Analysis of the effect of genetic heterogeneity on *de novo* genome assembly using *Xylocopa virginica* as a model.

Haimeng Tang

B.Sc. (Honours)

Gene Biotechnology

Submitted in partial fulfillment of the requirements for the degree of Master of Science

Faculty of Mathematics and Science, Brock University

St. Catharines, Ontario

Abstract

Next generation sequencing (NGS) technology has revolutionized genomic and genetic research, and as a result, *de novo* genome sequencing and assembly for non-model organisms has now become a common task in genome research. However, the integral properties of a genome such as ploidy, mutations, and repeat content impose issues for current genome assemblers. In this work, we used *Xylocopa virginica* (Eastern carpenter bees) as a unique model organism for examining on the effect of sequence heterozygosity on quality of *de novo* genome assembly. Using two de Bruijn graph genome assemblers, we assembled four bee genomes representing different sex and age (unworn male, worn male, unworn female, worn female) using standard Illumina sequencing and one genome using 10X linked-reads library for an unworn female. We discovered that there is a noticeable difference in a variety of genome assembly quality metrics, with the haploid unworn male genome having the highest quality and the worn diploid female genome having the lowest quality. In fact, the N50 value of the unworn male genome was >100 times higher than that of the worn female genome. The genome quality pattern supports the hypothesis that sequence heterozygosity resulting both from ploidy and somatic variants can affect the result of an assembly with former shown to be a much bigger player than the latter. Furthermore, we observed that the density of variants was moderately correlated to the density of breakpoints in the genome assemblies.

Overall, our results indicate that increased ploidy and accumulation of somatic variants both negatively affect the quality of the resulting assembly with the former being much more significant than the latter. When considering a *de novo* assembly project for a non-model organism, whenever possible, haploid samples at the youngest possible age are to be recommended. Furthermore, use of a long-read platform can lead to better genome quality.

However, at least for the 10x linked reads, having too much sequencing data does not necessarily lead to a better genome assembly.

Keywords: *de novo* assembly, non-model organism, variant detection, coverage, 10X barcode sequencing

Acknowledgements

I am very grateful to my supervisor, Dr. Ping Liang, for his support and patience throughout my time in his lab. I would also like to thank my committee members, Dr. Adonis Skandalis and Dr. Feng Li, for all their helpful insights and discussions. A big thank you goes out to Dr. Miriam Richards and Lyndon Duff for the sample collection and answering any related questions. An additional thank you to Dr. Adonis Skandalis for extracting the genomic DNA from the collected samples. I am also very appreciative of all my colleagues in the lab for giving helpful advice whenever I needed it. Finally, I would like to thank with gratitude, the love and unconditional support from my family and friends, without which I would not have been able to complete anything.

Table of Contents

A list of tables and figures

A list of abbreviations

bp –base pair

CVN - Copy number variation

DBG – De Bruijn graph

DSB – Double-strand break

Gbp – Giga base pair

kbp - Kilo base pair

LINE – Long interspaced nuclear element

LTR – Long terminal repeat

Mbp – Mega base pair

NGS – Next generation sequencing

NHEJ – Non-homologous end joining

OLC – Overlap-Layout-Consensus

PCR – Polymerase Chain Reaction

PE – Pair-end

SNP – Single nucleotide polymorphism

SDS – Sodium dodecyl sulfate

SLR – Synthetic long read

SV - Structural variant

SVA – SINE/VNTR/Alu

TE – Transposable element

CHAPTER 1: Introduction

1.1. *De novo* genome sequencing and assembly

Genome assembly has progressed at an incredible pace since next generation sequencing has become popular. The development of new techniques and programs allow for more species to be more quickly and accurately processed. Generally, assembly of reads is conducted according to a pre-existing reference genome. Sequenced reads are mapped onto the reference at the most likely position based on alignment. This process is often appropriately known as reference-based assembly. In comparison, *de novo* genome assembly refers to the process of reconstructing a genome sequence without the guidance of a reference. This raises the challenge of performing the assembly purely based on the limited relationships of sequence reads within the data for the testing sample, while *de novo* genome assembly permits us to study non-model organisms, for which, we usually have no high-quality draft genome available.

Attempting to assemble a genome with high repeat content, either tandem repeats or gene duplications can have a huge impact on the time and resources needed to create a high-quality assembly. Depending on the algorithm used, duplicate regions may be superimposed into a single region or discarded entirely (Sohn & Nam, 2018). This will result in variations in the length of the assembly and will thus impact the accuracy. Sequence heterozygosity from allelic differences in a diploid genome, which is the case for most eukaryotic samples, can cause uncertainties in the assembly process. This is an issue if the purpose of analysis is to phase haplotypes for maternal and paternal chromosomes in the offspring. Most assemblers are not equipped to handle such complexity and will integrate the two copies into one consensus sequence comprising of alleles randomly selected from one of the two diploid copies (Sohn & Nam, 2018).

Assemblers operate by taking one or several input DNA libraries and assembling the reads based on overlaps or using a graph-based approach (Baker, 2012). Different assemblers have different requirements for memory and CPU time depending on the algorithm applied and the specific goal it is for. Some assemblers, like `Supernova`, work well for diploid organism (Weisenfeld et al., 2017), while others, like `SPAdes`, work best with single cell sequencing libraries (Bankevich et al., 2012). Other examples include `SOAPdenovo-Trans`, a part of the `SOAPdenovo` package, being a tool specifically for *de novo* assembly of transcriptomes (Xie et al., 2014). Computational limits need to be considered when attempting an assembly. An assembler using the Overlap-Layout-Consensus algorithm will have a higher RAM and CPU requirement than an assembler using the de Bruijn Graph algorithm, and larger genomes will generally take more time and memory to assemble (Khan et al., 2018, Sohn & Nam, 2018).

Finally, even if an assembly is generated successfully, the accuracy of the draft genome may be difficult to determine. Some repetitive regions may have been discarded, shorter scaffolds may have been ignored in the final assembly and sequencing errors cause areas of mis-assembly in the genome. Nevertheless, certain quantitative measures can be used to give a general idea, although these measures can also change depending on how stringent the algorithms are. Measurements such as N50 values, number and length of contigs and scaffolds, and number of gaps in the assembly can all be indicative of assembly quality.

Despite these issues, *de novo* genome assembly is gaining popularity (Baker, 2012). Back in February of 2012, the Genome Online Database (GOLD) listed around fifteen thousand genome sequencing projects from a variety of sources including academic, sequencing centers, and private institutes (Baker, 2012). Of those, twelve thousand were either in the planning phase or in progress. Since then, the increase of interest by the general public in personalized medicine

as well as an improvement in sequencing technologies has led to an even greater influx of genomic data. In fact, as of the writing of this paper, there are over 171,000 whole genome sequencing projects listed on GOLD, among which over 68% represents de novo genome sequencing. Clearly, *de novo* assembly has become a critical part of current genetic analysis. With so much sequencing data already in the database and much more to come in the following years, it can be expected that biologists can unravel the intricacies of *de novo* genome assembly. In fact, an enormous amount of work has gone into refining the process of *de novo* assembly, and there have been noticeable improvements in the quality and speed at which assembly can be completed. Being able to rapidly produce genome assemblies for organisms will be reliant on the processing power of computers and the efficiency of the assembly tools, but the quality of the assembly is something that scientists can deduce even before assembly, simply based on knowledge of the organism. The factors affecting the quality of assembly can come from the genome itself, the nature of the data provided to the sequencer, computational limits (Sohn & Nam, 2018). International collaborations have been working to further perfect the assembly methods. The second Assemblathon, which concluded in 2013, compared a multitude of genome assembly tools (both standalone and pipelines) for data from a variety of sources in an attempt to determine which tools will provide the highest quality assembly. GAGE (Genome Assembly Gold-standard Evaluations) has compared 8 assemblers and their ability to create genomes from four different species (Salzberg et al., 2012). However, even if we know that a certain factor will affect the quality of genome assembly, we are not certain of the extent to which the assembly quality will change. Research needs be done to quantify the degree to which each of these factors listed above will change the quality of assembly.

1.1.1. Sequencing platforms

A major limitation in obtaining extensive genome data is the prohibitive costs associated with sequencing and assembling large eukaryotic genomes (Quail et al., 2012). The development of next-generation massively parallel sequencing technologies has significantly improved sequencing throughput, reduced costs, and advanced research in many areas, including large-scale resequencing of human genomes (Bentley et al., 2008, Li et al., 2010), transcriptome sequencing (Salzmann et al., 2019), and epigenetic studies (Richards et al., 2018). However, the read length of these sequencing technologies, which is mostly much shorter than that of traditional capillary Sanger sequencing reads, has prevented its use as the sole sequencing technology in *de novo* assembly of large eukaryotic genomes.

Compared to traditional Sanger capillary-based electrophoresis systems, these new technologies provide ultrahigh throughput with two orders of magnitude lower cost. For instance, Illumina technology has been shown to be feasible for use in human whole-genome resequencing and can be used to identify single nucleotide polymorphisms (SNPs) accurately by mapping the short reads onto a reference genome (Poen et al., 2020).

Currently, a number of NGS platforms are available, each with its pros and cons. Illumina sequencing utilizes the sequencing by synthesis method, in which the template strand is replicated using DNA polymerase and each newly incorporated base is recorded through light emission. Bridge amplification prior to sequencing generates a localized cluster of DNA fragments of the same sequence so the emitted signal light can be detected by instruments and ensures an accurate base call. In comparison to other NGS technologies such as Nanopore, which directly identifies nucleotides as they pass through a protein pore and alter the ionic current, reads sequenced by synthesis are much more accurate and has a higher signal-to-noise ratio but

is hindered in terms of read length due to the PCR amplification step, which has an optimal length below 500 bp for consistent result, and the limited half-life of the DNA polymerase, leading to lowering sequencing quality towards the 3'-end after a certain length. Other methods of sequencing include ion semiconductor (used by Ion Torrent) and pyrosequencing (used by 454) (Th & Ma, 2015). The main drawback of these two methods is the inability to accurately sequence long homopolymer regions, in additional to having short read length like Illumina. However, the efficiency and throughput of these methods counteract that by increasing the coverage of nucleotides at each position, thus minimising the effect of incorrect base call.

A recently developed data type known as 10X linked-reads uses molecular barcode tags to index Illumina short reads that come from the same long input DNA fragment. Linked-reads provide the long-range information missing from standard Illumina approaches while maintaining the accuracy of the sequencing platform by adding a barcode to every fragment generated. Fragments with the same barcode are then identified and grouped together. This degree of long-range information enables phasing of long-range haplotypes, permitting to call complex structural variants and to perform *de novo* genome assembly with a diploid output (Church, 2016).

When first starting a *de novo* assembly project, it is important to decide on a sequencing platform, the type of library to generate, and the amount of input DNA and sequencing data required. The latter is often limited by funding or availability of sample, while the platform selection may depend on which sequencing technology is readily available. Based off a survey of the completed whole-genome sequencing projects, there is a clear trend moving away from Sanger and 454 sequencing towards short read technologies such as Illumina HiSeq and Ion Torrent (Ekblom & Wolf, 2014). In the meantime, in recently years, long read sequencers, such as Pacific Biosciences and Nanopore, have established a foothold in the market, which offer

scientists a broader spectrum of read lengths to choose from. While this development of third generation sequencing blurs the initial dichotomy of short reads (e.g. 35 bp Illumina reads) versus long reads (~1 kbp Sanger reads), read length still has important bioinformatic implications, as assembly algorithms optimized for long reads are fundamentally different from approaches targeting short reads (Ekblom & Wolf, 2014). Table 1 gives an overview of a comparison between some next generation sequencers. Several recent studies began to combine data of different read length and from several different sequencing platforms (Tan et al., 2018, Wallberg et al., 2019). This strategy makes intuitive sense as the drawbacks of each method can be counterbalanced, although an agreement has yet to be reached as to whether such hybrid assemblies always outperform single data type approaches (Ekblom & Wolf, 2014). Here, for the discussion of issues associated with *de novo* genome assembly, we follow the principle of current common practice and base our considerations largely on sequencing of Illumina libraries of different lengths (we loosely refer to short reads at sequence lengths below 500 bp and long reads above this length).

Table 1. Comparison of common next generation sequencers.

Sequencer	*Cost per Gb*	*Sequence yield per run*	*Read length*	*Advantages*	*Disadvantages*	*References*
Illumina MiSeq	$502	1.5-2 Gbp	<150 bp	Low error rate (0.8%)	Short reads can not resolve repeats	(Allali et al., 2017)
Illumina NextSeq 2000	$20	330 Gbp	<150 bp	Low error rate (0.26%)	Short reads can not resolve repeats	[1]
Ion Torrent	$1000	20-50 Mbp	~200 bp	Cheaper instrument cost	Short reads can not resolve repeats	(Allali et al., 2017)
PacBio RSII	$400	0.5-1 Gbp	<60 kbp	Produces long reads to resolve repeats, sequences single molecules of DNA	Higher error rate (14%)	[2]
Nanopore MiniION Mk1B	$475	50 Gbp	<2 Mbp	Portable	Higher error rate (15%)	[3]

[1]: https://www.illumina.com/content/dam/illumina/gcs/assembled-assets/marketing-literature/nextseq-1000-2000-spec-sheet-770-2019-030/nextseq-1000-2000-spec-sheet-770-2019-030.pdf

[2]: http://allseq.com/knowledge-bank/sequencing-platforms/ pacific-biosciences/ http://www.pacb.com/blog/ new-chemistry-software-sequel-system-improve-read-length-lower-project-costs/ http://dnatech.genomecenter. ucdavis.edu/wp-content/ uploads/2014/07/Pacbio- Guidelines-SMRTbell- Libraries-v1.0.pdf

[3]: https://nanoporetech .com/products/comparison

1.1.2. Process of *de novo* assembly

De novo genome assembly starts from the raw sequencing data from a sequencing center in the form of FASTQ files, containing the reads generated by the sequencer. Sequencers can construct different libraries of reads based on what is required. Single-end libraries contain reads that each corresponds to a single DNA fragment. The length of each read in this type of library is generally shorter, around 100 to 300 bp. Pair-end libraries contain sequences which only have their end regions sequenced. These sequences flank an area of known approximate length but unknown bases. The advantage of this type of library is that they are longer than single end reads in that each insert length can go up to about 550 bp. This allows for assemblers to anchor neighboring contigs to create scaffolds or longer chains of sequences. Another common type of

reads library is the mate pair library. This library currently has the longest insert length and can also be used to bridge neighbouring contigs. Common lengths for mate pair libraries are 1 kbp, 3 kbp, 5 kbp, 10 kbp and 20 kbp. The approximate length of inserts helps minimize possible issues caused by repeat sequences by telling assemblers how far apart two sequences should be. With that information, any repetitive regions that fall between the two sequences will become apparent and can be resolved (Baker, 2012). There also exist some special libraries that are used only by certain assemblers, for instance the 10x library from 10x Genomics is used specifically by the `Supernova` assembler. For a single assembly, it is encouraged to use a combination of libraries so that all processes can function efficiently. By using multiple DNA libraries, the advantages of each type of library can be used to assemble genomes more efficiently (Liao et al., 2015, Mostovoy et al., 2016, Tan et al., 2018, Nowak et al., 2019, Wallberg et al., 2019, Liu et al., 2020). The general process of genome assembly can be separated into 5 steps: read preprocessing, contig construction, scaffold assembly, gap filling, and genome assembly quality assessment. Figure 1 shows the pipeline of a genome assembly workflow.

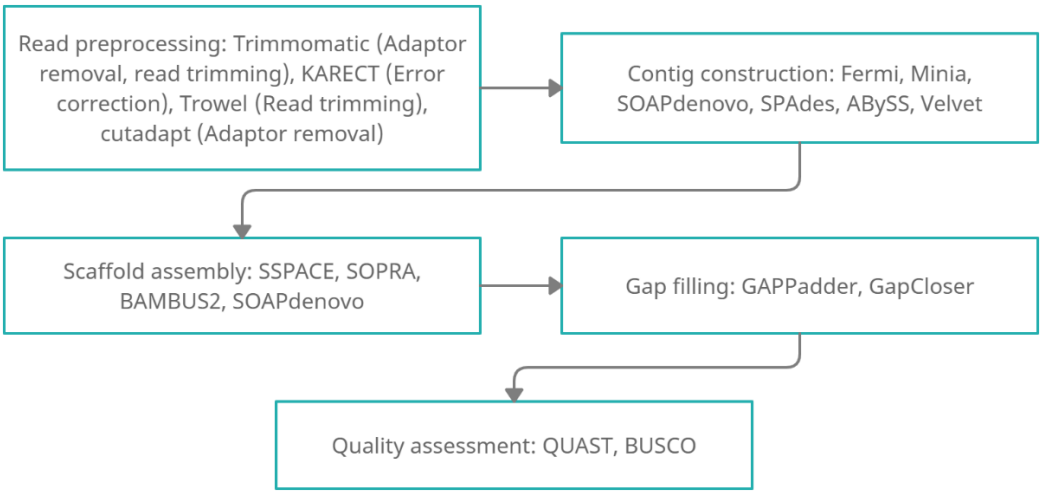


Figure 1. General workflow for genome assembly. Commonly used tools are listed after the corresponding step in the pipeline. For read preprocessing, the purpose of each tool is listed in brackets.

1.1.2.1. Read Preprocessing

The process of *de novo* genome assembly includes read preprocessing, contig construction, scaffold assembly, gap filling, and quality assessment (Sohn & Nam, 2018). Depending on the final goal, an appropriate assembler should be chosen, during which it is important to consider factors such as the size of genome that is being analyzed and the type of sequencing data (i.e. what types of libraries are given, short reads or long reads) (Baker, 2012).

The purpose of read preprocessing is to make sure the data is suitable in quality for assembly. Although sequencers such as the Illumina HiSeq can attain highly accurate reads, there is no guarantee that all reads produced (or all parts of a read) will be completely accurate. Illumina sequencers also introduce adapters during fragment amplification, which include read primers and indexes. If these adapters are not correctly removed from a DNA fragment before sequencing, the adapter sequence will get fully or partially incorporated in the resulting reads, resulting in missed alignments or increased amount mismatches. Sequencing errors, if left unchecked, will lead to erroneous nodes and unnecessary branch paths during contig assembly, leading to lower quality of final assembly product (Th & Ma, 2015). This can be remedied by adding a step for removal of low-quality reads and trimming the low-quality regions of reads from the input FASTQ files using the provided PHRED scores. Each entry can be trimmed based on length restrictions or quality thresholds. To assure the highest quality of assembly, chunks of sequences with low quality scores should be removed to minimize the likelihood of error by using a sliding window. If the average quality score of the base calls in that window dips below a certain point, the tool can cut off all the following bases in the sequence to ensure that the read maintains a sufficient level of accuracy. The location of the cut can vary depending on the length of the window and the threshold that is selected. Some assembly pipelines include an error

correction step. Individual tools like FastQC can be used to give a graphical summary of the reads, and the reads can be trimmed using bash scripts. For trimming the reads and removal of adapters, Trimmomatic (Bolger et al., 2014) is the most common tools used. There also exists standalone tools for adapter removal and read trimming (i.e. cutadapt (Martin, 2011), AdapterRemoval (Lindgreen, 2012)). However, some introduced errors are not recognized as a low-quality region and will escape notice. As such it is then important to have a step to correct the errors introduced by the sequencer. This step generally involves the use of high coverage data to isolate area of error within reads, although different tools have variations on the algorithm. K-spectrum methods use k-mer frequencies to determine error by first dissolving the reads into a set of k-mers. From there the tools diverge in operating principle. Examples of tools that use this method are Lighter (Song et al., 2014), Blue (Greenfield et al., 2014), Trowel (Lim et al., 2014). Another common approach is to use multiple sequence alignments. This approach involves aligning reads that share at least 1 k-mer with the reference read while storing the results in a graph, which is then used to identify the likely location of errors in each reference. The algorithm then repeats this process using every single read as the reference. As a result, this algorithm is thorough, but extremely time-consuming. Examples of tools that utilize this approach are Fiona (Schulz et al., 2014), SGA-EC (Simpson & Durbin, 2012) and KARECT (KAUST assembly read error correction tool (Allam et al., 2015)) which was used in this study.

1.1.2.2. Contig Construction

Following error correction and adaptor removal, the FASTQ files are input into the assembler. The assembler then takes the reads and joins them into short contiguous sequences.

Most assemblers utilize one of two algorithms: the de Bruijn graph (DBG) algorithm or the overlap-layout-consensus (OLC) algorithm.

In the DBG method, reads are broken into small pieces called k-mers. By determining the connections of k-mers, a consensus sequence can be made. This construction of k-mers and connections is called a graph, where the k-mers are termed 'nodes' and the connections are called 'edges'. When two reads partially overlap, they will share some, but not all, of their k-mers, and the assumption can be made that these reads originated from nearby regions on the genome. Therefore, overlapping k-mers can be used to extend the connections between reads and a continuous consensus sequence can be derived (Li et al., 2012). In OLC approach, the overlap between all reads is first calculated. In this stage, the reads are sorted according to the pattern of overlap they display. Moving left and right from the point of overlap, more reads and their overlaps can be incorporated to extend the size of the region. All reads that are captured in this extension are then aligned using multiple sequence alignment. From the alignment generated during layout, the most common base at each position is used to generate a consensus sequence that summarizes the alignment. Each consensus sequence is then considered a contig. Due to the multiple sequence alignment step, the OLC algorithm is more demanding on computer hardware, especially RAM, than the DBG method (Li et al., 2012). These two algorithms also differ in the way that repeat regions are treated. Figure 2 shows the difference in the two approaches. OLC will represent all repeat reads as nodes and make connections between all possible outcomes. DBG will split the sequence reads into shorter k-mers or fragments of length k, and links to flanking regions will the shown as edges. In DBG, the repeat is only represented once. Overall, the OLC algorithm is more suitable for assemblies with lower coverage and long reads, while DBG works better with high coverage and short reads (Li et al., 2012, Sohn & Nam, 2018).

Common contig assemblers include SPAdes (Bankevich et al., 2012), ABySS (Simpson et al., 2009), SOAPdenovo (Luo et al., 2012), Fermi (Li, 2012), and Minia (Chikhi & Rizk, 2013). The first three also complete steps beyond constructing a contig and are separated into a group called pipeline assemblers. SPAdes also assemble contigs into scaffolds, while ABySS and SOAPdenovo can do everything from read procession to gap filling. The latter two are used for contig construction only.

The contig construction step is where the difference between assemblers begin to make an impact. The result is greatly influenced by which algorithm is chosen, as selecting a more stringent algorithm can lead to the assembler making more accurate connections between reads or k-mers, but repetitive regions and heterozygous regions are more likely to be thrown out or ignored (Pop, 2009). On the other hand, choosing a lenient algorithm will decrease the likelihood of discarding repeat regions, but errors in sequencing and genome heterozygosity issues will propagate into the final assembly and decrease accuracy.

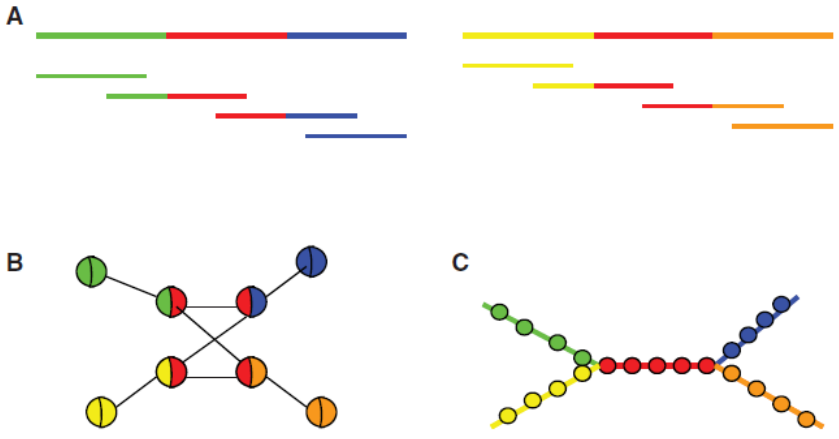


Figure 2. Differences in resolving repeats in overlap-layout-consensus (OLC) and de bruijn graph (DBG) based approach. A) Two genomic regions share a similar sequence region coloured in red. Four sequence reads are mapped to each of the original reference. B) The OLC approach. C) The DBG approach. Figure reprinted from Li et al. 2011.

The main difference between bacterial genome assembly and eukaryotic genome assembly also starts from this contig assembly step. Due to the lack of ploidy, much less repeat content, and being much smaller in size, bacterial genomes are less complex than eukaryotic genomes, and contig construction will require less time and resources. However, the presence of plasmids may hinder this process. Some assembler may try to aggressively force the plasmid sequence into the genome while others might see the plasmid as erroneous sequences and remove them from the assembly (Lantz et al., 2018).

1.1.2.3. Scaffold assembly and gap filling

After contig assembly comes scaffolding, the goal of which is to lengthen the connections between contigs and to find relative positions of each contig in a bigger region (Sohn & Nam, 2018). This is completed by using long insert pair-end libraries or mate-pair libraries. Once two reads are anchored to each other, the unknown bases in between are filled in as 'N's. Ideally, each scaffold will have the length of an entire chromosome and the assembly will have as many scaffolds as chromosomes in the genome. However, in reality, an assembly usually consists of scaffolds ranging in number from thousands or even hundreds of thousands, representing a fragmented genome assembly. The resulting genome assembly is usually presented as a text file containing these scaffold sequences in FASTA format. There exists both pipeline and independent scaffolders, each with a specific niche. Considering the type of input and data available is important when deciding on the scaffolder to use. Pipeline assemblers such as with `SOAPdenovo` and `ABySS` are specifically optimized to take the output from the previous contig creation step and generate scaffolding. Meanwhile, individual scaffolders require more input on the part of the users but allows for more customizability when taking into account the quality of

the output and data. Some of the more well-known stand-alone scaffolders are SSPACE (Boetzer et al., 2011), SOPRA (Dayarian et al., 2010), and Bambus2 (Koren et al., 2011).

However, most scaffolders, such as SSPACE and Supernova (Weisenfeld et al., 2017), will specifically focus on the assembly of contigs that are greater than 1 kbp in length. This means that sometimes shorter contigs under 1 kbp will be ignored and some sequences will end up underrepresented in the final assembly, unless specifically asked for.

A few post-assembly processes can be done to further improve the assembly. For examples, gaps left from the previous step can be filled or shortened. This is usually done by using a single-end library or shorter pair-end libraries (Sohn & Nam, 2018). Examples of tools for this step include GapCloser (Luo et al., 2012) which is part of the SOAPdenovo pipeline, and GAPPadder (Chu et al., 2017), a standalone tool.

1.1.2.4. Genome assembly quality assessment

Once a draft genome assembly is generated, it is important to check the quality of the assembly before moving forward with downstream analyses using the assembly. Quality assessment can help determine how successful the assembly is by examining a set of assembly statistics optionally in comparison to other assemblers.

A variety of parameters can be used for assembly quality assessment. The most common ones to use are the N50 value (along with L50 and N90) (Yandell & Ence, 2012), as well as scaffold lengths and number of scaffolds. The N50 value is defined as the length of the smallest scaffold or contig, with all sequences sorted by length, above which fifty percent of the entire assembly length is represented. The larger the N50, the better the genome assembly by indicating longer scaffolders and less fragmentation. Therefore, the N50 can only provide a general

guideline and can be used to quickly compare the efficacy of different assemblers. However, this statistic can change depending on how stringent the assembler is. Other similar parameters include N90 value, which is the length of the smallest scaffold or contig above which ninety percent of the entire assembly length is represented, and L50, which gives the smallest number of contigs or scaffolds that is required for fifty percent of the assembly to be represented. Some tools will also output values such as NA50 and NGA50, both requiring the presence of a reference genome. NA50 is determined by aligning contigs to the reference genome and splitting the aligned contigs into blocks whenever a mis-assembly is detected. The N50 statistic for these blocks is then calculated, which is then labeled as NA50. NGA50 is simply the NA50 value but using 50% of the reference genome size instead of the assembly size. When comparing assemblies of different samples, it is best to use the NGA50 value instead of the NA50 value. In order for these values to be meaningful when comparing the performance of different assemblers, the assembly size must be equal. Other statistics that can be used to measure the accuracy of an assembler are average contig or scaffold length, with larger lengths indicating a more complete assembly. Finally, the draft genome can be compared to that of a closely related organism. This comparison identifies similarities of gene sequences in both species and simplifies the process of annotating the new genome. If a crucial gene is missing from the new assembly but was present in the closely related genome, it may be assumed that the draft genome is incomplete.

The most common tool for quality assessment is `Quality Assessment Tool for Genome Assemblies` (`QUAST`) (Gurevich et al., 2013). This tool takes the genome assembly in FASTA format as the input and outputs a list of statistics of interest, including number of contigs, total length, GC content, N50, L50, size of the largest contig and more. Some

pipeline assemblers have their own assessment tool built in. For instance, `Supernova` outputs a file that lists certain qualities of the assembly (https://support.10xgenomics.com/de-novo-assembly/software/pipelines/latest/output/asm-stats).

In the future, it might be useful to consider how to implement more than one type of sequencing data from different sequencing platforms into the *de novo* assembly process. For example, for large eukaryotic genomes, having both an Illumina short pair-end reads as well as long reads will be effective in reducing the number of fragments caused by difficulty in assembling the repeat regions in the genome as well as ensuring a high sequence accuracy by using short-reads. In fact, some groups are already implementing this idea when producing their draft genomes (Tan et al., 2018, Wallberg et al., 2019).

An instance of using hybrid assembly is the generation of the draft genome for honeybee, *Apis mellifera*. To generate a higher quality reference genome of the honeybee, Wallberg et al. (2018) compared between a variety of approaches to assemble the genome, including PacBio long reads libraries, 10X Chromium linked-reads, BioNano technology, and Hi-C. PacBio reads produce contigs of the highest contiguity and were then supplemented by 10X data. This produced contigs, the longest of which were 40 times longer than those from individual platforms. Scaffolding of the contigs were done with BioNano's optical mapping technology, which identifies specific motifs on DNA fragments that can be used to order the fragments (Teague et al., 2010, Lam et al., 2012), and Hi-C data (now discontinued), which used chromatin interactions to position contigs (Burton et al., 2013, Kaplan & Dekker, 2013). After gap filling and final clean up, each chromosome is represented by a single scaffold. They then aligned sequences from the new draft genome (Amel_HAv3) to the 16 chromosomes of honeybee and found that 10% more sequences were anchored in comparison to the previous version

(Amel_4.5). The 16.4 Mbp of previously unplaced data were now aligned, and noticeable improvements were made in the repetitive regions, including centromeres and telomeres (Wallberg et al., 2019).

1.2. The types and sources of genetic heterogeneity and their impact on the quality of genome assembly

1.2.1 Types and sources of genetic heterogeneity

Genetic heterogeneity is defined as having similar phenotypes resulting from different genotypes, due to allelic heterogeneity, locus heterogeneity, or both. Allelic heterogeneity is defined as having two or more alleles for the same genomic locus in an individual or a population (Milholland et al., 2017). This is due to a gene usually spanning many kilobases in length and a variation at any position within that gene can lead to the rise of a new allele. In diploids such as humans, carriers of recessive hereditary diseases have one faulty allele on one chromosome and a normal allele on the homologous chromosome without showing a change in the phenotype. Locus heterogeneity refers to a disease or trait that manifests due to mutations or variations in different genes. An example of this is retinitis pigmentosa, which is governed by 16 genes, and any of the gene may develop a mutation resulting in the deterioration of vision (Milholland et al., 2017)). However, to inspect the fundamental processes behind *de novo* genome assembly, we should look into the more broader aspect of sequence heterogeneity. For the purposes of this analysis, I will be focusing on the impact of sequence heterozygosity on *de novo* genome assembly. I define sequence heterozygosity as one of the two following occurrences.

One is the variation existing between homologous chromosomes at the corresponding positions (Griffiths et al., 2000, Milholland et al., 2017). For diploid and polyploid organisms, homologous chromosomes may have heterozygous alleles for a gene due to random mutations, an issue which is not present in haploid organisms. With sequence data generated by most of current sequencing platforms, sequence reads have no way to be localized to a specific region in the chromosome, so the assembler will assign reads based on overlap (Khan et al., 2018, Weisenfeld et al., 2017). As a result, some reads may be misaligned into other regions of the genome, or multiple redundant contigs may be generated by the assembler. This causes many unnecessary breaks during contig extension, which will result in many short contigs as opposed to a long contig (Milholland et al., 2017).

Another is the accumulation of somatic mutations throughout an organism's lifetime. As an individual grows, somatic mutations may occur in the chromosomes within each cell. More often than not, these mutations are fixed by DNA repair mechanisms (Li & Heyer, 2008). However, there are some mutations that escape repair and manage to be fixed in the genome of that cell. These mutations will then propagate by passing from the original cell to its progeny via cell division (Basturea, 2018). If these cells are sampled along with other cells without this variation for making the DNA to be used for genome sequencing, the disagreement of base calls at the position of the variation will affect the extension of contigs. Somatic mutations differ from germline mutations in that not all cells in the organism will contain the same mutations. A certain mutation that occurs in one cell can be missing in another. On the other hand, germline mutations are present in the gametes (i.e., sperm and/or egg) before development and should be present in all cells of the individual after development. Mutations such as single nucleotide polymorphisms (SNPs) and insertion or deletion of individual nucleotides within a gene can

cause a new allele to arise. SNPs are common substitutions of a base that is observed in more than 1 percent of the population (Griffiths et al., 2000). Other forms of mutation include insertions and deletions that are within 50 bp in size (termed as indels) and larger sequence variations classified under structural variants, which come from a variety of mechanisms such as transposable element insertion, copy number variation, inversions, and translocations. In humans, SNPs and indels are most prominent by number. A study published in 2015 by the 1000 Genomes Project analyzed over 2,500 genomes spanning across 26 populations (Auton et al., 2015). They conclude that a typical human genome contains around 4.30 million SNPs and indel sites, while only 2,100 to 2,500 SVs are present, although the latter collectively impacts more sequences by length. Of the structural variations, transposable element insertions occur most frequently, with around 1094 being present in a typical genome. This count includes insertions if Alu, L1 and SVA (SINE-VNTR-Alus) transposable elements. This is followed by large insertions (~1000/genome), copy number variants (~160/genome), and inversions (~10/genome).

1.2.2 The role of homology-mediated DNA repair in somatic mutation

DNA replication introduces somatic variants when defects in DNA replication machinery fail to recognize a mismatched base or double-strand break (DSB) (Chatterjee & Walker, 2017). This could be in the form of an indel or a substitution in DNA sequence. Some of these replications errors and additional errors from DNA damaging factors can be repaired using the homology-mediated DNA repair in the diploid genomes (Li & Tye, 2011, Li & Heyer, 2008, Rodgers & Mcvey, 2015). When a double-strand break happens in the genome, the broken ends are trimmed away to leave a single-strand 3’ end. A homologous template is annealed and used to synthesize the missing bases onto this single-strand DNA. However, this procedure gives rise

to potential opportunities for mutagenesis and, as a result, sequence heterozygosity (Rodgers & Mcvey, 2015).

When homologous chromosomes are not present, a more error-prone approach of non-homologous end joining (NHEJ) is used to correct DSBs (Iyama & Wilson, 2013, Malzhan et al., 2017). Due to this difference in repair mechanisms, the accumulation of somatic variants in the haploid samples is expected to be higher compared to diploid samples, thus leading to a lower genome assembly quality. In this study, a comparison between the old and young samples of each sex provided an opportunity to determine how this specific repair mechanism affects quality of assembly. The observation that the difference in the genome assembly quality between old and young samples for males is greater than that in female samples will confirm the action of the homology-based DNA repair mechanisms.

1.2.3. Previous work on the impact of sequence heterozygosity on genome assembly quality

Very few studies have explored the effect of sequence heterozygosity on quality of *de novo* assembly. It is unknown whether one specific category of variation will affect the quality more than others. Diploid and polyploid organisms should, in theory, have a lower assembly quality than haploid organism due to the presence of different alleles on homologous chromosomes (Liang et al., 1998). Accumulation of somatic mutations in the genome is another cause of heterozygosity and will theoretically affect haploid organisms more than for diploid genome due to the lack of a DNA repair mechanisms which rely on homologous recombination. If a strand of DNA is nicked in both strands (i.e., a double-strand break), a diploid organism can use the other homologous chromosomes to repair damages. This is not an option for haploid organisms. These mutations, despite existing at variable frequencies depending on the tissue and timing of the mutations, can lead to problems during the contig construction step where reads or

k-mers are joined by overlap. If the flanking regions of the sequences are similar and the heterozygous alleles spans longer than the length of a read, then assemblers will be unable to distinguish which allele is the "correct" allele and will form an unresolved branch. Some assemblers will leave the branch as is, while others will attempt to condense the branch, either by looking at which variant has higher coverage or by combining the two paths into a consensus sequence, representing bases from both reads. Neither of these results will give a true representation of the genome.

A previous study has examined on the effect of sequence heterozygosity on *de novo* assembly of a fish genome by using DNA from a parthenogenic haploid larvae of the Chinook Salmon in comparison with that of diploid females (Iwasaki et al., 2016). In the study, the parthenogenic larvae was generated by inseminating eggs with UV irradiated sperms, leading to the organisms being haploid and grown to the larvae stage. The DNA sequencing was performed using the Ion Torrent platform, and assembly was performed using both an OLC assembler (`Newbler 2.9`) and a DBG assembler (`CLC Genomics Workbench`). They discovered that the haploid larvae produced assemblies of a higher quality than that of the diploid samples based on 8 metrics including the number of contigs, number of scaffolds, average contig length, average scaffold length, N50 contig length, N50 scaffold length, largest contig length and largest scaffold length. Out of these 16 comparisons (8 metrics x 2 assemblers), the haploid assembly only outranks the diploid assembly in 3 (number of scaffolds for DBG assembler, largest scaffold length for both OLC and DBG assembler). As the first of its kind, the authors conclude that haploid assemblies resulted in higher quality assemblies in comparison to the diploid counterpart. However, the study only managed to compare between the heterogeneity originating from ploidy (Iwasaki et al., 2016), and it did not examine the effect of somatic mutations, which

likely accumulated more in the haploid female than the diploid larvae, on the quality of assembly. Another caveat of this study is the uses artificial means involving the use of UV irradiated sperm to produce haploid samples, in which one cannot rule out the possibility of DNA contamination from the remanence of the treated sperm. A better research design or model is needed to quantify the difference in quality of the *de novo* assemblies when different types of sequence heterozygosity is present.

A newer study from 2019 analyzed the difference of de novo assembly between diploid and haploid samples from the Hymenoptera order (Yahav & Privman, 2019). The organism being studied was *Cataglyphis drusus*, a species of ants. Two haploid male samples and 2 worker diploid samples were prepared for genome assembly. *De novo* assembly was done with `SOApdenovo2` and `SPAdes`, both using the DBG algorithm to complete contig assembly. Similar to the previous study, it was discovered that haploid samples had higher N50 contig and scaffold sizes when compared to the diploid samples, indicating that reads from haploid genomes produce a higher quality assembly. This study also did not consider that impact of somatic mutations on the quality of *de novo* genome assembly, something that we aim to remedy.

1.2.4. Using *Xylocopa virginica* as a model organism

Xylocopa virginica, also known as the Eastern carpenter bee, is commonly found in eastern United States and Canada. Like all other species belonging to the order Hymenoptera, *X. virginica* undergoes sex determination through a process called haplodiploidy (King, 2007). In this system, the gender is determined by the ploidy of the genome. An egg laid by a female bee can either be fertilized by a male or unfertilized. An unfertilized egg will develop into a haploid male, while a fertilized egg will develop into a diploid female. Haplodiploidy has an advantage in that it can quickly remove all recessive lethal and deleterious alleles from the population, as

males carrying these alleles will be eliminated, unable to pass these mutations to offspring (White, 1984). This unique way of sex determination is found in all ants, wasps, and bees, which make them suitable model organisms for studying the effect of ploidy derived gene heterogeneity on *de novo* assembly, as natural haploid genomic DNA can be easily obtained from the males. Further, such models permit examining the impact of role of homology-based DNA repair on somatic mutation rate, as well as the impact of somatic mutation on genome assembly quality by comparing the mutation rate and spectrum between male and female bees over different time spans in the organisms.

1.3. Research Rationale and Objectives

The capacity to generating a high-quality genome assembly is critical for biologists to be able to take advantage of non-model organisms. As more and more genomic data is being produced, it is not only important to improve the efficiency of analysis, but to also evaluate the quality of the data. By using organisms with haplodiploidy as models, we can investigate the impact of sequence heterozygosity on the quality of *de novo* genome assembly, which can provide insights for improving the DNA assembly algorithms. To this end, we propose the use of *Xylocopa virginica* as the model organism. Using male samples as haploids and female sample as diploids, we can more directly compare the effect of sequence heterozygosity on assembly quality while limiting the effect of other factors on the quality. We hypothesize that sequence heterozygosity due to ploidy will negatively affect the assembly quality. Diploid genomes introduce heterogeneity through heterozygous alleles, which complicates the contig extension process. Thus, we expect that the use of diploid samples will result in a lower quality assembly than the haploid counterpart. We also hypothesize that the accumulation of somatic

variants will decrease the quality of the assembly. As an organism ages, its cells are exposed to all forms of chemical and physical insults from both internal and external factors. When DNA gets damaged, repair mechanisms try to correct any mistakes that get detected. However, if a mistake is not detected due to a faulty mechanism or the lack of a proper repair mechanism, then the variant will remain within the genome. Further replication of the cell will only serve to propagate the variant to its daughter cells. Theoretically, as time goes on and the organism ages, these cells will accumulate these mistakes and derive from the original genome sequence (Basturea, 2018). When these slight differences grow in number, so does the chance that they get captured in DNA samples. On the other hand, in non-dividing cell, mismatch repair and homologous recombination are both either low in activity or show no functional activity at all, which leads to accumulation of physical breaks and damaged DNA (Iyama & Wilson, 2013). As a result, when the samples get sequenced and assembled, the algorithm might misinterpret as the divergence in the consensus sequence and introduce more breaks in the assembly, thus lowering quality (Khan et al., 2018). In this case, it would make sense for the worn organisms will have lower quality in comparison to unworn organisms due to the accumulation of somatic variants over time. In addition, we should see that the decrease in quality is greater in haploid than diploid samples due to the lack of homologous chromosome DNA repair mechanism. Based on the results of this experiment, we can also determine the degree to which each of the two factors affect quality. If the difference between haploid and diploid sample assembly qualities are greater than the difference between worn and unworn qualities, we can deduce that somatic variants accounts for a smaller effect on assembly quality than ploidy. If the opposite is true, then it is likely that ploidy affects the assembly quality less than the accumulation of somatic variants does. Furthermore, analysis of the additional variants found in the worn male can provide

insights regarding the specific types of DNA damages handled by the homology-based DNA repair mechanisms.

CHAPTER 2: Materials and Methods

2.1 Sample preparation

Xylocopa virginica in the Niagara region of Southern Ontario were captured using peso-cup-traps. Captured bees (2 males and 3 females) were placed on ice for sedation and transferred to a -80°C freezer after observation. Unworn bees were captured immediately after emergence. The classification of worn versus unworn individuals was based on the degree of wear on their wings; unworn individuals have a wing wear rank of 0, while worn bees have a wing wear of rank of 5. Wing wear was given a classification based on the completeness of the wing margin and divided into 6 categories, ranging from 0 (complete margin) to 5 (margin showing major cuts). Ranking of wing wear was done according to a visual guide described by Muller and Wolf-Muller (1993). Both wings for each sample were considered and the results were averaged (Mueller & Wolf-Mueller, 1993).

Extraction of DNA for the bees was done according to the steps outlined in DNA Extraction of Single Insects from the 10X Genomics website (https://support.10xgenomics.com/permalink/7HBJeZucc80CwkMAmA4oQ2). Briefly, a bee sample was homogenized using a razor blade in a solution mixed from 600 µl of lysis buffer, 40 µl of 10% SDS, and 100 µl Proteinase K solution. DNA was precipitated in 5 M NaCl solution and subject to centrifugation at 4°C.

Bee sample collection was performed by students in the laboratories of Dr. Miriam Richard and DNA extraction was performed by Dr. Adonis Skandalis.

2.2 DNA Sequencing

Standard Illumina sequencing using the HiSeq X sequencer for this study was outsourced to Génome Québec Innovation Center. DNA libraries were prepared according to the TruSeq DNA PCR-Free protocol. Compressed FASTQ files in pairs (read 1 and 2 of pair-end sequencing) of the whole genome sequencing for each of the 4 samples were downloaded from the Génome Québec server to Dr. Liang's research space on Compute Canada High Performance Computing (HPC) systems for analysis.

10X Linked-reads sequencing was performed by The Center for Applied Genomics using the Chromium instrument and the Illumina HiSeq X sequencer for one female sample. The data files consisting of a trio of compressed FASTQ files (read 1, read 2 and index file) were downloaded using the tcag-client app to Compute Canada servers as for the other sequencing data.

2.3 *De novo* genome assembly

2.3.1 Input datasets

A total of 5 whole genome sequencing datasets including one Illumina PE library each for an unworn male, an unworn female, a worn male, and a worn female, plus one 10X Genomics linked read library for a unworn female. All PE libraries have an insert size of ~450 bp and an approximate read length of 151 bp before pre-assembly processing. The amounts of sequence data and coverage depth for these five samples are summarized in Table 2. The coverage was estimated using the following equation, in which the value for the length of

genome was based on the estimated genome size of 228 Mbp (haploid) by Kmergenie (Chikhi & Medvedev, 2014) estimate:

$$Coverage = \frac{\#\ of\ reads * length\ of\ read}{length\ of\ genome}$$

Table 2: Summary statistics of genome sequencing

Sample library	Input Quantity (ng)	Number of bases	Number of reads	Length of read	Coverage*
Worn Male	4,761.50	30,805,038,578	102,003,439	151	134.91x
Worn Female	2,267.30	34,716,667,304	114,955,852	151	152.04x
Unworn Male	2,290.50	33,395,056,414	110,579,657	151	146.25x
Unworn Female	1,753.30	37,336,205,338	123,629,819	151	163.51x
Unworn Female (10X)	Not Specified	54,320,510,480	359,738,480	128 + 16 nt barcode	250.95x

*Coverage was calculated using average haploid genome size estimate (228 Mbp)

2.3.2 Sequence pre-assembly processing

Before the reads were used for assembling, they were subject to preprocessing using `Trimmomatic-0.36` (Bolger et al., 2014). This step serves to remove sequencing adapters (R1: AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC, R2: AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT) as well as sections of reads that fall below a quality score threshold. The sliding window approach was taken when examining the quality score. The size of the window was 4 bp and an average quality score of 15 and above was required for the read not to be trimmed. Any bases at the beginning or end of the read that had a quality score of less than 3 were trimmed. After all processing, any reads that were less than 36 bps in length were dropped.

In the second pre-assembly processing step, the reads were corrected for likely sequencing errors. This was done using `KARECT (KAUST Assembly Read Error Correction Tool)` (Allam et al., 2015). The most common errors for Illumina sequenced reads are substitutions, thus we went with the Hamming distance for "-matchtype". The detailed commands and options for the above two steps are provided in Appendix A.

2.3.3 *De novo* assembly using Soapdenovo2 with Illumina PE library

2.3.3.1 Preparation of sequence reads with the same coverage across samples.

The 4 regular Illumina sequence data sets have variable sequence amounts as shown in Table 2. To eliminate this variability in comparison for the genome assembly quality, reads were trimmed so that the same coverage was used for all samples, and this was achieved by trimming sequence amounts of 3 samples to that of the sample with the lowest coverage, i.e., the worn male assembly at 76x coverage (Table 2).

2.3.3.2 *De novo* genome assembly using Soapdenovo2

A *de novo* genome assembly was performed for all samples with Illumina PE sequencing data. All the steps in the `SOAPdenovo2` pipeline (Luo et al., 2012) were run sequentially for all assemblies. Input data was specified in a separate configuration file, and parameters for assembly were kept at default values using 50GB of memory and 32 CPU cores (Appendix A, Table S1). The assembling was run on Compute Canada high-performance servers. Detailed commands and parameter settings are provided in Appendix A.

2.3.4 *De novo* assembly of 10X linked-reads using Supernova

The 10X sequence library was used for assembly without any preprocessing to avoid accidentally removing barcode sequences. The assembly was run using `Supernova v2.1`(Weisenfeld et al., 2017) with the default setting except for the "--maxreads" flag, which determines maximal number of reads to be used in the assembly on a random selection basis in case where excessive reads are available. For analysis of the effect of coverage on `Supernova` assembly quality, the value of the "--maxreads" flag was adjusted to achieve the expected coverages ranging from 28x to 250x. For the assembly using all reads, a setting of "--maxreads=all" and "--accept-extreme-coverages" were used (see Appendix A for the detailed command and parameter settings). Assembly metrices for a resulting assembly were obtained from the report.txt output file of `Supernova` runs, while the non-gapped genome length was measured with `fatools`, a tool developed at Dr. Liang's lab (https://github.com/pliang64/fatools).

2.3.5 Generation of a reference genome sequence for *X. virginica*

2.3.5.1 Post-assembly processing

Gap filling was done for all Illumina sequenced pair-end assemblies with the built-in gap closing function (`GapCloser`) of `SOAPdenovo2` (Luo et al., 2012). The unworn male assembly, which had the highest assembly quality among all samples, was chosen to undergo further processing to generate a reference genome. Two rounds of gap filling were done with `GAPPadder` (Chu et al., 2017) and `GapCloser` using reads from both unworn male and unworn female samples, and the final product was used as the reference genome for all downstream analyses.

2.3.5.2 Creating a randomized, linear pseudogenome

The FASTA entries from the above reference assembly were randomly selected and combined into a single fasta entry to be used as a pseudo linear reference genome to determine relative location of variants, repeats, and breakpoints more easily. The reference assembly was first filtered to keep only scaffolds of greater than 500 bp in length. Randomization of the scaffold position in the genome was then done by collecting entry IDs using a perl script, `list2randomIDset.pl` (Appendix A). The scaffolds were then concatenated to a linearized genome assembly which combines all scaffolds as one continuous FASTA entry to be the pseudo linear reference genome.

2.4 Detection of variants

For identification of SNPs and indels, the sequence reads of worn male and females were mapped to the haploid reference sequence using `BWA-MEM` (`Burrows-Wheeler Aligner`) (Li & Durbin, 2010) and variants were identified using the `HaplotypeCaller` module for germline variants and `Mutect2` module for somatic variants included in the `GATK` (Genome Analysis ToolKit) package (McKenna et al., 2010). Variants from output of the two modules were combined to generate a non-redundant final list using `vcftools` (Danecek et al., 2011).

For identification of structural variants (insertions and deletions >=50 bp) in the 4 genomes other than the unworn male, `pblat` (Wang & Kong, 2019) was used to align assemblies to the reference genome. Mismatches and indels, as well as the ends of the scaffolds representing breakpoints of the individual genome assemblies, were identified for each assembly using in-house Perl scripts (Appendix B).

2.5 Comparative assessment of *de novo* genome assembly quality

Comparisons of assembly metrics were performed to determine the order of quality of the assemblies from samples representing different ploidy levels and ages. The quality of assembly was determined by looking at a combination of N50 value, total number of bases with and without gaps, number of Ns per 100 kbp, as well as length and number of scaffolds. All genome assemblies were filtered to only keep scaffolds that were greater than 500 bp in length and their assembly metrices were obtained using an in-house tool, `fatools`, and `QUAST` (Appendix A).

To compare the quality between assemblies, we would usually use the N50 size, which is a measure of the continuity of the assembly. However, it is subject to change according to the number of scaffolds that is selected as a cut off. In our case, we used a cut off threshold of 500 bp, which ensures that the scaffolds in the assembly are contigs that have at least 5 reads joined together, and not simply single reads that did not join to any contig, possibly due to sequencing errors. This minimizes the risk of having incorrect bases being added into the assembly. Similarly, average length of scaffolds can also be used but suffers from the same pitfalls and biases as N50 size.

2.6 Analysis of effect of sequence coverage on Supernova assembly quality

Using the same 10X Linked-reads input file, we selected a variety of coverages spanning from 28x to 250x to generate assembly. The number of reads required as input for `Supernova` (`v2.1`) was calculated based the intended coverage, the length of a read and the estimated size of the genome. The reads were sampled randomly from the entire original sequence file. An addition trial was added by using all 359,738,480 reads (263X coverage). The quality of

assembly was compared across all assemblies by plotting the N50 values, average scaffold length, and non-gapped genome length against the coverages.

2.7 Determining correlation between variant density and breakpoints

The scaffolds of individual genomes other than the unworn male genome, which was used as the reference genome, were mapped to the linearized pseudo reference genome to identify their location in this reference genome. The location of variants identified from 2.4 on scaffolds were converted to position on the linearized pseudo reference by using the position given on the VCF file and the location of the scaffold in relation to the pseudo genome. Breakpoints in the assemblies (identified as the ends of each scaffold) were detected and were converted to the pseudo genome positions in a similar method to variants. A sliding window with a size of 2 Mbp and step size of 500 kbp was used to generate a density plot of variants and breakpoints for each of the four genomes (pair-end unworn male, pair-end unworn female, pair-end worn male, pair-end worn female) in reference to the linearized reference genome. The Pearson correlation coefficient between density of breakpoints and density of variants for each sample was calculated using Microsoft Excel.

2.8 Determining correlation between density of various repeats and breakpoints

`RepeatMasker` (http://repeatmasker.org) was used to annotate all repetitive sequences in the reference genome, in which `RepeatModeler` (http://repeatmasker.org/RepeatModeler) was first used to generate a list of consensus repetitive element sequences based on the assembly. These sequences are then compared to known repeats in the `RepeatMasker` library to

categorize them and used as the repeat library to run `RepeatMasker`. For the top 5 most prominent types of repeats, a table of densities was generated using a sliding window of 500 kbp along the linearized pseudo reference to count the number of repeats of a certain type. The Pearson correlation coefficient between density of breakpoints and density of each type of repeat was calculated for each type.

CHAPTER 3: Results

To examine the effect of sequence heterozygosity on the quality of *de novo* genome assembly, we included in our study four different samples with two genome ploidy levels (haploid for males and diploid for females) and two age groups (unworn and worn bees for each sex). The comparison between the male and female unworn bee genome assemblies would allow us to examine the impact sequence heterozygosity contributed by genome ploidy; while the comparison of that between the young (unworn) and old (worn) bee genome in the same sex would permit us to assess the impact of sequence heterozygosity contributed from somatic mutations. Furthermore, the degree of the differences between the two age groups for male and female bees may provide insight regarding the role of DNA repair mechanism in somatic mutations, more specific the role of homology-based DNA repair.

For this purpose, all four samples were sequenced using the Illumina standard PE sequencing protocol at coverages around 90x (83 to 104) (Table 2) and performed *de novo* genome assembly using `SOAPdenovo2`. To eliminate the variation associated with the coverage depth, in addition to assembly with all available reads for each sample, we also trimmed the read input data size for three of the samples (Unworn male, unworn female, worn female) to match the sequence coverage of the worn male, which has the lowest sequence coverage, such that the same sequence coverage was used for all 4 samples in the comparison.

3.1. Effect of sequencing coverage on genome assembly quality

3.1.a Effect of sequencing coverage on genome assembly quality with standard Illumina pair-end reads

Upon observation, it was apparent that the coverages of the samples were not the same and in fact differed by quite a bit (~70x coverage difference between highest and lowest). The lowest coverage was that of the worn male sample with 76x coverage and the greatest coverage was that of the unworn female sample with 146.25x. To remove any bias in the assembly quality that is contributed by the difference in coverage, a coverage normalization step was added. The two male genomes were also assembled using half of the estimated female genome size to account for the difference in ploidy. Furthermore, we assembled each of the 4 samples with different coverages for input reads (5x to 40x).

Figure 3 shows the relationship of coverage and N50 values for each of the 4 assemblies. Supplementary table S2 shows the numerical values of the N50s. In comparison to the female samples, male samples have a more distinct trend in assembly quality, that is, as coverage increases, the assembly quality also showed steady increase within the entire coverage range tested (5X to 40X) (Figure 3). It is interesting to notice the significant difference between the unworn and worn males: while both showing an up-trend with increase sequencing coverage, the unworn male sample showed a much larger benefit from more sequencing than the worn male. More specifically, at 5X coverage, the N50 are 890 bp and 575 bp for unworn and worn male, respectively, with the unworn male being ~1.5 times better than the worn male, but at 40X coverage, the difference becomes ~10 times (21.3 kbp vs. 2.2 kbp) (Figure 3A). With all available reads for a coverage at ~150X, the N50 increases to 207 kbp for the unworn male, and for worn male at the highest coverage of ~130X, the N50 reaches ~80 kbp (Figure 3B). This

result seems to suggest that the benefit of more sequences is likely damped by the increase sequence heterogeneity from somatic mutations in the worm male. In comparison with the situation in the males, for the female samples, quality of genome assembly increases between 5X to 20X and then it started to drop beyond 20X (Figure 3A), suggesting that an increase in coverage does not equal to an increase in quality of the assembly, likely indicating an increasing negative impact combining between the increased sequence heterogeneity from diploid and somatic mutations. As a summary, among the four samples, the unworn male shows the biggest genome quality increase from 5X to 40X sequence coverage, followed by the worn male, unworn female, and worn female assemblies.

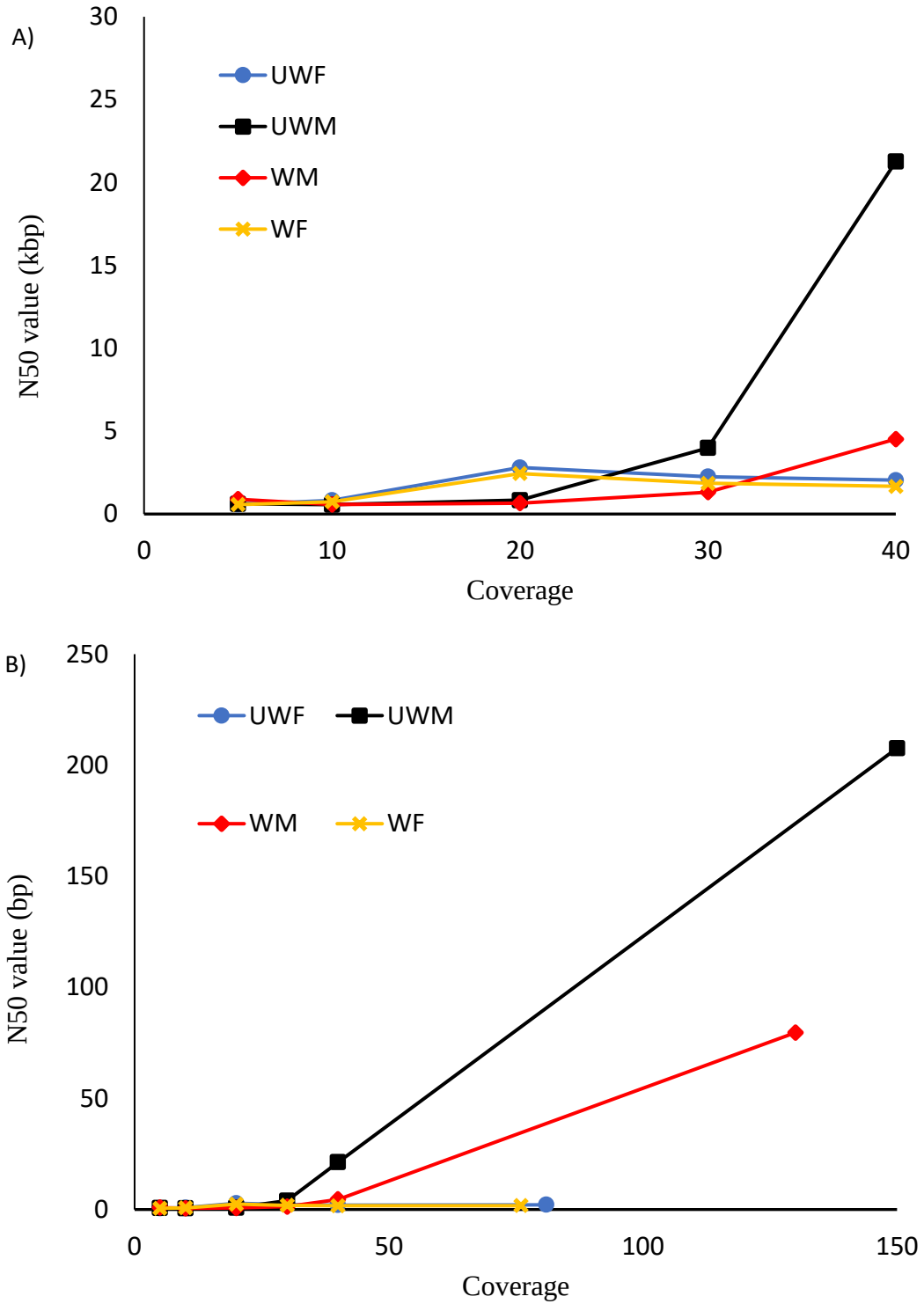


Figure 3. Effect of sequencing coverage on assembly quality. Unworn male (black), Worn male (red), Unworn female (blue), and Worn female (yellow) are shown with A) 5X, 10X, 20X, 30X, and 40X coverages and B) all reads. Coverage calculations were based on estimated genome sizes 456 Mbp (diploid/female) and 228 Mbp (haploid/male).

3.1.b Effect of sequencing coverage on genome assembly quality with linked Illumina reads

To examine the effect of the sequencing coverage on genome assembly quality, 10X linked-reads assembly using `Supernova v2.1` was performed using different sequencing coverages ranging from 28x to 263x. Assembly metrics were retrieved from the output result.txt from the `Supernova` runs. `Supernova` coverage was calculated using the estimated genome size provided by the `Supernova` run output. This measurement is slightly different than the genome length used to calculate number of input reads and resulted in `Supernova` giving a different coverage to what was expected. Table 3 shows the assembly metrics for the `Supernova` assemblies obtained at different coverages, calculated based on the estimated genome length of 456 Mbp by random sampling from the same FASTQ file using the built-in --maxreads flag.

With a step size of 5x, between 28x and 63x coverage, the scaffold N50 increases drastically with the increase of sequencing coverage. However, from 63x up until 118x, the quality decreases from the peak at 63x coverage. The assembly quality improves again as the coverage increases from 250x to 263x. However, at 263 coverage, the contig N50 and number of long scaffolds do not measure up to the assembly at 63x coverage, while the scaffold N50 size improves (Figure4).

The estimated genome length by `Supernova` follows a similar trend to N50 lengths, starting from 392.77 Mbp for a diploid genome at 28x coverage and increasing up until a peak value of 418 Mbp at 63x coverage. The length then decreases back to 392 Mbp at a coverage of 118x. At 263x, the genome size is ~433 Mbp, the highest value of all `Supernova` assemblies.

In summary, with 10X linked-reads for the diploid genome, sequencing at 63X coverage seems to offer the best genome quality, after which the increase sequencing coverage does not seem to offer much benefit if not drawbacks besides higher sequencing cost.

Table 3. Supernova assembly qualities at different coverage

Coverage	N50 values (kbp)	# of scaffolds (x1000)	Contig N50 (kbp)	Est. genome size (Mbp)	Non-gapped genome length (Mbp)
28	14.73	2.06	12.4	392.77	167.37
33	19.3	3.94	15.6	406.1	179.81
38	28.21	4.79	21.05	415.74	184.61
43	40.25	4.78	29.4	418.81	187.60
48	45.92	4.43	39.57	415.13	190.99
53	47.33	4.26	41.49	414.3	194.81
58	48.74	4.04	43.1	411.14	195.81
63	49.02	4.06	43.4	408.22	195.92
68	47.65	4.28	42.49	408.52	196.64
73	46	4.47	40.22	404.13	195.82
78	44.07	4.59	37.41	402.08	194.56
83	41.26	4.72	32.42	402.78	194.20
88	38.86	4.71	29.21	400.09	192.34
93	35.54	4.75	26.77	398.72	191.58
98	33.29	4.76	24.7	400.49	190.43
103	31.92	4.67	23.16	398.37	189.78
108	31.79	4.54	21.89	395.91	188.65
113	30.69	4.55	21.33	396.44	188.01
118	31.16	4.43	21.24	392	187.96
128	36.09	4.04	21.57	397.9	186.26
138	36.95	4.16	21.55	398.88	185.52
150	41.2	4.18	23.28	403.87	185.51
175	46.91	4.18	25.42	406.76	185.54
200	49.03	4.31	27.33	416.27	187.54
225	52.3	4.18	28.97	419.76	189.38
250	54.26	4.2	30.75	428.76	193.14

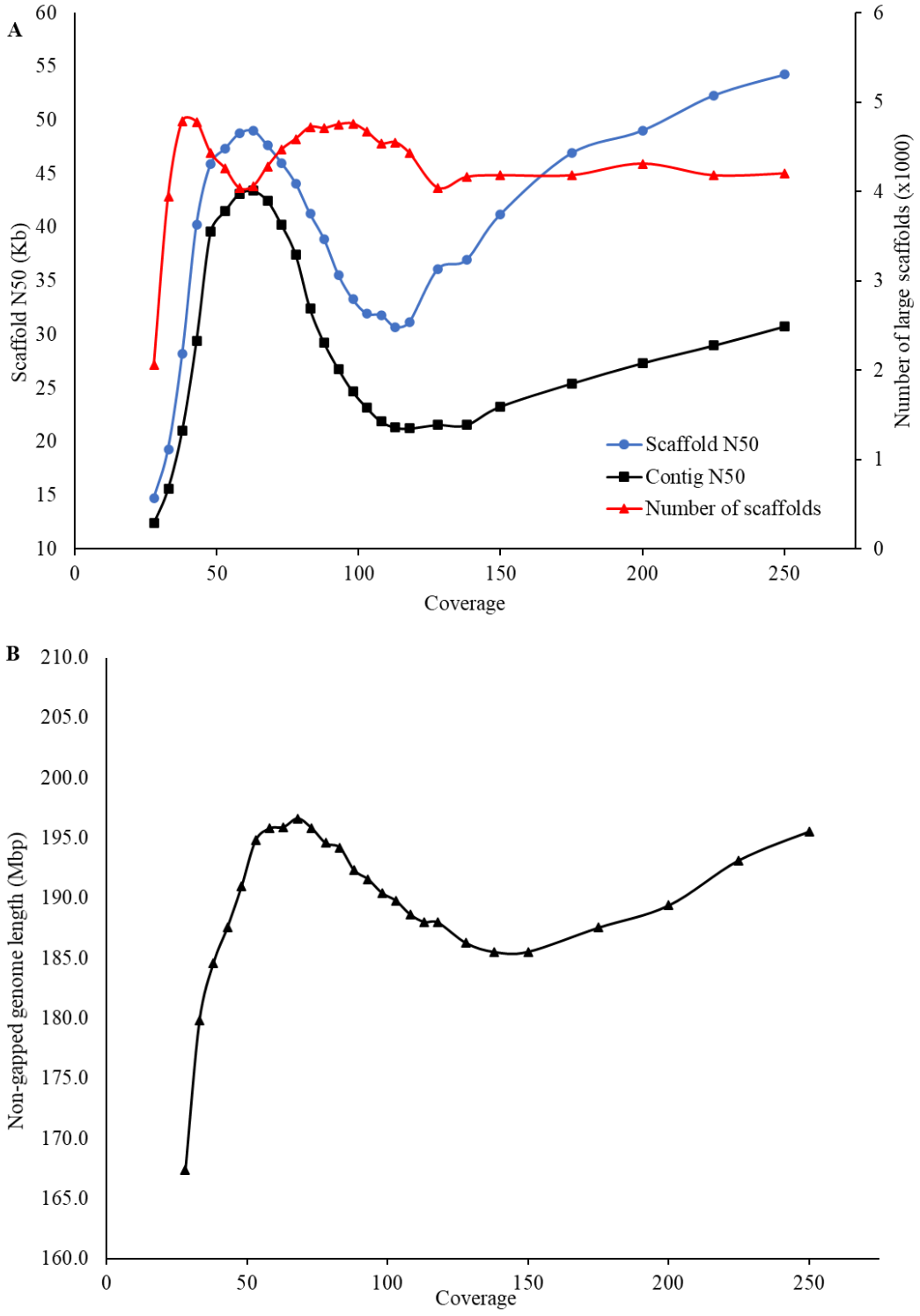


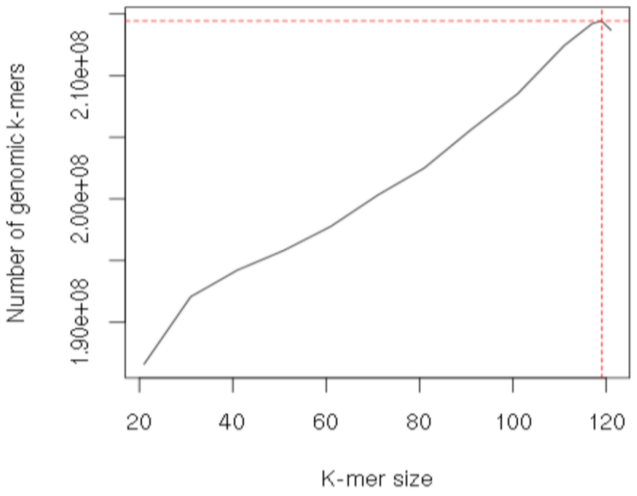
Figure 4. Comparison of four assembly quality metrics for Supernova assembly at various sequencing coverages of 10X linked reads. Scaffold N50 value in bp (A, blue), contig N50 in bp (A, black) and non-gapped assembly length in Mbp (B) were plotted for each sample at different coverages. The number of scaffolds larger than 10 kbp (A, red) is shown on the secondary Y-axis.

3.2: Characteristics of the *X. virginica* genome: reference genome, genome size estimation, and repeat content

Based on the comparison of the quality of genome assemblies for the 5 samples, for generation of the reference genome, we chose to use the unworn male assembly, which is by far the best among the 5 assemblies, as the base. The `SOAPdenovo` assembly obtained using all available PE reads was subjected to post-assembly improvements via two rounds of gap-filling using all PE reads including those from other samples. This led to an increase of more than 440 kbp non-gap sequences, representing a decrease of 5 bp per 100 kbp genome wide. The final reference genome consists of 5,440 scaffolds for a total of 193,543,232 bp sequences including gaps or 193,532,873 bp excluding sequence with a N50 at 207kb and minimal length of 500 bp and the largest scaffold being ~1.7Mb in length. According to QUAST, the G/C content of the assembly is 42.40%.

In order to provide an estimation of the genome size for *X. virginica,* we also run `Kmergenie` using the raw pair-end reads of all four samples and taking an average of the estimated genome sizes. As shown in Figure 5A, the output of `Kmergenie` gives the estimated genome size based on the reads provided. Figure 5B shows the genome sizes of all 4 samples (Linked-reads library not included due to barcoded region interfering with estimation) as well as the average size.

A)



B)

Sample	Genome Size (bp)
UWM	214,426,434
WM	220,158,655
UWF	237,510,849
WF	241,265,962
Average	228,340,475

Figure 5. Genome size estimation using `Kmergenie` based on Illumina pair-end sequencing data. A) `Kmergenie` output for unworn male sample with a predicted genome size at 214,426,434 bp; B) A table showing the estimated genome sizes for all 4 samples and the average estimate at 228 Mbp.

Combining the data on genome size prediction from `Kmergenie` and the length of the assemblies, we estimated the genome size of *X. virginica* to be approximate 220 Mb for a haploid genome.

The summary table from `RepeatMasker` shows a list of the repeats and the percentage each major type (Table S5). This includes *de novo* repeats as identified by `RepeatModeler`. The largest type of classified repeats are simple repeats/low complexity repeats (1.54%), followed by DNA transposons (0.93%), LINEs (0.20%), and LTR elements (0.13%) with ~6% being unclassified. For reference, the Amel_HAv3.1 assembly on NCBI lists the *Apis mellifera* draft genome as having a larger proportion of the genome comprising of simple and low complexity repeats (~5.70%) when compared to our assembly. DNA transposons are also more

prevalent in *A. mellifera* than *X. virginica*, while LINEs and LTR elements are present in similar levels for both (0.19% and 0.15%).

3.3. The effects of sequence heterozygosity on genome assembly quality

Another main objective of this study is to assess the impact of sequence heterozygosity on the quality of the genome assembly, specifically the effect of the allelic heterogeneity and somatic mutations. For this purpose, we compared the genome assembly quality of 4 bee samples representing two different ploidy levels, each at two different age groups. In this case, all four genomes were assembled using `SOAPdenovo2` with Illumina PE reads using both all available reads and equalized amounts of sequence data. For the unworn male, the raw genome assembly from `SOAPdenovo2` was used instead of the improved the reference genome to make a fair comparison. The assemblies were evaluated and ranked according to 5 metrics (N50 scaffold, length of scaffold, average length, non-gapped genome length, number of Ns per 100 kbp).

We also examined the characteristics of sequences associated with the break points of the scaffolds as way to understand the way(s) the genetic variants impacting the genome assembly.

3.3.1 Sequence heterozygosity derived from ploidy has a major impact on genome assembly quality.

To assess the impact of ploidy on the quality of the genome assembly, we compared genome assembly quality between the male and female unworn genomes and between worn male and worn female genomes. Table S3 shows the assembly statistics before and after coverage normalization. This step only slightly altered N50 value and average length. An increase in the number of sequences and total length could be due to the loss of crucial reads that used to bridge contigs but are now missing. From this, we can conclude that the difference in coverage does not

affect the order of quality of the assemblies. To ensure an unbiased comparison, the assemblies with equal coverage was used for further analyses.

Table 4 shows the metrics that were provided from `fatools` and `QUAST`. Number of scaffolds, scaffold N50 value, size range, average length, total assembly length with gaps and without gaps, and gap size distribution were all obtained using `fatools`. Number of Ns per 100 kbp was given by `QUAST`. The assemblies were ordered from highest quality to lowest quality.

All measures of assembly quality display a similar trend when comparing between male (haploid) and female (diploid) regardless of whether the sample was worn or unworn with the haploid genome assemblies being significantly better than that of the diploid genome assemblies. Specifically, the scaffold N50value of the unworn male (207,668 bp) is 97 times higher than that of the unworn female (2,145 bp). For the worn bees, N50 of the male (79,615) was also significantly higher than that of worn female (1,769 bp) but was only half of the difference seen for the unworn bees (97X vs. 45X). Average scaffold length and non-gapped assembly length are both larger for male samples than the corresponding female sample. Specifically, the unworn female assembly has a total of 96,719 sequences larger than 500 bp, while the unworn male assembly consists of only $1/20^{th}$ the number of sequences due to its much larger N50, and the worn male has $1/13^{th}$ the number of sequences of the worn female assembly.

The female genomes are ~50Mbp shorter than the age corresponding male genomes, being 144.4 Mbp vs 193.7 Mbp for unworn female and male genomes, and 145 Mbp vs 194.7 Mbp for the worn female and male genomes, respectively. This difference represents ~25% of the genome size, so it is very significant. In the meantime, female genomes have much more gap sequences than the male genomes, leading to larger difference in the non-gap sequence length, being 51.8 Mbp more in the unworn male than unworn female genome assemblies, and 51.2 Mbp

more in the worn male than in the unworn female genome assemblies. This is also visible by the gap density, being 0.52 kbp/100 kbp vs. 2.1 kbp/100 kbp for unworn male vs female genomes and 0.57 kbp/100 kbp vs. 1.56 kbp/100 kbp for worn male vs. female genome. After normalizing the sequencing coverage for ploidy, the density of gaps in the male genomes becomes more or less similar to that of the female genomes, indicating a contribution of gaps from insufficient sequences. However, the male genomes still have more than 50 Mbp non-gap sequences. The loss of these sequences was due to the possibility that female genome assemblies having a larger number of small scaffolds (<500 bp) which were filtered out from the final assembly for each genome. Female samples have, on average, 3 times more scaffolds between the lengths of 151 bp and 500 bp, than their male counterparts. The existence of more small scaffolds is either a result of less contigs being assembled or loss of connections for some contigs during scaffolding or a combination of the two situations.

In this case, the differences between the unworn male and female genome assemblies should be a better representation of the true effect of allelic heterogeneity on genome assembly quality, since in the worn bees, the loss of homology-based DNA repair, which leads to a higher rate of mutation may have offset the degree of heterozygosity from the ploidy level. Our result indicate that the allelic heterogeneity resulted from diploid genomes has a major effect on the genome assembly quality, including total genome length, amounts of gaps, and the N50. Specifically, in our case, the allelic heterogeneity led to the total genome assembly length being ~25% shorter, more gaps, and ~100 times lower by N50, indicating overall a much higher degree of assembly fragmentation and gaps.

Table 4: *De novo* genome assembly quality comparison

Samples/matrix	# of sequences*	N50 (bp)	Size range	Average length	Total length with gaps (bp)	Non-gap total length (bp)	Gap size:number	# of Ns per 100 kbp
Unworn Male	5,444	207,668	500-1,687,437	33,576	193,675,872	193,181,777	<11bp: 0; <101bp: 15; <500bp: 5078	515.16
Unworn Male**	12,436	39,836	500-286,814	15,707	195,340,310	192,049,465	<11bp: 4; <101bp: 12090; <500bp: 18597	1,668.96
Worn Male	8,570	79,615	500-586,123	22,719	194,708,439	194,002,450	<11bp: 0 <101bp: 209; <500bp: 4605	570.84
Worn Male**	61,769	5,330	500-48,411	3,190	197,057,028	191,042,912	<11bp: 3; <101bp: 2072; <500bp: 40685	3,030.27
Unworn Female	97,921	2,157	500-450,468	1,475	144,448,209	141,377,281	<11bp: 10; <101bp: 56; <500bp: 14604	2,115.82
Worn Female	104,671	1,770	500-342,394	1,385	145,027,269	142,755,704	<11bp: 14; <101bp: 30; <500bp: 10276	1,559.19
Unworn female (Supernova)	17,634	48,863	500-312, 088	11,911	210,053,875	206,849,082	<11bp: 0; <101bp: 5132; <500bp: 916; >1kb: 825	1522.43
Unworn Male (Reference)	5,440	207,531	500-1,686,392	35,577	193,543,232	193,532,873	<11bp: 892; <101bp: 46; <500bp: 43	4.85

* assembly was performed all at 76X coverage unless otherwise indicated; sequences from SOAPdenovo assembly were filtered to have a minimum size of 500 bp; ** assembled using only half the amount of provided reads to achieve similar coverage after normalizing for ploidy level

3.3.2 Assembly quality comparison between SOAPdenovo and Supernova assemblies.

Only one 10X linked-reads library was constructed for this study. The unworn female sample (linked-reads) and the unworn female sample (pair-end reads) were not from the same source and both were assembled with all available reads. As seen in Figure 6, the Supernova assembly has almost 23 times higher scaffold N50 size than that of the Illumina pair-end reads assembled with SOAPdenovo. In comparison (Table 4), the Supernova assembly quality is far below that of the haploid genome assemblies from SOAPdenovo in terms of N50 scaffold size, average length, and number of sequences. In addition, the Supernova assembly contained more gaps per 100 kbp than either of the male assemblies. The total length of the genome with gaps is 210,053,875 bp and without gaps is 206,849,082 bp, closer to the estimated haploid genome size of 228 Mbp for this species. This result indicates that while to 10X Linked reads does significantly improve the genome assembly quality of a diploidy genome by ~20 times in our specific case, it is far from being able to offset the negative impact of allelic heterozygosity on genome assembly quality.

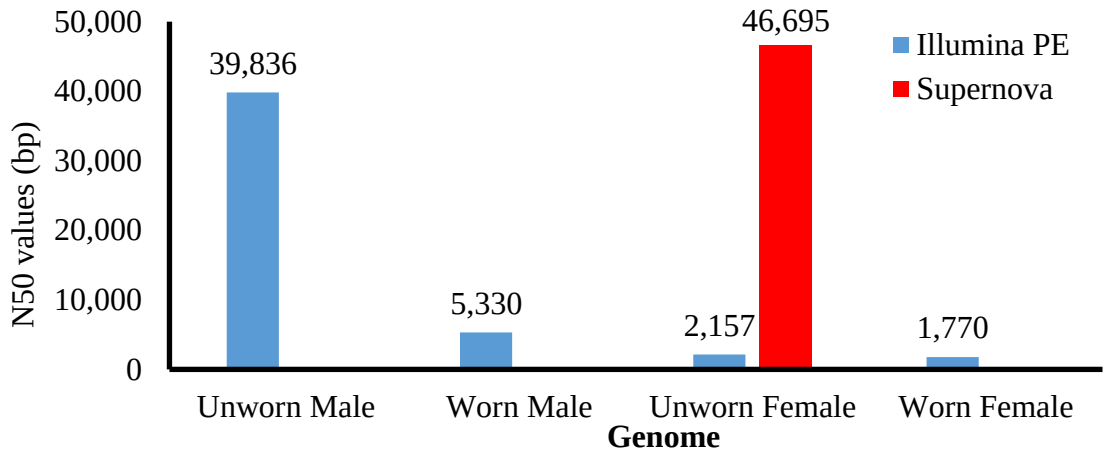


Figure 6. N50 statistic for genome assemblies. N50 values were calculated using an in-house Perl tool for scaffolds generated through SOAPdenovo2 (blue) and Supernova (red). All assemblies were generated at 76X.

3.3.3 Sequence heterozygosity from somatic mutation has a larger impact on genome assembly quality in male genomes than in female genomes, but less than that of allelic heterozygosity

To examine the effect of somatic variations on genome assembly quality, we compared the genome assemblies of worn bees in each gender to the respective assemblies of unworn bees in regard to the N50 value, average scaffold length, number of sequences, and non-gapped assembly length. For both genders, scaffold N50 is much lower for the genome assembly of worn bee relative to that of the unworn bees (Table 4). Specifically, at full sequence coverage for all samples, the N50 of the worn male is ~2.5 times lower that of the unworn male (79 kbp vs. 207 kbp), while the N50 of the worn female genome is ~3 times lower than that for the unworn female genome. For the male genomes, we also used only half of the reads in genome assembly to count for ploidy level to be at equivalent coverage with the female genomes in a stricter sense, and the degree of difference is even larger to be ~8 times lower in worn male (5.3 kbp vs. 39.8 kbp). As expected, the numbers of scaffolds and average length also support the same trend seen with N50. Interestingly, the effect on total assembly length is minimal (~0.5%) for both genders,

There are some interesting patterns for the differences with regard to the genome length and number and length of gaps in these assemblies (Table 4). At the equivalent sequence coverage of 76X for all samples (not normalize for ploidy level), the worn male genome is ~1 Mbp longer in total length than that of the unworn male (194.7 Mbp vs, 193.7 Mbp), likely due to more redundant fragments and more gap sequence (706 kbp vs 494 kbp). The density of gap sequences is ~10% higher in worn male than unworn male genomes (570 bp/100 kbp vs 515 bp/100 kbp) by having more gaps between 11 and 100 bp in length (209. Vs 15). However, for females, it is the unworn genome assembly having a high density of gaps than the worn genome

(~2.1 kbp/100 kbp vs 1.6 kbp/100 kbp) with the unworn female genome assembly having 40% more gaps between 101 and 500 bp in length than that of worn female genome assembly (14,604 vs. 10,276) (Table 4).

In summary, our results indicate that sequence heterozygosity resulted from somatic mutation also has an effect on genome assembly quality by N50, and the effect is slightly higher in male than in female, with little to no effect on the total genome assembly length. In our specific cases, the somatic mutation led to 7.5 times drop of N50 in males and 1.2 times in female, and it also need to some increase in gap sequence in the male but not in the female. The higher impact of these variants in male is likely a reflection of the higher mutation rate in the male genome than in the female genome from lack of homology-based DNA damage repair.

3.3.4 Assembly quality has some correlation with variant density

To understand how variants affect the assembly quality, the density of variants in each sample was measured and compared to the density of breakpoints. The detection of the variants in variable samples was based on the reference assembly, which is based on the unworn male genome, therefore, the analysis here is limited to the genomes of the worn male and unworn and worn females.

Table 5 shows the total variant counts and the quality of assembly matrixes of the three genome assemblies. The unworn male assembly was left out as it was used to generate the reference genome. Total variant count contains both SNPs and indels and was generated by combining the output from both `Mutect2` somatic variants and `Haplotypecaller` germline variants. By the total number of variants, worn female assembly contained most (865,819) variants, while worn male contained the least variants (587,381) among the three assemblies.

However, by density of variants adjusted for the genome ploidy, worn male has the highest (2.40 variants/kbp). The higher variant density in the worn male (2.40 variants/kbp vs. 0) genome corresponds to the higher impact on genome quality in male vs. female (2.12 vs. 2.07 variants/kbp for worn vs. unworn female genomes) (Table 5).

Table 5. Effect of total variant count on assembly quality

Sample Name	Variant count	N50 Scaffold	Average length of scaffold	Non-gapped assembly length (Mbp)	Variant density (per kbp)*
Worn male	587,381	79,615	22,719	194	2.40
Unworn female	846,563	2,145	1,454	138	2.07
Worn female	865,819	1,769	1,374	140	2.12

*Variant density calculated according to estimated genome length for haploid (228 Mbp) and diploid (456 Mbp)

3.3.5 The breakpoint in genome assemblies correlates best with variant density in worn male assembly

To examine the relationship between the assembly quality and accumulation of somatic variants, we focused on the breakpoints of contig extension and variant distribution. The distribution of SNPs and indels derived from the `pblat` alignment and variant calls were computed by mapping the assemblies to the pseudo linear as the reference (Figure 7). All three assemblies and the combined values showed a moderate correlation between the density of variants and breakpoints (Unworn female: 0.5340, worn female: 0.5699, worn male: 0.6973, combined: 0.5848). The worn male sample has the highest Pearson correlation coefficient while the female samples had lower coefficients.

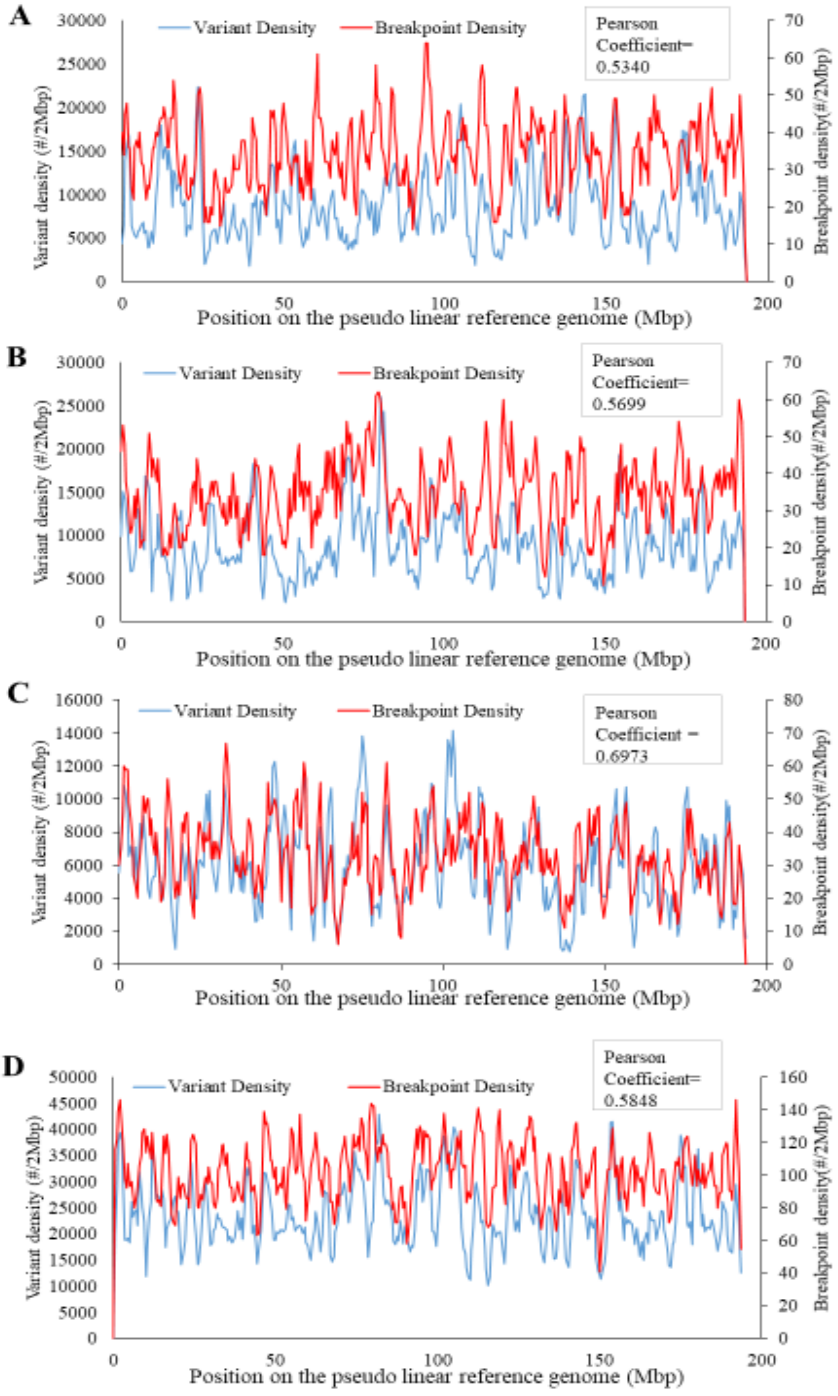


Figure 7. Distribution of variants and scaffold breakpoints in the genomes. The densities of variants (SNPs and indels) and scaffold breakpoints in counts/ 2 Mbp were calculated and plotted along the pseudo linear reference genome based on the unworn male *de novo* genome assembly for the unworn female genome (A), worn female genome (B), worn male genome (C), and combined data for these three genomes (D). In all panels, the variant density is shown in the primary (left) Y-axis and the breakpoint density is shown in the secondary (right) Y-axis.

3.3.6. Repeat density has low correlation with assembly breakpoints

To examine what factor(s) contribute to the fragmentation of genome assembly as reflected by the lower scaffold N50 and larger number of scaffolds compared to the unworn male genome, we analyzed the degree of correlation between break point density with repeats. For this analysis, only the 5 most prominent types were selected to undergo analysis. These include (in order of prevalence) Unclassified (5.83%), Simple repeats (1.27%), DNA transposons (0.93%), LINEs (0.20%), and LTR elements (0.13%). As shown in Table 6 disregarding the direction of correlation, among the ME types, "Unclassified" repeats show the most correlation to the density of breakpoints, followed by "DNA transposons", "Simple repeats", "LINEs" and "LTR elements". Simple repeats and LTR elements are the only types to show a weak positive correlation to breakpoint density, indicating that repeat sequence is not a major issue for genome assembly in the case of *Xylocopa virginica* genome due to the relative low content of repeats (8% vs. 50% in human) (Tang et al. 2018)

Table 6. Pearson's correlation between repeat types and breakpoints

	Coefficient
Simple repeats	0.109294605
LTR	0.030208168
LINEs	-0.085019237
DNA	-0.121626524
Unclassified	-0.147770165

CHAPTER 4: Discussion

From this study, we assembled and analyzed 5 assemblies of the species *X. virginica*, with the intent to understand the effect that sequence heterozygosity has on *de novo* assembly. We observe that the five samples provided have a distinct ranking of assembly quality, with the highest ranking being the unworn male sample and the lowest ranking being the worn female sample. We also noticed that coverage plays a role in the quality of the assembly and made an estimate regarding the minimum effective coverage for the input library of a species such as this. When comparing between assemblies, the `Supernova` assembly, using 10X linked-reads as input, vastly outranked the corresponding sample that used Illumina pair-end sequencing. As the first study to sequence *X. virginica*, we also made an estimate regarding its genome size. We discuss below several aspects of our data and observations.

4.1 Genome assembly

4.1.1 The optimal sequence coverages for different sequencing platforms

What would be the optimum sequence coverage to use for a *de novo* genome sequencing project is an intriguing question commonly faced by researchers. The answer really changes depending on the nature of the genome and the selection of the sequencing platform(s). In this study, we used two types of sequencing platforms, the commonly used short read platform, Illumina, and a less commonly used platform, which is the 10X Chromium Linked-reads. While based on the Illumina PE reads, the latter does work as a long-read platform in a sense by being able to rely on barcode data appended to DNA amplicons before sequencing. With regard to the type of genomes, here we used an insect as our model, which has a relatively small genome size

among animals with relatively low level of repeats. Therefore, the information we generated from this study might need to be limited to this group of species and these two specific sequencing platforms. Nevertheless, we think the data is still valuable to the research communities, especially the insect genomics community, with some general trend appliable to a wide scope beyond the insects.

For the impact of sequence coverage on genome quality, our results indicate that, for haploid samples, an increase in the coverage leads to improvement of the assembly quality, while for diploid samples, increasing the number of input reads does not necessarily always lead to a higher assembly quality. This might be because the increasing number of reads also introduces a lot of variability in the sequence due to the heterozygosity of the diploid genome, thus reducing the confidence of the assembler in making connections between reads. Changing the coverage of the input assemblies shows an optimal coverage of 20X for diploid samples, at which the N50 values were greatest. This number may vary depending on the assembler used, as `Supernova` results show that the diploid sample should have a 63X coverage for highest quality assembly. The trends we observe in this study is in agreement with a study by Zhang et al. (2019) on the human sample. They demonstrate that total coverage affects the quality of assembly and that the optimal coverage for human samples is around 56X (Zhang et al., 2019). Illumina guidelines suggests 30 to 50X coverage for human whole genome sequencing projects (https://www.illumina.com/science/technology/next-generation-sequencing/plan-experiments/coverage.html).

4.1.2 The impact of sequence heterozygosity on genome assembly quality

The ranking of the assemblies showed that the haploid, young sample had the overall highest assembly quality of the four samples, while the diploid, older sample was ranked lowest. We observe the negative effect of sequence heterozygosity on the assembly quality in this study. The difference in ploidy seems to play a much bigger role in determining the overall quality. Somatic variants, on the other hand, do not have as big an impact but would explain why the older samples ranked lower than the young samples. This was as expected and aligns with other studies, and it supports our initial hypotheses that male assemblies will have a higher quality than females, meaning an increase in ploidy will negatively affects the genome assembly quality. The sample with the longest average scaffold is the unworn male, indicating a more continuous assembly for that sample.

Non-gapped assembly length is a good measure of the completeness of the assembly. The closer the assembly length is to the actual genome size, the more likely that the assembly is of a higher quality. The sample with the longest non-gapped assembly size is the unworn female with 10X sequencing, followed by the worn male, unworn male, worn female, and unworn female. In general, worn samples have a higher non-gapped assembly length than its unworn counterpart. Considering the accumulation of somatic mutations within the genome as the sample ages, the above result is not what was expected and suggests that other factors, such as quality of the DNA, may play a bigger role. In addition, the larger number of scaffolds found in diploid assemblies in comparison to haploid assemblies indicate that sequence redundancy may be a factor which can negatively affect the non-gapped assembly length. Theoretically, somatic mutation in individuals introduce variants in the genome, which when sequenced as reads, are

sometimes labeled as errors during sequencing due to their low frequency. This becomes an issue when an abundance of somatic variants forces the assembler to introduce a breakpoint between scaffolds, thereby reducing the quality of the assembly. Interestingly, when we examined what percentage of the total assembly length with gaps is covered by the non-gapped assembly length among samples, the worn samples have very slightly lower values than the unworn samples (Unworn male vs worn male – 99.74% vs 99.63%; Unworn female vs worn female – 99.58% vs 99.53%), suggesting that the slightly higher amounts of gap might be contributed by the higher sequence heterozygosity resulted from higher somatic mutations in the worn samples.

The overall ranking of the assemblies is in line with similar studies. A study by Iwasaki, et al. (2016) investigated the use of haploid larvae in *de novo* genome assembly of *Seriola quinqueradiata*. Using two different assembly algorithms, DBG and OLC, they generated assemblies for the artificially produced haploid larvae and their diploid dam. The haploid assembly had an N50 scaffold value of 7502 bp while the diploid assembly had 6924 bp. The number of scaffolds and average length were significantly improved for the haploid sample. Their result concluded that using a haploid genome not only improved the efficacy of scaffolding by an estimated 40%, but also the quality of scaffolds as well. A more recent paper published in 2019 looked at the *de novo* assembly of hymenopteran genomes using the different ploidy samples in *Cataglyphis niger* using `SOAPdenovo` and `SPAdes`, both being DBG assemblers (Yahav & Privman, 2019). Their results indicate that the contig N50 for the haploid assembly was three times larger than the diploid. However, the completeness of the assembly was split for the two assemblers. The haploid assembly had a higher completeness than the diploid when using `SOAPdenovo`, but the opposite was true for `SPAdes`. In terms of N50 for scaffolds, haploid sample qualities show a ten-fold increase over the diploid samples regardless of the

assemblers used. In comparison, our results show that haploid samples may improve scaffold N50 values by up to 97-fold increase. This difference might be due to the age differences of the samples being used in the studies, as worn samples showed a smaller degree difference for the N50 scaffold sample in comparison to the unworn samples. The species in the study by Yahav and Privman did not specify the age of the samples, but it is possible that their samples were equivalent to our worn samples, which would explain the smaller differences in genome assembly quality observed between haploid and diploid samples. For future reference, it may be necessary to indicate some measurement of age for samples, especially when factors such as somatic variation may play a role in the variation of N50 values. Furthermore, based on our results, the use of young samples is recommended for extracting DNA for genome sequencing.

An alternative to comparing haploid and diploid genomes is to use a double haploid versus a diploid genome, similar to an earlier study by Zhang et al. (2014), which used mitotic gynogenesis to produce double haploid individuals of *Takifugu rubripes*. Then, by assembling and comparing with wild type diploid assemblies, they deduce that the heterozygosity of diploid genomes negatively affects the assembler into introducing split paths.

As can be seen, this is not the first study to examine the effect of sequence heterozygosity on assembly quality using an organism from the Hymenoptera. However, in addition to focusing on the effect of ploidy as in the above prior studies and comparing between samples at different ages, we also examined the effect of the variants (both location and density) and repeats on genome assembly quality.

4.1.2.a Correlation of variant density to the density of breakpoints in the assembly

This study is designed based on the assumptions that i) somatic variants accumulate steadily throughout an organisms' lifespan; ii) somatic variation and unrepaired, damaged DNA within the genome are sampled during sequencing and confuse assemblers by having differing base identities and making it difficult to ascertain the true sequence; and iii) assemblers will introduce a break point in the resulting contigs resulting in shorter contigs and a more fragmented genome assembly. In this study, a moderate positive correlation was identified using the density plot of variant density and breakpoint along the pseudo-linearized genome (Figure 7). While correlation is not equal to causation, this provides an argument as to whether the effect of somatic variants can introduce breakpoints in the assembly during the contig extension process. With this result, we cannot be certain that somatic variants are the main cause of the decrease in quality between young and old samples. Another factor that could have caused the decrease in quality is from a lower quality of DNA i.e., damaged DNA and physical DNA breakpoints.

4.1.2.b Correlation of 5 groups of repeat density to the density of breakpoints in the assembly

As shown in Table 6, all repeat types show weak correlation with the breakpoint density, indicating that repeats are likely not a reason behind the introduction of breakpoints in assemblies. LINEs and LTRs occur rarely (<= 10 and <20 in each 500 kbp window) in the genome when compared to simple repeats and DNA transposons (often > 100). This is likely the reason why the former two types have a lower correlation to breakpoints than the latter two types. In addition, I reason that the repeat age may play a role in affecting the genome assembly with recent repeats likely having more impact due to their higher level of sequence similarity

between copies. The lack of recent repeat copies from the low activity of the repeats in this species' genome as indicated by the low percentage of repeats may also help explain their low impact on genome assembly quality.

4.2 Bee biology

4.2.1 The genome size of *X. virginica*

Based on the results from Chapter 3.2, an appropriate genome size estimate would be around 220 Mbp. This is similar to the genome length provided on NCBI for other related organisms (*Ceratina australensis* (closest organism with completed genome assembly) - 219.30 Mbp, *A. mellifera* – 236 Mbp) (https://www.ncbi.nlm.nih.gov/genome/?term=txid78185[Organism:noexp], https://www.ncbi.nlm.nih.gov/genome/?term=txid7460[Organism:exp]). Supernova algorithm for estimating genome size that may not work well in all circumstances for all species. Insects are generally more problematic than mammals, and for some plants, reptiles, and butterflies there are cases where genome size estimate from Supernova does not line up with estimates from other data sources or varies significantly as a function of coverage as defined by number of reads during the input step (https://support.10xgenomics.com/de-novo-assembly/software/pipelines/latest/output/asm-stats).

4.2.2 Genome content comparison

The repeat content for *X. virginica* shares many similarities in repeat content to *A. mellifera.* As shown in table S5, the total repeat content of *X. virginica* is 8.65% of the assembly

length of 193 Mbp, while *A. mellifera* is slightly higher at 9.58% of 236 Mbp. In terms of sequence percentage, the two organisms have similar values for SINEs (0.012% vs 0.00%), LINEs (0.188% vs 0.20%), LTR elements (0.153% vs 0.13%), and DNA transposons (1.384% vs 0.93%). However, when comparing between the total interspersed repeats, *X. virginica* has 1.8 times higher than that of *A. mellifera*. The difference in this value could mean that *X. virginica* has a more active TE library in comparison to that of the honeybee. The GC content of the reference assembly for X. virginica was determined to be 42.4%. In comparison, humans are known to have an average of 41% G-C content (Romiguier et al., 2010) while *A. mellifera* has a G-C content of 34.1% (Wallberg et al., 2019). The higher percentage of G-C content in *X. virginica* could be an indication that G-C rich regions are over-represented in the assembly as a result of GC bias during sequencing.

4.2.3 Implications of haplodiploidy

In this analysis, we use the mechanism of sex determination known as haplodiploidy as a natural substitution for artificial ploidy manipulation in samples (Iwasaki et al., 2016). The difference in ploidy between females and males can allow for a direct comparison of the effect of ploidy on genome assembly. This phenomenon opens up another venue to investigate other direct effects of ploidy on the genome, such as the mechanism of homology directed DNA repair. Since the haploid males lack this repair pathway in comparison to the diploid females, it would be interesting to observe how the rate of propagation of somatic variants is affected and the specific forms of impact in sequence, likely among increased breakpoints and gaps, as well as the nature of the sequence for the regions are mostly impacted.

4.3. Conclusions and future perspectives

The ploidy of the organism is an important factor in the design of a *de novo* genome sequencing project. This study shows that utilizing the haploid genome of *X. virginica* resulted in very significant improvements in the *de novo* assembly. The haploid male assemblies outranked the diploid female assemblies in both age groups, worn and unworn. The assembly of a haploid genome effectively decreased the total number of contigs/scaffolds, resulting in an increase in the average and N50 scaffold lengths, as well as 25% more total genome sequences. Moreover, the haploid assembly also improved the quality of scaffold sequences by reducing the number of regions with gaps (i.e., unassigned nucleotides (Ns)). Thus, the outcome of our study reinforces the strategy for constructing the reference genomes from non-model diploid organisms using a haploid alternative if available. Furthermore, our study indicates that genomic heterogeneity from somatic mutation can also have a negative impact on genome assembly quality, despite not as significant as the allelic heterogeneity, nevertheless suggesting the use of younger samples should be used for sequencing a non-model organism with the aim of generating a high-quality reference genome sequence.

In addition, our study results provided some useful guidelines regarding the optimal sequencing platform and sequence coverage for a genome sequencing project in two folds. First, the use of long read platform is preferred over the short-read platform, especially for diploid genomes. Second, while in general, more sequencing coverage is beneficial, especially for haploid genomes, it can become detrimental to genome assembly quality passing certain threshold for diploid genomes. The latter applies to both the Illumina PE reads and the 10X linked reads, with the negative impact being more significant for the latter. Last, but not least,

our study demonstrated that organisms, such as *X. virginica*, offer ideal models for studying the role of homology-based DNA repair.

The lack of information on the relationship of the samples represents a shortfall for our research design, as it does not allow us to distinguish between variants from somatic mutation verse germline mutations between the worn and unworn samples. By tracking the relationship between samples, it would be easier to track the hereditary lineage of the samples and use that information to distinguish between germline and somatic variants. In future studies, efforts can be made to use samples from the parents and their offspring. It is also important to determine the population of the samples, as organisms from the same population are more likely to share the same external stress, and thus develop variants in similar fashion. Furthermore, studies can be designed to examine how different types of sequence heterozygosity specifically interfere the assembly algorithms and how can existing algorithms can be improved or new assembly algorithms can be developed to deal with such interferences.

Adding gene annotation to the reference genome can be valuable for better understanding the biology of *X. virginica* by allowing comparative analysis with the genomes of closely related organisms, such as the honeybee (*Apis mellifera*) and small carpenter bee (*Cataglyphis niger*). Once the reference assembly is more polished, it can be used in further studies regarding homology-based DNA repair mechanism or to examine the level of genetic diversity within this species.

References

Angel, V. D., Hjerde, E., Sterck, L., Capella-Gutierrez, S., Notredame, C., Pettersson, O. V., . . . Lantz, H. (2018). Ten steps to get started in Genome Assembly and Annotation. *F1000Research, 7*, 148. doi:10.12688/f1000research.13598.1

Allam, A., Kalnis, P., & Solovyev, V. (2015). Karect: accurate correction of substitution, insertion and deletion errors for next-generation sequencing data. *Bioinformatics, 31*(21), 3421–3428. doi: 10.1093/bioinformatics/btv415

Auton, A., L. D. Brooks, R. M. Durbin, E. P. Garrison, H. M. Kang, J. O. Korbel, J. L. Marchini, S. McCarthy, G. A. McVean, G. R. Abecasis and G. P. Consortium (2015). "A global reference for human genetic variation." Nature 526(7571): 68-74.

Baker, M. (2012). *De novo* genome assembly: What every biologist should know. *Nature Methods,9*(4), 333-337. doi:10.1038/nmeth.1935

Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., … Pevzner, P. A. (2012). SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. *Journal of Computational Biology*, *19*(5), 455–477. doi: 10.1089/cmb.2012.0021

Basturea, G. N. (2018). Somatic Mutations. *Materials and Methods, 8*. doi:10.13070/mm.en.8.2673

Bentley, D. R., Balasubramanian, S., Swerdlow, H. P., Smith, G. P., Milton, J., Brown, C. G., . . . Smith, A. J. (2008). Accurate whole human genome sequencing using reversible terminator chemistry. *Nature, 456*(7218), 53-59. doi:10.1038/nature07517

Boetzer, M., Henkel, C. V., Jansen, H. J., Butler, D., & Pirovano, W. (2010). Scaffolding pre-assembled contigs using SSPACE. *Bioinformatics*, *27*(4), 578–579. doi: 10.1093/bioinformatics/btq683

Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*, *30*(15), 2114–2120. doi: 10.1093/bioinformatics/btu170

Bradnam, K. R., Fass, J. N., Alexandrov, A., Baranay, P., Bechner, M., Birol, I., . . . Korf, I. F. (2013). Assemblathon 2: Evaluating de novo methods of genome assembly in three vertebrate species. *GigaScience, 2*(1). doi:10.1186/2047-217x-2-10

Chaisson, M. J., Wilson, R. K., & Eichler, E. E. (2015). Genetic variation and the *de novo* assembly of human genomes. *Nature Reviews Genetics, 16*(11), 627-640. doi:10.1038/nrg3933

Chatterjee, N., & Walker, G. C. (2017). Mechanisms of DNA damage, repair, and mutagenesis. *Environmental and Molecular Mutagenesis*, *58*(5), 235-263. doi:10.1002/em.22087

Chikhi, R., & Rizk, G. (2013). Space-efficient and exact de Bruijn graph representation based on a Bloom filter. *Algorithms for Molecular Biology*, *8*(1). doi: 10.1186/1748-7188-8-22

Chu, C., Li, X., & Wu, Y. (2017). GAPPadder: A Sensitive Approach for Closing Gaps on Draft Genomes with Short Sequence Reads. doi: 10.1101/125534

Church, D. (2016). A basic introduction to linked-reads - 10x Community. Retrieved from https://community.10xgenomics.com/t5/10x-Blog/A-basic-introduction-to-linked-reads/ba-p/95.

Dayarian, A., T. P. Michael and A. M. Sengupta (2010). "SOPRA: Scaffolding algorithm for paired reads via statistical optimization." BMC Bioinformatics 11: 345.

Ekblom, R., & Wolf, J. B. W. (2014). A field guide to whole-genome sequencing, assembly and annotation. *Evolutionary Applications, 7*(9), 1026–1042. doi: 10.1111/eva.12178

Greenfield, P., Duesing, K., Papanicolaou, A., & Bauer, D. C. (2014). Blue: Correcting sequencing errors using consensus and context. *Bioinformatics, 30*(19), 2723-2732. doi:10.1093/bioinformatics/btu368

Griffith, A. J., Miller, J. H., Suzuki, D. T., Lewontin, R. C., & Gelbert, W. M. (2000). *An Introduction to Genetic Analysis* (7th ed.). doi:0-7167-3520-2

Iwasaki, Y., Nishiki, I., Nakamura, Y., Yasuike, M., Kai, W., Nomura, K., … Ototake, M. (2016). Effective *de novo* assembly of fish genome using haploid larvae. *Gene*, *576*(2), 644–649. doi: 10.1016/j.gene.2015.10.015

Iyama, T., & Wilson, D. M. (2013). DNA repair mechanisms in dividing and non-dividing cells. *DNA Repair, 12*(8), 620-636. doi:10.1016/j.dnarep.2013.04.015

Khan, A. R., Pervez, M. T., Babar, M. E., Naveed, N., & Shoaib, M. (2018). A Comprehensive Study of *De novo* Genome Assemblers: Current Challenges and Future Prospective. *Evolutionary Bioinformatics,14*, 117693431875865. doi:10.1177/1176934318758650

King, R.C; Stansfield, W.D.; Mulligan, P.K. (2006). *A dictionary of genetics* (7th ed.). Oxford University Press. p. 194. ISBN 978-0-19-530761-0

Koren, O., Goodrich, J. K., Cullender, T. C., Spor, A., Laitinen, K., Bäckhed, H. K., … Ley, R. E. (2012). Host Remodeling of the Gut Microbiome and Metabolic Changes during Pregnancy. *Cell*, *150*(3), 470–480. doi: 10.1016/j.cell.2012.07.008

Koren, S., T. J. Treangen and M. Pop (2011). "Bambus 2: scaffolding metagenomes." Bioinformatics 27(21): 2964-2971.

Li, H. (2012). Exploring single-sample SNP and INDEL calling with whole-genome *de novo* assembly. *Bioinformatics*, *28*(14), 1838–1844. doi: 10.1093/bioinformatics/bts280

Li, R., Fan, W., Tian, G., Zhu, H., He, L., Cai, J., . . . Wang, J. (2009). The sequence and de novo assembly of the giant panda genome. *Nature, 463*(7279), 311-317. doi:10.1038/nature08696

Li, X., & Heyer, W. (2008). Homologous recombination in DNA repair and DNA damage tolerance. *Cell Research, 18*(1), 99-113. doi:10.1038/cr.2008.1

Li, X. C., & Tye, B. K. (2011). Ploidy Dictates Repair Pathway Choice under DNA Replication Stress. *Genetics, 187*(4), 1031-1040. doi:10.1534/genetics.110.125450

Li, Z., Chen, Y., Mu, D., Yuan, J., Shi, Y., Zhang, H., . . . Fan, W. (2011). Comparison of the two major classes of assembly algorithms: Overlap-layout-consensus and de-bruijn-graph. *Briefings in Functional Genomics,11*(1), 25-37. doi:10.1093/bfgp/elr035

Liang, F., Han, M., Romanienko, P. J., & Jasin, M. (1998). Homology-directed repair is a major double-strand break repair pathway in mammalian cells. *Proceedings of the National Academy of Sciences, 95*(9), 5172-5177. doi:10.1073/pnas.95.9.5172

Lim, E., Müller, J., Hagmann, J., Henz, S. R., Kim, S., & Weigel, D. (2014). Trowel: A fast and accurate error correction module for Illumina sequencing reads. *Bioinformatics, 30*(22), 3264-3265. doi:10.1093/bioinformatics/btu513

Lindgreen, S. (2012). AdapterRemoval: Easy Cleaning of Next Generation Sequencing Reads. *BMC Research Notes, 5*(1), 337. doi:10.1186/1756-0500-5-337

Luo, R., Liu, B., Xie, Y., Li, Z., Huang, W., Yuan, J., … Wang, J. (2012). SOAPdenovo2: an empirically improved memory-efficient short-read *de novo* assembler. *GigaScience*, *1*(1). doi: 10.1186/2047-217x-1-18

Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal, 17*(1), 10. doi:10.14806/ej.17.1.200

Malzahn, A., Lowder, L., & Qi, Y. (2017). Plant genome editing with TALEN and CRISPR. *Cell & Bioscience, 7*(1). doi:10.1186/s13578-017-0148-4

Milholland, B., Dong, X., Zhang, L., Hao, X., Suh, Y., & Vijg, J. (2017). Differences between germline and somatic mutation rates in humans and mice. *Nature Communications*, *8*(1). doi:10.1038/ncomms15183

Mueller, U. G., & Wolf-Mueller, B. (1993). A method for estimating the age of bees: Age-dependent wing wear and coloration in the Wool-Carder beeAnthidium manicatum (hymenoptera: Megachilidae). *Journal of Insect Behavior, 6*(4), 529–537. doi: 10.1007/bf01049530

Nadalin, F., Vezzi, F., & Policriti, A. (2012). GapFiller: a *de novo* assembly approach to fill the gap within paired reads. *BMC Bioinformatics, 13*(S14). doi: 10.1186/1471-2105-13-s14-s8

Phillippy, A. M., Schatz, M. C., & Pop, M. (2008). Genome assembly forensics: Finding the elusive mis-assembly. *Genome Biology, 9*(3). doi:10.1186/gb-2008-9-3-r55

Pop, M. (2009). Genome assembly reborn: recent computational challenges. *Briefings in Bioinformatics, 10*(4), 354–366. doi: 10.1093/bib/bbp026

Quail, M., Smith, M. E., Coupland, P., Otto, T. D., Harris, S. R., Connor, T. R., … Gu, Y. (2012). A tale of three next generation sequencing platforms: comparison of Ion torrent, pacific biosciences and illumina MiSeq sequencers. *BMC Genomics*, *13*(1), 341. doi: 10.1186/1471-2164-13-341

Richards, R., Patel, J., Stevenson, K., & Harbison, S. (2018). Evaluation of massively parallel sequencing for forensic DNA methylation profiling. *Electrophoresis, 39*(21), 2798-2805. doi:10.1002/elps.201800086

Romiguier, J., Ranwez, V., Douzery, E. J., & Galtier, N. (2010). Contrasting GC-content dynamics across 33 mammalian genomes: Relationship with life-history traits and chromosome sizes. *Genome Research, 20*(8), 1001-1009. doi:10.1101/gr.104372.109

Rodgers, K., & Mcvey, M. (2015). Error-Prone Repair of DNA Double-Strand Breaks. *Journal of Cellular Physiology, 231*(1), 15-24. doi:10.1002/jcp.25053

Salzberg, S. L., Phillippy, A. M., Zimin, A., Puiu, D., Magoc, T., Koren, S., . . . Yorke, J. A. (2012). GAGE: A critical evaluation of genome assemblies and assembly algorithms. *Genome Research,22*(3), 557-567. doi:10.1101/gr.131383.111

Salzmann, A. P., Russo, G., Aluri, S., & Haas, C. (2019). Transcription and microbial profiling of body fluids using a massively parallel sequencing approach. *Forensic Science International: Genetics, 43*, 102149. doi:10.1016/j.fsigen.2019.102149

Schulz, M. H., Weese, D., Holtgrewe, M., Dimitrova, V., Niu, S., Reinert, K., & Richard, H. (2014). Fiona: A parallel and automatic strategy for read error correction. *Bioinformatics, 30*(17), I356-I363. doi:10.1093/bioinformatics/btu440

Simpson, J. T., & Durbin, R. (2011). Efficient *de novo* assembly of large genomes using compressed data structures. *Genome Research, 22*(3), 549-556. doi:10.1101/gr.126953.111

Sohn, J., & Nam, J. (2018). The present and future of *de novo* whole-genome assembly. *Briefings in Bioinformatics,19*(1), 23-40. doi:10.1093/bib/bbw096

Song, L., Florea, L., & Langmead, B. (2014). Lighter: Fast and memory-efficient sequencing error correction without counting. *Genome Biology, 15*(11). doi:10.1186/s13059-014-0509-9

Tan, M. H., Austin, C. M., Hammer, M. P., Lee, Y. P., Croft, L. J., & Gan, H. M. (2018). Finding Nemo: Hybrid assembly with Oxford Nanopore and Illumina reads greatly improves the clownfish (Amphiprion ocellaris) genome assembly. *GigaScience,7*(3). doi:10.1093/gigascience/gix137

Tang, W., Mun, S., Joshi, A., Han, K., & Liang, P. (2018). Mobile elements contribute to the uniqueness of human genome with 15,000 human-specific insertions and 14 Mbp sequence increase. *DNA Research, 25*(5), 521-533. doi:10.1093/dnares/dsy022

Th, A., & Ma, S. (2015). Next Generation Sequencing Technologies: A Short Review. *Journal of Next Generation Sequencing & Applications, 01*(S1). doi: 10.4172/2469-9853.s1-006

Wallberg, A., Bunikis, I., Pettersson, O. V., Mosbech, M.-B., Childers, A. K., Evans, J. D., … Webster, M. T. (2018). A hybrid *de novo* genome assembly of the honeybee, Apis mellifera, with chromosome-length scaffolds. doi: 10.1101/361469

Weisenfeld, N. I., Kumar, V., Shah, P., Church, D., & Jaffe, D. B. (2016). Direct determination of diploid genome sequences. *Genome Research,27*, 757-767. doi:10.1101/070425

White, M. J. (1984). Chromosomal mechanisms in animal reproduction. *Bolletino Di Zoologia, 51*(1-2), 1-23. doi:10.1080/11250008409439455

Xie, Y., Wu, G., Tang, J., Luo, R., Patterson, J., Liu, S., … Wang, J. (2014). SOAPdenovo-Trans: *de novo* transcriptome assembly with short RNA-Seq reads. *Bioinformatics, 30*(12), 1660–1666. doi: 10.1093/bioinformatics/btu077

Yahav, T., & Privman, E. (2019). A comparative analysis of methods for *de novo* assembly of hymenopteran genomes using either haploid or diploid samples. *Scientific Reports, 9*(1). doi: 10.1038/s41598-019-42795-6

Yandell, M., & Ence, D. (2012). A beginner's guide to eukaryotic genome annotation. *Nature Reviews Genetics, 13*(5), 329-342. doi:10.1038/nrg3174

Zerbino, D. R., & Birney, E. (2008). Velvet: Algorithms for *de novo* short read assembly using de Bruijn graphs. *Genome Research, 18*(5), 821–829. doi: 10.1101/gr.074492.107

Zhang, H., Tan, E., Suzuki, Y., Hirose, Y., Kinoshita, S., Okano, H., … Asakawa, S. (2014). Dramatic improvement in genome assembly achieved using doubled-haploid genomes. *Scientific Reports, 4*(1). doi: 10.1038/srep06780

Zhang, L., Zhou, X., Weng, Z., & Sidow, A. (2019). Assessment of human diploid genome assembly with 10x Linked-Reads data. *GigaScience, 8*(11). doi:10.1093/gigascience/giz141

Supplementary Tables and Figures

Table S1: Computational resources required for different steps of the genome assembly

Process	Tool	CPU	Memory (GB)	Wall clock (h)
Preprocessing	Trimmomatic-0.36	32	0.8	~10
	KARECT	32	643	~7.5
Assembly	Supernova v2.1	1	220	12
	SOAPdenovo2	32	50	~0.75
Mutation rate	bwa	1	18	2
	GATK (Haplotypecaller)	4	23	11
	GATK (Mutect2)	4	101	17
Post-assembly processing	In-house perl tool (fatools)	1	4	0.03
	GAPPadder	1	15	37
	Gapcloser	1	22	10

Table S2: N50 values for SOAPdenovo assemblies at various coverages

Samples/matrix	# sequences*	N50 (bp)
UWF 5X	9,145	575
UWF 10X	113,066	825
UWF 20X	94,564	2,806
UWF 30X	99,532	2,254
UWF 40X	99,144	2,037
Unworn Female (81X)	97,921	2,157
UWM 5X	452	612
UWM 10X	8,355	565
UWM 20X	136,712	834
UWM 30X	79,173	3,984
UWM 40X	20,604	21,272
Unworn Male (146X)	5,444	207,668
WM 5X	530	890
WM 10X	3,007	572
WM 20X	95,674	656
WM 30X	146,940	1,318
WM 40X	70,354	4,524
Worn Male (134X)	8,570	79,615
WF 5X	6,221	567
WF 10X	113,101	742
WF 20X	102,610	2,440
WF 30X	108,389	1,852
WF 40X	106,765	1,670
Worn Female (76X)	104,671	1,770

* sequences from SOAPdenovo assembly were filtered to have a minimum size of 500 bp

Table S3. Effect of coverage normalization on assembly quality

Samples/matrix	# of sequences**	N50 (bp)	Size range	Average length	Total length with gaps (bp)
Unworn Male	5,440	207,948	500-1,689,041	35,678	194,093,716
Worn Male	8,555	79,907	500-586,997	22,802	195,073,119
Unworn Female	69,719	2,145	500-450,379	1,454	140,650,204
Worn Female	103,372	1,769	500-416,173	1,374	142,112,360
Coverage normalized*	**# of sequences****	**N50 (bp)**	**Size range**	**Average length**	**Total length with gaps (bp)**
Unworn Male	5,440	207,948	500-1,689,041	35,678	194,093,716
Unworn Female	97,921	2,157	500-450,468	1,475	144,448,209
Worn Female	104,671	1,770	500-342,394	1,385	145,027,269

*Worn male was omitted due to no change

**sequences from SOAPdenovo assembly were filtered for a minimum size of 500 bp

Table S4. N50 value comparison between samples

Sample/Matrix	Illumina PE	Supernova
Unworn Male	39,836	N/A
Worn Male	5,330	N/A
Unworn Female	2,157	46,695
Worn Female	1,770	N/A

Table S5. Output table from RepeatMasker

==========

file name: reference_assembly.500.randomized_linearGenome.fa
sequences: 1
total length: 193543232 bp (193534764 bp excl N/X-runs)
GC level: 42.40 %
bases masked: 16733350 bp (8.65 %)

==========

	number of elements*	**length occupied**	**percentage of sequence**	**percentage of sequence (*Apis Mellifera*)****
Retroelements	1721	643858 bp	0.33%	-
SINEs:	0	0 bp	0.00%	0.012%
Penelope		0 bp	0.00%	-
LINEs:	1194	385025 bp	0.20%	0.188%
CRE/SLACS	0	0 bp	0.00%	-
L2/CR1/Rex	0	0 bp	0.00%	-
R1/LOA/Jockey	88	265308 bp	0.14%	-
R2/R4/NeSL	159	53559 bp	0.03%	-
RTE/Bov-B	0	0 bp	0.00%	-
L1/CIN4	199	16725 bp	0.01%	-
LTR elements:	527	258833 bp	0.13%	0.153%
BEL/Pao	116	62563 bp	0.03%	-
Ty1/Copia	34	25077 bp	0.01%	-
Gypsy/DIRS1	231	148513 bp	0.08%	-
Retroviral	0	0 bp	0.00%	-
DNA transposons	9531	1801632 bp	0.93%	1.384%
hobo-Activator	3287	534897 bp	0.28%	-
Tc1-IS630-Pogo	4463	973501 bp	0.50%	-
En-Spm	0	0 bp	0.00%	-
MuDR-IS905	0	0 bp	0.00%	-
PiggyBac	1193	189821 bp	0.10%	-
Tourist/Harbinger	169	20608 bp	0.01%	-
Other (Mirage, P-element, Transib)	0	0 bp	0.00%	-
Rolling-circles	0	0 bp	0.00%	-

Unclassified:	56324	11278966 bp	5.83%	0.107%
Total interspersed repeats:		13724456 bp	7.09%	3.822%
Small RNA:	64	17478 bp	0.01%	-
Satellites:	46	16305 bp	0.01%	0.030%
Simple repeats:	56504	2451256 bp	1.27%	5.698%[1]
Low complexity:	10007	523855 bp	0.27%	

* most repeats fragmented by insertions or deletions have been counted as one element

***Apis Mellifera** version Amel_4.5 repeat data obtained from NCBI (genome size: 236 Mbp)

[1] Simple repeats and low complexity repeats have been combined in the *A. mellifera* data

Appendix A: Commands and configuration settings

`Trimmomatic` run command:

```
java -jar $EBROOTTRIMMOMATIC/trimmomatic-0.36.jar PE -trimlog
trimlog1.log /home/jt18hz/projects/def-
pliang/jt18hz/raw_reads/HI.4950.006.IDT_i7_156---
IDT_i5_156.Xylocopa_virginica_unworn_male_R1.fastq.gz
/home/jt18hz/projects/def-
pliang/jt18hz/raw_reads/HI.4950.006.IDT-i7_156---
IDT_i5_156.Xylocopa_virginica_unworn_male_R2.fastq.gz -baseout
Unwornmale_trimmed.fq.gz ILLUMINACLIP:TruSeq3-PE.fa:2:30:10
LEADING:3 TRIALING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

`KARECT` run command:

```
./karect -correct -
inputfile=/home/jt18hz/scratch/Virginica_reads/HI.4950.006.IDT_i
7_168---IDT_i5_168.Xylocopa_virginica_unworn_female_R1.fastq -
inputfile=/home/jt18hz/scratch/Virginica_reads/HI.4950.006.IDT_i
7_168---IDT_i5_168.Xylocopa_virginica_unworn_female_R2.fastq -
celltype=diploid -matchtype=hamming -
resultdir=/home/jt18hz/scratch/karected_untrimmed/
```

`Fastq_split.pl` run command:

```
/home/pliang/bin/FASTQ_split.pl -n 102,003,439
/home/jt18hz/scratch/unworn_female/karected/
karect_Unwornfemale_trimmed_1P.fq >
karect_Unwornfemale_trimmed_1P_0000.fq
```

`SOAPdenovo2` configuration file and run commands:

Configuration file:

```
max_rd_len=151

[LIB]

avg_ins=380

q1=/home/jt18hz/scratch/unworn_female/karected/karect_Unwornfema
le_trimmed_1P.fq
```

```
q2=/home/jt18hz/scratch/unworn_female/karected/karect_Unwornfema
le_trimmed_2P.fq
```

Run command:

```
SOAPdenovo-127mer all -p 32 -K 121 -L 500 -F -s
/scratch/jt18hz/SOAP-runs/config_fileUWF_new -o Unwornfemale_out
1>assembly1.log 2>assembly1.err
```

`Fatools.pl` run command:

```
fatools -l 500 Unwornfemale_out.scafSeq | fatools -pS -
2>Unwornfemale_out_500.stats >>Unwornfemale_out_500.stats
```

`Supernova` run command and output generation:

Run command:

```
supernova run --id=Xvirginica_43x --maxreads=89735869 --
fastqs=/home/jt18hz/scratch/Virginica_reads/10X_reads --
localcores=32 --localmem=350
```

Output generation:

```
supernova mkoutput --style=pseudohap2

-asmdir=../Xvirginica_noI1/outs/assembly

--outprefix=Xvirginica_noI1
```

`Gappadder` configuration file and run command:

Configuration file:

```
{"draft_genome": {"fa":
"/home/jt18hz/scratch/unworn_male/SOAP_UWM_trimmed/Unwornmale_out.scafSeq"},


"raw_reads": [{"left":
"/home/jt18hz/scratch/Virginica_reads/HI.4950.006.IDT_i7_168---
IDT_i5_168.Xylocopa_virginica_unworn_female_R1.fastq",

"right": "/home/jt18hz/scratch/Virginica_reads/HI.4950.006.IDT_i7_168---
IDT_i5_168.Xylocopa_virginica_unworn_female_R2.fastq"},
```

{"left": "/home/jt18hz/scratch/Virginica_reads/HI.4950.006.IDT_i7_156---IDT_i5_156.Xylocopa_virginica_unworn_male_R1.fastq",

"right": "/home/jt18hz/scratch/Virginica_reads/HI.4950.006.IDT_i7_156---IDT_i5_156.Xylocopa_virginica_unworn_male_R2.fastq"}],

"alignments": [{"bam": "/home/jt18hz/scratch/bwa/UWFaln.sorted.bam", "is": "250","std": "73"},{"bam": "/home/jt18hz/scratch/bwa/UWMaln.sorted.bam", "is": "226","std": "64"}],

"software_path": {"bwa": "bwa", "samtools": "samtools", "velvet": "/cvmfs/soft.computecanada.ca/easybuild/software/2017/avx2/Compiler/intel2016.4/velvet/1.2.10/bin/", "kmc": "/home/jt18hz/scratch/KMC/bin/", "TERefiner": "./TERefiner_1", "ContigsMerger": "./ContigsMerger"},

"parameters": {"working_folder": "/home/jt18hz/scratch/GAPPadder", "min_gap_size": "100", "flank_length": "300", "nthreads": "32", "verbose": "1"},

"kmer_length": [{"k": 30,"k_velvet": [{"k": 29},{"k": 27}]},{"k": 40,"k_velvet": [{"k": 39},{"k": 37}]},{"k": 50,"k_velvet": [{"k": 49},{"k": 47}]}]}

Run command:

```
python ./main.py -c All -g config_edited.json
```

Gapcloser run command:

```
./GapCloser -a /home/jt18hz/scratch/GAPPadder/new_UWM_seq -b
/home/jt18hz/scratch/SOAP-runs/config_fileUWF -o
GapCloser_output -l 151 -t 32
```

Bwa-mem run command:

```
bwa mem -t 32
/home/jt18hz/scratch/ref_seq/reference_assembly.500.fasta
/home/jt18hz/scratch/Virginica_reads/10X_reads/A1_S12_L008_R1_00
1.fastq
/home/jt18hz/scratch/Virginica_reads/10X_reads/A1_S12_L008_R2_00
```

```
1.fastq | samtools view -Shb -@ 32 -o
/home/jt18hz/scratch/variant_call/10x_to_ref/10x.bam –
```

Pblat run command:

```
pblat -dots=1000000 -threads=24
/home/jt18hz/scratch/ref_seq/reference_assembly.500.fasta
/home/jt18hz/scratch/unworn_female/SOAP_new/Unwornfemale_out_500
.fasta unwornfemale_output.psl
```

Variant calling Preprocessing:

SortSAM run command:

```
java -jar $EBROOTPICARD/picard.jar SortSam
I=./ref500/${1}.500.bam O=${1}_500.sorted.bam
SORT_ORDER=coordinate
```

Remove duplicates run command:

```
java -jar $EBROOTPICARD/picard.jar MarkDuplicates
I=${1}_500.sorted.bam
O=/home/jt18hz/scratch/variant_call/read_preprocessing/ref500/${
1}_markdups_500.bam
M=/home/jt18hz/scratch/variant_call/read_preprocessing/ref500/${
1}_marked_dup_metrics_500.txt
```

Add read groups run command:

```
java -jar $EBROOTPICARD/picard.jar AddOrReplaceReadGroups
I=Unwornmale_markdups_500.bam O=Unwornmale_markdupsrg500.bam
RGID=2 RGLB=Lib1 RGPL=illumina RGPU=HVJ7CCXY.6.NCAAGACT+NTCGGTAA
RGSM=uwm
```

Indexing run command:

```
samtools index
```

```
/home/jt18hz/scratch/variant_call/read_preprocessing/ref500/Worn
male_markdupsrg500.bam
```

LACER tool run command:

```
samtools index Wornmale_markdups.bam
```

```
lacer.pl -bam Wornmale_markdups.bam -reference
/home/jt18hz/scratch/ref_seq/reference_assembly.fasta -rgfield
PU -output Wornmale_recal.txt
```

Haplotypecaller run command:

```
gatk --java-options "-Xmx30G" HaplotypeCaller -R
/home/jt18hz/scratch/ref_seq/reference_assembly.500.fasta -I
/home/jt18hz/scratch/variant_call/read_preprocessing/ref500/Worn
male_markdupsrg500.bam -O HTC_Wornmale.vcf
```

Mutect2 run command:

```
gatk --java-options "-Xmx150G -Xms40G" Mutect2 -R
/home/jt18hz/scratch/ref_seq/reference_assembly.500.fasta -I
/home/jt18hz/scratch/variant_call/read_preprocessing/ref500/Worn
male_markdupsrg500.bam -I
/home/jt18hz/scratch/variant_call/read_preprocessing/ref500/Unwo
rnmale_markdupsrg500.bam -O MT2_Wornmale.vcf
```

Appendix B Perl Scripts

Fastq_split.pl script:

```
#!/usr/bin/perl -w

##########################################################################

# This script splits a FASTQ format file into smaller chunks

##########################################################################

use strict;

use Getopt::Std;

my %opt;

getopts("p:n:t:z:h:",\%opt);


(@ARGV==1 && ! (! $opt{n} && ! $opt{p}) && ! $opt{h}) or die

"Usage: $0 [-p num_parts] [-n num_entries/part] <-t totalEntryNum><-z1 gzout>
fastq_file

     options:

     -p: number of parts (required if no -n)
```

```
     -n: number of entries per part (required if no -p)
     -t: total number of fastq entries (optional)
     -z: gz compress output
     notes: input fastq can be gz compressed, use -p1 and -n X to extract the
top X entries and leave out the rest.\n
";

my ($infile)  = @ARGV;
my $gz;
if ($infile=~/\.gz$/){$gz=1}
my ($base,$ext)=$infile=~/(^.*?)\.(fq|fastq)/;

my ($ne,$np,$nfq); #number of entries/part, number of parts, total number of
fastq entries
if ($opt{t}){$nfq=$opt{t}}
else{
    if ($gz){$nfq =`zcat $infile |wc -l |cut -d " " -f 1`}
    else{$nfq =`wc -l $infile |cut -d " " -f 1`}
    chomp $nfq; $nfq=int($nfq/4);  #obtain the total number of fast
    print STDERR "$infile has $nfq entries\n";
}
if ($opt{n}){
    $ne=$opt{n};
    $np=int($nfq/$ne);
}else{$np=$opt{p}}

my $nep; #number of entries per parts
if ($ne){$nep=$ne} #usee the user specififed number
else{ #determine the number of entries per part
    if($nfq%$np ==0){$nep=$nfq/$np}
    else{$nep=int($nfq/$np)+1}
}
if ($opt{p} && $opt{n}){$np=$opt{p}}
```

```
if ($gz){open(IN, "zcat $infile |") or die "cannot open $infile: $!\n";}
else{open(IN, "<$infile") or die "cannot open $infile: $!\n"}


my ($NP,$C)=(0,0);  #keep track of parts and total number of entries
generated


while ($NP<$np){
    my $NF=sprintf("%02s",$NP); #number of parts to be used in file name for
subparts
    if ($opt{z}){
        open(OUT,"|gzip >${base}_$NF.$ext.gz") or die "can't open
${base}_$NF.$ext\n";
    }else{
        open(OUT,">${base}_$NF.$ext") or die "can't open ${base}_$NF.$ext\n";
    }
    my $c=0;
    while($c<$nfq && $c<$nep){
        my $fq= <IN>.<IN>.<IN>.<IN>;
        print OUT $fq;
        $c++; #number of entries in the current part
        $C++; #total number of entries processed
    }
    $NP++
}
close IN;
close OUT;
exit 0;
```

`fatools.pl` script:

Refer to Github page: https://github.com/pliang64/fatools

Combining VCF and PSL output script:

```
#!/usr/bin/env perl


#this script takes 2 in files, a psl and vcf file for converting the variant position in scaffold

#to genome location based on the blat result of the scaffolds to the genome


use strict;

if (!@ARGV || @ARGV !=2){die "Usage: $0 pslFile VCFfile\n"};


( -s $ARGV[0] && $ARGV[0]=~/psl$/ ) or die "$ARGV[0] doesn't exist or zero in size or not in psl format.\n";

( -s $ARGV[1] && $ARGV[1]=~/vcf$/ ) or die "$ARGV[1] doesn't exist or zero in size or not in vcf format.\n";


my (%SF2GENOME,%BS,$lpsl,$nsf,@SF);

open(PSL, $ARGV[0]) or die "$!\n";

while (<PSL>){#process psl file and collect the best match

    if ($_!~/^\d+/){next}

    my @f =split /\t/;

    my ($qid,$qs,$qe,$tchr,$ts,$te,$st)=@f[9,11,12,13,15,16,8];

    if (!$BS{$qid} || $BS{$qid}<$f[0]){#update if better blat score match is seen

        $SF2GENOME{$qid}={QS=>$qs,QE=>$qe,TC=>$tchr,TS=>$ts,TE=>$te,ST=>$st};

    }

    $lpsl++;

}

close PSL;

$nsf=scalar (keys %SF2GENOME);

print STDERR "$nsf unique scaffold was processed from $lpsl lines in $ARGV[0].\n";


my (@VAR,$nv,$lostv,$tv,%seenBK);

open(VCF, "<$ARGV[1]") or die "$!\n";

while (<VCF>){
```

```
    if ($_=~/^#/){next}#skip vcf header
    $tv++;
    my @f=split /\t/;
    my($sf,$s,$ref,$alt)=@f[0,1,3,4];
    my ($gid,$gs,$ge,$vtype,$val);
    if (length($ref) ne length($alt)){$vtype=2}else{$vtype=1}
    my $r=$SF2GENOME{$sf};
    if (!$r){$lostv++; next}
    my
($qs,$qe,$st,$tch,$ts,$te)=($r->{QS},$r->{QE},$r->{ST},$r->{TC},$r->{TS},$r->
{TE});
    $val="$sf|$s|$ref:$alt";
    $gid=$tch;
  if (!$seenBK{$sf}){#adds two breakpoints for each scaffold only once
          push @VAR, {GID=>$gid,GS=>$ts,VT=>3,VA=>"$sf|$qs-$qe|$qs",ST=>$st};
          push @VAR, {GID=>$gid,GS=>$te,VT=>3,VA=>"$sf|$qs-$qe|$qe",ST=>$st};
          $seenBK{$sf}=1
      }
      if ($st eq "+"){#scaffold aligned with the gneome on plus strand
          $gs=$ts+$qs+$s-1;
      }else{#scaffold aligned with the gneome on plus strand
          $gs=$ts+($qe-$s);
      }
      push @VAR, {GID=>$gid,GS=>$gs,VT=>$vtype,VA=>$val,ST=>$st};
  }
  close VCF;


  @VAR=sort {$a->{GID} cmp $b->{GID} ||$a-> <=> $b-> } @VAR; #sort all
variants by genomeID and start position
  $nv =scalar @VAR;
  print STDERR "$nv variants collected from $tv in $ARGV[1] with $lostv
entries discarded due to no mapping positions.\nNow printing variants in and
by the order of their genomic locations.\n";


  #sort and print all variants in the genome location
```

```
foreach my $v (@VAR){
    print "$v->{GID}\t$v->\t$v->\t$v->{VA}\t$v->{VT}\t$v->{ST}\n";
}
exit 0;
```