AN ALGORITHM FOR STRUCTURAL VARIANT DETECTION WITH THIRD GENERATION SEQUENCING

BY

HUI-JOU CHOU

A thesis submitted to the

Graduate School—Camden

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Scientific Computing

Written under the direction of

Dr. Andrey Grigoriev

and approved by

Dr. Andrey Grigoriev

Dr. Sunil Shende

Dr. Jean-Camille Birget

Camden, New Jersey

May 2017

THESIS ABSTRACT

An Algorithm for Structural Variant Detection with Third Generation Sequencing

by

HUI-JOU CHOU

Thesis Director:

Dr. Andrey Grigoriev

Structural variations are large variations in chromosome sturcture, including deletions, duplications, insertions, inversions, and translocations. Many studies have shown the importance of structural variants in genetic diversity and disease susceptibility. Most structural variants are identified using next-generation sequencing paired-end or mate-pair reads. Primarily due to short read lengths, structural variant detection methods for NGS data tend to have low sensitivity and precision. Long-read sequencing technologies generate continuous long reads which can span large genomic regions, especially critical for insertions, and be mapped with high accuracy.

We developed an algorithm to predict structural variants using long reads. We applied multiple signals including split-read, alignment mismatching and read-depth to identify structural variants in PacBio whole genome sequencing and found our results to be comparable to other structural variant detection algorithms. Our approach provides an effective way of detecting structural variants in long read datasets that can compensate for and complement the limitations and benefits of short reads.

Acknowledgements

I would like to express my gratitude towards my adviser, Prof. Andrey Grigoriev, who patiently supported me throughout my research. He guided me to this research area where I have really learned a lot. I would also like to sincerely thank our lab pos-doc Sean Smith, who gave me a lot of idea about the algorithm design and discussed with me when I had questions so that I can keep going to improve my algorithm. A special thanks to my defense committee, Prof. Sunil Shende, and Prof. Jean-Camille Birget for their interests in my work and giving me their treasured ideas.

I would like to acknowledge our lab member Spyros Karaiskos, whose suggestion about the content of my presentation slides and correcting the English grammar for me; Joe Kawash, who provided the valuable idea on the machine learning; Lingyu Guan, whose immediately support really encouraging me on the defense date.

Finally, I would especially like to thank the support and love of my family. They all kept me going, and this thesis would not have been possible without them.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

SV	**S**tructual **V**ariant
SNV	**S**ingle **N**ucleotide **V**ariant
NGS	**N**ext **G**eneration **S**equencing
WGS	**W**hole **G**enome **S**equencing
TGS	**T**hird **G**eneration **S**equencing
SMRT	**S**ingle **M**olecule **R**eal **T**ime
SAM	**S**equence **A**lignment **M**ap
IGV	**I**nterative **G**enomics **V**iewer
GROM	**G**enome **R**earrangement **O**mni **M**apper
RD	**R**ead **D**epth
SR	**S**plit **R**ead

Chapter 1

Introduction

Genetic variation is the genetic differences within populations. For example, in the human population, there may be multiple variants of any given gene leading to polymorphism. On average, all human genome are 99.5% similar to any other human genomes [7]. No two humans are genetically identical. Even twins who develop from one zygote have infrequent genetic differences because of mutations during development and gene copy-number variation [2]. At the gene level, the variation can be identified as a single nucleotide variant (SNV); while at the chromosome level, the variation can be identified as a structural variant (SV). The study of genetic variation has been applied in the evolutionary field and medical field. From the evolutionary view, it can help scientists better understand how different human groups are biologically related to one another. From the medical view, human genetic variation may be the reason of why many diseases happen.

Although smaller-scale forms of genetic variation such as single nucleotide polymorphisms (SNPs) are more common, SVs have greater functional potential due to their larger size and higher likelihood to alter gene structure and dosage. Structural variant consists of many kinds of variation in the genome of the species, usually including deletions, duplications, insertions, inversions and translocations (Figure 1.1). Generally, SVs are defined as a region of DNA larger than 50 bp in size. Many SVs have been implicated in human health with associated phenotypes ranging from disabilities to obesity, cancers and other diseases [17]. For example, Charcot-Marie Tooth (CMT) disease was the first autosomal dominant disease associated with a gene dosage effect due to an inherited DNA rearrangement in 1991. The disease phenotype results from having three copies of the normal gene [10].

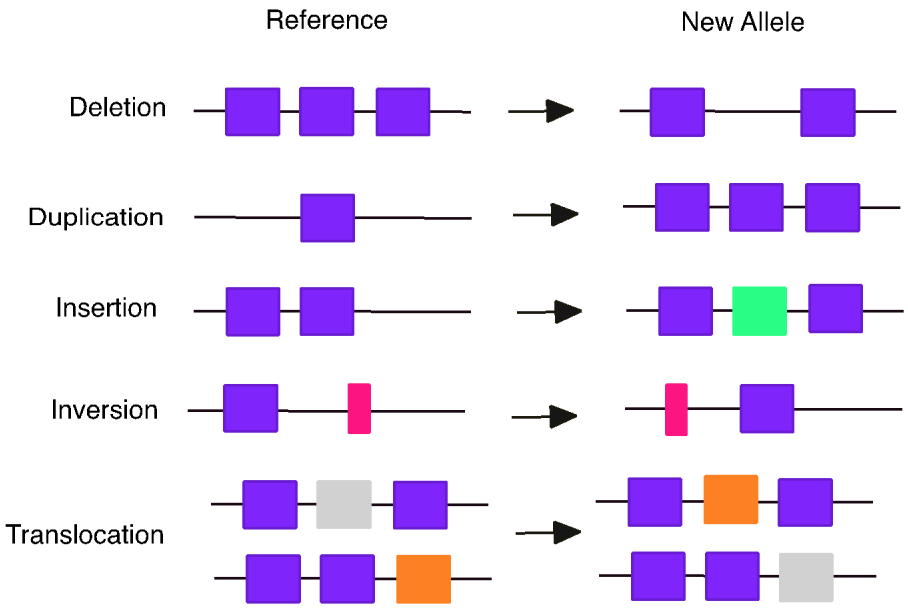


Figure 1.1: Structural variant classification

Due to the biological impact of SVs, we need technologies to call SVs. The technology advances in next-generation sequencing (NGS), or called second-generation sequencing, have become more prevalent across a large number of species. It allows us to perform whole genome sequencing (WGS), so that we can study SVs by mapping the reads to the reference genome. The commercialization of high-throughput technologies including Roche/454 pyrosequencing in 2005 and Illumina sequencing in 2007 have been applied to sequence many new genome along with widespread resequencing effort to analyze genomic diversity. Basically, SV detection methods require comparing the sample DNA sequences with a reference genome, known as mapping-based method [5], which identify SV candidates from abnormally mapped reads. Such mapped reads have different features including pair-end, single-end, mate-pair, soft-clipped and so on. SV callers use these features to detect different types of variants. Second-generation sequencing has improved large-scale analyses of single nucleotide and small variants, and a number of scientists have developed algorithms to analyze structural variation using NGS dataset [14, 6, 3], however, the limitations from the nature of NGS technologies itself is difficult to be overcome.

One of the limitations is the read length. NGS generates shorter reads ranging from dozens to hundreds of base pairs (bp), it will result in less confident mappings in the repeat region and cause the SV callers to fail to detect a confident matching breakpoint (Figure 1.2). Moreover, it is known that certain genomic regions are tough to sequence such as GC-rich regions, which are difficult to map as well. According to the mechanisms of SVs, SVs tend to occur more frequently in repetitive regions of the genome [12]. Another limitation is from artificial chimeric reads, which are formed during polymerase chain reaction (PCR) amplification step of NGS [1]. These artificial reads may be misinterpreted as formed by SVs. Therefore, these flaws could result in tens of thousands of structural variants missing, especially large size SVs.

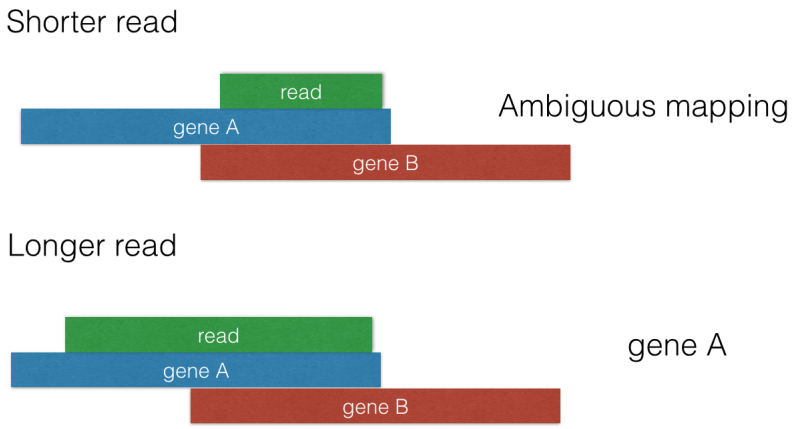


Figure 1.2: Ambiguities in read mapping

The new single-molecule sequencing technologies that produces average read lengths of more than 10,000 bp have greatly improved analysis of genomic structural variation. This technology is also known as third-generation sequencing (TGS) (Table 1.1). The most commercially established sequencer is Pacific Biosciences (PacBio) Single Molecule Real Time (SMRT) Sequencing, which was introduced in 2010 [15](Roberts, Carneiro, and Schatz 2013). Longer read length can span more repetitive elements and thus increase mapping confidence. Also, longer reads enable an improved split-read method so that deletions, inversions, insertions and other structural changes can be more easily recognized. Furthermore, TGS can produce more uniform coverage of the genome,

because it is not as sensitive to GC content as NGS, which tends to have reduced or completely absent coverage over regions [16].

Technology	Generation	Read length (bp)	Single pass error rate (%)
Illumina HiSeq 2500	2nd	125~250	0.1
PacBio RS II	3rd	$1.0–1.5 \times 10^4$ on average	13

Table 1.1: Performance comparison of sequencing platforms

In this research, we present an algorithm of integrating multiple signals including split-read alignment, read depth and alignment mismatching in an efficient parallel pipeline to identify different types of structural variants via PacBio sequencing. Our aim is to provide a method that can complement the limitation of short reads. We evaluated our algorithms performance by comparing with two recent well-known long-read SV callers: PBHoney [4] and Sniffles. We take advantage of seven different sets of high-confidence SV calls for NA12878 human genome as gold standard SV call sets to validate our results. We also evaluated the results of these three SV callers with Illumina call sets generated from GROM, which is a short-read caller developed by our lab post-doc. Our results have shown improved sensitivity in deletions, insertions and inversions discovery.

Chapter 2

Materials and Methods

2.1 Description of Data

2.1.1 BAM/SAM data format

Sequence Alignment Map (SAM) format consists of a header section and an alignment section [9]. Each alignment line has 11 mandatory fields for essential alignment information and a variable number of optional fields for aligner specific information. It typically represents the linear alignment of a segment. Table 2.1 shows an overview of the mandatory fields in the SAM format. A BAM file is the binary version of a SAM file. The goal of BAM along with BAM index file is to achieve fast retrieval of alignments without going through the whole alignments.

Table 2.1: An overview of the mandatory fields in the SAM format.

col	Field	Brief description
1	QNAME	Query template name
2	FLAG	bitwise FLAG
3	RNAME	Reference sequence name
4	POS	mapping position
5	MAPQ	mapping quality
6	CIGAR	cigar string
7	RNEXT	Ref. name of the next read
8	PNEXT	Position of the next read
9	TLEN	observed Template length
10	SEQ	segment sequence
11	QUAL	Phred-scaled base quality
12	option field	various tags

2.1.2 PacBio datasets used for this study

Three whole-genome PacBio sequencing datasets, which are 10x, 12x and 44x coverage of the NA12878 human genome, were used to test the performance of SV calling pipelines. The 10x NA12878 datasets mapped with NGM-LR aligner was downloaded from Pacbio website. The 12x NA12878 datasets was from Mount Sinai Hospital which has been mapped using BWA-SW aligner. The 44x NA12878 dataset was obtained from NCBI SRA database [13]. After we obtained raw data with fastq format, we mapped the sample reads to the human reference hg19 genome (GATK resource bundle https://software.broadinstitute.org/gatk/download/bundle) using BWA-MEM (version 0.7.15) [8]. The output bam files were input to our algorithm, PBHoney or Sniffles.

2.1.3 Illumina dataset

Raw short-read Illumina platinum WGS data for NA12878 was obtained from the Illumina website (https://www.illumina.com/platinumgenomes/). Its mapping coverage was approximately 51x. NA12878 Illumina platinum fasta files were mapped to human reference hg19 using BWA-MEM (version 0.7.15) with the -M parameter to mark shorter read splits as secondary. The output bam file was input to GROM (Genome Rearrangement Omni-Mapper), which is a comprehensive variant detection method for analysis of a wide range of variants including SNVs, indels ($<$ 50 base insertions and deletions), and SVs. Then, the generated SVs call sets with mapping quality $\geq$ 35, SV length $\geq$ 50 bp and at least two variant-supporting reads were used for validation.

2.1.4 Gold standard SV call sets

We applied three deletions call sets, two duplications sets, one insertions set and an inversion set as gold standard call sets. Deletion and Insertion benchmarks for NA12878 were obtained from Genome in a Bottle (GIAB), in which most of the calls were refined by experimental validation or other independent technologies. Deletion and duplication benchmarks for NA12878 were from the LUMPY [6] and Mills [11] papers, respectively. We also downloaded the Database of Genomic Variants Gold Standard (DGV-GS) from

dgv.tcag.ca/dgv. For the NA12878 DGV-GS benchmarks, all deletions and duplications with NA12878 tag were extracted from the DGV-GS. An inversion benchmark was obtained from Pendleton et al.(Pendleton et al. 2015)

2.2 Variant detection

In our algorithm, we analyze various features through a BAM file generated by PacBio SMRT sequencing. Our algorithm can detect all types of SVs using evidence from read depth, alignment reads mismatching and split-read alignment. The BAM file can be from NGM-LR, BWA-MEM or BWA-SW aligner. The workflow is designed for parallelization. We calculate variants in each chromosome individually, therefore it can be constructed in parallel. We output our SV call sets with at least two reads support, mapping quality threshold 35 and minimal SV length 50 bp for evaluation. Figure 2.1 summarizes our algorithm workflow.

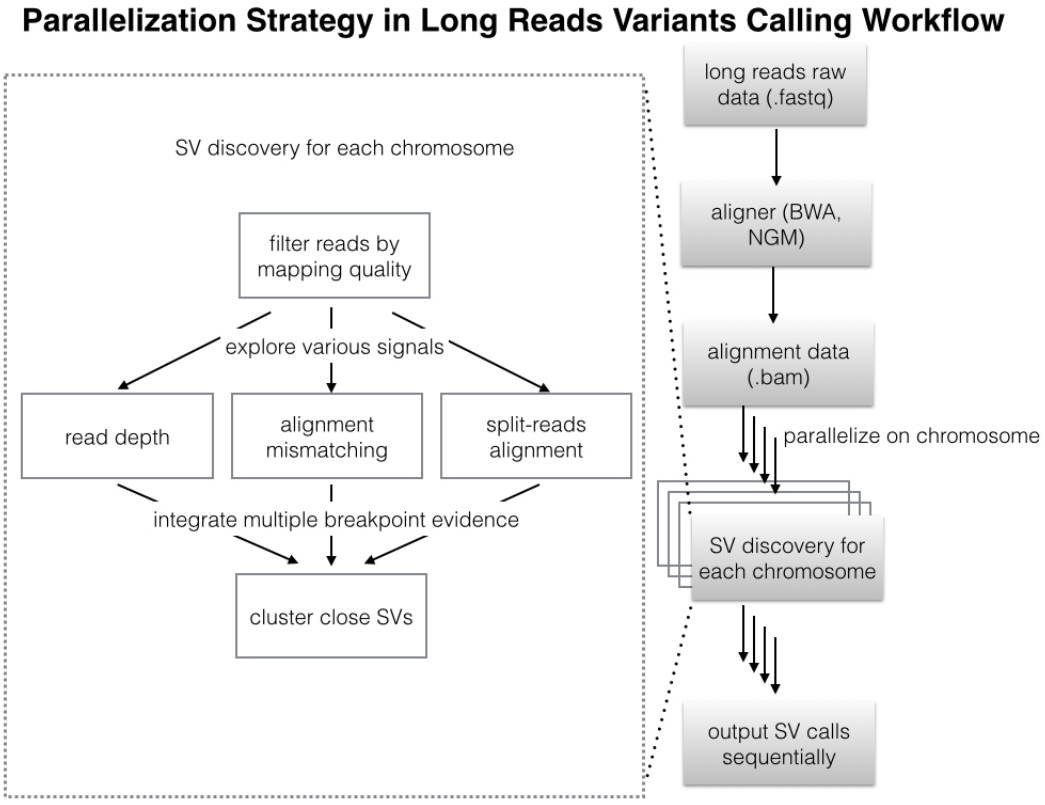


Figure 2.1: The workflow of our algorithm

2.2.1 Read depth

Read depth (RD) analysis is a method to detect deletions and duplications. By looking for genome regions that have significantly reduced or increased coverage, we can identify the potential variants. To detect such an event, we first calculate the average read depth (r) and its standard deviation (sd) from a set of windows in the dataset. Then, we estimate RD in non-overlapping intervals with fixed-size windows of 1000 bp. For an interval of consecutive windows, we call it an unusual event if the read depth mean in the region is $> r + 3 \times sd$, a potential duplication, or $< r - 3 \times sd$, a potential deletion. Next, we merge the consecutive events. More formally, a variant event is a tuple $X = \langle E, s, e \rangle$, where $X.E$ is the set of evidence type; $X.s$ and $X.e$ are start breakpoint and end breakpoint. If there are two events A, B in the set of the variants, $B.s$ is equal to $A.e$, then A and B are merged to event M, $M.s = min(A.s, B.s)$, $M.e = max(A.e, B.e)$. The merge step for deletion events and duplication events is performed separately. The variants identified by read depth analysis are annotated with RD in evidence type field.

2.2.2 Alignment reads mismatching

We use linear alignment mapping information to identify deletions and insertions, whose information is included in the CIGAR field of the alignment section. When sample reads map to the reference genome, every spot in the sample reads can agree with the reference or produce a mismatch, deletion or insertion. This information is collected in CIGAR string, and we apply it along with mapping position to compute the breakpoints of a deletion or insertion within the alignment. The minimum SV length is 50 bp. The variants found by alignment mismatching are annotated with MM in the evidence type field.

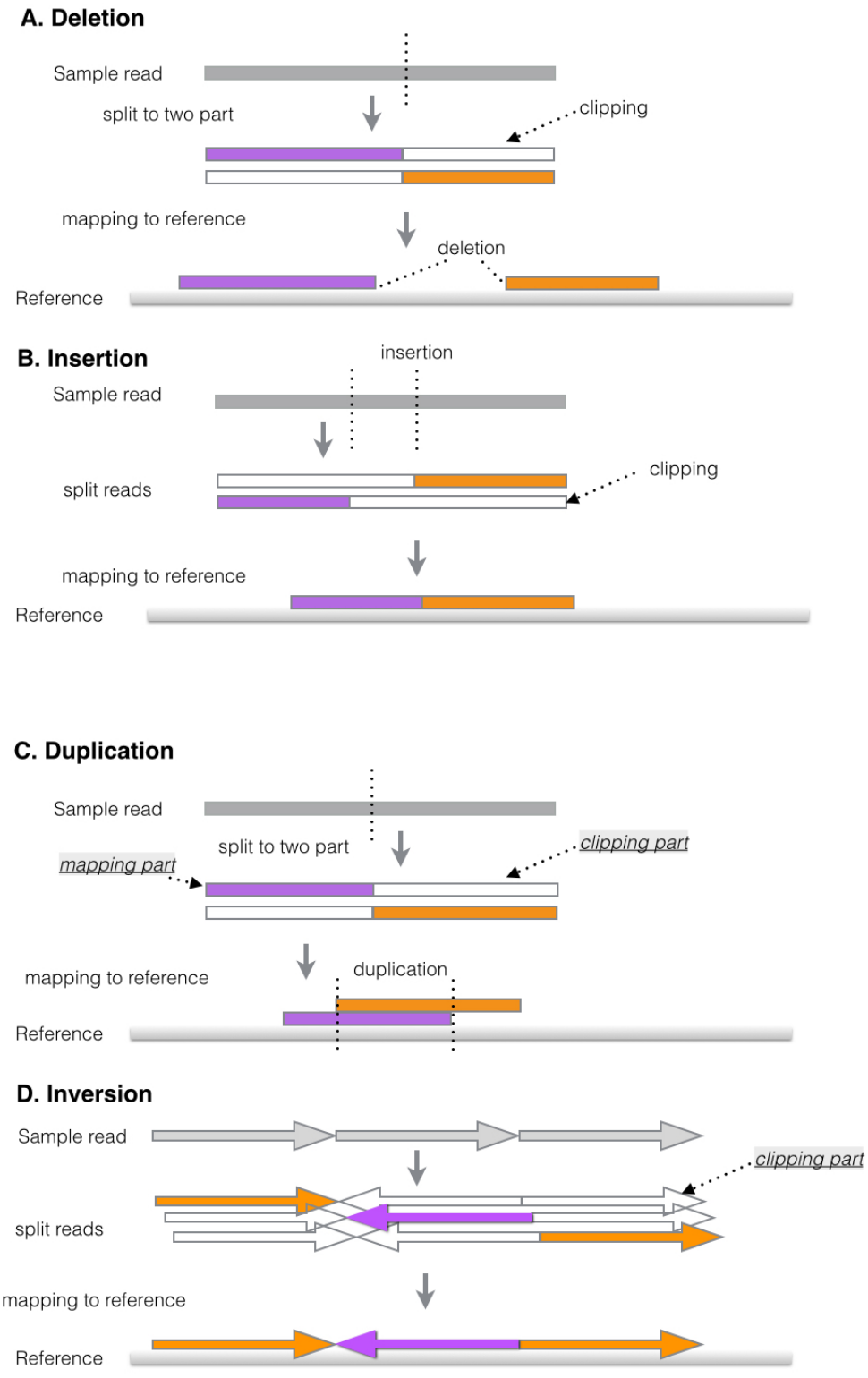
2.2.3 Split-read alignment

A split-read alignment is a single read X that does not contiguously align to the reference genome. Therefore, X contains a set of linear alignments ($X = x_1, x_2, ..x_n.$).

We call these linear alignments subreads in X. All the linear alignments in a split-read alignment have same the QNAME in the SAM records. If a linear alignment belongs to a part of a split-read alignment, there will be a SA tag in the option field recording the supplementary alignments information (the other parts of the split-read) of this linear alignment. We apply the SA tag record with the representative alignment record to identify all types of SVs.

Assuming each subread is denoted as $x_i = \langle c, s, e, o, rc, lc, ql \rangle$, where c represents chromosome; s and e are the start mapping position and end mapping position; o is orientation; rc and lc are right clipping length and left clipping length; ql is sample read mapped length. We consider mapping location, orientation and chromosome to infer breakpoints and SV type. When the orientations and chromosome match $x_i.o = x_{i+1}.o$, $x_i.c = x_{i+1}.c$, the event could be a deletion, duplication or insertion. If $x_i.e < x_{i+1}.s - min\_sv\_length$, the default min_sv_length is 50 bp, it indicates a gap caused by a deletion; if $x_i.e > x_{i+1}.s + min\_sv\_legnth$, it indicates a duplication; if $x_i.e$ and $x_{i+1}.s$ are close within 10 bp , it could be a potential insertion location. When the orientations do not match $x_i.o! = x_{i+1}.o$, the event is marked as inversion. When x_i and x_{i+1} align to different chromosomes, the event is marked as translocation (Figure 2.2).

We introduce a clipping difference in order to filter out possible false positives. Because $rc + ql + lc$ equals the total length of a split-read alignment X, we can know the relative position of a subread in a single sample read by checking rc or lc. The relative distance between xi and xi+1 is computed as $x_{i+1}.lc - (x_i.lc + x_i.ql)$, we call this clipping difference, denoted by *cd*. For deletion, duplication, inversion and translocation, $\langle x_i, x_{i+1} \rangle$ should satisfy $-cd\_threshold < cd < cd\_threshold$, we want the relative location of x_{i+1} right after x_i, it means if x_{i+1} is far from x_i, or x_{i+1} overlap with x_i, there could be more complicated structure variation involved that we dont consider here. For insertion, if $\langle x_i, x_{i+1} \rangle$ are very close on the reference coordinate, then we check *cd*; if $cd > abs(x_i.e - x_{i+1}.s)$, it indicates an insertion. We further illustrate the algorithm for inversion. When $x_i.o = +$ and $x_{i+1}.o = -$, the potential breakpoints pair could be $\langle x_i.s, x_{i+1}.s \rangle$ or $x_i.e$ could be another potential start breakpoint, then the algorithm


A. Deletion
Sample read
split to two part
clipping
mapping to reference
deletion
Reference
B. Insertion
insertion
Sample read
split reads
clipping
mapping to reference
Reference
C. Duplication
Sample read
split to two part
mapping part
clipping part
mapping to reference
duplication
Reference
D. Inversion
Sample read
clipping part
split reads
mapping to reference
Reference

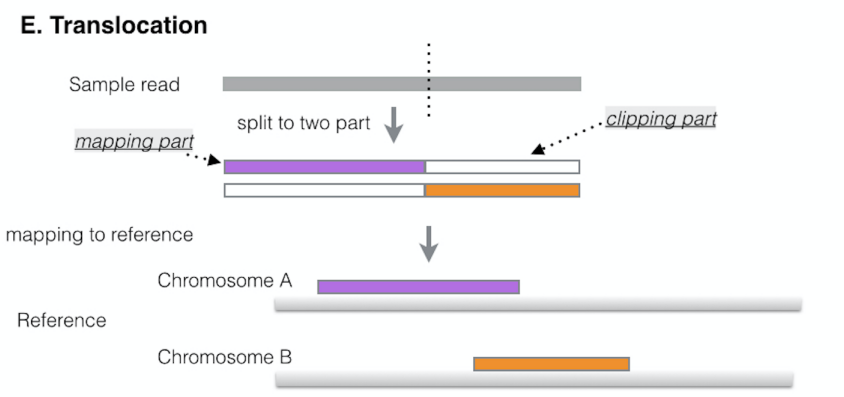


Figure 2.2: SV detection method of split read

searches for the next subread until it finds a subread that has different orientation $x_{i+n}.o = +$, so another breakpoints pair could be $\langle x_i.e, x_{i+n-1}.e \rangle$. Therefore, there are two potential events for inversion when orientation changes.

2.2.4 Clustering

To cluster two or more SV calls with close breakpoints, we provide two clustering methods. The first clustering method is that we first sort the SV list by start breakpoints, then sort end breakpoints. Then, the combined condition is to check if the breakpoints of one item close to the breakpoints of the other item by considering the starting point and end point at a distance less than a buffer length. Buffer length is set to 150 bp in this method. After combining the close events, the breakpoints of the new merged event is the average position of each event which have been combined. The second clustering method is similar to the first one, but the combined condition is different. We check if the two SV calls have 90% reciprocal overlap, so buffer length will be automatically changed based on the seed SV length every time. Due to insertions only having one breakpoint on the reference coordinate and translocations having two breakpoints in different chromosomes, our clustering method 2 is not applied to cluster insertions and translocations. The clustering step for different SV types is performed separately.

2.3 Representative algorithms for comparison

2.3.1 Sniffles

Sniffles, developed by Cold Spring Harbor Laboratory is a structural variation caller written in C++, and it analyzes TGS to detect SVs using evidence from split-read alignments, high-mismatch regions, and coverage analysis. sniffles analyzes noise regions by extracting the differences in the alignment and detecting the noisy regions by using plane sweep algorithm. Then the potential regions are stored in a self-balancing binary tree. We ran Sniffles on the 10x, 12x and 44x NA12878 datasets, the command line used here is

$ ./sniffles -m mapped.sort.bam -v output.vcf -s 2 -q 35 -l 50

We modified the default parameter in order to compare with our algorithm. The minimum number of reads that support a SV was 2, minimum mapping quality was 35, minimum length of SV to be reported was 50.

2.3.2 PBHoney

PBHoney identifies genomic variants via two algorithms, long-read discordance (PBHoney-Spots) and interrupted mapping (PBHoney-Tails). PBHoney-spots used intra-read discordance to identify deletions and insertions. They obtained the error rate at each position, then they applied a smoothing kernel and a slope kernel on this error rate. Error rate :

$$E_{ij} = \frac{A_{ji}}{C_i}$$

where A_{ji} is the value of the jth channel at position i in the reference and C_i is the coverage at that position.

Smoothing kernel:

$$M_{ji} = \frac{1}{2B+1} \sum_{k=i-B}^{i+B} E_{jk}$$

Slope kernel:

$$S_{ji} = \frac{1}{B} \left(\sum_{k=i-B}^{i-1} M_{jk} - \sum_{k=i+1}^{i+B} M_{jk} \right)$$

using the above approach, they identify possible structural variants by extracting regions that contain increases in discordance followed by decreases in discordance, which corresponding to the starts and ends of genomic variants, respectively.

We used PBHoney-Spots data run on 10x NA12878 BAM file mapped with NGM-LR aligner downloaded from http://www.pacb .com/blog/identifying-structural-variants-na12878-low-fold-coverage-sequencing-pacbio-sequel-system/. The command line used here is

$ Honey.py spots mapped.sort.bam –reference ref.fa -E 2 -i 50 –consensus None

The default minimal read support was 3, we modified to 2 (-E), and minimal SV length was modified to 50 bp for comparison.

2.4 Performance Evaluation of SV callers

The predicted SVs of each caller were compared with the gold standard SV sets. We used 50% reciprocal overlap for the matching. We used sensitivity, precision and F1 score to evaluate the performance of the callers. Sensitivity is the probability that a reference variant is called as a variant. Precision estimates the probability that a variant call is truly a reference variant. Sensitivity, precision and F1 were calculated as follows:

$$Sensitivity = \frac{TP}{TP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

$$F1 = 2 \times \frac{sensitivity \times precision}{sensitivity + precision}$$

Where TP is the number of true positives, FP is the number of false positives, and FN is the number of false negatives. We defined TP as variants called by a SVs caller and matching with the gold standard set, FP are variants called by a SVs caller but not in the gold standard set, and FN are variants in the gold standard set but not called

by a SVs caller.

Chapter 3

Results

3.1 IGV visualization for deletion and duplication

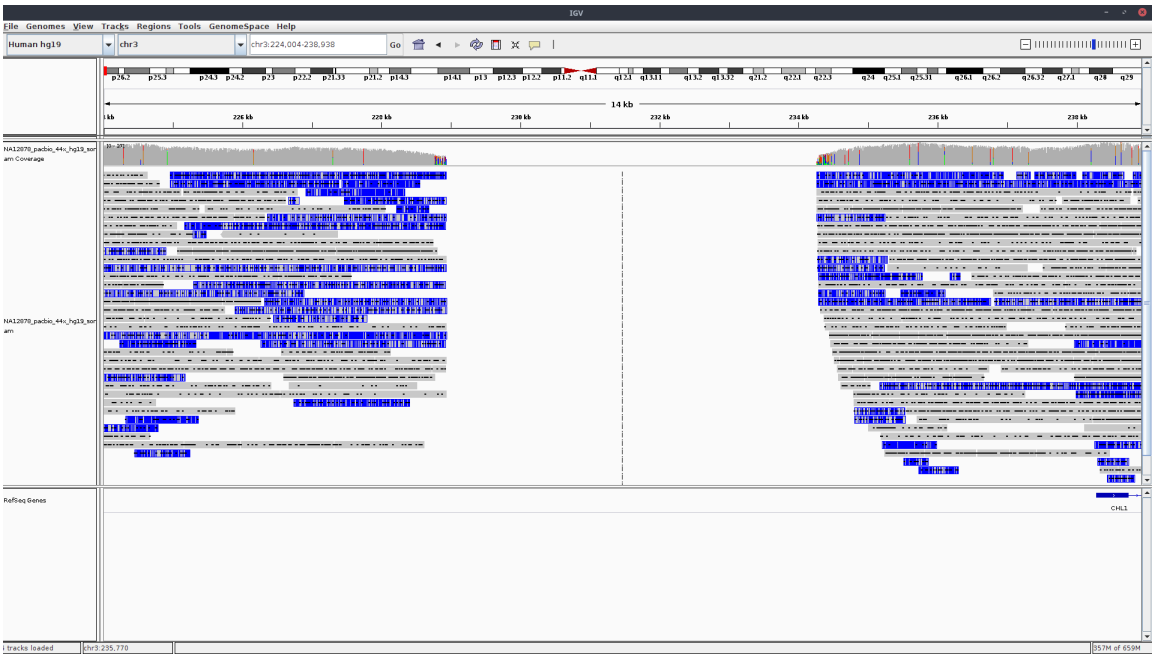
Integrative genomics viewer (IGV) is a visualization tool that enables intuitive real-time exploration of diverse, large-scale genomic datasets. As shown in figure 3.1, we used IGV to show examples of a deletion in chromosome 3 starting from position 228971 to position 233971 which was found by RD algorithm and a duplication in chromosome 17 starting from position 122686 to position 123426 which was found by SR algorithm.

3.2 Clustering method 1 and method 2 comparison

We compared the performance of clustering method 1 and clustering method 2 in 12x NA12878 data to evaluate which method is optimal. The combined condition of clustering method 1 is to check if the distance of the breakpoints of two predicted SVs is within a buffer length; while the combined method of clustering method 2 is to check if two SVs are 90% reciprocal overlapping. As Table 3.1 shown, Method 1 has higher sensitivity for calling deletions, duplications and inversions. Though Method 2 has higher precision for deletions, we desire higher sensitivity in the SV calling algorithm to achieve the completeness of a SV call set. Thus, we chose Method 1 to do the following evaluation.

3.3 10x NA12878 dataset with NGM-LR aligner

To evaluate the SV calling performance of our algorithm, we downloaded a 10x PacBio data set of NA12878 aligned with NGM-LR and compared with Sniffles, and PBHoneyspots. NGM-LR is a long-read mapper designed to correctly aligned reads spanning complex structural variations. The resulting calls were compared with the gold standard



(a) Deletion evidence

(b) Duplication evidence

Figure 3.1: IGV visualization

Algorithm	Sensitivity	Precision	F1 score
Deletion (truth set n=3558)			
Method1	**0.66469**	0.35293	0.46105
Method2	0.64193	**0.50657**	**0.56627**
Duplication (truth set n=1169)			
Method1	**0.03763**	**0.00667**	**0.01134**
Method2	0.02822	0.00548	0.00919
Inversion (truth set n=486)			
Method1	**0.54526**	**0.04876**	**0.08937**
Method2	**0.54526**	0.04131	0.07680

Table 3.1: Clustering methods comparison

SV set including 2676 deletion calls and 68 insertion calls from the Genome in A Bottle (GIAB) consortium. We calculated sensitivity, precision and F1 scores to evaluate the SV callers. For deletions, we consider a predicted call is TP if it has 50% reciprocal overlapping with a SV in true set, otherwise, it is FP. For insertions, there is only one breakpoint identified, so we defined a called SV is TP if its breakpoint is within 50bp of the breakpoint of a true SV. As shown in Figure 3.2 and Table 3.2, our algorithm shows somewhat higher sensitivity for deletions. For insertions, the sensitivity of our algorithm and Sniffles were both 0.51, which is higher than PBHoney-spots. In Table 4, our algorithm identified more TPs than Sniffles for insertions, because there may be multiple called SVs matched to one true SV.

3.4 12x NA12878 dataset with BWA-SW aligner

We next tested our algorithm with a higher coverage dataset. The resulting calls were compared with different gold standard SV sets from different sources including 3 deletion sets, 2 duplication sets, 1 insertion set and 1 inversion set, as shown in Table 3.3. 83% deletions and 51% insertions in the GIAB gold standard set can be detected by our algorithm, which is higher than Sniffles. Although only 57% deletions in the DGV gold standard set and 61% deletions in Mill gold standard set were detected by our

Algorithm	True Positives	False Positives	Sensitivity	Precision	F1 score
Deletion, Genome in a Bottle (truth set n=2676)					
Mypbsv	2205	3986	**0.81875**	0.35616	0.49639
Sniffles	2086	3042	0.78101	**0.40678**	**0.53494**
PBHoney	1662	3025	0.62182	0.35459	0.45164
Insertion, Genome in a Bottle (truth set n=68)					
Mypbsv	45	25057	**0.5147**	0.00179	0.00357
Sniffles	35	70546	**0.5147**	0.00049	0.00009
PBHoney	31	6240	0.45588	**0.00494**	**0.00978**

Table 3.2: SV calling performance for each SV caller on the 10x NA12878 with NGM-LR aligner

algorithm, which is slightly less than Sniffles which detected 66% deletions in DGV and 61% deletions in Mill, our algorithm had higher precision in deletions, so that F1 scores for deletions based on three gold standard sets are higher. The precision and sensitivity for insertion calls of our algorithm was higher.

We also used NA12878 SV set from Illumina data, called by GROM, as validation set to assess our result. As shown in Figure 3.4 and Table 3.4, 65% deletions, 20% insertions, 52% inversions in illumina SV sets were matched by our algorithm. The precision for deletion calls and insertion calls of our algorithm shown higher than Sniffles. However, our algorithm has lower sensitivity for duplications and translocations than Sniffles either in the gold standard set or the Illumina set.

3.5 44x NA12878 dataset with BWA-MEM aligner

We further assessed our algorithm with 44x NA12878 dataset with a different aligner. As shown in Figure 3.5, 3.6 and Table 3.5, 3.6, the sensitivity and precision for insertions of our algorithm in the GIAB set and Illumina set are higher. The sensitivity and precision for deletions of our algorithm in all Gold standard sets and illumina set are slightly lower than Sniffles. For inversions, our algorithm perform better in the Illumina set. For duplications and translocations, our algorithm still showed lower sensitivity and

precision. The entire performance of SV calling in 44x NA12878 dataset was 6% $\sim$ 30% less than the performance of SV calling in 12x NA12878 dataset.

3.6 Running time and memory cost

Table 3.7 summarizes running time and memory cost of our algorithm for different datasets in both parallel and serial modes to show workload distribution and method efficiency. We used four threads in our parallel mode. All timings were performed on a Intel Xeon CPU E5-1620 v2 processor, 3.7 GHz with 16GB RAM. For a 20 gigabytes BAM file, it took about 0.2 hr to run. The largest file, which has 267 gigabytes, took about 2.4 hr in parallel mode. The parallel mode is about 3 times faster than serial mode.

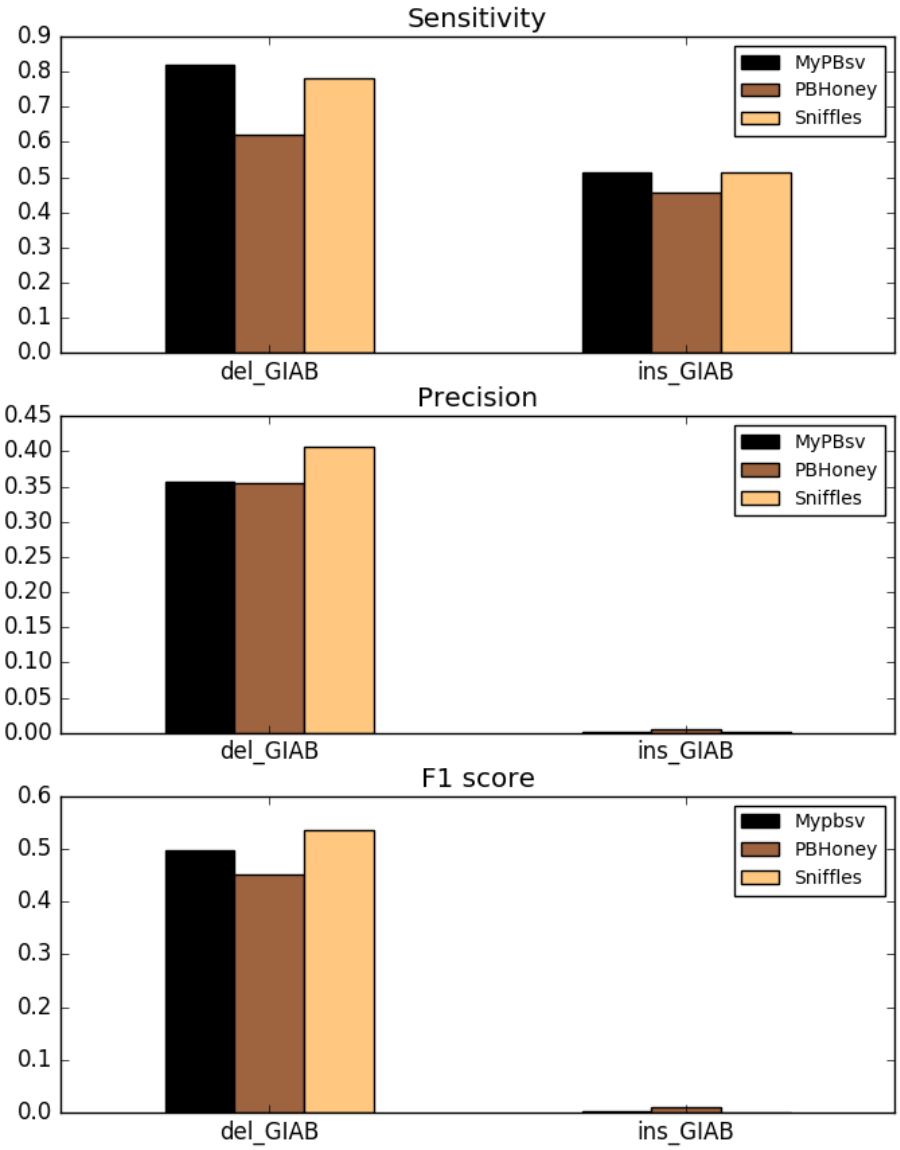


Figure 3.2: SV calling performance for each SV caller on the 10x NA12878 with NGM-LR aligner

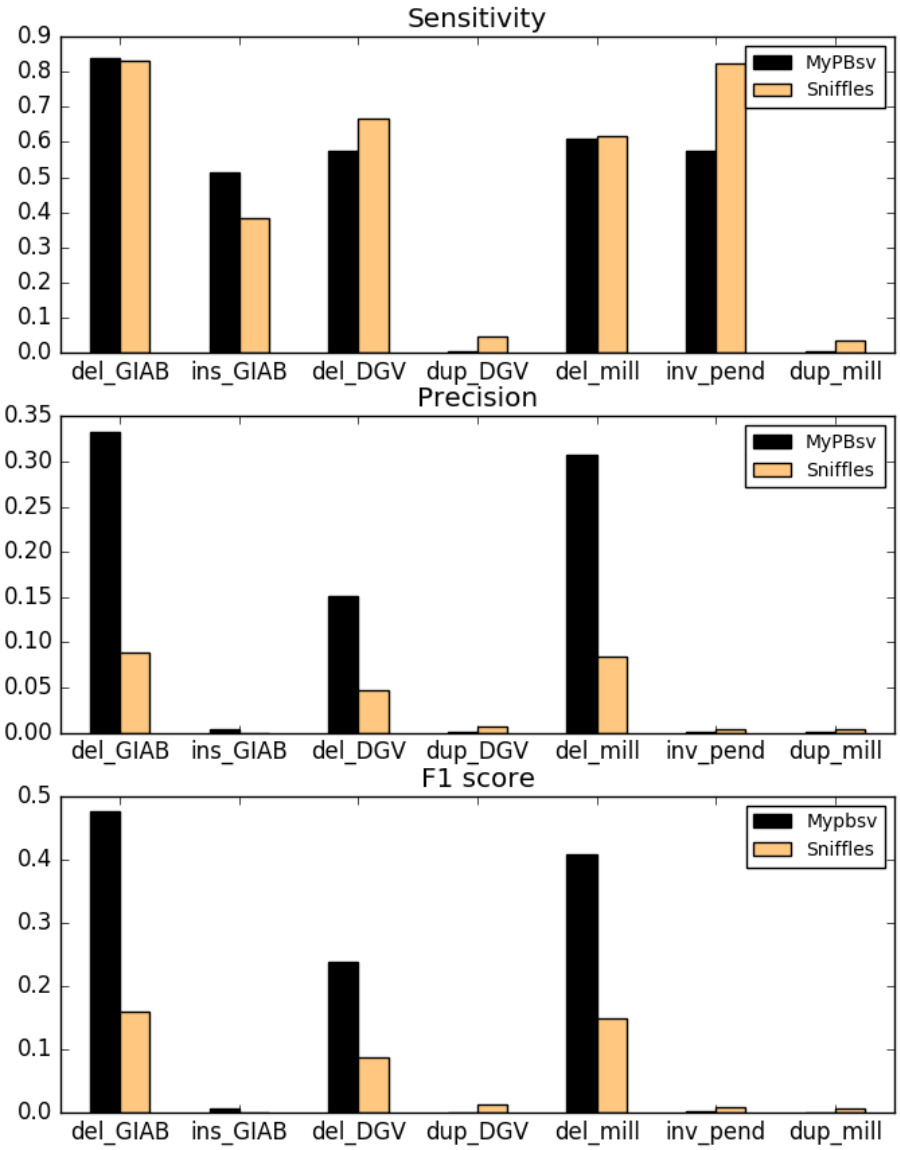


Figure 3.3: SV calling performance for each SV caller on the 12x NA12878 with BWA-SW with Gold standard

Algorithm	True Positives	False Positives	Sensitivity	Precision	F1 score
Deletion, Genome in a Bottle (truth set n=2676)					
Mypbsv	2289	4582	**0.83893**	**0.33313**	**0.47690**
Sniffles	2256	23280	0.83071	0.08834	0.15970
Insertion, Genome in a Bottle (truth set n=68)					
Mypbsv	47	14484	**0.51470**	**0.00323**	**0.00642**
Sniffles	28	456525	0.38235	0.00001	0.00012
Deletion, DGV (truth set n=1935)					
Mypbsv	1034	5837	0.57409	**0.15048**	**0.23846**
Sniffles	1192	24344	**0.66580**	0.04667	0.08724
Duplication, DGV (truth set n=570)					
Mypbsv	2	6735	0.00350	0.00029	0.00054
Sniffles	27	3564	**0.04736**	**0.00751**	**0.01297**
Deletion, Mill (truth set n=3376)					
Mypbsv	2108	4763	0.61078	**0.30679**	**0.40843**
Sniffles	2165	23371	**0.61759**	0.08478	0.14909
Duplication, Mill (truth set n=298)					
Mypbsv	2	6735	0.00671	0.00029	0.00056
Sniffles	14	3577	**0.03691**	**0.00389**	**0.00705**
Inversion, pendleton (truth set n=40)					
Mypbsv	30	29204	0.575	0.00103	0.00204
Sniffles	37	9216	**0.825**	**0.00399**	**0.00795**

Table 3.3: SV calling performance for each SV caller on the 12x NA12878 with BWA-SW with Gold standard

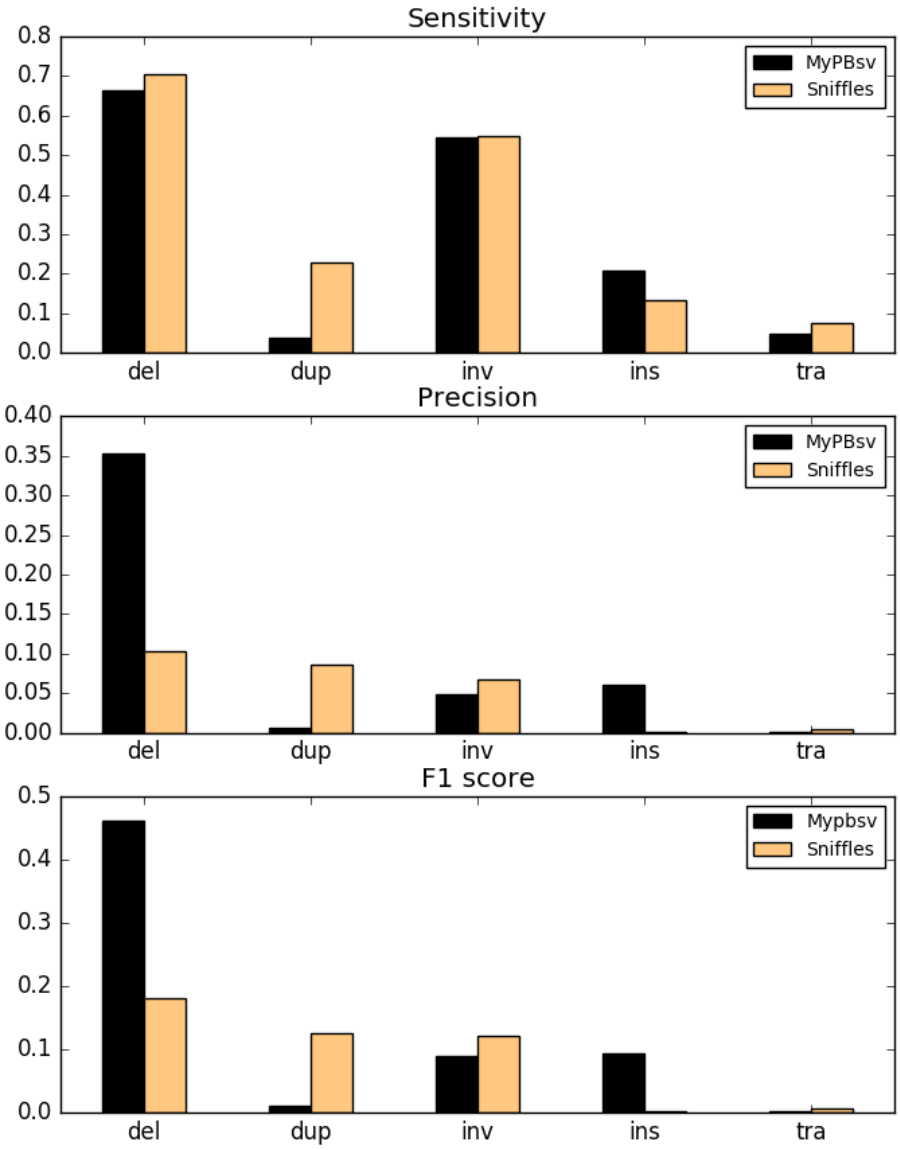


Figure 3.4: SV calling performance for each SV caller on the 12x NA12878 with BWA-SW with Illumina dataset

Algorithm	True Positives	False Positives	Sensitivity	Precision	F1 score
Deletion (truth set n=3558)					
Mypbsv	2425	4446	0.66469	**0.35293**	**0.46105**
Sniffles	2632	22904	**0.70573**	0.10307	0.17987
Insertion (truth set n=3615)					
Mypbsv	882	13649	**0.20774**	**0.06069**	**0.09394**
Sniffles	481	456072	0.13333	0.00105	0.00209
Duplication (truth set n=1169)					
Mypbsv	45	6692	0.03763	0.00667	0.01134
Sniffles	310	3281	**0.23011**	**0.08632**	**0.12555**
Inversion (truth set n=486)					
Mypbsv	1423	27811	**0.54526**	0.04867	0.08937
Sniffles	631	8622	0.54938	**0.06819**	**0.12132**
Translocation (truth set n=1419)					
Mypbsv	34	24990	0.04792	0.00135	0.00264
Sniffles	55	14392	**0.07610**	0.00380	0.00725

Table 3.4: SV calling performance for each SV caller on the 12x NA12878 with BWA-SW with Illumina dataset

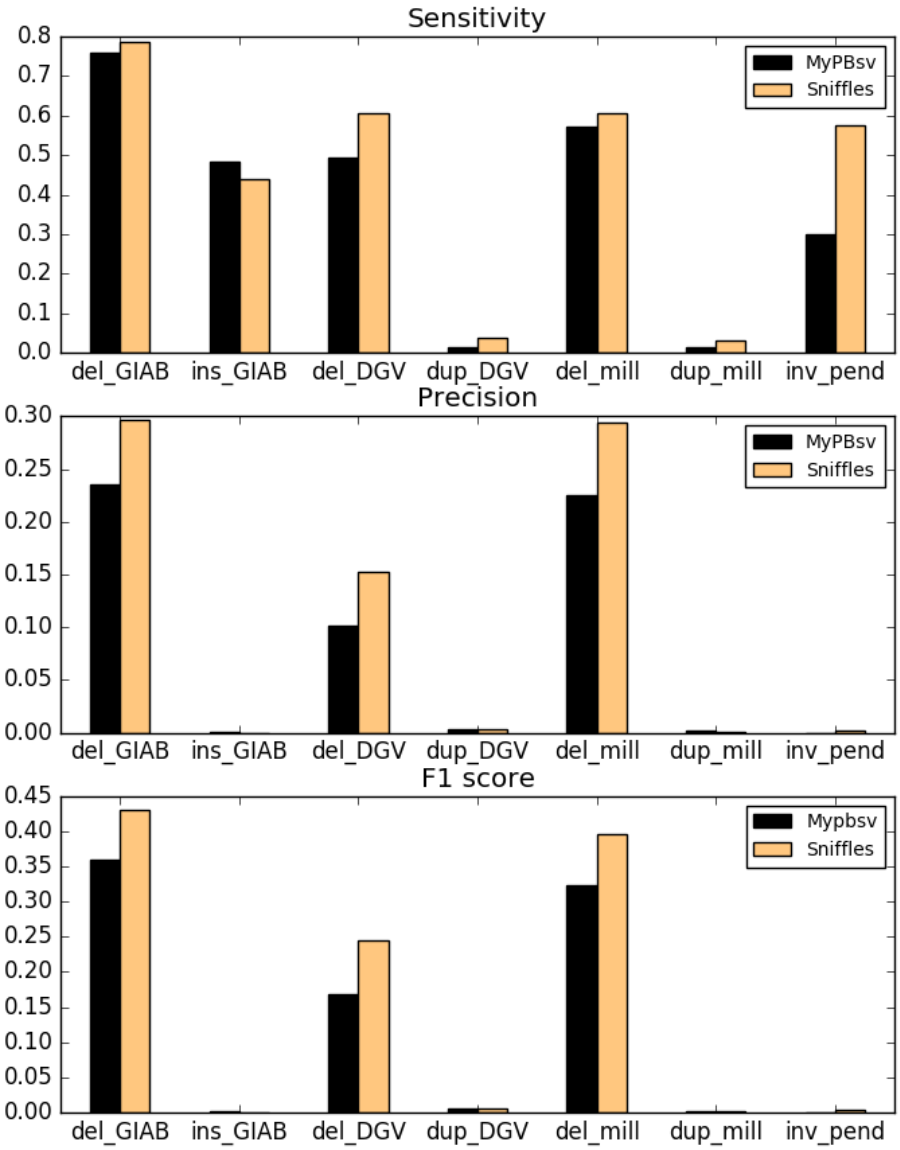


Figure 3.5: SV calling performance for each SV caller on the 44x NA12878 with BWA-MEM with Gold standard

Algorithm	True Positives	False Positives	Sensitivity	Precision	F1 score
Deletion, Genome in a Bottle (truth set n=2676)					
Mypbsv	2079	6758	0.76046	0.23526	0.35935
Sniffles	1978	5336	**0.78512**	**0.27044**	**0.40230**
Insertion, Genome in a Bottle (truth set n=68)					
Mypbsv	45	39956	**0.48529**	**0.00112**	**0.00224**
Sniffles	32	576768	0.44117	0.00001	0.00011
Deletion, DGV (truth set n=1935)					
Mypbsv	893	7944	0.49378	0.10105	0.16777
Sniffles	1070	6244	**0.60621**	**0.14629**	**0.23570**
Duplication, DGV (truth set n=570)					
Mypbsv	9	2536	0.01578	**0.00353**	**0.00577**
Sniffles	21	7629	**0.03859**	0.00209	0.00397
Deletion, Mill (truth set n=3376)					
Mypbsv	1990	6847	0.57435	0.22518	0.32338
Sniffles	2003	5311	**0.60723**	**0.27385**	**0.37747**
Duplication, Mill (truth set n=298-30)					
Mypbsv	4	2541	0.01342	**0.00157**	**0.00281**
Sniffles	7	7638	**0.03020**	0.00091	0.00177
Inversion, pendleton (truth set n=40)					
Mypbsv	12	80948	0.3	0.00014	0.00029
Sniffles	24	15095	**0.575**	**0.00158**	**0.00316**

Table 3.5: SV calling performance for each SV caller on the 44x NA12878 with BWA-MEM with Gold standard

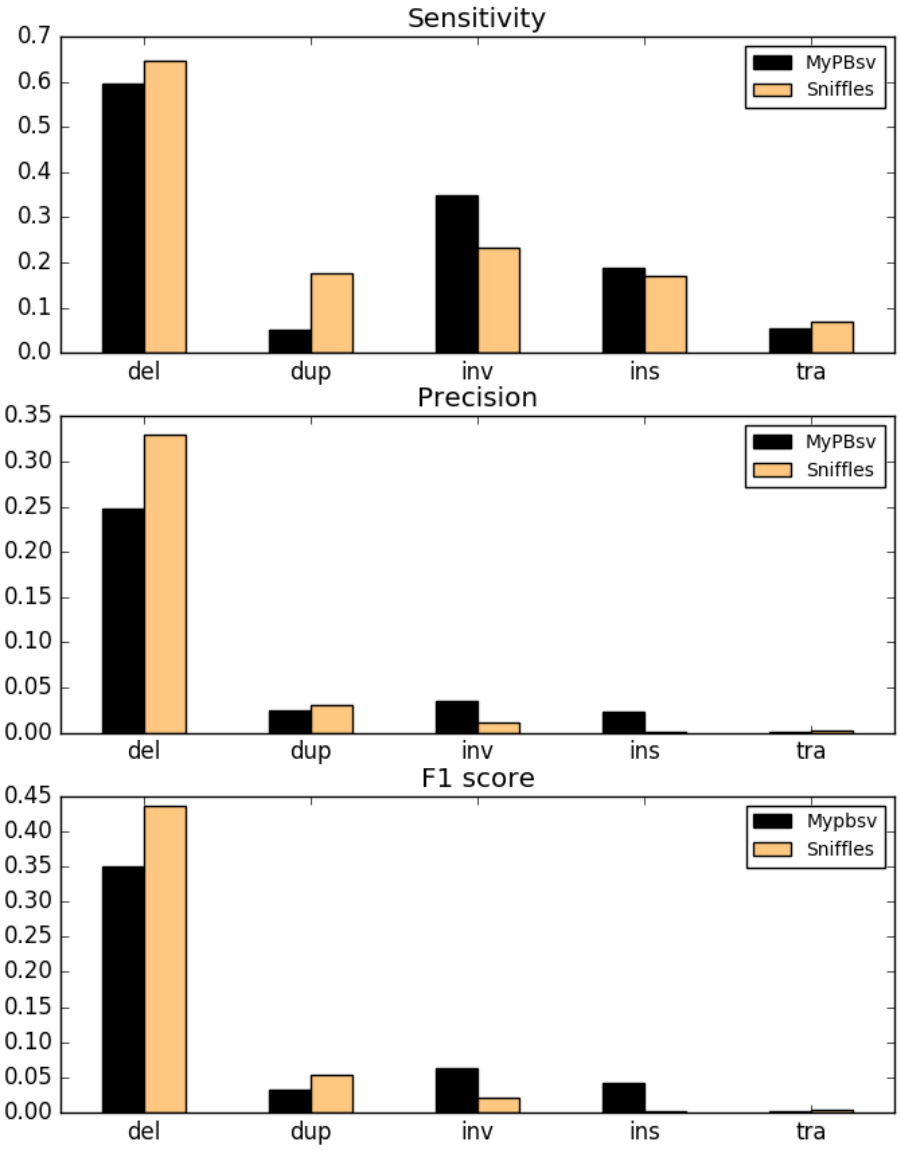


Figure 3.6: SV calling performance for each SV caller on the 44x NA12878 with BWA-MEM with illumina dataset

Algorithm	True Positives	False Positives	Sensitivity	Precision	F1 score
Deletion (truth set n=3558)					
Mypbsv	2185	6652	0.59724	0.24725	0.34972
Sniffles	2001	5313	**0.64558**	**0.27358**	**0.38430**
Insertion (truth set n=3615)					
Mypbsv	930	39071	**0.18865**	**0.02324**	**0.04139**
Sniffles	614	576186	0.17040	0.00106	0.00211
Duplication (truth set n=1169)					
Mypbsv	62	2483	0.05218	0.02436	0.03321
Sniffles	194	7451	**0.17621**	**0.02537**	**0.04436**
Inversion (truth set n=486)					
Mypbsv	2802	78158	**0.34773**	**0.03460**	**0.06295**
Sniffles	128	14991	0.23251	0.00846	0.01633
Translocation (truth set n=1419)					
Mypbsv	38	30268	0.05355	0.00125	0.00245
Sniffles	50	24917	**0.07047**	0.00125	0.00389

Table 3.6: SV calling performance for each SV caller on the 44x NA12878 with BWA-MEM with Illumina dataset

Sample	File size (GB)	Walltime (h)		Memory (Gb)	
		Parallel	Serial	Parallel	Serial
10x NA12878	20	0.2	0.57	8	1.99
12x NA12878	120	1.23	2.8	8.5	2.10
44x NA12878	267.1	2.4	6.3	10	2.38

Table 3.7: Running time and memory cost summary

Chapter 4

Discussions

Due to the higher per-nucleotide error rates ($\sim$ 15%) in single-molecule sequencing technologies, to mitigate false positive caused by high error rates is a challenge. As shown in Table 4.1, we analyzed a 44x SV call set to categorize SV calls based on different evidence support. We observed that split-read alignment method has significantly higher precision than read depth and alignment mismatching. Alignment mismatching contributed most of the false positives for deletions and insertions, which might be due to the nature of high per-nucleotide error rates in SMRT sequencing. To reduce FPs, we might increase the threshold in read support for alignment mismatching approach, but strict filtering criteria may loss some potential TPs and leave the call set incomplete. The SV call sets with two signals support have higher precision for deletions. For example, SV call set with SP, MM signals has 86% precision. Although read depth method contribute fewer TPs and the breakpoints resolution is lower than split-read, it can find unique and large size SVs which split-read might not find. For instance, our read-depth algorithm found 4 duplications in DGV gold standard set that Sniffles didnt find. We acknowledge that the read-depth algorithm needs to be improved in future work, and we believe that the read-depth approach is a complement method that will help us to find more potential SVs.

We used different coverage datasets to assess the performance of our algorithm. The results showed that our algorithm performed better on low coverage NA12878 dataset rather than high coverage dataset. We might need to test different datasets such as a recently published Chinese genome HX1 to avoid a bias performance on our algorithm. Different aligners used might also be a factor affecting our algorithms performance. Our algorithm showed better results when the genome data was aligned by NGM-LR and BWA-SW. Allowing BAM files from different aligners as input is a merit of our algorithm. Compared with some SV callers that depend on a specific

Method	Deletion (GIAB)			Duplication (DGV)			Insertion (GIAB)		
	TP	FP	precision	TP	FP	precision	TP	FP	precision
SR	861	402	0.68171	4	1287	0.00309	18	1084	0.01633
MM	410	5896	0.06501	NA	NA	NA	18	36466	0.00049
RD	44	342	0.11398	5	1241	0.00401	NA	NA	NA
SR,MM	761	118	0.86575	NA	NA	NA	9	2406	0.00372
SR, RD	3	0	1	0	8	0	NA	NA	NA

Table 4.1: SV signals comparison

aligner, for example, PBHoney-tails have to depend on BLASR, which is also a long read aligner, and now BLASR have been changed to a new version, PBHoney-tails have been incompatible with BLASR and it might no longer be used. Our algorithm accept BAM file from at least three different aligners which provides flexibility and less limitations for users. We also attempted to run a BAM file from BLASR. The RD and MM algorithms were adequate to detect SVs, but our split-read algorithm was not applied because BLASR doesnt find chimeric reads and mark SA tags to be applied by our algorithm.

Our duplication call set is called by SR and RD algorithms which we expected to find more TPs and increase sensitivity, but it was not as our expectation. To improve duplication detection, we observed that the length of our duplications ranged from 50 $\sim$ 10000 bp, but the length of gold standard duplications are distributed from 100 $\sim$ 100000 bp (Figure 4.1). We might try to merge overlapping duplications to produce longer duplications in the SR algorithm. For the RD algorithm, we selected duplication regions which have abnormal coverage of three times standard deviation more than average. This threshold might be too stringent. An alternative way, we might increase the window size which we used to calculate our read depth mean and standard deviation.

Our algorithm integrates split-read, read-depth and alignment mismatching signals to detect SVs which is comprehensive discovery of structural variation. Table 4.2 summarize the methods used by each caller. Compared with PBHoney-spots and Sniffles,

our approach included read depth signal which might find more different variants from them. In addition to facilitating signal integration, our use of parallel computing during SV discovery should reduce time consummation which could help analyze increasing amounts of data on WGS efficiently.

Algorithm	Methods			multithreading
	aln mismatching	split read	read depth	
Mypbsv	✓	✓	✓	✓
Sniffles	✓	✓		✓
PBHoney-spots	✓			

Table 4.2: Algorithms comparison

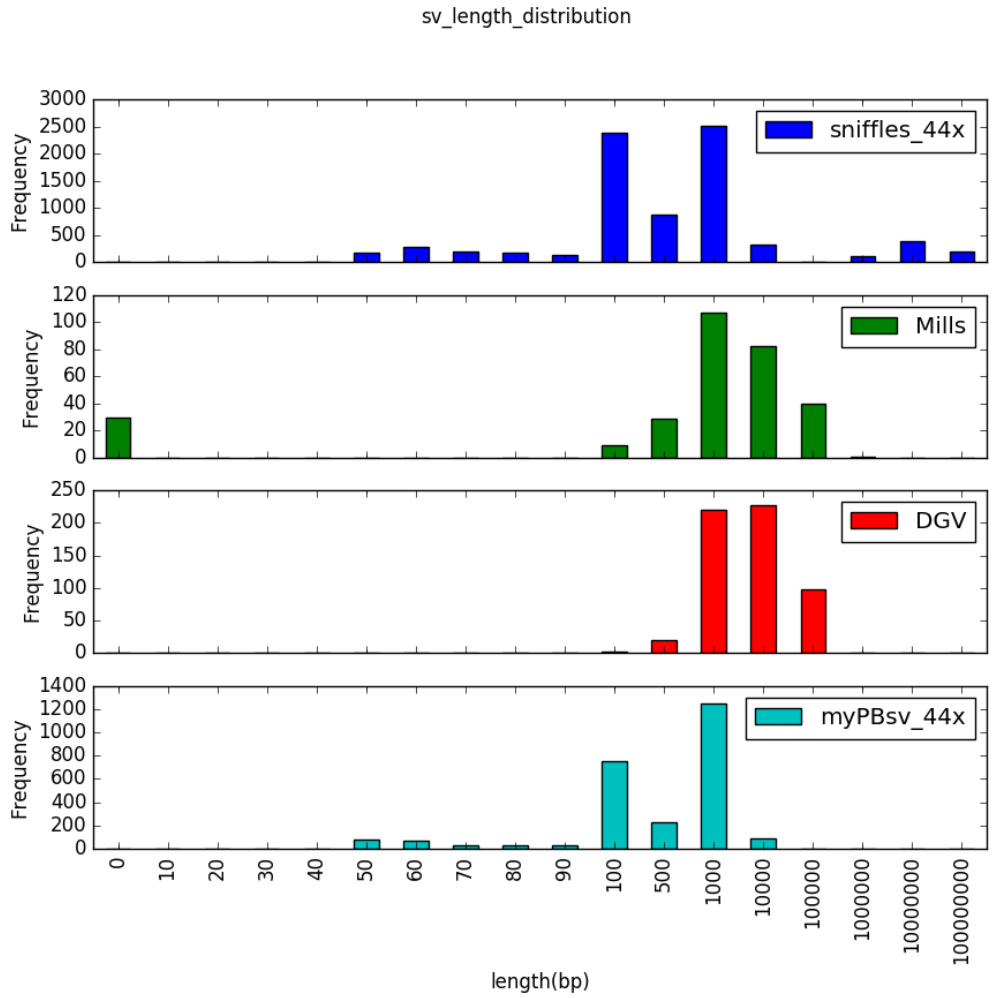


Figure 4.1: Length distribution of duplication

Chapter 5

Conclusion

Structural variation detection faces many challenges when creating a completely characterized genome with identified large and complex variants. Here, we describe an algorithm simultaneously integrating multiple SV detection signals with high mappability of long-reads during structural variation discovery. Our approach is flexible to support bam file from several aligners and analyze large-scale SV calling efficiently. We expect our method will be a community resource to facilitate practical and routine structural variant analysis in genome sequencing research.

References

[1] K. E. Ashelford et al. “At least 1 in 20 16S rRNA sequence records currently held in public repositories is estimated to contain substantial anomalies”. In: *Appl. Environ. Microbiol.* 71.12 (Dec. 2005), pp. 7724–7736.

[2] C. E. Bruder et al. “Phenotypically concordant and discordant monozygotic twins display different DNA copy-number-variation profiles”. In: *Am. J. Hum. Genet.* 82.3 (Mar. 2008), pp. 763–771.

[3] K. Chen et al. “BreakDancer: an algorithm for high-resolution mapping of genomic structural variation”. In: *Nat. Methods* 6.9 (Sept. 2009), pp. 677–681.

[4] A. C. English, W. J. Salerno, and J. G. Reid. “PBHoney: identifying genomic variants via long-read discordance and interrupted mapping”. In: *BMC Bioinformatics* 15 (June 2014), p. 180.

[5] P. Guan and W. K. Sung. “Structural variation detection using next-generation sequencing data: A comparative technical review”. In: *Methods* 102 (June 2016), pp. 36–49.

[6] R. M. Layer et al. “LUMPY: a probabilistic framework for structural variant discovery”. In: *Genome Biol.* 15.6 (June 2014), R84.

[7] S. Levy et al. “The diploid genome sequence of an individual human”. In: *PLoS Biol.* 5.10 (Sept. 2007), e254.

[8] H. Li and R. Durbin. “Fast and accurate long-read alignment with Burrows-Wheeler transform”. In: *Bioinformatics* 26.5 (Mar. 2010), pp. 589–595.

[9] H. Li et al. “The Sequence Alignment/Map format and SAMtools”. In: *Bioinformatics* 25.16 (Aug. 2009), pp. 2078–2079.

[10] J. R. Lupski et al. “DNA duplication associated with Charcot-Marie-Tooth disease type 1A”. In: *Cell* 66.2 (July 1991), pp. 219–232.

[11] R. E. Mills et al. “Mapping copy number variation by population-scale genome sequencing”. In: *Nature* 470.7332 (Feb. 2011), pp. 59–65.

[12] A. W. Pang et al. “Mechanisms of formation of structural variation in a fully sequenced human genome”. In: *Hum. Mutat.* 34.2 (Feb. 2013), pp. 345–354.

[13] M. Pendleton et al. “Assembly and diploid architecture of an individual human genome via single-molecule technologies”. In: *Nat. Methods* 12.8 (Aug. 2015), pp. 780–786.

[14] T. Rausch et al. “DELLY: structural variant discovery by integrated paired-end and split-read analysis”. In: *Bioinformatics* 28.18 (Sept. 2012), pp. i333–i339.

[15] R. J. Roberts, M. O. Carneiro, and M. C. Schatz. “The advantages of SMRT sequencing”. In: *Genome Biol.* 14.7 (July 2013), p. 405.

[16] M. G. Ross et al. “Characterizing and measuring bias in sequence data”. In: *Genome Biol.* 14.5 (May 2013), R51.

[17] J. Weischenfeldt et al. “Phenotypic impact of genomic structural variation: insights from and for human disease”. In: *Nat. Rev. Genet.* 14.2 (Feb. 2013), pp. 125–138.