Simulation of Next Generation Sequencing Short Reads for Mutation Spectrum Analysis

By

Ahmad Ghadiri Modares

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Master of Applied Science

in Electrical and Computer Engineering

Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada
January 2015

Abstract

Next generation sequencing (NGS) of mutant reporter transgenes is increasingly being used for mutation spectrum analysis (MSA) to characterize the genomic effects of mutagens. The ability to simulate NGS-MSA experimental data will permit the tuning of various parameters in the downstream analysis pipeline. However, no simulator currently exists that is capable of producing the read depths (up to 100,000x) required for MSA experiments. In this study, we introduce MutSim, a short read mutation simulator that enables researchers to generate NGS data for simulated samples exposed to a mutagen. MutSim generates data following the Ion Proton™ instrument error model and also a mutational model of a given mutagen. MutSim is shown to be highly scalable both with respect to genome length and read depth coverage. MutSim simulated data is validated against real experimental data in several aspects including genotype content, quality scores, read depth coverage and read length.

Acknowledgments

I would like to express my deepest appreciation to my supervisor Professor James Green for his continuous support, excitement and assistance in this thesis. I would also like to thank Rémi Gagné, Marc Beal, and all other collaborators at Health Canada who provided me with guidance throughout this study.

Thanks to NSERC and Carleton University for providing financial support and funding this research.

Lastly, I would also like to thank my parents, my sister and brother-in-law who encouraged me to pursue my studies at this level.

Table of Contents

List of Tables

List of Figures

List of Equations

List of Abbreviations

NGS	Next Generation Sequencing
MSA	Mutation Spectrum Analysis
JVM	Java Virtual Machine
SNP	Single Nucleotide Polymorphism
TGR	Transgenic Rodent Assay

1 Introduction

1.1 Background

Next Generation Sequencing (NGS) has dramatically changed the way research is conducted in the field of bioinformatics. In recent years, a variety of sequencing technologies, each of which suitable for different types of DNA analyses, have given scientists the ability to explore increasingly complex biological processes. Commonalities among these technologies are improved throughput, decreased cost, and enhanced flexibility of the sequencing process [1], [2] relative to the previous state-of-the-art technology, namely Sanger sequencing. However, each technology uses fundamentally different sequencing processes, which in turn leads to different biases and systematic error tendencies. For instance, while one sequencer may maximize speed for re-sequencing, the other may feature a slower and more accurate approach, making it suited for *de novo* sequencing. The diversity and uniqueness of NGS technologies introduce undesirable complexity in modeling the behaviors and attributes of each technology [3]. As a result, more advanced characteristic capturing schemas for error modeling are required in order to obtain more accurate and consistent data simulation.

As our ability to sequence DNA increases, more sophisticated problems become tractable, which in turn, require more advanced data analysis software pipelines. When conducting DNA analysis, it is essential to establish a proper experiment design as well as to optimize the many configuration parameters in each stage of the analysis pipeline. This process typically involves numerous sequencing runs in order to iteratively evaluate and improve the performance and efficiency of the analysis pipeline [4]. Even though the cost of the sequencing

process has dramatically decreased over the past decade, accurate and representative simulated data would significantly expedite and simplify this process at a significantly less cost [5].

Furthermore, we are particularly interested in using NGS for mutation spectrum analysis (MSA) [6], [7]. MSA is useful to biologists because it can reveal insights into the mechanisms by which chemicals mutate DNA. Mutations often cause disruptions in structural and behavioural features of the target sequence [8], [9]. In an MSA study, changes in mutation frequency and pattern between control and exposed groups are analyzed to characterize the mutagen under study [10].

1.2 Motivation

With new NGS technologies offering more affordable and faster sequencing approaches, a growing variety of analyses exist. As a result, there is an increasing growth in need for simulation of data for newer and more complex DNA analysis. Simulation of DNA short reads would greatly facilitate the interpretation of DNA analysis and configuration of pipeline tools and their parameters. As illustrated in Figure 2, instead of designing repetitive and costly experiments from the beginning and sequencing more samples, we propose that simulated data could be leveraged to assist analysis pipeline parameter tuning and refining of experiment design. [11]–[15].

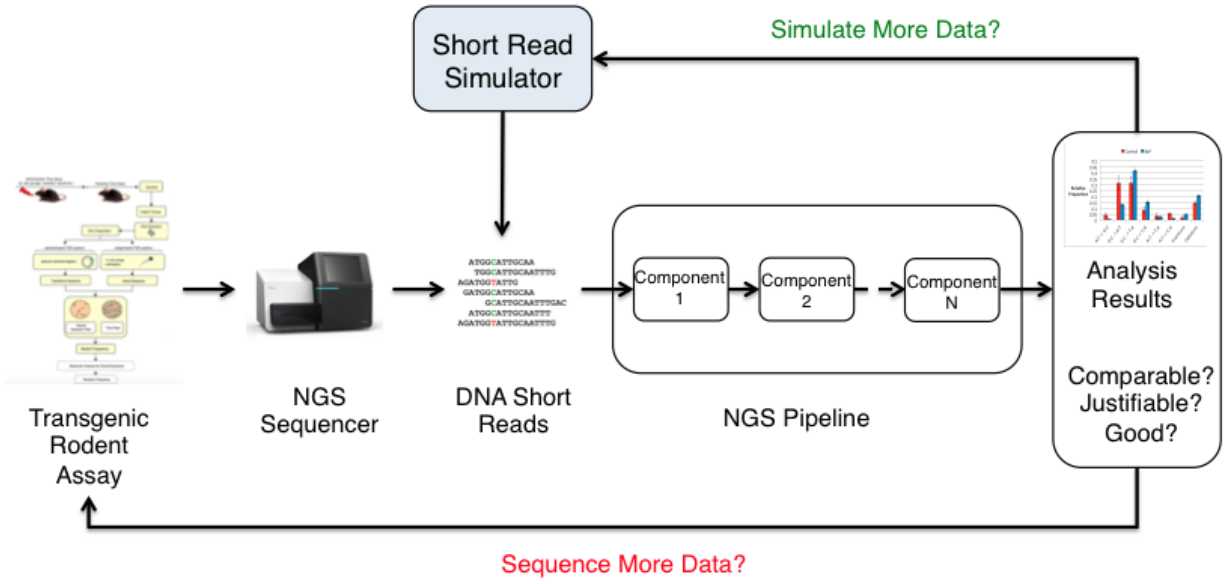

Figure 1: Usefulness of DNA Short Read Simulation

Illustrates the overall process of DNA analysis for a transgenic rodent assay used for mutation spectrum analysis (see section 2.3 for more explanations about MSA). If the MSA results are not satisfactory, one would normally have to collect more biological data. The simulator developed in this thesis provides a low-cost convenient alternative.

Using NGS for MSA imposes a number of new requirements on a simulator including read depths, which are orders of magnitude larger (3000-fold) than those typically used in genome-wide sequencing experiments. These requirements pose the challenge of introducing new practical algorithms which can scale with the read depth coverage required for simulation of MSA experiments. In other words, the simulator should be able to generate reads in a reasonable amount of time given the high read depth coverage required for mutation spectrum analysis.

Furthermore, instrument errors characteristic of various technologies should be accurately modeled as each technology tends to introduce diverse classes of variations. Particularly, the Ion Proton™ instrument, which employs semi-conductor technology, is not

well-studied in the area of simulation nor is it supported by any currently available simulation package.

Another important requirement of generating short reads for MSA studies is the ability to model the effects of arbitrary mutagens on the DNA sequence under study and the way it influences the generation of short reads. As more inclusive information, such as genotype composition and positional probability of mutation play vital role in this type of simulation, more elaborated modeling schemes are required to produce simulated data comparable with actual sequenced data for MSA studies.

1.3 Statement of the Problem

We seek to address new challenges and complexities added to the simulation of data produced as the result of MSA assays. In order to facilitate the parameterization and benchmarking of software pipelines configured for MSA, accurate and inclusive simulated data are required. Existing simulators are effective within their intended domain, however, these simulators provide minimal or no support for modeling mutations. When using software to generate short reads containing biologically-induced mutations, ideally the researcher would want to specify the genotype composition proportions of mutations as well as the locations at which mutations are more likely to occur. Ideally, the designed simulator should be able to accept and process the mutational model as they are presented in MSA studies. This would render the generality and the convenience of employing the simulator.

In addition, during MSA experiments, DNA fragments are typically sequenced at very high coverage (e.g. x80,000) in such assays in order to obtain sensible results. No existing

simulator is able to generate short reads with depth coverage required for MSA studies since they were designed with different goals in mind. The significant increase in data requirements demands more scalable algorithms with respect to read depth coverage for read generation in order to produce simulated data in a satisfactory amount of time for practical usage.

Lastly, as each NGS technology employs a different methodology to accomplish the sequencing process, instrument-specific error modeling schema is required for valid data generation. Simulators currently provide no support for the instrument Ion Proton™ (Life Technologies) which incorporates semi-conductor technology.

1.4 Contributions

In this thesis, we develop a novel short read simulator (MutSim) that features a high level of generality with respect to modeling and also focuses on precision of data generation. In the design of this simulator, both instrument error modeling and mutational variation profiling are incorporated. A clear distinction between them is enforced from the user perspective, but they are ultimately integrated into a consistent profile during read generation. This design decision is based on the fact that mutation data is universally interpreted independent of the sequencing technology, and yet, at the time of read generation both types of variations should be applied. Concisely, the mutation profile acts as an optional layer on top of the instrument error profile.

Furthermore, Scalable standards and algorithms enable this simulator to meet the requirement for high read depth coverage posed by MSA studies in the area of short read simulation. MutSim's design and implementation meet the modern requirements of software design including scalability, maintainability and portability. More specifically, MutSim is built on top of the Java Virtual Machine (JVM), leveraging its performance maximization and

portability of developed software. Proper use of industrial-strength data structures and careful algorithm design decisions enable this simulator to operate efficiently, while satisfying the requirement for read generation with high read depth coverage.

In addition, the software structure of MutSim exploits object-oriented design coupled with modern software design patterns, enhancing modularity thereby facilitating system maintenance and extension to future sequencing technologies. In other words, modular parts and well-defined interactions between them while avoiding excess coupling which is a requirement of maintainable software. This work not only addresses the challenges above but also focuses on providing a core framework that is easily extendable as more requirements in simulation appear in future. Moreover, this project has been released in open-source with the intention to provide access for all researchers to contribute to this research or use it as basis for future work.

1.5 Organization of Thesis

Chapter 2 is composed of introduction sections relevant to concepts from biology, next generation sequencing, and mutation spectrum analysis. A few existing well-established and pertinent short read simulators and their contribution to the literature are also discussed in Chapter 2. Next, Chapter 3 details the simulation requirements of instrument errors and mutational variations. Moreover, the algorithms and methodologies introduced by this study and how they fit into the structure of MutSim are also discussed in Chapter 3. Simulated and empirical datasets are compared from various aspects in Chapter 4 to validate the results of utilizing MutSim for short read generation. Lastly, this thesis concludes with discussing and summarizing the work presented and providing guidelines for prospective work in Chapter 5.

2 Literature Review

This chapter includes three introductory sections. A brief review of relevant concepts from biology is provided immediately below. After that, next generation sequencing technologies and their impact and applications in bioinformatics and computational biology are reviewed. Introduction to NGS technologies is followed by exploring a more specialized field of application: mutation spectrum analysis. Then, existing well-known simulators suited for NGS short read generation are reviewed.

2.1 Biology

Genomics, in general, is comprised of many levels to analyze the function and structure of genomes. Short read simulation is mainly concerned with concepts at the DNA level. This section is dedicated to explaining the relevant concepts at this level to provide context to the contributions made in this thesis.

From the bioinformatics point of view, DNA constitutes chains of small molecules called nucleotides or bases. Nucleotides have four different types: Adenine (A), Cytosine (C), Guanine (G) and Thymine (T). Due to the hydrogen binding rules and nucleotides internal structure, A tends to form base pair with T, and G with C. DNA molecules are double stranded and, because of the mentioned tendencies, the forward strand and the reverse strand are complementary to each other. The reverse strand is often referred to as the "complement" strand. Figure 2 represents a double stranded DNA sequence. Because of the carbon terminal atom in the backbone structure, each DNA strand has two ends which are called 3' and 5', respectively. Double stranded DNA strings are called DNA helices by virtue of their physical characteristics.

Figure 2: Sample double stranded DNA region

Dashes in between the two strands indicate the bindings of nucleotides which form base pairs. Each strand has two ends, and they are reversed. The building blocks of a DNA strand are A, C, G and T. (Reproduced from [47])

Chromosomes are comprised of long DNA helices, and the genome is the complete DNA of an individual. For instance, the human genome contains 23 pairs of chromosomes, and almost all cells of the body contain a complete genome. Genes are segments of DNA which contain heredity information. This information is typically used to build protein structures through the processes of transcription and translation.

Transcription is a biological process during which a DNA helix segment is copied into a single RNA strand. As a result, all Thymine (T) nucleobases are transformed to Uracil (U). Not all arbitrary segments of genome are typically transcribed into an RNA strand. In fact, only gene expressive fragments (i.e. segments that comprise a gene) and a number of classes of non-coding RNA molecules are normally transcribed *in vivo*. When a gene is transcribed, a messenger RNA (mRNA) molecule is created.

Translation is the biological process by which a cell produces a protein from a mRNA molecule. In this process, more specifically, a mRNA strand is transformed into a protein sequence. Proteins are made of chains (heteropolymers) of amino acids. Bearing in mind that 20 different amino acids exist, protein sequences can be imagined as strings containing 20 different alphabetic letters, each letter representing an amino acid name. Codons represent consecutive triples within RNA sequences that encode amino acids based on genetic codes presented in Table

1. Figure 3 illustrates a sample DNA genetic sequence transcribed into a mRNA and then translated into the corresponding protein sequence. This process is also referred to as gene expression.

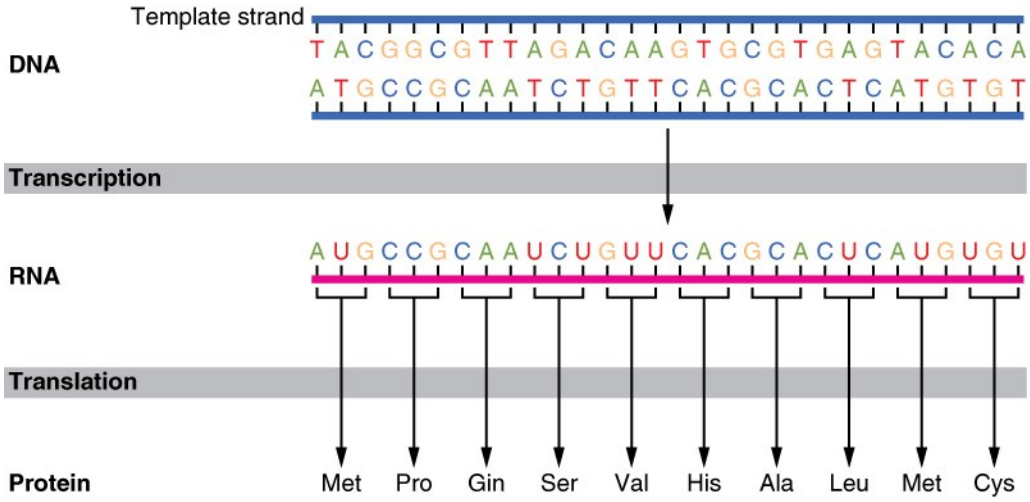

Figure 3: Sample Representation of Gene Expression

Represents a DNA template transcribed into RNA and then translated into protein. This process is also called gene expression. (Reproduced from [48].)

The concept of gene expression is important in MSA since results of mutations are reflected at the protein level although mutations themselves occur at the DNA level. In fact, mutation types are determined by codon changes. More explanations are provided in section 2.3 about different types of mutations.

2.2 *Next Generation DNA Sequencing*

DNA sequencing was first introduced in 1970's by Sanger [16]. At that time, the technology was primarily aimed at creating reference genomes for modelling several organisms (such as *Drosophila melanogaster, Escherichia coli,* and *Mus musculus*). Sanger sequencing technology underwent frequent improvements in the following decades to achieve faster and

more accurate DNA sequencing. The Human Genome Project was conducted using this technology which took over 10 years to sequence a complete human genome of 3 billion base pairs [17]–[19]. This project remains the most collaborative work and, arguably, the greatest scientific discovery of the century. Sections of the human genome were sequenced in many research institutes separately across the globe and successfully mapped to a cohesive framework to render the first complete genome. Sanger sequencing requires *in vivo* cloning and amplification which involves numerous periods of preparation to become accessible, and also the cyclic sequencing using electrophoresis allows only one read/sample at a time which is quite time consuming.

First Position	Second Position: U	Second Position: C	Second Position: A	Second Position: G	Third Position
U	UUU Phe / F	UCU Ser / S	UAU Tyr / Y	UGU Cys / C	U
	UUC Phe / F	UCC Ser / S	UAC Tyr / Y	UGC Cys / C	C
	UUA Leu / L	UCA Ser / S	*UAA STOP*	*UGA STOP*	A
	UUG Leu / L	UCG Ser / S	*UAG STOP*	UGG Trp / W	G
C	CUU Leu / L	CCU Pro / P	CAU His / H	CGU Arg / R	U
	CUC Leu / L	CCC Pro / P	CAC His / H	CGC Arg / R	C
	CUA Leu / L	CCA Pro / P	CAA Gln / Q	CGA Arg / R	A
	CUG Leu / L	CCG Pro / P	CAG Gln / Q	CGG Arg / R	G
A	AUU Ile / I	ACU Thr / T	AAU Asn / N	AGU Ser / S	U
	AUC Ile / I	ACC Thr / T	AAC Asn / N	AGC Ser / S	C
	AUA Ile / I	ACA Thr / T	AAA Lys / K	AGA Arg / R	A
	AUG Met / M	ACG Thr / T	AAG Lys / K	AGG Arg / R	G
G	GUU Val / V	GCU Ala / A	GAU Asp / D	GGU Gly / G	U
	GUC Val / V	GCC Ala / A	GAC Asp / D	GGC Gly / G	C
	GUA Val / V	GCA Ala / A	GAA Glu / E	GGA Gly / G	A
	GUG Val / V	GCG Ala / A	GAG Glu / E	GGG Gly / G	G

Table 1: Genetic Code

Codons represent consecutive triples within RNA sequences that encode amino acids. Here genetic codes are used to determine how genes are expressed at protein level as mutations reflect their structural behaviors on cells at this level. (Reproduced from [49])

Over the past decade a number of new technologies have emerged in order to make DNA sequencing more affordable and less time consuming. A typical NGS sequencing process involves five steps as illustrated in Figure 4. The DNA template to be sequenced is firstly isolated from the cell and fragmented into millions of pieces. Then, instrument-specific adaptors are attached to the DNA fragments. The first three steps are often referred to as *library preparation*. Next, the fragments are cloned and amplified (i.e. multiple copies are made of each fragment). Subsequently, the resulting DNA fragments are inputted to the instrument for actual sequencing and determination of genotype compositions. After that, short reads representing the many DNA fragments are available for NGS data analysis. These steps are slightly different among new technologies. However, novel DNA fragmentation procedures and *in vitro* adapter ligation of adaptors has made library preparation less costly and quicker [20]. Furthermore, new polymerase chain reaction (PCR) techniques including emulsion PCR, used in pyro-sequencing, semi-conductor sequencing, and sequencing-by-ligation, and bridge PCR, used in sequencing-by-synthesis, provide faster and less expensive clonal amplification [1], [21]. In general, the employment of microarrays and reduction of sample sizes have enabled massively parallel sequencing and have introduced high-throughput next generation sequencing instruments [1]. Table 2 summarizes a number of well-known commercial platforms and their underlying sequencing technology. Each vendor also offers diverse instruments for a variety of applications ranging from whole genome, to transcriptome, to small targeted sequencing.

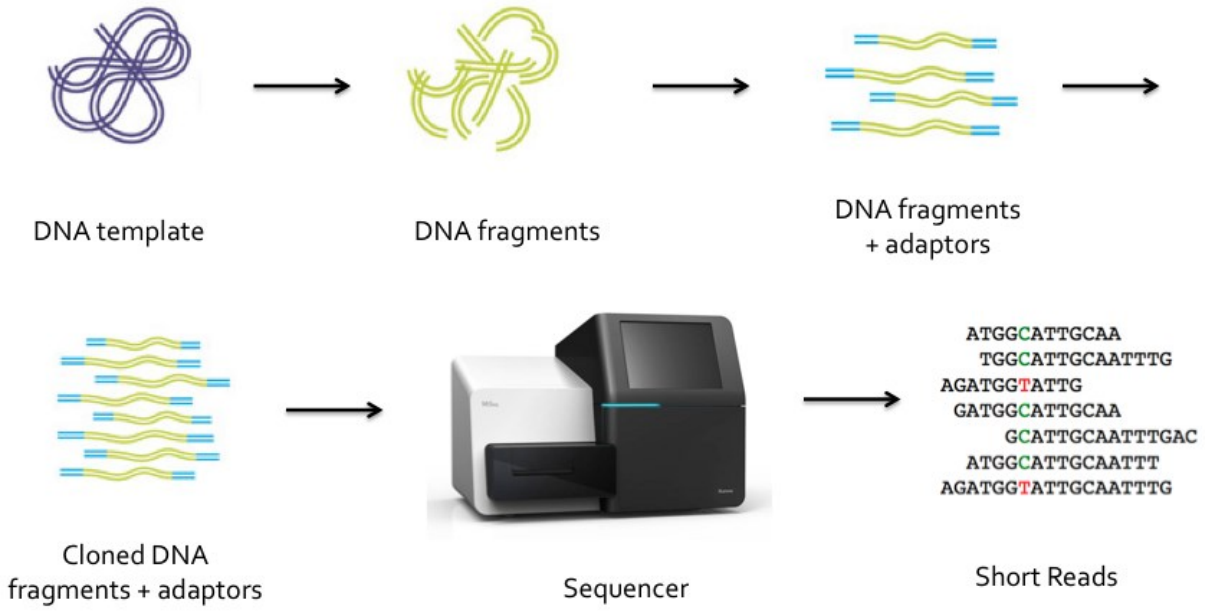

Figure 4: Next Generation Sequencing Process

DNA template is firstly fragmented and instrument-specific adaptors are attached to fragments. Then, they are cloned and amplified in order to be fed to the sequencer. Data in form of short reads is generated after sequencing. (Partially adapted from [50])

NGS data analyses are now branched into numerous fields as sequencing has become less expensive and specialized sequencing instruments have been developed for more targeted analyses. Sequencing applications fall into either whole-genome sequencing, re-sequencing, *de novo* sequencing, quantification of gene expression, transcriptome analysis, paleogenomics, metagenomics, or epigenetic changes. Mutation spectrum analysis is mainly concerned with topics and concepts related to gene expression and re-sequencing.

Pyro-sequencing and semi-conductor sequencing technologies use emulsion PCR [3], [22]. Figure 5 represents DNA emulsion PCR library preparation leading to production of many clonally amplified DNA templates. This provides more accuracy in base reading since these technologies are not sufficiently sensitive for a proper detection of bases if only one single template is present. Although they both use emulsion PCR for library preparation, their

sequencing mechanism is almost completely different leading to various classes of instrument induced variations on the short reads.

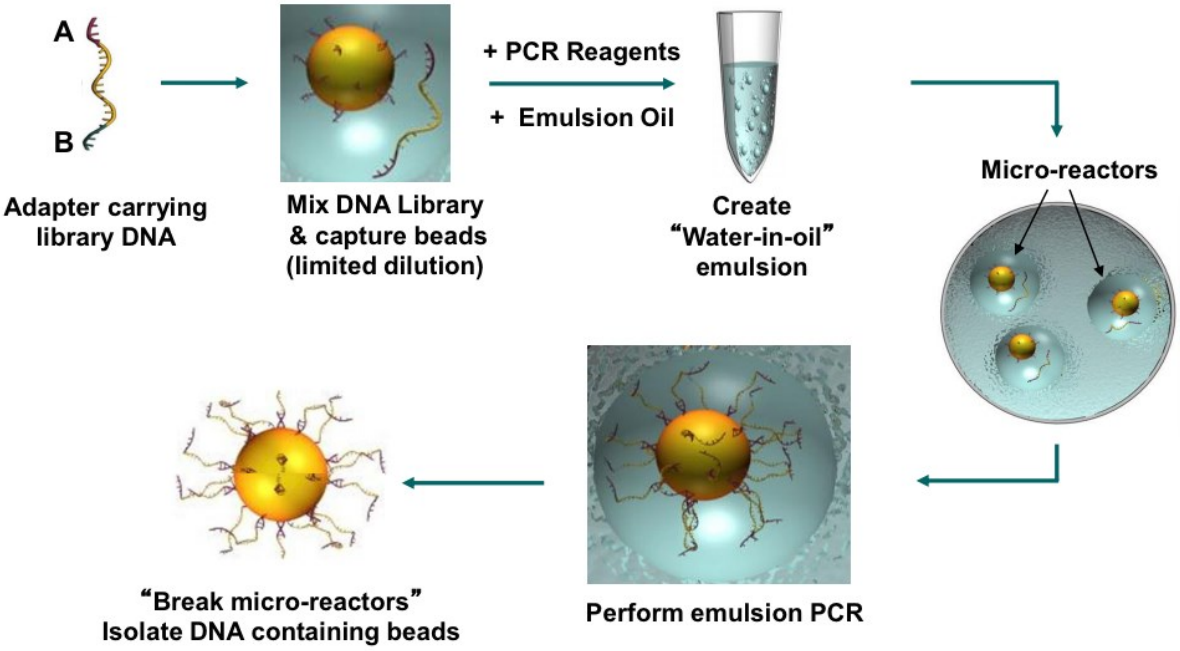

Figure 5: PCR Emulsion Illustration

Illustrates emulsion PCR library preparation used in pyro-sequencing and semi-conductor sequencing technologies. Millions of clonally amplified DNA templates are generated from one. This results in more accurate base detection since ion sensors employed in semi-conductor technology are not sufficiently sensitive to correctly identify bases using only one DNA template [51].

Table 1 Comparison of next generation sequencing platforms

Company	Sequencing Principle	Detection	System platform	Read length (bp)	Number of Reads	Time/run	Throughput/run	Accuracy	Machine cost ($)	Advantage	Disadvantage
Illumina	Reversible terminator sequencing by synthesis	Fluorescence/ Optical	HiSeq 2500/1500	36/50/100	3 billion (SE)	2~11 days	600 GB	> 99%	740,000	Very high throughput; Cost-effectiveness; Steadily improving read lengths; Massive throughput	Long run time; Short read lengths; Expensive instrument; Lower error rate
			Genome Analyzer IIx	35/50/75/100	320 million (SE)	2~14 days	95 GB	> 99%	250,000	High throughput; The most widely used platform	Low multiplexing capability of samples
			MiSeq	25/36/100/ 150/250	17 million (SE)	4~27 hours	8.5 GB	> 99%	125,000	High throughput; Cost-effectiveness; Short run times; Appropriate throughput for microbial applications; Minimal hands-on time; High coverage	Short read lengths
Roche	Pyrosequencing	Optical	454 GS FLX+	700	1 million	23 hours	0.7 GB	99.997%	450,000	High throughput; Longer read lengths; Short run times; High coverage	Appreciable hands-on time; High reagent costs; Higher error rate in homopolymers regions
			454 GS Junior	400	1 million	10 hours	0.035 GB	> 99%	108,000	Longer read lengths; Short run times	
Helicos Biosciences	Single molecule sequencing	Fluorescence/ Optical	Heliscope	25~55 (average: 32)	600~800 million	8 days	37 GB	99.99%	999,000	Single-molecule nature of technology; Non-bias representation of templates for genome	Expensive instrument; Very short read lengths (increase cost and difficulty of assembly); Higher error rate
ABI Life Technologies	Ligation	Fluorescence/ Optical	5500 SOLiD	75+35	1.4 billion	7 days	90 GB	99.99%	350,000	High throughput; Lowest reagent cost	Long run times; Very short read lengths (increase cost and difficulty of assembly)
			5500xl SOLiD	75+35	2.8 billion	7 days	180 GB	99.99%	595,000	Very high throughput; Low error rate; Massive throughput	
	Proton detection	Change in pH detected by Ion-Sensitive Field Effect Transistors (ISFETs)	Ion Personal Genome Machine (PGM)	35/200/400	12 million	2 hours	2 GB	> 99%	80,000	Short run times; Low cost per sample; Appropriate throughput for microbial applications; Direct measurement of nucleobase incorporation events	Appreciable hands-on time; High reagent costs; Higher error rate in homopolymers (sequential washing steps)
			Ion Proton Chip I/II	Up to 200	60-80 million	2 hours	10 GB / 100 GB	> 99%	243,000	Short run times; Flexible chip reagents	Instrument not available at time of writing
Pacific Bioscience	Real-time, single molecule DNA sequencing	Fluorescence/ Optical	PacBio RS	Average: 3000	~50 K	2 hours	13 GB	84~85%	750,000	Short run times; Very long read lengths; Low reagent costs; Simple sample preparation	No paired reads; Highest error rates; Expensive instrument; Difficult installation
Oxford Nanopore	Nanopore exonuclease sequencing	Electrical Conductivity	gridION	Tens of Kb	4~10 million	According to experiment	Tens of GB	96%	According to experiment	Extremely long read lengths; Low cost of α-HL nanopore production; Customization; No fluorescent	4% error rates; Cleaved nucleotide may be read in the wrong order; Difficult to fabricate a device with multiple parallel

Table 2: Next Generation Sequencing Platforms

Illustrates most well-known NGS sequencers and the technologies they use. Mainly, each technology features a combination of characteristics suitable for different DNA analysis. Key differences among technologies are speed, cost, accuracy and read length. (Reproduced from [52])

Ion Torrent products, including the Ion Proton™ instrument, employ semi-conductor technology [23]. In this technology, after DNA fragmentation and PCR emulsion, sequencer-specific adaptors are attached to the both ends of the fragments. Adaptors are solely used for the sequencing process and they are immediately discarded thereafter. For each sequencing experiment run, a new Ion Chip™, which contains millions of micro wells, must be used. Each well contains a different short segment of DNA to be sequenced (called a template sequence). Sequencing is accomplished by iteratively building the complementary sequence to each template sequence, and detecting each base pair as it is added to the growing complementary strand. This is accomplished by repeatedly exposed all wells on the chip to different nucleotides (A,C,G,T). When the leading unpaired nucleotide on a template DNA sequence in a well is exposed to a complementary base pair, an ion-sensitive sensor embedded in that well detects this event by monitoring the pH of the solution in the well [23]. The sequencing process used in this technology takes 2-4 hours for the production of 60-80 million reads.

Common in all sequencing technologies, a Phred quality score is assigned to each nucleobase in each sequenced read representing the sequencer's confidence in reading that nucleobase on the DNA fragment [24]. Sequencers use various forms of Phred scores. The Ion Proton™ presents quality scores using Phred+33, which ranges from 0 to 40. Phred scores are logarithmically related to the probability of errors. For instance, Phred scores of 20 and 40 indicate 99% (1 in 100 incorrect) and 99.99% (1 in 10,000 incorrect) accuracy of base sequencing, respectively.

Once data are available after the sequencing process, pipelines for NGS analysis involve the alignment of short reads to the reference sequence and reconstruction of the sequenced DNA template. DNA alignment for NGS short reads is fairly complex and numerous studies have been

conducted in order to compensate for miscellaneous variations introduced by different technologies (e.g. [25]–[27]). Figure 5 illustrates the general short read alignment problem. Base pair substitutions, deletions, insertions, and other edits of the newly sequenced DNA template may make the process of read alignment to a reference genome more challenging and time consuming. Primarily, the difference between alignment algorithms is the trade-off between sensitivity and speed. Several parameters (e.g. number of allowed mismatches, size of gaps, etc.) can be manually set for a more fine-tuned alignment, although it is rarely required nowadays since alignment profiles are typically provided with the aligner packages (e.g. very sensitive or fast).

Another aspect in choosing the proper aligner is the type of short reads available (i.e. single-end reads or paired-end reads) and also size of reads. Well-established aligners (such as bowtie and Burrow's Wheeler Aligner) offer more flexibility in this aspect to cover a variety of reads types and sizes [25], [26].

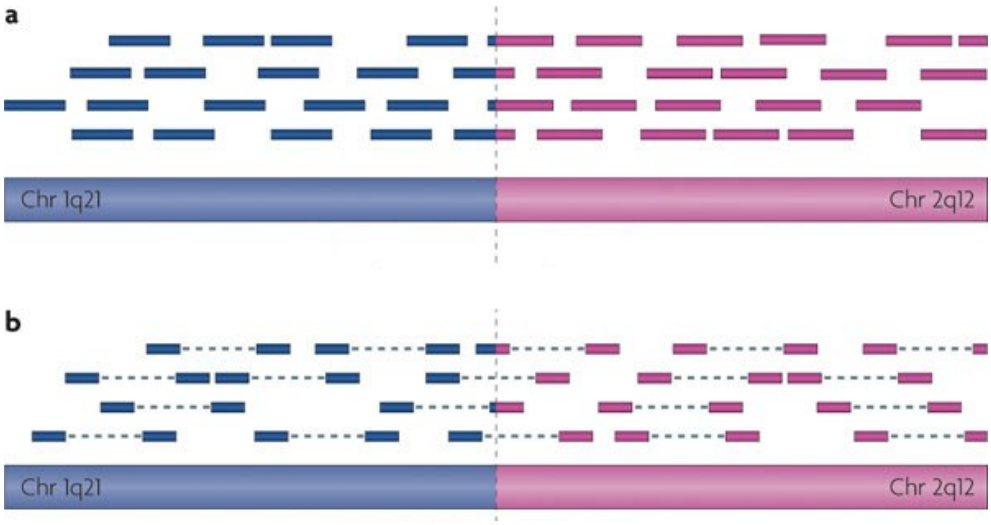

Figure 6: Sample Representation of Different Short Read Types

Different short read types as produced by NGS platforms [53]. Single-end reads, indicated by bars (a) are aligned to 2 chromosomes having different colors. In (b), paired-end sequencing mode is used in which only the first and last few base pairs of each fragment are sequenced (i.e. bars in figure). The distances between the 2 reads (i.e. dashed lines in figure) are also known. The latter protocol allows more repetition of overlapping regions and more accurate alignment.

Different types of reads are illustrated in Figure 6. Single-end reads are sequenced from either the 3' or 5' end of the DNA template while paired-end reads are sequenced from both 3' and 5' ends. Paired-end reads produce more redundant data, but prior knowledge of the distance between paired-end reads allows more precise alignment for repetitive overlapping regions. Alignment of paired-end reads, however, requires more sophisticated algorithms.

2.3 Mutation Spectrum Analysis

Mutation spectrum analysis seeks to characterize the mutational effect of a substance or condition. Typically, this involves identifying all induced mutations within the sequenced DNA following exposure to the mutagen. Most simple mutations, or sequence edits, are base pair substitutions and indels. Base pair substitution is the change of a nucleobase on the forward strand and its corresponding nucleobase on the reverse strand. Base pairs may also be inserted or deleted from the sequence, referred to as insertions or deletions, respectively or indels for short. Indels may take any size, but the likelihood of occurrence of an indel significantly drops as indel size increases [28], [29]. Figure 7 represents sample sequence edits including a base pair substitution, an insertion, a deletion, and a few other forms.

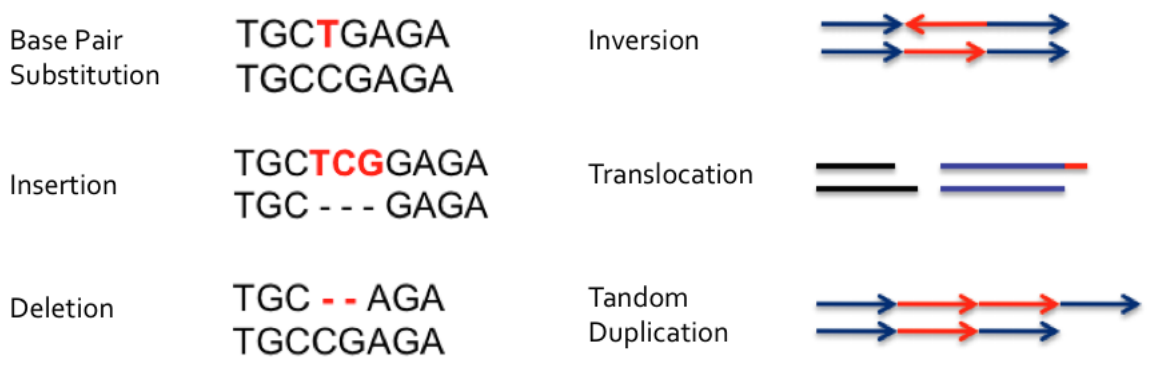

Figure 7: Sample Sequence Edits

The most notable and well-studied sequence edits are base pair substitution and indels (insertions and deletions). Indels bigger than 2 bp in length and other edits are rare. This figure is not a complete list of sequence edits.

Mutations are classified into three categories with respect to structure: silent, missense, and nonsense. Silent mutations are base pair substitutions or indels that result in one codon transforming into an equivalent codon (i.e. the encoded amino acid does not change). Since the encoded protein is not altered, these mutations have the least impact on cellular processes. The second class of mutations are missense in which a base pair substitution or indel at the DNA level results in a modification of the encoded amino acid at the protein level. Depending on the location of the mutation on the genome and the resulting amino acid type, missense mutations may or may not cause diseases, other ill-effects, or even beneficial effects. Lastly, when a change at the DNA level leads to the production of a premature stop codon, a nonsense mutation occurs. This type of mutation is the most dangerous since the resulting protein sequence is truncated and is therefore likely to no longer function properly.

A typical MSA experiment involves two groups of mice, i.e. control and treated. The treated group is given a dose of a suspected mutagen while the control group receives only the

solvent used to dissolve the mutagen. After exposure, mice are sacrificed and tissues are collected. The DNA of the tissue of interest is then isolated, packaged into a phage and then transfected to *E. coli* where a positive selection technique (e.g. p-gal or x-gal) is employed to select mutant plaques (where a mutant plaque is a colony of genetically identical *E. coli* in which the gene of interest contains a mutation) [30]. Mutant frequency is calculated by counting mutant plaques, and the gene of interest is then sequenced for all sampled plaques for both groups. Then, data is analyzed to detect differences in mutations between the control and treated samples in terms of mutation pattern, location, and frequency.

MSA data have traditionally been obtained by Sanger sequencing, a technology that is quite reliable but of low throughput [31]. The advent of NGS provides opportunities to characterize MSA more quickly and at a lower cost. Sanger sequencing of longer genes (e.g. *LacZ* and *LacI*) requires multiple sequencing reactions and is a laborious task. Therefore, MSA has traditionally been limited to smaller reporter genes (e.g. *cII*). Such a limitation is not present for NGS. However, NGS introduces new challenges, such as shorter reads and larger gaps, making sequence alignment more difficult and also increasing the need for careful data pre-processing, quality assurance, and analysis.

In transgenic mutation assays, the estimated mutant frequency determined during plaque collection can be rendered incorrect as a result of clonal expansion (CE). This phenomenon occurs when multiple mutant plaques arise from a single true mutation in the mouse (i.e. the same mutated *lacZ* gene is inadvertently transfected into multiple *E. coli* colonies).

As a consequence, multiple identical mutations (same type and at the same position) may be identified during MSA. Since mutations are very rare, it is extremely unlikely to observe two identical mutations among the 100 or so sequenced plaques. These observed repeated mutations are much more likely to be, in fact, due to clonal expansion. Therefore, it is important to look for instances of CE and remove these duplicate samples before proceeding with the MSA. In this case, typically only a single "independent" mutation is included and the rest is referred to as "recurrent" mutations and are excluded from subsequent analyses. Pyro-sequencing and semi-conductor sequencing technologies are more likely to result in noticeable clonal expansion due to their library preparation, template amplification, and sequencing procedures [28], [32].

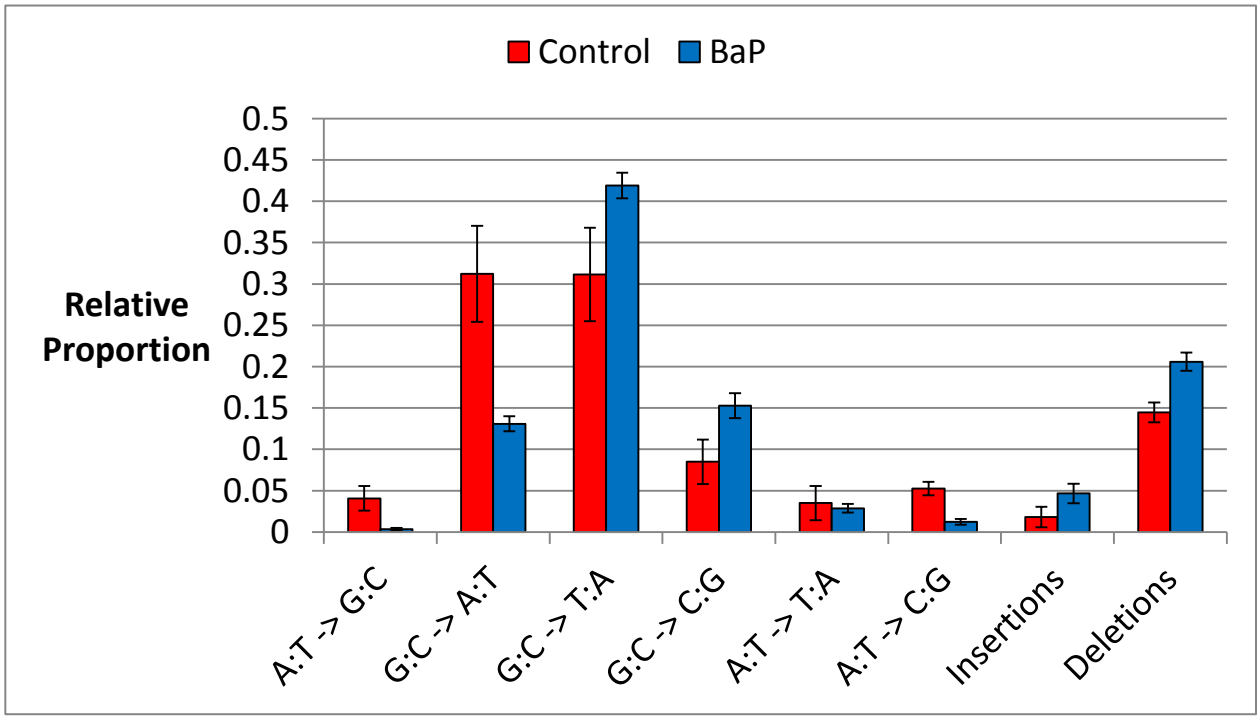

Figure 8: Sample Representation of Relative Proportion of Mutations

Illustrates global proportion of mutations across *LacZ* gene for *BaP* induced and controlled samples. Only independent mutations are considered to represent the results without any instrument or library preparation specific errors. The error bars represent the standard error between samples. (Reproduced from [28])

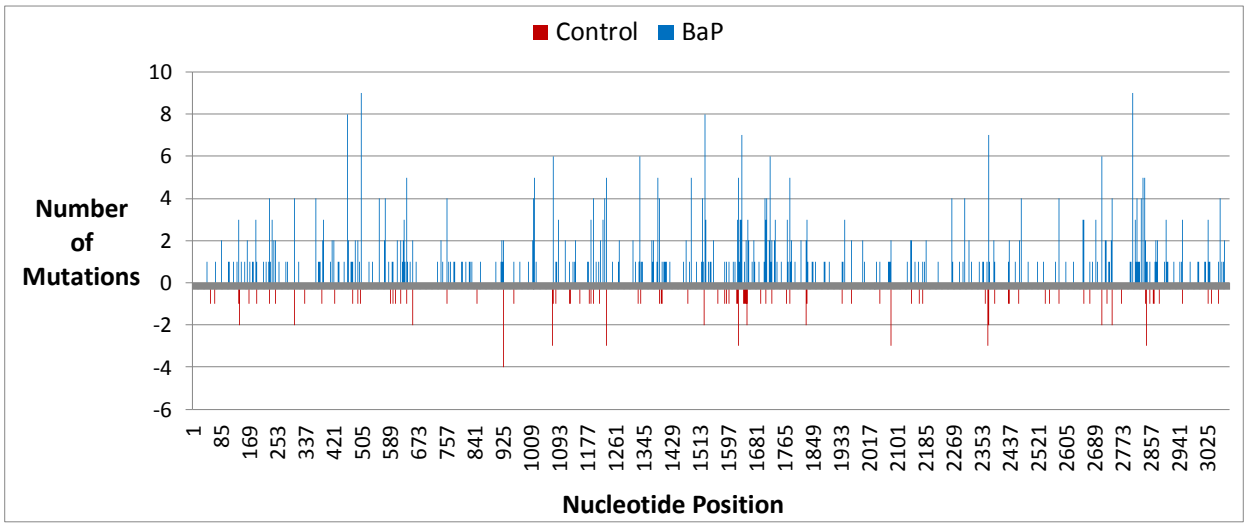

Figure 9: Sample Representation of a Positional Mutation Spectrum

In addition to global proportions, biologically-induced mutations aroused from a mutagen, are presented by positional chance of occurrence on the gene. This figure shows the number of independent mutations after conducting a mutation spectrum analysis in which Ion Proton instrument was used to perform the sequencing process [28].

However, clonal expansion can be successfully corrected by employing standard statistical quality assurance tests before interpretation of the mutation spectrum [33]–[35]. As MSA assays interpret their results independent of technology used, ideally only independent mutations should be used for an appropriate comparison. A sample representation of global mutation proportions is illustrated in Figure 8. Furthermore, Figure 9 shows positional base pair substations of the same study.

2.4 Short Read Simulation

Although MutSim provides unique features, it also has common characteristics with a number of existing widely-used simulators. Here we discuss these simulators by explaining their core properties in short read simulation and their support level for modeling instrument errors and mutational variations, where applicable. Additionally, the advantages and drawbacks of utilizing each for short read simulation are further described. This section is completed by stating

the critical requirements of NGS data simulation in the area of MSA, and why existing simulators are inapplicable for such data simulation.

ART uses a random approach for placing erroneous base pair substitutions and parameterizes error models based on empirical data [11]. It, however, shapes its error model by taking pure empirical data and applying the exact observed values when simulating new data, which is time consuming and impractical for producing MSA data requiring high coverage. This simulator provides support for short read generation of Roche 454™, Illumina™, and SOLiD™ instruments. The latter two employ completely different technologies from the Ion Proton™, but the Roche 454™ performs the sequencing process by pyro-sequencing technology which shares a number of characteristics with semi-conductor technology as discussed in section 2.2. Pyro-sequencing is more likely to produce indel type errors which is also observed in reads sequenced by semi-conductor given the cyclic read process they both perform. The types of errors, however, vary due to different base reading techniques used by each technology. The semi-conductor technology determines the base by measuring pH in each well using an ion-sensitive layer. This results in different indel errors with respect to length and location on reads compared to those caused by pyro-sequencing [24]. Furthermore, *ART* provides no support for applying mutational variations. Impractical and pure empirical data modeling, no explicit and proper support for semi-conductor technology, and, most importantly, the absence of a component to apply biologically-induced mutations, makes this simulator unsuitable for simulation of NGS-MSA data described in this study.

pIRS offers a complex model for base substitution and indel calling based on read cycles and pure experimental data [12]. More specifically, the error profile is modelled by including several factors including read cycle, reference base, called base, and called quality. Base

substitutions are determined by only taking the current read cycle into account while quality scores are determined by both the current and previous cycle. Read cycles in the semi-conductor technology used in Ion Torrent™ products, such as the Ion Proton™, are an insignificant source of error when compared with indels arising from homopolyer regions [22], [24]. While *pIRS* performs well for other technologies, its error modeling schema is not well-suited to instruments using pyro-sequencing such as the Roche 454™ products or semi-conductor sequencing such as the Ion Torrent™ instruments. Moreover, *pIRS* solely provides paired-end read simulation while studies in MSA are typically conducted using single-end sequencing.

WgSim, even though it is a small simulator and its feature set is limited, is widely used for testing and education purposes [14]. This simulator provides support for simulating diploid genomes with single nucleotide polymorphism (SNPs) in the form of single base pair substitutions and indels. Sequencing errors are simulated by following a uniform distribution and by supplying an overall error rate. Although this simulator is broadly used, given its lack of support for technology-specific errors and mutational variants, it does not meet the requirements of simulating NGS short reads for MSA studies.

XS provides introductory support for mutational variation, but only as an overall rate (see section 3.4.2 for a complete mutation model). Furthermore, *XS* takes a different approach to read generation where no reference sequence is provided, but only sequence composition proportions [13]. This simulator features a fair set of options to generate reads for multiple technologies. More specifically, *XS* is mainly concerned with speed and portability with the goal of providing data simulation for large-scale projects via cloud services. Simulation is accomplished by simply setting the desired gene length and global nucleobase compositions, since reference is not supplied. Exact or approximate repeats can also be optionally added during the simulation. An

interesting aspect observed in the development of this simulator is the break-down of the problem into multiple independent software modules. The modules are re-used for simulation of different technologies where applicable. However, since in generating mutation data, a specific reference gene and inclusive mutation model must be specified, *XS* does not qualify for NGS-MSA simulation. Furthermore, *XS* does not provide support to indicate the desired read depth coverage.

ArtificialFastqGenerator features a new approach in order to bias data based on manipulating GC-content percentage [15]. This simulator is reported to be fast and involves numerous innovations. In general, the simulation is approached by giving the user the option to supply many parameters instead of tuning them using sequenced data. Specifics of paired-end reads (i.e. read length, template length, and gap size between pairs) can be supplied for a more realistic simulation. The desired read depth coverage can be specified for relatively low levels (below 50x). It, however, is unable to provide the read coverage typically used by MSA experiments due to memory requirements and in-scalable algorithms used for read generation. Furthermore, this simulator only supports paired-end reads for Illumina™ technology. Although findings are useful in this simulator, given the reasons outlined above, it is ultimately unsuitable for NGS data simulation for MSA studies.

While the above mentioned simulators offer many features and are effective in their target areas, they do not meet the requirements of NGS-MSA experimental data simulation. In particular, none of these simulators are capable of producing reads for high coverage since they were typically designed for low-coverage genome-wide experiments. As a consequence, their memory consumption and/or runtime make it impossible to simulate high coverage experiments with any of these methods. Furthermore, they do not permit a detailed mutation modeling

schema for simulating the effects of arbitrary mutagens. Lastly, no simulator currently exists specifically targeted at the Ion Proton™ NGS instrument or in general Ion Torrent Products.

3 Read Simulation for Mutation Spectrum Analysis Studies

3.1 Introduction

When using NGS for mutation spectrum analysis, two distinct classes of variations appear in short read DNA sequences: those arising due to sequencer errors, and those caused by biologically-induced changes to the underlying sequence due to the mutagen or spontaneous mutations. The first type of variant is caused by instrument errors, where the sequenced short read does not accurately reflect the DNA sequence under study. Characterizing such errors must be approached in a vendor- and instrument-specific way as each vendor uses a different underlying technology to perform the sequencing process, and every instrument offered by each vendor applies slight modifications to the data collection protocols, targeting each instrument to specific types of analysis [3]. For instance, the Ion Torrent™ instruments (Life Technologies) employ semi-conductor technology which captures data by utilizing ion-sensitive chips, and observing their reactions during DNA polymerase synthesis [23]. This approach is more susceptible to producing indel errors compared to base pair substitution errors. Since the reactions of each base type are sensed using an electronic signal, determining the number of identical consecutive bases is difficult, which often leads to improper detection of homopolymer repeat lengths in the sequence under study [22], [24]. Furthermore, Life Technologies feature a range of instruments targeting applications such as targeted, exome, transcriptome, and genome sequencing, with each instrument differing in run time, read length, and throughput to meet various criteria for each analysis type. Since the underlying technologies and short read format is identical, their error models can be accurately incorporated into one integrated profile. MutSim, primarily employs Ion Proton instrument read data generated by two different chip sets and

toolkits for sequencing error profiling. However, the instrument error modeling schema is designed to be flexible and independent of sequencing technology and therefore, with slight modification, it can be used to simulate data from other NGS instruments.

The second category of sequence variation originates from biological factors. In this case, the sequence under study actually differs from the reference sequence. For MSA experiments, the most noteworthy of such variations are mutational variations arising from exposure to a mutagen [36]. Induced human gene mutations *in vivo* are rather difficult to study [37]. However, transgene rodent mutation assays offer a more feasible proxy approach [38]. The MutSim mutation profile schema is modeled after the thorough and position-dependent characterization of mutagens typically produced by such transgene rodent mutation assays [6], [7].

MutSim begins by accepting a reference file in fasta format, a sequencing error profile, and an optional mutation profile. As a result, one or more fastq files are generated based on the number of desired samples [39]. In this section, the overall simulation workflow using MutSim is described. Then, the MutSim read generation approach is described in detail. After that, both profile schemas and their influence on short read simulation is further expanded. Mutation spectrum analysis methods are briefly discussed. Next, the scalability of MutSim with respect to read depth coverage and genome length is investigated. The section concludes with key implementation details of the simulator.

3.2 *Read Generation*

Read generation begins by placing an arbitrary window of fixed-size *windowWidth* on the reference sequence. Figure 10 illustrates the read generation approach designed in this work and implemented in MutSim. Reads are generated within the window until *fill-threshold* % of the

total specified coverage is met (default=5%), then the window is relinquished and a new one is selected. This strategy results in a uniform coverage with systematic fluctuations across the gene which was observed in all datasets (see section 4.3). Both read depth coverage and window size can be optionally set prior to read generation. The first step in generating a read is to determine the read length by sampling from the read length distribution. The range of valid start positions for that read length is calculated, and then the read is placed in the window by drawing a start position uniformly from the valid range. Read coverage and valid window placement ranges are always cached during the overall process, making the window selection highly efficient while maintaining read generation validity and uniformity. Analogous caching of valid positions is also used when placing reads within the window. By pre-calculating and caching effective ranges for read placement, MutSim eliminates any time-consuming trial and error approaches during simulation.

When generating a read, both sequencing errors and mutational variations are intelligently combined and applied to the corresponding reference sequence. More specifically, for each position of the read, sequencing errors are uniformly drawn based on the instrument error profile values for that position. For example, the tendency for the Ion Proton™ instrument to generate indel errors within homopolymer regions is handled by MutSim given its position-specific instrument error profile schema [24] (see section 3.4.1 for an accurate definition of values). Any resulting instrument errors are applied to the reference base at that position within the read. Then, in the case of no errors, the reference base is also passed through another filter in which mutational profile changes may be applied. After that, the generated read is immediately pushed to the buffer to be written to file, significantly minimizing memory usage since reads are regularly released. In other words, reads are produced on the fly, keeping only metrics

representing previously generated data (i.e. cumulative read depth at each position and etc.) and not any detailed read state in memory. Releasing resources pertaining to a read, and only retaining its effect on the overall statistics, enables MutSim to simulate reads for extremely deep coverage as well as for genomic sequences of any length (see section 3.5).

Phred quality scores are simulated using a normal distribution for each genotype and indel type. This enables the profile creator to accurately capture Phred quality scores from experimental data, while featuring a generalized approach that effectively models the error characteristics of multiple technologies.

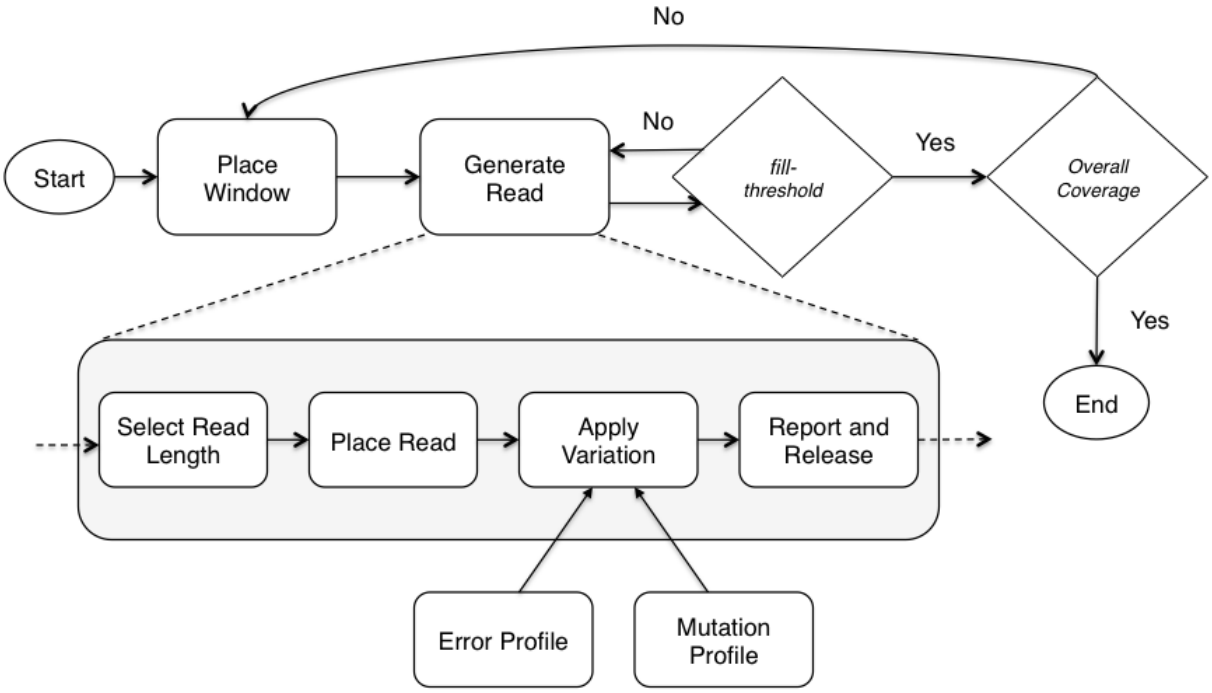

Figure 10: Read Generation Strategy in MutSim

The Generate Read module is further expanded in this figure illustrating its internal components. Concisely, read generation starts with placing a window on the reference gene. Reads are generated for that window until the fill-threshold is met, and another window is repetitively selected continuing this process until the overall coverage is met (see text for a full description of the process).

Furthermore, read length distributions were fit for two different commercially available Ion Proton chips (Ion PIv1 and PIv2), giving the user the option to select the desired chip

specifications. Another chip-specific aspect of Ion chips is the range of well X-Y-coordinates. These coordinates are presented in the header section of short reads indicating the well in which the DNA fragment was sequenced. Although header sections of reads have no effect on downstream analysis, MutSim also simulates this behavior for an authentic and complete format which is required for all fastq file processing tools. A uniform distribution of DNA fragments over the wells on chip was observed in all sequencing runs in this study. Therefore, uniform sampling is performed to assign well coordinates for each simulated short read.

3.3 Simulation Workflow

MutSim accomplishes read generation by accepting a flanked reference gene, an instrument error profile, and a mutational profile (see sections 3.4.1 and 3.4.2). Here, the "flanked" reference sequence includes the reference gene sequence plus some adjacent sequence from the chromosome to either ends of the gene (GenBank: J01636.1).

Figure 11 illustrates the workflow of tasks in order to prepare the data for MutSim and generate NGS short reads for MSA studies. One or more fastq file(s) are first aligned to the flanked reference gene. We used the alignment program *bowtie2* since it provides the option of very sensitive alignment at the cost of time to produce more reliable alignment results and to make a more representative error profile [40]. SAM files contain alignment information as well as all other read information which are converted to BAM format (binary format of SAM) for faster processing [41]. BAM files are sorted by start position along the reference gene (a requirement of pile-up generation) and then indexed for faster access time by other tools. Figure 12 illustrates a number of sample short reads aligned on the *LacZ* gene (910bp –1450bp).

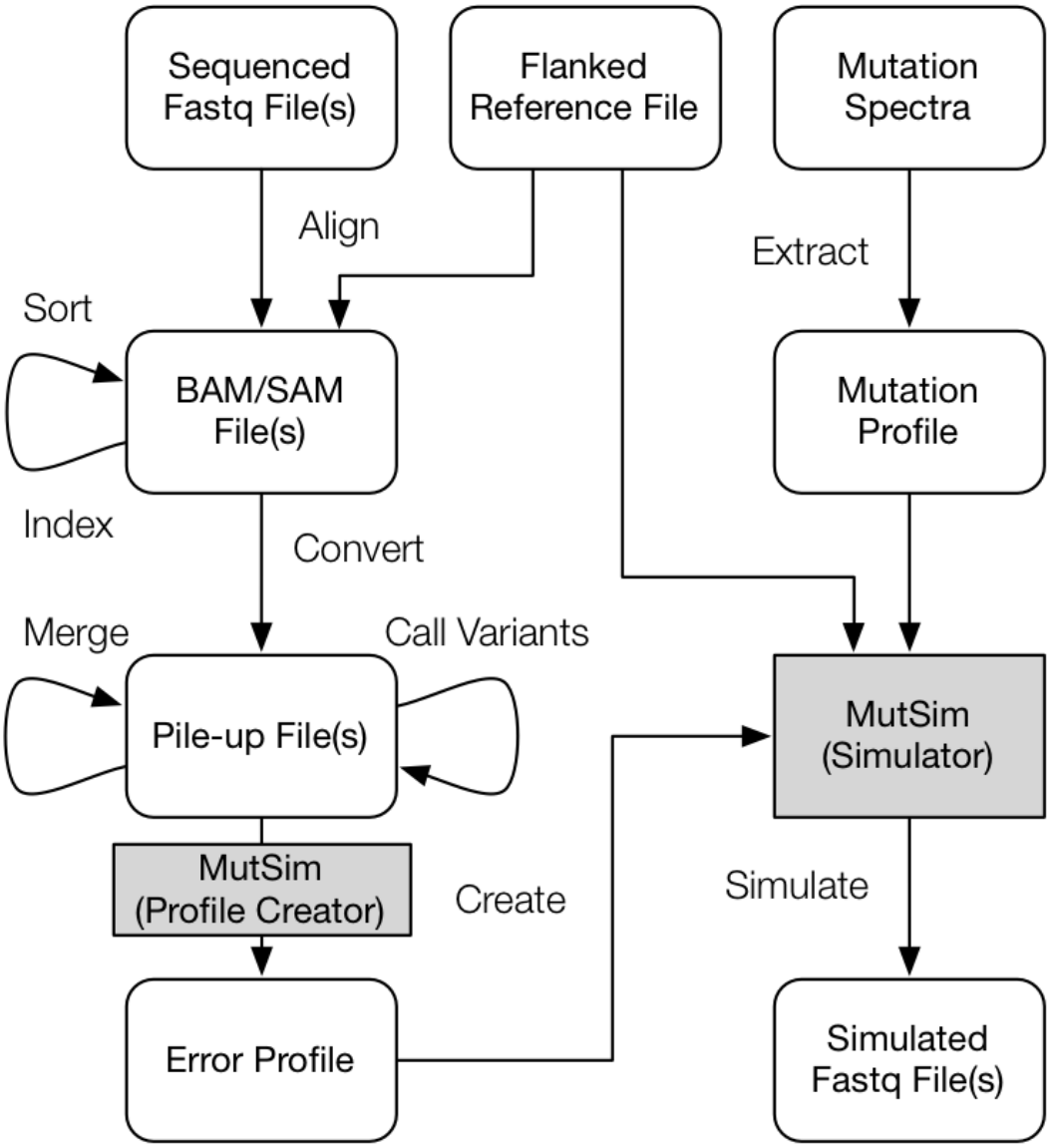

Figure 11: Workflow of Read Generation Using MutSim

Illustrates the workflow of short read generation using MutSim. The flowchart on the left illustrates the creation of an instrument error profile from empirical data. The flowchart on the right illustrates the simulation of short read data. An error profile and a flanked reference gene are required for read simulation. A mutation profile can be conveniently extracted/imported from MSA studies since MutSim is able to directly process this type of data as they are typically represented in such studies.

Using *SAMTools*, raw pile-up file(s) are generated from sorted and indexed BAM files [41]. The pile-up format represents aligned short read information at each position in a base-wise fashion. At this stage, various tools can be used to call and exclude variants from the pile-up file(s), as we wish to focus on instrument-induced sequencing errors. The profile creation module is designed to accept one pile-up file. Therefore, if multiple pile-up files are presented, they are first merged and then fed as input to the profile creation module embedded in MutSim. As a result, an integrated and compound error profile is produced which is later passed into MutSim for applying sequencing errors during simulation. This provides users with the option to

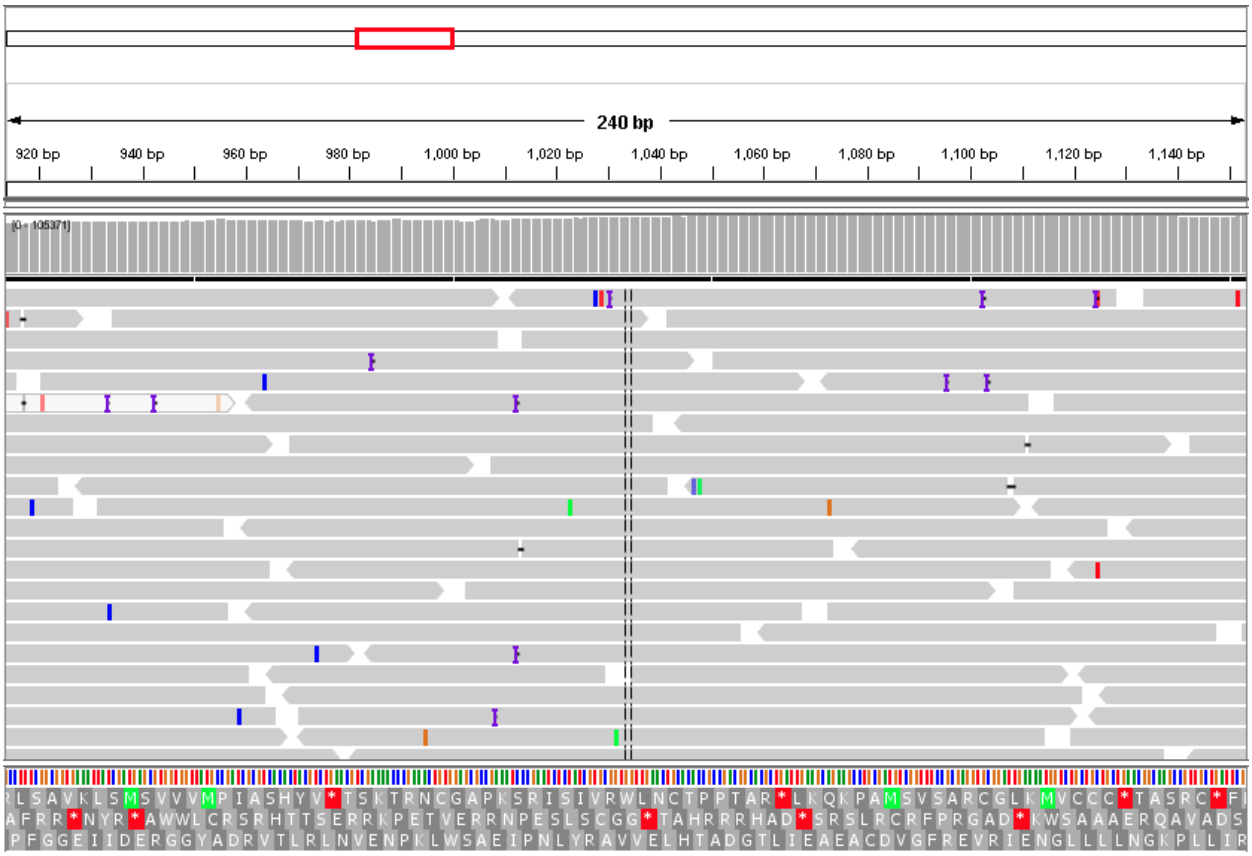

Figure 12: Sample Representation of Aligned Short Reads

Read are produced by the Ion Proton™ instrument from the LacZ gene (zoom: 910bp – 1450bp). Small gray bars at top represent base-wise read depth coverage. The main pane illustrates reads and their aligned locations, indels, and base pair substitutions. Insertions and deletions are indicated by symbol I and The lower pane represents the amino acid sequence resulting from all possible codons of consecutive base 3-tuples.

use the desired and advanced tools for calling variants if required. Pile-up files inputted to the profile creation module must be free of any variants for a more realistic simulation.

In MSA experiments in order to ensure sufficient read depth coverage at both ends of the gene, a flanked version of gene is used. Following this protocol, MutSim is designed to work with flanked version of the gene of interest. Using this approach, realistic coverage simulation is achieved by this design decision as shown in section 4.3.

The mutation profile is the third input to MutSim. Mutation spectra presented in MSA studies can be conveniently imported to form the mutation profile. Minimal effort is required for inputting mutational fingerprint of arbitrary mutagens since MutSim is able to directly process data as represented in most MSA studies. Mutational profile is further explained in section 3.4.2.

3.4 Profile Description

3.4.1 Instrument Error Profile Description

Instrumentation errors are modeled in MutSim using a position-specific profile. The propensity for base substitutions and indels, as well as the quality scores, are captured using observed position-specific frequencies of genotypes, deletion lengths, and specific insertions. The resulting profile is employed to build statistical distributions at the time of read generation. Precise statistical modeling not only saves tremendous data storage and retrieval time, but also introduces stochastic behavior which is essential when multiple samples are generated from a single set of inputs. This deviation between simulation runs is in accordance with observed variations among different actual experiment runs (see section 4). Given the comprehensiveness yet concise design of the error profile, its usage can be further expanded into other similar fields where accurate error modeling is desired.

In order to generate the error profile, MutSim features a profile creation module. Fastq files produced by actual sequencing must first be aligned and converted to a raw pile-up format. This gives the option to the user to call any existing non-sequencing error variations with related tools. For a proper simulation, the pile-up file used to generate an error profile must only contain sequencing error (see section 4.1), so any non-sequencing error variations should be removed prior to profile creation. The profile creation module parses the pile-up file and extracts genotype composition rates, indel types and their respective rates, and Phred quality score distribution parameters for each position. More details of the error profile are provided in Appendix 1.1.

3.4.2 Mutation Profile Description

Mutation spectrum analysis seeks to characterize the types of mutations that a mutagen is likely to cause in a genomic sample. The resulting spectra are complex, comprising of both position-specific characteristics (e.g. hotspot analysis) and also composition-specific characteristics (e.g. nucleotide substitution rates). This complexity of representation reflects the complex biological events which are typically induced by a mutagen. Various computational methods have been developed to obtain and interpret mutation spectra from empirical data [10], [36], [38]. While these approaches differ significantly and use a variety of technologies, they all seek to elucidate true mutation types, their positions, and associated frequencies [7], [28], [38]. Determining true mutation frequencies is highly dependent on the sequencing technology employed and the experiment design used to collect the data. Subsequent analysis of these data can also lead to differences in mutation spectra, but general agreement has been observed in most cases across multiple studies [28], [42], [43].

Another important factor associated with transgenic mutation assays is clonal expansion (CE) which is the phenomenon that occurs when a bacterial cell with a given mutation replicates

to produce multiple clones arising from a single independent mutation. CE biases the portrait of mutation spectra since multiple mutations may actually arise from the DNA originating from a single mutation event within the mouse. In other words, CE leads to recurrent mutations observed on a base shared between the overlapping short reads. Additionally, more than one independent mutation can occur in one position. MutSim is able to model the effects of CE and multiple independent mutations when simulating a MSA experiment. Since in addition to overall mutation frequencies of each type, positional mutation frequency can be supplied for each base in the reference sequence, by assigning the appropriate frequency and proportional genotype compositions CE can be adjusted and independent mutations can be modeled, respectively.

During MSA, when computing the final mutation spectrum, one seeks to identify and discard all sequence variation arising from instrument errors, leaving only true mutations. Furthermore, the final spectrum must be presented in a manner that is independent of technology or any other systematic artifacts caused by the experiment design. However, when simulating read generation for an NGS MSA experiment, instrument-induced sequencing errors, mutagen-induced mutational variations, and all relevant factors arising from the experiment design must be considered simultaneously. MutSim is able to achieve this by effectively integrating both profiles during read generation. More details of error profile are provided in Appendix 1.2.

3.5 Scalability

Although NGS technologies offer many advantages over Sanger sequencing, they also introduce a new set of challenges. Since the DNA sequence is broken into many more pieces during the fragmentation process prior to sequencing, NGS sequencers are able to process many more tiny fragments in parallel taking a considerably shorter time. Consequently, a consistent

increase in read depth coverage is seen among various types of DNA analyses as they make their transition into NGS. The re-construction of the DNA sequence, in contrast, poses more challenges since reads are shorter and more numerous, making their alignment to the reference sequence more difficult, especially if other variations are introduced in the collected data. The alignment of NGS short reads is a well-studied field, and modern tools leverage sophisticated approaches to address this problem [25], [26]. Typical genome-wide analyses are conducted using x20-30 coverage. When conducting mutation spectrum analysis, however, an extreme read depth is required (e.g. x60,000) in order to accurately characterize the mutational fingerprint of a mutagen, owing to the rarity of mutation events.

Among existing NGS simulators, MutSim is uniquely capable of achieving such read depths by introducing a highly scalable approach in its design. As mentioned above, during read generation a single read is drawn from the reference sequence, manipulated using stateful modeling statistics based on the instrument error and mutation profiles, then written to file and immediately released from memory. Keeping any supplementary and unnecessary read specifications in memory is avoided and only concise information about previously generated data, which is also partially influenced by profile schemas, are preserved in memory. In this way, MutSim is extremely efficient and is able to produce authentic data regardless of both read depth coverage and genome length. Numerically speaking, while only consuming less than 150 MB of memory, dozens of GB of read data can be produced. In order to determine the growth in runtime as a function of read depth and genome length, an empirical performance analysis can be utilized [44]. Equation 1 describes the order of growth of simulation runtime. Equation 2 is derived from Equation 1 using the power-law relationship. Equation 3 examines the system-

independent factors of runtime by applying the doubling hypothesis, where we investigate the effect on runtime when doubling the size of the problem (see section 4.6).

$$\boldsymbol{log_2\big(T(N)\big) = b\, log_2\, N + c}$$

Equation 1: Logarithmic Order of Runtime growth

$$\boldsymbol{T(N) = \ 2^c\, N^b}$$

Equation 2: Power-law Relationship Applied to Equation 1

$$\boldsymbol{b = log_2(ratio)}$$

Equation 3: System-independent Factor of Runtime

In these equations, $T(N)$ is run-time, N is the input size, b is the slope of $T(N)$ vs. N in a log-log scale, *ratio* refers to the ratio of two consecutive runtimes where N has doubled, and c is a constant. In brief, the parameters of Equation 2 can be fit based on empirically observed runtimes for varying read depths (see section 4). System-independent effects (i.e. algorithm and input size) determine the exponent b and a combination of system-dependent and -independent effects (i.e. hardware, software, system) determine 2^c in Equation 2. Since the results are interpreted independent of a machine and are instead based on run-time ratios, system dependencies are intentionally factored out in this approach. In other words, by modeling runtime in this way, MutSim will perform the exact same way on any machine with regards to the order of growth of runtime.

3.6 *Additional Implementation Details*

Hash-tables are used throughout the implementation when unordered data retrieval by-value instead of by-index is required, thereby providing fast constant time access as opposed to linear time search in regular arrays. When ordered data access based on value is required, a form of red-black balanced binary tree is used as it offers increased access speed. Furthermore, the *strategy*, *builder*, *composite*, and *facade* design patterns are used to shape consistent associations, to minimize dependency, and to establish proper interactions between modules. Object-oriented design principals and polymorphic class structures with dynamic run-time binding are used throughout the architecture of MutSim. As a result, modules are properly defined, unit tested, and reused. Integrity between modules is preserved, and therefore, maintenance of the overall and modular parts of the software system is facilitated during and after development. Figure 13 illustrates a number of places where design patterns are used. For instance, the ReadGenerator class uses encapsulated algorithms in an instance of the Profile class for applying changes on the instance of the Reference class. These changes occur without ReadGenerator requiring knowledge of the type of change, but the actual type is determined by the underlying relationship bound at runtime when the profiles are loaded. Here the Profile class represents all common attributes and actions required by the two profiles used. Additional relevant details to each type of profile are included in subclasses ErrorProfile and MutationProfile (not shown here). The strategy pattern is used to hide the unnecessary implementation details from outside the class and also to bind the correct type of change at runtime. Furthermore, the Profile class maintains a list of Variant instances used to keep base-wise variation statistics which makes Profile a composite class that loses its meaning without the existence of class Variant. Interestingly, the Variant here is itself a composite.

In addition, as presented in Figure 14, two sub-systems of MutSim which are Simulate and ProfileCreator define *facades* used by MutSim. In general, a *facade* defines the level of abstraction necessary for other classes to depend on or to use the class. These two classes are endpoints to which simulation or profile creation tasks are submitted. Also, the *builder* pattern is used since reads are generated on the fly and in great amounts. This pattern simplifies the construction of reads given any desired coverage and profiles. For instance, the ReadGenerator class only specifies at what coverage reads should be generated without concern for how changes from the instrument error and mutational profiles should be applied in different circumstances (i.e. different error profile, mutation profile, gene length used to model and etc.).

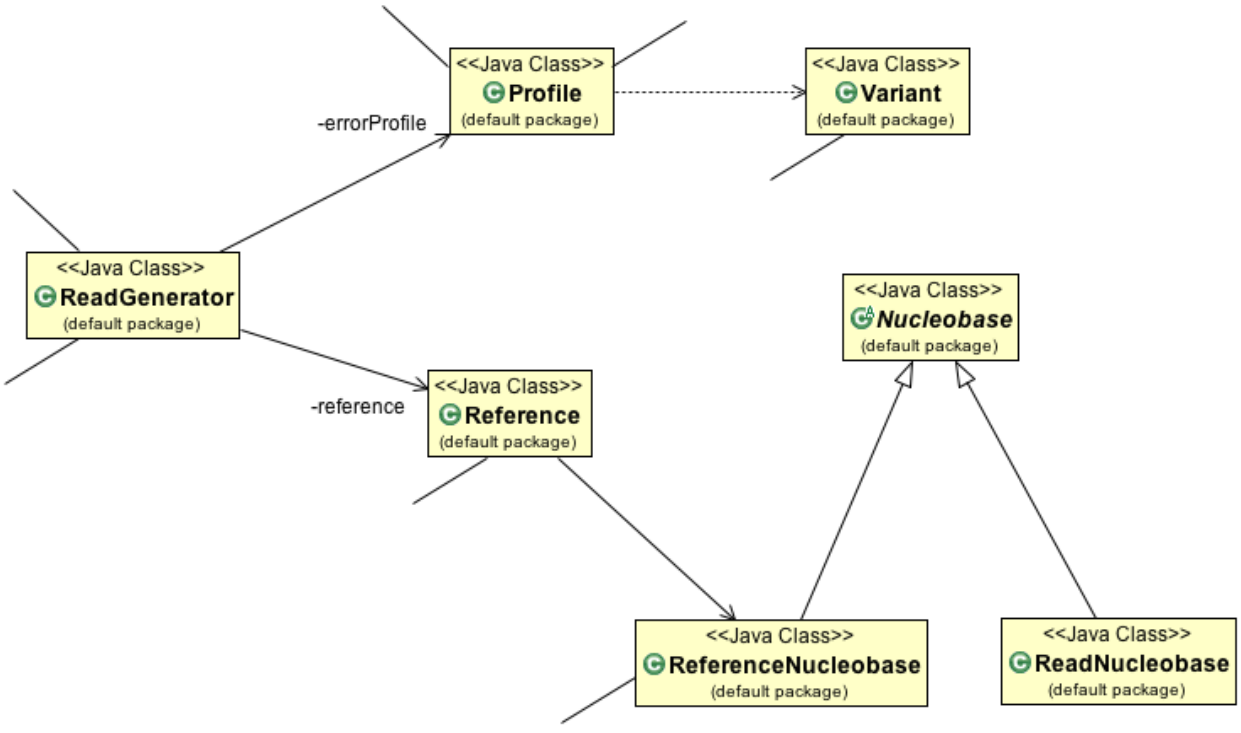

Figure 13: *Strategy* and *Composite* Design Patterns

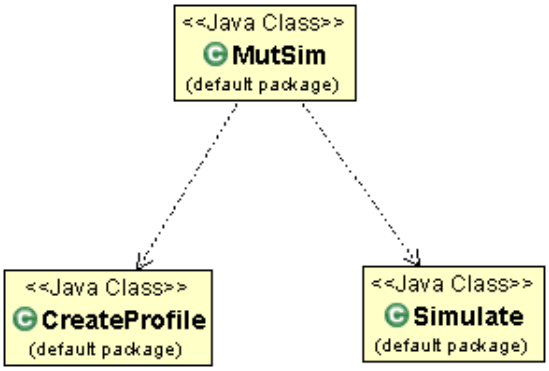

Figure 14: Facade Design Pattern in MutSim

4 Results

This section first describes the two datasets used in this study. Two separate runs of the MutSim simulator are then defined, using each of these datasets respectively. Run I simulates only instrument errors while Run II simulates both instrument errors and mutational variants. The simulated and actual datasets are then compared in terms of GC-content, read depth coverage, Phred quality scores, and read length. Lastly, an analysis of MutSim runtime and memory usage with respect to read depth is provided. The Kolmogorov–Smirnov test is used for statistical comparison and determination of goodness of a fit where applicable.

4.1 Datasets

In this study two datasets are used: one containing only short read sequencing errors and the other containing both sequencing errors and mutational variations. Both datasets come from transgenic rodent assay (TGR) experiments conducted by our collaborators at Health Canada, where the reporter transgene is the bacterial *lacZ* gene. *Escherichia coli* is used as the bacterial host such that *lacZ* genes containing mutations can be detected, retrieved, and sequenced. The datasets are fully described in [28]. Dataset A contains 2 technical duplicates (1 biological sample) from a single control mouse, which was fed olive oil for 28 days. *E. coli* plaques were collected 48 days after the mice were sacrificed and the *lacZ* genes therein were sequenced. Collected plaques were tested against mutations using the P-gal positive selection assay to make sure they do not contain any mutant plaques. Dataset A is therefore considered to be free of mutations and is used to evaluate the instrument error profile portion of MutSim. Dataset B contains 12 technical duplicates (6 biological samples) from 6 mice treated with benzo[a]pyrene

(BaP) blended in oil olive for 28 days. Genomic DNA was screened using the P-gal positive selection assay to identify *lacZ* mutants [28]. The Ion Proton™ NGS system (Life Technologies) was used to sequence all genomic samples using the Ion PIv1™ (Dataset A) and PIv2™ chips (Dataset B).

Two runs of the simulator are used for comparison. Run I results from the simulator being only supplied with instrument error models derived from Dataset A. Run II results from MutSim being provided with both instrument error models and also mutational models (for BaP) derived from Dataset B. In order to have comparable dataset sizes, Runs I and II were set to produce 1 and 6 samples, respectively. Where applicable, either one or both empirical datasets are compared with the results of its corresponding simulation run.

4.2 GC-Content Analysis

In order to determine the validity of simulated data, proportional genotype content was first investigated. In this step of the analysis, GC-content was individually determined for all sequences of the sample, and then samples were combined to form the overall GC-content distribution. Figure 15 illustrates the GC-content distribution over all sequences for Dataset B vs. the respective sequences from simulation Run II. A test for normality confirmed the goodness of fit between sequenced and simulated data (Run I: p-value = 0.3625 and Run II: p-value = 0.5324). Preserving the initial GC-content is the most important aspect of short read simulation as even slight changes may lead to different outcomes downstream in the NGS analysis pipeline. These results indicate that MutSim produces simulated data with valid GC-content. Furthermore, it was observed that the biologically-induced mutations caused by BaP in this study do not introduce any bias in GC contents. However, MutSim is capable of manipulating overall GC-

content through pertinent mutation proportions in the mutational profile if required by the mutagen being simulated.

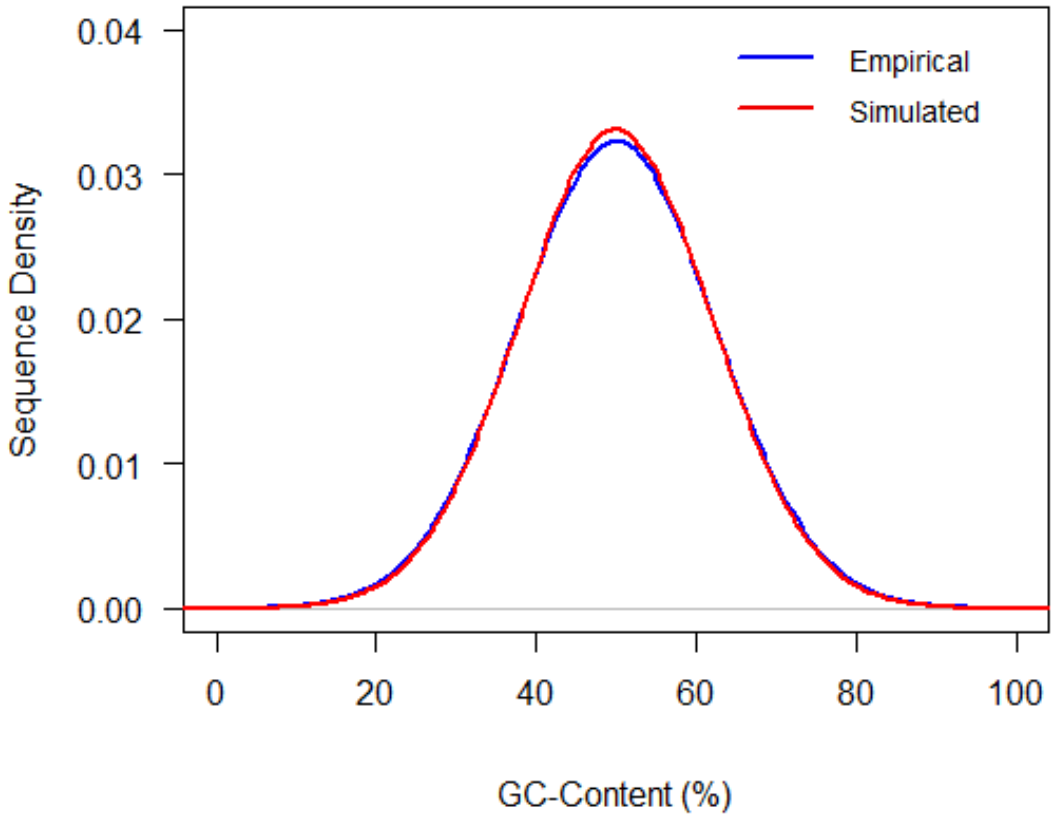

Figure 15: GC-Content % Over All Reads

Empirical Dataset B and simulated data from Run II both contain 6 samples of the *lacZ* gene with a read depth of 88,000x. The results are averaged within each dataset. MutSim successfully preserves GC-Content % regardless of size and number of the simulations. No statistical significance in distributions was observed (p-value = 0.5324)

4.3 Read Depth Coverage Analysis

Read depth coverage refers to the number of times each reference base is read during the sequencing process. Read depth generally follows a uniform distribution with non-zero variance in almost all DNA analyses. As mentioned in the Read Generation section above, MutSim

employs a windowing approach in order to generate short reads. This windowing pattern not only avoids significant memory consumption, but also can be leveraged to introduce fluctuations in read depth coverage by adjusting the *window-width* and *fill-threshold* parameters. Figure 16 represents read depth coverage of simulation Run I and its corresponding NGS Dataset. Run I was performed using a window size of 400bp, fill threshold of 0.15, and read depth coverage of 88,000x. The parameter *window-width* determines the width and number of peaks in the distribution, while *fill-threshold* influences the magnitude of the peaks. These parameters can both be adjusted by extracting the metrics above from sequenced data. However, such adjustment would be only very rarely required since the chip well fill ratio of the Ion Proton™ is typically consistent among different sequencing runs. No systematic behavior in the specific location of peaks was observed among the Datasets used in this study. However, all sequencing runs expressed a consistent variance of 15±5% (in the form of fluctuations) with respect to the coverage which is employed to set *fill-threshold*.

4.4 Phred Quality Score Analysis

MutSim provides explicit support for simulating Phred quality scores which is rarely observed among other simulators [12], [13], [15]. The profile creation module in MutSim captures quality scores in the form of distributions. More specifically, for each genotype and indel type observed at each position, the module calculates and stores the mean and variance of observed quality scores across all reads. These are later used to parameterize a normal distribution of Phred quality score during simulation. This approach more realistically captures the variability in quality scores when compared to other methods which simply assign a fixed constant quality score to all positions [14], [15] or apply an empirically-observed series of

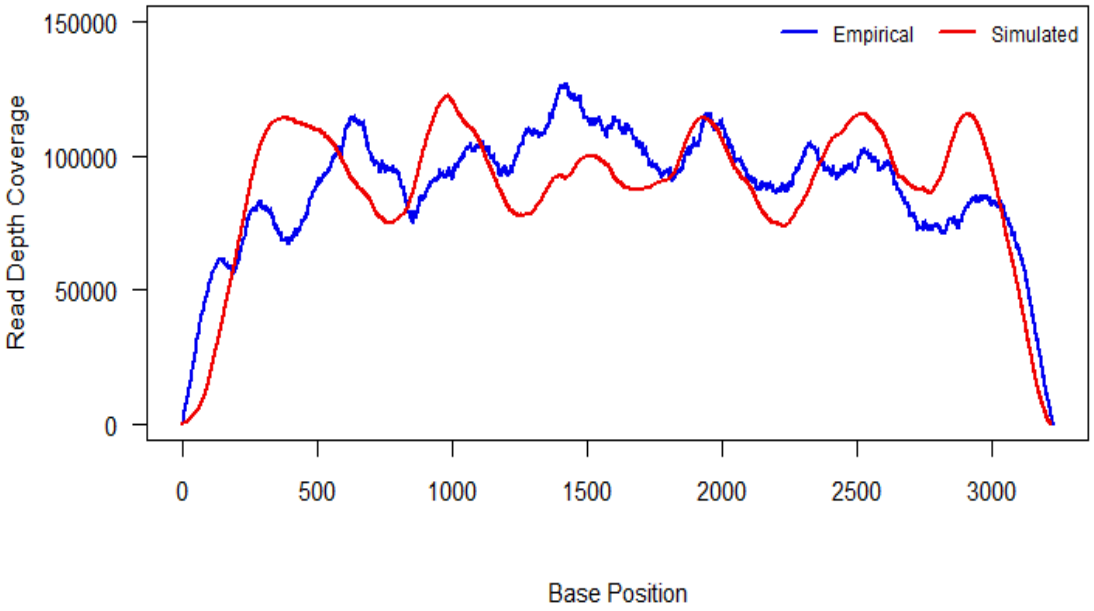

Figure 16: Overall Read Depth Coverage

Represents the overall read depth coverage of both empirical Dataset A and simulation Run I. In both cases, the read depth generally follows a uniform distribution. Fluctuations in read depth for simulated data are quite comparable with those of the empirical data. The locations of fluctuations are not consistent between sequencing runs although the magnitude of fluctuation is roughly consistent.

quality scores [11], [12]. Furthermore, this approach results in performance increase since sampling from a distribution is significantly faster than retrieving stored firm quality scores from empirical data. Figure 17 illustrates the overall quality score distributions of both empirical and simulated data. Although slight modifications are introduced, downstream analysis will produce the same results since low score reads are discarded with respect to their relative Phred score. In addition, the distribution of Phred quality scores in a sequence run should follow a normal distribution otherwise there may have been severe errors included in the data during the sequencing process [24]. A test for normality confirms that no statistical significance is observed between the distribution of Phred quality scores of actual and simulated data (p-value = 0.2043).

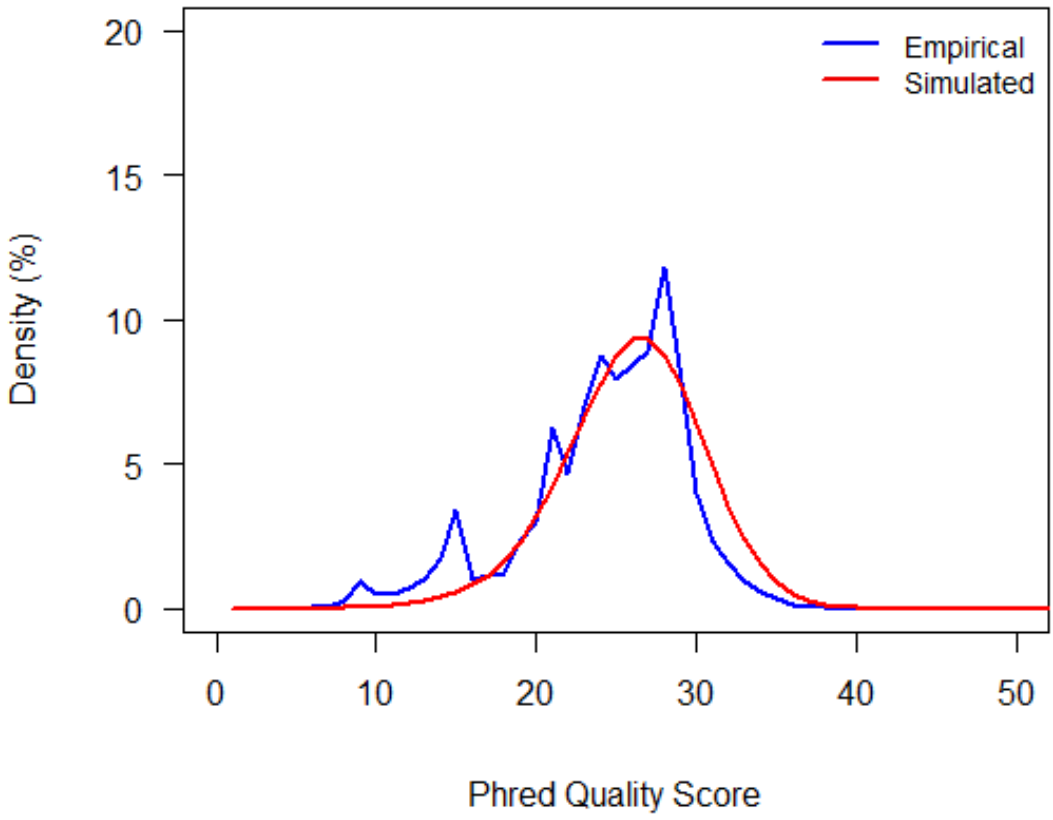

Figure 17: Phred Quality Scores Distributions

Distribution of Phred quality scores of empirical vs. simulated data over all short read sequences in Dataset B and Run II (6 samples each). The density shape follows a normal distribution as a result of capturing the statistical characteristics of the empirical data. This allows considerably faster performance without sacrificing any substantial information as no statistical significance was observed when comparing empirical and simulated Phred quality scores (p-value = 0.2043).

4.5 Read Length Analysis

Instruments using semi-conductor technology, such as the Ion Proton™, produce short reads with variable lengths but have been shown to follow a normal distribution [22], [24]. Both Datasets A and B (gathered using Ion Iv1™ and Ion Iv2™ Chips, respectively) were used to investigate the lengths of short reads. Read lengths differed substantially between the two Datasets. However, they both follow a normal distribution. Therefore, for each chip, a distinct

distribution was fitted (μ=95, σ=30 for Chip1; μ=120, σ=38 for Chip2). In MutSim, the chip configuration can be set by supplying the chip number as an input parameter and additional chip configurations can be added. The valid range of short read well coordinates are also a chip-specific feature of Ion chips and are incorporated as part of the chip component in this simulator (see section 3.2). Given the modular structure of MutSim, new chip settings can be conveniently added without disrupting the overall arrangement of internal constructs.

4.6 Runtime Analysis

To illustrate the linear growth in simulation runtime as a function of the required number of reads to be generated, the simulator was run multiple times, doubling the size of read depth coverage each time. Results are presented in Table 3. The doubling hypothesis is then used to demonstrate linear growth of time with respect to problem size (i.e. read depth coverage). In other words, the log of the ratio of consecutive runtimes approaches 1 as the input size grows, indicating linear growth in runtime with respect to input size. Furthermore, Figure 18 illustrates the steady memory footprint of MutSim over time during simulation with 64,000x coverage using the *LacZ* gene as the reference sequence. As can be seen, the memory requirements remain constant over time during a simulation, owing to careful memory management within MutSim. Stable and deterministic memory requirements ensure scalable performance of the simulator, such that any arbitrary read depths may be simulated using this approach. Each of the state-of-the-art NGS simulators reviewed in section 2.4 were evaluated for their ability to simulate read depths typically used in MSA experiments. None of them were able to simulate depths greater than 1000.

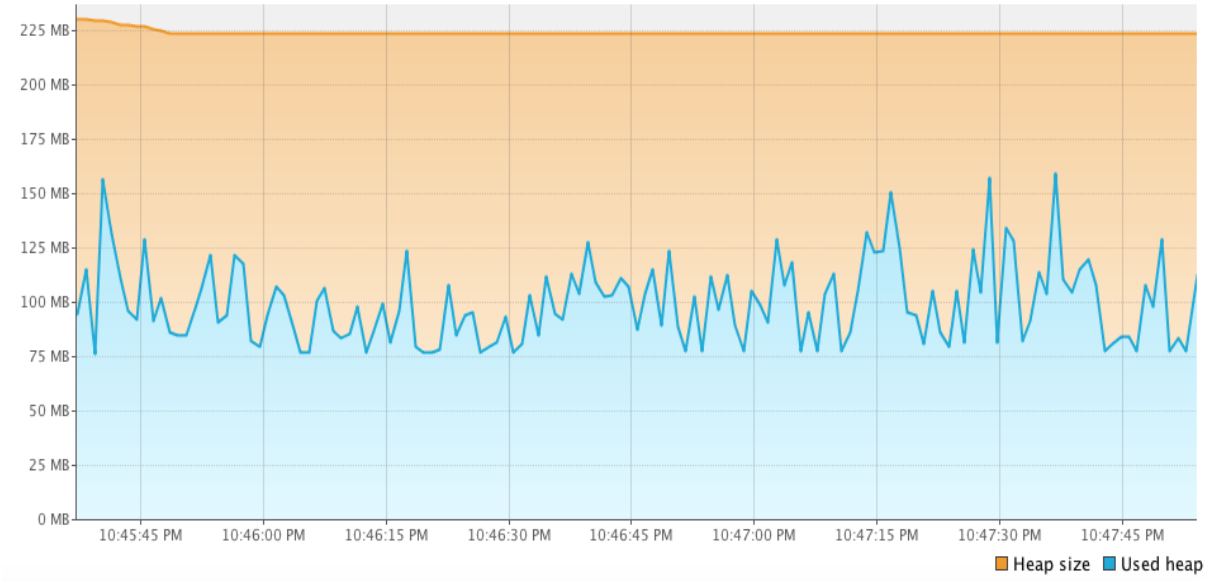

Figure 18: Memory Footprint of MutSim

Memory usage of MutSim with respect to time during a simulation with 64,000x coverage when supplying both instrument error and mutational profiles. Careful memory management guarantees that memory requirements do not grow with time or problem size. Heap size is proportionally and automatically allocated and released by the JVM.

Read Depth Coverage	Simulation Time (sec)	Ratio	log2(ratio)
1000	9		
2000	15	1.6667	0.7370
4000	19	1.2667	0.3410
8000	35	1.8421	0.8814
16000	68	1.9429	0.9582
32000	134	1.9706	0.9786
64000	265	1.9776	0.9838
128000	523	1.9735	0.9808

Table 3: Runtime Summary of MutSim

Demonstrates runtime summary of MuSim when both instrument error and mutational profiles are supplied and LacZ gene is used as the reference sequence. Results indicate linear growth of simulation runtime with respect to input size. In other words, $\boldsymbol{b = log_2(ratio)}$ approaches 1 as data size increases, where $\boldsymbol{b}$ represents system-independent growth of the algorithm.

5 Thesis Summary and Future Recommendations

5.1 Summary of contributions

In this study we present MutSim, an NGS data simulator that concisely models the Life Technology Ion Proton™ sequencing errors and mutational variations induced by a mutagen which is required for simulation of NGS short read data for MSA experiments. Fast and accurate simulation is achieved by employing modern software design principles, practical algorithms, and industrial-strength data structures. Scalable standards and algorithms enable this simulator to produce the high read depth coverage required by MSA studies in the area of short read simulation. Furthermore, this simulator introduces new notions such as software maintainability and scalability in this domain which distinguishes MutSim among short read simulators. The process of creating an error profile in MutSim provides flexibility since it begins by accepting a pile-up file. Such a pile-up file can be generated by calling variants in several fastq files and combining the results. Moreover, the mutational profile models both composition-based and position-based probabilities for substitutions and indels, permitting accurate modeling of both independent mutations and clonal expansion.

NGS technologies continue to advance at a high rate, with new capabilities being continuously added to DNA sequencers. NGS data analysis pipelines likewise continue to evolve and increase in complexity, requiring additional data to fully parameterize and optimize the experiment design and analysis stages. MutSim is designed not only to accurately and efficiently simulate data for existing technologies, but also to facilitate adding new components in future. Modularity in MutSim's structure allows rapid design and development of new modules with

maximizing the usage of currently implemented components. Lastly, the algorithms introduced in this simulator are scalable as data size increases.

5.2 *Discussion*

MutSim introduces scalability of short read simulation with respect to read depth coverage. Existing NGS simulators were designed and developed without this requirement in mind. A runtime experiment with x1,000 coverage was attempted on all simulators discussed in section 4.6. However, given the fact that every package failed due to memory requirements, a direct comparison of runtime was not possible. In contrast, since MutSim introduces a highly scalable approach with careful memory management, no limit is expected on supplying read coverage. In fact, only the simulation time would increase as more data is to be generated.

Furthermore, no simulator was found to model the effects of a mutagen beyond the ability to supply an overall random mutation rate, ignoring both composition and position dependent aspects of mutagen tendencies. This abstraction of simulating mutations leads to unsatisfactory mutagen modeling scheme which is the most important aspect of short read generation for MSA studies. The effects of mutations on short reads are fairly complex. Current MSA studies, in order to form a comparison structure and evaluate the results of an assay, include both compositional and positional mutation effects. MutSim is the only NGS simulator that models both of these aspects and, therefore, as with runtime, no sensible comparison was possible to relate the ability of different simulators in applying biologically-induced mutations on short reads, given the fact that they provide no support for modeling mutations, nor can they produce read depth coverage approaching that required for MSA studies.

MutSim was designed in order to address the requirements posed by NGS (using the Ion Proton™ instrument) for conducting MSA experiments. In this work, all of the challenges outlined in the Problem Statement in section 1.3 have been addressed. However, as more sophisticated analysis may appear over the next years, additional modules may be introduced to meet the new requirements. As mentioned, in the design of MutSim, modern software design principals were employed to facilitate further expansion of this work as required. MutSim has been released as an open source project at https://github.com/jrgreen7/MutSim to encourage use and extension by other research groups.

This work has a number of limitations. Firstly, MutSim has been developed and validated over a relatively small number of experimental data (one instrument, from one lab). MutSim should be further validated using additional data, potentially from other labs and using other mutagens. Secondly, MutSim is currently specifically targeted at the Ion Proton™ instrument. However, it has been designed to easily generalize to other instruments.

5.3 Future Work

The mutational differences observed between samples in MSA assays are in fact quite complicated. Even though the variations among samples are not typically reflected in most MSA studies due to their insignificance in overall results, such information could be useful to adjust the simulator read generation mechanism when multiple samples are of interest. We have not observed a consistent and deterministic difference between samples in MSA studies, even though the differences can be identified conveniently. MutSim, emulates this variation between samples through the natural stochastic behaviors introduced by sampling numbers from statistical distributions for sequencing errors. The investigation of mutational samples and their differences

is beyond the scope of this study given their non-trivial characteristics. A few notable suggested methods of comparison among mutation spectra include multinomial models, regression analysis, and pairwise comparison [45], [46]. Accurate analysis can be performed by selecting an appropriate comparison model in order to identify the differences. Nevertheless, the important aspect is that the in-between sample differences are highly dependent on the environmental conditions (e.g. animal and tissue type) of each study, hindering any generalizations to be drawn.

MutSim is designed to model error profile of Ion Proton instrument due to the lack of support by other simulators for the technology used in this sequencer. As an extension of this work, error profiles of other instruments may be modeled and used within MutSim. Given the proper design of MutSim, no disruption to other components is expected in the case of substituting the error profile module.

Lastly, parameter optimization and selection of proper analysis tools are important aspects of constructing pipelines for DNA data analysis. MutSim enables researchers to investigate the performance of MSA pipelines more conveniently and at a considerably less cost as it permits the generation of synthetic data avoiding the need for repetitive re-sequencing in order to tune pipeline tools and their parameters. In other words, as no simulators support the requirements for such data simulation, by the work presented in this study, it is now possible to leverage data simulation for optimization and parameter adjustment of MSA pipelines.

Appendix A: Profile Descriptions

This appendix describes both the error profile and the mutation profile used in MutSim in detail. Profiles are kept in separate files following different formats as each of them is suited for convenient data extraction from related various resources.

1.1 Error Profile

The error profile is designed to capture important characteristics of empirical data. This profile extracts base-wise information from pile-up files. A sample error profile for the *LacZ* gene is presented in Figure 19. For each position on the reference sequence, genotype compositions along with all indel types are stored. Compositions are calculated using proportions instead of raw frequencies in order to provide more flexibility and to enable the use of the same profile for simulation runs with different read depth coverage. Furthermore, Phred quality scores are represented by estimating statistical distributions rather than storing actual empirically observed scores across all reads for each consensus in each position. This schema resulted in a concise and accurate modeling of errors as discussed in section 4.2 and 4.4.

lacz	2226	G	+CGT	1	0.00001	13.00000	0
lacz	2226	G	+G	4	0.00005	25.75000	0
lacz	2226	G	+GA	1	0.00001	30.00000	0
lacz	2226	G	+T	20	0.00023	26.70000	7.11000
lacz	2226	G	+TA	2	0.00002	26.50000	0
lacz	2226	G	-A	396	0.00446	25.79040	11.47880
lacz	2227	A	A	87223	0.99565	23.86432	24.30797
lacz	2227	A	C	24	0.00027	24.20833	19.91493
lacz	2227	A	G	290	0.00331	23.30000	28.20310
lacz	2227	A	T	26	0.00030	22.88462	26.40976
lacz	2227	A	+A	1	0.00001	28.00000	0
lacz	2227	A	+AAC	1	0.00001	27.00000	0
lacz	2227	A	+AC	3	0.00003	27.00000	0
lacz	2227	A	+ACG	1	0.00001	24.00000	0
lacz	2227	A	+ACGT	1	0.00001	23.00000	0
lacz	2227	A	+G	1	0.00001	22.00000	0
lacz	2227	A	-AG	22	0.00025	24.27273	23.19835
lacz	2227	A	-AGC	8	0.00009	24.62500	8.48438
lacz	2227	A	-AGCC	3	0.00003	23.33333	0
lacz	2228	A	A	87249	0.98736	22.96827	18.86222
lacz	2228	A	C	59	0.00067	21.08475	32.48434
lacz	2228	A	G	308	0.00349	22.66883	19.01370
lacz	2228	A	T	15	0.00017	22.13333	15.71556
lacz	2228	A	+AAG	1	0.00001	24.00000	0
lacz	2228	A	+AC	2	0.00002	23.00000	0
lacz	2228	A	+ACG	1	0.00001	23.00000	0
lacz	2228	A	+AG	5	0.00006	21.40000	10.64000
lacz	2228	A	+ATG	1	0.00001	24.00000	0
lacz	2228	A	+C	271	0.00307	23.10701	21.01807
lacz	2228	A	+CG	17	0.00019	23.35294	17.64014
lacz	2228	A	+CGCC	1	0.00001	24.00000	0
lacz	2228	A	+CT	1	0.00001	30.00000	0
lacz	2228	A	+G	412	0.00466	22.71359	15.40341
lacz	2228	A	+GC	1	0.00001	21.00000	0

Figure 19: Sample Error Profile of Ion Proton

The partial instrument error profile: follows the format of gene name, base number, reference base, observed consensus base/indel, observed frequency of occurrence, probability of occurrence, Phred quality score mean, and Phred quality score variance. Note, Phred quality score variance is only calculated if sufficient observations of the same consensus are presented. For the consensus column, entries that begin with a '-' indicate deletions while '+' indicate insertions. Substitutions are represented with a single nucleotide.

1.2 Mutation Profile

The mutation profile used in MutSim is design for efficiency of data access, complete representation of mutagen fingerprints as typically presented in MSA studies, and convenience of data import to follow this format. As described in section 2.3, mutation spectra are typically described by their global mutation (base pair substitutions and indels) proportions as well as position-specific frequency data. Table 4 illustrates a sample mutational profile imported from a MSA assay investigating the effects of the *benzo[a]pyrene* mutagen on the *LacZ* gene.

C	A	0.22580645
G	A	0.05241936
C	T	0.05241936
A	T	0.00403226
T	A	0.00403226
A	G	0.00403226
T	C	0.00403226
A	C	0
T	G	0
+		
I	1	0.01612903
I	2	0
D	1	0.16705069
D	2	0.06682028
+		
67	G	0.002
136	G	0.002
163	C	0.002
187	G	0.002
236	G	0.002
246	C	0.002
281	C	0.002
416	G	0.002

Table 4: Sample Mutation Profile used in MutSim

Sections are separated by + symbol. Section 1 describes global base pair substitution rates (column 1 is reference base, column 2 is variant base). Section 2 describes indel mutation proportions where column 1 indicates I for insertion and D for deletion. Column 2 indicates the length of the indel and column 3 indicates the relative frequency (frequencies in sections 1&2 are normalized together). Sections three and four are positional probabilities of mutation occurrences of base pair substitutions and indels if non-zero. Column 1 indicates the position, 2 indicates the variant base, and 3 indicates the probability of occurrence (normalized across all observed variants at all positions).

References

[1] J. Shendure and H. Ji, "Next-generation DNA sequencing.," *Nat. Biotechnol.*, vol. 26, no. 10, pp. 1135–1145, 2008.

[2] E. R. Mardis, "The impact of next-generation sequencing technology on genetics," *Trends in Genetics*, vol. 24, no. 3. pp. 133–141, 2008.

[3] E. R. Mardis, "Next-generation sequencing platforms.," *Annu. Rev. Anal. Chem. (Palo Alto. Calif).*, vol. 6, pp. 287–303, Jan. 2013.

[4] A. S. Gargis, L. Kalman, M. W. Berry, D. P. Bick, D. P. Dimmock, T. Hambuch, F. Lu, E. Lyon, K. V Voelkerding, B. A. Zehnbauer, R. Agarwala, S. F. Bennett, B. Chen, E. L. H. Chin, J. G. Compton, S. Das, D. H. Farkas, M. J. Ferber, B. H. Funke, M. R. Furtado, L. M. Ganova-Raeva, U. Geigenmüller, S. J. Gunselman, M. R. Hegde, P. L. F. Johnson, A. Kasarskis, S. Kulkarni, T. Lenk, C. S. J. Liu, M. Manion, T. A. Manolio, E. R. Mardis, J. D. Merker, M. S. Rajeevan, M. G. Reese, H. L. Rehm, B. B. Simen, J. M. Yeakley, J. M. Zook, and I. M. Lubin, "Assuring the quality of next-generation sequencing in clinical laboratory practice.," *Nat. Biotechnol.*, vol. 30, no. 11, pp. 1033–6, Nov. 2012.

[5] F. Sanchez, E. Salami, A. Ramirez, and M. Valero, "Performance Analysis of Sequence Alignment Applications," in *2006 IEEE International Symposium on Workload Characterization*, 2006, pp. 51–60.

[6] T. Ono, H. Ikehata, S. Nakamura, Y. Saito, Y. Hosoi, Y. Takai, S. Yamada, J. Onodera, and K. Yamamoto, "Age-associated increase of spontaneous mutant frequency and molecular nature of mutation in newborn and old lacZ-transgenic mouse.," *Mutat. Res.*, vol. 447, no. 2, pp. 165–77, Feb. 2000.

[7] A. Besaratinia, H. Li, J.-I. Yoon, A. Zheng, H. Gao, and S. Tommasi, "A high-throughput next-generation sequencing-based method for detecting the mutational fingerprint of carcinogens.," *Nucleic Acids Res.*, vol. 8111, no. 10, pp. 1–13, 2012.

[8] N. Tokuriki and D. S. Tawfik, "Stability effects of mutations and protein evolvability.," *Curr. Opin. Struct. Biol.*, vol. 19, no. 5, pp. 596–604, Oct. 2009.

[9] S. D. Bruner, D. P. Norman, and G. L. Verdine, "Structural basis for recognition and repair of the endogenous mutagen 8-oxoguanine in DNA.," *Nature*, vol. 403, no. 6772, pp. 859–66, Feb. 2000.

[10] I. B. Rogozin, V. N. Babenko, L. Milanesi, and Y. I. Pavlov, "Computational analysis of mutation spectra.," *Brief. Bioinform.*, vol. 4, no. 3, pp. 210–227, 2003.

[11] W. Huang, L. Li, J. R. Myers, and G. T. Marth, "ART: a next-generation sequencing read simulator.," *Bioinformatics*, vol. 28, no. 4, pp. 593–4, Feb. 2012.

[12] X. Hu, J. Yuan, Y. Shi, J. Lu, B. Liu, Z. Li, Y. Chen, D. Mu, H. Zhang, N. Li, Z. Yue, F. Bai, H. Li, and W. Fan, "pIRS: Profile-based Illumina pair-end reads simulator.," *Bioinformatics*, vol. 28, no. 11, pp. 1533–5, Jun. 2012.

[13] D. Pratas, A. J. Pinho, and J. M. O. S. Rodrigues, "XS: a FASTQ read simulator.," *BMC Res. Notes*, vol. 7, p. 40, Jan. 2014.

[14] "wgsim." [Online]. Available: https://github.com/lh3/wgsim.

[15] M. Frampton and R. Houlston, "Generation of artificial FASTQ files to evaluate the performance of next-generation sequencing pipelines.," *PLoS One*, vol. 7, no. 11, p. e49110, Jan. 2012.

[16] F. Sanger, S. Nicklen, and A. R. Coulson, "DNA sequencing with chain-terminating inhibitors.," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 74, no. 12, pp. 5463–7, Dec. 1977.

[17] M. P. Sawicki, G. Samara, M. Hurwitz, and E. Passaro, "Human Genome Project," *Am. J. Surg.*, vol. 165, no. 2, pp. 258–264, Feb. 1993.

[18] F. S. Collins, M. Morgan, and A. Patrinos, "The Human Genome Project: lessons from large-scale biology.," *Science*, vol. 300, no. 5617, pp. 286–90, Apr. 2003.

[19] F. S. Collins, "Implications of the Human Genome Project for Medical Science," *JAMA*, vol. 285, no. 5, p. 540, Feb. 2001.

[20] S. Brenner, M. Johnson, J. Bridgham, G. Golda, D. H. Lloyd, D. Johnson, S. Luo, S. McCurdy, M. Foy, M. Ewan, R. Roth, D. George, S. Eletr, G. Albrecht, E. Vermaas, S. R. Williams, K. Moon, T. Burcham, M. Pallas, R. B. DuBridge, J. Kirchner, K. Fearon, J. Mao, and K. Corcoran, "Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays.," *Nat. Biotechnol.*, vol. 18, no. 6, pp. 630–4, Jun. 2000.

[21] R. Williams, S. G. Peisajovich, O. J. Miller, S. Magdassi, D. S. Tawfik, and A. D. Griffiths, "Amplification of complex gene libraries by emulsion PCR.," *Nat. Methods*, vol. 3, no. 7, pp. 545–50, Jul. 2006.

[22] M. A. Quail, M. Smith, P. Coupland, T. D. Otto, S. R. Harris, T. R. Connor, A. Bertoni, H. P. Swerdlow, and Y. Gu, "A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers.," *BMC Genomics*, vol. 13, no. 1, p. 341, Jan. 2012.

[23] J. M. Rothberg, W. Hinz, T. M. Rearick, J. Schultz, W. Mileski, M. Davey, J. H. Leamon, K. Johnson, M. J. Milgrew, M. Edwards, J. Hoon, J. F. Simons, D. Marran, J. W. Myers,

J. F. Davidson, A. Branting, J. R. Nobile, B. P. Puc, D. Light, T. A. Clark, M. Huber, J. T. Branciforte, I. B. Stoner, S. E. Cawley, M. Lyons, Y. Fu, N. Homer, M. Sedova, X. Miao, B. Reed, J. Sabina, E. Feierstein, M. Schorn, M. Alanjary, E. Dimalanta, D. Dressman, R. Kasinskas, T. Sokolsky, J. A. Fidanza, E. Namsaraev, K. J. McKernan, A. Williams, G. T. Roth, and J. Bustillo, "An integrated semiconductor device enabling non-optical genome sequencing.," *Nature*, vol. 475, no. 7356, pp. 348–52, Jul. 2011.

[24] L. M. Bragg, G. Stone, M. K. Butler, P. Hugenholtz, and G. W. Tyson, "Shining a light on dark sequencing: characterising errors in Ion Torrent PGM data.," *PLoS Comput. Biol.*, vol. 9, no. 4, p. e1003031, Apr. 2013.

[25] H. Li and R. Durbin, "Fast and accurate short read alignment with Burrows-Wheeler transform.," *Bioinformatics*, vol. 25, no. 14, pp. 1754–60, Jul. 2009.

[26] B. Langmead, C. Trapnell, M. Pop, and S. L. Salzberg, "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome.," *Genome Biol.*, vol. 10, no. 3, p. R25, Jan. 2009.

[27] R. Luo, T. Wong, J. Zhu, C.-M. Liu, X. Zhu, E. Wu, L.-K. Lee, H. Lin, W. Zhu, D. W. Cheung, H.-F. Ting, S.-M. Yiu, S. Peng, C. Yu, Y. Li, R. Li, and T.-W. Lam, "SOAP3-dp: fast, accurate and sensitive GPU-based short read aligner.," *PLoS One*, vol. 8, no. 5, p. e65632, 2013.

[28] F. M. and C. L. Y. Marc A Beal, Rémi Gagné, "Characterizing Benzo[a]pyrene-induced lacZ Mutation Spectrum in Transgenic Mice Using Next-Generation Sequencing," *Mutat. Res.*, 2015.

[29] S. Ossowski, K. Schneeberger, J. I. Lucas-Lledó, N. Warthmann, R. M. Clark, R. G. Shaw, D. Weigel, and M. Lynch, "The rate and molecular spectrum of spontaneous mutations in Arabidopsis thaliana.," *Science*, vol. 327, no. 5961, pp. 92–4, Jan. 2010.

[30] E. J. Mientjes, M. J. Steenwinkel, J. H. van Delft, P. H. Lohman, and R. A. Baan, "Comparison of the X-gal- and P-gal-based systems for screening of mutant lambda lacZ phages originating from the transgenic mouse strain 40.6.," *Mutat. Res.*, vol. 360, no. 2, pp. 101–6, Jun. 1996.

[31] M. A. Ihle, J. Fassunke, K. König, I. Grünewald, M. Schlaak, N. Kreuzberg, L. Tietze, H.-U. Schildhaus, R. Büttner, and S. Merkelbach-Bruse, "Comparison of high resolution melting analysis, pyrosequencing, next generation sequencing and immunohistochemistry to conventional Sanger sequencing for the detection of p.V600E and non-p.V600E BRAF mutations.," *BMC Cancer*, vol. 14, no. 1, p. 13, Jan. 2014.

[32] M. H. Brugman, J. D. Suerth, M. Rothe, S. Suerbaum, A. Schambach, U. Modlich, O. Kustikova, and C. Baum, "Evaluating a ligation-mediated PCR and pyrosequencing method for the detection of clonal contribution in polyclonal retrovirally transduced samples.," *Hum. Gene Ther. Methods*, vol. 24, no. 2, pp. 68–79, Apr. 2013.

[33] J. G. Boer de, S. Provost, N. Gorelick, K. Tindall, and B. W. Glickman, “Spontaneous mutation in lacI transgenic mice: a comparison of tissues,” *Mutagenesis*, vol. 13, no. 2, pp. 109–114, Mar. 1998.

[34] S. Zhang, B. W. Glickman, and J. G. de Boer, “Spontaneous mutation of the lacI transgene in rodents: absence of species, strain, and insertion-site influence.,” *Environ. Mol. Mutagen.*, vol. 37, no. 2, pp. 141–6, Jan. 2001.

[35] J. A. Heddle, “On clonal expansion and its effects on mutant frequencies, mutation spectra and statistics for somatic mutations in vivo,” *Mutagenesis*, vol. 14, no. 3, pp. 257–260, May 1999.

[36] P. D. Lewis and J. M. Parry, “An exploratory analysis of multiple mutation spectra.,” *Mutat. Res.*, vol. 518, no. 2, pp. 163–180, 2002.

[37] V. Thybaud, S. Dean, T. Nohmi, J. de Boer, G. R. Douglas, B. W. Glickman, N. J. Gorelick, J. A. Heddle, R. H. Heflich, I. Lambert, H.-J. Martus, J. C. Mirsalis, T. Suzuki, and N. Yajima, “In vivo transgenic mutation assays.,” *Mutat. Res.*, vol. 540, no. 2, pp. 141–51, Oct. 2003.

[38] I. B. Lambert, T. M. Singer, S. E. Boucher, and G. R. Douglas, “Detailed review of transgenic rodent mutation assays.,” *Mutat. Res.*, vol. 590, no. 1–3, pp. 1–280, 2005.

[39] P. J. A. Cock, C. J. Fields, N. Goto, M. L. Heuer, and P. M. Rice, “The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants.,” *Nucleic Acids Res.*, vol. 38, no. 6, pp. 1767–71, Apr. 2010.

[40] B. Langmead and S. L. Salzberg, “Fast gapped-read alignment with Bowtie 2,” *Nature Methods*, vol. 9, no. 4. pp. 357–359, 2012.

[41] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin, “The Sequence Alignment/Map format and SAMtools.,” *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, 2009.

[42] G. R. Douglas, J. D. Gingerich, J. A. Gossen, and S. A. Barlett, “Sequenc spectra of spontaneous lacZ gene mutations in trransgenic mouse somatic and germline tissues,” *Mutagenesis*, vol. 9, no. 5, pp. 451–458, Sep. 1994.

[43] M. J. Ligtenberg, S. Kemp, C. O. Sarde, B. M. van Geel, W. J. Kleijer, P. G. Barth, J. L. Mandel, B. A. van Oost, and P. A. Bolhuis, “Spectrum of mutations in the gene encoding the adrenoleukodystrophy protein.,” *Am. J. Hum. Genet.*, vol. 56, no. 1, pp. 44–50, Jan. 1995.

[44] D. Knuth, *Art of Computer Programming, Volume 1: Fundamental Algorithms*. Addison-Wesley Professional, 1997.

[45] W. W. Piegorsch and K. A. Richwine, “Large-sample pairwise comparisons among multinomial proportions with an application to analysis of mutant spectra,” *J. Agric. Biol. Environ. Stat.*, vol. 6, no. 3, pp. 305–325, Sep. 2001.

[46] I. B. Rogozin and Y. I. Pavlov, “Theoretical analysis of mutation hotspots and their DNA sequence context specificity.,” *Mutat. Res.*, vol. 544, no. 1, pp. 65–85, Sep. 2003.

[47] “Double stranded DNA.” [Online]. Available: http://www.ucl.ac.uk/~sjjgsca/DNAreplication.html. [Accessed: 13-Jan-2015].

[48] “Gene Expression.” [Online]. Available: http://sciencesoup.tumblr.com/post/92776449818/translation-in-the-process-of-transcription-an. [Accessed: 13-Jan-2015].

[49] “Genetic Code Table.” [Online]. Available: https://biologywarakwarak.wordpress.com/2012/01/15/the-3-magical-rules-to-determine-the-amino-acid-chain-from-a-dna-piece-without-error/. [Accessed: 13-Jan-2015].

[50] “Next Generation Sequencing Process.” [Online]. Available: http://www.thermoscientificbio.com/ngs-library-preparation-kits/. [Accessed: 13-Jan-2015].

[51] “Institute of Translational Health Sciences.” [Online]. Available: https://www.iths.org/. [Accessed: 04-Jan-2015].

[52] C.-Y. Lee, Y.-C. Chiu, L.-B. Wang, Y.-L. Kuo, E. Y. Chuang, L.-C. Lai, and M.-H. Tsai, “Common applications of next-generation sequencing technologies in genomic research,” *Translational Cancer Research*, vol. 2, no. 1. pp. 33–45, 03-May-2013.

[53] M. Meyerson, S. Gabriel, and G. Getz, “Advances in understanding cancer genomes through second-generation sequencing.,” *Nat. Rev. Genet.*, vol. 11, no. 10, pp. 685–96, Oct. 2010.