

# Analysis of the Impact of Ploidy on the Genotypic Effects of Directed Evolution

---

*By*  
*Phillip Richmond*  
*MCDB, University of Colorado—Boulder*

*Defense Date: April 11<sup>th</sup>, 2012*

*Thesis Advisor:*  
*Robin Dowell*

*Thesis Defense Committee:*  
*Robin Dowell, MCDB-CSCI-Biofrontiers*  
*Debra Goldberg, CSCI*  
*Christy Fillman, MCDB*

## Abstract

The role of ploidy—copy number of chromosomes in an organism—on cells under environmental stress is poorly understood. Variations in ploidy such as aneuploidy—deviations in copy number of chromosomes compared to an organisms base ploidy—and polyploidy—more than two sets of each chromosome—are common in cancer so it is important to understand how ploidy impacts evolution. We performed an evolution experiment on strains of *Saccharomyces cerevisiae* with different ploidy (haploid, diploid, and tetraploid) under selection for growth on a low-glucose, high raffinose media. After 240 generations, the parental strain and seven of the surviving strains (2 diploid and 5 tetraploids) were sequenced via next generation sequencing technology. We analyzed the sequenced strains to identify mutations causal for survival and compared the mutations between the strains of different ploidy. More mutations in the tetraploid strains relative to their diploid counterparts were discovered hinting at the possibility of a positive correlation between ploidy and mutation rate. However, the inability to identify structural variations from the current sequencing dataset prevents complete identification of mutations present in the evolved strains. In addition, we have benchmarked variation-calling software with both “*in silico*” generated synthetic datasets, as well as our evolved strains for calling SNPs and indels at higher ploidy ( $N > 2$ ).

## Introduction

### Biological Relevance

#### Polyploidy and Aneuploidy

Polyploidy, or having more than two sets of homologous chromosomes, occurs frequently in nature. Polyploidy can be the result of genome duplications and can have implications in evolutionary divergence and emergence of species [16]. One example of a species undergoing genome duplication is the *Xenopus laevis*, which is stable in its tetraploid form—four copies of each chromosome (4N), while its close relative *Xenopus tropicalis* maintains a stable diploid genome (2N) [17]. Other examples of polyploidy exist in solid tumor masses, where the cells exhibit increased as well as the added confounding factor of aneuploidy—abnormal copy number of chromosomes relative to the organisms’ accepted euploidy (base ploidy) [1,2,3]. Recently, numerous studies have taken a closer look at aneuploidy and its role in cancer; a connection that was first identified in the early 1900s when Theodor Boveri found rampant aneuploidy via multipolar mitosis in several different tumors [2]. Since cancer is a genetic disease in which cells gain mutations that allow them to out-compete their neighboring cells and attain a more rapid rate of cell division and proliferation, it can be modeled in an evolution study [3]. In an evolution study, cells can be placed under stress, and will be forced to acquire mutations that allow them to have increased fitness and to more rapidly divide and proliferate (similar to cancer) [3].

#### Experimental Setup

Evolution studies have been common over the course of the past 20 years in many different model organisms. One such model organism is *Saccharomyces cerevisiae*—budding yeast—which is used for its efficiency at modeling the single cell level of higher eukaryotes. In this project, a haploid *Saccharomyces cerevisiae* strain (BY4741) was utilized to generate an essentially isogenic (identical

copies of each chromosome) ploidy series with haploid, diploid, and tetraploid progeny. The diploid strains were generated using classical yeast mating techniques. In order to produce the tetraploid strains, the diploids were manipulated at the mating type locus—MAT-a/MAT- $\alpha$  heterozygous in a diploid—mutating it to be a homozygous of either MAT-a or MAT- $\alpha$ ; thus confusing the diploid strain into mating with itself, creating identical copies of the chromosomes, but not completing mitosis due to the mating type homozygosity. The end result being that even though there are a higher copy number of chromosomes in the diploids and tetraploids, each copy is identical to each other (isogenic). Then, in an *in vitro* experiment that doesn't allow for mating or meiosis, haploid, diploid, and tetraploid strains are passaged for 240 generations under the selective pressure of a low-glucose/high-raffinose media—which is not a good quality carbon source for the wild type yeast. Using controlled experimental procedures not detailed here, the strains competed with each other under this selective pressure, and at the end of the 240 generations surviving strains showed an increase in fitness for survival and growth on the low-glucose/high raffinose media. From the resultant strains, some were subjected to genomic analysis to evaluate how they evolved. All of the experiments described above were performed by our collaborators Anna Selmecki and David Pellman at the Dana Farber Cancer Institute.

### Strain characterizations

Strains were chosen from the evolution experiment after comparative genomic hybridization assays (aCGH) and flow cytometry analyses were performed. These assays were performed on all of the evolved progeny, and allowed for the identification of strains that contained particular traits that could be causal for survival in the evolution experiment, as well as strains that lacked these traits and may have evolved through an alternative mechanism. The results of these individual analyses can be summed up in the following tables, including HXT 6/7 amplification, certain aneuploidy characteristics or lack thereof, and base of ploidy. The selection of these strains from the larger pool of evolved progeny as well as the assays was performed by our collaborators at the Dana Farber Cancer Institute. Through the sequencing analysis, these characteristics will all be reconfirmed (with the exception of base level of ploidy). In addition to the progeny strains described below, the parental strain, 6040, was sequenced in its tetraploid form.

Strain	Ploidy	Aneuploidies	HXT6/7 amplification
G2	4N	13+, 14-, Seg4-	YES
D9	3N+	3+, 13+	
F2	4N		YES
A8	4N+	2x12+, 13+	YES
G11	3N+	9+, 13+, 14+	
B2	2N		YES
F12	2N		

**Table 1:** This table (generated by our collaborators Anna Selmecki and David Pellman at the Dana Farber Cancer Institute) represents the evolved progeny to be sequenced, and their ploidy as identified by flow

cytometry, and specific aneusomies and copy number variations at the HXT6/7 locus as identified by aCGH data.

A common characteristic of the strains that started at a higher ploidy, is that the vast majority of them exhibited some form of aneuploidy at the end of the evolution experiment, probably due to the instability of polyploidy in the *Saccharomyces cerevisiae* system.

## Sequencing

### *Next Generation Sequencing*

Next generation sequencing has been giving genetic researchers insights into the complexities of the genomes (DNA) and transcriptomes (RNA) of numerous organisms over the past 5-10 years. The need for a new way to sequence the DNA arose due to limitations in the cost of sequencing DNA via the Sanger method (1-1000 bp for ~\$20-50)[16]; thus, sequencing an entire genome with millions (or billions) of basepairs becomes very expensive to accomplish. To meet this need next generation sequencing was invented with the efforts of both academia and industry combined. There are several platforms currently available to perform next generation sequencing, and they all follow a similar process wherein the genome is fragmented into smaller pieces, and then the pieces are all sequenced independent of each other. Different platforms have different chemistry, and since this sequencing dataset came off of ABI's SOLiD sequencer at the Dana Farber Cancer Institute, a brief SOLiD primer will be described below.

### *SOLiD Sequencing Primer*

The first step to Whole Genome Sequencing (WGS) is the library preparation. The SOLiD system offers a mate-pair library setup, in which (1) the genomic DNA is sheared, (2) size selected for fragments at about 2kb, (3) internal adapters are ligated to the sheared ends, (4) the internal adapters circularize the DNA, (5) the circular DNA is then cut at restriction sites by restriction enzymes, (6) external adapters are ligated to the cleaved ends [Figure 1]. Once the library is constructed, it is combined with DNA polymerase and beads that have the complement primer attached, and then the mixture is subjected to emulsion PCR: where the properties of oil and water are manipulated to create small bubbles that contain the beads, the DNA, the polymerase, and additional primers with streptavidin attached. After the thermocycler has allowed for amplification of the entire DNA in the system, the beads that amplified are pulled down via the streptavidin-amplified primers, and placed on a flow-cell. Once on the flow cell, the actual sequencing of DNA begins.

The sequencing on the SOLiD system involves primer shifting and repetition of ligation reactions, and a breakdown of the process is as follows: (1) universal primer starting at  $n=0$  (first position of the read to be sequenced) is added, (2) 16 different di-base combinations of probes (with two specific bases followed by 3 non-specific bases) coded into four different colors are added, (3) the probes are cleaved and release a fluorophore corresponding to one of the four colors, (4) the process is repeated through the length of the read, (5) the primer is removed and replaced by a primer that is shifted by one position, (6) repeat the steps 1-5. At the end of the sequencing, the data from each primer is condensed and each read is produced with corresponding quality scores that indicate the efficacy of the color calls.

In order to better understand the sequencing of the SOLiD system, a comprehensive tech summary is available at <http://seqanswers.com/forums/showthread.php?t=10>.

### ***The Reads***

The end result of the sequencing run produces a set of mate-paired reads in a standardized format where each pair of reads is given a unique identifier (with the exception of which end was sequenced either F3 or R3 for the corresponding mate), as well as the DNA sequenced in color-encoded format, and a set of quality values (QVs) that correspond to the individual color calls. The quality values are determined based on the accuracy of the color call on a phred scale, and are assigned to each color call in the set of reads as they come off of the sequencer. A description of the phred quality scale and the meaning behind the quality scores is available at [http://en.wikipedia.org/wiki/Phred\\_quality\\_score](http://en.wikipedia.org/wiki/Phred_quality_score).

### ***Whole Genome Resequencing***

This project deals with Whole Genome Resequencing (WGS), in which the entire genome—or collection of the organisms' DNA—is extracted and sequenced. The idea of re-sequencing comes into play because the reference genome from the organism has already been sequenced and had its genome assembled and finished. The genome assembly for *Saccharomyces cerevisiae*—the organism of interest in this project—has a widely-accepted genome assembly that also contains high quality annotations of genes. Since a well manicured “reference sequence” is available and the BY4741 strain is closely related, assembly of the fragmented reads into a genome is unnecessary for this project. Thus, in order to analyze the data for this project, alignment to the already sequenced “reference” genome will allow for identification of what the true sequence is of the strain; and allow for comparisons between the different strains to identify commonalities relative to the reference as well as determining the unique characteristics of each strain.

### **The Goal**

This thesis describes the computational analysis of the whole genome sequencing of 8 strains to identify mutations in the form of single nucleotide polymorphisms (SNPs), small insertions and deletions (indels), and large structural variations (SVs). Strain-unique variations in the evolved diploids and tetraploids could be causal for their increased fitness under the selective pressure of a compromised growth media at the end of 240 generations. In order to efficiently determine mutations present in these strains a pipeline was implemented and tested with the aid of benchmarking through *in silico* short read generation as well as verification of identified mutations through Sanger sequencing.

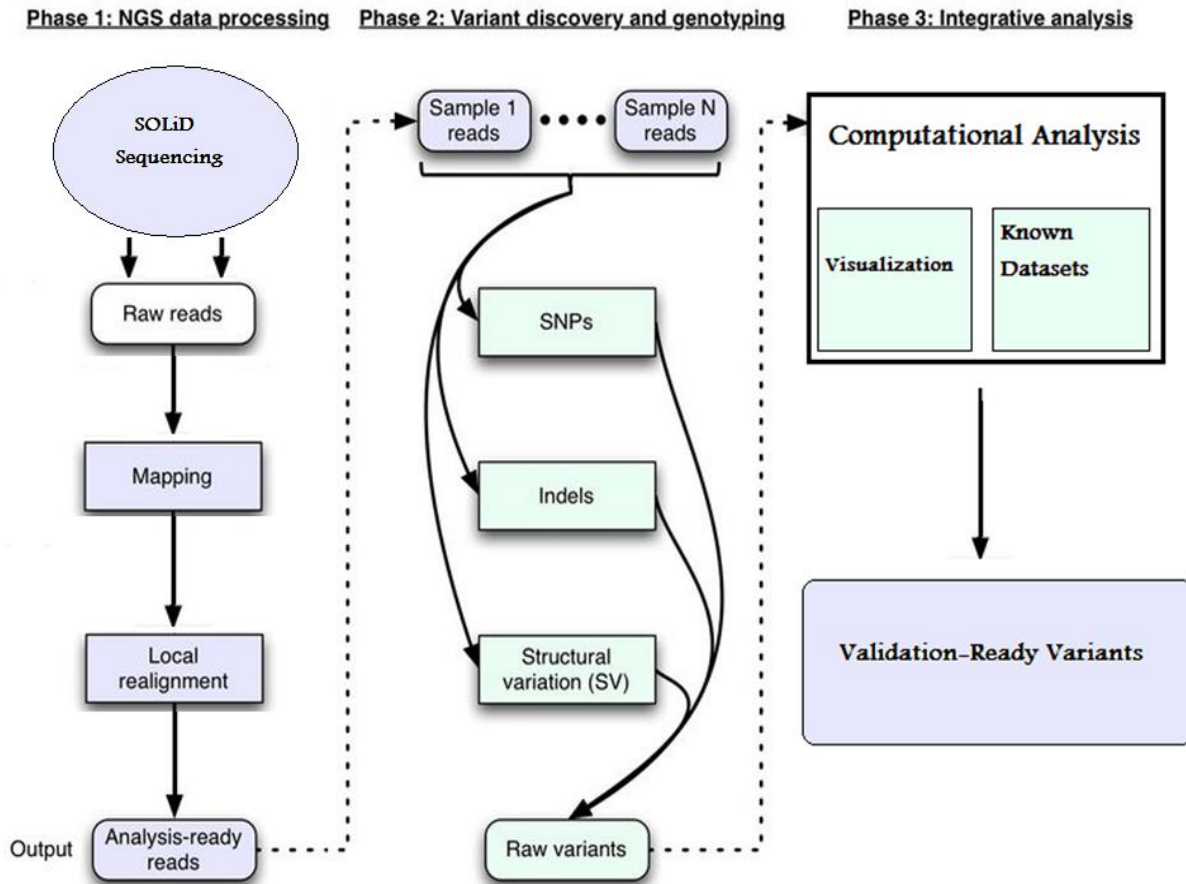
## **Methods**

### **Computational Workflow**

#### ***Identification of Variants***

In order to analyze the data and identify variations in the sequences relative to the reference strain that they will be aligned to, a computational pipeline was implemented. The backbone of the pipeline was derived from the BROAD's “Best Practice for Multi-Sample Variant Calling Pipeline v.3.”

[BROAD WIKI]. However, since the pipeline at the BROAD is designed for variation calling on low coverage (read depth per position) Illumina-sequenced human datasets, it required a little tweaking to be applicable to the project at hand. The pipeline workflow is laid out in Figure 4, but a brief description of the analysis is as follows: (1) raw QV analysis on the unmapped reads, (2) alignment to a reference genome, (3) local realignment to detect small indels, (4) copy number variation analysis, (5) variation calling to identify SNPs, indels, and structural variants, (6) processing of variants to determine true-positives, (7) Sanger-sequencing of variations identified in original samples.



**Figure 1:** Analysis Pipeline adapted from the BROAD, but modified to incorporate SOLiD sequencing, computational benchmarking and manual visualization through IGV.

### *Parental and Strain-Unique Variants*

This pipeline will identify, for each sample, the set of variations present in that sample relative to the reference strain. However, since the parental strain doesn't match the reference strain perfectly, the vast majority of the variations identified in all the samples will be common (e.g. differences between BY4741 and the reference). Thus, in order to identify the variations that are unique to each strain, intersections between the strains together must be leveraged to produce a set of "parental" variations, and then any variations that exist in addition to the parental in any of the evolved strains can be considered to be strain-unique and possibly due to the selective pressure induced during the evolution.

Also, since the experimental setup was such that all of initial strains are isogenic, the identification of parental variants will be limited to cases where the variations relative to the reference are homozygous; occurring in all copies of a chromosome and therefore occurring in all the overlapping reads at the specific position.

### Limitations Due to Issues of Ploidy

The identification of variants in sequencing datasets has been largely limited to the resequencing of haploid or diploid organisms. However, this project entails the sequencing of *S. cerevisiae* strains exhibiting polyploidy and aneuploidy, it is possible that identification of variations occurring in only one chromosome may not be identified. What it means to be “diploid-restricted” is best described by analyzing the underlying equation that is used by the vast majority of variation callers:

$$GL_{(i,j)}(g) = P(\mathbf{B}_{(i,j)}, \mathbf{Q}_{(i,j)} \mid G = g)$$

This equation manipulates Bayesian statistics by determining the possibility of a certain variation where the Genotype Likelihood (GL) is a function of the probability (P) of a base (B) at a given position (i), with a quality score (Q), for all the reads (j) that align over that position (i), given the possible genotypes (G). However, in a diploid-restricted situation, the fifteen possible genotypes are all combinations of two of any of the possible four bases or an indel (Table 1).

--				
-A	AA			
-T	AT	TT		
-G	AG	TG	GG	
-C	AC	TC	GC	CC

**Table 2:** The possible genotypes using the four nucleotides Adenine (A), Thymine (T), Guanine (G), and Cytosine (C). An indel is marked with a (-).

However, in addition to the possible genotypes, the variation callers assume a distribution of reads at a given location such that in order to call a variation at a given position, generally each allele must be supported by  $1/N$  of the reads where  $N$ =assumed ploidy (within a reasonable error range). Thus, when the assumed ploidy is restricted to 2, there will only be variations reported where roughly 50%(±15%) of the reads support one allele, and the other 50%(±15%) support the other allele (15% is an arbitrary standard deviation to accommodate for error). However, strain-unique variations in the higher ploidy progeny will be represented in the reads at proportions closer to 75% reference allele, and 25% alternate allele.

Due to this limitation, and the confounding effect it *could* have in identification of strain-unique variations, an *in silico* experiment was implemented as a precautionary measure. The idea behind the *in silico* experiment is to generate a dataset that mimics the output of sequencing, but leverages the ability to insert variations at known loci and then determine the efficacy of variation callers to identify those variations. The insertion of variations can also be controlled to only appear in a certain fraction of the reads, thus mimicking the idea of ploidy. This process is known as “benchmarking,” and a large part of

the data analysis for this experiment is directed by the results of the benchmarking on higher ploidy data.

### *Software List*

Throughout the pipeline and the analysis of the data numerous mapping tools, realignment tools, variation calling tools, and benchmarking tools were utilized. The benefits and pitfalls of each individual tool were determined by benchmarking and the final results on the ploidy data were determined using combinations of outputs from each. Briefly, the mapping software used in this project were BWA, Bfast, Bowtie, and NovoAlignCS [4,8,9,15]. Due to similar results to other aligners, performance (speed) issues, and licensing restrictions NovoAlignCS were removed from the analysis. The variation calling software used were VARiD, Breakdancer, Samtools Mpileup/BCFtools, and Freebayes [5,6,7,18]. Freebayes was the only tool that claims to have the ability to call variations at a higher ploidy, as discussed in the “Limitations Due to Issue of Ploidy” section. The realignment software SRMA and Samtools BAQ were used, with largely similar results [7,10]. The raw QV analysis was done using a package developed at Rutgers called SOLiD QV Analysis [14]. All visualization was done using IGV (Integrative Genomics Viewer) [11]. Copy Number Variation analysis was done using BEDtools in combination with custom perl scripts [19]. Read generation was done using the read generator TrainQual RS (Alex Poole, Dowell lab, unpublished), as well as using a set of custom perl scripts. Version information is detailed in the specific usage of software below.

### *SOLiD QualityScore Preprocessing and Analysis*

Quality score analysis on the unmapped reads gives an indication of the raw quality of the reads from the sequencer. Developed at Rutgers, the SOLiD QualityScore Preprocessing and Analysis is a set of tools that allows for analysis and possible prefiltering of the raw data. The quality analysis of the raw-unfiltered data was used on each strain individually, yielding results in different categories including: number of reads with an average QV > 10(or 20), and the number of reads that pass through an erroneous error filter (e) only allowing one QV in the first ten bases to be less than 10, as well as a polyclonal error filter (p) only allowing 5 QV scores less than 25. Although analysis was performed, the filtering out of reads prior to downstream processing was not utilized and will be discussed later.

### *Mapping*

Mapping of the short reads to a reference genome involves finding the most likely location for where a read—or a pair of reads—belongs. Different mappers implement different algorithms in this process including hashing and indexing, Burrows-Wheeler Transformation (BWT), and sequence alignment. Details on the algorithms utilized by the software can be found in the software’s corresponding publications. The reads were mapped to the S288c genome acquired from Saccharomyces Genome Database (SGD) on July 28<sup>th</sup> 2010.

#### *Bowtie (v 0.12.7)*

Bowtie is an ultra-fast mapper based on Burrows-Wheeler Transformation (BWT) that is used in most pipelines as an initial alignment for the reads. Bowtie is efficient in handling single-end reads, and performs a non-gapped alignment while allowing for very few variations/mismatches per read.



Bowtie indexes were built using the default parameters with the added option for colorspace via the command:

```
$ bowtie-build <in.fasta> <out.prefix>
```

Bowtie alignments were done both using the default parameters as well as using the unique-mapping option. Reads were aligned in both the mate-paired conformation as well as aligned treating each of the reads in the pair as an independent fragment (allowing reads to only map to one location)

```
(default mate-pair) $ bowtie -v 2 -C -S -f <reference.index> -1 <in.F3.csfasta> -Q1 <in_QV.qual> -2 <in.R3.csfasta> -Q2 <in.R3_QV.qual> > <out.sam>
```

```
(default single-end) bowtie -v 2 -C -S -f <reference.index> <in.csfasta> -Q <in_QV.qual> <out.sam>
```

```
(unique single-end) $ bowtie -k 1 -v 2 -C -S -I 0 -X 4000 -f <reference.index> <in.csfasta> -Q <in_QV.qual> <out.sam>
```

### Bfast (v. 0.6.5a)

Bfast is a slower read aligner that has the ability to handle mate paired reads and provides a gapped alignment. In general, Bfast and also provide a very loose alignment (ability to map to locations where other aligners cannot) because it utilizes rigorous (but computationally expensive) sequence alignment rather than hashing or BWT strategy. Of all the mappers used, Bfast performs the best around small indels due to the loose alignment abilities, but at the same time will align reads of poor quality score and will align reads to locations of lower sequence identity (homopolymeric regions, repeat regions, etc.). There are three steps to the Bfast alignment including match, localalign, and postprocess.

```
bfast match -A 1 -f <reference.fasta> -t -n 8 -T <working directory> -l -K 10 -M 50 -r <in.fastq> > <bfast.match.bmf>
```

```
$ bfast localalign -A 1 -f <reference.fasta> -t -n 8 -M 50 -m <bfast.match.bmf> > <bfast.localalign.baf>
```

```
$ bfast postprocess -A 1 -f <reference.fasta> -t -n 8 -O 1 -a 2 -i <bfast.localalign.baf> -r <readgroup_header> > <out.sam>
```

### BWA (v. 0.5.9)

BWA (v. 0.5.9) was released in April 2011, and served as a mapper with an intermediate strategy, i.e. it wasn't as loose at mapping indels as Bfast, but still provided a limited gapped alignment that also leveraged mate paired alignment. BWA is limited in its gapped alignment capabilities which will be discussed later on. BWA is currently the recommended aligner for colorspace and base-space reads by the BROAD.

BWA indexes were built using the parameters specific to a smaller genome and the alignment of short reads (<200 bp)—as specified in the manual for BWA.

```
$ bwa index -a is -c <in.fasta>
```

The reads were converted from their SOLiD format into fastq format using the default solid2fastq.pl script. So, the reads went from strain.F3.csfasta + strain.F3\_QV.qual and strain.R3.csfasta +

strain.R3\_QV.qual to a strain.F3.fastq and strain.R3.fastq.

```
$ perl solid2fastq.pl <in.title> <out.prefix>
```

They were then aligned using default parameters, but setting the colorspace option and using 8 threads.

```
$ bwa aln -c -t 8 <reference prefix> <in.F3.fastq> > <out.F3.sai>
```

```
$ bwa aln -c -t 8 <reference prefix> <in.R3.fastq> > <out.R3.sai>
```

Then the alignments were combined and finalized using the default parameters, with a maximum insertion of 4000.

```
$ bwa sampe -a 4000 -f <out.sam> <reference prefix> <in.F3.sai> <in.R3.sai> <in.F3.fastq> <in.R3.fastq>
```

### NovoAlignCS (acquired 01/30/2011)

NovoAlignCS ranked high among the recommended aligners for mate-paired reads, and its recent addition to handle colorspace (CS) made it another useful alignment tool. However, NovoAlignCS became software only available by commercial license in February 2011, so it was only used to produce a single alignment for each sample.

Since NovoAlignCS is no longer open source, it is difficult to reproduce the exact command line arguments for the index:

```
$ novoalignCS index -c <in.fasta> <out.prefix>
```

The reads are able to be aligned in their .csfasta and QV.qual formats, and were aligned using a mate paired insert of 2000 with a standard deviation of 1000, using 5 threads and outputting SAM format.

```
$ novoalignCS -c 5 -d <reference prefix> -F CSFASTAnQV -f <in.F3.csfasta> <in.R3.csfasta> -I MP 2000,1000 -o SAM
```

### Realignment

Mapping software treats each read independently and therefore is limited in its ability to distinguish small indels from SNPs. Realignment attempts to consider the set of reads overlapping a continuous region in order to identify small indels.

### Samtools BAQ (v 0.1.16)

BAQ uses a fast heuristic to identify regions that could contain a small indel, and then realigns the reads around the potential indel to produce consensus at a position supported by the majority of the reads. Realignment by BAQ was run on all samples, regardless of whether or not they had been realigned by SRMA. The realignment command used was:

```
$ samtools fillmd -bA <in.sorted.bam> <reference.fasta> > <out.sorted.baq.bam>
```

### SRMA (v 0.1.15)

SRMA is the recommended realignment tool to accompany the Bfast alignment [10]. The realignment uses a rigorous graph-based strategy, and is used in conjunction with BAQ to provide the most comprehensive realignment possible.

```
$ srma REFERENCE=<reference.fasta> INPUT=<in.bam> OUTPUT= <out.realigned.bam>
```

## Copy Number Variation

The aCGH data performed on the strains (by collaborators at Dana Farber Cancer Institute) identified specific aneusomies (individual chromosomes with a copy number different than the rest of the chromosomes) and segmental amplifications around the HXT6/7 genes. Since sequencing equally represents all the DNA in the system, the read pileups should give insight into the presence or absence of copy number variations. The comparison between the read pileups and aCGH data allow for both a secondary verification of the copy number variations as well as a confirmation that the reads are being mapped in a way that corresponds to the expected genome-wide distribution.

## Bedtools (v 2.12.0)

In order to generate a read pileup that can be visualized in IGV and be compared to the aCGH data, a series of custom scripts and Bedtools commands were used. The following is the series of steps used to generate the short read pileup:

1. A tab delimited Bed file with chromosome, start, and stop was generated using a custom perl script that takes in a fasta, a window size, and a step size.
  - a. This was done using two different types of steps, one was a step of 50 and a window size of 50 (no overlap), and the other was a step of 50 and a window of 500 (contains ample overlap).
2. These files were then input into the Bedtools coverageBed command where the alignment and the interval file are taken and produce a new tab delimited Bed file with the interval information, as well as the number of reads that pileup underneath each interval.
3. The files were then passed through a different perl script in order to convert them into a WIG file format.
4. The WIG file format used was a fixedStep wig, with a step of 50 and a span of either 50 or 500 (depending on the input).

This output can then be visualized in IGV.

## CGH translation

In order to easily compare the aCGH data with the read pileup data, the aCGH files were translated into a file format compatible with IGV. The aCGH files (Excel worksheet formatted files) were translated into files that can be visualized in IGV in the following steps:

1. Download the sequences of the corresponding probes from Agilent eArray website. This will produce a fasta formatted file with the probe name and corresponding sequence.
2. Align the probes to the reference genome of interest (S288c) via bowtie with the following command:
  - a. `$ bowtie <index prefix> -S -f <probes.fasta> <mapped_probes.sam>`
3. Parse the file to pull out from each read in the sam file the probe ID and the alignment location.
  - a. This should produce a 2 column tab-delimited file that can be viewed in Excel, and then sorted based on probe ID.
4. In excel, sort the entire aCGH file by the ID of the probes.

5. Insert the 2 columns from the mapped probes (which should also be sorted).
6. Double check that the probe ID's matchup between the new 2 inserted columns and the original probe ID's of the aCGH file.
7. Copy from the aCGH file the locations of the mapped probes into a new excel file called <sample>.wig.
8. Copy from the aCGH file the column of values for a given sample and place them next to the corresponding mapped locations in the sample.wig file.
9. Insert a new row at the top of the file, and fill it with (don't include the quotation marks):
  - a. "track type=wiggle\_0 name=sample.name description=variableStep format visibility=full autoscale=off color=50,150,255 yLineMark=0 priority=10"
10. Now, insert a new row whenever there is a change of chromosome, starting with a row before chr1 (don't include the quotation marks).
  - a. "variableStep chrom=chr1 span=60"
  - b. Repeat this for each chromosome changing the chrom=chr# to fit the corresponding chromosome
11. Now, take this excel file, convert it to a tab-delimited text file, and bring it up in a text editor.
  - a. If there are any extra quotation marks where there shouldn't be due to the translation of excel.xlsx to sample.wig.txt remove them

The end result is a WIG file, with objects that map to the correct location of the genome, are of length 60, and have a value corresponding to the value of the probe in the aCGH file.

### *SNP/Indel calling*

#### Samtools Mpileup/BCFtools (v0.1.16)

Samtools Mpileup is a tool that is a part of the larger Samtools package, and is a well-supported variation calling tool. The Samtools package is used in many aspects of post-mapping modifications and analyses, and the Mpileup tool in conjunction with BCFtools (another tool in the Samtools package) analyze the sum of all the reads on a per-position basis to produce variation calls and various annotations.

Samtools Mpileup requires preprocessing of the references into an indexed reference via the command:  
\$ samtools faidx <in.fasta>

Then the pileup command can be run which will generated a binary pileup readout for every position in the reference:

```
$ samtools mpileup -uf <in.reference.fasta> <in.sorted.bam> > <out.raw.bcf>
```

Then SNPs and indels will be extracted from the pileup and outputted into a raw vcf file:

```
$ bcftools view -bvcg <in.raw.bcf> > <out.raw.vcf>
```

Then the raw SNPs/Indels will be filtered to exclude any variants that occur at a depth greater than 500 (since any region with a depth of > 500 are the result of alignment artifacts).

```
$ bcftools view <in.raw.vcf> | vcfutils.pl varFilter -D500 > <out.sorted.vcf>
```

### VARiD (v. 1.0.7)

The nature of the VARiD algorithm is to load the entire reference sequence into memory, so that the HMM forward-backward algorithm can be utilized to find the best variants over a region. However, this is impractical due to the fact that the complete genome is ~12 million bases long. Therefore, in order to use the VARiD software, a series of scripts was used to preprocess the reference.fasta sequence into 50kb (kilo-base) fragments with 500 bp overlap, extract the reads that align to those fragments, and then run VARiD on each fragment. The 500 bp overlap allowed the forward-backward algorithm to maintain continuity between segments. Once all the fragments had been run individually, they were concatenated into a single variation file, and another postprocessing step was required to filter out any resultant variations that were duplicates due to the overlap. The actual command used to run VARiD including expected frequencies for A,T,G, and C (calculated from the reference genome) was:

```
$varid-exec --threads 4 -r <50kb fasta fragment> -a <isolated reads.sam> --format vcf --freq-a .309 --freq-c .192 --freq-t .309 --freq-g .309 -o 50kb.location.vcf
```

### Freebayes (v 0.8.9)

Freebayes was used as the higher ploidy/variable ploidy variation caller for the strains that are of higher ploidy and exhibit aneuploidy. In early releases of Freebayes, the false-positive error rate was too high to be worth analyzing. However, in the most recent release (v 0.8.9 released 8/23/2011), much more manageable numbers of reported variations in the higher ploidy strains are reported.

Freebayes allows for an option in which a separate bed file containing the copy number on a per-region basis is specified, and the identification of variants in each region corresponds to the copy number. The separate bed file was determined based on analysis of aCGH data as well as the depth of coverage data. The command used to run the strains through freebayes was:

```
$ freebayes -f <genome> -b <alignment.sorted.bam> -p <base ploidy> -A <specific bed file> -v <out.vcf>
```

Example lines from the region bed file in tab-delimited form:

```
chrI 0 230208 6040.bwa 4
chrII 0 813178 6040.bwa 4
chrIII 0 316617 6040.bwa 4
chrIV 0 776999 6040.bwa 4
chrIV 777000 1149999 6040.bwa 4
chrIV 1150000 1165000 6040.bwa 4
```

(Note, segmental aneuploidies on chrIV can also be designated).

Freebayes was also run on each strain with different base ploidies for each strain corresponding to what their base ploidy or aneuploidy levels are as determined by the combination of the aCGH data, and the flow cytometry analysis (performed by our collaborators at the Dana Farber Cancer Institute).

```
$ freebayes -f <genome> -b <alignment.sorted.bam> -p <base ploidy> -C <minimum # of reads supporting alternate allele> -v <out.vcf>
```

## *Structural Variation Calling*

Despite the relatively small insert size between our mate paired reads (2kb), we sought to identify any large structural variations by analysis of mate pairs.

### Breakdancer (v. 1.1\_2011\_02\_21)

Breakdancer was run on all the strains in order to identify large chromosomal translocations, interchromosomal translocations, transversions, inversions, and large deletions.

The first step to running Breakdancer is to find the mean insert size between mate paired reads, as well as a standard deviation. This is accomplished using the perl script that accompanies the Breakdancer package:

```
$ perl bam2cfg.pl <in.matepaired.sorted.bam> > out.sample.cfg
```

The output .cfg (configuration) file contains the mean and standard deviations for the insert sizes of the reads. The next step is to combine the .cfg file with the bam alignment file and run Breakdancer:

```
$ breakdancer_max <in.cfg> <in.sorted.bam> > <output.SVs.txt>
```

## *Benchmarking*

The goal of benchmarking in this analysis was to determine the efficiency and capability of our pipeline, and the specific variation callers we used in the pipeline, at identifying variations at a higher ploidy.

### Custom Scripts

A set of custom Perl scripts were written in order to generate simulated reads from a reference sequence file. Briefly, the script would input a reference.fasta, manipulate object oriented Bioperl package programming to turn the reference into reads at a specified depth along the genome—which was 100bp. After the reads were generated, another set of scripts would define locations for SNPs and incorporate them into a certain fraction of the reads in order to mimic ploidy, where the mutation would be incorporated into  $1/N$  reads with  $N$ =ploidy. The incorporation of a variation into a fraction of the reads was based on a random number generator to produce a distribution of allelic frequencies.

To test the ability of the variation callers to identify SNPs at a higher ploidy, 150 SNPs were incorporated into chromosome 1 at ploidies of 1N, 2N, 3N, 4N, 5N, and 6N. The SNPs were each 500 bp apart and spanned from 10kb to 85kb. After the reads with the SNPs incorporated were generated, they were mapped using BWA default settings (described above) and called using both Freebayes and VARiD.

### TrainQual RS

TrainQual RS was used due to the limitation of the custom perl scripts at incorporating variations into the reads after they have been generated. This is not feasible for indels, since deletions or insertions would cause the reads to change in size and give errors to the alignment software. The read generator (Alex Poole, Dowell lab, unpublished) uses a variation file that contains variant information on a location basis. Then, while it is generating the reads it incorporates the variation information specified, instead of a post-processing step like the above described read generator by custom Perl scripts.

To test the ability of variation callers to identify indels at a higher ploidy, 25 deletions were incorporated into chromosomes 1-5, at a ploidy respective to the chromosome (chr1=1N, chr2=2N, etc.). The deletions were each 500 bp apart and spanned from 10kb to 22.5kb. Then the reads were aligned using BWA default settings (described above), and called by the variation caller Freebayes, which claims to identify indels at higher ploidy.

### *Sanger Sequencing Verifications*

Sanger sequencing verifications were all performed at the Dana Farber Cancer Institute by our collaborator Anna Selmecki of the Pellman Lab. Briefly, the locations that needed to be verified by Sanger sequencing were sent to the Pellman Lab, and then they designed primers flanking the region of interest, PCR amplified the region, and then Sanger sequenced at the Dana Farber Cancer Institute. The resulting AB trace files from the Sanger sequencing center were then interpreted by our collaborator Anna Selmecki. This allowed for the confirmation of SNPs and Indels in the sequencing data relative to the reference genome, as well as identification of strain-unique variations at variable ploidy.

### *GATK (v 1.4-7-gc96fee4)*

The Genome Analysis Toolkit (GATK) is a set of tools implemented by the BROAD for the purposes of variation calling in the 1000-Genomes project (sequencing of 1000+ human genomes). It isn't designed for the dataset at hand, but it provides a different set of standardized tools to be used on the current dataset in identification of variations. The primary difference between the pipeline we've implemented and the one offered by the BROAD is the idea of base quality score recalibration (BQSR). BQSR is designed to leverage information from the alignment, as well as known loci for variations, and recalibrate the quality scores of the reads to more accurately represent the error rate they are intended to indicate. The expectation from using the toolkit on this dataset is that it will produce largely similar results in identification of variations, but since it is designed for use with an Illumina dataset and is currently restricted to analyzing diploid organisms, expectations for its effectiveness at identifying strain-unique variations are not very high.

### *BWA alignment*

Since BWA is the recommended aligner for the GATK's Multi-Sample Variation Calling Analysis Pipeline, it was used for this pipeline. The alignment used contained default BWA settings, with the exception of adding an Illumina read group tag (@RG) for the sequencing platform. This is because the output of BWA is in base-space, so the downstream GATK tools expect to see base-space reads. Bfast, which does output colorspace read alignments, failed the mate-read-filter with 100% of the reads and was therefore no use in the GATK pipeline (follow-up on this issue still in progress). NovoAlignCS alignments were also pushed through the GATK but with largely similar results to the BWA, so the results will not be shown.

### *LeftAlign Indels*

GATK manipulates an indel-realigner that incorporates a list of known sites of indels in the human genome and then realigns around them. However, since this is not available for *S. cerevisiae*, this tool wasn't used in the GATK pipeline analysis. In its place, BAQ and SRMA (described above) were used to realign around indels.

## Base Quality Score Recalibration (BQSR)

### *CountCovariates using parental locations*

In order to manipulate the BQSR, a set of known locations of variations is needed. Generally, in the main usage of this tool in analyzing human genomes, a dbSNP file containing the known sites of variations is available. However, since this is not available for the organism this project concerns (*S cerevisiae*), a file containing sites of variations must be made. The process of making this file is to run a variation caller, determine what the true, most confident SNPs are for the parental strain, and then use that “parental” list to recalibrate all the samples. This was accomplished by taking the intersection of all of the output variation call format files (VCFs) for all the strains, and then stepping through and manually confirming or denying each call as either a true variation, an artifact of the alignment, or a sequencing error pileup. This was done largely by eye; leveraging the alignments from multiple aligners, with the additional power of manipulating Sanger verifications for regions where the true sequence identity was unclear (examples given in Results/Discussion).

```
java -Xmx10g -jar /usr/local/src/GenomeAnalysisTK-1.4-7-gc96fee4/GenomeAnalysisTK.jar -R <in.fasta>
-I <in.bam> -T CountCovariates -cov ReadGroupCovariate -cov DinucCovariate -cov CycleCovariate -
knownSites <parental.vcf> -recalFile <out.strain.covariates> --solid_nocall_strategy PURGE_READ
```

### *AnalyzeCovariates on Covariates output*

AnalyzeCovariates is a tool that uses the output of CountCovariates as its input, and will run analyses on the data to produce plots that describe both the empirical and the actual representations of the quality scores in the data and their efficacy at identifying true sequencing error based on the Phred quality scores. Since the pre-recalibration data uses a quality score of 0-60 (color-space) and the output is designated for Illumina 0-40, the efficacy of this tool on SOLiD data is unclear.

```
java -Xmx10g -jar /usr/local/src/GenomeAnalysisTK-1.4-7-gc96fee4/AnalyzeCovariates.jar -recalFile
<strain.covariates.output> -outputDir analyzeCovariates/ -maxQ 60
```

### *TableRecalibration using Covariates output*

TableRecalibration is the tool that takes the CountCovariates output and applies it to the alignment to produce a recalibrated alignment. The resulting recalibrated alignment then should have quality scores that are more indicative of error probability, as well as a more normal distribution of quality scores.

```
java -Xmx10g -jar /usr/local/src/GenomeAnalysisTK-1.4-7-gc96fee4/GenomeAnalysisTK.jar -R <in.fasta>
-I <in.bam> -T TableRecalibration -recalFile <strain.covariates.output> -o <recal.strain.bam> -
solid_nocall_strategy PURGE_READ
```

### *CountCovariates on recalibrated alignment*

As described above, CountCovariates can be run on any alignment, including a recalibrated alignment. This step is used to produce a new set of Covariates that can be analyzed and produce similar plots that describe the empirical and actual representations post-recalibration.



```
java -Xmx10g -jar /usr/local/src/GenomeAnalysisTK-1.4-7-gc96fee4/GenomeAnalysisTK.jar -R <in.fasta>
-I <in.bam> -T CountCovariates -cov ReadGroupCovariate -knownSites <parental.vcf> -recalFile
<strain.postrecal.covariates> --solid_nocall_strategy PURGE_READ
```

### AnalyzeCovariates on recalibrated Covariates

As described above, this same step is run on the recalibrated set of Covariates produced by running CountCovariates on the recalibrated alignment. The same plots will be produced.

```
java -Xmx10g -jar /usr/local/src/GenomeAnalysisTK-1.4-7-gc96fee4/AnalyzeCovariates.jar -recalFile
<strain.covariates.output> -outputDir analyzeCovariates/PostRecal/ -maxQ 60
```

### UnifiedGenotyper SNP calling

The last step to the pipeline is running the variation caller built into GATK. This was done on both the recalibrated alignments as well as the raw alignments. The input is the alignment and the output is a VCF file containing variations.

```
java -Xmx10g -jar /usr/local/src/GenomeAnalysisTK-1.4-7-gc96fee4/GenomeAnalysisTK.jar -R <in.fasta>
-I <in.bam> -T UnifiedGenotyper -glm BOTH > <out.vcf>
```

## Results

### SOLiD QualityScore Preprocessing and Analysis

Ploidy	Strain	Percentage QV>20		Percentage Passing p=1 e=5	
		F3	R3	F3	R3
4N	6040	22.2	46.6	13.2	30.4
4N	A8	32.1	56.0	20.6	38.2
2N	B2	35.8	63.8	19.7	40.2
4N	D9	66.8	74.7	48.0	59.0
4N	F2	56.9	71.5	40.4	55.5
2N	F12	34.7	59.8	19.6	37.9
4N	G2	49.9	70.3	36.0	53.3
4N	G11	29.81	58.8	19.2	39.4

**Table 3:** The results from the SOLiD QualityScore Analysis tool run on each strain for both the F3 and R3 reads (each read in the mate pair). The middle columns represent the percentage of the reads that pass the requirement of having an average quality score of at least 20. The right columns represent the number of reads that pass through a filter with a p=1 and an e=5 (described in Methods).

## Alignment

<u>Strain</u>	<u>Reads</u>	<u>Avg QV &gt; 10:</u> <u>F3</u>	<u>Avg QV &gt; 10:</u> <u>R3</u>	<u>Bowtie F3</u> <u>Mapped %</u>	<u>Bowtie R3</u> <u>Mapped %</u>	<u>Bfast Pairs</u> <u>Mapped %</u>	<u>NovoAlignCS</u> <u>Pairs Mapped</u> <u>%</u>	<u>BWA Pairs</u> <u>Mapped %</u>
<b>6040</b>	57733300 P:(28866557)	41.60%	84.60%	14.60%	35.88%	12.28%	15.02%	17.66%
<b>A8</b>	47429857 P:(23714918)	53.70%	87.90%	22.04%	43.70%	17.19%	21.33%	24.14%
<b>B2</b>	56111357 P:(28055673)	64.00%	90.40%	22.46%	47.57%	18.08%	18.61%	21.59%
<b>D9</b>	53487836 P:(26743915)	90.20%	91.70%	51.22%	64.28%	38.70%	41.14%	48.63%
<b>F12</b>	60789022 P:(30394507)	79.80%	91.30%	22.50%	45.64%	17.96%	19.46%	24.41%
<b>F2</b>	55747962 P:(27873981)	60.40%	88.70%	45.31%	51.37%	34.90%	42.51%	44.01%
<b>G11</b>	65509215 P:(32754607)	70.40%	90.60%	38.65%	59.24%	16.64%	19.54%	24.04%
<b>G2</b>	57290351 P:(28645153)	53.40%	88.00%	38.65%	59.24%	28.78%	34.31%	40.30%

**Table 4:** The results from the different alignments generated from each of the mapping software. From left, the strain, the total number of reads (number of pairs below), average number of reads in F3 and R3 for each strain that pass an average QV score of 10 (determined using QV analysis above), Bowtie single fragment mapping of F3 and R3 individually, Bfast percentage of pairs mapped, NovoAlignCS percentage of pairs mapped, BWA percentage of pairs mapped.

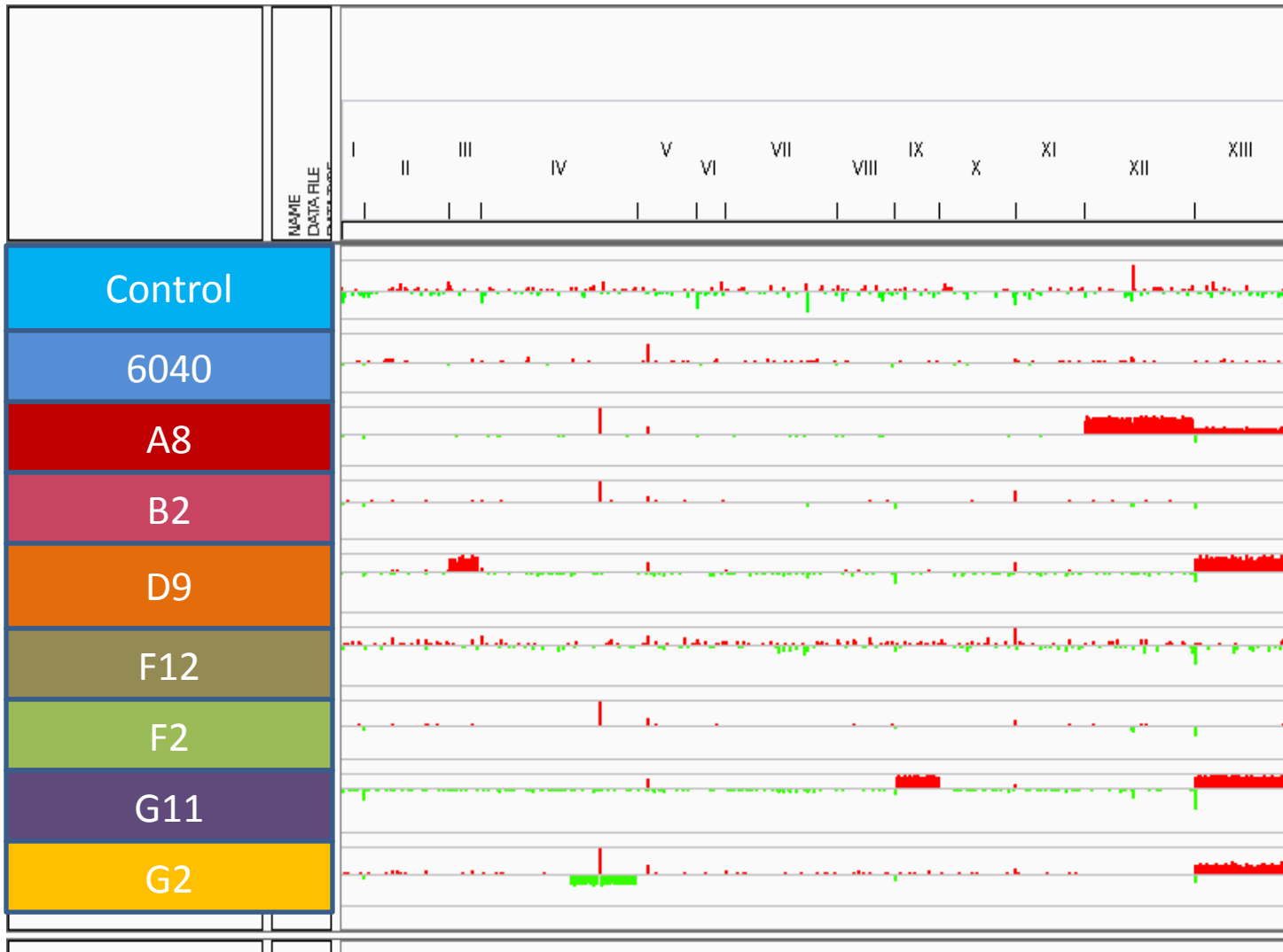
## Realignment

Realignment was performed on all the alignments using both SRMA and BAQ (data not shown).

## Copy Number Variation

### aCGH conversion

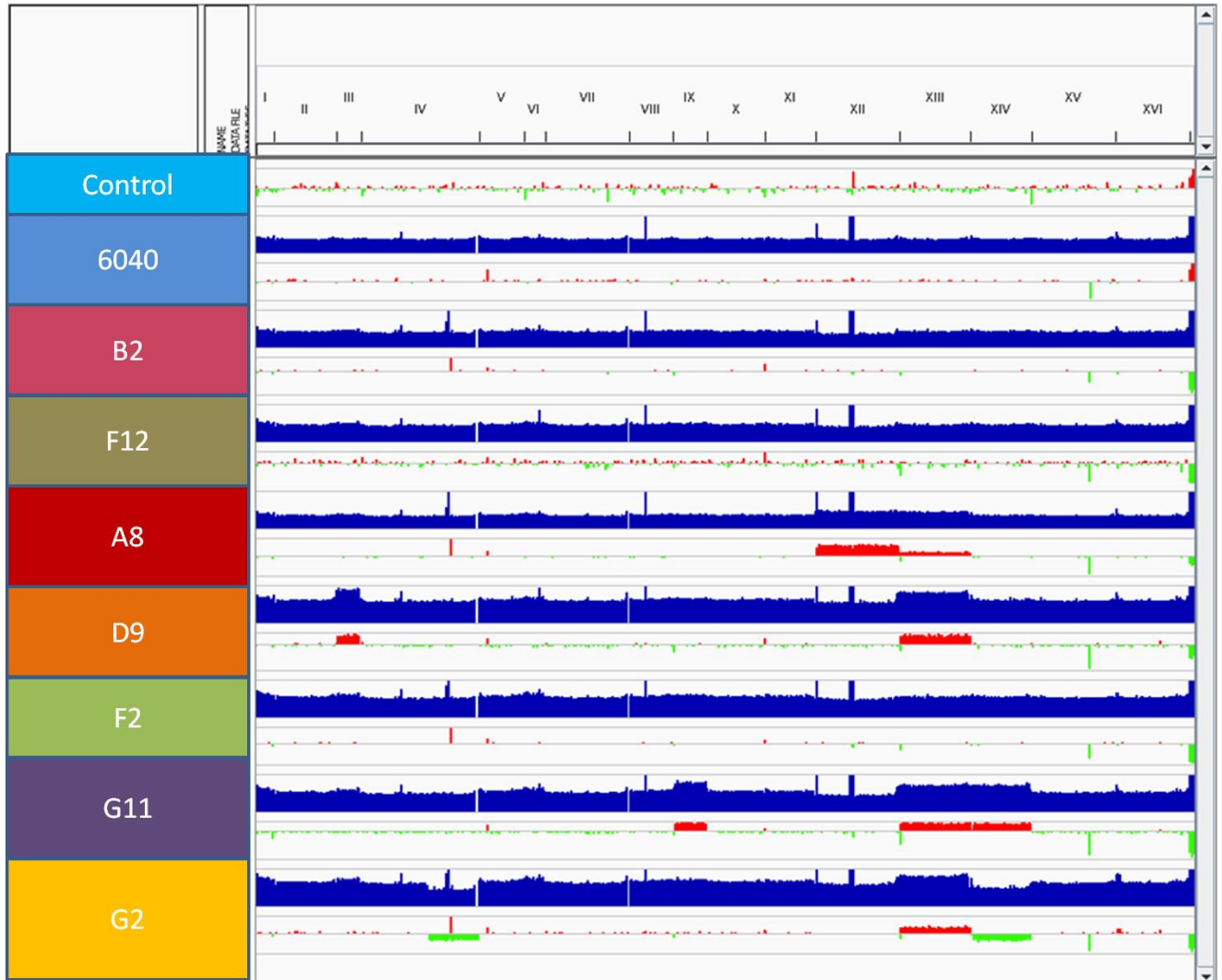
Below are the results from the aCGH conversion into a format that can be visualized in IGV. Across the top on the x-axis are the chromosomes ordered from one to sixteen in roman numerals, flanked on the right side by the mitochondrial DNA. The control lane is the parental strain in its haploid form, run on a CGH array against itself. The 6040 is the tetraploid form of the parental strain, compared against the haploid control. The rest of the lanes are the evolved progeny against the control parental haploid strain. Green represents  $\log_2()$  values less than 0, indicating that the copy number is decreased relative to the normal level for the rest of the genome, and Red represents  $\log_2()$  values greater than zero, indicating that the copy number is increased relative to the normal level for the rest of the genome. Interpretation of one of the progeny, D9, shows an aneuploidy of chromosome 3 (III), as well as chromosome 13 (XIII), but doesn't exhibit the HXT 6/7 amplification.



**Figure 2:** An IGV (Integrative Genomics Viewer) snapshot of the converted aCGH data into a file format utilized by IGV. The control against itself on the top, and then the parental strain followed by the evolved progeny.

#### Depth of Coverage

Below are the results for the depth of coverage (blue tracks) placed above their corresponding CGH conversions (red/green tracks). Across the top on the x-axis are the chromosomes 1-16 labeled via roman numerals. The first snapshot shows the control, 6040 (parent), and B2 and F12 (diploid progeny). The second snapshot shows the control, A8, D9, F2, G11, G2 (tetraploid progeny).



**Figure 3:** IGV snapshot of the depth of coverage (blue tracks) placed above their corresponding CGH conversions (red/green tracks). Aneusomies identified by the aCGH (red/green) are also visible in the depth of coverage data of the read pileup.

### *SNP/Indel calling*

Below are the SNP/indel calls for the BWA alignments using different variation callers. These alignments were chosen to sort through for identification of strain-unique variations and common variations due to the ability of BWA to perform a gapped alignment, as well as the high percentage of parental SNPs present in each individual sample, as well as the low number of strain-unique SNPs in the parental strain. This is expected, since the number of strain-unique SNPs after 240 generations should not be several hundred (Bfast-VARiD data not shown) and the number of strain-unique SNPs in 6040 should be close to zero. The tables below show the total number of variation calls per strain, an intersection of all the variations with 1/2 strains required to have a given variation, and from that intersection the homozygous SNPs and indels (which will be later submitted to parental categorization and verification).

## Samtools Mpileup

Samtools Mpileup data not shown as VARiD was able to call all variations that Samtools Mpileup could call.

## VARiD

Below are the raw, unfiltered numbers of SNPs and indels generated from the BWA-VARiD alignment. The Unique variations are created by comparing each individual strain's variations against an intersection of at least 4/8 strains together.

	BWA-VARiD	BWA-VARiD Unique
6040	1138	113
A8	1094	69
B2	1065	40
D9	1146	121
F12	1046	21
F2	1198	173
G11	1135	110
G2	1160	135
Intersect (4/8 req)	1025	-
Intersect (homo)SNPs	552	-
Intersect (homo) INDELS	325	-

**Table 5:** The number of variations called by VARiD on the BWA alignment, as well as the unique variations for each strain. The left column contains the strains, the middle column contains the number of variations per VARiD run, and the right column contains the strain-unique variations per strain.

## Freebayes

Below are the raw, unfiltered SNPs and indels from the Freebayes alignment. For the evolved progeny the alignments were run through Freebayes using different base ploidy options (n=1, n=2, n=3, n=4, n=5, n=6). The Intersect is generated with the variations from Freebayes when each strain is run with the lowest ploidy shown (A8 at n=4, B2 at n=2, etc.). The Unique variations however, are the highest ploidy Freebayes runs compared to the intersect.

	BWA-Freebayes	BWA-Freebayes Unique
6040 (n=1)	433	1
A8 (n=4)   (n=5)   (n=6)	631   736   1164	360
B2 (n=2)   (n=3)	621	52
D9 (n=3)   (n=4)	748   751	55
F12	504	54
F2 (n=4)	796	235
G11 (n=3)   (n=4)	716   719	159
G2 (n=3)   (n=4)   (n=5)	806   812   1029	467
Intersect (4/8 req)	613	-

**Table 6:** The number of variations called by Freebayes on the BWA alignment, as well as the unique variations for each strain. The left column contains the strains, and the ploidy at which they were run. Evolved progeny (except for F12) were run at multiple ploidies separated by “|”. The middle column contains the corresponding number of variations per Freebayes run, and the right column contains the strain-unique variations per strain.

### Parental Variations

The total number of variations that occurred in either VARiD or Freebayes on the BWA alignment data in an intersection of at least 4/8 strains was 580 SNPs, and 325 indels. Manual Post processing of the SNPs and indels occurred through the usage of IGV, wherein the variations were categorized into true parental, artifact of the alignment, or questionable. In order to be a true parental, the variation must occur with consensus in multiple alignments, and not be a part of a homopolymer or occur in a region of low coverage (read depth < 10)—and if they do not meet both these criteria then they are considered artifacts of the alignment and not true parentals. However, in this identification process, certain interesting results occurred including situations where the different aligners could not come to a consensus on what the true identity of the sequence was, and it was not due to a lower coverage or homopolymeric underlying sequence. These were marked as questionable, and submitted for Sanger verification. The final set of parental SNPs and Indels contains 396 SNPs and ~250 indels (still waiting on some indel confirmations). Examples of cases where the true identity of the SNPs or indels were unclear include dinucleotide SNPs, and multiple insertions and deletions within a single read. Figures with the different alignments and the Sanger results are shown in the Supplemental Figures.

### Structural Variation Calling

The first step in the structural variation calling produces configuration files in which the mean and standard deviation of the insert size between the two mate paired reads for each sample is determined from the alignment. The following table is the results of the mean and the standard deviations:

Strain	BWA Mean and Standard Deviation	NovoAlign Mean and Standard Deviation
6040	1654 ± 410	1607 ± 401
A8	1575 ± 376	1538 ± 369
B2	1180 ± 498	1132 ± 483
D9	1113 ± 341	1076 ± 322
F2	1813 ± 394	1773 ± 378
F12	1166 ± 397	1121 ± 375
G11	1731 ± 354	1697 ± 344
G2	1595 ± 373	1532 ± 340

**Table 7:** For each strain (left column) the mean ± standard deviation for the BWA (middle) and NovoAlignCS (right) alignments.

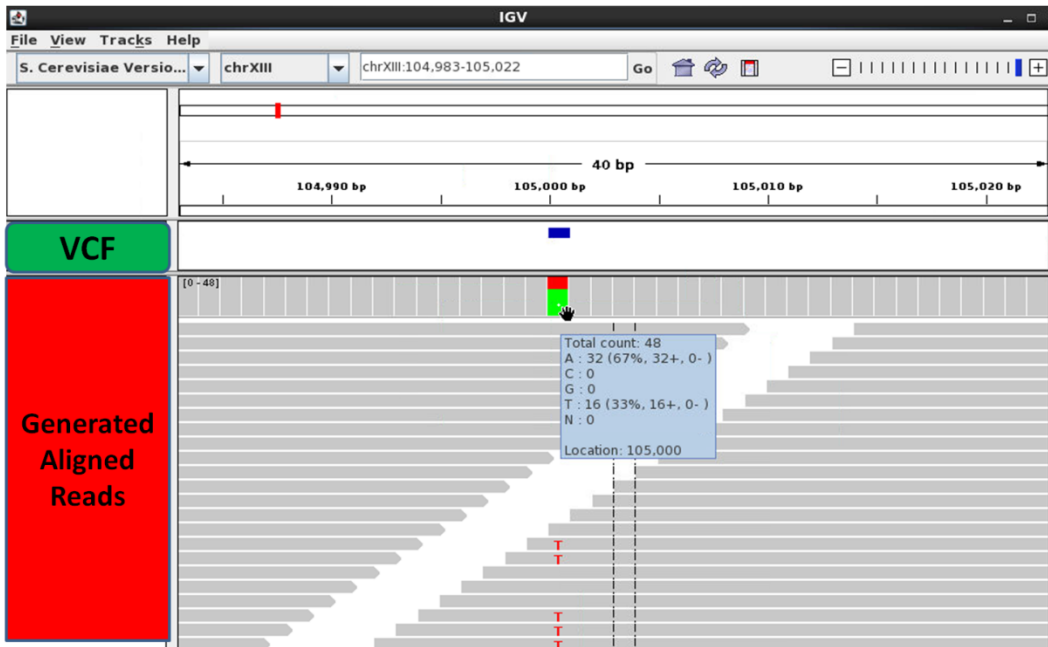
Due to the high standard deviation in the insert sizes, the number of structural variations reported per strain was roughly 20000/strain, including that many in the parental strain. In addition, when a mapping program allows non-unique locations this contributes to SV overcalling. Further analysis on identifying structural variations is currently ongoing and thus final results for strain-unique structural variations are unavailable.

### *Benchmarking*

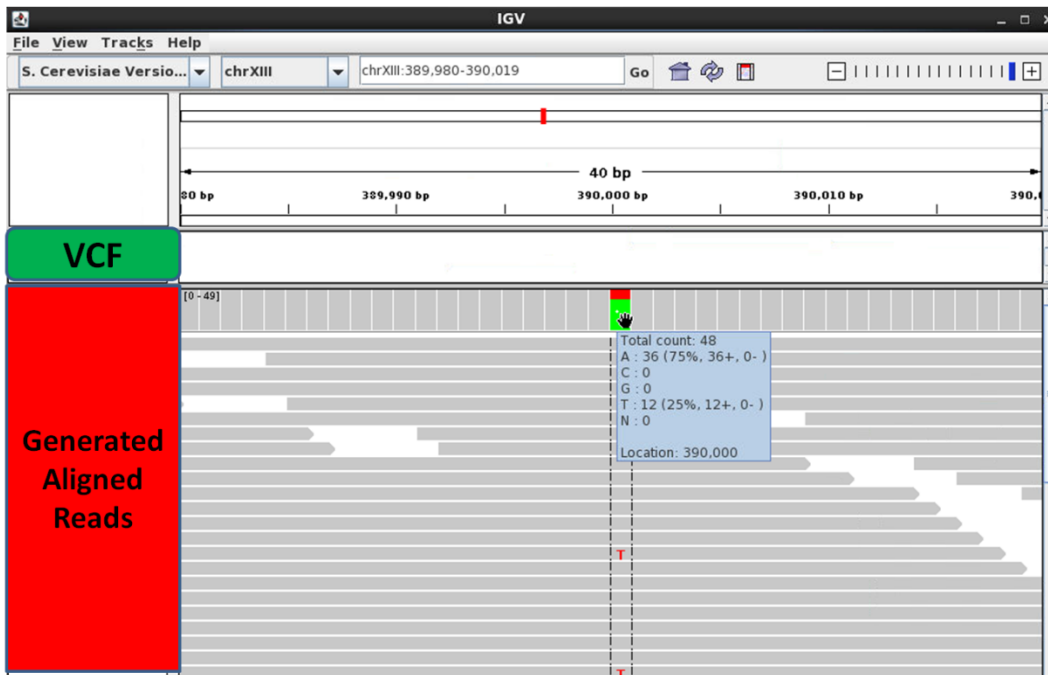
Below are the results from the benchmarking of SNPs and Indels using the set of custom perl scripts and TrainQual RS respectively.

#### Custom Perl Scripts

After the benchmarking of SNPs via custom perl scripts, interesting results ensued. Instead of definitive yes or no answer to the limitations of polyploidy on the diploid-restricted variation callers, a cutoff was observed. In this cutoff, only when >32% of the reads supported the alternate allele was it possible for the variation caller to identify the SNP (blue bar above the coverage track). When the alternate allele was <32% supported by the reads, the SNP was not called (absence of the blue bar above the coverage track).



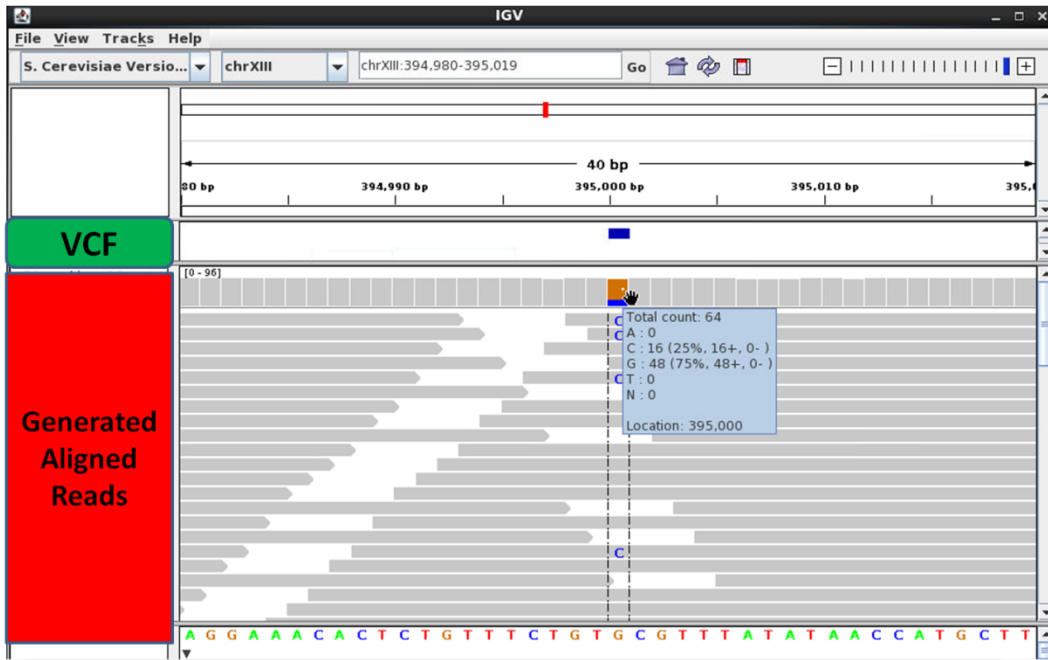
**Figure 4:** An IGV snapshot containing the mapped reads (Generated Aligned Reads), a histogram denoting read pileup, and the variation calls (VCF). VARiD is able to identify a SNP at an alternate allelic frequency of 33%: The reference is A (67% of the reads) the alternate allele is T (33% of the reads).



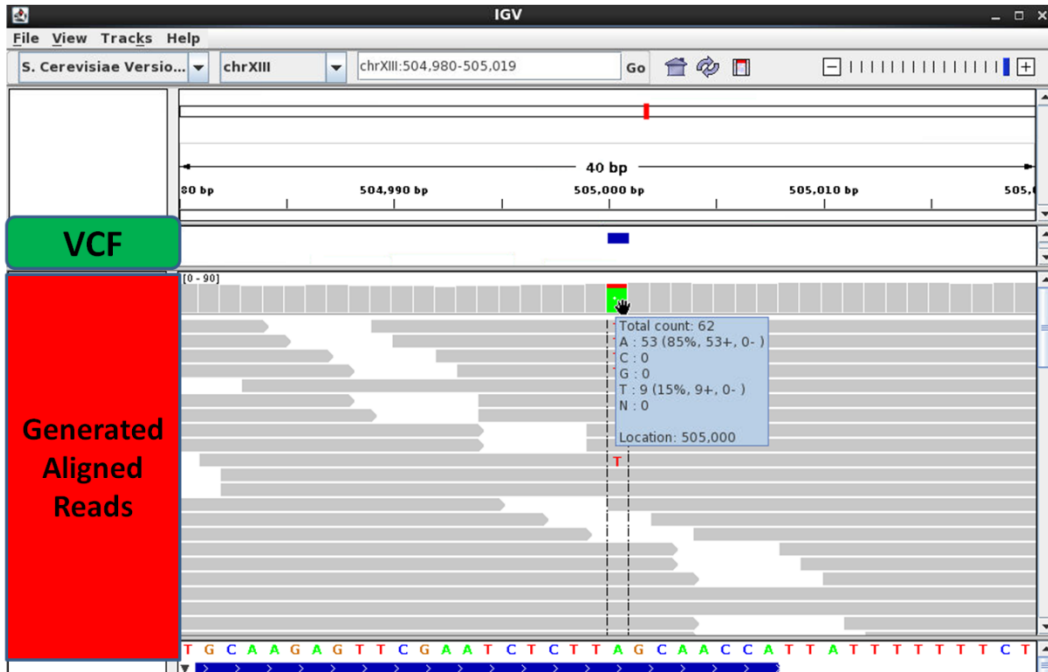
**Figure 5:** An IGV snapshot containing the mapped reads (Generated Aligned Reads), a histogram denoting read pileup, and the variation calls (VCF). VARiD is not able to identify a SNP at an alternate allelic frequency of 25%: The reference is A (75% of the reads) the alternate allele is T (25% of the reads).



However, when Freebayes was run on aligned reads with SNPs incorporated at variable ploidy, all of the SNPs were identified. The only exceptions to this were cases where the SNPs were incorporated into reads in a region where an artifact of the alignment existed—including any region where the coverage was not the expected 100bp.



**Figure 6:** An IGV snapshot containing the mapped reads (Generated Aligned Reads), a histogram denoting read pileup, and the variation calls (VCF). Freebayes is able to identify a SNP at an alternate allelic frequency of 25%: The reference is G (75% of the reads) the alternate allele is C (25% of the reads).



**Figure 7:** An IGV snapshot containing the mapped reads (Generated Aligned Reads), a histogram denoting read pileup, and the variation calls (VCF). Freebayes is able to identify a SNP at an alternate allelic frequency of 15%: The reference is A (85% of the reads) the alternate allele is T (15% of the reads).

### TrainQual RS

Below is the table of the number of indels (out of 25) that Freebayes was able to report for each chromosome (with a given ploidy). The right hand column is the set of total SNPs/indels in each call of Freebayes minus the true positives, resulting in the number of false positives. These false positives are artifacts of the alignment and often have much lower quality scores relative to the true positives that were incorporated.

Chromosome (ploidy)	True Positive	False Positive
chrI (1N)	25	127
chrII (2N)	25	427
chrIII (3N)	18	663
chrIV (4N)	19	1002
chrV (5N)	24	1350
chrVI (6N)	17	1771

**Table 8:** The number of indels that were able to be identified in the chromosome with a given ploidy (left column) split into true positive or incorporated (middle column), and false positive right column.

### Sanger Sequencing Verifications

Sanger sequencing verifications were used for three reasons during the data analysis. First, they were used to confirm parental homozygous SNPs/indels where the different alignments and variation callers shared a consensus with the variation. Since there are several hundred parental variations, a subset were selected at random to be verified, which yielded the expected results of the parental strain containing the variations. Sanger verification was also used to identify the true underlying sequence when the variation callers and aligners could not come to an agreement on what the true sequence was. Examples of this include dinucleotide SNPs, multiple SNPs in a short region, and numerous insertions and deletions in a single readlength. For the most part, these were confirmed and the patterns of which alignment was correct were applied to other situations where this occurred. However, some regions could not be PCR amplified even after using two different primer sets, and these regions were determined to be sub-telomeric, repeat regions, or non-unique regions and the variations of interest were considered to be artifacts of the alignments. Finally, Sanger sequencing was used to verify the strain-unique variations in the evolved progeny, which are summarized in a table below.

Strain (HXT amplification)	Strain/SNP location	Annotation	Syn/Non-Syn	Amino Acid Changes	Allele Frequency
<b>B2 + HXT6/7</b>	B2_chrXIII:746,160	DFG5 promoter	N/A		1/2
<b>F12</b>	F12_chrl:103,998	LTE1	Non-Syn.	S 626 C	1/2
	F12_chrIV:112,896	SNF3	Non-Syn.	G 439 E	1/2
<b>F2 + HXT6/7</b>	F2_chrX:722,854	PGU1	Non-Syn.	I 17 N	1/4
	F2_chrII:289,179	SCO2 promoter	N/A		1/4
	F2_chrIV:995,263	DIN7	Non-Syn.	T 90 S	1/4
	F2_chrVII:892,626	SNG1	Non-Syn.	A 507 V	1/4
	F2_chrXII:841,308	RSC2 promoter	N/A		1/4
	F2_chrXIII:646,177	SPG5	Nonsense	Q 176 -	1/4
<b>G11</b>	G11_chrXII:841,308	RSC2 promoter	N/A		1/3
	G11_chrV:465,825	SPT15	Syn.		1/3
	G11_chrVII:1,029,503	YTA7	Non-Syn.	M 710 L	1/3
	G11_chrXI:61,897	TOR2	Non-Syn.	M 488 I	1/3
<b>G2 + HXT6/7</b>	G2_chrIV:609,668	PDC2	Non-Syn.	V 138 I	1/4
	G2_chrVI:133,989	VTC2	Non-Syn.	A 729 S	1/4
	G2_chrVII:1,023,892	YGR266W	Non-Syn.	A 411 T	1/4
	G2_chrXV:332,208	YSP3	Syn.		1/4
	G2_chrXVI:671,171	TFB4	Non-Syn.	Q 17 K	1/4
	G2_chrVIII:23,427	EFM1	Non-Syn.	I 550 L	(16%/84%)
<b>A8 + HXT6/7</b>	A8_chrVII:852082	PBP1	Non-Syn.	F 380 Y	1/4
	A8_chrXI:103753	FAS1	Non-Syn.	E 1026 D	1/4
	A8_chrVIII:375275	SPL2 Promoter	N/A		1/4
<b>D9</b>					

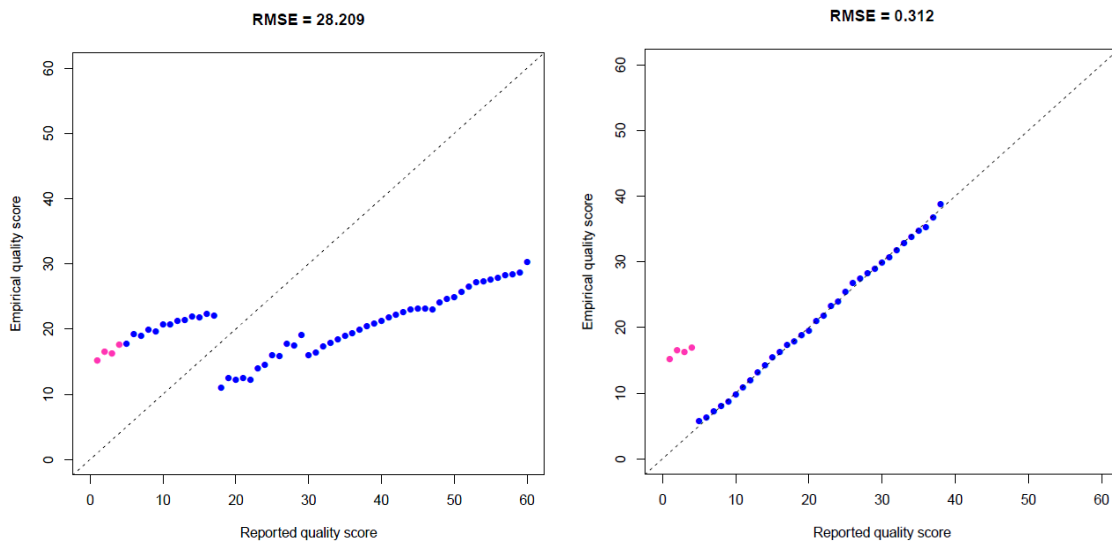
**Table 9:** The final list of strain-unique SNPs identified in the evolved strains, with their location, the

annotation they occur in, whether they are synonymous or non-synonymous, the amino acid change, and the frequency at which they are present in the reads (allelic frequency).

### GATK

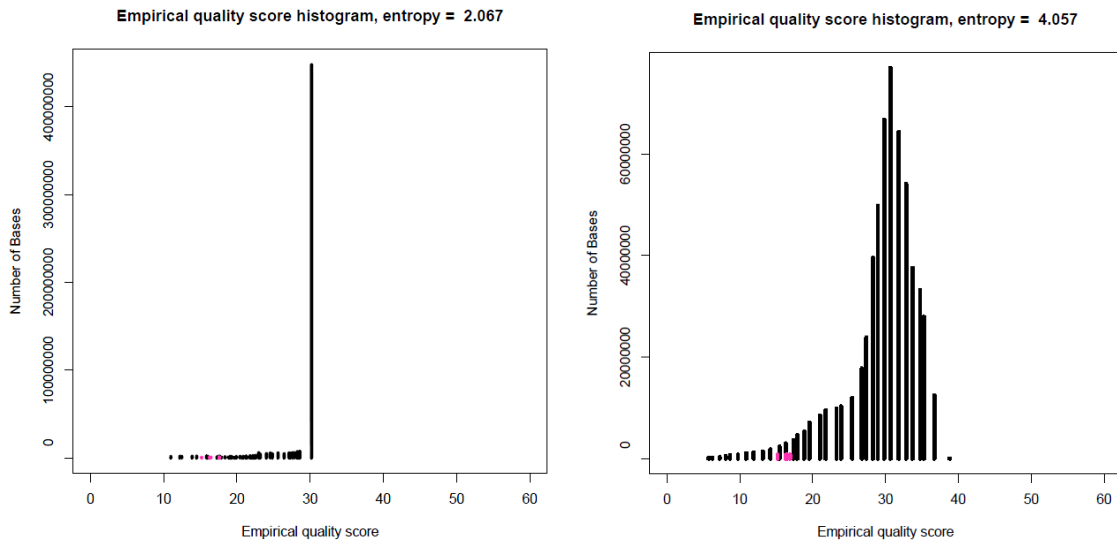
Below are the plots that result from the AnalyzeCovariates on the pre-recalibration as well as the post-recalibration Covariates output. The plots produced are of the empirical vs. reported QV score, as well as the empirical QV distribution, and the reported QV distribution. The plots on the left indicate the pre-recalibration output, and the right correspond to the post-recalibration output. All of the graphs are from the recalibration of the 6040 (parental strain) but similar plots exist for all of the 8 strains (Supplemental).

### Empirical vs. Actual Quality Score



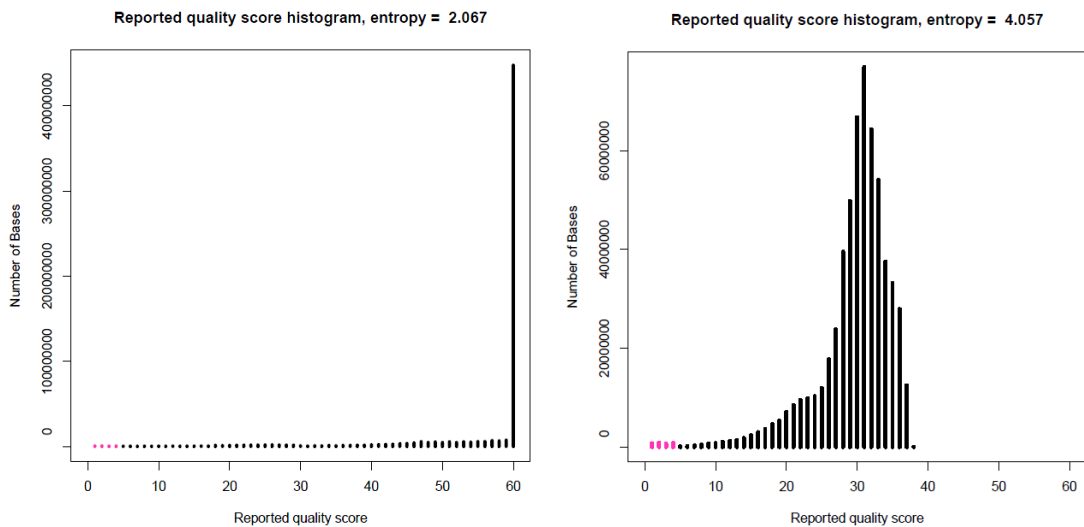
**Figure 8:** The empirical vs. actual quality scores represented in the aligned BWA reads for the strain 6040 before BQSR (left) and after BQSR (right). The scale is from 0-60 before BQSR because of the SOLiD quality score scale, and is changed to the Illumina quality score scale post-BQSR.

## Empirical Quality Score Distribution



**Figure 9:** The empirical quality score distribution represented in the aligned BWA reads for the strain 6040 before BQSR (left) and after BQSR (right).

## Reported Quality Score Distribution



**Figure 10:** The actual quality score distribution represented in the aligned BWA reads for the strain 6040 before BQSR (left) and after BQSR (right). The scale is from 0-60 before BQSR because of the SOLiD quality score scale, and is changed to the Illumina quality score scale post-BQSR.

Below is the table of SNP/indel counts for each strain pre/post-recalibration. The unique variations are SNPs in a given strain that didn't occur in at least 4/8 strains, the difference is the number of variations pre-BQSR minus the number of variations post-BQSR, and the Unique-diff is the number of strain-unique variations in the pre-BQSR minus the strain-unique variations in the post-BQSR.

	BWA-GATK	unique	BWA.BQS R-GATK	Unique	Difference	Unique- diff
6040	1848	642	1539	388	309	254
A8	1573	388	1324	186	249	202
B2	1799	639	1367	258	432	381
D9	1516	316	1353	187	163	129
F12	1815	632	1446	310	369	322
F2	1629	377	1487	279	142	98
G11	1613	332	1414	233	199	99
G2	1585	376	1405	196	180	180
Intersect (4/8 req)	1288		1227			

**Table 10:** The SNP/indel counts for each strain pre/post-BQSR. The total SNPs/indels for each strain, the unique for each strain, and the difference between pre-BQSR and post-BQSR total and unique variations.

## Discussion

### *SOLiD QualityScore Preprocessing and Analysis*

Upon seeing the low quality score trends for each of the strains, a prefiltering tactic was considered. However, upon contacting Ariel Sasson, the developer of the QV preprocessing and analysis tools at Rutgers, she discouraged the use of preprocessing tools on a resequencing dataset. Instead, the need for preprocessing is largely important when *de novo* assembly is the end result, since *de novo* assembly is highly sensitive to sequencing errors. Whereas in resequencing, when a very close relative to the strain being sequenced is already well annotated and available, alignment will mitigate the prefiltering step by not aligning the reads that have poor quality or contain sequencing errors. This is why there is a strong correlation between the number of reads passing a filter and the number of reads that get aligned.

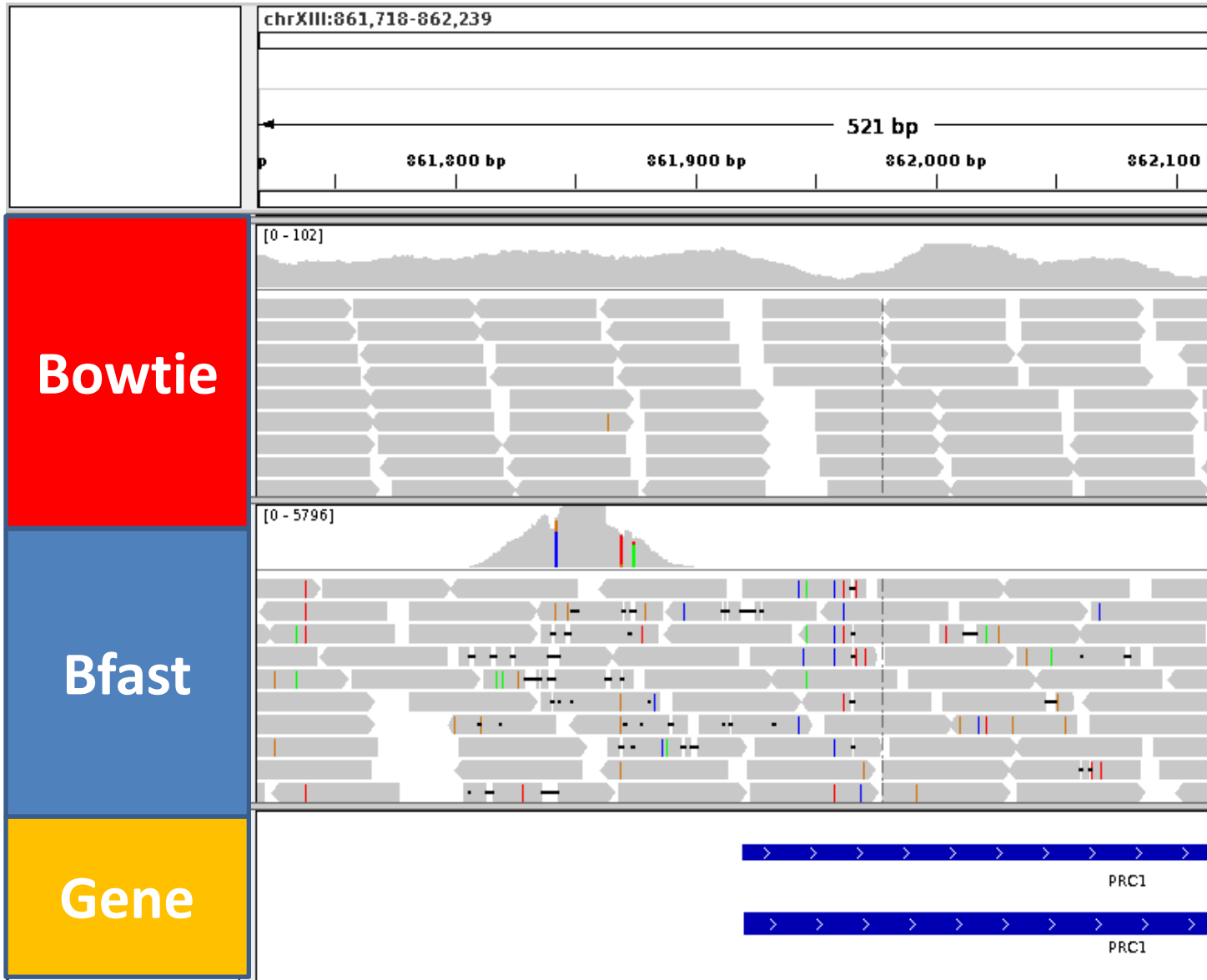
### *Alignment*

Numerous alignments were used in the data analysis of this project, each with their own pros and cons. However, altogether the alignments allowed for true identification of what the underlying sequence is. Below are examples of the merits of using multiple different alignments.

### *Bfast Artifacts*

As mentioned in the Quality Score preprocessing, reads containing sequencing errors and reads with poor quality were submitted to the alignment software. Since Bfast is able to map these low quality reads, the runtime of a Bfast alignment per strain was much longer than the other alignment software. Also, in the Bfast alignments, which allowed for the mapping of more reads and poor quality

reads, large read pileups around locations of low sequence identity were common. Below is an example of a common spike in coverage seen in the Bfast alignment, but not in the other alignments produced by the other software (Bowtie below and BWA as well).



**Figure 11:** An IGV snapshot of the Bowtie and Bfast read alignments, as well as a gene track showing the gene PRC1. This is an example of an artifact of Bfast’s ability to align reads loosely over regions of low sequence identity, resulting in what appears to be a copy number variation. This is not supported by the Bowtie alignment at all.

### Bowtie non-gapped alignment

Bowtie’s implementation provides an ungapped alignment, and also restricts the number of mismatches to 1-2 per read. Together, these two attributes make Bowtie the “Ultra-fast” aligner that can map the reads much faster than the other mapping software available. However, in the event of a

true indel, bowtie is unable to map any reads that span this type of variation. Since insertions and deletions are not allowed, a true indel in the reads will cause Bowtie to map only reads where the end of the read spans the indel. An explanation for this is that if an insertion or a deletion was in the middle of a read mapped without allowing gaps, then all of the bases downstream of the indel would be counted as mismatches—and since Bowtie is restricted to only 1-2 mismatches, only reads that have the indel at the end of the read can map and identify the indel incorrectly as a SNP or a mismatch relative to the reference. Below is an example of this, with another aligner (Bfast) that provides a gapped alignment exposing the true underlying sequence as containing two distinct deletions roughly 50 basepairs apart.



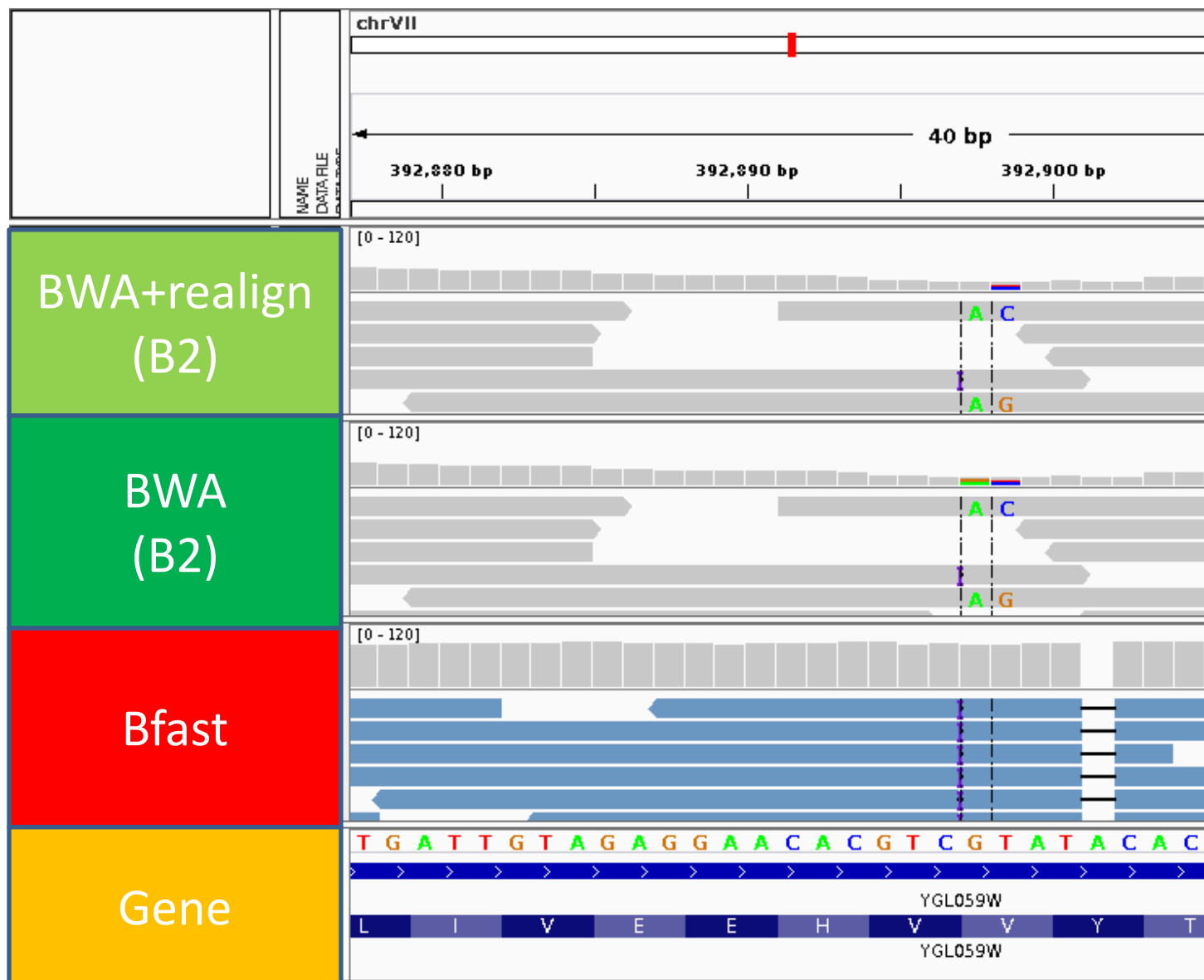
**Figure 12:** An IGV snapshot of the Bowtie and Bfast read alignments, as well as a gene track with the reference sequence bases along the bottom. This is an example of a merit of Bfast over Bowtie, due to



the fact that Bfast can align over small indels with its gapped alignment, whereas Bowtie fails to align any reads over the indels with the exception of the ends of reads thus producing false SNPs.

### BWA multiple indel confusion

BWA emerges as an all-encompassing aligner that can handle multiple mismatches per read, provide a gapped alignment, and handle mate pairing effectively. It was not the first choice of aligner, due to the fact that at the time of acquiring the data, BWA was outperformed by both Bowtie and Bfast. However, a later release (April 2011) became a viable option later on in the data analysis when Bowtie and Bfast together still encountered roadblocks. Despite being all-encompassing, BWA still has its pitfalls in the form of true identification of some indels—most notably when both an insertion and deletion occur within the span of one read. This pitfall is the reason why consensus through multiple alignments on any variations identified was still necessary. Below is an example of a scenario where Bfast is still useful in identification of indels when the BWA alignment failed.



**Figure 13:** An IGV snapshot of the BWA (B2, before and after realignment) and Bfast read alignments, as well as a gene track with the reference sequence bases along the bottom. A confirmed insertion and deletion are present, but BWA is unable to correctly identify the insertion and deletion when they are so close to each other—unlike the Bfast alignment which correctly identifies both.

### *Copy Number Variation*

The copy number variation identified prior to the sequencing was successfully converted into a file format that can be visualized alongside the alignments and the read pileups in IGV. The result was that for all strains, any aneuploidy identified via the aCGH data was also confirmed in the read pileup data. This was a more of a confirmation of previously identified strain characteristics than a new analysis.

## *SNP/Indel calling*

### VARiD

Varid was used as a variation caller to identify both parental variations common to all strains, as well as strain-unique diploid variations. VARiD was a looser variation caller than Samtools Mpileup, allowing it to identify more variations than Samtools with more information provided about each variant including number of supporting reads, allelic frequency, p-value, and others. The increased variations that Samtools missed largely ended up being false positives that were artifacts of the alignment. However, since it is limited to diploid cases, Varid was not able to identify any of the strain-unique variations in the evolved tetraploids.

### Freebayes

In the initial Freebayes releases, the number of variations per strain was between 10,000 and 200,000, where the number of false positives was too large to deal with through previously described visualizations and Sanger confirmations. However, the latest release greatly decreased the number of false positives that were artifacts of the alignment, and from this more reasonable number strain-unique variations were identified in the evolved tetraploids. Also, it is important to note that as the ploidy is increased, so is the number of false positives reported. This explains why the parental (run with a ploidy of  $N=1$ ) has fewer total variations than the intersection of 4/8 (since there are at least 4 strains run with a ploidy of  $N=4$ ). It is also important to note that Freebayes is still an unpublished variation caller, and as the software keeps being updated, these numbers may be subject to change.

### Parental

The final set of parental variations includes about 600 SNPs/indels, which is reasonable considering how evolutionarily close the samples being sequenced were to the reference strain. In this identification of true positives versus false positives, certain liberties were taken when considering a variation to be parental, and the parental set is not entirely complete. The liberties taken include manual visualization to verify the variations, and in cases where all alignments couldn't reach a consensus Sanger verification was used. However, several of the indels included in the parental set may just be due to sequencing error surrounding homopolymers, and if the primary objective of this project was to collect all the parental variations, then a more thorough analysis would need to be completed on several of the indels called. The parental set is also not entirely complete, because it was generated using an intersection of different variant callers and alignments. Thus, if a true parental variation was found in one alignment but not the other, then the variation would not be included (note that this is not the practice used for identification of strain-unique variations). This is not of much concern since the parental set was primarily used for GATK's BQSR, as well as being used to compare the individual strains against to look for strain-unique variations.

## *Structural Variation Calling*

Structural variations can be identified in sequencing datasets by leveraging the insert size between aligned mate paired reads. Since there is a known insert size between two reads, when they align if they align with a much smaller insert size then this can be indicative of a large deletion, a much larger insert size would indicate an insertion, in reverse order would be probable cause for an inversion,

and on two different chromosomes could mean a chromosomal translocation. However, this dataset is limited at identifying these structural variations for the following reasons. First, in order to determine what is “much larger” or “much smaller,” the algorithm will look for reads with an insert size outside of two-to-three standard deviations. In this dataset, there was a very large standard deviation, which confounds the ability to look for these insertions or deletions. This is shown in the reads with large standard deviation, and also exhibited in the bioanalyzer traces with broad peaks instead of sharp spikes around the target insert size (Supplemental data). Second, the ability to call chromosomal translocations hinges on the ability of the insert size to span across a transposable element. In yeast, chromosomal translocations and recombinations occur in transposable elements, and if the insert size isn't large enough to provide unique locations for each pair of reads on either side, then these structural variations won't be able to be discovered. Finally, a large part of the analysis in this project hinges on identification of variations at a higher ploidy, and to date there is no structural variation software that can handle this. Instead, the structural variation software looks for variations represented in the majority of the reads at a given position, which is not what would be expected for a chromosomal structural variant in one chromosome of an evolved tetraploid. Altogether, the structural variation identification for this data is incomplete, and hinges on the release of new higher/variable ploidy software, as well as another whole genome sequencing run that incorporates an insert size that is larger than the transposable elements (roughly 6-10 kb).

## *Benchmarking*

### Custom Perl Scripts

From the SNP incorporation into the reads at a given percentage, it is apparent that VARiD cannot call SNPs outside of a set allelic frequency. After looking into the VARiD algorithm, there is a preprocessing step that filters out any SNPs that do not fall within a certain range of allelic frequencies. Thus, SNPs can be considered to be diploid, and thus reported as heterozygous if they fall within the expected 50%:50%—where the alternate allele is present in half of the reads and the other half of the reads support the reference—plus or minus a standard deviation, which in this case is a fixed 18%. So, positions that have a SNP in 1/3 of the reads, or supported by 33% of the reads (as expected by a ploidy of  $N=3$ ), will be reported. However, as the ploidy increases, and the expected allelic frequency present in the reads is 1/4, VARiD will not report the SNPs. However, since Freebayes has the ability to look for SNPs present in a lower proportion in the reads (as expected for higher ploidy SNPs), these SNPs will be reported. Altogether, VARiD serves as a variant caller for SNPs at a ploidy of  $2N$ , with the ability to call some SNPs at a ploidy of  $3N$  (as long as they fall within the allowed distribution), and Freebayes serves as a variant caller that can identify SNPs at a higher ploidy.

### TrainQual RS

The indel incorporation into the reads at increasing ploidies was successful in showing that Freebayes has the ability to identify indels at a higher ploidy. The lower numbers of indels reported out of the 25 incorporated in chromosomes 3, 4, and 5 are due to artifacts in the alignment—where reads were not able to map to the location that the indel was supposed to be incorporated. These artifacts were the result of different lengths of telomeric repeats in these select chromosomes, and were not influenced by the incorporation of indels.

Although Freebayes was able to identify these indels at a higher ploidy, the increase in ploidy setting on Freebayes comes at a cost. The number of false positives increases constantly—by about 200-400—for each additional ploidy increase. This creates problems when analyzing the strains of higher ploidy due to the number of false positives that have to be removed in order to attain the true higher ploidy SNPs.

### *Sanger Sequencing Verifications*

Sanger sequencing has given insights into the true sequence identity throughout the entire analysis. Sanger was used in situations to confirm homozygous parental SNPs and indels that were easily identified with consensus in the alignments and the variant callers, in cases where the alignments and variant callers didn't agree with one another and the true identity of the sequence was unable to be conclusive from the sequencing dataset, and to identify the strain unique variations in the progeny; variations that could be causal for their survival in the evolution experiment. No strain-unique indels were identified in this experiment, and the candidate strain-unique indels were all artifacts of the alignment due to homopolymers.

### *GATK*

GATK is designed primarily for the 1000 Genomes Project implemented at the BROAD, and therefore makes assumptions based on the type of datasets being analyzed. Those assumptions are the following: (1) low coverage reads across the genome (read depth < 30), (2) large genome size, (3) Illumina sequencing platform, (4) well annotated loci of known SNP/indel sites. For this project, none of these assumptions hold true: there is deep coverage across the entire genome with read depth ~50-75, the genome is only 12.6 Mb (as opposed to the ~3.2 Gb human genome), the sequencing was done on a SOLiD platform in colorspace, and there is no dbSNP database containing known sites of SNPs and indels. Due to these factors, the efficacy of using the GATK tools on this data as a primary analysis tool is not sound, but using it as a secondary pipeline and as a model for a variation calling analysis pipeline was useful. The recalibration of the alignments to better represent the true error probability meant to be incorporated by the phred quality scores may be statistically sound when the known sites to recalibrate on is a list of over 200,000, but whether or not it is feasible using ~600 known sites identified in this project has not been calculated. The effect of the BQSR on variation calling is a clear decrease in the total number of SNPs/indels post-recalibration compared to pre-recalibration, but the effect on the strain-unique and parental variations has not been investigated.

## **Conclusion**

Through this project new insights into the variation calling in higher ploidy have been revealed, novel SNPs have been identified in evolved progeny—and may prove to be causal for their adaptation to the selective pressure—and array technology has been parallelized to deep sequencing technology allowing for confirmations of specific aneuploid chromosomes and segmental amplifications. The initial goal of the project was to analyze these strains and perhaps draw conclusions about the differences in how the starting ploidy affected a strain's evolutionary path. However, a definitive answer as to the entire genotype of these evolved strains was not reached due to limitations in the sequencing dataset, limitations in the software available, and limitations on the number of generations that the strains

evolved through. These limitations gave rise to new computational solutions including *in silico* generation of reads and modeling of polyploidy/aneuploidy to test the efficacy of current software in alignment and variation calling. Future work on this project includes modeling of structural variations, testing new structural variation software, and RNA-seq. The goal of the RNA-seq is to identify transcriptional differences between the strains, to quantify transcription levels of specific aneuploid chromosomes relative to their euploid counterparts, and possibly gain insight into allele-specific expression of genes based on locations where SNPs were present in one copy of the chromosome.

## Best Practice for Sequencing Analysis

Below are detailed some of the sequencing analysis practices that I have acquired through large amounts of trial and error in this data analysis pipeline. This section contains some hints and directions about data analysis practices that will save time and improve efficiency.

### Alignments

The usage of multiple alignments has proven to be very effective in establishing the true sequence identity. Tweaking the alignment parameters will produce alignments specific to the situation—including loosening of the insert size restrictions, allowing for only unique mapping, providing both a loose alignment and a strict alignment based on the number of errors allowed while mapping, and other settings as well. Not mentioned in the methods above, Samtools is an essential tool used for converting between the SAM (standard text alignment format) and the BAM (standard binary alignment format), as well as other tools including sorting and indexing that are required for downstream tool usage. Also, for downstream software usage, including @RG (read group) tags is important to do right after the alignment because most post-processing steps and downstream tools will require it.

### File Formats

Restrictions on file formats are not always followed by the developers of software, and file format issues can confound data analysis. One example of a file format issue in this project is the VCF (Variation Calling Format) file. Different variation calling software will use different output for this file format—although they do remain within the tab delimited one-variant-per-line format. The differences of a single column in the file can be subtle, but when pushing these files through downstream visualizations, errors can occur. For example, the VCF file produced by the GATK is not compatible with the visualizer IGV.

### Organization

Organization of the data becomes very important in the data analysis for any project—be it single sample or multiple sample. One simple way of organizing things is to use a cascade of directories with the following levels:

1. Sample (Strain)
  - a. Fastq directory—contains quality analysis and raw fastq
  - b. Alignment with BWA
    - i. Variation calling with VARiD

- ii. Variation calling with Freebayes
  - c. Alignment with Bfast
    - i. Variation calling with VARiD
    - ii. Variation calling with Freebayes
- 2. Genome
  - a. Fasta directory—contains the reference sequence and annotations
  - b. BWA\_index—contains the indexed reference for all BWA alignments
  - c. Bfast\_index—contains the indexed reference for all Bfast alignments

This organization is complemented well by file-naming practices, where the name of the file includes information about the sample, the aligner, the variation calling tools, etc. The last note about organization of data analysis is a current README file, where information regarding the contents of a directory and how those contents came to be (command line arguments, order of tool usage, etc.).

## Other Work

In addition to this project, I have collaborations with three other projects in the field of bioinformatics. One collaboration I have with Jamie Prior of the Copley lab includes an *E. coli* gene amplification project that is looking to identify possible SNPs in individual copies of the amplifications based on an evolutionary model. This project entails read generation, amplification modeling, SNP incorporation, and quality score analysis. Another collaboration with Max Cohen of the Han lab entails implementation and running of a variation calling analysis pipeline on a *C. elegans* shotgun sequencing dataset. This project allows for experience in variation calling in an Illumina dataset on whole genome sequencing data from an organism that is more complex and has a larger genome than *S. cerevisiae*. Lastly, a collaboration with Alex Poole of the Dowell lab has proven to be quite beneficial to the both of us through the development and testing of a read generator. My work with identification of certain patterns in the dataset is the basis for work Alex is pursuing in automatically detecting patterns across multiple aligners to identify true variations.

## References

1. Pavelka N, Rancati G, Li R. **Dr Jekyll and Mr Hyde: role of aneuploidy in cellular adaptation and cancer.** *Curr Opin Cell Biol.* 2010;22:809–815. doi: 10.1016/j.ceb.2010.06.003.
2. Torres, E.M., B.R. Williams, A. Amon. 2008. **Aneuploidy: cells losing their balance.** *Genetics.* 179:737–746. doi:10.1534/genetics.108.090878
3. Duesberg P, Li R, Fabarius A, Hehlmann R. **The chromosomal basis of cancer.** *Cell Oncol.* 2005;27:293–318.
4. **BWA:** Li H. and Durbin R. (2010) Fast and accurate long-read alignment with Burrows-Wheeler Transform. *Bioinformatics*, Epub. [PMID: [20080505](https://pubmed.ncbi.nlm.nih.gov/20080505/)]

5. **VARiD**: Adrian V. Dalca; Stephen M. Rumble; Samuel Levy; Michael Brudno. VARiD: A variation detection framework for color-space and letter-space platforms. *Bioinformatics* 2010 26: i343-i349.
6. **Freebayes**: Marth Lab, Boston College, <http://bioinformatics.bc.edu/marthlab/FreeBayes>
7. **Samtools**: Li H.\*, Handsaker B.\*, Wysoker A., Fennell T., Ruan J., Homer N., Marth G., Abecasis G., Durbin R. and 1000 Genome Project Data Processing Subgroup (2009) The Sequence alignment/map (SAM) format and SAMtools. *Bioinformatics*, 25, 2078-9.
8. **Bfast**: Homer N, Merriman B, Nelson SF. Local alignment of two-base encoded DNA sequence. *BMC Bioinformatics*. 2009 Jun 9;10(1):175. PMID: 19508732 <http://dx.doi.org/10.1186/1471-2105-10-175>
9. **Bowtie**: Langmead B, Trapnell C, Pop M, Salzberg SL. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. (2009) *Genome Biol* 10:R25.
10. **SRMA**: Nils Homer, Stanley F Nelson. Improved variant discovery through local re-alignment of short-read next-generation sequencing data using SRMA. *Genome Biology* 2010, 11:R99 doi:10.1186/gb-2010-11-10-r99. Published: 8 October 2010
11. **IGV**: James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. [Integrative Genomics Viewer](#). *Nature Biotechnology* 29, 24–26 (2011)
12. **GATK**: DePristo, M., Banks, E., Poplin, R., Garimella, K., Maguire, J., Hartl, C., Philippakis, A., del Angel, G., Rivas, M.A, Hanna, M., McKenna, A., Fennell, T. Kernytsky, A., Sivachenko, A, Cibulskis, K., Gabriel, S., Altshuler, D. and Daly, M. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*. 2011 Apr; 43(5):491-498.
13. **GATK**: McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res*. 2010 Sep; 20(9):1297-303. Epub 2010 Jul 19.
14. **SOLiD QV Analysis and Filtering**: Sasson A., Michael T.P. Filtering error from SOLiD output. *Bioinformatics* 2010;26:849-850.
15. **NovoAlignCS**: Produced by Novocraft. <<http://www.novocraft.com/main/index.php>>
16. "Next Generation Sequencing," BBSRC - Biotechnology and Biological Sciences Research Council (BBSRC). BBSRC. Web. 08 Apr. 2012. <<http://www.bbsrc.ac.uk/web/FILES/Reviews/1102-next-generation-sequencing.pdf>>.



17. Enrique Amaya, Martin F Offield, Robert M Grainger. **Frog genetics: *Xenopus tropicalis* jumps into the future.** *Trends in Genetics*, Volume 14, Issue 7, 1 July 1998, Pages 253-255, ISSN 0168-9525, 10.1016/S0168-9525(98)01506-6.  
(<http://www.sciencedirect.com/science/article/pii/S0168952598015066>)
18. **BreakDancer:** Ken Chen, John W Wallis, Michael D McLellan, David E Larson, Joelle M Kalicki, Craig S Pohl, Sean D McGrath, Michael C Wendl, Qunyuan Zhang, Devin P Locke, Xiaoqi Shi, Robert S Fulton, Timothy J Ley, Richard K Wilson, Li Ding & Elaine R Mardis. BreakDancer: an algorithm for high-resolution mapping of genomic structural variation *Nature Methods* 6, 677 - 681 (2009)Published online: 9 August 2009 | doi:10.1038/nmeth.1363
19. **BEDtools:** Quinlan AR and Hall IM, 2010. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*. 26, 6, pp. 841–842.