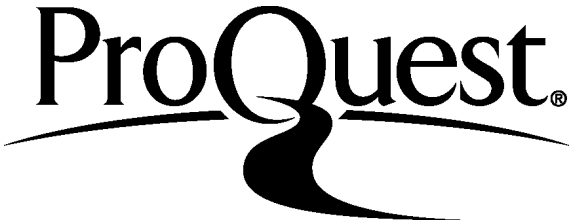
BISSEQ: A REDUCED REPRESENTATION BISULFITE SEQUENCING SIMULATION AND ANALYSIS TOOL

by

Yubo Xu

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Master of Science in Bioinformatics and Computational Biology

Spring 2016



BISSEQ: A REDUCED REPRESENTATION BISULFITE SEQUENCING SIMULATION AND ANALYSIS TOOL

by

Yubo Xu

Approved: _____
Adam G. Marsh, Ph.D.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: _____
Kathleen F. McCoy, Ph.D.
Chair of the Department of Computer and Information Sciences

Approved: _____
Babatunde A. Ogunnaike, Ph.D.
Dean of the College of College of Engineering

Approved: _____
Ann L. Ardis, Ph.D.
Senior Vice Provost for Graduate and Professional Education

ACKNOWLEDGMENTS

I would like to first thank my thesis advisor, Dr. Adam Marsh for seeing the potential in me and taking me into his lab to work on such an interesting project. I would also like to thank him for providing continuous guidance and support on my project. It has been a great pleasure working with him.

My genuine gratitude goes to my committee members and DBI colleagues for advising on my thesis project. Special thanks to Karol Miaskiewicz, Hongzhan Huang for their help in deploying tasks on Biohen server. This thesis would not be possible without their patience and kindness.

I must also thank Yiyan Zhang, Kehui Zhang, Chaoyu Chen, Shiyi Chen, Chen Peng, and all my friends, who have supported and helped me throughout my master study. Thanks you all for inspiration through your encouragement, sense of honesty, selflessness and perseverance.

This manuscript is dedicated to my parents, Jian Xu and Hua Zhang for their unconditional love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Reduced representation bisulfite sequencing is gaining popularity among researchers who focus on epigenetics. But with an ever-increasing availability of downstream mapping software, no clear standard has been established, hence impairing confidence of experimental results. Simulation of NGS data coupled with software performance analysis provides an alternative way.

BisSeq is a new next-generation sequencing simulator capable of generating single-end reduced representation bisulfite sequencing reads in FASTA format. BisSeq allows users to configure different sequencing parameters, facilitating various research purposes. BisSeq integrates data visualization methods to help researchers assess the performance of sequencing read aligners after bisulfite conversion of cytosine's to thymidines. Currently BisSeq supports simulation against single genome. We demonstrate BisSeq's value by using Bismark to map simulated sequencing reads. Working with reads simulated with different read length; we profiled performance of BSseeker2, BSMAP, Bismark using a comparison metric. BisSeq provides researchers with a good tool to benchmark reads mapping tools and to identify appropriate parameter values for experimental design.

.

Chapter 1

INTRODUCTION

With advancements in recent epigenetic research and Next-Generation-Sequencing (NGS) (Metzker et al., 2010) technology, more and more pharmaceutical companies gradually realized that epigenetics could be the place where next generation diagnosis and cure begins. A noteworthy fact is the number of patients diagnosed as negative for mutations in well-studied disease-causing genes has been increasing, which indicates that traditional diagnostic approaches have limited power. A novel diagnosis and cure method is in urgent need at this same time. Also, traditional genotyping cannot help much in diagnosing process. It has been suggested that epigenetic modification could be a potential contributor to these diseases.

Different from genetic mutation that involves change of DNA sequence that may take generations to come into effect, epigenetic mechanisms act in a comparatively more transient way that does not change genomic sequence. Two important components of epigenetic modifications are DNA methylation and histone acetylation (Fig 1). DNA methylation could prevent transcription by silencing promoter-binding activity, and its role in disease formation process has become more and more evident (Fig 2). Specific methyl binding protein will bind to methyl cytosine within promoter area and then recruit transcriptional repressive complex to retain a negative transcriptional status (Webb et al., 2001). Given the importance of DNA methylation, a comprehensive and precise profile of methylation status (Chatterjee et al., 2012) may enable researchers to identify epigenetic markers to support diagnosis

of diseases and make personal treatment possible. Nowadays, there are many technologies available for scientists to choose from, including: Whole genome bisulfite sequencing (WGBS) (Mill et al., 2006, Bibikova et al., 2010), reduced representation bisulfite sequencing (RRBS) (Cokus et al., 2008, Choi et al., 2015, Gu et al., 2010, Harris et al., 2010), infinium methylation microarray (Moran et al., 2014), methylated DNA immunoprecipitation (MeDIP) (Mohn et al., 2009), MeDIP and high-throughput sequencing (MeDIP-seq) (Taiwo et al., 2012). Among these techniques, Reduced representation bisulfite sequencing (RRBS) is usually chosen to analyze clinical sample, given that it only requires relatively low amount of sample input while clinical sample are usually hard to acquire. Moreover, RRBS is costly efficient, costing $400-500 compared to WGBS, which usually cost $5000-7000 to generate 50-fold coverage. After sequencing reads generated, RRBS requires software to align them to bisulfite-converted reference genome or vice versa. There are many bisulfite sequencing mapping programs for scientist to choose from, which will be discussed in more detail later. However, none of them can achieve result accuracy and time efficiency at the same time.

To facilitate methylation profiling in a more efficient and accurate way, Dr. Marsh developed a proprietary algorithm to map bisulfite-sequencing reads and to determine cytosine methylation rate. As a pilot project, this work aims to develop a pipeline, BisSeq, to generate simulated bisulfite-sequencing reads, and then evaluate bisulfite-sequencing mapping tools by using these data. Given the popularity of RRBS within the research community, we build our simulation based on RRBS protocol. The architecture of the pipeline can be divided into several steps, (1) The reads simulation part, users choose their own reference genome to build simulated sequencing reads on.

(2) Tune sequencing parameters to user desired value. (3) Connect selected mapping program to analyze simulated reads. (4) Evaluate mapping results by comparing methylation rate to preset methylation rate.

In the following, we will first describe RRBS technique, and some popular programs applied to analyze RRBS data. Then we will present detail about how BisSeq is build and its underlying algorithm. Finally, a case study using human chromosome 1 sequence will be presented. In this case, we show how BisSeq can be applied to fit into various research frameworks by plug in Bismark (Krueger et al., 2011), a RRBS mapping program to BisSeq and evaluate its performance. Another use case is to study read length effect on different mapping tools, we evaluate performance of BSMAP (Xi et al., 2009), BSseeker2 (Guo et al., 2013), Bismark mapping simulated sequencing reads with various read length.

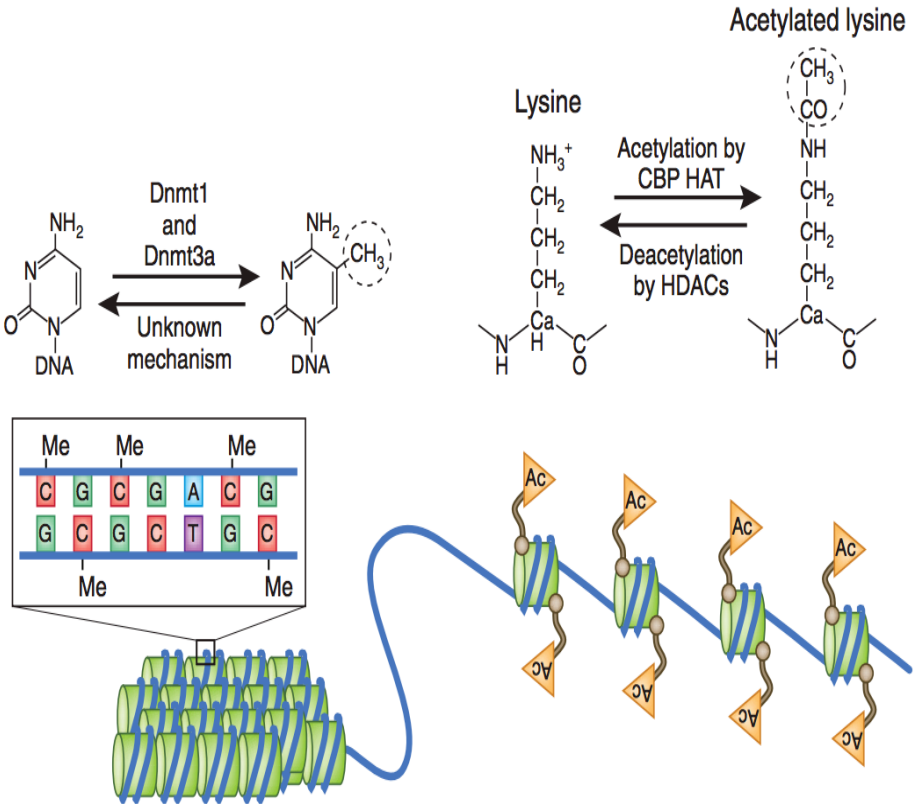


Figure 1 DNA methylation and histone acetylation are two critical epigenetic mechanisms controlling chromatin structure and function in postmitotic mammalian neurons. Hypermethylated DNA recruits silencing transcription chromatin remodeling complexes with histone deacetylases (HDACs) and promotes chromatin condensation. Hypomethylated DNA unfolds into a 'beads-on-a-string' structure in which histones are accessible for chromatin remodeling factors such as CREB-binding protein histone acetyltransferase (CBP HAT), the transcriptional coactivator implicated in epigenetic mechanisms controlling memory consolidation3. Ac, acetyl group; methyl group.(Korzus et al., 2010)

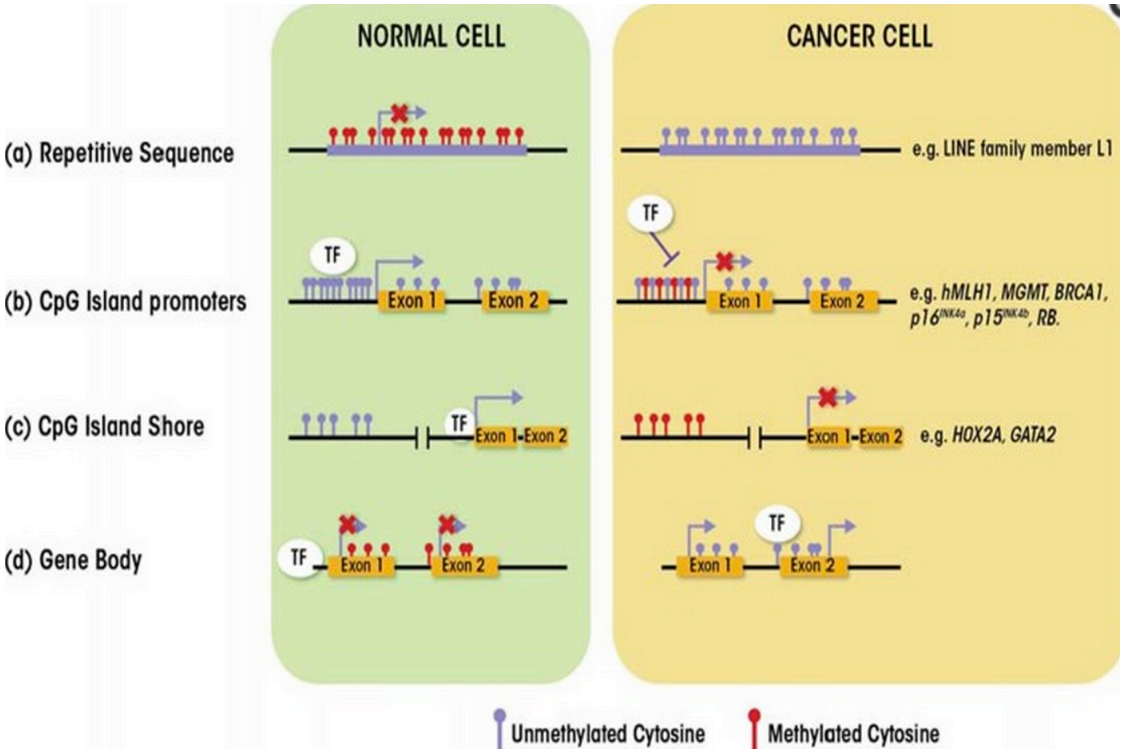


Figure 2 DNA methylation patterns in normal and cancer cells. (a) Repetitive sequences generally are methylated at cytosine nucleotides in normal cells. Global loss of methylation in cancer cells leads to chromosomal instability and activation of endoparasitic sequences. (b) CpG islands in promoter sequences typically are unmethylated in normal cells whereas they can become hypermethylated in cancer cells, leading to transcriptional repression. Examples of genes affected are shown on the right. (c) Similar patterns are seen in CpG island shores, located in front (i.e., upstream) of promoters. (d) CpGs located in gene bodies frequently are methylated in normal cells; this pattern is reversed in cancer cells, leading to initiation of transcription at several incorrect sites (Marta et al., 2013).

Chapter 2

RELATED WORK

2.1 Reduced Representation Bisulfite Sequencing

Previous study revealed that different genetic pathways control various types of methylation. A comprehensive map of methylation at single base pair resolution across the genome could provide directions for researchers to better understand the details of epigenetic mechanisms. Multiple methods have been developed to study the distribution of 5-methylcytosine across whole genome. They can be divided into two main categories. The method based off bisulfite conversion before sequencing: WGBS and RRBS. Another method is affinity purification based approach, which is less popular compared to previous one. While delivering single base resolution information about cytosine methylation, WGBS often requires a large volume of genomic analysis, which is not very feasible for some clinical cases and result in high cost. RRBS (Sun et al., 2015, Bentley et al., 2008) only interrogates a portion of original genome that is heavily methylated (Fig 3).

As the RRBS protocol shown in Fig 3, after genomic DNA has been extracted from cell or tissue sample, the first step is to digest with DNA restriction enzyme (Wang et al., 2013, Cokus et al., 2008). There are two reasons for doing so: first, by cleaving genomic DNA into fragments, consequent electrophoresis can be used to perform the size selection and extract those fragments of research interest, usually the size range used is the range within which exist most promoter segments, the length range is usually pre-calculated by *in silico* enzyme restriction treatment and could

adjusted to fit different experimental need. The second purpose is to expose the CpG sites to the end of the fragments, so the CpG sites would more likely to be detected by sequencing and improve sequencing data quality in terms of how many CpG sites are recovered. Now the most commonly used restriction enzyme is MspI, which cut at “CCGG” sites and leave a "CGG" overhang. To prevent the cohesive ends from annealing back with each other, the following step after enzyme treatment is to repair the sticky end and addition of an adenine overhang. The adenine overhang will be used to connect with primers for next step PCR. After ends have been repaired, the adapters for PCR amplification will be added to the adenine overhang. Then critical step in RRBS is the bisulfite conversion. The sodium bisulfite will efficiently deaminate unmethylated cytosine to uracil without affecting 5-methyl cytosine. After that size-selected fragments are equipped with end adapters, denatured and treated with bisulfite to convert all unmethylated cytosine to uracil. Then these fragments are cloned into vector plasmid for sequencing, and go through PCR to amplify their enrichment.

The last step is to map the sequencing reads to the reference genome and extract methylation information from the mapping results. In the following section, we will introduce some popular bisulfite sequencing mapping tools, their advantages and drawback.

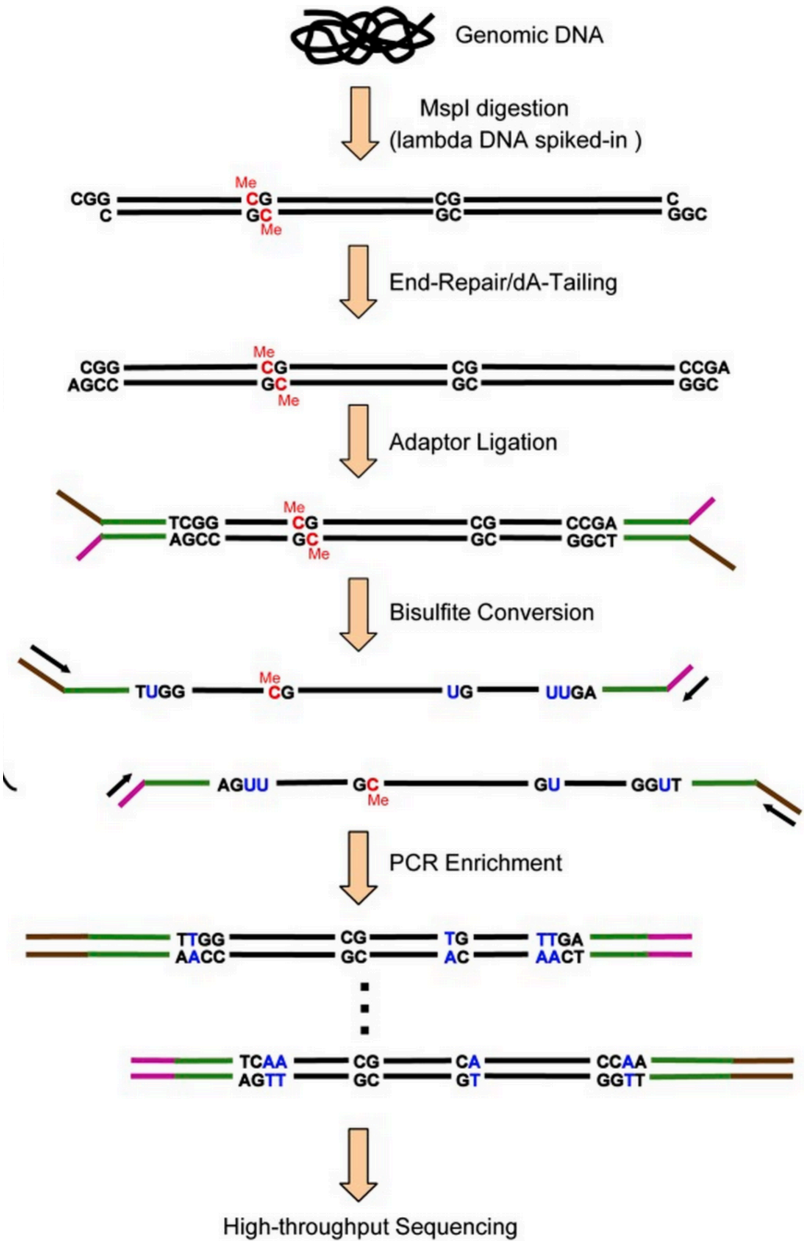


Figure 3 A schematic of the single-cell RRBS (RRBS) technique. 1) lysis of an individual cell, 2) release of the naked double-stranded genomic DNA, 3) spiking with lambda DNA, 4) digestion of the genomic DNA using a restriction enzyme, 5) end-repair and dA-tailing of the DNA fragments, 6) ligation of the adaptors to the DNA fragments, and 7)bisulfite conversion of the ligated DNA (Guo et al., 2013).

2.2 Bisulfite Sequencing Reads Analysis Tool

To acquire a correct inference of methylation status, a tool that can map short reads accurately and efficiently is very desirable. There are lot of tools (Tárraga et al., 2015, Xi et al., 2011, Fonseca et al., 2012) have been developed to tackle this challenge including BSMAP, Bismark, BS-Seeker2. Most of these tools perform some kind of conversion at the very beginning of the mapping process (e.g., Cs to Ts and Gs to As) either on the short reads or the reference genome sequence, or both and then use existing regular aligners such as Bowtie, Bowtie2 (Langmead et al., 2009), BLAT (Hancock et al., 2004) to map short reads to reference genome. Based on the underlying index algorithms, they can be categorized into two groups: Burrow-wheeler transform, and hash table (Fig 4).

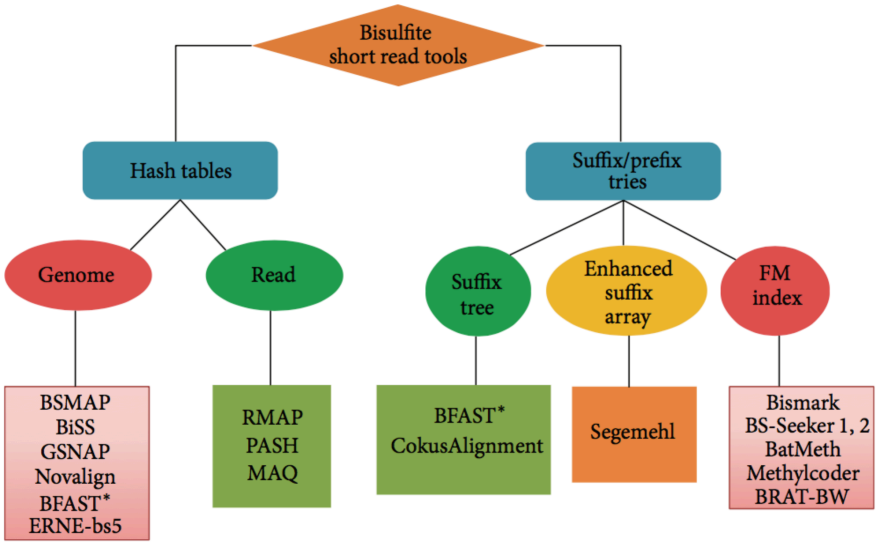


Figure 4 Bisulfite mapping tools classification. The tools can be divided into two groups based on indexing strategies: hash tables or suffix/prefix tries. Each of the groups is classified further into subgroups where some example programs are shown. BFAST uses multiple index strategies: both hashing and suffix tree.

2.2.1 Bismark

Bismark (Krueger et al., 2011) is a methylation profiling software based on indexing using Burrow-Wheeler transform. Essentially, it converts the inexact string matches to exact matching problem. The algorithm is composed of two sections: 1) identifying exact matches, 2) building inexact alignments supported by it. There are multiple choices for searching exact matches in suffix/prefix tries: suffix tree, enhanced suffix array, FM-index, Bismark utilizes FM-index. Given the fact that there will be four DNA strands to be analyzed after bisulfite sequencing, determine the strand origin of a bisulfite read could be a challenge. Bismark tackles this challenge in an effective way. First, It transforms bisulfite reads into a C-to-T and G-to-A version (equivalent to a C-to-T conversion on the reverse strand). After that, each of them is aligned to equivalently pre-converted reference genome using four parallel Bowtie/Bowtie2 process. Bowtie starts by building an FM-index for the reference genome and uses the modified FM index to the matching location. Bowtie2 is designed to support reads longer than 50bps. This feature enables Bismark to uniquely identify strand origin for each read, hence distinguish itself from other software. In addition to mapping reads to reference, Bismark directly produces methylation status of each cytosine position, saving bench scientists a lot of time post-processing mapping data. Bismark also enables methylation analysis in different sequence context by discriminating cytosines in CpG, CHG, and CHH context.

2.2.2 BSMAP

BSMAP (Xi et al., 2009) is a C++ application based on Short Oligonucleotide Alignment Program (SOAP) (Li et al., 2008)aligner. One challenge comes with the nature of RRBS is the asymmetry of C-to-T mapping, the Ts in bisulfite read can be

mapped to C/Ts in reference genome, whereas not vice versa. A common approach to tackle this issue is to convert all Cs to Ts and then map converted reads to equivalently converted reference. Post-processing is needed to calculate false-positive bisulfite C/T alignments for mismatches. This might be feasible and could do well for reads from C-poor strands, but when processing with reads from G-poor strands, where all the Cs are actually transcribed from Gs by PCR, it is not appropriate to do so. Also, ignoring C/T asymmetry will generate large number of false-positive bisulfite mappings and would significantly increase the computational load in a quadratic manner when working against large reference genome. To work out this bottleneck, BSMAP masks Ts in bisulfite reads as Cs, only at C positions in the original reference while keeping all other Ts in the bisulfite reads unchanged. So using bitwise masking, the asymmetric C/T conversion is achieved, which is very fast. In addition, it indexes reference genome for a series of k-mer seeds using a more efficient hash table. The seed length and patterns are also adjustable to allow different mismatches.

As to which tool is better, there are two evaluation criteria to consider: CPU running time and mapping efficiency. Mapping efficiency is determined by the number of short reads that have been uniquely identified divided by the total number of reads, and the CPU running time basically is the time a tool needed to finish a mapping job. According to existing tests, Bismark has the highest mapping efficiency, at the same time it need longer time to finish the job. This unbalanced performance suggests that there could be a tradeoff between mapping efficiency and CPU running time. Despite the performance indicators, appropriately preprocessing data before mapping can help increase the mapping efficiency regardless of what tools are chose. Also, adjusting parameters within tools can affect the mapping results.

2.3 Sequencing Reads Simulator

NGS technologies, Illumina Sequencing by Synthesis, Roche/454 GS FLX, produces large volume of data. Increasing availability of these large volume data opens more opportunities for researchers. For example, deep sequencing and metagenomic sequencing has made it possible to study rare variants in viral population. Whereas mining meaningful information from sequencing data could be very difficult due to the error rates associated with NGS. Separating true variants from sequencing errors remains challenging. In addition to that, it is difficult to select appropriate analysis software since there are more and more software available now. So NGS data simulation (Xi et al., 2011) combined with downstream software benchmarking is needed. Here we introduce three packages aimed to generate NGS reads, underlying different simulation models of them, also their advances and drawbacks. It's noteworthy that none of these tools are developed to generate RRBS reads, leaving this area blank.

2.3.1 Wgsim

Comes with SamTools (Li et al., 2009) - the widely used sequencing alignment tool, wgsim is among the few tools available early for sequencing reads simulation. However, it only supports a uniformly increasing error rate, while NGS normally generates with heterogeneous error profiles.

2.3.2 ART

Equipped with different models for all three main sequencing platforms, ART (Huang et al., 2012) simulates both single-end and paired-end sequencing reads off 454 (Balzer et al., 2010), Illumina, and SOLiD. It features built-in, platform-specific read error models and base quality value profiles, which are parameterized empirically

from large sequencing datasets. Although, ART comes with technology-specific read-error profiles, it still retains the flexibility to take user-supplied configuration and generate sequencing data with various read length and error characteristics. Illumina sequencing by synthesis operates in base-by-base style, where each base are determined when they are incorporated into growing DNA template that complementary to template. Hence the error model for Illumina is mainly substitution. ART simulates substitutions based on an empirically, position-dependent base quality distribution, the mean quality score decreases as the base position increases. Aside from substitution, ART simulates insertion and deletion based on empirical model derived from their training datasets. Roche/454 tests the presence of A, T, G, C in cyclical fashion, where results are produced as intensity signal based on number of incorporated based in a single cycle. Hence the error model for it is indel resulting from base over- or under-call. Given that sequencing error rarely changes with increasing flow cycle for 454 sequencing, ART adopt the empirical model where error profile are based on homopolymer length-dependent base over- or under-call. ABI's SOLiD is a relatively outdated sequencing platform, whereas ART still support simulation of it by generating nucleotide transition color, where distribution of DNA fragment size is determined by gaussian distribution. As to data output, ART can generate simulated reads in FASTA formal, and alignment in the ALN format. Also ART can output alignment in SAM format or UCSD BED file format. All in all, implemented in C++, ART is optimized with specific algorithms for different sequencing platform and is highly efficient in read simulation.

2.3.3 GemSIM

GemSIM (McElroy et al., 2012) is a python application with a command line interface. It consists of four components: GemErr, GemHaps, GemReads, and GemStats. Similar to ART, GemSIM supports most of mainstream sequencing platform by using empirically derived fragment length model and error distribution profile. In addition to that, GemSIM also can generate simulation reads from several reference genomes, which makes it possible to simulate deep sequencing, metagenomic, and resequencing projects.

Chapter 3

METHOD

3.1 Overview of The Pipeline

Simulation of bisulfite sequencing usually appears in publication as pilot assay part, but it could be very valuable to have a comprehensive pipeline that fulfill multiple tasks from sequencing reads generation to analysis, which makes experimental design more convenient and feasible.

BisSeq fills this need. As shown in Figure 5, it can be divided into three steps in general. First, users will configure parameters to be used for simulation. Currently we support a range of parameters that are essential to sequencing experiment, which including read length, genome copy number, seqCycles. Since this project is still in pilot stage, our pipeline does not support sequencing error models, which we'll discuss in more detail later. With user-set parameters, sequencing reads are produced based on reference genome. Then simulated data can be used for downstream mapping and analysis. Although BisSeq does not map sequencing reads itself, it allows user to plugin an aligner to do the mapping. Currently, users can choose from BSMAP, BSseeker2, and Bismark. Also BisSeq provides functions to analyze and visualize mapping results. Throughout the workflow of BisSeq, a log file system will record running time parameters and necessary statistics to aid the final methylation profiling analysis. In case of system failure, these log files may also help to find out the break point and recover the task from there. Some prerequisites of BisSeq are listed as following:

Pypy: 2.7.9,

Bismark: v0.14.5,

BSMAP: 2.90

BSseeker2: 2.0.3

Bowtie2: 2.2.6,

Samtools: v1.4

Perl: v5.18.2

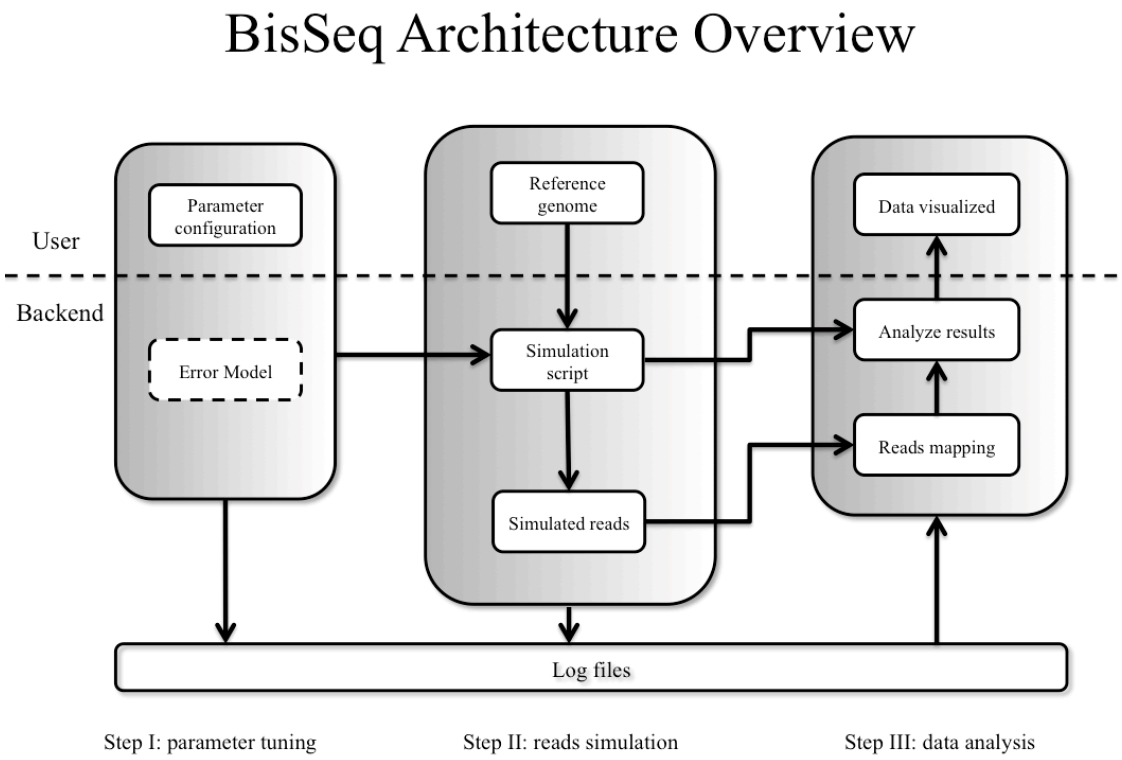


Figure 5 Overview of the BisSeq architecture

3.2 Configure Simulation Parameters

Before simulation actually starts, users need to configure the parameters first. Starting with whether to use existing reference genome, the value of "ReadGenFile" is set to 0 if user chooses to simulate reference genome from *de novo*. Otherwise, a genome will be loaded from the directory where "LoadGenome" specified. "genCopyNum" defines how many genome copies will be used to form a population with certain methylation pattern. "fracBIS" set the fraction of bisulfite conversion rate.

Parameter Name	Description
TAG	Unique Id string for folders/files
gMBsize	Set the genome size to be generated if choose to simulate reference genome
genCopyNum	Number of genome copies (each with methylation patterns)
seqCycles	Number of sequencing cycles (depth of fragment sampling)
readLen	Sequencing read length (bp)
simRefGenFolder	Specify working project of BisSeq
ReadGenFile	Decide whether to generate reference to use or load existing
fracBIS	Percent efficiency of bisulfite converting reaction
fastQ	Whether use Fasta file. 0=fasta, 1=fastQ with phred score
coinToss	Probability of a fragment in sample population to be sequenced
LoadGenome	Location for reference genome to load if ReadGenFile is set to 1

Table 1 Parameters that users need to configure before run simulation

3.3 Simulation of RRBS Reads

As indicated by diagram in Figure 5, simulation is the core step of the whole pipeline. A python script generates the simulated reads. It can be further divided into several steps: 1) load input genome and initiate global variables, 2) generate differentially methylated genome population, 3) *in silico* restriction enzyme digestion, 4) creating sequencing read tags, 5) output reads to a fasta file and statistics table about simulation.

At the very beginning of the python script, it will first read in all the user-set parameters. If user set the "ReadGenFile" to 0, a reference genome will be simulated, using the build-in genome simulator. It is implemented in a base-by-base fashion with the genome size and nucleotide percentage predefined. Since "CCGG" sites will be used to digest the reference genome later, we need to control the number of occurrence of "CCGG" when simulating the genome. Our method is to set up a random number generator and visit this generator every time adding a nucleotide. If the number returned by this generator exceed threshold set by us, then a "CCGG" fragment will be incorporated into growing sequence instead. If not, other if-conditions will be visited to decide which nucleotide to add. This genome simulator is developed at early stage of BisSeq to provide small and simple reference genome for software testing purpose. This function has been retained in case users might need small reference genome to get a quick view about how BisSeq works. In the next edition, this function will be deprecated.

With reference genome loaded into memory, the python script will then scan the genome to find out all the "CpG" sites (even though there are many types of DNA methylation, BisSeq focus on the 5'-C methylation, which are mostly likely to occur within "CpG" islands). A methylation rate will be randomly assigned for each "CpG"

site, and this information will be stored in a dictionary. Dictionary is a data structure in python, which is similar to hash table in other programming language, here "CpG" position index is stored as key and the methylation rate is stored as value. The pseudo code is shown in upper panel of Fig 6, the dictionary here is named "MetTable".

Methylation table generation pseudo code:

```
methystates = [0, 5, 10 … , 100]
While not End of refGenome:
    index = find.next("CG", refGenome)
    MetTable[index] = random.choice(methystates)
```

Differentially methylated copies:

```
For genome in range(1, genCopyNum):
    for cgPOS,fracMET in MetTable.iteritems():
        cg = random_integer(0,100)
        if (cg < fracMET):
            refGenome[cgPOS] = 'x'
            refMetCount[cgPOS] += 1
#  'x' indicates a methylated Cytosine here.
```

Figure 6. Pseudo code to generate methylation table and to differentially methylated genome copies. Here genome sequence is stored as an array in "refGenome".

Before we start to generate sequencing reads, we need to acquire a heterogeneous methylated population first. This resembles the fact that in the real sequencing sample each different cell may have their own unique methylation profile and together they form a methylation pattern for that sample. So now the problem is: how can we methylate each reference genome copy to finally achieve a population,

which represent methylation pattern we pre-calculated in the last step. Our solution is to follow a genome-by-genome fashion, where we run a for-loop for number of genome copy times (genome copy number is defined by user at the parameter tuning stage). As show in lower panel of Fig 6, at the beginning of each loop we iterate through all the "CpG" sites store in "MetTable" and decide whether to methylate it or not. The method we use here is to visit a random number generator, which produce random integer between 0 and 100. If the number generated is less than the methylation rate we designate for this position in "MetTable", then we swap the cytosine in "CpG" site with an "x" mark to indicating that cytosine is methylated. Otherwise, we pass this site and leave cytosine unchanged. For instance, we have 100 genome copies to methylate. And for position index "10045", we designate a methylation rate of 25%. For 100 genome copies, we will visit the random integer generator 100 times. Assuming that the function is random and each number has equal probability to be returned, then around 25 out of total 100 "cg" generated will be smaller than 25, and those genome copies will be methylated at position index "10045". Every time we methylate a site, we record it by increase the methylation count of position by 1, as reflected by "refMetCount[cgPOS] += 1" in Fig 6. So at the end, we can calculate exact methylation rate based on "refMetCount" dictionary.

The next step is restriction enzyme digestion. We currently support MspI digestion, since it is the most widely used enzyme for RRBS. In fact, there are many enzymes also eligible for RRBS as long as they have "CG" dinucleotide in their recognition site, we will discuss this possibility in future work section. Noticing that some cytosine ("C") is now represented by "x" due to methylation, the recognition site used for *in silico* digestion is "CCGG" or "CxGG" (MspI will recognize "CCGG" site

and cut at the second cytosine from the 5' end, leaving a "CGG" end). After scanning through the genome string, digested fragments are stored in an array.

Before we start generating sequencing reads, there is an additional size selection step. When the RRBS technique first comes out, no size selection was applied before the sequencing. However, researchers find out that by doing *in silico* against whole genome they can acquire a length range where resides most of fragments come from promoter area. This progress leads to an addition of size selection to RRBS protocol, which enables scientists to obtain methylation information from promoter area. Also size selection can be adjusted for specific research interest, which we will discuss later. According to published data (Gu et al., 2010, Cokus et al., 2008), we implement size selection to retain all fragments within range of 40~150bp and 150~220bp. The size selection could also be tuned per users' request. Then python script will perform the bisulfite conversion. Similar to the method we used before, a random number generator is set up to determine whether an unmethylated cytosine will be converted. The "x"s in fragments are converted back to "C", which represent methylated cytosine. Until now, all enzyme digested and bisulfite converted fragments are stored in an array. These fragments are cropped at one end with predefined "readLen" and written to .fa or .fq file. Noted that there is also a probability of whether a fragment will be cropped, reflected in the reads file; the probability used here is "coinToss". This process is repeated "seqCycles" times for each fragment to generate desired depth of coverage.

The reads simulation is largely organized around the for-loop. The reason is that by incorporating reads generation into each loop instead of methylating all

genome copies first, we avoid storing all those differentially methylated genomes in memory, saving huge amount of memory space.

3.4 Reads Mapping and Data Visualization

Based on which alignment tool the user choose, it will be loaded to map sequencing reads to reference genome. Normally RRBS alignment tools require a pre-processing step to either convert sequencing reads or reference to a G-to-A and C-to-T version, sometimes both. After reads are mapped and methylations are extracted, BisSeq will take the results to generate tables to visualize the performance of alignment tool. The final representation of result profiling the performance of tool questioned contains two figures: one is scatter plot with observed methylation rate as X axis and expected methylation rate as Y axis; another one is bar plot with methylation rate call value as X axis and corresponding counts as Y axis.

Chapter 4

RESULT

4.1 BisSeq: The Command Line Tool of RRBS Simulation and Profiling

BisSeq is now working as a command line package, consisting of four scripts. 1) 00-Bis-SimQuantPipe.sh, 2) 01-GenerateBisulfiteSeqTagData.py, 3) 02-bis-ScoreSimMethyl.py, 4) 03-bis-SimMetPlots.R . The workflow is as follows:

4.1.1 Core Shell Script

This shell script, 00-Bis-SimQuantPipe.sh, is responsible to set up folders and file location for later simulation results. It reads in simulation parameters (as shown in Figure 6) from users and decides which steps within BisSeq pipeline will be executed. There are several gate variables set up in this script; they will be sequentially visited to decide whether certain step is going be run. This design provides an easy-to-use function that facilitates users to flexibly use BisSeq to implement different simulation purposes, and may potentially save substantial amount of time by avoiding repeated work. In addition to flow control, core shell script stores configuration in a text file as run log every time it is executed, in case users may need to retrieve run-specific setting later.

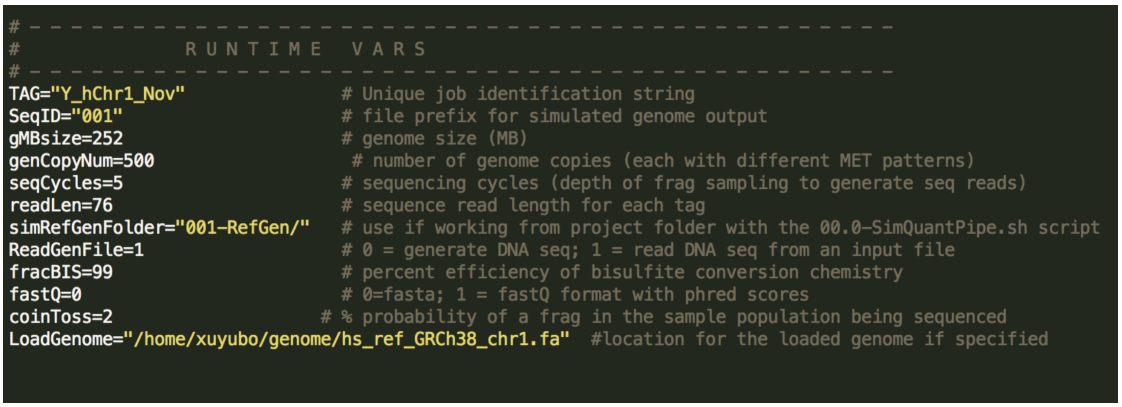


Figure 7 Example of parameters setting in the shell script

4.1.2 Simulation Script

Simulation script, 01-GenerateBisulfiteSeqTagData.py, is called by core shell script when its corresponding gate variable is set to 1. It parses the arguments calling it and extracts variable values sent by the core shell script for simulation. As mentioned in method section, this script will first generate a methylation table and provide it to downstream code; this table will be stored in a text file as a run log. The first column is index position of the "CpG" sites (0-indexed), the second column is the methylation rate designated for that site. While simulation going on, methylation status for each "CpG" site is also recorded and then actual methylation rates in simulated reads will be calculated base on them. This actual methylation rates table reflects the real methylation information in the genome population used for subsequent reads production; it is unknown in real-world experiment which makes it difficult to objectively benchmarking RRBS analyzing tool. This table is also used in the final sequencing error evaluation in the later section. Running log printed out in command line is also stored for debugging purpose.

4.1.3 Results Processing Script

After reads are simulated, aligned to reference genome and methylation call has been made, the results need to be filtered to accommodate for downstream visualization. This script, 02-bis-ScoreSimMethyl.py, comb raw results from aligner tool and generate ready-to-plot tables. Three tables will be generated depending on whether a site is profiled having positive methylation rate, and whether a site have a expected positive methylation rate. The first table-"output table", which contains sites have positive expected methylation rate and positive profiled rate, has 7 columns: 1) position index of "CpG" site, 2) expected methylation rate, which extracted from table generated simulation script, 3) observed methylation rate that calculated by aligning tool under study, 4) number of methylation calls at this position, 5) number of unmethylated calls at this position, 6) the percent of guanine and cytosine within 40 base pair range, taking "CpG" site in center position, 7) the distance from this site to nearest methylated site. The second table, "lost table", contains the lost sites, which have positive expected methylation rate but profiled having no methylation. The third table, "other table", stores the sites are not expected to be methylated but profiled to have positive methylation rate.

4.1.4 Data Visualizing Script

This script, 03-bis-SimMetPlots.R, is written in R. It produces two figures: 1) obsMet vs expMet plot, which communicates how far the RRBS analyzing tool deviated from correct result. 2) methylation rates distribution bar plot, which represent the distribution of profiled methylation rates.

Currently beta version of BisSeq is still under construction, developments of more features are in progress and will be added to the package. Interests and inquiries

about BisSeq could send to Dr. Marsh at amarsh@udel.edu and Yubo Xu at xuyubo@udel.edu.

4.2 Case Study: RRBS Simulation of Human Chromosome 1

4.2.1 Sequencing Reads Simulation

BisSeq makes it possible to simulate RRBS process and the following software benchmarking. We use human chromosome 1 as reference genome, with some key parameters setting shown in table 2:

Parameters
TAG="Y_hChr1_Feb"
SeqID="001"
genCopyNum=500
seqCycles=5
readLen=60
fracBIS=99
coinToss=2

Table 2 Parameters setting for simulation

All tasks were deployed on Biohen server hosted at Center for Bioinformatics and Computational Biology, University of Delaware. The Node37 was reserved for BisSeq tasks. It has two Intel Xeon E5-2630 @ 2.30GHz processors, each with 6 cores, It takes ~1 hour to finish the simulation step, generating sequencing reads file of size

2.25 Gb, containing 18617698 sequencing tags. Among 98187000 "CpG" sites, there are 51282435 sites are unmethylated, the remaining ones are methylated. The alignment tool we choose here is Bismark. Before running reads alignment, Bismark needs to convert reference genome into a G-to-A and C-to-T version. After reads are aligned, Bismark methylation extractor was executed to acquire methylation status. Results profiling Bismark are shown in Figure 7. It shows that Bismark profiled methylation rate generally resemble that of real data, even with a small deviation. However, this performance is from an experiment simulating against a small fraction of genome, whether Bismark can provide an accurate profiling of reads against large and complex genome is still waiting to be answered.

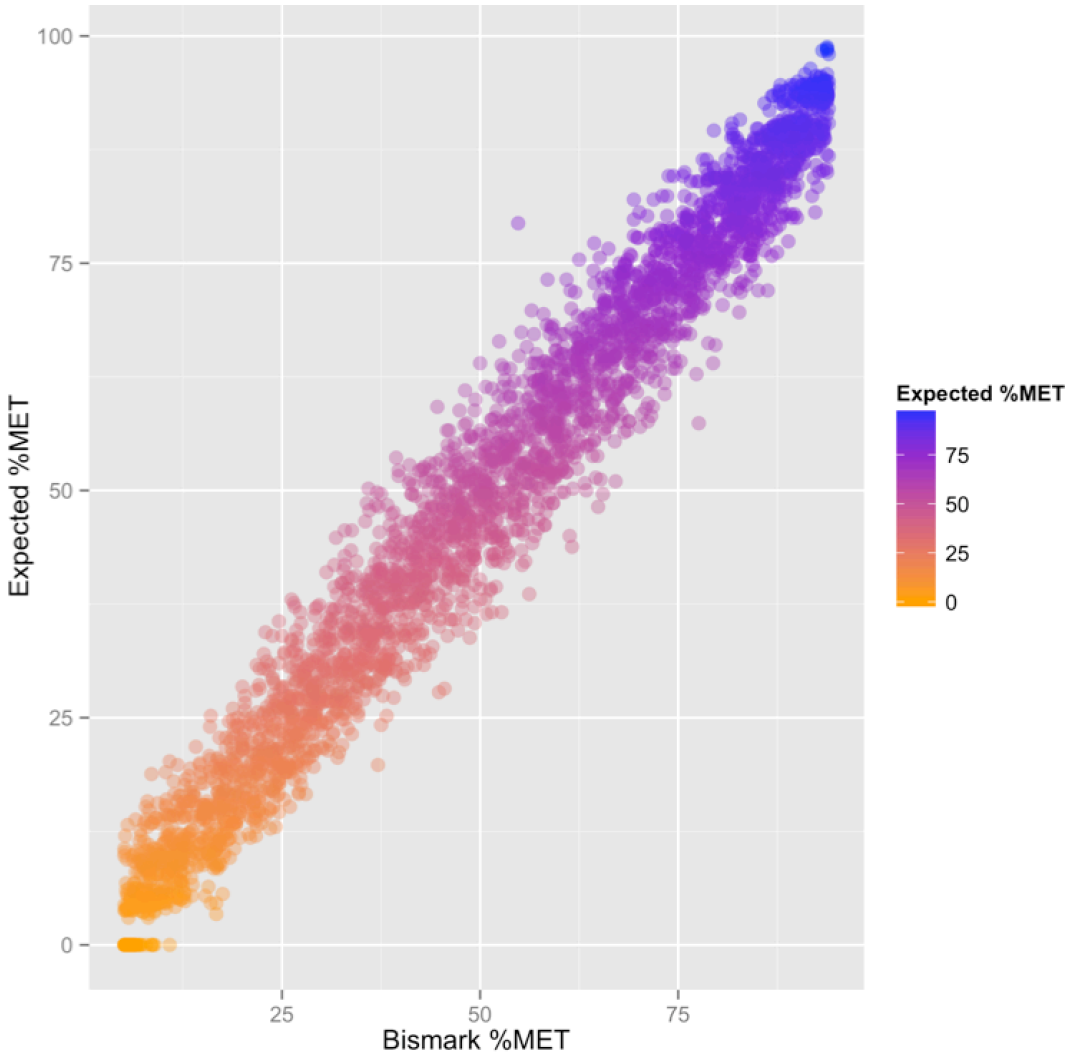


Figure 8 Methylation rate profiled by Bismark vs Actual methylation rate.

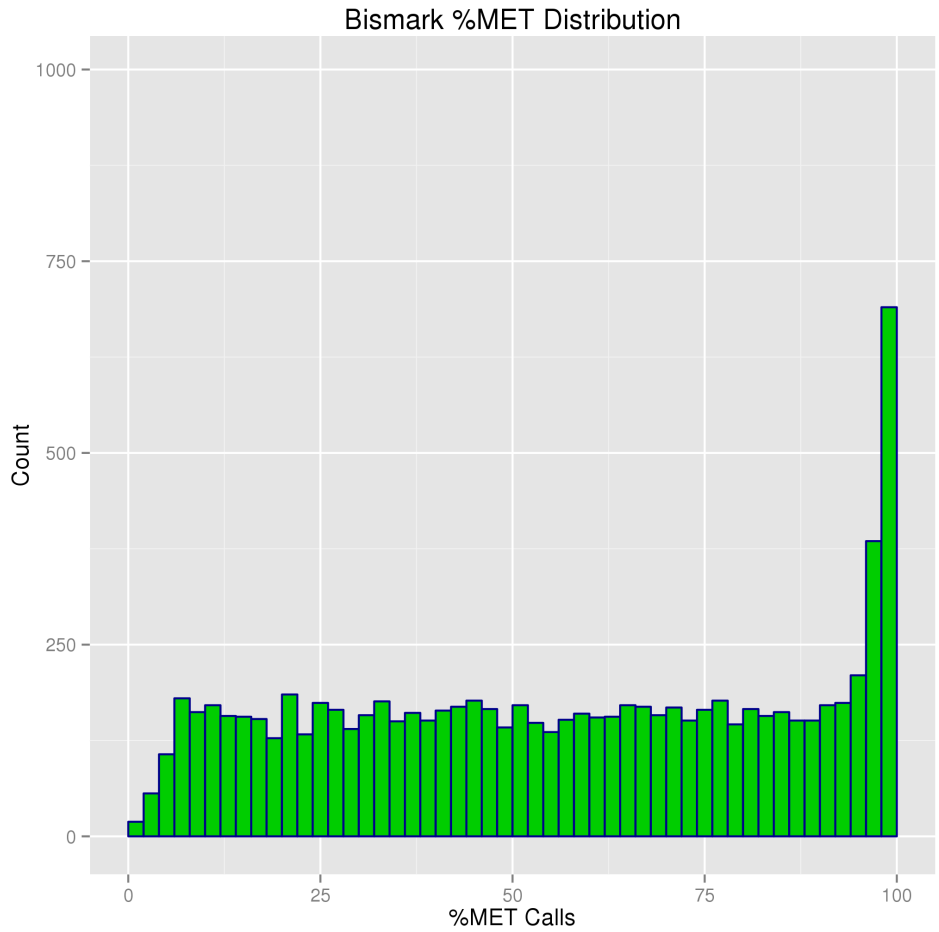


Figure 9 Methylation Distribution profiled by Bismark on Y_hChr1_Feb_001

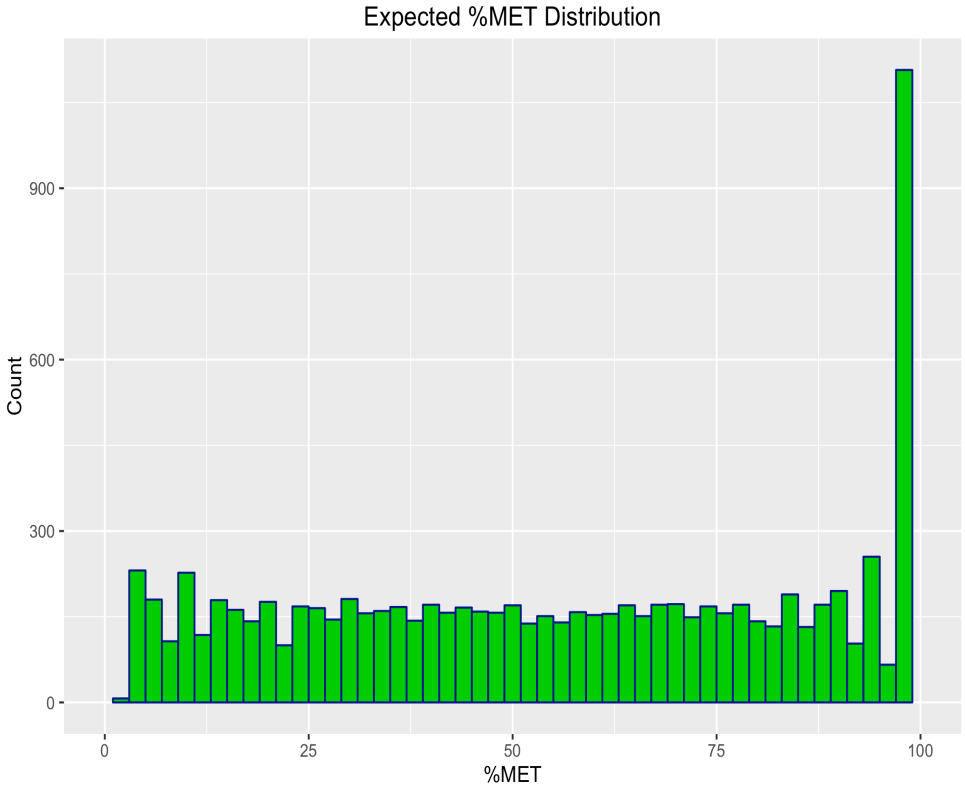


Figure 10 Expected methylation distribution of Y_hChr1_Feb_001

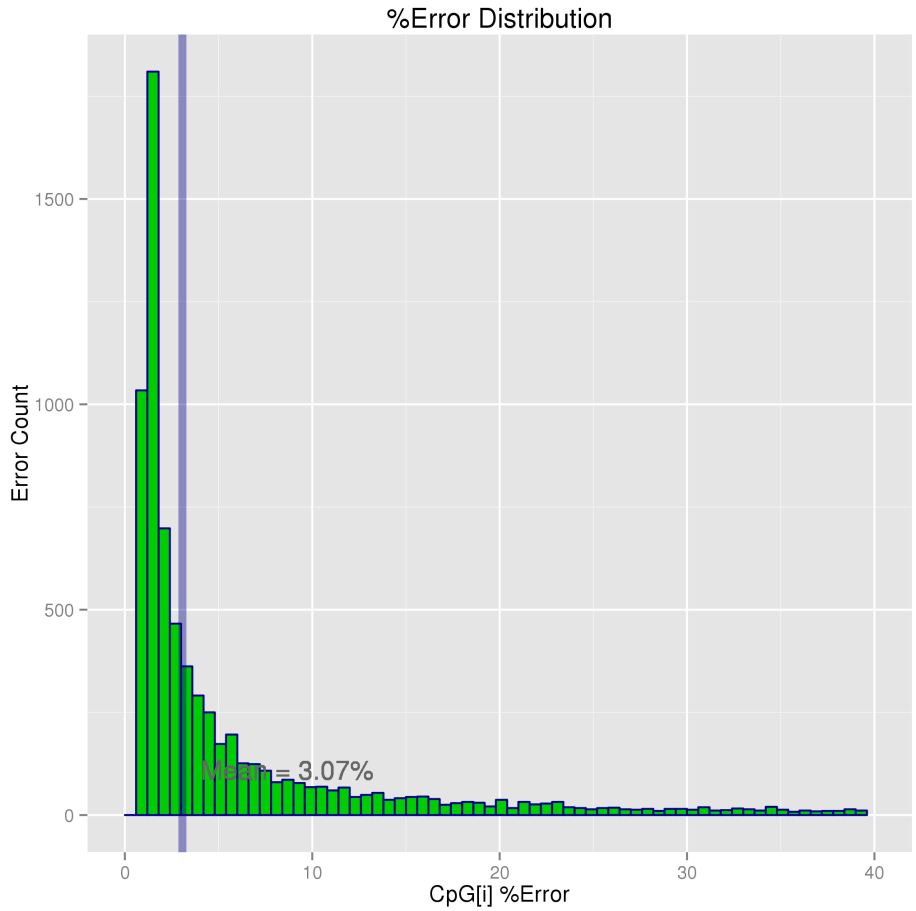


Figure 11 Error rate distribution of Y_hChr1_Feb_001

4.2.2 Read Length Effect On Different RRBS Mapping Tools

Another question about RRBS has been read length. With the advancement of sequencing technique, nowadays we are able to achieve sequencing reads with longer and longer length. However, which sequencing analysis tools to use still has no clear answer across different research groups. Here we design this use case to profile the

performance of different RRBS mapping tools when dealing with various read-length sequencing samples, we choose the range from 40 ~ 140bp with 10bp increment per group. There are three bisulfite sequencing mapping tools selected, Bismark, BSseeker2 (Guo et al., 2013), BSMAP, based on the their popularity and maintenance effort from their development team.

As goal is to benchmark the performance of bisulfite mapping tools, we designed a profiling metric accordingly. There are three factors that we take into consideration here: mapping efficiency, CPU running time, and the methylation error. Here by mapping efficiency, we mean the number of uniquely mapped read divided by all sequencing reads. And the "uniquely" does not indicate the read mapped exactly one time to reference genome, sequencing reads usually have multiple matches where each match has alignment score. As long as there is one match for a read has much higher alignment score than the other matches, then this read is "uniquely" mapped. Our experiment result shows that, across all read length groups, BSseeker2 achieved highest rate of uniquely mapped reads, followed by BSMAP and then Bismark. As shown in Fig. 8, both three tools' mapping efficiency increase from around 70% to 95% as the read length of sequencing sample increases, the performance of three tools are very close to each other. The second factor is CPU running time, which is time the tool needed to finish mapping. For some tools that convert reference genome before mapping, the time spend on conversion are also included. In Fig. 9, we can tell BSMAP is fastest in terms of CPU running time, followed by Bismark, and BSseeker2. Last factor is methylation error. Knowing the expected methylation rate, as we record that information during simulation process, we calculate R-squared value for each experiment to evaluate the accuracy of methylation call for each tool. Fig. 10 shows

that R-squared value of BSMAP is lower than other two tools across all read-length groups, whereas BSseeker2 and Bismark have similar values. This indicates that even BSMAP runs faster than BSseeker2 and Bismark, it produces less accurate methylation profile. Fig 11 shows the visualization of the data behind how we calculate the R-squared value plot. Combined the comparison metrics together, Bismark can produces relatively accurate methylation profile within reasonable amount of time, while BSseeker2 require longer running time. BSMAP can finish mapping quick, but the quality of methylation profile is not as well as those generated by other two tools.

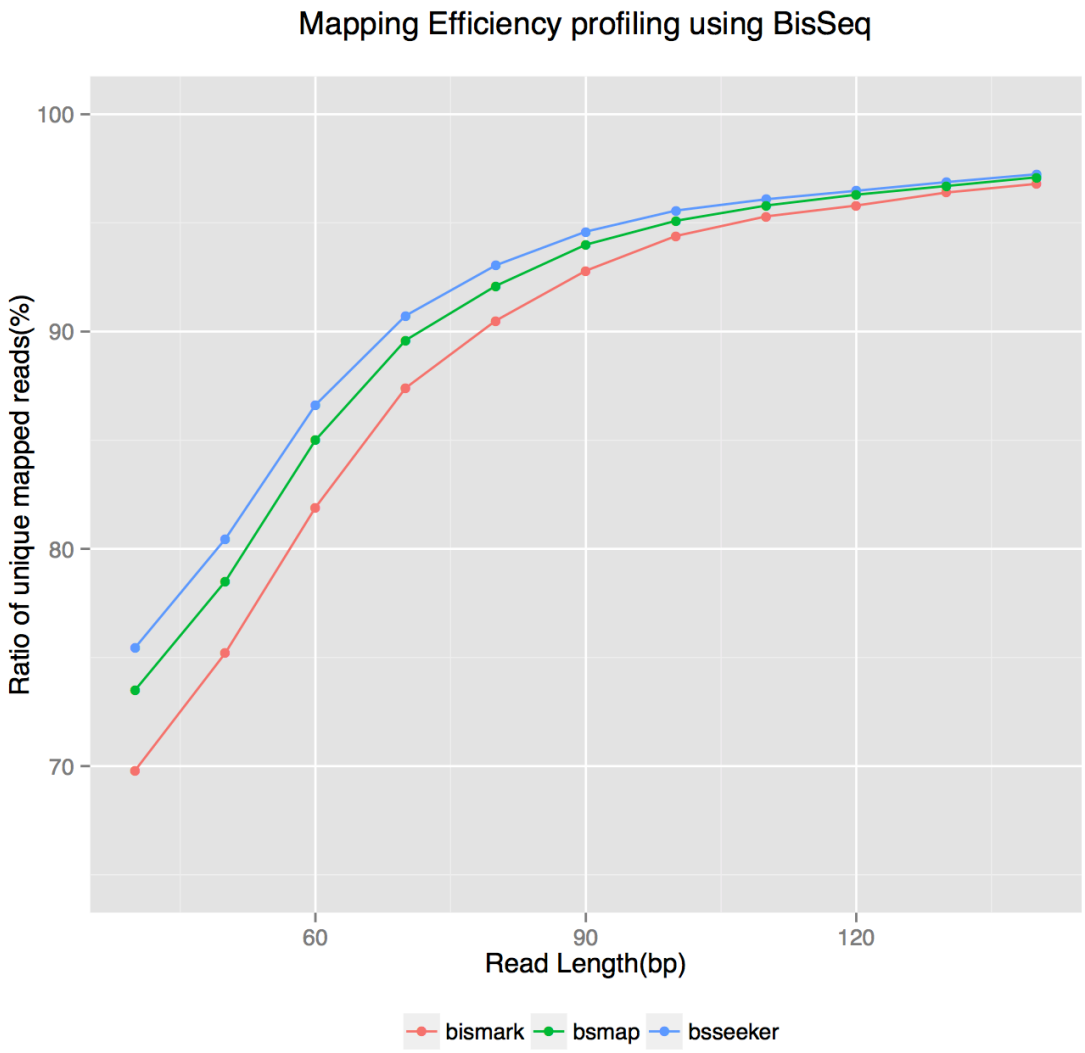


Figure 12 Mapping efficiency profile using BisSeq. The sequencing reads are generated with read length increase from 40 bp to 140 bp with 10 bp interval. The number of mismatch allowed is set to zero, and the rest of parameter are using default for each program.

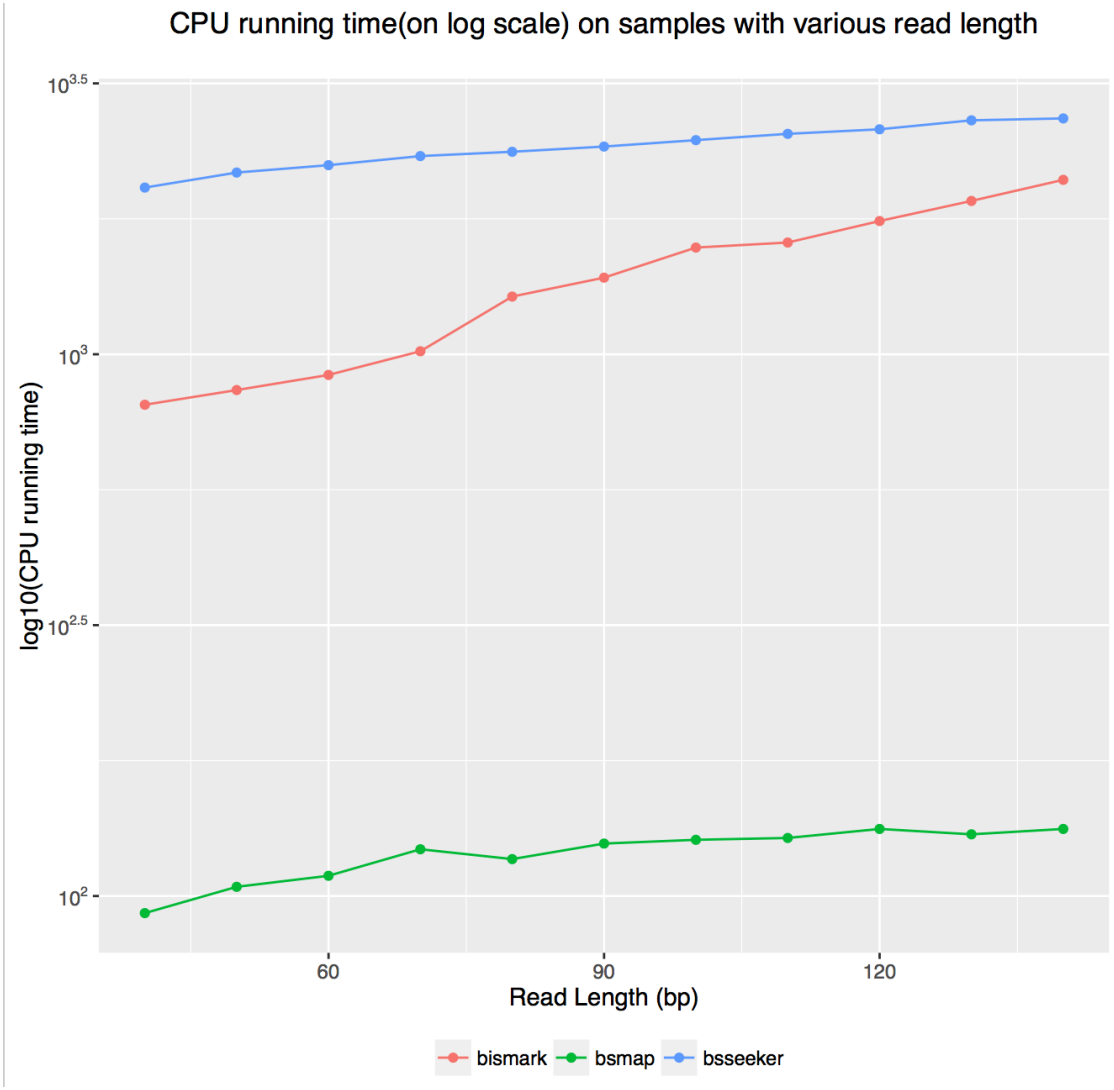


Figure 13 CPU running time evaluation of Bismark, BSseeker2, BSMAP. Here Bismark, BSseeker2 convert reference genome before mapping, the time needed for conversion are also included in running time.

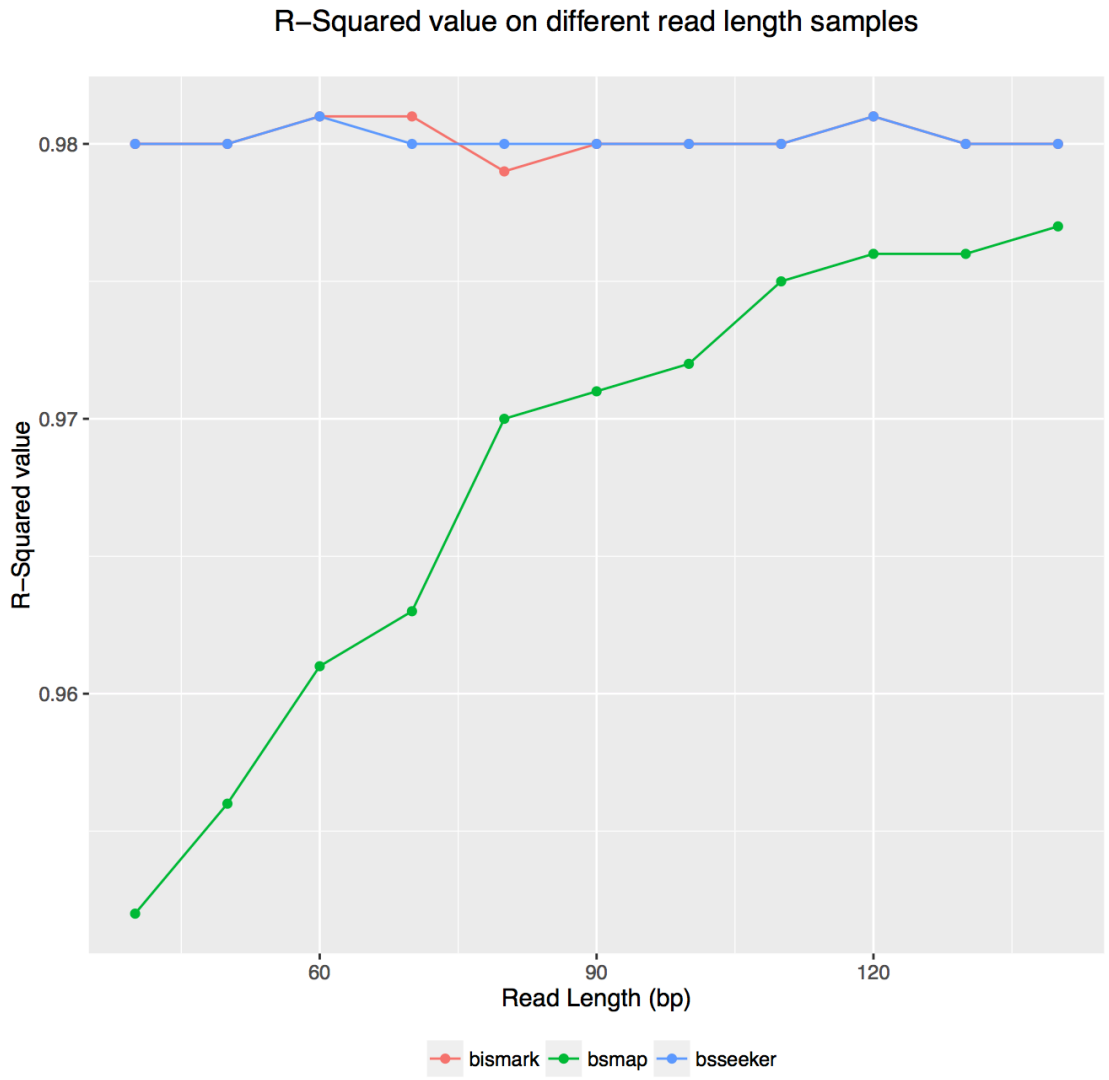


Figure 14 R-squared value plot for each bisulfite mapping tools across various read-length sequencing samples. .

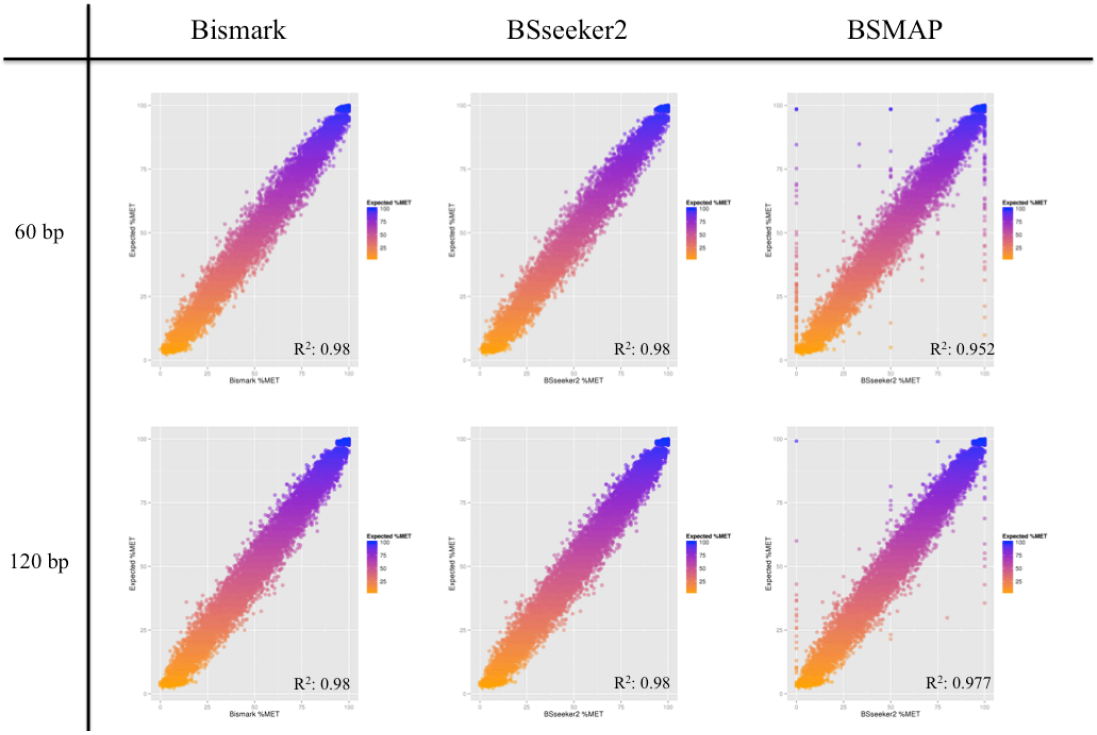


Figure 15 Visualization of Observed methylation rate vs Expected methylation rate for Bismark, BSseeker2, BSMAP, across 60bp read-length group and 120bp read-length group.

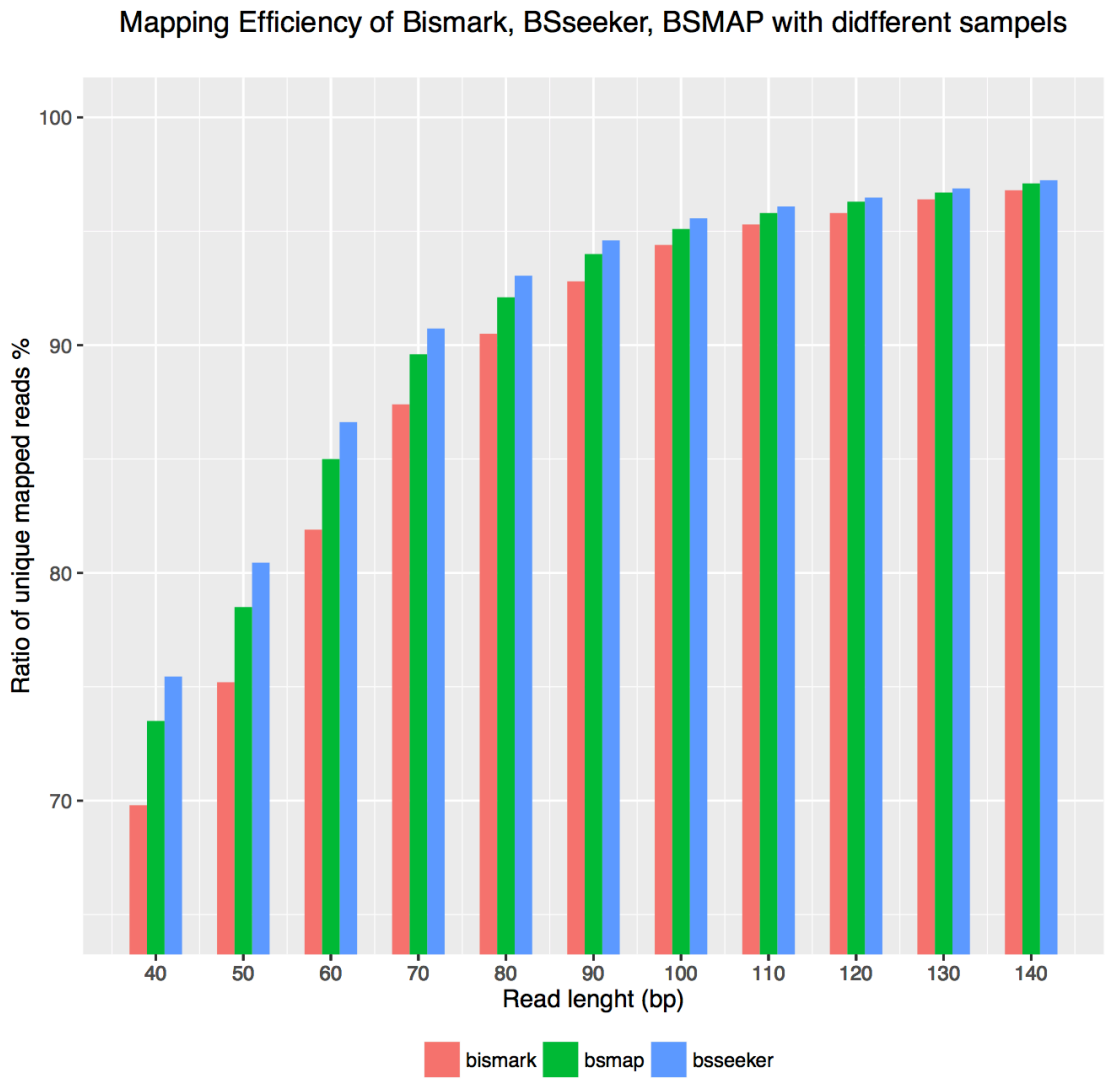


Figure 16 Mapping efficiency profile using BisSeq shown in bar plot grouped by various read length..

Chapter 5

DISSCUSSION AND FUTURE WORK

5.1 Dicussion

As a pilot project, there are still many features could be improved in BisSeq pipeline. The RRBS simulator now only supports Illumina sequencing platform. Different sequencing platform employ various techniques that leading all kinds of error model. A good simulation tool should provide service across different platform.

In terms of simulation, BisSeq's error model is currently very simple and may not able to resemble all the detail that occurred in real sequencing process, which may affect the confidence level of research outcome using BisSeq. Now BisSeq only allows single input reference genome, does not enable cross species simulation, this prevent metagenomic (Richter et al., 2008) researchers from using our tool. All in all, even BisSeq provides users with opportunities to simulate RRBS data; there are still certain drawbacks in BisSeq that confine its functionality. We describe some our solutions and next step plan in the following section.

5.2 Future Work

Our future work will devote to modify the simulation section of BisSeq to provide a more comprehensive simulation across multiple sequencing platforms. In near future, an improved version of error model will be loaded into BisSeq. It will

support sequence context based error model (Nakamura et al., 2011) and possibly enable users to generate their own error model from previous sequencing data.

As a critical step in RRBS, enzyme digestion exposes "CpG" to end of fragment, which makes it easier for them to be sequenced. But for a certain enzyme, take MspI as example, it can only has limit number of recognition sites within reference genome. We are planning to use multiple restriction enzymes in combination, as long as they have "CG" in their recognition sites, to identify more effective enzyme combo. Further improvements to BisSeq will include but not limited to the previous mentioned parts.

Chapter 6 CONCLUSION

At the current stage, we developed a RRBS simulation and analyzing pipeline capable of generating single-end reads for Illumina sequencing platform. BisSeq read in users' parameter configuration and generates corresponding reads tag in fasta format. It provides users with comprehensive log files to track each execution. Users can choose their own alignment tool to achieve specific experiment results. BisSeq communicates the performance of alignment tool in forms of easy-to-interpret figures. Read-length effect use case shows the potential BisSeq have on benchmark bisulfite mappings tools. Even certain drawback exists, BisSeq fill the blank in area of RRBS simulation. It is a good prototype that can be optimized in the future.

REFERENCES

Metzker, M.L. (2010). Sequencing technologies - the next generation. Nat Rev Genet 2010, 11:31-46.

T, Webb., F, Latif. (2001). Rett syndrome and the MECP2 gene. J Med Genet. 38(4): 217–223.

Cokus, S., Feng, S., Zhang, X., Chen, Z., Merriman, B., Haudenschild, C., Jacobsen, S. et al (2008). Shotgun bisulphite sequencing of the Arabidopsis genome reveals DNA methylation patterning. Nature, 215-219.

Choi, M., Lee, J., Le, M., Nguyen, D., Park, S., Soundrarajan, N., . . . Park, C. (2015). Genome-wide analysis of DNA methylation in pigs using reduced representation bisulfite sequencing. DNA Research DNA Res, 343-355.

Gu, H., Bock, C., Mikkelsen, T., Jäger, N., Smith, Z., Tomazou, E., . . . Meissner, A. (2010). Genome-scale DNA methylation mapping of clinical samples at single-nucleotide resolution. Nature Methods Nat Meth, 133-136.

Harris,R.A. et al. (2010) Comparison of sequencing-based methods to profile DNA methylation and identification of monoallelic epigenetic modifications. Nat. Biotechnol., 28, 1097–1105

McElroy KE1, Luciani F, Thomas T.(2012) GemSIM: general, error-model based simulator of next-generation sequencing data. BMC Genomics. 1471-2164-13-74.

Xi, Y., & Li, W. (2009). BSMAP: Whole genome bisulfite sequence MAPping program. BMC Bioinformatics, 232-232.

Langmead, B., Trapnell, C., Pop, M., & Salzberg, S. L. (2009). Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol Genome Biology, 10(3). doi:10.1186/gb-2009-10-3-r25

Hancock, J. M. (2004). BLAT (BLAST-like Alignment Tool). Dictionary of Bioinformatics and Computational Biology. doi:10.1002/0471650129.dob0071

Wang, J., Xia, Y., Li, L., Gong, D., Yao, Y., Luo, H., Gao, F. et al. (2013). Double restriction-enzyme digestion improves the coverage and accuracy of genome-wide CpG methylation profiling by reduced representation bisulfite sequencing. BMC Genomics, 11-11.

Tárraga, J., Pérez, M., Orduña, J., Duato, J., Medina, I., & Dopazo, J. (2015). A parallel and sensitive software tool for methylation analysis on multicore platforms. Bioinformatics, 3130-3138.
Krueger, F., & Andrews, S. (2011). Bismark: A flexible aligner and methylation caller for Bisulfite-Seq applications. Bioinformatics, 1571-1572.
Xi, Y., Bock, C., Muller, F., Sun, D., Meissner, A., & Li, W. (2011). RRBSMAP: A fast, accurate and user-friendly alignment tool for reduced representation bisulfite sequencing. Bioinformatics, 430-432.
Li, R., Li, Y., Kristiansen, K., & Wang, J. (2008). SOAP: Short oligonucleotide alignment program. Bioinformatics, 24(5), 713-714. doi:10.1093/bioinformatics/btn025
Sun, Z., Cunningham, J., Slager, S., & Kocher, J. (2015). Base resolution methylome profiling: Considerations in platform selection, data preprocessing and analysis. Epigenomics, 813-828.
Chatterjee, A., Stockwell, P., Rodger, E., & Morison, I. (2012). Comparison of alignment software for genome-wide bisulphite sequence data. Nucleic Acids Research.
Fonseca, N., Rung, J., Brazma, A., & Marioni, J. (2012). Tools for mapping high-throughput sequencing data. Bioinformatics, 3169-3177.
Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP, Evers DJ, Barnes CL, Bignell HR, et al: Accurate whole human genome sequencing using reversible terminator chemistry. Nature 2008, 456:53-59.

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R: The Sequence Alignment/Map format and SAMtools. Bioinformatics 2009, 25:2078-2079.

Weichun Huang, Leping Li, Jason R Myers, and Gabor T Marth. ART: a next-generation sequencing read simulator, Bioinformatics (2012) 28 (4): 593-594
Nakamura K, Oshima T, Morimoto T, Ikeda S, Yoshikawa H, Shiwa Y, Ishikawa S, Linak MC, Hirai A, Takahashi H, et al: Sequence-specific error profile of Illumina sequencers. Nucleic Acids Res 2011, 39:e90.

Richter DC, Ott F, Auch AF, Schmid R, Huson DH: MetaSim: a sequencing simulator for genomics and metagenomics. PLoS One 2008, 3:e3373.

Balzer S, Malde K, Lanzen A, Sharma A, Jonassen I: Characteristics of 454 pyrosequencing data–enabling realistic simulation with flowsim. Bioinformatics 2010, 26:i420-425.

Guo W, Fiziev P, Yan W, Cokus S, Sun X, Zhang MQ, et al. BS-Seeker2: a versatile aligning pipeline for bisulfite sequencing data. BMC Genomics. 2013;14:774.

Mill J, Yazdanpanah S, Guckel E, Ziegler S, Kaminsky Z, Petronis A: Whole genome amplification of sodium bisulfitetreated DNA allows the accurate estimate of methylated cytosine density in limited DNA resources. Biotechniques 2006, 41: 603e607
Bibikova M, Fan JB: Genome-wide DNA methylation profiling. Wiley Interdiscip Rev Syst Biol Med 2010; 2:210-23; PMID:20836023;
Moran, S., Vizoso, M., Martinez-Cardús, A., Gomez, A., Matías-Guiu, X., Chiavenna, S. M., . . . Esteller, M. (2014). Validation of DNA methylation profiling in formalin-fixed paraffin-embedded samples using the Infinium HumanMethylation450 Microarray. Epigenetics, 9(6), 829-833. doi:10.4161/epi.28790
Mohn, F., Weber, M., Schübeler, D., & Roloff, T. (2009). Methylated DNA Immunoprecipitation (MeDIP). Methods in Molecular Biology DNA Methylation, 55-64. doi:10.1007/978-1-59745-522-0_5
Taiwo, O., Wilson, G. A., Morris, T., Seisenberger, S., Reik, W., Pearce, D., . . . Butcher, L. M. (2012). Methylome analysis using MeDIP-seq with low DNA concentrations. Nat Protoc Nature Protocols, 7(4), 617-636. doi:10.1038/nprot.2012.012