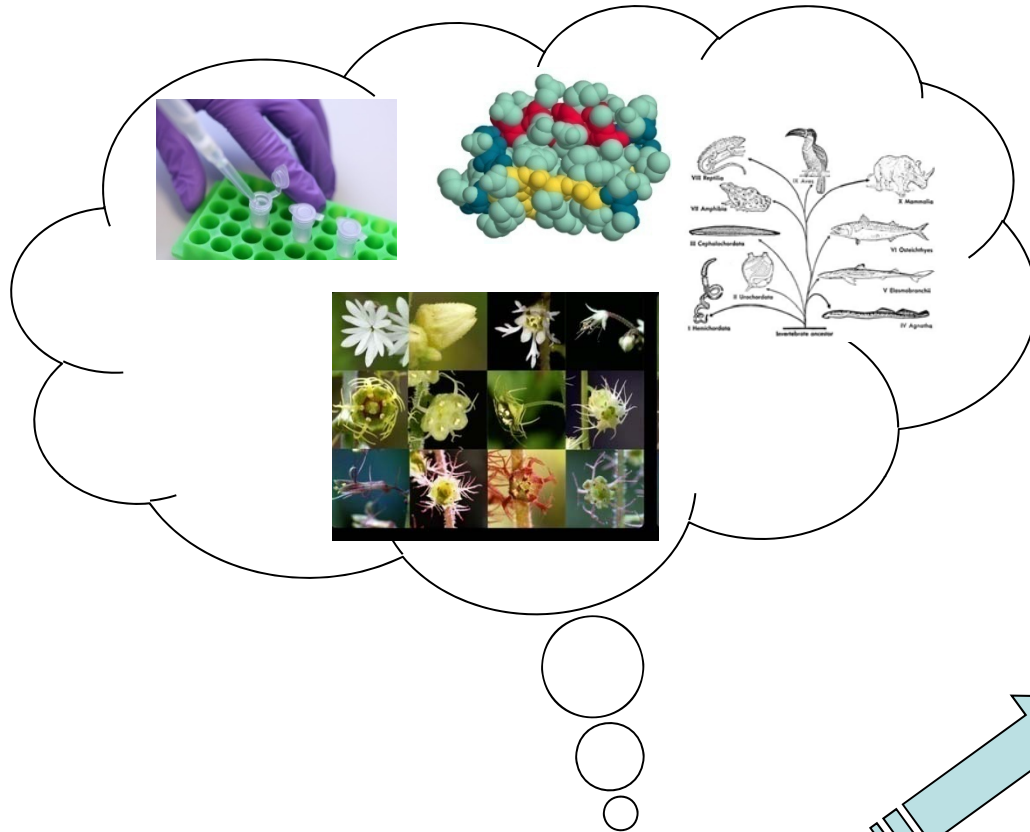
University of Arizona



- Motivation
- Project Goals
- Hardware Architecture (Tesla GPU)
- Software Architecture (Smith-Waterman)
- Parallelization Strategy
- Results



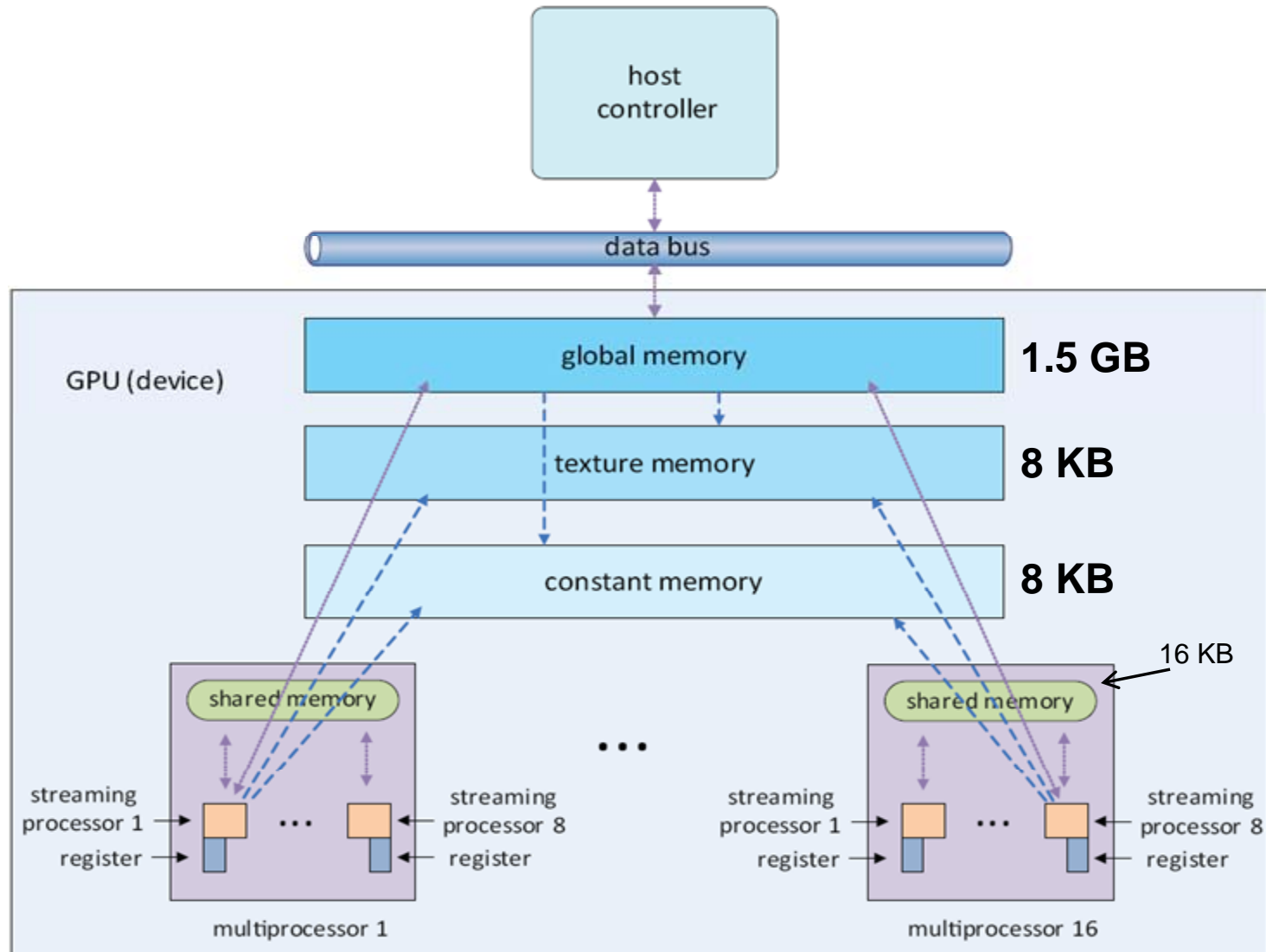
Motivation



- Explore the GPU as a massively parallelized scientific computing device
 - Learn the GPU architecture and its intricacies
 - Study how programs must be parallelized to work with the GPU architecture
 - Find ways to exploit GPU memory hierarchy
- Mapping software architecture onto hardware architecture
 - Demonstrate the feasibility of mapping software architectures to the GPU and the drawbacks
 - Explore what makes software architectures suitable for the GPU
- Run software purely on the GPU
 - Design software mapping to be scalable



Hardware Architecture



Sequence Alignment

$$H_{ij} = \max \left\{ \begin{array}{l} H_{i-1,j-1} + S_{ij}, \\ H_{i-1,j} - G, \\ H_{i,j-1} - G, \\ 0 \end{array} \right\} \quad //G = 10$$

		Database Sequence			
		A	C	D	W
Query Sequence	A	0	5	0	0
	C	0	0	18	8
	F	0	0	8	13
	W	0	0	0	3
	W	0	0	0	28

	A	B	C	D	F	W
A	5	-2	-1	-2	-3	-3
B	-2	5	-3	-3	-4	-5
C	-1	-3	13	-4	-2	-5
D	-2	-3	-4	8	-5	-5
F	-3	-4	-2	-5	8	4
W	-3	-5	-5	-5	4	15

Substitution Matrix



Manavski

- Poor memory utilization of GPU
 - Query Sequences are limited to 356 characters in length.
- Highly CPU Dependent
- Not scalable
- Requires large amounts of memory to store substitution table created from substitution matrix and query sequence

Farrar

- CPU must have SSE2 SIMD support.
Not compatible with AMD
- Does not utilize larger register size supported by SSE3
- Requires large amounts of memory to store substitution table created from substitution matrix and query sequence



Cost Function

New Cost Function

$$S_{ij} = (\text{ascii}(B) - 65, \text{ascii}(D) - 65)$$

$$S_{ij} = (66 - 65, 68 - 65)$$

$$S_{ij} = (1, 3) = -3$$

$$S_{ij} = -3$$

*Does not use modulo

Sorted Substitution Table

	A	B	C	D	...	Z
A	5	-2	-1	-2	...	-3
B	-2	5	-3	-3	...	-5
C	-1	-3	13	-4	...	-5
D	-2	-3	-4	8	...	-5
...
Z	-3	-5	-5	-5	...	15

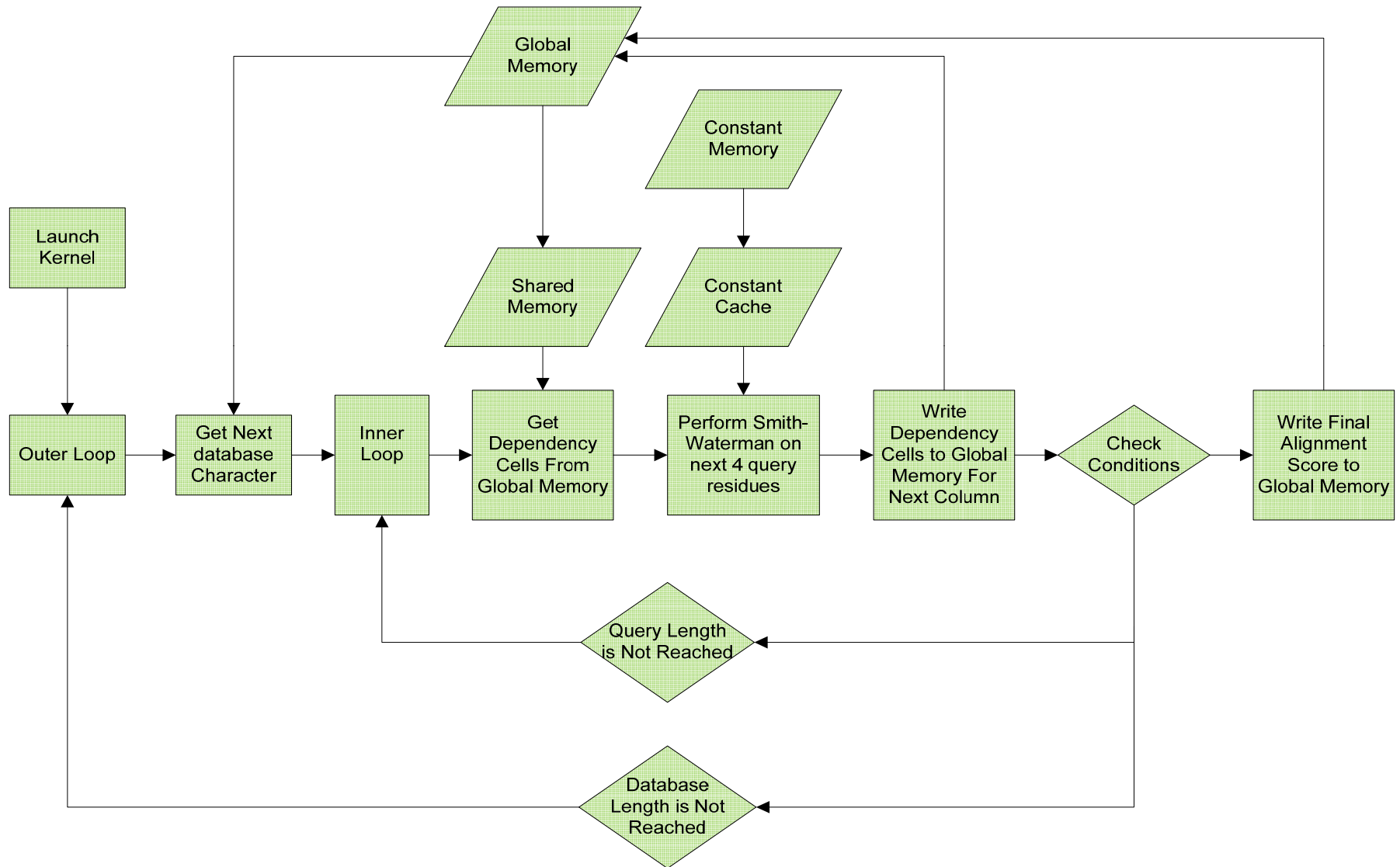
Only need space for 26x26 matrix

Previous Methods

Computed new table from substitution matrix with substitution characters for top row and query sequence for column

	A	R	N	D	...	X
Q	-1	1	0	0	...	-1
U	-1	-1	-1	-1	...	-1
E	-1	0	0	2	...	-1
R	-1	5	0	-2	...	-1
Y	-2	-2	-2	-3	...	-1
...

Space needed is 23x(Query Length)



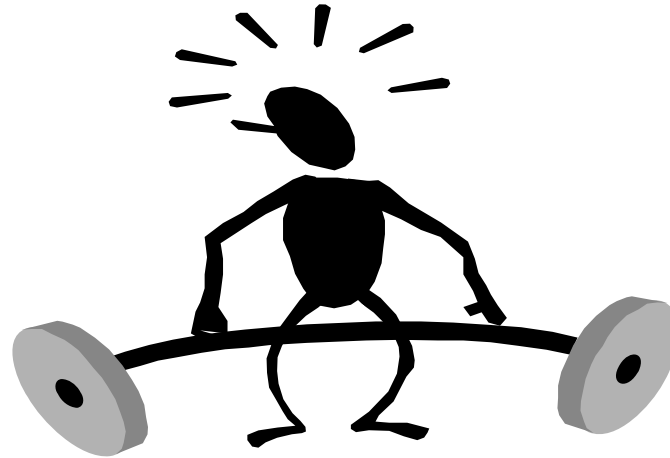
Alignment Database: Swissprot (Aug 2008), containing 392,768 sequences. GSW vs SSEARCH.

Protein Length	GPU (1.35GHz) Time (s)	SSEARCH (3.2GHz) Time(s)	Speedup	GPU Cycles (Billions)	SSEARCH Cycles (Billions)	Cycles Ratio
4	1.3	10	8	1.69	32.00	18.96
8	1.8	12	6.8	2.38	38.40	16.11
16	2.8	26	9.2	3.83	83.20	21.71
32	5.8	47	8.1	7.79	150.40	19.30
64	11.2	99	8.8	15.17	316.80	20.89
128	22.0	212	9.7	29.64	678.40	22.88
256	43.8	428	9.8	59.14	1369.60	23.16
512	92.4	886	9.6	124.78	2835.20	22.72
768	144.6	1292	8.9	195.15	4134.40	21.19
1024	279.7	1807	6.5	377.55	5782.40	15.32

Alignment Database: Swissprot (Aug 2008), containing 392,768 sequences. GSW vs Farrar.

Protein Length	GPU (1.35GHz) Time (s)	Farrar (2.4GHz) Time(s)	Speedup	GPU Cycles (Billions)	Farrar Cycles (Billions)	Cycles Ratio
4	1.3	2.78	2.14	1.69	6.67	3.95
8	1.8	2.81	1.56	2.38	6.74	2.83
16	2.8	2.96	1.06	3.83	7.10	1.85
32	5.8	3.35	0.58	7.79	8.04	1.03
64	11.2	4.46	0.40	15.17	10.70	0.71
128	22.0	6.50	0.30	29.64	15.60	0.53
256	43.8	12.77	0.29	59.14	30.65	0.52
512	92.4	23.37	0.25	124.78	56.09	0.45
768	144.6	26.61	0.18	195.15	63.86	0.33
1024	279.7	37.47	0.13	377.55	89.93	0.24

Questions



Farrar's SIMD Implementation

Execution Time vs Sequence Length

