UNIVERSITY OF CALIFORNIA
RIVERSIDE

Computational Methods for Epigenetic Studies

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Elena Yavorska Harris

June 2010

Dissertation Committee:
Dr. Stefano Lonardi, Chairperson
Dr. Karine Le Roch
Dr. Eamonn Keogh

The Dissertation of Elena Yavorska Harris is approved:

_____

_____

_____

Committee Chairperson

University of California, Riverside

Acknowledgements

I would like to thank the professors at UCR who contributed to my knowledge in bioinformatics and algorithm design: Dr. Stefano Lonardi, Dr. Tao Jiang, Dr. Neal Young, Dr. Marek Chrobak and Dr. Eamonn Keogh.

I deeply appreciate the expert advice and training, helpful suggestions, guidance, encouragement and support of my adviser, Dr. Stefano Lonardi.

I am very grateful to Dr. Karine Le Roch for giving me the opportunity to work on cutting edge projects and for the training that she provided during our collaborative work. I also would like to express my sincere gratitude to Dr. Nadia Ponts, an outstanding scientist in Dr. Le Roch's laboratory, for her helpful training and encouragement.

I also would like to thank Dr. Stefano Lonardi, Dr. Karine Le Roch and Dr. Eamonn Keogh for their helpful suggestions that improved this thesis.

Chapter 3 appeared in *Bioinformatics*, in the paper titled "BRAT: Bisulfite-treated Reads Analysis Tool", co-authored with Nadia Ponts, Alexandr Levchuk, Karine Le Roch and Stefano Lonardi [34].

Portions of Chapter 4 appeared in *Genome Research*, in the article titled "Nucleosome landscape and control of transcription in the human malaria parasite", co-authored with Nadia Ponts, Jacques Prudhomme, Ivan Wick, Colleen Eckhardt-Ludka, Glenn R. Hicks, Gary Hardiman, Stefano Lonardi, and Karine Le Roch [65].

Materials included in Chapters 2 and 5 are the results of collaboration with Nadia Ponts, Karine Le Roch and Stefano Lonardi.

I would like to thank my family: Keith, my husband; Vera Agapova, my mother; and Oksana and Alona, my daughters, for their love, encouragement and support.

For Vera Agapova

ABSTRACT OF THE DISSERTATION

Computational Methods for Epigenetic Studies

by

Elena Yavorska Harris

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, June 2010
Dr. Stefano Lonardi, Chairperson

The epigenome has increasingly been the focus of research over the past decade. Epigenetic control occurs in two primary ways: DNA *methylation* and chemical modification of *histones*. The latter mechanism determines whether the chromatin is tightly packed, in which case gene expression is repressed, or relaxed, in which case gene expression is enhanced. DNA methylation is the addition of the methyl groups to cytocines. Methylation of DNA is involved in a variety of biological processes, including embryogenesis and development, silencing of transposable elements, and regulation of gene transcription. In our research, we developed a set of computational methods and software tools that enable genome-wide epigenetic studies. In particular, our computational methods and software tools allow for (1) the determination of dynamic nucleosome positioning and the analysis of the correlation between the nucleosome landscape and gene expression, (2) the identification of methylation patterns from raw data obtained from next-generation sequencing technologies such as the Illumina Genome Analyzer, and (3) the discovery of transcription factors binding sites from data on the dynamic chromatin structure remodeling. Our methods and tools were used to study nucleosome landscape,

methylation and control of transcription in the human malaria parasite which is responsible to one million deaths world-wide every year.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Increasing evidence is becoming available that regulation of cell mechanisms is much more complex than previously thought. New discoveries in epigenetic studies shed light on some of the most puzzling questions, such as how differentiation of cells is regulated or why identical twins can have very different physical constitutions and develop different diseases despite their common genome. Now molecular biologists believe that they have answers to both questions.

A relatively recent discovery is that second layer of information exists, which is embedded on chemical markers added to DNA and in special proteins that package DNA into chromosomes. This second layer, called the epigenome, controls access to the genes, allowing each cell type to activate its own special genes and silence others. The epigenome is involved not just in defining what genes are accessible in each type of cell, but also in controlling when the accessible genes may be activated. In other words, an individual organism has one genome but many “spacewise” (i.e., tissue-specific) and “time-wise” (i.e., specific to developmental stage) epigenomes. Molecular biologists have also confirmed that the epigenome is affected by the interactions of the organism with the environment. In particular, the epigenomes of identical twins slowly differentiate over time, giving rise to distinct phenotypes.

Epigenetic control occurs in two primary ways: DNA *methylation* and chemical modification of *histones*. DNA methylation is the addition of methyl groups to cytocines. The second, more complex kind of alteration affects another class of proteins called histones, the proteins that package DNA into *chromatin* (the main component of chromosomes). Chemical

modifications of histones determine whether the chromatin is tightly packed, in which case gene expression is repressed, or relaxed, in which case gene expression is enhanced. Chromatin is packed into *nucleosomes* with short distances between them. Each nucleosome is a formation of eight histones locked together to form a miniature spool with 147 base pairs of DNA twisted around the histones. Each histone has a protruding "tail" to which more than twenty chemical tags can attach. Some of these tags, or certain combinations of them (which define a new "language" called the histone code) give rise to relaxed chromatin (free from nucleosomes); others have the opposite effect. When DNA has to replicate for cell division, methylation marks and histone modifications pass only to the two parent strands and all the nucleosomes are disassembled; yet the cell has ingenious methods for reconstituting the same marks on the two daughter genomes. The whole system is called the epigenome because these chemical marks are inherited across cell division despite not being encoded directly in DNA.

The epigenome has increasingly been the focus of research over the past decade, and several international projects are currently under way. The objectives of the International Human Epigenome Project are "to identify, catalog and interpret genome-wide DNA methylation patterns of all human genes in all major tissues" (http://www.epigenome.org/). So far, IHEP has analyzed methylation patterns in a few dozen individuals in selected regions of three chromosomes [25, 66]. The Alliance for Human Epigenomics and Disease is also coordinating a comprehensive human epigenome mapping project [41]. The focus is on developing a suitable bioinformatics infrastructure and performing epigenome mapping in a selection of normal tissues, which may provide the reference for subsequent mapping in cancer cells. Finally, the NIH-funded Encyclopedia of DNA Elements project aims to map all functional elements in the human genome sequence including large-scale mapping of DNA methylation and histone modifications [11]. Epigenetic therapy is also becoming a reality: while epigenetic marks are heritable in

somatic cells, drug treatments could potentially reverse them. This has significant implications for the prevention, diagnosis and treatment of major human diseases and for the normal effects of aging. Potential targets include diabetes, cardio-pulmonary diseases, autoimmune diseases and cancer, in which missteps in epigenetic programming have been directly implicated. Several inhibitors of chromatin-modifying enzymes have now been approved by the FDA or are in clinical trials with good prognosis for tumor regression.

In our research, we developed a set of computational methods and software tools that enabled comprehensive epigenetic studies. In particular, our computational methods and tools allow for (1) the determination of dynamic nucleosome positioning and the analysis of the correlation between the nucleosome landscape and gene expression, (2) the identification of methylation patterns from raw data obtained from high-throughput (next-generation) sequencing instruments such as the Illumina Genome Analyzer, and (3) the discovery of transcription factors binding sites using the data on the dynamic chromatin structure remodeling.

The organism on which we developed our algorithms is *Plasmodium falciparum*, the parasite responsible for 350–500 million cases of malaria each year and between one and three million human deaths worldwide, the majority of whom are young children in Sub-Saharan Africa.

Due to its modest genome size, malaria is an ideal platform to test the proposed methodologies and to develop algorithms and tools. The genome of *P. falciparum* is, however, not lacking of technical challenges. Its fourteen chromosomes represent the most (A+T)-rich genome sequenced to date. The average (A+T) fraction is 80.6% genome-wide, and higher than 90% in the intergenic regions. Due to the effective reduction of the alphabet cardinality from four symbols to two, the genome contains an abundance of low-complexity repeats that complicates the mapping of short reads to the genome.

The methods and tools developed here are applicable to a variety of other eukaryotes, including human.

1.1 Nucleosomes mapping

In eukaryotic cells, genomic DNA exists as a highly compacted structure called chromatin. Because DNA is negatively charged, the resulting electrostatic repulsion would prevent this polymer from fitting into the small confines of the nucleus. The problem is solved by the presence of histones that bind to the DNA and neutralize the negative charge.

The fundamental unit of chromatin is a nucleosome composed of 147 base pairs (bp) of DNA wrapped 1.65 turns around a protein complex of eight histones. There are 14 points of contact between histones and DNA, which make the nucleosome one of the most stable protein-DNA complexes known. The presence of nucleosomes directly affects a variety of cellular and metabolic processes, including transcription (gene expression), recombination, replication, centromere formation, DNA repair, and so forth. The more condensed the chromatin, the harder it is for transcription factors and other proteins to access the DNA and carry out their tasks. The more accessible the DNA, the more likely it is that surrounding genes are actively transcribed. Nucleosomes represent an additional layer of information that directly affects gene regulation in addition to the well-known regulatory elements, such as promoters, enhancers, silencers, and response elements.

Most of genomic DNA is organized in densely packed chromatin. For example, it is estimated that 75-90% of the human DNA is wrapped in nucleosomes. Nucleosomes tend to be spaced quite regularly along the genome, at a distance from each other that is organism specific: ≈ 18 bp in yeast [47, 58, 73], ≈ 28 bp in drosophila [59] and *C. elegans* [81], and ≈ 38 bp in

humans [7]. Very long linkers or nucleosome-free regions (NFRs) tend to occur at the beginning and the end of genes (see Figure 8). The positioning preference of nucleosomes depends on a variety of biochemical factors, including the ability of DNA to bend sharply around the histone complex, which in turn depends on the DNA composition. Since the 1980s, several studies have attempted to discover recurrent patterns in DNA sequences that favor nucleosome positioning (see, e.g., [80, 68, 85, 71, 39, 63, 15, 33, 93, 72, 20, 67]). Several authors have observed periodic occurrences of AA, TT, and TA dinucleotides spaced at 10 bp intervals, and occurrences of GC also spaced at 10 bp but 5 bp out of sync with respect to AA, TT and TA. It turns out that these patterns are "statistically weak", and thus difficult to detect. Despite the statistical enrichment of AA, TT, TA and GC associated with nucleosomes, the presence of these dinucleotide patterns in individual nucleosomes only occurs modestly above a random distribution.

Nucleosomes are not static units. They possess dynamic properties that are tightly regulated by chromatin remodeling protein complexes. These complexes bind nucleosomes and the adjacent linker, then move nucleosomes in the direction of the linker (see, e.g., [48] and references therein).

In order to correlate nucleosomes and gene expression, it is essential, therefore, to study the nucleosomes positioning preference over time. A handful of experimental techniques have been developed for genome-wide mapping of nucleosomes. For nucleosomes, one can enrich for genomic regions that are bound to histones (typically via chromatin immuno-precipitation or ChIP) or for genomic regions that are free of nucleosomes. Then, tiling microarrays (ChIP-chip) or sequencing (ChIP-seq) is applied to the enriched DNA. Finally, software tools are used to infer the epigenetic information from the tiling array or sequencing data. Maps of nucleosome occupancy produced via tiling microarrays include *S. cerevisiae* [53, 47, 74, 42, 94], *C. elegans* [40], and human [69] in various cell types and under a variety of physiological perturbations.

High throughput sequencing was used to produce nucleosome maps for *S. cerevisiae* [1, 73, 58], *C. elegans* [81], and drosophila [59].

1.2 DNA methylation

The other major epigenetic player is methylation, which affects mostly cytosine bases in DNA. An enzyme called *DNA methyltransferase* affixes a methyl group to cytosine, creating a different but stable nucleotide called 5-methylcytosine. Accounting for a significant fraction of the nucleotides within eukaryotic genomes, 5-methylcytosine imparts an additional layer of heritable information on top of the DNA code.

In mammals, DNA methylation happens almost exclusively in a symmetrical way in a CG context in somatic cells. In the recent publication [51], embryonic stem cells contain a higher and significant proportion of CHG and CHH methylation, where H represents A, C or T. In *Arabidopsis* cytosines are methylated in CG, CHG, and CHH contexts [18]. DNA methylation is involved in a variety of biological processes, including embryonic development, genomic imprinting, silencing of transposable elements, and regulation of gene transcription. Methylation in the promoter region of a gene typically results in gene silencing. Methylation has been shown to be essential for mammalian embryonic development: mice that lack DNA methyltransferase die within a few days [23]. In the earliest stages of life, DNA methylation influences differentiation of embryonic stem cells into the different cell types that constitute the diverse organs, tissues and systems of the body.

Recent research has shown, however, that environmental influences and experiences, such as the type of care a young rat receives from its mother, can also result in characteristic methylation patterns and corresponding behaviors that are heritable for several generations [84].

Methylation patterns depend on the cell type [10, 16]. They also change throughout development, and under normal and disease states. A strong link between DNA methylation and cancer has been discovered: not only CpG islands in the promoters of tumor-suppressor genes are hyper-methylated, but also genome-wide hypo-methylation is observed as the tumor progresses (see, e.g., [79]).

Although DNA methylation is considered one of the most stable epigenetic marks, it is not permanent. There are known enzymes that can demethylate DNA genome-wide, but the mechanisms that drive demethylation at specific active sites are still poorly understood. Demethylation of DNA can occur passively by failure to methylate newly synthesized DNA during replication or actively in the absence of DNA replication [86, 95]. It is crucial, therefore, to study methylation as a dynamic process.

Assays that can be used to detect DNA methylation include methylation-sensitive restriction enzymes, bisulfite conversion, and affinity purification. In restriction enzyme-based approaches, a methyl restriction enzyme is used to cut DNA at specific patterns such as CCGG if the internal C is unmethylated; followed by sequencing and subsequent mapping to a reference genome, these methods indirectly reveal methylated cytocines. In affinity purification methods (MedIP, MAP, CAP) antibodies to methylated cytosines are used to immunoprecipitate the methylated portion of the genome, which is followed by sequencing of the selected genome regions. These first two methods are relatively inexpensive, but they suffer from several limitations and the resolution of methylation detection is not high (see [70, 50] for details). Bisulfite conversion is considered the “gold standard” in terms of resolution, but it can be very costly. Treatment of DNA with sodium bisulfite converts unmethylated cytosines to uracils, but leaves methylated cytosines unchanged. A few runs of PCR followed by sequencing can distinguish between methylated and unmethylated bases. Due to its cost, whole-genome bisulfite

sequencing has been applied only to a few species such as *Arabidopsis* [20, 51], to a few human chromosomes [25, 66], and to the mouse [60]. Recently, whole-genome bisulfite sequencing has been used on the entire human genome [52].

1.3 Gene regulation in *P. falciparum*

Despite almost a decade of investigation into the 23-million bp genome of *P. falciparum* [31], more than half of its 5,595 predicted genes are still annotated as "hypothetical", and the mechanisms driving gene expression are almost completely unknown. Genome-wide gene expression analysis has shown remarkable variations of mRNA levels with a tight regulation throughout the parasite life cycle [12, 46]. Recent studies have demonstrated that the parasite has a limited capacity to regulate the expression of its genes in response to metabolic stresses [30, 45]. To date, however, only a handful of transcription factors have been validated [5, 19, 54]. How does the malaria parasite regulate its genes in response to changes in the environment with such a small number of transcription factors? We believe that malaria regulates most of its genes via epigenetic mechanisms (i.e., methylation, histone tail modification, and nucleosome positioning).

As stated above, in eukaryotic cells alterations of chromatin structure are known to modulate DNA accessibility to DNA-binding proteins and thus regulate gene expression. In *P. falciparum*, several recent studies indicate that chromatin remodeling controls the exclusive expression of an important family of genes responsible for antigenic variation at the surface of infected red blood cells (see [17, 22, 24, 64, 28, 55]).

We developed a methodology for discovery of transcription factors binding sites or motifs using the data from our study on the dynamic chromatin structure remodeling.

1.4 Next generation sequencing technology

Next-generation sequencing technologies (454, Illumina/Solexa, and ABI SOLiD) are rapidly expanding the horizon of possibilities to study DNA and RNA. These technologies help scientists to discover SNP, CNV and other chromosomal rearrangements, DNA-protein interactions, coding and non-coding RNAs, and to study the chromatin packaging and methylation of DNA [57]. Most of these studies (including methylation identification and nucleosomes mapping) rely on a reference genome on which the sequenced reads are mapped. The problem of mapping or anchoring of the reads to a reference genome is computationally challenging due to the quantity of data produced by the next generation sequencers. For example, Solexa and ABI sequencers produce reads ranging between 24-75 and 25-35 base pairs respectively and have throughputs of hundreds of millions of reads per run. Mapping hundreds of millions of reads accurately and quickly turns out to be a non-trivial computational problem. It has been shown that widely known software such as BLAST [2, 3] and BLAT [43] cannot efficiently accomplish this new task, and several software tools have been developed in the last couple of years to solve it.

We believe that molecular biologists want to know the strengths and weaknesses of the mapping software tools in order to make an informed choice when it is time to analyze their data, and therefore our first project is to compare and analyze some of the new mapping tools.

1.5 Contributions and structure of this thesis

The tools and methods presented in this thesis enabled genome-wide studies on the role of nucleosomes and methylation in regulating *P. falciparum*'s transcripts [65]. Both of these research projects on dynamic chromatin structure remodeling and methylation patterns in the malaria parasite used the sequenced DNA from the Illumina Genome Analyzer, and therefore

involved mapping sequenced reads to the reference genome as the first step in the analysis pipeline. In order to study which particular mapping tool is the most suitable for our projects, we benchmarked the most popular mapping tools comparing their mapping accuracy and time/space efficiency. We believe that molecular biologists will benefit from our benchmarking as well when they are to choose between the mapping tools for their projects. Chapter 2 provides the details and results of the benchmarking of the representative mapping tools for short reads from the next generation sequencing technology.

For the study of methylation regulation in *P. falciparum*, our collaborators used bisulfite-treatment of DNA that changes unmethylated cytocines to uracils that eventually are converted to thymines during following PCR (polymerase chain reaction). The mapping of bisulfite-treated reads to a reference genome is even more challenging than mapping regular short reads. Along with the normal mappings of a base in a read to the same base in a genome, the mapping of bisulfite-treated reads must allow for additional matches of T in a read to C in a genome and A in a read to G in a genome. Using the existing mapping tools with bisulfite-treated reads turned out to be inefficient both in time and accuracy of mapping. We developed a new, accurate and highly efficient mapping tool for bisulfite-treated reads called Busilfite-treated Reads Analysis Tool (BRAT). This tool was used to study methylation in the human malaria parasite. Chapter 3 describes BRAT and additional tools for end-trimming of low quality bases of the reads and for nucleotide counting of the mapped reads at each base of the reference genome.

The human malaria parasite offers unprecedented opportunities for studying dynamic nucleosome landscape due to the changes the parasite undergoes during the erythrocytic cycle or the human blood cycle. The erythrocytic cycle takes 36-40 hours, during which the parasite goes through different stages of development: ring, trophozoite and schizont. The study on dynamic nucleosome remodeling during the erythrocytic cycle shed the light on the mechanisms that the

parasite employs to regulate gene expression. Chapter 4 describes methods that enable study on dynamic chromatin structure remodeling during the erythrocytic cycle of the human malaria parasite.

Our method for motif discovery that used the data on dynamic chromatin structure remodeling helped to identify *cis*-regulatory elements very similar to the previously validated motifs as well as to discover new motifs statistically enriched in functional gene sets [35]. Our approach for motif discovery and the results of its application to the human malaria parasite are described in Chapter 5.

Chapter 6 provides a summary of the proposed methods and tools and demonstrates contributions of this research to the epigenetic studies.

Chapter 2

Benchmarking of mapping tools

Next-generation sequencing technologies, specifically 454, Illumina/Solexa, and ABI SOLiD, are capable of producing hundreds of millions of sequenced reads in a single run. Due to the quantity of data produced, the downstream analysis can be computationally challenging. Among the variety of tasks in the analysis pipeline, the most computationally intensive is the mapping of the reads to a reference genome. Many software tools have been developed in the last couple of years to solve the mapping problem efficiently. Dependent on the goal of a software application, there are three major categories of tools for short reads: tools for de novo sequence assembly, tools for mapping RNA-sequenced reads and tools for mapping short reads to a reference sequence. The latter tools can be further classified dependent on the techniques that the tools employ: tools that use Burrows-Wheeler transform [14]; tools that use Smith-Waterman algorithm [77]; and tools that use hash index table. We focused our study on tools for mapping reads to reference sequences, and in this chapter we present a comparative analysis of these tools. We chose representative tools that employ different techniques: Efficient Large-scale Alignment of Nucleotide Databases (ELAND) (http://bioinfo.cgrb.oregonstate.edu/docs/solexa/), Short Read Mapping Package (SHRiMP) [90], Oligonucleotide Alignment Program (SOAP) [49], Mapping and Assembly with Qualities (MAQ) (http://maq.sourceforge.net/index.shtml), RMAP [78], and Bowtie [44]. ELAND, SOAP, MAQ and RMAP utilize hash index table; SHRiMP employs Smith-Waterman algorithm; and Bowtie uses Burrows-Wheeler transform. We evaluated and compared the accuracy, sensitivity and need for computational resources (memory and time) of

these tools with respect to different types and quantity of sequencing errors, read length, genome length and choices of parameters. Our results will help biologists choose wisely (depending on the quality of the sequence data and the specific task at hand) among the mapping tools presented here.

2.1 Background

Next-generation sequencing technologies (454, Illumina/Solexa, and ABI SOLiD) are rapidly expanding the horizon of possibilities to study DNA and RNA. These technologies help scientists discover SNP, CNV and other chromosomal rearrangements, DNA-protein interactions, coding and non-coding RNAs, and study the chromatin packaging and methylation of DNA [57]. Most of these studies rely on a reference genome on which the sequenced reads are mapped. The problem of mapping or anchoring the reads to a reference genome is computationally challenging due to the quantity of data produced by the next generation sequencers. Here, we focus on the problem of anchoring short reads obtained with Illumina/Solexa Genome Analyzer and ABI SOLiD System. Solexa and ABI sequencers produce reads ranging between 24-100 and 25-50 base pairs respectively and have throughputs of hundreds millions reads per run. Mapping hundreds of millions of reads accurately and quickly turns out to be a non-trivial computational problem. It has been shown that widely-known software such as BLAST and BLAT cannot efficiently manage this new task, and several software tools have been developed in the last couple of years to address the problem of anchoring efficiently. We believe that molecular biologists want to know the strengths and weaknesses of each software tool in order to make an informed choice when they analyze their data.

In this chapter we report on a comprehensive comparison of six computational tools that are available to the scientific community to anchor short reads to a reference genome (a thorough list of mapping tools can be found at http://seqanswers.com/forums/showthread.php?t=43). These tools are: MAQ, SOAP, SHRiMP, RMAP, ELAND, and Bowtie. The comparison involves measuring mapping accuracy, time and memory requirements as well as the sensitivity of the tools to different parameters. We tested the tools both on synthetic data obtained by shredding a genome *in silico*, and on real reads obtained by the Solexa 1G sequencing machine.

The first set of experiments used *E. Coli* and human *X* chromosome as reference genomes and the reads generated from these genomes *in silico*. We simulated the reads sets containing different number and types of sequencing errors: mismatches as well as insertions and deletions (*indels*). Then we tested each tool on each of the reads sets measuring mapping accuracy, time and memory usage, and the distribution of mapped reads by the number of mismatches or indels. Since the mapping on a small genome is performed much faster, we used *E. Coli* as a reference genome to study sensitivity of RMAP, SOAP and SHRiMP to different parameters such as a seed width and the number of hits per window (the latter parameter is used with SHRiMP only).

The second part of experiments used human genome (hg18) and a real set of the reads sequenced by the Solexa Genome Analyzer.

The subsequent sections are organized as follows. In the next section, we describe briefly algorithms that are used by the tools and give a short overview of each tool. In Materials and Methods, we discuss the system on which we ran tests, the parameters that we chose for the tools, test data sets, and the measurements for the experiments. In the Results section, we present the results of the tests and provide an overview of the specific behavior of each tool. Finally, the Discussion summarizes the main points of this chapter.

2.2 Overview of the representative set of tools

From the algorithmic perspective, all of the tools (except Bowtie) are relatively similar. Each program uses a variant of the strategy called *k*-mer filtering method [6] to speed up the process of finding potential matches in the genome. Seeds are short substrings of a fixed length on which a hash table is constructed. The hash table is built in the preprocessing phase as follows. If the j^{th} read in the input contains an exact occurrence of seed *x* then the entry in the table corresponding to *x* contains the index *j* and the offset of *x* in *j*. In order to determine where the reads match the genome, the keys in the hash table are used as "seeds" to identify possible candidates.

While one can build the hash table either on the reads or on the genome, four tools, namely ELAND, RMAP, MAQ and SHRiMP, index the reads. The authors of SOAP index the reference genome and keep the genome in memory. All of the tools that employ a hash table use binary representation of the genome and reads together with binary logic operations to identify approximate matches (up to a specified number of mismatches). SHRiMP uses an improved Smith-Waterman algorithm to align possible candidates reads to the selected regions of the genome. In what follows, we give a short description of each tool (specifics of the tools refer to the time when this project was carried out).

Bowtie performs alignments using the Burrow-Wheeler transform, which involves a preprocessing step in which the tool builds a compressed data structure on a reference genome. Bowtie expands the usual algorithm of finding a pattern in a given text using the Burrow-Wheeler transform to allow for up to three mismatches. Reads are aligned one character at a time. Bowtie supports paired-end mapping. ELAND, MAQ and RMAP are the tools that use a hash index table. At the time when this benchmarking was done, none of these tools allowed gaps; now ELAND allows a single gap up to 32 bases.

ELAND is a part of Solexa-Illumina data processing package. We tested ELAND as a stand-alone program. At the time of these tests, ELAND was capable of finding ungapped alignments with up to 2 mismatches in the first 32 bases of the reads.

MAQ performs ungapped alignments with the number of mismatches up to a user-specified parameter. In addition to mapping, MAQ also identifies SNPs, performs the assembly of the reference genome (when the reference genome is known), supports paired-end alignment and can map reads to itself (MAQ identifies only overlaps of the reads over the entire length). In addition, MAQ uses base quality score to measure the error probability of each alignment and reports a mapping quality score for each read. MAQ does not support variable length of reads (all the reads in the input file must be of the same length.)

The RMAP program finds ungapped alignments up to a user-specified number of mismatches. RMAP allows reads with different lengths, but aligns only the first *k* bases of each read, where *k* is the length of the shortest read in the input set. RMAP supports paired-end reads and unlimited read length.

SOAP and SHRiMP both allow gaps in alignments. SOAP first tries to find alignments with the smallest number of mismatches up to a user-specified number of mismatches, and then looks for the alignments with a single continuous insertion or deletion up to 3 bases long. SOAP does not support mapping with substitutions and gaps in the same alignment, nor insertions and deletions in the same alignment. SOAP is the only program of the tools tested here that hashes the reference genome and keeps the reference genome in the memory. SOAP supports paired-end alignment and multithread parallelism.

SHRiMP performs alignments with the Smith-Waterman algorithm and therefore allows for mismatches and indels in same alignment. The input file can contain reads of different lengths with a user-specified length of the longest read; however, since the alignment score threshold is

also specified by the user, there is one threshold for all the reads, which allows alignments of longer reads to have significantly more mismatches or/and indels per read length than alignments of the shorter reads. SHRiMP also provides a tool that calculates the probability of the alignment occurring by chance, and another tool that prints the actual alignments for visual inspection.

2.3 Materials and methods

2.3.1 System and parameters

All tests were performed on SGI SAL Version 1.15 with 32 GenuineIntel processors, IA-64 architecture, 1594 CPU MHz and 64,232,368 KB memory. The operating system was SUSE Linux with gcc version 4.1.0.

This project was carried out Spring-Fall, 2008. The five mapping tools were MAQ (2), SOAP (3), SHRiMP (4), RMAP (5) and ELAND (6); and the versions of these tools used were: maq-0.6.5_i686-linux, soap_1.05, SHRiMP_1_0_5.lx26.i686, rmap_v0.3 and ELAND as a part GAPipeline-0.3.0.

2.3.2 Test data

For reference genomes we chose human *X* chromosome (http://www.sanger.ac.uk/HGP/ChrX/), hg18 (http://hgdownload.cse.ucsc.edu/goldenPath/hg18/bigZips/) and *E. Coli* genome (9). The lengths of the genomes are 154,826,262 bp, 3,080,419,480 bp and 5,153,376 bp respectively. For human X chromosome and *E. Coli* genome, the read sets were simulated, and for hg18 the reads were sequenced with Solexa-Illumina sequencer. The reads for hg18 are publicly available at (http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=table&f=run&m=data&s=run); we took

the reads with accession number SRX000168, run10_lane2, files numbered from 1 to 40 (reads containing N(s) were removed from the set); the length of reads in this set was 24 and the total number of reads was 1,007,029.

MAQ requires read files in FASTQ format that provides quality score for the bases of each read. We assigned a uniform base score of 30 to all bases of the reads used in our tests.

To study the sensitivity of the tools to different types and number of errors, time and space requirements and mapping accuracy, we used reads randomly generated from the human X chromosome and the *E. Coli* genome. First, we randomly generated 50-bases reads and recorded the positions of the reads in the corresponding genome. We chose reads that only contained characters from the DNA alphabet (A, C, G and T). The number of reads in each of the test sets is one million. To generate reads with different length, we took the first *k* bases of the 50-bases for several *k* (16, 24, 32 and 40).

To study the sensitivity of the tools to different types and number of errors, we used 32-bases reads and generated different reads sets each with a different type/number of sequencing errors: substitutions, insertions and deletions. We ran tests on sets with 1, 2 and 3 introduced errors per read. For the sets with substitutions, we selected a base within each read uniformly at random and substituted this base with a randomly chosen nucleotide from the DNA alphabet. For the sets with *m* insertions, we uniformly at random chose each position within 32 nucleotides, then inserted a randomly chosen nucleotide at this position, and finally took the first 32 bases of the modified read. Thus the sets with insertions had at most *m* insertions at different positions of each 32-base read. For the tests with deletions, we randomly chose a single position within the first 32 bases of the 50-bases read, deleted *m* nucleotides starting at the selected positions and took the first 32 nucleotides of the resulting reads. The total number of reads sets for each reference genomes (human chromosome X and *E. Coli*) was thirteen: four sets with reads of

length 16, 24, 32 and 40, three sets with substitutions (1, 2 and 3 mismatches), three sets with insertions (1, 2 and 3 insertions at different positions in a read), and three sets with deletions (1, 2 and 3 deletions at a single position of a read).

2.3.3 Mapping accuracy

To address the question whether a program maps reads to the original positions in the genome accurately, we measured two parameters: the number of missed ambiguous reads identified by a tool and the *mapping accuracy* as defined below. Both parameters were measured only for *in silico* reads. If a read can be mapped with the least number of mismatches to a unique position in the genome (considering both forward and reverse orientation), then the read is *unique*; otherwise, it is *ambiguous*.

For some applications it is important to distinguish between unique and ambiguous reads; that is why we chose a number of missed ambiguous reads identified by a program as one of the measurements of the program's performance. Each tool tested here has a way to distinguish between unique and ambiguous reads. If a tool claims that a mapped read is unique, but there exist two or more equally good alignments for it, then the read is considered to be a missed ambiguous read. To determine truly unique reads, we wrote a tool that identified ambiguous reads for the sets of reads with no errors and for the sets of reads with substitutions (ELAND and RMAP showed the same results as our program in identifying the number of ambiguous reads.)

Since ambiguous reads can be mapped to many locations, we defined *mapping accuracy* only for unique reads. The mapping accuracy of a program is the ratio between the number of unique reads mapped by the program to correct positions and the total number of unique reads in the input set (as defined by our tool). Here the correct positions are the positions in the genome from which the *in silico* reads were taken. If a tool has a mapping accuracy less than 100%, then

there are two possible sources of inaccuracy: (1) the mapping tool has a different definition for ambiguous reads and identifies more ambiguous reads (which are not included in the calculation of the mapping accuracy); or/and (2) the mapping tool cannot find all possible alignments for some unique reads.

For the data sets with substitutions, mapping accuracy was measured slightly differently. Since *m* substitutions were introduced randomly in each read, there is a non-zero chance that the corrupted read can be mapped to the genome with a smaller number of mismatches. We took this into account and with an *m*-substitution set we measured mapping accuracy only for the reads mapped uniquely with exactly *m* mismatches. For a set with *m* substitutions, the mapping accuracy was calculated as the ratio of unique reads mapped to the correct positions in the genome with *m* substitutions and the total number of unique reads whose best alignments had *m* substitutions. The exact formula for mapping accuracy with the sets with substitutions is given below. Let *T* be the total number of reads in the input, *R* denote the total number of uniquely mapped reads to correct positions with *m* mismatches, *A* denote the total number of true ambiguous reads as calculated by our tool and M_i be the number of unique alignments with *i* mismatches with $0 \leq i <$ m, where *m* is the number of substitutions introduced to the set. Then the mapping accuracy *MA* is given by

$$MA = 100R/(T - A - \sum_{0}^{m-1} M_i)$$

The mapping accuracy for the sets with *m* indels was not calculated because of the ambiguity in interpretation of the trimmed ends of reads with SHRiMP. There were cases when SHRiMP reported alignments of the trimmed reads without counting the trimmed bases as mismatches or indels, i.e., in some cases SHRiMP finds alignments that do not include end bases

of the reads. One can interpret the missing bases as mismatches, insertions or deletions, and the results for the mapping accuracy differ significantly for these different cases.

Here we provide the ways employed by different tools to let users distinguish between unique and ambiguous reads. MAQ sets the mapping quality score to 0 for ambiguous reads, RMAP has an option of producing ambiguous reads into a separate file, and ELAND identifies ambiguous reads by using type of a match output parameter. SHRiMP has an option that sets the number of best hits to be printed out. SOAP has an option that counts the number of the best hits for each read.

2.3.4 Time and space measurements

Time was measured using *time* in the command line when running a program; in particular, we recorded *real time*. Space was measured using the command *top*. We ran tests with the same parameters at different times, when the computer load varied. For most of the programs tested, time was not significantly affected, with the exception of SOAP, which was very sensitive to current CPU load. For example, for the set with 32-reads with no errors, SOAP showed 326 and 185 seconds at different times. We report the worst time measured.

2.4 Results

2.4.1 Human *X* chromosome

2.4.1.1 Different length reads

Tables 2.1, 2.2 and 2.3 show space (MB), time (seconds) required by each program, and parameters used when running tests on different length reads with exact alignments (no

mismatches). ELAND does not offer any options. MAQ finds alignments up to *m* mismatches (option *n* in the command line), and when running a test with the number of mismatches set to 0, MAQ was extremely slow; that is why in these tests, we set *n* to 1.

Program	16 bases	24 bases	32 bases	40 bases
maq	539	539	539	539
soap	893	1058	3215	3215
shrimp	881	1,003	1,132	1,254
rmap	N/A	510	526	526
eland	91	217	345	N/A
bowtie	139	139	139	139

Table 2.1: Space in MB in exact alignment with different length reads.

Program	16 bases	24 bases	32 bases	40 bases
maq	3,143	436	449	611
soap	67,947	340	326	341
shrimp	10,326	2,540	4,842	4,860
rmap	170	176	178	179
eland	1,611	361	326	N/A
Bowtie	37	45	53	60

Table 2.2: Time in seconds in exact alignment with different length reads.

Program	16 bases	24 bases	32 bases	40 bases
maq	n1	n1	n1	n1
soap	s6,v0	s10,v0	s12,v0	s12,v0
shrimp	s10,w32,n2,h16	s13,w24,n2,h24	s13,w32,n2,h32	s13,w32,n2,h40
Rmap	S12,m0	s12,m0	s12,m0	s12,m0

Table 2.3: Parameters in exact alignment with different length reads.

Mapping accuracy was 100% for all programs in this test. MAQ, RMAP, ELAND and Bowtie used less than 1GB of memory, with Bowtie using the least amount. SOAP used a little more than 3GB of memory for reads of length greater than 32. Time shown for 24, 32- and 40-bases reads was comparable for MAQ, RMAP, SOAP and ELAND with RMAP showing the best results among these tools, the tools that use hash index table. SHRiMP took longest due to time required by Smith-Waterman alignment algorithm, and Bowtie was the fastest among all the tools. Most of the programs excluding RMAP and Bowtie were sensitive to smaller read length – the time usage increased with 16-bases reads. SOAP had a restriction on a seed length dependent on the length of reads (that is why in the test with 16 reads, the seed was 6); the small seed size caused running time to rise drastically.

2.4.1.2 Data sets with substitutions

Figure 2.1 shows the number of alignments found with 0, 1, 2 and 3 mismatches (calculated only for unique reads), the number of ambiguous reads, and missed ambiguous reads for each program when the sets with substitutions (1, 2 and 3) were used.

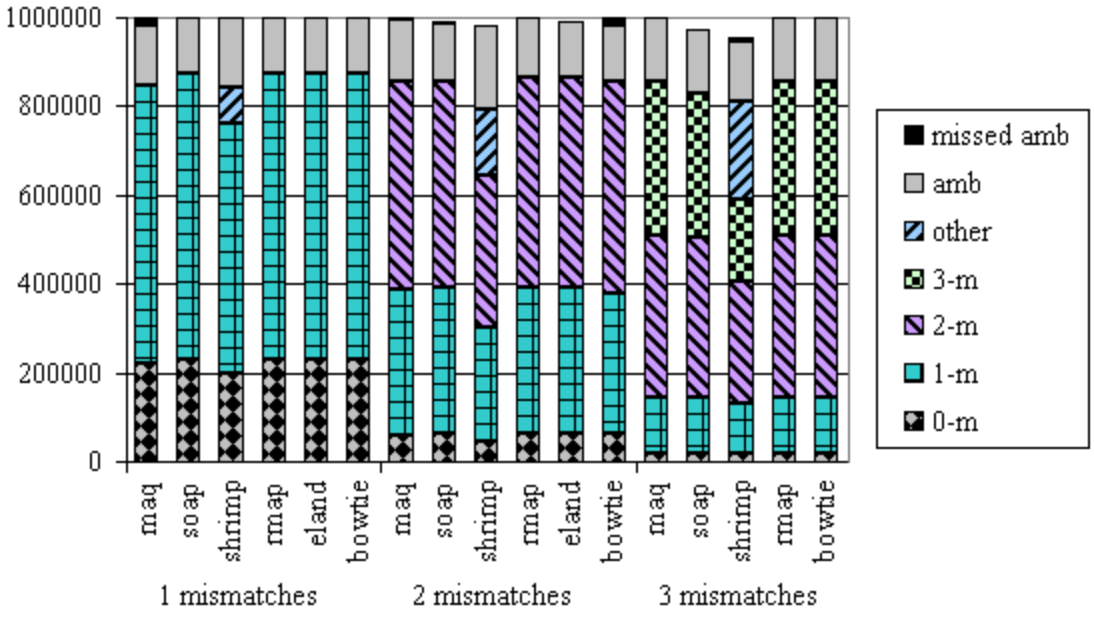

Figure 2.1: Data sets with substitutions.

In Figure 2.1, the category *other* is applicable only to SHRiMP. SHRiMP trims the ends of reads in some alignments; specifically, the program reports some alignments with read start alignment position greater than one or/and with read end alignment position less than the read's length. SHRiMP does not count these missing nucleotides at the end of a read as mismatches or indels. In addition, SHRiMP finds alignments that have mismatches and indels in the same alignment. In Figure 2.1, the category *other* is the sum of the alignments with mixed types errors and the alignments with mismatches whose reads' ends are trimmed.

In tests with 1 and 2 substitutions, the seed width was the same for all programs, and the results were slightly different but comparable. RMAP and ELAND showed the same results. The number of ambiguous reads identified by MAQ was slightly higher (5% more than that of RMAP for sets with 1 and 2 substitutions); it might be the case that MAQ defines ambiguous reads

differently or that MAQ assigns the mapping score of zero to some alignments that are not ambiguous. The number of ambiguous reads identified by SHRiMP was significantly higher: 23% and 39% more ambiguous reads than reported by our program for sets with 1 and 2 substitutions respectively.

MAQ missed some ambiguous reads: 15% of all ambiguous reads in case with 1-substitution data set, and 2.6% of the 2-substitution data set. SOAP missed 1% and 3% of ambiguous reads in tests with 2 and 3 substitutions respectively. SHRiMP missed 7.7% of the ambiguous reads in the 3-substitution data set. Bowtie missed 10.8% of ambiguous reads in the 2-substitution set.

The total number of alignments reported by the programs was comparable, with Bowtie finding the least number of alignments, 95%, 92% and 90%, with 1-, 2- and 3-substitution sets respectively. For a 3-substitution set, the seed width parameters were different for different programs: the seed for RMAP was 8 and for SOAP and SHRiMP the seed was 9. SHRiMP is also sensitive to another parameter, namely the number of hits per window (*n*), which we discuss later. Since in tests with the human X chromosome as the reference genome we set *n* to 2, SHRiMP did not show the best results in term of the total alignments found. However, the results improved when the seed width was decreased as shown in the results with *E. Coli* genome.

Figure 2.2 shows the mapping accuracy of the programs with sets with substitutions. The mapping accuracy of RMAP and ELAND was 100% in all applicable cases. SOAP showed 100% mapping accuracy in the 1-substitution data set, and 97.8% and 92.6% in the 2- and 3-substitution data sets respectively. MAQ showed slightly lower mapping accuracy due to the reason mentioned above: it defines ambiguous reads differently.

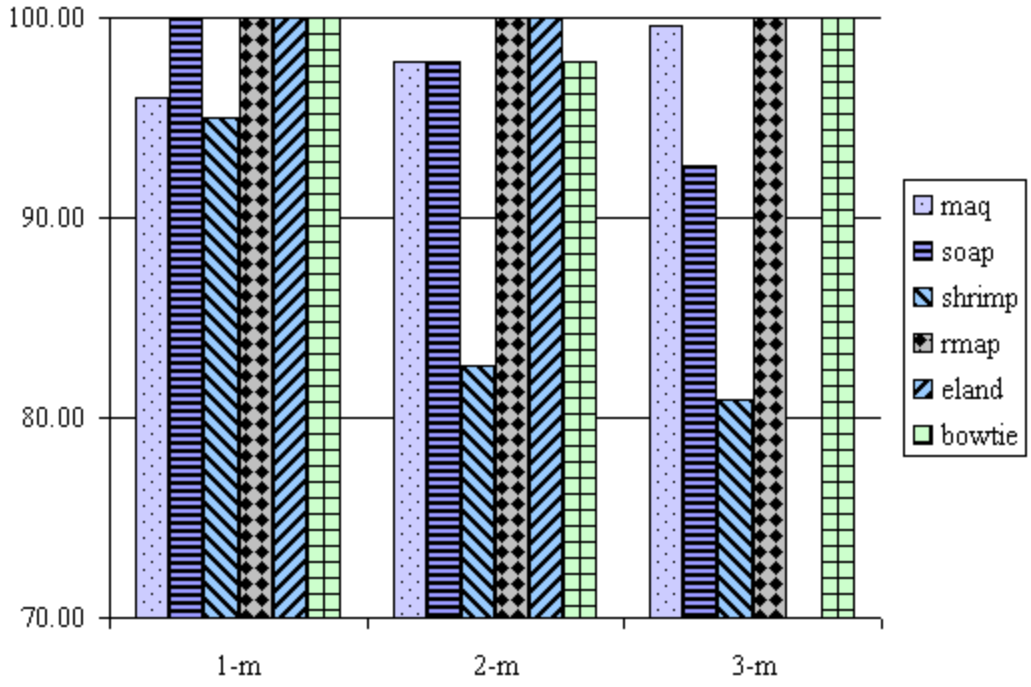

Figure 2.2: Mapping accuracy in alignments with substitutions.

The larger number of ambiguous reads identified by SHRiMP and the smaller number of mapped reads also caused the decrease of its mapping accuracy. The total number of alignments found depends on the seed width, so the mapping accuracy of both SOAP and SHRiMP could be improved at the cost of time by adjusting seed width and, for SHRiMP, the number of hits per window. Bowtie showed 97.3% of mapping accuracy in the 2-substitution data set. The mapping accuracy was improved to 100% at the cost of 40% of time increase with another set of parameters: *a*, *best* and *strata*.

Tables 2.4 and 2.5 show time and parameters in these tests respectively.

Program	1 mism	2 mism	3 mism
maq	387	875	2,492
soap	268	1,882	22,173
shrimp	6,187	17,380	40,523
rmap	238	966	7,604
eland	339	342	N/A
bowtie	103	437	1,660

Table 2.4: Time in seconds with data sets with substitutions.

Program:	1 mism	2 mism	3 mism
maq	n1	n2	n3, e90
soap	s12,v1	s10,v2	s9,v3
shrimp	s12,w31,n2,h30	s10,w31,n2,h28	s9,w31,n2,h26
rmap	h12,m1	h10,m2	h8,m3
bowtie	v1,k2	v2,k2	v3,k2

Table 2.5: Parameters used for the data sets with substitutions.

The results for time in the sets with substitutions confirm once more that SOAP is very sensitive to the seed width. With sets with 1 and 2 substitutions, when the seed width was the same one used in RMAP, SOAP and SHRiMP, SOAP was slower than MAQ, RMAP and ELAND. In the set with 3 substitutions, SOAP had greater seed width, but the program was 2.9 times slower than RMAP and 6 times slower than MAQ. The latter was the fastest program with this set of reads (ELAND does not handle 3 mismatches). Bowtie was the fastest with 1- and 3-substitution sets, and ELAND with 2-substitution set.

2.4.1.3 Data sets with indels

Figure 2.3 shows the results of tests with sets with indels. Since sets with insertions have insertions at different positions in a read, SOAP is tested with only 1 insertion set (recall that SOAP handles only single indels).

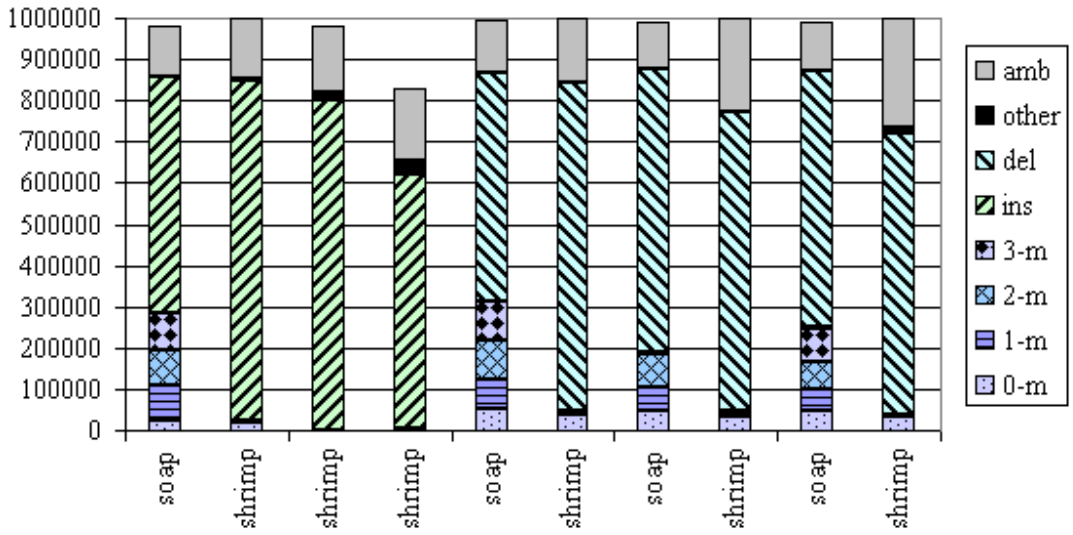

Figure 2.3: Data sets with indels.

SOAP mapped 98-99 percent of reads with indels, and SHRiMP for the same sets mapped 100 percent of the reads. SHRiMP found more alignments with insertions and with deletions in the corresponding sets. The values used for match (+1), mismatch (-1), gap opening (0) and gap continue (-1) were chosen with SHRiMP so that the alignments with *m* mismatches had the same or smaller score than alignments with *m* indels. This explains why the number of ambiguous reads was larger and the alignments with 1, 2 or 3 mismatches were absent in SHRiMP's results.

Table 2.6 and 2.7 show time in seconds and parameters for the sets with indels. Interestingly, for sets with 3 deletions, SHRiMP was faster than SOAP with the same seed size.

Program	1 ins	2 ins	3 ins	1 del	2 del	3 del
soap	1,105	N/A	N/A	1,065	15,349	16,788
shrimp	6,694	16,041	13,317	6,828	20,733	10,208

Table 2.6: Time in seconds with data sets with indels.

Progr	1 ins	2 ins	3 ins	1 del	2 del	3 del
soap	s12,v3,g1	N/A	N/A	s12,v3,g1	s10,v2,g2	s11,v3,g3
shrimp	h29,s12	h28,s10	h26,s10	h29,s12	h28,s10	h26,s11

Table 2.7: Parameters used for the data sets with indels.

2.4.2 *E. Coli* genome

The results in alignment distribution, mapping accuracy and the number of missed ambiguous reads for tests with *E. Coli* genome as a reference genome were similar to those obtained on the human X chromosome. Therefore, we omit figures and tables for the sets with substitutions and indels. Since *E. Coli* genome is much smaller, clearly the time and space results were different. In this section, we present the results for tests with different length reads and we study the sensitivity of the programs to seed width with regards to time, mapping accuracy and the total number of mapped reads.

Table 2.8 and 2.9 show space and time respectively for the tests with different read length. The most noticeable difference in space usage when human X chromosome and *E. Coli*

genome were used was observed with SOAP. The tendency of some programs to spend more time on 16 reads than 24 and 32 reads was confirmed. All programs were noticeably faster on the smaller genome of *E. Coli*.

Program	16 bases	24 bases	32 bases	40 bases
maq	468	468	468	468
soap	87	246	1307	1307
shrimp	322	902	1,032	1,154
rmap	526	526	541	541
eland	55	180	310	N/A
bowtie	18	18	18	18

Table 2.8: Space in MB in exact alignments of different length reads.

Program:	16 bases	24 bases	32 bases	40 bases
maq	92	43	42	45
soap	536	35	28	30
shrimp	119	92	128	167
rmap	25	26	27	29
eland	70	36	45	N/A
bowtie	37	40	43	47

Table 2.9: Time in seconds in exact alignments of different length reads.

We tested sensitivity of RMAP, SOAP and SHRiMP to the choice of seed width parameter with *E. Coli*. Recall that ELAND and MAQ do not provide this option.

We chose read set with 3 substitutions, changed seed and measured time, the total number of mapped reads and mapping accuracy. The results are given in Tables 2.10, 2.11 and

2.12 respectively. The results for MAQ are given for comparison (ELAND identifies up to 2 mismatches, so it was not compared here). For SHRiMP, we also measured the sensitivity to the number of hits per window.

Program	seed 10	seed 9	seed 8	MAQ
Soap	129	860	965	135
shrimp, n=1	2,533	9,430	36,060	
Rmap	51	90	246	
shrimp, n=2	270	726	2,728	

Table 2.10: Time in seconds for different width seeds.

Program	seed 10	seed 9	seed 8	MAQ
soap	939,303	969,906	969,906	1,000,000
shrimp, n=1	989,913	998,318	1,000,000	
rmap	938,891	973,933	1,000,000	
shrimp, n=2	870,744	949,273	988,320	

Table 2.11: Total number of mapped reads.

Program	seed 10	seed 9	seed 8	MAQ
soap	85.23%	92.57%	92.57%	100.00%
shrimp, n=1	93.37%	99.34%	99.95%	
rmap	85.53%	93.74%	100.00%	
shrimp, n=2	57.16%	81.56%	95.68%	

Table 2.12: Mapping accuracy.

Observe in Tables 2.10 and 2.11 that the smaller number of hits per window (*n*) for SHRiMP increased both the total number of mapped reads and the mapping accuracy at the cost of increased time. Changing of *n* from 2 to 1 caused an average increase of the total number of mapped reads by 6.6%, an average increase of the mapping accuracy by 29.8%, and an average time increase by 1,100%. If one keeps *n* fixed, but reduces the seed size, it improves results in terms of both the total number of mapped reads and the mapping accuracy. The increase of time with smaller seed was significant but not as drastic as in the case of manipulating the parameter *n.* For SOAP and RMAP the change of seed width from 10 to 8 increased time by 648% and 382% respectively with the increase of the total number of mapped reads by 3.2% and 6.5% respectively, and with an improvement of the mapping accuracy by 8.6% and 16.9% respectively. Observe that for this particular data set, SOAP showed the same results with seeds 9 and 8. MAQ also showed the best results with this data set: it mapped 100% of reads with a mapping accuracy of 100%, faster than other programs.

2.4.3 Human genome

In this experiment we tested the tools with regard to space and time requirements and with regard to overall performance in identifying the alignments with up to 2 mismatches and up to 2 bases

gap (insertions or deletions). We used real data set of 24-bases reads with the total number of 1,007,029 here. The purpose of this test was to study space and time requirements of the tools when a large genome was used as well as to study the performance of the tools on real reads in terms of the total number of found alignments. Table 2.13 shows space, time and total alignments found for each program and Table 2.14 shows the parameters used in this experiment.

Program	Space, MB	Time, sec	Total mapped reads
maq	585	24,115	937,827
soap	11,300	25,680	805,641
shrimp	570	203,336	791,012
rmap	414	19,674	794,412
eland	239	6,156	808,432
bowtie	128	838	808,866

Table 2.13: Space, time and total mapped with real human reads on hg18.

Program	Parameters
maq	n 2, e 80
soap	v 2, s 10
shrimp	n 2, w 31, o 2, s 1111111111, m 1, i –1, g –10, e –10, h 20
rmap	w 24, m 2, h 10
eland	N/a
bowtie	v 2, k2

Table 2.14: Parameters with real human reads on hg18.

MAQ found more alignments than any other program. SOAP, SHRiMP and RMAP had the seed width set to 10, which is relatively for this read length; that is why all three programs found fewer alignments than the other tools. With the same seed width, RMAP was 1.3 times faster than SOAP, but found slightly fewer alignments (1.3% fewer). Bowtie mapped about the same number of reads as ELAND, but was 7.3 times faster. ELAND showed the second best results both in time and space.

2.5 Discussion

We tested six tools for mapping sequenced reads to a reference genome on different data sets. Bowtie was the fastest tool in our experiments although its mapping accuracy was not 100% on the data set with 2 substitutions with human X chromosome. SHRiMP was the slowest tool. Among the tools that use a hash table RMAP and SOAP ran faster than ELAND and MAQ on reads with length greater than 16 and zero mismatches. RMAP was the fastest program on very short reads of length 16 with exact-matches alignment. For mappings with mismatches, RMAP and SOAP were faster with the seed width of 12 (as in the case with human simulated reads, 1 substitution set). MAQ and ELAND were faster when the seed width $s = k/(m+1)$ is less than 12 (here, k is a read length and m is the number of mismatches allowed). RMAP, SOAP and SHRiMP are quite sensitive to the seed width. SHRiMP is also sensitive to a number of hits per window. The longer is the seed (and the greater is the number of hits per window with SHRiMP), the faster the programs run (but suffer some sensitivity loss).

ELAND and MAQ require additional disk space to keep binary representation of the reference genome, and Bowtie requires disk space to keep the compressed genome. Bowtie and

ELAND use less memory than other tools in a single run. SOAP uses more memory than other tools with large genomes and relatively small data sets.

Depending on the biological task at hand, one can choose a tool among those tested to ensure the best results. If only alignments with up to two mismatches needed to be found with reads shorter than 33, then ELAND and Bowtie are the fastest. If the length of the reads is longer, then with the number of mismatches allowing the seed width of 12, RMAP and SOAP are two next fastest. MAQ maps reads faster when the seed width is less than 12. If only a single short continuous gap is allowed per alignment, then SOAP can be used with longer reads. SHRiMP undeniably takes precedence when the purpose of mapping is to study DNA variations. To speed up mapping in the study of DNA variations, one can use ELAND or Bowtie to find alignments with up to 2 mismatches fast, and then use SHRiMP to align unmapped reads.

We demonstrated that the tools tested here can be used with large genomes as well. All programs showed comparable results, with some programs showing better results in both time and mapping accuracy. ELAND and RMAP had the same mapping accuracy. SHRiMP showed the lowest mapping accuracy due to identifying alignments with trimmed reads' ends; trimming ends makes reads shorter, and shorter reads are more likely to be found at different locations in the genome.

In terms of user-friendliness, all of the programs are relatively easy to use. MAQ, ELAND and Bowtie require some preprocessing of a genome before mapping, with Bowtie taking the longest to build the index. SHRiMP requires more careful consideration for the parameter setting. SOAP provides a long list of options that give users more control and better choice for accomplishing a specific task. RMAP is probably the easiest to use.

Chapter 3

Methylation analysis

3.1 Introduction and background

Methylation of DNA is involved in a variety of biological processes, including embryogenesis and development, silencing of transposable elements, and regulation of gene transcription. In [50] the authors provide an overview of methods and techniques for identifying methylation sites in genome and discuss the advantages of new generation sequence technology that enables methylation studies with high resolution.

Some of previous methods included digestion of DNA by restriction enzymes or methylated DNA immunoprecipitation (selection of DNA's methylated locations using antibodies), both methods followed by hybridization to oligonucleotide arrays. In 1992, Frommer et al. developed a new high resolution method for methylation detection [29]. The method uses bisulfite sodium treatment of DNA that converts unmethylated cytocines to uracils and leaves methylated cytocines unchanged; followed by PCR and sequencing, bisulfite conversion allows to detect methylated cytocines with a single base resolution.

The emergence of next generation sequencing technologies and availability of several sequenced genomes enable studies of genome-wide methylation with higher accuracy than ever. Dependent on the focus of a study, scientists use different methylation detection approaches that can be broadly categorized into two groups based on whether the study focuses on specific locations within a genome or on the entire genome.

One of the technique that does not use bisulfite conversion employs restriction enzymes followed by sequencing (Methyl-seq). DNA is cut at CCGG sites by methyl-insensitive and methyl-sensitive digestion enzymes (MspI and HpaII respectively) and then selected-length DNA fragments are sequenced. The sites cut with Mspl but not with HpaII are methylated, and the sites that cut with both are partially unmethylated in a population of cells. The limitation of this approach is that it determines methylation at specific sites and misses other possible patterns of methylation.

The methods that use bisulfite conversion allow detection of methylated sites at a single base resolution. Some of them involve sequencing of entire genome and the other sequencing methylation-enriched regions only. The latter techniques first identify regions that are methylation-enriched and then apply bisulfite conversion on the selected regions. The approaches that sequence an entire genome differ in the steps of DNA preparation prior sequencing. In BS-seq, two sets of adapters are used: the first set is used prior to bisulfite conversion and the second set is used prior to sequencing. The steps of size selection and PCR are repeated twice with this method. In the MehylC-seq, purified DNA is fragmented, then selected fragments are ligated to methylated adaptors, amplified with PCR and finally sequenced. In a reduced representation bisulfite sequencing method (RRBS), DNA is digested with methyl-insensitive enzyme (Mspl) then selected-length DNA fragments are ligated to methylated adapters, bisulfite converted, amplified with PCR and sequenced. This method is biased to CpG regions.

The combination of bisulfite conversion and next generation sequencing has already enabled some genome-wide studies of DNA methylation [12, 51, 52]. The success of these methods critically depends on the availability of accurate and time-efficient tools capable of mapping hundreds of millions of bisulfite treated (*BS-treated*) short reads to a reference genome.

This latter task, called *BS-mapping*, can be computationally intensive. Due to the effect of the bisulfite conversion, BS-mapping must allow Ts in the sequenced reads to align to Cs in the reference genome and similarly As in the reads to align to Gs in the genome. Hereafter, these types of T-C and A-G allowed mismatches are called *BS-mismatches*. In order to allow for BS-mismatches during the mapping, one can (1) allow a large number of mismatches, about 1/4 of the read length assuming that methylation is rare; (2) use an exhaustive search where for each read all possible combinations of Ts are converted to Cs; or (3) apply different kinds of reference/reads conversions, usually involving the reduction of the alphabet cardinality.

Allowing a large number of mismatches introduces many false positives due to non-BS-mismatches and can be very computationally expensive, which makes this strategy impractical. Similarly, the second option generates a very large number of candidates and presents similar problems. The conversion of a genome and/or reads has been shown to be a successful strategy. For instance, in [51] the authors mapped sequenced reads to three versions of the genome: the original genome, the genome in which Cs are replaced with Ts, and finally the genome in which Gs are changed to As. Reads were allowed up to two mismatches to capture methylated Cs. The shortcoming of this method is that it does not handle instances where a read contains both unmethylated and methylated Cs with the number of Cs higher than the number of allowed mismatches. Another strategy was proposed in [18], where the reads are transformed in position-weight matrices and alignment is carried out in probability space. Due to its computational complexity, the authors suggest that their approach is not practical unless the reference genome is small.

To meet these challenges several BS-mapping tools have been designed such as mrsFAST [36], BSMAP [88], VerJInxer [92] and RMAP-bs [76]. The description of the algorithm used in mrsFAST currently is not publicly available. RMAP-bs uses hashing on the

reads and employs wildcard matching to allow BS-mismatches. VerJinxer uses q-grams that simulate all possible methylation patterns. BSMAP uses hashing on the reference genome, where seeds are words of a fixed length expanded to account for all possible combinations of substitutions Cs to Ts. These latter two approaches can be very slow due to the large search space induced by the additional seeds.

While the mapping method plays an important role, increasing the read length and employing paired-end sequencing further improves the number of uniquely mapped reads [50]. To accommodate users who prefer paired-end sequencing, we have developed a new time efficient BS-mapping tool called Bisulfite-treated Reads Analysis Tool (BRAT). Our tool supports single and paired-end short reads. BRAT uses a specially designed binary representation of the reference genome and reads that allows for BS-mismatches without affecting the search space. Our tool seamlessly handles input files containing reads/mates of various lengths aligning all the bases of the reads/mates. Experimental results show that (1) on paired-end reads, our tool is much faster, maps more unique pairs and has higher mapping accuracy than BSMAP and mrsFAST, and (2) on single reads, BRAT's performance is comparable to the performance of RMAP-bs, which to our knowledge is currently the best BS-mapping tool for single reads.

3.2 Basic concepts and notations

As said, the 'gold-standard' method to study genome-wide DNA methylation takes advantage of the effect of sodium bisulfite (BS) conversion on DNA. After BS treatment, several steps of PCR amplification are applied, and the resulting DNA is sequenced. Figure 3.1 illustrates the effect of BS conversion and subsequent PCR amplification. The protocol for next-generation sequencing instruments (e.g., Illumina Genome Analyzer) requires adding special methylated adaptors before

BS treatment. The presence of the adaptors controls which DNA strands are sequenced: although there are four strands of PCR product (PCR1+, PCR1–, PCR2+ and PCR2− in Figure 3.1), the actual sequencing is carried out only for PCR1+ and PCR2−, which are the original genomic strands. Methylated Cs in the genome remain Cs after BS conversion, while unmethylated Cs are transformed. Unmethylated Cs in the positive strand of the genome (DS+) are converted into Ts in PCR1+ and stay Cs in PCR2+. The same is true for unmethylated Cs in DS–, which stay Cs in PCR1– and turn into Ts in PCR2–.

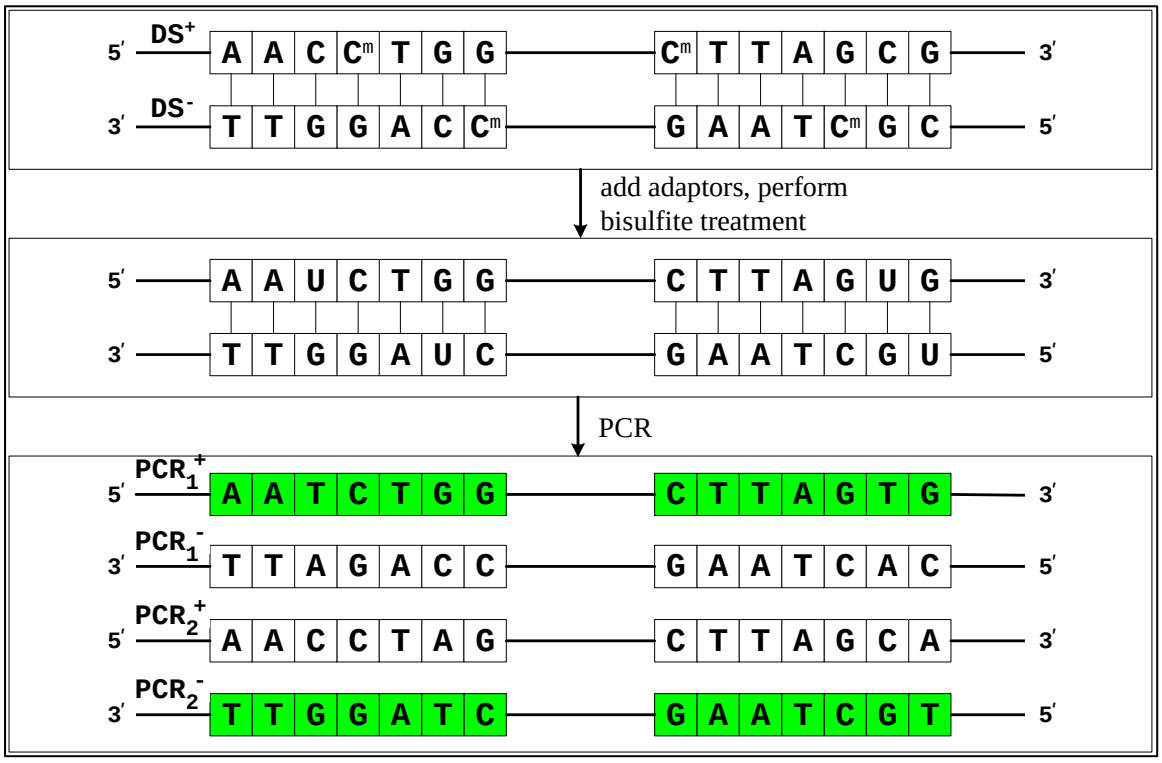

Figure 3.1: The effect of BS-conversion followed by PCR.

Figure 3.2 illustrates BS sequencing (following PCR) and BS-mapping. The order of mates in paired-end reads is essential for BS-mapping. When the 5' mate maps to the positive strand of the genome (DS+), then only T-C mismatches are legal. If the reverse-complement of the 5' mate maps to DS+, then only A-G mismatches can be allowed. Similarly, if the 3' mate maps to DS+, then only A-G mismatches are legal, and if the reverse-complement of the 3' mate maps to DS+ then only T-C mismatches can be allowed. Similar rules apply to single reads sequencing: these reads must follow the same rules as for 5' mate for paired-ends (in Figure 3.2, the 5' mate is called read 1 and the 3' mate is called read 2).

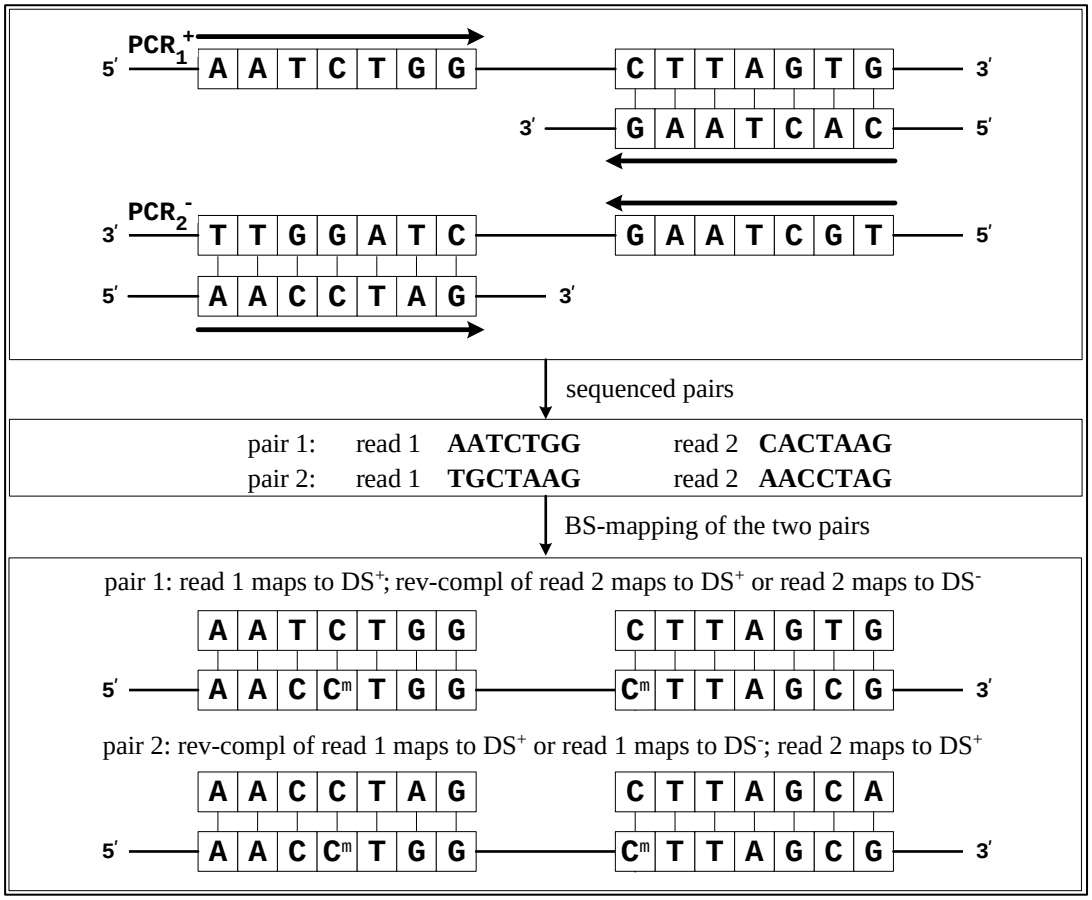

Figure 3.2: BS-sequencing and BS-mapping.

BS-mapping refers to the computational process of mapping short reads obtained after bisulfite treatment to a reference genome. Due to the effect of BS conversion, out of the sixteen possible mappings between a symbol in a read and a symbol in a reference genome we allow six, namely A-A, A-G, C-C, G-G, T-C and T-T. Mappings A-G and T-C are called *BS-mismatches*. The other ten mappings are considered true mismatches (hereafter they are called *non-BS-mismatches)*.

We say that a paired-end (or single) read is *unique* if it maps with any number of BS-mismatches (including zero) to a unique location in the reference genome. If it maps to multiple locations under the same conditions, the read is called *ambiguous*. When we allow non-BS-mismatches, a single read is *unique* if it maps to a unique location with the smallest number of non-BS-mismatches (a BS-mismatch is equivalent to zero non-BS-mismatches); a paired-end read is *unique* if it maps to a unique location with the smallest number of non-BS-mismatches in both mates. If it maps to multiple locations (under the same conditions), that read/pair is called *ambiguous*. We define the *mapping accuracy* as the ratio between the number of correctly mapped unique pairs/reads and the total number of mapped unique pairs/reads.

Let us illustrate the definition of uniqueness with a few examples. Consider a single read that maps with one BS-mismatch to one location in the reference genome and with two BS-mismatches to another location. This read is ambiguous because it maps to more than one location with the allowed BS-mismatches. Let us now consider a single read that maps to one location with three BS-mismatches and to two additional locations with one non-BS-mismatch. This read is unique because it maps with the smallest number of non-BS-mismatches (zero in this case) to a single location. Finally, let us consider an example with paired-end reads. Assume that a paired-end read maps to two distinct locations: in the first, the mapping has two BS-mismatches for the left mate and five BS-mismatches for the right mate; in the second location, the mapping

is a perfect match for the left mate and has one non-BS-mismatch for the right mate: in this case the smallest number of non-BS-mismatches is zero (mapping to the first location has only BS-mismatches that are counted as zero non-BS-mismatches) and the mapping is unique.

We define a *k-mer* as any word (or substring) of *k* consecutive bases obtained from the reference genome.

3.3 BRAT: the algorithm

Each read, its reverse complement and the reference genome are represented in BRAT according to their *ta*- and *cg*- binary representations. In the *ta*-representation, all Cs and Ts are converted to ones, and As and Gs are converted to zeroes, which can be interpreted that all Cs are converted to Ts and, similarly, that all Gs are converted to As. In other words, the *ta*-representation is the reduced genome over two letters, namely T and A, which is reflected in the chosen name. In the *cg*-representation, Cs and Gs are converted to ones, and As and Ts are converted to zeroes. This representation is solely used to verify C-C and G-G mappings described below, and its name reflects this functional role. In both cases, each base is represented by one bit. Hereafter, we call these two induced representations for the reference genome as the *ta-genome* and the *cg-genome.*

During the preprocessing step, BRAT builds a hash table on the reference genome. The algorithm first identifies the shortest read in the input reads (let *w* be its length). The hashing function uses *w* consecutive bases of the forward strand of the *ta*-genome and the *cg*-genome as *seeds* to calculate the corresponding key for the entry in the hash table. The entry is updated with the corresponding reference name and position within the reference where the seeds come from. Similarly, each read of length $k \geq w$ is hashed on the first *w* bases to find the corresponding key for that read. We used variants of a hashing function designed by Bob Jenkins (see

http://burtleburtle.net/bob/c/lookup8.c). There are four hashing functions used in BRAT: one for normal mapping, one for BS-mapping with large genomes and two for BS-mapping with small genomes (details of the hashing functions are given in Figure 3.3).

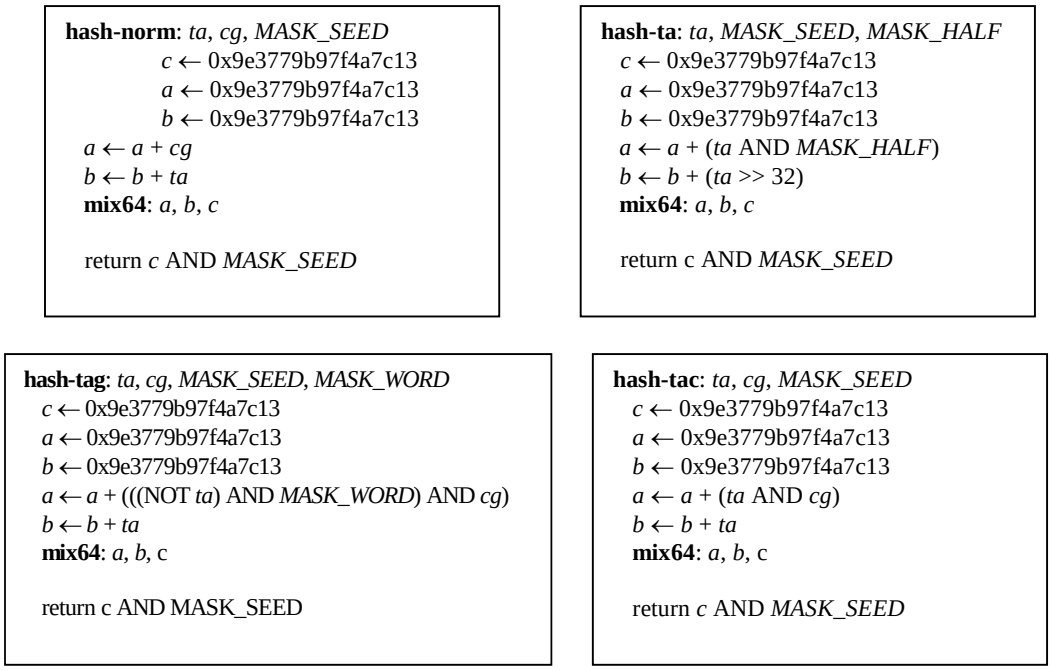

Figure 3.3: Hashing functions used in mapping. BRAT uses *hash-norm* for normal mapping, and *hash-ta*, *hash-tac* and *hash-tag* for BS-mapping. The subrouting *mix64* is designed by Bob Jenkins. All variables here are 64-bit unsigned long integers. MASK_SEED masks 24 least significant bits with ones, MASK_HALF masks 32 least significant bits with ones, MASK_WORD masks *w* consecutive bits with ones (*w* is the length of the shortest read), and *ta* and *cg* are seeds of *ta*- and *cg*- binary representations of reads and genome *k*-mers. AND here is a logic bit wise operation, and (*x* >> *y*) is a shift of a binary *x y* bits to the right.

After the entry corresponding to a read has been identified, that read is aligned to all *k*-mers of the genome whose starting positions are stored at this entry (hereafter, in order to

distinguish between hashing and mapping, we use *w* for hashing, and *k* for mapping, where *w* is the length of the shortest read and *k* is the length of a read with *k* greater than or equal to *w*).

The alignment process is carried out on the binary representations in three steps: *ta*-check, *cg*-check and final verification. Figure 3.4 depicts ta- and cg-check steps of the BS mapping.

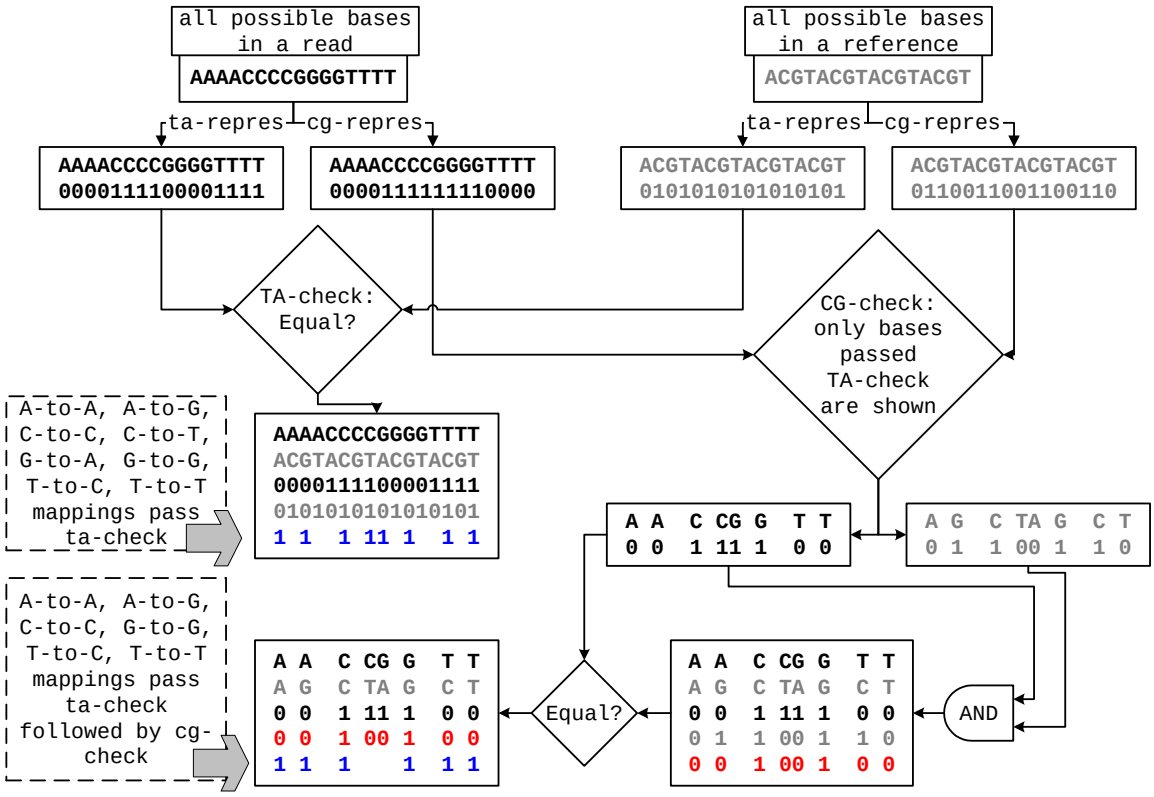

Figure 3.4: BS-mapping induces only normal matches and BS-mismatches.

First, we describe the procedure for zero non-BS-mismatches. We use *ta*-check to verify that Ts in sequenced reads map only to Ts in the *ta*-genome and similarly that As in reads map to As in *ta*-genome; the corresponding mappings on the original genome are: T-C, T-T, A-G and A-

A (which are allowable mappings) and C-T and G-A (which are erroneous mappings that should not be allowed, but the *cg*-check takes care of these). During the *ta*-check, the *ta*-representation of a read (hereafter called *ta*-read) is compared with *k*-mer from the *ta*-genome: if the bits in the *ta*-read and the *k*-mer are equal, we proceed to the next step. We use *cg*-check to ensure that Cs in sequenced reads map only to Cs in the reference genome and similarly that Gs in reads map exclusively to Gs in the reference. This is done by checking the Boolean condition [*cg*-read == (*cg*-read AND (*k*-mer in the *cg*-genome))], where AND is the normal bitwise-AND operation. Figure 4 demonstrates that *ta*-check followed by *cg*-check induces only BS-mismatches and perfect matches. The two input strings are chosen so that all possible combinations of alignments of bases in a read to bases in a reference are obtained. In the final check, we verify the correct combination of a read orientation and BS-mismatches types using the rules described in Section 3.2 and illustrated in Figure 3.2.

Next we describe the mapping procedure when the number of non-BS-mismatches is greater than zero. The mapping procedure when the number of non-BS-mismatches is greater than zero is similar to the one for zero non-BS-mismatches. The difference lies in using XOR instead of an equality operation in *ta*-check and *cg*-check (where XOR is the normal bitwise exclusive-or operation). The number of non-BS-mismatches is counted as the number of bits set to 1 by XOR. To allow for non-BS-mismatches, BRAT as RMAP-bs uses spaced (or masked) seeds. We still use seeds of *w* consecutive bases, but mask some of the bases with zeros to allow for non-BS-mismatches. BRAT uses a total of four seeds: the first seed is for the reads that could be mapped with zero non-BS-mismatches (these reads are not considered in further mapping steps) and the other three seeds are used to find all mappings with one non-BS-mismatch. To allow one non-BS-mismatch, it would be sufficient to use just two masked (or spaced) seeds, but

using three masked seeds retain more bit information for hashing. Examples of seeds are given in Figure 3.5.

```
00000000000000000000000000000000111111111111111111111111111111
00000000000000000000000000000000000000000011111111111111111111
00000000000000000000000000000000111111111111111111110000000000
00000000000000000000000000000000111111111000000000001111111111
```

```
00000000000000001111111111111111111111111111111111111111111111
00000000000000000000000000000000111111111111111111111111111111
00000000000000001111111111111111111100000000000000000000000000
```

Figure 3.5: Example of seeds used in mapping.

If the shortest read has length *w*, a third of *w* is taken as a measure by which we subdivide a seed (for paired-end reads of length greater or equal to 48 bases, a half of *w* is used). The figure shows examples of seeds for $w = 32$ and $w = 49$ (bits with values 1 here correspond to the bits not masked with zeros, i.e. bits whose values could be 0 or 1 in *ta*- or *cg*- binary representations). The first set of seeds is used with single-end reads and with each mate of paired-end reads of length less than 48 bases, and the second set of seeds with each mate of paired-end reads of length greater than or equal to 48 bases. The first seed finds all alignments with zero non-BS-mismatches, and the rest of seeds guarantee to find all alignments with 1 non-BS-mismatch. To guarantee to find all alignments with one non-BS-mismatch in each mate of a pair, the mapping runs in rounds, and two seeds for each pair are used (one seed for one mate and another seed for another mate). There are total of 10 pairs of seeds for paired-end alignment of reads with length less than 48 bases and total of 5 pairs of seeds for reads of length greater than or equal to 48 bases

(one pair for zero non-BS-mismatches: both seeds are the first seeds in the sets of Figure 3.5; the rest of pairs are the combinations of the two seeds, each of which is chosen from the rest of the seeds shown on Figure 3.5).

3.4 BRAT: the software suite

We offer users two versions of BRAT, namely BRAT and BRAT-large. Both programs run on 64-bit architecture under Linux: we made sure that they can compile on the five major Linux variants, namely Ubuntu, CentOS (RedHat), Debian, Suse, and Fedora. Input reads have to be at least 24 bases long. There is no upper limit on reads lengths. Users can specify any number of non-BS-mismatches, but BRAT guarantees to find all read-genome mappings with up to one non-BS-mismatch in the first 24 bases of reads. BRAT-large guarantees to find all read-genome mappings as long as the first 24 bases of each read can be mapped either perfectly or with any number of BS-mismatches.

BRAT accepts up to 65,536 reference sequences (genomes/chromosomes), where the maximum size of the cumulative reference sequence is 4.2GB. BRAT-large also allows up to 65,536 references where the size of the largest reference sequence is 4.2GB.

BRAT works best with relatively small genomes because it requires significantly more memory than BRAT-large. However, BRAT is faster than BRAT-large because the latter uses only one hashing function and maps the reads to the reference genome sequentially: first it maps all reads to the first reference, and then all reads to the second reference and so on. The memory space used by BRAT-large depends on the size of the largest reference in the set of input references, whereas the space used by BRAT depends on the total size of all input references (measured in base pairs). If we define N to be the total size of a genome in base pairs, P to be the

size of the largest reference in base pairs and R to be the total number of reads (each read is counted, *i.e.* a pair has 2 reads), the space (Bytes) required by the two programs is bounded by

$$\text{Space}_{\text{BRAT}} = 269 \cdot 10^6 + (2 \cdot 4 + 3/8)N + 25R$$

$$\text{Space}_{\text{BRAT-large}} = 269 \cdot 10^6 + (2 \cdot 4 + 3/8)P + 25R \quad \text{(option } S\text{)}$$

$$\text{Space}_{\text{BRAT-large}} = 269 \cdot 10^6 + (4 + 3/8)P + 25R$$

Where 269MB accounts for the space required supporting the data structure for the hash table. A total of 4 bytes is used to store a reference position in the hash table for BRAT and BRAT-large. In BS-mapping when non-BS-mismatches are not allowed, BRAT uses two different hashing functions: one for a read and one for its reverse-complement; therefore, each genome position can be hashed to at most two different entries of the hash table. With BS-mapping and non-BS-mismatches, BRAT uses a single hashing function (*hash-ta* shown in Figure 3), so each genome position can be hashed to at most two different entries of the hash table (one for each mate).

When the memory available is limited, a user has a choice to use BRAT-large. For example, with paired-end BS-mapping and non-BS-mismatches, BRAT uses 2.5GB of memory to index human chromosome 1, whereas BRAT-large needs just 1.7GB (or 2.7GB when option *S* is used) of memory for the entire human genome.

The package includes two additional tools: trim and acgt-count. The tool trim accepts FASTQ files with reads/pairs as input and trims the ends of reads whose base quality scores are lower than user specified threshold and finally filters reads/pairs, which contain internal Ns. The other tool, acgt-count, aligns mapped reads to the genome and counts the number of occurrences of A, C, G and T at each base in the forward and the reverse strands separately.

3.5 Benchmarking experiments

We have compared our tool with RMAP-bs, mrsFAST and BSMAP using real BS-treated reads on *P. falciparum* obtained with Illumina GAII and *in silico* reads on *H. sapiens* and *P. falciparum*. *H. sapiens* has long CpG islands whereas *P. falciparum* is AT-rich.

All tests were carried out on a cluster of 4x Quadcore Intel Xeon CPU running at 2.40GHz (16 CPU cores), with a total of 64GB of RAM. We used *P. falciparum*'s genome version PlasmoDB 6.0 (downloaded from http://plasmodb.org/plasmo/) and *H. sapiens*, hg18 (http://hgdownload.cse.ucsc.edu/goldenPath/hg18/bigZips/).

Our first experiment tested tools on real and *in silico* data when only perfect matches and BS-mismatches were allowed. Table 3.1 reports the results of this experiment. Our real dataset contains 21.5M reads, whereas for the simulation we generated 1M and 10M randomly chosen pairs/reads with 90% of Cs converted to Ts (no sequencing errors were introduced for this experiment). The parameters used in the experiments were RMAP-bs (m 0, S 1, h 26), BSMAP (s 9, v 0, r 0, m 106, x 306), and mrsFAST (e 0, n 2, min 106, max 306). For BSMAP, we used the largest seed allowed by the program (parameter *s*).

		Genome, read length, and number of reads/pairs	Time	RAM (MB)	Total mapped unique reads/pairs	Correctly mapped unique reads/pairs
single-read	RMAP	*P.falciparum*, 26bp, 21.5M	8m3s	1,500	7,413,261	n/a
	BRAT		1m59s	982	7,379,870	n/a
	RMAP	*H. sapiens*, chr X, 32bp, 10M	4m52s	2,100	7,906,395	7,906,395
	BRAT		6m28s	2,000	7,915,050	7,915,050
paired-end	BSMAP	*P.falciparum*, 32bp, 1M	1160m0s	171	402,602	393,810
	BRAT		0m40s	982	913,225	913,225
	mrsFAST		48m10s	687	635,784	620,622

Table 3.1: Performance and sensitivity on BS-mapping with exact matches.

With single reads, both RMAP-bs and BRAT had 100% mapping accuracy. The mapping accuracy is calculated as the ratio between unique reads/pairs mapped correctly and total number of unique reads/pairs, where unique reads/pairs are reads/pairs that are mapped perfectly or with BS-mismatches to a single location. There is a slight difference in the number of mapped reads because RMAP-bs, in addition to BS-mismatches, allows a C in the reads to align to a T in a genome only when C is followed by a G. On paired-end reads, BRAT mapped 1.47 and 2.3 times more unique pairs (correctly) than mrsFAST and BSMAP respectively while retaining higher accuracy: BRAT had a mapping accuracy of 100%, whereas mrsFAST was 97.6% and BSMAP was 97.81%.

To compare our tool with the better performing tool for paired-end reads (mrsFAST) in the presence of sequencing errors, we used *in silico* 1M paired-end 24 bases reads and 64 bases reads from *P. falciparum* with 90% of BS-conversion and 1% of sequencing errors. Sequencing

errors were introduced at random positions: first a read was chosen randomly, then a base within the read was chosen uniformly at random, and finally, the chosen base was substituted by a letter randomly chosen out of the three remaining letters. The parameters used for this experiments were: mrsFAST (*n* 2; *e*: 0, 1, 2; *min* 136; *max* 324) and BRAT (*i* 113; *a* 301; *m*: 0, 1, 2). The minimum and maximum values for insert size were taken different than the corresponding values for generated *in silico* pairs (mrsFAST: *min* and *max* and BRAT: *i* and *a*). Here the values for min/max insert size for both programs are different due to different definitions for these values used by the programs: BRAT counts an insert size as the distance between the leftmost points of the alignments of the two mates on the forward strand, and mrsFAST as the distance between the outmost end points of the alignments of the two mates. The options *e* and *m* set the number of non-BS-mismatches in mrsFAST and BRAT respectively. The two programs differ in the way they output the results: BRAT outputs only unique alignments and mrsFAST outputs a user-specified number of alignments (parameter *n*). Therefore, to distinguish between unique and ambiguous reads/pairs, users must use the option *n* in mrsFAST. Unfortunately, this option does not guarantee to find necessarily unique reads or the best alignments. For example, we found cases when mrsFAST reported mapped locations for pairs with a larger number of non-BS-mismatches than another location that was missed (in this case, a read is counted as ambiguous even though it is unique). Another source of poorer mapping accuracy in mrsFAST is due to missing ambiguous reads, i.e., reads for which only one result is found (with *n* 2) even though there exists an equally good alignment somewhere else in the genome. We computed the number of correctly mapped unique reads for mrsFAST by excluding all the ambiguous reads as identified by mrsFAST (occurring twice in the output) and all ambiguous identified by BRAT (occurring in at least two valid and equally good alignments). For total unique pairs, we counted

the reads occurring once in the resulting output. Figure 3.6 shows the number of correctly mapped unique pairs (bars) as well as mapping accuracy of both tools (lines).

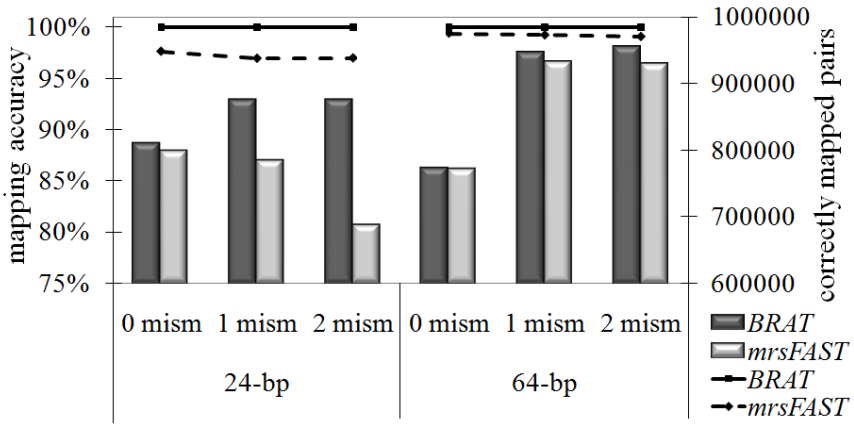

Figure 3.6: BRAT vs. mrsFAST.

When mapping with non-BS-mismatches, we define a pair to be unique if it maps to a single location with the smallest number of non-BS-mismatches in both mates. BRAT mapped up to 21% more unique pairs than mrsFAST on 24 bases reads. In both experiments, BRAT had higher mapping accuracy. BRAT was also significantly faster than mrsFAST: on 24 bases reads, BRAT was 67, 12 and 18 times faster with 0, 1 and 2 mismatches respectively and on 64 bases reads it was 55, 20 and 37 times faster with 0, 1 and 2 mismatches respectively.

The parameters used with mrsFAST were taken from the tool's description on-line and insert size between mates of a pair was deduced from output file. This latter parameter turned out to be not optimal, and when using optimal insert size, mrsFAST had slightly improved performance both in time efficiency and mapping accuracy.

Chapter 4

Dynamic chromatin remodeling in *P.falciparum*

4.1 Introduction

In eukaryotic cells, DNA is packed into chromatin with a nucleosome being a building unit block of chromatin structure. In order for transcription to take place, DNA has to be accessible for various transcription factors to bind to it and initiate the transcription. In classical eukaryotic model, chromatin reorganizes within promoters immediately prior transcription: a few nucleosomes either slide or completely disassemble to make DNA accessible for the transcription machinery. In case of *P. falciparum*, the role of chromatin in transcription regulation was not clearly understood. In this chapter, we investigated dynamic chromatin remodeling during the parasite' erythrocytic cycle in order to gain insights in regulation of transcription in *P. falciparum*.

In order to infer the positions of nucleosomes in malaria, two complementary assays were employed, namely Formaldehyde-Assisted Isolation of Regulatory Elements (FAIRE) and Micrococcal Nuclease (MNase) digestion. FAIRE isolates protein/nucleosome-depleted DNA [32], whereas MNase digestion isolates nucleosomes-enriched DNA from the chromatin. In order to study the dynamic landscape of nucleosomes, fourteen distinct libraries were produced for sequencing: seven libraries for DNA obtained with FAIRE at time points T = 0, 6, 12, 18, 24, 30, and 36 hours, corresponding to early ring, late ring, early trophozoite, late trophozoite, early schizont, late schizont and merozoite stages during the malaria erythrocytic cycle, respectively,

and seven libraries obtained after MNase digestion (at same time points). The fourteen libraries were sequenced using the Illumina 1G Genome Analyzer. Each time point was assigned to a lane of the Analyzer, which produced about 10 million short reads per lane. We processed sequenced reads as follows. First, we trimmed a base from each end of the reads to decrease the chance of having bases from adaptors used during sequencing, and then we trimmed the ends of the reads based on quality scores. Then, we mapped the trimmed reads to the genome of *P. falciparum* strain 3D7 v5.5. Illumina's ELAND was used to map reads of lengths 18 to 32 bases, whereas RMAP [76] was employed to process reads of lengths 33 and 34 bases. Sequenced reads giving a perfect match at a unique location in the reference genome were immediately mapped to that location. The remaining reads were mapped allowing up to two mismatches, but only if their chastity score was at least 0.6. Remaining reads were aligned with the human genome to measure the degree of contamination with human DNA, which is unavoidable given that *P. falciparum* is cultured in human blood. The raw coverage was normalized per million mapped reads and with the percentage of the genome covered by reads. In order to remove high-frequency noise, the normalized raw coverage distribution of mapped reads of each chromosome and each strand was smoothed using a kernel density estimation method [61]. To obtain the overall smoothed coverage distribution (or *coverage profile*) the smoothed coverage for the positive and negative strands was added at each position [81]. For each time point the total number of reads that mapped to the genome and the percentage of nucleotides actually covered by at least one read was computed. Smoothed coverage profile was used to determine nucleosome positioning, and raw normalized data was employed for the rest of our analysis. We compared the genome-wide distributions of mapped reads obtained with FAIRE-seq to the ones measured with MNase-seq. We computed Pearson's product-moment coefficient between corresponding profiles of coverage

across the seven investigated time points. The average correlation coefficient is strongly negative ($R^2 = -0.70$), indicating excellent complementarity between FAIRE-seq and MNase-seq.

4.2 Methods

4.2.1 Nucleosomes positioning

The positions of nucleosomes were determined from the coverage profiles obtained in the MAINE-seq procedure, as follows. First, regions of coverage were defined as the continuous spans of the genome where coverage was at least one. Then each such region of coverage was processed individually: positive and negative strands were scanned for local maxima using a sliding window. The size of the sliding window was empirically determined to be 15 bases. For each local maximum found on the positive strand, the negative strand was scanned for a corresponding negative maximum within a distance range determined empirically according to the sizes of the libraries and the sizes of the sequenced reads. The chosen distance range was [-15, 180]. Half of the distance between the two corresponding peaks was defined as the shift. The center of a nucleosome was placed in the middle between the two corresponding positive and negative peaks, and the left and right boundaries were set at distance 146/2 = 73 bp around the center. In some cases, a local maximum was found only on one strand, with no matching peak on the other. Visual inspection of the dataset revealed that most of the time the missing matching peak was actually present as a shoulder of a neighboring peak (and not as a local maximum). In these cases a nucleosome was placed according to a single detected peak, using the shift calculated for the nearest nucleosome. The center of a nucleosome was placed at the position of the local positive peak plus a shift if only positive peak was detected and at the position of the local negative peak minus a shift if only negative peak was detected. Each nucleosome was

assigned a confidence score equal to the sum of the smoothed normalized coverage in a region of 147 nucleotides centered at the assigned nucleosome location.

4.2.2 Measurements for modification of chromatin structure

The modification of chromatin structure was measured by the depth of FAIRE and MAINE coverage. For the analysis of correlation between the dynamic chromatin structure remodeling and gene expression profiles, we used 1000 bases 5' upstream of the genes start codons (promoter regions), 1000 bases downstream of the genes stop codons (downstream regions), and 600 bases inside the genes from 5' to 3' downstream from the start codons and upstream from the stop codons. Another measurement of chromatin structure was the degree of chromatin opening calculated as log base two of the ratio between FAIRE coverage and MAINE coverage at each base of the genome.

4.2.3 Finding the centromeres

Chromosomes were scanned to detect putative centromeres. Scanning parameters were empirically determined using the known centromeric regions (www.plasmoDB.org v.5.5). For both FAIREseq and MAINE-seq datasets (time points combined together), the average number of (A+T) and the average number of (C+G) per base were calculated for all known centromeric region. Then, chromosomes were scanned for 2 kb-long regions with (A+T) and (C+G) content within the range [average – 5*SD, average +5*SD] for both FAIRE-seq and MAINE-seq. For each time point, consecutive windows were merged into a single contig corresponding to putative centromeric region.

4.3 Results and discussion

4.3.1 Global chromatin organization

The chromatin structure of *P. falciparum* undergoes drastic changes throughout the erythrocytic cycle. It achieves its most open state with the most enriched FAIRE coverage and least pronounced MAINE coverage at T18 time point, which corresponds to late throphozoite stage. The chromatin is mostly closed (packed into nucleosomes) at T36, late schizont stage. The percentage of the genome covered with nucleosomes at T18 and T36 respectively was 26% and 50%. We observed that with *P. falciparum*, nucleosomes mostly located inside genes, and the promoters and downstream of the genes are predominantly free from nucleosomes for the longest duration of the erythrocytic cycle. Figure 4.1 shows the degree of chromatin opening T0, T18 and T36 time points around the start and stop codons for all genes.

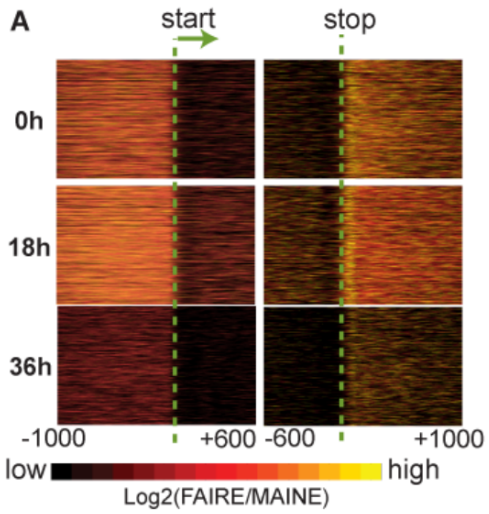

Figure 4.1: The degree of chromatin opening around start and stop codons.

Observe in Figure 4.1, that regions inside genes are predominantly covered with nucleosomes, whereas promoters and downstream of the genes are free from nucleosomes at T0 and T18. However, at T36 these regions are packed into nucleosomes, ready for the next erythrocytic cycle. Table 4.1 shows the total number of nucleosomes, percentage of genome covered with nucleosomes and average raw scores of the nucleosomes for seven time points across the erythrocytic cycle. The genome is most covered by nucleosomes at T36, and the least covered at T18. The total number of nucleosomes also follows the same pattern: the least number of nucleosomes was observed at T18 and the largest number at T36.

Time point	Total number of nucleosomes	Percentage of genome covered	Average score
0	62,092	35%	718.93
6	53,365	31%	807.58
12	75,156	43%	1221.65
18	45,384	26%	748.33
24	62,287	36%	1235.84
30	72,890	42%	1030.19
36	89,115	50%	1221.69

Table 4.1: Nucleosomes statistics across the erythrocytic cycle.

The centromeric regions showed distinct pattern of coverage with very low FAIRE coverage and almost no MAINE coverage. We used FAIRE and MAINE coverage to identify putative centromeres for chromosomes 9, 11 and 12, for which centromeres are not identified to date. Using a sliding window as described in Section 4.2.3, we were able to detect all existing centromeres and to discover putative centromeric regions ranging from 2 to 3 kb (Table 4.2).

Chromosome	Previously identified loci		Loci identified de novo	
	Start	Stop	Start	Stop
MAL1	459,080	461,461	458,604	461,814
MAL2	447,300	450,455	447,229	450,490
MAL3	594,492	596,951	594,410	597,532
MAL4	648,984	651,511	648,534	651,840
MAL5	455,737	457,250	454,956	458,019
MAL6	478,646	480,966	478,217	481,309
MAL7	864,620	867,966	864,368	867,647
MAL8	300,194	302,519	300,048	303,299
MAL9	1,242,126	1,244,473	1,241,760	1,244,815
MAL10	undetermined	undetermined	70,843	72,932
MAL11	undetermined	undetermined	831,726	834,921
MAL12	1,282,761	1,285,056	1,281,951	1,285,347
MAL13	1,168,327	1,170,626	1,167,891	1,170,876
MAL14	undetermined	undetermined	1,071,399	1,074,951

Table 4.2: Centromeres *loci* identified *de novo.*

In Table 4.2, “Start” and “Stop” refer to the start and end positions of known and putative centromeres.

4.3.2 Analysis of dynamic chromatin structure remodeling

We used FAIRE and MAINE coverage to analyze whether there is correlation between dynamic chromatin structure remodeling and gene expression in *P. falciparum*. As said, genome-wide gene expression analysis has shown remarkable variations of mRNA levels with a tight regulation throughout the parasite life cycle. We used mRNA steady state levels measured at seven time points throughout the erythrocytic cycle of the malaria parasite reported in [46] for this analysis. Recall that DNA libraries for FAIRE and MAINE were collected for the same time points;

therefore, the comparison between various measurements obtained using FAIRE and MAINE coverage and gene expression data was consistent. In what follows, we describe different techniques that we used to determine whether dynamic chromatin structure remodeling is correlated with gene expression in the human malaria parasite.

In [46], a total of 2235 genes of *P. falciparum* that showed significant change of mRNA steady state levels throughout the erythrocytic cycle were clustered into fifteen clusters. Our first approach was to cluster genes using approximately the same number of clusters (from 10 to 20) either by FAIRE or MAINE coverage and to study whether these latter clusters were similar to gene expression clusters. We used the *Jaccard* coefficient as a metric of similarity between two clusters. Given two clusters of genes, the Jaccard coefficient is defined as the ratio between the number of genes shared by the clusters and the total number of distinct genes in the clusters. To cluster genes by coverage, we considered two sets of DNA regions: the first set included regions of promoters and portion of each genes and the second set included exclusively promoters. We defined a promoter as 1000 bases upstream of the genes start (ATG). When we considered gene beginning, we took 600 bases starting at the first codon. We clustered genes using raw normalized FAIRE and MAINE coverage of the chosen regions. The Jaccard pairwise coefficients between clusters by coverage (either FAIRE or MAINE) and clusters by gene expression profiles below 0.5 indicated low similarity. In this analysis, no similar clusters were identified.

Next, the nucleosome scores of the two first nucleosomes inside each gene were added at each time point resulting in seven values for each gene. The Pearson coefficients between these values and corresponding values of mRNA steady state level for gene expression showed good correlation only for a few genes. The vast majority of the genes had Pearson coefficients below 0.5.

Next, we observed that the majority of the genes had a single highly enriched peak within 1000 bases upstream of the genes start with FAIRE coverage. We first studied whether these peak s change their locations relative to the genes start across time points throughout the erythrocytic cycle. Figure 4.2 shows that although the intensity of the most enriched peaks change across time points, the positions are relatively stable.

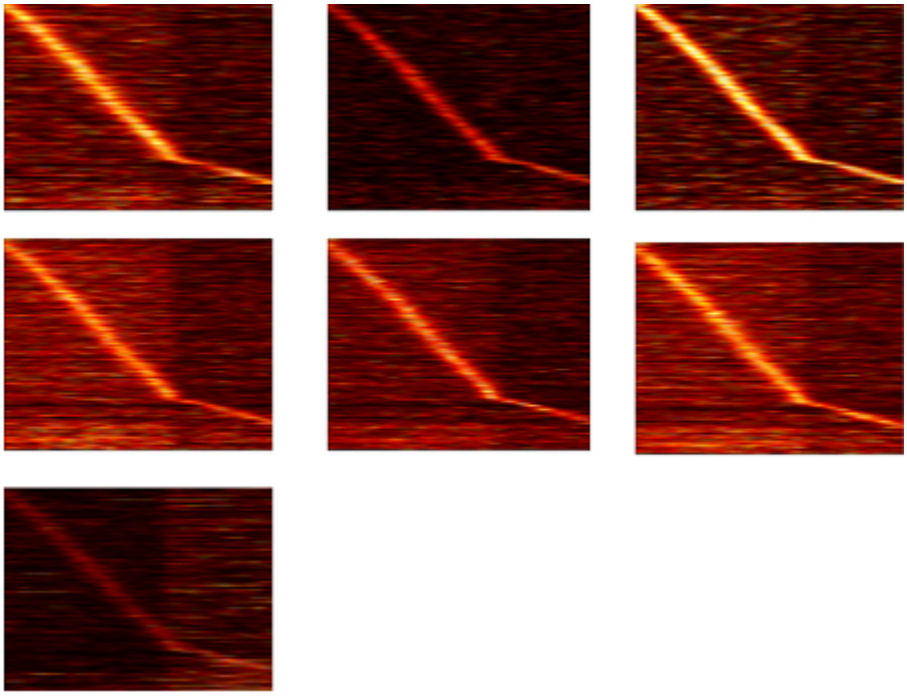

Figure 4.2: The most enriched peaks with FAIRE at seven time points.

Each heat map of Figure 4.2 corresponds to a time point (first row corresponds to T0, T6 and T12, second row to T18, T24 and T30 and the last heat map to T36). Each heat map shows the regions 1000 bases upstream from each gene start and 600 bases inside each gene. Genes are sorted by positions of the most enriched peaks within the promoters with the farthest peaks from start codons being at the top of the map. The fact that the most enriched FAIRE peaks do not

change location might signify the biological relevance of the genome regions under the peaks. That is why in the next approach we used 300 bases (approximately equal to the size of two nucleosomes) centered at the peaks. We calculated the average FAIRE coverage over the selected 300 bases for seven time points and calculated Pearson correlations between these values and gene expression data. Again, the results showed no significant correlation.

In the following final analysis, we selected 400 bases regions within promoters containing the most enriched FAIRE peaks for each gene and measured the degree of chromatin opening for these regions (see Section 4.2.2) at seven time points across the erythrocytic cycle. Then we clustered the genes using kmeans by these seven values. Figure 4.3 shows the resulting four clusters of genes.

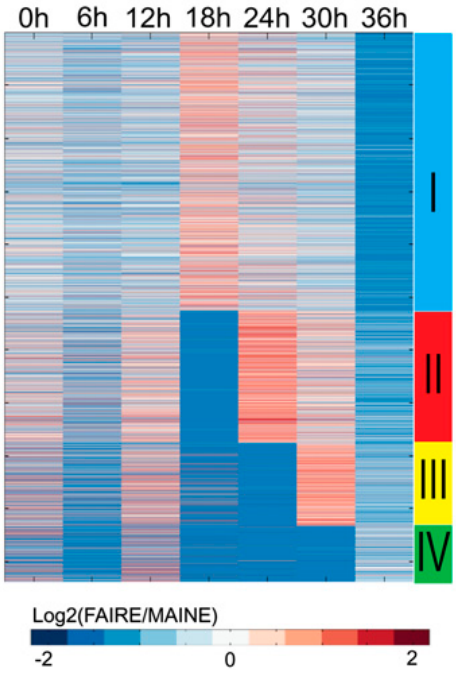

Figure 4.3: Clustering by the degree of chromatin opening.

In Figure 4.3, genes are sorted by the resulting four clusters. The rows of the heat map correspond to genes and columns show the degree of chromatin opening at seven time points. The more closed chromatin is, the darker is color in Figure 4.3.

We obtained four well-defined clusters that were further analyzed for enrichment of the genes from the gene expression clusters using the hypergeometric test. Genes expressed in sporozoites and gametocytes and genes involved in invasion of red blood cells (genes corresponding to gene clusters 1-3, 14 and 15 in [46]) are over represented in cluster I with p-value of $10^{-7.5}$. Genes corresponding to ring, trophozoite and schizont stages (corresponding to gene clusters 4-7) are over represented in cluster III (p-value of $10^{-3.4}$) and in cluster IV (p-value of 10^{-5}). Cluster II did not have any significant enrichment of the genes from functional gene expression sets.

Overall, our analysis suggests that mRNA steady state levels could not be explained by fine changes in chromatin. Our study showed that the chromatin of *P. falciparum* goes through two extreme stages: open and closed, which contradicts the classical eukaryotic model of initiation transcription after precise removal of a few nucleosomes. This binary state of chromatin together with accessibility of promoters and downstream of the genes to transcription factors for the most of the erythrocytic cycle, allowed to suggest a transcription regulation model for *P. falciparum* as shown in Figure 4.4. Genome-wide variations of densities for MAINE coverage are shown in red and for FAIRE in black. In this model, chromatin structure and nucleosome turnover control massive transcription during the erythrocytic cycle.

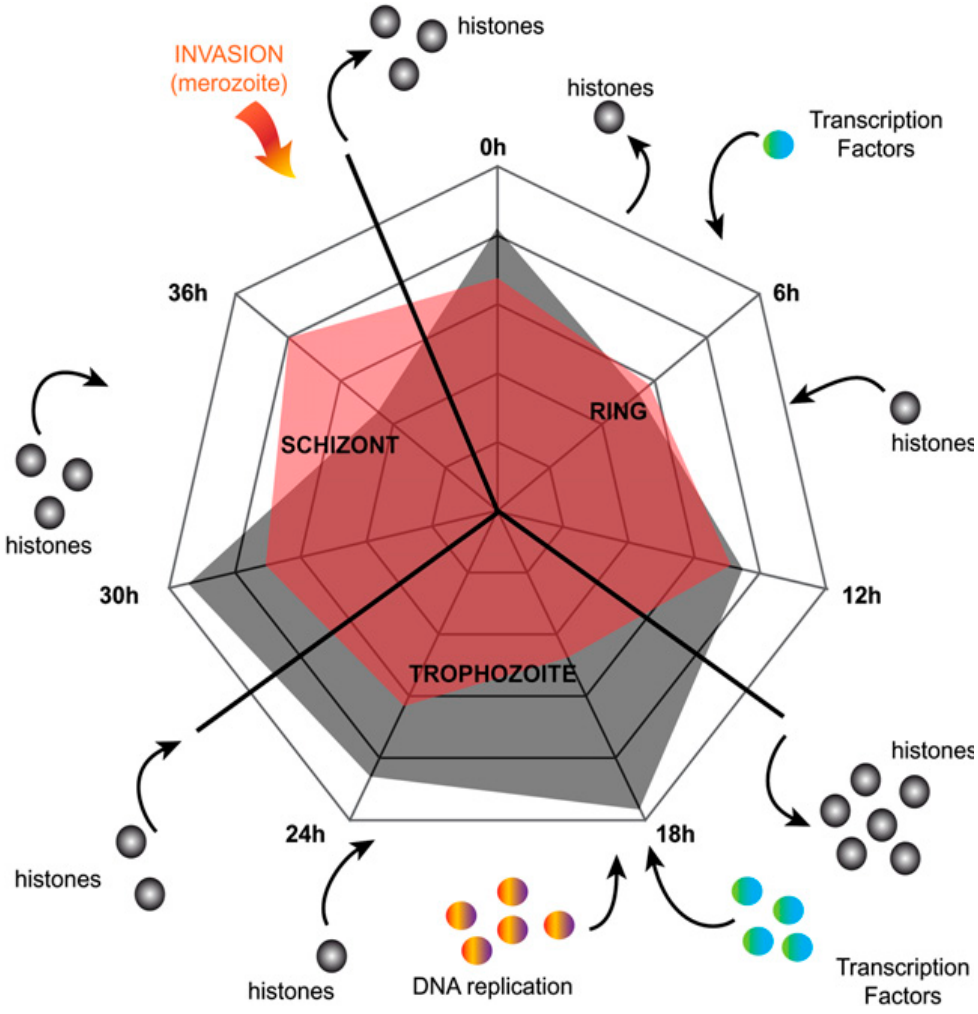

Figure 4.4: Proposed model of chromatin remodeling throughout *P. falciparum*'s erythrocytic cycle.

After the invasion of a red blood cell by a parasite at its merozoite morphological stage, histones are depleted and chromatin loosens. During the ring stage, transcription occurs locally in a regulated manner. At the early trophozoite stage, histones are massively depleted; transcription factors can freely bind and replication can take place. DNA-related metabolic processes are permitted. After replication, DNA re-packs at the schizont stage and the parasites divide and escape the host as differentiated merozoites that are ready for a new cycle of infection.

Chapter 5

Motif discovery using dynamic chromatin remodeling

5.1 Introduction and background

Despite extensive efforts to discover transcription factors binding sites in the human malaria parasite, only a few motifs have been experimentally validated to date. Most of the previous methods for *in silico* motif discovery heavily rely on gene clusters by function or by gene expression profiles. Recent research in chromatin remodeling of *P.falciparum* suggests that chromatin architecture plays an important role in transcriptional control [65]. Here we propose a methodology for discovering *cis*-regulatory elements using our new data on dynamic chromatin structure remodeling described in Chapter 1 and Chapter 4.

In general, the problem of motif discovery for transcription factors binding sites has been studied intensely, and many algorithms and tools have been developed to date. Algorithms are usually classified into two categories: enumerative and alignment-based. The latter method uses a probabilistic modeling approach together with an optimization technique (expectation-maximization or Gibbs sampling) to identify sequence patterns that are over represented in input sequences. The enumerative approach involves enumeration of a vast majority of words in input sequences, an appropriate score assignment using statistical measurement, and finally selection of the most statistically significant words as putative motifs. An extensive overview of existing approaches for motif discovery and computational challenges associated with this problem is given in [56]. Some of the most widely-known tools for motif discovery are MEME [8], Weeder

[62], and AlignACE [37]. Successful as they are with model organisms, these tools might not be the best choice with an extremely AT-rich like human malaria parasite genome.

Some efforts to discover *cis*-regulatory elements *in silico* in the human malaria parasite have been made in the past five years [26, 27, 87, 91, 38, 75]. These approaches define different criteria for gene clustering and/or different statistical models to identify over represented motifs in a given set of sequences. Below we give a short overview of some of these methods for motif discovery and their findings as well as the methods that used classical biological assays to identify motifs.

Elemento et al. proposed an algorithm named Finding Informative Regulatory Elements (FIRE) to discover putative motifs [26]. FIRE uses motif profiles and gene expression profiles to discover transcription factors binding sites. A motif is present in the promoter if one of its instances occurs in either of the DNA strands within the promoter. Gene expression profiles are used to cluster genes. For each motif, FIRE calculates the value of the objective function, which is used to select the most statistically significant motifs. This study resulted in twenty-one highly informative putative motifs in *P. falciparum.*

Wu et al. used comparative genomics to identify putative motifs given a set of upstream sequences from two different species [87]. The authors improved on a previous study [27] by introducing a statistical model that learns parameters from orthologous sequences (taking into account orthologous relationships) and allows for different sequence length. The method uses the conservation of the *z*-score as a scoring metric for statistical significance of the resulting motifs. The algorithm starts by learning parameters, namely the probability distribution due to a background model and the probability distribution due to orthologous relations from the input sequences. Then it enumerates all *k*-mers (words of size *k*), calculates their *z*-scores and sieves the *k*-mers with a given threshold. For the selected *k*-mers, the method iteratively introduces

mismatches and picks at each iteration the motif with the highest *z*-score. After motifs clustering, the authors reported 38 motifs.

Young et al. designed a new enumerative tool called Gene Enrichment Motif Searching (GEMS) that uses a hypergeometric-based scoring function and a position-weight matrix optimization procedure to search for putative motifs [91]. Twenty-one clusters of functionally related and co-expressed genes were used. GEMS starts with assigning *p*-value scores to the set of all unique words of a fixed size (5-9 bp), then sorts the words in increasing order of their *p*-values and processes each word at a time. An initial position weight matrix (PWM) of a word is built from its one mismatch edit neighborhood. Then an optimization routine is used to find the best parameters, namely similarity threshold and number of mismatches. Similar motifs are clustered using the *Tanimoto* distance metric that accounts for positional information of motif occurrences within promoters. The authors used phylogenetic analysis to sieve the remaining motifs further, which resulted in the 34 putative motifs in promoter sequences and 21 in introns-derived sequences. One of the identified motifs was experimentally validated by the authors.

Iengar and Joshi used three popular motif discovery tools: MEME [8], AlignACE [37] and Weeder [62], to identify putative motifs in promoters of co-expressed genes [38]. The authors used strict cut-offs and selected only motifs that were found by all three programs. This study resulted in 27 strong motif groups that were further combined into four unique motif patterns: G-rich (having GGG or GGGG), C-rich (having CCC or CCCC), CACA and TGTG (having patterns TGTG or TGTATG).

In contrast to *in silico* motif finding strategies, Silva *et al.* used Protein-binding microarray assay to identify transcription factors binding sites for two members of the Apicomplexan AP2 family of putative transcriptional regulators [75]. The microarray contained

all possible 10-mers. Two specific binding sites were identified in this study, namely TGCATGCA (the PF14_0633 associated motif) and GTGCAC (the PFF0200C motif).

Most of the *in silico* methods described above heavily rely on gene clusters by function or by gene expression profiles. Here we endeavor to identify motifs using the dynamic chromatin structure remodeling data described in Chapter 4.

Recall that FAIRE coverage isolates nucleosome-free regions of the genome, which identifies DNA that is potentially accessible to transcription factors. For this reason, we used FAIRE coverage to discover putative motifs and their associated target genes. Our approach identified 144 putative motifs, 69 of which showed statistically significant enrichment in at least one functional gene set. To our knowledge, this is the first study that uses exclusively FAIRE coverage for motif discovery.

5.2 Methods

Our hypothesis is that regions within promoters with the highest variance of FAIRE coverage across the erythrocytic cycle are more likely to contain transcription factors binding sites. Here we provide an overview of our approach for motif discovery using FAIRE coverage. Figure 5.1 shows a flowchart of our method. First for each gene we identified a *functional window* within its promoter, a 151-bases region with the highest variance of FAIRE coverage across seven time points of the erythrocytic cycle. Then for each gene we calculated average FAIRE coverage within the functional window at seven time points and clustered the genes by these seven values using *k*-means clustering. We call the resulting clusters preliminary clusters by FAIRE coverage. Next we identified putative motifs using hypergeometric enrichment score (defined below). For

each motif we selected all genes with occurrences of this motif within functional windows and called these genes a *target genes cluster*.

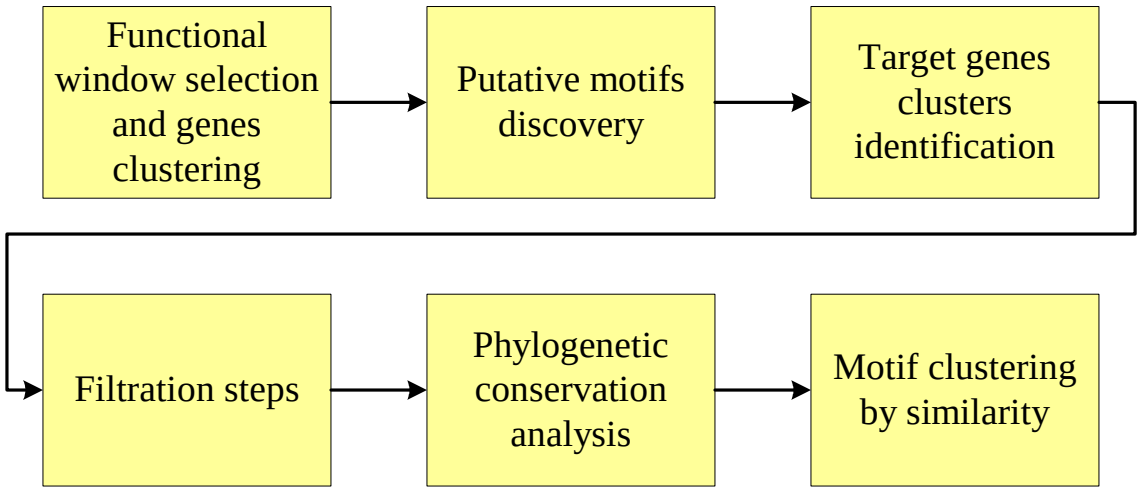

Figure 5.1: Flowchart of motif discovery using FAIRE coverage.

Next we applied two filtration steps described below, phylogenetic conservation analysis, and finally clustered the resulting motifs by similarity. For the final set of selected motifs we used additional analysis, namely we compared our motifs with the motifs identified in previous studies (described above), we evaluated positional bias of our motifs relative to transcription start sites and to the predicted promoters from [13, 82,83], and finally we studied enrichment of our motifs in functional gene sets by gene ontology and gene expression profiles. In the following sections we provide more details for each step of our approach for motif discovery.

5.2.1 Motifs scoring

We use hypergeometric probabilistic model with the background as a set *S* of all *k*-mers inside 1000-bases promoters of all genes. Let *N* be the size of *S* and *n* be the size of a subset *M* of *k*-mers inside 151-bases functional windows within the promoters of a given gene cluster. This latter subset of *k*-mers is the positive set. We use the hypergeometric enrichment score as a scoring metric for statistical overrepresentation of a motif inside the positive set *M*. The hypergeometric enrichment score (HES) is defined as the negative log of the hypergeometric probability of a motif (*p*-value) having an equal or greater number of occurrences in the positive set if the set was drawn randomly. The smaller is the *p*-value, the smaller is the chance that the observed number of occurrences is to random factors. This scoring metric estimates not only the overrepresentation of a motif within a given cluster of genes, but also incorporates statistical overrepresentation of the motif within the selected windows inside the promoters. Let *r* denote the number of occurrences of motif in *S* and *y* denote the number of occurrences of the motif in the positive set *M*. The hypergeometric *p*-value is

$$P(N,r,n,y) = \sum_{i=y}^{\min(n,r)} \frac{\binom{r}{i}\binom{N-r}{n-i}}{\binom{N}{n}}$$

and *HES* is calculated as

$$HES = -\log_{10} P(N,r,n,y)$$

5.2.2 Motif representation

Let *t* be a *k*-mer in the positive set *M*. We define *N*(*t*, 1) as the 1-mismatch neighborhood of *t* in *M*: a *k*-mer *s* in *M* belongs to *N*(*t*, 1) if a Hamming distance between *s* and *t* is less than or equal to one. Given *M* and *t*, we first compute *N*(*t*, 1), then we calculate HES for each *k*-mer in the *N*(*t*, 1), sort the *k*-mers in *N*(*t*, 1) in decreasing order of their HES, and then use dynamic programming to select a subset of *N*(*t*, 1) that maximizes HES. We represent *t* as a subset of *k*-mers from *N*(*t*, 1) that maximizes HES calculated over these *k*-mers, and we call the *k*-mers in this subset *mutants* of *t*. We observed in practice that the number of mutants is usually less than ten; to speed up calculations we restrict the list of *N*(*t*, 1) to the top ten *k*-mers with the highest HES.

To ensure the selection of motifs according to the probabilistic model without replacement, we do not consider motifs whose mutants' occurrences overlap each other.

5.2.3 Motifs and their target gene clusters

Here we describe our methodology to identify putative motifs and their associated clusters of target genes, or *expanded* clusters. We preprocessed each *k*-mer in the genome by counting the number of its exact occurrences inside all 1000-bases promoters, and by calculating the probability of its occurrence given the background distribution of A, C, G and T obtained from the genome. Under the i.i.d. model, the probability of an occurrence of a *k*-mer is simply the product of probabilities of its bases. For each preliminary cluster by FAIRE coverage, we first applied a procedure that selects potential *k*-mers for further analysis. To select this preliminary subset of *k*-mers from all *k*-mers in a cluster, we process each *k*-mer in the positive set one at a time. For a *k*-mer *t*, we find its 1-mismatch neighborhood *N*(*t*, 1), and verify that a set of conditions (described next) on *N*(*t*, 1) is satisfied: if all requirements are satisfied, *t* is selected.

The following conditions are used: (1) *t* has to occur in at least 5 distinct input sequences in a given cluster and (2) the expected number of distinct sequences in which *k*-mers from *N*(*t*, 1) occur has to be less than the actual number of distinct sequences with *k*-mers from *N*(*t*, 1). The size of the smallest preliminary cluster was 33, and square root of 33 is approximately 5; that is why 5 distinct input sequences where *t* occurs is one of the requirements stated above [8]. The expected number of occurrences and the expected number of distinct sequences are computed on the positive set of *k*-mers within the given sequences of a cluster of genes. Let *i* be a *k*-mer from *N*(*t*, 1), then the expected number of occurrences of *i* in the positive set is $\mu_{occ}^{i} = p_i n$, where p_i is the probability of one occurrence of *i* in the genome, and *n* is the total number of *k*-mers in the positive set; then the expected number of distinct sequences containing *i* is

$$\mu_{seq}^{i} = 1 - e^{-\mu_{occ}^{i}}$$

and the expected number of distinct sequences for a motif *t* is calculated as

$$\mu_{seq} = G \sum_{i \in N(t,1)} \mu_{seq}^{i} ,$$

where *G* is the number of sequences in the positive set.

The first threshold on the least number of distinct sequences in which *k*-mers from *N*(*t*, 1) must occur filters out motifs that have a small probability of being over represented in the positive set. The second threshold filters out motifs that are in general abundant in the genome and therefore overrepresentation of these motifs in the positive set is more likely due to a random chance.

After the potential motifs have been identified, we study each of them in the context of corresponding preliminary cluster by FAIRE coverage. For each *k*-mer *t*, we find the subset of mutants in $N(t, 1)$ that maximizes HES for *t* in the cluster by FAIRE coverage. Then we identify all genes that have exact occurrences of the mutants of *t* within the selected 151-bases windows. These genes constitute a new *expanded* cluster for which the set of mutants for *t* is recomputed. This final set of mutants in the expanded cluster is a putative motif and the expanded cluster is its associated set of putative target genes. We performed this strategy with three different motif length *k*, namely $k = 6$, $k = 7$ and $k = 8$. The three validated motifs for *P. falciparum* are of length 6 and 8 bases; therefore, we restricted our analysis for motifs of length 6, 7 and 8.

We provided different representations for a motif: as a subset of mutants, as a position weight matrix (PWM), as a sequence logo and as a regular expression. A regular expression is computed as follows: for each base of a motif *t*, its regular expression has either a single nucleotide or a subset of nucleotides inside square brackets if more than one nucleotide occurs with a frequency of at least 25% at the corresponding position of *t*. For example, the regular expression AAC[TG]TT indicates that T and G occur with a frequency of at least 25% at fourth position of the given motif, and nucleotides at other positions occur with frequency greater than 75%. This will allow biologists to make an informed decision about which particular sequence of a motif to use in experimental validation.

For all identified motifs and their expanded clusters, we applied a stringent pipeline of filtration steps described below to distinguish between true and false positives putative motifs.

5.2.4 Additional filtration steps

Since we calculate HES over the number of mutants occurrences rather than over the number of distinct genes where mutants occur, we had to use additional filtration steps to ensure that a high

HES is not due to multiple occurrences within a few genes of the cluster. We required the number of distinct genes where the motif occurs to be close to the number of occurrences of the motif in the target genes cluster. We used the ratio of the number of distinct genes where the motif occurs and the number of occurrences of the motif in the cluster to filter out motifs whose ratio is less than 0.8. Since HES of a motif is calculated over the sequences of 151 bases in length, we expect the exact occurrences of mutants inside a single sequence to be close to one.

Our method identified thousands of potential motifs within preliminary clusters. To filter out putative motifs we considered only the motifs whose HES was greater than or equal to the mean HES in the distribution of HES for all motifs of a given length k = 6, 7 and 8.

5.2.5 Phylogenetic conservation analysis

We used phylogenetic conservation information to filter out non-conserved motifs that were considered to be false positives. Similarly to the approach by Young et al. [91], we calculated HES for the putative motifs in four orthologous species: *P. berghei*, *P.chabaudi*, *P. vivax* and *P. yoelii*. For each putative motif, we calculated the enrichment score for genes orthologous to the genes with mutants occurrences in the motif's expanded cluster within 1000 bases promoter 5' upstream of the start codon. The motifs that had HES of 2.0 (corresponding to *p*-value of 0.01) in at least one of the four orthologous species were kept for further analysis. To avoid artificially high orthologous HES, we required the ratio of the number of distinct genes where the mutants occur and the number of exact occurrences being greater than or equal to 0.5 (this threshold is lower here due to increased length of input sequences). This ensures that HES is supported with enough orthologous genes and that all occurrences do not fall into the promoters of a few genes.

5.2.6 Motifs clustering

After the filtration of putative motifs by the steps described above, some of the resulting motifs were very similar. Hence, we selected only unique motifs with the highest HES and clustered the remaining putative motifs by similarity using the Tanimoto distance [62, 91] and Pearson correlation coefficients of their position weight matrices (PWMs) [89] to filter out duplicates and highly similar motifs. The Tanimoto distance accounts for positional similarity of the putative motifs. It takes on the values from 0 to 1: if two motifs share all positions, then the Tanimoto distance is 0. We used the Tanimoto distance to calculate pairwise similarity of the motifs, and we chose the motif with the highest HES to represent the motifs with the similarity distance smaller than 0.5. This means that two motifs must share at least half of their positions to belong to the same motif cluster. Since our method does not rely on co-regulated clusters, we assume that for a given motif not all genes in the target genes cluster are co-regulated. That is why we allow more flexibility on the threshold for the Tanimoto distance (compare our threshold of 0.5 with 0.8 that was used for co-regulated clusters in [62, 91]). Given the positional occurrences for two motifs represented in sets A and B respectively, we define two occurrences (one from A and one from B) to be overlapping if they share at least one base. The Tanimoto distance was calculated as follows

$$1 - \frac{|A \cap B|}{|A \cup B|}$$

The intersection of A and B includes all overlapping positional occurrences.

Next, we clustered motifs by the Pearson coefficient of their PWMs. This step filters out the motifs very similar by their content. Some of the motifs might be shifted versions of each

other. We define that two motifs are shifted versions of each other if they share a *k*-mer where *k* is a half of a motif's length. We considered two motifs similar if the Pearson coefficient of their PWMs or of their PWMs built on shifted versions of the motifs was greater than or equal to 0.75 (used in [87, 89]). To build PWMs for two motifs that are shifted versions of each other, first we built PWM for the motif with the highest HES and then we constructed PWM for the other motif that we shifted to align with the highest-HES-motif.

5.2.7 Gene orthology and gene functional sets

Orthologous gene maps between *P.falciparum* and *P. berghei*, *P.falciparum* and *P. chabaudi*, *P.falciparum* and *P. vivax*, and *P.falciparum* and *P.yoelii* were generated by the OrthoMCL database (http://www.orthomcl.org/). We retained only those orthologous genes that had 1000 bases upstream of start codon available. A total of 1918 orthologous genes of *P. berghei*, 1249 of *P. chabaudi*, 4126 of *P. vivax*, and 2688 of *P.yoelii* were used in this study.

In order to study the motifs enrichment in functional gene sets, we used gene ontology (GO) functions from *P.falciparum* version 6.0 together with ontology-based pattern identification (OPI) clusters from (http://carrier.gnf.org/publications/OPI/) to retrieve a total of 13859 GO/gene pairs, with 1288 distinct GO names.

As an alternative for GO annotation, we used gene sets categorized by gene expression profiles. We employed fifteen clusters of genes previously obtained by gene expression profiles [46] to obtain four gene expression (GE) groups as follows. Gene expression group I (GEI) contains genes expressed in sporozoites and gametocytes (clusters 1-3), GEII contains genes corresponding to ring, schizont and trophozoite stages (clusters 4-7), GEIII contains genes expressed at the throphozoite stage (clusters 8-13) and group GEIV contains genes expressed at

sporozoite, gametocyte and schizont stages and involved in red blood cell invasion (clusters 14-15). Finally, we used fifteen clusters by gene expression profiles [46] as functional gene sets.

5.3 Results and discussion

5.3.1 Selecting windows inside promoters and gene clustering

As said our study explores the hypothesis that modification of chromatin structure controls transcription in malaria. In the course of our project described in Chapter 4, we observed that FAIRE coverage surrounding a validated motif undergoes drastic changes throughout the erythrocytic cycle indicating that putative motifs most probably are located within promoter regions with the highly variable FAIRE coverage. Therefore, we restricted our search for putative motifs to the windows within the promoters with the highest variance of FAIRE coverage across the seven time points in the parasite erythrocytic cycle.

Using the start position of the uniquely mapped reads, we determined the FAIRE raw coverage by extending the mapped reads up to 200 bases (the average length of DNA fragments in FAIRE library) and then normalized the raw coverage per million mapped reads and per percentage of area covered. For each gene, we selected a window within its promoter as follows. We slid a 151-bases window (approximate size of a nucleosome) within the promoter and calculated average FAIRE coverage on all 850 positions (two consecutive windows overlapped by a single base). We repeated this procedure on all seven time points; thus, each 151-bases window at position i within a promoter, $-1000 \le i \le -151$, is presented by seven values for average FAIRE coverage. Next, for each gene we chose the 151-base window with the highest variance of average FAIRE coverage across seven time points. Figure 5.2 shows an example of the functional window for a single gene.

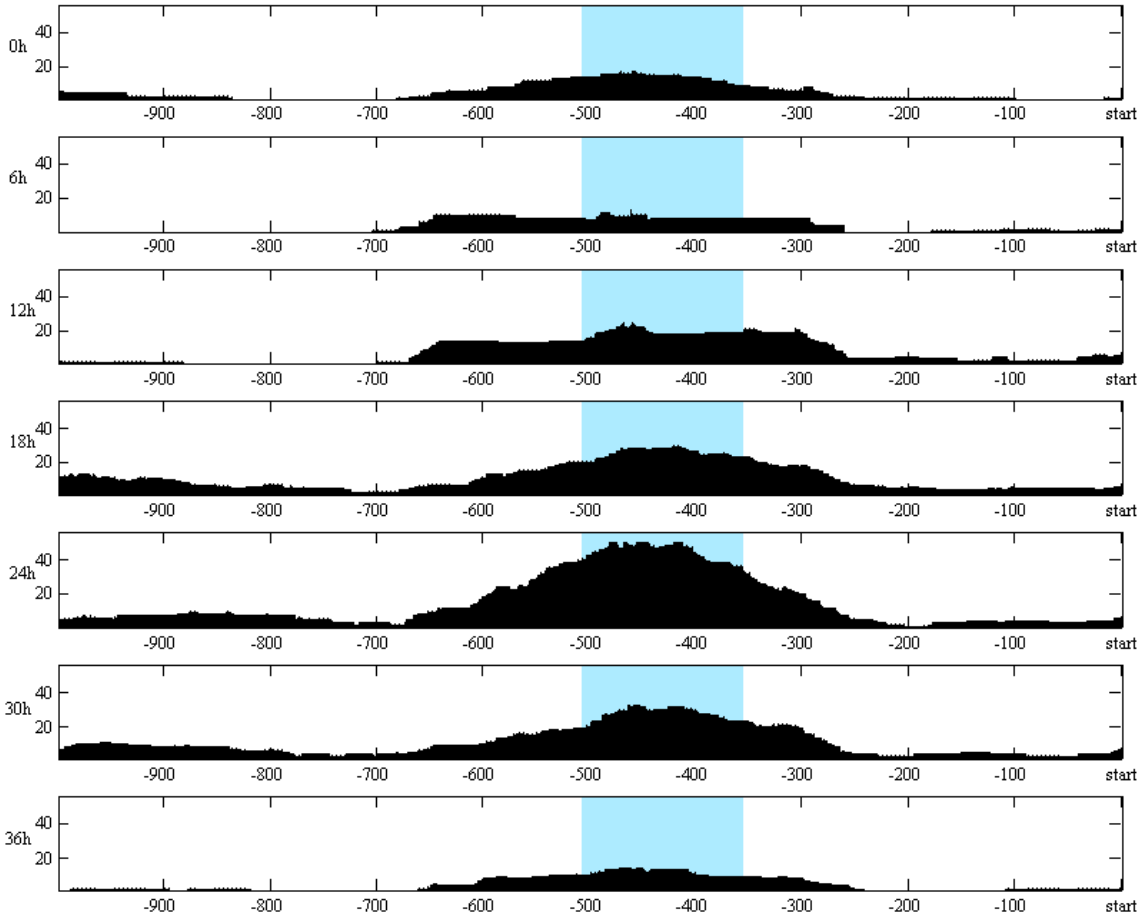

Figure 5.2: Example of a functional window with the highest variance of FAIRE coverage.

FAIRE coverage is shown in black for seven time points across the erythrocytic cycle of the parasite. The shaded region corresponds to the functional window.

The seven values for the FAIRE average coverage of the selected windows were used to cluster genes. We used *k-mean* clustering to build fifteen preliminary gene clusters that were used in our search for discovery of motifs and their associated clusters of target genes. We also tried other values for k = 5, 10 and 20. For each choice we computed motifs of length 6 and compared them using the Pearson coefficient of their PWMs (see Section 5.2.6). When the thresholds on Pearson coefficient is 0.6, the resulting motifs for k = 5, 10 and 20 turned out to be very similar to

the motifs for k = 15. In other words, the number of preliminary clusters did not affect the resulting motifs. In the rest of this study, we used fifteen clusters as a good tradeoff between a cluster size and the quality of cluster separation. The fifteen resulting clusters have cluster sizes in the range from 33 to 841 with average of 364. Figure 5.3 shows the values of the FAIRE average coverage for the centroids of the fifteen clusters. X-axis corresponds to 15 clusters, and for each cluster seven bars correspond to the FAIRE average coverage at seven time points (Y-axis).

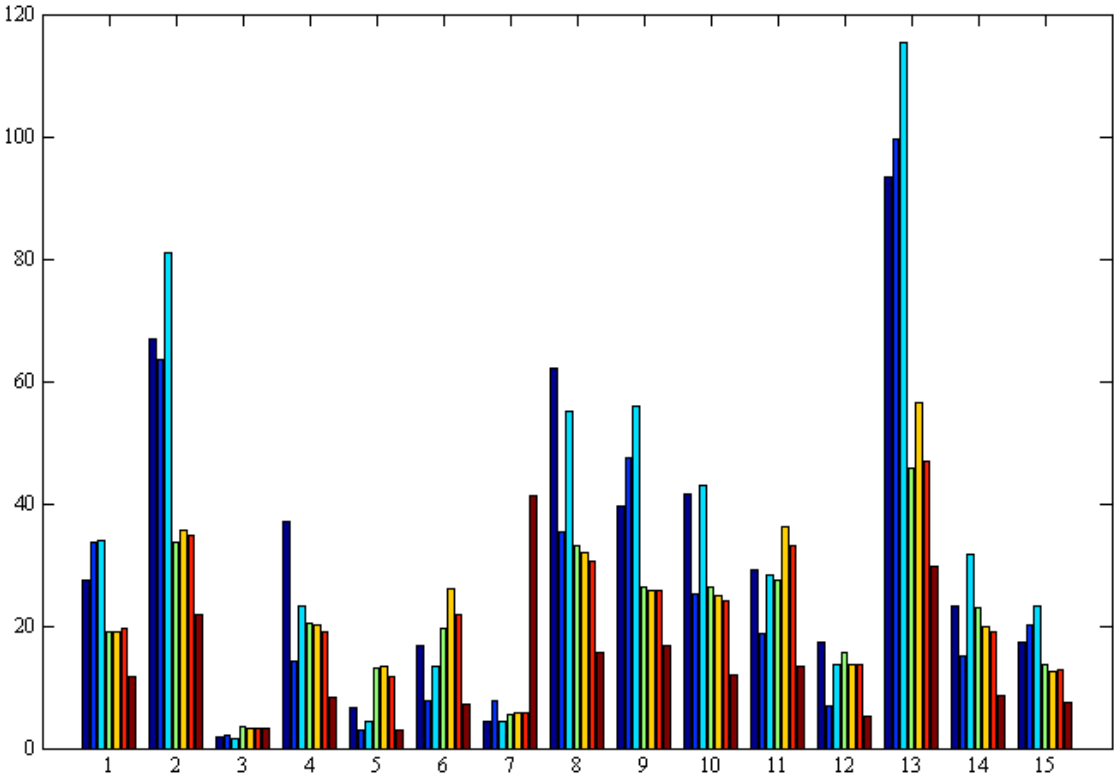

Figure 5.3: FAIRE average coverage for the centroids of fifteen clusters.

5.3.2 Motifs and their target gene clusters

Much of the success of *in silico* motif discovery methods depends on the gene clustering and on the beliefs *a priori* about which genes are co-regulated and are likely to share the same transcription factors binding sites. In the analysis of *P. falciparum*, the problem of gene clustering remains challenging: out of about 5460 genes, over 2200 genes have no known function, and 3241 genes have no distinct gene expression profiles. Thus, clustering by function may result in incomplete clusters and clustering by gene expression profiles or by function does not cover all genes. Here, we attempted to find motifs and their associated clusters of target genes using only FAIRE coverage and conservation information with orthologous genes. Instead of using clusters of genes by function or gene expression, we used preliminary clusters obtained by processing FAIRE coverage. Then we identified putative motifs and their target genes clusters as described in Section 5.2.3. Figure 5.4 shows the distribution of the sizes for target genes clusters of 144 putative motifs identified in this study.

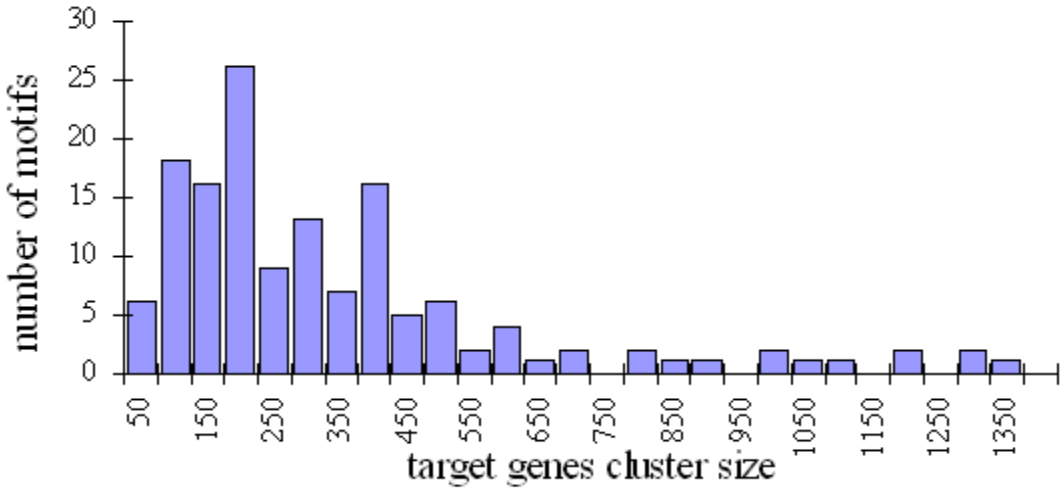

Figure 5.4: Distribution of sizes for target genes clusters.

Observe in Figure 5.4 that the majority of motifs have target genes clusters with sizes from 100 to 400. The average size was 139.07, the standard deviation was 279.1, and the smallest size was 41 and the largest 1333.

As said, after obtaining a cluster of putative target genes for a motif, we recalculated the subset of mutants representing the motif inside the target genes using HES. For instance, the distribution of HES for motifs in their target genes clusters for motifs of length 6 is given in Figure 5.5. Motifs of length 7 and 8 had similar HES distributions with average and standard deviation of (88.19 and 62.07) and (73.17 and 60.57) respectively. The total of motif/clusters considered by our method was 2727 for 6-mers, 8813 for 7-mers and 19435 for 8-mers.

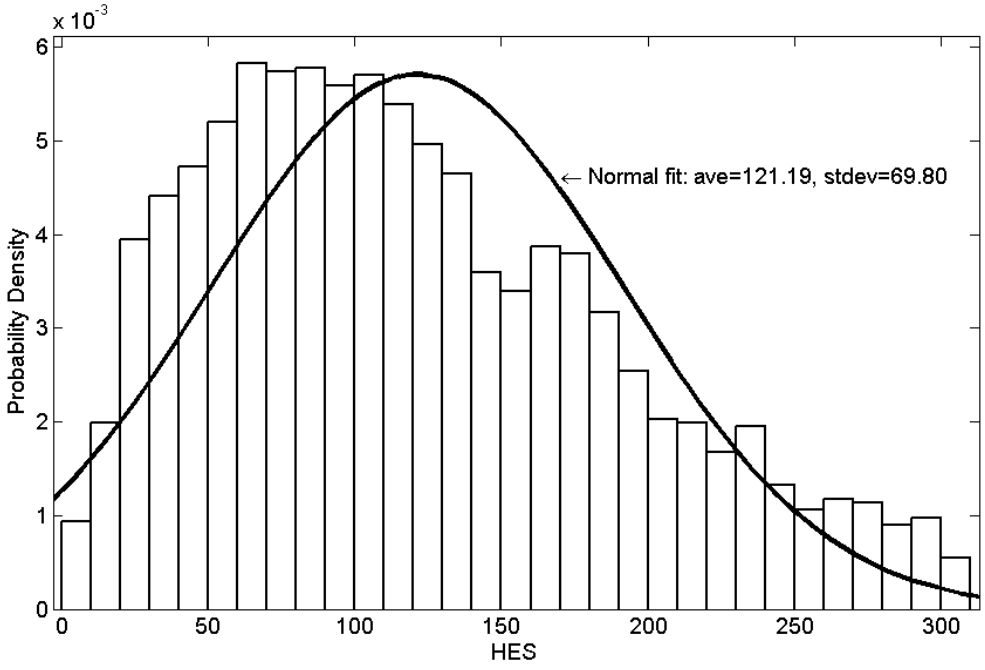

Figure 5.5: HES distribution for 6-mers inside their target gene clusters.

Recall that in order to select biologically significant motifs from these sets, we used phylogenetic conservation information. After filtration by HES in orthologous species, we obtained total of 229, 128 and 222 motifs/clusters with motifs of length 6, 7 and 8 respectively.

To obtain the final set of motifs, we further chose only unique motifs with the highest HES and clustered the remaining motifs by *Tanimoto* distance and by Pearson coefficient of PWMs of the motifs (see Section 5.2.6). The final list of motifs consists of 26 6-mers, 49 7-mers and 69 8-mers.

5.3.3 Comparison with previously found motifs

We investigated the question whether our approach was capable of identifying motifs that had been previously validated in literature [75, 91]. We compared our motifs with PF_140633 motif (also known as TGCATGCA) and PFF0200c motif (known as GTGCAC) of [75], and with the motif NGGTGCA of [91].

We used the Pearson coefficient on the corresponding positions of the PWMs of two motifs of interest as a similarity metric. To build PWM for Silva's motifs, we used the occurrences of the motifs in all genes clusters provided in Supplemental methods as potential target genes for these motifs [75]. Our top-scoring motif GTGCAC has a similarity score of 0.9969 with Silva's motif GTGCAC, and a shifted version of GTGCAC and Young's motif NGGTGCA resulted in the score of 0.833. The most similar motif to Silva's motif TGCATGCA was TATGCAT with the similarity score of 0.704. Table 5.1 shows the results of this comparison.

Our Motif	Others Motif	Reference	Our PWM	Others PWM	Score
GTGCAC	GTGCAC	PFF0200c, De Silva et al., 2008			0.997
GTGCAC shift 1b left extend 1b left	NGGTGCA	PfM18.1, Young et al., 2008			0.834
TATGCAT shift 2b right extend 1b right	TGCATGCA	PF14_0633, De Silva et al., 2008			0.704

Table 5.1: Comparison of our motifs and the motifs from previous studies.

In addition, we compared our 144 motifs with 50 motifs reported in [91] study using the Pearson coefficient of their PWMs and the threshold of 0.75 for selection of similar motifs. We found a total of 30 our motifs were similar to 23 motifs from [91]. The fact that we identified motifs highly similar to those that were validated in published studies provides validity to our method. Recall that we do not make any priori assumptions about the genes clusters, and we use exclusively chromatin architecture dynamics to discover *cis*-regulatory elements.

5.3.4 Additional supporting evidence

For the final collection of selected motifs, we estimated the enrichment of the motifs in functional gene sets using hypergeometric statistics. For this analysis, we used clusters of genes by GO, fifteen clusters by gene expression from [46] and four gene expression (GE) groups formed from the fifteen clusters of [46] as defined before (Section 5.2.7). For each motif we calculated HES on mutants' exact occurrences within 151-bases chosen windows of a functional gene set. To select statistically significant scores, we applied a randomized analysis as follows. For each motif we

selected 100 random clusters of genes and we computed HES for the motif's occurrences within the chosen 151-bases windows of the random genes. We considered the HES of a motif in a functional gene set to be statistically significant if the score was at least 2 and its z-score had *p*-value of 0.01 or lower. The z-score was calculated using average and standard deviation of the distribution of HES on the random clusters of the same size as the size of the functional gene set. Figures from 5.6 to 5.9 show z-scores of motifs' enrichment in GO, fifteen clusters by gene expression and GE functional gene sets. A total of 47.9% of the selected motifs showed enrichment in at least one functional gene set.

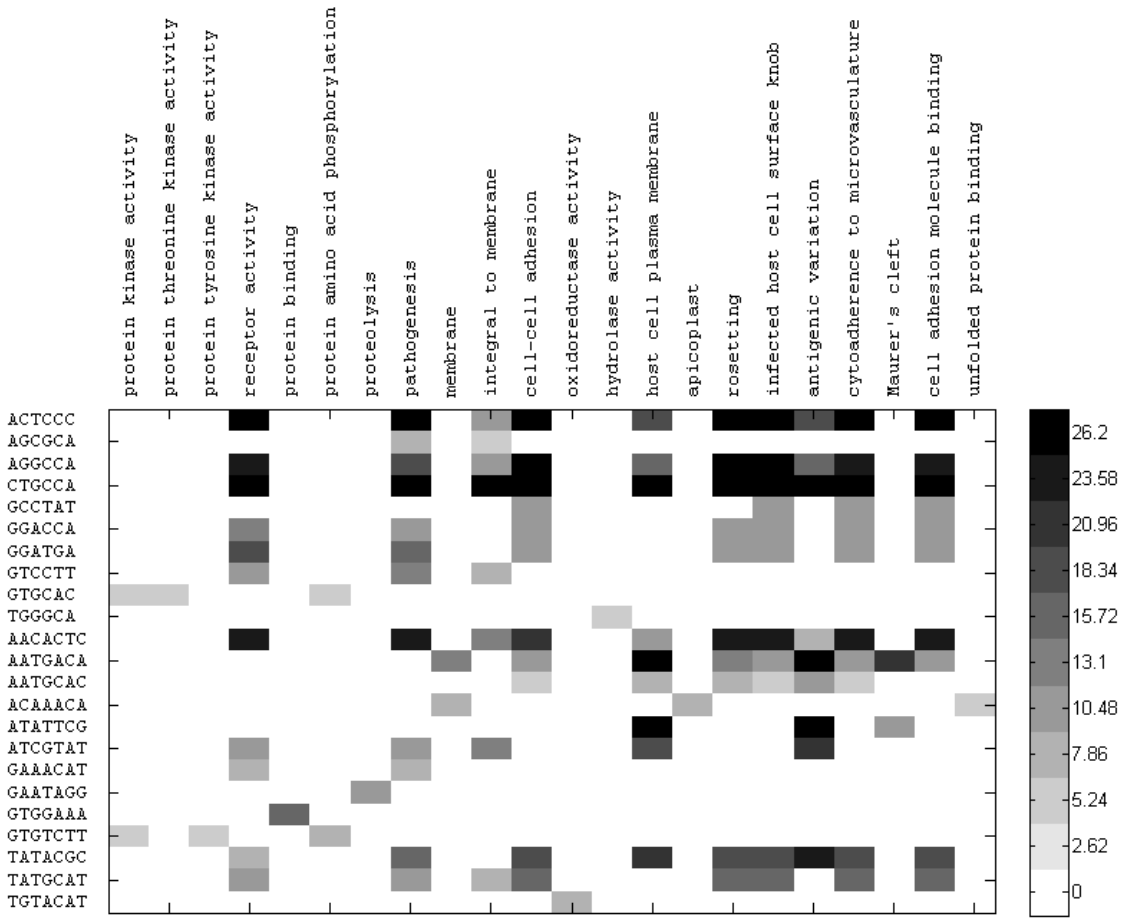

Figure 5.6: Enrichment in GO-functional gene sets for 6 and 7 bases motifs.

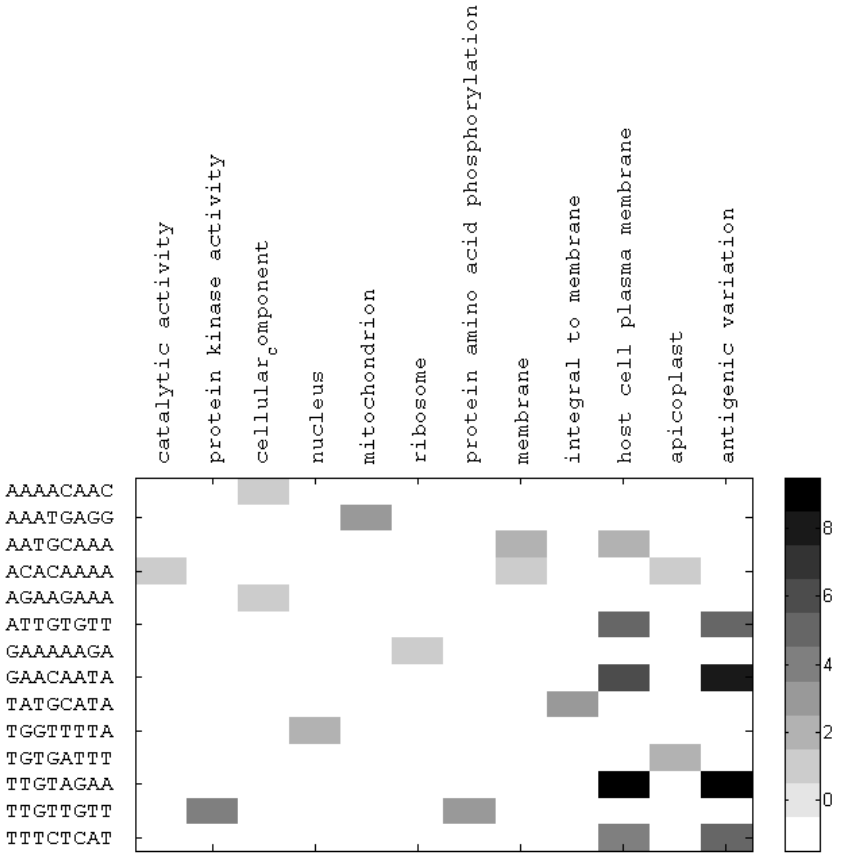

Figure 5.7: Enrichment in GO-functional gene sets for 8 bases motifs.

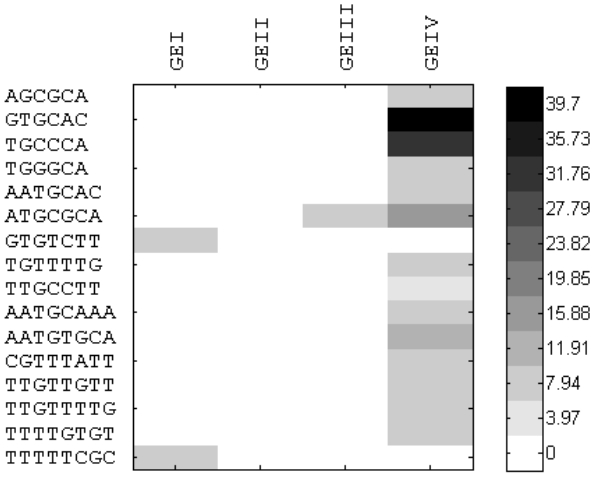

Figure 5.8: Enrichment of motifs in GE groups.

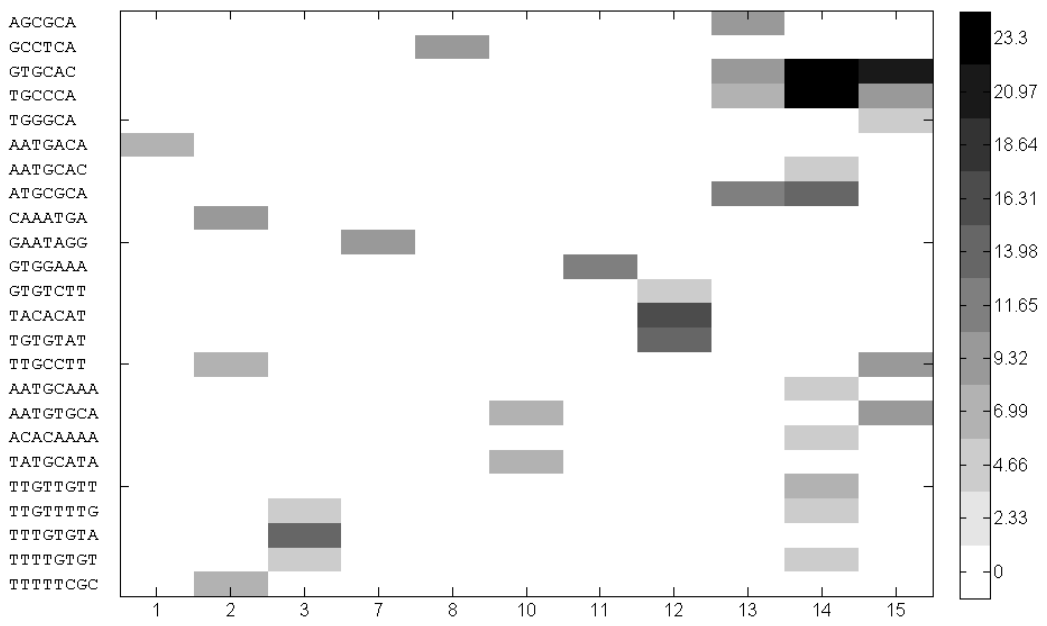

Figure 5.9: Enrichment of motifs in the clusters by gene expression.

In Figures 5.6-5.9 the darker the color is, the larger the z-score of the corresponding motif inside the corresponding given functional gene set.

5.3.5 Positional bias of the motifs relative to TSS and the predicted promoters

We conducted analysis of positional bias of identified motifs relative to the transcription start site (TSS) and relative to the predicted promoters published in [13, 82, 83]. We considered 2084 of TSS and 1027 of the predicted promoters from [13, 82, 83] located upstream of the genes within 1000-bases promoters. Here we describe our analysis for TSS (analysis for the predicted promoters was carried out similarly). We calculated the number of occurrences of 144 motifs inside [-1000, +1000] region around the TSS. A motif occurs at position *i* if it starts at *i*. Figures 5.10 and 5.11 show the distribution of positional occurrences of our motifs relative to TSS and

the predicted promoters respectively. We also computed the distribution of expected positional occurrences for our motifs if they were uniformly randomly distributed under the i.i.d. model within 1000-bases promoters. The actual probability density is shown in black and the expected in gray.

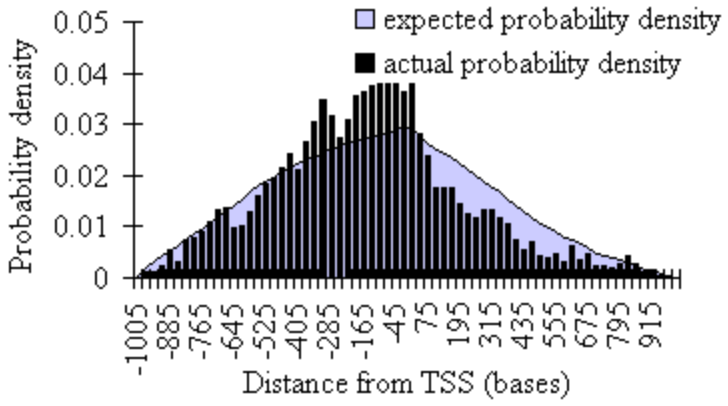

Figure 5.10: Positional bias of our motifs relative to TSS.

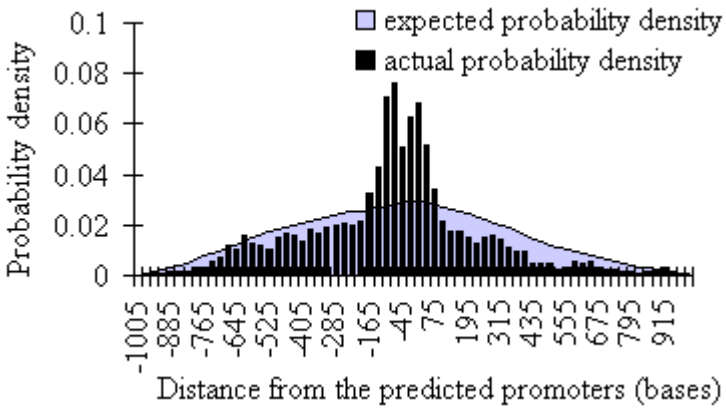

Figure 5.11: Positional bias of our motifs relative to the predicted promoters.

Observe in Figures 5.10 and 5.11 that some of our motifs have positional bias relative to TSS and to the predicted promoters respectively: the actual probability density for these motifs does not fit the expected probability density distribution. For comparison, Figures 5.12 and 5.13 show the results of similar analysis for 50 motifs from [91].

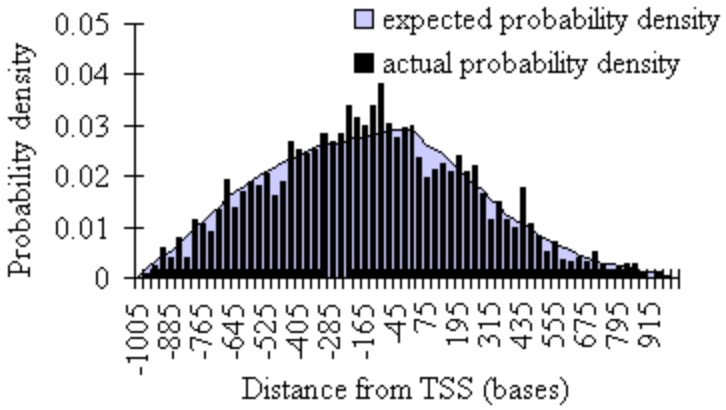

Figure 5.12: Positional bias of others motifs relative to TSS.

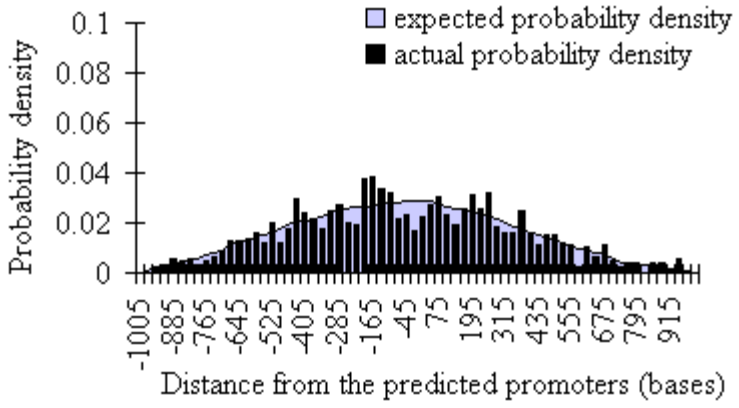

Figure 5.13: Positional bias of others motifs relative to the predicted promoters.

Next, we applied analysis of the positional bias relative to TSS and the predicted promoters for individual motifs similar to [9]. For each motif we computed the distribution of its positional occurrences within 1, 5, 10, 15, 20, 25 and 30-bases windows inside the background and inside [-300, +300] region around TSS (and the predicted promoters). We used [-1000, -301] region upstream of TSS as the background. Using the positional distribution of the motif in the background, we computed the 99% confidence line using a simple linear regression. We considered the number of positional occurrences of a motif inside a *k*-bases window to be statistically significant if it was greater than or equal to the threshold corresponding to *p*-value of 0.01 (*k* = 1, 5, 10, 15, 20, 25 and 30). Table 5.2 shows the percentage of our motifs that showed statistically significant positional bias relative to TSS. For comparison, we also provide the percentage of 50 motifs from [91] that showed statistically significant positional bias to TSS.

Window width, bases	Percentage of our motifs having *p*-value of 0.01 or lower	Percentage of others motifs having *p*-value of 0.01 or lower
1	97.0%	90.0%
5	91.6%	88.0%
10	80.0%	76.0%
15	65.2%	72.0%
20	62.5%	52.0%
25	53.4%	52.0%
30	41.0%	44.0%

Table 5.2: Percentage of motifs with positional bias relative to TSS.

Chapter 6

Conclusion

The work presented here contributed to both computer science and molecular biology. Methods and tools that we developed facilitated epigenetic studies in the human malaria parasite, which is one of the most infectious lethal diseases known to mankind. Our work encompasses different stages of the pipeline analysis of the data produced by the next generation sequencing technology: mapping the sequenced reads to a reference genome, descriptive and inferential analysis of the read coverage, and utilization of the coverage for motif discovery.

6.1 Summary of results

The throughput as well as the accuracy of the sequenced reads produced by the next generation sequencing technology has improved greatly in the past few years. Our benchmarking of the representative mapping tools provides biologists with comprehensive insights on which tools will be best to use depending on the volume of the data and its quality. As the quality of reads improves and the length of sequenced DNA increases, the advantages of tools that use the Burrow-Wheeler transform will be unmatched because these tools are capable of mapping huge amount of reads to a reference genome extremely fast. The increased length of sequenced reads will significantly affect the efficiency of the tools that use Smith-Waterman algorithm or hash index table, but will not appreciably change the performance of the tools that use the Burrow-Wheeler transform. The only drawback of the latter tools is the small number of mismatches

allowed in a read, but this limitation can be remedied with the improvement of quality of the reads produced.

Whereas most applications do not require mapping with indels or with a large number of mismatches, more specialized applications like DNA variations studies rely on efficient mapping that allows for indels and a large number of mismatches. For these applications, tools that use the Smith-Waterman algorithm are the best options at present, though their efficiency needs to be improved. For now, when the accuracy of the sequenced reads is far from ideal, the mapping tools that use a hash index table allow relatively efficient mapping with a large number of mismatches, which makes these tools the best choice for some applications.

Methylation and its regulation role in cell mechanisms is one of the most active areas of research in molecular biology. The bisulfite sodium treatment of DNA enables identification of methylated cytocines with a single base resolution. The use of next generation sequencing technologies coupled with bisulfite treatment of DNA facilitates the studies of methylation. However, existing mapping tools are not equipped to map bisulfite-treated reads efficiently. When we started working on the project with the aim of studying methylation in the malaria, there were no tools for mapping bisulfite-treated reads. We designed a new highly efficient tool for mapping bisulfite-treated reads called BRAT (described in Chapter 3) that was used to study methylation in malaria. BRAT uses specially designed binary representation of a reference genome and reads that does not increase search space and allows mapping of reads directly to the positive strand of the genome. When other tools became available for this purpose, we showed that our tool outperforms the tools for paired-end mapping of bisulfite-treated reads and is comparable with the best tool for single reads available at the time, RMAP-bs. BRAT has certain advantages over other existing tools: (1) it aligns all bases of the reads/pairs even though reads or mates of paired-end reads can have different length; (2) it offers two additional tools for end-

trimming of the low quality bases and for nucleotide count at each base of a reference genome. BRAT uses a hash index built on a reference genome, which makes the space usage independent from the amount of input reads, which can be a distinct advantage as the throughput of the next generation sequencing machines continues to rise. Since BRAT processes chromosomes sequentially, the total space used by the program is kept relatively small even with large genomes such as human genome, which makes BRAT suitable for users who cannot afford computers with large primary memory.

Mapping reads to a reference genome is only the first step in the pipeline analysis. The analysis of the resulting coverage is used to facilitate new molecular biological discoveries. In Chapter 4, we showed how coverage from FAIRE and MAINE sequencing was used to study dynamic chromatin structure remodeling of the human malaria parasite. We determined nucleosome positioning from MAINE coverage and assigned scores to nucleosomes to evaluate a nucleosome's stability in the population of cells. We analyzed the nucleosome landscape across seven time points of the erythrocytic cycle of the parasite and established that malaria's chromatin has essentially only two states, open and closed. Chromatin is open at 18 hours after blood cell invasion, and it is closed by the end of the erythrocytic cycle (36 hours after invasion). Only 26% of the malaria's genome is covered with nucleosomes at 18 hours, while 50% of the genome is occupied by nucleosomes at 36 hours. We then studied the preferences of different genomic regions to nucleosomes. Unlike yeast whose genome is mostly packed in nucleosomes (80% of the yeast's genome is occupied by nucleosomes), malaria has nucleosomes predominantly located inside genes, while promoters and downstream regions of genes are free from nucleosomes for most of the erythrocytic cycle.

In addition to descriptive analysis, we applied inferential analysis to study whether there is a correlation between FAIRE or MAINE coverage and gene expression profiles. We developed

a variety of approaches to group data to be able to apply statistical techniques to measure the correlations between different data sets. While some approaches did not show any correlation between the coverage and gene expression data, one approach showed evidence of a correlation between certain groups of genes and the degree of chromatin opening. Finally, we used FAIRE and MAINE coverage to determine putative centromeric regions in chromosomes 9, 11 and 12, which were previously unknown.

The study of dynamic chromatin structure remodeling in the human malaria parasite produced several interesting observations that were used in our project of motif discovery. In particular, we examined FAIRE coverage surrounding a validated motif from [75] and noticed that this coverage varies drastically throughout the erythrocytic cycle. Another observation was that most of the genes had a well-defined enriched FAIRE peaks within their promoters, and while the intensity of the peaks changed across the time points, the locations of the peaks corresponding to the start codons remained the same. These two observations allowed us to formulate the hypothesis that if chromatin remodeling regulates gene transcription, then the most likely regions that have motifs are the windows within the promoters with the highest variance in the FAIRE coverage. We tested this hypothesis by developing methodology for *in silico* motif discovery using FAIRE coverage. By restricting the search for motifs to 151-base windows within the promoters with FAIRE coverage with the highest variance across time points of the erythrocytic cycle, we were able to identify a total of 144 6 to 8 bases motifs together with their corresponding target gene clusters. Some of the motifs showed high similarity to previously discovered motifs, and 69 motifs showed statistically significant enrichment in at least one functional gene set. In order to strengthen our results, it would be necessary to validate the identified motifs using molecular biological assays, which we are considering at the time of writing.

Our collaborative research also contributed new knowledge on cytocine methylation in the human malaria parasite. Our tool BRAT was used to discover that *P. falciparum* is methylated with methylation patterns highly distinct from the methylation patterns of model organisms such as *Arabidopsis* or *H. sapiens*. The binary representations of a reference genome and reads that are used with BRAT make this tool flexible for adaptation to new requirements: by employing new binary logic operations, we made BRAT capable of detecting adenine methylation. The methods and tools designed in this research can be used with other organisms and can benefit epigenetic research.

6.2 Future research

Despite extensive efforts toward the development of new mapping tools, several computational challenges remain to be met. The continuous increase of throughput of the next-generation sequencing technologies necessitates that mapping tools adapt to these new capabilities. For instance, tools like SHRiMP will be extremely slow for the mapping of hundreds of millions of reads. Therefore, for DNA variation detection, new mapping tools of improved efficiency are needed.

Tools such as Bowtie are the most efficient mapping tools in terms of space and time. However, these tools currently handle only a small number of mismatches and do not handle insertions and deletions. One might profitably explore the possibility of making these tools capable of handling a larger number of mismatches as well as indels.

For methylation detection, we designed BRAT to use a hashing index table as the underlying data structure. Future research might involve improving BRAT's efficiency by using a combination of the techniques proven to be effective in speeding up the mapping. For example,

BRAT can be redesigned to work with the Burrows-Wheeler transform and keep binary representations of a genome in memory to allow a more efficient approach to handling mismatches than that used by Bowtie. This could allow a larger number of mismatches in the reads. Another enhancement of BRAT's flexibility would be to tailor it work with a wide range of operating systems. We also want to extend the range of platforms on which BRAT could run.

Other improvements are also possible for BRAT. For example, BRAT does not map ambiguous reads, but many applications involve studying correlation of methylation and genomic locations of sRNA and piRNA that often occur in repeat regions of the genome. Therefore it would be helpful to add the capability of mapping ambiguous reads. Moreover, the mapping of ambiguous reads should be done according to different strategies, some of which are the mapping according to distribution of uniquely mapped reads, or according to the distribution of BS-mismatches in the ambiguous reads, or mapping to a random location, or mapping using a more sophisticated statistical model that depends on application at hand.

Finally, BRAT currently does not include any tools for analysis of methylation. One might expand BRAT in this direction. For example, we could add a tool that analyzes methylation in different genomic regions. Another tool might analyze the correlation of methylation and nucleosomes. Other tools could apply statistical methods to infer correlation between methylation and genomic locations of piRNA. Ideally, we would like to turn BRAT into a sophisticated suite that would permit a full statistical analysis of methylation.

Future research might expand the analysis of the nucleosome landscape in malaria. The MAINE coverage described in Chapter 4 provides a unique opportunity to study nucleosomes fluctuation in a population of parasites across the erythrocytic cycle. In the future, we could develop a methodology that identifies well-positioned nucleosomes and nucleosomes whose locations are not stable within a population of parasites. This future approach could include

analysis of these positions across the erythrocytic cycle and perhaps the identification of a possible correlation between stable well-defined nucleosomes and DNA content (or methylation, or specific genomic regions).

For motif discovery using FAIRE coverage, one might look into motifs inside introns and downstream of the genes. As more data becomes available for orthologous species, a more sophisticated analysis of positional bias of motifs relative to TSS and the predicted promoters could be developed. Our method can be improved by applying a more refined selection of target genes for the motifs using DNA content and/or other available information about the genes: GO-function, gene expression profiles and TSS positions.

Bibliography

[1] I. Albert, T. N. Mavrich, L.P. Tomsho, J. Gi, S. J. Zanton, S. C. Schuster, and B. F. Pugh. Translational and rotational settings of H2A.Z nucleosomes across the *Saccharomyces cerevisiae* genome. *Nature,* 446:572–576, 2007.

[2] S. F. Altshul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403-10, 1990.

[3] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acid Research*, 25:3389-3402, 1997.

[4] A. Apostolico, M. E. Bock, and S. Lonardi. Monotony of surprise and large-scale quest for unusual words. *Journal of Computational Biology, 10*(2/3):283–311, 2003.

[5] S. Balaji, M. M. Babu, L. M. Iyer and L. Aravind. Discovery of the principal specific transcription factors of apicomplexa and their implication for the evolution of the AP2-integrase DNA binding domains. *Nucleic Acids Research, 33(*13):3994–4006, 2005.

[6] R.A. Baeza-Yates and C.H. Perleberg. Fast and practical approximate pattern matching. *Information Processing Letters*, 59:21-27, 1996.

[7] A. Barski, S. Cuddapah, K. Cui, T. Roh, D. Schones, Z. Wang, G. Wei, I. Chepelev and K. Zhao. High-resolution profiling of histone methylations in the human genome. *Cell,* 129*(4):*823–837, 2007.

[8] T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 28-36, 1994.

[9] V. Bernard, V. Brunaud, and A. Lecharny. TC-motifs at the TATA-box expected position in plant genes: a novel class of motifs involved in the transcription regulation. *BMC Genomics*, 11:166, 2010.

[10] M. Bibikova, E. Chudin, B. Wu, L. Zhou, E. W. Garcia, Y. Liu, S. Shin, T. W. Plaia, J. M. Auerbach, D. E. Arking, R. Gonzalez, J. Crook, B. Davidson, T. C. Schulz, A. Robins, A. Khanna, P. Sartipy, J. Hyllner, P. Vanguri, S. Savant-Bhonsale, A. K. Smith, A. Chakravarti, A. Maitra, M. Rao, D. L. Barker, J. F. Loring and J.-B. Fan. Human embryonic stem cells have a unique epigenetic signature. *Genome Research*, 16(9):1075–1083, 2006.

[11] E. Birney, J. A. Stamatoyannopoulos, A. Dutta, R. Guigó, T. R. Gingeras, E. H. Margulies, Z. Weng, M. Snyder, E. T. Dermitzakis, R. E. Thurman, M. S. Kuehn, C. M. Taylor, S. Neph, C.

M. Koch, S. Asthana, A. Malhotra *et al*. Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature,* 447*:* 799–816, 2007.

[12] Z. Bozdech, M. Llinas, B. L. Pulliam, E. D. Wong, J. Zhu and J. L. Derisi. The transcriptome of the intraerythrocytic developmental cycle of *Plasmodium falciparum*. *PLoS Biology,* 1(1):e5, 2003.

[13] K. Brick, J. Watanabe, and E. Pizzi. Core promoters are predicted by their distinct physicochemical properties in the genome of *Plasmodium falciparum. Genome Biology*, 9:R178, 2008.

[14] M. Burrows and D. J. Wheeler. A Block Sorting Lossless Data Compression Algorithm. Technical Report. 124 Palo Alto, CA. Digital Equipment Corporation, 1994.

[15] K. Chen, Q. Meng, L. Ma, Q. Liu, P. Tang, C. Chiu and S. Hu. A novel DNA sequence periodicity decodes nucleosome positioning. *Nucleic Acids Research*, 36(19):6228-6236, 2008.

[16] T.-T. Ching, A. K. Maunakea, P. Jun, C. Hong, G. Zardo, D. Pinkel, D. G. Albertson, J. Fridlyand, J.-H. Mao, K. Shchors, W. A. Weiss, and J. F. Costello. Epigenome analyses using BAC microarrays identify evolutionary conservation of tissue-specific methylation of SHANK3. *Nature Genetics*, 37(6):645–651, 2005.

[17] T. Chookajorn, R. Dzikowski, M. Frank, F. Li, A. Z. Jiwani, D. L. Hartl and K. W. Deitsch. Epigenetic memory at malaria virulence genes. *Proceedings of the National Academy of Sciences*, 104(3):899–902, 2007.

[18] S. J. Cokus, S. Feng, X. Zhang, Z. Chen, B. Merriman, C. D. Haudenschild, S. Pradhan, S. F. Nelson, M. Pellegrini and S. E. Jacobsen. Shotgun bisulphite sequencing of the *Arabidopsis* genome reveals DNA methylation patterning. *Nature*, 452:215–219, 2008.

[19] R. M. Coulson, N. Hall, and C. A. Ouzounis. Comparative genomics of transcriptional control in the human malaria parasite *Plasmodium falciparum*. *Genome Research*, 14(8):1548–1554, 2004.

[20] F. Cui and V. B. Zhurkin. Distinctive sequence patterns in metazoan and yeast nucleosomes: Implications for linker histone binding to AT-rich and methylated DNA. *Nucleic Acids Research*, 37(9):2818–2829, 2009.

[21] J. P. Daily, K. G. Le Roch, O. Sarr, D. Ndiaye, A. Lukens, Y. Zhou, O. Ndir, S. Mboup, A. Sultan, E. A. Winzeler and D. F. Wirth. In vivo transcriptome of *Plasmodium falciparum* reveals overexpression of transcripts that encode surface proteins. *The Journal of Infectious Diseases*, 191(7):1196–1203, 2005.

[22] K. W. Deitsch, M. S. Calderwood, and T. E. Wellems. Malaria: Cooperative silencing elements in *var* genes. *Nature*, 412:i875–876, 2001.

[23] J. E. Dodge, M. Okano, F. Dick, N. Tsujimoto, T. Chen, W. Wang, Y. Ueda, N. Dyson and E. Li. Inactivation of DNMT3b in mouse embryonic fibroblasts results in DNA hypomethylation,

chromosomal instability, and spontaneous immortalization. *Journal of Biological Chemistry*, 280(18): 17986–17991, 2005.

[24] M. T. Duraisingh, T. S. Voss, A. J. Marty, M. F. Duffy, R. T. Good, J. K. Thompson, L. H. Freitasjunior, A. Scherf, B. S. Crabb and A. F. Cowman. Heterochromatin silencing and locus repositioning linked to regulation of virulence genes in *Plasmodium falciparum*. *Cell*, 121(1):13–24, 2005.

[25] F. Eckhardt, J. Lewin, R. Cortese, V. K. Rakyan, J. Attwood, M. Burger, J. Burton, T. V. Cox, R. Davies, T. A. Down, C. Haefliger, R. Horton, K. Howe, D. K. Jackson, J. Kunde, C. Koenig, J. Liddle, D. Niblett, T. Otto, R. Pettett, S. Seemann, C. Thompson, T. West, J. Rogers, A. Olek, K. Berlin and S. Beck. DNA methylation profiling of human chromosomes 6, 20 and 22. *Nature Genetics*, 38(12):1378–1385, 2006.

[26] O. Elemento, N. Slonim, and S. Tavazoie. A universal framework for regulatory element discovery across all genomes and data types. *Molecular Cell*, 28:337–350, 2007.

[27] O. Elemento, and S. Tavazoie. Fast and systematic genome-wide discovery of conserved regulatory elements using a non-alignment based approach. *Genome Biology*, 6:R18, 2005.

[28] C. Epp, F. Li, C. A. Howitt, T. Chookajorn, and K. W. Deitsch. Chromatin associated sense and antisense noncoding RNAs are transcribed from the *var* gene family of virulence genes of the malaria parasite *Plasmodium falciparum*. *RNA*, 15(1):116–127, 2009.

[29] M. Frommer, L. E. Mcdonald, D. S. Millar, C. M. Collis, F. Watt, G. W. Grigg, P. L. Molloy and C. L. Paul. A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands. *PNAS*, 89:1827-1831, 1992.

[30] K. Ganesan, N. Ponmee, L. Jiang, J. W. Fowble, J. White, S. Kamchonwongpaisan, Y. Yuthavong, P. Wilairat and P. K. Rathod. A genetically hard-wired metabolic transcriptome in *Plasmodium falciparum* fails to mount protective responses to lethal antifolates. *PLoS Pathogen*, 4(11):e1000214, 2008.

[31] M. J. Gardner, N. Hall, E. Fung, O. White, M. Berriman, R. W. Hyman, J. M. Carlton, A. Pain, K. E. Nelson, S. Bowman, I. T. Paulsen, K. James, J. A. Eisen, K. Rutherford, S. L. Salzberg, A. Craig, S. Kyes, M.-S. Chan, V. Nene, S. J. Shallom, B. Suh, J. Peterson, S. Angiuoli, M. Pertea, J. Allen, J. Selengut, D. Haft, M. W. Mather, A. B. Vaidya, D. M. A. Martin, A. H. Fairlamb, M. J. Fraunholz, D. S. Roos, S. A. Ralph, G. I. McFadden, L. M. Cummings, G. M. Subramanian, C. Mungall, J. C. Venter, D. J. Carucci, S. L. Hoffman, C. Newbold, R. W. Davis, C. M. Fraser and B. Barrell. Genome sequence of the human malaria parasite *Plasmodium falciparum*. *Nature*, 419:498–511, 2002.

[32] P.G. Giresi, J. Kim, R. M. McDaniell, V. R. Iyer and J. D. Lieb. FAIRE (Formaldehyde-Assisted Isolation of Regulatory Elements) isolates active regulatory elements from human chromatin. *Genome Research*, 17(6):877–885, 2007.

[33] S. Gupta, J. Dennis, R. E. Thurman, R. Kingston, R., J. A. Stamatoyannopoulos, W. S. Noble and P. E. Bourne. Predicting human nucleosome occupancy from primary sequence. *PLoS Computational Biology*, 4(8):e1000134, 2008.

[34] E. Y. Harris, N. Ponts, A. Levchuk, K. Le Roch, S. Lonardi, "BRAT: Bisulfite-treated Reads Analysis Tool", *Bioinformatics*, 26(4):572-573, 2010.

[35] E. Y. Harris, N. Ponts, K. Le Roch, and Lonardi. Motif discovery in *P. falciparum* using dynamic chromatin remodeling data (in preparation).

[36] F. Hormozdiari, C. Alkan, E. E. Eichler and S. C. Sahinalp. Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Research*, 19:1270-1278, 2009.

[37] J.D. Hughes, P.W. Estep, S. Tavazoie, and G.M. Church. Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. *Journal of Molecular Biology*, 296:1205-1214, 2000.

[38] P. Iengar and N. V. Joshi. Identification of putative regulatory motifs in the upstream regions of co-expressed functional groups of genes in Plasmodium falciparum. *BMC Genomics*, 10:18, 2009.

[39] I. P. Ioshikhes, I. Albert, S. J. Zanton, and B. F. Pugh. Nucleosome positions predicted through comparative genomics. *Nature Genetics*, 38(10):1210–1215, 2006.

[40] S. M. Johnson, F. J. Tan, H. L. McCullough, D. P. Riordan, and A. Z. Fire. Flexibility and constraint in the nucleosome core landscape of *Caenorhabditis elegans* chromatin. *Genome Research*, 16(12):1505–1516, 2006.

[41] P. Jones, T. Archer, S. Baylin, S. Beck, S. Berger, B. Bernstein, J. Carpten, S. Clark, J. Costello and R. Doerge. Moving ahead with an international human epigenome project. *Nature*, 454:711–715, 2008.

[42] T. Kaplan, C. L. Liu, J. A. Erkmann, J. Holik, M. Grunstein, P. D. Kaufman, N. Friedman, O. J. Rando and B. V. Steensel. Cell cycle– and chaperone-mediated regulation of H3K56ac incorporation in yeast. *PLoS Genetics*, 4(11):e1000270, 2008.

[43] W. J. Kent. BLAT -- The BLAST-Like Alignment Tool. *Genome Research*, 4:656-664, 2002.

[44] B. Langmead, C. Trapnell, M. Pop and S. L. Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biology*, 10:R25, 2009.

[45] K. Le Roch, J. Johnson, H. Ahiboh, D.-W. Chung, J. Prudhomme, D. Plouffe, K. Henson, Y. Zhou, W. Witola, J. Yates, C. Mamoun, E. Winzeler and H. Vial. A systematic approach to understand the mechanism of action of the bisthiazolium compound T4 on the human malaria parasite, *Plasmodium falciparum*. *BMC Genomics*, 9(1):513, 2008.

[46] K. Le Roch, Y. Zhou, P. L. Blair, M. Grainger, J. K. Moch, J. D. Haynes, P. D. L. Vega, A. A. Holder, S. Batalov, D. J. Carucci and E. A. Winzeler. Discovery of gene function by expression profiling of the malaria parasite life cycle. *Science*, 301:1503–1508, 2003.

[47] W. Lee, D. Tillo, N. Bray, R. H. Morse, R. W. Davis, T. R. Hughes and C. Nislow. A high-resolution atlas of nucleosome occupancy in yeast. *Nature Genetics*, 39(10):1235–1244, 2007.

[48] B. Li, M. Carey and J. Workman. The role of chromatin during transcription. *Cell*, 128(4):707–719, 2007.

[49] R. Li, Y. Li, K. Kristiansen and J. Wang. SOAP: short oligonucleotide alignment program. *Bioinformatics*, 24(5):713-714, 2008.

[50] R. Lister and J. R. Ecker. Finding the fifth base: Genome-wide sequencing of cytosine methylation. *Genome Research*, 19(6):959–966, 2009.

[51] R. Lister, R. O'Malley, J. Tonti-Filippini, B. Gregory, C. Berry, A. Millar and J. Ecker. Highly integrated single-base resolution maps of the epigenome in *Arabidopsis*. *Cell*, 133(3):523–536, 2008.

[52] R. Lister, M. Pelizzola, R. H. Dowen, R. D. Hawkins, G. Hon, J. Tonti-Filippini, J. R. Nery, L. Lee, Z. Ye, Q. Ngo, L. Edsall, J. Antosiewicz-Bourget, R. Stewart, V. Ruotti, A. H. Millar, J. A. Thomson, B. Ren and J. R. Ecker. Human DNA methylomes at base resolution show widespread epigenomic differences. *Nature*, 462:315-322, 2009.

[53] C. L. Liu, T. Kaplan, M. Kim, S. Buratowski, S. L. Schreiber, N. Friedman and O. J. Rando. Single-nucleosome mapping of histone modifications in *S. cerevisiae*. *PloS Biology*, 3(10):e328, 2005.

[54] M. Llinas, Z. Bozdech, E. D. Wong, A. T. Adai, and J. L. Derisi. Comparative whole genome transcriptome analysis of three *Plasmodium falciparum* strains. *Nucleic Acids Research*, 34(4):1166–1173, 2006.

[55] J. J. Lopez-Rubio, A. M. Gontijo, M. C. Nunes, N. Issar, R. H. Rivas and A. Scherf. 5' flanking region of *var* genes nucleate histonemodification patterns linked to phenotypic inheritance of virulence traits in malaria parasites. *Molecular Microbiology*, 66(6):1296–1305, 2007.

[56] K. D. MacIsaac and E. Fraenkel. Practical strategies for discovering regulatory DNA sequence motifs. *PLoS Computational Biology*, 2(4):201-210, 2006.

[57] E. R. Mardis. The impact of next-generation sequencing technology on genetics. *Trends in Genetics*, 24(3):133-141, 2008.

[58] T. N. Mavrich, I. P. Ioshikhes, B. J. Venters, C. Jiang, L. P. Tomsho, J. Qi, S. C. Schuster, I. Albert and B. F. Pugh. A barrier nucleosome model for statistical positioning of nucleosomes throughout the yeast genome. *Genome Research*, 18(7):1073–1083, 2008.

[59] T. N. Mavrich, C. Jiang, I. P. Ioshikhes, X. Li, B. J. Venters, S. J. Zanton, L. P. Tomsho, J. Qi, R. L. Glaser, S. C. Schuster, D. S. Gilmour, I. Albert and B. F. Pugh. Nucleosome organization in the drosophila genome. *Nature*, 453:358–362, 2008.

[60] A. Meissner. Reduced representation bisulfite sequencing for comparative high-resolution DNA methylation analysis. *Nucleic Acids Research*, 33(18):5868–5877, 2005.

[61] E. Parzen. On estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33:1065–1076, 1962.

[62] G. Pavesi, P. Mereghetti, G. Mauri, and G. Pesole. Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Research*, 32:W199-W203, 2004.

[63] H. E. Peckham, R. E. Thurman, Y. Fu, J. A. Stamatoyannopoulos, W. S. Noble, K. Struhl and Z. Weng. Nucleosome positioning signals in genomic DNA. *Genome Research*, 17(8):1170–1177, 2007.

[64] K. Perez-Toledo, A. P. Rojas-Meza, L. Mancio-Silva, N. A. Hernandez-Cuevas, D. M. Delgadillo, M. Vargas, S. Martinez-Calvillo, A. Scherf and R. Hernandez-Rivas. *Plasmodium falciparum* heterochromatin protein 1 binds to tri-methylated histone 3 lysine 9 and is linked to mutually exclusive expression of *var* genes. *Nucleic Acids Research*, 37(8):2596–2606, 2009.

[65] N. Ponts, E. Y. Harris, J. Prudhomme, I. Wick, C. Eckhardt-Ludka, G. R. Hicks, G. Hardiman, S. Lonardi, K. G. Le Roch. Nucleosome landscape and control of transcription in the human malaria parasite. *Genome Research*, 20(2):228-238, 2010.

[66] V. K. Rakyan, T. Hildmann, K. L. Novik, J. Lewin, J. Tost, A. V. Cox, T. D. Andrews, K. L. Howe, T. Otto, A. Olek, J. Fischer, I. G. Gut, K. Berlin and S. Beck. DNA methylation profiling of the human major histocompatibility complex: A pilot study for the human epigenome project. *PLoS Biology*, 2(12):e405, 2004.

[67] S. Reynolds, J. Bilmes and W. Noble. On the relationship between DNA periodicity and local chromatin structure. *Proceedings of the International Conference on Research in Computational Molecular Biology (RECOMB)*, LNCS 5541:434–450, 2009.

[68] S. C. Satchwell, H. R. Drew, and A. A. Travers. Sequence periodicities in chicken nucleosome core DNA. *Journal of Molecular Biology*, 191(4):659-675, 1986.

[69] D. Schones, K. Cui, S. Cuddapah, T. Roh, A. Barski, Z. Wang, G. Wei, and K. Zhao. Dynamic regulation of nucleosome positioning in the human genome. *Cell*, 132(5):887–898, 2008.

[70] D. E. Schones and K. Zhao. Genome-wide approaches to studying chromatin modifications. *Nature Reviews Genetics*, 9(3):179–191, 2008.

[71] E. Segal, Y. Fondufe-Mittendorf, L. Chen, A. Thåström, Y. Field, I. K. Moore, J.-P. Z. Wang and J. Widom. A genomic code for nucleosome positioning. *Nature*, 442:772–778, 2006.

[72] M. R. Segal. Re-cracking the nucleosome positioning code. *Statistical Applications in Genetics and Molecular Biology*, 7 (Article 14) 2008.

[73] S. Shivaswamy, A. Bhinge, Y. Zhao, S. Jones, M. Hirst and V. R. Iyer. Dynamic remodeling of individual nucleosomes across a eukaryotic genome in response to transcriptional perturbation. *PLoS Biology*, 6(3): e65, 2008.

[74] S. Shivaswamy and V. R. Iyer. Stress-dependent dynamics of global chromatin remodeling in yeast: Dual role for SWI/SNF in the heat shock stress response. *Molecular and Cellular Biology*, 28(7):2221–2234, 2008.

[75] E. K. De Silva, A. R. Gehrke, K. Olszewski, I. Leon, J. S. Chahal, M. L. Bulyk, and M. Llinas. Specific DNA-binding by Apicomplexan AP2 transcription factors. *PNAS*, 105(24):8393–8398, 2008.

[76] A. D. Smith, W. Chung, E. Hodges, J. Kendall, G. Hannon, J. Hicks, Z. Xuan, and M. Q. Zhang. Updates to the RMAP short-read mapping software. *Bioinformatics*, 25(21*)*:2841-2842, 2009.

[77] T. E. Smith and M.S. Waterman. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147:195-197, 1981.

[78] A. D. Smith, Z. Xuan and M. Q. Zhang. Using quality scores and longer reads improves accuracy of Solexa read mapping. *BMC Bioinformatics*, 9(1):128, 2008.

[79] K. H. Taylor, R. S. Kramer, J. W. Davis, J. Guo, D. J. Duff, D. Xu, C. W. Caldwell and H. Shi. Ultradeep bisulfite sequencing analysis of DNA methylation patterns in multiple gene promoters by
454 sequencing. *Cancer Research*, 67(18):8511–8518, 2007.

[80] E. N. Trifonov and J. L. Sussman. The pitch of chromatin DNA is reflected in its nucleotide sequence. *PNAS*, 77(7):3816–3820, 1980.

[81] A. Valouev, J. Ichikawa, T. Tonthat, J. Stuart, S. Ranade, H. Peckham, K. Zeng, J. A. Malek, G. Costa, K. McKernan, A. Sidow, A. Fire and S. M. Johnson. A high-resolution, nucleosome position map of *C. elegans* reveals a lack of universal sequence-dictated positioning. *Genome Research*, 18(7):1051–1063, 2008.

[82] Watanabe, M. Sasaki, Y. Suzuki, and S. Sugano. FULL-malaria: a database for a full-length enriched cDNA library from human malaria parasite, *Plasmodium falciparum*. *Nucleic Acids Research*, 29:70-71, 2001.

[83] Watanabe, Y. Suzuki, M. Sasaki, and S. Sugano. Full-malaria 2004: an enlarged database for comparative studies of full-length cDNAs of malaria parasites, *Plasmodium* species. *Nucleic Acids Research*, 32:D334-338, 2004.

[84] I. C. G. Weaver, N. Cervoni, F. A. Champagne, A. C. D'Alessio, S. Sharma, J. R. Seckl, S. Dymov, M. Szyf and M. J. Meaney. Epigenetic programming by maternal behavior. *Nature Neuroscience*, 7(8):847–854, 2004.

[85] H. R. Widlund, H. Cao, S. Simonsson, E. Magnusson, T. Simonsson, P. E. Nielsen, J. D. Kahn, D. M. Crothers and M. Kubista. Identification and characterization of genomic nucleosomepositioning sequences. *Journal of Molecular Biology*, 267(4):807–817, 1997.

[86] A. P. Wolffe, P. L. Jones and P. A. Wade. DNA demethylation. *PNAS*, 96(11): 5894–5896, 1999.

[87] J. Wu, D. H. Sieglaff, J. Gervin, and X. S. Xie. Discovering regulatory motifs in the Plasmodium genome using comparative genomics. *Bioinformatics*, 24(17):1843–1849, 2008.

[88] Y. Xi and W. Li. BSMAP: whole genome bisulfite sequence MAPping program. *BMC Bioinformatics*, 10:232-240, 2009.

[89] X. Xie, J. Lu, E. J. Kulbokas, T.R. Golub, V. Mootha, K. Lindblad-Toh, E.S. Lander, and M. Kellis. Systematic discovery of regulatory motifs in human promoters and 3' UTRs by comparison of several mammals. *Nature*, 434:338–345, 2005.

[90] V. Yanovski, S. M. Rumble and M. Brudno. Read mapping algorithms for single molecule sequencing data. In *Proceedings of Algorithms in Bioinformatics, WABI*, 5251:38-49, 2008.

[91] J. A. Young, J. R. Johnson, C. Benner, S. F. Yan, K. Chen, K. G. Le Roch, Y. Zhou, and E. A. Winzeler. *In silico* discovery of transcription regulatory elements in Plasmodium falciparum. *BMC Genomics*, 9:70, 2008.

[92] M. Zeschnigk, M. Martin, G. Betzl, A. Kalbe, C. Sirsch, K. Buiting, S. Gross, E. Fritzilas, B. Frey, S. Rahmann, and B. Horsthemke. Massive parallel bisulfite sequencing of CG-rich DNA fragments reveals that methylation of many X-chromosomal CpG islands in female blood DNA is incomplete. *Human Molecular Genetics*, 18(8):1439-1448, 2009.

[93] G. C. Yuan and J. S. Liu. Genomic sequence is highly predictive of local nucleosome depletion. *PloS Computational Biology*, 4(1):e13, 2008.

[94] G. C. Yuan, Y. J. Liu, M. F. Dion, M. D. Slack, L. F. Wu, S. J. Altschuler and O. J. Rando. Genome-scale identification of nucleosome positions in *S. cerevisiae*. *Science*, 309:626–630, 2005.

[95] F. Zhang, J. H. Pomerantz, G. Sen, A. T. Palermo and H. M. Blau. Active tissue-specific DNA demethylation conferred by somatic cell nuclei in stable heterokaryons. *PNAS*, 104(11):4395–4400, 2007.