Contents

Acknowledgements **I**

Contents **II**

Index **V**

List of Figures **VII**

List of Tables **IX**

List of Algorithms **XI**

1 Introduction **1**
1.1 Overview 1
1.2 Motivation 2
1.3 Contents of the PhD thesis 3

2 State of the Art **5**
2.1 Backward search with the FM-Index 5
2.1.1 BWT calculation 6
2.1.2 Suffix Array sorting algorithms 7
2.1.3 FM-Index data structures 8
2.1.4 FM-Index of the complementary strand 9
2.1.5 Recursive Backward Search algorithm 9
2.1.6 Iterative Backward Search algorithm 13
2.2 General Purpose GPU 14
2.3 Sensitivity (TPR) 17
2.4 Sequence alignment tools 18

3 Objectives and contributions **25**
3.1 Objectives 25

4 FM-Index search on hybrid CPU-GPU environments 29
4.1 FM-Index compression 29
4.2 GPU exact search algorithm 30
4.2.1 Implementation details 31
4.2.2 Pthread multi GPU version 32
4.2.3 Experimental results 33
4.3 Hybrid CPU-GPU inexact search algorithm 39
4.3.1 Implementation details 41
4.3.2 Bounding techniques explanation 44
4.3.3 Experimental results 45
4.3.4 Conclusions 48

5 Faster and more accurate inexact mapping using advanced tree exploration on backward search methods 51
5.1 Suffix Array compression 52
5.1.1 Vector S compression 52
5.1.2 Vector R compression 53
5.2 Search tree exploration prototype 55
5.3 Search tree exploration complete algorithm 59
5.4 Compatibility interface 62
5.5 Experimental results 64
5.5.1 Comparison with other FM-Index only algorithms 64
5.5.2 Preprocessing step for modern aligners 66
5.5.3 Comparison between BWT and *csalib* runtimes 67
5.5.4 Asymptotic analysis 69
5.6 Conclusions 70

6 General conclusions 71
6.1 Exact mapping on GPU algorithm 71
6.2 Hybrid CPU-GPU pair-end algorithm 72
6.3 Advanced search tree exploration inexact mapping algorithm . . . 73

7 Relevant publications and source code 77

Bibliography 79

List of Figures

2.1 Nvidia Tesla GPU architecture 21
2.2 Nvidia GPU memory hierarchy 22
2.3 Nvidia kernel 23

4.1 Multiple threads in CPU 33
4.2 Profiling of the algorithm without operative system disk cache . . 35
4.3 Profiling of the algorithm with operative system disk cache 35
4.4 Obtaining the optimum GPU block size 37
4.5 Speedup impact of the number of reads per GPU kernel execution 37
4.6 Short sequence mapping tools comparison finding only the first occurrence 40
4.7 Short sequence mapping tools comparison finding all occurrences . 40
4.8 Hybrid CPU-GPU, execution times 47
4.9 Hybrid CPU-GPU, mapping locations found 47

5.1 Number of partial solutions. Values of *res* during a forward search of string "AGGATC" against the reference "AGGAGC$" 57
5.2 Complete inexact search algorithm. Example for 2 errors. 59
5.3 BWT only tools. 2 Million 250bps reads. Execution times from 0 to 7 errors. 65
5.4 BWT and SW tools. 2 Million 250bps reads. Execution times comparing the new algorithm, the modern mappers and the combination of both. 68
5.5 BWT and SW tools. 2 Million 400bps reads. Execution times comparing the new algorithm, the modern mappers and the combination of both. 68
5.6 BWT and csalib runtimes. 2 Million 250bps reads. Execution times from 0 to 7 errors. 69

List of Tables

4.1 Effectiveness of the hybrid model 45
4.2 Sequential execution of the hybrid CPU-GPU algorithm 46
4.3 Comparison with SOAP3-dp. 48

5.1 Results for Soap 2, Bowtie 1 and the new algorithm. The dataset contains 2 million 250bps reads. 66

List of Algorithms

2.1 Exact Backward Search 14
2.2 Search iteration with the FM-Index 14

4.1 Matrix O decompression 30
4.2 GPU exact search algorithm 31
4.3 Sequential execution 41
4.4 Backward Vector GPU 42
4.5 Backward Helper CPU 43

5.1 Vector S decompression 52
5.2 Vector R decompression 54
5.3 Search prototype 56
5.4 Exact Subroutine 57
5.5 Branch Subroutine 58
5.6 Complete Inexact Search (Step I figure 5.2) 61
5.7 Calculate D forward (Step I figure 5.2) 63