ACCURATE ALIGNMENT OF SEQUENCING READS FROM VARIOUS GENOMIC ORIGINS

LIM JING QUAN

NATIONAL UNIVERSITY OF SINGAPORE

2014

ACCURATE ALIGNMENT OF SEQUENCING READS FROM VARIOUS GENOMIC ORIGINS

LIM JING QUAN
(B.CompSc.(Hons), NUS)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
NATIONAL UNIVERSITY OF SINGAPORE

2014

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has not been submitted for any degree in any university previously.

_____

Lim Jing Quan

18/July/2014

Acknowledgements

I thank my thesis supervisor Dr Sung Wing-Kin for his impeccable patience, selfless guidance and sharing of his invaluable knowledge over the course of my candidature. I am also glad to have Prof. Wong Lim Soon and Prof. Tan Kian Lee to be my thesis advisory committee members. I am thankful to Dr Wei Chia-Lin, Dr Li Guoliang, Dr Eleanor Wong and Dr Chandana Tennakoon for successful collaboration on some of the projects which I have worked on and have eventually made up parts of this thesis. I would also like to thank Dr Teh Bin Tean, Dr Lim Weng Khong, Sanjanaa and Saranya from Duke-NUS graduate medical school for accommodating me while I was still working on this thesis.

The pursuit for knowledge over these years has not been a bed of roses for me. There was a point of time when I had wanted to quit my candidature. I am grateful that I have still managed to turn back, pull through and reach 'this' particular point of the thesis. To my comrades whom have made the lab an enjoyable place to work in, I thank you all in no particular order of favor or seniority: Sucheendra, Chuan Hock, Javad, Hugo Willy, Hoang, Zhizhuo, Xueliang, Chandana, Rikky, Gao Song, Peiyong, Ruijie, Narmada, Liu Bing, Difeng, Tsung Han, Benjamin G., Wang Yue, Michal, Wilson, Hufeng, Chern Han, Mengyuan, Kevin L., Alireza, Ramanathan and Ratul for inspiration and for contributing to the finishing of this thesis in various ways.

Finally, I would like to thank my family and Chu Ying for their patience. Once again, I thank all of you for keeping me aspired and hopeful towards the end of my candidature.

Contents

1 Introduction 1
1.1 History of DNA Sequencing 3
1.1.1 First-Generation sequencing 3
1.1.2 Second-Generation sequencing 4
1.1.3 Third-Generation sequencing 5
1.2 Motivation 6
1.2.1 Looking at the DNA with an intent 6
1.3 General workflow on sequencing reads 7
1.4 The mapping challenge 8
1.5 Contribution of thesis 9
1.6 Organization of the thesis 10
2 Basic Biology and Sequencing Technologies 11
2.1 Basic Biology 11
2.2 Central Dogma of Molecular Biology 13
2.2.1 DNA-DNA Replication 15
2.2.2 DNA-RNA Transcription 17
2.2.3 RNA-Protein Translation 20
2.3 Next Generation Sequencing Technologies 21
2.3.1 Roche/454 Sequencing 22
2.3.2 Ion Torrent Sequencing 23

2.3.3 Illumina/Solexa Sequencing 23
2.3.4 ABI/SOLiD Sequencing 25
2.3.5 Comparison 26
2.4 Origins and representations of sequenced data 27
2.4.1 Whole-genome and targeted sequencing 27
2.4.2 RNA-seq – mRNA 28
2.4.3 Epigenetic sequencing 29
2.4.4 Base-space and color-space reads 30
2.4.5 Computational representation of data 32
3 Survey of Alignment Methods 33
3.1 Basics of Genomic Alignments 33
3.2 Bisulfite-treated DNA-seq aligners 35
3.2.1 Challenges in aligning BS-seq reads 35
3.2.2 BS-aligner for Base-space reads 37
3.2.3 BS-aligner for Color-space reads 37
3.2.4 Methylation-aware mapping 38
3.2.5 Unbiased-Methylation mapping 39
3.2.6 Semi Methylation-aware mapping 40
3.2.7 Comparison of BS-Seq Aligners 41
3.3 Gapped DNA-seq aligners 43
3.3.1 Challenges in Gapped Alignment 44
3.3.2 Hash/Seed based Approaches 45
3.3.3 Prefix/Suffix trie based approaches 48
3.3.4 Hardware acceleration of seed-extension 52
3.3.5 Comparison of Gapped DNA-Seq Aligners 53
3.4 RNA-seq aligners 58
3.4.1 Challenges in RNA-seq Alignment 59
3.4.2 Unspliced/Annotation-guided Aligners 60

3.4.3 Spliced Aligner 61
3.4.4 Comparison of RNA-seq Aligners 64
4 Bisulfite Sequencing Reads Alignment 69
4.1 Introduction 69
4.2 Related Work 70
4.3 Results 73
4.3.1 Evaluated programs and performance measures 73
4.3.2 Evaluation on the simulated Illumina data 75
4.3.3 Evaluation on the real Illumina data 76
4.3.4 Evaluation on the simulated SOLiD data 79
4.3.5 Evaluation on the real SOLiD data 80
4.4 Materials and Methods 81
4.4.1 Methods for base reads 81
4.4.2 Methods for color reads 87
4.5 Prediction of Imprinting Genes using BatMeth 94
4.5.1 Results 95
4.5.2 Methods and Material on Prediction of Imprinting Genes 108
4.6 Discussion 110
4.7 Conclusions 111
5 Gapped Alignment Problem 113
5.1 Introduction 113
5.2 Related Work 114
5.3 Results 115
5.3.1 Simulation study on variant-spanning reads 115
5.3.2 Compared methods and method of cross-comparison 119
5.3.3 Simulation of data 120
5.3.4 Evaluation on ART-simulated reads 121
5.3.5 Evaluation on simulated pure-indel reads 125

5.3.6 Evaluation on paired reads 128
5.3.7 Evaluation on real reads 132
5.3.8 Evaluation on life-sized dataset 133
5.4 Methods 139
5.4.1 Problem definition and overview of the method 139
5.4.2 Reverse-alignment 139
5.4.3 Determining F 140
5.4.4 Deep-scan 140
5.4.5 BatAlign algorithm 141
5.4.6 Handling long reads 141
5.4.7 Enumerating hits 141
5.4.8 Finding indel hits. 142
5.4.9 Faster semi-global alignment and SW alignment 142
5.4.10 Alignment score and mapping quality 144
5.4.11 Accelerating alignment 144
5.5 Conclusion 144
6 Spliced Alignment Problem 147
6.1 Introduction 147
6.2 Challenges in Spliced Alignment 148
6.3 Related Work 149
6.4 Results 150
6.4.1 Setup of experiments and performance measures used 150
6.4.2 Evaluation on the simulated RNA-seq Illumina-like reads 151
6.4.3 Evaluation on real RNA-seq Illumina-like reads 154
6.5 Evaluation on running time 158
6.6 Methods 159
6.6.1 Simulation of data and validation of simulated data 160
6.6.2 Overview of Method 160

6.6.3 Motivation for using BatAlign as a seeding tool 161
6.6.4 Phase 1 – Resolve exonic region within a single read 162
6.6.5 Phase 2 – Search for junctions from an anchored region 163
6.6.6 Phase 3 – Refine alignments due to splice junction near ends of reads 165
6.6.7 Data structure for efficient pairing of genomic coordinates 167
6.6.8 Details of implementation 168
6.6.9 Discussion 169
7 Conclusion 173
7.1 BatMeth 173
7.2 BatAlign 174
7.3 BatRNA 175
7.4 Future Developments 175
Bibliography 177
Appendix A 192
A.1 Additional information on profiling methylation libraries 192

Summary

Sequencing technologies have revolutionized the study of genomes by generating high throughput data for various studies which are not cost-efficient when done with Sanger sequencing. The first step in analyzing these high throughput data is often to find the original location from which the data reads are sequenced from a reference genome. Moreover, references genomes can be very large (human genome ~3.2GB). This calls for better methodologies in aligning reads onto a reference genome.

In this thesis, we present three methodologies in producing accurate alignments of DNA-sequencing reads with bisulfite-induced nucleotide conversion, DNA-sequencing reads with mismatches and gaps, and RNA-sequencing reads with intronic spliced junctions.

Our first contribution is BatMeth; a fast, sensitive and accurate aligner for DNA-sequencing reads derived from sodium bisulfite treatment. BatMeth is designed to handle both base-space and color-space bisulfite-treated reads. Based on List-Filtering, Mismatch-Stage-Filtering, BatMeth was able to avoid examining spurious hits and improve the efficiency and specificity of our alignment. Our experiments also show that BatMeth can produce better methylation callings across samples of different bisulfite conversion rates.

BatAlign is our next contribution which can align DNA-sequencing reads in the presence of both mismatches and insert-delete (indel) accurately. Two novel

strategies called Reverse-Alignment and Deep-Scan are developed to enable the efficient reporting of accurate alignments for these reads. Reverse-Alignment starts the alignment of a read by looking for the most probable preliminary alignments incrementally. Deep-Scan refines the preliminary alignments by searching for a targeted subset of less probable alignments to better distinguish the best alignment from the rest. BatAlign was able to achieve competitive runtime efficiency with SIMD-enabled Smith-Waterman algorithm for the extension of seeds from a long read in our seed-and-extend strategy.

Our last contribution is BatRNA is designed to recover splice alignment of a RNA-sequencing read sensitively and efficiently. As RNA-sequencing datasets can have very varying mixture of exonic and spliced reads in them, BatAlign was introduced in BatRNA as a pre-mapping tool to draft up the possible spliced sites of the genome. After which, we filtrate the reads from the mappings of BatAlign to be mapped by BatRNA for possible spliced alignments of the reads. The resultant mappings from both BatAlign and BatRNA are considered for the final alignment of a read. Compared with other popular and recent RNA-sequencing aligners, BatRNA was able to produce very sensitive and accurate alignments in a dataset of mixed exonic and spliced reads, while maintaining competitive runtimes.

In summary, we have developed various methodologies to align reads on to a reference genome, sequenced from various genomic origins, accurately and sensitively.

List of Tables

2.1. Comparison between some commercialized sequencing platforms in the market..... 27

3.1. The possible text-edit operations which can be represented by a CIGAR for the alignment of a query string onto a reference text 34

3.3. Methods for gapped alignment and their respective main indexing/mapping strategies 54

3.4. Methods for RNA-seq alignment and their respective mapping strategies and usage of annotations for spliced alignments 65

4.1. Comparison of mapping efficiencies and estimation of methylation levels in various genomic contexts 78

4.2. Comparison of speed and unique mapping rates on three lanes of human BS data... 78

4.3. Unique mapping rates and speed on 100,000 real color reads 83

4.4. Possible ways to map a BS read onto the converted genome 84

4.5. Cutoffs for list filtering on simulated reads from the Results section..... 85

4.6. Possible ways to map a BS color read onto the converted color genome 89

4.7. Top pMRs across cell types 103

4.8. Characterization of partial methylated region (pMRs) 108

5.1A-C. A..Number of first (or best) alignment reported by various methods on simulated 75bp dataset. B. Number of first (or best) alignment reported by various methods on simulated 100bp dataset. C. Number of first (or best) alignment reported by various methods on simulated 250bp dataset 126

5.2. Alignments on a 5% discordant simulated paired-end dataset of various read-lengths by all compared methods. 131

5.3A. Number of indel-variants called from the sub-samples at 70 PCR-validated sites137

5.3B. Number of SN-variants called from the subsamples at 67 PCR-validated sites ... 137

5.4. Comparison of running times across all discussed programs on 1 million reads from SRR315803..... 138

6.1. The F1-scores of the compared methods on BEERS-simulated 2M datasets. 152

6.2. Breakdown of alignment performance by exonic and spliced reads using simulation. 153

6.3a. Tabulation of correct hits ranked by the order in which they were reported for a read. 154

6.3b. Tabulation of wrong hits being reported alongside a rank-k correct hit. 154

6.4. Wall-clock time of compared methods on different sets of 2 million reads. 159

A1.1. Bisulfite libraries information. 195

A1.2. Statistics of bisulfite library mapping and partial-methylation calling 196

List of Figures

1.1. General workflow on sequencing reads 8

2.1. Schematic diagram of a typical animal cell. Source: [33] 11

2.2. Two main types of genomic tasks and their respective downstream analysis. De novo tasks involve the manipulation of read data without a reference. Profiling tasks use the alignment of the read on a reference for analysis..... 13

2.3. The general cases of the central dogma of molecular biology for eukaryotic cells ... 14

2.4. Three postulated methods for DNA replication prior to Meselson-Stahl experiment 16

2.5. Schematic diagram of DNA replication at a replication fork. Source: [52]..... 17

2.6. Illustration of introns and exons in pre-mRNA and the maturation of mRNA by splicing. 19

2.7. Schematic diagram of bridge amplification forming cluster stations. Source: [68]... 24

2.8. Workflow of ligase-mediated sequencing approach from ABi SOLiD. Source: [68] 25

2.9. 2-base encoding scheme used by SOLiD sequencers. Source: [68] 31

3.1. PCR amplification of bisulfite treated genomic DNA. The original strands of the DNA undergo bisulfite conversion with unmethylated-C changing to U and methylated-C remaining unchanged after the treatment. Methylated (Red) and Unmethylated (Green). 36

4.1. (a,b) Base call error simulation in Illumina and SOLiD reads reflecting one mismatch with respect to the reference from which they are simulated in their respective base- and color-space. (b) A naïve conversion of color read to base space, for the purpose of mapping against the base space reference, is not recommended as a single color base error will introduce cascading mismatches in base space. (c) A BS conversion in base space will introduce two adjacent mismatches in its equivalent representation in color space. 72

4.2. Benchmarking of programs on various simulated and real data sets (a) Benchmark results of BatMeth and other methods on the simulated reads: A, BatMeth; B, BSMAP; C, BS-Seeker; D, Bismark. The timings do not include

index/table building time for BatMeth, BS-Seeker, and Bismark. These three programs only involve a one-time index-building procedure but BSMAP rebuilds its seed-table upon every start of a mapping procedure. (b) Insert lengths of uniquely mapped paired reads and the running times for the compared programs. (c) Benchmark results on simulated SOLiD reads. Values above the bars are the percentage of false positives in the result sets. The numbers inside the bars are the number of hits returned by the respective mappers. The graph on the right shows the running time. SOCS-B took approximately 16,500 seconds and is not included in this figure. (d) BS and non-BS induced (SNP) adjacent color mismatches. 77

4.3. A total of 106, 75 bp long reads were simulated from human (NCBI37) genomes. Eleven data sets with different rates of BS conversion, 0% to 100% at increments of 10% (context is indicated), were created and aligned to the NCBI37 genome. (a-e) The x-axis represents the detected methylation conversion percentage. The y-axis represents the simulated methylation conversion percentage. (f) The x-axis represents the mapping efficiency of the programs. The y-axis represents the simulated methylation conversion percentage of the data set that the program is mapping. (a,b) The mapping statistics for various genomic contexts and mapping efficiency with data sets at different rates of BS conversion for BatMeth and B-SOLANA, respectively. (c-e) Comparison of the methylated levels detected by BatMeth and B-SOLANA in the context of genomic CG, CHG and CHH, respectively. (f) Comparison of mapping efficiencies of BatMeth and B-SOLANA across data sets with the described various methylation levels. 82

4.4. Outline of the mapping procedure. (a) Mapping procedure on Illumina BS base reads. (b) Mapping procedure on SOLiD color-space BS reads. 86

4.5. Partial methylation callings. a) Categories of methylation levels; b) Validation of DNA methylation callings from bisulfite sequencing with 27K DNA methylation array in embryonic stem cell H9. The darker the color, the higher ratio from the observed consistent DNA methylation callings to the randomly expected consistent callings; c) Numbers of partially-methylated Cs from individual cell-lines and tissues. 98

4.6. Genomic profile of partial methylated C. a) Distributions of partially-methylated Cs in CpAs, CpCs, CpGs and CpTs from individual cell lines. The cell lines are sorted by the proportion of partially-methylated Cs from CpGs. b) Distributions of partially-methylated Cs in CAAs, CACs, CAGs and CATs from embryonic stem cells H1, H9, induced pluripotent stem cell iPS19.11, and blood cells. Cell-lines H1, H9, and iPS19.11 are enriched with CAGs at CpA sites, while blood cells are depleted with CAGs at CpA sites. c) Profile of partially-methylated Cs from cell-lines H1, H9 and IMR90 along chromosome 1 in 100Kb bins. d) Hierarchical clustering of the profile of partially-methylated Cs from different cell lines. ES cluster includes H9, H1, H1NPC, H1_BMP4, H1_me_BMP4, and iPS19.11. Sperm cluster includes two sperm replicate DNA methylation data. HSF1 is an embryonic stem cell with low mapping rate (due to extensive "N"s in the reads). Other cell lines are in the differentiated cell cluster (Diff. cells). 100

4.7. Partial methylation across samples. a) QQ plot of the average percentage of partially-methylated Cs in 5Kb bins across the genome from all studied cell-lines and tissues. There are regions without partially-methylated Cs as percentage 0. (b-d) Screenshots around imprinting genes *GNAS*, *PEG10*, and *MAGEL2/NDN* enriched with partially-methylated Cs in the studied cell lines. 102

4.8. Cell specificity of partial methylation. a) Heatmap of percentages of partially-methylated Cs from gender-specific partially-methylated regions. b) Screenshot of gender-specific partial methylation around gene AMELX. c) Heatmap of percentages of partially-methylated Cs from differentiated-cell-specific partially-methylated regions. d) Screenshot of partial methylation around gene PCDHB12 for differentiated-cell-specific partial methylation. Refer to Table A1.1 for male/female cell-lines, and refer to Figure 4.6d for embryonic stem cells and differentiated cells. 106

4.9. Histone modification profile and gene expressions of partially-methylated regions from embryonic stem cell H9. (a) Boxplot of the percentage of fully-methylated Cs from all the cell-lines and cell-line H9. a: average percentage of fully-methylated Cs across samples; a.au: average percentage of fully-methylated Cs across samples overlapped with conserved pMRs; a.g.m: average percentage of fully-methylated Cs across male samples overlapped with gender-specific pMRs; a.g.f: average percentage of fully-methylated Cs across female samples overlapped with gender-specific pMRs; a.Ed.E: average percentage of fully-methylated Cs across embryonic stem cells and the likes overlapped with differentiated-cell-specific regions; a.Ed.d: average percentage of fully-methylated Cs across differentiated samples overlapped with differentiated-cell-specific regions; H9: percentage of fully-methylated Cs from cell line H9; H9.au: percentage of fully-methylated Cs from cell line H9 overlapped with pMRs; H9.g: percentage of fully-methylated Cs from cell line H9 overlapped with gender-specific pMRs; H9.Ed: percentage of fully-methylated Cs from cell line H9 overlapped with differentiated-cell-specific pMRs; (b-c) Profiles of histone marks H3K4me3 (b) and H3K27ac (c) around conserved pMRs, gender-specific pMRs, and differentiated-cell-specific pMRs. (d) distribution of pMRs around gene models; (e) Boxplot of gene expressions from cell-line H9: 1/3 highly expressed genes(high), 1/3 intermediately expressed genes (inter), 1/3 lowly expressed genes (low), genes whose promoter regions overlapped with conserved pMRs (pMRs.c), gender-specific pMRs (pMRs.g), and differentiated-cell-specific pMRs (pMRs.d). 107

5.1. The difference in sensitivity and specificity between mapping paired-end datasets with simulated concordant and discordant mate-pair information. 117

5.2. a) The sensitivity and specificity of compared methods on indel reads which do not have an alternative alignment with up to 5 mismatches. b) The sensitivity and specificity of compared methods on indel reads which have an alternative alignment of at most 5 mismatches. 118

5.3. a) The sensitivity and specificity of compared methods on k-mismatch reads which can be mapped uniquely with k-mismatch. b) shows similar statistics to a) by mapping k-mismatch reads which have alternate unique alignment of ≤ k-mismatch. 119

5.4. Sensitivity and accuracy for aligning simulated reads from ART. Cumulative counts of correct and wrong alignments from high to low mapping quality for simulated Illumina-like (A) 75 bp and (B) 100 bp (C) 250bp datasets. 124

5.5. Specificity (A-B) / F-measure (C-D) of alignments using simulated Pure-Indel reads of various indel-lengths each with 1 million 75 bp reads. A/C are delete datasets. B/D are insert datasets. 127

5.6. Mappings of concordant and discordant datasets using paired-end mapping mode of various methods. 130

5.7. Concordant and discordant alignments using real reads from Illumina. Cumulative number of concordant and discordant alignments from high to low mapping quality for real Illumina (A) 76 bp (B) 101 bp (C) 150bp datasets. 134

5.8. Example of recovering a delete in a reference from a read..... 143

6.1. The counts of (a) correct alignments and (b) wrong alignments from the compared methods on 76 bp and 100 bp BEERS-simulated datasets. 151

6.2. Chromosome-1 reads were mapped to a chromosome-1-deficit hg19. False positive rate was calculated by the number of simulated reads that were mapped to the modified hg19, divided by the total number of reads. 155

6.3. The counts of correct and wrong alignments for simulated RNA-seq 76bp and 100bp of 2 million reads each stratified by edit-distances of 0 to 3. 157

6.4. The cumulative counts, over edit distances of 0-3, of all non-ambiguous mappings from the various spliced mappers on 2 million real reads taken from Sample 11T of ERP00196. 157

6.5. The cumulative counts, over edit distances of 0-3, of all non-ambiguous spliced mappings from the various spliced mappers on 2 million real reads taken from Sample 11T of ERP00196. 158

6.6. A schematic flowchart showing how input RNA-seq reads is aligned using the 3-phased methodology of BatRNA..... 161

6.7. Possible alignments on RNA-seq read from BatAlign. 163

6.8: A flowchart showing how the splice alignment algorithm in BatRNA performs splice alignment. 165

6.9. Schematic sketches of some possible scenarios that can happen in BatRNA splice algorithm. a) Adjacent non-overlapping seeds do not span across exon-exon junctions. b) Anchored seed is near to a exon-exon junction and next immediate 18-mer is used to seed the alignment. c) After successfully pairing of seeds within spanning distance of 20 kbp, alignments are extended towards each other to recover the splice junction on the reference genome. d) New seed is selected for the continual extension of a current partially anchored alignment. 166

6.10. Possible short overhangs being recovered with local alignment by using preceding prediction as a guide in an unsupervised manner. 167

A1.1. Percent of partially methylated Cs with regards to CpG islands. 192

A1.2. Genomic profile of partially methylated Cs along chromosome 1. 193

A1.3. Proportion of partially methylated CpGs in different chromosomes 194

A1.4.Definition of gene model used in our results. 195

Chapter 1

Introduction

Earth has been brimming with life for as long as we can remember. With ourselves being intellectually revolutionized agents, it is imminent for us to question and understand the ravels of life. As known to some as The Code of Life, DNA has been understood to be the determinant material that guides the operations and propagation of organisms on a molecular basis. Charles Darwin and Gregor Mendel first studied the rules on how life propagates between 1856 and 1865.

In 1859, Charles Darwin published his theory of evolution with inspiring evidences in a book titled “On the Origin of Species” [1]. He showed that all species of life have descended from common ancestors and rejected competing explanations of species being transmuted from one and another. This scientific theory proposed a branching pattern of evolution for different species resulted from a process which he has coined as Natural Selection. While the theory of evolution was centered on the communal pressure for existence in an ecosystem, Gregor Mendel focused on the passing of phenotypes to the next generation of the same species. Mendel’s experiments on plant hybridization led to the understandings of dominant and recessive phenotypes propagation in a species under the form of inheritable materials [2], which we now

call it as genes. It was not until the 1940s that Darwin's theory of Natural Selection and Mendel's Law of Inheritance were combined and gave rise to evolutionary biology.

The genes which gave phenotypic traits to an organism are made up of DNA. It was first isolated as a weak acid and identified as the genetic material in 1944 by Oswald Avery, Colin MacLeod and Maclyn McCarty [3]. Within the next decade, science celebrated ground-breaking discovery on the structure of DNA with the publication of three papers by Nature: one from James Watson and Francis Crick of Cambridge University that proposed the double helix sugar-phosphate backbone structure of the DNA [4], and two accompanying papers from Franklin Rosalind [5] and Maurice Wilkins [6] of King's College, London, who used X-ray diffraction images to support the helix hypothesis.

After the DNA double helix structure was discovered, scientists moved on to investigate the contents of what our genetic materials hold, namely, the sequence of the nucleotides which form the genes. DNA is sequenced for the first time in early 1970s and methods by Fred Sanger [7], Walter Gilbert, and Allan Maxam [8] were published independently in 1977. Sanger sequencing was the first established method to sequence the long stretches of DNA and has partially been used to produce the first draft of the human genome, known as the Human Genome Project (HGP), starting from 1990 and to its completion in 2003 with the working draft of the human genome [9].

Due to the influx of funding and talent into the field of genomics, huge advances in sequencing technologies were achieved and this gave rise to a new generation of sequencing technologies which we call second-generation sequencing (SGS) technologies.

With SGS technologies at the disposal of scientists, landmark projects were launched. After the HGP, scientists went on to sequence the genomic sequences of a wide variety of species from various clades such as mammal, nematode and insect. Some examples include humans from different ethnic groups and different strains of influenza virus due to antigenic shift. Alongside with DNA sequencing projects, studies were also started on functional genetic elements such as RNA and proteins. As such, Human Encyclopedia of DNA Elements (ENCODE) was started in 2003 to build a comprehensive parts list of functional elements of the human genome, including elements that act at the protein and RNA levels, and regulatory elements that control cells and circumstances in which a gene is active [10]. As of 2012, ENCODE has claimed to have assigned biochemical functions for 80% of the human genome [11].

1.1 History of DNA Sequencing

1.1.1 First-Generation sequencing

DNA sequencing is the process of determining the precise order of nucleotides within a DNA molecule. The first whole DNA genomic sequence was obtained in 1977 from the entire genome of bacteriophage Φ–X174 using chain-termination methods [12]. This sequencing method was developed in 1975 by Sanger [13] and followed independently by Maxam and Gilbert in 1977. The Maxam-Gilbert method was more laborious and hazardous to handle with as the chemicals used in the sequencing procedures were more radioactive than Sanger's method. Due to these reasons, Sanger sequencing became dominant and was ubiquitous in first-generation sequencing. Even till now, Sanger sequencing is still practiced due to the longer reads, ~800 bases in average, that it generates as compared to ~100 bases reads from Illumina GA IIx machines [14]. Sample preparation for Sanger sequencing starts by generating randomly-sized fragments of the DNA from the same starting location. The ends of these different-sized fragments are then labeled with one of the four

fluorescent/radioactive dyes which substitutes for each of the four nucleotides of the DNA – adenine, cytosine, guanine and thymine. Next, the dye-ended fragments are placed into a 2D gel and will be sorted in order of their lengths via electrophoresis across the gel. Lastly, the sequence of the DNA sample is determined from the last base of the fragments as depicted by the order of their relative positions in the 2D gel. Although, this method can be fully automated to sequence long stretches of DNA, it still took about 13 years and three billion dollars to produce the first working draft of the human genome for the HGP. The main drawback of Sanger sequencing is that the throughput of each run is too low to perform in-depth studies on the complex dynamics of the human genome.

1.1.2 Second-Generation sequencing

This wave of technologies aims to offer numerous advantages over Sanger sequencing in the form of (1) shorter runtime (sequencing speed is increased); (2) higher throughput (more bases sequenced within short period of time); (3) cheaper sequencing costs (less reagents needed for the experiments) and (4) higher accuracy (enabled discovery of rare-occurring variants).

The second generation of sequencing (SGS) was first described by two publications in 2005 [15, 16]. The initial impacts that polony sequencing had brought about was the lower sequencing costs and the potential for scientists to capture the complex dynamics of the genome at high resolutions. A year later, two Cambridge scientists developed the Solexa 1G sequencer and it was able to produce a throughput of 1 giga-base in a single experimental run for the first time in history using reversible terminator chemistry [17]. In the same year of 2006, Agencourt was purchased by Applied Biosystems which introduced SOLiD sequencing [18] which too had the ability to sequence a genome as complex as the human genome. Other NGS technologies include Roche 454 pyrosequencing [19], IonTorrent semiconductor sequencing [20], DNA nanoball sequencing [21] and Heliscope single molecule

sequencing [22]. With most SGS technologies, strands of identical DNA are anchored to a fixed location to be read by a sequential series of label-scan-wash cycles. Each of this cycle will yield a read-base and is no longer continued until the series of label-scan-wash cycles exceeds a threshold of quality. Due to the extremely high density of DNA that can be packed into a single sequencing template platform, the throughput from such technologies far exceeds of those of Sanger sequencing [14]. This is the most outstanding advantage of SGS over first-generation sequencing technology and it has directly made quantification of transcripts, genome-wide methylation profiling and many other studies possible.

More cost-effective methods were also developed to compromise between the competing goals of genome-wide coverage and cost-effective targeted-coverage; such an example will be "exome sequencing" whereby ~1% of the human protein-coding genome is sequenced [23, 24].

1.1.3 Third-Generation sequencing

Sanger sequencing and SGS technologies have by far revolutionized the field of genomics. However, there are still aspects of genome biology that are beyond the capabilities of SGS technologies. The main shortcomings of SGS technologies are the long runtime (a few days), short read-lengths and potentially high sequence bias and/or errors. Due to the large number of synchronized label-scan-wash cycles required to generate a read, the time needed to generate viable reads of long read-lengths is long. It is also due to the fact that the label-scan-wash cycles have to be synchronized in between cycles, meaning that the yield of each step of the long series of cycles will be <100%; the cycles sometimes get "dephased" and out-of-synchronization which produced erroneous reads. As such, this causes an increase in sequencing errors as the read extends during sequencing. The average read lengths generated by SGS technologies are generally less than the lengths achieved by Sanger sequencing. Another source of read errors in the form of sequencing bias come from

PCR amplification [25] which is involved as an intermediate step in SGS technologies. A new generation of single-molecule sequencing (SMS) technology aims to resolve these shortcomings of SGS technologies.

Unlike sequencing-by-synthesis (SBS) technologies in SGS, SMS interrogate single molecules of DNA using SBS too but in an asynchrony manner. In this manner, tens of thousands of reads can be sequenced within hours as compared to days as experienced with SGS technologies. In addition, since molecules are interrogated individually, there is no need to amplify the DNA sample prior to sequencing by SGS technologies; this removes any amplification bias or defects which may be introduced by PCR. The nucleotides used in SGS technologies are usually 'color-coded' with a dye and this makes them different from natural-occurring nucleotides which make up the DNA. This chemical bias is further removed in SMS technologies and the reagent used to replace the dyed nucleotides is none other than DNA polymerase which is responsible for DNA duplication whenever cells divide.

The main idea in SMS comes from the tangible measurements that can be measured when the new DNA fragment is synthesized upon the template fragment. The measurements can then be interpreted as an ordered sequence of nucleotides. Some technologies of this new generation are based on but not limited to the use of nanopores, tunneling currents during DNA synthesis, mass spectrometry, micro-fluidic chips and electron microscopes.

1.2 Motivation

1.2.1 Looking at the DNA with an intent

Little was known about the functions of DNA when it was identified as the genetic materials of organisms in 1944. It was also unclear on how DNA polymorphisms played a part in the molecular operations of the cells in an organism. It was not until 1956 when Vernon Ingram successfully associated a single amino acid substitution

with Sickle Cell Disease [26]. Since then, scientists have moved forward with the intention to better understand of diseases caused by genetic variations and discover new ways to treat them [27]. Other common genetic diseases include Cystic Fibrosis, Glucose-6 Phosphate Dehydrogenase deficiency and Color Blindness.

Genetic diseases were first thought to be a direct causal effect of mutations in the DNA. This thought could not be more wrong. The products of such processes can also affect the level of transcription of DNA to RNA and translation of RNA to proteins. The study on the causal effects of the DNA and its products other than the changes in the underlying sequence is now termed as epigenetics [28]. The two epigenetic modifications to the genome are histone modifications and DNA methylation [29].

Regardless of genomic or epigenetic factors, the important challenge is to understand the mechanisms that control the expression of each gene in a genome. By learning about these processes, we can uncover more ways to treat, cure or even prevent such adverse phenotypes in a diseased genome.

1.3 General workflow on sequencing reads

Due to the limitations of technology, the sequenced reads is almost always shorter than the reference genomes, such as that of the human and mouse. As such, from the raw sequenced reads of a sample to downstream analysis in the dry-lab, the common step in most processing pipeline is to map the sequenced reads onto a reference genome as depicted by Figure 1.1. In the field of genomics, the two most front-lined computational tasks are 1) mapping reads back onto a reference genome and 2) de novo or guided assembly of the genome with sequenced reads to produce a reference for reads to be mapped on. These tasks are generally the most computationally intensive tasks in a pipeline and the problem is made worse by the voluminous amount of data (~600 Gb in a single run).

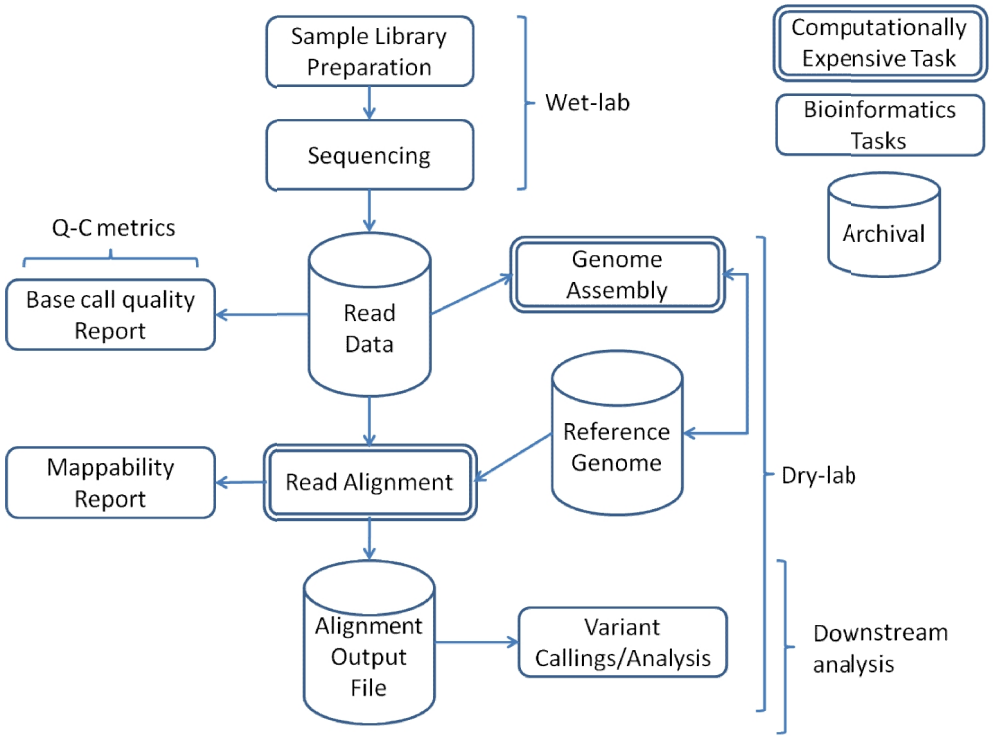


Figure 1.1. General workflow on sequencing reads

1.4 The mapping challenge

One can see the problem of mapping a read onto a reference genome as a computational problem of string matching. The aim is to find the original location from where the read was sequenced from the genome. The alignment of SGS reads poses some challenges in the forms of different error profiles of sequenced reads from different sequencing technologies, short read lengths (reads from SGS can be ~36 bases long), large reference length which the reads need to be mapped on and the voluminous data that are generated from SGS machines [30]. Since mapping the reads is a prelude to many downstream analyses such as and not restricted to variant-callings, quantification of rare transcripts and annotation of epigenetic factors on the DNA, it is important to map the sequenced reads with high sensitivity, specificity and speed.

Many scientists and classic software have tackled this problem. For instance, BLAST [31] (~50k citation count) and BLAT [32] have shown the demand and impact of

bioinformatics in understanding genomic data. However, classic software cannot handle SGS well as they are not designed for SGS reads in mind. Therefore, new methods have to be developed to align SGS reads. This thesis aims to reports on the new algorithms which we have developed to align SGS reads with high sensitivity, specificity and speed.

1.5 Contribution of thesis

The first contribution of this thesis is the development of BatMeth which is a fast and efficient algorithm for the alignment of bisulfite-treated DNA sequencing reads back onto a genome allowing mismatches. BatMeth is based an exact algorithm, namely BatMis, for the alignment of reads onto a genome allowing mismatches. By designing the appropriate heuristics, BatMeth has shown to be an improved aligner in our benchmarks. In addition, it was also shown to have less bias when mapping bisulfite treated samples across a wide range of bisulfite conversion rates on both Illumina base-space and SOLiD color space reads. Dr Guoliang Li also used BatMeth to predict potential imprinting genes and his results are also included in this thesis.

The second contribution is the development of BatAlign for the accurate alignments DNA sequencing reads allowing both mismatches and indels. The algorithm of BatAlign was designed to discriminate between polymorphisms and sequencing errors with high precision. The initial method for aligning short reads (~100 bases) was also extended to handle longer reads of (150-250 bases). Dr Chandana Tennakoon developed underlying data structures used to implement the algorithm for space-time efficiency. BatAlign was benchmarked on a wide class of simulated and real reads and have shown to be more accurate than other popular aligners in terms of mapping accuracy and variant callings on published PCR-validated SNV/indel mutation in gastric cancer.

The third contribution is the development of BatRNA for the alignment of reads allowing mismatches, indels and large intronic gaps to be in a single read. The algorithm of BatRNA was designed to resolve large gaps due to introns with high accuracy and speed while maintaining the capability to avoid mapping to pseudogene areas and resolve short exonic overhangs due to spliced junctions near the ends of reads. The algorithm was extended and modified from BatAlign to make better use of RNA-specific features for more accurate mappings of RNA-seq reads. Benchmarks showed that BatRNA gives sensitive and accurate mappings in a mixed sample of exonic and spliced read across varying sequenced read lengths.

In summary, we have developed three novel alignment algorithms on improved data structures for the efficient and accurate mappings of sequencing reads from various genomic contexts. In addition, we also include results on the biological insights which Dr Guoliang Li and I have uncovered on the prediction of imprinting genes by using BatMeth.

1.6 Organization of the thesis

The remaining contents of the thesis are organized as follows. Chapter 2 presents a preliminary of biological background and survey of SGS technologies required for the proper understandings of the thesis. Chapter 3 will present the survey of bisulfite-treated DNA-seq aligners, gapped DNA-seq aligners and spliced RNA-seq aligners in their respective subsections. Chapter 4-6 will present our algorithms for an improved alignment of bisulfite-treated DNA-seq reads (I am the first-author of this paper published in *Genome Biology*), gapped DNA-seq reads and spliced RNA-seq reads respectively. Chapter 4 also includes research findings on the prediction of imprinting genes using BatMeth. Chapter 7, the last chapter, will conclude the thesis with a summary of all the presented work and a brief discussion on the possible future developments which still can be carried out on alignment algorithms.

Chapter 2

Basic Biology and Sequencing Technologies

2.1 Basic Biology

In this chapter, we present the background knowledge on molecular biology and describe some of the SGS technologies that are widely used today. We also describe the types of reads which can be obtained from the wet-lab which will be mapped onto the reference genome by aligners.

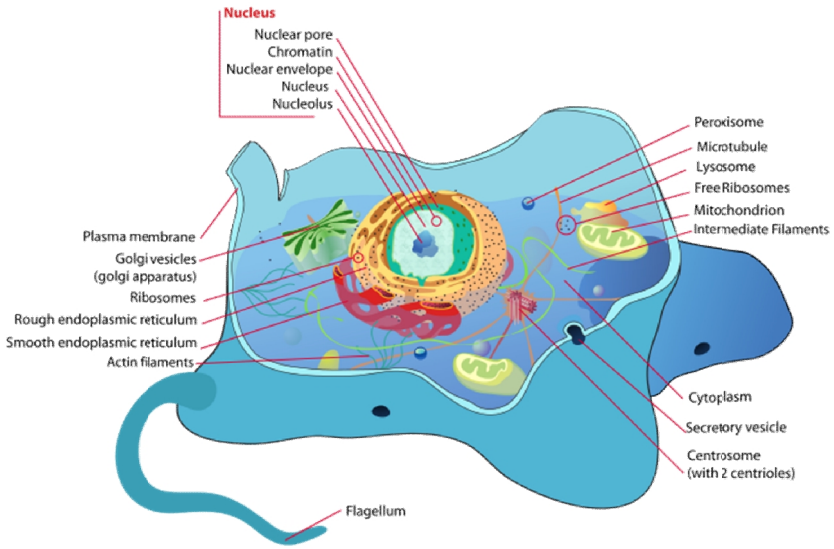


Figure 2.1. Schematic diagram of a typical animal cell. Source: [33]

Cells are the building blocks of organisms and complex organisms such as an adult human will contain trillions of cells. Cells are also referred to be the building blocks of life as they are essential to maintain the bodily functions of the organisms which they make up. On a cellular level, a cell has a cell membrane, nucleus, golgi apparatus, cytoplasm and mitochondrion as drawn in Figure 2.1. On a macro-molecular scale, it typically contains carbohydrates, amino acids, lipids and nucleic acids. With the advantages made in the protocols for experiments, studies can be carried out to study the activities of the various macro-molecules in the cell. For instance, genome-wide analysis of gene expression can be measured using high throughput methods such as RNA-seq data [34], spliced alignment tools [35-42] and transcripts-isoforms quantification tools [43-47]. Repetitive regions of the genome can be hard to be studied with SGS data as alignment tools will not be able to report the putative original location of the read in the genome with high confidence of uniqueness. These repetitive regions can include the telomeres and centromeres of the human genome and florescence immuno-staining techniques are used to study these genomic regions [48].

To first study the genomes using high throughput data using SGS, scientists have to use sequencing machines to 'read' out the genomic sequences of the prepared sample. Many SGS technologies are now widely used by scientists worldwide and some of these technologies come from Illumina, Life Technologies, Roche and Ion Torrent. Depending on the type of samples, method of preparation on the experiments and the sequencing technologies being used, a wide range of analysis can be carried out. Figure 2.2 shows some of the analysis which can be carried out on sequencing data. This is the important initial phase in identifying and understanding the dynamics of macromolecules in a cell.

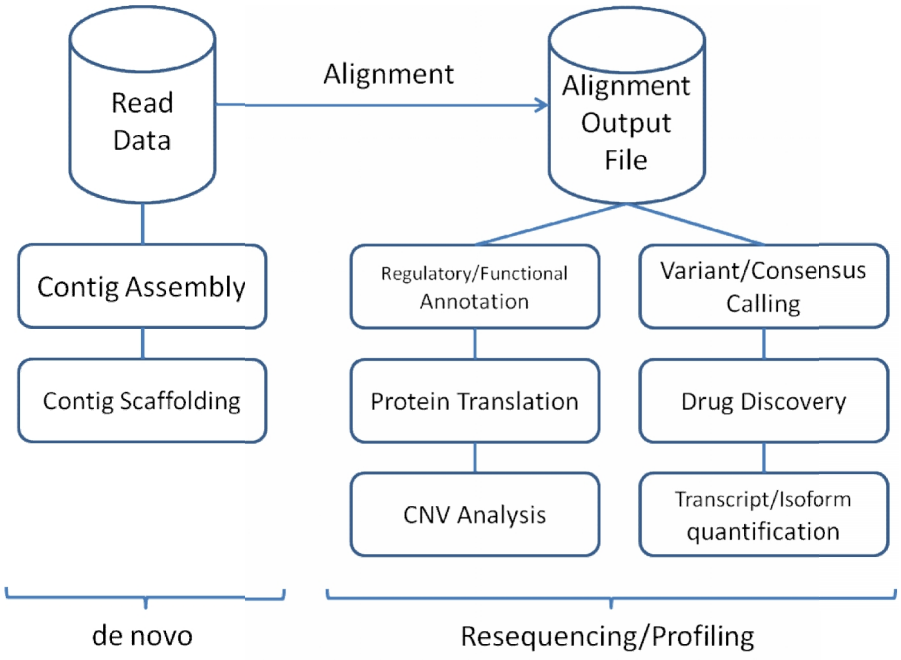


Figure 2.2. Two main types of genomic tasks and their respective downstream analysis. De novo tasks involve the manipulation of read data without a reference. Profiling tasks use the alignment of the read on a reference for analysis.

2.2 Central Dogma of Molecular Biology

The Central Dogma of Molecular Biology is one of the main principles in understanding molecular biology. Although there are exceptions pointing against it, it is still widely accepted that the transfer of genetic information is from the gene sequences of the DNA to proteins which carries out various cellular functions. Figure 2.3 depicts the general passing of sequential information between genetic materials as stated by the general cases in the Central Dogma of Molecular Biology as formulated in 1970 [49].

The central dogma states that all genetic information is encoded in the DNA molecules. This genetic information can be visualized as linear sequences of nucleotides in the cells. When cells grow and divide, genetic information is transmitted from the parent cell to the daughter cells by replication to form a duplicate of the DNA molecule during the synthesis phase of the cell-cycle. During the synthesis of messenger ribonucleic acid (mRNA), part of the original DNA sequence acts as a template for the mRNA sequence to be synthesized on. This process of mRNA synthesis is known as transcription.

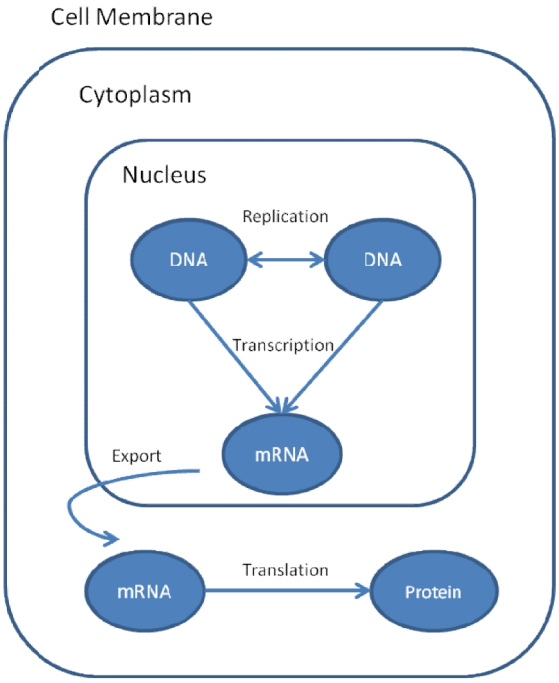


Figure 2.3. The general cases of the central dogma of molecular biology for eukaryotic cells

For eukaryotic cells, the mRNA molecules are then transported out of the nucleus, into the cytoplasm, where consecutive triplets of nucleotides are read as codons by protein complexes known as ribosome. In the ribosome-mRNA complex, aminoacylated transfer RNAs (tRNA) are recruited and used to link the amino acids according to the mRNA sequence to form the protein polypeptide chains. The process of reading mRNA and form the protein complex from it is known as translation. This

protein polypeptide chains will undergo post-translation modifications to a stable folded 3D structure which attributes to its functions and will then drive various cellular functions in an organism.

From the central dogma, DNA-DNA replication, DNA-RNA transcription and RNA-Protein translation are the main processes for understanding the transfer of genetic information. In addition, we can also see that there are several ways in which cells can be regulated. For instance, the amount of mRNA transcribed from the DNA, known as gene expression level, will be translated into varying concentrations of proteins which, in turn, will up/down-regulate transcription, affecting the gene expression levels and subsequently their corresponding protein concentration levels in the form of a feedback loop. Post-modifications to the proteins such as phosphorylation and acylation can also affect the functional properties of proteins. By mutating the DNA sequences, changing the levels of mRNA and protein abundances can lead to the onset of diseases such as Sickle-cell anemia and Cystic Fibrosis.

In the following subsections, we will describe replication, transcription and translation in detail.

2.2.1 DNA-DNA Replication

DNA comprises of nucleotides and each of them contains a deoxyribose sugar, a phosphate and a nucleobase. It is usually double-stranded and both strands are bonded together to form a double-helix structure. The deoxyribose sugar and phosphate will form the backbone of the double-helix structure and the nucleobase (Adenine, Cytosine, Guanine and Thymine; ACGT) will be forming hydrogen bonds with another nucleobase on the reverse-complementary strand of the DNA. The base pair makeup of the DNA was also hinted by Chargaff's 1950 experiment and provides a general but not exclusive rule that adenine and cytosine pairs up with thymine and guanine respectively on opposing strands of the DNA [50].

DNA replication is the process whereby a new copy of the DNA molecule is replicated from one original template DNA molecule. This is possible as DNA is composed of two strands and each strand of the original DNA molecule serves as a template for the replication of the new reverse-complementary strand. This results in two copies of double-stranded DNA molecules with each of them consisting of an 'old' template strand and a 'new' replicated strand; this is why DNA is semi-conservatively replicated and is demonstrated to be so in 1958 by Meselson-Stahl experiment [51]. Figure 2.4 shows three postulated methods of replication before Meselson-Stahl experiment.

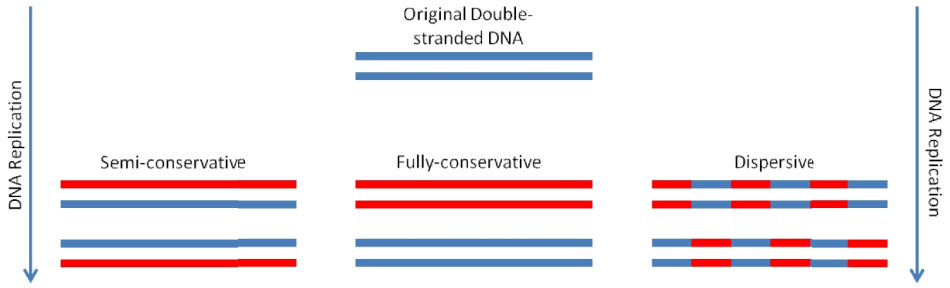


Figure 2.4. Three postulated methods for DNA replication prior to Meselson-Stahl experiment

As DNA replicates prior to mitosis, it must involve initiation of replication, elongation of DNA fragments and termination of synthesis. For a cell to divide, it must replicate its DNA first and this process can initialize at various sites known as replication origins. Initiator proteins will target A-T rich regions of the DNA and recruit other proteins, unzips the double-stranded DNA and prepares it for replication. As the new DNA is being synthesized and elongated on the old template DNA, the helicases keep breaking the hydrogen bonds between the two DNA strands to unwind more regions of the DNA for elongation.

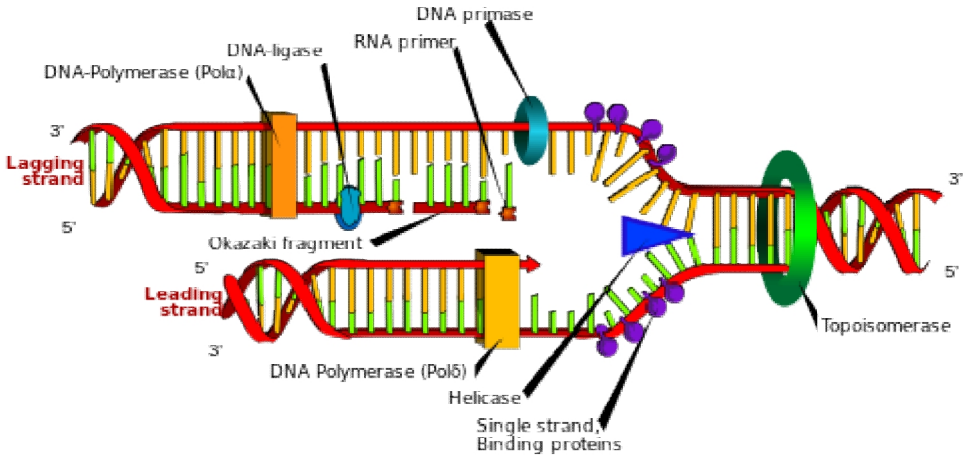


Figure 2.5. Schematic diagram of DNA replication at a replication fork. Source: [52]

As DNA is always synthesized from the 5' to 3' direction, there will be one strand of the DNA that will be in the 'wrong' direction and this is called the lagging strand in DNA replication; the other strand will be the leading strand. The DNA polymerase will start to add complementary bases to the template strand after a small RNA fragment attaches itself to the site of replication origin to prime the elongation process. With respect to the leading strand, the DNA polymerase will move in the same direction of the helicase. However, for the lagging strand, the DNA polymerase can only add bases away from the direction of the helicase and results in replicating the DNA in disjoint but adjacent fragments called Okazaki fragments. Figure 2.5 depicts the process of DNA replication at one instance of the DNA replication fork. Since there are multiple points of replication origins, termination of elongation happens when a replication forks meet and this can occur at many points in a single chromosome.

2.2.2 DNA-RNA Transcription

RNA comprises of nucleotides and each of them contains a ribose sugar, a phosphate and a nucleobase. It is usually single-stranded. However, RNA can form intra-strand double helix structure as in the case of the double-stranded DNA by complementary

base-pairing with hydrogen bonds too; as in the case of tRNAs. The ribose sugar and phosphate will form the backbone of the structure for RNA and the nucleobase (Adenine, Cytosine, Guanine and Uracil; ACGU). Three main types of RNA are transcribed from a region of the DNA as a template and they are messenger-RNA (mRNA), transfer-RNA (tRNA) and ribosomal RNA (rRNA) [53]. mRNA is a near-duplicate of a region of the template DNA which will code for a protein sequence. tRNA is a short sequence of ~80 nucleotides that transfers amino acid to the site of protein synthesis. rRNA is responsible to link the amino acids from the tRNA to grow the polypeptide chain to form a protein.

The first step in achieving molecular function is to transcribe a gene region of the DNA into mRNA in a process called transcription. The mRNA will act as a blueprint for a protein to be translated from it. In eukaryotes, the process starts by having the RNA polymerase and other transcription factor(s) to bind to a core promoter sequence in the DNA which is usually within a hundred bases upstream from the transcription start site (TSS) of a gene. In prokaryotes, protein factors bind to the RNA polymerase which affects the binding of the polymerase to the DNA. The RNA polymerase will next start to move along the promoter region and towards the TSS. Once the RNA polymerase enters the gene region, it will use base pairing complementarily with the DNA template (non-coding strand) to create an RNA copy. Different transcription levels of genes are usually resulted from multiple rounds of transcription or multiple RNA polymerases on a single DNA template. Elongation of the RNA terminates when the newly synthesized RNA segment contains a GC rich and subsequent Us rich sequence or the 'Rho' protein destabilize the interaction between the template DNA and the mRNA. These two mechanisms cause the template DNA and RNA polymerase to disengage from one another and the synthesis of any new RNA segments to cease.

2.2.2.1 Genes and Splicing

A gene is a biological unit of hereditary material. It can also refer to subsequences of DNA and it provides the blueprints for the RNA polymerase to synthesize proteins from it. In eukaryotic cells, the RNA that is transcribed from the DNA will undergo more post-transcription modifications [54]. At the 5' end of the pre-mRNA, a single G will have its 5' end attached to it, whereas at the 3' end, a poly-A tail will be added. This capping on both ends of the untranslated regions (UTRs) of the pre-mRNA fragment will result in 3' endings and protect the fragment from being cleaved at the 5' end by exonucleases. Figure 2.6 shows the differences in the markup of genomic features between pre-mRNA and mRNA.

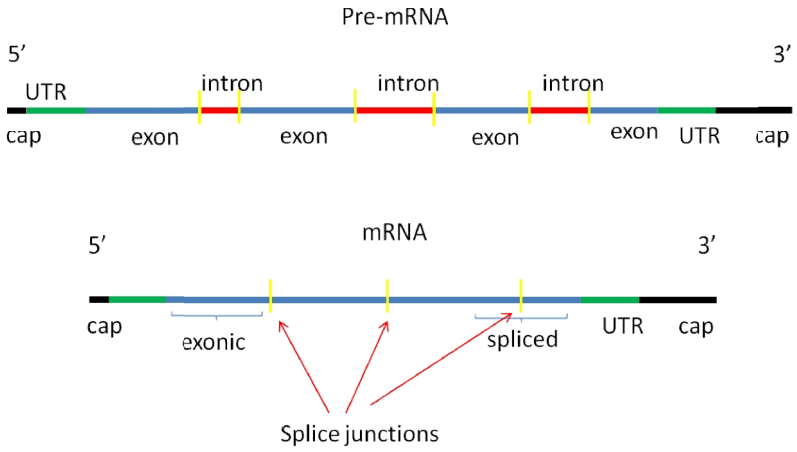


Figure 2.6. Illustration of introns and exons in pre-mRNA and the maturation of mRNA by splicing.

A pre-mRNA fragment contains adjacent sequences of nucleotides that will either be translated to protein or not; namely, exons and introns respectively [55]. In eukaryotic cells, the pre-mRNA fragment will be matured by cleaving the introns away from the original pre-mRNA fragment which will leave the exons behind. This event is known as splicing and the genomic locations where introns are being cleaved at are called splice sites. From the literature, we can observe that these splice sites tends to be

conserved with canonical signals (GT-AG, donor-acceptor) at rate of >98% on splicing events in humans [56].

Splice sites can sometimes reside completely in exonic or intronic regions. In other words, splicing can sometimes happen or not happen at a splice site and this is known as alternate splicing [57]. This gives the possibility of a single gene to code for several proteins which makes it more efficient as a single gene region may have more than one functional product. In fact, the human DNA is so efficient in this sense that ~95% of multi-exons gene regions can express more than one functional product [58].

Currently, SGS technologies produce RNA-seq data from sequencing matured mRNA fragments. As such, the intronic regions are left out from the spliced sequencing read. Before scientists can study the transcription levels of genes, they have to map the RNA-seq reads back to the human DNA reference genome by taking these intronic gaps into account too. The alignment of RNA-seq read proved to be a challenge as seen from the myriad of computational methods developed to solve it. In the following chapter, we will review on the techniques developed for the alignment of RNA-seq reads.

2.2.3 RNA-Protein Translation

Proteins are chains of polypeptide sequences that are made up of some combinations of amino acids. The polypeptide chain folds into a 3-D structure which will define its cellular functions. Generally, proteins are studied at four levels of granularity. At the finest level, the structure of a protein can be studied by the sequence of amino acids which represents it. Next, secondary local structures such as the α-helix and β-pleated sheets are formed when amino acids of the same polypeptide are joined together by hydrogen bonds. Thirdly, tertiary structures are folded into configurations due to the attractive/repulsive forces between secondary local structures. Lastly, quaternary

structures are formed when two or more proteins come together to form a more complex 3-D structure.

Proteins are synthesized from an mRNA sequence by a ribosome complex through a process called translation. Translation starts with the ribosome binding to the 5' end of the mRNA. The ribosome will then decode the mRNA in consecutive non-overlapping frames of 3 bases called a codon. The start codon for translation is "ATG" and serves as an initiation site for translation. While the ribosome traverses across the mRNA, tRNAs carrying specific amino acids with complementary anti-codon sequences to that of the mRNA will have the amino acids chain together into a polypeptide. The chain will terminate when the ribosome faces a stop codon (UAA, UAG or UGA) and this recruit a release factor protein to disassemble the entire ribosome-mRNA complex. The synthesized chains of polypeptide will then give itself the molecular functions with the structure which it folds itself into or by integrating with other secondary or tertiary structures as mentioned before.

2.3 Next Generation Sequencing Technologies

Chapter 1 gave a brief history of sequencing technologies and the motivation to uncover insights that genomic sequences contain. In this section, we will briefly describe the computational challenges that these technologies bring about and the main ideas behind some sequencing technologies which the thesis is focused on. Currently, sequencing technologies support sequencing materials from a wide range of starting materials, such as genomic DNA, PCR products, bacterial artificial genome (BAC) and complementary DNA. Without loss of generality, we will describe the sequencing of genomic DNA in the following subsections by various technologies.

2.3.1 Roche/454 Sequencing

454 sequencing is arguably the first high throughput sequencing technology that is available to the market. This technology eradicates the need for DNA sample fragments to be cloned in bacterial hosts. By removing bacterial clonal copies of the DNA, we also remove any amplification bias which may be introduced by the hosts into the DNA sample. Instead of in vivo cloning of the DNA sample using bacterial hosts, the amplification process is replaced by a more efficient in vitro DNA amplification method called emulsion PCR [59]. In emulsion PCR, fragmented DNA will attach to a streptavidin bead covered with adapter probes with bases complementary to that of the fragmented DNA. The ideal scenario will be one fragment to one bead and then this bead will be suspended in an emulsion so that individual beads can be trapped in amplification micro-reactors. The whole emulsion of beads will be amplified in parallel to create millions of clonal copies of each DNA fragment on each bead. After amplification, the emulsion is removed from the mixture of beads as like removing the oil from an oil-and-water mixture and the beads are loaded onto a picotiter plate prior to being sequenced by a machine [16].

The loaded picotiter plate will have hundreds of thousands of sequencing processes to be carried out in parallel, obtaining massive increase in throughput as compared to Sanger sequencing [60]. As sequencing takes place, a nucleotide is added one by one to the immobilized template DNA on the bead. Whenever a complementary nucleotide is added to the template DNA, a chemiluminescent enzyme present in the reaction mix will produce a detectable light by releasing inorganic pyrophosphate [19, 61]. This is also why 454 sequencing is also known as pyrosequencing and SBS.

Since the produced light signal is directly proportional to the number of bases incorporated onto the template DNA in one sequencing cycle, pyro-sequenced reads will often have lengths of homo-polymeric nucleobases wrongly estimated.

2.3.2 Ion Torrent Sequencing

Ion Torrent invented the first semiconductor sequencing chip that is commercially available for the market. Similar to 454 sequencing, Ion Torrent clonally amplify DNA fragments by using emulsion PCR. After which, the beads with the amplified DNA materials will each sit inside a micro-well. The main difference in Ion Torrent sequencing from 454 sequencing is that the chip itself sequences the read [62].

The scanning of the DNA fragment starts by flooding the bead-loaded well with one nucleotide after another sequentially. When the DNA fragment is extended by the incorporation of nucleotides, it releases hydrogen ions into the well and this changes the pH of the solution in the well changes. This chemical change of pH in the solution can directly be recorded by a sensor plate at the bottom of the well into voltage readings [63]. Since the chip directly detects the nucleotides which are being synthesis on the DNA template fragments on the bead, no external optical instruments are needed. This is the fundamental difference of Ion Torrent sequencing from 454 sequencing.

Although Ion Torrent sequencing uses a different methodology from Roche-454 to sequence genomic materials, its sequenced read will also have lengths of homo-polymeric nucleobases wrongly estimated [64]. This is due to the produced voltage being directly proportional to the number of bases incorporated onto the template DNA in one sequencing cycle.

2.3.3 Illumina/Solexa Sequencing

DNA molecules are first fragmented into varying length-sizes through the use of a nebulizer through a process called sonication [65] or nebulization [66]. The subset of these randomly sized DNA fragments of similar length-size is then selected to be sequenced. Illumina uses ‘bridge’ amplification reaction that occurs on the surface of the flow cell to sequence a DNA fragment [67]. The surface of the flow cell is coated

with single stranded oligonucleotides as complementary probes that correspond to the priming adapters ligated to both ends of the DNA fragment. These single stranded oligonucleotides are bounded to the surface of the flow cell exposed to the reagents for polymerase-based extension. Priming occurs at the free end of a ligated fragments and 'bridges' to a complementary oligonucleotide on the surface.

Repeated denaturation and extension result in localized amplification of single molecules in millions of unique locations across the flow cell. This process is referred to as "cluster station"; an automated flow cell processor. Figure 2.7 shows the process of bridge amplification of DNA fragments prior to obtaining clusters of amplified DNA materials [68]. The flow cell, with millions of clusters, is then loaded into the Solexa sequencer for cycles of extension and imaging. The first cycle of sequencing consists first of the incorporation of a single fluorescent nucleotide and followed by laser imaging of the entire flow cell. These images represent the respective base being synthesized at each individual location of the flow cell. Any laser signals above background will identify the physical location of a cluster and the fluorescent emission identifies which of the four bases was incorporated at that position. The cycle is then repeated, one base at a time, generating images each representing a single base extension at a cluster.

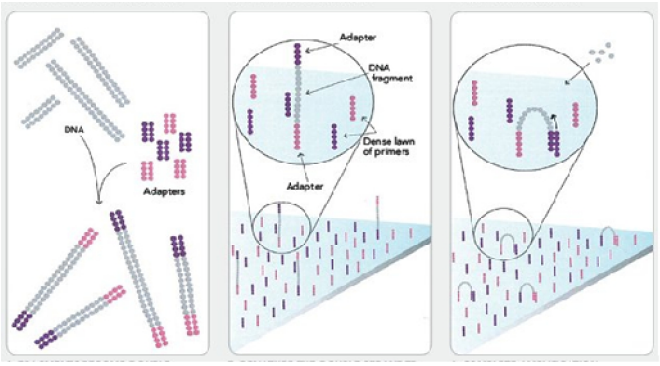


Figure 2.7. Schematic diagram of bridge amplification forming cluster stations. Source: [68]

The result of sequencing a DNA fragment with Solexa will be a string of ATCG which is the read representation of a cluster on the flow cell.

2.3.4 ABI/SOLiD Sequencing

Similar to 454 and Ion Torrent sequencing, SOLiD also uses emulsion PCR which generates 'bead' clones which each contains a single nucleic acid species. Each bead is then attached to the surface of a flow cells via 3' modifications to the DNA strands. At this point, we have a flow cell with millions of beads, each of a single genomic DNA species, with distinct adaptors on either end being monitored simultaneously via sequential digital imaging.

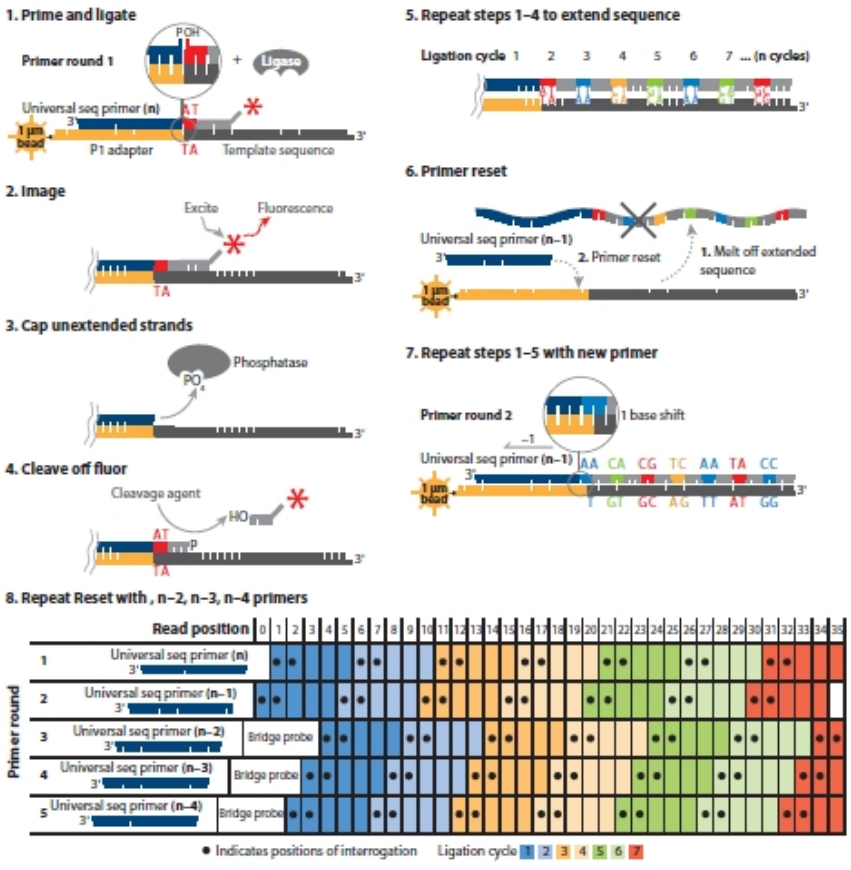


Figure 2.8. Workflow of ligase-mediated sequencing approach from ABi SOLiD. Source: [68]

Now, the actual base interrogation is no longer done by the polymerase-driven incorporation of labeled dideoxy terminators but rather by a mixture of labeled oligonucleotides and queries the input strand with ligase [15]. The technology is also strange such that each oligonucleotide has degenerated positions at base positions from 3 to 5, and one of the 16 specific dinucleotides at positions 1-2 from the 3' end. Base positions from 6 till the 5' end of the oligonucleotide are also degenerated and will hold one of the four fluorescent dyes.

Figure 2.8 provides an overview of steps involved in sequencing DNA fragments using SOLiD sequencers [68]. The sequencers initially involve annealing a primer, hybridizing and ligating a mixture of fluorescent oligonucleotides of 8-mers whose 1st and 2nd 3' bases match that of the template. The unextended fragments are then capped with the same mixtures of non-fluorescent probes. Following which, phosphatase treatment is applied to prevent out of phase ligation and detection of specific fluorescent dyes takes place. After imaging, the dyes are removed via a two step chemical cleavage of the three 5' bases, leaving behind a 5-base ligated probe, a 5' phosphate. We repeat these steps, this time, querying the 6th and 7th bases. After ~10 cycles, a 'reset' of primer is initiated. The initial primer and all ligated parts of the template are melted and washed away. A new primer that is N-1 in length takes over and the whole process of sequencing, starting with annealing the primer restarts.

Sequencing by ligation method used in SOLiD sequencers has been reported to have problem sequencing palindromic sequences [69].

2.3.5 Comparison

Having discussed some of the most popular sequencing technologies, we can infer that there is no best sequencer for all types of experiments. The type of sequencer used is largely dependent on the type of data which the experimenters wish to collect and the budget allowed in these sequencing projects. Table 2.1 shows the

specifications of some commercial sequencers which are commonly used today [14, 70].

Table 2.1. Comparison between some commercialized sequencing platforms in the market.

Sequencer models	Illumina HiSeq2000	Ion Torrent 318	Roche 454 GS FLX	ABI SOLiDv4	Sanger 3730xl
Sequencing mechanism	By Synthesis	By Synthesis	Pyrosequencing	By Ligation and dibase coding	Dideoxy chain termination
Cost of machine	$$$	$	$$	$$	$
Cost per Gb	$70	$1,000	$10,000	$130	$2,400
Run Time	11 days	2 hours	24 hours	7 days for SE 14 days for PE	20 mins ~ 3 hours
Read Length	up to 150 bp	~200 bp	700 bp	up to 50 bp	400~900 bp
Throughput	600 Gb	1 Gb	0.7 Gb	120 Gb	1.9~84 Kb
Accuracy	98% (100bp read)	98%	99.9%	99.94%	99.999%
Paired reads	Yes	Yes	Yes	Yes	No
Insert Size	up to 700 bp	up to 250 bp	up to 20 Kbp	up to 10 Kbp	-

2.4 Origins and representations of sequenced data

Chapter 1 has outlined some of the challenges in aligning sequencing reads to a reference genome which this thesis tackles. In this subsection, we will highlight some of the genomic materials which are commonly being sequenced. In addition, we also describe the two main representations of sequenced reads. In doing so, we present an overview on how alignment challenges can arise from various sequencing technologies.

2.4.1 Whole-genome and targeted sequencing

High throughout sequencing technologies have successfully been used to sequence genome-wide dataset for the study of genome in its entirety. Machines from Illumina and Roche have much higher throughput and shorter sequencing times as compared to Sanger sequencing. As such, it is almost always that NGS is picked over Sanger sequencing to sequence whole genomes. Two main types of projects which are performed with NGS are the de novo assemblies of whole genomes and the alignment

of sequencing reads onto assembled reference genomes. In the former type of projects, scientists construct a reference genome from the sequencing reads to allow the study of various disease-causing genomic features such as SNPs, indels, structural variants and epigenetic profiles on it. In the latter, the alignments of the sequenced sample onto a reference genome are used to uncover some of the mentioned disease-causing genomic features which are of interest to the scientists. [71] showed that massively sequenced reads are able to reconstruct mutational signatures on a genome-wide scale in gastric cancer samples.

However, whole-genome sequencing (WGS) is not preferred when the cost of reagents far exceeds the requirements of a study. For instance, it will not be cost-efficient to use GWS to study 10 genomic locations (the human diploid genome has about 6G locations). Thus, targeted sequencing was developed to sequence a specific genomic region of interest with high coverage. With high coverage at targeted genomic regions, mutations can still be studied at the same level of resolution as it is with WGS. However, by doing so, the amount of reagents used is directly proportional to the extent of the study and the time used to sequence the sample is also reduced as compared to Sanger sequencing and WGS. [24] allowed the study of a rare Mendelian disease through targeted sequencing on a small population and the identification of the genes responsible for Millers syndrome. Examples of targeted sequencing are exome sequencing, amplicon sequencing and reduced representation bisulfite sequencing.

2.4.2 RNA-seq – mRNA

RNA-seq is used to create a profile of transcription levels of all genes in a genome called transcriptome [34]. The transcriptome is clinically important in genetic diagnosis as the functional consequences in a cell can be viewed as transcript sequences in the transcriptome. The transcriptome was first profiled from using Sanger sequencing on the DNA fragments that are complementary to mRNA

fragments called Expressed Sequenced Tags (EST) [72]. However, due to the low throughput of Sanger sequencing, lowly transcribed genes are eluded from detection. Since NGS can sequence genomic samples with high coverage and throughput, lowly-expressed transcripts can also be detected and be dynamically quantified directly to the number of mRNA fragments being sequenced. In addition, the cost of RNA-seq is also lower than EST sequencing as the required amount of RNA is less than what is needed for EST sequencing.

2.4.3 Epigenetic sequencing

Epigenetic is the study on the causal effects of the DNA and its products other than the changes in the underlying sequence. The main types of epigenetic studies enabled by high throughput sequencing technologies are Chromatin-Immuno Precipitation sequencing (ChIP-seq) and methylation studies.

ChIP-seq is used to identify the protein-binding sites on the DNA [73]. This is important as it helps scientists to understand how DNA-protein interaction affects gene expression. ChIP-seq was preceded by ChIP-chip which requires a pre-designed microarray and this makes ChIP-chip susceptible to hybridization bias as microarrays come with a fixed number of probes. ChIP-seq lacks this form of bias as sequencing technologies can amplify all ChIP-enriched regions and can be applied to genome-wide discovery of transcription factors, structural proteins and DNA modifications.

DNA methylation is the process of adding a functional methyl group to the cytosine of the DNA [74]. Changes in DNA methylation influences the expression of genes in cells and as differentiated cells matured from embryonic stem cells. The methylation profiles, methylome, of differentiated cells from different tissues are vastly unique to one another. Knowing that sites of methylation in differentiated cells are specific and permanent, these cells are prevented to revert back to their pluri-potent state [75]. The current golden standard to produce a genome-wide methylome of a sample is to do

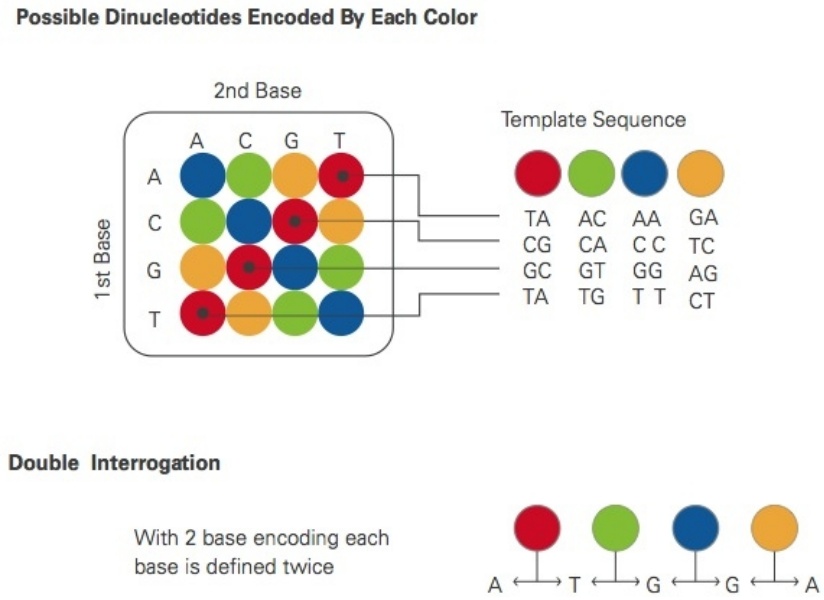
bisulfite treatment on the DNA sample. Bisulfite treatment modifies the unmethylated cytosines to uracils and leaves methylated cytosines unchanged [76]. Upon subsequent PCR amplification of the bisulfite treated sample, uracils will be amplified as thymine on the + strand and adenine on the – strand; unmodified cytosines will be amplified as if DNA-DNA replication is taking place. By using NGS, each possible sites of methylation can be surveyed at high coverage and give a finer granularity of methylation rates at each site.

2.4.4 Base-space and color-space reads

Sequenced reads can be stored in various representations and the two most distinctive representations are base-space and color-space reads.

Base-space reads are stored as a string of characters consisting of "ACGTN". This sequence of characters usually represents and can directly translate to the genomic sequence which was being scanned by the sequencing machines. The character 'N' is to denote a base in the read which the sequencing machine cannot represent with any of the usual "ACGT" nucleotide characters with high confidence. Illumina/Solexa, Roche/454 and Life Technologies/Ion Torrent produces base-space reads.

For SOLiD, a genomic base is interrogated dinucleotide-ly; each color dye represents two adjacent genomic bases. Figure 2.9 shows how the 16 combinations of di-nucleotides are encoded by 4 of the color codes. An example color read would be "T000123100122331100". The first character, T, is the last base of the primer used during sequencing and the numbers represents the transitions of genomic bases during



the dinucleotides interrogation. To obtain the reverse-complement of a color read, we can simply reverse the numerical portion of the read and flip the terminal base from 'T' to 'G'.

Figure 2.9. 2-base encoding scheme used by SOLiD sequencers. Source: [68]

In addition, SNPs are easily identified by two adjacent color mismatches in a read while aligned to a reference genome. However, it is also this strength that got turned into a weakness for SNP-rich and bisulfite-treated data. Each base-letter mismatch will be represented by two adjacent color mismatches instead of one letter mismatch in other technologies. As there will more color-space mismatches in a color-space read than base-space mismatches in an equivalent base-space read, computational

time dramatically increases as we do read-mapping at a much high mismatch number with color-space reads.

2.4.5 Computational representation of data

Computers run on software which are compiled and then executed as a string of 1s and 0s on the hardware level. Behind layers of abstraction, the data which we store in a computer is an ordered string of 1s and 0s; the binary data format. The smallest unit of storage in a computer is a bit which can represent either a 1 or a 0. Data are often stored in pre-defined data structure which can be 4 bytes long (1 byte = 8 bits). A 4 bytes long structure, holding 32 bits, can effectively express a large range of numbers. If we were to use units of 4 bytes structures to store each DNA nucleotide then this would be putting a lot of bits to waste. As DNA is comprised of only 4 unique characters, a 2-bit data structure is enough to represent a nucleotide uniquely with other 3 nucleotides. In the literature, 2-bit encoding is used extensively to optimize use of space in the storage of DNA and RNA sequences.

In this thesis, we discuss mainly on aligning a read onto a reference genome with high accuracy and efficiency. To achieve this, we have to build a 1-time index of our reference genome. The reference genome is now represented as an FM-index [77]; an opportunistic data structure based on BWT [78] to optimize both space and time complexity in our alignment algorithms. It supports linear time complexity query operation, in terms of the query read length, with a Backwards Search routine. In the following chapter, we will review on other alignments algorithms and the indexing techniques which they have employed in their respective methods.

Chapter 3

Survey of Alignment Methods

Alignment of genomic sequences has been tackled since the advent of sequencing technologies. Pioneering works such as Smith-Waterman [79] and Needleman-Wunsch [80] has guided the development of genomics when sequence alignment was still in its infancy. However, as huge technological advancements are made, the amount of data and the types of sequencing reads which can be generated has increased dramatically. Therefore it is now essential to have methods which can align high volume of reads from various wet-lab/dry-lab origins accurately and efficiently. Hence, a salvo of alignment methods was designed to handle these reads.

3.1 Basics of Genomic Alignments

The aim of all genomic alignment algorithms is to map each query read to a reference genomic location in which it is originally sequenced from. Given a reference genome T and a read R, the alignment algorithm may report a list of putative genomic locations which R is from. These locations are sometimes called hits or mappings. For each reported location which represents R, the aligner can also report the sequence of text-edit operations which can transform R into T. The text-edit

operations are often stored as a string of characters as CIGAR string in a SAM formatted mapping file [81]. CIGAR is made up of the alphabet {M, I, D, N, S, H, P, =, X} and the details of each text-edit operation are described in Table 3.1.

Table 3.1. The possible text-edit operations which can be represented by a CIGAR for the alignment of a query string onto a reference text

Operation	Description
M	alignment match (can be a sequence match or mismatch)
I	insertion to the reference
D	deletion from the reference
N	skipped region from the reference
S	soft clipping (clipped sequences present in query sequence)
H	hard clipping (clipped sequences NOT present in query sequence)
P	padding (silent deletion from padded reference)
=	sequence match
X	sequence mismatch

Given that R can be transformed into T with a sequence of text-edit operations, we would often want to find the location in T such that the number of edit operations needed to transform R into T[loc .. loc + |R| + |gap|] is minimized. Ideally, an aligner would want to align R onto T perfectly, with no mismatches or gaps between R and T. However, in the presence of polymorphisms and sequencing errors, it is uncommon to map R onto T perfectly. As such, a scoring function and scoring matrix can be designed to account demerits for mismatches, opening gaps and extending gaps between the alignment of R and T; the function is also known as affine gap penalty [82]. Thereafter, the best-scoring hit can be chosen from a list of preliminary candidate hits reported by an aligner.

Aside from alignment-score, an important measure of accuracy in an alignment is the Phred-scaled [83] mapping quality score or mapQ [84]. This score equals to -10 $\log_{10}$ Pr{mapping position is wrong}. If an alignment is deemed to be wrong, Pr{mapping position is wrong} = 1 then its mapQ will be assigned 0. In many cases, an alignment with mapQ=0 is deemed as ambiguous. As the mapQ increases, the likelihood of the query read being sequenced from the reported location increases too. However, we

should also keep in mind that a higher mapQ of one read does not mean that it should be trusted more than an alignment of a lower mapQ. This unreliability of mapQ can be attributed from how the mapQ calculation functions are designed or even the alignment algorithms itself.

3.2 Bisulfite-treated DNA-seq aligners

To study the methylation state of a whole genome, the methylome, bisulfite (BS) conversion of the genomic DNA is performed and the resultant BS-converted DNA is sequenced with NGS (BS-seq). BS-seq is then mapped to a reference genome and single nucleotide resolution methylome can be obtained. Although NGS has advanced the study of the methylome, there are still various challenges to infer the methylome accurately from NGS data. In this subsection, we will review on these challenges and the developed approaches which are used to analyze BS-seq data.

3.2.1 Challenges in aligning BS-seq reads

Bisulfite treatment on a DNA fragment causes unmethylated and methylated cytosines to change to uracils and remains as cytosines respectively [76]. As uracils behave as a thymine, unmethylated cytosines will be amplified as adenine upon subsequent PCR amplification for the complementary DNA strand after bisulfite conversion. As a reference genome will not contain any information of methylation of its bases, an allowance of mismatches has to be given when aligning unmethylated cytosines against the reference genome. This will inevitably reduce mapping efficiency.

Since the methylated state of a base in the sequenced read can only be inferred by comparing it to its corresponding mapped base on the reference genome, accurate alignment of BS-seq reads is critical in correctly deriving the methylome. Thus, special consideration has to be made for induced BS-mismatches when aligning a read onto a reference genome, discriminating them from sequencing errors and

polymorphisms. In addition, cytosine methylation is not symmetrical on both strands of the DNA and candidate alignments on each strand must be examined. If the BS-reads are from a directional library, then only the DNA fragments from the top (Watson) and bottom (Crick) strands are sequenced. However, if the BS-reads are from a non-directional library, all four possible orientations of the DNA fragments (Watson-forward/reverse and Crick-forward/reverse) can be sequenced. Non-directional libraries will require aligners to align a BS-read in all of its four possible strand orientations before the best-alignment can be picked for the construction of the methylome. Figure 3.1 shows the possible BS-induced conversions that can take place on cytosines of bidirectional library after bisulfite treatment.

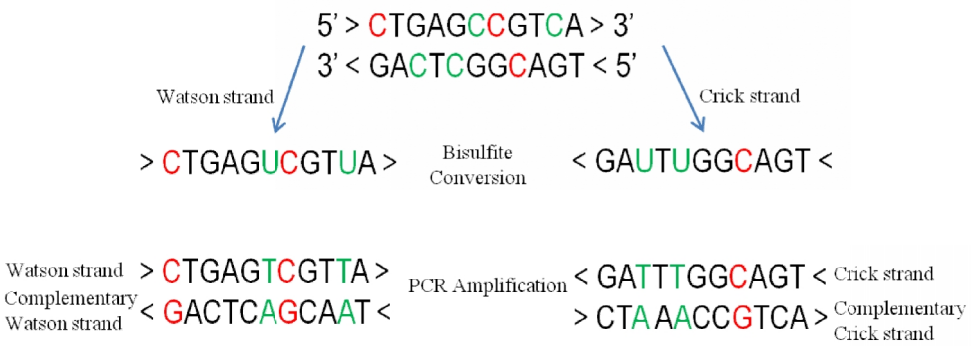


Figure 3.1. PCR amplification of bisulfite treated genomic DNA. The original strands of the DNA undergo bisulfite conversion with unmethylated-C changing to U and methylated-C remaining unchanged after the treatment. Methylated (Red) and Unmethylated (Green).

Apart from allowing a higher number of mismatches when aligning a BS-treated read onto a reference genome than with an untreated DNA read, an even higher number of mismatches should be allowed in aligning a BS-treated color read. This is so as a color base in a color read is called from the consensus of two adjacent nucleotides, a BS-conversion on one nucleotide will result in two adjacent color mismatches. Thus, it is also computationally more expensive to align BS color-space reads than BS letter-space reads. In Chapter 4, we will describe the problem of BS-seq alignment and our solution to it in more details.

3.2.2 BS-aligner for Base-space reads

Base-space reads generated by Illumina sequencing technologies will represent bisulfite mismatches as nucleotides mismatches during alignment. Generally, alignments of BS-reads are classified into two main types: Methylation-aware and Methylation-unbiased. After we obtain the mapping locations of the BS-reads, cytosines on the reference genome will be compared with the letter bases which are mapped to it. After which, the state of methylation on each possible site of methylation will be calculated.

3.2.3 BS-aligner for Color-space reads

Each color base in a color read dictates the transition from one letter base to the next adjacent base. Based on the terminal letter base of a color read, a color read can be converted into letter-space, a base at a time, by using the color-to-base transition matrix described in Chapter 2. However, if any of the color bases are erroneous resulting either from sequencing errors or genomic variations, this naïve conversion from color-space to letter-space will not be suitable. By using the color-to-base transition matrix on a mismatched color base, cascading base-letter mismatches will be introduced after the mismatched color base as the color error is carried forward throughout the whole read when the conversion takes place. Due to this problem, in-silico conversion of cytosine to thymine is not advisable and unbiased methylation mapping is often opted out in these alignment tasks.

As each color base is interrogated from two nucleotides of a read, a letter-mismatch in the sequenced read will introduce two adjacent color-base mismatches into a color read. Hence, the same number of BS-induced conversion in a read will usually need to be aligned at a higher mismatches setting when in color-space than in base-space. Due to this, more computations are needed to align color-space BS-reads. However, with the prior knowledge of methylation in various genomic contexts, we can apply in-silico bisulfite conversion to the reference genome in hope to reduce the number of

mismatches needed to scan a BS-read against a reference genome. For most of the eukaryotic genomes, less than 5% of the methylation happens in non-CpG context [85]. Given this information, we can prepare an in-silico conversion of cytosine to thymine in non-CpG context of the reference genome prior to having BS-reads mapping on it. In all, this is not an unbiased approach to do bisulfite mapping but it does reduce the required number of mismatches which an aligner need to scan a color-space BS-read against the reference genome. Semi methylation-aware mapping can be incorporated into an unbiased aligner to improve mapping sensitivity but remapping reads which cannot be mapped unbiasedly. With the right set of heuristics, this 2-phase alignment strategy can improve mapping sensitivity and accuracy without much impedance on its speed [86].

3.2.4 Methylation-aware mapping

In methylation-aware aligners, cytosine in a BS-read is assumed to be sequenced from an original methylated cytosine, whereas, a thymine is assumed to be either from an unmethylated cytosine or thymine. These assumptions encapsulate all possible combinations of cytosine and thymine that a BS-read can have at the positions reading cytosine and thymine. For example, if a BS-read is to be sequenced with 10 cytosines and thymines then a methylation-aware aligner will try to map 2^{10} possible representations of this BS-read onto the reference genome; permuting between cytosine and thymine at those 10 positions. Due to the amount of search-space which the aligner searches through, it is able to produce the highest possible sensitivity of mapping rates.

At the expense of high mapping sensitivity, there is a reduction in speed and also overestimation of methylation levels. In methylation-aware alignment, the reference genome is used in its whole original state; assuming full methylation throughout the genome. As such, methylated sequences will map to the original genome more easily

than sequences of lower methylation rates. This directly causes an overestimation of methylation levels from the mappings of methylation-aware aligners.

An example of methylation-aware aligner is SOCS-B [87]. SOCS-B starts an alignment of a color read by first converting it into base-space. Four translations are computed, starting from all four possible nucleotides as the terminal base instead of the terminal primer base provided by the original color read. The substrings of the translated reads are enumerated in ternary to form a partial hash over positions represented by a cytosine or thymine. The mapping algorithm is based on an iterative version of the Rabin-Karp algorithm and generates candidate genomic locations of the partial hash. SOC-B then uses dynamic programming and base qualities to compute the most probably methylation state for each cytosine. The optimal alignment should have the least number of color-space mismatches with respect to the reference genome.

3.2.5 Unbiased-Methylation mapping

Fundamentally different from methylation-aware aligners, unbiased-methylation aligners convert cytosines in the BS-reads and reference genome to thymines prior to alignment. This assumes the BS-reads to be fully BS-converted due to the absence of methylation throughout the experimental data. Since both the reference and read has all its cytosines being changed to thymines, the methylation state of either the reference or the read will not affect the alignment of such an in-silico converted read and will not incur any biased estimation of methylation state after alignment. Since the BS-read and genome now assume same state of methylation and assuming we have error-free reads, using a DNA-seq aligner will already enable us to align the converted BS-read onto the converted reference genome with an exact match.

With unbiased mapping, this type of alignment for BS-reads also has some shortcomings. Due to this in-silico conversion of cytosines to thymines, the alphabet

size of the data gets reduce from 4 to 3; the complexity of the data is now greatly reduced. Thus, it has become harder to map such an in-silico converted BS-read onto a similarly converted genome unambiguously. Hence, unbiased-methylation aligners generally yield lower mapping efficiency than methylation-aware aligners.

Some of the BS-aligners which fall into this category are Bismark [88], BRAT [89, 90] and BS-Seeker [91]. Bismark and BS-Seeker are based on Bowtie as a pre-mapping tool. These two methods prepare in-silico fully converted references prior to alignment. Bismark synchronized the threads of Bowtie to consider methylation level for each read on-the-fly but is slowed down due to synchronization of threads. BS-Seeker outputs the preliminary alignments of each thread into separate files and post-process these alignments but have to take up additional storage prior to the consideration of methylation levels for each read. BRAT-BW implements an FM-index alignment routine from scratch to avoid the problem of synchronization and large temporary storage from using an auxiliary program as a pre-mapping tool. BRAT-BW also guarantees to find all alignments if there is at most one mismatch in a prefix of length 32-64 bp (user defined) of the read.

3.2.6 Semi Methylation-aware mapping

Due to the difference of methylation levels on different genomic contexts, in silico conversion of the reference genome can be done to allow for methylation-aware mapping or unbiased methylation mapping on different parts of the genome. A semi methylation-aware mapping approach to profiling human methylome is to do unbiased mapping only in CpG context and methylation-aware mapping in non-CpG context. This approach is used to improve mapping sensitivity of reads by utilizing prior knowledge of expected methylation levels in different genomic contexts as studied in [92]. If such an aligner is used to map BS-reads from flowering plants to a reference genome, the aligner would probably do unbiased mapping in non-CHH (H = A, C and T) context, and methylation-aware mappings in CG and CHG contexts.

The gain in mapping sensitivity comes at the expenses of similar but milder drawbacks seen by methylation-aware aligners.

An example of an aligner that depends on such a mapping strategy is RMAP [93] and PASS-bis [94]. RMAP uses wildcard matching for positions represented by thymines and thus only maps unbiasedly in CpG genomic context; it performs biased mapping in non CpG genomic context. PASS-bis can map both base-space and color-space reads. While it does map base-space reads unbiasedly, it does not do so for color-space reads. Due to the fact that PASS-bis converts a color-space read to base-space read prior to mapping, the base-space read could be mis-represented by the reference due to cascading errors due to this conversion. In order to maximize the mappability of each color read, PASS-bis performs a secondary phase of mapping based on the combinatorial assortment of genomic C-T conversions which is methylation-aware mapping. As this second phase of mapping is slow, it is implemented as an option in PASS-bis and even if it is used, it will only be activated when the read fails to map onto the in-silico fully converted reference genomes unbiasedly.

3.2.7 Comparison of BS-Seq Aligners

In the previous section, we have reviewed on three approaches which are used to align BS-seq reads and the two types of reads which can represent such reads. Below, we summarize the details of the different BS-seq alignment methods for the analysis of methylome in Table 3.2.

Method	Bismark	BRAT-BW	BSMAP	BS-Seeker	PASS-bis	RMAP-BS	SOCS-B	B-SOLANA
Reference	[88]	[90]	[95]	[91]	[94]	[93]	[87]	[96]
Mapping strategy	Bowtie	FM-index	SOAP	Bowtie	PASS	Positional weight matrix matching	Robin-Karp algorithm	Bowtie
Read-space	Letter	Letter	Letter	Letter	Letter/Color	Letter	Color	Color
Paired-end mode	Y	Y	Y	N	Y	N	N	N
Methylation-aware mapping	Unbiased	Unbiased	Biased	Unbiased	Semi	Semi	Biased	Biased
Best Alignment criteria	Lowest number of non BS-mismatches	Lowest number of mismatches OR non BS-mismatches	Lowest number of mismatches	Lowest number of mismatches	Lowest number of non BS-mismatches	Lowest number of mismatches	Lowest number of non BS-mismatches	Lowest number of mismatches
Output	a,b,c,d	a,c	a,b	a,b	a,b	a	a,b	a,b
Advantages	Speed	Speed	-	Speed	Sensitive	-	Full methylation-aware	Speed
Disadvantages	-	-	Speed	-	-	Speed and semi-biased mapping	Speed	-

a. Alignment output.
b. Methylation call output.
c. Methylation caller.
d. Summary of methylation level.

3.3 Gapped DNA-seq aligners

In numerous studies of Mendelian diseases, periodic sequences are found to be mutagenic and context of deletions and insertions in human coding sequences are investigated for possible onsets of diseases [97, 98]. In the case of cancers, differential mutational studies are also carried out to identify somatic differences between normal and tumor tissues. For instance, the identification of aberrant gapped-integrations of hepatitis B virus into the genomes of its hosts will increase the chances for the onset of malignant hepatoma [99]. All these studies using DNA-seq data would not have been possible without the advent of gapped DNA-seq aligners.

In order to improve space-time efficiency in the alignment of the voluminous data brought about by NGS, aligners use indexing strategies to achieve this. Indexing approaches can be sub-divided into two main groups based on whether the reference genome or query reads are indexed. Methods such as BWA [100, 101], Bowtie [102, 103], SOAP [104, 105], Novoalign [106], Stampy [107], PASS [108], CUSHAW [109, 110], SRmapper [111], SeqAlto [112] index the genomic reference. On the other hand, Eland [113], RMAP [114], MAQ [84], SHRiMP [115, 116] and ZOOM [117] index the query reads and map them back onto the genomic sequences.

In general, gapped DNA-seq aligners can be classified in many ways. In this thesis, we classify aligners based on their indexing strategies: hash based, suffix-trie based and merge-sort based approaches. With the context of this thesis in mind, we will not describe the details of the merge-sort based approach; SliderI/II. Readers who wish to understand how merge-sort is applied to the alignment of genomic data can refer to [118, 119].

Gapped aligners mostly involve finding a list of preliminary candidate mapping locations by aligning a subpart of the read onto the reference with a technique called seeding. After which, a secondary step takes place by locally aligning the query reads

with each of the candidate locations, also known as extension phase, before the best-scoring local aligned location is reported to the user. The secondary step is computationally expensive and is the main reason why gapped alignment was avoided in the past for short (~36 bp) reads. There have also been some works revolving around hardware acceleration to improve the execution timings of local alignment [120, 121]. In general, seed-and-extend strategy dominates the field of aligning NGS reads.

In recent advancements of alignment methods, the reference is indexed instead of the query reads. This has the advantages of the programs' working memory being independent and execution times directly proportional to the input query sizes.

3.3.1 Challenges in Gapped Alignment

During the early development of alignment algorithms for NGS reads, a number of aligners [84, 100, 102, 114] were developed. These aligners map a query read onto a reference genome within a number of mismatches only; this type of alignment is also known as ungapped alignment. Due to the limitations of past sequencing technologies, the lengths of reads range from ~25 bp to ~36 bp long. The short read-lengths slow down most algorithms due to its redundant representation in the large reference genome and makes gapped alignment infeasible. However, due to recent advancements in sequencing technologies, read-lengths can reach as long as ~100 bp and ~250 bp from Illumina GAIIx and MiSeq machines respectively. The lengthened read lengths have now made gapped alignment tractable.

As reads get longer, they will more likely contain more SNPs, indels and structural variations in them than shorter reads. Ungapped alignment was not sufficient to align them back onto a reference genome and gapped alignment becomes critically important in aligning NGS reads. In Chapter 5, we will describe the problem of gapped RNA-seq alignment and our solution to it in more details.

3.3.2 Hash/Seed based Approaches

Hash based approaches stem from the first hash-based algorithm, BLAST [31], and follow the seed-and-extend paradigm. Since the publication of the BLAST paper, many developments have been made to its original seeding idea to handle more features which are present in NGS reads. As mentioned before, many aligners follow the seed-and-extend paradigm in the alignment of a query read. In the following subsections, we will report on the different seeding methodologies which have been developed.

3.3.2.1 Seeds

The most primitive type of a seed is a contiguous substring of the query read. Seed is also termed k-mer and is usually referred to a specific n-tuple of nucleotides or amino acid sequences. Pioneering aligner for NGS reads such as BLAST uses 11-mers (for DNA sequences) to seed the alignment query. Subsequently, BLAT [32], MegaBLAST [122] and YAHA [123] were developed to use 11-mer, 28-mer and 15-mer respectively.

By using a seed instead of the original query string for alignment, we can theoretically increase the sensitivity of the method. As a seed is much shorter than the original query read, it will have a higher chance of finding an exact representation of itself in the reference genome. However, due to the reduced informational content in the seed (trimmed from the query read), it is now less unique and can be spuriously represented by many regions of the reference genome. In a seeding approach, the first task is to identify all possible locations to which the original read can be aligned to. Next, an extension step is performed on the seed on the candidate locations to pick the best scoring alignment candidate location to report to the user.

3.3.2.2 Mismatch-seeds

If the correct alignment of a query read lies in a SNP-dense region of the DNA, a k-mer seed might miss it and, worse, other seeds may report a false-positive hit to the user. To resolve this shortcoming, mismatches can be allowed in a seed to avoid missing the correct alignment during the initial seeding-phase of the alignment.

To the best of my knowledge, RMAP [114] is the first method to use mismatch-seed in the alignment of sequenced reads. RMAP uses a different set of seeds to achieve full sensitivity of k-mismatches through the use of k+1 seeds [124]. According to the pigeonhole principle, if we are to partition a k-mismatch read into k+1 equal adjacent and non-overlapping seeds, then at least one of the k+1 seeds can be represented exactly in the reference text. RMAP first identifies locations in the reference genome where the seeds can be matched exactly. Exact matching is preferred as it can be executed more efficiently than approximate matching and only the regions outside of the seeds need to be realigned during the extension-phase of the alignment. The disadvantage of this seeding approach becomes obvious when k is large and each mismatch-seed is small. Ultra-short seeds will return too many spurious candidate locations for the extension-phase to work with and such seeding approaches will take a great hit on its running time.

3.3.2.3 Spaced-Seeds

From the use of contiguous bases as a seed, we are faced with two conflicting performance factors which aligners are designed to improve on: Speed and Sensitivity. In a seed-and-extend paradigm, an aligner would want to minimize the number of local alignments as it is a computational expensive procedure. With this in mind, better filtration methods are designed so that the seed-phase of the alignment will return a minimal set of candidate locations, preferably with one of the seeds representing the correct location of the query read, for the extend-phase of the aligner

to work with. As such, spaced-seeds are developed as a filtration technique to achieve a balance between these two conflicting performance factors [125].

A spaced-seed is a seed which can be specified using a sequence of 1's and 0's. From the name, we can guess that some of the positions in a spaced-seed will be sampled and some will not be sampled, allowing mismatches between itself and the reference sequence internally. A query performed with a spaced-seed will use it as a template and skip the sampling of bases between the reference and the underlying read-sequence which are marked by a 0 in the template spaced-seed. For instance, use of spaced seed in PatternHunter showed that a template '111010010100110111' can be ~50% more sensitive than BLAST's default 11-mer seed for two sequences of 70% similarity [126].

Pioneering aligner based on the use of spaced-seed for filtration, Eland [113] used six spaced-seeds to span the entire query read. The scanning of the six seeds will ensure that a two-mismatch query read (with respect to the reference), regardless of the mismatches' positions in the read, will be represented by at least one of the seeds. MAQ extended the idea of 6-template-2-mismatches from PatternHunter to guarantee recovery of k-mismatches hit of a query read. However, to provide full sensitivity, MAQ required $\binom{2k}{k}$ spaced-seeds to guarantee full sensitivity of k-mismatches mappings. Due to the large number of seeds needed to be scanned, MAQ guarantees full sensitivity by only using the spaced-seed seeding approach on the first 28 bp segment of the read with at most two mismatches. Usually the first k-bases from the 5' end of a sequenced read is selected to seed a query as it is shown to contain less sequencing errors [127]. Once the spaced-seed returns a partial match, the seed match is then fully extended to the full read length.

For the design of a minimum set of spaced-seeds to achieve certain sensitivity requirement and memory usage on a given read length, readers can refer to ZOOM! [117] for more details.

3.3.2.4 q-gram Filter

The primitive approach to recover indels from a short read is to anchor parts of the query string onto the reference in the seeding-phase and the indels are recovered in the extension-phase by using SW-algorithm. Indels can be recovered using this primitive seeding approach, albeit small indels of 1-3 bp [104] and mis-alignments.

In the previous seeding approaches, the candidate hits from one long seed will undergo the extension-phase of alignment. A q-gram is similar to a contiguous seed but by using q-gram filter as a filtration step, the extension-phase is only initiated on a cluster of localized seeds which shares t matching q-grams instead of partial matches from a single long contiguous seed. The q-gram filter is based on the observation that if the query string has at most k mismatches and gaps, then both the query string and the reference of length w will shares at least $t = (w+1) - (k+1)q$ common q-grams [128, 129]. Based on q-gram filter, SHRiMP [115, 116] and RazerS [130, 131] are able to build an index which innately allowing gaps during the seeding-phase of an alignment. A more recent variant of q-gram filter can be seem in MASAI [132] where a set of multiple seeds are searched simultaneously on an additional index to speed up alignment by 11.9x as compared to RaserS 3.

3.3.3 Prefix/Suffix trie based approaches

In trie based aligners, the seed-phase and extension-phase of alignment correspond to the exact string matching problem and inexact string matching problem respectively. For these aligners to find an exact match to substrings of the query reads, they have to build an index of the reference genome using data structures based on FM-index [77], suffix array [133] and suffix tree [134]. The advantage of searching a query string

against a trie-based index is that identical substrings of a reference genome need to be searched only once as identical substrings will collapse into a single traversal path in the trie. As opposed to the trie-based index, identical substrings of the reference genome are not always represented by the same entry in a hash table and thus alignment needs to be performed on each identical copy of the reference.

The first uses of trie in aligners are mainly based on suffix tree and can be traced back to MUMmer [135] and OASIS [136]. However, the disadvantage of using suffix trees as the search index is huge memory-space requirements. An immediate improvement on suffix tree, with respect to space-efficiency, is the development of suffix array (SA) based on Farach's [137] optimal linear time suffix tree construction algorithm. However, the sizes of the index built, based on suffix tree and suffix array still require more or equal memory than the reference itself. Due to the fact that reference genomes can be very large and is best to reside entirely in the working physical memory of the computer during alignment, the development of genomic index-building algorithms was motivated towards building a space-efficient index such as one based on enhanced suffix array (ESA) [138] and FM-index which require only space within the size of the reference or even less.

Vmatch [138] and Segemehl [139] are based on ESA which consists of an SA and auxiliary arrays. Theoretically, ESA is able to store each nucleotide at the cost of 6.25 bytes. Since ESA is a succinct representation of the suffix tree, they allow exact queries at the same time complexity of suffix tree while requiring lower space-requirement needed to index the reference than SA. A further improvement on space-efficiency was achieved through the use of FM-index which is a compressed full-text substring index based on Burrows-Wheeler transform [78]. It was also observed by the inventors of the FM-index that the descendant of a node in a prefix trie can be located in constant time by performing a backwards search on this data structure which allows it to have the same time complexity of performing exact matches with

that of a trie. Some pioneering genomic aligners which use FM-index are Bowtie, BWA and SOAP2. The FM-index is the most used trie-based index due to its minute memory footprint. GEM [140] is shown to be the fastest aligner by our benchmarks in the later chapter of this thesis and is based on the FM-index.

3.3.3.1 Inexact Matching using Trie

As noted from above, aligners are based on different trie-related data structures but all of them can be translated into one another without any loss of information. Trie is excellent in finding exact matches as all identical copies of substrings from the reference are collapsed into a single traversal path but it is not ideal for using inexact matches. Inexact matching is performed on trie by introducing mismatches and/or gaps when the alignment progresses in a depth-first traversal on the index. These introduced mismatches make the search space grow exponentially and affects alignment speed dramatically.

In order to curb the effects of performing inexact matching on trie, aligners have their algorithms designed to only explore a portion of the search space. With the illusion of a pruned search space, aligners hope to achieve speedups with minimal impact on sensitivity and accuracy of alignments in such a designed search/trie-traversal algorithm.

MUMmer, Vmatch, CUSHAW2 and YAHA anchor the alignment with exact matches and join these exact matched segments with gapped alignment. In addition, Segemehl tries to align the longest exact prefix of each suffix but also introduces mismatches at certain positions of the query read to reduce false alignments.

OASIS and BWT-SW searches substrings of the reference by a depth-first traversal on the trie and align these substrings with the query strings by dynamic programming. BWA-SW extends from BWT-SW [141] by representing the query string as a directed words graph which enables it to deploy heuristics to speed up alignments.

As dynamic programming using SW/NW-algorithms is much slower than a linear-time exact matching between the query string and the reference BWT-index, it was avoided as much as possible in Bowtie and BWA. Instead of realigning the short substrings of the reference with the query string, the query and substrings of the reference are only being compared if they are within a number of mismatches else those substrings will not be considered for alignment with the query read. As BWA and Bowtie align a query read by the traversal of an FM-index, it can determine the pruning of some branches in the search space that will result in excessive number of text-edit operations between the query read and reference genome on-the-fly. BWA further speeds up gapped alignment by performing a banded-SW algorithm and employing MegaBLAST's X-Dropoff heuristic for the extension of it seeds.

Bowtie2 samples a set of 22-mer seeds from the query string using exact matching. The seeds are extended to their full length by dynamic programming in order of their frequencies of occurrences in the reference genome as indicated by their suffix array intervals. The prioritized seeds are realigned using hardware accelerated versions of SW/NW-algorithms with Streaming SIMD Extensions 2 (SSE2) hardware instructions for speed.

GEM uses region-based filtration technique to speed up exhaustive alignment of its query string. This technique identifies non-overlapping regions which are non-repetitive (less than certain number of occurrences in the reference) for the extension-phase. The seeds are extended using Myer's fast bit-vector algorithm in GEM. GEM can align up to several times faster than Bowtie2 and BWA as the filtration technique used greatly reduced the number of candidate reference positions needed to be extended by dynamic programming.

3.3.4 Hardware acceleration of seed-extension

During the implementation of algorithms, source codes are written in a rather sequential manner but they do not need to be interpreted and executed in a sequential manner. By exploiting the features of modern hardware and application programming interface, performance of sequential programs can be improved. Many aligners are able to achieve decent speedups by introducing elements of concurrency into their algorithms. Three main exploits which current aligners have in them are multi-threading capabilities on multi-core system, Single-Input-Multiple-Data (SIMD) instructions and Graphics Processing Units (GPUs) accelerations.

Since the introduction of multiple-core central processing units (CPUs) into bench-top personal computers, coders have tried to fully utilize the availability of computational power on these processors by having multiple threads of their single program to run in parallel on a single computer. Since the memory-overheard incurred by an addition thread of operation in genomic alignment is small, using a shared-memory policy within a single execution process such as CUSHAW2. As such, multi-threading is preferred by current users of genomic applications.

In genomic applications, due to the alphabet size of data being handled, 2-bit encoding is often used, and many indexing and alignment operations can be seen as bit-based operations. In the extension-phase of alignment, the binary bits that represent the query string and reference text can be fetched into the registers of the CPUs such that a single instruction can operate more bits than it would normally. A common application of SIMD acceleration [120, 121] is in the SW/NW-algorithm routine used in the extension-phase of aligners such as Bowtie2, SHRiMP and Novoalign.

GPUs are also gaining popularity in genomic applications. CPUs are designed with a few computational cores for serial processing while general purpose GPUs consists of

thousands of smaller computational cores which are designed for parallel processing of users-customized code [142]. More than often, the alignment of a genomic DNA fragment is independent of the alignment of other fragments and executing them in parallel is possible without affecting the end-results of each individual alignment. By using GPUs over CPUs in genomic alignment, SOAP3 [143] was able to achieve tens of times speedup over SOAP2 [105].

3.3.5 Comparison of Gapped DNA-Seq Aligners

In the previous section, we have reviewed on two indexing strategies and two mapping approaches for aligning gapped reads. In Table 3.3, we summarize the details of different gapped alignment methods together with a short description to each of them.

Table 3.3. Methods for gapped alignment and their respective main indexing/mapping strategies

Methods	Index		Type of Mapping Approach		Description	Reference
	Reference	Read	Hash-based	BWT-based		
BFAST	X		X		Uses empirical derived seed template for mapping fixed read lengths and genome sizes	[144]
BLASR	X			X	Maps PacBio reads with successive refinements to the local alignments of the seed locations	[145]
Bowtie	X			X	Bowtie1/2 aim at fast and sensitive mappings of reads. Version 2 targets longer reads and can do gapped alignment too	[102, 103]
BWA	X			X	BWA-short targets short reads of ~100bp with low (~3%) error rate. BWA-SW targets longer reads up to 10kbp with higher error rate	[100, 101]
CloudBurst		X	X		Uses Hadoop MapReduce framework to do alignment in the CLOUD	[146]
CUSHAW2	X			X	CUSHAW1 is targeted for CUDA-enabled GPUs. CUSHAW2 (-GPU) is targeted for long read alignment for CPUs (GPUs).	[109, 110, 147]
Eland		X	X		First NGS short read aligner. Allows up to two mismatches in an alignment	[113]
GEM	X			X	Based on adaptive region based filtration technique for sensitive and extremely fast alignment efficiency	[140]

GNUMAP	X		X		Targets accurate gapped alignment of Illumina reads	[148]
Hobbes	X		X		Reports multiple putative mappings fast	[149]
MAQ		X	X		First program to use posterior mapping score to disambiguate multiple candidate mappings	[84]
Masai	X			X	Uses approximate seeds to speed up alignments	[132]
MOM	X		X		Identifies the maximal length match within the short read.	[150]
Mosaik	X		X		Uses banded SW-algorithm for extending seed locations	[151]
mrFAST	X		X		Uses cache oblivious memory technique to minimize memory miss-transfers to speed up gapped alignments of letter-space reads. mrsFAST is ungapped version of mrFAST. drFAST is designed for color-space reads.	[152-154]
Novoalign	X		X		High sensitivity and specificity alignments. Uses base qualities in all steps of alignments and output good calibrated posterior mapping quality scores	[106]
PASS	X		X		Alignment of words are pre-computed from the hashed index of the genome	[108]

PerM	X			X		Uses periodic seeds to quickly find alignments of up to four mismatches with full sensitivity	[155]
ProbeMatch	X			X		Uses gapped q-grams and q-grams of various pattern to identify seeding locations from a reference	[156]
RazerS		X		X		No restriction on read length. Seeds can be designed with predictable tradeoff between sensitivity and speed	[130, 131]
REAL	X			X		Targeted at fast, accurate and sensitive mappings of single-end reads	[157]
RMAP		X		X		Can map reads with an arbitrary numbers of mismatches	[114]
SeqAlto	X			X		Uses adaptive seeding approach to terminate alignment when alignment reaches certain confidence for reporting	[112]
SeqMap		X		X		Can align up to a mixture of 5 mismatches and gaps between the reference and the read	[158]
SHRiMP		X		X		Aims at accurate mapping of color-space reads. Version 2 index the reference instead of the reads	[115, 116]
Slider			Merge-sort			Reduces the percentage of base call error mismatches in an alignment; produces high SNP discovery rate	[118, 119]
SOAP2	X				X	Fast and accurate alignments on a wide range of read lengths. Improved version of SOAP1. SOAP3 is akin to GPU-enabled SOAP2.	[104, 105, 143]

SRmapper	X		X	Small memory footprint. ~2.5GB for human genome.	[111]
SSAHA2	X		X	Fast alignments for reads of small number of variants	[159]
Stampy	X		X	High sensitivity of reads with high percentages of variants in them. Very slow but can be sped up by using BWA as a pre-mapping tool	[107]
Subread	X		X	Uses novel 'seed-and-vote' paradigm to perform fast alignments	[160]
YAHA	X		X	Recover optimal breakpoints of alignments for structural variation detection	[123]
ZOOM		X	X	100% sensitivity of reads between 15-240bp with reasonable number of mismatches and gaps.	[117]

3.4 RNA-seq aligners

RNA, together with DNA and proteins, is one of the three major macromolecules which are needed for life. Pre-mRNA is synthesized from the DNA in a process called transcription and is matured by having its introns removed in eukaryotic cells [161]. In mammalian genomes, alternate splicing of the same gene region adds onto the genomic complexity by generating multiples variants of a single gene known as mRNA isoforms [162]. The disruption in the synthesis of mRNA isoforms can cause genetic diseases [163, 164].

Since it is motivating to produce a map of genes together with their expression level on the genome-wide scale across various cell types, it is critical to annotate a transcriptome efficiently and accurately. The prevalent method for producing a genome-wide gene-map requires the costly and low-throughput method of applying capillary sequencing on cDNAs or expressed sequence tag (EST) fragments [72]. Due to the usage of low-throughput sequencing, the true complexity brought about by alternate splicing to isoforms and cell-type specific splicing events cannot be studied in depth without the advent of high-throughput method. Alternatives to capillary sequencing of ESTs are tiling arrays and splice-aware microarrays. Tiling arrays are able to interrogate larger transcribed regions but at limited resolution [165]. As for SJ-aware microarrays, they are fabricated with probes which hybridize to known RNA sequences and will not be suitable to quantify expression levels of novel or unrepresented genes [166, 167].

Due to the advent of NGS technologies, we are able to sequence the cDNAs derived from RNA fragments using NGS technologies [34]. This gave rise to high throughput sequencing of RNA fragments which we know as RNA-seq. Methods such as Exonerate [168] and BLAT [32] which are designed for the alignment of capillary sequencing technologies are now unable to map voluminous NGS data within competitive timings. In order to improve space-time efficiency in the alignment of

RNA-seq data, computational tools have to be developed to deliver unparalleled performance for the alignment of RNA-seq reads.

Akin to the analysis of DNA-seq datasets, the first step of analyzing RNA-seq datasets is to align the RNA-seq reads back onto a reference genome or transcriptome. Given the myriad of aligners developed in the recent years, we are able to group those aligners which targets RNA-seq reads into two main groups based on their read mapping strategies: Unspliced and Spliced aligners.

3.4.1 Challenges in RNA-seq Alignment

The goal of RNA-seq alignment is the resolution of the gene-map with all exact splice junctions annotated in it for different types of cells. Although the main challenge in gapped DNA-seq alignment is similar to RNA-seq alignment, the task of RNA-seq alignment is tougher as reads now need to be split into smaller k-mer for identification of small-exons (<20 bp) too. Shorter read fragments will be harder to map unambiguously and will be more computationally expensive to resolve during the extension phase of the alignment. In addition, accurate detection of split junction without prior knowledge of splice signals is still an open problem especially in lowly transcripted regions. To make matter worse, canonical splice signals are ubiquitous in both transcripted and non-transcripted regions.

The presence of unexpressed genomic sequences, which are similar to concatenated sequences of multiple exons also, poses problem to the accurate alignment of RNA-seq reads. These regions, which are known as pseudogenes [169], are not transcripted from the genomic DNA into mRNA sequences and should not have RNA-seq reads mapping to it. However, due to the case whereby multi-exons spanning reads may map to these regions, without splicing, poses a great challenge to exon-first method. Seed-and-extend methods will also face problem in determining if an unspliced full alignment of a RNA-seq read should actually be spliced or not.

In Chapter 6, we will describe the problem of spliced RNA-seq alignment and our solution to it in more details.

3.4.2 Unspliced/Annotation-guided Aligners

The unspliced aligners are mostly as described in the previous section of gapped aligners. In the aspect of RNA-seq, unspliced aligners are mostly used to align RNA-seq reads to the assembled transcriptome without having the need to allow for large intronic gaps during alignment. Due to the use of the assembled transcriptome, unspliced aligners are also known as annotation-based aligners in the literature. Unspliced aligners are used when de novo detection of splice junctions is not needed and are great for mappings reads against a well annotated transcriptome for quantification studies [170-172]. Some examples of unspliced aligners are ERANGE [170] and RNA-MATE [173].

ERANGE begins by mapping reads onto the DNA reference genome. Reads that cannot map onto the DNA reference will be mapped again onto a known transcriptome. Highly reliable ambiguous mappings from the previous 2-step alignment will be used to calculate the Reads Per Kilo Megabases (RPKM) of putative transcripts. Lastly, the assignment of ambiguous mappings to the current transcriptome will be based on the previously calculated RPKM as a form of weightage.

RNA-MATE is developed for aligning color-space RNA-seq reads. It follows a similar 2-step alignment strategy used in ERANGE. However, it is largely based on a recursive methodology as a read is truncated if it fails to map to a known transcriptome or DNA reference. The process of truncation is repeated until the truncated read length reaches a certain lower limit or is can be mapped using the 2-step alignment strategy. RNA-MATE also provides an option to use ambiguous

mappings from the alignment step for the quantification of expression transcripts. RNA-MATE is now superseded by X-MATE [174].

3.4.3 Spliced Aligner

Unlike unspliced aligners, spliced aligners align RNA-seq reads back onto a genomic DNA reference genome consisting of adjacent exons and introns in it. Spliced alignment was generally evoked by the longer read lengths that are introduced by improved sequencing technology. The transcripts represented by NCBI Reference Sequence Database (RefSeq) [175, 176] are downloaded and used as an oracle set for BEERS [177] to simulate RNA-seq reads from. On 76 bp, 100 bp and 120 bp of 2 millions simulated reads each, it was observed that there is 17.8%, 22.4% and 25.5% of reads spanning across two or more exons respectively. With increasing read lengths generated by improving sequencing technologies, it has become more important for aligner to handle spliced alignments more efficiently and accurately. Spliced aligners can generally be classified into two categories based on their method of detecting splice junctions - Exon-first and Seed-and-Extend. We will also describe learning-based approaches which is a sub-class of spliced aligners here.

3.4.3.1 Exon-first Approaches

Aligners which are categorized as exon-first approaches map the original RNA-seq reads onto a DNA reference first. This initial alignment step will only align reads which do not span across exon-exon junctions successfully. Hence, they are named "exon-first" approaches. This step essentially quantifies transcript abundance using only exonic reads and does not identity the exon-exon breakpoints. The mapped exonic reads are used as a guide to guide the detection of splice junction in the latter extension step. TopHat [41] and G-Mo.R-Se [178] to incorporate the mappings of the exonic reads to guide the alignment of non exonic reads. The downside of this approach is that sufficient coverage is needed to be provided by the exonic mappings before it can be used to align non exonic reads confidently.

The unmapped reads from the initial alignment step are now split into shorter fragments and aligned independently. Since the fragments are now shorter, they stand a better chance of aligning exactly onto the DNA reference. From before, we know that a shorter read will produce spurious seeding locations but will allow full-read which are unable to map exactly before to be able to be mapped onto the reference now. Due to this, more computational effort needs to be spent on realigning the many alignments that may be returned from the shorter mapped read lengths to their full read-length. However, exon-first aligners can be very efficient as minority of reads would need this computationally expensive step.

Some examples of exon-first aligners are GEM (splice alignment module) [140], TopHat1/2 [41, 42], MapSplice [37], SpliceMap [40], SOAPsplice [179] and PASSion [38].

3.4.3.2 Seed-and-Extend Approaches

This class of spliced aligners begins aligning reads onto a DNA reference by splitting them up into smaller fragments. The candidate alignments of these fragments are then used to localize the actual alignment of the original read. By merging initial seeding alignments, local realignment, the seeding alignments of the split fragments can extend toward one and another to the original full read length. Some methods of this approach are QPALMA [39], GSNAP [180], Supersplat [181] and STAR [35].

Recently, seed-and-extend strategy are also extended to consider a read as a concatenation of multiple smaller read fragments by using multiple seeds in the alignment of the reads. These methods include CRAC [182], OLego [36] and Subjunc [160].

For both exon-first and seed-and-extend approaches, it is possible to align flanking intronic dinucleotides to known canonical splice signals to increase the reliability of detecting novel splice junctions and recover short-overhangs.

3.4.3.3 Learning-based Approaches

One of the earliest RNA-seq aligner is QPALMA [39] which is a spliced aligner. It is based on a learning algorithm, support vector machines (SVM) [183], to learn how splice junctions are positioned on a reference genome by training with a known set of spliced mappings. However, the performance of this strategy relies heavily on the completeness of the underlying reference transcriptome for efficient and accurate alignment. Unspliced aligners do have the same pitfall as microarray as they cannot interrogate expression levels of novel genes or unrepresented transcription regions in the used reference transcriptome. QPALMA is succeeded by PALMapper [184] which is a combination of the learning-based spliced alignment method QPALMA and the short read alignment tool GenomeMapper [185]. PALMapper improves on the speed by using a banded semi-global and spliced alignment algorithm of GenomeMapper to align the RNA-seq reads while taking advantage of base quality information and the predictions of splice junctions from the SVM algorithm.

Also based on a learning approach is HMMSplicer [186] which is a tool developed for the discovery of novel and known splice junctions. HMMSplicer trains a hidden markov chain model (HMM) by using the halves of aligned reads which initially cannot align its entirety onto the reference genome. From the trained HMM model, the method tries to find the splice junctions within the other halves of these aligned reads and match the remaining portion of the read downstream of the spliced sites. As HMMSplicer trains using data from the input itself, it is capable of detecting novel splice junctions.

Since the objective of using machine-learning in RNA-seq alignment is for the accurate discovery of known splice junctions, learning-based approaches can also be regarded as a subset of spliced aligners.

Most aligners assume a known gene model for the sequenced reads and can be biased towards the detection of canonical (~98.7% of the splicing junctions in mammalian sample [56]) and semi-canonical junctions . Non-canonical junctions such as splicing of exons that does not lie on the same RNA transcript (trans-splicing [187]) may not be detected. However, learning-based approached can learn from sample-specific data and train a sample-specific model for unbiased detection of splicing junctions without the annotations of known splicing motifs. As such, this approach might be more suitable for de novo discovery of splicing junctions of less studied organisms.

3.4.4 Comparison of RNA-seq Aligners

In the previous section, we have reviewed on two main mapping strategies, possible usage of known canonical signals and annotated intron gaps for biased detection of splice junctions. We summarize and characterize different RNA-seq alignment methods for the analysis of transcriptome in Table 3.4.

Table 3.4. Methods for RNA-seq alignment and their respective mapping strategies and usage of annotations for spliced alignments

Methods	Mapping Strategy		Use of Annotations		Splice junction Model		Description	Reference
	Exon-first	Seed-Extend	Yes	No	Biased	Unbiased		
ABmapper	X			X	X		k-mer from both ends of seeds are searched against Suffix Array index and extended towards each. Still essentially an exon-first approach as seeds is extended for exonic mapping first.	[188]
CRAC		X		X		X	Uses k-mer profiling to detect candidate mutations, indels, splicing and chimeric junctions	[182]
GEM-rna-mapper	X		X	X		X	Based on GEM. (unpublished)	[140]
GSNAP		X		X	X		Detection of novel splice junctions is based on a probabilistic model implemented as a maximum entropy model on user-specified known splice junctions.[189]	[180]
HMMsplicer	X			L	X		Uses half-read mapping to train a HMM to detect most probable splice position	[186]
MapNext		X	X	X	X		Using un-annotated mode, searches paired k-mer in a hash table within 10kbp and with the same strand	[190]
MapSplice	X			X		X	Sensitive for exonic reads	[37]

OLego	X			X	L	X		Targeted at finding small exon and good specificity on exonic reads. Based on logistic regression for detecting splice junctions	[36]
OSA		X		X	X	X		Trims poor-quality 3' ends of reads and improves alignment speed	[191]
PALMapper		X		L		X		Combined QPALMA and GenomeMapper	[184]
PASSion	X				X	X		Use of pattern growth algorithm and splicing signals to detect both novel and known splicing junctions	[38]
PASTA		-			L	X		Similar to seed-extend strategy, it uses patterned alignments of 2 subreads split at various points for spliced mapping	[192]
QPALMA		X		L		X		Used in silico spliced reads from annotated genome to train a 'weighted degree' kernel with SVMs	[39]
RNASEQR		X		X			X	Reduces false identifications of SNVs near splice junctions	[193]
RUM	X			X	X	X		Combination of Bowtie (exonic) and BLAT (spliced) are used to align reads to both the transcriptome and genome	[177]
SeqSaw		X			X		X	Based on SeqMap [158]. High specificity in detecting splice junctions	[194]
SOAPsplice	X				X	X		Use two filtration strategies to produce low false positive rates	[179]

SpliceMap	X			X	X		50bp reads cannot be extended for more than 40bp and residual overhang must be >10bp	[40]
SplitSeek		X		X		X	Suitable for detecting novel splicing junctions and chimeric transcripts	[195]
STAR		X	X	X		X	Ultrafast aligner that can discover non-canonical junctions and fusion junctions	[35]
Subread/Subjunc		X		X		X	Uses a seed-and-vote strategy on sub-reads for alignment	[160]
Supersplat		X		X		X	Finds every possible splice junction by mapping different 2-chunk reads for alignment	[181]
TopHat 1/2	X		X	X	X		Construct exon islands with exonic reads to determine localize final splice junctions. TopHat2 can handle indels	[41, 42]
TrueSight	X			X	X		Takes all possible splice junctions of a transcriptome from the aligning reads and learn a regression model to find best assignments for them	[196]
X-Mate	X			X		X	Upgraded version of RNA-Mate [173]. Designed for color-space reads but can align base-space reads too	[174]

L is for machine-based learning.

Chapter 4

Bisulfite Sequencing Reads Alignment

4.1 Introduction

DNA methylation modifies the nucleotide cytosine by the addition of methyl groups to its C5 carbon residue by DNA methyltransferases [197]. This modification can be inherited through cell division and it plays an important role in many biological processes, such as heterochromatin and transcriptional silencing [198, 199], imprinting genes [200], inactivating the X chromosome [201] and silencing of repetitive DNA components in healthy and diseased (including cancerous) cells [202, 203]. Methylation analysis can also be used to diagnose pre-natal Down's syndrome [204]. Thus, the genome-wide methylation profiles of different tissues are important to understand the complex nature and effects of DNA methylation.

In the past decade, quantum leaps have been made in the development of sequencing technologies by vendors such as Illumina-Solexa and Applied BioSystems (AB)-SOLiD. These can generate millions of short reads at a lower cost compared to traditional Sanger methods [75, 205-208]. Bisulfite (BS) treatment converts unmethylated cytosines (Cs) to uracils (which are then amplified by PCR as thymine

(T) without affecting the other nucleotide bases and methylated cytosines [209]. Next-generation sequencing coupled with bisulfite treatment enables us to produce a methylome of a genome at single base resolution and low cost.

4.2 Related Work

One important step in calling methylation of a genome is to map BS reads. Mapping of BS reads is different from that of ChIP-Seq and RNA-Seq data since the non-methylated Cs are converted to Ts by BS treatment and subsequent PCR. The BS reads are difficult to map to the reference genome due to the high number of mismatches between the converted Ts and the original Cs. For mapping Illumina BS reads, the pioneering published methods are BSMAP [95] and RMAP [93]. BSMAP aligns a BS read to the reference genome by first enumerating all C-to-T combinations within a user-defined length k seed of the reads; then, through hashing, BSMAP aligns the seeds onto the genome and putative alignments are extended and validated with the original reads. After this step, BSMAP can output an unambiguous hit for each read, if available. BRAT [89] uses a similar strategy as BSMAP. It converts the reference genome into a TA reference and a CG reference (each converted reference uses one bit per base). Using a 36-mer hash table, BRAT aligns the first 36 bases of every read and its 1-neighbors on the two converted references to identify possible alignments. RMAP uses layered seeds as a bit-mask to select a subset of the bases in the reads and constructs a hash table to index all the reads. However, these seed-hash-based approaches are slow.

Subsequently, several methods were proposed to map BS reads onto the converted genomes. MethylCoder [210] surfaced as a BS read mapper that uses GSNAP [180] to do a primary mapping of *in silico* converted reads (that is, all Cs in the reads are converted to Ts) onto a converted reference genome (that is, all Cs in the genome are converted to Ts). Those reads that fail to map onto the converted genome will be remapped again in their original forms onto the original reference. BS-Seeker [91]

and Bismark [88] use a similar conversion strategy as BSMAP except that they align the reads with Bowtie [102] and unique hits are found by a seed-then-extend methodology. (Note that every tool has its own uniqueness criterion. A tool will denote a read to have a unique hit if it finds exactly one occurrence of the read in the reference genome.) Both methods trade accuracy for efficiency.

AB-SOLiD color reads are different from Illumina reads since they encode every pair of bases with four different colors. (For more details on this sequencing technology and how it differs from sequencing by synthesis, see [18, 68, 211, 212].) Unlike BS mapping of Illumina reads onto converted genomes, mapping BS color reads onto converted genomes produces many mismatches when the regions are highly methylated [213]. This also causes a dramatic decrease in the unique mapping rate and unbiased measurements of hypomethylation sites. In addition, a single color error in a read will lead to incorrect conversions throughout the rest of the read (Figure 4.1a, b). Although *in silico* conversion of Cs to Ts guarantees unbiased alignments in base space, this is not preferred for color reads.

SOCS-B [214] and B-SOLANA [96] were developed to map BS color reads. SOCS-B splits a color read into four parts and tries to get hits for any combination of two parts via an iterative Rabin-Karp approach [215]. SOCS-B uses a dynamic programming (DP) approach to convert an aligned read to the aligned portion of the reference genome. The conversion starts with all possible four nucleotides as the pseudo-terminal base (rather than just the terminal base from the read). Subsequently, the sub-strings of the four translations are used to generate partial hashing seeds that are then mapped onto the hashed reference genome. However, the running time of SOCS-B is long and the unique mapping rate is too low to be practical. B-SOLANA improves speed and unique mapping rate by aligning against both fully converted and non-CpG converted references simultaneously with Bowtie. The final hits are determined by checking their number of mismatches.

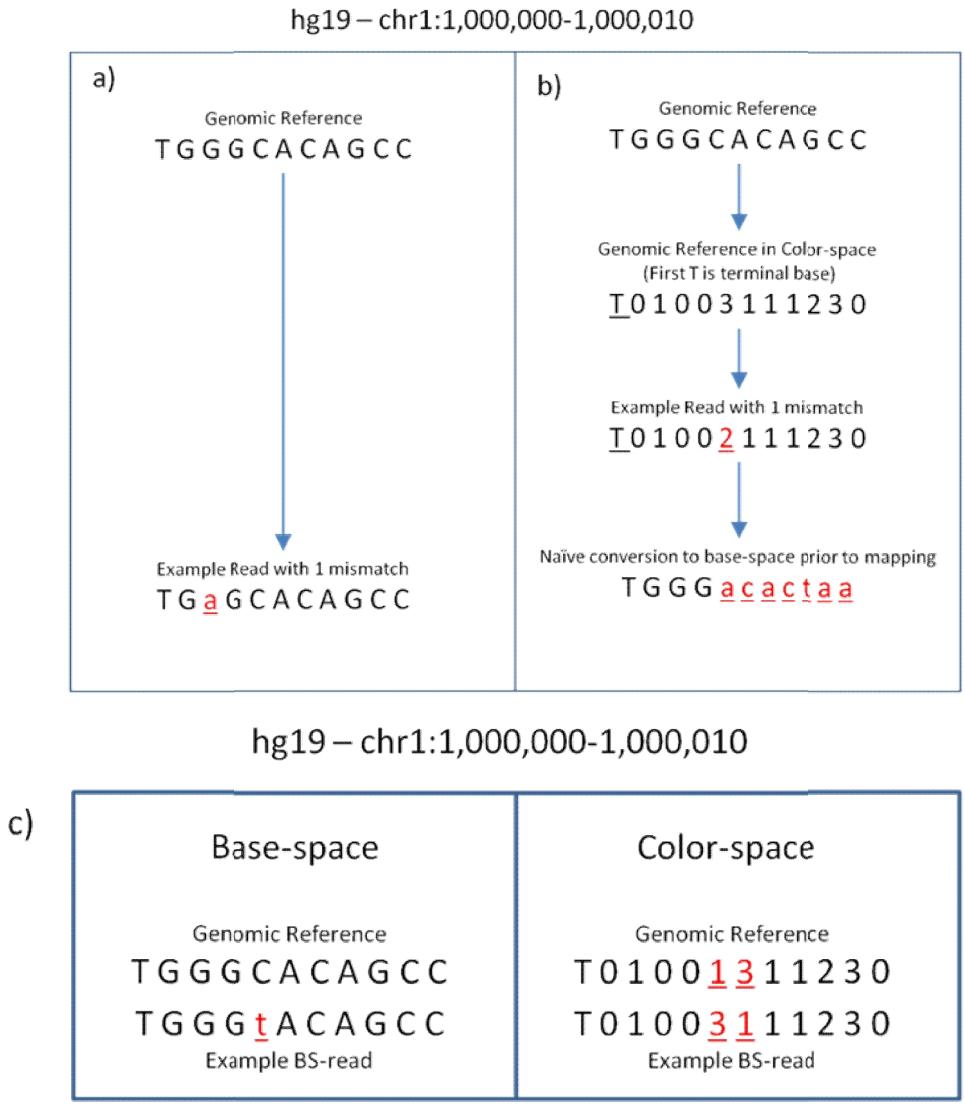


Figure 4.1. (a, b) Base call error simulation in Illumina and SOLiD reads reflecting one mismatch with respect to the reference from which they are simulated in their respective base- and color-space. (b) A naïve conversion of color read to base space, for the purpose of mapping against the base space reference, is not recommended as a single color base error will introduce cascading mismatches in base space. (c) A BS conversion in base space will introduce two adjacent mismatches in its equivalent representation in color space.

A recent *Nature* review paper [213] reported that Bismark and BS-Seeker are the most recent published methods for mapping BS base reads whereas B-SOLANA is the most recent published method for mapping BS color reads. This review also highlighted the main challenges to develop methods that can map reads unbiasedly and to improve unique mapping rates for mapping color reads.

4.3 Results

BatMeth (Basic Alignment Tool for Methylation) was developed by us to address the issues of efficiency and accuracy on mapping BS reads from Illumina and BS color reads from SOLiD. Unlike existing algorithms, BatMeth does not map the BS reads in the initial stage. Instead, BatMeth counts the number of hits of the BS reads to remove spurious orientations of a read. This idea has significantly sped up the mapping process and has also reduced the number of false positives. When dealing with color reads, BatMeth reduced bias on hypomethylation measurements with high initial mismatch scanning. BatMeth also employed a DP conversion step for the color reads to account for BS mismatch accurately and an incremental processing step to produce higher unique mapping rates and speed (refer to the Materials and methods section for details).

4.3.1 Evaluated programs and performance measures

In order to evaluate the performance of our pipeline, we have tested the following programs: BSMAP, BS-Seeker, and Bismark for base-space mapping; and SOCS-B and B-SOLANA for color-space mapping. BS-Seeker and Bismark only output unique hits for each read. BSMAP, SOCS-B and B-SOLANA will output at most one hit per read, with a flag to indicate if a hit is unique. Some reads can map to multiple genomic locations and since a read can only come from one origin, retaining such non-unique mappings will affect the accuracy of downstream analysis such as unbiased methylation site calls. To avoid the problem of wrong methylation calls, all six programs were thus compared with their unique mapping rates.

All our experiments were run on a server equipped with an Intel Xeon E7450 @ 2.40GHz and 128 GB of RAM. We allowed the same mismatch number and CPU threads on all the compared programs in our experiments. Other parameters were kept at default.

We have compared the performance of BatMeth with recent stable versions of BSMAP (2.4.2), BS-Seeker, Bismark (0.5.4), SOCS-B (2.1.1) and B-SOLANA (1.0) using both simulated and real data sets (BS-Seeker, Bismark and B-SOLANA used Bowtie 0.12.7 in our experiments). With simulated Illumina and SOLiD reads, BatMeth (default mode) recovered the highest number of hits, has the lowest noise rate and is the fastest among the compared programs. BatMeth is also able to produce better unbiased results than the other programs by comparing the detected methylation levels in different genomic contexts over simulated data sets (Illumina and SOLiD reads) of different methylation levels. With a paired-end library, we show the specificity of our Illumina results by counting the pairs of concordant paired reads that fall within the expected insert size of the library. With a directional library, we indicate the specificity of our results with direction-specific information. In summary, BatMeth is an improved BS mapper in terms of speed, recovery rate and accuracy, and, in particular, has addressed the main challenges of mapping color reads identified in [213].

We have not included RMAP in our comparisons as it only performs biased mapping in a non-CpG context. MethylCoder was also not included because a newer variant of it, namely B-SOLANA, has been released (MethylCoder's release notes mention that it is now deprecated due to the release of B-SOLANA). BRAT was considered impractical as it only considers one base error in the first 36 bp of a read and therefore was not included in our experiments.

Below, we define ‘recovery’ to be the portion of the unique hits recovered by the programs. We also define ‘accuracy’ to be the portion of the recovered hits that are correct. All recorded timings are wall clock times. A ‘hit’ is a genomic location to which a read is aligned. Lastly, due to sequencing errors and BS mismatches, we allow k (>0) mismatches when mapping a BS read onto a reference. A genomic location is deemed to be unique for a read if it is the only location with the lowest number of mismatches with respect to the read.

4.3.2 Evaluation on the simulated Illumina data

We generated 1 million reads, each 75 bp long, which were randomly simulated from the human genome hg19 using the simulator found in RMAP-bs [216]. The data set was built by allowing a maximum of three mismatches per read. Each C in the simulated read, regardless of its context, was BS converted at a uniform rate of 97%. We benchmarked BatMeth and the other methods, BSMAP, BS-Seeker and Bismark, on this data set. Since the original coordinates in the simulated reads are known, we can evaluate the accuracy of all the programs by comparing their outputs with the original coordinates. We mapped the reads onto the reference allowing at most three mismatches. BatMeth recovered the most number of true positives and the lowest number of false positives and is the fastest program, as shown in Figure 4.2a.

We further illustrate that BatMeth can achieve better unbiased methylation calls than the best published method, Bismark, by replicating the experimental settings of Figure 4.2b in [213]. We used the same simulator, Sherman [217], the same number of reads (1 million), the same length of read (75 bases) and the same reference genome (NCBI37) for this comparison. We used Sherman to simulate 11 sets of data, from 0% to 100% of BS conversion in increments of 10%. Sherman emulates BS conversion by converting all Cs regardless of their genomic context with a uniform distribution. No non-BS mismatches were allowed in the reads, during the scanning phase, for both BatMeth and Bismark. The results produced by Bismark show exactly

the same trends as the graph that was presented in [213]. Table 4.1 presents the performance of BatMeth and Bismark in terms of mapping efficiency, detected methylation levels in different genomic contexts from various *in silico* methylation rates in different contexts (CG, CHG and CHH genomic contexts, where H stands for base A/C/T only). BatMeth has an average of approximately 1.1% better mapping efficiency and about twice the accuracy as Bismark in estimating methylation levels of Cs from different genomic contexts with different initial methylation levels.

4.3.3 Evaluation on the real Illumina data

We downloaded about 850 million reads sequenced by Illumina Genome Analyzer II (Gene Expression Omnibus (GEO) accession number GSE19418) [218] on H9 embryonic stem cells. Since BSMAP is not efficient enough to handle the full data set, 2 million paired-end reads were randomly extracted from one of the runs in GSE19418 for comparative analysis with BSMAP. Reads were observed to have a lot of Ns near the 3' end and were trimmed down to 51 bp before being mapped onto hg19 with at most two mismatches per read.

For this sample data set, BatMeth mapped 1,518,591 (75.93%) reads uniquely compared to 1,511,385 (75.57%) by BSMAP, 1,474,880 (73.74%) by BS-Seeker and 1,498,451 (74.92%) by Bismark. Out of all the hits reported by BatMeth, 1,505,190, 1,464,417 and 1,481,251 mapped loci were also reported by BSMAP, BS-Seeker and Bismark, respectively. BatMeth found 13,401, 54,174 and 37,340 extra hits when compared to BSMAP, BS-Seeker and Bismark, respectively. BSMAP, BS-Seeker and Bismark also found 6,195, 10,463 and 17,220 extra hits, respectively, when compared to our result set.

Next, we mapped the two reads of every paired-end read independently to investigate the mapping accuracy of the compared programs. Since the insert size of this set of

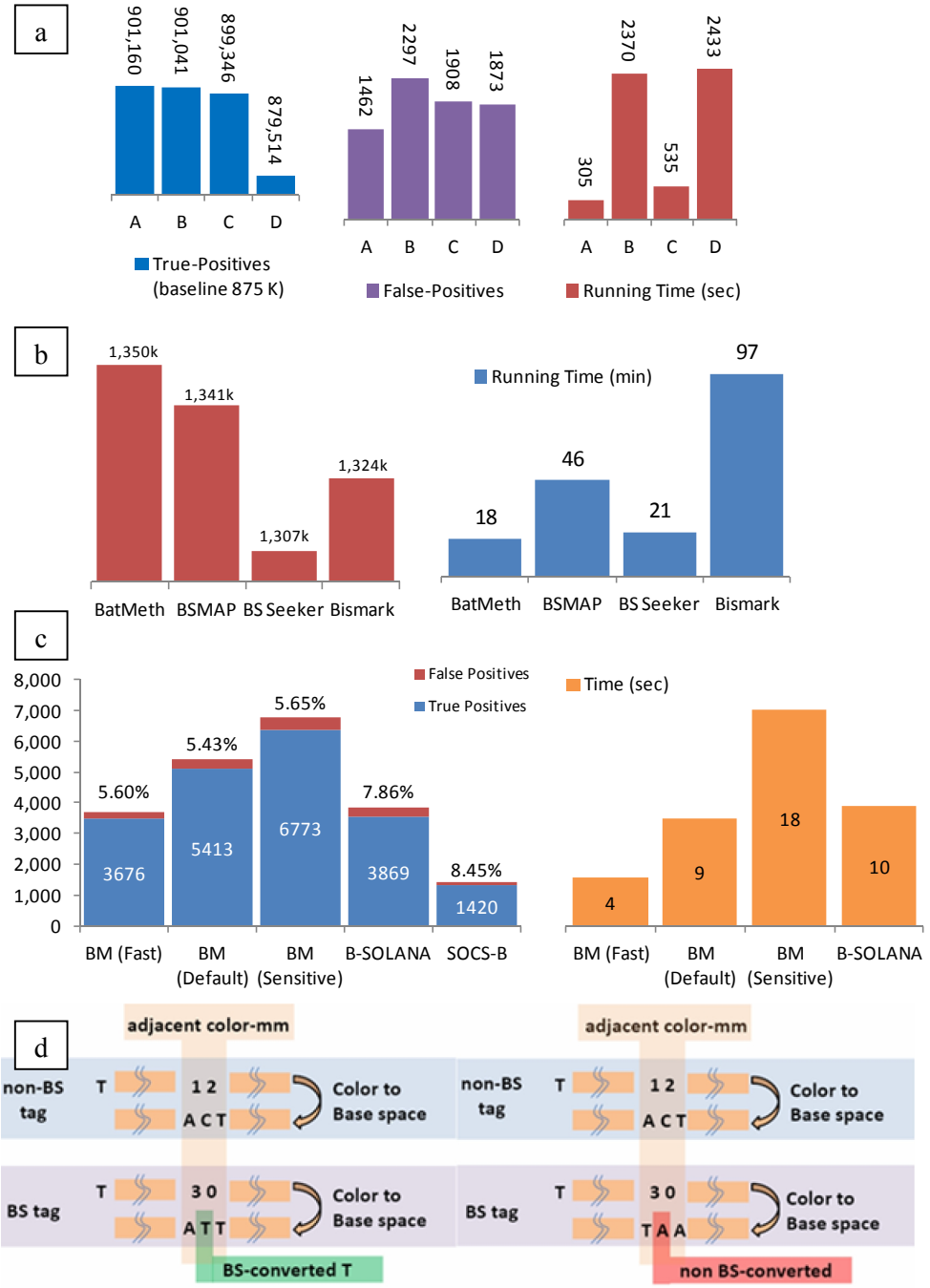


Figure 4.2. Benchmarking of programs on various simulated and real data sets (a) Benchmark results of BatMeth and other methods on the simulated reads: A, BatMeth; B, BSMAP; C, BS-Seeker; D, Bismark. The timings do not include index/table building time for BatMeth, BS-Seeker, and Bismark. These three programs only involve a one-time index-building procedure but BSMAP rebuilds its seed-table upon every start of a mapping procedure. (b) Insert lengths of uniquely mapped paired reads and the running times for the compared programs. (c) Benchmark results on simulated SOLiD reads. Values above the bars are the percentage of false positives in the result sets. The numbers inside the bars are the number of hits returned by the respective mappers. The graph on the right shows the running time. SOCS-B took approximately 16,500 seconds and is not included in this figure. (d) BS and non-BS induced (SNP) adjacent color mismatches.

Table 4.1. Comparison of mapping efficiencies and estimation of methylation levels in various genomic contexts

BatMeth (%)				Bismark (%)				Oracle BS rate (%)
Mapping efficiency	CG	CHG	CHH	Mapping efficiency	CG	CHG	CHH	
94.2	0.0	0.0	0.0	91.1	0.0	0.0	0.0	0.0
94.0	10.0	10.0	10.0	92.1	10.0	10.0	10.0	10.0
93.9	20.0	20.0	20.0	92.4	20.0	20.1	20.0	20.0
93.8	30.0	30.0	30.0	92.5	29.9	30.0	30.0	30.0
93.6	39.9	40.0	40.0	92.5	40.0	40.0	40.0	40.0
93.5	50.0	50.0	50.0	92.6	50.0	50.0	50.0	50.0
93.4	60.0	60.0	60.0	92.6	60.0	60.1	60.0	60.0
93.2	70.0	70.0	70.0	92.7	70.0	70.0	70.0	70.0
93.0	79.9	80.0	80.0	92.6	79.9	80.0	80.0	80.0
92.8	90.0	90.0	90.0	92.6	90.1	90.0	90.0	90.0
92.6	100	100.0	100.0	92.6	100.0	100.0	100.0	100.0

Methylation levels in various genomic contexts, such as CG, CHG and CHH (H is A/C/T only), are called by BatMeth and Bismark and validated against the oracle BS rate used in Sherman.

Table 4.2. Comparison of speed and unique mapping rates on three lanes of human BS data

Read file	Number of reads	Unique mapping (%) [a]		Running time (minutes) [a]	
		BatMeth	BS-Seeker	BatMeth	BS-Seeker
SRR019048	15,331,851	37.4	37.2	30	87
SRR019501	7,217,883	44.7	44.5	16	41
SRR019597	5,943,586	58.2	58.1	13	37

[a] Threshold of two mismatches used.

paired-end reads is approximately 300 bp, a pair of partner reads can be expected to be mapped correctly with a high probability if they are mapped concordantly within a nominal distance of 1,000 bp. The high number of such pairable reads (Figure 4.2b) indicates that BatMeth is accurate. Figure 4.2b also shows that BatMeth is fast.

We have also downloaded approximately 28.5 million reads sequenced by Illumina Genome Analyzer II on the human H1 embryonic cell line (GEO accession numbers

SRR019048, SRR019501 and SRR019597) [91]. We only compared BatMeth with BS-Seeker since BSMAP and Bismark are too slow. Furthermore, Krueger and Andrews [88] mention that Bismark is both slower and less likely to report unique hits than BS-Seeker. Table 4.2 shows the unique mapping rates and running times of BatMeth and BS-Seeker. In summary, BatMeth achieved the best mappability rate, lowest estimated false positive rate and was the fastest on real Illumina data.

4.3.4 Evaluation on the simulated SOLiD data

We generated 10,000 simulated reads, each having 51 color bases, that were randomly extracted from chromosome 1 of UCSC hg19 using the simulator from RMAP-bs [216]. RMAP-bs was used to convert the Cs in the reads, regardless of its context, to Ts at a uniform rate of 97% to simulate BS conversions. In addition, for each read, zero to two non-BS base mismatches were introduced with equal chance before the read was converted to color space. Lastly, sequencing errors were added at a uniform rate of 5% to the reads.

The simulated color reads were mapped using BatMeth, SOCS-B and B-SOLANA allowing resultant unique hits to have at most three mismatches. Precisely, BatMeth and SOCS-B allowed at most three non-BS mismatches while B-SOLANA did not discount BS mismatches. Figure 4.2c summarizes the results of the three programs together with the verification against the oracle set. BatMeth gave many more correct hits and fewer wrong hits than both SOCS-B and B-SOLANA. BatMeth can be made to offer a flexible tradeoff between unique mapping rates and speed. In the ‘default’ mode, BatMeth was found to be more sensitive (approximately 15%) and faster (approximately 10%) than the most recent published B-SOLANA. In the ‘sensitive’ mode, BatMeth was found to be more sensitive (approximately 29%) and slower (approximately two times) than B-SOLANA. In addition to producing approximately 15% to 29% more correct hits, BatMeth had a precision of 94.5% while that of B-

SOLANA and SOCS-B was 92.1% and 91.5%, respectively. These statistics show that BatMeth is an accurate mapper for color reads.

To illustrate that BatMeth can achieve better unbiased methylation calls for color reads than the best published method, B-SOLANA, we replicated the experimental settings of Figure 4.2c in [213] to compare the two programs; we used the same simulator (Sherman), the same number of reads (1 million), the same length of read (75 bp) and the same reference genome (NCBI37) for this comparison. We used Sherman to simulate 11 sets of data, from 0% to 100% of BS conversion at increments of 10%. Sherman emulates BS conversion by converting all Cs regardless of their genomic context with a uniform distribution. Default parameters were used for BatMeth and B-SOLANA. The graph produced by us for B-SOLANA shows the same trends as that presented in [213]. We further broke down the graphs as well as those in Figures 4.3a (BatMeth) and 3b (B-SOLANA), which show rates of methylation calling for various *in silico* methylation rates (0% to 100% at divisions of 10% of BS conversion) in different contexts (CG, CHG and CHH genomic contexts, where H stands for base A/C/T only) of the genomes, into separate series of data. Subsequently, we did a direct comparison between BatMeth and B-SOLANA to show that BatMeth is better than B-SOLANA in all contexts of methylation calling, namely, CG (Figure 4.3c), CHG (Figure 4.3d), CHH (Figure 4.3e) and non-unique mapping rates (Figure 4.3f). To be exact, BatMeth was approximately 0.7%, 0.7% and 2.2% more accurate than B-SOLANA in the methylation callings of the CG, CHG and CHH sites, respectively, and had an average of approximately 9.2% more non-unique mappings than B-SOLANA on the tested data sets.

4.3.5 Evaluation on the real SOLiD data

We downloaded about 495 million reads sequenced by AB SOLiD system 3.0 (Sequence Read Archive (SRA) accession number SRX062398) [208] on colorectal cancer. Since SOCS-B is not efficient enough to handle the full data set, 100,000

reads were randomly extracted from SRR204026 to evaluate BatMeth against SOCS-B and B-SOLANA. The mismatch threshold used was 3.

Table 4.3 compares the unique mapping rates and running times between BatMeth, SOCS-B and B-SOLANA. Note that BatMeth always has a higher unique mapping rate (from 39.6% to 52.1%; from fast to sensitive mode) than the next best method, B-SOLANA with 37.4%. At the same time, BatMeth maintained low rates of noise (from 0.47% to 1.75%; from fast to sensitive mode). Hence, it is still more specific than the other programs. In terms of running time, BatMeth fast mode is approximately 1.7 times faster and BatMeth sensitive mode is approximately 4 times slower than B-SOLANA. It was also observed that 3.26% of the resultant hits from B-SOLANA are duplicated; some of the reads were given two hit locations as B-SOLANA traded speed for checking the uniqueness of hits.

Based on the experiments performed, BatMeth's memory usage peaked at 9.3 GB (approximately 17 seconds of load time) for Illumina reads and 18.8 GB (approximately 35 seconds of load time) for color reads while BSMAP and BS-Seeker peaked at 9+ GB and Bismark peaked at 12 GB. SOCS-B peaked at 7+ GB and B-SOLANA peaked at 12 GB. In summary, the experiments in this section show that BatMeth is the fastest among all the compared programs. Furthermore, BatMeth also has the highest recovery rate of unique hits (exclusive of false positives) and the best accuracy among all the compared programs.

4.4 Materials and Methods

4.4.1 Methods for base reads

4.4.1.1 Problem definition and overview of the method

The problem of mapping BS reads is defined as follows. A BS treatment mismatch is defined as a mismatch where the aligned position is a T in the read and the

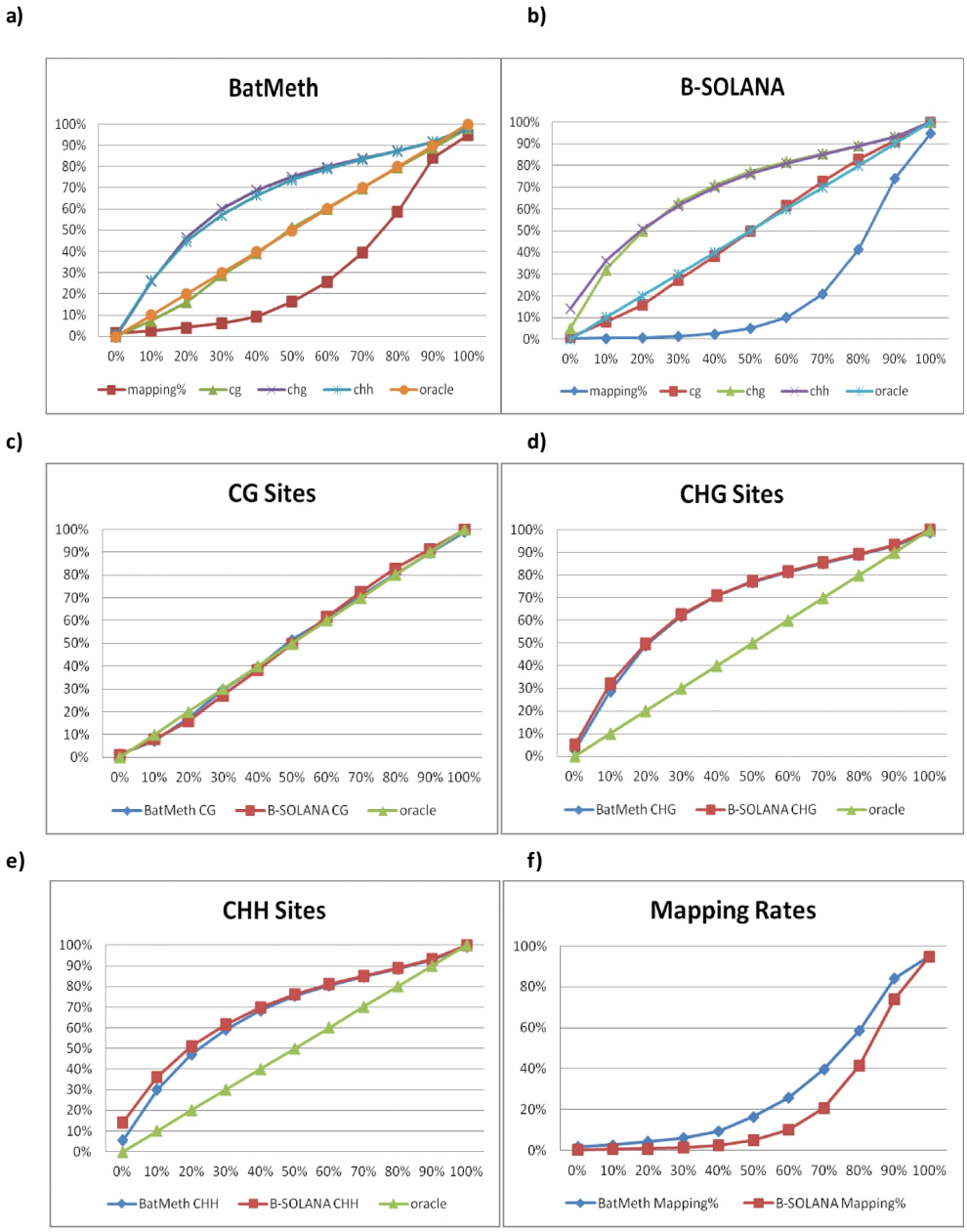


Figure 4.3. A total of 106, 75 bp long reads were simulated from human (NCBI37) genomes. Eleven data sets with different rates of BS conversion, 0% to 100% at increments of 10% (context is indicated), were created and aligned to the NCBI37 genome. (a-e) The x-axis represents the detected methylation conversion percentage. The y-axis represents the simulated methylation conversion percentage. (f) The x-axis represents the mapping efficiency of the programs. The y-axis represents the simulated methylation conversion percentage of the data set that the program is mapping. (a,b) The mapping statistics for various genomic contexts and mapping efficiency with data sets at different rates of BS conversion for BatMeth and B-SOLANA, respectively. (c-e) Comparison of the methylated levels detected by BatMeth and B-SOLANA in the context of genomic CG, CHG and CHH, respectively. (f) Comparison of mapping efficiencies of BatMeth and B-SOLANA across data sets with the described various methylation levels.

Table 4.3. Unique mapping rates and speed on 100,000 real color reads

SRR204026	Unique mapping (%)[a]	Estimated noise (%)[b]	Timing
BatMeth (fast)	39.6	0.47	77 s
BatMeth (default)	45.8	0.94	247 s
BatMeth (sensitive)	52.1	1.75	521 s
B-SOLANA[c]	37.4	2.06	130 s
SOCS-B[d]	28.3	4.55	~71 h

a We tabulated the unique mapping rates of the 100,000 reads. b The error rates are estimated from the number of reverse-strand mappings as stated by Equation 2 in Materials and methods. c Note that 3.26% of B-SOLANA's resultant reads are double-counted as B-SOLANA reported two hits for them. One of the two hits is assumed to be correct for the estimation of the noise rate of B-SOLANA. d Reverse-strand mapping is allowed by enabling G-A transitions in SOCS-B. BatMeth fast, default, and sensitive modes were run with -n0-N3, -n0-N4, -n0-N5 as parameters, respectively.

corresponding position in the reference genome is a C. Given a set of BS reads, our task is to map each BS read onto the reference genome location, which minimizes the number of non-BS mismatches.

The algorithm of BatMeth is as follows. BatMeth starts off by preparing the converted genome and does a one-time indexing on it. Next, low complexity BS reads are discarded; otherwise, we obtain counts of the hits for BS reads and discard the hits according to list filtering. After this, each of the retained hits will be checked for BS mismatches by ignoring C to T conversions caused by the BS treatment. BatMeth reports the unique hit with the lowest non-BS mismatches for each read. Figure 4.4a outlines the algorithm and we discuss the novel components that aid BatMeth to gain speed and accuracy below.

4.4.1.2 Converted genome

Similar to BS-Seeker and Bismark, we prepare a converted reference genome with all Cs converted to Ts. Since the plus and minus strands are not complementary after Cs are converted to Ts, we have to create two converted references where one is for the plus strand and the other is for the minus strand. Burrows-Wheeler transform (BWT) indexing of the two new converted references is done before the mapping.

4.4.1.3 Low complexity BS reads

BatMeth does not map BS reads with low complexity. The complexity of the raw read is computed as Shannon's entropy, and raw BS reads with a differential entropy $H < 0.25$ are discarded. In BatMeth, differential entropy is estimated from the discrete entropy of the histogram of A/C/G/T in a read. Depending on the design of the wet-lab experiment, the amount of reads being discarded by this entropy cutoff varies. In our experiments on Illumina reads, approximately 0.5% of the reads were discarded.

4.4.1.4 Counting hits of BS reads and list filtering

For those reads that pass the complexity filter, we first convert all Cs to Ts and map them against the converted genomes. In contrast to existing methods, BatMeth does not obtain the best or second best hits (for example, BS-Seeker and Bismark) from each possible orientation of a converted read and reports the lowest-mismatch locus to be the resultant hit for a read. In the case of hyper-methylation, the correct hit may not be the best or second best hit as it might contain more mismatches. Thus, this approach will miss some correct solutions. BatMeth also does not enumerate all hits like BSMAP, which is slow. Instead of mapping the reads directly, BatMeth counts the number of hits where the read or its reverse complement can occur on the two converted genomes using an in-house short read mapper, BatMis Aligner [219]. Table 4.4 shows the four ways of aligning the converted reads onto the converted genomes, which yield four counts of hits.

Table 4.4. Possible ways to map a BS read onto the converted genome

	Reference (C→T)	RC reference (C→T)
Read (C→T)	Count 1	Count 2
RC Read (C→T)	Count 3	Count 4

RC, reverse-complement.

Out of the four counts on the four lists, only one list contains the true hit. List filtering aims to filter away those spurious lists of hits (represented by the counts) that are unlikely to contain the true hit. Note that a read can appear to be repetitive on one

strand but unique on the opposite strand of the DNA. Hence, if a list has many hits (by default the cutoff is set to be 40 hits) with the same number of mismatches, we discard such a list since it is likely to be spuriously reported for one strand of the reference genome. Another reason for rejecting such lists is that they may contain hits that may be of the same mismatch number as the hit that is unique on the opposite strand, rendering all hits as ambiguous.

Apart from improving the uniqueness of the putative resultant hit among all reported hits of a BS read, filtering also reduces the number of candidate hits that need to be checked.

This improves the efficiency of the algorithm. For example, consider the simulated BS-converted read 'ATATATATGTGTATATATATATATATATATGTGTATA TATATGTGTGTATATATATATA TATATATGTATATAT' being mapped onto the converted hg19 genomes as discussed earlier. We obtained four counts of 1, 0, 40 and 40 hits by mapping the converted reads onto the converted genomes. The last two lists are filtered away since they have too many hits, leaving us to check only one hit instead of 81 for BS mismatches. Since the data are simulated, the unfiltered hit is found to be the correct unique hit for this read, which the other mappers cannot find.

Table 4.5. Cutoffs for list filtering on simulated reads from the Results section

List size	Mismatch counting in seconds[a]	Correct hit	Wrong hit	Total hit
20	136	901,164	1,516	902,680
40	165	901,160	1,462	902,622
60	191	901,165	1,454	902,619
100	279	901,166	1,448	902,614
200	475	901,166	1,447	902,613
500	1,197	901,167	1,450*	902,617
1,000	2,942	901,167	1,450*	902,617

Asterisks indicate increased false-positives produced with large list filtering cutoffs.

a)

1. Prepare the converted *Reference Indexes for both plus and minus strands.*
2. **For** each input read **do**
3. Prepare the plus and minus conversions of the read
4. Count the number of hits using 4 possible ways to map the converted reads on the *Converted Genome*
5. Using *List Filtering*, we filter the lists whose number of hits > cutoff
6. For each hit in the unfiltered lists, compute the number of mismatches ignoring the BS-treatment mismatches.
7. **If** the least mismatch hit is unique **then**
8. Report its location.
9. **Else**
10. Report it as non-unique.
11. **EndIf**
12. **EndFor**

b)

1. Prepare 4 *Reference Indexes* for the two fully-converted color genomes and the two non-CpG converted color genomes.
2. **For** every read **do**
3. Count the number of hits for 2 possible ways to map the read and its reverse on the fully-converted color genomes
4. Apply *List Filtering* on the counts obtained from Step 3.
5. Apply *Mismatch Stage Filtering* to the unfiltered list from Step 4.
6. Apply *Conversion of Bisulfite Color reads to Base reads* to the hits from Step 5.
7. Determine the *Color Mismatch Counts for the hits* on the ordered hits from Step 6.
8. **If** the least mismatch hit is unique **then**
9. Report it. Goto Step 14.
10. **ElseIf** the least mismatch hit is non-unique
11. Reported it as non-unique. Goto Step 14.
12. **ElseIf** no hits found on fully-converted color genomes **then**
13. Repeat Steps 3 to 14 with non-CpG-converted color genomes
14. **EndIf**
15. **EndFor**

Figure 4.4. Outline of the mapping procedure. (a) Mapping procedure on Illumina BS base reads. (b) Mapping procedure on SOLiD color-space BS reads.

Table 4.5 shows the effect of using list filtering on the same set of simulated data from the evaluation on the simulated Illumina reads. We ran BatMeth with different cutoffs for list filtering and we can see that the time taken increased linearly with increasing cutoffs for list filtering while sensitivity and accuracy dropped. With large cutoffs such as ≥500 (marked by asterisks in Table 4.5), the number of wrong hits increased while sensitivity still continued to drop. Thus, we have chosen a cutoff of 40 for a balance of speed, sensitivity and accuracy. (Disabling list filtering will cause BatMeth to check through all the reported candidate locations for a read and will slow BatMeth down by approximately 20-fold, as shown in Table 4.5.)

4.4.2 Methods for color reads

4.4.2.1 Overview of the method

Due to the di-nucleotide encoding and sequencing errors in SOLiD color reads, a naïve conversion from color space to base space is hardly possible without errors. As a color error in a read will introduce cascading base-space errors, we cannot use the method described in 'Methods for base reads' above to map BS color reads. This section describes how we aim to map each BS color read uniquely to the reference genome while minimizing the number of non-BS treatment mismatches.

The algorithm of BatMeth is as follows. BatMeth starts by preparing the converted genome and non-CpG converted genome and does a one-time BWT indexing on them. For every color read, we do a 'counting hits of BS color reads' for it on the references and discard the list of hits according to list filtering. After applying mismatch stage filtering, the unfiltered hits are converted to base space as described in 'Conversion of bisulfite color reads to base reads' below to allow for the checking of BS mismatches. The color mismatch count for the retained hits is then determined and the unique locus with the lowest mismatch count reported; otherwise, no hits will be reported for this read. We have also utilized additional heuristics, such as fast mapping onto two

indexes and handling hypo- and/or hyper-methylation sites to speed up and improve the accuracy of BatMeth, which we discuss below. All the components, namely list filtering, mismatch stage filtering, conversion of BS color reads to base reads, color mismatch count, fast mapping onto two indexes and handling hypo- and/or hyper methylation sites differ from existing methods. Figure 4.4b outlines the algorithm and shows how the components are assembled for SOLiD color-space BS read mapping.

4.4.2.2 Non-CpG converted genome

The reference genome and its reverse-complement were first prepared by converting all its Cs to Ts as described in the base reads mapping procedures; then, the two converted genomes are encoded into color space. These two genomes are called fully converted color genomes. In addition, the reference genome and its reverse-complement are similarly converted except that the Cs in CpG are left unchanged. We call these the non-CpG converted color genomes. Finally, the BWT indexes for these four color genomes are generated.

In the algorithm, the BS color reads will be mapped to the fully converted color genomes to identify unique hits first; if this fails, we will try to map the reads onto the non-CpG converted color genomes and BatMeth will label which reference a hit is from.

The reason for using the non-CpG converted genome is that the conversion step for BS color reads is different from that for Illumina. In Illumina reads, the C-to-T mismatches between the raw BS reads and the reference genome are eliminated by converting all Cs to Ts in both the reads and the reference genomes. However, we cannot make such a conversion in BS color reads as we do not know the actual nucleotides in the reads. Based on biological knowledge, we know that CpG sites are expected to be more methylated [220]. Hence, such conversion reduces the number of mismatches when the color reads are mapped onto the reference genome in color

space. This aids in gaining coverage in regions with high CpG content. Thus, BatMeth maps BS reads to both hyper- and hypo-methylation sites.

4.4.2.3 Counting hits of BS color reads and list filtering

Unlike sequencing by Illumina, SOLiD only sequences reads from the original BS-treated DNA strands. During PCR amplification, both strands of the DNA are amplified but only the original forward strands are sequenced. Subsequently, during the sequencing phase, reverse-complement reads are non-existent as a specific 5' ligated P1 adaptor is used. As such, matches to the reverse-complement of the BS-converted reference genome are invalid.

In other words, although a BS color read has four possible orientations to map on the non-CpG converted color genomes (or the fully converted color genomes), only two orientations are valid as opposed to the four orientations in the pipeline on Illumina reads

Table 4.6. Possible ways to map a BS color read onto the converted color genome

	Reference (C→T)	RC reference (C→T)
Read	Count 1	Invalid
RC read	Invalid	Count 4

RC, reverse-complement.

(Table 4.6). As opposed to the mapping of Illumina reads, it is not preferred to do a naïve conversion of color reads to base space prior to mapping. Figure 4.1a shows that a single base call error in an Illumina read will introduce one mismatch with respect to the reference. However, Figure 4.1b shows that a single base color call error in a color read will introduce cascading base mismatches instead of just one color mismatch if we are to map the color read as it is onto the reference in color space.

Thus, we will need to do a primary map onto a converted genome with a higher mismatch parameter (by default, 4) than what we usually use for Illumina BS reads as

a BS mismatch will introduce two adjacent color mismatches (see Figure 4.1c for an example of BS-induced adjacent color mismatches). Similar to mapping Illumina reads, we count the number of possible hits from the two valid orientations. Then, the list filtering step is applied to filter the lists with too many hits (by default, more than 10). (Note that this property also helps us to estimate the noise rate; we discuss this further in 'Noise estimation in color reads' below.)

4.4.2.4 Conversion of bisulfite color reads to base reads

After the color BS reads are aligned to the reference genome, we can convert the color BS reads to their most-likely nucleotide equivalent representation. In the context of BS mapping, we discount all the mismatches caused by BS conversions.

We use a DP formulation as presented in [100] to convert color reads to base reads except that the costs for BS-induced mismatches have to be zeroed when the reference is C and the read is T. This conversion is optimal and we use the converted base read to check against the putative genomic locations from list filtering to interrogate all mismatches in the read to determine if they are caused by BS conversion, base call error or SNP.

4.4.2.5 Color mismatch count

After converting each color read to its base-space equivalent representation, we can calculate the number of base mismatches that are actually caused by BS treatment in the color read. Figure 4.2d shows two different types of adjacent color mismatches that are caused by BS conversion (left) and non-BS conversion (right). For BS-induced adjacent mismatches, we assign a mismatch cost of 0 to the hit. For non-BS-induced adjacent mismatches, we assign a mismatch cost of 1 to the hit.

To be precise, we consider a color read as C[1..L], where L is the read length, and let B[1..L-1] be the converted base read computed from the DP described previously and

mm[i] as a mismatch at position i of C, which is computed using Equation 4.1. The mismatch count of C is computed as mm[1]+…+mm[L-1], where:

$$mm[i] = \begin{cases} 1, if\ C[i]\ and\ C[i+1] \text{ are color mismatches, } B[i] \text{ is non-BS mismatch} \\ 0, otherwise \end{cases}$$

Eq (4.1)

4.4.2.6 Mismatch stage filtering

We have developed a set of heuristics to improve the rate of finding a unique hit among the set of candidate hits. First, we sort and group the initial hits by their number of color mismatches; then, we try to find a unique hit with the minimum non-BS-mismatch count within each group of hits.

As the bound of color mismatches is known, we can apply a linear time bucket sort to order all the candidate hits according to their mismatch counts. The group of initial mapping loci with the lowest mismatch number is recounted for their number of base mismatches using the converted read in base space obtained from the previously discussed DP formulation. If a unique lowest base mismatch hit exists among them, we report this location as unique for this read. Otherwise, we proceed to recount the base mismatches for the group of mapping loci with the next highest color mismatch count. We continue this procedure until a unique hit is found or until there are no more color-space mismatch groups to be examined. A unique hit must be unique and also minimizes the base mismatch counts among all previously checked hits in the previous groups.

Mismatch stage filtering enables us to check less candidate hits, which speeds up the algorithm. It also improves the unique mapping rate as there are less ambiguous hits within a smaller group of candidate hits.

When the above components are applied, the mapping rates on SOLiD data improve progressively as seen below. By using Equation 1 to count color mismatches,

BatMeth was able to increase the number of unique mappings by approximately 9% and by employing mismatch stage filtering, unique mapping rate is approximately increased by another 3%. With this increase in unique mappings of approximately 12%, BatMeth had an estimated noise level of approximately 1% as based on Equation 2 while B-SOLANA and SOCS-B had estimated noise levels of approximately 2.06% and 4.55%, respectively, on the same set of 100,000 reads. These statistics agree with the results on the simulated data and indicate that BatMeth is capable of producing low-noise results.

4.4.2.7 Fast mapping onto two indexes

As mentioned in the 'Non-CpG converted genome' section above, we map BS color reads onto four converted references, two of which have their Cs converted to Ts at non-CpG sites and the other two have all their Cs converted to Ts. It was observed that mappings on both non-CpG converted and fully converted references highly coincide with each other with an approximately 95.2% overlap. Due to this observation, we try to map onto the fully converted reference first to give us a mapping to regions of hypo-methylation status. If there are no mappings found on the fully converted references, then BatMeth maps the same read again onto the non-CpG converted references, which biases hyper-methylation sites. This allows the simultaneous interrogation of canonical CpG hyper-methylation sites with reduced biased mapping on the fully converted genome. BatMeth also labels each hit with the type of converted references it was mapped to. Overall, this approach can save time by skipping some scanning of the non-CpG-converted references.

4.4.2.8 Handling hypo- and/or hyper-methylation sites

With prior knowledge of the methylation characteristics of the organism to be analyzed, different *in silico* conversions to the reference can be done and the best alignments can be determined from the combined set of results of different mapping runs. BatMeth uses two types of converted genomes to reduce mapping biases to both

hyper- and hypo-methylation sets. Since the two sets of hits from the two genomes coincide to a large extent, we can save time by scanning a read on one genome with a much lower mismatch number than on the other genome.

BatMeth allows users to choose the mismatch number they want to scan on each of the two types of genomes. We now introduce M1 and M2 (capped at 5) as the mismatch numbers used in the scans against the fully converted and non-CpG-converted genomes, respectively. For the best sensitivity, BatMeth scans at M1 = M2 = 5 for both hyper- and hypo-methylation sites. For the highest speed, BatMeth scans at [M1 = 0, M2 = 3] and [M1 = 3, M2 = 0], which will perform biased mapping to hyper- and hypo-methylation at CpG sites, respectively. Figure 4.2c shows the results of running the various modes of BatMeth (fast, default and sensitive) on a set of 10,000 simulated color reads.

4.4.2.9 Noise estimation in color reads

To estimate noise rates, we map the real reads in their two possible orientations onto the genome. If a hit is found for a read from the original strands of the genome, we try to map the same read onto the complement strand of the genome too. If a lower mismatch hit can be found from the complement strand of the genome, then we mark the result for this read as noise. We use the proportion of marked reverse-complement unique mappings to estimate the noise level, given by Equation 4.2:

$$err = \frac{\#\ of\ reverse-complement\ mappings}{\#\ of\ mappings}$$

Eq (4.2)

4.4.2.10 Handling ambiguous bases

For base reads, non-A/C/G/T bases are replaced by A so they will not affect the callings of methylation sites. Similarly, color reads with non-A/C/G/T bases are replaced with 0. Non-A/C/G/T bases on the reference genome are converted to A to

avoid affecting downstream methylation callers. We have avoided converting them to random nucleotides as it may produce false hits in regions containing ambiguous bases. We mapped 1 million 75 bp reads and have seen reads being mapped to poly-N regions. This can be mostly attributed to the reduced alphabet size, from four to three, due to BS conversions.

4.5 Prediction of Imprinting Genes using BatMeth

From genome-wide mappings of bisulfite data, Laurent et al and Lister et al [218, 221] showed that the pattern of fully-methylated Cs changes throughout the cell differentiation process and affects the gene expressions. Hansen et al [208] found that different cancer types have variable profiles in their methylomes, which may contribute to tumor heterogeneity. Stadler et al [222] showed that transcription factor binding affects DNA methylation and may create low-methylation regions (i.e., regions with low percentages of fully-methylated Cs).

Partially-methylated Cs are as important as fully-methylated Cs. Gene promoter regions enriched with partially-methylated Cs (i.e. high percentage of partially-methylated Cs over the all Cs in the regions) are functionally different from regions enriched with fully-methylated Cs. For example, X-chromosome inactivation, in which one allele of chromosome X is inactivated, is believed to be related to partial DNA methylation [223]. In another example, partial methylation is known to be related to imprinting genes [224], in which only one parental-specific allele of the gene is expressed. The effect of partial methylation on imprinting genes was tested in mouse models [225] and mutant mice that were deficient in DNA methyltransferases activity [200].

Abnormal pattern of partially-methylated Cs causes disease and affects the progression of cancers [226-229]. For example, it was showed that partial methylation of $p14^{ASF}$ affects its gene expression in colorectal cancer (CRC) [230].

Prader-Willi Syndrome and Angelman Syndrome are associated with partial methylation and can be detected by parent-of-origin specific DNA methylation [231-233].

Although the patterns of partially-methylated Cs are important, little attention has been put on partially-methylated Cs. In this work, we generated profiles of partially-methylated Cs from the genome-wide DNA methylation bisulfite sequencing datasets for 18 cell-lines and tissues. We observed that partially-methylated Cs were widespread in the genome. Moreover, partially-methylated Cs were clustered in kilobase regions to form partially-methylated regions (pMRs). Some pMRs are conserved in most of the studied cell-lines, while some are gender-specific or differentiated-cell-specific. These pMRs mark genes with intermediate level of expressions and are enriched with active histone modifications and transcription factors. The gender-specific pMRs are enriched in chromosome X with X-linked inheritance function; the differentiated-cell-specific pMRs show development related functions. Furthermore, we observed that the conserved pMRs are significantly overlapped with known imprinting genes, which can be a method to identify new imprinting genes.

4.5.1 Results

4.5.1.1 Partially-methylated Cs from 18 individual bisulfite libraries

We studied 18 bisulfite sequencing libraries [218, 221, 234-238], including 8 libraries from female samples and 10 libraries from male samples (see Appendix Table A1.1 for details of the libraries used, including gender information and embryonic-cell-differentiated-cell information). They were re-mapped with BatMeth [239] to human reference genome hg19. The methylation level of each individual C is defined as the percentage of reads covering the C that are not modified by the bisulfite treatment. The percentage ranges from full methylation (100%) to no methylation (0%). The

DNA methylation level of each C is stratified into five categories (Figure 4.5a): 1) methylated (as M, 80%-100%), 2) between methylated and partially-methylated (as Mp, 60%-80%), 3) partially-methylated (as P, 40%-60%), 4) between partially methylated and un-methylated (as pU, 20%-40%), and 5) un-methylated (as U, 0%-20%). To validate our bisulfite mapping and methylation callings, we compared the methylation callings of cell-line H9 from bisulfite sequencing [218] and from Illumina Infinium Human-Methylation27 BeadChip microarray (as 27K array in the following text). 27K array heatmap in Figure 4.5b shows the ratios of the observed counts and the expected counts in all 25 combinations. The darker colors (as high observed/expected ratios) along the diagonal of the heatmap suggest that the methylation callings from our bisulfite sequencing are concordant with the callings from 27K array. The high observed/expected ratio for the category with "P" callings from both bisulfite sequencing and 27K array shows that the partial methylation callings from bisulfite reads are quite reliable. Our following studies are mainly focused on partially methylated Cs (i.e. Cs with methylation level between 40%-60%) from bisulfite sequencing. The numbers of partially methylated Cs in CpGs from individual cell-lines are shown in Figure 4.5c (see Appendix Table A1.2 for more information about bisulfite mapping and partial-methylation calling).

4.5.1.2 Individual partially-methylated Cs widely spread in the genome

First, we characterized the individual partially-methylated Cs. Figure 4.6a shows the distributions of partially-methylated Cs in CpA, CpC, CpG and CpT compositions. Similar to fully-methylated Cs, partially-methylated Cs are significantly enriched in CpGs, while vast majority of the partially-methylated Cs are not in CpG islands (Figure A1.1). Apart from CpGs, partially-methylated Cs are enriched in CpAs of embryonic stem cells H1, H9 and induced-pluripotent-stem (iPS) cells, but not in differentiated cells.

This is similar to the previous observations that non-CpG methylated Cs are enriched in embryonic stem cells [218, 221]. When partially-methylated Cs in CpAs were checked in details, CAGs were enriched in partially-methylated Cs at CpA sites from embryonic stem cells, while diminished in blood cells (Figure 4.6b). The functions of such non-CpG methylated Cs need further investigation. Since partially-methylated Cs are mainly from CpGs in most of the cell-lines, the following analyses are based on partially-methylated Cs from CpGs.

Second, we checked the profiles of the partially-methylated Cs along the chromosomes. Figure 4.6c shows the profile of percentages of partially-methylated Cs over the covered Cs in 100Kb bins along chromosome 1 from cell-lines H1, H9 and IMR90. The partially-methylated Cs spread widely along chromosome 1 (and the same for other cell-lines and tissues, see Figure A1.2, and other chromosomes). In fact, the number of partially-methylated Cs in each chromosome is proportional to the chromosome length (see Figure A1.3). We also observed that, in the embryonic stem cells, 3%-6% of Cs per 100Kb are generally partially-methylated, while in the differentiated cell-lines, higher percentages (6%-10%) of the Cs per 100Kb are partially-methylated (Figure A1.2). This observation is consistent with Figure 4.5c, which showed that differentiated cells have more partially-methylated Cs.

Figure 4.6d is a heatmap showing the correlation of partially-methylated Cs among different cell-lines. In general, the correlations of partial methylation profiles of Cs among embryonic stem cells are higher than those between embryonic stem cells and differentiated cells. The hierarchical clustering of the correlation matrix shows the separation of the cell lineages. Roughly speaking, there are two clusters: one cluster is for the embryonic stem cells and the likes, and another cluster is for the differentiated cells. H1 and H9 are the male and female embryonic stem cell respectively. H1_BMP4 and H1_mesendoderm_BMP4 are H1 derived embryonic-stem-like cells. H1NPC is a neural progenitor cell derived from H1. Sperm cells are similar to the

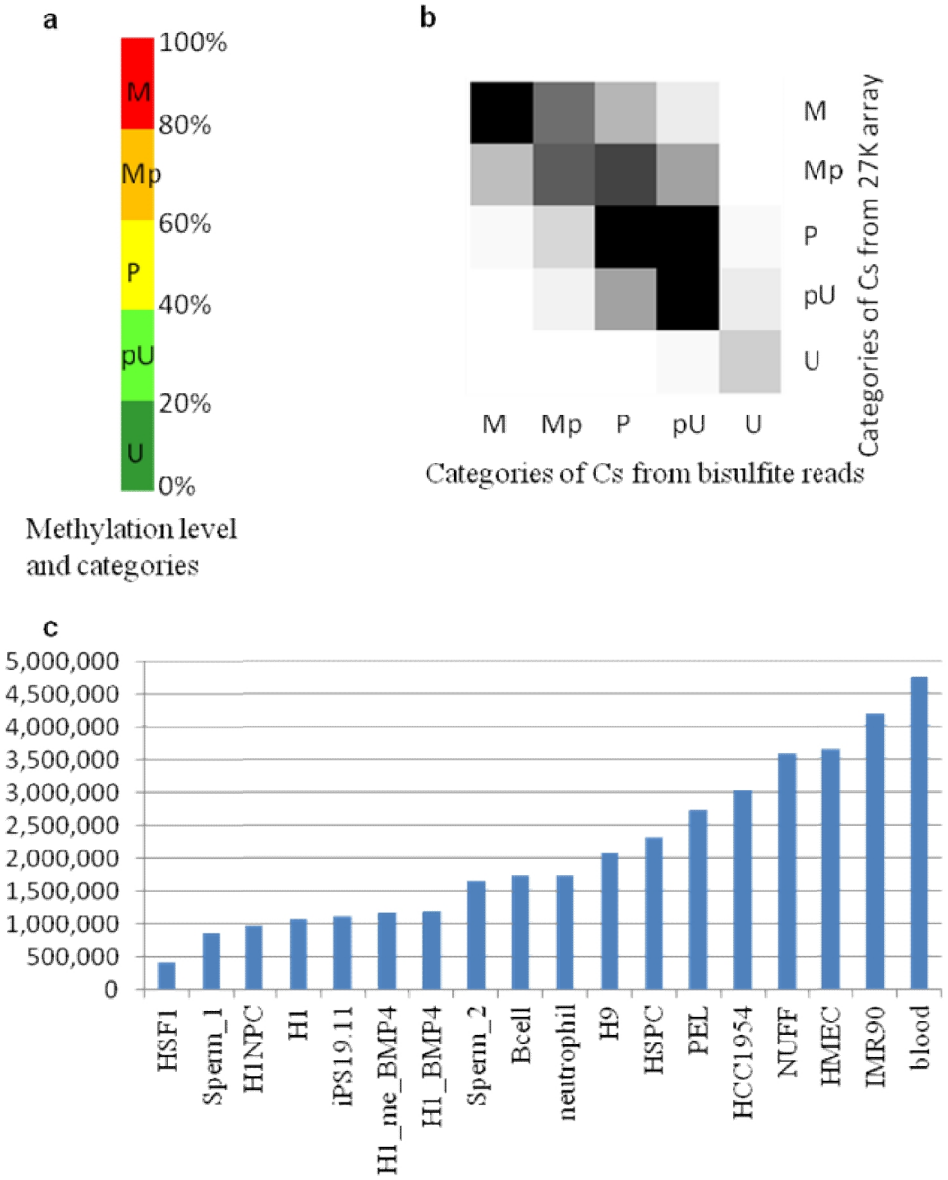


Figure 4.5. Partial methylation callings. a) Categories of methylation levels; b) Validation of DNA methylation callings from bisulfite sequencing with 27K DNA methylation array in embryonic stem cell H9. The darker the color, the higher ratio from the observed consistent DNA methylation callings to the randomly expected consistent callings; c) Numbers of partially-methylated Cs from individual cell-lines and tissues.

embryonic stem cells. HSF1 cell-line is an embryonic stem cell and has a low mapping rate (due to extensive Ns in the reads). Cell-line iPS19.11 is an induced pluripotent stem cell. The other cell-lines are in the differentiated cell cluster.

4.5.1.3 Regions enriched with partially-methylated Cs across the samples overlap with imprinting genes

Although the numbers of partially-methylated Cs in the individual chromosomes are approximately proportional to the chromosome lengths (Figure A1.3), Figure 4.6c shows that partially-methylated Cs are not uniformly distributed along the chromosomes – some partially-methylated Cs are clustered in small regions. Based on this observation, we partitioned the genome into 5Kb bins to identify the regions enriched with partially-methylated Cs. For every bin in each cell-line, we computed its percentage of partially-methylated Cs over the covered Cs (precisely, the percentage of Cs that are partially methylated within the bin). Then, the average percentage of partially-methylated Cs of each bin among all cell-lines was computed. Figure 4.7a shows the QQ plot of the average percentage of partially-methylated Cs of all the bins. From the plot, we saw that some bins were significantly enriched with partially-methylated Cs. 0.15 was determined as the cutoff to identify bins enriched with partially-methylated Cs. We further merged the bins enriched with partially-methylated Cs if they were less than 5Kb apart and filtered the bins that overlapped by 40% or more with RepeatMasker regions [240]. This resulted in 94 regions from autosomes as the conserved partially-methylated regions (pMRs). Figure 4.7b shows the conserved pMR with the highest average percentage of partially-methylated Cs. This region locates around the promoter of the known imprinting gene *GNAS*. Figure 4.7c and 4.7d show two more pMRs which are around known imprinting genes *PEG10*, *MAGEL2* and *NDN*. In particular, the pMR around the imprinting genes *MAGEL2/NDN* is located in the Prader-Willi syndrome deletion region.

The above examples show that conserved pMRs are good candidates of imprinting genes. In fact, the number of conserved pMRs is approximately the same as the

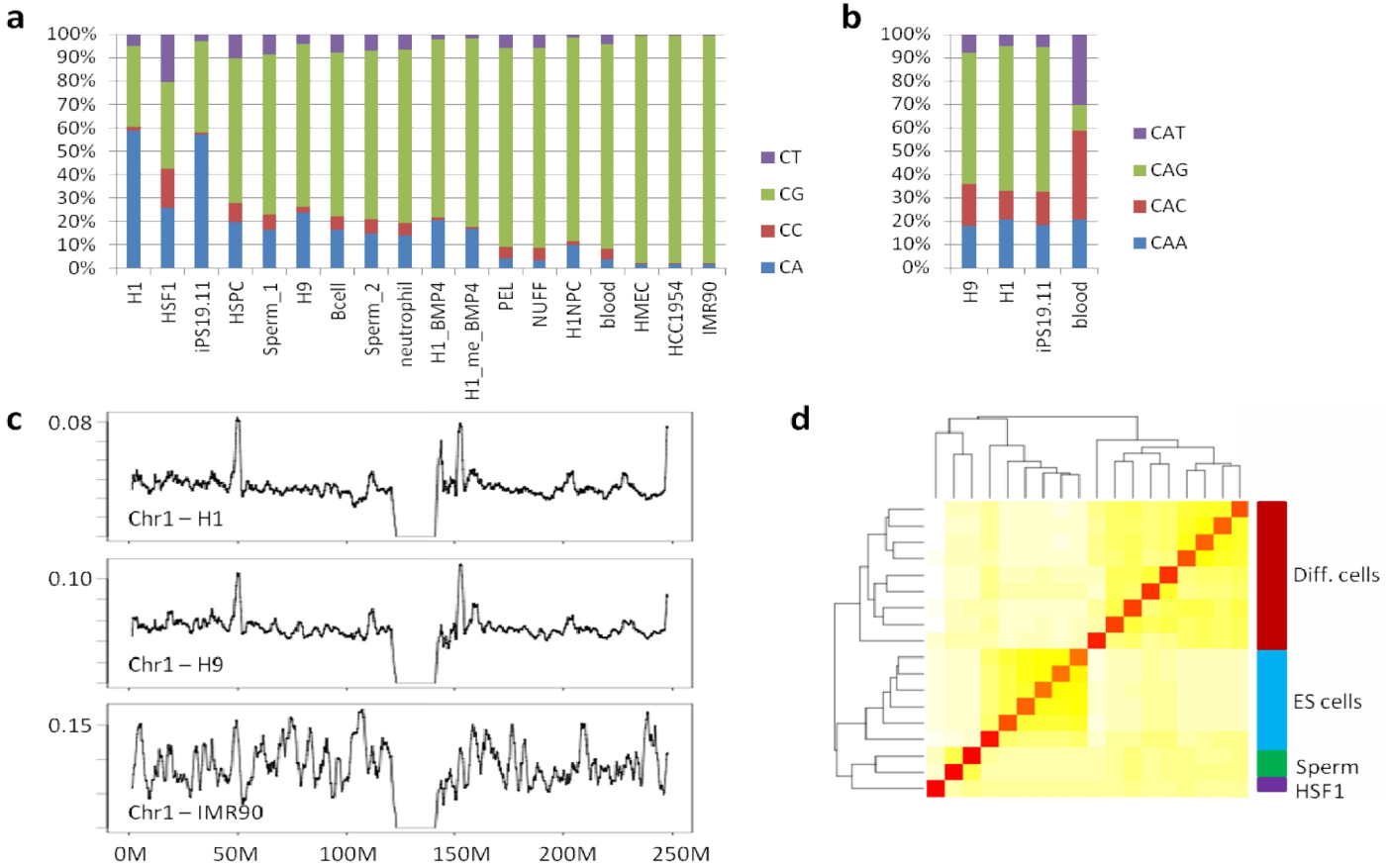


Figure 4.6. Genomic profile of partial methylated C. a) Distributions of partially-methylated Cs in CpAs, CpCs, CpGs and CpTs from individual cell lines. The cell lines are sorted by the proportion of partially-methylated Cs from CpGs. b) Distributions of partially-methylated Cs in CAAs, CACs, CAGs and CATs from embryonic stem cells H1, H9, induced pluripotent stem cell iPS19.11, and blood cells. Cell-lines H1, H9, and iPS19.11 are enriched with CAGs at CpA sites, while blood cells are depleted with CAGs at CpA sites. c) Profile of partially-methylated Cs from cell-lines H1, H9 and IMR90 along chromosome 1 in 100Kb bins. d) Hierarchical clustering of the profile of partially-methylated Cs from different cell lines. ES cluster includes H9, H1, H1NPC, H1_BMP4, H1_me_BMP4, and iPS19.11. Sperm cluster includes two sperm replicate DNA methylation data. HSF1 is an embryonic stem cell with low mapping rate (due to extensive “N”s in the reads). Other cell lines are in the differentiated cell cluster (Diff. cells).

estimated number of the imprinting genes, which is around 100-200 [241]. Gene Ontology (GO) analysis with GREAT [242] also showed that the conserved pMRs are enriched with genomic imprinting. Then, we overlapped the conserved pMRs with 98 known imprinting genes. 18 out of 94 conserved pMRs overlap with known imprinting genes. To test if the overlap is statistically significant, we randomly simulated the same number of regions from the genome and calculated the overlapped regions with known imprinting genes. We repeated the simulation 1000 times and found the average number of overlapped regions was 0.74 in the simulation, which is much smaller than 18, which is the actual number of conserved pMRs overlapped with imprinting genes (empirical p-value = 0).

Especially, 11 out of top 20 conserved pMRs overlap with known imprinting genes (Table 4.7). From the remaining 9 pMRs that did not overlap with the set of known imprinting genes, the *FANK1* gene was validated to have allele-specific methylation in blood [235] and another gene *MAP2K3* was validated using strain-biased expression [243]. Another region, *chr11:2,020,000-2,025,000*, is just 1Kb upstream of a well-characterized imprinting gene *H19*. Taken together, this list is highly enriched with imprinting genes.

4.5.1.4 Cell specificity of partial methylations

To study if partially-methylated Cs mark genes with cell specificity, we used t-test (see Method) to identify pMRs which were gender-specific and differentiated-cell-specific. 210 regions were identified as gender-specific pMRs and 272 regions were identified as differentiated-cell-specific pMRs. Figure 4.8a shows the heatmap of the percentages of partially-methylated Cs from gender-specific pMRs in different cell-lines. Majority of such gender-specific pMRs (201 out of 210 regions) are from chromosome X, and they have higher percentage of partially-methylated Cs in the female cell-lines. The remaining gender-specific pMRs are from chromosomes other

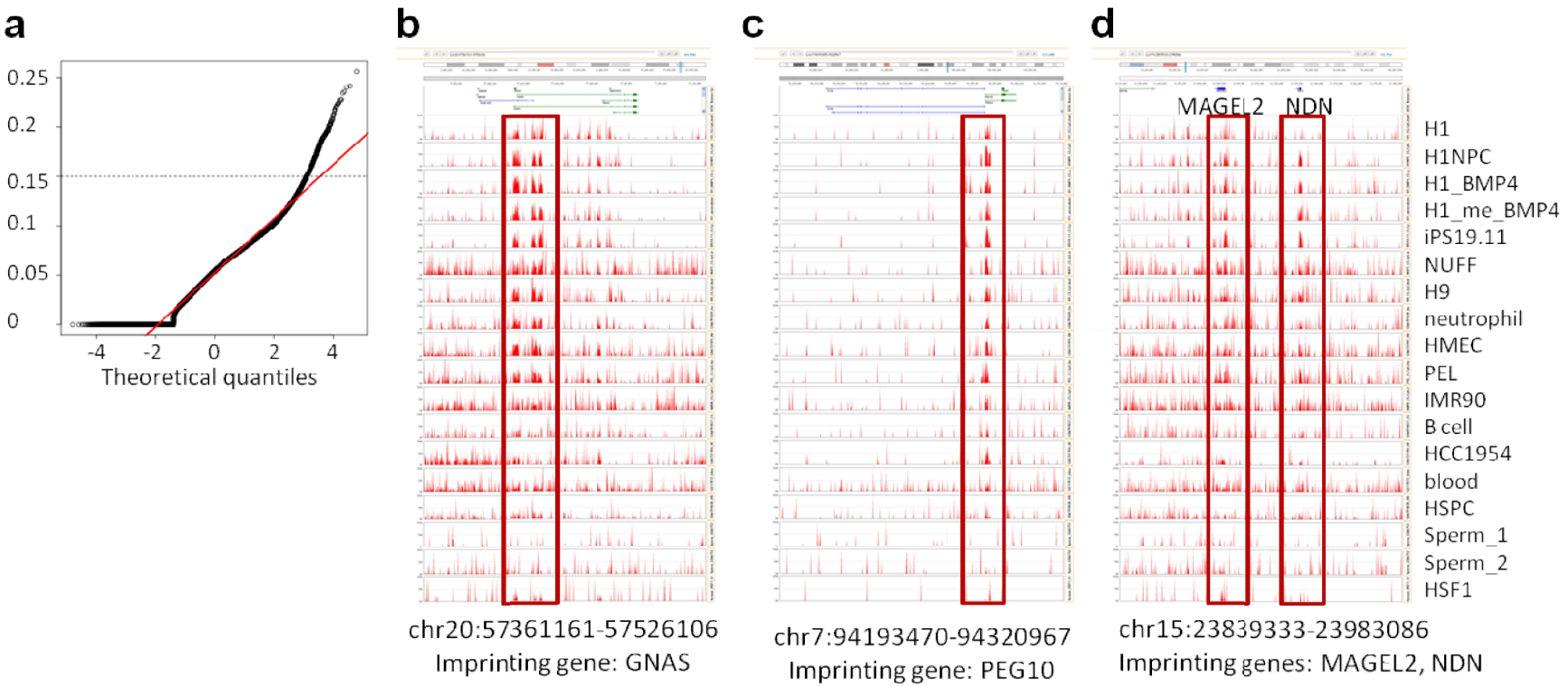


Figure 4.7. Partial methylation across samples. a) QQ plot of the average percentage of partially-methylated Cs in 5Kb bins across the genome from all studied cell-lines and tissues. There are regions without partially-methylated Cs as percentage 0. (b-d) Screenshots around imprinting genes *GNAS*, *PEG10*, and *MAGEL2/NDN* enriched with partially-methylated Cs in the studied cell lines.

Table 4.7. Top pMRs across cell types

genomic regions with partial methylation across samples	*genes overlapped at promoter regions(known imprinting genes in BOLD)*
chr20:57,415,000-57,435,000	**GNAS**
chr7:94,285,000-94,290,000	**SGCE**;**PEG10**
chr6:57,360,000-57,420,000	**PRIM2**
chr7:130,130,000-130,135,000	**MEST**;**MESTIT1**
chr10:135,490,000-135,495,000	DUX4L3;DUX2
chr21:9,890,000-9,920,000	---
chr19:54,040,000-54,045,000	**ZNF331**
chr11:2,720,000-2,725,000	**KCNQ1**;**KCNQ1OT1**
chr17:21,245,000-21,255,000	---
chr15:25,200,000-25,205,000	**SNRPN**;**SNURF**
chr1:161,420,000-161,425,000	TRNA_Glu;TRNA_Gly;TRNA_Asp;TRNA_Leu
chr20:29,625,000-29,635,000	MLLT10P1;FRG1B
chr20:42,140,000-42,145,000	**L3MBTL1**
chr11:2,020,000-2,025,000	AK311497;MIR675;**H19**
chr17:21,200,000-21,230,000	MAP2K3
chr10:127,575,000-127,590,000	DHX32;U2;FANK1
chr7:61,050,000-61,060,000	---
chr6:57,205,000-57,210,000	**PRIM2**
chr15:23,930,000-23,935,000	**NDN**
chr19:57,345,000-57,355,000	**ZIM2;PEG3;MIMT1**

than chromosome X, with higher percentage of partially-methylated Cs in male cell-lines. Figure 4.8b shows an example gender-specific pMR which is around the gene *AMELX*. *AMELX* is known to associate with X-linked forms of amelogenesis imperfecta. Gene Ontology analysis with GREAT [242] showed that the gender-specific pMRs are highly enriched for X-linked inheritance and intellectual disability. This means that genes around these pMRs are important for gender-related functions.

Figure 4.8c shows the heatmap of the percentages of partially-methylated Cs from differentiated-cell-specific pMRs in different cell-lines. Figure 4.8d shows an example differentiated-cell-specific pMR which is around the *PCDHB* gene clusters. *PCDHB* gene clusters are related to Wilms' tumor, which is a pediatric tumor of the kidney with failure of the fetal developmental program [244]. In summary, both the

gender-specific pMRs and differentiated-cell-specific pMRs are enriched with cell-specific functions.

4.5.1.5 Characterization of partially-methylated regions with methylated Cs, histone modification marks and gene expressions

With conserved pMRs, gender-specific pMRs and differentiated-cell-specific pMRs, we characterized them with the profile of fully-methylated Cs, histone marks and gene expressions. First, we checked the profile of fully-methylated Cs in the identified pMRs. Figure 4.9a shows the percentages of fully-methylated Cs in the whole genome, the conserved pMRs, the gender-specific pMRs and the differentiated-cell-specific pMRs. For the whole genome, the average percentages of fully-methylated Cs from all cell-lines and cell-line H9 are generally high (the medians are around 80%). The conserved pMRs have the lowest percentage of fully-methylated Cs (around 50%-60%). The gender-specific pMRs have higher percentage of fully-methylated Cs in the male samples than in the female samples. The differentiated-cell-specific pMRs have higher percentage of fully-methylated Cs in embryonic stem cells and the likes than those in the differentiated cells. This indicated that the conserved pMRs are not suppressed by fully-methylated Cs, gender-specific pMRs are suppressed by fully-methylated Cs in male samples and differentiated-cell-specific pMRs are suppressed by fully-methylated Cs in embryonic stem cells and the likes. Consistent with the findings from all cell-lines, conserved pMRs and gender-specific pMRs have lower percentage of fully-methylated Cs in female

embryonic stem cell-line H9, while differentiated-cell-specific pMRs have higher percentage of fully-methylated Cs in H9. Next, we characterized the pMRs with the histone modification data [245] from the cell-line H9. Figures 4.9b and 4.9c show that the active histone marks H3K4me3 and H3K27ac are enriched in the conserved pMRs and gender-specific pMRs, but not enriched in the differentiated-cell-specific

pMRs. The findings are consistent with the profile of the fully-methylated Cs in the cell-line H9. Since conserved pMRs and gender-specific pMRs in the cell-line H9 have both high percentage of partially-methylated Cs and relatively low percentage of fully-methylated Cs, they are expected to associate with both active histone marks and potentially active chromatin regions. For the differentiated-cell-specific regions, Figure 4.9a indicates that the differentiated-cell-specific pMRs have high percentage of fully-methylated Cs in H9; hence, they are expected to be suppressed and, thus, the histone marks H3K4me3 and H3K27ac are depleted in H9.

Lastly, we checked the distribution of pMRs relative to gene models (Appendix Figure A1.4) and the expressions of genes whose promoter regions overlapped with pMRs. Figure 4.9d shows that, conserved pMRs and gender-specific pMRs are enriched around gene promoter regions and inter-genic regions. Differentiated-cell-specific pMRs are enriched in inter-genic regions and introns.

Figure 4.9e shows the boxplots of the expression levels of genes in the cell-line H9 [218] whose promoter regions overlapped with the conserved pMRs, gender-specific pMRs, and differentiated-cell-specific pMRs. The figure also shows the boxplots of gene expression levels of the 1/3 highly expressed genes, 1/3 intermediately expressed genes and 1/3 lowly expressed genes in the cell-line H9. Clearly, the genes overlapped with conserved pMRs and gender-specific pMRs have intermediate expression levels (the expression levels are similar to that of the 1/3 intermediately expressed genes and higher than that of the 1/3 lowly expressed genes). Also, as expected, the genes associated with differentiated-cell-specific pMRs have low expression levels in the cell-line H9. Table 4.8 summarizes the properties of the conserved pMRs, gender-specific pMRs and differentiated-cell-specific pMRs studied.

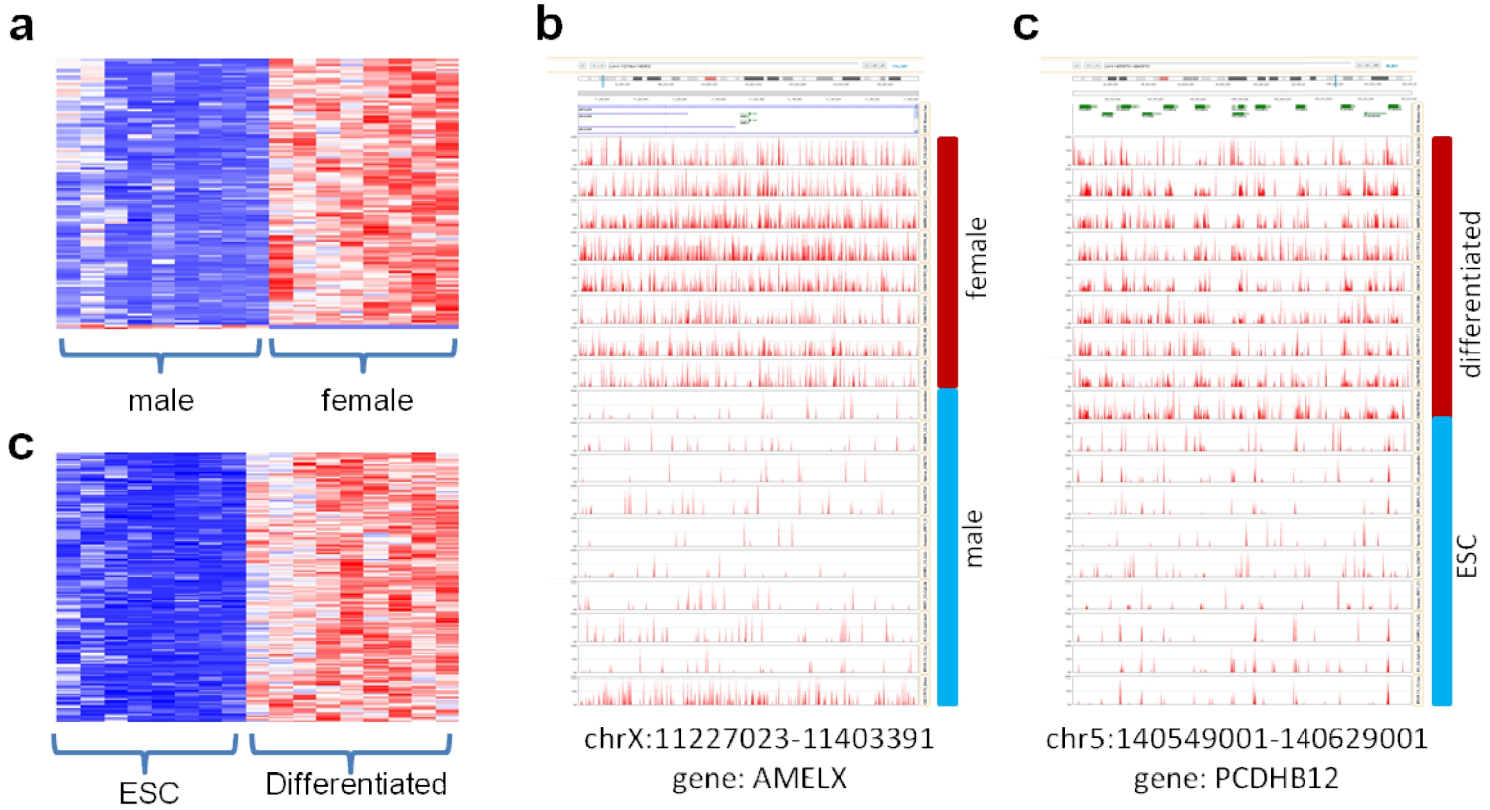


Figure 4.8. Cell specificity of partial methylation. a) Heatmap of percentages of partially-methylated Cs from gender-specific partially-methylated regions. b) Screenshot of gender-specific partial methylation around gene AMELX. c) Heatmap of percentages of partially-methylated Cs from differentiated-cell-specific partially-methylated regions. d) Screenshot of partial methylation around gene PCDHB12 for differentiated-cell-specific partial methylation. Refer to Table A1.1 for male/female cell-lines, and refer to Figure 4.6d for embryonic stem cells and differentiated cells.

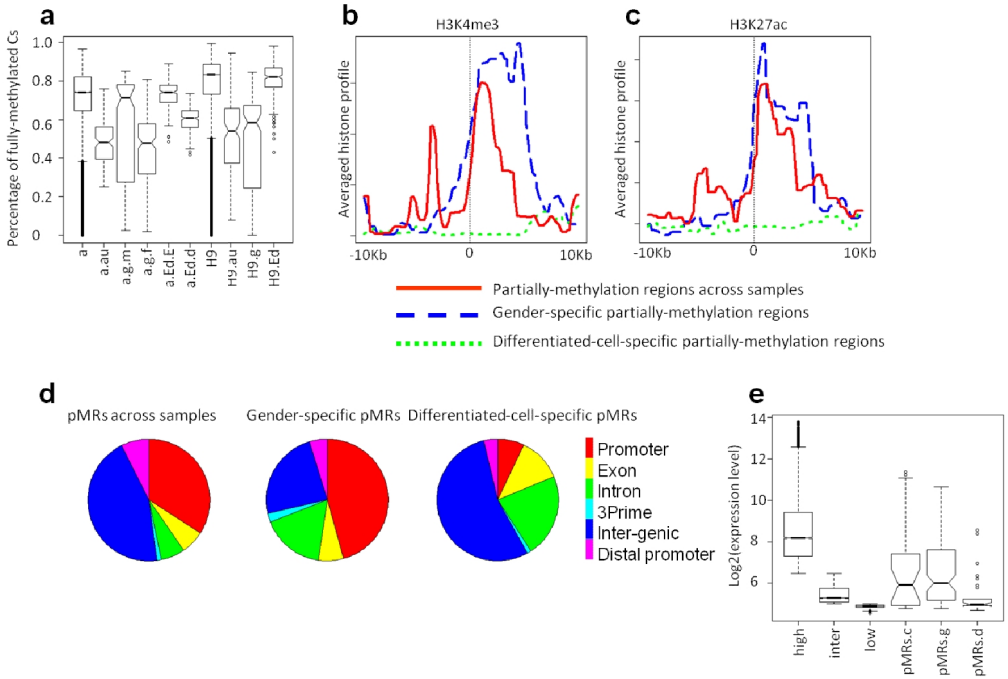


Figure 4.9. Histone modification profile and gene expressions of partially-methylated regions from embryonic stem cell H9. (a) Boxplot of the percentage of fully-methylated Cs from all the cell-lines and cell-line H9. a: average percentage of fully-methylated Cs across samples; a.au: average percentage of fully-methylated Cs across samples overlapped with conserved pMRs; a.g.m: average percentage of fully-methylated Cs across male samples overlapped with gender-specific pMRs; a.g.f: average percentage of fully-methylated Cs across female samples overlapped with gender-specific pMRs; a.Ed.E: average percentage of fully-methylated Cs across embryonic stem cells and the likes overlapped with differentiated-cell-specific regions; a.Ed.d: average percentage of fully-methylated Cs across differentiated samples overlapped with differentiated-cell-specific regions; H9: percentage of fully-methylated Cs from cell line H9; H9.au: percentage of fully-methylated Cs from cell line H9 overlapped with pMRs; H9.g: percentage of fully-methylated Cs from cell line H9 overlapped with gender-specific pMRs; H9.Ed: percentage of fully-methylated Cs from cell line H9 overlapped with differentiated-cell-specific pMRs; (b-c) Profiles of histone marks H3K4me3 (b) and H3K27ac (c) around conserved pMRs, gender-specific pMRs, and differentiated-cell-specific pMRs. (d) distribution of pMRs around gene models; (e) Boxplot of gene expressions from cell-line H9: 1/3 highly expressed genes(high), 1/3 intermediately expressed genes (inter), 1/3 lowly expressed genes (low), genes whose promoter regions overlapped with conserved pMRs (pMRs.c), gender-specific pMRs (pMRs.g), and differentiated-cell-specific pMRs (pMRs.d).

Table 4.8. Characterization of partial methylated region (pMRs)

	Conserved pMRs	Gender-specific pMRs		Differentiated-cell-specific pMRs	
		Male	Female	ESC cells	Differentiated Cells
Percentage of partially-methylated Cs	High	Low	High	Low	High
Percentage of fully-methylated Cs	Low	High	Low	High	Low
Number of regions	94	210		272	
Overlap of promoters (+/- 5Kb around TSS)	32	102		33	
H3K4me3 in cell-line H9	Enriched	Enriched		Depleted	
H3K27ac in cell-line H9	Enriched	Enriched		Depleted	
Gene expression in cell-line H9	Intermediate	Intermediate		Low	

4.5.2 Methods and Material on Prediction of Imprinting Genes

4.5.2.1 Data used

18 bisulfite-treated DNA methylation libraries from next-generation sequencing were used in this study, which were summarized in Appendix Table A1.1. H9 histone modification and transcription factor ChIP-Seq data were extracted from [245] with GEO accession number GSE24447. The H9 gene expression data and 27K array DNA methylation data were from [218].

4.5.2.2 Mapping of DNA methylation bisulfite sequence reads

BatMeth [239] was used for efficient and accurate bisulfite sequence reads mapping. At most three non-bisulfite mismatches were allowed for each read; bisulfite mismatch was defined as a T in the read and a C in the reference genome at their corresponding genomic locations. Only uniquely-mapped reads were kept for further processing. In addition, each C needs to be supported by three or more tags before it is interrogated for downstream analysis. The methylation level of the individual C is defined as the

percentage of reads whose Cs are not modified by the bisulfite treatment, which ranges from full methylation (100%) to no methylation (0%).

4.5.2.3 Validation of methylation callings

To validate our methylation callings, the methylation levels of the Cs in the 27K array are partitioned into the same five categories as in the bisulfite sequencing. Based on the 5 categories of Cs in methylation callings, there are 25 possible combinations of methylation callings from bisulfite sequencing and 27K array. The total numbers of different combinations are counted for all the common sites from our methylation calling and 27K array methylation calling. ~77% of the Cs called by both platforms have the same methylation categories. To show how the categories from our methylation callings and 27K array are consistent in general, we generated the ratio of the real counts in each of the 25 combinations and the expected numbers of the counts in the corresponding categories in Figure 4.5b.

4.5.2.4 Conserved partially-methylated regions

We used a sliding window of 5Kb to scan the genome. For each window, we computed the percentage of the partially-methylated Cs over the covered Cs in CpGs in each cell-line. From all the candidate regions, we calculated the average percentage of partially-methylated Cs over the covered Cs from all samples. Regions enriched with partially-methylated Cs were determined by the QQ plot (the cutoff was 0.15) and the regions within distance less than 5Kb were merged. Then, the regions were filtered out, if they had more than 40% overlap with RepeatMasker repeat regions [240].

To evaluate the significance of the overlap of the partially-methylated regions with imprinting genes, we randomly generated the same number of 5Kb regions, and counted the number of the randomly-generated regions overlapped with the known imprinting

genes. Such simulations were repeated 1000 times and the empirical p-value was defined as the number of times of simulations with more regions overlapped with known imprinting genes.

4.5.2.5 Gender-specific partially-methylated regions and Differentiated-cell-specific partially-methylated regions

For the 5Kb sliding windows across the genome, we performed t-test on the percentage of partially-methylated Cs between male and female cells. A sliding window was selected as a gender-specific partially-methylated region if the t-test p-value is less than 0.001, and the average percentage of partially-methylated Cs from male cells or female cells is more than 0.15. The gender-specific regions were further merged if their distance is less than 5Kb. The regions were filtered out if they had more than 40% overlap with RepeatMasker repeat regions. The calling of differentiated-cell-specific partially-methylated regions is similar to that of gender-specific pMRs, except that the t-test was performed between the DNA methylation libraries from the embryonic stem cells and differentiated cells and the cutoff for t-test p-value as 0.0001.

4.6 Discussion

DNA methylation is an important biological process. Mapping the BS reads from next-generation sequencing has enabled us to study DNA methylation at single-base resolution. Our proposed method aims to develop efficient and accurate methods to map BS reads.

This study employed three methods to evaluate the performance of BS read mapping methods. The first method measured the ratio of correct and wrong unique unambiguous mappings. This method only applies to simulated data when the actual locations of the reads are known. For real data, the number of unambiguous mappings alone may not be a good criterion to evaluate accuracy (we can map more reads at a higher mismatch number,

which results in lower specificity). The second method evaluated the accuracy using the number of reads that were mapped in consistent pairs, and can only be employed when paired-end read information is available. The third method used the directionality of the mapped reads from SOLiD sequencing. For the SOLiD reads, we mapped reads unbiasedly onto both forward and reverse directions of our reference genome. From the unambiguous mappings, we estimated the error rate of our unique mappings from the proportion of reverse direction unique mappings in the result sets. All these measures were used on different sets of simulated and real data and they suggest that BatMeth produces high quality mapping results.

For future work, our team will be working on more time-efficient data structures to better streamline our algorithm.

4.7 Conclusions

We report a novel, efficient and accurate general-purpose BS sequence mapping program. BatMeth can be deployed for the analysis of genome-wide BS sequencing using either base reads or color reads. It allows asymmetric BS conversion to be detected by labeling the corresponding reference genome with the hit. The components discussed in the Materials and methods section, such as list filtering, mismatch stage filtering, fast mapping onto two indexes, handling hypo- and hyper-methylation sites and other heuristics have offered increased speed and mappability of reads. In addition, BatMeth reduces biased detection of multiple CpG heterogeneous and CpH methylation across the whole reference by mapping onto both fully converted and non-CpG references and then labeling the reference to which the hits are from to aid biologists to discriminate each hit easily. Users can also choose to bias against either reference with varying mismatch scans. In assessing the uniqueness of a hit for BS color reads, BatMeth considers both strands of the DNA simultaneously while B-SOLANA considers both DNA strands separately.

Hence, BatMeth has a stronger uniqueness criterion for hits as B-SOLANA may produce two hits for a read, one hit for each separate DNA strand. Lastly, BatMeth uses an optimal DP algorithm to convert the color read to base space to check for non BS mismatches.

Chapter 5

Gapped Alignment Problem

5.1 Introduction

Genomic variations include insertions [246], deletions [247] and polymorphisms. These genomic variations can be captured by second-generation sequencing (SGS) at a high resolution. However, these genomic variations introduce mismatches and gaps when the SGS reads are aligned on to a reference genome. Hence, it is a challenge to accurately align SGS reads on to a reference genome. Furthermore, indels represent 7-8% of human polymorphisms [248]. This motivates us to develop an accurate gap-aligner.

Alignment tools can be divided into two main categories as mismatch-only and gapped aligners. A number of mismatch-only mapping algorithms have been proposed, including SOAP [104], RMAP [93], Bowtie [102], PerM [155], and BatMis [219]. They enable us to map reads allowing SNVs and sequencing errors.

However, mismatches cannot capture all types of genomic variations. Technically, indels may introduce a contiguous chain of mismatches in a read, with respect to a reference, which is computationally expensive for aligners to handle. With the increasing evidence

of indels being involved in a number of diseases [249], it is important to produce accurate mapping results for downstream variant callers such as VarScan [250], SAMtools [81], microindels [251] and Dindel [252]. Hence, it is important to align reads which have a mixture of mismatches and indels efficiently and accurately.

5.2 Related Work

Many gapped-aligners have been designed to allow both mismatches and indels in an alignment, including GEM [140], SeqAlto [112], ZOOM [117], BWA [100, 101], SHRiMP2 [116], Stampy [107], Bowtie2 [103] and others [144, 158, 180, 253]. Here we review some of the most popular officially published gapped aligners. GEM is currently the fastest published method and is based on filtration to leverage string matching with increased efficiency and precision. Based on FM-index, GEM implements a region-based adaptive filtering, tailored for each read, by leaving out regions which yield too many candidate matches that need to be checked, resulting in efficient alignments.

BWA-short [100] is a fast and accurate gapped aligner for short reads which performs a backward search of a read against the BWT-index of the reference. BWA-SW [101] also uses BWT-index and it represents the reference in a prefix trie and the query read in a prefix directed acyclic word graph (DAWG). All local matches of the query in the reference are then found by applying Smith-Waterman (SW) algorithm between the prefix trie and the prefix DAWG.

Stampy [107] is a hybrid method based on both BWT indexing and hash-based method. Stampy uses BWA-short as a pre-mapping tool to map reads onto a reference. For the remaining unmapped reads, Stampy identifies candidate genomic locations which match the length-15 seeds of these unmapped reads; then a Single Input Multiple Data (SIMD)

version of SW algorithm is applied to these candidate locations and the hit with the best-score returned by the SW algorithm is reported.

Similar to the seeding approach of Stampy, Bowtie2 [103] uses a length-22 seed with no mismatches to identify putative mapping locations and an SIMD-SSE2 implementation of striped SW algorithm to recover the best alignment from the seeded locations. However, such heuristics of using 0 or 1-mismatch seeds may miss the correct alignment of a read if the seeded regions of the reads contain more than one mismatch w.r.t. the reference genome.

SeqAlto is a method that hashes all k-mers of the reference genome. It aims to align long reads. This is achieved by the usage of longer k-mer sizes compared to some other approaches, examining less repetitive k-mers first and by adaptively stopping the k-mer search. Large contiguous k-mers greatly reduce the number of locations of each k-mer in the reference to mostly unique hits and improves accuracy of alignment too.

In summary, these methods assume that the seed does not contain gaps, have a small number of mismatches and/or that the seed does not come from repetitive regions. This affects the accuracy of existing methods. BatAlign is proposed by allowing high-mismatch and/or gaps in a seed. Candidate locations of a read to repetitive regions are also examined to avoid missing any correct alignments.

5.3 Results

5.3.1 Simulation study on variant-spanning reads

The alignment of reads in the presence of SNVs, indels, and/or SVs still remain challenging despite developments already made by published aligners. This section

intends to study the alignment accuracy of existing published methods using simulated reads that span across genomic regions with high number of SNVs, indels or SVs.

Mate-pair information can falsely disambiguate alignments: Mate-pair information is useful in aligning the two individually repetitive mate-paired reads unambiguously to the locality of each other in the reference genome. An ideal aligner should be able to align concordant and discordant paired-reads without bias, i.e., same rates of specificity while maintaining high sensitivity on mapping these two types of reads. (A concordant paired-read is a pair of reads that are sequenced from the vicinity of each other, within the expected wet-lab insert-size, on the reference genome. A discordant paired-read is a pair of reads that are sequenced much more than the expected insert-size from each other) However, if mate-pair information is used too aggressively, an aligner might wrongly align a pair of discordant read-pair concordantly onto the reference genome.

We have studied the impact of mate-pair information on alignment performance by aligning two types of simulated paired-reads (see *Simulation of data*). The first set consists of paired-reads that were simulated with a mean insert-size of 500 bp (s.d. of 50 bp) and the other set consists of paired-reads simulated with interchromosomal insert size. Figure 5.1 reports on the differences in mapping sensitivity and specificity of each published method between these two sets of reads. An ideal aligner should exhibit minimal performance bias between these two types of reads. Also, we have observed that the performance of the compared methods varied greatly from one and another between the alignments of these two types of paired-reads. The estimated bias in sensitivity and specificity ranges from ~13% to ~20% and ~0.1% to ~7.8% respectively among the compared methods.

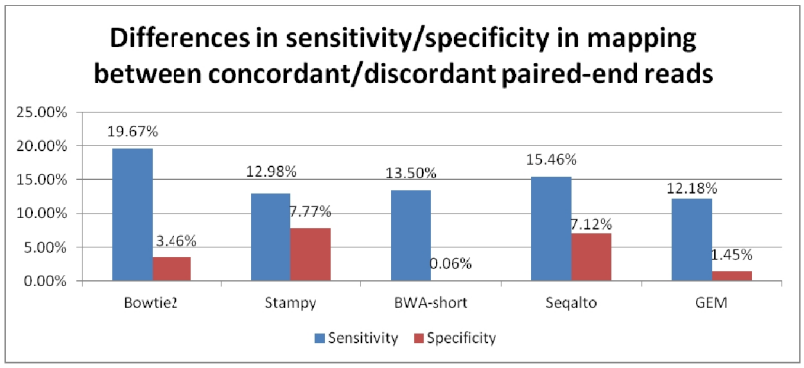


Figure 5.1. The difference in sensitivity and specificity between mapping paired-end datasets with simulated concordant and discordant mate-pair information.

Gapless seeding affects the alignment of indels: One delete of 1-8bp are simulated into reads of 75 bp long from hg19 reference genome (see *Simulation of Data*). We have noticed that up to ~4.9% of these reads can be represented by alternative genomic locations gaplessly with a low mismatch count of less than 5. This percentage is a lower-bound estimation as the seeds used in current published methods are much shorter (15-22 bp). This will affect indel callings.

To further investigate the impact of using short-seeds in aligning indels, we procured the current set of 1 bp indel reads into two groups. The first group of reads which have failed to be represented by alternate genomic positions with up to 5 mismatches while the second group of reads can be. On the first group of reads, all the compared methods have averaged sensitivity and specificity approaching 100%. However, on the second group of reads, the compared methods only obtained an averaged sensitivity and specificity of ~48% and ~53% respectively. As the reads are simulated with zero mismatches, the differences in mapping performance mainly stems from a gapless seed approach employed by the compared methods which did not capture the correct initial seeding alignment. (See Figure 5.2a-b for the detail of the sensitivity and specificity of different aligners for these two groups of reads.) This highlights the difficulty faced by current methods on mapping indel reads.

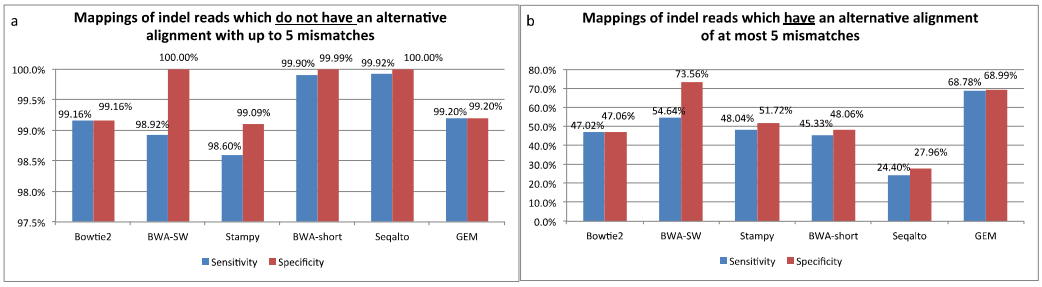


Figure 5.2. a) The sensitivity and specificity of compared methods on indel reads which do not have an alternative alignment with up to 5 mismatches. b) The sensitivity and specificity of compared methods on indel reads which have an alternative alignment of at most 5 mismatches.

Simulated reads with k mismatches can be mapped with less than k mismatches

Mismatches (like SNPs) can cause misalignments of reads, especially for reads sequenced from highly polymorphic regions, to homologous genomic regions of non-origins., We simulated reads (see *Simulation of data*) to study the effects of mismatches in producing misalignments..For each read, we reported the lowest-mismatch unique hits (using BatMis [219], an exact k-mismatch alignment algorithm). We then compared the number of mismatches at which the reads were simulated with (we call this value A) and mapped at (we call this value B). Interestingly, if $A=B$, the respective alignments from BatMis were mapped correctly. However, when $A \neq B$, the mappings were wrong, as it must be so due to alignment to a location different from where it was simulated.

We should note that with the increase of simulated mismatches in a read, the occurrences of it being misaligned with a lesser number of mismatches also increases; statistically, this is also true as mismatches act as wild-cards in string-matching problems. From the mappings of BatMis, the rates of misalignment for reads simulated with 1 to 5 mismatches increased from 0.3% to 0.9% respectively. This result implies that, in SNV-aberrant genomic reads, it is unwise to always pick the lowest-mismatch hit as it might misrepresent the original location of a read.

To further investigate the impact of high-mismatch reads on the performance of the current published methods, we procure two groups of reads from the current set of simulated reads. The first group and second group consist of k-mismatch reads which can be mapped uniquely by k-mismatch and less than k-mismatch respectively. On the first group of reads, all the compared published methods have averaged sensitivity of ~90% and specificity approaches 100%. However, on the second group of reads, both sensitivity and specificity never exceeded 4%. (See Figure 5.3a-b for the graph of sensitivity and specificity of different aligners for these two groups of reads.) This highlights the difficulty faced by current methods on mapping high-mismatch reads.

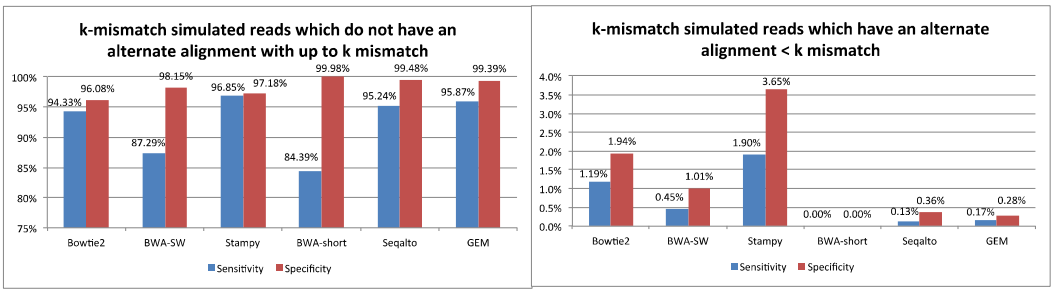


Figure 5.3. a) The sensitivity and specificity of compared methods on k-mismatch reads which can be mapped uniquely with k-mismatch. b) shows similar statistics to a) by mapping k-mismatch reads which have alternate unique alignment of ≤ k-mismatch.

5.3.2 Compared methods and method of cross-comparison

We have used the following gapped alignment tools for comparison: BatAlign, Bowtie2 (2.0.6), BWA-Short, BWA-SW (0.6.1-r104), Stampy (v1.0.16), GEM (3rd release) and SeqAlto (0.5-r123). These aligners are widely used and feature a wide range of mapping techniques. We run Stampy without using BWA as it was reported to be more sensitive. For each tool, the reference genome was indexed with default indexing parameters. hg19 was used for all experiments in this chapter. All experiments were run on a Linux workstation equipped with Intel X5680 (3.33 GHz) processor and 16GB RAM.

It was noted that GEM is the only method among the 7 compared methods that does not assign mapping quality to its alignments. We run the default modes of the compared programs unless otherwise stated. We have also adopted the performance measure "first correct" (or best) alignments from GEM's paper into our experiments to make sure our comparisons are extensive.

In this chapter, we have compared the full spectrum of mappings using ROCs. The ROC graphs are stratified by mapQ to visually compare the relative performances among the different methods. However, it is hard to compare the absolute differences in performance between methods as mapQ calculation differs from one method to another. To resolve this problem and to present the relative differences in performance numerically between the different methods, we will have to pick a baseline performance indicator for all methods to compare against with. For instance, if we want to compare the specificity of the methods, we will pick the method with the lowest number of incorrect/discordant mappings at mapQ>0 as the baseline for the other methods to compare with. In addition, any method that does not report a calibrated mapQ to its hits cannot be compared as described with other methods, but will still be plotted in the ROCs. In general, the ROCs in the later section show that if the sensitivity of a method drops, its specificity will increase. Hence, we can pick and compare the sensitivity and specificity of the various methods by picking a baseline as described.

5.3.3 Simulation of data

We generated three classes of simulated data. The first class mimicked Illumina-like reads, the second class has one indel in each of its reads and the third class is 'paired' reads. The first class of reads were generated by ART (Huang et al. 2011) from hg19 (excluding non-chromosomal sequences). We have chosen ART for our study since the substitution errors are simulated according to empirical, position-dependent distribution

of base quality scores; it also simulates insertion and deletion errors directly from empirical distributions obtained from the training data from the 1000 genome project [254]. Empirical read quality score distributions are provided for read lengths 75 bp, 100 bp and 250bp (these are the longest read lengths made available by ART). Although the error rate in the 75 bp and 250 bp data are generally <4%, we cap the number of mismatches and indels (SNVs or base-call errors or gaps) in all simulated read at 7% (an indel is counted as 1 error).

The second class of pure-indel reads was used to demonstrate the performance of BatAlign on the recovery of indels. 16 sets of 75 bp reads were created from hg19. Each pure-indel data set contains 1 million reads having one indel of a fixed length (since the average density of an indel is one in 7.2kb of DNA [247]). Indel lengths range from 1 bp to 8 bp with inserts and deletes being considered separately.

The third class of reads was used to demonstrate the efficacy of mate-pair information on the paired-end mapping mode of the compared programs. 6 sets of 1 million reads were created. Each set consists of 2 x 500k x (75/100/250) bp x (concordant/discordant) reads. The first set consists of concordant paired-end reads of mean insert size of 500 bp with a standard deviation of 50 bp. The second set consists of discordant paired-end reads, the 'left' and 'right' ends of the paired reads are simulated from chromosome 1 and chromosome 2 of hg19 respectively. This class of reads is to demonstrate the robustness of BatAlign when aligning reads with mate-pair information in the presence of genomic repetition and structural variations.

5.3.4 Evaluation on ART-simulated reads

As the original locations of simulated reads are known, we have assessed the sensitivity and accuracy of each method using simulated reads in this section. For each method and

each dataset, we discarded mappings with mapQ = 0 for all methods as they are deemed ambiguous. Then, we recorded the cumulative number of correct and wrong alignments by their respective decreasing mapQ and plotted these results in the form of an ROC curve; the corresponding cumulative number of correct and wrong alignments at a particular mapQ cutoff will be the respective x-axis and y-axis values for a single data point on the ROC curve. In addition, due to the inability to align some indels to their exact locations and the presence of soft-clippings, an alignment will be considered as a correct mapping if the leftmost position was within 50 bp of the position simulated by the simulator on the same strand.

Figure 5.4 shows the ROC curve (i.e. the cumulative correct alignments against cumulative wrong alignments) for each method and dataset. BatAlign was more sensitive and reported less wrong alignments than other methods over a large range of mapping quality cutoffs. We will cross-compare the methods for their sensitivity and specificity as described in *Compared methods and method of cross-comparison*. In terms of sensitivity on the 75 bp dataset, BatAlign, Bowtie2, BWA-SW, Stampy, BWA-Short and SeqAlto reported sensitivity of 91.0%, 84.1%, 74.9%, 85.6%, 82.2% and 85.5% respectively. In terms of specificity, BatAlign, Bowtie2, BWA-SW, Stampy, BWA-Short and SeqAlto reported specificity of 99.998%, 99.987%, 97.168%, 99.990%, 99.944% and 99.862% respectively. In terms of sensitivity on the 100 bp dataset (BWA-short was excluded from this comparison as it has a sensitivity of 47.9% as compared and Bowtie2/SeqAlto still reported higher than this level of sensitivity at their highest mapQ threshold), BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported sensitivity of 91.2%, 85.1%, 76.6%, 87.8% and 88.6% respectively. In terms of specificity, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported specificity of 99.996%, 99.992%, 96.644%, 99.995% and 99.786% respectively. In terms of sensitivity on the 250 bp dataset, BatAlign, Bowtie2,

BWA-SW, Stampy, BWA-Short and SeqAlto reported sensitivity of 88.8%, 85.1%, 85.8%, 86.4%, 88.5% and 88.0% respectively. In terms of specificity, BatAlign, Bowtie2, BWA-SW, Stampy, BWA-Short and SeqAlto reported specificity of 99.999%, 99.887%, 99.990%, 99.863%, 99.998% and 99.997% respectively.

We know that Stampy uses a short-seed of length-15 to get the candidate hits of a read. Based on this seeding strategy, Stampy should be able to have the correct hit among the candidate hits represented by the short-seed. However, Stampy was unable to produce accurate mappings and Figure 5.4 shows that over a large range of mapQ cutoffs, Stampy actually has the lowest specificity. As expected, Bowtie2 which also uses a short-seed mapping strategy also suffered high number of incorrect mappings within its results. It should be noted that both methods employed low mismatch/gap costs in their affine gap penalty cost matrix which are aimed at sensitive prediction of indels. This is done, however, at the expense of increased false-positive rates as shown in Figure 5.4.

In order to compensate for any bias which the ROCs might place against GEM, we have adopted the validation of first (or best) alignment used in GEM's paper for the mappings from all the compared methods on the 75/100/250 bp simulated datasets. Regardless of the mapQ values, this `correctness' measurement gives the number of mappings for which the simulated location was correctly retrieved by the mapper. We had the methods report the top 10 hits for each read (Stampy can only report at most 1 hit). Since all the compared methods report hits to a read ranked by their likelihood of being the correct hit, we improved the validation by reporting the rank of the correct hits too. The complete breakdown of our validation by of the first (or best) alignment by rank can be found in Table 5.1.

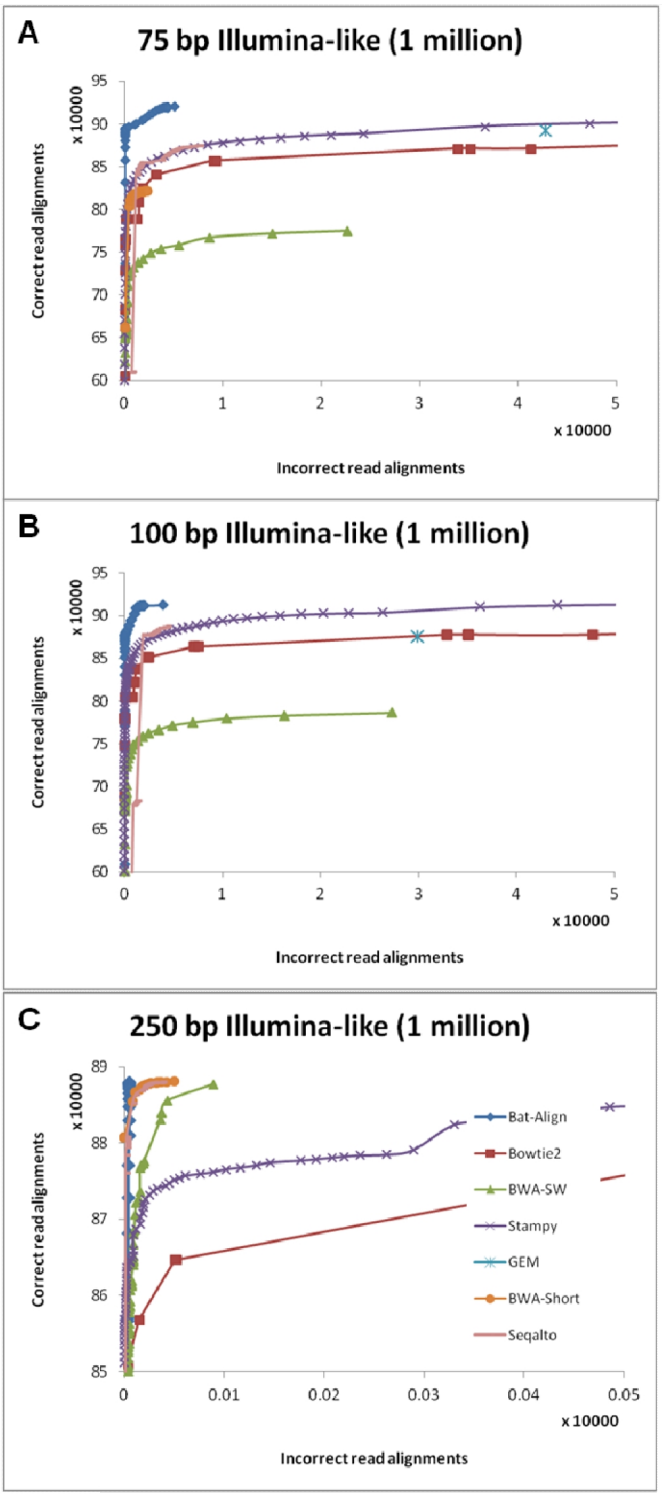


Figure 5.4. Sensitivity and accuracy for aligning simulated reads from ART. Cumulative counts of correct and wrong alignments from high to low mapping quality for simulated Illumina-like (A) 75 bp and (B) 100 bp (C) 250bp datasets.

From Table 5.1, we can see that BatAlign recovered the most number of correct hits within any of its reported 10 hits on the 75 bp and 100 bp dataset. On the 250 bp dataset, BatAlign is second to GEM by a margin of < 0.01%. Random inspection of the missing reads showed that the difference was caused by hits which are too repetitive and it was by chance that these hits were not ranked as any of the top 10 hits for their corresponding read.

5.3.5 Evaluation on simulated pure-indel reads

The reads generated by ART have ~0.01% probability of containing an indel. Therefore, Figure 5.4 does not show clearly the performance of the methods on reads containing indels. Therefore, we used the Pure-Indel read class to further highlight the performance of BatAlign on indel identification. For this comparison, we used sensitivity (SEN), accuracy (ACC) and F-measure to gauge the performance of the methods: SEN=TP/(TP+FN), ACC=TP/(TP+FP) where TP, FP and FN are true-positives, false-positives and false-negatives, respectively; F-measure =2(SEN*ACC)/(SEN+ACC).

Figure 5.5 plots the accuracy rates (Figure 5.5A-B) and F-measure (Figure 5.5C-D) graphs for all the compared methods across all the pure-indel datasets. BatAlign, Bowtie2 and Stampy had similar sensitivity across all the datasets of varying indel-lengths while the other programs generally have problems maintaining sensitivity when the size of the indel increases in our datasets. We also observed that BatAlign had the smallest drop from 92.0% to 91.4% with respect to increasing insert-lengths in the reads. Furthermore, the drop in sensitivity observed in BWA-SW was less than 1% on the delete-data sets but 5.4% on the insert-data sets. This hints that BWA-SW performs biased mapping of deletes over inserts. Nevertheless, BatAlign was the only method to have the highest F-measure and is unaffected by different indel-length in our experiments.

Table 5.1A-C. A..Number of first (or best) alignment reported by various methods on simulated 75bp dataset. B. Number of first (or best) alignment reported by various methods on simulated 100bp dataset. C. Number of first (or best) alignment reported by various methods on simulated 250bp dataset

Aligner \ Rank	75bp dataset #Correct hits 1	2	3	4	5	6	7	8	9	10	Sum of correct hits
BatAlign	933560	11604	732	622	745	335	146	116	90	68	948018
Bowtie2	866410	10873	4095	1541	551	314	192	142	112	77	884307
BWA-SW	786309	2	0	0	0	0	0	0	0	0	786311
Stampy	902757	-	-	-	-	-	-	-	-	-	902757
GEM	893162	13603	5243	2231	1074	688	555	375	323	272	917526
BWA-Short	834519	10008	3535	1226	408	160	83	52	43	26	850060
Seqalto	885208	5692	1712	723	306	164	102	63	33	32	894035

Aligner \ Rank	100bp dataset #Correct hits 1	2	3	4	5	6	7	8	9	10	Sum of correct hits
BatAlign	924272	7599	941	728	851	332	182	108	101	63	935177
Bowtie2	866310	8685	2833	948	254	110	69	42	23	5	879279
BWA-SW	794661	7	0	0	0	0	0	0	0	0	794668
Stampy	913874	-	-	-	-	-	-	-	-	-	913874
GEM	875333	10327	3533	1445	638	377	283	193	163	178	892470
BWA-Short	484558	5207	1747	662	211	112	79	48	20	13	492657
Seqalto	890336	1821	515	194	91	44	38	11	3	8	893061

Aligner \ Rank	250bp dataset #Correct hits 1	2	3	4	5	6	7	8	9	10	Sum of correct hits
BatAlign	894799	6394	732	824	225	175	107	98	53	50	903457
Bowtie2	892350	6245	1269	346	184	131	83	62	49	34	900753
BWA-SW	894395	1	0	0	0	0	0	0	0	0	894396
Stampy	893642	-	-	-	-	-	-	-	-	-	893642
GEM	895450	5999	1203	319	201	116	92	79	50	46	903555
BWA-Short	894658	5615	800	67	0	0	0	0	0	0	901140
Seqalto	894661	3775	296	77	41	33	14	9	9	9	898924

In terms of accuracy, Figure 5.5A-B shows that BatAlign has an average of 6.2%, 0.4%, 5.7%, 8.3%, 0.8% and 7.44% more specificity as compared to Bowtie2, BWA-SW, Stampy, GEM, BWA-Short and SeqAlto respectively. Figure 5.5A-B also shows that BatAlign maintained a high specificity even when the delete-insert-length increases.

In terms of F-measure, BatAlign clearly outperformed other methods and was the only program to have a stable F-measure of 95.5% across all types of simulated pure-indel reads. Figures 5.4 and 5.5 showed that BatAlign has better performance than the other methods on a general dataset with a mixture of mismatches and indels, as well as in identifying indels of various lengths. Thus, BatAlign can be used to identify a broad spectrum of variants, in the presence of sequencing errors.

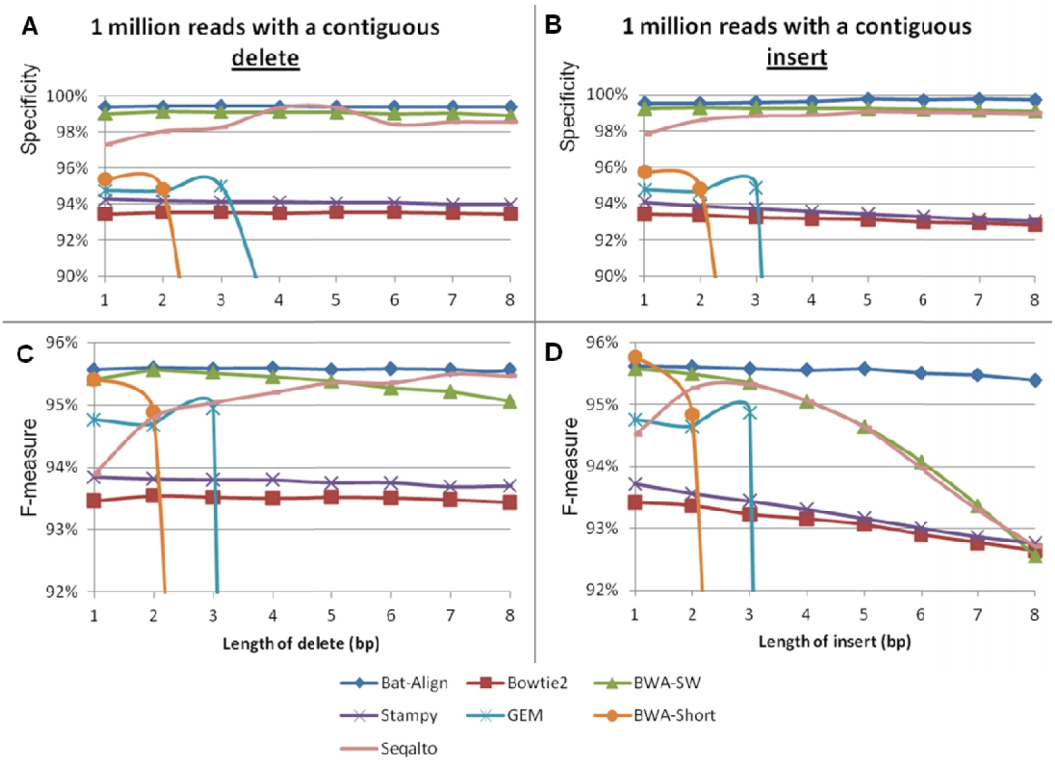


Figure 5.5. Specificity (A-B) / F-measure (C-D) of alignments using simulated Pure-Indel reads of various indel-lengths each with 1 million 75 bp reads. A/C are delete datasets. B/D are insert datasets.

5.3.6 Evaluation on paired reads

Mate-pair information was provided by the experimental setup from the wet-lab and was enabled by the paired-end sequencing capabilities of current sequencing technologies. After the PCR-amplification step of the genomic starting materials, the amplified sample undergoes sonication and fragments of similar size are separated with the use of centrifugal instruments. In this section, we present the results on mapping concordant (emulating a normal genome) and discordant (emulating large deletes and structural variations in a diseased genome) simulated reads using the paired-end mapping mode available in the compared methods. Due to the use of local alignment to sometime rescue an unmapped mate read, the rescued alignment of the initially unmapped read might be clipped at the ends and the reads cannot map to the exact locations as simulated by the simulator. As such, similar to the previous sections, an alignment will be considered as a correct mapping if the leftmost position was within 50 bp of the position simulated by the simulator on the same strand. The mappings of the discordant reads was unavailable from BWA-SW as the run did not complete after >2000 CPU hours. Post-processing are needed for the mappings of Stampy, SeqAlto and GEM as these methods modified the identification tags of the paired reads after their respective paired-end mapping mode.

On the dataset consisting fully of concordantly paired reads, BatAlign was more sensitive and reported less wrong alignment than most other compared methods at reported mapQ > 0. BatAlign, Bowtie2, BWA-SW, Stampy, GEM, BWA-Short and SeqAlto reported sensitivity with their running corresponding specificity of 98.0% (99.854%), 91.1% (92.601%), 93.0% (98.711%), 98.1% (98.907%), 97.4% (98.183%), 60.2% (99.702%) and 96.2% (99.881%) respectively. In the aspect of mapping concordant reads with paired-end mapping option of the compared methods, BatAlign has the highest sensitivity and second to SeqAlto in specificity (-0.027%). The slight loss in specificity on this

dataset is negligible as BatAlign still performed the best in terms of F-measure with 98.9% while SeqAlto measured at 98.0%.

On the discordant paired-end dataset, the sensitivity and specificity dropped for almost all the compared programs as compared to the rates from the previous concordant paired-end dataset. However, it is important to note that BatAlign now has the highest sensitivity and specificity of 88.6% and 99.643% respectively. At mapQ > 0, the sensitivity with their running corresponding specificity for Bowtie2, BWA-SW, Stampy, GEM, BWA-Short and SeqAlto are 88.6% (99.643%), 71.5% (96.058%), -(-), 85.1% (91.138%), 85.2% (96.735%), 46.7% (99.646%) and 80.7% (92.759%) respectively.

From doing paired-end mappings on these two datasets, an interesting trend of results was observed for the compared programs except for BatAlign. The initial observation was that methods which had lower specificity on the concordant-paired dataset, their specificity generally suffered a smaller drop on the discordant-paired dataset. For instance, Bowtie2 used to have a specificity of 92.601% on the concordant set but the specificity improved to 96.058% on the discordant set. The inverse of the initial observation on the results was also true. SeqAlto used to have the highest specificity of 99.881% on the concordant set but its specificity suffered a large drop of 7.122% to 92.759% on the discordant set. These fluctuations in specificity are due to the aggressiveness of the pairing algorithms in the various methods to map a pair of supposedly paired-end reads close to each other. These results showed the efficacy of mate-pair information which was the consequences of different approaches used for mapping paired-end reads in the different aligner. This will affect the alignment of SV-spanning read-pairs and will directly impact the callings of SVs.

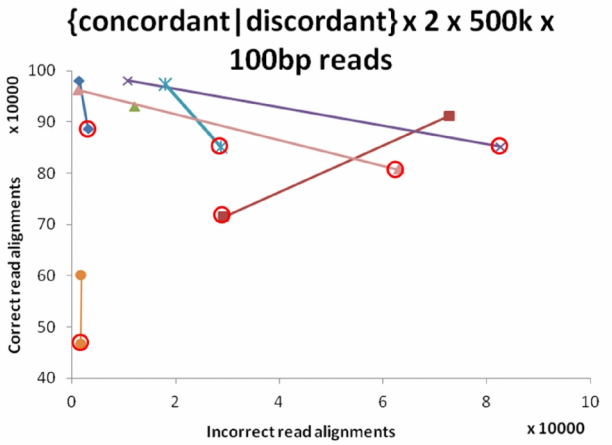


Figure 5.6. Mappings of concordant and discordant datasets using paired-end mapping mode of various methods. BWA-SW was unable to complete the alignment of 2 x 500k x 100bp discordantly paired reads and is plotted as a single data point in the graph. The 'red' circled plots are from the discordant dataset.

Figure 5.6 summarized the performance of the compared methods on the above-mentioned paired-end datasets. The mappings on the two datasets that are from the same method are joined together by a line. As from the above, we can also concur that the ideal method should not suffer any depreciation in its ability to detect the correct alignments for the reads from any of the two datasets. Thus, graphically-speaking, Figure 5.6 shows how biased a method is in its paired-end mapping mode in aligning reads with mate-pair information with the length of the lines that joins the data points in Figure 5.6. Overall, BatAlign was observed to have the smallest fluctuations in its sensitivity and specificity of 9.4% and 0.211% respectively between the two datasets.

One would normally argue that the results on the discordant set might be skewed as the frequency of such discordantly paired reads is too high even in a cancer genome as subjected to inter-chromosomal fusion or intra-chromosomal rearrangements. As a real dataset would have a mixture of concordant and discordant read-pairs in it, one can strongly infer the robustness of the paired-end mapping mode of a method from the line that joins the pair-data points of the corresponding method. This is why the results are

presented as it is in Figure 5.6. The results in this subsection are obtained from running datasets of 100 bp long. Experiments were also done using 75 bp and 250 bp datasets and the trend of results are consistent among all three datasets.

Although the proportion of discordant read-pairs in a real dataset may vary from what we have simulated but we still want to provide our readers the feeling of how various aligners will perform on a general dataset which will have these SV-spanning discordant read-pairs in them. We adjusted the proportion of discordant read-pairs to be 5% in each datasets of 2 x 500k x 75/100/250 bp reads. We then aligned them using the compared methods and verify the correctness of their respective alignments. Table 5.2 shows that BatAlign can better align reads which span across large gaps/SVs with higher performance. In terms of the number of misalignments, the other methods have at least ~0.5x to ~200x more misalignments than BatAlign. We also found out that these huge numbers of misalignments from the other methods comes from the 5% discordant reads which are simulated into the datasets.

Table 5.2. Alignments on a 5% discordant simulated paired-end dataset of various read-lengths by all compared methods.

	Methods	BatAlign		Bowtie2		BWA-short	
2 x 500k x ReadLength	Format	Wrong	Correct	Wrong	Correct	Wrong	Correct
	75bp	1,203	969,846	81,873	897,083	1,729	937,567
	100bp	1,231	979,930	76,401	904,722	2,356	595,437
	250bp	37	979,410	93,349	902,535	298	984,586
BWA-SW		GEM		SeqAlto		Stampy	
Wrong	Correct	Wrong	Correct	Wrong	Correct	Wrong	Correct
26,409	776,090	44,271	914,696	3,992	870,479	48,798	917,596
32,488	772,499	29,194	871,661	1,912	881,189	45,047	935,131
168	978,735	9,748	987,430	51	902,875	42,599	946,378

5.3.7 Evaluation on real reads

We have downloaded 500k reads of 2 x 76 bp (DRA accession DRR000614, Sample: NA18943), 2 x 101 bp (SRA accession SRR315803, Sample: NGCII082 Mononuclear blood) and 2 x 150 bp (SRA accession ERR057562, Sample: ERS054071) paired-end datasets. The sequencing platform used for the downloaded data was Illumina Genome Analyzer IIx for the 76/101 bp dataset and Illumina MiSeq for the 150 bp dataset.

In order to address the lack of an oracle set, we mapped the paired-end reads as single-end reads and calculated the fraction of reads that were mapped concordantly. We consider a pair of reads to be concordant if they have the correct orientation and maps within 1,000 bp of each other with a mapQ > 0. (The distance 1000 is chosen since Illumina machines normally cannot extract paired-end reads from DNA fragments of size longer than 1000bp.) If both ends of the paired-end reads were mapped but were not concordant, they were marked as discordant. This form of verification gauges the single-end mapping algorithms with reads containing the true spectrum of polymorphisms, substitutions and read errors. To plot the full spectrum of concordance/discordance in our experiments on real data for the ROCs, we recorded the number of concordant and discordant alignments stratified by the mapping quality of the forward read. We must also emphasize that although the rate of concordant mappings is taken as a measure of performance in aligning real reads, it is only a lower bound of performance when used on mapping datasets of expectedly high paired-end concordance sequenced rates. Paired-reads with an opposite unmapped end will not be considered as they only form a minimal portion of the mappings and there is no oracle data to discriminate wrong mappings from the correct mappings.

Figure 5.7 reported the mapping results on the 76 bp, 101 bp and 150bp real datasets. BatAlign reported more concordant mappings and less discordant mappings than other

methods over a large range of mapping quality cutoffs. We will cross-compare the methods on their concordance and discordance respectively, as described in *Compared methods and method of cross-comparison*.

In terms of concordance on the 75 bp dataset, using BWA-short as the base line of 74.1%, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported concordance rates of 77.6%, 46.8%, 71.9%, 28.0% and 72.4% respectively. In terms of concordance within the resultant mappings, using BWA-short as the baseline of 99.334%, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported rates of 99.558%, 97.990%, 99.022.%, 96.792% and 99.022% respectively. In terms of concordance on the 100 bp dataset, using BWA-short as the base line of 78.6%, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported concordance rates of 81.8%, 70.6%, 80.2%, 70.3% and 79.8% respectively. In terms of concordance within resultant mappings, using BWA-short as the baseline of 99.132%, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported rates of 99.578%, 98.823%, 99.434%, 97.706% and 99.285% respectively. In terms of concordance on the 150 bp dataset, using BWA-short as the base line of 71.8%,, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported concordance rates of 82.6%, 59.9%, 79.9%, 65.9% and 79.9% respectively. In terms of concordance within resultant mappings, using BWA-short as the baseline of 98.531%, BatAlign, Bowtie2, BWA-SW, Stampy and SeqAlto reported rates of 99.165%, 98.031%, 98.826 %, 98.126% and 98.908% respectively.

5.3.8 Evaluation on life-sized dataset

A life-sized dataset (GEO Accession: SRX084939, SRX084940, SRX084941) with ~33x coverage sequenced from the gastric tumor of a gastric cancer patient (GSM 764988), was also downloaded for comparing indel-mapping capabilities of the different methods. A total of 1,004,087,071 reads were gathered across 28 runs of the experiments. Duplicate mappings (mapQ>0) were removed by SAMtools (v0.1.18) within each

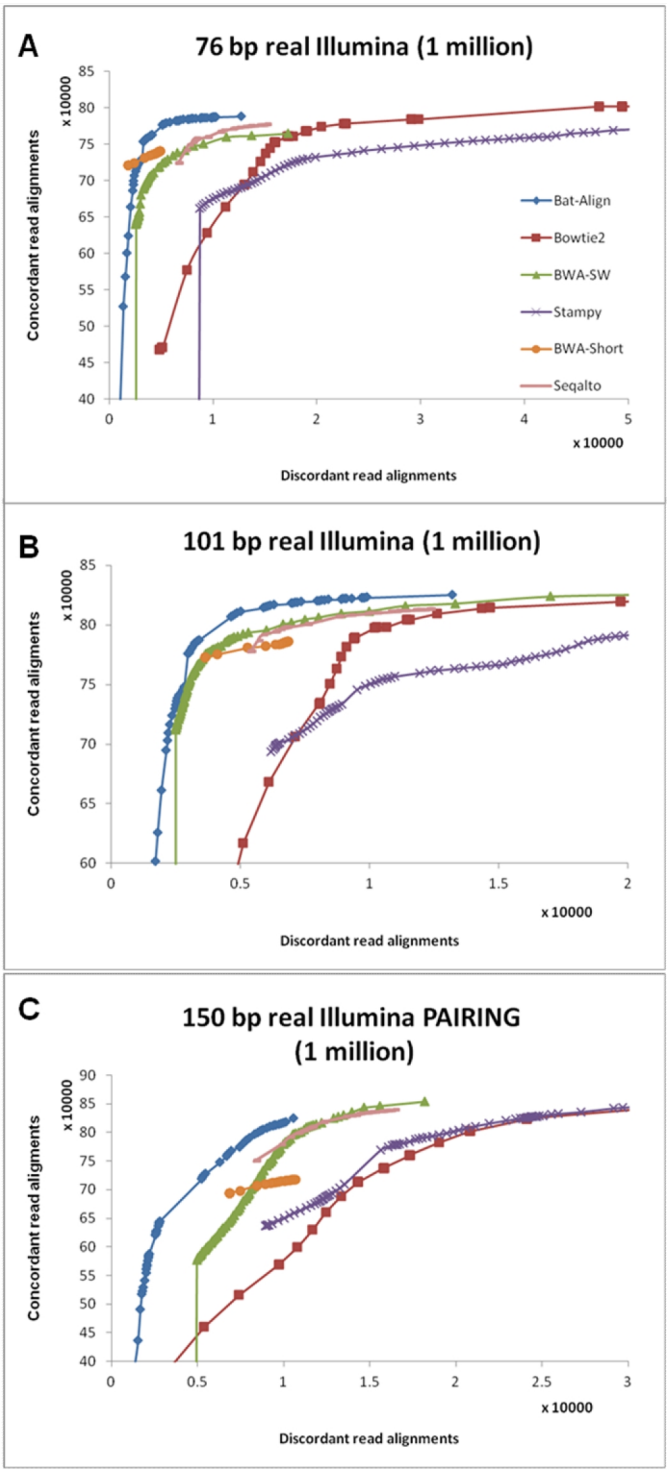


Figure 5.7. Concordant and discordant alignments using real reads from Illumina. Cumulative number of concordant and discordant alignments from high to low mapping quality for real Illumina (A) 76 bp (B) 101 bp (C) 150bp datasets.

sequencing run for downstream analysis. Stampy was left out from this analysis as it could not finish mapping the library within 2 weeks. GEM was unable to report mapping quality information which will affect the quality of variant-calling adversely and was left out from this analysis too. Since we are comparing mappings at sites of PCR-validated sites, we will pick the methods with the highest mappability as reported in the previous sections. In addition, we only picked two programs, namely Bowtie2 and BWA-SW, to compare with due to the intensive resources needed to perform variant-callings.

For purpose of variant-calling, we have used SAMtools to call variant-indels on all the mappings from the respective methods. Cutoffs of at least 10 supporting reads and variant score of 50 (30 for SNVs) were used to filter low-quality indels-variant calls.

5.3.8.1 Eliminating scoring bias

We have tried to eliminate mapping bias from the different methods by realigning the resultant mappings from the respective methods with the same set of alignment costs. First, we used Genome-Analysis-ToolKit (GATK-Lite-2.3.4) to target all the genomic intervals which overlapped with reads supporting an indel or clustered with mismatches. Next, all mappings which overlapped with the targeted intervals are realigned with GATK using the same set of affine gap penalty scores.

5.3.8.2 Accuracy of mappings over PCR-validated variants

The methods were checked for their ability to detect variant-indels by counting the number of reads which supported the 70 PCR-validated indels [71]. BatAlign was able to map 472 reads across these indels as compared to the 431 reads by Bowtie2 and 337 reads by BWA-SW. We also checked the proportion of reads which suggested an indel to the total number of mappings overlapping these 70 locations for each program. BatAlign, Bowtie2 and BWA-SW had 32.2%, 28.4% and 23.9% of mappings which called indels at

these validated sites and the concordance rate of the mappings at these validated sites were 88.5%, 85.7% and 88.2% respectively.

The methods were also checked for sensitivity and accuracy in detecting SNVs. The bases of each read that covered the validated SNVs were verified to see if they supported either the nucleotide of the main or alternate alleles. Over 67 PCR-validated SNVs [71], BatAlign, Bowtie2 and BWA-SW were able to map 2046, 2036 and 1913 reads across these validated sites respectively with only 1 wrong support from each program over chr1:32827126. To extend the analysis on accuracy in detecting SNVs, we took 50,000 consensus SNVs from the methods that coincide with dbSNP137 [255] to check if the mapped base from each corresponding method was able to support the SNV represented in dbSNP137 correctly. It turned out that BatAlign supported the consensus SNVs with the highest accuracy; 1,307,185 correct supports and 2,986 (ACC=99.772%) wrong supports while Bowtie2 had 1,279,525 correct supports and 2,969 (ACC=99.768%) wrong supports, and BWA-SW had 1,080,803 correct supports and 2,504 (ACC=99.769%) wrong supports.

5.3.8.3 Sensitivity of variant-callings over PCR-validated sites across sub-samples

The variants called from a robust method should be the least affected by sub-sampling of its own respective mappings. We investigated the robustness of each method by performing variant-callings on sub-samples of their resultant mappings. For each method, by comparing the percentages of variants being called in the sub-samples to the set of variants called from the mapping of the full dataset, we can deduce which method is most robust. In this experiment, Table 5.3A-B shows that BatAlign has the highest counts of detected variants from SAMtools on 9 out of 10 subsamples.

Table 5.3A. Number of indel-variants called from the sub-samples at 70 PCR-validated sites

read depth cutoff	2	4	6	8	10
Subsample size	20%	40%	60%	80%	100%
BatAlign	36	39	45	47	52
Bowtie2	35	40	45	47	51
BWA-SW	2	37	41	46	48
Ideal Program			70		

Table 5.3B. Number of SN-variants called from the subsamples at 67 PCR-validated sites

read depth cutoff	2	4	6	8	10
Subsample size	20%	40%	60%	80%	100%
BatAlign	4	11	14	23	30
Bowtie2	4	8	14	20	28
BWA-SW	4	8	11	20	20
Ideal Program			67		

5.3.8.4 Evaluation on running times

Up till now, we reported on the sensitivity/specificity (simulated data) and concordance/discordance (real data) of various methods. BatAlign was generally observed to have the highest performance on these two measures among the compared methods. BatAlign was developed to focus primarily on producing accurate alignments. However, by doing so, it trades off speed. In Table 5.4, we show that the default mode of BatAlign can run within relatively reasonable timings as compared to some of the fastest methods used in our comparisons.

In the spirit of accurate alignment, BatAlign trades speed for accuracy and was not as fast as GEM and Bowtie2. From *Evaluation on simulated/real reads*, the ROC curves showed

that BatAlign is more specific/concordant over a large range of mapping qualities. Since this is so, we try to allow more flexibility of usage for users by allowing BatAlign to run faster but with slight decrease in discordance. The analysis of runtime is also done on the same set of real data used in the previous section to present realistic timings of all the compared programs. Table 5.4 shows the relative runtimes and speed factors between the programs and we conclude that GEM is the fastest aligner.

We have also observed that discordance rates increased with decreasing running times. This was resulted from the reduction of search space in a bid for faster running times. Since BatAlign always try to scan for all candidate hits incrementally and report the best hit, a reduction in search space will actually discard some concordant hits prematurely and this have caused the increased discordance rates.

Table 5.4. Comparison of running times across all discussed programs on 1 million reads from SRR315803.

Program	Runtime (seconds)	Speedup factor
BatAlign - Default	583	6.3x
BatAlign - Fast	481	7.7x
BatAlign - Turbo	331	11.2x
Bowtie2	459	8.0x
BWA-Short	598	6.2x
BWA-SW	639	5.8x
GEM	214	**17.3x**
SeqAlto	677	5.5x
Stampy	3694	**1.0x**

5.4 Methods

5.4.1 Problem definition and overview of the method

The problem of mapping genomic reads is defined ideally as given a set of genomic reads, find the origin of each read in the reference genome, along with their correct alignments. However, in practice, this problem cannot always be solved and we have to resort to finding the most likely point of origin and alignment for each read.

The outline of BatAlign algorithm is as follows. As a pre-processing step, a one-time indexing of the reference genome is done. Next, it will start scanning for the most probable hits of the read in the reference by using *Reverse-alignment*. *Deep-scan* is then applied to scan and pick the most probable hit of the read in the reference. BatAlign then calculates a mapping quality (mapQ) for this hit and reports it. Below, we will discuss the novel components that aid BatAlign to gain accuracy and speed.

5.4.2 Reverse-alignment

Seed-based aligners search for candidate hits of its seeds; then, these hits are extended and the best alignment is selected based on a set of pre-defined criterion. In contrast, *Reverse-alignment* does the opposite by searching for the best possible hits in the reference first. Given a read R, *Reverse-alignment* incrementally finds the hits of R with the most likely combination of mismatches and indels. We define a function F such that $F(i) = (p_i, q_i)$, where p_i and q_i are non-negative integers representing the number of mismatches and indels in an alignment respectively. If $F(a) < F(b)$, then the probability of the correct alignment of a read having (p_a, q_a) mismatch/indel combination is higher than that of (p_b, q_b) mismatch/indel combination. *Reverse-alignment* incrementally tries to map R allowing F(i) mismatch/indel combinations for i = 1,..,9. We will describe the definition F(i) in the following subsection.

5.4.3 Determining F

In real-life, the likelihood of an indel in a genome is an order of magnitude less than that of a SNV. The likelihood of finding multiple indels within a read becomes small if the length of R is shortened. If sequencing is error-free, we can expect mismatches in a read R to appear at a rate equal to the expected number of SNVs in a segment of length |R| in the reference genome. However, empirical studies show that, for Illumina and SOLiD, the majority of mismatch errors are due to sequencing errors. A general heuristic for such platforms is to set the mismatches in a read due to sequencing errors and/or SNVs to be about ~5% of the read length. Furthermore, indels occur at a rate of ~0.02% [256]. Based on these statistics, for a read of length around 75 bp, we can set one indel and four mismatches as a reasonable upper bound for the number of indels and mismatches to be allowed in a mapping. For the default mode of enumerating candidate hits, we have 9 levels where F(1)=(0,0), F(2)=(1,0), F(3)=(2,0), F(4)=(3,0), F(5)=(4,0), F(6)=(5,0), F(7)=(0,1), F(8)=(1,1) and F(9)=(2, 1).

5.4.4 Deep-scan

The best-scoring alignment according to the function F need not be the correct alignment, even if it turns out to be the only hit with such a mismatches/indel combination. It is best if we can get the set of next-best alignments too. With these additional hits and using the quality information of the mapping, we might be able to find the correct alignment. Furthermore, these extra hits will help BatAlign to assess the quality of the final alignment better. *Deep-scan* enumerates hits according to F. If F(k) is the first successful mismatch/indel combination found during *Reverse-alignment*, and there are multiple hits, we return all these hits. Otherwise, if there is a unique hit and $k < 9$, we return all hits having the mismatch/indel combinations F(k) and F(k + 1).

5.4.5 BatAlign algorithm

We will first describe the BatAlign algorithm for a short read (75 bp). It consists of three steps. First it will perform a *Deep-scan* for each read R to build a set of candidate alignments. Next, if a set of candidate alignments can be found, each hit is assigned an alignment score based on quality information and the unique, highest scoring hit is reported (if it exists). Finally, a quality score is assigned to each reported hit as described in the next section.

5.4.6 Handling long reads

For reads longer than or equal to 150 bp, we will split the read into non-overlapping 75 bp reads. Each of the 75 bp segment will be aligned as described above. For instance, if we are given 250 bp reads, BatAlign will obtain 3 consecutive segments of a read starting from the first base of the read and map each of them individually. If the first or best hit from each segment are non-repetitive and fall within the locality of each other, we will try to align the original read onto this region of the reference. By doing this, we avoid realigning the original read to more than one location of the reference. However, if the first or best hits from each segment are repetitive or not mapped to the locality to one another, BatAlign will examine and align the whole read onto the putative locations reported by each of the segment. Among these alignments, the best-scoring hit is reported.

5.4.7 Enumerating hits

We use efficient BWT-based methods to enumerate hits corresponding to F(i). With the restriction stated above, i.e. assuming we allow only one indel, we have two cases where F(i) is of the form $(p_i, 0)$ or $(p_i, 1)$. When $F(i) = (p_i, 0)$, only mismatches are allowed in R, we will use the BatMis algorithm to solve this case. BatMis is a BWT-based algorithm that can enumerate all hits having k-mismatches exactly and efficiently.

5.4.8 Finding indel hits.

When F(i) = (p_i, 1), we allow pi mismatches along with an indel. For all these hits of R with a gap in them, there are two cases. The first case is that the indel appears in one half of the read. Then the other half of the read that does not contain the indel must contain at most pi mismatches. We map the left and the right halves of R allowing at most p_i mismatches; then a SW-extension is performed to recover alignments having pi mismatches and one indel. The second case happens when the indel is not completely contained in either half of the read, but is overlapping the midpoint of the read. To identify these hits, we apply BatMis algorithm to find <0, p_i> mismatch hits of the *l*-mer suffix and prefix of R, where *l* is set to one third of |R| by default. Then, using a novel data structure, we find the suffix and prefix locations whose total number of mismatches do not exceed pi and are at most *d* bp apart, where *d* is set to 200 by default. These potential hit locations are further examined by aligning R against a neighborhood of the possible hit locations using the SW-algorithm, and those alignments with p_i mismatches and one indel are reported.

Apart from the *Deep-scan* criterion, we will always perform an indel-scan if (1) an indel is detected during the full-read extension; (2) the current best alignment score is worse than an alignment possibly having an indel; or (3) the average base quality of mismatch positions in an alignment is higher than that of the average base quality of the read.

5.4.9 Faster semi-global alignment and SW alignment

After mapping a 75 bp seed, BatAlign can perform either SW alignment or semi-global alignment to extend the alignment of the read. Since we have devised a semi-global alignment method that is faster than SW-alignment by ~30%, the default mode of BatAlign is to extend the seed using semi-global alignment. When the alignment score of the semi-global alignment drops below 90% of the maximum alignment score (i.e. the

score for an exact match), a SW-alignment is done. If the user wants to perform SW-extensions only, an option is provided to do so. Below, we describe the SW alignment and the semi-global alignment methods.

The SW alignment is SIMD accelerated via SSE2 instructions. Our implementation is based on an extension of SSW library [121] that modifies Farrar's method [120]. This algorithm determines the best alignment in two steps: First it will calculate the best SW-score and then it will perform a banded alignment to get the optimal trace-back of the alignment from the DP-table.

For the semi-global alignment, we designed a new algorithm assuming that there is at most one gap in the pair-wise alignment. The algorithm will divide the read into two halves and first assume that the indel is in the left half. If the indel is in the left half, the right half of the read must align to the reference with only mismatches. Figure 5.8 shows the situation for the case of a deletion. The right half of the read (Part C) maps to location Y in the genome. Part A of the read maps to location X in the reference. Location X will be found by BatAlign algorithm where a seed of length |R|/2 will be mapped. Assume we allow a maximum of d bp for the indel, we will set j=1..d, and map part C of the read at j bp away from part A of the read.

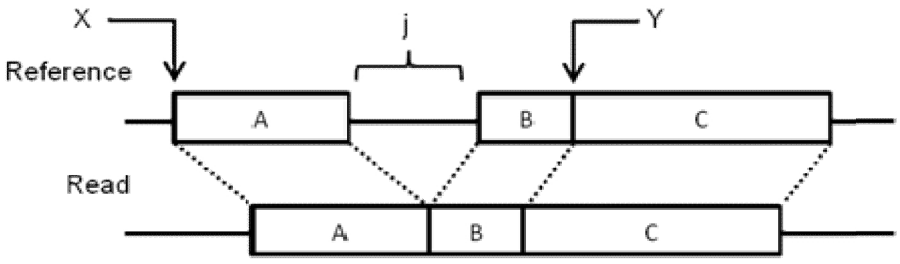


Figure 5.8. Example of recovering a delete in a reference from a read

5.4.10 Alignment score and mapping quality

Sequencing data can contain a per-base quality score that indicates the reliability of a base call. If the probability of a base call at position i being correct is P[i], the quality score Q[i] assigned to location i is given by the equation $P[i] = 1 - 10^{-Q[i]/10}$. Assuming that there is no bias to a particular set of nucleotides, the probability of a base being miscalled at location i can be calculated by the formula $1 - P[i]/3$. For a given alignment, we compute an alignment score based on an affine-gap scoring scheme, where the score for a match or a mismatch at R[i] is the Phred scaled value of P[i].

5.4.11 Accelerating alignment

The speed of the algorithm is improved by limiting the number of SW-alignments performed for each read. Another way is to stop performing SW-extensions when the best alignment score has failed to increase after a determined number of attempts. To trace back the optimal alignment path in the DP table, we need to perform a non-SIMD banded version of SW-algorithm. This step is time consuming. However, we can skip this step if the SW-score of the alignment falls below the current best SW-score. Furthermore, we can also restrict the number of candidate hits to check in the *Deep-scan* process. These heuristics will significantly speed-up the alignment process without much loss in sensitivity and accuracy as shown in *Evaluation on running times*.

5.5 Conclusion

We presented a method, called BatAlign, for the gapped alignment of genomic reads onto a reference genome with improved accuracy and sensitivity. The mapping strategies discussed in the Method section, such as *Reverse-alignment* and *Deep-scan*, produced mappings with increased accuracy as compared to other methods in both simulated (ART-simulated and pureIndel) and real datasets. In addition, BatAlign aligned over sites of PCR-validated indels and SNVs with more concordant mappings and coverage, and

was also shown to be more robust in calling variants in low-coverage samples. A new *faster semi-global alignment algorithm* and other heuristics have also been used to replace the traditional SIMD-SW routine to speed up BatAlign. BatAlign also outputs a well-calibrated mapQ score for each mapped read. In general, BatAlign is an improved gapped aligner for accurate gapped alignment of DNA reads.

Recently, a number of aligners such as YAHA [123] and CUSHAW2 [110] are developed to handle long reads (500 bp or more). These aligners are not able to produce accurate results due to their use of maximal exact-match seeds. A possible future work is to develop an accurate tool for the alignment of long reads.

Chapter 6

Spliced Alignment Problem

6.1 Introduction

RNA, together with DNA and proteins, is one of the three major macromolecules that are needed for life. Pre-mRNA is synthesized from the DNA through transcription and is matured by having its introns removed [161]. In mammalian genomes, alternate splicing of the same gene region adds onto the genomic complexity by generating multiples variants of a single gene known as mRNA isoforms [162]. The disruption in the synthesis of mRNA isoforms can cause genetic disease [163, 164]. Hence, it is critically important to accurately identify and quantify the splicing sites in both normal and diseased cell states.

RNA-seq can interrogate gene expression levels at genome-wide scale. *De novo* detection of splice junctions and quantification of novel gene expression was also not possible with microarray technologies before. Each sequencing run, from next-generation-sequencing (NGS) technologies, of an RNA-seq experiment can yield up to hundreds of millions of bases, allowing the accurate relative quantification of expressed transcripts. In all, RNA-

seq has provided a quantum leap to the analysis of novel features in the transcriptome [34] from hybridization-based microarray techniques.

6.2 Challenges in Spliced Alignment

The first step to analyzing RNA-seq data is to align the sequenced reads back onto a known reference genome or annotated transcriptome. The alignment of RNA-seq also brought along an additional set of challenges as compared to aligning DNA-seq data. The first challenge is to align in the presence of large gaps due to the presence absence of introns from the sequenced reads with respect to the reference genomic text which we are aligning the reads onto. From empirical studies, ~38% of 100 bp RNA-seq reads can span across two or more exons that can be thousands of bases apart [176]. Due to splicing junctions between adjacent exons in a read, different subparts of a read can map to different adjacent exonic regions of the reference genome but with a large intronic gap in between them. Other than the presence of large intronic gaps, alignment is further complicated by the presence of polymorphisms, indels and sequencing errors. In addition, it was also observed that ~25.8% of 100 bp long reads, has an exon-exon boundary within 10 bp on either ends of a read. This short residual exon, which we call short 'overhang', can be represented spuriously by the reference genome and is both computationally and algorithmically hard for aligner to accurately locate its correct alignment efficiently. Short exons can also appear in the middle of a read, sandwiched between two exon-exon boundaries within a single read. Without loss of generality, RNA-seq reads poses a new set of challenges for aligners to work with as compared to its siblings of DNA-seq aligners.

Other than intronic gaps, pseudogenes also make splice alignment harder than DNA gapped alignment. Pseudogenes are dysfunctional relatives of genes which are highly similar to RNA sequences [257]. An ideal RNA-seq aligner should be able to avoid

aligning reads to processed pseudogenes at all times as pseudogenic regions do not transcribe to mRNA sequences. The authors of TopHat2 [42] has also found that ~26.9% of reads in the RNA-seq data from [258] can be aligned to the full length of pseudogenic regions with at least 80% identity. This poses a challenge to us as reads can sometimes be aligned to pseudogenic locations with higher percentages of identity than to their original location of transcription. For instance, a read can align in an ungapped fashion onto a pseudogenic region as an exact match but the correct alignment should be exact matches of two non-overlapping and adjacent substrings of the read marked by an exon-exon boundary (intronic gap on the reference genome) between them.

6.3 Related Work

Several alignment algorithms have been developed to align mRNA-seq reads [35-39, 41, 177, 180]. Here, we review some of the published RNA-seq aligners which we compared our methodology with in this section of the thesis. MapSplice uses consecutive contiguous 20-25 bp long seeds of a read to determine the candidate alignments of the mRNA read. Based on the seed locations, MapSplice will determine the most likely alignment of each mRNA read to a reference genome. Similar to SpliceMap, OLego also uses sub-sequences of a read to obtain anchor locations of an mRNA read. However, OLego uses relatively shorter seeds of 12-14 bp long and is aimed at sensitive recovery of micro-exon (~20 bp). STAR uses the idea of finding a Maximal Mappable Prefix (MMP) as its seed finding routine. This concept is similar to the maximal exact unique match used by genome alignment tools Mummer [135] and MAUVE [259]. Due to this, STAR is very efficient. However, the use of small seeds and MMP as the seed finding routine may produce spurious candidate locations, which are hard to disambiguate, and correct candidate locations may be missed respectively. This affects the accuracy of existing methods.

Our work described in this chapter, BatRNA, is based on a fast BWT data-structure for efficient detection of splice junctions and focuses on distinguishing spliced reads from exonic reads by using phased aligned strategies to handle each type of reads automatically with high sensitivity and accuracy. BatRNA is also the fastest method among the compared programs which use similar amount of physical working memory.

6.4 Results

The simulated data is produced by BEERS using hg19 configuration files which can be downloaded from the RUM website. The performance of aligning real data was evaluated with reads from ERP00196 (Sample: 11T). We will report the performance of BatRNA based on evaluation of aligning simulated and real data below.

6.4.1 Setup of experiments and performance measures used

In order to evaluate the performance of our method, we have benchmarked against the following programs: OLego v1.1.1, MapSplice v2.1.2, STAR 2.3.0e and TopHat2 v2.0.8b. Some RNA-seq reads can map to multiple genomic locations and since a read can only come from at most one point of origin we only validate unambiguous mappings which were indicated by a non-zero mapping quality.

Whenever ground truth is available, we will use F-measure to compare the mapping performance of all the compared methods. F-measure is defined as $(2 * (R * P)/(R + P))$, where R is Recall and P is Precision. As for real datasets, we will use the cumulative number of spliced mappings over edit-distances of ≤ 3 to compare the performance of the methods.

To address the lack of ground truth for the alignments of real data, we have adopted a variant of validation used in TopHat2 paper [42] which we will further elaborate in the later section to evaluate the alignment performance on real data.

All our experiments were run on a server equipped with Intel Xeon X5680 @ 3.33GHz and 48 GB of RAM. We allowed the same CPU threads on all the compared programs in our experiments. Other parameters were kept at default.

6.4.2 Evaluation on the simulated RNA-seq Illumina-like reads

We have used BEERS in the RUM package to simulate two datasets of 75 bp and 100 bp read-lengths. We aligned the reads in these datasets and summarized the number of correct and wrong alignments in Figure 6.1.

In terms of recall, BatRNA is second to MapSplice with ~1% lower recall on the simulated datasets. In terms of precision, BatRNA and OLego tied as the methods with the best precision. However, BatRNA was able to obtain the highest F-measures on aligning these two simulated datasets. This can be seen from Figure 6.1, that MapSplice being the best recall method, was ranked 4th out of our 5-methods comparison on precision. Despite having good prevision, OLego was ranked last for its low recall rates.

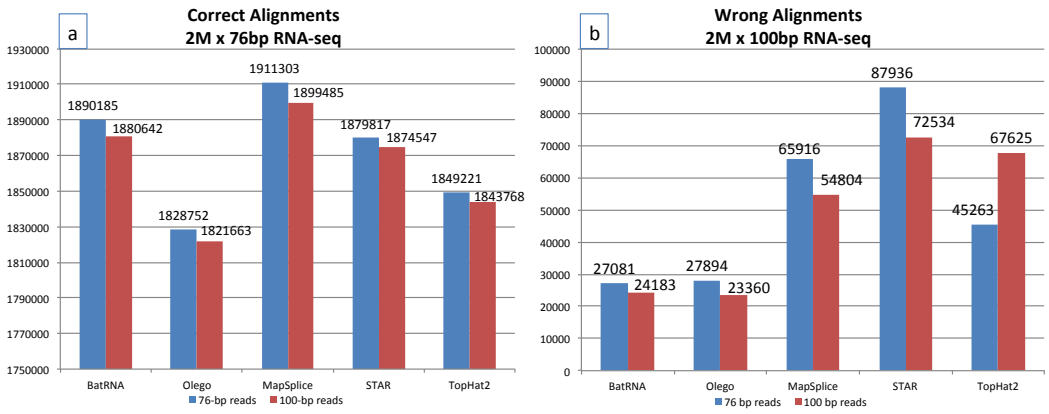


Figure 6.1. The counts of (a) correct alignments and (b) wrong alignments from the compared methods on 76 bp and 100 bp BEERS-simulated datasets.

The precision of alignments generally increases with increasing read-lengths as reads of longer read-lengths can be represented more uniquely on the reference that it is simulated

from than its shorter counterparts [114]. From Figure 6.1, we can observe such a general trend of increased specificity across all the compared methods except for TopHat2. Table 6.1 reports the F-measure of the various methods using oracle information available from the two simulated datasets.

Table 6.1. The F1-scores of the compared methods on BEERS-simulated 2M datasets.

Method	F-measure	
	76 bp	100 bp
BatRNA	**96.51%**	**96.32%**
OLego	94.84%	94.75%
MapSplice	96.11%	96.07%
STAR	94.75%	94.98%
TopHat2	94.97%	94.28%

RNA-seq datasets generally contain unequal proportions of exonic and spliced sequenced reads. When the read-length increases, the chances of a read spanning across an exon-exon boundary increase. At current popular RNA-seq read-lengths of ~100 bp, the proportion of exonic reads is expected to be ~82.2% from empirical studies of simulated reads using BEERS. As exonic reads are the dominant portion of alignments sequenced from a typical un-diseased human sample using reads of length ~100 bp, the accurate quantification of transcript abundance can be achieved solely with an unspliced aligner albeit the inability to identify exon-exon junctions in the sampled data. In order to gauge the true alignment performance of the compared splice alignment methods, we will procure the previously discussed simulated datasets, segregate the exonic and spliced reads from each other, align them and present their alignment results in Table 6.2 separately. Without loss of generality, this enabled us to gauge the alignment performance of the compared methods with better granularity on RNA-seq reads.

Table 6.2. Breakdown of alignment performance by exonic and spliced reads using simulation.

	Methods	Spliced read count	Exonic read count	Sensitivity (Spliced)	Precision (Spliced)	Sensitivity (Exonic)	Precision (Exonic)
2M x 76 bp	BatRNA	355811	1644189	**85.84%**	**92.93%**	96.39%	**99.76%**
	OLego			71.05%	90.93%	95.85%	**99.83%**
	MapSplice			**86.31%**	90.93%	**97.57%**	97.85%
	STAR			77.91%	82.46%	**97.47%**	98.22%
	TopHat2			75.51%	**93.28%**	96.13%	98.39%
2M x 100 bp	BatRNA	447170	1552830	**83.75%**	**94.80%**	96.99%	**99.76%**
	OLego			72.64%	93.95%	96.39%	**99.84%**
	MapSplice			**84.58%**	93.14%	**97.97%**	98.26%
	STAR			78.70%	87.52%	**98.05%**	98.55%
	TopHat2			76.97%	**94.54%**	96.57%	96.91%

The best-2 recalls and precisions scores of each experiment for this table are in bold.

Apart from reporting on non-ambiguous (mapQ > 0) hits, we also report on the sensitivity of the top-10 hits reported by each method to determine if multi-mappings from the other programs can correctly quantify transcript abundance. In addition to showing the rank of the reported correct hit among the top-10 multi-hits that a method has reported for a read, the tabulated statistics in Table 6.3a also indirectly showed the cumulative number of wrong hits that an aligner has reported by allowing the report of multi-hits. For instance, if aligner A was to report k number of rank-2 hits, it would also mean that aligner A has also reported k number of top-rank (rank-1) for k reads. It should also be noted that the correct hit for a read should preferentially be reported as a rank-1 hit. From Table 6.3a, BatRNA reported the least number of non rank-1 hits indicating its ability to discriminating against spurious hits effectively. Correspondingly, Table 6.3b tabulates the number of wrong multi-hits that were reported alongside with a rank-k correct hit.

Table 6.3a. Tabulation of correct hits ranked by the order in which they were reported for a read.

Methods	Rank of hits	75bp dataset #Correct hits 1	2	3	4	5	6	7	8	9	10
BatRNA	Spliced	335922	1272	312	96	56	49	23	14	5	2
OLego		280648	2381	651	197	111	66	26	23	18	6
MapSplice		336955	2232	462	122	48	29	5	2	1	0
STAR		335195	4085	826	241	170	104	38	21	7	5
TopHat2		283938	2278	627	133	77	27	6	21	20	11
BatRNA	Exonic	1585428	480	11	1	8	1	2	0	2	0
OLego		1595295	15038	2456	378	7	0	0	0	1	0
MapSplice		1604210	18779	5071	1986	1058	746	377	237	184	161
STAR		1603880	19726	5123	2059	1001	569	255	134	52	30
TopHat2		1582341	18820	5102	1971	1044	784	397	295	201	181

Table 6.3b. Tabulation of wrong hits being reported alongside a rank-k correct hit.

Methods	Rank of hits (k)	75bp dataset Cumulative #Wrong hits for reported correct rank-k hit 1	2	3	4	5	6	7	8	9	10
BatRNA	Spliced	0	1272	624	288	224	245	138	98	40	18
OLego		0	2381	1302	591	444	330	156	161	144	54
MapSplice		0	2232	924	366	192	145	30	14	8	0
STAR		0	4085	1652	723	680	520	228	147	56	45
TopHat2		0	2278	1254	399	308	135	36	147	160	99
BatRNA	Exonic	0	2163	940	306	208	115	54	63	88	0
OLego		0	480	22	3	32	5	12	0	16	0
MapSplice		0	15038	4912	1134	28	0	0	0	8	0
STAR		0	18779	10142	5958	4232	3730	2262	1659	1472	1449
TopHat2		0	19726	10246	6177	4004	2845	1530	938	416	270

k is an integer from 1 to 10 inclusive. k is used to denote the order of a hit in which it is reported. A correct hit of rank-k will also mean that it has generated (k-1) * #correct_rank-k_hits.

6.4.3 Evaluation on real RNA-seq Illumina-like reads

Although the lack of ground truth makes our validations much more difficult on real data, we would still like to use real data to provide a measure of corresponding performance in practice as if we were dealing with simulated data.

6.4.3.1 *Edit-distance as a measure of correctness in real-data set*

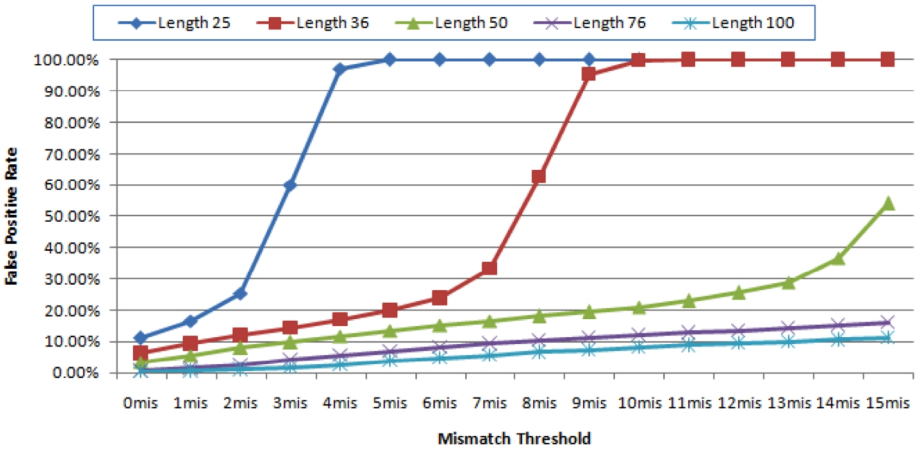


Figure 6.2. Chromosome-1 reads were mapped to a chromosome-1-deficit hg19. False positive rate was calculated by the number of simulated reads that were mapped to the modified hg19, divided by the total number of reads.

Due to the lack of ground truth, we adopted one experiment from TopHat2 paper whereby the authors estimated the performance of real-read alignment with cumulative number of alignments with edit distances of ≤ 3; assuming these candidate hits with low edit distance are correct. But first, we would have to study the behavior of this validation using simulated data. Using simulated reads from chromosome 1 of hg19, we mapped these reads back to a chromosome 1-deficit reference genome to investigate how increasing edit-distances actually correlate with the noise rates in our alignments. Figure 6.2 shows the false mapping rates of these simulated chromosome-1 reads, of various read-lengths, increased when the allowed maximum edit distance to their respective alignments also increased. As the false positive mapping rates for the reads become significant when edit distance of more than 3 was allowed for the alignments, we will only assume alignments with edit distance lower than or equals to 3 as correct in our experiments on real data.

First, we will apply this validation to the simulated dataset to observe the relationships of sensitivity and specificity with increasing edit-distances of alignments. Figure 6.3 is generated from the results of the immediate preceding section on simulated data. It gave us a feeling of how the trend of results for a method would behave with more/less and correct/wrong alignments. As we can see from Figure 6.3, the gain in recall rates is marginal as the edit distance of the alignments approaches 3. To add on, the relatively large drop in specificity with respective to the number of correct alignments with higher edit-distances discouraged us from using alignments with high edit-distance for downstream analysis too.

Using edit distances to test the goodness of spliced alignments is far from satisfactory as the results from Figure 6.3 and 6.4 do not agree with each other. From Figure 6.3, a method with many wrong and low edit distance alignments will have its wrong alignment eluded for scrutiny if they are represented as what has already been shown in Figure 6.4. Upon deeper investigation, we found out that the pseudogenic regions caused the disparity between the results. The reads were mapped to pseudogenic regions either with a lower edit distance or it is mapped preferentially without splicing junctions in them. For instance, a 0-edit distance spliced read can be mapped to a pseudogenic region in an unspliced fashion, both locations will yield 0-edit distance alignments, and still be considered as a good alignment under this form of validation. From this observation, the validation of alignment on real RNA-seq reads using edit distances will be more appropriate if it was applied solely to spliced alignments. This will directly prevent the wrong classification of spurious unspliced alignments as correct hits.

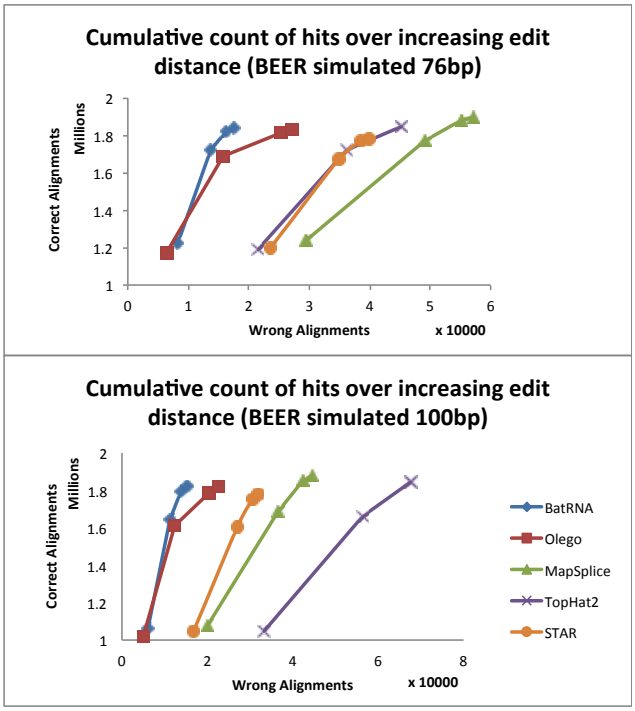


Figure 6.3. The counts of correct and wrong alignments for simulated RNA-seq 76bp and 100bp of 2 million reads each stratified by edit-distances of 0 to 3.

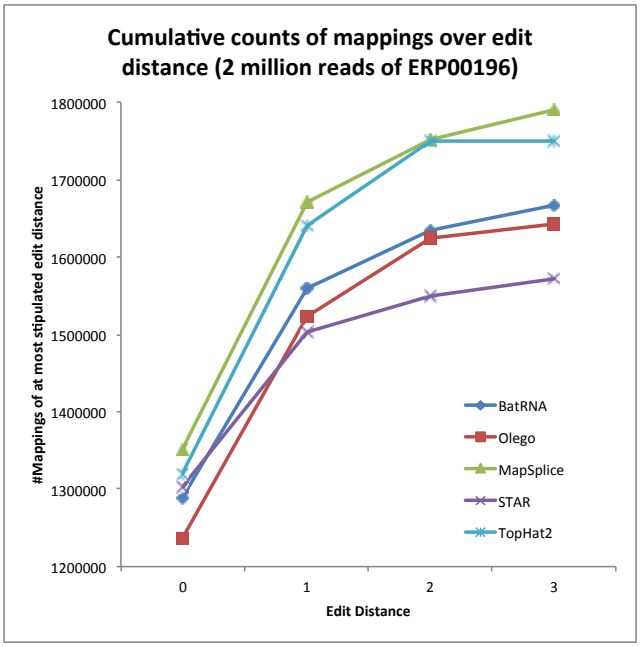


Figure 6.4. The cumulative counts, over edit distances of 0-3, of all non-ambiguous mappings from the various spliced mappers on 2 million real reads taken from Sample 11T of ERP00196.

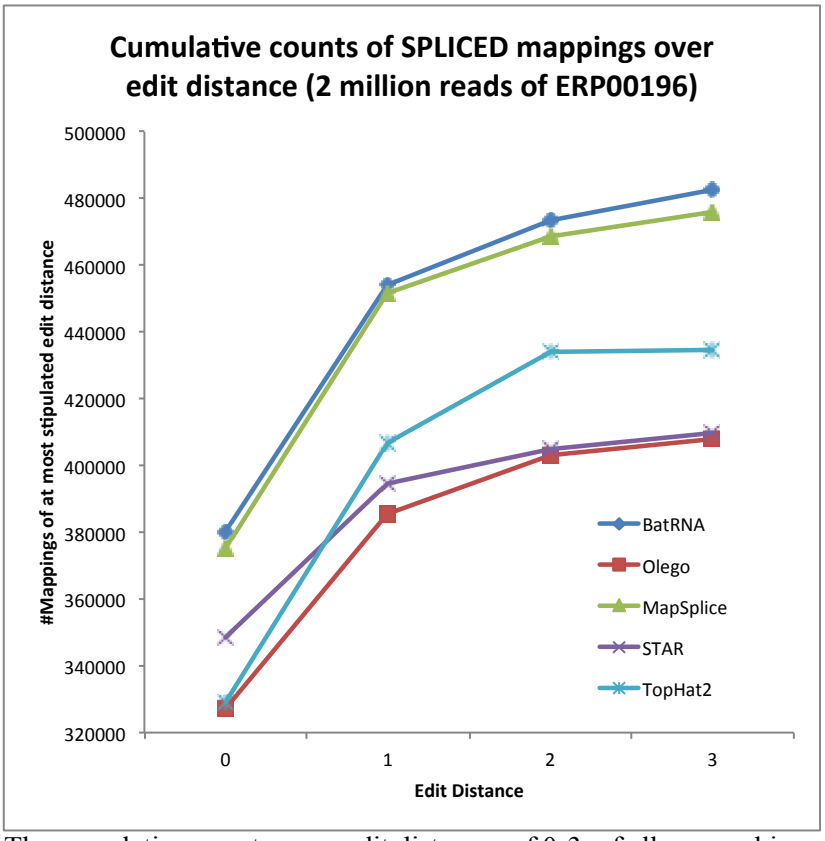


Figure 6.5. The cumulative counts, over edit distances of 0-3, of all non-ambiguous spliced mappings from the various spliced mappers on 2 million real reads taken from Sample 11T of ERP00196.

The results from the section on simulated data now coincide with the results shown in Figure 6.5. Without loss of generality, BatRNA was reported as the top performing method for both simulated and spliced alignments.

6.5 Evaluation on running time

The same sample of 2 millions reads, from the simulated datasets and patient 11T of ERP00196, were used to determine the runtime efficiency of the compared methods. The index-loading time was not recorded, as it does not reflect mapping efficiency and will be amortized to negligible timing over an actual life-sized dataset. The start time, wall-clock times, were recorded when the threads reached ~100% efficiency (indicating the

complete loading of the primary index of the reference genome) and the end times were marked by the termination of their execution. Table 6.4 reports the recorded wall-clock times for dataset of different origins and read-lengths. STAR is the fastest method due to the search for MMP on a 29.8 GB human reference genome index. The runner-up method, in terms of running time, would be BatRNA. We also executed Tophat2 with the parameters "--no-sort-bam" and "--no-convert-bam" to avoid it from incurring additional execution times due to non-mapping related operations.

Table 6.4. Wall-clock time of compared methods on different sets of 2 million reads.

	Runtime on 2 million reads (seconds)		
Method	BEER 76 bp	BEER 100 bp	Real 90 bp
BatRNA	72	82	92
OLego	239	237	272
MapSplice	235	277	418
STAR	13	14	11
TopHat2	630	709	694

We have also observed that although only ~20% of the reads will have their primary candidate locations passed onto the second phase, this small portion of reads will take up more than 80% of the total runtime needed to run datasets with read-length of ~100 bp. Overall, BatRNA offers considerable improvements in alignment efficiency over the other compared methods with similar physical working memory footprint of << 30 GB.

6.6 Methods

BatRNA (Basic Alignment Tool for RNA-seq) was developed to address the issues of efficiency and accuracy on performing spliced-alignment of RNA-seq reads. BatAlign was used to align and produce candidate alignments of the reads. By emulating paired-end information in a single-read, an efficient pairing data structure was used to exhaustively search for the presence of splice junctions in a read. The putative mappings

from both BatAlign and the splice-detection algorithm were then ranked according to their alignment score and reported to the users.

6.6.1 Simulation of data and validation of simulated data

The BEERS package in RUM is used to simulate the RNA-seq reads used in our benchmarks. 2 millions reads are simulated for current popular read lengths of 76bp and 100 bp. We have used BEERS as the simulator as it is built on an extensive platform of oracle information from 11 sets of annotations, namely, AceView, Ensembl, Geneid, Genscan, NSCAN, RefSeq, SGP, Transcriptome, UCSC, Vega, Other RefSeq databases. BEERS was also trained from these annotations and is able to simulate ~1.7M exons and ~1.1 introns, based on ~672K distinct gene models, with ground truth for validation.

For the aligned location of the simulated reads to be considered correct, the reported locations must be within 50 bp of the locations generated by the simulator. For simulated spliced reads, in additional to the condition required for simulated exonic reads to be considered correct, we required that at least one of its simulated intronic gap(s) correctly identified before its reported alignment is considered correct, else our verifier will consider the reported alignment as an erroneous alignment.

We have also further broken down the mixture of exonic and spliced reads in the BEERS-simulated dataset for clearer illustrations of how the compared spliced aligner perform on each type of these reads.

6.6.2 Overview of Method

RNA-seq aligners can be classified into two main approaches: Exon-first and Seed-extend approaches. To the best of our knowledge, BatRNA is the first method that uses a pre-mapping tools meant for DNA-seq reads but can still be considered as a seed-extend approach. The reason that we developed BatRNA as a seed-extend approach is that it

provides an unbiased mapping over both exonic and spliced reads. BatRNA is a three-phased method whereby the first phase is to find the list of candidate locations for the contiguous exonic region within a read, the second phase is to map the unmapped reads or low quality hits from the initial phase using a k-mer splicing seed-extend strategy and the last phase is to refine the alignments, from the previous phases, to identify splice junctions accurately. Figure 6.6 shows the schematic workflow of aligning RNA-seq reads using the methodology implemented in BatRNA.

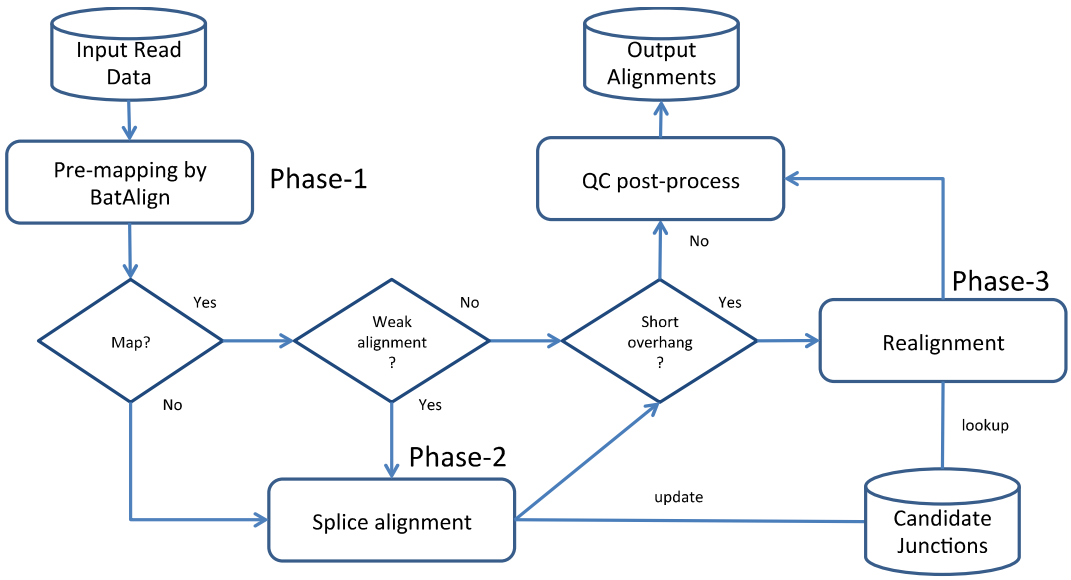


Figure 6.6. A schematic flowchart showing how input RNA-seq reads is aligned using the 3-phased methodology of BatRNA.

6.6.3 Motivation for using BatAlign as a seeding tool

As we have observed from BEERS-simulated reads of 100 bp long, ~62% of them can be mapped as if they are DNA-seq reads without the presence of large intronic gaps within them. Aside from this, less than 5% of exons have lengths shorter than 42 bases. This means that BatAlign can be used to seed the mapping location of the longest exonic

region within a junction read successfully with its long mismatch-gapped seed on a 100 bp RNA-seq fragment.

The efficacy of using BatAlign is further highlighted by its ability to accurately align genomic reads. On the dataset of 100 bp RNA-seq reads, it was able to map 97.3% and 74.4% of the simulated exonic and spliced junction reads with an accuracy of 99.8% on the exonic reads; leaving only 9.4% of the dataset unmapped. The percentage of reads, from a general 100 bp RNA-seq dataset, that was delegated onto the later phases of BatRNA to align is ~18%.

6.6.4 Phase 1 – Resolve exonic region within a single read

The first phase of BatRNA is to use BatAlign as a seeding tool to align the input RNA-seq reads. For the putative alignments from BatAlign, we will assign a mapping quality score and a text-edit CIGAR string to each of them. If the mapping quality score is low, Phase-2 and Phase-3 of BatRNA will remap these reads, and this could mean three things. Firstly, the putative exonic alignment is repetitive due to its location in repeat or pseudogenic regions. Secondly, the alignment is weak due to high number of text-edit operations required to align the read back onto the reference genome. Thirdly, the putative alignment is heavily clipped with only a small percentage of the read being aligned to the reference genome by a local alignment routine. The last, unmentioned and trivial case of a read from BatAlign is that it is left unmapped by BatAlign.

Figure 6.7 shows the possible alignments of an RNA-seq read which spans across single or multiple exon-exon boundaries. This figure shows the possibility of using a DNA-seq gapped aligner as a pre-mapping tool for RNA-seq reads but still retains the unbiased alignment property towards both exon and junction reads of the seed-extend methodology towards RNA-seq alignment. Unlike TopHat1/2 that only realign primary candidate

alignments with a certain threshold of edit distance on them from pre-mapping tools, BatRNA takes into account for the existence of splice junctions even at the DNA-seq gapped alignment step.

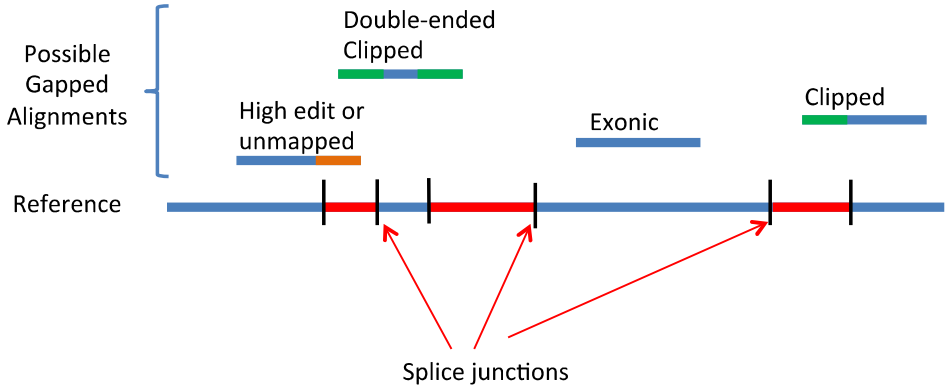


Figure 6.7. Possible alignments on RNA-seq read from BatAlign.

An example CIGAR string given to a simulated junction read by BatAlign would be "66M34S". The largest matching segment will be the first 66 bases of the read and this matching sequence will be treated as an exonic transcript that lies on the left of an exon-exon boundary of the sample transcriptome. As described later, Phase-2 of BatRNA will align the rest of the remaining clipped 34 bases of the read downstream, at most 20 kbp away, of the anchored longest exonic region of this read.

6.6.5 Phase 2 – Search for junctions from an anchored region

There are two types of reads that will be passed into this phase of BatRNA: low-quality alignments and unmapped alignments from Phase-1. Figure 6.8 shows a flowchart on how these two types of reads are processed by the splice alignment algorithm in BatRNA. We will start to explain Phase-2 with the similar but simpler case of unmapped alignments. BatRNA's splice mapping algorithm is based on a perfect-matching seed-extend-pairing strategy. It will first align the first two adjacent non-overlapping 18-mer

of a read segment. If the first 18-mer cannot be anchored, then 5 bases are trimmed from the 5' end of the read for each time it fails to anchor itself. If the first 18-mer is anchored then the second 18-mer of the read will be aligned and be paired using an efficient pairing (shown in Figure 6.9a) data structure to form a 36 bp exonic segment of the read. However, if the immediate 18-mer cannot be anchored (shown in Figure 6.9b), Phase-2 will try to pair the anchored portion of the read with the next adjacent and non-overlapping 18-mer of the read, this will continue until the end of the read. In the event that two 18-mer can be paired up successfully within the neighbor of each other (within 20 kbp), we will extend the paired candidate alignments of the two 18-mers, called gap-filling (shown in Figure 6.9c), towards each other, while respecting the donor-recipient canonical/non-canonical splicing signals. In the event, whereby there are more than one possible candidate location which can be paired with the anchored region of the read, the splice junctions detected by the gap-filling procedure are stored in a candidate junction files for use in Phase-3.

The second type of input to Phase-2 is partial-alignments from Phase-1. If the longest contiguous matching sequence of the partial-alignment is at least 25 bp then the partial-alignment is discarded and the read is treated as unmappable by Phase-1. We have decided on this threshold because the smallest seed used in BatAlign is 25 bp. From here, the longest contiguous matched sequence is treated as the anchored alignment and due to the presence of clippings in the partial-alignments, a junction is assumed to exist within or before the next 18-mer that needs to be aligned. Hence, the second 18-mer away from the clipped location will be aligned and paired with the already anchored partial alignment. If the 18-mer can be paired up then it will be extended towards the previously found partial-alignment (as shown in Figure 6.9d); if not, the algorithm will recursively proceed to the next non-overlapping 18-mer as described in the preceding paragraph.

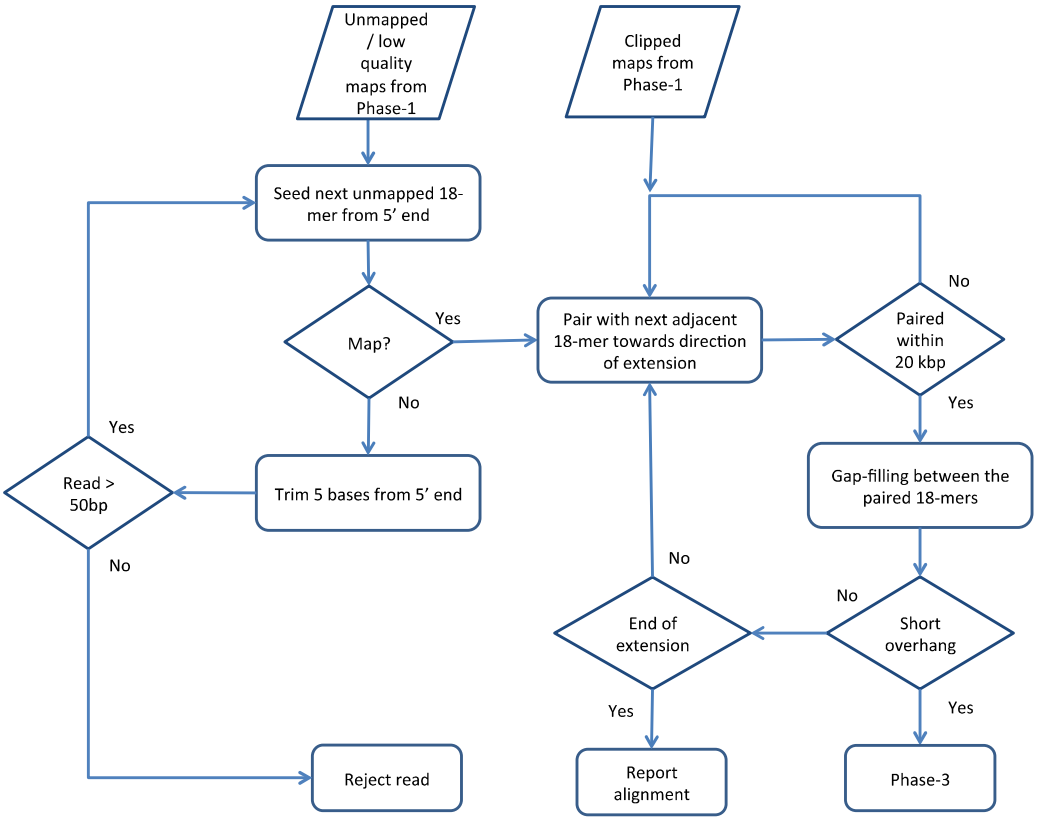


Figure 6.8: A flowchart showing how the splice alignment algorithm in BatRNA performs splice alignment.

During the extension of the partial-alignments, short-overhanging exons can exist at the ends of the reads that are due to the presence of splice junctions being sequenced into the near-ends of the reads, these short-overhangs will be soft-clipped by Phase-2. Phase-3 will refine these exon-exon junctions that appears as soft-clippings in the reads from both Phase-1 and Phase-2.

6.6.6 Phase 3 – Refine alignments due to splice junctions near ends of reads

Alignment has always been an independent event between reads until TopHat devised the idea of exon-islands to localize putative splice junctions without annotations. By assembling the consensus of regions covered by the alignments of exonic reads from a

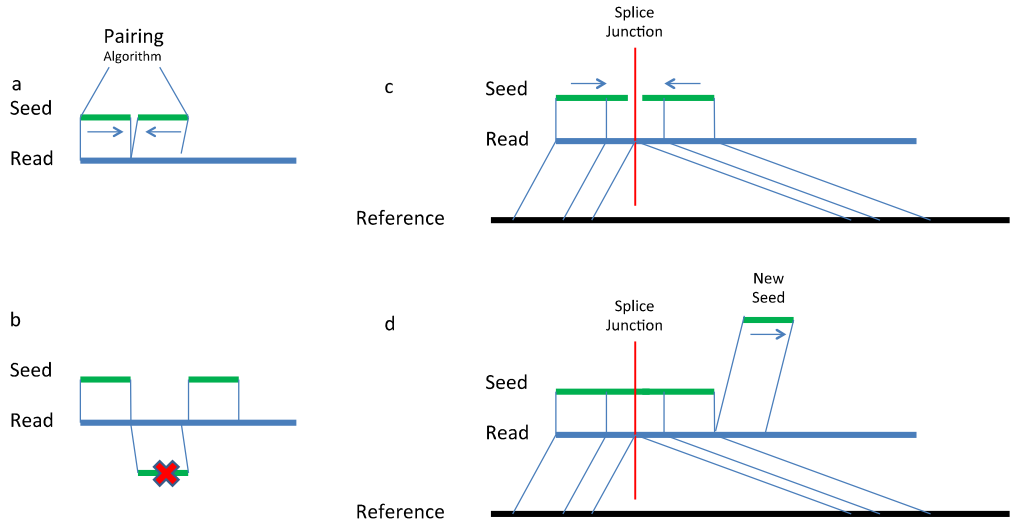


Figure 6.9. Schematic sketches of some possible scenarios that can happen in BatRNA splice algorithm. a) Adjacent non-overlapping seeds do not span across exon-exon junctions. b) Anchored seed is near to an exon-exon junction and next immediate 18-mer is used to seed the alignment. c) After successfully pairing of seeds within spanning distance of 20 kbp, alignments are extended towards each other to recover the splice junction on the reference genome. d) New seed is selected for the continual extension of a current partially anchored alignment.

Bowtie, exon islands can be obtained. Splice-junctions are then localized near the vicinity of these exon-islands.

Different from TopHat, the gap-filling component in Phase-2 of BatRNA has already identified the putative splice junctions. The unsupervised learning of splice junctions is done whenever two adjacent non-overlapping 18-mers from a read are aligned more than 20 bp apart and a gap-filling procedure is done to identify the splice junctions. These splice junctions are stored in a putative bed-coverage file for Phase-3 to refine the alignments of short-overhangs. For instance, the cigar string "12M2439N88M" was previously "11S89M" for read "CGAGAGCTAAAGGAGGTCTTTGGTGATGAC TCTGAGATCTCTAAAGAATCATCAGGAGTAAAGAAGCGACGAATACCCCGTT TTGAGGAGGTGGAACAAG". The 11 clipped bases are then locally aligned to each of the candidate splice junctions, within 20 kbp of the anchored 89 contiguous exonic bases,

recorded by Phase-2. Figure 6.10 shows the possible inputs into Phase-3 that are realigned around a putative splice junction from the splice alignment algorithm in Phase-2. After which, the mappings are scored similarly with a scoring function similar to BatAlign. In the event that there are more than two candidate alignments to a read, the donor-recipient splicing signals will precede over the total intronic gap sizes in an alignment as a tiebreaker.

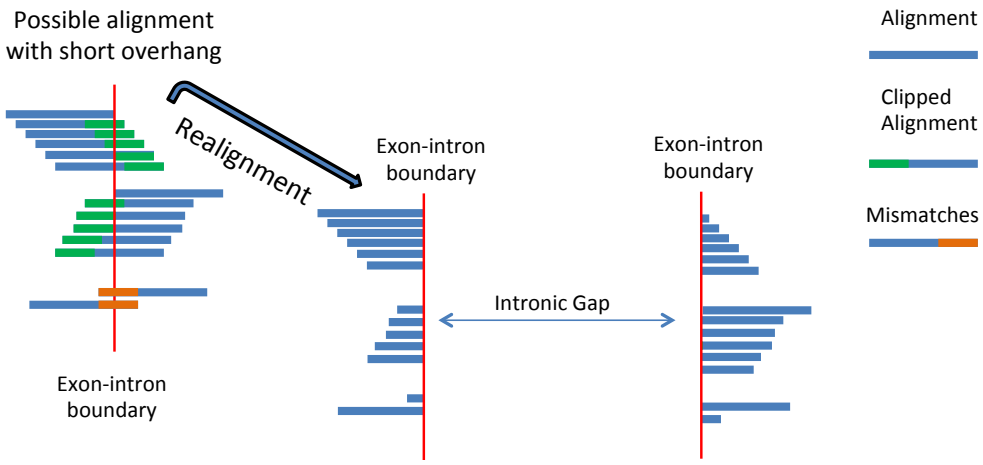


Figure 6.10. Possible short overhangs being recovered with local alignment by using preceding prediction as a guide in an unsupervised manner.

The coverage or the transcript-abundance of the dataset simulated or sequenced will matter for the performance of this realignment step. If the depth of coverage is low, Phase-2 may not be able to detect the splicing junctions needed for the realignment of short-overhang in a read. As such, the same read with alignment CIGAR "11S89M" will be left unchanged after Phase-3 is complete.

6.6.7 Data structure for efficient pairing of genomic coordinates

As the entire array of genomic locations is too large to fit into the main memory of common personal computers, genomic locations are often sampled at a fixed *k* interval to keep the index compact. However, after the alignment of a read is done on the suffix

array, the intermediate suffix-interval can only be converted to a genomic location by referencing the sampled array in $O(\log|Reference|)$ time. If s is the number of steps needed to invert the suffix-interval back onto a sampled location, then the actual location of the alignment can be calculated from $[(Sampled\ location) - s]$. As each occurrence of the aligned read needs to be inverted back to a genomic location seperately and independently, the total time needed to find all the genomic locations represented by the suffix-intervals is $O(|Occurrences| . \log|Reference|)$ time.

However, if we pre-compute and hash the genomic locations of the k-length string, the genomic locations can be retrieved in $O(|Occurrences| + \log|Reference|)$ time instead. Furthermore, if we pre-process the hashed genomic locations by sorting them, we can pair the genomic locations for two k-length strings within a distance D in $O(|Occurrences|)$ time. The data-structure used is a hash-map with the suffix-array interval and genomic locations as a key and value pair respectively. The building of this data structure is only a 1-time off effort. In order to avoid large memory overhead, only strings that have occurrences of more than 1 and less than 200 are hashed by our index-building routine. This data structure will incur 2.5 GB of the total memory footprint of BatRNA.

6.6.8 Details of implementation

The length of the seed is chosen as 18 bp long to represent more than 99% of UCSC RefSeq exon-lengths on a genomic reference without spanning over an exon-exon boundary [175, 176]. As 18 bp fragment can be over-represented spuriously on the reference genome for BatRNA to align efficiently, we do not allow any mismatches or/and gap in the 18-mer seed in our splice alignment algorithm. For the efficiency of the method, the maximum distance between adjacent exons within a read has to be within 20

kbp to each other. This threshold on the intronic gap size constitutes for less than 6.1% of the human genome [260].

6.6.9 Discussion

In this chapter, we represented BatRNA; a method that emulates pair-end information within a single RNA-seq read for efficient alignment of high throughput datasets from next generation sequencing technologies. Since the introduction of high throughput sequencing technologies, the dominant improvements brought about by it are increase in throughput and read-length. However, with longer read-lengths, RNA-seq alignment algorithms has to be developed to account for the long intronic gap that can exist in a RNA-seq read which are much larger than an indel gap in DNA-seq read. In order to handle these large intronic gaps, pioneering alignment tools for RNA-seq reads align reads gaplessly onto a pre-constructed reference of known transcripts, which is also known as transcriptome. This strategy is very efficient as gapless alignment can be performed efficiently using gapless BWT-based aligner in *O(Read-Length)* time for each read. However, using the transcriptome as the reference text to align RNA-seq reads with a gapless aligner will void us of doing de novo detection of novel splice junctions. Albeit a non *de novo* methodology, this strategy was extended and gave birth to the development of popular exon-first strategy such as TopHat (using Bowtie as premapping tool). By weeding out the exonic seeds out before the computationally dominant splice alignment algorithms align the unmapped spliced reads, exon-first approaches generally align faster than seed-extend approaches.

The main shortcoming of exon-first is that it favors towards the alignment of exonic reads over spliced reads. In other words, exonic reads from RNA-seq experiments may align more often to pseudogenic regions erroneously than seed-extend methods. TopHat2 tried to minimize this error rate by realigning reads of a certain threshold of edit-distance

(capped at 3) from its DNA-seq gapped aligner through its seed-extend splice alignment routine in hope of realigning the same read with an exonic alignment to a splice alignment instead. Seed-extend was introduced by pioneering methods such as BLAT and exonerate to unbiasedly map RNA-seq reads regardless of the existence of splice junctions in the reads to the genomic reference. By picking the correct seed length and the intervals between each subsequent seeds on a read are critical to the success of a seed-extend approach. A long read-length will cause the seed to incur a high edit distance and miss the correct alignment. The lengths of the seeds are generally short in order to achieve good sensitivity. Specificity is dependent on how well the seeds are sampled such that the short seeds can capture the correct alignment of the read. For instance, BLAT indexes all the non-overlapping 11 bp tiles to achieve good alignment of long EST/cDNA sequences [32]. Additional heuristics such as perfect seed matches are also used to limit the number of preliminary partial alignments for post-processing for reasonable alignment efficiency.

BatRNA was developed as a hybrid between exon-first and seed-extend approaches. This was done to complement the shortcomings of both approaches under one unified method. In order to reduce the number of computationally expensive splice alignments needed to recover splice junctions, Phase-1 uses a gapped aligner that can align ~91% of reads in a general 100 bp read-length RNA-seq dataset. From here, the unmapped and low quality alignments from Phase-1 are realigned with our splice algorithm. Next, we reckon that 18 bp will produce a lot of spurious partially alignment to post-process and, as such, an efficient pairing data-structure was developed to align pairs of 18 bp in our splice alignment efficiently. Using our pairing data structure, we are effectively aligning a much longer seed of 36 bp (2 x 18 bp) than the current seed-extend methods seed-lengths of 10-25 bp such as SubRead [160].

In summary, our experiments have shown that BatRNA have achieved better sensitivity and specificity in handling RNA-seq reads of current common read-lengths on a reference genome than other compared methods. BatRNA is also the most efficient program among the compared methods of similar memory footprints.

Chapter 7

Conclusion

In this chapter, we review on the main contributions of this thesis and discuss some of the possible future directions that can be adopted to further improve on the proposed methodologies.

You should have already known, the purpose of this thesis is to report on new methodologies that provide accurate alignment of sequence reads from various genomic origins. In this thesis, we have handled bisulfite-treated, gapped and spliced reads.

7.1 BatMeth

The alignment of a read, against a reference genome indexed by the FM-index, comprises of two main sequential steps. The first step involves the retrieval of the suffix-array intervals that represents the occurrences of the read in the reference-index. The second step, where the bulk of computation takes place, involves the conversion of the indices to genomic locations that will then be reported to the user.

In BatMeth, List Filtering performs the alignment of a read solely by counting the number of occurrences of the reads on each possible orientation of the DNA without using the second step of alignment. Counter-intuitively, List Filtering improved the sensitivity, specificity and speed of the methods. BatMeth was also developed to account for mismatches attributed from deamination or/and sodium bisulfite-induced base conversion in both base-space (Illumina reads) and color-space reads (SOLiD reads) correctly. Experiments have also shown that BatMeth aligned two types of sequenced reads on different genomic context (CG, CHH, CHG; where H ≠ G) and different levels of induced methylation with less bias. Bisulfighter [261] has also commented that BatMeth is the current best method in making the binary decision on whether a base is methylated or unmethylated.

7.2 BatAlign

The pioneering aligners for next generation sequencing reads were originally designed to handle only mismatches, with respect to the reference genome, in the reads. As genomic polymorphism can also comprises of indels and genomic rearrangements, gapped aligners were developed to better study the complex nature of polymorphisms in both normal and diseased genomes.

Since a sequenced read can be transformed back to the reference genome through a sequential order of text-edit operations, we can also score such a transformation by assigning scores to the text-edit operations, namely, match, mismatch and gap-open. By enumerating the number of text-edit operations to a read, we can rank the possible combinations of edit operations needed to align a read. BatAlign uses *Reverse-alignment* to incrementally align a read in increasing order of alignment cost with the combination of match/mismatch/gap defined by the scores assigned to them. In addition, *Deep-scan* was developed so that BatAlign will be able to better differentiate a real-SNP mismatch

from a false base-call mismatch during alignment. Experiments have shown that BatAlign was able to map SNV- and indel-spanning reads (75 to 250 bp long) with high sensitivity and accuracy over a large range of assigned mapping quality scores.

Paired-reads are first aligned in a single-read fashion and are later paired up unbiasedly to yield accurate alignments. Chimeric/supplementary alignments are also reported for a single read, under the pair-end mapping mode of BatAlign, to better support the identification of breakpoints caused by genomic rearrangements. In general, BatAlign is an improved method for gapped reads.

7.3 BatRNA

The advent of RNA-seq allows scientists to quantify gene expression on a genome-wide scale. As RNA-seq reads can span across different exons, they can be challenging to be aligned back onto a reference genome.

BatRNA was developed as a hybrid between both exon-first and seed-extend methodologies. BatAlign was used as a non-splice pre-mapping aligner and a splice-alignment routine, which emulated paired-end information within a single read, was used to align the clipped and unmapped reads from BatAlign. The experiments have also shown that BatRNA has achieved accurate alignments on both exonic and spliced reads from the human transcriptome. It is also time-efficient when compared to other methods of similar memory usage.

7.4 Future Developments

As sequencing technologies continue to develop, the error profiles, which come along with these technologies, will also evolve with them. For instance, when read length gets longer from the oncoming third generation sequencing technologies, the total edit distance in a single read will require re-development of existing algorithms to better

handle such challenges. Homopolymers might be a prevalent type of sequencing errors over wrong base-calls too and aligners will have to handle these types of gap-errors efficiently too.

Alongside with sequencing technologies and alignment algorithms, genomic assemblers are also producing better scaffold reference genomes for scientists to work with. For instance, the recent release of the GRCh38 human genome has included 261 alternate loci, which are highly similar to the main loci of the GRCh19 genome. The alignment of read to the newer, GRCh38, genome will of course yield more non-unique alignments as these 261 alternate loci are supposed to be of high similarity to the main chromosomal sequences. In the future, aligners should be aware if an alignment is from either the main or alternate loci and should not assign marked it with a low uniqueness score if a read is aligned to such regions.

Alignment of genomic reads can also be tackled from the reference index's point of view. Since metagenomic projects are producing whole genome data of high similarity, compressing algorithms and data structures can be developed for more efficient memory usage without comprising on alignment efficiency.

Bibliography

1. Darwin C: **On the origins of species by means of natural selection.** *London: Murray* 1859.
2. Mendel G: **Versuche über Pflanzenhybriden.** *Verhandlungen des naturforschenden Vereines in Brunn 4: 3* 1866, **44**.
3. Avery OT, MacLeod CM, McCarty M: **Studies on the chemical nature of the substance inducing transformation of pneumococcal types induction of transformation by a desoxyribonucleic acid fraction isolated from pneumococcus type III.** *The Journal of experimental medicine* 1944, **79:**137-158.
4. Watson JD, Crick FH: **Molecular structure of nucleic acids.** *Nature* 1953, **171:**737-738.
5. Franklin RE, Gosling RG: **Evidence for 2-chain helix in crystalline structure of sodium deoxyribonucleate.** *Nature* 1953, **172:**156-157.
6. Wilkins MH, Seeds WE, Stokes AR, Wilson HR: **Helical structure of crystalline deoxypentose nucleic acid.** *Nature* 1953, **172:**759-762.
7. Sanger F: **DNA Sequencing with Chain-Terminating Inhibitors.** *Proceedings of the National Academy of Sciences* 1977, **74:**5463-5467.
8. Maxam AM: **A New Method for Sequencing DNA.** *Proceedings of the National Academy of Sciences* 1977, **74:**560-564.
9. Noble I: **Human genome finally complete.** In *BBC News*; 2003.
10. Maher B: **ENCODE: The human encyclopaedia.** *Nature* 2012, **489:**46-48.
11. Bernstein BE, Birney E, Dunham I, Green ED, Gunter C, Snyder M: **An integrated encyclopedia of DNA elements in the human genome.** *Nature* 2012, **489:**57-74.
12. Sanger F, Air GM, Barrell BG, Brown NL, Coulson AR, Fiddes CA, Hutchison CA, Slocombe PM, Smith M: **Nucleotide sequence of bacteriophage phi X174 DNA.** *Nature* 1977, **265:**687-695.
13. Sanger F, Coulson AR: **A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase.** *J Mol Biol* 1975, **94:**441-448.
14. Liu L, Li Y, Li S, Hu N, He Y, Pong R, Lin D, Lu L, Law M: **Comparison of next-generation sequencing systems.** *J Biomed Biotechnol* 2012, **2012:**251364.
15. Shendure J, Porreca GJ, Reppas NB, Lin X, McCutcheon JP, Rosenbaum AM, Wang MD, Zhang K, Mitra RD, Church GM: **Accurate multiplex polony sequencing of an evolved bacterial genome.** *Science* 2005, **309:**1728-1732.
16. Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, Bemben LA, Berka J, Braverman MS, Chen YJ, Chen Z, Dewell SB, Du L, Fierro JM, Gomes XV, Godwin BC, He W, Helgesen S, Ho CH, Irzyk GP, Jando SC, Alenquer ML,

Jarvie TP, Jirage KB, Kim JB, Knight JR, Lanza JR, Leamon JH, Lefkowitz SM, Lei M, Li J, et al: **Genome sequencing in microfabricated high-density picolitre reactors.** *Nature* 2005, **437:**376-380.

17. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP, Evers DJ, Barnes CL, Bignell HR, Boutell JM, Bryant J, Carter RJ, Cheetham RK, Cox AJ, Ellis DJ, Flatbush MR, Gormley NA, Humphray SJ, Irving LJ, Karbelashvili MS, Kirk SM, Li H, Liu XH, Maisinger KS, Murray LJ, Obradovic B, Ost T, Parkinson ML, Pratt MR, et al: **Accurate whole human genome sequencing using reversible terminator chemistry.** *Nature* 2008, **456:**53-59.
18. McKernan KJ, Peckham HE, Costa GL, McLaughlin SF, Fu Y, Tsung EF, Clouser CR, Duncan C, Ichikawa JK, Lee CC, Zhang Z, Ranade SS, Dimalanta ET, Hyland FC, Sokolsky TD, Zhang L, Sheridan A, Fu H, Hendrickson CL, Li B, Kotler L, Stuart JR, Malek JA, Manning JM, Antipova AA, Perez DS, Moore MP, Hayashibara KC, Lyons MR, Beaudoin RE, et al: **Sequence and structural variation in a human genome uncovered by short-read, massively parallel ligation sequencing using two-base encoding.** *Genome Res* 2009, **19:**1527-1541.
19. Ronaghi M: **DNA SEQUENCING:A Sequencing Method Based on Real-Time Pyrophosphate.** *Science* 1998, **281:**363-365.
20. Eid J, Fehr A, Gray J, Luong K, Lyle J, Otto G, Peluso P, Rank D, Baybayan P, Bettman B, Bibillo A, Bjornson K, Chaudhuri B, Christians F, Cicero R, Clark S, Dalal R, Dewinter A, Dixon J, Foquet M, Gaertner A, Hardenbol P, Heiner C, Hester K, Holden D, Kearns G, Kong X, Kuse R, Lacroix Y, Lin S, et al: **Real-time DNA sequencing from single polymerase molecules.** *Science* 2009, **323:**133-138.
21. Drmanac R, Sparks AB, Callow MJ, Halpern AL, Burns NL, Kermani BG, Carnevali P, Nazarenko I, Nilsen GB, Yeung G, Dahl F, Fernandez A, Staker B, Pant KP, Baccash J, Borcherding AP, Brownley A, Cedeno R, Chen LS, Chernikoff D, Cheung A, Chirita R, Curson B, Ebert JC, Hacker CR, Hartlage R, Hauser B, Huang S, Jiang Y, Karpinchyk V, et al: **Human Genome Sequencing Using Unchained Base Reads on Self-Assembling DNA Nanoarrays.** *Science* 2010, **327:**78-81.
22. Thompson JF, Steinmann KE: **Single molecule sequencing with a HeliScope genetic analysis system.** *Curr Protoc Mol Biol* 2010, **Chapter 7:**Unit7 10.
23. Ng SB, Turner EH, Robertson PD, Flygare SD, Bigham AW, Lee C, Shaffer T, Wong M, Bhattacharjee A, Eichler EE, Bamshad M, Nickerson DA, Shendure J: **Targeted capture and massively parallel sequencing of 12 human exomes.** *Nature* 2009, **461:**272-276.
24. Ng SB, Buckingham KJ, Lee C, Bigham AW, Tabor HK, Dent KM, Huff CD, Shannon PT, Jabs EW, Nickerson DA, Shendure J, Bamshad MJ: **Exome sequencing identifies the cause of a mendelian disorder.** *Nat Genet* 2010, **42:**30-35.
25. Acinas SG, Sarma-Rupavtarm R, Klepac-Ceraj V, Polz MF: **PCR-induced sequence artifacts and bias: insights from comparison of two 16S rRNA clone libraries constructed from the same sample.** *Appl Environ Microbiol* 2005, **71:**8966-8969.
26. Ingram VM: **A Specific Chemical Difference Between the Globins of Normal Human and Sickle-Cell Anæmia Hæmoglobin.** *Nature* 1956, **178:**792-794.
27. Niidome T, Huang L: **Gene therapy progress and prospects: nonviral vectors.** *Gene Ther* 2002, **9:**1647-1652.

28. Riddihough G, Zahn LM: **Epigenetics. What is epigenetics? Introduction.** *Science* 2010, **330:**611.
29. Bernstein BE, Meissner A, Lander ES: **The mammalian epigenome.** *Cell* 2007, **128:**669-681.
30. Treangen TJ, Salzberg SL: **Repetitive DNA and next-generation sequencing: computational challenges and solutions.** *Nat Rev Genet* 2012, **13:**36-46.
31. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool.** *J Mol Biol* 1990, **215:**403-410.
32. Kent WJ: **BLAT - The BLAST-like alignment tool.** *Genome Res* 2002, **12:**656-664.
33. **Animal cell structure en by LadyofHats** [http://commons.wikimedia.org/wiki/File:Animal_cell_structure_en.svg#mediaviewer/File:Animal_cell_structure_en.svg]
34. Wang Z, Gerstein M, Snyder M: **RNA-Seq: a revolutionary tool for transcriptomics.** *Nat Rev Genet* 2009, **10:**57-63.
35. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, Batut P, Chaisson M, Gingeras TR: **STAR: ultrafast universal RNA-seq aligner.** *Bioinformatics* 2013, **29:**15-21.
36. Wu J, Anczukow O, Krainer AR, Zhang MQ, Zhang C: **OLego: fast and sensitive mapping of spliced mRNA-Seq reads using small seeds.** *Nucleic Acids Research* 2013.
37. Wang K, Singh D, Zeng Z, Coleman SJ, Huang Y, Savich GL, He X, Mieczkowski P, Grimm SA, Perou CM, MacLeod JN, Chiang DY, Prins JF, Liu J: **MapSplice: accurate mapping of RNA-seq reads for splice junction discovery.** *Nucleic Acids Research* 2010, **38:**e178.
38. Zhang Y, Lameijer EW, t Hoen PA, Ning Z, Slagboom PE, Ye K: **PASSion: a pattern growth algorithm-based pipeline for splice junction detection in paired-end RNA-Seq data.** *Bioinformatics* 2012, **28:**479-486.
39. De Bona F, Ossowski S, Schneeberger K, Ratsch G: **Optimal spliced alignments of short sequence reads.** *Bioinformatics* 2008, **24:**i174-180.
40. Au KF, Jiang H, Lin L, Xing Y, Wong WH: **Detection of splice junctions from paired-end RNA-seq data by SpliceMap.** *Nucleic Acids Research* 2010, **38:**4570-4578.
41. Trapnell C, Pachter L, Salzberg SL: **TopHat: discovering splice junctions with RNA-Seq.** *Bioinformatics* 2009, **25:**1105-1111.
42. Kim D, Pertea G, Trapnell C, Pimentel H, Kelley R, Salzberg SL: **TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions.** *Genome Biol* 2013, **14:**R36.
43. Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, Pimentel H, Salzberg SL, Rinn JL, Pachter L: **Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks.** *Nature Protocols* 2012, **7:**562-578.
44. Li B, Dewey CN: **RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome.** *Bmc Bioinformatics* 2011, **12:**323.
45. Anders S: **HTSeq: Analysing high-throughput sequencing data with Python.** *URL http://www-huber embl de/users/anders/HTSeq/doc/overview html* 2010.
46. Bohnert R, Ratsch G: **rQuant.web: a tool for RNA-Seq-based transcript quantitation.** *Nucleic Acids Research* 2010, **38:**W348-351.
47. Nicolae M, Mangul S, Mandoiu, II, Zelikovsky A: **Estimation of alternative splicing isoform frequencies from RNA-Seq data.** *Algorithms Mol Biol* 2011, **6:**9.

48. Szulwach KE, Li XK, Li YJ, Song CX, Han JW, Kim S, Namburi S, Hermetz K, Kim JJ, Rudd MK, Yoon YS, Ren B, He C, Jin P: **Integrating 5-Hydroxymethylcytosine into the Epigenomic Landscape of Human Embryonic Stem Cells.** *Plos Genetics* 2011, **7**.
49. Crick F: **Central dogma of molecular biology.** *Nature* 1970, **227:**561-563.
50. Chargaff E, Zamenhof S, Green C: **Composition of human desoxypentose nucleic acid.** *Nature* 1950, **165:**756-757.
51. Meselson M, Stahl FW: **The replication of DNA in Escherichia coli.** *Proceedings of the National Academy of Sciences* 1958, **44:**671-682.
52. **DNA replication en by LadyofHats** [http://commons.wikimedia.org/wiki/File:DNA_replication_en.svg#mediaviewer/File:DNA_replication_en.svg]
53. Holley RW, Apgar J, Everett GA, Madison JT, Marquisee M, Merrill SH, Penswick JR, Zamir A: **Structure of a Ribonucleic Acid.** *Science* 1965, **147:**1462-1465.
54. Kowalczyk J, Domal-Kwiatkowska D, Mazurek U, Zembala M, Michalski B, Zembala M: **Post-transcriptional modifications of VEGF-A mRNA in non-ischemic dilated cardiomyopathy.** *Cellular & Molecular Biology Letters* 2007, **12:**331-347.
55. Darnell JE, Jr.: **Implications of RNA-RNA splicing in evolution of eukaryotic cells.** *Science* 1978, **202:**1257-1260.
56. Burset M, Seledtsov IA, Solovyev VV: **Analysis of canonical and non-canonical splice sites in mammalian genomes.** *Nucleic Acids Research* 2000, **28:**4364-4375.
57. Early P: **Two mRNAs can be produced from a single immunoglobulin μ gene by alternative RNA processing pathways.** *Cell* 1980, **20:**313-319.
58. Pan Q, Shai O, Lee LJ, Frey BJ, Blencowe BJ: **Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing.** *Nat Genet* 2008, **40:**1413-1415.
59. Williams R, Peisajovich SG, Miller OJ, Magdassi S, Tawfik DS, Griffiths AD: **Amplification of complex gene libraries by emulsion PCR.** *Nature Methods* 2006, **3:**545-550.
60. Schuster SC: **Next-generation sequencing transforms today's biology.** *Nature Methods* 2008, **5:**16-18.
61. Ronaghi M, Karamohamed S, Pettersson B, Uhlen M, Nyren P: **Real-time DNA sequencing using detection of pyrophosphate release.** *Anal Biochem* 1996, **242:**84-89.
62. Pennisi E: **Genomics. Semiconductors inspire new sequencing technologies.** *Science* 2010, **327:**1190.
63. Purushothaman S, Toumazou C, Ou CP: **Protons and single nucleotide polymorphism detection: A simple use for the ion sensitive field effect transistor.** *Sensors and Actuators B-Chemical* 2006, **114:**964-968.
64. Metzker ML: **Emerging technologies in DNA sequencing.** *Genome Res* 2005, **15:**1767-1776.
65. Sambrook J, Russell DW: **Fragmentation of DNA by sonication.** *CSH Protoc* 2006, **2006**.
66. Sambrook J, Russell DW: **Fragmentation of DNA by nebulization.** *CSH Protoc* 2006, **2006**.
67. Adessi C, Matton G, Ayala G, Turcatti G, Mermod JJ, Mayer P, Kawashima E: **Solid phase DNA amplification: characterisation of primer attachment and amplification mechanisms.** *Nucleic Acids Research* 2000, **28:**E87.

68. Mardis ER: **Next-generation DNA sequencing methods.** *Annu Rev Genomics Hum Genet* 2008, **9:**387-402.
69. Huang YF, Chen SC, Chiang YS, Chen TH, Chiu KP: **Palindromic sequence impedes sequencing-by-ligation mechanism.** *BMC Syst Biol* 2012, **6 Suppl 2:**S10.
70. Quail MA, Smith M, Coupland P, Otto TD, Harris SR, Connor TR, Bertoni A, Swerdlow HP, Gu Y: **A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers.** *Bmc Genomics* 2012, **13:**341.
71. Nagarajan N, Bertrand D, Hillmer AM, Zang ZJ, Yao F, Jacques PE, Teo AS, Cutcutache I, Zhang Z, Lee WH, Sia YY, Gao S, Ariyaratne PN, Ho A, Woo XY, Veeravali L, Ong CK, Deng N, Desai KV, Khor CC, Hibberd ML, Shahab A, Rao J, Wu M, Teh M, Zhu F, Chin SY, Pang B, So JB, Bourque G, et al: **Whole-genome reconstruction and mutational signatures in gastric cancer.** *Genome Biol* 2012, **13:**R115.
72. Adams M, Kelley J, Gocayne J, Dubnick M, Polymeropoulos M, Xiao H, Merril C, Wu A, Olde B, Moreno R, et a: **Complementary DNA sequencing: expressed sequence tags and human genome project.** *Science* 1991, **252:**1651-1656.
73. Bernstein BE, Kamal M, Lindblad-Toh K, Bekiranov S, Bailey DK, Huebert DJ, McMahon S, Karlsson EK, Kulbokas EJ, 3rd, Gingeras TR, Schreiber SL, Lander ES: **Genomic maps and comparative analysis of histone modifications in human and mouse.** *Cell* 2005, **120:**169-181.
74. Bird AP: **CpG-rich islands and the function of DNA methylation.** *Nature* 1986, **321:**209-213.
75. Meissner A, Mikkelsen TS, Gu H, Wernig M, Hanna J, Sivachenko A, Zhang X, Bernstein BE, Nusbaum C, Jaffe DB, Gnirke A, Jaenisch R, Lander ES: **Genome-scale DNA methylation maps of pluripotent and differentiated cells.** *Nature* 2008, **454:**766-770.
76. Frommer M, Mcdonald LE, Millar DS, Collis CM, Watt F, Grigg GW, Molloy PL, Paul CL: **A Genomic Sequencing Protocol That Yields a Positive Display of 5-Methylcytosine Residues in Individual DNA Strands.** *Proceedings of the National Academy of Sciences of the United States of America* 1992, **89:**1827-1831.
77. Ferragina P, Manzini G: **Opportunistic data structures with applications.** *41st Annual Symposium on Foundations of Computer Science, Proceedings* 2000**:**390-398.
78. Burrows M, Wheeler D: **A block-sorting lossless data compression algorithm.** 1994.
79. Smith TF, Waterman MS: **Identification of common molecular subsequences.** *J Mol Biol* 1981, **147:**195-197.
80. Needleman SB, Wunsch CD: **A general method applicable to the search for similarities in the amino acid sequence of two proteins.** *J Mol Biol* 1970, **48:**443-453.
81. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R: **The Sequence Alignment/Map format and SAMtools.** *Bioinformatics* 2009, **25:**2078-2079.
82. Vingron M, Waterman MS: **Sequence alignment and penalty choice. Review of concepts, case studies and implications.** *J Mol Biol* 1994, **235:**1-12.
83. Ewing B, Hillier L, Wendl MC, Green P: **Base-calling of automated sequencer traces using phred. I. Accuracy assessment.** *Genome Res* 1998, **8:**175-185.

84. Li H, Ruan J, Durbin R: **Mapping short DNA sequencing reads and calling variants using mapping quality scores.** *Genome Res* 2008, **18:**1851-1858.
85. Pelizzola M, Ecker JR: **The DNA methylome.** *FEBS Lett* 2011, **585:**1994-2000.
86. Lim JQ, Tennakoon C, Li G, Wong E, Ruan Y, Wei CL, Sung WK: **BatMeth: improved mapper for bisulfite sequencing reads on DNA methylation.** *Genome Biol* 2012, **13:**R82.
87. Ondov BD, Cochran C, Landers M, Meredith GD, Dudas M, Bergman NH: **An alignment algorithm for bisulfite sequencing using the Applied Biosystems SOLiD System.** *Bioinformatics* 2010, **26:**1901-1902.
88. Krueger F, Andrews SR: **Bismark: a flexible aligner and methylation caller for Bisulfite-Seq applications.** *Bioinformatics* 2011, **27:**1571-1572.
89. Harris EY, Ponts N, Levchuk A, Roch KL, Lonardi S: **BRAT: bisulfite-treated reads analysis tool.** *Bioinformatics* 2010, **26:**572-573.
90. Harris EY, Ponts N, Le Roch KG, Lonardi S: **BRAT-BW: efficient and accurate mapping of bisulfite-treated reads.** *Bioinformatics* 2012, **28:**1795-1796.
91. Chen PY, Cokus SJ, Pellegrini M: **BS Seeker: precise mapping for bisulfite sequencing.** *BMC Bioinformatics* 2010, **11:**203.
92. Lee TF, Zhai J, Meyers BC: **Conservation and divergence in eukaryotic DNA methylation.** *Proc Natl Acad Sci U S A* 2010, **107:**9027-9028.
93. Smith AD, Chung WY, Hodges E, Kendall J, Hannon G, Hicks J, Xuan Z, Zhang MQ: **Updates to the RMAP short-read mapping software.** *Bioinformatics* 2009, **25:**2841-2842.
94. Campagna D, Telatin A, Forcato C, Vitulo N, Valle G: **PASS-bis: a bisulfite aligner suitable for whole methylome analysis of Illumina and SOLiD reads.** *Bioinformatics* 2013, **29:**268-270.
95. Xi Y, Li W: **BSMAP: whole genome bisulfite sequence MAPping program.** *BMC Bioinformatics* 2009, **10:**232.
96. Kreck B, Marnellos G, Richter J, Krueger F, Siebert R, Franke A: **B-SOLANA: An approach for the analysis of two-base encoding bisulfite sequencing data (In Press).** *Bioinformatics* 2011.
97. Kondrashov AS, Rogozin IB: **Context of deletions and insertions in human coding sequences.** *Hum Mutat* 2004, **23:**177-185.
98. Ma L, Zhang TT, Huang ZR, Jiang XQ, Tao SH: **Patterns of nucleotides that flank substitutions in human orthologous genes.** *Bmc Genomics* 2010, **11**.
99. Sung WK, Zheng H, Li S, Chen R, Liu X, Li Y, Lee NP, Lee WH, Ariyaratne PN, Tennakoon C, Mulawadi FH, Wong KF, Liu AM, Poon RT, Fan ST, Chan KL, Gong Z, Hu Y, Lin Z, Wang G, Zhang Q, Barber TD, Chou WC, Aggarwal A, Hao K, Zhou W, Zhang C, Hardwick J, Buser C, Xu J, et al: **Genome-wide survey of recurrent HBV integration in hepatocellular carcinoma.** *Nat Genet* 2012, **44:**765-769.
100. Li H, Durbin R: **Fast and accurate short read alignment with Burrows-Wheeler transform.** *Bioinformatics* 2009, **25:**1754-1760.
101. Li H, Durbin R: **Fast and accurate long-read alignment with Burrows-Wheeler transform.** *Bioinformatics* 2010, **26:**589-595.
102. Langmead B, Trapnell C, Pop M, Salzberg SL: **Ultrafast and memory-efficient alignment of short DNA sequences to the human genome.** *Genome Biol* 2009, **10:**R25.
103. Langmead B, Salzberg SL: **Fast gapped-read alignment with Bowtie 2.** *Nature Methods* 2012, **9:**357-359.

104. Li R, Li Y, Kristiansen K, Wang J: **SOAP: short oligonucleotide alignment program.** *Bioinformatics* 2008, **24:**713-714.
105. Li R, Yu C, Li Y, Lam TW, Yiu SM, Kristiansen K, Wang J: **SOAP2: an improved ultrafast tool for short read alignment.** *Bioinformatics* 2009, **25:**1966-1967.
106. Novocraft: **Novoalign.** www.novocraft.com.
107. Lunter G, Goodson M: **Stampy: a statistical algorithm for sensitive and fast mapping of Illumina sequence reads.** *Genome Res* 2011, **21:**936-939.
108. Campagna D, Albiero A, Bilardi A, Caniato E, Forcato C, Manavski S, Vitulo N, Valle G: **PASS: a program to align short sequences.** *Bioinformatics* 2009, **25:**967-968.
109. Liu Y, Schmidt B, Maskell DL: **CUSHAW: a CUDA compatible short read aligner to large genomes based on the Burrows-Wheeler transform.** *Bioinformatics* 2012, **28:**1830-1837.
110. Liu Y, Schmidt B: **Long read alignment based on maximal exact match seeds.** *Bioinformatics* 2012, **28:**i318-i324.
111. Gontarz PM, Berger J, Wong CF: **SRmapper: a fast and sensitive genome-hashing alignment tool.** *Bioinformatics* 2013, **29:**316-321.
112. Mu JC, Jiang H, Kiani A, Mohiyuddin M, Bani Asadi N, Wong WH: **Fast and accurate read alignment for resequencing.** *Bioinformatics* 2012, **28:**2366-2373.
113. Cox A: **ELAND: Efficient Local Alignment of Nucleotide Data.** 2006.
114. Smith AD, Xuan ZY, Zhang MQ: **Using quality scores and longer reads improves accuracy of Solexa read mapping.** *Bmc Bioinformatics* 2008, **9**.
115. Rumble SM, Lacroute P, Dalca AV, Fiume M, Sidow A, Brudno M: **SHRiMP: accurate mapping of short color-space reads.** *PLoS Comput Biol* 2009, **5:**e1000386.
116. David M, Dzamba M, Lister D, Ilie L, Brudno M: **SHRiMP2: sensitive yet practical SHort Read Mapping.** *Bioinformatics* 2011, **27:**1011-1012.
117. Lin H, Zhang ZF, Zhang MQ, Ma B, Li M: **ZOOM! Zillions of oligos mapped.** *Bioinformatics* 2008, **24:**2431-2437.
118. Malhis N, Butterfield YS, Ester M, Jones SJ: **Slider--maximum use of probability information for alignment of short sequence reads and SNP detection.** *Bioinformatics* 2009, **25:**6-13.
119. Malhis N, Jones SJ: **High quality SNP calling using Illumina data at shallow coverage.** *Bioinformatics* 2010, **26:**1029-1035.
120. Farrar M: **Striped Smith-Waterman speeds database searches six times over other SIMD implementations.** *Bioinformatics* 2007, **23:**156-161.
121. Zhao M, Lee WP, Marth GT: **SSW Library: An SIMD Smith-Waterman C/C++ Library for Use in Genomic Applications.** *arXiv preprint arXiv:12086350* 2012.
122. Zhang Z, Schwartz S, Wagner L, Miller W: **A greedy algorithm for aligning DNA sequences.** *J Comput Biol* 2000, **7:**203-214.
123. Faust GG, Hall IM: **YAHA: fast and flexible long-read alignment with optimal breakpoint detection.** *Bioinformatics* 2012, **28:**2417-2424.
124. Baeza-Yates RA, Perleberg CH: **Fast and practical approximate string matching.** In *Combinatorial Pattern Matching*. Springer; 1992: 185-192.
125. Ma B, Tromp J, Li M: **PatternHunter: faster and more sensitive homology search.** *Bioinformatics* 2002, **18:**440-445.
126. Li H, Homer N: **A survey of sequence alignment algorithms for next-generation sequencing.** *Brief Bioinform* 2010, **11:**473-483.

127. Nakamura K, Oshima T, Morimoto T, Ikeda S, Yoshikawa H, Shiwa Y, Ishikawa S, Linak MC, Hirai A, Takahashi H, Altaf-Ul-Amin M, Ogasawara N, Kanaya S: **Sequence-specific error profile of Illumina sequencers.** *Nucleic Acids Research* 2011, **39:**e90.
128. Burkhardt S, Karkkainen J: **Better filtering with gapped q-grams.** *Fundamenta Informaticae* 2003, **56:**51-70.
129. Jokinen P, Ukkonen E: **Two algorithms for approxmate string matching in static texts.** In *Mathematical Foundations of Computer Science 1991.* Springer; 1991: 240-248
130. Weese D, Emde AK, Rausch T, Doring A, Reinert K: **RazerS--fast read mapping with sensitivity control.** *Genome Res* 2009, **19:**1646-1654.
131. Weese D, Holtgrewe M, Reinert K: **RazerS 3: faster, fully sensitive read mapping.** *Bioinformatics* 2012, **28:**2592-2599.
132. Siragusa E, Weese D, Reinert K: **Fast and accurate read mapping with approximate seeds and multiple backtracking.** *Nucleic Acids Research* 2013, **41:**e78.
133. Manber U, Myers G: **Suffix Arrays: A New Method for On-Line String Searches.** *SIAM Journal on Computing* 1993, **22:**935-948.
134. Weiner P: **Linear pattern matching algorithms.** 1973**:**1-11.
135. Delcher AL, Kasif S, Fleischmann RD, Peterson J, White O, Salzberg SL: **Alignment of whole genomes.** *Nucleic Acids Research* 1999, **27:**2369-2376.
136. Meek C, Patel JM, Kasetty S: **OASIS: an online and accurate technique for local-alignment searches on biological sequences.** In *Proceedings of the 29th international conference on Very large data bases - Volume 29*. pp. 910-921. Berlin, Germany: VLDB Endowment; 2003:910-921.
137. Farach M: **Optimal suffix tree construction with large alphabets.** *38th Annual Symposium on Foundations of Computer Science, Proceedings* 1997**:**137-143.
138. Abouelhoda MI, Kurtz S, Ohlebusch E: **Replacing suffix trees with enhanced suffix arrays.** *Journal of Discrete Algorithms* 2004, **2:**53-86.
139. Hoffmann S, Otto C, Kurtz S, Sharma CM, Khaitovich P, Vogel J, Stadler PF, Hackermuller J: **Fast Mapping of Short Sequences with Mismatches, Insertions and Deletions Using Index Structures.** *PLoS Comput Biol* 2009, **5**.
140. Marco-Sola S, Sammeth M, Guigo R, Ribeca P: **The GEM mapper: fast, accurate and versatile alignment by filtration.** *Nature Methods* 2012, **9:**1185-1188.
141. Lam TW, Sung WK, Tam SL, Wong CK, Yiu SM: **Compressed indexing and local alignment of DNA.** *Bioinformatics* 2008, **24:**791-797.
142. Luebke D, Harris M, Govindaraju N, Lefohn A, Houston M, Owens J, Segal M, Papakipos M, Buck I: **GPGPU: general-purpose computation on graphics hardware.** 2006**:**208.
143. Liu CM, Wong T, Wu E, Luo R, Yiu SM, Li Y, Wang B, Yu C, Chu X, Zhao K, Li R, Lam TW: **SOAP3: ultra-fast GPU-based parallel alignment tool for short reads.** *Bioinformatics* 2012, **28:**878-879.
144. Homer N, Merriman B, Nelson SF: **BFAST: an alignment tool for large scale genome resequencing.** *PLoS One* 2009, **4:**e7767.
145. Chaisson MJ, Tesler G: **Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory.** *Bmc Bioinformatics* 2012, **13:**238.
146. Schatz MC: **CloudBurst: highly sensitive read mapping with MapReduce.** *Bioinformatics* 2009, **25:**1363-1369.

147. Liu Y, Schmidt B: **CUSHAW2-GPU: empowering faster gapped short-read alignment using GPU computing.** *IEEE Design & Test* 2013**:**1-1.
148. Clement NL, Snell Q, Clement MJ, Hollenhorst PC, Purwar J, Graves BJ, Cairns BR, Johnson WE: **The GNUMAP algorithm: unbiased probabilistic mapping of oligonucleotides from next-generation sequencing.** *Bioinformatics* 2010, **26:**38-45.
149. Ahmadi A, Behm A, Honnalli N, Li C, Weng L, Xie X: **Hobbes: optimized gram-based methods for efficient read alignment.** *Nucleic Acids Research* 2012, **40:**e41.
150. Eaves HL, Gao Y: **MOM: maximum oligonucleotide mapping.** *Bioinformatics* 2009, **25:**969-970.
151. Lee W-P, Stromberg M, Ward A, Stewart C, Garrison E, Marth GT: **MOSAIK: A hash-based algorithm for accurate next-generation sequencing read mapping.** 2013.
152. Alkan C, Kidd JM, Marques-Bonet T, Aksay G, Antonacci F, Hormozdiari F, Kitzman JO, Baker C, Malig M, Mutlu O, Sahinalp SC, Gibbs RA, Eichler EE: **Personalized copy number and segmental duplication maps using next-generation sequencing.** *Nat Genet* 2009, **41:**1061-1067.
153. Hach F, Hormozdiari F, Alkan C, Birol I, Eichler EE, Sahinalp SC: **mrsFAST: a cache-oblivious algorithm for short-read mapping.** *Nature Methods* 2010, **7:**576-577.
154. Hormozdiari F, Hach F, Sahinalp SC, Eichler EE, Alkan C: **Sensitive and fast mapping of di-base encoded reads.** *Bioinformatics* 2011, **27:**1915-1921.
155. Chen Y, Souaiaia T, Chen T: **PerM: efficient mapping of short sequencing reads with periodic full sensitive spaced seeds.** *Bioinformatics* 2009, **25:**2514-2521.
156. Kim YJ, Teletia N, Ruotti V, Maher CA, Chinnaiyan AM, Stewart R, Thomson JA, Patel JM: **ProbeMatch: rapid alignment of oligonucleotides to genome allowing both gaps and mismatches.** *Bioinformatics* 2009, **25:**1424-1425.
157. Frousios K, Iliopoulos CS, Mouchard L, Pissis SP, Tischler G: **REAL: an efficient REad ALigner for next generation sequencing reads.** 2010**:**154.
158. Jiang H, Wong WH: **SeqMap: mapping massive amount of oligonucleotides to the genome.** *Bioinformatics* 2008, **24:**2395-2396.
159. Ning Z, Cox AJ, Mullikin JC: **SSAHA: a fast search method for large DNA databases.** *Genome Res* 2001, **11:**1725-1729.
160. Liao Y, Smyth GK, Shi W: **The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote.** *Nucleic Acids Research* 2013, **41:**e108.
161. Sharp PA: **The discovery of split genes and RNA splicing.** *Trends Biochem Sci* 2005, **30:**279-281.
162. Breitbart RE, Andreadis A, Nadal-Ginard B: **Alternative splicing: a ubiquitous mechanism for the generation of multiple protein isoforms from single genes.** *Annu Rev Biochem* 1987, **56:**467-495.
163. Goedert M, Spillantini MG, Jakes R, Rutherford D, Crowther RA: **Multiple isoforms of human microtubule-associated protein tau: sequences and localization in neurofibrillary tangles of Alzheimer's disease.** *Neuron* 1989, **3:**519-526.
164. Licatalosi DD, Darnell RB: **Splicing regulation in neurologic disease.** *Neuron* 2006, **52:**93-101.
165. Kapranov P, Cawley SE, Drenkow J, Bekiranov S, Strausberg RL, Fodor SP, Gingeras TR: **Large-scale transcriptional activity in chromosomes 21 and 22.** *Science* 2002, **296:**916-919.

166. Pan Q, Shai O, Misquitta C, Zhang W, Saltzman AL, Mohammad N, Babak T, Siu H, Hughes TR, Morris QD, Frey BJ, Blencowe BJ: **Revealing global regulatory features of mammalian alternative splicing using a quantitative microarray platform.** *Mol Cell* 2004, **16:**929-941.
167. Kwan T, Benovoy D, Dias C, Gurd S, Provencher C, Beaulieu P, Hudson TJ, Sladek R, Majewski J: **Genome-wide analysis of transcript isoform variation in humans.** *Nat Genet* 2008, **40:**225-231.
168. Slater GS, Birney E: **Automated generation of heuristics for biological sequence comparison.** *Bmc Bioinformatics* 2005, **6:**31.
169. Vanin EF: **Processed pseudogenes: characteristics and evolution.** *Annu Rev Genet* 1985, **19:**253-272.
170. Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B: **Mapping and quantifying mammalian transcriptomes by RNA-Seq.** *Nature Methods* 2008, **5:**621-628.
171. Wang ET, Sandberg R, Luo S, Khrebtukova I, Zhang L, Mayr C, Kingsmore SF, Schroth GP, Burge CB: **Alternative isoform regulation in human tissue transcriptomes.** *Nature* 2008, **456:**470-476.
172. Nagalakshmi U, Waern K, Snyder M: **RNA-Seq: a method for comprehensive transcriptome analysis.** *Curr Protoc Mol Biol* 2010, **Chapter 4:**Unit 4 11 11-13.
173. Cloonan N, Xu Q, Faulkner GJ, Taylor DF, Tang DT, Kolle G, Grimmond SM: **RNA-MATE: a recursive mapping strategy for high-throughput RNA-sequencing data.** *Bioinformatics* 2009, **25:**2615-2616.
174. Wood DL, Xu Q, Pearson JV, Cloonan N, Grimmond SM: **X-MATE: a flexible system for mapping short read data.** *Bioinformatics* 2011, **27:**580-581.
175. Pruitt KD, Tatusova T, Maglott DR: **NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins.** *Nucleic Acids Research* 2005, **33:**D501-504.
176. Pruitt KD, Tatusova T, Brown GR, Maglott DR: **NCBI Reference Sequences (RefSeq): current status, new features and genome annotation policy.** *Nucleic Acids Research* 2012, **40:**D130-135.
177. Grant GR, Farkas MH, Pizarro AD, Lahens NF, Schug J, Brunk BP, Stoeckert CJ, Hogenesch JB, Pierce EA: **Comparative analysis of RNA-Seq alignment algorithms and the RNA-Seq unified mapper (RUM).** *Bioinformatics* 2011, **27:**2518-2528.
178. Denoeud F, Aury JM, Da Silva C, Noel B, Rogier O, Delledonne M, Morgante M, Valle G, Wincker P, Scarpelli C, Jaillon O, Artiguenave F: **Annotating genomes with massive-scale RNA sequencing.** *Genome Biol* 2008, **9:**R175.
179. Huang S, Zhang J, Li R, Zhang W, He Z, Lam TW, Peng Z, Yiu SM: **SOAPsplice: Genome-Wide ab initio Detection of Splice Junctions from RNA-Seq Data.** *Front Genet* 2011, **2:**46.
180. Wu TD, Nacu S: **Fast and SNP-tolerant detection of complex variants and splicing in short reads.** *Bioinformatics* 2010, **26:**873-881.
181. Bryant DW, Jr., Shen R, Priest HD, Wong WK, Mockler TC: **Supersplat--spliced RNA-seq alignment.** *Bioinformatics* 2010, **26:**1500-1505.
182. Philippe N, Salson M, Commes T, Rivals E: **CRAC: an integrated approach to the analysis of RNA-seq reads.** *Genome Biol* 2013, **14:**R30.
183. Cortes C, Vapnik V: **Support-vector networks.** *Machine Learning* 1995, **20:**273-297.
184. Jean G, Kahles A, Sreedharan VT, De Bona F, Ratsch G: **RNA-Seq read alignments with PALMapper.** *Curr Protoc Bioinformatics* 2010, **Chapter 11:**Unit 11 16.

185. Schneeberger K, Hagmann J, Ossowski S, Warthmann N, Gesing S, Kohlbacher O, Weigel D: **Simultaneous alignment of short reads against multiple genomes.** *Genome Biol* 2009, **10:**R98.
186. Dimon MT, Sorber K, DeRisi JL: **HMMSplicer: a tool for efficient and sensitive discovery of known and novel splice junctions in RNA-Seq data.** *PLoS One* 2010, **5:**e13875.
187. Iwasaki R, Kiuchi H, Ihara M, Mori T, Kawakami M, Ueda H: **Trans-splicing as a novel method to rapidly produce antibody fusion proteins.** *Biochem Biophys Res Commun* 2009, **384:**316-321.
188. Lou SK, Ni B, Lo LY, Tsui SK, Chan TF, Leung KS: **ABMapper: a suffix array-based tool for multi-location searching and splice-junction mapping.** *Bioinformatics* 2011, **27:**421-422.
189. Yeo G, Burge CB: **Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals.** *J Comput Biol* 2004, **11:**377-394.
190. Bao H, Xiong Y, Guo H, Zhou R, Lu X, Yang Z, Zhong Y, Shi S: **MapNext: a software tool for spliced and unspliced alignments and SNP detection of short sequence reads.** *Bmc Genomics* 2009, **10 Suppl 3:**S13.
191. Hu J, Ge H, Newman M, Liu K: **OSA: a fast and accurate alignment tool for RNA-Seq.** *Bioinformatics* 2012, **28:**1933-1934.
192. Tang S, Riva A: **PASTA: splice junction identification from RNA-sequencing data.** *Bmc Bioinformatics* 2013, **14:**116.
193. Chen LY, Wei KC, Huang AC, Wang K, Huang CY, Yi D, Tang CY, Galas DJ, Hood LE: **RNASEQR--a streamlined and accurate RNA-seq sequence analysis program.** *Nucleic Acids Research* 2012, **40:**e42.
194. Wang L, Wang X, Liang Y, Zhang X: **Observations on novel splice junctions from RNA sequencing data.** *Biochem Biophys Res Commun* 2011, **409:**299-303.
195. Ameur A, Wetterbom A, Feuk L, Gyllensten U: **Global and unbiased detection of splice junctions from RNA-seq data.** *Genome Biol* 2010, **11:**R34.
196. Li Y, Li-Byarlay H, Burns P, Borodovsky M, Robinson GE, Ma J: **TrueSight: a new algorithm for splice junction detection using RNA-seq.** *Nucleic Acids Research* 2013, **41:**e51.
197. Law JA, Jacobsen SE: **Establishing, maintaining and modifying DNA methylation patterns in plants and animals.** *Nat Rev Genet* 2010, **11:**204-220.
198. Keshet I, Lieman-Hurwitz J, Cedar H: **DNA methylation affects the formation of active chromatin.** *Cell* 1986, **44:**535-543.
199. Reik W, Dean W, Walter J: **Epigenetic reprogramming in mammalian development.** *Science* 2001, **293:**1089-1093.
200. Li E, Beard C, Jaenisch R: **Role for DNA methylation in genomic imprinting.** *Nature* 1993, **366:**362-365.
201. Heard E, Clerc P, Avner P: **X-chromosome inactivation in mammals.** *Annu Rev Genet* 1997, **31:**571-610.
202. Walsh CP, Chaillet JR, Bestor TH: **Transcription of IAP endogenous retroviruses is constrained by cytosine methylation.** *Nat Genet* 1998, **20:**116-117.
203. Gopalakrishnan S, Van Emburgh BO, Robertson KD: **DNA methylation in development and human disease.** *Mutat Res* 2008, **647:**30-38.
204. Hultén MA, Papageorgiou EA, Ragione FD, D'Esposito M, Carter N, Patsalis PC: **Non-invasive prenatal diagnosis: An epigenetic approach to the detection of common fetal chromosome disorders by analysis of maternal blood samples** In *Circulating Nucleic Acids in Plasma and Serum.* Edited by Gahan PB; 2011: 133-142

205. Lister R, O'Malley RC, Tonti-Filippini J, Gregory BD, Berry CC, Millar AH, Ecker JR: **Highly integrated single-base resolution maps of the epigenome in Arabidopsis.** *Cell* 2008, **133:**523-536.
206. Cokus SJ, Feng S, Zhang X, Chen Z, Merriman B, Haudenschild CD, Pradhan S, Nelson SF, Pellegrini M, Jacobsen SE: **Shotgun bisulphite sequencing of the Arabidopsis genome reveals DNA methylation patterning.** *Nature* 2008, **452:**215-219.
207. Chung CAB, Boyd VL, McKernan KJ, Fu Y, Monighetti C, Peckham HE, Barker M: **Whole methylome analysis by ultra-deep sequencing using two-base encoding.** *PLoS ONE* 2010, **5:**e9320.
208. Hansen KD, Timp W, Bravo HC, Sabunciyan S, Langmead B, McDonald OG, Wen B, Wu H, Liu Y, Diep D, Briem E, Zhang K, Irizarry RA, Feinberg AP: **Increased methylation variation in epigenetic domains across cancer types.** *Nat Genet* 2011, **43:**768-775.
209. Frommer M, McDonald LE, Millar DS, Collis CM, Watt F, Grigg GW, Molloy PL, Paul CL: **A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands.** *Proc Natl Acad Sci U S A* 1992, **89:**1827-1831.
210. Pedersen B, Hsieh TF, Ibarra C, Fischer RL: **MethylCoder: software pipeline for bisulfite-treated sequences.** *Bioinformatics* 2011, **27:**2435-2436.
211. Shendure J, Ji H: **Next-generation DNA sequencing.** *Nat Biotechnol* 2008, **26:**1135-1145.
212. Homer N, Merriman B, Nelson SF: **Local alignment of two-base encoded DNA sequence.** *BMC Bioinformatics* 2009, **10:**175.
213. Krueger F, Kreck B, Franke A, Andrews SR: **DNA methylome analysis using short bisulfite sequencing data.** *Nature Methods* 2012, **9:**145-151.
214. Ondov BD, Varadarajan A, Passalacqua KD, Bergman NH: **Efficient mapping of Applied Biosystems SOLiD sequence data to a reference genome for functional genomic applications.** *Bioinformatics* 2008, **24:**2776-2777.
215. Karp RM, Rabin MO: **Efficient randomized pattern-matching algorithms.** *IBM Journal of Research and Development* 1987, **31:**249–260.
216. Smith AD, Xuan Z, Zhang MQ: **Using quality scores and longer reads improves accuracy of Solexa read mapping.** *BMC Bioinformatics* 2008, **9:**128.
217. **Sherman** [http://www.bioinformatics.bbsrc.ac.uk/projects/sherman/]
218. Laurent L, Wong E, Li G, Huynh T, Tsirigos A, Ong CT, Low HM, Kin Sung KW, Rigoutsos I, Loring J, Wei CL: **Dynamic changes in the human methylome during differentiation.** *Genome Res* 2010, **20:**320-331.
219. Tennakoon C, Purbojati RW, Sung WK: **BatMis: A fast algorithm for k-mismatch mapping.** *Bioinformatics* 2012.
220. Bird A, Taggart M, Frommer M, Miller OJ, Macleod D: **A fraction of the mouse genome that is derived from islands of nonmethylated, CpG-rich DNA.** *Cell* 1985, **40:**91-99.
221. Lister R, Pelizzola M, Dowen RH, Hawkins RD, Hon G, Tonti-Filippini J, Nery JR, Lee L, Ye Z, Ngo QM, Edsall L, Antosiewicz-Bourget J, Stewart R, Ruotti V, Millar AH, Thomson JA, Ren B, Ecker JR: **Human DNA methylomes at base resolution show widespread epigenomic differences.** *Nature* 2009, **462:**315-322.
222. Stadler MB, Murr R, Burger L, Ivanek R, Lienert F, Scholer A, Wirbelauer C, Oakeley EJ, Gaidatzis D, Tiwari VK, Schubeler D: **DNA-binding factors shape the mouse methylome at distal regulatory regions.** *Nature* 2011, **480:**490-495.

223. Hellman A, Chess A: **Gene body-specific methylation on the active X chromosome.** *Science* 2007, **315:**1141-1143.
224. Paulsen M, Ferguson-Smith AC: **DNA methylation in genomic imprinting, development, and disease.** *J Pathol* 2001, **195:**97-110.
225. Swain JL, Stewart TA, Leder P: **Parental legacy determines methylation and expression of an autosomal transgene: a molecular mechanism for parental imprinting.** *Cell* 1987, **50:**719-727.
226. Ehrlich M: **DNA methylation in cancer: too much, but also too little.** *Oncogene* 2002, **21:**5400-5413.
227. Baylin SB, Esteller M, Rountree MR, Bachman KE, Schuebel K, Herman JG: **Aberrant patterns of DNA methylation, chromatin formation and gene expression in cancer.** *Hum Mol Genet* 2001, **10:**687-692.
228. Gao W, Kondo Y, Shen L, Shimizu Y, Sano T, Yamao K, Natsume A, Goto Y, Ito M, Murakami H, Osada H, Zhang J, Issa JP, Sekido Y: **Variable DNA methylation patterns associated with progression of disease in hepatocellular carcinomas.** *Carcinogenesis* 2008, **29:**1901-1910.
229. Hinoue T, Weisenberger DJ, Lange CP, Shen H, Byun HM, Van Den Berg D, Malik S, Pan F, Noushmehr H, van Dijk CM, Tollenaar RA, Laird PW: **Genome-scale analysis of aberrant DNA methylation in colorectal cancer.** *Genome Res* 2012, **22:**271-282.
230. Zheng S, Chen P, McMillan A, Lafuente A, Lafuente MJ, Ballesta A, Trias M, Wiencke JK: **Correlations of partial and extensive methylation at the p14(ARF) locus with reduced mRNA expression in colorectal cancer cell lines and clinicopathological features in primary tumors.** *Carcinogenesis* 2000, **21:**2057-2064.
231. Dittrich B, Robinson WP, Knoblauch H, Buiting K, Schmidt K, Gillessen-Kaesbach G, Horsthemke B: **Molecular diagnosis of the Prader-Willi and Angelman syndromes by detection of parent-of-origin specific DNA methylation in 15q11-13.** *Hum Genet* 1992, **90:**313-315.
232. Lalande M: **Parental imprinting and human disease.** *Annu Rev Genet* 1996, **30:**173-195.
233. Robertson KD: **DNA methylation and human disease.** *Nat Rev Genet* 2005, **6:**597-610.
234. Lister R, Pelizzola M, Kida YS, Hawkins RD, Nery JR, Hon G, Antosiewicz-Bourget J, O'Malley R, Castanon R, Klugman S, Downes M, Yu R, Stewart R, Ren B, Thomson JA, Evans RM, Ecker JR: **Hotspots of aberrant epigenomic reprogramming in human induced pluripotent stem cells.** *Nature* 2011, **471:**68-73.
235. Li Y, Zhu J, Tian G, Li N, Li Q, Ye M, Zheng H, Yu J, Wu H, Sun J, Zhang H, Chen Q, Luo R, Chen M, He Y, Jin X, Zhang Q, Yu C, Zhou G, Huang Y, Cao H, Zhou X, Guo S, Hu X, Li X, Kristiansen K, Bolund L, Xu J, Wang W, Yang H, et al: **The DNA methylome of human peripheral blood mononuclear cells.** *PLoS Biol* 2010, **8:**e1000533.
236. Hon GC, Hawkins RD, Caballero OL, Lo C, Lister R, Pelizzola M, Valsesia A, Ye Z, Kuan S, Edsall LE, Camargo AA, Stevenson BJ, Ecker JR, Bafna V, Strausberg RL, Simpson AJ, Ren B: **Global DNA hypomethylation coupled to repressive chromatin domain formation and gene silencing in breast cancer.** *Genome Res* 2012, **22:**246-258.
237. Hodges E, Molaro A, Dos Santos CO, Thekkat P, Song Q, Uren PJ, Park J, Butler J, Rafii S, McCombie WR, Smith AD, Hannon GJ: **Directional DNA**

methylation changes and complex intermediate states accompany lineage specificity in the adult hematopoietic compartment. *Mol Cell* 2011, **44:**17-28.
238. Molaro A, Hodges E, Fang F, Song Q, McCombie WR, Hannon GJ, Smith AD: **Sperm methylation profiles reveal features of epigenetic inheritance and evolution in primates.** *Cell* 2011, **146:**1029-1041.
239. Lim J-Q, Tennakoon C, Li G, Wong E, Ruan Y, Wei C-L, Sung W-K: **BatMeth: Improved Mapper for Bisulfite Sequencing Reads on DNA Methylation (accepted).** *Genome Biology* 2012.
240. **RepeatMasker Open-3.0, http://www.repeatmasker.org**
241. Luedi PP, Dietrich FS, Weidman JR, Bosko JM, Jirtle RL, Hartemink AJ: **Computational and experimental identification of novel human imprinted genes.** *Genome Res* 2007, **17:**1723-1730.
242. McLean CY, Bristor D, Hiller M, Clarke SL, Schaar BT, Lowe CB, Wenger AM, Bejerano G: **GREAT improves functional interpretation of cis-regulatory regions.** *Nat Biotechnol* 2010, **28:**495-501.
243. Tuskan RG, Tsang S, Sun Z, Baer J, Rozenblum E, Wu X, Munroe DJ, Reilly KM: **Real-time PCR analysis of candidate imprinted genes on mouse chromosome 11 shows balanced expression from the maternal and paternal chromosomes and strain-specific variation in expression levels.** *Epigenetics* 2008, **3:**43-50.
244. Maslov S, Ispolatov I: **Propagation of large concentration changes in reversible protein-binding networks.** *Proc Natl Acad Sci U S A* 2007, **104:**13655-13660.
245. Rada-Iglesias A, Bajpai R, Swigut T, Brugmann SA, Flynn RA, Wysocka J: **A unique chromatin signature uncovers early developmental enhancers in humans.** *Nature* 2010, **470:**279-283.
246. Bennett EA, Coleman LE, Tsui C, Pittard WS, Devine SE: **Natural genetic variation caused by transposable elements in humans.** *Genetics* 2004, **168:**933-951.
247. Mills RE, Luttig CT, Larkins CE, Beauchamp A, Tsui C, Pittard WS, Devine SE: **An initial map of insertion and deletion (INDEL) variation in the human genome.** *Genome Res* 2006, **16:**1182-1190.
248. Bhangale TR, Rieder MJ, Livingston RJ, Nickerson DA: **Comprehensive identification and characterization of diallelic insertion-deletion polymorphisms in 330 human candidate genes.** *Hum Mol Genet* 2005, **14:**59-69.
249. Yang H, Zhong Y, Peng C, Chen JQ, Tian D: **Important role of indels in somatic mutations of human cancer genes.** *BMC Med Genet* 2010, **11:**128.
250. Koboldt DC, Chen K, Wylie T, Larson DE, McLellan MD, Mardis ER, Weinstock GM, Wilson RK, Ding L: **VarScan: variant detection in massively parallel sequencing of individual and pooled samples.** *Bioinformatics* 2009, **25:**2283-2285.
251. Krawitz P, Rodelsperger C, Jager M, Jostins L, Bauer S, Robinson PN: **Microindel detection in short-read sequence data.** *Bioinformatics* 2010, **26:**722-729.
252. Albers CA, Lunter G, MacArthur DG, McVean G, Ouwehand WH, Durbin R: **Dindel: accurate indel calls from short-read data.** *Genome Res* 2011, **21:**961-973.
253. Rizk G, Lavenier D: **GASSST: global alignment short sequence search tool.** *Bioinformatics* 2010, **26:**2534-2540.

254. Durbin RM, Altshuler D, Abecasis GR, Bentley DR, Chakravarti A, Clark AG, Collins FS: **A map of human genome variation from population-scale sequencing.** *Nature* 2010, **467:**1061-1073.
255. Sherry ST, Ward M, Sirotkin K: **dbSNP-database for single nucleotide polymorphisms and other classes of minor genetic variation.** *Genome Res* 1999, **9:**677-679.
256. Zhang ZDD, Du J, Lam H, Abyzov A, Urban AE, Snyder M, Gerstein M: **Identification of genomic indels and structural variations using split reads.** *Bmc Genomics* 2011, **12**.
257. Karro JE, Yan Y, Zheng D, Zhang Z, Carriero N, Cayting P, Harrrison P, Gerstein M: **Pseudogene.org: a comprehensive database and comparison platform for pseudogene annotation.** *Nucleic Acids Research* 2007, **35:**D55-60.
258. Chen R, Mias GI, Li-Pook-Than J, Jiang L, Lam HY, Miriami E, Karczewski KJ, Hariharan M, Dewey FE, Cheng Y, Clark MJ, Im H, Habegger L, Balasubramanian S, O'Huallachain M, Dudley JT, Hillenmeyer S, Haraksingh R, Sharon D, Euskirchen G, Lacroute P, Bettinger K, Boyle AP, Kasowski M, Grubert F, Seki S, Garcia M, Whirl-Carrillo M, Gallardo M, Blasco MA, et al: **Personal omics profiling reveals dynamic molecular and medical phenotypes.** *Cell* 2012, **148:**1293-1307.
259. Darling AC, Mau B, Blattner FR, Perna NT: **Mauve: multiple alignment of conserved genomic sequence with rearrangements.** *Genome Res* 2004, **14:**1394-1403.
260. Gudlaugsdottir S, Boswell DR, Wood GR, Ma J: **Exon size distribution and the origin of introns.** *Genetica* 2007, **131:**299-306.
261. Saito Y, Tsuji J, Mituyama T: **Bisulfighter: accurate detection of methylated cytosines and differentially methylated regions.** *Nucleic Acids Research* 2014, **42:**e45.

Appendix A

A.1 Additional information on profiling methylation libraries

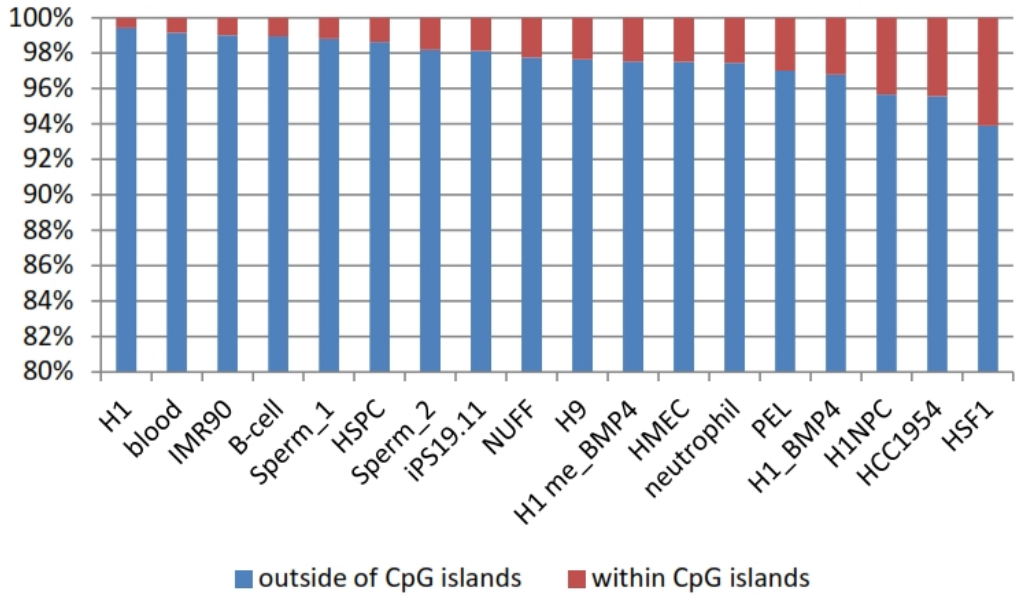


Figure A1.1. Percentage of partially methylated Cs with regards to CpG islands.

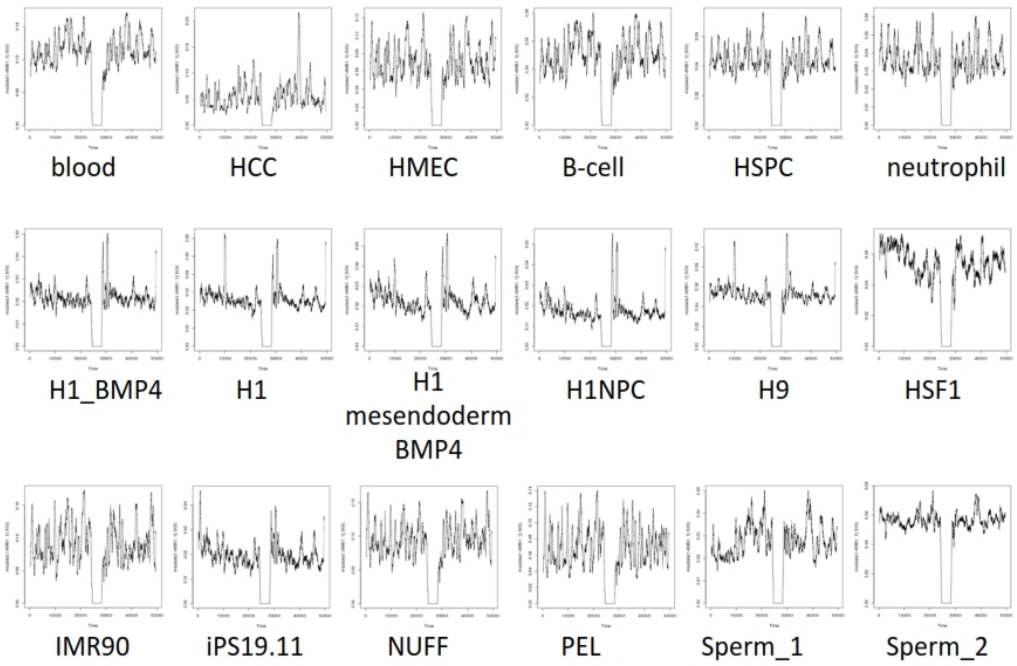


Figure A1.2. Genomic profile of partially methylated Cs along chromosome 1.

	H1	PEL	NUFF	IMR90	H1NPC	iPS19.11	H9	blood	HCC1954	HMEC	B cell	HSPC	neutrophil	H1_BMP4	H1_me_BMP4	HSF1	Sperm_1	Sperm_2	Chromosome length
chr1	0.083	0.076	0.078	0.072	0.085	0.085	0.082	0.078	0.075	0.074	0.074	0.074	0.074	0.084	0.085	0.084	0.075	0.078	0.081
chr2	0.074	0.074	0.078	0.078	0.069	0.072	0.071	0.082	0.083	0.075	0.078	0.074	0.070	0.073	0.071	0.075	0.077	0.078	0.079
chr3	0.058	0.054	0.057	0.061	0.053	0.055	0.052	0.064	0.060	0.055	0.061	0.058	0.053	0.056	0.054	0.054	0.062	0.060	0.064
chr4	0.052	0.052	0.055	0.059	0.046	0.048	0.045	0.062	0.067	0.054	0.061	0.057	0.052	0.049	0.047	0.051	0.065	0.058	0.062
chr5	0.053	0.053	0.056	0.058	0.050	0.052	0.048	0.060	0.055	0.054	0.059	0.055	0.053	0.052	0.051	0.050	0.060	0.056	0.058
chr6	0.056	0.048	0.051	0.053	0.051	0.053	0.048	0.057	0.057	0.050	0.054	0.053	0.049	0.054	0.052	0.050	0.059	0.056	0.055
chr7	0.056	0.054	0.057	0.055	0.056	0.056	0.053	0.056	0.061	0.055	0.053	0.053	0.053	0.055	0.055	0.058	0.055	0.057	0.051
chr8	0.047	0.051	0.051	0.052	0.046	0.046	0.045	0.052	0.048	0.050	0.051	0.048	0.046	0.047	0.046	0.047	0.056	0.052	0.047
chr9	0.041	0.039	0.040	0.038	0.042	0.042	0.041	0.041	0.037	0.038	0.039	0.039	0.040	0.041	0.041	0.043	0.038	0.041	0.046
chr10	0.047	0.051	0.051	0.049	0.049	0.046	0.046	0.050	0.046	0.049	0.049	0.047	0.046	0.049	0.048	0.047	0.050	0.050	0.044
chr11	0.048	0.047	0.049	0.045	0.051	0.050	0.047	0.048	0.045	0.047	0.047	0.047	0.048	0.050	0.051	0.050	0.049	0.048	0.044
chr12	0.048	0.044	0.045	0.044	0.047	0.048	0.044	0.046	0.044	0.043	0.045	0.044	0.043	0.047	0.047	0.044	0.048	0.047	0.043
chr13	0.028	0.027	0.030	0.031	0.028	0.026	0.025	0.033	0.030	0.029	0.032	0.030	0.028	0.029	0.027	0.029	0.034	0.032	0.037
chr14	0.032	0.031	0.031	0.030	0.032	0.030	0.030	0.031	0.032	0.031	0.030	0.030	0.029	0.031	0.032	0.031	0.031	0.031	0.035
chr15	0.031	0.028	0.029	0.027	0.030	0.029	0.030	0.029	0.028	0.027	0.028	0.028	0.027	0.030	0.030	0.029	0.027	0.028	0.033
chr16	0.037	0.039	0.041	0.035	0.043	0.039	0.040	0.035	0.033	0.040	0.033	0.036	0.040	0.040	0.041	0.044	0.036	0.040	0.029
chr17	0.039	0.034	0.038	0.030	0.044	0.044	0.042	0.033	0.031	0.035	0.030	0.034	0.038	0.042	0.043	0.044	0.033	0.038	0.026
chr18	0.025	0.026	0.027	0.027	0.023	0.023	0.022	0.028	0.022	0.025	0.027	0.025	0.024	0.024	0.023	0.023	0.028	0.027	0.025
chr19	0.041	0.034	0.035	0.027	0.051	0.047	0.045	0.027	0.034	0.036	0.025	0.030	0.037	0.045	0.048	0.055	0.035	0.038	0.019
chr20	0.027	0.028	0.029	0.026	0.030	0.029	0.029	0.026	0.027	0.026	0.026	0.026	0.028	0.029	0.030	0.030	0.024	0.026	0.020
chr21	0.015	0.014	0.015	0.014	0.016	0.018	0.014	0.015	0.015	0.014	0.015	0.014	0.015	0.015	0.015	0.017	0.017	0.016	0.016
chr22	0.020	0.019	0.022	0.016	0.024	0.024	0.023	0.016	0.017	0.019	0.015	0.017	0.020	0.022	0.023	0.025	0.016	0.020	0.017
chrX	0.038	0.076	0.029	0.074	0.027	0.034	0.075	0.028	0.053	0.072	0.066	0.080	0.086	0.032	0.032	0.016	0.023	0.020	0.050
chrY	0.006	0.001	0.004	0.000	0.009	0.004	0.001	0.003	0.000	0.000	0.000	0.000	0.001	0.007	0.007	0.003	0.004	0.003	0.019

Figure A1.3. Proportion of partially methylated CpGs in different chromosomes

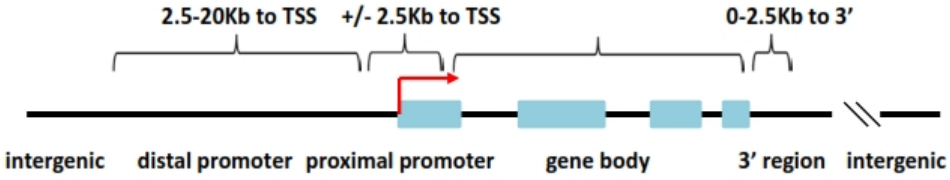


Figure A1.4. Definition of gene model used in our results.

Table A1.1. Bisulfite libraries information.

library label	**cell line/tissue**	**gender**	**differentiation stage**
H9	WA09 (H9)	female	embryonic stem cell
PEL	fibroblast derived from H9	female	fibroblast
NUFF	Newborn Human Foreskin Fibroblasts	male	Human Foreskin Fibroblasts
H1	H1	male	embryonic stem cell
IMR90	IMR90	female	lung fibroblast cell
H1NPC	H1 cell line differentiated into neural progenitor cells	male	Neural progenitor cells
iPS19.11	induced pluripotent cell	male	induced pluripotent cell
Blood	blood	male	blood
HCC1954	HCC1954	female	breast cancer cell
HMEC	HMEC	female	primary human mammary epithelial cells
Bcell	B cell	female	mature lymphocyte
HSPC	HSPC	female	progenitor
Neutrophil	neutrophil	female	mature granulocyte
H1_BMP4	H1_BMP4	male	embryonic stem cell differentiated by treatment with BMP4
H1_Mesendoderm_BMP4	H1_mesendoderm_BMP4	male	embryonic stem cell differentiated into mesendoderm cells
HSF1	HSF1	male	ES cells
Sperm1	sperm	male	sperm
Sperm2	sperm	male	sperm

Table A1.2. Statistics of bisulfite library mapping and partial-methylation calling

Library	Total reads	Uniquely mapped reads	Covered Cs (with 3+ coverage)	Partially-methylated Cs	Covered Cs in CpGs (with 3+ coverage)	Partially-methylated Cs in CpGs
H9	1,252,758,376	732,304,916	580,002,008	6,452,534	37,291,772	2,090,486
PEL	1,280,156,574	745,344,788	553,556,046	3,260,351	37,474,915	2,723,454
NUFF	1,321,093,122	663,911,302	536,680,438	4,021,523	36,734,335	3,584,060
H1	1,982,672,531	769,544,786	825,145,315	4,256,866	38,592,912	1,068,386
IMR90	2,817,649,029	742,213,396	833,934,828	4,342,363	40,914,135	4,183,597
H1NPC	1,970,863,133	1,024,456,755	926,880,865	1,161,963	49,664,560	967,331
iPS19.11	1,538,561,338	1,204,375,563	921,432,358	3,437,386	50,148,798	1,117,622
blood	1,587,460,142	935037280	915,203,119	5,604,701	39,927,494	4,760,027
HCC1954	2102598758	828619155	975,795,104	3,160,520	50,192,769	3,031,062
HMEC	1097792043	513359748	904,200,716	3,794,719	48,107,820	3,649,696
Bcell	1137586034	288198282	671,197,198	2,851,116	31,434,935	1,723,629
HSPC	650891167	325956458	1,010,825,010	4,173,993	46,540,622	2,306,948
neutrophil	662515394	308367492	758,996,943	2,887,905	41,439,759	1,738,892
H1_BMP4	1067309814	574658246	998,414,735	1,735,532	50,841,639	1,189,300
H1_mesendoderm_BMP4	1,248,076,985	950443316	958,197,979	1,626,897	48,331,716	1,175,966
HSF1	2063178999	89540695	71,222,576	1,302,939	4,670,588	403,290
Sperm_GSM752295	628135142	175265542	489,550,470	1,632,161	27,975,180	861,352
Sperm_GSM752296	940852056	204267987	406,088,738	2,914,756	25,880,438	1,650,318