

## DOCTORAL THESIS

### GPU accelerated sequence alignment

Zhao, Kaiyong

*Date of Award:*  
2016

[Link to publication](#)

#### General rights

Copyright and intellectual property rights for the publications made accessible in HKBU Scholars are retained by the authors and/or other copyright owners. In addition to the restrictions prescribed by the Copyright Ordinance of Hong Kong, all users and readers must also observe the following terms of use:

- Users may download and print one copy of any publication from HKBU Scholars for the purpose of private study or research
- Users cannot further distribute the material or use it for any profit-making activity or commercial gain
- To share publications in HKBU Scholars with others, users are welcome to freely distribute the permanent URL assigned to the publication

---

## Abstract

DNA sequence alignment is a fundamental task in gene information processing, which is about searching the location of a string (usually based on newly collected DNA data) in the existing huge DNA sequence databases. Due to the huge amount of newly generated DNA data and the complexity of approximate string match, sequence alignment becomes a time-consuming process. Hence how to reduce the alignment time becomes a significant research problem. Some algorithms of string alignment based on HASH comparison, suffix array and BWT, which have been proposed for DNA sequence alignment. Although these algorithms have reached the speed of  $O(N)$ , they still cannot meet the increasing demand if they are running on traditional CPUs.

Recently, GPUs have been widely accepted as an efficient accelerator for many scientific and commercial applications. A typical GPU has thousands of processing cores which can speed up repetitive computations significantly as compared to multi-core CPUs. However, sequence alignment is one kind of computation procedure with intensive data access, i.e., it is memory-bounded. The access to GPU memory and IO has more significant influence in performance when compared to the computing capabilities of GPU cores. By analyzing GPU memory and IO characteristics, this thesis produces novel parallel algorithms for DNA sequence alignment applications. This thesis consists of six parts. The first two parts explain some basic knowledge of DNA sequence alignment and GPU computing. The third part investigates the performance of data access on different types of GPU memory. The fourth part describes a parallel method to accelerate short-read sequence alignment based on BWT algorithm. The fifth part proposes the parallel algorithm for accelerating BLASTN, one of the most popular sequence alignment software. It shows how multi-threaded control and multiple GPU cards can accelerate the BLASTN algorithm significantly. The sixth part concludes the whole thesis.

To summarize, through analyzing the layout of GPU memory and comparing data under the mode of multithread access, this thesis analyzes and concludes a perfect optimization method to achieve sequence alignment on GPU. The outcomes can help practitioners in bioinformatics to improve their working efficiency by significantly reducing the sequence alignment time.

---

# Table of Contents

DECLARATION .....	i
Abstract .....	ii
Acknowledgements .....	iii
Table of Contents .....	iv
List of Tables .....	vii
List of Figures.....	viii
Chapter 1 Introduction and Background.....	1
1.1 The scoring model .....	1
1.2 Background of sequence alignment.....	2
1.3 Short read alignment algorithms.....	4
1.4 GPU computing .....	6
1.5 Motivation .....	7
Chapter 2 Research Problem and Literature Review .....	9
2.1 Sequence alignment in biology.....	9
2.2 Sequence alignment algorithms.....	11
2.3 Dynamic programming algorithms.....	12
2.4 Seed match and extend alignment algorithms .....	13
2.5 GPU based alignment algorithms .....	16
Chapter 3 CUDA Memory Model .....	17
3.1 Global memory .....	20
3.1.1 Shared memory.....	21
3.1.2 Constant memory.....	23
3.2 Test process design.....	24
3.2.1 Demand analysis.....	24
3.2.2 Experimental design .....	24
3.2.3 Testing code schema.....	27
3.3 Experimental results .....	31
3.3.1 Experimental environment .....	31
3.3.2 Experiment analysis.....	32

---

3.3.3	Multi-access with global and shared memory I/O.....	43
3.3.4	GPU memory model.....	47
Chapter 4	GPU Accelerated SOAP3.....	50
4.1	Background of SOAP3.....	50
4.1.1	Suffix string.....	50
4.1.2	The Trie structure.....	50
4.1.3	The suffix array.....	52
4.1.4	Introduction to BWT algorithm.....	53
4.1.5	FM-index.....	54
4.2	Reduce memory access.....	57
4.3	Coalescing memory accesses.....	58
4.4	Reduce branching effect.....	59
4.5	The division of the kernel.....	59
4.6	Experimental results.....	59
Chapter 5	G-BLASTN: accelerating nucleotide alignment by graphics processors.....	61
5.1	Introduction.....	61
5.2	BLASTN algorithms.....	63
5.3	First method of G-BLASN.....	65
5.3.1	The main parameters for CUDA implementation.....	66
5.3.2	Conclusions from this research.....	67
5.4	Design of G-BLASTN.....	69
5.5	Implementation.....	72
5.5.1	Accelerating the seeding step by GPU.....	72
5.5.2	Accelerating the mini-extension step by GPU.....	76
5.5.3	Optimizing the trace-back step.....	78
5.5.4	Pipeline mode for multiple queries.....	79
5.6	Results.....	80
5.6.1	General setup and data sets.....	80
5.7	Experimental results.....	83
5.7.1	Performance under normal mode.....	83
5.7.2	Performance under pipeline mode.....	94
5.8	Discussions and Conclusions.....	94

---

Chapter 6 Conclusion and future work .....	96
Publication List .....	98
References .....	101
CURRICULUM VITAE .....	107