Pragmatic approaches for genome analysis of ants and other emerging model organisms

Anurag Priyam

Submitted in partial fulfilment of the requirements of the Degree of Doctor of Philosophy

November 2022

Statement of originality

I, Anurag Priyam, confirm that the research included in this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, this is duly acknowledged in each chapter, and my contribution indicated. Previously published material is also acknowledged in each chapter.

I attest that I have exercised reasonable care to ensure that the work is original and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

Anurag Priyam

28th November 2022

Acknowledgements

I am grateful to my supervisor, Yannick Wurm, for providing this opportunity and for the guidance to see it through to the end. If I succeed in my career, it will be in no small part to what I have learnt from him. I am especially thankful to Yannick for making my PhD very enriching and memorable by supporting me to participate in conferences and courses across Europe. It's the most I ever travelled in my life!

My work would not would have been possible without the generous financial support provided by the college, my department, my supervisor, and my parents. Nor would it have been possible without the guidance provided by my advisors, Andrew Leitch and Stephen Rossiter. Thank you.

I am thankful to Richard Nichols, Yannick, and David Nash for lectures on ecology, genetics and evolution, to my department for the weekly seminars and lab jollies, to the members of Yannick's lab for the weekly lab-meetings, journal clubs, and many stimulating discussions. They provided the broad foundations for my work and contributed greatly to my expertise in bioinformatics.

The whole PhD experience in no way felt easy to me. I am grateful to my family, friends, and colleagues for supporting me throughout. I am especially thankful to my childhood friend Shashank Ghosh and colleague Magdalena Innes Schacht for illuminating some of the very difficult moments. Last but not the least, I am grateful to my wife, Ila, without whom my life in London would have been rather lonely.

Abstract

The dramatic drop in sequencing costs has created many opportunities for novel biological research. Many research questions depend on comparing sequenced reads to a "reference genome" to characterise genes, regulatory regions, genetic variation and gene expression. Typically, the reference genome is computationally assembled *de novo* from reads generated by "shotgun sequencing" and is often the first step in the molecular characterisation of a species. Unfortunately, this process is prone to errors, which results in several regions of the genome to be missing, fragmented, or mis-assembled in the reference. I review the sources of these errors, the challenges in evaluating the quality of *de novo* genome assemblies, present new metrics and a tool to overcome some of the limitations. Furthermore, I show that fine tuning the parameters of assembly software is an effective way to obtain higher quality genome assemblies. However, the quality of genome assemblies is ultimately tied to the length and the error profile of sequenced. I present a tool to facilitate rapid transfer of gene annotations to a new genome assembly with high confidence so that known regions of the genome do not need to be recharacterized. Furthermore, intermediate output of the tool can also be used to transfer variant annotation and other data formats. Finally, I present a graphical interface for the popular BLAST software. Among other things, it is useful for qualitative quality assessment of genome assemblies and gene annotations or to characterise regions of interest in newly assembled genomes. I believe the approaches presented here can play a key role in genomic characterisation of previously understudied organisms. As examples, I present two studies on social evolution in ants where I led specific analyses through the application of above-mentioned tools and related concepts.

Contents

List of figures

List of tables

Chapter 1: Introduction

Biologists can today query many aspects of an organism's genome. Such investigations often depend on comparing reads from various sequencing experiments to a reference genome, where the reference genome itself is derived from shotgun sequencing of the organism's DNA. Thus, obtaining a reference genome is often the first step in the molecular study of a species.

I review the process of obtaining a reference genome in the sections below, with a particular focus on highlighting the limitations and their consequences. This is followed by a summary overview of my thesis.

Genome sequencing and assembly

The genetic information of an organism is encoded in long molecules of deoxy ribonucleic acid (DNA). These are carried in thread-like structures inside the nucleus of each cell of the organism, called the chromosomes. DNA molecules consist of two complementary chains of nucleotide molecules that wrap around each other to form a double-helix structure. Four nucleotide molecules make up the chains: 'adenine', 'thymine', 'guanine', and 'cytosine'. The sequence of nucleotide molecules that make up the chain can be read as a string of 'A', 'T', 'G', and 'C' through a process known as sequencing. Sequencing technologies, however, can only read DNA molecules that are much smaller than the chromosomes of most organisms. Chromosomes are thus broken down into smaller fragments for sequencing and computationally reconstructed afterwards. This process is referred to as genome assembly.

DNA for genome assembly is ideally extracted from an inbred individual where large regions of homologous chromosomes are expected to be near identical. If it is difficult to rear the organism in laboratory conditions for inbreeding (e.g., pandas) or where enough DNA cannot be extracted from a single individual (e.g., small insects), DNA from wild-type or more than one individual may be used. However, there can be considerable differences

between the homologous chromosomes of wild-type individuals which must additionally be resolved by the assembly process (this allelic diversity further increases if DNA from several individuals is pooled together). The DNA is ideally extracted from gametes so that the reference represents the germline, although DNA from other tissue types or whole-body may also be used.

The extracted DNA is subject to a series of processing steps and then loaded on to the sequencing machine. The specifics depend on the sequencing technology but follow the general process of breaking the DNA into overlapping fragments, selecting DNA fragments of appropriate size, and ligating them to short "adapter sequences" that help initiate the sequencing reaction. This process is known as "library preparation". In some cases, it may be required to break the DNA into a single strand. The sequence of nucleotides is then determined through synthesis of the complementary strand – four fluorescent-labelled nucleotides are introduced in the reaction chamber and emission spectra of the nucleotide that incorporates into the chain is observed. Illumina and Pacific Biosciences take this approach. Alternatively, the sequence of nucleotides may be determined by pushing DNA fragments through a protein nanopore and monitoring the electrical current generated by this process, an approach pioneered by Oxford Nanopore technologies.

Sequencing technologies are not perfect. Mistakes are made in the process of reading the sequence of nucleotides in a DNA fragment. For example, a nucleotide 'A' in the DNA may be replaced by nucleotide 'G' in the sequenced read. This is termed 'substitution error'. However, insertion or deletion of a few (2-4) nucleotides compared to the input DNA are also observed. These are called 'indel' errors. The rate of substitution and indel errors depend on the sequencing technology. It can be as high as 15% for Pacific Biosciences and Oxford Nanopore sequencing to lower than 1% for Illumina sequencing (Schirmer *et al.*, 2016; Weirather *et al.*, 2017). Other sequencing errors include "adapter contamination" and "chimeric reads". Adapter readthrough is when the sequencing process reads through the end of DNA fragment and into the adapter sequence resulting in a portion of adapter sequence to remain fused with the read (Martin, 2011), while chimeric reads are formed by accidental fusion of two different DNA fragments (drive5.com/usearch/manual/chimeras).

Errors in genome assemblies

The outcome of whole-genome sequencing is readouts of the DNA fragments as a series of 'A', 'T', 'G', and 'C'. Depending on the sequencing technology, the reads may be of fixed length, as in Illumina sequencing, or of variable length as in Pacbio and Oxford Nanopore sequencing. Genome assembly works by aligning pairs of reads to determine the overlap between them. Overlaps between the reads are termed 'containment' when all bases in a read align with the other, or 'dovetail' when the overlap only involves the ends of the reads. Put simply, the genome is reconstructed by stitching together reads with dovetail overlaps and using both dovetail and containment overlaps to identify and correct sequencing errors along the way. This process should ideally give us as many sequences as the number of chromosomes in the organism and each base of the sequence would be exactly same as in the chromosome. However, except for a few species with very simple genome, this picture is far from reality.

For most eukaryotes, several regions of the genome end up fragmented, missing, or mis-assembled. This is because eukaryotic genomes are repetitive, meaning we find the exact same or near-identical sequences several times in the genome. To understand why repeats are problematic, let us assume that the sequenced reads are error free to begin with. Now, if the reads are shorter than the length of the sequence that is repeated in the genome, we can see how reads from exact copies of the repeat will cluster together during overlap detection. Accordingly, the different repeat copies are 'collapsed' into one in the assembly and the regions containing the other copies of the repeat are split into two. This is most often the case when the repeated sequences are interspersed. When the repeat copies are in tandem, other copies may be collapsed without fragmentation as commonly observed with large stretches of short repeat sequences (such as tandem array of di- and tri-nucleotide motifs). Finally, when the repeat copies are near-identical, e.g., with just one base difference between them, it is easy to see how the repeat copies can end up being 'shuffled' or 'rearranged' in the resulting assembly. If the shuffled repeats had opposing orientations, their shuffling can further result in the region between the repeats to be inverted (Phillippy, Schatz and Pop, 2008, fig. 4).

Sequencing error makes it harder to distinguish between exact or near exact and diverged repeats. Consider the ~15% sequencing error rate of Pacific Biosciences or Oxford Nanopore. All alignments with identity >70% (100 – 2 x 15) must be considered by the assembler or true overlaps may be missed. Now remember how each region of the genome is sequenced multiple times. Assuming a random error model (which fits our use case very well), not all reads will have an error at the same base. This knowledge is used to correct reads prior to assembly. In case of Pacific Biosciences and Oxford Nanopore, read correction can decrease the sequencing error considerably, thus allowing us to distinguish between repeat copies with up to 2-3% divergence (Koren *et al.*, 2017). Inability to resolve repeat copies beyond this limit results in collapse and fragmentation, or rearrangement. Furthermore, a mistake in distinguishing repeat copies during error correction can homogenise their reads. (Nurk *et al.*, 2020) indicate this can happen but do not provide any details. We can suppose that repeat homogenisation will further contribute to collapse and fragmentation, or rearrangement.

Collapse is the most common reason for missing sequences and assembly fragmentation, but not the only contributing factor. For each region of the genome, there must be enough reads to reliably distinguish true overlaps from false, repeat or sequencing-error induced overlaps. However, biases introduced during library preparation and sequencing can result in some regions of the genome to have less reads than the others. If the "read coverage" of a region falls below a threshold it may become impossible to reliably reconstruct the region. Accordingly, the region will be excluded from the assembly and the surrounding region fragmented into two. Ross et al. (Ross *et al.*, 2013) provide a detailed overview of the sources of coverage bias. A common reason is polymerase chain reaction (PCR) based DNA amplification step during library preparation and sequencing which lower the coverage of GC- and AT-rich regions of the genome. GC- and AT-rich regions also tend to have a higher proportion of sequencing errors than average.

Other assembly errors include "split alleles" and "consensus errors". Split alleles are like the opposite of collapse. Regions of homologous chromosomes that are diverged above a certain threshold may be assembled twice, one copy for each allele (Hahn, Zhang and Moyle, 2014). Consensus errors are residual sequencing errors. Despite error correction of reads and afterwards, error correction of the assembled sequences, some sequencing errors make it

through, leading to substitution and indel (consensus) errors in the assembly. Finally, efforts to place assembled sequences in their chromosomal order using linkage, optical or other form of chromosomal maps can result in mis-joins, inversions, and erroneous splitting.

Impact of assembly errors on genomic analyses

Errors in genome assemblies reduce the sensitivity and specificity of downstream analyses. Genomic regions missing in the assembly are difficult to characterise (Chapter 7). If the missing regions are the result of a repeat collapse, reads from the collapsed copy can map to the copy present in the assembly and result in small differences between the repeat copies to be incorrectly identified as polymorphisms (Vollger *et al.*, 2019). Assembly fragmentation can lead to underestimation of synteny relationships (Liu, Hunt and Tsai, 2018). While fragmentation, rearrangements, consensus errors and mis-joins can all lead to errors in gene prediction (Alkan, Sajjadian and Eichler, 2011; Florea *et al.*, 2011; Denton *et al.*, 2014). Finally, split alleles can create the illusion of gene or segmental duplications (Kelley and Salzberg, 2010).

Thesis overview

My thesis is divided into two parts. In Part 1, I present software tools and workflows to overcome some of the limitations of genome assemblies. While in Part II, I present two collaborative studies on social evolution in ants which provided the inspiration for software presented in Part I.

Part I

In Chapter 2, I discuss the challenges in evaluating the quality of *de novo* genome assemblies, present new metrics and a tool to overcome some of the limitations. Furthermore, I show that fine tuning the parameters of assembly software is an effective way to obtain higher quality genome assemblies.

In Chapter 3, I present a tool to transfer gene annotations with high confidence to a new, improved genome. Furthermore, intermediate output of the tool can also be used to transfer variant annotation and other data formats.

In Chapter 4, I present a graphical interface for the popular BLAST software. Among other things, it is useful for qualitative assessment of genome assemblies and gene annotations or to characterise regions of interest in newly obtained genomes.

In Chapter 5, I present a software and related workflows for quality assessment, filtering, and manual curation of gene annotations.

Part II

In Chapters 6 and 7, I present two studies investigating social evolution in the red fire ants, *Solenopsis invicta*, and in the big-headed ant, *Pheidole pallidula*. Both species have two types of colony organisation, either headed by a single queen or by multiple queens. In 2013, Wang and Wurm showed that the social polymorphism in fire ants is linked to a large region of suppressed recombination on chromosome 16, akin to Y-chromosome (Wang *et al.*, 2013). In 2017, I took part in a study led by Rodrigo Pracana to characterise the differences in odorant-binding proteins (OBPs) between the two social forms of fire ant, an important class of proteins responsible for communication in ants (Chapter 6). While for the big-headed ant, we investigated if the social polymorphism is linked to a Y-like chromosome similar to fire ants (Chapter 7), i.e., does evolution often take the same genomic path towards phenotypic convergence? However, the studies were fraught with high-level of assembly fragmentation and missing sequences. This set me on the path to understand their cause (presented in this chapter) and devise ways to overcome associated challenges (presented in Part I of the thesis).

Part 1: Methods

Chapter 2: Parameter exploration improves the accuracy of long-read genome assembly

Contributions

I led the design of the study and did most of the work. For some specific parts of the project, I reached out to others for their relevant expertise: Alicja Witwicka performed the test for statistical significance, Anindita Brahma performed the test for foreign DNA contaminants, Eckart Stolle collected the ants, genotyped them and extracted the DNA. All steps of the work were done under the guidance of Yannick Wurm. I wrote the chapter. Eckart provided very helpful comments on an initial draft. Yannick provided valuable guidance throughout the writing. Everyone involved read and contributed to improving the manuscript later.

The chapter is intended for submission as a Methods article to Genome Research:

A Priyam, A Witwicka, A Brahma, E Stolle, Y Wurm (in prep)

Abstract

Long-read sequencing is now routinely applied to generate high-quality reference genome assemblies. However, datasets differ in repeat composition, heterozygosity, read lengths and error profiles. The assembly parameters that provide the best results for any particular dataset could thus differ from the default settings of the assembly software. To determine the potential benefits of optimising assembly, we generated 44x genome coverage of Pacbio long-molecule sequences for the invasive red fire ant *Solenopsis invicta*. From this dataset, we generated 36 assemblies using the Canu software by systematically varying three key parameters that affect how the software handles raw sequence reads. We compared the generated assemblies using four complementary metrics: contiguity, presence of expected single-copy genes, resolved assembly length, and concordance with independently generated short Illumina sequences. We find that the assemblies vary considerably in terms of all four metrics, and that more than half of the parameter combinations led to higher assembly qualities than when using default parameters. The best assembly had 22% higher contiguity, 12.8% more of the expected single-copy genes, 0.2% higher concordance with Illumina sequences and was 1.8 Mb longer than if using default parameters. Our results demonstrate the benefits of fine-tuning assembly parameters. Furthermore, we provide a practical framework and a generic analysis tool for researchers wanting to pragmatically compare and choose among multiple assemblies.

Introduction

High-quality genome assemblies are essential for modern biological research (Schneider *et al.*, 2017). They serve as the reference for annotating genes and other genomic features (Raymond *et al.*, 2018; Shields *et al.*, 2018), identifying genetic and epigenetic variation (Kronenberg *et al.*, 2018), and quantifying gene expression (Srivastava *et al.*, 2019). Assemblies are in turn crucial for characterizing the genetic architecture of complex traits (Nadeau *et al.*, 2016) and patterns of genome structure evolution (Wicker *et al.*, 2018). Unfortunately, eukaryotic genome assemblies typically contain major errors. This is because eukaryotic

genomes include large amounts of repetitive sequences (Schatz, Delcher and Salzberg, 2010) that are difficult to resolve due to limitations of sequencing processes and assembly algorithms. The inability to resolve repetitive sequences leads to assembly fragmentation (Ye *et al.*, 2011), to collapsing of multiple occurrences of repetitive sequence into fewer assembled sequences (Alkan, Sajjadian and Eichler, 2011), and to misassembly of repetitive regions (Phillippy, Schatz and Pop, 2008). Such shortcomings of genome assemblies reduce the sensitivity and specificity of downstream analyses. For example, assembly fragmentation can lead to underestimation of synteny relationships (Liu, Hunt and Tsai, 2018), and to errors in gene prediction (Alkan, Sajjadian and Eichler, 2011; Florea *et al.*, 2011; Denton *et al.*, 2014). Furthermore, when the sequence reads from different copies of a repetitive element map to a collapsed representation of the repeat, small differences between the repeat copies can be incorrectly identified as polymorphisms (Vollger *et al.*, 2019).

Long-molecule sequencing has the potential to dramatically improve genome assemblies (Miga *et al.*, 2020). In particular, long reads can capture entire tandem arrays of repetitive elements, thus resolving such regions (Koren *et al.*, 2013). Furthermore, single-molecule long-read sequencing technologies from Pacific Biosciences and Oxford Nanopore are more robust to variation in GC composition than short-read technologies (Rhoads and Au, 2015). However, the ability of assembly software to reconstruct the correct genome sequence can depend on the dataset (Mikheenko *et al.*, 2018; Kolmogorov *et al.*, 2019) and on the algorithmic parameters used (Conte *et al.*, 2017; Minio *et al.*, 2019; Zhang, Jain and Aluru, 2019). Such variations likely arise because whole-genome sequence datasets differ in characteristics including repeat composition, heterozygosity, read lengths and read error profiles, whereas assembly software defaults are based on particular datasets. This suggests that testing different assemblers and assembly parameters may be advantageous. But how can *de novo* assembly projects apply this knowledge?

An exhaustive search of the parameter space of most assemblers is impractical because assemblers can have dozens of continuous parameters. Fundamentally, assemblers work by determining overlaps between pairs of reads and stitching together reads that overlap the best (Myers *et al.*, 2000). For the popular Canu and FALCON assemblers, Conte et al.'s work

(Conte *et al.*, 2017) suggests that modifying minimum read length and minimum overlap length parameters can affect assembly quality. Another parameter that may similarly affect assembly quality is the estimate of sequencing error. If the true sequencing error is higher than the estimate used by the algorithm, then true overlaps between reads may be missed. This would fragment the assembly. Alternatively, if the true sequencing error is lower than the estimate used by the algorithm, the number of false overlaps may increase. This can lead to assembly fragmentation, collapse, or mis-assembly of repetitive regions.

Choosing the best of multiple *de novo* genome assemblies is challenging. An assembly is better if it is more contiguous, complete, and accurate. The N50 metric provides a straightforward view of contiguity despite lacking direct biological relevance. Similarly, testing for the presence and completeness of protein-coding genes from related organisms (Simão *et al.*, 2015) or concordance with transcriptomic data (Riba-Grognuz *et al.*, 2011; Denton *et al.*, 2014) can indicate completeness and accuracy in genic regions. However, genome-wide measures of completeness or accuracy are less immediate. Most projects lack datasets that are ideal for such comparisons, including sequences from independent fosmid or BAC libraries, high-resolution genetic, optical, or chromatin interaction maps, or a high-quality reference assembly. Although of lower resolution, independently derived Illumina DNA sequences can be used in such cases due to the ubiquity of Illumina sequencing. Indeed, mapping of short insert size Illumina DNA sequences can detect structural errors in an assembly (Khelik *et al.*, 2020) or provide a base-by-base view of consensus accuracy (Thomas and Hahn, 2019). But there is a considerable overhead in applying tools implementing such ideas, interpreting their output and summarising them into general statements of assembly completeness and accuracy.

To test the impact of parameter optimisation on assembly quality and to establish a simple approach for selecting the best assembly, we obtained Pacbio reads for the red fire ant, *Solenopsis invicta* and generated 36 assemblies using Canu (Koren *et al.*, 2017). This species is a model for the study of social behaviour, and a globally invasive pest (Tschinkel, 2006). The draft genome assembly for this species (Wurm *et al.*, 2011) has been cited more than 300 times despite its high fragmentation (69,511 sequences) and capturing only 79% of the genome (estimated to be 450 Mb (Stolle *et al.*, 2019)). Importantly, the fragmentation and the missing

sequences affect genomic regions involved in environmental perception (Pracana, Levantis, *et al.*, 2017; Venthur and Zhou, 2018), complex behavioural and developmental traits (Privman, Wurm and Keller, 2013; Wang *et al.*, 2013; Buechel, Wurm and Keller, 2014; Pracana, Priyam, *et al.*, 2017; Martinez-Ruiz *et al.*, 2020), differences between long- and short-lived individuals, and potential pesticide targets (Venthur and Zhou, 2018). To compare the generated assemblies, we used four complementary metrics that characterise assembly completeness, contiguity, and accuracy. We show that varying error thresholds for finding overlaps between reads greatly improves contiguity and accuracy of Canu assemblies. Lastly, we provide a simple, generic tool that can be used to similarly select among multiple assemblies.

Results

Thirty-six assemblies by varying three key Canu parameters

We obtained 2.9 million Pacbio reads, totalling 20.2 billion bases (45x genome coverage) from a diploid sample of *S. invicta (*N50 read length of 8,876 bp; Figure 2.S1). We first assembled this dataset using default parameters of Canu. We then generated 35 additional assemblies to test the effects of three parameters (full details in Table 2.S1). We varied the estimated raw overlap error rate, using values corresponding to sequencing error rates of 12.5%, 13.75%, 15% (default), 16.25%, and 17.5%. We varied the stringency of trimming raw reads, requiring a minimum of 4 overlaps (default), a more relaxed setting of 2 overlaps, and disabling trimming of raw reads altogether. Finally, we varied the estimated error rate of the "corrected reads" generated by Canu using values corresponding to corrected error rates between 1.15% and 5.87% (default: 2.25%). We "polished" the consensus sequence of each assembly (Chin *et al.*, 2013) removed unresolved haplotypes (Roach, Schmidt and Borneman, 2018) to minimise the impacts of residual errors on measurements of assembly quality (Table 2.S2).

Measures of assembly contiguity, accuracy and completeness

To compare the 36 genome assemblies, we obtained four metrics of assembly quality. We first calculated NG50, which is the N50 metric normalised by estimated genome size. Second, we determined how many of the 4,415 expected single-copy genes are present and complete (Simão *et al.*, 2015). Third, we generated and mapped short-read Illumina sequences from a PCR-free library from two closely related individuals to each assembly. This mapping enabled us to measure the resolved length of each assembly, which we defined as the regions of the assembly that have greater than 5-fold coverage but less than twice the median coverage of the assembly (Figure 2.S2). Resolved length metric improves the total assembly length metric to show how much of the genome is potentially usable for analysis through standard approaches and how much is assembler "chaff" (Salzberg *et al.*, 2012). Indeed, regions with particularly low coverage can contain high amounts of sequencing errors, whereas regions with particularly high coverage typically contain collapsed repeats. Finally, we measured the number of solidly mapped Illumina reads, meaning reads that mapped to the resolved regions in their entirety (i.e., without clipping) and within the expected distance and orientation of its mate (i.e., concordantly), as a percentage of total reads. Clipped and non-concordant mapping patterns for large numbers of reads occur when there are assembly errors such as mis-joins, inversions, collapses, and consensus errors (Liu *et al.*, 2015). The reason we exclude unresolved regions is because mappings in such regions are noisy: we cannot be sure if the reads truly originated from that region. The reason we divide by the number of total reads instead of the number of mapped reads is that the former is closer to the ground truth. After all, the entirety of whole-genome sequencing output should map to a completed genome assembly. An advantageous side-effect of following this approach is that the metric simultaneously summarises assembly completeness and accuracy in a single number. Furthermore, unlike likelihood-based approaches (Rahman and Pachter, 2013), the metric is meaningful in a non-comparative context: the value would tend to 100% for a perfect assembly. This makes solid pairs a useful metric to show assembly quality in centralised databases.

Four complementary metrics reveal extensive variation in assembly quality

We found a 2.3-fold difference in the NG50 metric of contiguity between assemblies (237,734 bp to 543,457 bp) and 1.4-fold variation in the number of missing or incomplete single-copy genes (141 to 202). Furthermore, resolved assembly lengths vary up to 12.6 Mb, *i.e.,* by up to ~2.8% of genome size. Finally, there was a 2.6% range in the proportion of Illumina read pairs that map concordantly to resolved regions of the assemblies. These four measurements of assembly quality have positive but weak correlations (average 0.66), highlighting their complementarity and the importance of considering multiple measures of genome quality (Figure 2.S3).

To select the best assembly, we summed the ranks of the assemblies in each metric, weighted by the complement of the average correlation of the metric with other metrics (Figure 2.1). Twenty-three assemblies (64%) had higher overall quality than obtained through default parameters. In particular, the best ranked assembly had 17.2% higher NG50 (518,074 vs 441,945 bp), had 11.3% less missing or incomplete expected single-copy genes (141 vs 159), had 1.8 Mb higher resolved length and had 0.33% more Illumina reads mapping correctly (57.81% vs 57.62%) than the default assembly. This best ranked assembly was based on a raw error rate of 13.75%, no trimming of raw reads, and a corrected read error rate corresponding to 3.45%.

In this experiment, the estimated error rate for corrected reads had the most significant impact on the overall assembly quality (generalised linear model; $p < 10^{-5}$), followed by the estimated error rate for raw reads ($p < 0.05$). There was no general trend for the impact of raw read trimming on assembly quality ($p = 0.5$).

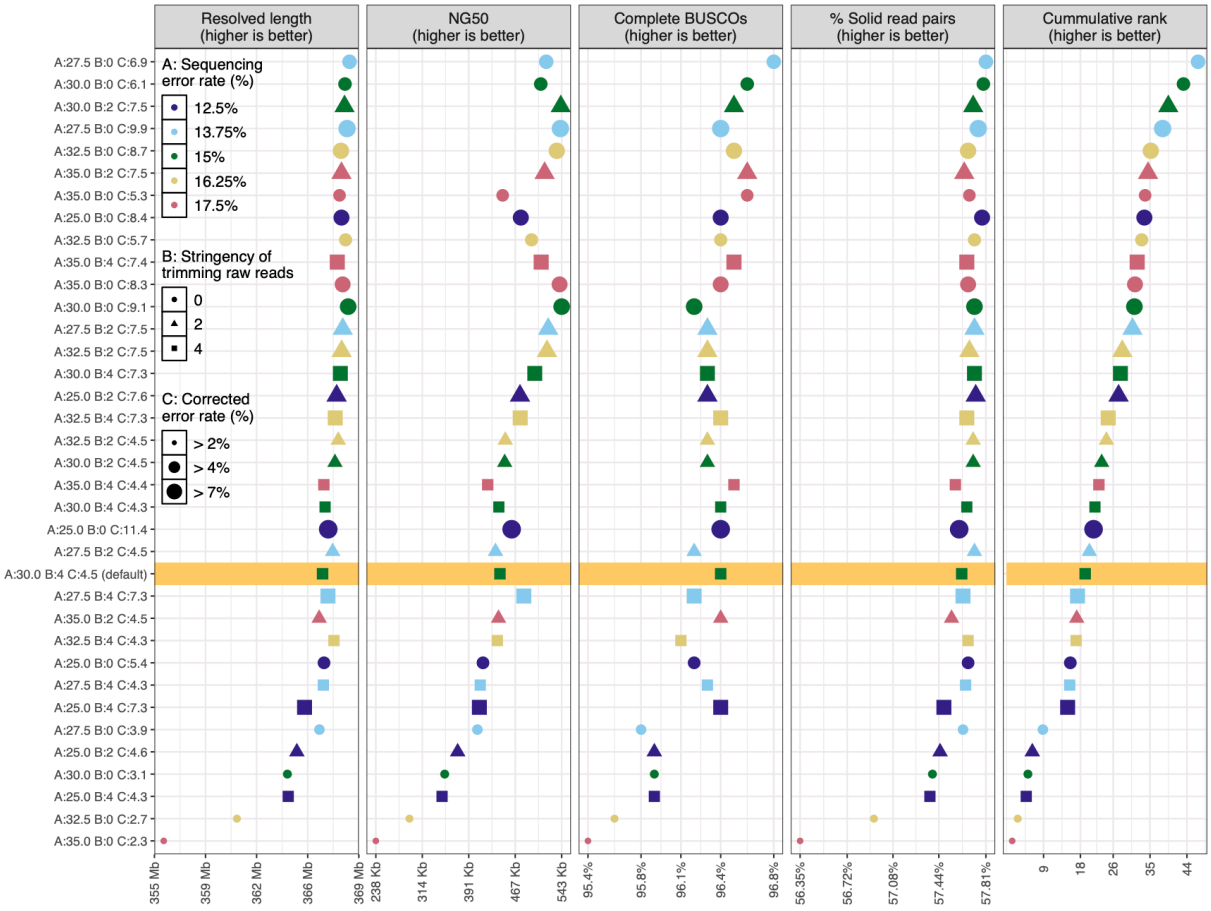


Figure 2.1: Thirty-six assemblies compared using four measures of assembly quality

Thirty-six polished genome assemblies ordered on the y-axis from best (top) to worst (bottom) based on the weighted sum of their ranks (rightmost panel) in each of the four metrics (other panels). The x-axis shows the range of values of each metric, or of the weighted rank in case of the rightmost panel. The assembly generated using default parameters is highlighted in yellow. Twenty-three assemblies scored higher than the 'default' assembly.

Processing and chromosome-level scaffolding of best assembly for use as the reference assembly

To make the best assembly suitable for use as the reference assembly for the red fire ant, we improved the consensus sequence by mapping short read population-sequencing datasets (270x genome coverage (Stolle *et al.*, 2019; Martinez-Ruiz *et al.*, 2020)) to correct residual sequencing errors (Logsdon, Vollger and Eichler, 2020) and replace rare alleles in the

assembly (Ballouz, Dobin and Gillis, 2019). Additionally, we removed likely contaminating contigs that appear to be from bacteria, fungi or plants (Table 2.S3). Finally, we ordered and oriented the contigs into chromosome-level scaffolds using genetic maps, complemented by optical maps, and paired RNA sequencing reads. The resulting assembly captures 347 Mb of the fire ant genome in 16 chromosomes, with 10% of the assembly being in 916 unplaced contigs (Figure 2.S4). This is the most contiguous, accurate and complete genome assembly of the red fire ant *Solenopsis invicta* (Table 2.S4). A comparison with the draft genome assembly of the species shows the inclusion of many more sequences into chromosomes (Figure 2.S5).

Discussion

Estimates of sequencing error is a key parameter for optimisation

Although genome assemblers can produce relatively accurate consensus sequences from long-molecule sequences, we show that small changes in parameters that indicate estimated sequencing errors substantially improve assembly contiguity, completeness and accuracy. This is likely because such fine-tuning helps to resolve lower-complexity regions of the genome. Our general finding that tuning these parameters improves assembly outcomes should similarly apply to other datasets. However, the specific levels of these parameters and their impact will depend on dataset specific features including repeat composition of the genome and the lengths and the error profiles of sequenced reads. For example, we obtained the highest quality assembly from a different fire ant Pacbio dataset by increasing the overlap error thresholds for raw reads by 2% and decreasing the overlap error threshold for corrected reads by 0.5% (data not shown).

Tool for comparing genome assemblies and selecting the best one

Our work also shows the importance of considering multiple metrics that can reveal independent aspects of assembly quality. Our approach of weighing the metrics by their

relative independence provides a robust framework for comprehensive comparison of assembly quality. To simplify the application of our genome comparison approach we have created a standalone tool, CompareMyGenomes, that will derive these complementary metrics and rank the assemblies based on weighted sum of ranks, producing summary tables and figures analogous to our Figure 1. This tool is agnostic to the sequencing approach: as inputs it requires a set of genome assemblies, and a set of paired Illumina sequences (Supplementary methods). Additional metrics (Mikheenko *et al.*, 2018) can be included for ranking and visualisation using simple tabular files. This tool can become key for effectively and efficiently obtaining high-quality assemblies for the thousands of species whose genomes are now being sequenced.

Methods

Sample collection and sequencing

We collected male pupae of the fire ant *Solenopsis invicta* from one single-queen colony from Campo Grande, Brazil (GPS coordinate: 20°38'46.85"S 50°38'36.58"W, permit number: 14BR015531/DF). Since the pupae are from a single-queen colony they are full brothers. Males of this species are haploid, while the females are diploid. Samples were flash-frozen and preserved at -80° centigrade until further processing. Species was confirmed using partial sequencing of the mitochondrial cytochrome c oxidase I gene and colony organisation (i.e., single- or multiple-queen) was verified using a Gp-9 marker assay (Stolle *et al.*, 2019).

Pacbio sequencing of a pool of 21 haploid brothers for assembly

We extracted DNA from the twenty-one haploid brothers (whole body) using a CTAB-phenol-chloroform protocol (Hunt and Page, 1992). From this DNA, the Centre for Genomics Research in Liverpool prepared a SMRT library with a size selection of 10 kb and sequenced the library using 5 SMRT cells on a Pacbio Sequel (V2 chemistry).

Assembly parameters and workflow

We generated a total of 36 assemblies from the Pacbio sequences, using Canu (version 1.6 (Koren *et al.*, 2017)). One assembly was generated using default parameters to serve as a reference point for all comparisons. The remaining 35 assemblies were generated to test the effects of three parameters: overlap error rate for detecting overlaps between raw reads (rawErrorRate), minimum number of overlaps required to not trim or split raw reads (corMinCoverage), and error rate for detecting overlaps between corrected reads (correctedErrorRate). For rawErrorRate we tested the values 0.25, 0.275, 0.30 (default), 0.325, and 0.35 corresponding to sequencing error rates of 12.5%, 13.75%, 15% (default), 16.25%, and 17.5%. For corMinCoverage we tested the values 4 (default), 2, and 0. Zero disables trimming and splitting of raw reads whereas two represents a more relaxed trimming and splitting stringency compared to the default. For correctedErrorRate we tested values specific to each combination of rawErrorRate and corMinCoverage. Specifically, we used the -correct option of Canu to generate corrected reads for the fifteen combinations of rawErrorRate and corMinCoverage. We then estimated error rate of the corrected reads by mapping them to the GCF_000188075.1 reference assembly (Wurm *et al.*, 2011) using minimap2 (2.5-r574 (Li, 2018)) and calculating the total edit distance between the reads and the reference divided by the total number of bases mapped. We only considered highly conserved, single-copy, protein-coding genes for the calculation. This is because we expected that the reads mapping to these regions are extremely unlikely to be mismapped. The genes (n=988) were downloaded from Ensembl BioMart matching the criteria: orthologous to the nematode *C. elegans* and without a paralog. Furthermore, because coding regions of genes may be shorter than the reads and read mapping tools typically provide edit distance of the whole read in the SAM format, we derived the mismatch rate by obtaining a pileup of the reads in the regions of interest using samtools (version 1.4.1 (Li *et al.*, 2009)). The fifth column of the pileup format provided the number of mismatches and the fourth column provided the number of mapped bases. At first, we set correctedErrorRate to twice the estimated error rate (Figure 2.S6) and generated fifteen assemblies, one for each combination of rawErrorRate and corMinCoverage. However, ten out of the first fifteen assemblies came out highly fragmented (N50 < 100 kb). This suggested

to us that there is more noise in corrected reads than estimated. Indeed, for the set of corrected reads obtained using default parameters our estimate of error threshold deviated from the default value by almost 3%. We thus assembled each set of corrected reads twice more by increasing the calculated error threshold by 3% and by 6% and generated 30 more assemblies. From the initial fifteen assemblies, we only retained five that had N50 > 100 kb for further comparison. Overall, we tested values of correctedErrorRate corresponding to error rate of corrected reads between 1.15% and 5.87%.

For all except the default assembly we changed two other parameters from their default values. By default, Canu's read correction step only corrects the longest input reads that would represent 40x genome coverage. However, as trimming of raw reads alone can discard up to 28% of data, we were apprehensive of losing more and disabled such subsetting of input reads by setting corOutCoverage to 100 (canu.readthedocs.io). Additionally, we changed the corMhapSensitivity parameter from "normal" to "high" to increase the sensitivity of overlap detection between raw reads (Berlin *et al.*, 2015).

We polished all assemblies and removed unresolved haplotigs from all assemblies prior to comparison as residual sequencing errors and "unresolved haplotigs" can impact BUSCO and read mapping metrics (Table 2.S2). For polishing, we used raw Pacbio data in BAM format with the SMRTLink software suite (version 5.1.0.26412) which takes into account quality signals inherent to SMRT sequencing (Chin *et al.*, 2013). To remove unresolved haplotigs, we used Pacbio reads with the purge_haplotigs pipeline (commit 0b9afdfd (Roach, Schmidt and Borneman, 2018)) which works on the principle that redundantly assembled loci will have high sequence similarity to some region of the genome and have half the mean genome coverage. Minimap2 (2.5-r574 (Li, 2018)) was used to map Pacbio reads to the assemblies; reads shorter than 1000 bp were discarded prior to mapping. Figure 2.S7 shows coverage histogram of the best assembly before and after running purge_haplotigs.

Assembly quality metrics and ranking

For each assembly, we obtained measures of contiguity, completeness and accuracy. First, we used quast (version 4.6.1 (Gurevich *et al.*, 2013)) to get the NG50 metric of contiguity.

Second, we used BUSCO (version 3.0.1 (Simão *et al.*, 2015)) to determine how many of the genes expected to be present in a single copy in Hymenopteran species (n=4,415) are indeed present and intact in each assembly. This provides a measure of assembly accuracy and completeness in genic regions. For a genome-wide measure of accuracy and completeness, we downloaded Illumina reads derived from a brother of the individuals used for Pacbio sequencing and from another male of a nearby colony (SRA runs SRX4907869 and SRX4907871 respectively (Stolle *et al.*, 2019)). We cleaned the Illumina reads (Supplementary methods) and mapped them to the assemblies using default parameters of bwa-mem (version 0.7.17 (Li, 2013)). Next, for each assembly, we used mosdepth (version 0.2.6 (Pedersen and Quinlan, 2018)) to obtain read depth at each base of the assembly in a BED file. Using custom scripts, we then filtered the bases with depth lower than 5x (assembler chaff) or higher than twice the median coverage (collapsed regions). The number of bases retained after filtering is the resolved length of the assembly, a measure of assembly completeness. Next, we used bedtools (version 2.28.0 (Quinlan and Hall, 2010)) to select reads that mapped to resolved regions of the genome. Finally, using a custom script, we counted the reads that mapped to resolved regions of the genome, were not clipped, and mapped concordantly with respect to their mate. The number of solidly mapped Illumina reads as a percentage of total reads is a measure of assembly accuracy and completeness.

To consolidate the four metrics of assembly quality into an overall assembly rank, we first ranked the assemblies by each metric. We then calculated Spearman's rank correlation coefficient between pairs of metrics and from this, average correlation of a metric with other metrics. Finally, we summed the ranks of the assemblies in each metric, weighted by one minus the average correlation of the metric with other metrics (i.e., complement of the average correlation of the metric).

Determining significance of assembly parameters

We modelled the overall assembly rank as a function of the three assembly parameters (Figure 2.S8). Interaction terms were removed from the model in a stepwise procedure, based on their level of significance. To ensure that the data fit the assumptions of a linear

model, we inspected homoscedasticity, multicollinearity, the relationship between residuals and predicted values, and recognised them as satisfactory across the model.

Removal of residual sequencing errors and rare alleles from the best assembly

To remove residual sequencing errors and rare alleles from the best assembly we used the Pacbio reads and eighteen Illumina whole-genome sequence datasets: all thirteen "bigB" labelled SRA runs from BioProject PRJNA542606 (Martinez-Ruiz *et al.*, 2020) and five such SRA runs from BioProject PRJNA396161 (Stolle *et al.*, 2019). First, we cleaned the Illumina reads as described in the "quality control of Illumina reads" section. Next, we mapped the cleaned Illumina reads to the assembly using bwa-mem (version 0.7.17 (Li, 2013)). Third, we mapped the raw Pacbio reads to the assembly using minimap2 (version 2.17 (Li, 2018)); reads shorter than 1000 bp were discarded prior to mapping. Finally, we used pilon (--fix snps,indels; version 1.23 (Walker *et al.*, 2014)) on the assembly and the resulting alignments to generate a polished assembly.

Identification of foreign DNA in the best assembly

To identify foreign DNA in the best assembly we used Kraken2 (version 2.0.8 (Wood, Lu and Langmead, 2019)) to compare the contigs to NCBI's non-redundant databases of nucleotide sequences (downloaded on April 22, 2020) and 231 new insect viral sequences from the literature (Käfer *et al.*, 2019).

Ordering and orienting contigs

To assign the polished and filtered contigs to one of the sixteen fire-ant chromosomes, we generated genetic maps from RAD sequencing (RAD-seq) of seven families (Wang *et al.*, 2013), and complemented them with contig connectivity information derived from Bionano optical maps (Stolle *et al.*, 2019) and from RNA sequencing of multiple tissue types and developmental stages (Calkins *et al.*, 2018) all SRA runs from BioProjects PRJNA542606

(Martinez-Ruiz *et al.*, 2020), PRJNA422376 (Calkins *et al.*, 2018), PRJNA266847, and PRJNA393960. These were then input to ALLMAPS (version 0.8.12 (Tang *et al.*, 2015)) to order and orient the contigs; all datasets were equally weighted to reduce propagating biases of any one dataset.

To create genetic maps, we first demultiplexed the RAD-seq reads using a custom script and cleaned the demultiplexed reads using default parameters of stacks2 (version 2.5 (Rochette, Rivera-Colón and Catchen, 2019)). Second, for each family, we mapped the cleaned RAD-seq reads to the assembly using bwa-mem (version 0.7.17 (Li, 2013)) and genotyped the individuals using stacks2 (-X "populations: -e ecoRI --vcf"). The VCF output of stacks contained only bi-allelic sites. Next, for each family, we plotted the number of called sites for each individual on x-axis and the corresponding number of homozygous sites on the y-axis (Figure 2.S9). Because the individuals are haploid, we expect an almost 1:1 correlation between the number of called sites and the number of homozygous sites. Based on the plot, we eliminated individuals that were 2 standard deviations away from the regression line. We additionally removed individuals that jumped out as having too few called sites. Next, we filtered variant sites based on the number of missing observations (because the individuals are haploid males, we treated heterozygous calls as missing observation), mean site depth, mean genotype quality, and minor allele frequency. The respective thresholds were chosen by inspecting a frequency histogram of each parameter for each family and testing several values (Figure 2.S10-S13). We found a suitable threshold for the number of missing observations to be around 25-30% of the number of individuals in the family, for mean site depth to be around 99th percentile, for mean genotype quality to be around 10th percentile, and for minor allele frequency to be either around 0.38 or 0.10. Next, we phased the filtered genotypes using a haplotype doubling method (Wang *et al.*, 2013) and converted the phased and filtered genotypes matrix to a format suitable for MSTmap (downloaded on December 17, 2019 (Wu *et al.*, 2008)). For MSTmap, we used the distance_function kosambi and population_type DH for all the families and family specific values for the parameters cutoff_p_value and missing_threshold, ranging from 10^{-6} to 10^{-10} for cutoff_p_value and either 0.25 or 0.30 for missing_threshold. The variant sites clustered into expected 16 linkage groups for six out of the seven families. However, one family had very few markers: only 389 while the other families had between 5,000 and 17,000 markers. We discarded it. Linkage

groups from the five remaining families were then converted to ALLMAPS compatible format. Scripts used for linkage map creation and conversion to ALMMAPS format, including from the steps below are available at the following GitHub repository: github.com/wurmlab/to_allmaps.

For Bionano optical maps, we first scaffolded the assembly using hybrid scaffolding option of IrysView software using the aggressive preset (version 2.5.1). Next, we used bionano2Allmaps.pl script (github.com/tanghaibao/jcvi/issues/37) to convert contig connectivity information from IrysView's output to ALLMAPS compatible format. We then eliminated paths with less than four markers.

For RNA-seq data, we mapped them to our assembly using bwa-mem (-M; version 0.7.17 (Li, 2013)) and eliminated reads that mapped to more than one location in the genome (bioinformatics.stackexchange.com). Next, we generated *ab initio* gene predictions using AUGUSTUS (--gff3=on --species=fly; version 3.2.3 (Stanke *et al.*, 2008)). Next, we used AGOUTI (version 0.3.3-25-ga7e65d6 (Zhang, Zhuo and Hahn, 2016)) to generate contig connectivity information from read mappings and *ab initio* gene predictions. Finally, we used a custom script to convert AGOUTI's output to ALLMAPS compatible format.

Data availability

The Pacbio data that were used to generate the 36 assemblies as well as the scaffolded best assembly are available from NCBI (BioProject PRJNA609320).

The code written for this project is split into two repositories. First, a tool to compare genome assemblies: github.com/wurmlab/CompareMyGenome. Second, a set of scripts to create linkage maps, and to convert linkage maps and contig connectivity information from Bionano and RNA-seq data to ALLMAPS compatible format: github.com/wurmlab/to_allmaps.

Acknowledgements

We thank Maria Cristina Arias (Universidade de São Paulo, Brazil) for providing the samples as permit holder; Andrew Leitch, Richard Nichols, Stephen Rossiter, Richard Durbin, Mark Blaxter, James Borrell, and Marian Priebe for valuable discussions that has shaped the work; Simon Butcher, Chris Walker, Peter Childs, Dan Whitehouse, and Tom Bradford for their help in utilising QMUL HPC system; Emeline Favreau for comments on the manuscript; Philip Howard and Martin Tran for Bionano hybrid scaffolding; and USDA Agricultural Research Service for additional Pacbio data.

This research was possible thanks to the funding available to the authors from Biotechnology and Biological Sciences Research Council (BB/K004204/1 to YW), Natural Environment Research Council (NE/L00626X/1 and NERC EOS Cloud to YW, and NE/S007229/1 to AW), Deutscher Akademischer Austauschdienst (DAAD) Postdoc Program (570704 83 to ES), European Commission Marie Curie Actions (PIEF-GA-2013–623713 to ES and YW), Marie Sklodowska-Curie Individual Fellowship (H2020-MSCA-IF-2018-842592) to AB and YW; and QMUL's Apocrita MidPlus computational facilities (doi.org/10.5281/zenodo.438045) and the JASMIN super-data cluster (www.jasmin.ac.uk).

Supplementary methods

CompareMyGenomes tool usage

```
docker run wurmlab/comparemygenomes --help

CompareMyGenomes 1.0
Calculate measures of contiguity, completeness, accuracy and rank the
assemblies.

comparemygenomes --genome-size 450000000 --busco-lineage hymenoptera
directory_containing_assemblies_and_illumina_reads/

-g, --genome-size       Expected genome size in base pairs.
-b, --busco-lineage     One of the 44 BUSCOv3 lineages, e.g., mamalia,
                        insecta, nematoda and so on. Required, unless
                        --rank-only is specified.
--rank-only             Don't compute metrics. Only rank assemblies based
                        on given tabular files.
--help                  View this message
```

Comparison of Canu, flye, and wtdbg2 assemblers

Assemblies generated by three popular long-read genome assembly software: Canu (Koren *et al.*, 2017), flye (Kolmogorov *et al.*, 2019), and wtdbg2 (Ruan and Li, 2020) were compared using our CompareMyGenomes tool (Table 2.S5). Assemblies were generated using default parameters. In case of Canu, we additionally used the purge_haplotigs pipeline (Roach, Schmidt and Borneman, 2018) to remove unresolved haplotigs which is typical of Canu assemblies. None of the assemblies were polished.

wtdbg2 generated the most contiguous assembly, but the assembly generated by Canu had the most resolved regions (13 Mb more than the next best) and considerably higher

proportion of solidly mapped Illumina reads (57.62% compare to 55.25% of the runner up), followed by Flye. There is a 0.01% difference in the BUSCO score (Simão *et al.*, 2015) of Canu and Flye assemblies. However, this difference is minor and likely to be eliminated by any subsequent polishing steps.

Quality control of Illumina reads

We filtered and trimmed Illumina datasets prior to use. First, we removed optical duplicates using clumpify.sh (version 37; biostars.org). Second, we removed reads with mean quality threshold lower than 15 using htqc (Yang *et al.*, 2013). Third, we compared the mean base quality per cycle, per tile to the mean base quality of that cycle across all tiles to test for air-bubbles becoming trapped in the flow cell (sequencing.qcfail.com). For this, we obtained the difference between per-cycle mean base quality for a tile and the per-cycle mean base quality for all tiles from FastQC's text output (version 0.11.5; bioinformatics.babraham.ac.uk). Where this difference was greater than 4, we changed the corresponding base in the reads to 'N'. This was done by creating a BED file of positions from the tile and cycle information and then using seqtk (version 1.2; github.com/lh3/seqtk) to convert bases at the positions specified in the file. Next, we considered that reads with multiple occurrences of low-quality bases may be problematic. To eliminate such reads, we turned bases with quality scores lower than 12 to 'N' using seqtk (reads with excessive Ns are removed in the next step). Finally, we used cutadapt (version 1.13 (Martin, 2011)) to trim adapter sequences, to trim low-quality bases from 3' and 5' ends, to trim any leading and trailing 'N's, to eliminate after trimming reads shorter than 50 bp and those with more than 4 'N's. For the Illumina sequences used for assembly comparison, we retained 64,850,542 pairs of 50-150 bp reads (*i.e.*, 79.23% of sequenced bases) after filtering.

Supplementary figures

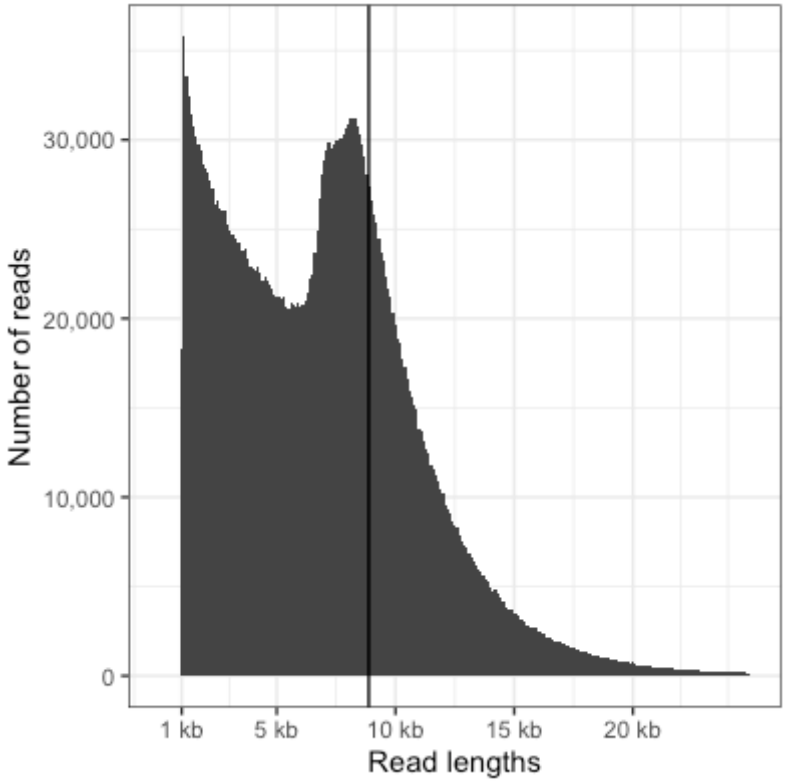


Figure 2.S1: Length distribution of raw Pacbio reads

Read lengths on x axis vs count of read lengths on y axis. The black vertical line is the N50 read length (8,876 bp).

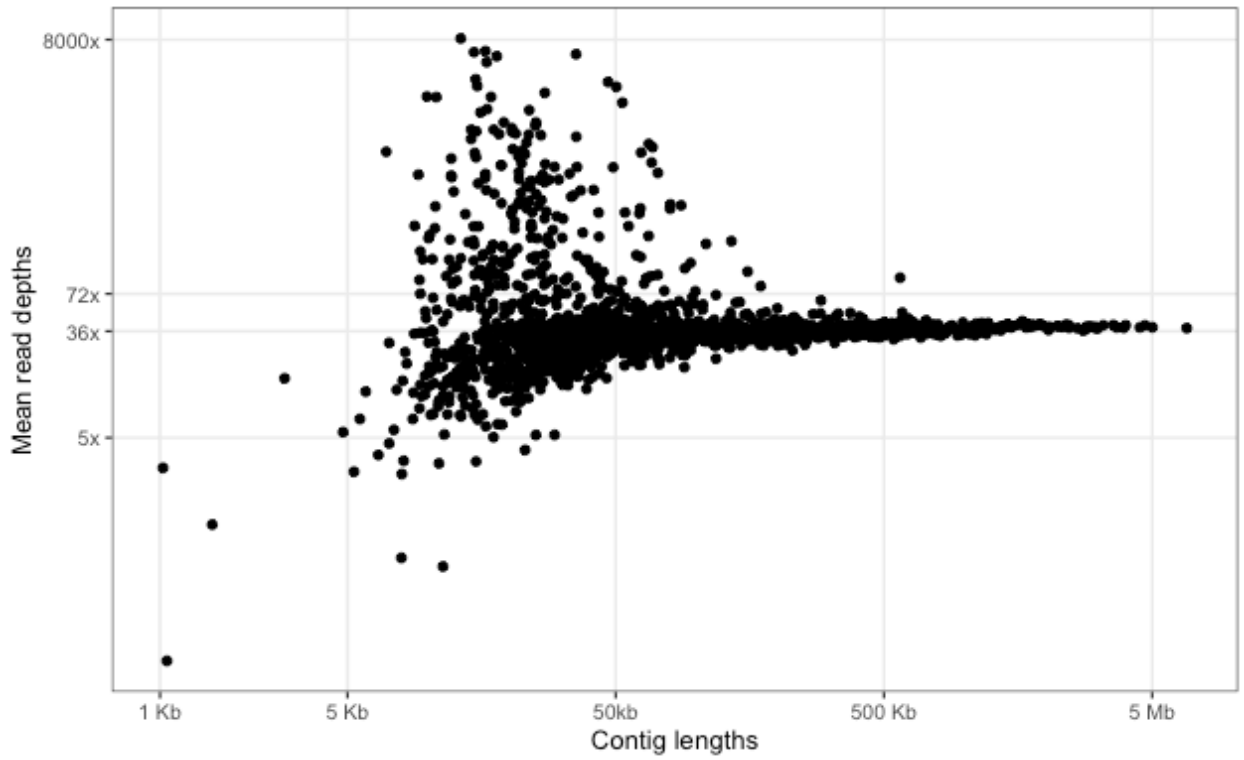


Figure 2.S2: Contig length vs average coverage

Distribution of contig lengths (x axis) and mean read depths (y axis) of the best assembly. The axes are log scaled. Reads were mapped using default parameters of minimap2 (version 2.17 (Li, 2018)). Average read depth of contigs were calculated using mosdepth (version 0.2.6 (Pedersen and Quinlan, 2018)). Some contigs have average depth as high as 8000x. Contigs with average depth higher than twice the median coverage (36x) are likely to contain collapsed representation of larger regions of the genome. Contigs with average depth lower than 5x are likely to contain higher amounts of sequencing error. This is because the SMRTLinks polishing step, which is critical for long-read genome assemblies, excludes regions with coverage lower than 5x.

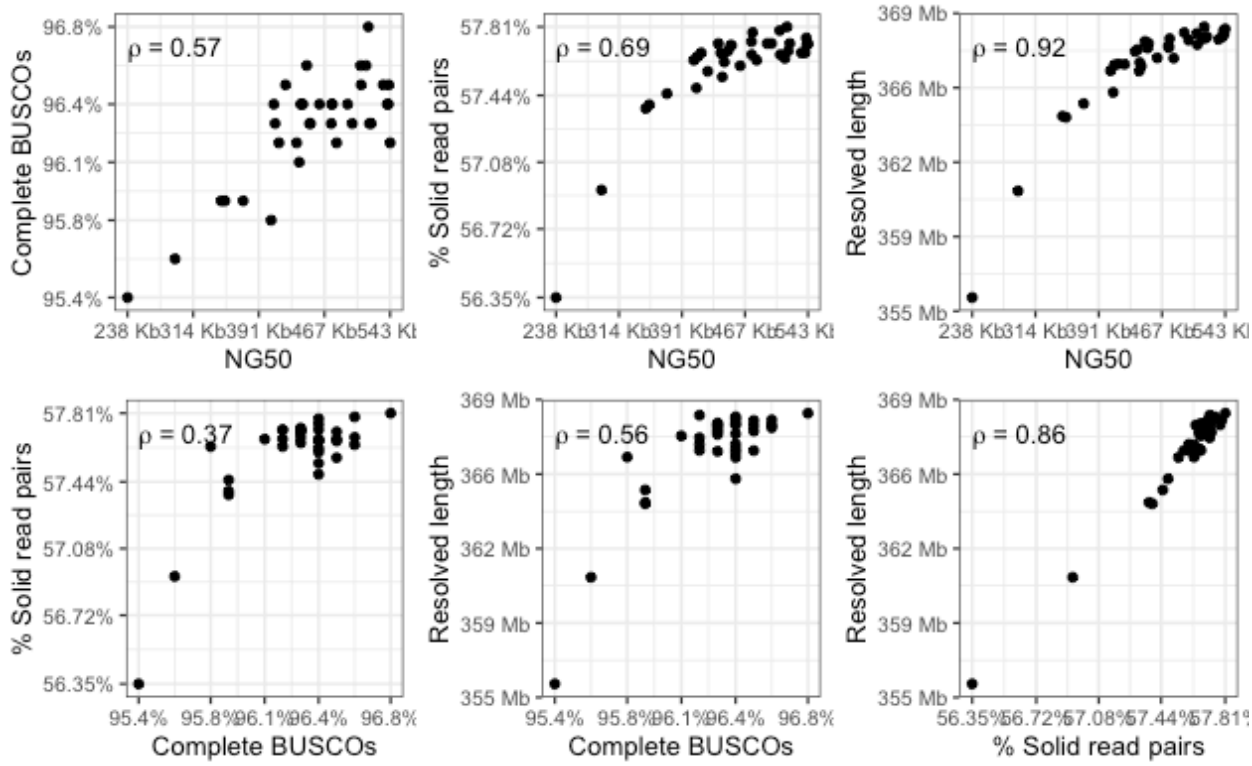


Figure 2.S3: Correlation between the metrics

Each panel shows the values taken by a pair of metrics on the x and the y axes, and Spearman's rank correlation coefficient (ρ) between the metrics.

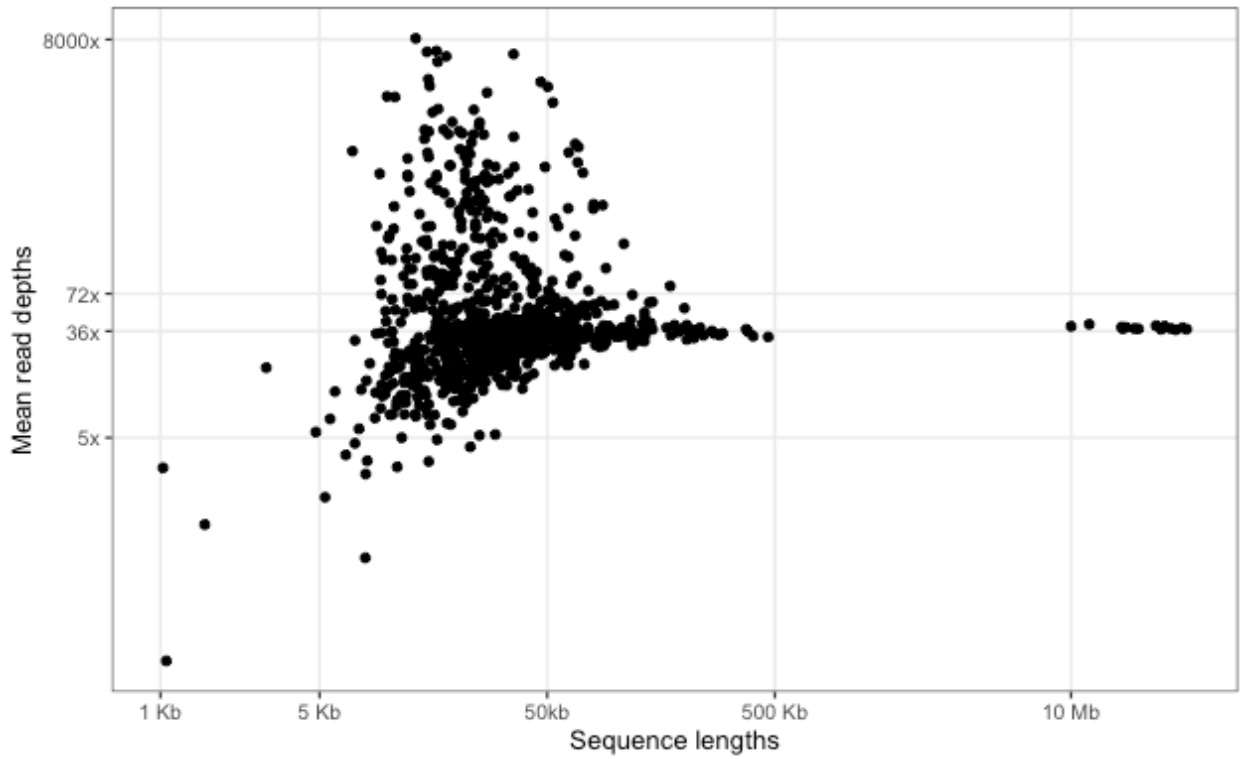


Figure 2.S4: Length vs coverage after scaffolding

Distribution of sequence lengths (x axis) and mean read depths (y axis) of the best assembly, like Figure 2.S2, but after scaffolding. Sequences longer than 10 Mb are the chromosomes, while the cloud of sequences on the left are unplaced contigs.

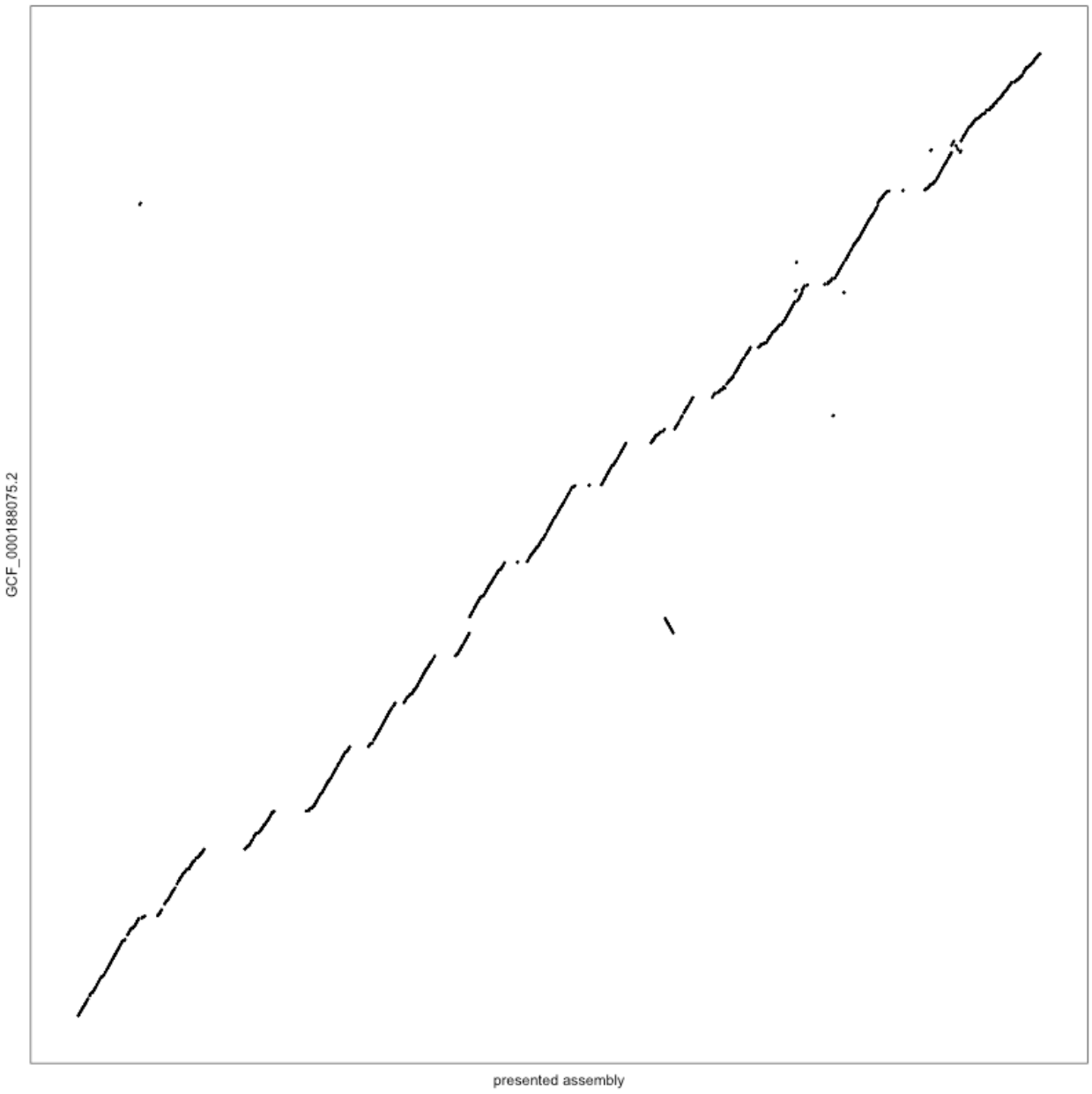


Figure 2.S5: Dot-plot of the presented assembly and the draft assembly

Dot-plot of the presented assembly (x axis) against the draft fire ant genome (y axis). The assemblies were aligned using minimap2 (version 2.17; -c -P -k19 -w19 -m200 (Li, 2018)) and visualised using dotPlotly (github.com/tpoorten/dotPlotly; -m 100000). Most breaks in collinearity are along the x axis. These show inclusion of new sequences in the presented assembly.

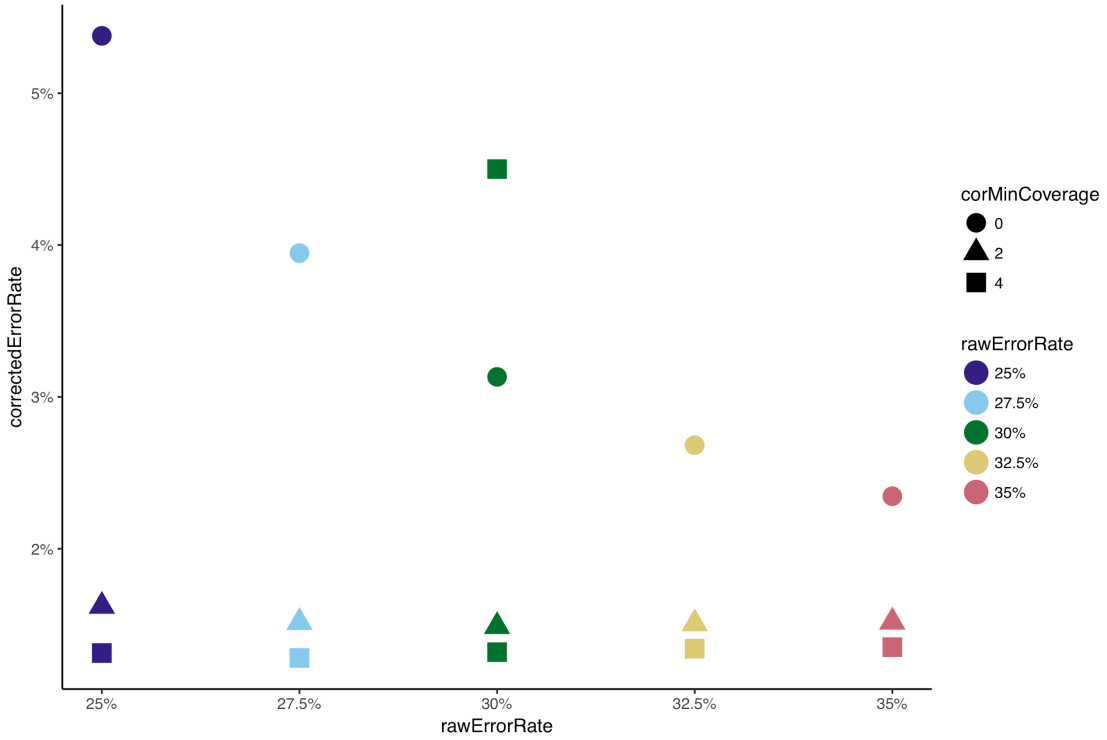


Figure 2.S6: Estimated error rate of corrected reads

The estimated value of the parameter correctedErrorRate (y axis) against rawErrorRate (x axis) and trimming stringency (shape).

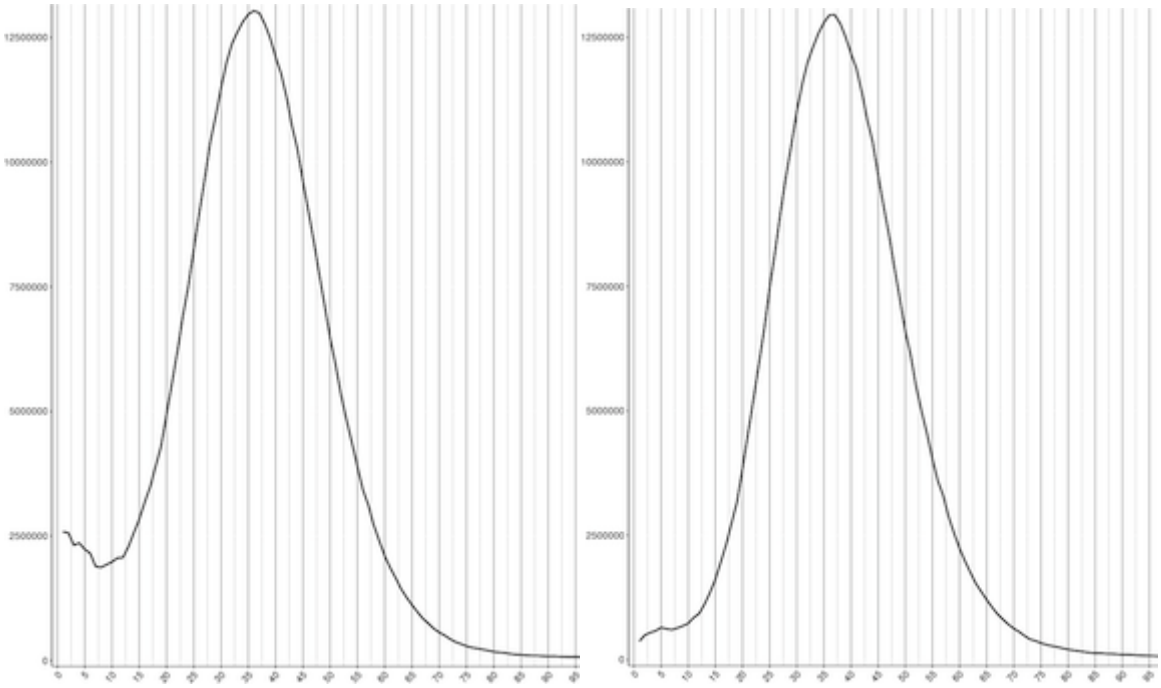
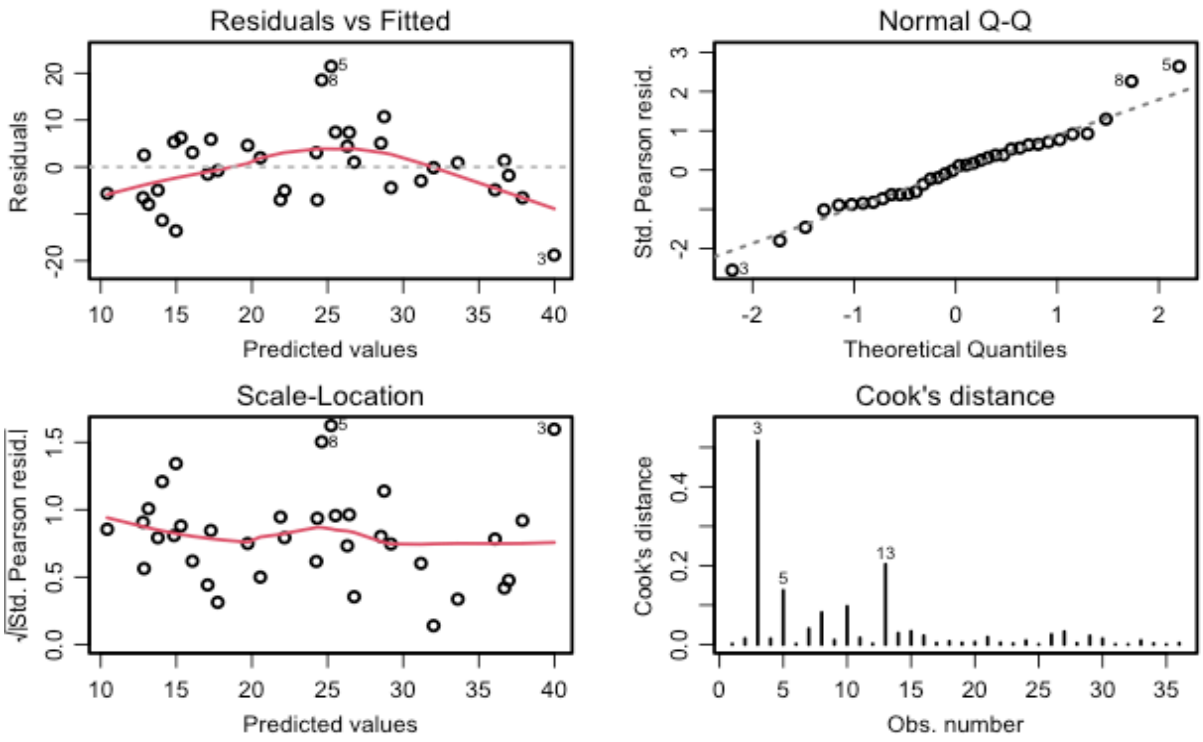


Figure 2.S7: Coverage histogram before and after removing unresolved haplotigs

Coverage histogram of the best assembly before removing unresolved haplotigs (left) and after. There is a clear enrichment of bases under 18x coverage (half the median coverage) which is largely resolved after using purge_haplotigs.




```
Call:
glm(formula = rank ~ raw + cov + cor, data = assemblies)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-18.7776   -5.2063    0.3749    4.7336   21.4776

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -27.9000    14.2227  -1.962   0.0586 .
raw           0.9740     0.4137   2.354   0.0249 *
cov          -0.6101     0.8473  -0.720   0.4767
cor           3.8188     0.6958   5.488  4.8e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 71.47563)

    Null deviance: 4734.3  on 35  degrees of freedom
Residual deviance: 2287.2  on 32  degrees of freedom
AIC: 261.62

Number of Fisher Scoring iterations: 2
```

Figure 2.S8: R general linear model output

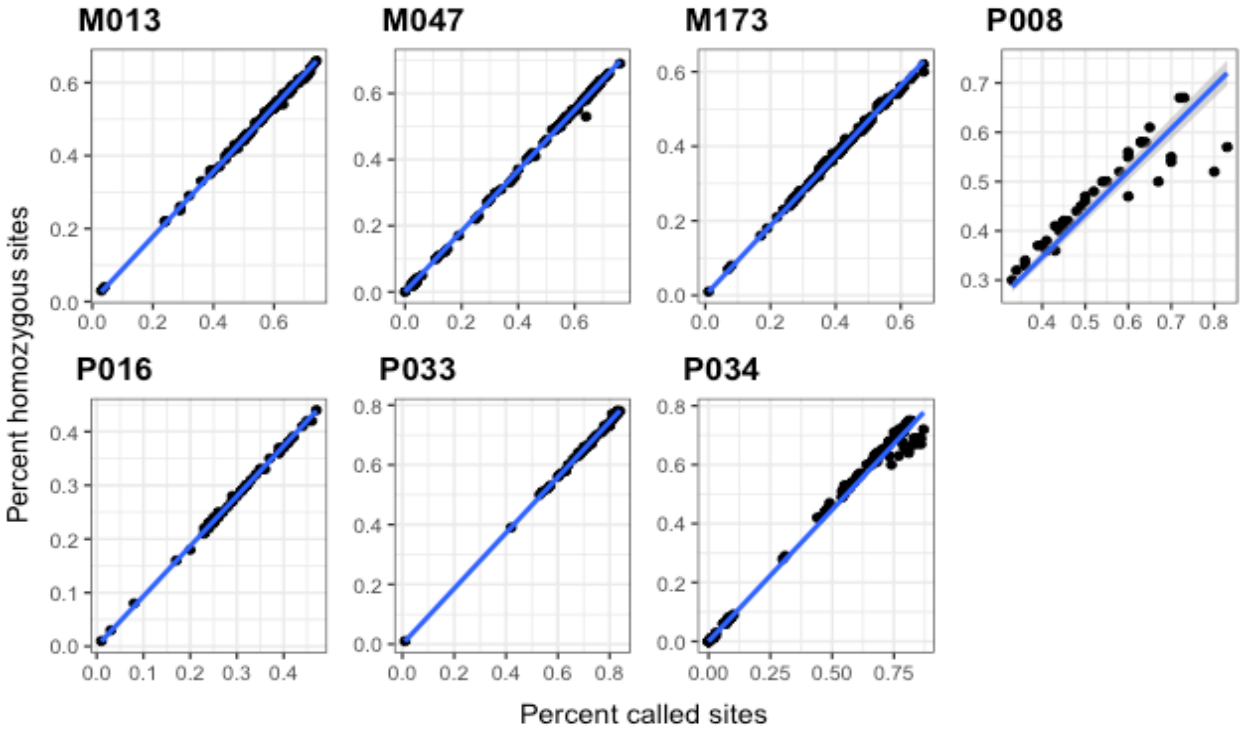


Figure 2.S9: Proportion of genotype calls against proportion of homozygous calls

Proportion of individuals genotypes per site (x axis) against proportion of homozygous individuals per site (y axis).

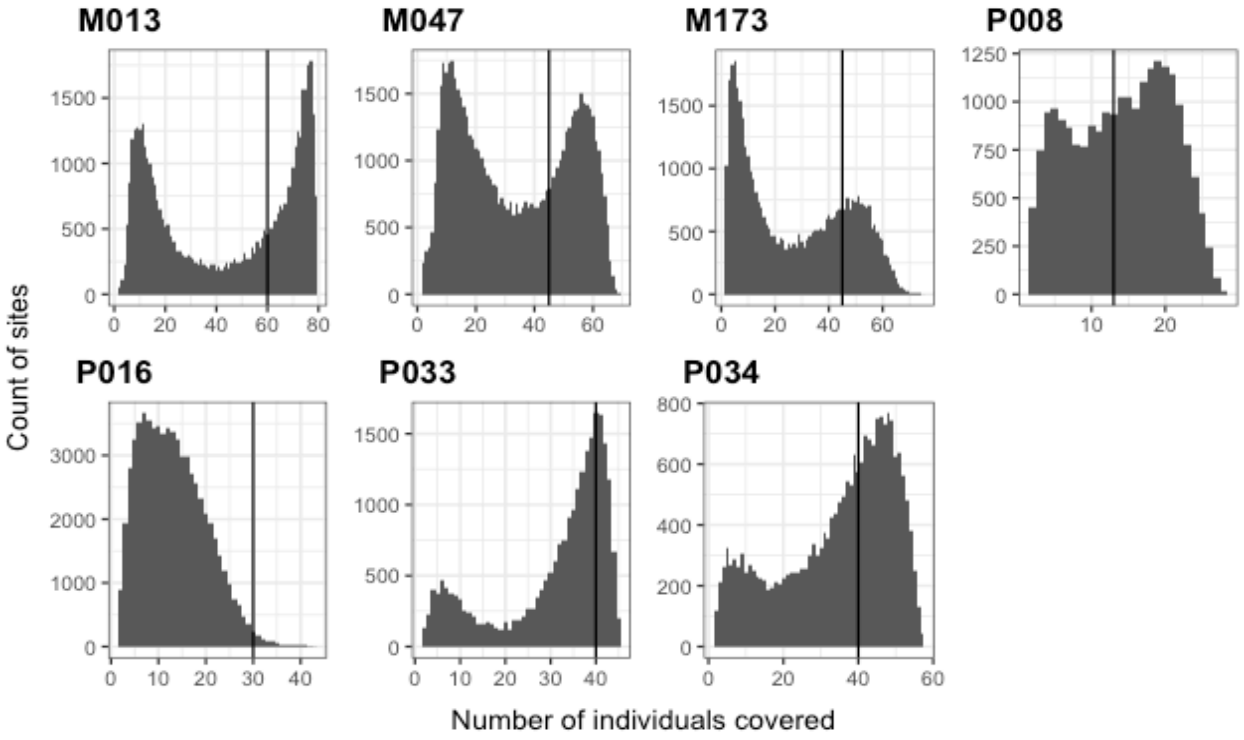


Figure 2.S10: Histogram of number of genotype individuals at each site

Number of individuals genotyped per site (x axis) against their count (y axis), for each of the seven families (M013 - P034). Black vertical line shows the threshold chosen for each family.

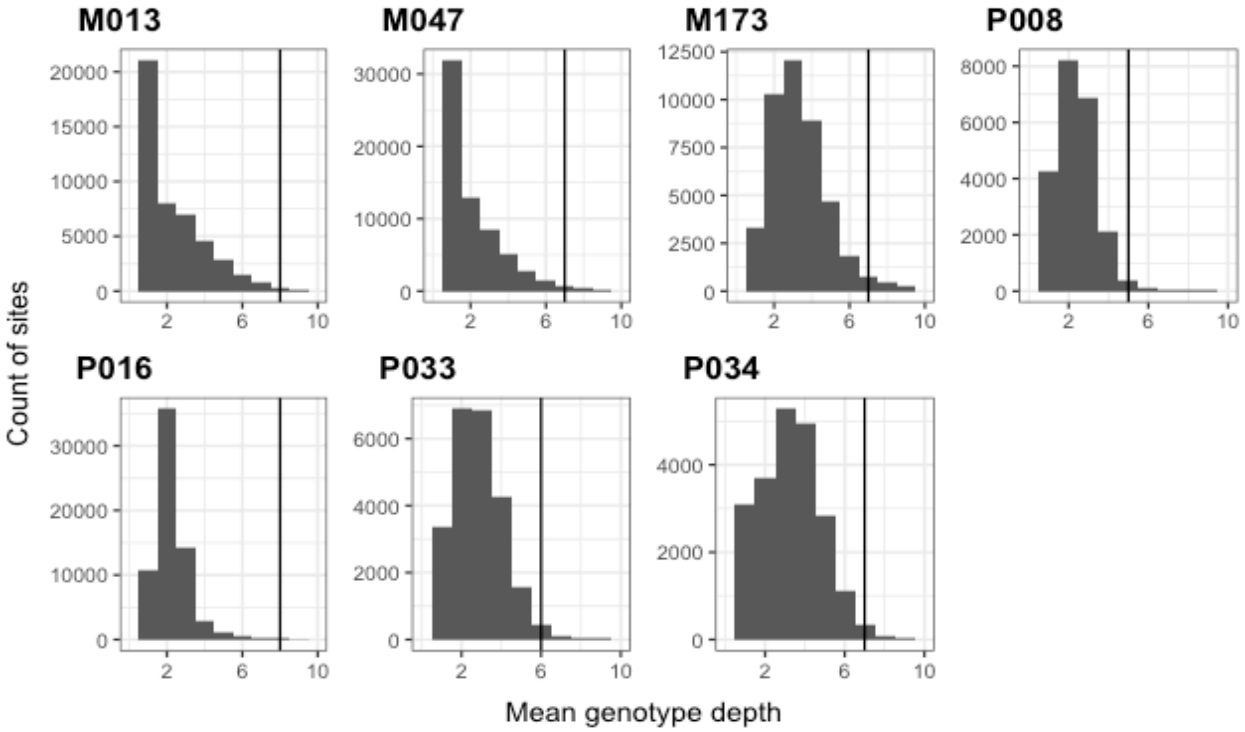


Figure 2.S11: Histogram of mean genotype read depth of sites

Mean read depth of genotypes per site (x axis) against their count (y axis), for each of the seven families (M013 - P034). Black vertical line shows the threshold chosen for each family.

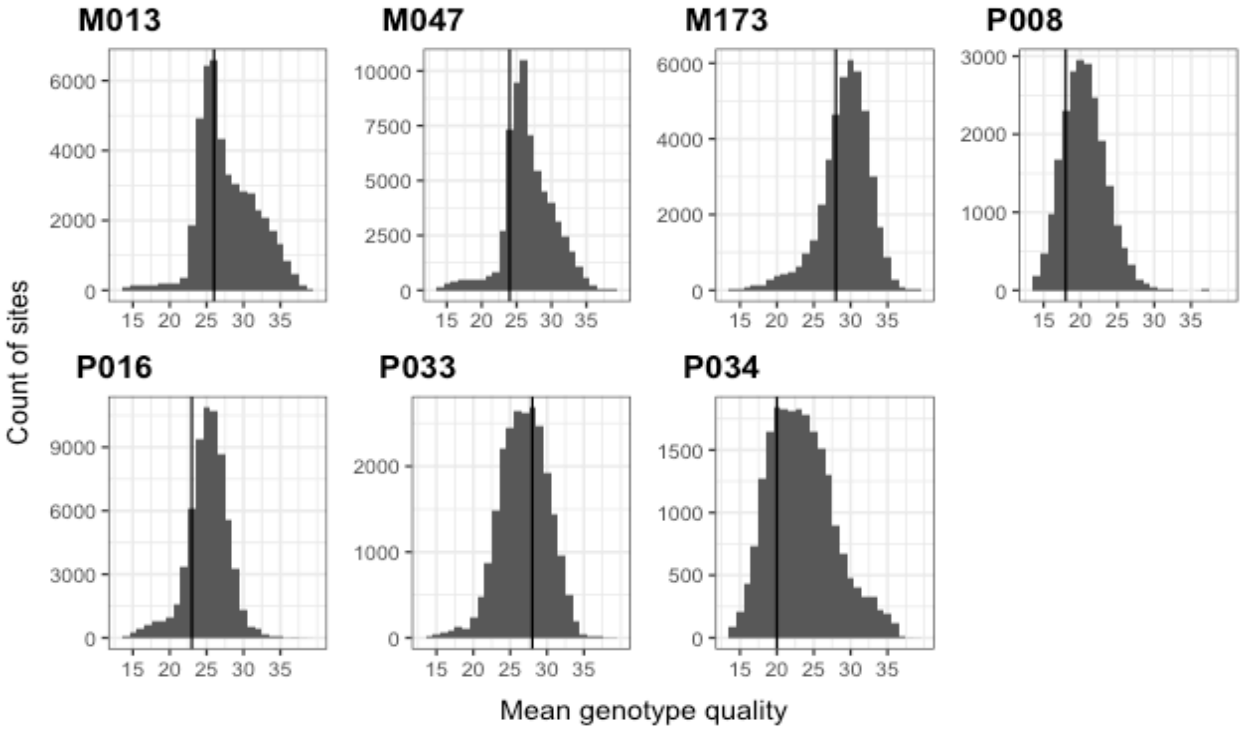


Figure 2.S12: Histogram of mean genotype quality of sites

Mean genotype quality per site (x axis) against their count (y axis), for each of the seven families (M013 - P034). Black vertical line shows the threshold chosen for each family.

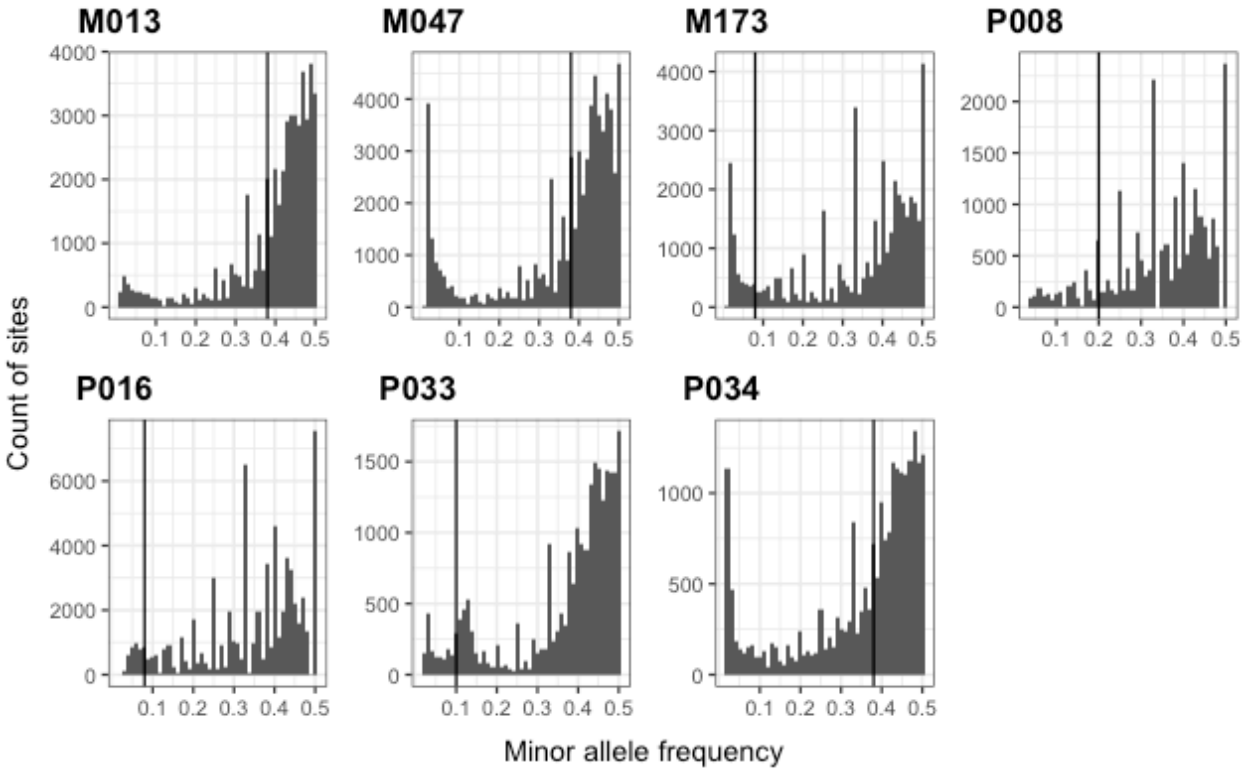


Figure 2.S13: Minor allele frequency spectrum of sites

Minor allele frequency per site (x axis) against their count (y axis), for each of the seven families (M013 - P034). Black vertical line shows the threshold chosen for each family.

Supplementary tables

Table 2.S1: Assembly parameters tested

We tested the effect of varying overlap error thresholds for raw and corrected reads, and stringency of trimming of raw reads (first three columns). Two other parameters were changed from default values when generating these assemblies, the sensitivity of overlap detection and the proportion of reads to use for correction (last two columns).

rawErrorRate Overlap error threshold for raw reads (twice the estimated sequencing error rate)	**corMinCoverage** Stringency of trimming raw reads	**correctedErrorRate** Overlap error threshold for corrected reads (twice the estimated error rate of corrected reads)	**corOutCoverage** What proportion of input reads to correct	**corMhapSensitivity** Sensitivity level of finding overlaps between raw reads
0.300	4	0.045	40x	normal
0.250	0	0.05378	100x	high
0.250	0	0.08378	100x	high
0.250	0	0.11378	100x	high
0.275	0	0.03947	100x	high
0.275	0	0.06947	100x	high
0.275	0	0.09947	100x	high
0.300	0	0.03132	100x	high
0.300	0	0.06132	100x	high
0.300	0	0.09132	100x	high
0.325	0	0.02682	100x	high
0.325	0	0.05682	100x	high
0.350	0	0.02346	100x	high
0.350	0	0.05346	100x	high
0.250	2	0.04622	100x	high
0.250	2	0.07622	100x	high
0.275	2	0.04515	100x	high
0.275	2	0.07515	100x	high
0.300	2	0.04489	100x	high
0.325	2	0.04507	100x	high
0.350	2	0.04518	100x	high
0.250	4	0.04313	100x	high
0.250	4	0.07313	100x	high
0.275	4	0.04281	100x	high

0.275	4	0.07281	100x	high
0.300	4	0.04319	100x	high
0.300	4	0.07319	100x	high
0.325	4	0.04342	100x	high
0.325	4	0.07342	100x	high
0.350	4	0.04352	100x	high
0.350	0	0.08346	100x	high
0.300	2	0.07489	100x	high
0.350	2	0.07518	100x	high
0.350	4	0.07352	100x	high
0.325	2	0.07507	100x	high
0.325	0	0.08682	100x	high

Table 2.S2: Improvements made by polishing and haplotigs removal

Improvements made by each round of polishing and haplotype filtering.

	Raw contigs	**Pacbio polished contigs**	**Illumina polished contigs**	**Haplotype-filtered**
General error rate	1.34	1.30	1.26	-
%reads with insertion	6.48%	3.91%	1.79%	-
%reads with deletion	2.75%	2.82%	2.19%	-
Mean mapping quality	25.73	27.49	28.24	-
BUSCO	D:3.4%,F:1.3%,M:0.7%	D:2.2%,F:0.8%,M:0.4%	D:2.3%,F:0.5%,M:0.5%	D:0.7%,F:0.6%,M:0.6%

Table 2.S3: Contaminant species identified in the best assembly

Kingdom	**Species**
Bacteria	*Bordetella bronchialis* (taxid 463025)
Bacteria	*Streptococcus respiraculi* (taxid 2021971)
Bacteria	*Pseudomonas silesiensis* (taxid 1853130)
Bacteria	*Streptomyces albus* (taxid 1888)
Bacteria	*Stappia sp.* ES.058 (taxid 1881061)
Bacteria	*Streptomyces griseoviridis* (taxid 45398)
Bacteria	Agrobacterium (taxid 357)
Bacteria	Terrabacteria group (taxid 1783272)
Bacteria	*Euhalothece natronophila* Z-M001 (taxid 522448)

Bacteria	*Streptomyces sp.* CC0208 (taxid 2306165)
Bacteria	*Pontibacter korlensis* (taxid 400092)
Bacteria	*Cardiobacterium hominis* (taxid 2718)
Bacteria	*Legionella anisa* (taxid 28082)
Bacteria	*Streptomyces griseoviridis* (taxid 45398)
Bacteria	*Bradyrhizobium symbiodeficiens* (taxid 1404367)
Fungi	*Pichia membranifaciens* NRRL Y-2026 (taxid 763406)
Fungi	Aspergillus (taxid 5052)
Plants	*Ipomoea trifida* (taxid 35884)
Plants	*Vigna unguiculata* (taxid 3917)
Plants	*Gossypioides kirkii* (taxid 47615)
Plants	*Populus trichocarpa* (taxid 3694)
Plants	*Gossypium raimondii* (taxid 29730)
Plants	*Brassica oleracea* (taxid 3712)
Plants	Viridiplantae (taxid 33090)
Plants	*Arachis hypogaea* (taxid 3818)
Plants	*Brassica rapa* (taxid 3711)
Plants	*Solanum pinnatisectum* (taxid 50273)
Plants	*Gossypioides kirkii* (taxid 47615)
Plants	*Vigna unguiculata* (taxid 3917)
Plants	*Sesamum indicum* (taxid 4182)
Plants	rosids (taxid 71275)
Plants	Mesangiospermae (taxid 1437183)
Plants	*Hordeum vulgare* subsp. vulgare (taxid 112509)
Plants	*Cannabis sativa* (taxid 3483)

Table 2.S4: Comparison of the published fire ant genome assemblies with the presented assembly

	Unique resolved length	**NG50**	**Complete single-copy BUSCO genes**	**Duplicated single-copy BUSCO genes**	**% Solid read pairs**	**Additional problematic sequences**
presented assembly	369 Mb	19.8 Mb	97.5%	0.6%	58.02%	16Mb unresolved
(Yan *et al.*, 2020)	365 Mb	12.6 Mb	97.0%	1.7%	57.76%	25Mb unresolved 17Mb duplicated haplotypes

(Fontana *et al.*, 2019)	328 Mb	7.14 Mb	88.4%	0.3%	51.27%	21 Mb unresolved
(Wurm *et al.*, 2011)	333 Mb	0.44 Mb	96.2%	0.3%	51.08%	66 Mb unresolved

Table 2.S5: Comparison of three popular assemblers using the presented CompareMyGenomes tool

Three popular assemblers compared using our CompareMyGenomes tool. Assemblies were generated using default parameters. In case of Canu, we additionally used the purge_haplotigs pipeline to remove unresolved haplotigs, which is typical of Canu assemblies. None of the assemblies were polished. Although wtdbg2 generated the most contiguous assembly, the assembly generated by Canu has the most resolved regions and considerably higher proportion of solidly mapped Illumina reads. The 0.01% difference in the BUSCO score of Canu and Flye assemblies is minor, and likely to be eliminated by polishing.

	Resolved length	**NG50**	**Completed BUSCO**	**Solid pairs**
Canu+ purge_haplotigs	366,814,754 bp	441,945 bp	96.4%	57.62%
Flye	353,678,069 bp	402,671 bp	96.5%	55.25%
Wtdbg2	320,860,502 bp	502,081 bp	68.6%	48.12%

Chapter 3: Rapid transfer of annotation to *de novo* genome assemblies

Contributions

Yannick Wurm identified the conceptual need for the software. I researched the specifics of chain file creation, limitations of UCSC liftOver, implemented the software, and wrote the chapter. Rodrigo Pracana provided very helpful feedback on an initial draft of the chapter.

The chapter is intended for submission as a brief communication to Bioinformatics:

A Priyam, R Pracana, Y Wurm (in prep)

Introduction

Reference genomes are subject to change. This creates the need to transfer coordinates of genes, single nucleotide polymorphisms (SNPs) and other annotations (a point coordinate or a coordinate interval) to a new genome assembly. For model organisms, this is commonly done using the programs liftOver (Kuhn, Haussler and Kent, 2013) or CrossMap (Zhao *et al.*, 2014) and a "chain file" (Kent *et al.*, 2003) downloaded from UCSC Genome Browser. Chain files describe regions of similarity between the two genome assemblies. Using chain files, the tools liftOver and CrossMap can transfer annotations stored in a variety of file formats. Alternatively, researchers may use NCBI's Remap web service (ncbi.nlm.nih.gov). However, both the services are limited in terms of the species and the genome builds they can transfer the coordinates between. This is a major disadvantage for researchers working with other organisms or custom genome builds. Furthermore, the tools liftOver and CrossMap do not correctly process hierarchical or connected features in Generic Feature Format (GFF) files, a popular file format for gene annotations. This can result in non-biologically meaningful output such as coding sequence annotation without a parent transcript annotation, or the coding sequence of a transcript split across two sequences.

We present flo, a command line tool to generate chain files suitable for coordinate transfer between genome assemblies of the same or very closely related species, and to transfer genes and other annotations in Generic Feature Format (GFF) files. The generated chain file can in turn be used with liftOver and CrossMap to transfer annotations in BED, BAM, VCF and other file formats supported by the two.

Methods

Chain file generation

Flo creates chain file following the process described in Kent 2003. Briefly, the two genome assemblies are first aligned to identify short segments of high sequence similarity. These

segments are then 'chained' to form longer alignments using a scoring scheme that allows for large gaps that can span local insertion or inversion events. Next, the gaps in the longest chain at each locus are filled by progressively stacking shorter chains and trimming the parts of the shorter chain that overlap in a process known as 'netting'. The resulting chains cover each locus of the genome exactly once and can be used for annotation lift over.

We use BLAT (Kent, 2002) to align the two genome assemblies. The alignments are chained and netted using the programs axtChain, chainSort, and chainNet (Kent 2003). The chains retained after the netting process are extracted for lift over using netChainSubset. To speed up the alignment and the chaining process, we split the target genome assembly into a user defined number of files and process them in parallel using GNU Parallel (Tange, 2011). To further speed up the alignment process, we split the query sequences (target assembly) into 5000 bp chunks and use -fastMap option of BLAT. Finally, only the alignments with identity 95% or higher are used by default.

Lift over of annotations in GFF format files

Feature annotations in the widely used General Feature Format (GFF) file are defined as a hierarchy of genomic intervals, each with its associated sequence ontology term and other meta data. For example, a gene is defined as an interval containing one or more messenger RNA intervals, which are in turn composed of intervals defining exons and parts of coding sequences. To capture this hierarchical relationship, we sort the annotations in a GFF file by their start coordinates and length and read them into a tree data structure that preserves the order of child nodes. The annotations are sorted so that parent intervals are read before child intervals. Next, we map the intervals represented by leaf nodes to the new assembly using liftOver (-minMatch=0.5 (Kuhn, Haussler and Kent, 2013)) and the chain file generated above. The node is considered mapped if 50% or more of the interval represented by the node mapped to the new assembly. We then consider the parents of the leaf nodes: If one or more child of a parent node mapped to the same sequence, strand, and in the same order on the target assembly, we mark the parent node as mapped and update its sequence identifier, strand, and start and end coordinates based on the child nodes that mapped. Otherwise, the node and all its children are recursively marked as unmapped. We repeat this process with

the parents of the parent nodes, their parents, and so on, till all the nodes have been processed. Finally, we write the nodes that mapped and those that did not to two output GFF files.

Quality control of transferred gene annotations

Flo reports both full-length and partial mappings of annotations in a GFF file. However, partial matches of annotations may not always be biologically meaningful. For example, a partially mapped gene annotation may be frameshifted or yield a chimeric protein product in other ways. Even full-length match of a gene may contain a nonsense mutation resulting in truncated protein product, or the gene may have mapped to a paralog. Thus, as a final step we compare the translated amino-acid sequence of gene annotations before and after mapping and report the difference in their length and the Levenshtein distance between the sequences. Researchers can use this table output to obtain a subset of gene annotations with the desired level of confidence: mappings that yield exactly identical protein-product, or up to those with one mismatch, and so on. For example, partial matches can serve as a useful bait in RNA sequencing studies, while comparative analyses may prefer identical mappings. Furthermore, this arrangement allows future investigation of why some gene models do not yield exactly identical sequences and can hint at errors in the target assembly or annotation errors in the source assembly. Although, some differences will necessarily be a result of differences between the samples from which the assemblies are derived.

Results

We tested flo on genome assemblies of the red fire ant (Wurm *et al.*, 2011) and with annotations downloaded from NCBI. First, annotation transfer between the draft assembly (Wurm *et al.*, 2011) and itself resulted in 99.9% gene models to be mapped with exactly identical protein sequence, as it should. Next, we transferred annotations from the draft assembly to the genome assembly generated in the previous chapter. A match was reported for 96.5% of all intervals and all transcripts. 88.7% of all transcripts had no frameshift or

missense mutations. 64.4% of all transcripts yielded identical protein product, 13.6% had up to one mismatch, 7.8% had up to five mismatches, and 3% of the transcripts had more than five mismatches. 4.9% of all transcripts mapped with a truncated protein product. Interestingly, 2.9% of the transcripts had longer protein product after mapping, suggestive of a frameshift error in the source annotation. The remaining 3.5% of the transcripts either did not map at all, i.e., were deleted in the new assembly, or mapped inconsistently, i.e., their child features mapped to different sequences, strand, or were not in the same order as input.

The entire process ran in less than 200 CPU hours for the moderately sized 450 Mb fire ant genome. Whole-genome alignment was the most time-consuming step. Increasing BLAT's minimum identity threshold from 95% to 98% reduced the runtime by 5-fold with a modest loss of about hundred features.

Discussion

We present flo, a command line tool to generate chain files and lift over annotations in GFF format to a new assembly. The source and the target genome assemblies, the number of parallel processes to run, and the GFF files containing annotations on the source assembly are provided through a configuration file. Flo transfers all annotations in GFF files including protein-coding genes, non-coding RNA, transfer RNA and others. The software provides a detailed breakdown of annotations that couldn't be lifted and why. For protein-coding genes, flo additionally generates a table describing similarity between the input and output protein product for quality filtering. If no GFF files are given, flo only generates the chain file. Users can optionally specify BLAT parameters to use for chain file generation and use the included helper script to download the required dependencies.

We show that flo achieves high sensitivity on the fire ant genome and runs relatively fast. In our experience, flo runs faster and achieves higher sensitivity compared to approaches that perform spliced alignment of a gene's amino-acid sequence to the target genome such as exonerate (Slater and Birney, 2005) or spaln (Gotoh, 2008). Although we did not specifically test inter-species annotation transfer, we expect the approach will work for closely related

species that have a high level of similarity between the genomes. Finally, how flo compares to approaches such as liftOff (Shumate and Salzberg, 2020) remain to be seen.

Data availability

flo is available at https://github.com/wurmlab/flo. Fire ant annotations transferred to the new assembly are available on request.

Acknowledgements

This research was possible thanks to the funding available to the authors from Biotechnology and Biological Sciences Research Council (BB/K004204/1 to YW), Natural Environment Research Council (NE/L00626X/1 and NERC EOS Cloud to YW, and NE/S007229/1 to AW) and QMUL's Apocrita MidPlus computational facilities (doi.org/10.5281/zenodo.438045). We thank Philip Bayer for fixing a bug in the install script and are grateful to the early adopters of the software for the encouragement.

Chapter 4: Sequenceserver: a modern graphical user-interface for BLAST

Contributions

Yannick Wurm conceived the software and outlined the software development and the user-experience design principles detailed in Methods. I created the software, including the underlying architecture, wrote 90% of the code, reviewed and integrated code contributed by co-authors, directly supervised the work of seven co-authors, contributed to the user-interface design (how the software looks) and to the software development and the user-experience design principles. Yannick Wurm and I wrote the manuscript. Ben Woodcroft provided significant input on an initial draft of the manuscript. All authors read and contributed to improving the manuscript later.

A variation of the chapter has been published as brief communication:

A Priyam, B Woodcroft, V Rai, I Moghul, A Munagala, F Ter, H Chowdhary, I Pieniak, L Maynard, M Gibbins, H Moon, A Davis-Richardson, M Uludag, N Watson-Haigh, R Challis, H Nakamura, E Favreau, E Gómez, T Pluskal, G Leonard, W Rumpf, Y Wurm (2019) "Sequenceserver: a modern graphical user interface for custom BLAST databases" Molecular Biology and Evolution 36(12): 2922–2924.

Introduction

The dramatic drop in sequencing costs has created many opportunities for individuals and groups of researchers to generate genomic or transcriptomic sequences from previously understudied organisms. Many research questions require small- or large-scale sequence comparisons, and BLAST (Basic Local Alignment Search Tool) is the most established tool for many such analyses (Altschul *et al.*, 1990; Camacho *et al.*, 2009). Unfortunately, BLAST analysis of new data can be challenging. There are delays before new data are submitted to and become publicly available on central BLAST repositories such as the NCBI (National Center for Biotechnology Information), and only small queries are feasible on such repositories. BLAST can be downloaded and installed locally, but its usage can be challenging for researchers without experience of command-line interfaces. Finally, commercial software to overcome such hurdles is too costly for many laboratories.

Here, we present Sequenceserver, a free graphical interface for BLAST designed to increase the productivity of biologist researchers performing and interpreting BLAST searches on custom data sets, and of bioinformaticians setting up shared laboratory or community databases. It has a user-centric focus (Garrett, 2010) on accompanying researchers through their work process. Below, we provide an overview of Sequenceserver features that facilitate BLAST query submission and interpretation.

Results

Assisted installation and BLAST query submission

Installing Sequenceserver on computers running macOS or Linux is typically rapid, requiring only one or few commands (see online documentation). If necessary, Sequenceserver automates the download of BLAST (Camacho *et al.*, 2009) binaries and can manage the conversion of FASTA files to BLAST databases. A user accesses Sequenceserver's graphical interface in a web browser at localhost:4567 (Figure 4.1A). All

detected BLAST databases are automatically listed here. The user types, pastes or drag-and-drops FASTA format query sequences into a text-field (Figure 4.1A). To prevent common errors, an alert message is shown, and query submission is disabled if the query is invalid (e.g., combining nucleotide and protein sequences). The user then selects databases. The appropriate basic BLAST algorithm will automatically be used (Figure 4.2). When multiple algorithms are appropriate, a pull-down in the BLAST submission button allows the user to toggle between them. An "advanced parameters" field provides access to all standard BLAST parameters.

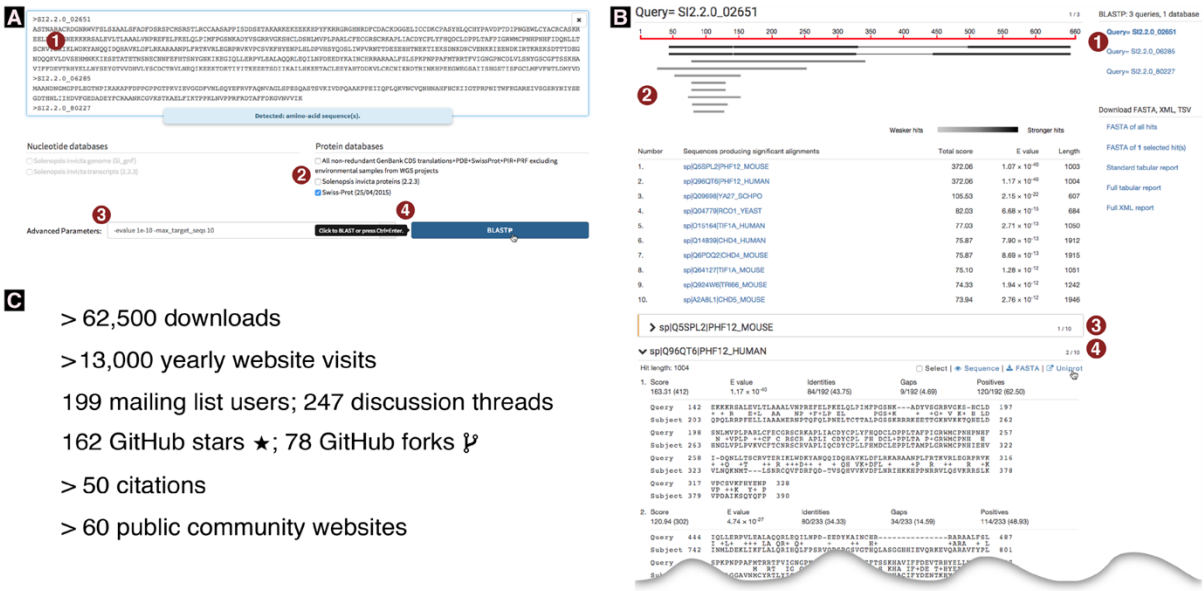


Figure 4.1: Sequenceserver's user-interface and usage statistics

(A) Partial screenshot of the query interface. Numbers circled in red highlight the steps involved and some specific features. (1) Three or more sequences were pasted into the query field (typewriter font; only the identifier is visible for the third sequence); a message confirms to the user that these are amino acid sequences. (2) The Swiss-Prot protein database was the first database to be selected. As a result, additional database selections are limited to protein databases; nucleotide databases are disabled. (3) Optional advanced parameters were entered which constrain the results to the ten strongest hits with *E*-values stronger than 10^{-10}. (4) The BLAST button is automatically activated and labelled "BlastP" as this is the only possible basic BLAST algorithm for the given query-database combination. As the user's mouse pointer hovers over the BlastP button, a tooltip indicates that a keyboard shortcut exists for this button. (*B*) Partial screenshot of a Sequenceserver BLAST report. An interactive version of this figure is online

at sequenceserver.com/paper/resultsinteractive (last accessed August 25, 2019). Three amino acid sequences were compared against the Swiss-Prot database using BlastP with an *E*-value cutoff of 10^{-10} and keeping only the ten strongest hits per query. This screenshot shows a portion of the results for the first query. Numbers circled in red highlight some specific features of this report. (1) An index overview summarizes the query and database information and provides clickable links to query-specific results. (2) Results for the first query are shown. These include a graphical overview indicating which parts of the query sequence align to each hit, a tabular summary of all hits, and alignment details for each hit. (3) The first hit is selected for download; its alignment details have been folded away. (4) The user is studying the second hit; the mouse pointer hovers over the link to the hit's UniProt page. (*C*) Sequenceserver usage as of June 11, 2019. These include download statistics from rubygems.org/gems/sequenceserver, Google Analytics statistics for sequenceserver.com, and citation statistics from app.dimensions.ai/details/publication/pub.1085102830, and GitHub statistics from github.com/wurmlab/sequenceserver.

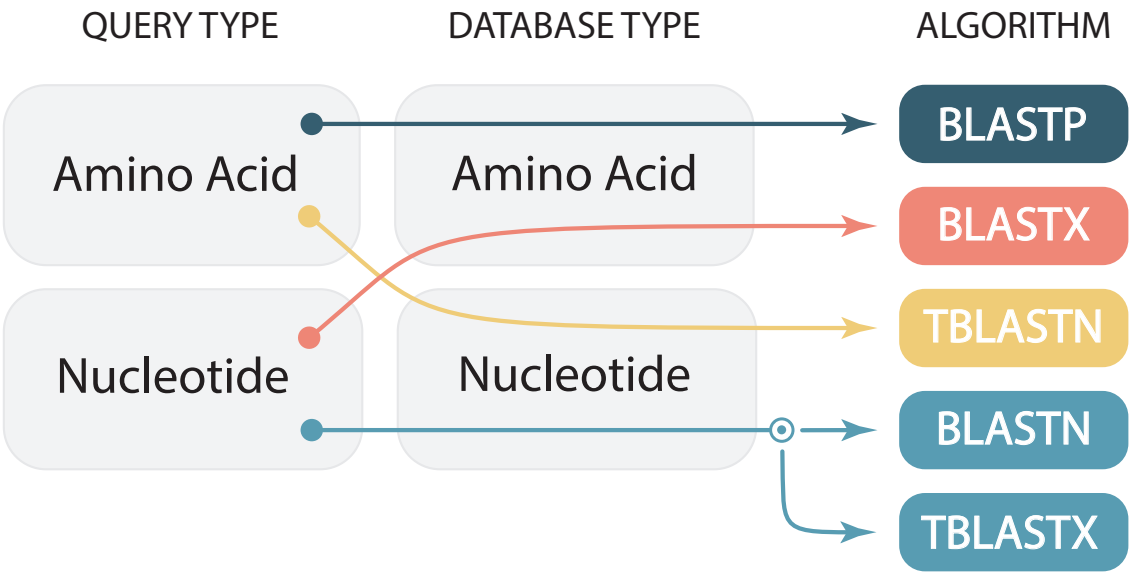


Figure 4.2: Automatic BLAST algorithm selection

BLAST includes five basic algorithms (right column). Arrows indicate how Sequenceserver automatically selects an appropriate BLAST algorithm based on the sequence types of the query (left column) and selected databases (middle column). For the first three combinations of query and database types, only one algorithm is possible. The circle indicates that for nucleotide query and nucleotide database, the user can choose between BLASTN and TBLASTX.

Usage by individual researchers and as part of community databases

Usage statistics including downloads, preprint citations, GitHub, and mailing list participation (Figure 4.1C) indicate that Sequenceserver is extensively used for molecular-genetic research on emerging model organisms (Table A1.1). For example, Sequenceserver installations on personal computers helped characterize the evolution of tunicate genomes (Blanchoud *et al.*, 2018), fire ant olfactory genes (Pracana, Levantis, *et al.*, 2017), and loci affecting Sorghum shoot architecture (McCormick, Truong and Mullet, 2016). Sequenceserver has also been used to analyse human prostate cancer genomes (Seim *et al.*, 2017) and to identify bacteria affecting shelf life of milk (Reichler *et al.*, 2018).

Importantly, Sequenceserver also represents a main querying mechanism for more than 50 community genome databases (Table A1.2), including the PHI-base database of genes underpinning pathogen–host interactions (Winnenburg *et al.*, 2006), an initiative to sequence 1,000 wild yeast genomes (Shen *et al.*, 2016), and the reefgenomics.org coral genomics database; last accessed August 25, 2019 (Liew, Aranda and Voolstra, 2016). Such community resources typically integrate Sequenceserver as part of larger web servers (e.g., Nginx (Reese, 2008)) and customize it by adding links from BLAST hits to genome browsers or other gene-specific information. Additionally, many password protected Sequenceserver instances exist for unpublished data.

Outlook

In creating Sequenceserver, we aimed to respect user-centric design principles, open-source, and sustainable software engineering practices. Our software is built using Ruby and Javascript frameworks commonly used for professional software development. The resulting robust architecture and flexibility facilitate customization and integration with other tools. This has led to contributions of improvements and bug-fixes by 21 bioinformaticians unrelated to the initial project; many are now co-authors. Our community is testing the ability to import pre-existing BLAST or DIAMOND XML result files (Buchfink,

Xie and Huson, 2015), and new manners of visualizing results (Wintersinger and Wasmuth, 2015; Cui *et al.*, 2016). Such efforts will continue to improve the ability of researchers to analyse and interpret genomic data.

Analytics and A/B testing can provide a data-driven framework for the development of new features and user-interface optimisations. For example, we can suppose that a new feature or user-interface optimisation should reduce the time it takes to run a BLAST search, or the time taken to find matching database sequences on results pages (e.g., how long it takes for a user to click on a FASTA download button after the results page has loaded). A random sample of users would then be exposed to the new feature, while a control group would continue to use the software without any changes. Usage statistics derived from the two groups can then be used to tell if the new feature improved or degraded the user-experience and by how much. Such an approach has two benefits. First is the reduction of developer's bias due to the large sample size of tens or hundreds of users. This is a rather important point as a single developer, or a small development team cannot anticipate all the ways in which a global user base comprising of different age groups and backgrounds will perceive the software changes. The second benefit is the formalisation of abstract design principles into tangible goals (e.g., time taken to find relevant database sequences) that can be more easily communicated, discussed, and acted upon.

A challenge in conducting A/B tests is that our software is typically installed on user's computer for use by an individual or a small group, and maybe seldom updated. However, this can be overcome by collaborating with large community databases (Table A1.2). Finally, conducting A/B tests will require integration with an analytics software like Matomo (matomo.org) that can handle data collection in an anonymised and GDPR compliant manner. Integration with analytics software will enable consent-based collection of further usage statistics that can help identify a baseline for A/B testing (e.g., randomisation strategy, how long to run the test) as well as avenues for software improvements, thus providing a complete data-driven framework for iterative evolution of our software.

Methods

Technical implementation details

We developed Sequenceserver from scratch rather than basing our work on the NCBI's initial Perl/CGI wwwblast wrapper to reduce technical debt (Lehman, 1980). The core of Sequenceserver is written in the Ruby language (Flanagan and Matsumoto, 2008) popular for creating websites (Ruby, Copeland and Thomas, 2020) and bioinformatics tools (Goto *et al.*, 2010), while JavaScript and HTML/CSS are used for layout and interactions in the web browser. We use pre-existing tools and libraries to facilitate development: The lightweight framework Sinatra (Harris and Haase, 2011) is used to create URL endpoints to load the search form and run BLAST searches from the browser. BLAST searches are delegated to the compiled command line version of BLAST (Camacho *et al.*, 2009); we use Ox (github.com/ohler55/ox) to parse BLAST XML and create the HTML report. Underscore (underscorejs.org), HTML5 Shiv (github.com/afarkas/html5shiv), jQuery (jquery.com), jQuery UI (jqueryui.com), Webshim (afarkas.github.io/webshim/demos), and Bootstrap (getbootstrap.com) libraries create a uniform scripting environment (for dynamic aspects of the user interface) and a consistent look-and-feel (for visual layout) across browsers. The d3 (d3js.org) and BioJS (Gómez *et al.*, 2013) libraries are used respectively for generating the graphical overview and the sequence viewing interface. Details regarding versions of the different software libraries are indicated in the source code repository at github.com/wurmlab/sequenceserver.

Sustainable software development approach

We followed six software engineering practices to facilitate and accelerate development while increasing robustness, improving the long-term sustainability of the software (Prlić and Procter, 2012; Wilson *et al.*, 2014). First, we used an open source and agile development approach (Shore and Warden, 2007) involving frequent incremental improvements, peer review and frequent deployment on our servers and within the community. Second, we structured the software according to the object-oriented programming paradigm (Weisfeld,

2013) to cleanly separate different parts of code. Third, we followed two important software development principles: "don't repeat yourself" (DRY) leads to fewer lines of code and thus fewer bugs, and makes it easier to read and understand code than if similar commands are repeated in several places (Hunt and Thomas, 2000); "keep it simple, stupid" (KISS) reduces unnecessary complexity and thus lowers risks and leads to higher maintainability (Raymond, 2003). Fourth, we reuse widely established software packages and libraries (see above) to benefit from work done by others. This accelerates our work and reduces the amount of Sequenceserver-specific code, which in turn further reduces the likelihood of adding bugs (Sametinger, 1997). Fifth, we implemented unit and integration tests (Ammann and Offutt, 2016) for many parts of Sequenceserver's code, and use continuous integration (travis-ci.org) to ensure these tests are automatically run whenever a change is made to the code, thus increasing the likelihood and speed of detecting errors. Sixth, we use automatic code checkers including rubocop (github.com/bbatsov/rubocop) and w3 validator to ensure that our code respects relevant style guides and development principles. Such respect of style standards (e.g., names of variables and methods, code structure and formatting) makes code more accessible to others than if we had chosen no or different conventions (Martin, 2009; Wurm, 2015). Finally, we use the Code Climate platform (codeclimate.com) for automated reviews of code quality.

User centric design of graphical user interface

To ensure a fluid user experience that increases researcher productivity, we designed Sequenceserver around eight modern user interface design principles. First, the interface contains only essential information to minimize distractions for the user. Second, the information is laid out in a clear and hierarchically structured manner. As part of this, we paid special attention to typography, using typefaces specifically designed for legibility and aesthetics on electronic devices (Roboto and Open Sans). Third, we used automation where possible to minimize the amount of decisions the user must make. For example, we limit the choices for algorithm selection based on query type and databases selection – this is because only a single basic BLAST algorithm is possible for all cases except for nucleotide-nucleotide search (Figure 4.2). Fourth, we use interactive visual feedback and cues for step-by-step

discovery of the workflow. For example, the BLAST button remains disabled until the user has provided query sequence(s) and selected target databases. If the user tries to click the BLAST button while it is disabled, a tooltip indicates that a required input is missing. Similarly, the selection of protein databases is automatically disabled if the user has already selected a nucleotide database (and vice versa). Fifth, we remain consistent and contextual with regards to user interaction. For example, notification of detection of sequence type does not depend on how the query sequence was provided. This notification is shown below the query sequence input field – where the user is likely to look after query input – instead of using a global designated notification area or displaying pop-up windows that can be disruptive or are ignored. Similarly, a "clear query" button is shown only after the user has provided query sequence(s) and is positioned where a user is likely to look for it. Sixth, we try not to let the advantages of a graphical interface and efforts to create an easily accessible user experience limit the scope of what the user can do. For example, all possible advanced BLAST search options can be entered via a generic input field. Similarly, tooltips over report download links are only shown after the mouse pointer has hovered for at least 500ms. This delay means most users will not be bothered by tooltips after they have used the interface a few times. Seventh, we exploit intuitive human notions of colours. For example, if the user erroneously tries to combine nucleotide and amino acid sequences in the query, the query input-area is gently highlighted using a red border to indicate an error. At a different level, in the graphical overview shown for each query, the colour of each hit indicates its strength, with stronger e-values being darker. Finally, the wording of error messages is similar to an informal human conversation to create empathy and familiarity, which may also clarify that Sequenceserver is built by a community of scientists.

Data Availability

Source code is available under GNU Affero General Public License (AGPL) 3.0 at github.com/sequenceserver (last accessed August 25, 2019). Additional documentation is available online at sequenceserver.com (last accessed August 25, 2019).

Acknowledgments

We thank the many Sequenceserver users and contributors for their input. During the creation of Sequenceserver, Y.W. was funded by a European Research Council grant to Laurent Keller. B.J.W. was supported by the United States Department of Energy (DE-SC0004632). While writing this manuscript, Y.W. and A.P. were supported by the Biotechnology and Biological Sciences Research Council (BB/K004204/1) and the Natural Environment Research Council (NE/L00626X/1).

Chapter 5: Choosing the best gene predictions with GeneValidator

Contributions

I wrote most of the chapter with inputs from Ismail Moghul and Yannick Wurm. Ismail provided the figures and the code snippets included in the chapter, modified GeneValidator software where it was required, and contributed to the writing with significant input from myself. The section “Merging gene predictions from different sources” was Yannick’s idea.

The annex has been published as a book chapter:

I Moghul*, A Priyam*, Y Wurm (2019) "Choosing the Best Gene Predictions with GeneValidator" Gene Prediction, Methods in Molecular Biology (2019), Volume 1962; Chapter 16.

Abstract

GeneValidator is a tool for determining whether the characteristics of newly predicted protein-coding genes are consistent with those of similar sequences in public databases. For this, it runs up to seven comparisons per gene. Results are shown in an HTML report containing summary statistics and graphical visualisations that aim to be useful for curators. Results are also presented in CSV and JSON formats for automated follow-up analysis.

Here, we describe common usage scenarios of GeneValidator that use the JSON output results together with standard UNIX tools. We demonstrate how GeneValidator's textual output can be used to filter and subset large gene sets effectively. First, we explain how low-scoring gene models can be identified and extracted for manual curation – for example, as input for genome browsers or gene annotation tools. Second, we show how GeneValidator's HTML report can be regenerated from a filtered subset of GeneValidator's JSON output. Subsequently, we demonstrate how GeneValidator's GUI can be used to complement manual curation efforts. Additionally, we explain how GeneValidator can be used to merge information from multiple annotations by automatically selecting the higher scoring gene model at each common gene locus. Finally, we show how GeneValidator analyses can be optimised when using large BLAST databases.

Introduction

Using accurate gene annotations is important because they affect subsequent analyses (Yandell and Ence, 2012). For some species, annotations can be downloaded directly from a public database such as Ensembl or NCBI (Benson *et al.*, 2018). For newly sequenced species, approaches to identify protein-coding genes in a genome sequence typically combine evidence from multiple data sources (including *ab initio* models, ESTs, RNA-seq and protein alignments) (Holt and Yandell, 2011; Hoff *et al.*, 2016; Keilwagen *et al.*, 2018). Whether gene feature annotations are downloaded from a public database or newly generated, they may contain errors resulting from biases of the underlying data, algorithmic choices (Schnoes *et*

al., 2009), and the general limitations of a one-dimensional representation of DNA sequences. Common errors include frameshifts, incorrect exon-intron structure, incorrect merging of adjacent genes, and incorrect splitting of genes at long intron positions (Steijger *et al.*, 2013).

We previously described GeneValidator (GV), a tool to evaluate the quality of protein-coding gene predictions based on comparisons with a database of known proteins (Drăgan *et al.*, 2016). In brief (Figure 5.1), GV first runs a BLAST search against the given database, retaining sequences of hits with e-value stronger than 10^{-5}. Next, GV runs up to seven validations on each gene prediction. Each validation tests if the characteristics of the query gene deviate from those of similar sequences in the reference database. Based on predefined thresholds, the result of each validation is a pass or a fail. The overall score of the prediction is a scaled percentage of the validations that passed. Predictions with a score lower than 75 (i.e., more than one failed validation) may be regarded as potentially problematic. Explanation of the approach and an overview of the data underlying each validation is included in the HTML report, along with several visualisations to facilitate interpretation. Detailed results are also available in CSV and JSON format for spreadsheet and programmatic access.

Results produced by GV are dependent on the quality and coverage of the database used for validation. Furthermore, higher scores indicate consistency with database sequences and not biological truths. Several publicly available databases of protein sequences such as Swiss-Prot (The Uniprot Consortium, 2017), UniRef50 (Suzek *et al.*, 2015; The Uniprot Consortium, 2017), TrEMBL (The Uniprot Consortium, 2017), or NR (Benson *et al.*, 2018) can be used with GV. The GV approach becomes increasingly reliable as proteomes of more species are submitted to these databases by the global research community, and as the quality of submitted sequences improve due to experimental validation, manual verification by experts, and technological and algorithmic advances in sequencing and automated gene prediction. We created GV to be flexible. Many of GV's features are designed to facilitate automatic processing of large gene sets (*e.g.*, whole-genome annotation) as part of custom workflows. These include GV's versatile JSON output, ability to leverage HPC facilities, and the possibility to use advanced BLAST search options. GV also includes a web server that

can be used as a shared resource. Here, we discuss five common use cases of GV that can be easily incorporated into custom workflows.

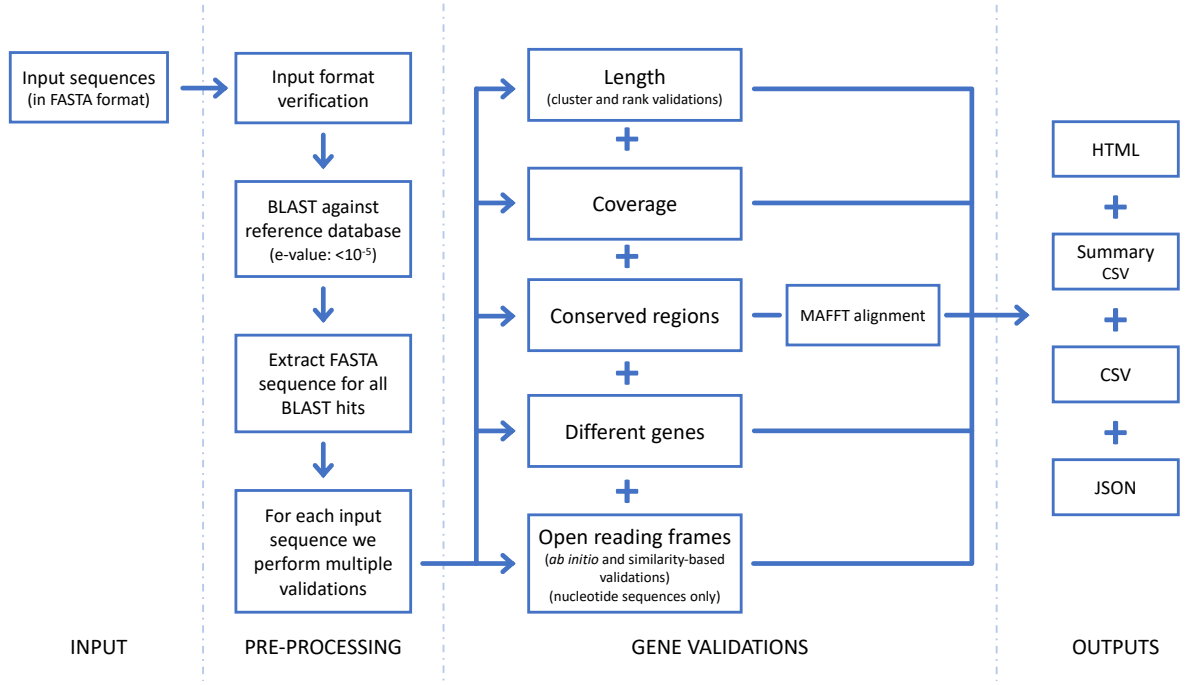


Figure 5.1: High-level schematic of the steps carried out by GeneValidator.

Installing and running GeneValidator

GV runs on Linux and macOS. To install GV, run the command shown below. This will install GV and all its dependencies to a directory called `"genevalidator"` in the current working directory.

```
sh -c "$(curl -fsSL https://install-genevalidator.wurmlab.com)"
```

The software includes example sequences to test the installation. The following command can be used to run GV on these example sequences with the included Swiss-Prot database. GV will print the results of validations for each gene prediction to the terminal, ending with a summary, and the directory where detailed results were saved to.

```
genevalidator --db genevalidator/blast_db/swissprot \
    --num_threads 4 \
    genevalidator/exemplar_data/protein_data.fa
```

GeneValidator workflows

A gene set will almost inevitably contain some gene predictions with low scores. It can be desirable to curate these manually. Here, we begin by providing two approaches to facilitate inspection of these low-scoring predictions. First (Subheading 3.1), we show how to use GV's JSON output to extract the sequence identifiers of low scoring gene predictions. Among other things, these can be used to subset the initial gene set, to prioritise inspection in a genome browser (Buels *et al.*, 2016), or for annotation editing in a tool such as Apollo (Lee *et al.*, 2013). Second (Subheading 3.2), we show how to create a new HTML report by subsetting GV's JSON output. This can reduce the need to navigate through a long HTML report. Subsequently (Subheading 3.3), we introduce GV's graphical interface. This is helpful for rapidly viewing how GV's validation results change during manual curation.

We also provide guidance on two more general challenges based on our applications of GV. First (Subheading 3.4), we show how GV can be used to automatically select the best gene model from multiple gene sets at each common gene locus. Furthermore (Subheading 3.5), we show how to restrict GV to use a specific subset of a BLAST database. This is to avoid BLAST searching against sequences unlikely to be informative.

Extracting sequence identifiers of low scoring gene predictions

GV's JSON output can be used with JQ (stedolan.github.io/jq), a command-line JSON processor (included in GV package), to select gene predictions matching a particular criterion and access validation results and associated metadata. In the example below, we extract identifiers of predictions with a score lower than 75 (i.e., having failed more than one validation) and having at least two BLAST hits for manual curation. The idea is that while having two BLAST hits is insufficient for GV's statistical tests (and thus results in a low

score), they may provide sufficient evidence for biologically interpreting whether the prediction could be appropriate.

1. Extract FASTA header of gene predictions that have more than two BLAST hits and an overall score of less than 75.

```
jq --raw-output ".[] |
  select(.no_hits >= 2 and .overall_score < 75) |
  .definition" input_file_results.json \
  > sequence_definitions.txt
```

2. Extract sequence identifier (first word of the FASTA header) using the cut command.

```
cut -d ' ' -f 1 sequence_definitions.txt \
  > sequence_ids.txt
```

Subsetting the HTML report to only low scoring gene predictions

GV's JSON output can be filtered using JQ and input back to GV to reproduce results for the selected gene predictions. This is useful to create smaller HTML report, for example focusing on a particular gene family. In the example below, we subset GV's output for the low scoring gene predictions selected in 3.1.

1. Select gene predictions that have more than two BLAST hits and an overall score of less than 75.

```
jq "[ .[] |
     select(.no_hits >= 2 and .overall_score < 75) ] |
     sort_by(.overall_score)" input_file_results.json \
     > input_file_results_subset.json
```

2. Reproduce GV's output.

```
genevalidator --json input_file_results_subset.json
```

Using GeneValidator web server to iteratively refine gene models

Although running GV from the command-line is ideal for processing of large datasets and custom workflows, a graphical user interface can facilitate iterative usage. For example, during manual curation of gene models, running GV repeatedly as a gene model is revised can help a curator verify that changes, they are making indeed improve the gene model. Building on the lessons learnt when developing the Sequenceserver BLAST interface (Priyam *et al.*, 2019), we also built a graphical user interface (app) for GV that is accessible through a web browser.

1. Launching GV app requires the path to a directory containing one or more BLAST databases; the interface (accessible at localhost:5678) is opened automatically in the default browser.

   ```
   genevalidator app --num_threads 4 \
     --database_dir genevalidator/blastdb/
   ```

2. To validate gene predictions, paste the corresponding FASTA sequences into the text area, select the database to compare to, and click "Analyse Sequences" (Figure 5.2). The results are then shown on the same page.

We also host a GV web server at genevalidator.wurmlab.com with two caveats: first, it is suitable for up to 10 queries at a time, and second, given computational constraints on this server, we only provide the Swiss-Prot and the UniRef50 databases.

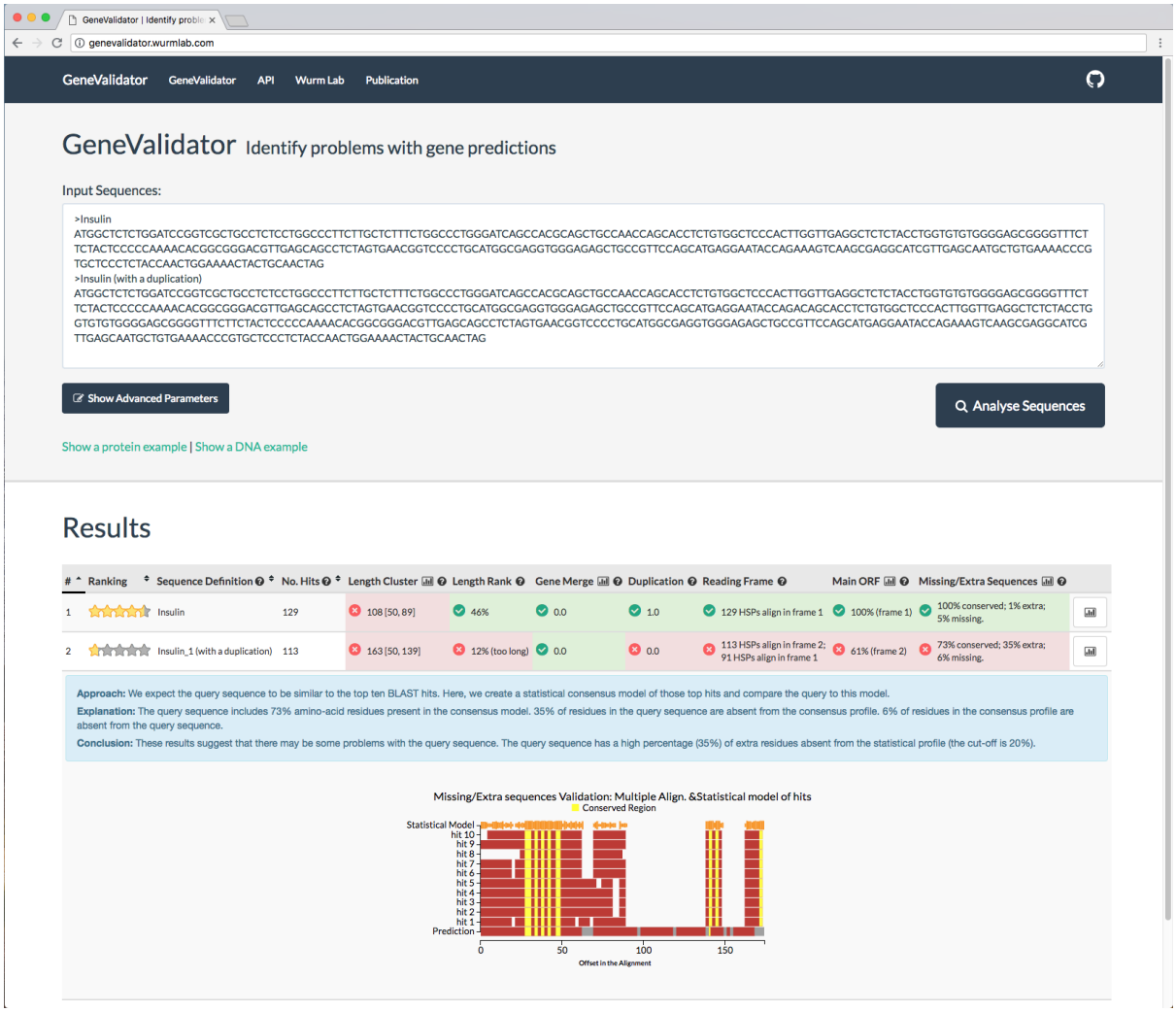


Figure 5.2: Screenshot of GeneValidator web application

A screenshot of the GeneValidator web application as launched from the command line via `"genevalidator app"` or by accessing genevalidator.wurmlab.com.

Merging gene predictions from two different sources

Different gene prediction approaches are unlikely to generate identical gene models for a locus. GV can be used to select the higher scoring gene model for each locus from multiple gene sets. Briefly, we first identify annotations corresponding to the same locus from the different sources (steps 1 – 3 below). Subsequently, we generate a FASTA file containing

alternative predictions for each locus and use GV's "--select_single_best" option to select the higher scoring one (step 4 below).

We make multiple simplifying assumptions to generate a mapping of annotations corresponding to the same locus from the different sources (steps 1 – 3 below). Specifically, we assume that we have a single transcript (splice-form) per source per locus, that gene predictions from different loci do not overlap, and that annotations are available in a GFF3 format file. Often, additional pre-processing of gene sets will be necessary to take these into account.

1. Intersect the transcript annotations in the GFF3 files (requires prior installation of bedtools). We require that both hits are on the same strand ("-s"). If comparing more than two GFF3 files, see the bedtools documentation ("-b" can take multiple values). The output file contains the entire input record from both input files ("-wa -wb").

```
awk '/\tmRNA\t/' geneset1.gff > geneset1_mrnas.gff
awk '/\tmRNA\t/' geneset2.gff > geneset2_mrnas.gff
bedtools intersect -wa -wb -s \
  -a geneset1_mrnas.gff -b geneset2_mrnas.gff \
  > geneset_overlaps.bed
```

2. Extract the GFF3 attributes columns (i.e., the 9th and 18th column) which contain the sequence identifiers.

```
awk '{printf ("%s;\t%s;\n", $9, $18)}' \
  geneset_overlaps.bed > attributes_columns.tsv
```

3. Extract the sequence identifiers from the attributes columns.

```
perl -nle '@ids = /ID=(.*?);/g;
  print join("\t", @ids) if @ids' \
  attributes_columns.tsv > mapping_ids.tsv
```

4. Now that we have identifiers of the annotations corresponding to the same locus from both the gene sets, their respective sequences can be extracted and then used with GV's "--select_single_best" option.
   a. Create indexes for each of the FASTA files (requires prior installation of samtools).

```
samtools faidx geneset1.fasta
samtools faidx geneset2.fasta
```

   b. Create output FASTA file.

```
touch output.fa
```

   c. Loop over the "mapping_ids.tsv" file. Extract FASTA sequence for each ID and write them to a temporary FASTA file. Run GV using the "--select_single_best" option on the temporary FASTA file. The "--select_single_best" mode prints the highest scoring sequence to STDOUT in FASTA format, which is written to the output file previously created.

```
cat mapping_ids.tsv | while read -r line; do
  echo "$line" | cut -f 1 | \
        xargs samtools faidx geneset1.fasta \
        > gv_run_tmp.fa
  echo "$line" | cut -f 2 | \
        xargs samtools faidx geneset2.fasta \
        >> gv_run_tmp.fa
  genevalidator --select_single_best gv_run_tmp.fa \
        >> output.fa
  rm gv_run_tmp.fa
done
```

It may be desirable to include gene models unique to both sets in the final output. We leave this as an exercise for the reader.

Using NCBI's non-redundant database of protein sequences with GV

While it is desirable to validate gene predictions against a gold standard database like Swiss-Prot, its limited coverage (The Uniprot Consortium, 2017) makes this challenging for many species. At the same time, technological advances continue to increase the quality of automated predictions (Minoche *et al.*, 2015). This makes it tempting to use a more comprehensive database such as NCBI's non-redundant collection (NR) of manually reviewed as well as automatically generated protein sequences for validation. However, the large size of the NR database means BLAST searches can take days. We show how to use BLAST's ability to restrict searches to a list of identifiers (ncbi.nlm.nih.gov/books/NBK279673) to accelerate a GV analysis. For this, we first restrict the BLAST search to a particular taxonomic lineage to avoid BLAST searching against sequences unlikely to be informative. Additionally, we exclude sequences from the focal species to avoid circular self-validation.

For the implementation below, we consider the example of the red fire ant, *Solenopsis invicta* (Wurm *et al.*, 2011). We first obtain taxon identifiers of all species in Eukaryota (id: 2759). Subsequently, we exclude all *Solenopsis* species (taxonomy id: 13685). We then obtain GenInfo identifiers (GI numbers) of all sequences in the retained taxa. We finally run GV using this list.

1. Obtain a list of eukaryotic taxon identifiers (this requires prior installation of Taxonkit).

```
taxonkit list --ids 2759 --indent "" \
  > taxon_ids_eukaryotes.txt
```

2. Obtain a list of *Solenopsis* taxon identifiers.

```
taxonkit list --ids 13685 --indent "" \
  > taxon_ids_solenopsis.txt
```

3. Subtract the two.

```
grep -Fvx -f taxon_ids_solenopsis.txt \
  taxon_ids_eukaryotes.txt > taxon_ids.txt
```

4. Download a tab-delimited file from NCBI linking taxon ids and GI Numbers.

```
curl -L -O
ftp://ftp.ncbi.nih.gov/pub/taxonomy/accession2taxid/prot.accession2t
axid.gz
```

5. Use csvtk (github.com/shenwei356/csvtk), a multithreaded CSV/TSV processor (packaged with GV), to extract the rows where the taxid is in the taxon_ids.txt file.

```
zcat prot.accession2taxid.gz | \
      csvtk --tabs grep --fields taxid \
            --pattern-file taxon_ids.txt | \
      cut -f 4 | tail -n +2 > gi_list.txt
```

6. Finally, we pass this file to GV using "`--blast_option`" option.

```
genevalidator --blast_options "-gilist gi_list.txt" --db nr \
  --num_threads 40 geneset1.fa
```

Starting with BLAST+ version 2.8.0 (in development at the time of this writing) steps 4 & 5 can be skipped and the list of taxon ids from step 3 can be passed directly to BLAST using the new "`-taxidlist`" option.

Tips and tricks

1. GV's overall score is based on the percentage of validations that pass, *i.e.,* where the score is above a threshold that we have determined to be appropriate. To emphasise the fact that GV results are highly dependent on the quality of information in databases and cannot be solely relied upon to classify a 'perfect' gene prediction, the overall score is decreased by 10%. The highest possible score is thus 90%.

2. GV will run the validations provided there are at least five BLAST hits for a given prediction. This can be changed using the "`--min_blast_hits`" option. A higher number of BLAST hits will increase the relevance of the comparisons.

3. GV generates several summary statistics for the input gene set. These include first, second and third quartiles of the overall scores, number of good and bad predictions, and number of predictions with insufficient BLAST hits. In addition to providing an overview of the quality of the input gene set, the summary statistics can be used to choose between predictions from two different sources.

4. GV includes a tool for downloading sequence databases from NCBI to use for comparisons (i.e., "`genevalidator ncbi-blast-dbs`"). This is a parallelised alternative to the "update_blastdb.pl" script included in BLAST+ package.

5. GV is also able to run BLAST searches on NCBI servers using BLAST's "`-remote`" option (*e.g.*, "`genevalidator --db 'swissprot -remote' geneset.fa`"). This has the benefit of being able to immediately use the most up-to-date version of a given database. However, using a remote BLAST database is very slow. We recommended using this for validating only a few genes (*e.g.,* fewer than 25).

6. It is possible to run BLAST independently and to subsequently provide the output XML ("`-outfmt 5`") or tab-delimited ("`-outfmt 6`") to GV. This can be particularly useful if BLAST results have already been produced for other analyses, or when BLAST can be run on a cluster.

7. BLAST is often the slowest step of GV pipeline, especially when working with large datasets. In such cases, DIAMOND (Buchfink, Xie and Huson, 2015) can be used instead of BLAST for (up to 20,000x!) faster database searching. Since DIAMOND's XML output is compatible with BLAST, it can be used directly with GV along with one additional input, i.e., a FASTA file of hit sequences (when used with BLAST, GV is able to automatically extract hit sequences from BLAST database). Our wiki

(github.com/wurmlab/genevalidator/wiki) provides detailed instructions for using GV with DIAMOND.

8. To resume a terminated analysis, GV can be run with "`--resume`" option. In resume mode, GV skips previously successful steps, including running BLAST. Gene predictions that were successfully processed are skipped as well.

9. It is possible to split an input gene set into multiple chunks, run GV on each chunk across multiple compute nodes, and combine the results for each chunk into a single report.

    1. After splitting the input file and running GV on each input file, the following command can be used to merge the individually produced GV JSON files.

        ```
        cat */*.json | jq '.[] ' | jq --slurp '.' > MERGED_JSON
        ```

    2. The merged JSON can then be used to produce a single report for the whole gene set.

        ```
        genevalidator --json MERGED_JSON
        ```

Acknowledgements

This work was supported by the Natural Environment Research Council [grant NE/L00626X/1] and the Biotechnology and Biological Sciences Research Council [grant BB/K004204/1 and BB/M009513/1]. This research used Queen Mary's Apocrita HPC facility, supported by QMUL Research-IT (doi.org/10.5281/zenodo.438045).

Part 2: Applications

Chapter 6: Fire ant social chromosomes: Differences in number, sequence and expression of odorant binding proteins

Contributions

Rodrigo Pracana and Ilya Levantis led the study of fire ant odorant binding proteins (OBPs) under supervision of Yannick Wurm. They did most of the biological analyses. The OBPs reported in this study were at first identified using a newer, more contiguous assembly and later transferred back to the reference assembly. I performed automated annotation of the new assembly, created the software used for manual curation of the OBPs, transferred the annotations back to the reference assembly together with Ilya and discovered some of the errors in the reference assembly reported in this study together with Rodrigo. I helped draft the 'OBP discovery and manual gene model curation' section and contributed to later versions of the manuscript.

The chapter has been published:

R Pracana, I Levantis, C Martínez-Ruiz, E Stolle, A Priyam, Y Wurm (2017) "Fire ant social chromosomes: Differences in number, sequence and expression of odorant binding proteins" Evolution Letters, 1(4): 181– 228

Introduction

Variation in social behaviour is common yet our knowledge of the mechanisms underpinning its evolution is limited (Robinson, Grozinger and Whitfield, 2005; Johnson and Linksvayer, 2010). The fire ant *Solenopsis invicta* provides a rare, textbook example of variation in a fundamental social trait: some colonies have one queen, whereas others have up to dozens of queens. Queens that will form their own single-queen colony typically disperse over greater distances and can effectively colonize newly available habitats. In contrast, multiple-queen colonies can outcompete single-queen colonies in saturated habitats and harsh environments and can split by fission (Bourke and Heinze, 1994; Ross and Keller, 1995; Tschinkel, 2006). Multiple additional traits differ between the two social forms, including in queen fecundity, colony size, worker size distribution, and worker aggressiveness (Ross and Keller, 1995; DeHeer, Goodisman and Ross, 1999; Keller and Ross, 1999; Goodisman, DeHeer and Ross, 2000; DeHeer, 2002; Buechel, Wurm and Keller, 2014; Huang and Wang, 2014).

A series of landmark studies (Ross, 1997; Keller and Ross, 1998; Ross and Keller, 1998) demonstrated that the two social forms are under the control of a Mendelian element. This element was first identified in a screen of electrophoretic markers as a polymorphic protein coding gene, *Gp-9*, with two alleles: *Gp-9B* and *Gp-9b* (Ross, 1997). If a colony includes only *Gp-9 BB* workers, they will accept a single *Gp-9 BB* queen and execute any additional queens. In contrast, if more than ~20% of the workers in a colony are *Gp-9 Bb* heterozygotes, they will execute reproductively active *Gp-9 BB* queens but accept dozens of *Gp-9 Bb* queens (Ross, 1997; Keller and Ross, 1998; Ross and Keller, 1998; Keller and Ross, 1999; DeHeer, Goodisman and Ross, 1999; Ross and Keller, 2002; Gotzek and Ross, 2007). In contrast, *Gp-9 bb* queens die before becoming reproductively active (Ross, 1997; DeHeer, Goodisman and Ross, 1999; Keller and Ross, 1999; Gotzek and Ross, 2007; Trible and Ross, 2016). The workers discriminate between queens of alternate genotypes based on olfactory cues (Keller and Ross, 1998; Ross and Keller, 1998, 2002), such as differences in the queens' cuticular hydrocarbon profiles (Eliyahu *et al.*, 2011; Trible and Ross, 2016). Because workers carrying the *Gp-9b* allele recognize whether queens also carry this allele and execute those that do

not, this system represents a rare example of a "green beard gene" (Keller and Ross, 1998), named after a theoretical model of a behavioural selfish genetic element (West and Gardner, 2010).

In another landmark study, Krieger and Ross (Krieger and Ross, 2002) demonstrated that *Gp-9* encodes an odorant binding protein (OBP). OBPs are essential components of insect communication systems: they bind and transport pheromones and other semiochemicals, generally mediating their perception and sometimes their secretion (Pelosi *et al.*, 2006, 2014; Leal, 2013). Furthermore, tests of historical selection on *Gp-9* reveal a significant excess of nonsynonymous (amino acid replacing) substitutions relative to synonymous (silent) substitutions between the lineage of *Gp-9* b-like alleles and *Gp-9* B-like alleles in the fire ant and its relatives. This implies that directional or diversifying selection has driven the molecular evolution of *Gp-9* and is associated with differentiation between the two forms of social organization in these ants (Krieger and Ross, 2002, 2005). Several models lay out the potential function of *Gp-9*, generally involving differential production or perception of pheromones in queens as well as workers of alternate genotypes (Krieger, 2005; Gotzek and Ross, 2007, 2009).

However, recent genome-wide analyses of the social dimorphism revealed that the association between genotype and form of social organization is not limited to *Gp-9* (Wang *et al.*, 2013). Instead, genetic maps obtained using Restriction site Associated DNA (RAD) markers from crosses in seven families showed that this association extends over a large chromosomal region of suppressed recombination. The two variants of this region, respectively, marked by the *Gp-9B* and *Gp-9b* alleles are carried by a pair of "social chromosomes" named SB and Sb. The region is genetically differentiated over 10.8 Mb (55%) of the mapped assembly of the social chromosomes, although its total length could be 19.4–31.5 Mb given the estimated size of the non-assembled portion of the genome (Pracana, Priyam, *et al.*, 2017). Based on the current NCBI gene set, this region contains at least 443 protein coding genes, including *Gp-9*. The two chromosomes differ by at least one large inversion affecting a large portion of the region and an additional small (48 kb) inversion. The region of suppressed recombination can be described as a supergene, a locus containing multiple genes with tightly linked allelic combinations that control a complex polymorphic

phenotype (Linksvayer, Busch and Smith, 2013; Schwander, Libbrecht and Keller, 2014; Thompson and Jiggins, 2014).

A study of general patterns of divergence and diversity showed that Sb has two orders of magnitude lower diversity than SB and the rest of the genome, and that there is high ratio of nonsynonymous to synonymous substitutions between SB and Sb (Pracana, Priyam, *et al.*, 2017). These results suggest that the evolution of Sb has been shaped by Hill–Robertson effects (the effects of selection on linked loci) due to the rarity of recombination in Sb (Wang *et al.*, 2013; Pracana, Priyam, *et al.*, 2017). However, little work has been done to characterize the genes present in the supergene region and to identify the mechanisms by which SB and Sb control the phenotypic differences between single- and multiple-queen colonies. Studies using cDNA microarrays representing 3673 genes demonstrated that the supergene region is enriched for genes that are differentially expressed between queens (Nipitwattanaphon *et al.*, 2014) and workers (Wang, Ross and Keller, 2008; Wang *et al.*, 2013) of the two colony types. This suggests that genes other than *Gp-9* could be responsible for the social dimorphism. Given that the determination of queen number requires the differential production and perception of semiochemicals by individuals of each genotype, it remains likely that OBPs play a part in determining the dimorphism.

Here, we determine to which extent OBPs have potentially functional divergence between social forms. For this, we identify all OBPs in the fire ant reference genome and map them to their genomic locations. Subsequently, we use population-sequencing data to identify allelic differences between OBPs found on alternate variants of the social chromosome supergene. We also sequence an outgroup species, *Solenopsis geminata*, which allows us to determine which supergene variant carries the derived allele for each substitution. Finally, we compare gene expression profiles of all OBPs and gene coexpression modules between social forms. We show that there are nucleotide and amino acid sequence level differences between SB and Sb in the supergene OBPs, and that OBPs inside and outside the supergene are differentially expressed between single- and multiple-queen colonies.

Results

The fire ant reference genome assembly contains 23 putative OBPs

We combined automatic and manual curation approaches incorporating genomic and gene expression data to identify the sequence, exon structure, and location of 23 putative OBP genes in the *S. invicta* reference genome. Seventeen of these matched fire ant OBP gene sequences that had been previously reported, although with differences in sequence or in their inferred location in linkage groups (Table A2.S1 and Annex 3 Supplementary Methods). The remaining seven putative OBP genes are novel to *S. invicta* (Table A2.S3). Interestingly, the coverage depth of SiOBPZ6 is fourfold higher (95% confidence interval [3.66–4.78]; t-test $t_{df = 6} = 14.0$, $P < 10^{-5}$) than that of 1000 randomly selected genes, suggesting that there are four copies of this gene. There is little genetic variation among reads mapping to this gene across the 14 individuals in our dataset (4.2 Single Nucleotide Polymorphisms [SNPs] per 1000 bp). The alignment of whole- genome sequencing reads of the outgroup species *S. geminata* to the *S. invicta* reference assembly shows that all OBPs are covered in this outgroup species. The coverage depth of SiOBPZ6 is three- fold higher in *S. geminata* (95% confidence interval [2.78–3.16]; t-test $t_{df = 999} = 20.7$, $P < 10^{-15}$), suggesting that this species also carries multiple copies of this gene.

Nine of the 23 OBPs in the genome are adjacent to unrelated genes, the remainder are organized into gene clusters. There are three locations in the genome each containing a cluster of four OBPs (two in linkage group 16, one in linkage group 3) and one containing a cluster of two OBPs (in linkage group 6). Intriguingly, none of these clusters appear to be completely monophyletic (Figure 6.1). For previously known OBPs, the topology of our phylogenetic tree agrees with previously published trees (Gotzek *et al.*, 2011; Zhang *et al.*, 2016), with the exception of the position of SiOBP15 (low bootstrap values in all trees) and SiOBP5.

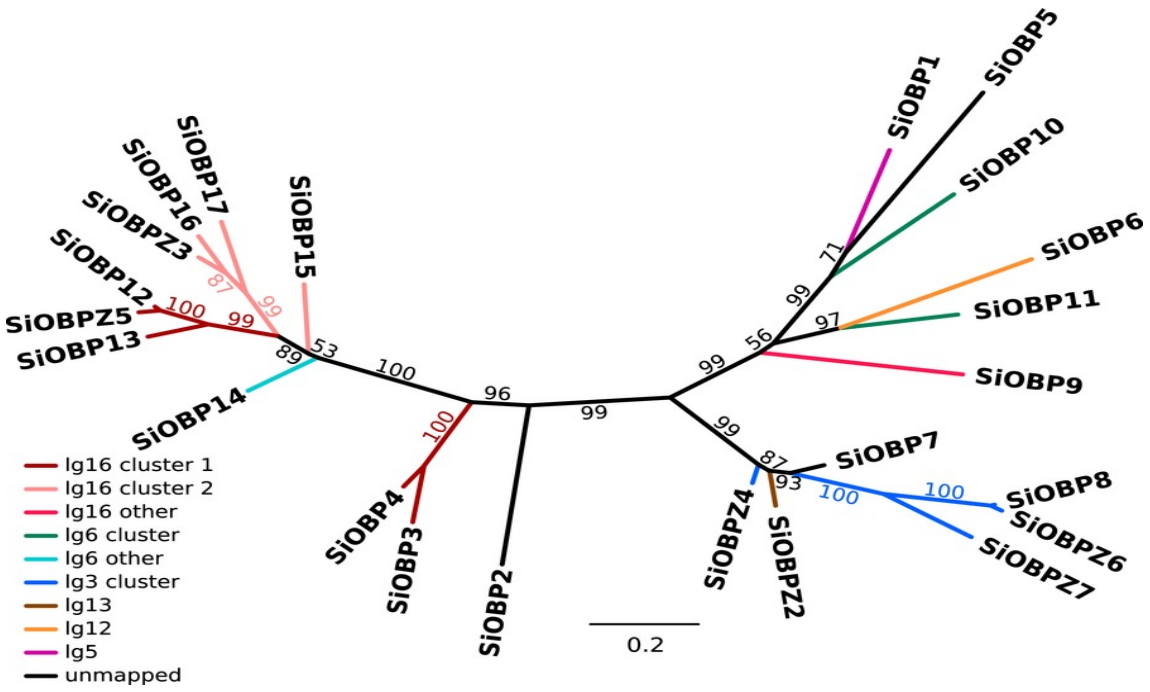


Figure 6.1: Phylogenetic tree of fire ant OBPs

Phylogenetic tree based on a codon-level alignment of revised gene predictions for previously described OBPs (*SiOBP1–17*) and novel OBPs (*SiOBPZ2–Z6*). Branches are colored by gene cluster and linkage group (lg). *SiOBPZ1* was removed from this analysis because the high divergence of its sequence led to unreliable alignments and positioning in the phylogeny. All OBPs on linkage group 16 (lg16) are within the supergene-like region of the social chromosomes (Figure 6.2).

Nonsynonymous differentiation between SB and Sb in OBPs

Eight of the OBPs are located in scaffolds of the SB fire ant genome assembly that map to the supergene region, with two clusters of four OBPs (Figure 6.2). One of the clusters includes Gp-9 (which was named SiOBP3 in (Gotzek *et al.*, 2011)). A ninth gene, SiOBP9, is located in an unmapped scaffold that likely also belongs to the supergene region based on high levels of SB-Sb differentiation (Figure 6.2). To determine whether the supergene OBPs have allelic differences between SB and Sb, we used whole-genome sequence data from seven SB males and seven Sb males.

These data confirmed the previous finding that Gp-9/SiOBP3 has eight nonsynonymous and one synonymous fixed single nucleotide substitutions between SB and Sb in the North American study population (Krieger and Ross, 2002). Of the other OBPs in the supergene region, SiOBP4 has three nonsynonymous and two synonymous substitutions. Two additional supergene OBPs have one fixed nonsynonymous substitution between SB and Sb (Table 6.1). Performing an analysis of the ratio of nonsynonymous to synonymous substitutions between alleles (dN/dS) was only possible for the two genes with the most divergent alleles: Gp-9/SiOBP3 had the highest ratio of nonsynonymous to synonymous substitutions (dN/dS = 1.48), followed by SiOBP4 (dN/dS = 0.74).

We analysed the OBP sequences from an outgroup species, *S. geminata*, estimated to have diverged from *S. invicta* 3–3.5 million years ago (Moreau and Bell, 2013; Ward, Sean G. Brady, *et al.*, 2015), that is, before the divergence between SB and Sb in *S. invicta* (estimated 0.35–0.42 million years ago (Wang *et al.*, 2013)). These sequences allowed us to determine the ancestral allele in each substitution. Sb carried the derived allele in most of the positions with nonsynonymous substitutions between SB and Sb (seven out of eight in Gp-9/SiOBP3 and all in SiOBP4 and SiOBPZ3; we could not derive the two SiOBP13 substitutions, as *S. geminata* read coverage was too low for this gene). This pattern is consistent with most nonsynonymous substitutions between SB and Sb having arisen in the lineage leading to Sb.

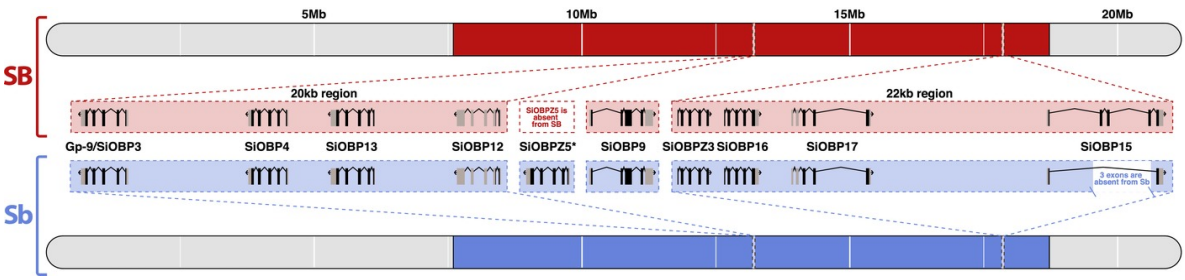


Figure 6.2. Position of the OBPs on the social chromosome

Relative positions on the social chromosome (i.e., linkage group 16) of 10 OBP loci, highlighting intron–exon structures and differences between the supergene region of Sb (blue) and SB (red). SiOBPZ5 is specific to Sb but we do not know its exact location; SiOBP15 is missing a 3-exon region in Sb; SiOBP9 is in an unmapped scaffold that likely belongs to the supergene region based on high levels of SB-Sb differentiation (Pracana et al. 2017).

Table 6.1: OBP differentiation between SB and Sb

The number of sequence-level differences between SB and Sb and differential OBP gene expression between multiple- and single-queen colonies.

S. invicta OBP locus	Nonsynonymous differences	Synonymous differences	Total differences	Significant differential expression between colonies types	
				In queens	In workers
SiOBP3 (*Gp-9*)	8	1	9	Yes	No
SiOBP4	3	2	5	Yes	Yes
SiOBP13	1	1	2	Yes	Yes
SiOBP12	Frameshift insertion in SB and duplication in Sb			Yes	No
SiOBPZ5	Present exclusively in Sb				
SiOBPZ3	1	0	1	No	No
SiOBP9	0	1	1	No	No
SiOBP16	0	0	0	Yes	No
SiOBP17	0	0	0	Yes	Yes
SiOBP15	~2600 bp deletion in Sb			No	No

All differentially expressed genes between social forms were overexpressed in multiple-queen colonies.

Copy number and structural differentiation between SB and Sb in OBPs

We also found structural differences between SB and Sb affecting two OBPs. For the first, SiOBP15, we detected a ~2600 bp deletion unique to Sb individuals (Figure 6.2, Table 6.1). This deletion is derived (i.e., it is not present in the outgroup species, *S. geminata*) and causes the loss of three out of five coding exons (89 out of 139 amino acids), although it does not cause a frameshift. The second OBP with a major structural difference is SiOBP12. In Sb individuals, this gene is duplicated, forming the Sb-specific SiOBPZ5 (Figure 6.2, Table 5.1). This gene increases the total OBP count of *S. invicta* to 24. There are 18 fixed amino acid differences between SiOBPZ5 and the SB allele of SiOBP12 sequence (one deleted codon, 21 nonsynonymous and four synonymous nucleotide-level fixed differences; four codons each contain two single-nucleotide fixed differences; dN/dS = 2.67). Intriguingly, SiOBP12 has an early stop codon (TAG) at codon position 16 of 176 in all seven SB individuals and the

reference genome. These individuals are also affected by six nonsynonymous SNPs and two polymorphic indels downstream of the early stop codon. Sb individuals have the CAG allele at position 16 of SiOBP12 but have a slightly later early stop codon at position 37 due to a frameshifting insertion of 17 bp at codon position 25 (nucleotide position 74). The outgroup species *S. geminata* has neither of the early stop codons. However, the very low *S. geminata* read coverage observed in the two terminal exons of this gene (median < 3; $t_{df=999} = -11.29$, $P < 10^{-27}$) could indicate a deletion in this species. SiOBP12 is thus non-functional in Sb and SB individuals, and putatively non-functional in the outgroup species. The Sb-specific gene SiOBPZ5 appears to be functional as it has no early stop codons. None of the other OBPs showed differences in structure or in copy number between SB or Sb.

Fourteen OBPs are differentially expressed between social forms

We compared expression levels between single- and multiple-queen colonies in workers and in queens (Figure 6.3; Table 5.1) using RNA-seq data from Morandin et al. (2016). General expression patterns showed an enrichment in differentially expressed genes in the supergene region in queens (expected proportion = 0.022, observed proportion = 0.059, $Chi^2_{df=1} = 32.84$, $P = 10^{-8}$) but not in workers (expected proportion = 0.021, observed proportion = 0.024, $Chi^2_{df=1} = 0.05$, $P = 0.82$).

In queens, fourteen OBPs, including seven in the supergene region, were significantly differentially expressed between multiple-queen and single-queen colonies (DESeq2 Wald test; Benjamini–Hochberg adjusted $P < 0.05$). Consistent with this, the entire group of 24 fire ant OBPs showed significantly stronger P-values for differential expression between queens from single- and multiple-queen colonies than would be expected by chance (tested among 12,693 transcripts, two-sided Kolmogorov–Smirnov test; $P < 10^{-11}$; Figure A2.S1). Surprisingly, all of the OBPs that were differentially expressed between social forms in queens were more highly expressed in multiple-queen colonies than in single-queen colonies (14 significant OBPs in queens, binomial test 14 out of 14; null probability = 0.5; $P < 10^{-4}$). In workers, only four OBPs (all in the supergene region) were significantly differentially expressed between social forms (DESeq2's Wald test Benjamini–Hochberg adjusted $P < 0.05$). All the differentially expressed OBPs in workers were also differentially expressed in queens. For

one of these OBPs (SiOBP17), a different splice form was differentially expressed between colony types in queens than in workers (Figure 6.3).

We additionally obtained qualitative gene expression profiles of all OBPs across 18 additional samples, in total representing seven different conditions of body part, social form, and caste (Table A2. 2). We find generally consistent expression patterns for OBPs across all independent samples (Figure 6.3). For instance, in every sample, Gp-9/SiOBP3 was the most highly expressed of all OBPs, whereas SiOBPZ3 was only residually expressed (0.26 or fewer transcripts per million reads). Six OBPs had only residual expression in queen antennae and in heads, although most of these showed at least some expression in whole-body samples. The expression of one of these genes, SiOBP9, appears to be limited to males.

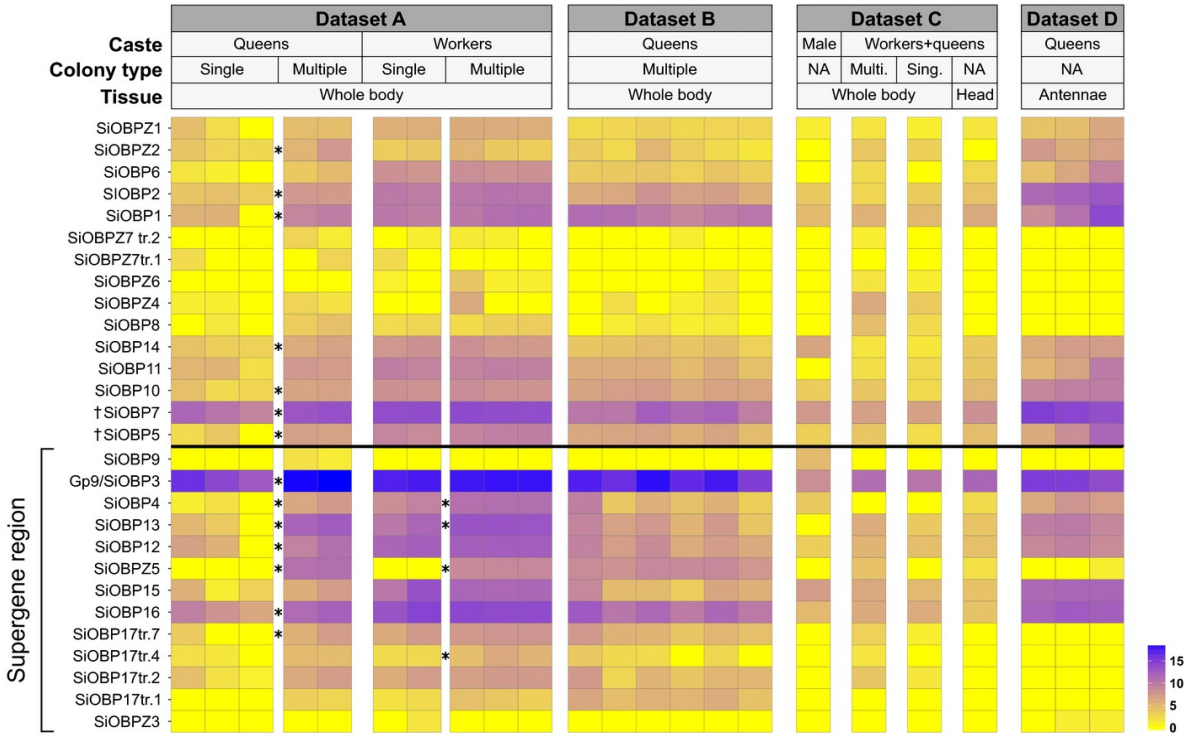


Figure 6.3. Expression patterns for all analysed RNA-seq datasets

Each tile represents the logarithm base 2 of DESeq normalized transcript counts. The rows with asterisks (*) correspond to those OBPs with significant differential expression between social forms within castes in dataset A (Morandin et al. 2016). Information about each dataset is available in Table A2.S2 (A: PRJDB4088, B and C: PRJNA49629, D: PRJNA266847). † The exons of SiOBP5 and SiOBP7 are split across three unmapped scaffolds; we do not know whether these genes are within or outside the supergene region.

Gene coexpression modules correlated with social form

We used WGCNA (Langfelder and Horvath, 2008) to produce modules of coexpressed genes from a set of worker samples (Wang, Ross and Keller, 2008) and a set of queen samples (Nipitwattanaphon *et al.*, 2013). Both datasets compare SB/SB and SB/Sb samples. The queen and worker datasets, respectively, clustered into 30 and 37 coexpression modules (Table A2.S4). Most modules in one dataset share a significant number of probes with a module in the other dataset (30 out of 31 in queens and 35 out of 37 in workers; Fisher's exact test for the overlap of the pairs of modules across the datasets, Bonferroni corrected $P > 0.05$; Figure A2.S2). However, in most cases there was no one-to-one correspondence between datasets (17 out of 31 modules in queens and 19 out of 37 in workers have significant overlaps with more than one module). Eight of the OBPs discovered in the present study are represented in the microarray (Table A2.S4). In the worker dataset, the module "worker_D" includes four of the OBPs (SiOBP3, SiOBP12, SiOBP13, and SiOBP16), accounting 25% of the 16 genes in the module. OBPs were present in nine other modules, although in all nine cases the OBP represented a very small proportion of the genes in the module (Table A2.S4).

We tested whether there were gene coexpression modules with differential eigengene expression between genotypes or social forms. In queens, four modules had differential expression between genotypes (Table A2.S6). In workers, one module had differential eigengene expression between genotypes, and one module had differential gene expression between social forms (Table A2.S6). One of the modules that had differential expression between genotypes in queens ("queen_X") corresponded with the module with differential expression between genotypes in workers ("worker_Z"). Only one of the modules with differential eigengene expression includes an OBP (SiOBP15 in "queen_D"). None of these modules were enriched for any GO term.

Three OBPs are in a region of the genome with characteristics of a recent selective sweep

We used measurements of π among SB individuals in nonoverlapping 10 kb windows from Pracana et al. (2017) to determine whether any OBPs are in regions of low π, characteristic of recent selective sweeps. Among windows overlapping OBPs, two neighbouring windows had π within the lower quartile of the whole-genome distribution ($\pi < 0.0004$; Figure A2.S3). These two windows overlap the loci SiOBPZ4, SiOBPZ7, and SiOBPZ6, which are within 19 kb of each other on linkage group 3. We did not perform an equivalent analysis on Sb individuals because the entire region of suppressed recombination has the signature of a recent sweep in Sb (Pracana, Priyam, *et al.*, 2017).

Discussion

The putative role of OBPs in determining social dimorphism

The description of Gp-9 as a green beard gene (Keller and Ross, 1998) and its subsequent characterization as an OBP (Krieger and Ross, 2002) led to the proposal of different models of how this single gene can control the dimorphism in social organization (reviewed by (Gotzek and Ross, 2007). At their most basic level, these models propose that Gp-9 controls the production of a green-beard odour in queens and the differential perception of this odour by workers of alternate genotypes. However, it was also proposed that Gp-9 additionally controls differential odour production in workers (Gotzek and Ross, 2007), as well as a number of physiological and morphological traits in queens (Keller and Ross, 1995; DeHeer, Goodisman and Ross, 1999; DeHeer, 2002) and males (Lawson, Vander Meer and Shoemaker, 2012). The discovery that Gp-9 is tightly linked to hundreds of other genes (Wang *et al.*, 2013; Pracana, Priyam, *et al.*, 2017)—including the nine additional OBPs we report here—suggests that the roles previously attributed to Gp-9 could be split between multiple genes.

The key roles of OBPs in semiochemical perception (Leal, 2013) and secretion (Li *et al.*, 2008; Iovinella *et al.*, 2011; Sun *et al.*, 2012) lead to the prediction that such proteins are involved in determining the two colony types. Our results support this hypothesis, as we find divergence in protein coding sequence between SB and Sb in the OBPs in the supergene region, as well as differences in the regulation of OBP expression between single- and multiple-queen colonies.

The differences in protein coding sequence affect seven of the ten OBPs in the supergene region, including Gp-9/SiOBP3. The biggest differences are in SiOBPZ5, absent in SB, and in SiOBP15, which is missing three exons in Sb. Such differences could have a major effect on semiochemical communication. Additionally, among the four intact OBPs with nonsynonymous divergence between SB and Sb, both Gp-9/SiOBP3 and SiOBP4 have dN/dS ratios indicative of adaptive differentiation between the alleles of these genes (Krieger and Ross 2002). This interpretation comes with some caution due to our relatively low sample size (14 individuals from an invasive population).

Additionally, 14 out of the 24 fire-ant OBPs were differentially expressed between social forms in queens or in workers. Our analysis uncovers three potentially important aspects of the differential regulation of OBP expression in the two social forms. First, all of the differentially expressed OBPs are more highly expressed in multiple-queen colonies than in single-queen colonies, suggesting that multiple-queen colony traits are associated with the activation of semiochemical communication pathways. Second, this activation seems to be stronger in queens, as more OBPs were differentially expressed between social forms in queens (14 OBPs) than in workers (four OBPs). This result reflects the more general pattern that the supergene region was enriched for differentially expressed genes between colony types in queens, but not in workers. The pools of workers from multiple-queen colonies contain a mix of individuals of both genotypes (36% SB/SB and 64% SB/Sb workers expected (Buechel, Wurm and Keller, 2014)), which could mask differences between SB/SB workers from single-queen colonies and SB/Sb workers from multiple-queen colonies. Indeed, previous studies using cDNA microarray data and a different gene set suggest that the supergene region is enriched for differentially expressed genes in both queens (Nipitwattanaphon *et al.*, 2013) and workers (Wang *et al.*, 2013). Third, several of the queen-

specific differentially expressed OBPs are located outside the supergene, implying that they are regulated in trans by elements in the supergene. It is important to note that all three patterns could be affected by our use of samples from whole bodies, which is known to introduce several types of biases if the differences in expression are tissue specific (Johnson, Atallah and Plachetzki, 2013; Montgomery and Mank, 2016). A particular issue is differences in allometry (i.e., relative body-size proportions) between the individuals of different groups, for instance the larger gaster of queens in single-queen colonies relative to queens in multiple-queen colonies (Tschinkel, 2006). These biases cannot be resolved by standard normalization methods, which are designed to normalize by entire library size rather than by the relative abundance of different transcripts (Dillies *et al.*, 2013). Tissue-specific gene expression profiling (Bastian *et al.*, 2008; Robinson *et al.*, 2013; Jasper *et al.*, 2016) would be needed to control for such allometric differences.

Our results also support the idea that along with OBPs, other genes are likely involved in defining the social polymorphism of *S. invicta*. For instance, only one of the coexpression modules with significantly different eigengene expression contained an OBP. Furthermore, other genes inside and outside the supergene region were differentially expressed between social forms. Thus, a venue of further investigation would be to examine the potential roles of other genes, including genes from families known to be involved in communication, including chemosensory proteins (Kulmuni, Wurm and Pamilo, 2013), desaturases (Helmkampf, Cash and Gadau, 2015), fatty-acid reductases (Lassance *et al.*, 2010; Niehuis *et al.*, 2013), and olfactory (Wurm *et al.*, 2011), gustatory (Robertson, Warr and Carlson, 2003; Zhou *et al.*, 2012), and ionotropic receptors (Benton *et al.*, 2009; Zhou *et al.*, 2012). It is important to note that additional experimental work would be necessary to demonstrate whether OBPs or any of these proteins have a functional role. An interesting approach would be to measure the effect of artificially modifying the sequence or expression level of each gene to test their specific function (Gaj, Gersbach and Barbas, 2013; Mohr *et al.*, 2014).

General evolutionary patterns of OBPs in *S. invicta*

The evolution of the OBP gene family is generally thought to follow the birth-and-death model, where gene duplication is followed by either the pseudogenization or the rapid

functional divergence of the duplicate gene (Nei and Rooney, 2005; Vieira, Sánchez-Gracia and Rozas, 2007). The *S. invicta* OBPs are organized in clusters along the genome, as in other insect species (Xu, Zwiebel and Smith, 2003; Forêt and Maleszka, 2006; Vieira, Sánchez-Gracia and Rozas, 2007). However, none of these clusters appear to be monophyletic (Figure 6.1). This is consistent with the birth– death model, where the fast evolution of genes can mask their true phylogenetic relationship (Vieira, Sánchez-Gracia and Rozas, 2007; Gotzek *et al.*, 2011; Vieira and Rozas, 2011). Alternative explanations include translocations affecting the OBPs during or after duplication, or ectopic gene conversion across different clusters after duplication (Arguello and Connallon, 2011). Another argument in support of the birth-and-death model is that we find evidence of expansions in OBP number. One example is the putative ant specific OBP expansion reported previously (the OBP cluster including SiOBP14 in Figure 6.1 (Gotzek *et al.*, 2011)). We found no one-to-one orthologous sequences for these genes in other ants or in other arthropods (the 11 genes in this group of OBPs have BLAST similarity to only three genes in the ant *Monomorium pharaonis*; phylogenetic group 1 in Table A2.S6). A cluster with several novel genes identified in our study (the group including SiOBP7 and SiOBP8 in Figure 6.1) follows a similar pattern (five OBPs have BLAST similarity to one *M. pharaonis* gene, two have BLAST similarity to one *Pogonomyrmex barbatus* gene; phylogenetic group 2 in Table A2.S6). These groups of genes may have expanded in the lineage leading to *S. invicta* and *S. geminata*, although this conclusion would require the exhaustive identification of OBPs in the present study to be replicated for other ant species. An example of a putatively recent expansion is SiOBPZ6, which seems to be present in multiple copies both in *S. invicta* and in *S. geminata*. Lack of heterozygosity in the region suggests that the gene copies have been recently affected by ectopic gene conversion (Arguello and Connallon, 2011). Furthermore, finding that the *S. invicta* SiOBPZ6 quadruplication is in a region that has a signature of a recent selective sweep makes it tempting to speculate that SiOBPZ6 is involved in a recent adaptive process (Kondrashov, 2012)—for example, to the invasive range of this species (Ascunce *et al.*, 2011).

Conclusion

Previous studies have focused on how the evolution of the social chromosomes has been affected by restricted recombination (Wang *et al.*, 2013; Pracana, Priyam, *et al.*, 2017), whereas the work presented here focuses on the putative mechanisms by which these chromosomes control social organization. In summary, our analyses provide a comprehensive overview of OBPs in the fire ant genome, describing patterns of differentiation and expression that are consistent with the predicted roles of OBPs in determining social organization in this species. Our study highlights the need for tissue-specific expression profiles, as well as for broader taxonomic sampling to understand OBP evolution during the origin of the multiple-queen colony organization. Finally, our work provides a starting point for future functional studies on the roles of OBPs in the social chromosome system.

Methods

OBP discovery and manual gene model curation

The sequences of 18 fire ant OBP genes were previously reported, based on searches of Sanger-sequenced Expressed Sequence Tag (EST) libraries (Table A2.S1; (Xu, Zwiebel and Smith, 2003; Wang *et al.*, 2007; Gotzek *et al.*, 2011; Wurm *et al.*, 2011). We used a curation approach similar to those previously used on other genes (Ingram *et al.*, 2012; Corona *et al.*, 2013; Kulmuni, Wurm and Pamilo, 2013; Privman, Wurm and Keller, 2013) to find the position of these OBP genes in the fire ant genome assembly (Wurm *et al.*, 2011) and to discover previously unreported OBP genes. Our curation pipeline is described in detail in Annex 3 Supplementary Methods. Briefly, we iteratively per- formed blastn and blastp (Camacho *et al.*, 2009; Priyam *et al.*, 2019) searches of the fire ant genome assembly (Wurm et al. 2011) using as queries the previously known fire ant OBP sequences as well as UniProt sequences that are part of the Pfam family "PBP_GOBP" (Finn *et al.*, 2014; The Uniprot Consortium, 2015). We manually curated the results of these searches by inspecting alignments of transcriptomic and genomic reads, which allowed us to infer intron–exon

boundaries and coding sequences of these OBPs. We labelled the curated gene predictions that correspond to the previously known OBP genes (SiOBP1–17) according to the notation used by Gotzek et al. (2011) and we labelled newly discovered loci SiOBPZ1–Z7. We used a genetic map (Pracana, Priyam, *et al.*, 2017) to assign OBPs to linkage groups. We generated a codon-level alignment of the *S. invicta* OBPs using MAFFT-linsi (version 6.903b (Katoh and Toh, 2008)) and PRANK (version 120626 (Löytynoja and Goldman, 2005)), and built a phylogenetic tree using RaxML (version 8.2.9 (Stamatakis, 2006)).

Identifying allelic differences for OBPs carried by alternate variants of the social chromosome

We used whole-genome sequences from one SB and one Sb male from each of seven colonies that had been sequenced at low coverage (Illumina 2*100 bp paired-end genome shotgun sequences; ~6×–8× coverage) in 2012 (NCBI SRP017317 (Wang *et al.*, 2013)). Each of these samples is a haploid male (ants have a haplodiploid sex determination system). We filtered the reads, aligned them to the reference genome using bowtie2 (version 2.1.0 (Langmead and Salzberg, 2012)), and used samtools and bcftools (version 1.3.1 for both (Li *et al.*, 2009)) to call variants among the individuals (Annex 3 Supplementary Methods).

We produced whole-genome sequencing reads of the out- group species *S. geminata*. We sequenced a pool of 10 workers (sampled in Thailand by Dr. Adam Devenish, University College London, United Kingdom) using Illumina HiSeq 4000 (11× coverage; Annex 3 Supplementary Methods). We called variants between the sample and the reference assembly (using freebayes version 1.0.2-33-gdbb6160 (Garrison and Marth, 2012)) within the coding sequence of each OBP using freebayes (Annex 3 Supplementary Methods). We classed the alleles in each SB-Sb substitution as ancestral or derived based on the allele carried in the outgroup species. We estimated the rate of synonymous and non-synonymous divergence (dS and dN, respectively) between SB and Sb using seqinR (version 3.0-7 (Charif *et al.*, 2007)).

Detection of copy number and structural variation in OBPs

We visually inspected the alignments of the seven SB and the seven Sb haploid male samples against each OBP region. Deletions were identified as regions with no coverage and duplications were identified as regions where the coverage was higher than the background (Annex 3 Supplementary Methods). Using the *de novo* assembler MIRA (version 4.0.2 (B. Chevreux, Wetter and Suhai, 1999)), we produced the sequence of the duplicate copy of SiOBP12, which we named SiOBPZ5 (approach detailed in Annex 3 Supplementary Methods).

Gene expression of *S. invicta* OBPs in publicly available RNA sequencing datasets

We analysed all available RNA sequencing (RNA-seq) data from the NCBI SRA database for *S. invicta* (data from Wurm et al. 2011; Morandin et al. 2016 and PRJNA266847; details in Table A2.S2). We determined the expression levels of *S. invicta* transcripts using the Kallisto count mode (version 0.43.0 (Bray *et al.*, 2016a)). Each sample was independently normalized using the DESeq2 method (version 1.14.1 (Love, Huber and Anders, 2014)). Additionally, we performed genome-wide analysis of differential expression of data from Morandin et al. (Morandin *et al.*, 2016), comparing three pools of queens from multiple-queen colonies with two pools from single-queen colonies, as well as two pools of workers from multiple-queen colonies with three pools from single-queen colonies. The pools of workers from multiple-queen colonies contain a mix of individuals of both genotypes, whereas the pool of queens from multiple-queen colonies has only SB/SB queens. We used a standard DESeq2 approach to identify expression differences between single- and multiple-queen samples in queens and in workers. Additional details regarding these analyses are in Annex 3 Supplementary Methods.

Differential expression of gene coexpression modules across social forms

We created gene coexpression modules from two cDNA microarray datasets (Platform GPL6930, with 25,344 probes representing 3673 genes; Annex 3 Supplementary Methods; Wang et al. 2007), one with queen samples (GSE42062 (Nipitwattanaphon *et al.*, 2013)), the other with worker samples (E-GEOD-11694 (Wang, Ross and Keller, 2008)). Both datasets included SB/SB and SB/Sb samples. We created modules for each set using weighted gene coexpression network analysis (WGCNA) (version 1.49 (Langfelder and Horvath, 2008)). We used t-tests to determine whether any module eigengene is correlated with genotype or social form. In queens, we compared SB/SB to SB/Sb samples because all samples originate in multiple-queen colonies. In workers, we separated the effect of genotype from the effect of social form following the approach in Wang et al. (2008): we compared genotypes (SB/SB vs SB/Sb) using samples from multiple-queen colonies, and we compared across social forms (single queen vs multiple queen) using SB/SB samples only.

Evidence for selection based on nucleotide diversity

Genomic regions that underwent recent selective sweeps are characterized by low nucleotide diversity (π) (Smith and Haigh, 1974; Nei, 1987; Nachman, 2001). We used measurements of π along a sliding window of the genome, originally produced by Pracana et al. (2017), to identify selection pressure acting on *S. invicta* OBPs. Measurements of π were taken from nonoverlapping 10 kb windows (Annex 3 Supplementary Methods).

Data availability

This analysis relies on the following data:

- Illumina sequences from 15 fire ant males: NCBI SAMN00014755
- Fire ant reference genome assembly: GCA_000188075.1.

We deposited the genomic reads of the *Solenopsis geminata* sample on NCBI SRA (SRX3045159). We manually produced gene models for 24 OBPs, which we deposited to NCBI. Additionally, all data is available at wurmlab.github.io/data.

Acknowledgements

We thank K. G. Ross, R. A. Nichols, C. Eizaguirre, L. Henry, E. Favreau, T. Colgan, two anonymous reviewers, the editor and the associate editor for advice and comments on the manuscript, and QMUL's SBCS Evolution group for support and stimulating discussion. We thank A. Devenish for supplying *Solenopsis geminata* samples. This work was supported by the Biotechnology and Biological Sciences Research Council (grant BB/K004204/1), the Natural Environment Research Council (grant NE/L00626X/1), NERC EOS Cloud, the Deutscher Akademischer Austauschdienst (DAAD) Postdoc-Programm (570704 83), Marie Curie Actions (PIEF-GA-2013-623713), and QMUL Research-IT and MidPlus computational facilities (The Engineering and Physical Sciences Research Council grant EP/K000128/1).

Chapter 7: No supergene despite social polymorphism in the big-headed ant *Pheidole pallidula*

Contributions

Emeline Favreau led the study under supervision of Yannick Wurm. She did most of the work, including in the field work, laboratory experiments, data analysis and drafting the manuscript. I generated the genome assembly that forms the basis of all analysis, provided considerable input in determining potential reference bias in the analyses, and in revising the draft manuscript and interpreting the results.

The chapter is intended for submission to Nature Communications:

E Favreau, C Lebas, E Stolle, A Priyam, R Pracana, S Aron, Y Wurm (in prep)

Abstract

Phenotypic polymorphisms that are maintained over time within a population are sometimes associated with genetic structures that hinder unfavourable allele recombination, such as supergenes. Recent studies in socially polymorphic ant species have demonstrated that large supergene regions of suppressed recombination are responsible for determining alternate forms of social organisation in at least two distinct lineages. Such findings suggest that supergenes may be required for maintaining social polymorphism, in line with the theory that such regions can resolve conflict. To test this, we focus on an independent lineage, the Mediterranean big-headed ant *Pheidole pallidula*, in which single- and multiple-queen colonies co-occur in the same population. We perform extensive genomic comparisons and show that a large supergene region does not underpin social polymorphism in this system. Our work highlights that even complex social polymorphisms can be maintained by other mechanisms.

Introduction

Evolutionary success within variable environments favours complex phenotypes, involving an important suite of characters and associated genes (Stearns, 2010; Thompson and Jiggins, 2014). Selecting for genetic diversity (*e.g.*, against inbreeding, for local adaptation) while maintaining allelic combinations associated with those complex phenotypes can be controlled by genetic architectures altering recombination levels. Supergenes, loosely defined as two or more tight linked loci associated with alternating complex phenotypes (Thompson and Jiggins, 2014), combine favourable alleles by hindering recombination, resulting in populations with balanced phenotypic polymorphisms. Supergenes are at the basis of sex chromosomes, plant mechanisms preventing self-fertilisation (Mather, 1950), butterfly's Müllerian mimicry (Joron *et al.*, 2011), and more recently they have been detected in social organisation of ants (Wang *et al.*, 2013; Purcell *et al.*, 2014).

Ants ancestrally have one queen per colony, yet many ant species have transitioned to having exclusively multiple-queen colonies (Hughes, Ratnieks and Oldroyd, 2008). A smaller number of species exhibit both social forms, which is called social polymorphism (Boulay *et al.*, 2014). In the socially polymorphic species, there are different benefits associated with the number of queens, depending on relatedness, phenotypic differences and local competition for instance (Bourke and Franks, 1995). Generally, multiple-queen colonies benefit from more worker resources (sharing brood care, foraging, fighting local competition (Hölldobler and Wilson, 1977)); in contrast, single-queen colonies benefit from high relatedness of offspring (Hölldobler and Wilson, 1977). In some species the two colony types are associated with different dispersal strategies (Boulay *et al.*, 2014).

The genetic basis of social polymorphism has been investigated in two socially polymorphic lineages, including the red fire ant *Solenopsis invicta* and the silver alpine ant *Formica selysi* (Wang *et al.*, 2013; Purcell *et al.*, 2014; Brelsford *et al.*, 2020). These two lineages are distantly related (119 million years (My) of divergence (Blanchard and Moreau, 2017)). However, in both the lineages, chromosomal inversions have led to the formation of a large region of the genome (several megabases (Mb)) where recombination is suppressed in the heterozygous state. The resulting supergene is associated with social form, in that all queens in multiple-queen colonies bear non-recombining alleles at this locus. In the single-queen genotype, all diploids are homozygous for the region and the supergene recombines. This suggests that supergene architecture may be required for maintenance of intra-specific variation of ant social organisation (Rubenstein *et al.*, 2019).

While it is currently impossible to validate the presence of a social supergene in every single socially polymorphic ant species, we sought to investigate the presence or absence of a social supergene in one other socially polymorphic lineage, the Mediterranean big-headed ant *Pheidole pallidula*. This species is distantly related to Solenopsis and Formica (> 100 My) yet presents the same social polymorphism (Aron *et al.*, 1999). If *P. pallidula* contains a supergene architecture associated with social form, we expect to find chromosomally linked variants that are associated with the phenotype. We further expect to find evidence of degeneracy in the non-recombining allele.

Results

To determine whether a supergene is associated with social polymorphism in *P. pallidula* we first constructed a reference genome assembly. We then mapped sequenced reads (whole-genome) from individuals of single- and multiple-queen colonies to the reference genome to identify single-nucleotide polymorphisms (SNPs) associated with social forms. Finally, we used simulations to demonstrate the validity of our analyses.

Reference genome for *Pheidole pallidula*

We constructed a *de novo* assembly of the genome of *P. pallidula* from 17x coverage of Oxford Nanopore long reads and 93x coverage of pairs of Illumina reads (150 bp). The resulting assembly (*Ppal_gnE*) has a length of 287 Mb, with an N50 length of 588 kb. Out of the 1,658 single-copy orthologous genes in the BUSCO insect reference dataset, 98.8% are present and complete, while only 0.7% are duplicated. The genome assembly length is reasonably close to the flow cytometry-based estimate for the genome size of another Pheidole species (326 Mb (Tsutsui *et al.*, 2008)). With 4,130 assembled contigs, this assembly is more fragmented than the ten expected chromosomes (Lorite and Palomeque, 2010). Nevertheless, our assembly's contig N50 is ranked in the top decile of 137 Hymenoptera genome assemblies (Table A3.S1).

No evidence of social supergene in genome-wide SNP survey

We collected workers from 108 colonies across three populations in France, Italy and Spain (Figure 7.1a), and determined the social form of each colony by genotyping six polymorphic microsatellite loci (average allele number per locus = 27.5, Table A3.S2) in eight workers per colony. We identified a total of 37 single-queen colonies and 71 multiple-queen colonies (Table A3.S3). We sequenced the genome of one worker from each colony using Illumina technology and obtained 6x coverage for each sample or 1380x coverage in total. We used bowtie2 (Langmead and Salzberg, 2012) to map the Illumina reads to the reference genome and freebayes (Garrison and Marth, 2012) for variant calling. We selected 121,786 biallelic

SNPs (one every 2,463 bp on average) that were polymorphic within each population and present in at least 75% of the samples.

We performed a Principal Component Analysis (PCA) of the selected SNPs. The first two principal components each explained less than 4% of the variation. Furthermore, the samples did not cluster by social form (Figure 7.1b). Instead, the first principal component splits Vigliano samples from the other two populations. Further exploration of principal components did not reveal social clusters either (Figure A3.S1). This suggests that the social forms are not strongly differentiated at the genomic level. This is further confirmed by consistently low fixation index (F_{ST}) in each of the population (< 0.25 using 10kb slide and 30kb window sizes; Figure 7.1e). In comparison, the average F_{ST} in *Solenopsis* and *Formica* supergenes is 0.9 (Pracana, Priyam, *et al.*, 2017; Brelsford *et al.*, 2020).

Next, we performed a genome-wide association test for social organisation. Out of the 121,786 SNPs, we found that 46 were significantly associated with social form (Fisher's exact test P_{adj} < 0.05, Bonferroni correction, Figure 7.1c). However, the SNPs were located on 42 different contigs, both large and small. Furthermore, none of the contigs had a strong association in both the populations (Figure 7.1d). Instead, it seems each population carries a unique, weak association with social organisation. Finally, only two of these contigs showed similarity to the *Solenopsis* supergene, the closest socially polymorphic lineage (Table A3.S4).

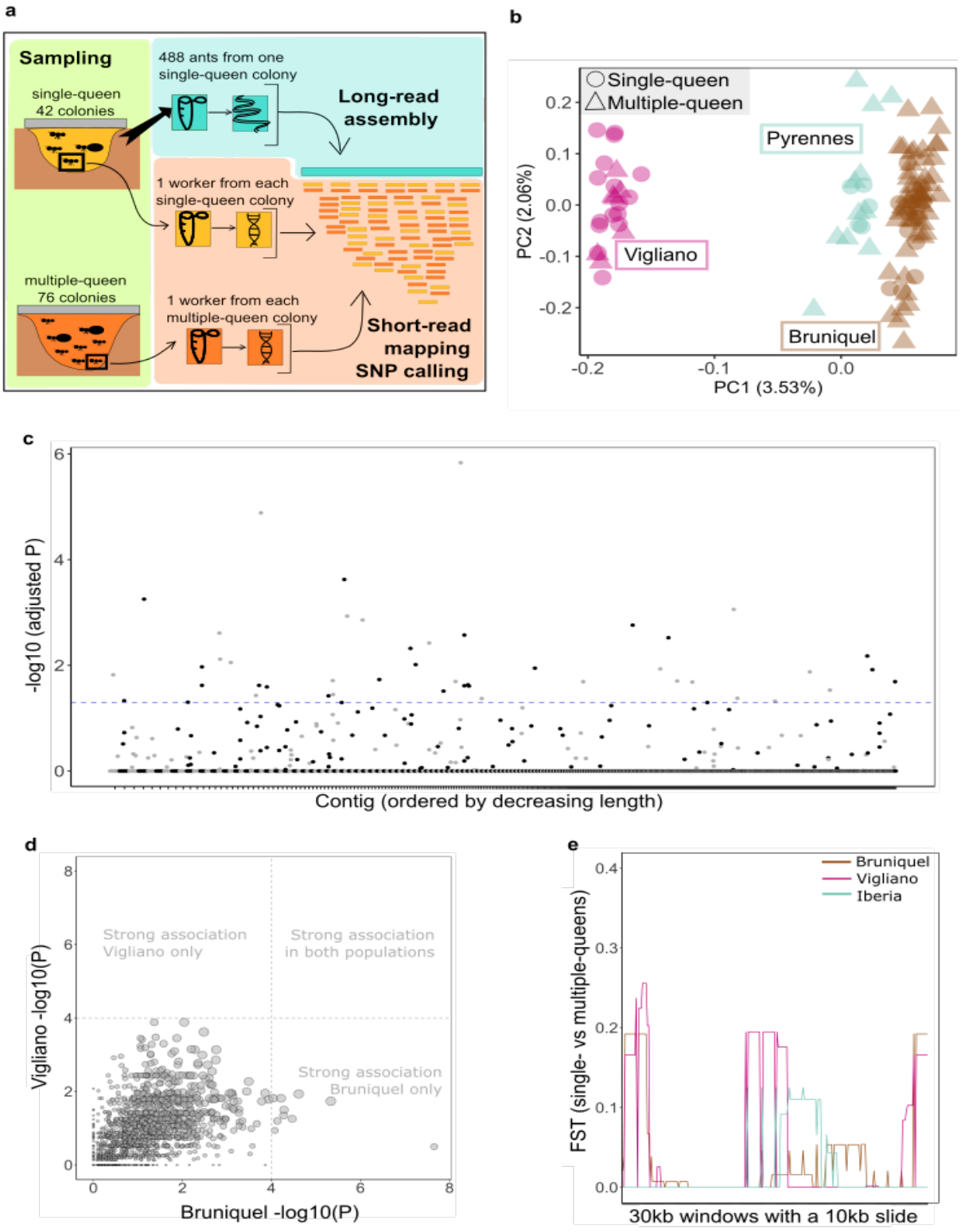


Figure 7.1: SNPs associated with social type are not linked

a) Experimental design: 108 sequenced workers, one per colony, originating from 3 populations containing

both single-queen and multiple-queen colonies. Short-reads were mapped to long-read assembly, followed by SNP calling.

b) PC1 and PC2 derived from the whole genomic dataset (121,786 within-population polymorphic SNPs, supported by 75% of samples, analysis from variance-standardised relationship matrix), explaining just over 5% of the total variance. Samples group by geographical locations.

c) Manhattan plot for association test across the whole dataset (121,786 Fisher's exact test *P* values, Bonferroni adjustment). The 2,555 contigs are ordered by length. There are 46 SNPs that are significantly associated with social form (*P* = 0.05, represented by the dashed line). The colour of the SNPs alternate based on their contig membership.

d) No common contigs yielding stronger within-population association signals (Pearson's correlation R = 0.443). 1,748 contigs containing SNPs within each population are represented by the lowest *P* value from each population (Fisher's exact test on SNP data, raw *P* value). The size of the circle radius equals the product of the *P* values from the association test in each population.

e) F_{ST} between social forms within each population, in an overlapping sliding window analysis.

Simulations demonstrate sufficient power to detect social supergene

We investigated if our association tests based on SNPs from one single worker per colony was powerful enough to detect an association between phenotype and genotype. We first simulated one single-nucleotide variant (homozygote in all single-queen samples, heterozygote in all multiple-queen samples) which we added to the *Pheidole* dataset with 121,786 SNPs. The simulated SNP was by far the most significant after multiple comparison adjustments (Figure A3.S2).

We then investigated if our association tests were powerful enough to detect an association with a fixed allele frequency for a simulated supergene region. We based this simulation on known systems by adding to our original dataset a realistic number of SNPs modelled on the *Solenopsis* supergene. 2.5% of the simulated SNPs were fixed for social form in the supergene: homozygote in all single-queen and a third of multiple-queen samples, heterozygote in two-thirds of multiple-queen (Buechel, Wurm and Keller, 2014; Pracana, Priyam, *et al.*, 2017). We randomly assigned a missing genotype to 25% of our samples, reflecting our original dataset. All simulated SNPs (n = 3,054) were detected by our method, with a *P* value lower than the real SNPs (Figure A3.S3). We additionally simulated the

Formica supergene: 3.6% of SNPs fixed for social form, homozygote in all single-queen samples, heterozygote for 68% of multiple-queen samples, homozygote alternative for 32% of the multiple-queen (Purcell *et al.*, 2014). Again, all simulated SNPs were detected, with a *P* value even lower than in the *Solenopsis* simulation (Figure A3.S4).

Finally, we investigated the effect of misgenotyping some of the samples on our analysis. We expect that the strength of association is so important in our dataset that the 46 significant SNPs will be detected by Fisher's exact tests, even when the categorical values (social type of the colony: single-queen or multiple-queen) is wrongly assigned. We set the alternative social type to 10% of our samples and investigated the subsequent association tests. We find that a large proportion of the 46 real significant SNPs are always recovered as significant in the simulations (Figure A3.S5).

Absence of coverage discrepancies underlying social supergene

We investigated if the genomic region potentially associated with social type may be missing in the reference genome. Since the reference assembly is derived from individuals of single-queen colonies, there are two possibilities. First, the region is present only in the individuals of multiple-queen colony. Second, the region is carried by individuals of both social forms but collapsed in the reference assembly.

We thus investigated the proportion and sequence similarity of unmapped reads (Table A3.S5), proportion and content of regions unique to each social type. We find that the regions unique to one social type are small and without significant sequence similarity to known ant social features (Table A3.S5). We also find that the contigs that are extremely enriched in either single-queen colony reads (top y axis in Figure 7.2) or multiple-queen colony reads (bottom y axis) are small and without significant SNP associated with social type. These coverage discrepancies investigations show that it is unlikely that the genome contains a region associated with social type.

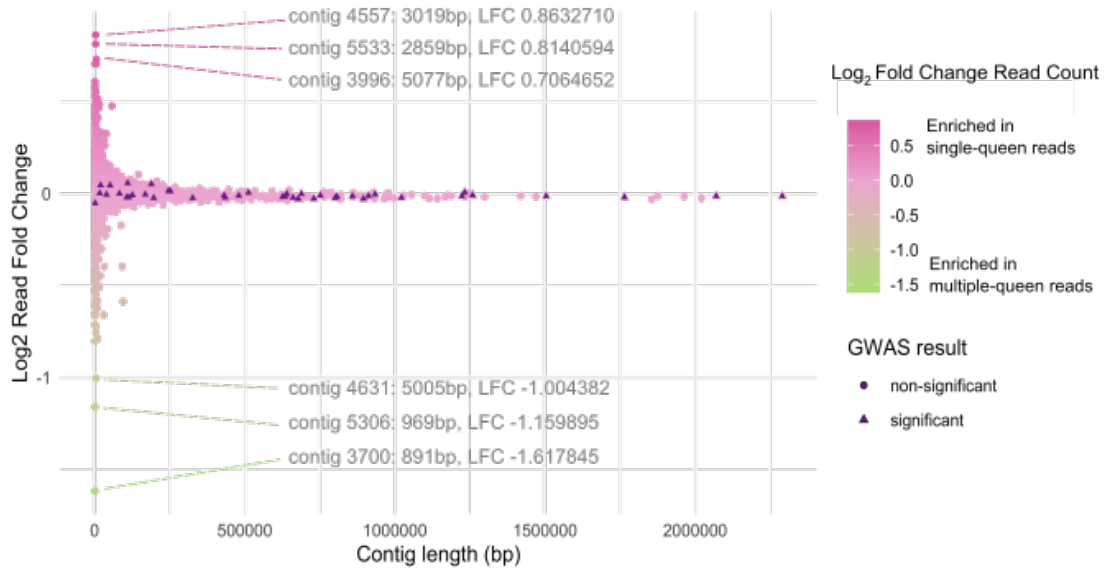


Figure 7.2: Contigs with biased coverage are small

Each contig is a round point, ordered by length on the x axis. Log_2 read fold change is based on the median-normalised mean read depth of 777,165 Bruniquel unfiltered SNPs. Contigs that are enriched in either single-queen colony reads (pink, top y axis) or multiple-queen colony reads (green, bottom y axis) are very small (left x axis) and without significant SNP (If the contig contains significant SNPs from the GWAS, the contig is represented by a triangle).

Discussion

We investigated whether the social polymorphism of *P. pallidula* is associated with a supergene. We genotyped 108 colonies for their social form and sequenced short reads from a representative worker of each colony. We assembled the genome *de novo* and tested for association with social form. In the whole dataset, we find 949 SNPs associated with social form, but none in genomic proximity to one another. We demonstrate that those loci are not significantly associated with social form within a population, and that each population has unique high FST regions. Furthermore, our datasets are complete, as there is no reference bias that could have supported the evidence of an absent (non-mapped) Y-like region. Our sampling effort led to a relatively small (n = 108) and heterogeneous dataset (each population has a skew towards one social form). Our experimental design included a PCR-based library

(non-random amplification) and a relatively low coverage per sample (6x). Yet, our simulation models show that the analysis would have detected an association as small as one significant locus.

The lack of social supergene brings into question the evolutionary origin of the social polymorphism in *P. pallidula*. There are many factors, other than a large non-recombining region, that could be at the origin of this polymorphism. First, life history, such as natural succession, can see a multiple-queen colony become single-queen as it matures (Hölldobler and Wilson, 1977). Second, other abiotic factors could be associated, such as phenotypic plasticity linked to environmental predictability, like in the system of facultatively sexual rotifer *Brachionus plicatilis* (Franch-Gras *et al.*, 2018). Third, it could be one (or several) very young supergene(s) in which two or more multiple-queen alleles are linked without signals of divergence. Alternatively, we could expect unlinked but associated polymorphisms to signal the early stages of a supergene, such as the *LaLal2* paralog that is involved in the self-incompatibility mechanism but is unlinked to other heterostyly genes in *Arabidopsis lyrata* (Thompson and Jiggins, 2014). Fourth, a social supergene could have evolved in the genome, altered by gene flow from subsequent population divergence and/or selection pressures from mutation load, such as in a hot potato scenario (Jeffries *et al.*, 2018). There, the supergene accumulates mutations in the non-recombining allele to the extent that the focal genes move to another chromosome (turnover). Fifth, social form could still have a genetic basis at a polygenic threshold trait, such as human red hair (Hysi *et al.*, 2018). There, each locus would have a very small effect that our analyses would not have detected. Sixth, social forms could vary depending on a threshold of allele frequencies, such as in the wing dimorphism of the sand cricket (Roff, Stirling and Fairbairn, 1997), which our dataset could not have detected. Finally, the social polymorphism of *P. pallidula* might be molecularly associated at another level of selection pressure (e.g., with differentiated gene expression, methylation or proteomic patterns). Further work, ideally combining long-term field observation and complete dataset analyses, will produce a more extensive survey of the potential mechanisms at play here.

The fact that there is no social supergene in *P. pallidula* contrasts with the social supergenes detected in two socially polymorphic ant species, *S. invicta* and *F. selysi* (Wang *et al.*, 2013;

Purcell *et al.*, 2014). It seems that the social polymorphism of *P. pallidula* is less bimodal than in these two lineages, in which a chromosomal inversion stops recombination at a specific locus that is associated with the social form and corresponding phenotypic characteristics. Cases of inversion associated with phenotypic characteristics are quite common, such as speciation in *Drosophila* species (Noor *et al.*, 2001). In the well-studied case of the colour polymorphism in *Heliconius* butterflies, the genus contains only one known species with an inversion associated with colour polymorphism (Joron *et al.*, 2011), while the rest of the species with colour polymorphism hold a polygenic system associated with colour (Nadeau, 2016). We hypothesize that the evolution of social organisation in ants follows a similar pattern in which there are many ways to evolve into a socially polymorphic system, in the ecologically diverse, 13,000-species strong genus. Moreover, the two examples of social chromosomes are tinted with local adaptation accents: *S. invicta* has more variability in the native range than in the North America population, *F. selysi* encompasses variation due to assortative mating and maternal effect (Avril *et al.*, 2019, 2020). A large-scale comparative analysis of social polymorphisms among the ants will give us a better idea of the mechanisms underlying the seemingly diverse case of social supergenes.

Methods

Sample collection

From each of 108 *Pheidole pallidula* colonies, 58 colonies were sampled in 2002-04 from France (Fournier, Aron and Milinkovitch, 2002) and Spain, and 57 were sampled in 2016-17 from France, Spain and Italy. Based on their geographic distribution, we expect all samples to be from the subspecies *Pheidole pallidula pallidula* (Seifert, 2016). We collected minor and major workers, either selected from within the colony, or attracted with bait outside the colony. All were stored in 100% molecular grade ethanol. We named three populations based on the location of the majority of samples: Bruniquel, Vigliano, Iberia (Figure A3.S6).

Microsatellite genotyping

We genotyped each colony using microsatellite markers and estimated its number of queens. We first extracted DNA from eight workers of each colony, using a slightly adapted version of a protocol based on a co-polymer solution (Gadau, 2009). Briefly, we reduced each individual into small fragments on a FastPrep homogeniser (MP Biomedicals) for two two-minute cycles of 8,000 rpm separated by one minute of rest, in a 5% Chelex solution with 1g ceramic beads. We then proceeded to the incubation and centrifuge steps as per the protocol. We evaluated DNA yields using fluorometry (Qubit 2.0, Life Technologies).

We then amplified six microsatellite loci, using species-specific markers developed by (Fournier, Aron and Milinkovitch, 2002), with fluorescent forward primers (VIC *Ppalo1T,* NED *Ppal33,* PET *Ppal84*, FAM *Ppalo3*, FAM *Ppal73,* VIC *Ppal12*) and non-fluorescent reverse primers. We performed multiplex PCRs using Type-It PCR kit (Qiagen) with 1µl of extracted DNA, following the manufacturer's cycling conditions modifying the annealing step (90s at 61 °C) and the total number of cycles (35). The microsatellite genotyping was performed on a 3730 DNA Analyser (Applied Biosystems); we subsequently determined the microsatellite length for each marker with GeneMarker software (v.2.4.0, SoftGenetics).

For each colony, we estimated the number of queens by counting the number of alleles present in each primer. With the assumptions that this species is singly-mated (Fournier, Aron and Milinkovitch, 2002) and that eight workers are a fair representation of the colony genotype, we use a simple rule: if more than three alleles are present at a locus within a colony dataset, the colony has multiple queens.

We subsequently evaluated the level of potential sample outliers with a principal component analysis, inputting the genotypes of each colony in the R package adegenet (Jombart, 2008; R Core Team, 2014).

DNA extraction for Illumina library preparation and sequencing

We first extracted high-quality DNA from one representative worker from 39 single-queen colonies and from 76 multiple-queen colonies (Table A3.S3). We followed a phenol-chloroform extraction protocol (Hunt and Page, 1995), with slight modifications: 10µl Proteinase K were added to the CTAB step, and we omitted the NaCl-Tris-Cl step. We further cleaned the extractions using part of Sigma Aldrich GenElute™ Mammalian Genomic DNA Miniprep Kit protocol (from step 4; catalogue number G1N70) and reduced the extraction volume to 20µl with an Evaporator centrifuge (Uniequip, Univapo 100H).

We then prepared individual libraries for whole-genome sequencing, using NEBNext® Ultra™ II FS DNA Library Prep Kit for Illumina (catalogue number E7805) and a combination of two primer sets (NEBNext® Multiplex Oligos for Illumina® Dual Index Primers Set 1 catalogue number E7600 and Set 2 catalogue number E7780). We altered the protocol by reducing in half the reagent volumes, to improve performance while reducing the costs, inspired by (Tan and Mikheyev, no date). Each resulting library was controlled for fragment size (300bp) using TapeStation High-Sensitivity tape (Agilent). 115 libraries were pooled in equimolar quantities. The final 15nM pool was sequenced on three lanes of Illumina HiSeq 4000 platform with 150 bp paired-end (Genewiz). We obtained 2,762,930,432 short-read raw sequences (Table A3.S6), which is the equivalent of more than 1,300x genome coverage (genome size estimated from *Pheidole* genus average in (Tsutsui *et al.*, 2008)). Each sample contributes to an average of 16x coverage, with one outlier sample E15 with 66x coverage. The number of sequenced reads is significantly different between single-queen and multiple-queen samples (Kolmogorov-Smirnov test P = 0.00558, Wilcoxon test P = 0.01009).

Species identification

We downloaded every COI barcode sequence for the taxon “Pheidole” in the BOLD database (v3.boldsystems.org), as well as the unique COI sequence of *Pheidole pallidula* from NCBI (GenBank: EF518381.1), whose sample originated from France (pers. comms Corrie Moreau). We reduced the number of BOLD sequences by collapsing redundant sequences

(cd-hit-est v4.6.8, overlap c = 0.97, word size n=10, length of description in .clstr file d= 0). The final database file contained 600 BOLD sequences and one NCBI sequence. For each sample, we compared the Illumina raw reads (forward and reverse) against that database (Magic-BLAST v1.4.0 -dbtype nucl -parse_seqids (Boratyn *et al.*, 2019)). To identify each sample, we assign the taxon name of the database sequences that bore the following criteria: 100% identity and the highest alignment score. All samples were identified as *P. pallidula*, with predictable regional proximity: our Italian and Corsican samples match BOLD barcodes from an Italian sample, French and Spanish samples match the NCBI barcode from the French sample.

Long read library preparation and sequencing

Prior to this study, there was no reference genome for this species and the closest species *Atta cephalotes* (Ward, Seán G. Brady, *et al.*, 2015) with an assembled genome (Suen *et al.*, 2011) is 83.5 My apart from *Pheidole.* We thus obtained one queenless French colony, kept alive in the laboratory, for MinION sequencing (colony 12). Workers were fed twice a week on Bhatkar diet (Bhatkar and Whitcomb, 1970) and water *ad libitum*.

We selected males and workers from two single-queen colonies, to reduce the allelic diversity of the resulting assembly in the potential supergene, under the assumptions of a *Solenopsis/Formica* system with homozygous supergene in single-queen samples. We extracted high-molecular weight DNA from each sample. We first reduced the samples in pellets using either a hand pestle, or a tissueRuptor. We then applied the phenol-chloroform extraction method mentioned above, with special modifications to ensure the preservation of long molecules. We prepared six libraries based on two chemistry kits (2D and $1D^2$); following ONT MinION protocols with several modifications. The libraries were subsequently sequenced following the manufacturer's instructions. Six sets of reads were obtained from the MinION runs (17x), basecalled with ONT Albacore version 2.1.7. We obtained raw reads with an average sequence length of 2.6Kb and an overall genome coverage of 17x (Table A3.S7).

De novo assembly Ppal_gnE

We assembled all MinION reads, using Flye v2.4 (Kolmogorov *et al.*, 2019) and a genome size estimation of 300Mb (Tsutsui *et al.*, 2008). We used Pilon v1.22 (Walker *et al.*, 2014) to improve the assembly, by reducing error rate and increasing contiguity. Pilon parameters were: --fix snps, indels --diploid. The samples used to polish were a set of 10 single-queen Italian samples. The reads were cleaned using the approach described in Chapter 2 and sub-sampled to obtain equal coverage from each sample. The final assembly (Ppal_gnE) was improved after ten Pilon iterations.

We controlled the quality of the finished assembly, by obtaining an estimation of continuity (N50) and other diagnostics with QUAST v4.6.1 (Gurevich *et al.*, 2013) and an estimation of gene completeness with BUSCO v3.0 using a total of 1,658 insect references (Simão *et al.*, 2015).

Scaffolding Ppal_gnE

We scaffolded Ppal_gnE assembly (generated with Flye and polished with Pilon) using AGOUTI (Zhang, Zhuo and Hahn, 2016) using publicly available RNA-seq reads (GenBank: EF518381.1). We first mapped these RNA-seq reads to each other with bwa (Li and Durbin, 2009), and we generated an annotation file with MAKER (Cantarel *et al.*, 2007) for AGOUTI input. We performed AGOUTI scaffolding with a minimum of five supporting RNA reads, with 318 contigs being scaffolded. The resulting assembly statistics were improved: N50 is 587,760 (previously 446,424bp), number of contigs is 3,954 (previously 4,130).

Reference-based analysis (mapping, variant calling, filtering)

We performed a reference-based variant calling using the assembly *Ppal_gnE* and Illumina raw reads on the QMUL HPC facility (King, Butcher and Zalewski, 2017). We first mapped raw reads of each sample to the assembly using Bowtie2 v2.3.4 (Langmead *et al.*, 2009) local alignment, obtaining 115 BAM files with alignments private to each sample.

We first quantified the variation between single-queen and multiple-queen samples in our alignment dataset using R. The proportion of mapped reads is not significantly different between social form (Figure A3.S7, T test P = 0.58). The mapping quality is significantly different between social form (Figure A3.S8, Kolmogorov-Smirnov test on average mapping quality of each sample, P = 0.002, Wilcoxon test P = 0.00036). This can be explained in part by the geographical origins of samples: samples from the same geographical locality as the reference assembly have a smaller difference between social forms (Kolmogorov-Smirnov test P = 0.04, Wilcoxon test P = 0.01).

We then used FreeBayes (--use-best-n-alleles 2; version 1.2.0 (Garrison and G., 2012)) to call the variants, obtaining 587,048 variants. We filtered the variant file with BCFtools v1.8 (Li *et al.*, 2009; Garrison and G., 2012), Tabix v0.2.5 (Li, 2011) and VCFtools v0.1.15 (Danecek *et al.*, 2011). Briefly, we sorted and indexed the VCF file, we kept biallelic SNPs, with a minimum quality phred of 30 and minimum sample support of 75% (--remove-indels --minQ 30 --min-alleles 2 --max-alleles 2).

We initially investigated this variant dataset by conducting a PCA using PLINK (--allow-extra-chr --allow-no-sex --pheno –cluster; version 1.90b4.6 (Chang *et al.*, 2015)) and visualised the results using R. We asked if some samples were outliers based on regional groups, and we estimated the proportion of variance explained by each principal component (n = 20 PCs). Seven samples were removed from the analysis with VCFtools, due to their outlier nature after the coverage analysis and after the PCA: three Spanish samples under the hypothesis of species misidentification, three samples under the hypothesis of mislabelled (PCA cluster in contradiction with geographical origin), one sample under the hypothesis of contamination (ten times more read depth).

We further filtered the variant database with VCFtools by keeping SNPs that are monomorphic within each population. Using the resulting 121,786 SNPs, we calculated Fisher's exact tests for association with social organisation using PLINK and visualised the results using R.

We replicated the association tests and visualisation for each population, enabling us to compare at the contig level the correlation between unadjusted *P* values of Bruniquel and Vigliano (Figure 2b). We also calculated FST values between social forms for each population using R package PopGenome (2.7.5) (Pfeifer *et al.*, 2014).

Simulations of association test with supergene region

We first performed a simulation in which one simulated variant (homozygote in single-queen samples, heterozygote in multiple-queen samples) was added to the VCF (109 samples, within coding regions). We ran a Fisher's exact test (allele count in two sample categories: single-queen and multiple-queen) using PLINK (Chang *et al.*, 2015), adjusted for multiple comparisons in R (p.adjust, Benjamini & Hochberg method, see (Benjamini and Hochberg, 1995)). This simulated SNP was by far the most significant (Figure A3.S5).

To test whether our analysis would detect a realistic supergene, we additionally simulated a dataset that replicates the model of *S. invicta*. We first selected 3,757 real *Pheidole* SNPs (100% sample-coverage, in coding regions, within-population polymorphic, from 108 diploid workers). Based on Pracana (Pracana, Priyam, *et al.*, 2017), 2,5% of all SNPs are fixed in *S. invicta* supergene region. We therefore added 95 simulated SNPs to the dataset, represent what we expect from the supergene system: homozygous SNPs for all single-queen and 1/3 of multiple-queen samples, heterozygous SNPs for 2/3 of multiple-queen samples (Buechel, Wurm and Keller, 2014). We then ran Fisher test and adjusted for multiple comparisons. All simulated SNPs are associated with social form (P_{adj} value < 0.05), strongly segregated away from the real dataset points (Figure A3.S7).

We additionally simulated a dataset that replicates the model of *F. selysi*. Based on (Purcell *et al.*, 2014), 3.7% of all SNPs are fixed in the supergene region. We therefore added 139 simulated SNPs to the dataset, represent what we expect from the supergene system: homozygous SNPs for all single-queen, 68% heterozygote multiple-queen samples and 32% multiple-queen samples that are homozygous for the alternative allele. All simulated SNPs are associated with social form (P_{adj} < 0.05), strongly segregated away from the real dataset points. As predicted, the Fisher test detects more easily the *Formica* SNPs (in which there is

strictly no homozygote for reference in multiple-queen samples) than the *Solenopsis* SNPs, with respectively *P* values of 7.659e-09 and 6.402e-18.

Assembling non-mapping reads

We used SPAdes (version 3.12.0 (Bankevich *et al.*, 2012)) to assemble the reads from polygynous samples that did not map to the monogynous assembly.

Data availability

Raw reads, the genome assembly, and analysis scripts are available on request. They will be uploaded respectively to NCBI and GitHub.

Acknowledgements

We thank Xavier Espalader, Nilo Ortiz de Zugasti Carron, Pedro Lorite Martínez, Janine Remoué and Carlos Martinez Ruiz for assisting in the sampling effort; Richard A. Nichols and Christophe Eizaguirre for advice and discussion. This work was supported by the Natural Environment Research Council [grant NE/L002485/1 to EF and NE/L00626X/1 to YW] and the Biotechnology and Biological Sciences Research Council [grant BB/K004204/1 to YW]. This research utilised Queen Mary's Apocrita HPC facility, supported by QMUL Research-IT.

Chapter 8: Discussion

In the previous chapters, I presented the limitations of reference genome assemblies, tools to overcome some of the limitations, and examples of studies that are now possible because of lower costs of genome sequencing than ever before. In this chapter, I will discuss some of the recent developments in genome sequencing and assembly and put my thesis in perspective of these developments. Next, I will discuss some of the challenges in biological data sharing, integration, and their potential solutions.

Technology trickles down slowly

Sequencing technologies are advancing and so are the assembly approaches. Human genome continues to demonstrate what is possible. Just this year, Nurk *et al.*, (2022) demonstrated a complete, telomere-to-telomere (T2T) assembly of all human chromosomes except Y. One might then assume that it is just a matter of time before a complete, T2T genome assembly can be generated for other species. Let us put that in perspective.

First, the human T2T genome assembly was generated by sequencing a hydatidiform cell line which is almost perfectly homozygous (Fan *et al.*, 2002). Researchers working with other species may not be so lucky. Instead, they need to work with what is effectively a diploid or a polyploid sample, such as a population of haploid sperm cells, or DNA from whole-body containing both the copies of parental DNA as well as somatic mutations from the different cell types. Considerable advances have made in the genome assembly using diploid samples (Cheng *et al.*, 2022; Jarvis *et al.*, 2022). However, the applicability of these approaches for a genome that is to be sequenced for the first time depends on two factors. First, some of the approaches require sequencing parents-offspring trio (Jarvis *et al.*, 2022) which may limit their applicability (Cheng *et al.*, 2022). Second, genome organisation of a newly sequenced species may violate the assumptions of the underlying computational methods as Cheng *et al.*, (2022) themselves found in the case of the sterlet genome. Indeed, genome assembly algorithms often make assumptions or have thresholds derived from the datasets they were

tested with, and the assumptions may not hold, or the thresholds may not be optimal for a different species. In fact, this was one of the key points of Chapter 2.

Second, the human T2T genome assembly was generated and validated using an array of complementary sequencing technologies (e.g., Pacific Biosciences HiFi, Oxford Nanopore ultra-long, Hi-C, Illumina, Strand-seq). This may not only be prohibitively expensive for many labs that work on a much smaller budget, but also, they may lack the expertise to reliably generate libraries for the different sequencing technologies. An example from personal experience is the Oxford Nanopore reads for the big-headed ant, *Pheidole pallidula*, in Chapter 7. While the words ultra-long reads (>40,000 bp) are often associated with Oxford Nanopore, the average read length we obtained was a mere 2,600 bp. Unfortunately, due to funding constraint it was not possible to get the sample re-sequenced at a sequencing centre with greater experience, and due to time (and funding) constraint it was not possible to improve the sequencing run by ourselves. One finds many such stories at conferences or stories where appropriate expertise had to be developed before long-molecular sequencing could be applied.

Third, the human T2T genome assembly was not the outcome of an automatic assembly pipeline. It required significant manual curation. To quote Jarvis *et al.* (2022) "Completing the T2T-CHM13 assembly also required a substantial amount of manual curation by dozens of people over many months, with different groups focused on each chromosome." Such manpower and expertise are also beyond the reach of many labs working on different species. Due to funding and project-specific constraints many small labs restrict themselves to the final output of genome assembler or adopt some heuristics (e.g., polish the assembly using Illumina reads), without truly understanding their pitfalls (like how polishing can homogenise repeats) and validating the results (e.g., quantifying assembly accuracy before and after polishing). Instead, the onus of accounting for assembly errors is often left for downstream analysis (Chapters 6 and 7 are good examples of how individual analyses must deal with assembly imperfections).

Thus, generating completed, T2T genome assemblies for all species, or more specifically, from any given sample (for the ultimate high-resolution analysis) still requires further

advances in the handling of diploid and polyploid samples, including sequencing library preparation, faster and more adaptive assembly algorithms that can automatically produce the optimal assembly for any given species and sequence dataset, and further reduction of sequencing costs (Pacific Biosciences sequencing costs approximately four times more than Illumina at the time of this writing). The advances required are numerous and complex enough that they will take a long time to materialise. However, based on the importance of the research questions and the funding landscape some researchers may push the envelope and generate a complete, T2T genome assembly for their species. While others will continue to answer biological questions that they can using the technologies and tools at their disposal given the project and funding constraints.

We are already at a stage where long-molecule sequencing has been applied to many species to generate higher-quality genome assemblies and answer biological questions that were impossible with short-read sequencing. This has already added to our collective knowledge of what works and what doesn't and made it possible for more researchers to obtain higher quality genome assemblies for their species. As more and more researchers continue to apply long-molecule sequencing and other forthcoming advances in genome sequencing and assembly approaches, the technology and the know-how will slowly trickle down. Until one day, we may indeed have a complete, T2T gnome assemblies for all species. However, in the process, we will generate many incomplete and imperfect genome assemblies, build communities around them, and answer biological questions despite their limitations.

My contributions

My thesis is set against the above-described backdrop of iterative technological progress. Chapter 2 encourages researchers to make the best of the available technology by testing different genome assembly software and assembly parameters. I describe a tool that can be used to rank the generated assemblies using multiple metrics of assembly quality and select the best. The chapter further demonstrates the benefit of such assembler-parameter-space exploration. Next, as technologies advance and the research questions move forward, newer

genome assemblies will inevitably be generated to either replace the old one or to complement it (e.g., to better capture the allelic diversity of a population). The tool described in Chapter 3 can then be used to carry over annotations to the newer, or alternate assembly. Finally, as discussed in the previous section, the onus to account for assembly errors often falls on downstream analyses, Chapters 4 and 5 empowers the researchers with tools that can be used to QC genomic regions and gene annotations in a targeted and visual manner.

Data sharing and integration

The current generation of sequencing technologies already produce dramatically better genome assemblies than their predecessors. This has spurred an ambitious international collaboration to produce reference genome assemblies for millions of species (Lewin *et al.*, 2018). The reference assemblies will likely be accompanied by reference gene annotations and datasets that were used to generate them. The result will be treasure trove of data. However, the scale of data will also bring with it challenges in their dissemination and analysis. I discuss two such challenges below and present potential solutions.

Data qualities

Genome assemblies and annotations are typically deposited to databanks upon publication, from where they can be downloaded by anyone for further analysis. However, the databanks do not often do a very good job of communicating the quality of the available data. Genome assemblies are presented only in terms of their contiguity and genome annotations in terms of counts of feature types and their size distribution. As a result, each researcher or research lab must repeat the same quality control steps, which is wasteful and increases the chances of propagating errors.

For genome assemblies, the databanks can provide quality information at three levels to reduce the above-mentioned inefficiencies. First, they can compute and display the metrics presented in Chapter 2 to provide a summary overview of genome assembly quality and

help researchers choose if multiple competing genome assemblies are available. Second, by assigning a Phred-like quality score to each base in the assembly using an approach such as Referee (Thomas and Hahn, 2019) and making this information available for download. Third, by identifying regions likely to be problematic (i.e., containing mis-assemblies), for example, using an approach similar to (Warr *et al.*, 2015) and making the information as BED file so that such regions they can be eliminated or specifically-treated by dependent analysis.

A similar approach can work for genome annotations. Approaches such as BUSCO (Simão *et al.*, 2015) can be used to provide biologically meaningful summary overview of the of gene-set's quality. While, quality of individual feature annotations can be indicated using metrics such as Annotation Edit Distance (Holt and Yandell, 2011). Alternatively, approaches such as GeneValidator (Drăgan *et al.*, 2016) can be used not only to assign a score to each annotation, but also indicate the type of error that may be present.

Infrastructure for secondary databases

Researchers often create custom databases focussed on a particular species or a lineage to foster collaboration, to reduce inefficiencies in accessing data directly from large primary databases, and to provide unique integrations in the form of additional data types, or search, download, analysis and visualisation tools that are not supported by the primary databases (see Annex 1, Table A1.2 for examples). This has led to the development of software tools that can be pieced together to create such databases (gmod.org). However, these software are not always easy to install and provision. Furthermore, there is the associated cost of procuring and maintaining the hardware on which the database will run and dedicating human resource for developing custom integrations and providing long-term maintenance such as, software updates or incorporating newly published datasets. It is not uncommon to a) hire dedicated software developer(s) for the creation and maintenance of such databases and b) leave the databases unmaintained if the funding runs dry or due to other operational challenges. The costs can limit smaller research communities and individual labs from creating and operating custom databases. Arguably, individual labs and small research communities stand to benefit the most from such centralisation of efforts. While databases

that could be but were never created or left unmaintained represent missed opportunities for the scientific community as a whole.

I created SequenceServer with Yannick (Chapter 4) in response to some of these challenges. We purposefully designed it to be easy to install, integrate, operate and maintain. Today, it has become a popular way to provide BLAST-search functionality in several custom databases, both large and small (Annex 1, Table A1.2). JBrowse genome browser is another example of a database component that has become popular due to its ease of setup, use, and integration (Skinner *et al.*, 2009; Buels *et al.*, 2016; Yao *et al.*, 2020). Such purposeful design of software components that are used to build custom databases can considerably lower the costs of their creation and maintenance. Furthermore, setup of a private company dedicated to providing hardware, software, and human resources for creation and maintenance of custom databases can simplify many of the challenges by amortising costs and developing standard integrations.

Annex

Annex 1: Supplementary information for chapter 4

Supplementary tables

Table A1.1: Research using Sequenceserver

Interplay of chimeric mating-type loci impairs fertility rescue and accounts for intra-strain variability in *Zygosaccharomyces rouxii* inter-species hybrid ATCC42981	Bizzarri et al., 2019
A genome-wide association study of non-photochemical quenching in response to local seasonal climates in *Arabidopsis thaliana*	Rungrat et al., 2019
Taraxacum kok-saghyz (rubber dandelion) genomic microsatellite loci reveal modest genetic diversity and cross-amplify broadly to related species	Nowicki et al., 2019
Developmental expression and evolution of hexamerin and haemocyanin from *Folsomia candida* (Collembola)	Liang et al., 2019
Disentangling the mechanisms of mate choice in a captive koala population	Brandies et al., 2018
Evidence for sexual reproduction: Identification, frequency, and spatial distribution of *Venturia effusa* (pecan scab) mating type idiomorphs	Young et al., 2018
Pseudomonas fluorescens group bacterial strains are responsible for repeat and sporadic post pasteurization contamination and reduced fluid milk shelf life	Reichler et al., 2018
Complete pathway elucidation and heterologous reconstitution of *Rhodiola salidroside* biosynthesis	Torrens-Spence et al., 2018
Evolution of the shut-off steps of vertebrate phototransduction	Lamb et al., 2018
De novo draft assembly of the *Botrylloides leachii* genome provides further insight into tunicate evolution	Blanchoud et al., 2018
Whole-genome sequence of the metastatic PC3 and LNCaP human prostate cancer cell lines	Seim et al., 2017
Fire ant social chromosomes: Differences in number, sequence and expression of odorant binding proteins	Pracana et al., 2017
Ecological genomics for the conservation of dwarf birch.	Borrell, 2017
Transcriptomic discovery and comparative analysis of neuropeptide precursors in sea cucumbers (Holothuroidea)	Suwansa-ard et al., 2018
High-throughput genotyping analyses and image-based phenotyping in *Sorghum bicolor*	McCormick, 2017
Bacteriocins of non-aureus staphylococci isolated from bovine milk	Carson et al., 2017

Naturally occurring high oleic acid cottonseed oil: Identification and functional analysis of a mutant allele of *Gossypium barbadense* fatty acid desaturase-2	Shockey et al., 2016
3D sorghum reconstructions from depth images enable identification of quantitative trait loci regulating shoot architecture	McCormick et al., 2016
A workflow for studying specialized metabolism in nonmodel eukaryotic organisms	Torrens-Spence et al., 2016
Transcriptomic identification of starfish neuropeptide precursors yields new insights into neuropeptide evolution	Semmens et al., 2016
Multi-species sequence comparison reveals conservation of ghrelin gene-derived splice variants encoding a truncated ghrelin peptide	Seim et al., 2016
Characterization of a second secologanin synthase isoform producing both secologanin and secoxyloganin allows enhanced de novo assembly of a *Catharanthus roseus* transcriptome	Dugé de Bernonville et al., 2015
Identification and heterologous expression of the chaxamycin biosynthesis gene cluster from *Streptomyces leeuwenhoekii*	Castro et al., 2015
Discovery of sea urchin NGFFFamide receptor unites a bilaterian neuropeptide family	Semmens et al., 2015
Comparative analysis reveals loss of the appetite-regulating peptide hormone ghrelin in falcons	Seim et al., 2015
Reconstructing SALMFamide neuropeptide precursor evolution in the phylum Echinodermata: Ophiuroid and crinoid sequence data provide new insights	Elphick et al., 2015
Molecular biology approaches in bioadhesion research	Rodrigues et al., 2014
Discovery of a novel methanogen prevalent in thawing permafrost	Mondav et al., 2014
Neuropeptides and polypeptide hormones in echinoderms: New insights from analysis of the transcriptome of the sea cucumber *Apostichopus japonicus*	Rowe et al., 2014
Discovery of a novel neurophysin-associated neuropeptide that triggers cardiac stomach contraction and retraction in starfish	Semmens et al., 2013
The evolution and diversity of SALMFamide neuropeptides	Elphick et al., 2013
The protein precursors of peptides that affect the mechanics of connective tissue and/or muscle in the echinoderm *Apostichopus japonicus*	Elphick, 2012

Table A1.2: Public community websites using Sequenceserver

Reference /description	**URL**
Dieterich et al., 2007. Genomic resources for the nematode, *Pristionchus pacificus*	pristionchus.org
Amborella Genome Project, 2013. Amborella genome database	amborella.uga.edu
Chiu et al., 2013. Spotted wing fly-base	spottedwingflybase.org
Petrillo et al., 2015. JRC GMO-amplicons: Database of amplicon sequences related to genetically modified organisms	gmo-crl.jrc.ec.europa.eu jrcgmoamplicons/db scans/blast
Kirmitzoglou and Promponas, 2015. LCR-eXXXplorer: Explore low complexity regions in protein sequences	repeat.biol.ucy.ac.cy/fgb2 gbrowse/swissprot/
Brandl et al., 2016. Planmine: Data and tools to mine planarian biology	planmine.mpi-cbg.de

Mun et al., 2016. Lotus-base: Resources, tools, and datasets for the model legume *Lotus japonicus*	lotus.au.dk
Liew et al., 2016. ReefGenomics: Genomic and transcriptomic data for marine organisms	reefgenomics.org
Shen et al., 2016. Y1000+ project: Initiative to sequence 1000 wild yeasts	y1000plus.wei.wisc.edu
Nakagawa and Takahashi, 2016. gEVE: Database of genome-based endogenous viral elements	geve.med.u-tokai.ac.jp
Janies et al., 2016. EchinoDB: Database of orthologous transcripts from echinoderms	echinodb.uncc.edu
Louro et al., 2016. Assembled transcriptomes of sea bass and sea bream	sea.ccmar.ualg.pt:4567
Hane et al., 2016. Lupin genome portal: Genome assembly and annotations for the narrow-leafed lupin	lupinexpress.org
Challis et al., 2016. Lepbase: Lepidopteran genome database	lepbase.org
Zhu et al., 2017. CottonFGD: Cotton functional genomics database	cottonfgd.org
Hill et al., 2017. Hopbase: Database for genomics of *Humulus lupulus* (hop)	hopbase.org
Torres et al., 2017. LeishDB: Database for leishmania genomic information	leishdb.com
Naas et al., 2017. BLDB: Beta-lactamase database	bldb.eu:4567
Elsik et al., 2018. Hymenoptera genome database	hymenopteragenome.org
Hagen et al., 2018. Bovine genome database	bovinegenome.org
Meng et al., 2019. CircFunBase: A database for functional circular RNAs	bis.zju.edu.cn/CircFunBase
Ravindran et al., 2018. Daphnia stressor database: Gene expression database for Daphnia	www.daphnia-stressordb.uni-hamburg.de/dsdbstart.php
Gene expression database for Alvinella pompejana, and *Platynereis dumerilii*	jekely-lab.tuebingen.mpg.de
EFISH Genomics 2.0: web portal for electric fish genomic resources	efishgenomics integrativebiology.msu.edu
NBIGV, Non-B cell derived immunoglobulin variable region database	nbigv.org
iBeetle-base: Database of Tribolium RNAiphenotypes	ibeetle-base.uni-goettingen.de
Cacao genome database	cacaogenomedb.org
Ant genomes, predicted transcripts and proteome	antgenomes.org
Aplysia transcriptome	aplysiagenetools.org:4567
Ash tree genome	ashgenome.org
Asparagus genome project	asparagus.uga.edu
Dwarf birch genome project	birchgenome.org
Fallon et al., 2018. Firefly genome database	blast.fireflybase.org
Genome, predicted transcripts and proteins of tardigrades	blast.tardigrades.org
Botulinum neurotoxin database	bontbase.org
eplant.org: Sequenced genomes of all plants to facilitate comparative genomic studies	eplant.org:4567

FusoPortal: A Fusobacterium genome and bioinformatic repository	fusoportal.org
NCHU fish genome database	lep-fish.nchu.edu.tw:4567
Fish genome database	brcwebportal.cos.ncsu.edu:4567
MarpolBase: Genome database for the common liverwort, Marchantia polymorpha	marchantia.info
MitoFun: A curated resource of complete fungal mitochondrial genomes	mitofun.biol.uoa.gr
Oat genome	oatgenomeproject.org
Spiny mouse transcriptome	spinymouse.erc.monash.edu
Measles, mumps, and rubella viruses database and analysis resource	mmrdb.org
Whole-genome sequence of the metastatic PC3 and LNCaP human prostrate cancer cell lines	ghrelinlab.org
10.1093/dnares/dsz003 Genome database for Iberian ribbed newt	inewt.nibb.ac.jp:8111
Crop genomics lab's BLAST server	plantgenomics.snu.ac.kr
Exome of Kronos durum wheat and Cadenza bread wheat mutants	wheat-tilling.com
Gene expression analysis and visualisation for wheat	wheat-expression.com
Fungal genomics	fungalgenomics.science.uu.nl
Stazione Zoologica Anton Dohrn	glossary-blast.bioinfo.szn.it
Georgia State University	db.cbn.gsu.edu:4568
Desplan Lab (Drosophila developmental biology)	desplan-lab.bio.nyu.edu
Commonwealth Scientific and Industrial Research Organisation	hieracium.csiro.au
Institute of Cytology and Genetics of Siberian Branch of the Russian Academy of Sciences	seqserver.sysbio.cytogen.ru
Taiwan Agricultural Genomics Resource Center	tagrc.org:4568, tagrc.org:4569

Annex 2: Supplementary information for chapter 5

Supplementary Methods

OBP discovery and manual gene model curation

We used MAKER2 (version 2.31 (Cantarel *et al.*, 2007)) to generate consensus gene models for the *S. invicta* genome assembly (Wurm *et al.*, 2011) from TopHat2 (version 2.0.11 (Kim *et al.*, 2013) alignments of RNA-seq reads (SRA accessions SRX757226-SRX757228) to the reference genome, an assembly of fire ant Expressed Sequence Tag (EST) libraries, protein sequences from SwissProt (downloaded June, 2014), *A. mellifera* (amel_OGSv3.2_pep.fa) and *N. vitripennis* (Nvit_OGSv1.2_pep.fa) genome projects, and *de novo* predictions from SNAP (Korf, 2004) and Augustus (Stanke *et al.*, 2006) using HMM models that had been generated during the fire ant genome project (Wurm *et al.*, 2011). To identify regions of the genome putatively containing OBPs, we performed blastn and tblastn (Camacho *et al.*, 2009; Priyam *et al.*, 2019) searches of the fire ant genome on antgenomes.org (Wurm *et al.*, 2009) using as queries previously published fire ant OBP sequences (Gotzek *et al.*, 2011) and Uniprot sequences that are part of the Pfam family 'PBP_GOBP' (Finn *et al.*, 2014; The Uniprot Consortium, 2015). We integrated all aforementioned data using the genomic annotation editor Afra (github.com/wurmlab/afra), the genome browser JBrowse (Skinner *et al.*, 2009), the tool GeneValidator (which assesses the quality of annotations by comparing them to public databases (Drăgan *et al.*, 2016)) and custom scripts. Manual curation followed a the standard approach based on Web Apollo (Lee *et al.*, 2013), including the inspection and adjustment of exon boundaries to ensure that the exon-intron structure of gene models was consistent with mappings of RNA sequence reads, and that the gene models had canonical splice sites, translation start and stop sites, and appropriate open reading frames. We also identified alternative spliced transcripts by visualising the alignments of the RNA sequence

reads. After producing high quality gene models through this method, we used these sequences for further blastn and tblastn searches against the reference genome to identify further putative OBPs. These were curated as above; this process was repeated iteratively until no new putative OBP loci were discovered.

Our pipeline identified seventeen out of the eighteen OBP genes that had been previously reported, although eight had differences in sequence and/or length relative to the published sequences (Table A2.S1). We found no genomic region with more than 80% identity to the remainder gene, *SiOBP18* (Table A2.S1), which had been identified from a single Sanger-sequenced EST (Wang et al. 2007; Gotzek et al. 2011), suggesting that this gene is either missing from the reference genome assembly, or that the sequence of the original EST was an artefact. We found evidence of alternative splicing for *SiOBP17* (four splice forms) and for the newly discovered *SiOBPZ7* (two splice forms). We found no support for the suggestion that *SiOBP12* and *SiOBP13* share an exon (Zhang *et al.*, 2016). All the genomic and transcriptomic sequences analysed in the OBP discovery pipeline included an insertion relative to the reference assembly affecting *SiOBP14* (insertion of a T in position NW_011802221.1:1,287,729), suggesting an error in the assembly. The sequence reported for this gene includes this insertion.

(Pracana, Priyam, *et al.*, 2017) (Pracana, Priyam, *et al.*, 2017) (Pracana, Priyam, *et al.*, 2017) to assign OBPs to linkage groups. We were able to position 20 of the OBPs in linkage groups, including all novel OBPs (Figure 6.1). Four OBPs were in unmapped scaffolds. Of these, *SiOBP9* is in a scaffold that we previously classed as putatively in the supergene (Pracana, Priyam, *et al.*, 2017) given its high SB-Sb differentiation. *SiOBP2* was in a scaffold without any divergence, thus classified as outside the supergene. It was not possible to confidently determine the positions of the remaining two OBPs (*SiOBP5* and *SiOBP7*), as each had exons in multiple small unmapped scaffolds.

Phylogenetic analysis

S. invicta OBPs are a highly divergent gene family (Gotzek *et al.*, 2011). We aligned the coding sequences of the 24 *S. invicta* OBPs using MAFFT-linsi (version 6.903b (Katoh and Toh,

2008)). We removed ambiguous sections from this alignment using trimAL (version 1.4.1; (Capella-Gutiérrez, Silla-Martínez and Gabaldón, 2009)) with the -gappyout option and built a "guide" tree using RaxML (version 8.2.9; (Stamatakis, 2006)) with the GTRGAMMAI model. We then used PRANK (version 120626; (Löytynoja and Goldman, 2005)) to generate a codon-level alignment of the original sequences, guided by the tree obtained above. Using the same parameters as above, we removed ambiguous sections from this alignment using trimAl and built a final tree using RaXML (10,000 bootstraps).

Read filtering of *S. invicta* whole-genome sequences

We used whole-genome sequences from one *SB* and one *Sb* male from each of seven colonies that had been sequenced at low coverage (Illumina 2*100bp paired-end genome shotgun sequences; ~6x-8x coverage) in 2012 (NCBI SRP017317) (Wang *et al.*, 2013). Each of these samples is a haploid male (ants have a haplo-diploid sex determination system), and the sequencing coverage is sufficiently homogeneous (Pracana, Priyam, *et al.*, 2017) for the analysis reported here, including high confidence genotype calling. We used seqtk (version 1.0-r31 (github.com/lh3/seqtk) to trim 2bp from the start and 5bp from the ends of the reads. We removed any read where more than 25% of the bases had a quality score smaller than 25 using fastq_quality_filter in the fastx toolkit (version 0.0.14 (hannonlab.cshl.edu/fastx_toolkit)). We used GNU parallel to parallelise this pipeline (Tange, 2011).

Detection of copy number and structural variation in OBPs

We used bowtie2 (version 2.1.0 (Langmead and Salzberg, 2012)) to align the cleaned reads to the reference genome assembly. We visually inspected the alignments of each of our curated gene predictions, searching for regions with no coverage to identify deletions and high coverage to identify duplications.

The genomic region that includes three exons of *SiOBP15* (scaffold NW_011801067.1:293,460-296,015) had no reads in any *Sb* individual, consistent with a deletion of this region (it is

impossible to determine the exact size of the deletion as the region is directly upstream of a non-assembled portion of the scaffold). We observed no other such pattern of deletion.

SiOBP12 (which is within the supergene region) had approximately two times higher coverage in *Sb* individuals relative to *SB* individuals. Approximately half of the reads from *Sb* individuals had a small number of consistent sequence differences to the other reads. This suggested that a recent duplication of the gene occurred. To obtain consensus sequences for *Sb* individuals for both duplicates, we extracted all pairs of reads from *Sb* individuals for which at least one pair mapped to the contig containing the transcribed region of *SiOBP12* and performed *de novo* assembly using MIRA (version 4.0.2 (Bastien Chevreux, Wetter and Suhai, 1999)). This resulted in assemblies on separate contigs of two genes: *SiOBP12* and the Sb-specific duplicate we named *SiOBPZ5*.

Visual inspection of *SiOBPZ6* (outside the supergene region) revealed that this gene had a much higher number of mapped reads than other genes. To estimate the number of copies of *SiOBPZ6*, we measured the median coverage per base pair of the seven *SB* individuals for this gene and for 1000 additional randomly sampled genes using bedtools coverage (with argument -d; version 2.25.0 (Quinlan and Hall, 2010)). For each individual, we calculated the ratio between the coverage of *SiOBPZ6* and the mean coverage of the 1000 random genes. For a single-copy gene, we expect these ratios to be one; we used a one-sample t-test to determine if the distribution of these ratios had a mean different from one. We did not produce individual sequences for each *SiOBPZ6* copy because there was an insufficient number of variable sites to differentiate the copies. The sample used for genome assembly (NCBI SAMN00014755 (Wurm *et al.*, 2011)) was not included in this test because it was sequenced using an earlier (noisier) Illumina technology.

Orthology in other species

Using a reciprocal blast approach, we searched for the closest orthologous sequence of each OBP gene. First, we ran a tblastn search of all *S. invicta* OBPs against all non-*S. invicta* arthropod sequences available on NCBI nr on 2017-03-21, accepting hits where e-value < 10^{-3}. We then ran a blastx search of these hits against the *S. invicta* gene predictions (including

our newly curated OBP set). We report the hits with the lowest e-value (Table A2. 7). We repeated this analysis by searching non-ant arthropods (not Formicidae).

Variant Calling in *S. invicta* OBPs

We added the contig with the *Sb*-specific *SiOBPZ5* to the reference assembly. Using bowtie2 (version 2.1.0 (Langmead and Salzberg, 2012)), we aligned the cleaned reads of the seven *Sb* individuals (see above) to the revised assembly and the seven *SB* individuals to the original reference assembly. We called single nucleotide polymorphisms (SNPs) in the protein coding regions of the supergene OBPs using samtools mpileup (Li *et al.*, 2009) and bcftools call (--ploidy 1 and –m; version 1.3.1 (samtools.github.io/bcftools/bcftools.html). We manually inspected the read alignments at each SNP position using the genome viewer IGV (Thorvaldsdóttir, Robinson and Mesirov, 2013).

Sequencing and variant calling of the OBPs of an outgroup species

We produced whole-genome sequencing reads of the outgroup species *Solenopsis geminata*. DNA was extracted from a pool of ten workers (sampled in Thailand by Dr Adam Devenish, University College London, United Kingdom) using the Phenol-Chloroform method in Hunt and Page (1994) and sequenced using Illumina HiSeq 4000 (11x coverage). We filtered the reads using skewer (version 0.2.2 (Jiang *et al.*, 2014)), with the following parameters: --mean-quality 20, --end-quality 15, -l 100, -n yes and -r 0.1. The reads were aligned to the *S. invicta* reference genome assembly using bowtie2 (version 2.1.0 (Langmead and Salzberg, 2012)). All OBPs were covered by *S. geminata* reads, although there was very low coverage (median coverage < 3) in the two terminal exons of *SiOBP12* and *SiOBP13*. Freebayes (version 1.0.2-33-gdbb6160 (Garrison and Marth, 2012)) was used to call variants between the sample and the reference assembly in the regions within 1000 bp of each OBP (excluding the two terminal exons of *SiOBP12* and *SiOBP13*). We filtered the variants using the parameter RO < 2, chosen based on visual inspection of the alignment using IGV (Thorvaldsdóttir, Robinson and Mesirov, 2013), and limited our analysis to homozygous positions within the coding sequence of each OBP.

Gene expression of *S. invicta* OBPs in publicly available RNA sequencing datasets

We analysed all available RNA sequencing (RNA-seq) data from the NCBI SRA database for *S. invicta* as of January 2017 (data from Wurm et al. 2011; Morandin et al. 2016 and PRJNA266847; details in Table S2). These included Illumina and Roche 454 sequences. Read quality was assessed using FastQC (version 0.11.5 (bioinformatics.babraham.ac.uk/projects/fastqc). Low quality bases were removed using the default options in fqtrim (version 0.9.5 (ccb.jhu.edu/software/fqtrim).

We determined the expression levels of *S. invicta* transcripts using count mode in Kallisto (version 0.43.0 (Bray *et al.*, 2016b)). As a reference, we modified the *S. invicta* protein-coding gene annotation release 100 (NCBI) by removing all automatically annotated OBPs and instead adding the OBP sequences we manually curated above. We masked regions of *SiOBP12* and *SiOBPZ5* that were identical between these recent duplicates to prevent reads from one gene to be misassigned to the other. *SiOBP15* lacks three exons in its Sb variant, so to prevent misalignment, we treated each variant of *SiOBP15* as a different transcript. The total read count for *SiOBP15* is the sum of its two variants. To control for the potential effects of sequence differences between SB and Sb, we repeated the analysis twice: first using the SB alleles of the OBPs, then using the Sb alleles. We only show the analysis done using the Sb alleles of the OBPs because both analyses produced qualitatively identical results.

For paired-end reads, we used the default counting options of Kallisto. For single-end reads, we provided Kallisto the average fragment length of each sample (as indicated on NCBI SRA) and we set the estimated standard deviation to 20 bp. To be able to analyse at least >50% of low-coverage Roche 454 reads with Kallisto, we set the average and the standard deviation of fragment length to 1.

We used Tximport (version 1.3.0 (Soneson, Love and Robinson, 2015)) to import the estimated counts produced by Kallisto into the R programming language implementation of DESeq2 (version 1.14.1 (Love, Huber and Anders, 2014)). Each sample was independently normalised using the DESeq2 method. Additionally, we performed genome-wide analysis

of differential expression on data from (Morandin *et al.*, 2016) using a standard DESeq2 approach to identify expression differences between social forms in queens and in workers. Queens and workers were analysed separately because they were sampled using different collection methods, which resulted in different variance patterns in each dataset (Morandin *et al.*, 2016). We included the Sb-specific *SiOBPZ5* in the analysis as control. As expected, this gene is significantly differentially expressed between single- and multiple-queen colonies in both workers and queens. We performed a standard Chi^2 test to determine whether the supergene region is enriched in differentially expressed loci relative to the rest of the genome.

Differential expression of gene co-expression modules across social forms

We created gene co-expression modules from two microarray sets comparing single-queen with multiple-queen colonies, one with queen samples (GSE42062 (Nipitwattanaphon *et al.*, 2013)) and the other with worker samples (E-GEOD-11694 (Wang, Ross and Keller, 2008)). We did not use the RNA-seq data because it does not include a sufficient number of samples of each social form to create gene co-expression modules. Both microarray sets use the same microarray platform (Platform GPL6930), which includes 25,344 probes (Wang *et al.*, 2007). To determine the number of genes that these probes represented, we aligned the sequences of all probes against the gnG assembly for *S. invicta* using the 'est2genome' mode with a minimum 95% identity in Exonerate (version 2.2.0 (Slater and Birney, 2005)). The positions of the probes in the genome were then intersected against the annotation release 100 for *S. invicta* used in the rest of analyses with the R package 'GenomicRanges' (Lawrence *et al.*, 2013). The probe sequences intersected with 3,673 unique genes.

We downloaded the normalised expression values of each dataset from NCBI GEO. For the queen set, we removed the 16 samples with reproductive age class because *SB/SB* reproductive samples had very low variance in gene expression relative to individuals of other age classes. The remaining set included 31 *SB/SB* samples and 31 *SB/Sb* samples (all virgin queens originating from multiple-queen colonies). The worker set included 20

samples from single-queen colonies and 40 from multiple-queen colonies (20 *SB/SB* and 20 *SB/Sb*; we removed two *Sb/Sb* samples). For each set, we removed any probe that had "null" expression in more than five individuals. For the remaining probes, individuals with "null" expression were imputed to the median expression of the probe. After filtering, there were 18,291 probes in common between the two datasets, representing 3,046 genes. We used the ComBat function in the sva R library (version 3.18.0 (Leek *et al.*, 2012)) to adjust both sets for the year in which the microarrays were produced. We used Weighted Gene Co-expression Network Analysis (version 1.49 (Langfelder and Horvath, 2008)) to create signed modules for each set. We used a soft-thresholding power of 5 for both sets. Modules were detected using the Dynamic Tree Cut method and merged using an eigengene dissimilarity threshold of 0.3. We used t-tests to determine whether any module eigengene is correlated with genotype or social form. In queens, we compared *SB/SB* to *SB/Sb* samples because all originate in multiple-queen colonies. In workers, we separated the effect of genotype from the effect of social form following the approach in Wang *et al.* (2008): we compared genotypes (*SB/SB* versus *SB/Sb*) using samples from multiple-queen colonies, and we compared across social forms (single-queen versus multiple-queen) using *SB/SB* samples only and corrected with the p-values Bonferroni correction. In the worker dataset, *SB/Sb* samples originate from both single- and multiple-queen colonies, so we also tested whether any module eigengene is correlated with social form in this dataset.

Gene Ontology (GO) term annotation of the *Solenopsis invicta* genome

We used the modified *S. invicta* annotations created for the RNA-seq alignments (above) as a query for blastp against the nr database of NCBI. We limited the hits to the 20 best matches, with a minimum e-value of 10^{-5}. The results were used with Blast2GO (version 4.1.9; (Conesa *et al.*, 2005)) to obtain the GO terms for each protein coding gene in *S.invicta*.

We tested whether any GO terms were overrepresented in any co-expression modules that were significantly correlated with social form or genotype using TopGO (version 2.26.0 (Alexa and Rahnenführer, 2009)) with the 'elim' algorithm and a Fisher's exact test, with p-values corrected for multiple-testing (Benjamini and Hochberg, 1995).

Evidence for selection based on nucleotide diversity

Genomic regions that underwent recent selective sweeps are characterised by low nucleotide diversity (π) (Smith and Haigh, 1974; Nei, 1987; Nachman, 2001). We used measurements of π along a sliding window of the genome, originally produced by Pracana et al. (2017), to identify selection pressure acting on *S. invicta* OBPs. These measurements were produced from SNPs identified de novo from the 7 SB samples mentioned above and an additional SB sample (NCBI SAMN00014755, ~33x coverage) using Cortex (version 1.0.5.20 (Iqbal *et al.*, 2012)). Measurements of π were taken from non-overlapping 10kb windows. *Sb* samples were excluded to avoid measuring diversity across sibling pairs, and because of the very low diversity in the Sb supergene variant ($\pi \approx 0$), which may be the result of low recombination in Sb and a putative recent fixation of this variant in the sampled population (Pracana, Priyam, *et al.*, 2017).

Supplementary figures

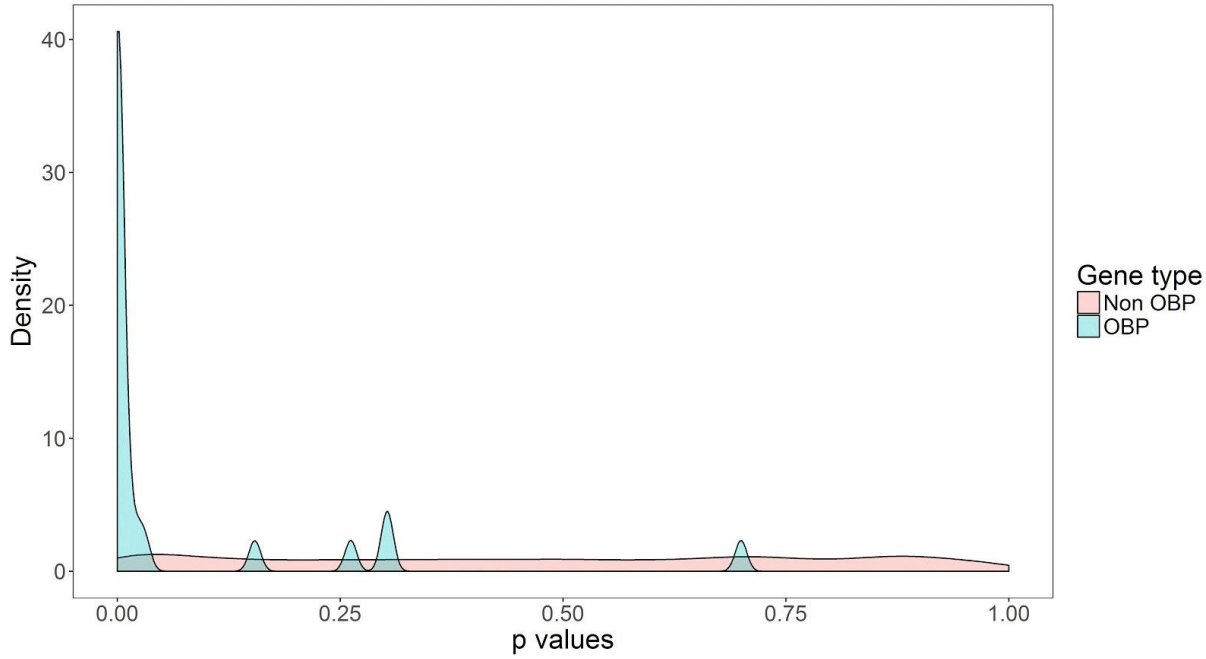


Figure A2.S1: Density distribution of p-values for differential expression between social forms

Density distribution of the p-values for differential expression between social forms in queens for OBPs (in green) and all other protein-coding genes (red). The p-values for OBPs are strongly skewed towards 0. This result is based on the expression levels from the Morandin et al. (2016) dataset.

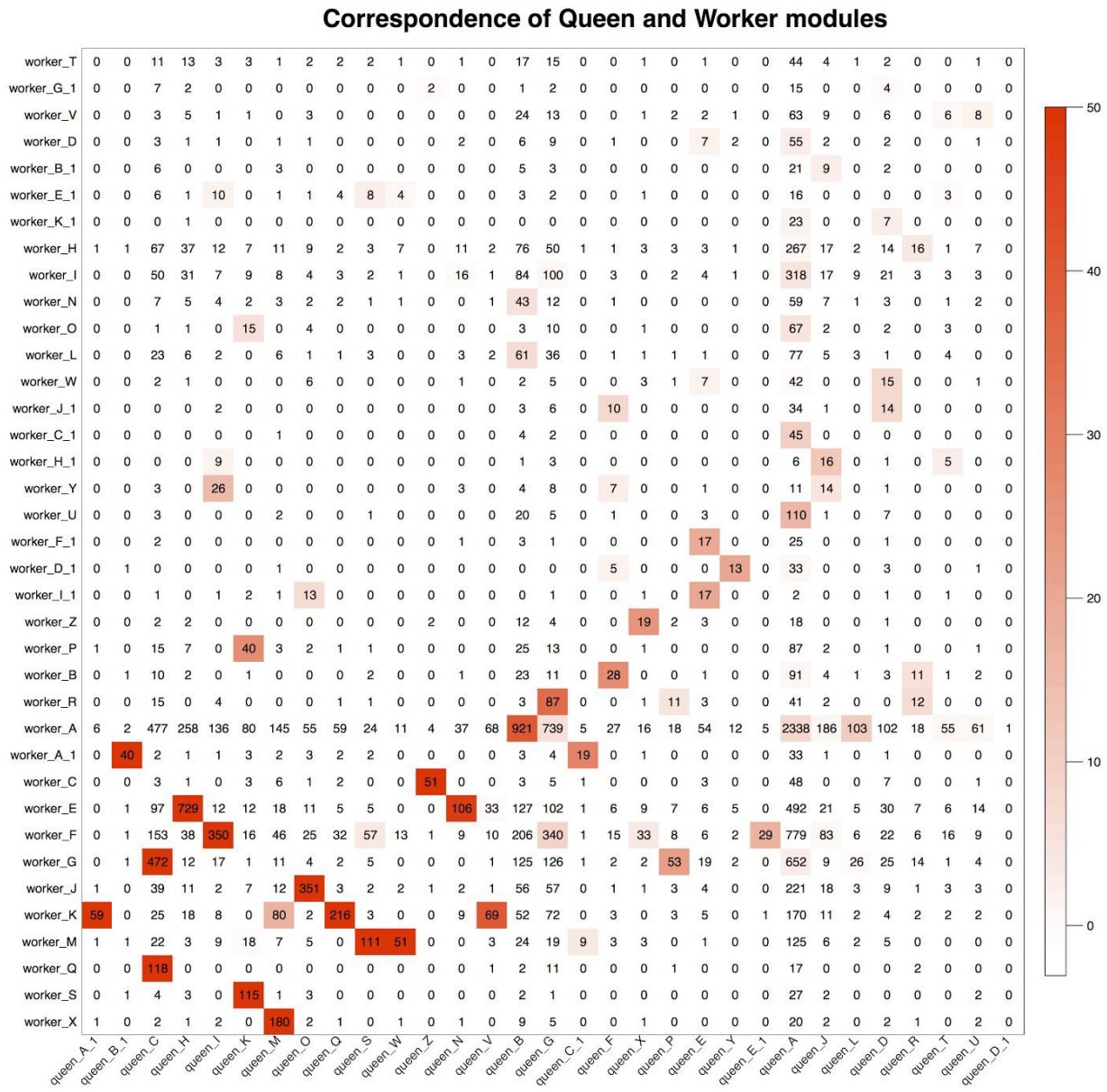


Figure A2.S2: Correspondence between queen and worker modules.

Numbers in the table indicate probe counts in the intersection of the corresponding modules. Coloring of the table encodes – log(p), with p being the Fisher's exact test p-value for the overlap of the two modules. A module in one dataset would be preserved across both sets if it had a single corresponding module in the other dataset with a large number of probes in common.

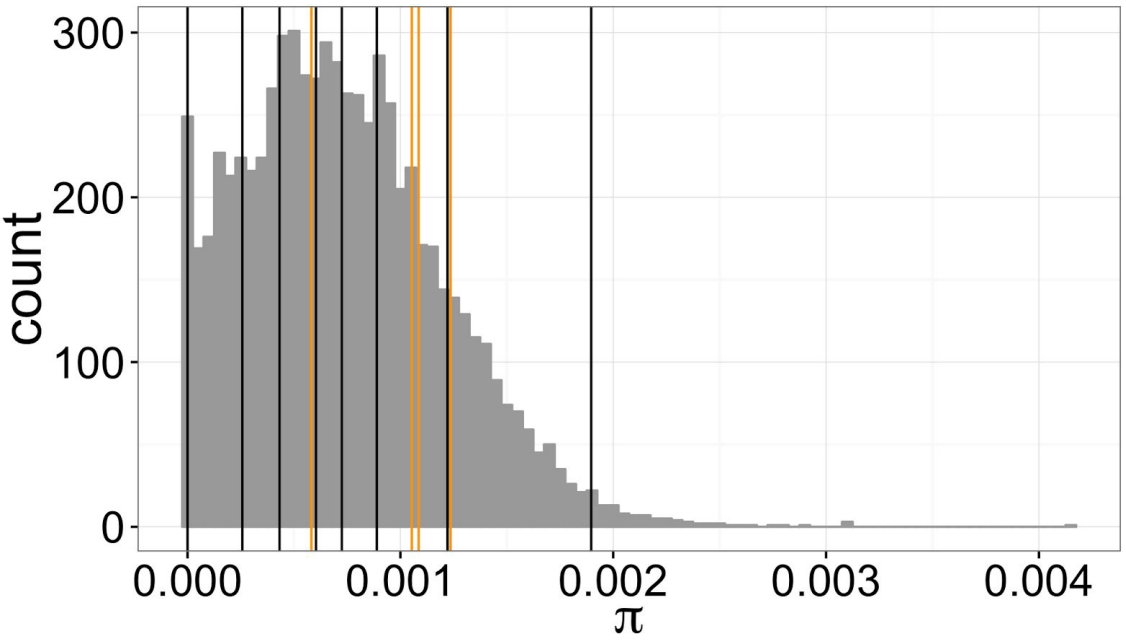


Figure A2.S3: Nucleotide diversity along the genome

Nucleotide distribution (π, measured from *SB* individuals in (Pracana, Priyam, *et al.*, 2017)) of 10kb windows of the assembled genome that overlap coding sequences. Vertical bars represent π of windows overlapping OBPs; orange bars representing those overlapping supergene OBPs.

Supplementary tables

Table A2.S1: Correspondence between presented and previously published sequences

Summary of correspondences between identifiers of sequences produced in this project and previously published sequences, including the number of sequence differences between the two groups.

New annotation		Gotzek et al. 2011			Other publication			
Name	CDS length	Accession	Nucleotide mismatch	Amino acid mismatch	Accession	Nucleotide mismatch	Amino acid mismatch	Publication
SiOBP1	417	HQ853350	0	0	FJ215314	0	0	Xu et al. 2009
SiOBP2	456	HQ853351	0	0	-	-	-	-
SiOBP3/ Gp-9	459	(not submitted)	-	-	AF427893	0	0	Krieger & Ross 2002
SiOBP4	459	HQ853352	1	0	-	-	-	-
SiOBP5	432	HQ853353	0	0	-	-	-	-
SiOBP6	438	HQ853354	1	0	-	-	-	-
SiOBP7	399	HQ853355	0	0	EFZ13147	43	14	Wurm et al. 2011
SiOBP8	459	HQ853356	0	0	-	-	-	-
SiOBP9	387	HQ853357	0	0	EFZ09576	45	22	Wurm et al. 2011
SiOBP10	429	HQ853358	12	4	FJ215319	85	14	Xu et al. 2009
SiOBP11	447	HQ853359	1	0	EFZ10447	90	30	Wurm et al. 2011
SiOBP12	525	HQ853360	9	5	FJ215315	8	4	Xu et al. 2009
SiOBP13	480	HQ853361	0	0	FJ215318	1	frameshift	Xu et al. 2009
SiOBP14	486	HQ853362	1	1	-	-	-	-
SiOBP15	486	HQ853363	4	2	FJ215316	5	frameshift	Xu et al. 2009
SiOBP16	513	HQ853364	2	2	FJ215317	112	51	Xu et al. 2009
SiOBP17	339/366/288/246	(not submitted)	-	-	-	-	-	-
SiOBP18	-	(not submitted)	-	-	-	-	-	-

Table A2.S2: Accession numbers of the gene expression data used.

"Project" and "SRA" columns indicate NCBI identifiers. The descriptions provided and the sequencing method used are based on metadata available on NCBI and in the manuscripts. Two samples (marked with an asterisk) were discarded because of very low coverage after aligning the reads to the *S. invicta* genome.

Publication	Project	Sequencing method	SRA	Description
Morandin et al 2016	PRJDB4088	Illumina paired-end	DRS023318	Pool of 9 polygyne workers 1
			DRS023319	Pool of 9 polygyne workers 2
			DRS023320	Pool of 9 polygyne workers 3
			DRS023315	Pool of 3 polygyne queen 1
			DRS023316*	Pool of 3 polygyne queen 2
			DRS023317	Pool of 3 polygyne queen 3
			DRS023312	Pool of 9 monogyne workers 1
			DRS023313*	Pool of 9 monogyne workers 2
			DRS023314	Pool of 9 monogyne workers 3
			DRS023309	Pool of 3 monogyne queen 1
			DRS023310	Pool of 3 monogyne queen 2
			DRS023311	Pool of 3 monogyne queen 3
Wurm et al 2011	PRJNA49629	454 EST	SRS084972	Pool of polygyne workers and queens from different developmental stages
			SRS084971	Pool of monogyne workers and queens from different developmental stages
			SRS084970	Pool of 100 heads from workers and queens
			SRS084969	Pool of 24 adult males
		Illumina single-end	SRS377035	4 pooled polygyne queens 1
			SRS376911	4 pooled polygyne queens 2
			SRS376910	4 pooled polygyne queens 3
			SRS376905	4 pooled polygyne queens 4
			SRS376904	4 pooled polygyne queens 5
			SRS376903	4 pooled polygyne queens 6

NA	**PRJNA266847**	**Illumina paired-end**	SRS742422	Antennae from queen, age 1
			SRS742424	Antennae from queen, age 2
			SRS742423	Antennae from queen, age 3

Table A2.S3: Closest BLASTP hit of newly produced *S. invicta* OBP sequences in NCBI "nr" database

***S. invicta* OBP**	**Blast hit organism**	**Blast hit description**	**Blast hit e-value**	**Blast hit identity**	**Blast hit query coverage**	**Blast hit accession**
SiOBP1	*S. invicta*	general odorant-binding protein 69a-like precursor	3.41E-98	100.00%	100.00%	NP_001291522
SiOBP2	*S. invicta*	OBP2 precursor, partial	5.71E-105	100.00%	100.00%	ADX94399
SiOBP3 (Gp-9)	*S. invicta*	PREDICTED: pheromone-binding protein Gp-9	3.48E-109	100.00%	100.00%	XP_011157711
SiOBP4	*S. invicta*	PREDICTED: pheromone-binding protein Gp-9	3.36E-110	100.00%	100.00%	XP_011157725
SiOBP5	*S. invicta*	PREDICTED: general odorant-binding protein 72, partial	1.06E-103	100.00%	100.00%	XP_011156042
SiOBP6	*S. invicta*	PREDICTED: general odorant-binding protein lush, partial	2.70E-104	100.00%	100.00%	XP_011165204
SiOBP7	*S. invicta*	PREDICTED: general odorant-binding protein 56d-like	3.98E-93	100.00%	100.00%	XP_011167532
SiOBP8	*S. invicta*	PREDICTED: uncharacterized protein LOC105203183	1.34E-110	100.00%	100.00%	XP_011170254
SiOBP9	*S. invicta*	PREDICTED: general odorant-binding protein 83a-like	5.12E-88	100.00%	100.00%	XP_011173007
SiOBP10	*S. invicta*	PREDICTED: general odorant-binding protein 69a-like isoform X2	1.76E-100	100.00%	102.80%	XP_011171270
SiOBP11	*S. invicta*	PREDICTED: pheromone-binding protein-related protein 6	1.31E-106	100.00%	100.00%	XP_011171271
SiOBP12	*S. invicta*	OBP12 precursor, partial	6.40E-117	97.70%	100.00%	ADX94408
SiOBP13	*S. invicta*	PREDICTED: pheromone-binding protein Gp-9-like	3.40E-112	100.00%	100.00%	XP_011157738
SiOBP14	*S. invicta*	PREDICTED: pheromone-binding protein Gp-9-like isoform X1	1.24E-115	99.40%	99.40%	XP_011171390

SiOBP15	*S. invicta*	PREDICTED: pheromone-binding protein Gp-9-like	2.67E-116	100.00%	100.00%	XP_011169815
SiOBP16	*S. invicta*	PREDICTED: pheromone-binding protein Gp-9-like	2.29E-117	100.00%	100.00%	XP_011169816
SiOBP17.1	*S. invicta*	OBP16 precursor, partial	1.31E-14	41.50%	79.30%	ADX94412
SiOBP17.2	*S. invicta*	OBP16 precursor, partial	1.28E-14	35.40%	67.70%	ADX94412
SiOBP17.4	*S. invicta*	OBP16 precursor, partial	5.71E-33	45.90%	84.40%	ADX94412
SiOBP17.7	*S. invicta*	OBP16 precursor, partial	4.22E-26	46.00%	91.20%	ADX94412
SiOBPZ1	*S. invicta*	PREDICTED: general odorant-binding protein 71	0	100.00%	100.00%	XP_011161522
SiOBPZ2	*S. invicta*	PREDICTED: uncharacterized protein LOC105199157	5.54E-90	100.00%	100.00%	XP_011164418
SiOBPZ3	*S. invicta*	PREDICTED: uncharacterized protein LOC105202825	9.07E-111	100.00%	100.00%	XP_011169818
SiOBPZ4	*S. invicta*	PREDICTED: general odorant-binding protein 56d-like	1.55E-63	73.10%	100.80%	XP_011167532
SiOBPZ5	*S. invicta*	OBP12 precursor, partial	4.81E-92	87.40%	100.60%	ADX94408
SiOBPZ6	*S. invicta*	PREDICTED: uncharacterized protein LOC105203183	1.67E-100	92.20%	100.00%	XP_011170254
SiOBPZ7.1	*S. invicta*	PREDICTED: general odorant-binding protein 56d-like	1.89E-67	64.10%	73.10%	XP_011161856
SiOBPZ7.2	*S. invicta*	PREDICTED: general odorant-binding protein 56d-like	1.48E-66	89.30%	99.10%	XP_011161856

Table A2.S4: Number of genes represented in each co-expression module

Number of genes represented in each co-expression module; OBPs represented in each module. Each gene may be represented by multiple probes. Probes in queen_E_1 are not assigned to a module.

Dataset	Module	Probe Number	Gene number	OBP Number	OBP Probes as % of Total	OBPs Represented
queen	-	18291	3046	8	0.26	*SiOBP1,SiOBP10,SiOBP12,SiOBP13,SiOBP15,SiOBP16,SiOBP3,SiOBP7*
worker	-	18291	3046	8	0.26	*SiOBP1,SiOBP10,SiOBP12,SiOBP13,SiOBP15,SiOBP16,SiOBP3,SiOBP7*
queen	queen_A	6492	1342	5	0.37	*SiOBP1,SiOBP12,SiOBP13,SiOBP16,SiOBP3*
queen	queen_B	1953	561	2	0.36	*SiOBP1,SiOBP10*

queen	queen_C	1651	348	1	0.29	*SiOBP1*
queen	queen_D	319	75	1	1.33	*SiOBP15*
queen	queen_E	168	39	1	2.56	*SiOBP7*
queen	queen_F	115	15	1	6.67	*SiOBP15*
worker	worker_A	6023	1599	2	0.13	*SiOBP1,SiOBP10*
worker	worker_B	193	27	1	3.7	*SiOBP15*
worker	worker_C	135	24	1	4.17	*SiOBP7*
worker	worker_D	94	16	4	25	*SiOBP12,SiOBP13,SiOBP16,SiOBP3*
queen	queen_G	1879	454	0	0	-
queen	queen_H	1190	296	0	0	-
queen	queen_I	619	109	0	0	-
queen	queen_J	451	102	0	0	-
queen	queen_K	335	77	0	0	-
queen	queen_L	164	72	0	0	-
queen	queen_M	550	63	0	0	-
queen	queen_N	203	62	0	0	-
queen	queen_O	510	41	0	0	-
queen	queen_P	115	34	0	0	-
queen	queen_Q	338	29	0	0	-
queen	queen_R	93	24	0	0	-
queen	queen_S	233	24	0	0	-
queen	queen_T	111	22	0	0	-
queen	queen_U	125	22	0	0	-
queen	queen_V	192	16	0	0	-
queen	queen_W	92	11	0	0	-
queen	queen_X	99	10	0	0	-
queen	queen_Y	39	8	0	0	-
queen	queen_Z	61	7	0	0	-
queen	queen_A_1	70	5	0	0	-
queen	queen_B_1	50	4	0	0	-
queen	queen_C_1	38	1	0	0	-
queen	queen_D_1	1	1	0	0	-
queen	queen_E_1	35	0	0	NA	-
worker	worker_E	1867	486	0	0	-
worker	worker_F	2312	408	0	0	-
worker	worker_G	1587	401	0	0	-
worker	worker_H	632	217	0	0	-

worker	worker_I	700	195	0	0	-
worker	worker_J	814	144	0	0	-
worker	worker_K	818	135	0	0	-
worker	worker_L	238	99	0	0	-
worker	worker_M	428	88	0	0	-
worker	worker_N	157	55	0	0	-
worker	worker_O	109	46	0	0	-
worker	worker_P	200	45	0	0	-
worker	worker_Q	152	44	0	0	-
worker	worker_R	181	43	0	0	-
worker	worker_S	161	38	0	0	-
worker	worker_T	125	36	0	0	-
worker	worker_U	153	30	0	0	-
worker	worker_V	148	25	0	0	-
worker	worker_W	86	24	0	0	-
worker	worker_X	233	16	0	0	-
worker	worker_Y	78	13	0	0	-
worker	worker_Z	65	12	0	0	-
worker	worker_A_1	118	11	0	0	-
worker	worker_B_1	49	11	0	0	-
worker	worker_C_1	52	10	0	0	-
worker	worker_D_1	57	10	0	0	-
worker	worker_E_1	60	10	0	0	-
worker	worker_F_1	50	9	0	0	-
worker	worker_G_1	33	8	0	0	-
worker	worker_H_1	41	6	0	0	-
worker	worker_I_1	41	6	0	0	-
worker	worker_J_1	70	6	0	0	-
worker	worker_K_1	31	5	0	0	-

Table A2.S5: Gene co-expression modules

Gene co-expression modules with module eigene differential expression between genotypes in queens (SB/SB versus SB/Sb), between genotypes in workers from multiple-queen colonies (SB/SB versus SB/Sb), and between social forms in SB/SB workers (single-queen colony versus multiple-queen colony). Differential expression was tested with t-tests within each comparison within each dataset, with p-values corrected for multiple testing using Bonferroni correction.

Dataset	Module	Comparison	Gene Number	Mean Difference	t	d.f.	p-value	Corrected p-value
queen	queen_X	genotype	10	0.12	-17.52	48.99	9.73E-23	3.02E-21
queen	queen_E_1	genotype	0	0.06	-4.2	58.75	9.21E-05	2.86E-03
queen	queen_Y	genotype	8	0.05	-3.72	50.23	5.10E-04	1.58E-02
queen	queen_D	genotype	75	0.05	-3.42	51.91	1.22E-03	3.77E-02
worker	worker_Z	genotype	12	-0.17	18.18	37.31	3.95E-20	1.46E-18
worker	worker_A_1	social_form	11	-0.08	11.11	29.18	5.35E-12	1.98E-10

Table A2.S6: Putative OBP orthologs in other species.

First, we ran a tblastn search of all *S. invicta* OBPs against all non-*S. invicta* arthropod sequences, accepting hits where e-value < 1x10-3. We then ran a blastx search of these hits against the *S. invicta* gene predictions (including our newly curated OBP set). We report the hits with the lowest e-value of the blastx search. We repeated this analysis by searching non-ant arthropods (not Formicidae). The coloured cells represent cases where the same non-*S. invicta* sequence aligns to multiple *S. invicta* OBPs.

	Grouping	OBP	Accession	Sequence Name	OBP coverage	e-value
S. invicta OBPs against all non-*S. invicta* arthropod sequences		**SiOBP1**	XM_012373322.1	PREDICTED: *Linepithema humile* general odorant-binding protein 69a-like (LOC105675856), transcript variant X1, mRNA	100.00%	2.02E-101
		SiOBP2	XM_012675063.1	PREDICTED: *Monomorium pharaonis* uncharacterized LOC105833365 (LOC105833365), mRNA	98.68%	2.06E-74
		SiOBP3	AF427903.1	*Solenopsis richteri* putative odorant binding protein precursor (Gp-9) gene, Gp-9-b' allele, complete cds	100.00%	4.57E-113
		SiOBP4	AY818614.1	*Solenopsis sp.* Bo_178 putative odorant binding protein precursor (Gp-9) gene, Gp-9-B allele, complete cds	100.00%	3.39E-103
		SiOBP5	XM_012671527.1	PREDICTED: *Monomorium pharaonis* general odorant-binding protein 19a-like (LOC105831417), mRNA	100.00%	2.41E-93
		SiOBP6	XM_012670912.1	PREDICTED: *Monomorium pharaonis* general odorant-binding protein 99b-like (LOC105831060), mRNA	100.00%	2.42E-82

		SiOBP9	XM_011693717.1	PREDICTED: *Wasmannia auropunctata* general odorant-binding protein 83a-like (LOC105452530), transcript variant X1, mRNA	88.37%	1.05E-73
		SiOBP10	XM_012684740.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein-related protein 6-like (LOC105838856), mRNA	100.00%	4.69E-103
		SiOBP11	XM_011692838.1	PREDICTED: *Wasmannia auropunctata* general odorant-binding protein 83a-like (LOC105452039), mRNA	100.00%	1.64E-78
		SiOBPZ1	XM_012019767.1	PREDICTED: *Vollenhovia emeryi* general odorant-binding protein 71 (LOC105566056), transcript variant X2, mRNA	93.28%	2.04E-125
	Phylogenetic Group 1	**SiOBPZ3**	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	100.00%	7.12E-61
		SiOBP16	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	77.78%	1.86E-52
		SiOBP14	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	89.51%	7.72E-45
		SiOBP13	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	91.25%	5.92E-44
		SiOBP17-mRNA.4	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	80.33%	5.48E-35
		SiOBP17-mRNA.7	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	78.76%	8.67E-28
		SiOBP17-mRNA.2	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	76.04%	1.48E-17
		SiOBP17-mRNA.1	XM_012679489.1	PREDICTED: *Monomorium pharaonis* pheromone-binding protein Gp-9-like (LOC105835869), mRNA	73.17%	6.24E-17
		SiOBP12	XM_012669319.1	PREDICTED: *Monomorium pharaonis* uncharacterized LOC105830146 (LOC105830146), mRNA	85.14%	4.73E-47

		SiOBP15	XM_011070365.1	PREDICTED: *Acromyrmex echinatior* pheromone-binding protein Gp-9-like (LOC105154687), mRNA	95.06%	1.67E-43
		SiOBPZ5	XM_012678755.1	PREDICTED: *Monomorium pharaonis* uncharacterized LOC105835453 (LOC105835453), mRNA	85.06%	1.52E-43
	Phylogenetic Group 2	**SiOBP7**	XM_011693711.1	PREDICTED: *Wasmannia auropunctata* B2 protein-like (LOC105452527), transcript variant X2, mRNA	99.25%	1.59E-76
		SiOBPZ4	XM_011693711.1	PREDICTED: *Wasmannia auropunctata* B2 protein-like (LOC105452527), transcript variant X2, mRNA	97.69%	1.03E-61
		SiOBPZ2	XM_011693711.1	PREDICTED: *Wasmannia auropunctata* B2 protein-like (LOC105452527), transcript variant X2, mRNA	96.18%	2.05E-53
		SiOBP8	XM_011645916.1	PREDICTED: *Pogonomyrmex barbatus* general odorant-binding protein 56d-like (LOC105431629), partial mRNA	73.20%	3.60E-29
		SiOBPZ6	XM_011645916.1	PREDICTED: *Pogonomyrmex barbatus* general odorant-binding protein 56d-like (LOC105431629), partial mRNA	73.20%	4.52E-29
		SiOBPZ7-mRNA.1	XM_012685128.1	PREDICTED: *Monomorium pharaonis* general odorant-binding protein 56d-like (LOC105839081), mRNA	69.23%	9.17E-29
		SiOBPZ7-mRNA.2	XM_012685128.1	PREDICTED: *Monomorium pharaonis* general odorant-binding protein 56d-like (LOC105839081), mRNA	91.07%	1.47E-28
S. invicta OBPs against non-ant arthropods (not		**SiOBP1**	XM_015321671.1	PREDICTED: *Polistes dominula* uncharacterized LOC107066747 (LOC107066747), mRNA	97.12%	2.28E-70
		SiOBP2	XM_015656912.1	PREDICTED: *Neodiprion lecontei* general odorant-binding protein 19d-like (LOC107218882), mRNA	62.50%	6.19E-04
		SiOBP3	HE578203.1	*Nasonia vitripennis* OBP18 gene for putative odorant binding protein 18, strain AsmCX	94.77%	1.17E-08
		SiOBP4	XM_015580922.1	PREDICTED: *Dufourea novaeangliae* uncharacterized LOC107191806 (LOC107191806), mRNA	90.20%	3.94E-10
		SiOBP5	KP963692.1	*Sclerodermus sp.* MQW-2015 odorant binding protein 7 (obp6) mRNA, complete cds	100.00%	1.61E-46
		SiOBP6	XM_015327234.1	PREDICTED: *Polistes dominula* general odorant-binding	93.15%	9.51E-26

				protein 56d-like (LOC107069711), mRNA		
		SiOBP9	XM_015579325.1	PREDICTED: *Dufourea novaeangliae* general odorant-binding protein 56d-like (LOC107190515), mRNA	85.27%	5.28E-53
		SiOBP10	KP717060.1	*Apis cerana cerana* odorant binding protein 10 mRNA, complete cds	89.51%	3.14E-88
		SiOBP11	XM_017934140.1	PREDICTED: *Habropoda laboriosa* general odorant-binding protein 99b-like (LOC108571970), mRNA	97.99%	2.89E-46
		SiOBPZ1	XM_015749472.1	PREDICTED: *Cephus cinctus* general odorant-binding protein 71-like (LOC107272374), mRNA	99.21%	2.12E-53
	Phylogenetic Group 1	**SiOBPZ3**	XM_017910159.1	PREDICTED: *Eufriesea mexicana* uncharacterized LOC108554787 (LOC108554787), mRNA	76.43%	1.75E-15
		SiOBP16	XM_018027175.1	PREDICTED: *Ceratina calcarata* uncharacterized LOC108626482 (LOC108626482), mRNA	50.29%	5.29E-10
		SiOBP14	XM_018027175.1	PREDICTED: *Ceratina calcarata* uncharacterized LOC108626482 (LOC108626482), mRNA	89.51%	1.11E-14
		SiOBP13	XM_006608556.1	PREDICTED: *Apis dorsata* uncharacterized LOC102678956 (LOC102678956), mRNA	71.25%	4.89E-12
		SiOBP17-mRNA.4	KT965294.1	*Melipona scutellaris* odorant-binding protein-4 mRNA, partial cds	63.93%	1.67E-08
		SiOBP17-mRNA.7	FN432786.1	*Glossina morsitans morsitans* mRNA for odorant binding protein 8 (obp8 gene), isolate Gmm_1	58.41%	1.06E-03
		SiOBP17-mRNA.2	FN432786.1	*Glossina morsitans morsitans* mRNA for odorant binding protein 8 (obp8 gene), isolate Gmm_1	56.25%	1.89E-03
		SiOBP17-mRNA.1	FN432786.1	*Glossina morsitans morsitans* mRNA for odorant binding protein 8 (obp8 gene), isolate Gmm_1	65.85%	1.98E-03
		SiOBP12	XM_015580922.1	PREDICTED: *Dufourea novaeangliae* uncharacterized LOC107191806 (LOC107191806), mRNA	82.29%	4.06E-11
		SiOBP15	XM_018027175.1	PREDICTED: *Ceratina calcarata* uncharacterized	91.98%	3.08E-14

				LOC108626482 (LOC108626482), mRNA		
		SiOBPZ5	XM_006608556.1	PREDICTED: *Apis dorsata* uncharacterized LOC102678956 (LOC102678956), mRNA	50.57%	2.73E-10
	Phylogenetic Group 2	**SiOBP7**	XM_015747897.1	PREDICTED: *Cephus cinctus* general odorant-binding protein 56d-like (LOC107271663), mRNA	99.25%	1.18E-51
		SiOBPZ4	XM_015323907.1	PREDICTED: *Polistes dominula* general odorant-binding protein 56d-like (LOC107067956), mRNA	95.38%	3.43E-44
		SiOBPZ2	XM_015323907.1	PREDICTED: *Polistes dominula* general odorant-binding protein 56d-like (LOC107067956), mRNA	93.89%	2.80E-39
		SiOBP8	XM_015323907.1	PREDICTED: *Polistes dominula* general odorant-binding protein 56d-like (LOC107067956), mRNA	97.39%	3.58E-19
		SiOBPZ6	XM_015323907.1	PREDICTED: *Polistes dominula* general odorant-binding protein 56d-like (LOC107067956), mRNA	97.39%	1.39E-17
		SiOBPZ7-mRNA.1	XM_017898310.1	PREDICTED: *Eufriesea mexicana* general odorant-binding protein 56h-like (LOC108546299), mRNA	69.23%	4.36E-18
		SiOBPZ7-mRNA.2	XM_003708502.2	PREDICTED: *Megachile rotundata* general odorant-binding protein 56a-like (LOC100883755), mRNA	92.86%	5.19E-18

Annex 3: Supplementary information to chapter 6

Supplementary figures

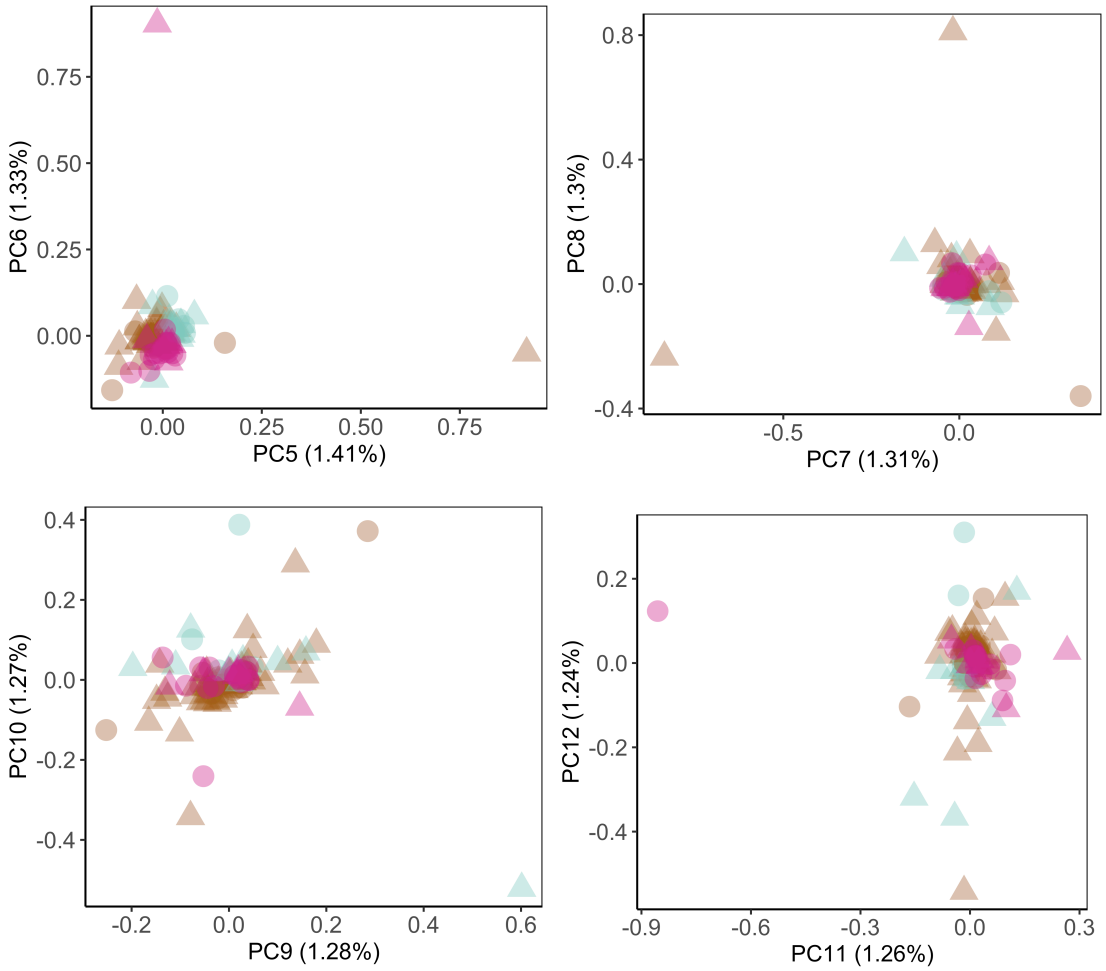


Figure A3.S1: PCA for minor PCs

PCA for the whole dataset (121,786 within-population polymorphic SNPs, supported by 75% of samples; 108 samples from Bruniquel (France), Vigliano, Iberia (France and Spain); analysis from variance-standardised relationship matrix.

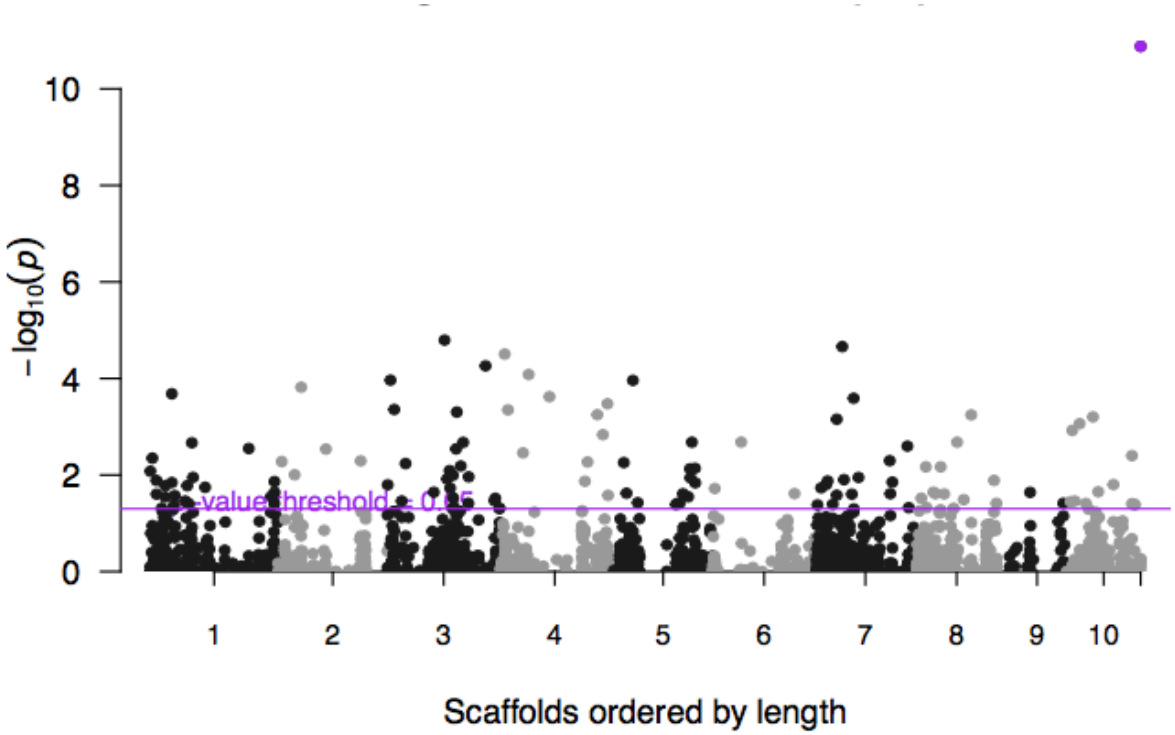


Figure A3.S2: Purple simulated SNP is the most significant variant in Fisher's exact test
Monogynous samples are homozygous at this locus, polygynous samples are heterozygous.

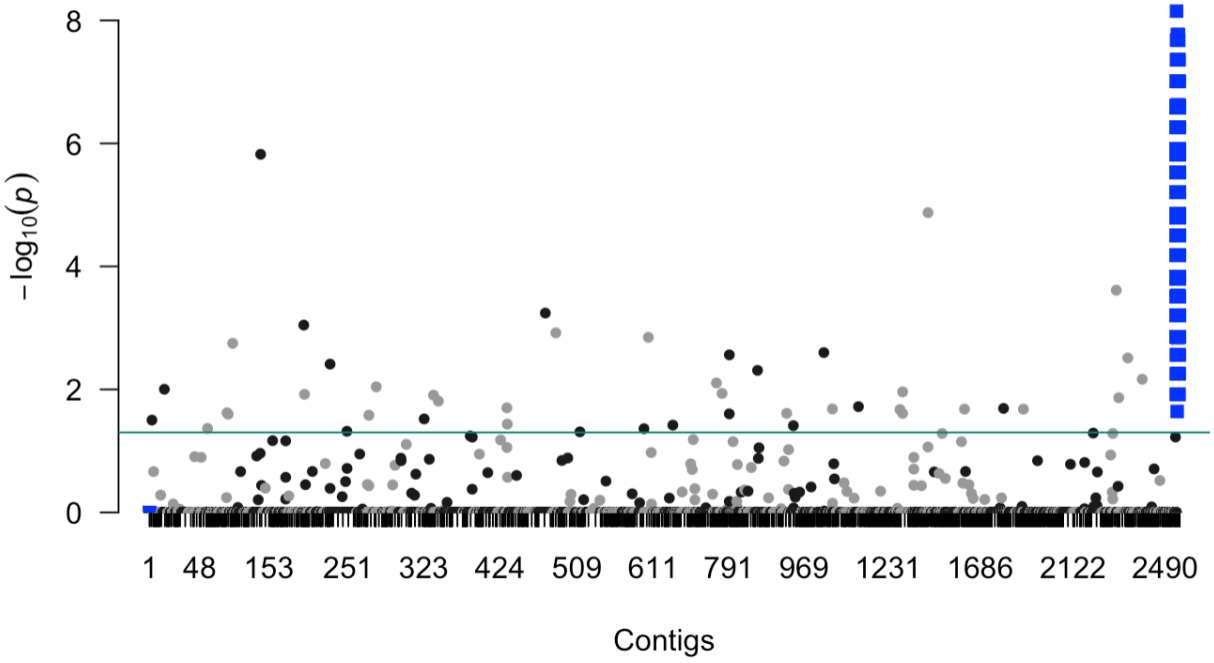


Figure A3.S3: *Solenopsis invicta* simulation

Manhattan plot from Fisher's exact test of allele count (association test for social type).

Total number of SNPs: 124, 840 (121,786 real and 3,054 simulated).

The 121,786 real Pheidole SNPs are from the main text analysis: supported by 75% samples, within-population polymorphic.

The 3,054 simulated SNPs (blue squares) reflect Solenopsis system: all monogynous samples are homozygous for the reference at the 3,054 simulated loci. A third of the polygynous samples are homozygous for the reference and two-thirds are heterozygous.

The adjustment for multiple comparisons is Bonferroni.

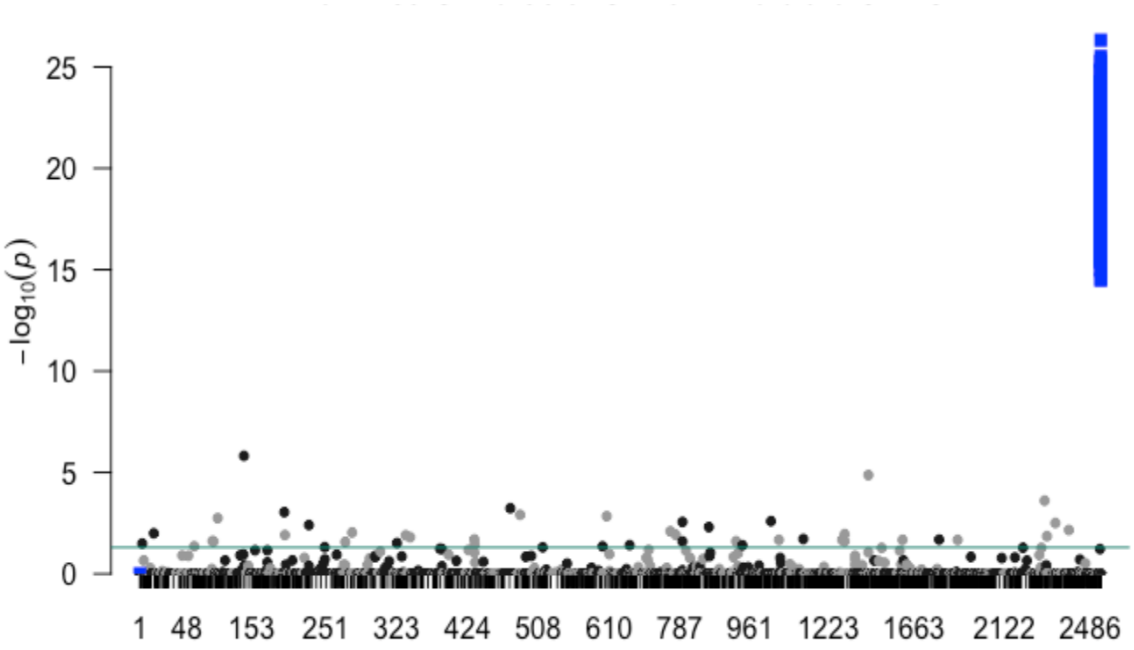


Figure A3.S4: *Formica selysi* simulation

Manhattan plot from Fisher's exact test of allele count (association test for social type).

Total number of SNPs: 126,291 (121,786 real and 4,505 simulated).

The 121,786 real *Pheidole* SNPs are from the main text analysis: supported by 75% samples, within-population polymorphic.

The 4,505 simulated SNPs (blue squares) reflect *Formica* system: all monogynous samples are homozygous for the reference at the 4,505 simulated loci. A third of the polygynous samples are homozygous for the alternative and two-thirds are heterozygous.

The adjustment for multiple comparisons is Bonferroni.

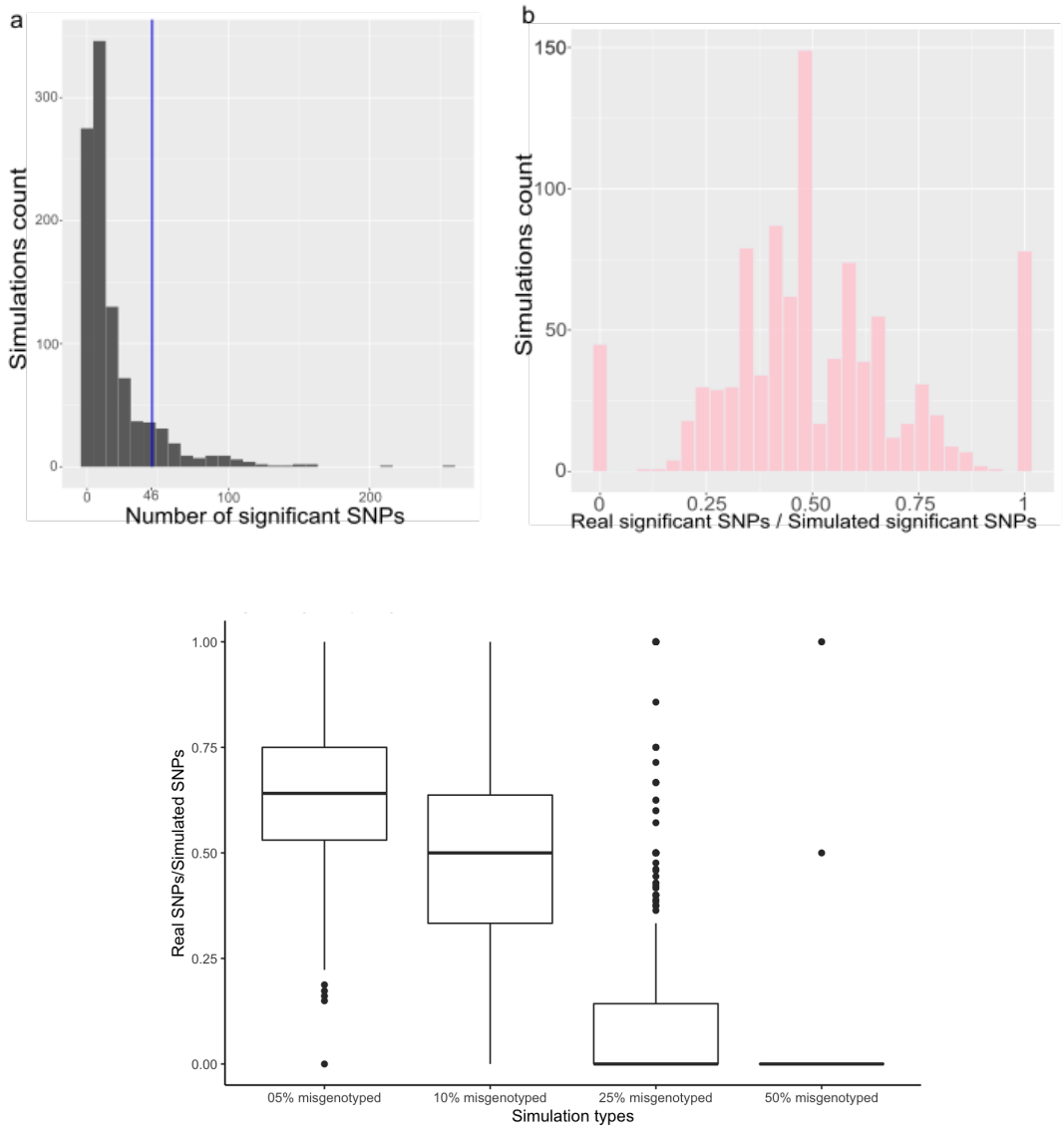


Figure A3.S5: Mis-genotyping simulations

a) Histogram of the numbers of significant SNPs over 1000 simulations, in which 10% of the samples are mis-genotyped (i.e., assigned the alternative social type). Significance is measured by Fisher's exact tests and Bonferroni adjustment. The 46 real SNPs are indicated with the blue line. Most simulations contain less significant SNPs than the real dataset (regardless of the exact loci).

b) The association analysis is powerful enough to include 10% of mislabelling the colonies' social type. Indeed, for more than 950 simulations, the real significant SNPs (n = 46) are recovered in the simulated significant SNPs set.

c) Proportion of real significant SNPs recovered by mis-genotyping simulations for 5% to 50% of samples being assigned the wrong labels. Assuming that our analysis contains up to 10% of mis-genotyping, the analysis design (GWAS using Fisher's exact test and Bonferroni adjustment) is expected to recover at least 50% of true positive significant SNPs.

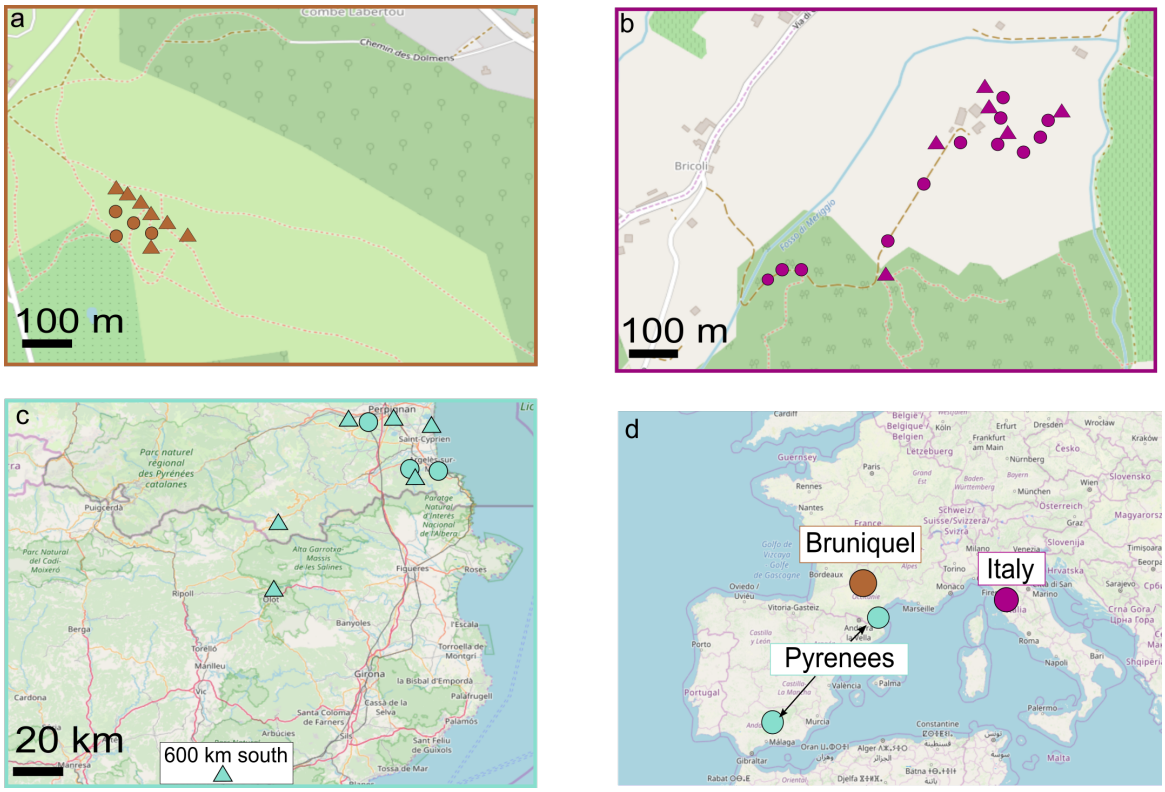


Figure A3.S6: Geographical location map of samples.

a) Bruniquel population: 53 polygynous and 16 monogynous samples.

b) Italian population: 7 polygynous and 16 monogynous samples.

c) Iberian population: 11 polygynous and 5 monogynous samples.

d) Overview of the populations.

Background map © OpenStreetMap contributors.

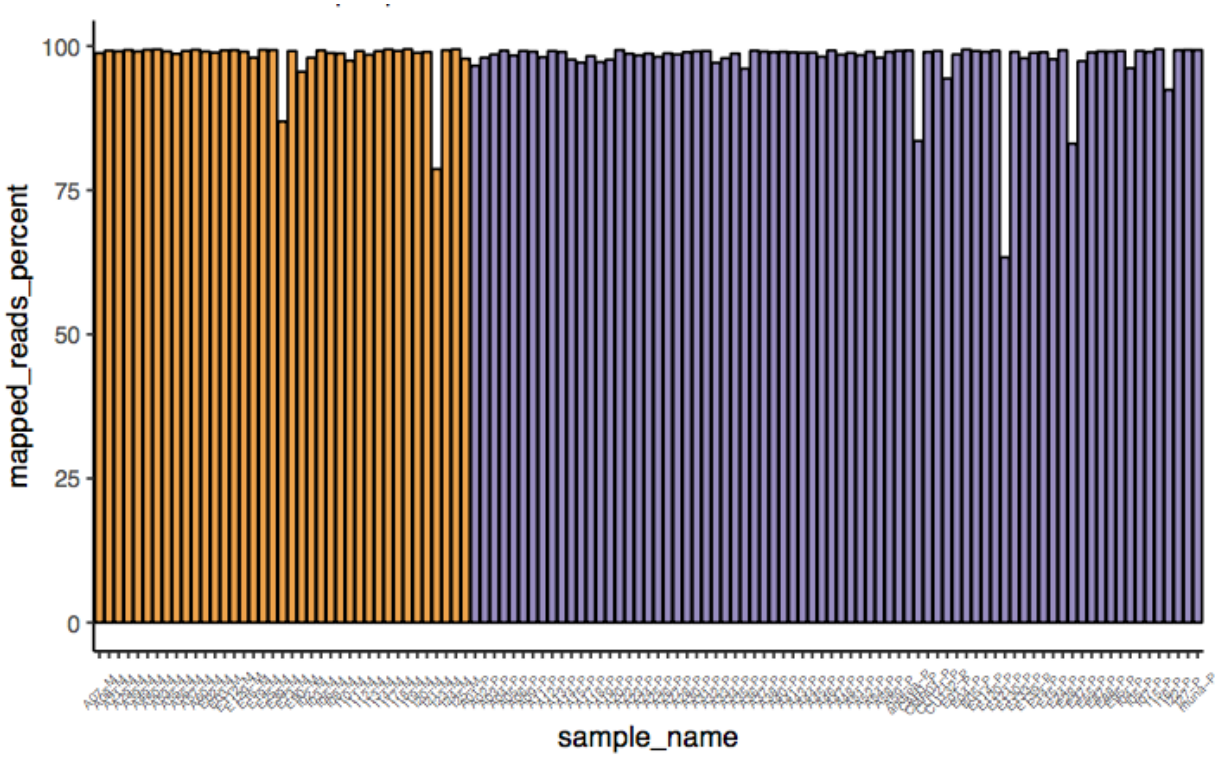


Figure A3.S7: Mapped read proportion by social type

Proportion of reads mapping to the reference for single-queen samples in orange and multiple-queen samples in purple. There is no significant difference between the means of proportion of read mapping between social types (T-test $P = 0.58$).

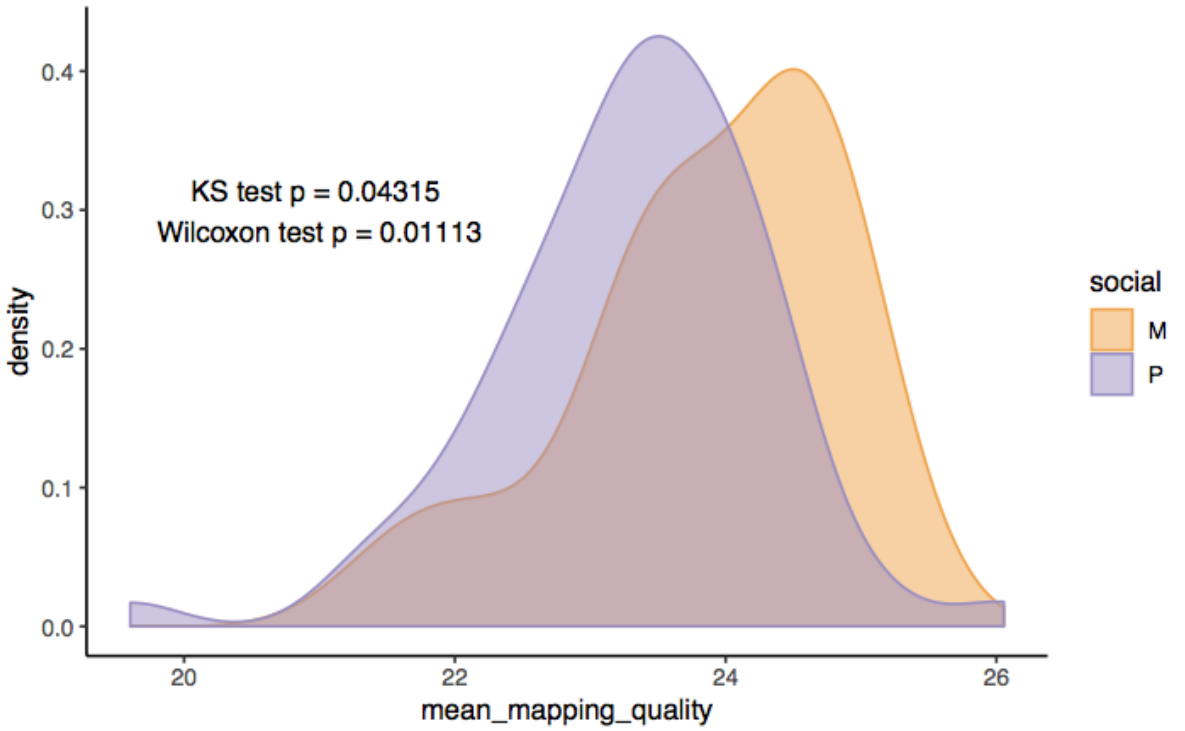


Figure A3.S8: Mean mapping quality by social type

Data from Qualimap report. The quality of mapping is statistically different between social groups (KS test p = 0.002, Wilcoxon test p = 0.0003). We hypothesize that the genome reference influences these results. We thus subset the data for the samples that share geographical origins with the genome reference (Bruniquel and Iberia, 86 samples), the differences are statistically valid but to a lesser strength (KS test p = 0.04, Wilcoxon test p = 0.01). We conclude that the geographical origin has an impact on these results.

Supplementary tables

Table A3.S1: Comparison of *P. pallidula* reference assembly with Hymenopteran genomes

Assembly Accession	Assembly Name	Organism	Contig N50	Scaffold N50
GCA_009193385.1	Nvit_psr_1	*Nasonia vitripennis* (jewel wasp)	7180486	7180486
GCF_003254395.2	Amel_HAv3.1	*Apis mellifera* (honey bee)	5382476	13619445
GCA_003254395.1	Amel_HAv3	*Apis mellifera* (honey bee)	5381094	13615080
GCA_003314205.1	INRA_AMelMel_1.0	*Apis mellifera mellifera* (German honeybee)	5131172	13573435
GCF_003672135.1	Obir_v5.4	*Ooceraea biroi* (clonal raider ant)	3735272	16888278
GCF_003227725.1	Cflo_v7.5	*Camponotus floridanus* (Florida carpenter ant)	1278439	1585631
GCA_009650705.1	Solenopsis_invicta_SB1.0	Solenopsis invicta (red fire ant)	945877	13114153
GCF_003227715.1	Hsal_v8.5	*Harpegnathos saltator* (Jerdon's jumping ant)	911506	1078644
GCA_009299975.1	ASM929997v1	*Solenopsis invicta* (red fire ant)	874937	16736736
GCF_000599845.2	Tpre_2.0	*Trichogramma pretiosum* (wasps, ants, and bees)	573774	1825723
GCF_000344095.2	Aros_2.0	*Athalia rosae* (coleseed sawfly)	571016	943070
GCF_000612105.2	Oabi_2.0	*Orussus abietinus* (hymenopterans)	508621	612083
GCF_004153925.1	Obicornis_v3	*Osmia bicornis bicornis* (red mason bee)	454559	607766
This study	**Ppal_E**	***Pheidole pallidula***	**452000**	**NA**
GCF_005281655.1	TAMU_Nfulva_1.0	*Nylanderia fulva* (ants)	320712	443094
GCA_009299965.1	ASM929996v1	*Solenopsis invicta* (red fire ant)	278331	11613644
GCA_901521435.1	Clev_1.0	*Crematogaster levior* (ants)	255727	383244
GCF_001272555.1	ASM127255v1	*Dufourea novaeangliae* (bees)	191865	2549405
GCA_002217905.1	Apiscer_1.0	*Apis cerana japonica* (Asiatic honeybee)	179487	180259
GCF_001263575.1	Nlec1.0	*Neodiprion lecontei* (redheaded pine sawfly)	87373	243810
GCA_001263575.2	Nlec1.1	*Neodiprion lecontei* (redheaded pine sawfly)	87373	243810
GCA_900474305.1	Cfun	*Cecidostiba fungosa* (wasps, ants, and bees)	87218	87417
GCF_000204515.1	Aech_3.9	*Acromyrmex echinatior* (Panamanian leafcutter ant)	80630	1110580
GCA_000599845.1	Tpre_1.0	*Trichogramma pretiosum* (wasps, ants, and bees)	78771	3706225
GCF_000599845.1	Tpre_1.0	*Trichogramma pretiosum* (wasps, ants, and bees)	78655	3706225
GCF_000214255.1	Bter_1.0	*Bombus terrestris* (buff-tailed bumblebee)	76043	3506793
GCF_000503995.1	CerSol_1.0	*Ceratosolen solmsi marchali* (wasps, ants, and bees)	74702	9558897

GCF_001594065.1	Ccosl1.0	*Cyphomyrmex costatus* (ants)	74312	1159032
GCA_002156465.1	MCINOGS1.0	*Macrocentrus cingulum* (wasps, ants, and bees)	65089	65089
GCF_000220905.1	MROT_1.0	*Megachile rotundata* (alfalfa leafcutting bee)	64153	1699680
GCF_000188095.3	BIMP_2.2	*Bombus impatiens* (common eastern bumble bee)	59072	1399493
GCF_000188095.2	BIMP_2.1	*Bombus impatiens* (common eastern bumble bee)	59072	1399493
GCF_000188095.1	BIMP_2.0	*Bombus impatiens* (common eastern bumble bee)	58885	1399493
GCA_900474275.1	Synjap	*Synergus japonicus* (wasps, ants, and bees)	55627	61479
GCF_000612105.1	Oabi_1.0	*Orussus abietinus* (hymenopterans)	54038	2372050
GCF_001594055.1	Tzet1.0	*Trachymyrmex zeteki* (ants)	52131	1333945
GCF_000806365.1	ASM80636v1	*Fopius arisanus* (wasps, ants, and bees)	51867	978588
GCF_000344095.1	Aros_1.0	*Athalia rosae* (coleseed sawfly)	51418	1366867
GCA_000956155.1	ASM95615v1	*Cotesia vestalis* (diamondback moth parasitoid)	46055	46055
GCF_000002195.4	Amel_4.5	*Apis mellifera* (honey bee)	45688	997192
GCA_001412515.2	Dall2.0	*Diachasma alloeu*m (wasps, ants, and bees)	45480	657001
GCF_001412515.1	Dall1.0	*Diachasma alloeum* (wasps, ants, and bees)	44932	645483
GCF_000341935.1	Ccin1	*Cephus cinctus* (wheat stem sawfly)	44905	622163
GCF_001442555.1	ACSNU-2.0	*Apis cerana* (Asiatic honeybee)	43751	1421626
GCA_900474325.1	Sumb	*Synergus umbraculus* (wasps, ants, and bees)	42371	49302
GCF_001465965.1	Pdom r1.2	*Polistes dominula* (European paper wasp)	42260	1625592
GCF_000648655.2	Cflo_2.0	*Copidosoma floridanum* (wasps, ants, and bees)	40744	1210516
GCF_003070985.1	ASM307098v1	*Temnothorax curvispinosus* (ants)	38942	223562
GCF_001483705.1	ASM148370v1	*Eufriesea mexicana (bees)*	38936	351926
GCF_002006095.1	ASM200609v1	*Pseudomyrmex gracilis* (ants)	38830	317681
GCF_000147195.1	HarSal_1.0	*Harpegnathos saltator* (Jerdon's jumping ant)	38321	601965
GCF_000956235.1	wasmannia.A_1.0	*Wasmannia auropunctata* (little fire ant)	37912	1175369
GCA_003055095.1	ASM305509v1	*Goniozus legneri* (wasps, ants, and bees)	37816	167330
GCA_003260585.1	UPENN_Mphar_2.0	*Monomorium pharaonis* (pharaoh ant)	35919	18352397
GCF_000217595.1	Lhum_UMD_V04	*Linepithema humile* (Argentine ant)	35858	1402257
GCA_900480045.1	Eadl	*Eurytoma adleriae* (wasps, ants, and bees)	34678	38307

GCF_000611835.1	CerBir1.0	*Ooceraea biroi* (clonal raider ant)	34211	1350650
GCF_000949405.1	V.emery_V1.0	*Vollenhovia emeryi* (ants)	32417	1346088
GCA_900474385.1	Opom	*Ormyrus pomaceus* (wasps, ants, and bees)	30505	30557
GCA_004794745.1	tlon_1.0	*Temnothorax longispinosus* (ants)	30134	514432
GCF_001313825.1	ASM131382v1	*Dinoponera quadriceps* (ants)	29911	1361239
GCF_001313835.1	ASM131383v1	*Polistes canadensis* (wasps, ants, and bees)	29465	521566
GCF_001594075.1	Tcor1.0	*Trachymyrmex cornetzi* (ants)	29356	760749
GCA_003710045.1	USU_Nmel_1.2	*Nomia melanderi* (Alkali bee)	28892	2351152
GCA_900480025.1	Eann	*Eupelmus annulatus* (wasps, ants, and bees)	28345	28524
GCF_000184785.3	Aflo_1.1	*Apis florea* (little honeybee)	24915	2863240
GCA_000188095.1	BIMP_1.0	*Bombus impatiens* (common eastern bumble bee)	24802	1017298
GCF_000184785.2	Aflo_1.0	*Apis florea* (little honeybee)	24704	2863240
GCF_000184785.1	Aflo_1.0	*Apis florea* (little honeybee)	24704	2863240
GCF_003651465.1	ASM365146v1	*Formica exsecta* (ants)	24299	997654
GCA_003063835.1	AZXXR	*Aphaenogaster floridana* (ants)	23448	439114
GCF_001652005.1	ASM165200v1	*Ceratina calcarata* (bees)	23399	632424
GCA_900474335.1	Onit	*Ormyrus nitidulus* (wasps, ants, and bees)	22971	22984
GCF_001263275.1	ASM126327v1	*Habropoda laboriosa* (bees)	22370	1784116
GCF_000980195.1	M.pharaonis_V2.0	*Monomorium pharaonis* (pharaoh ant)	21806	75377
GCA_003063805.1	AZXXQ	*Aphaenogaster ashmeadi* (ants)	21672	336807
GCF_003260585.2	ASM326058v2	*Monomorium pharaonis* (pharaoh ant)	21277	15645999
GCA_000980195.2	M.pharaonis_V2.0	*Monomorium pharaonis* (pharaoh ant)	21277	73835
GCF_000188075.2	Si_gnH	*Solenopsis invicta* (red fire ant)	21161	621039
GCA_002290385.1	ApisCC1.0	*Apis cerana cerana* (Asiatic honeybee)	21160	1393515
GCA_003063725.1	AZXXO	*Aphaenogaster miamiana* (ants)	20738	351517
GCF_000147175.1	CamFlo_1.0	*Camponotus floridanus* (Florida carpenter ant)	19487	451320
GCA_003063745.1	AZXXM	*Aphaenogaster rudis* (ants)	18941	269776
GCF_000002325.1	Nvit_1.0	*Nasonia vitripennis* (jewel wasp)	18865	708988
GCF_000002325.3	Nvit_2.1	*Nasonia vitripennis* (jewel wasp)	18840	708988
GCF_000002325.2	Nvit_2.0	*Nasonia vitripennis* (jewel wasp)	18840	698296
GCA_900480035.1	Euro	*Eupelmus urozonus* (wasps, ants, and bees)	17904	18995
GCA_001045655.1	ASM104565v1	*Lasius niger* (ants)	17048	17057
GCA_000346575.1	ASM34657v1	*Lasioglossum albipes* (bees)	16944	628061

GCA_004916985.1	UKY_Npine_v1	*Neodiprion pinetum* (white pine sawfly)	15816	609994
GCA_003063765.1	AZXXP	*Aphaenogaster fulva* (ants)	15753	255328
GCA_003063815.1	AJDMW	*Aphaenogaster rudis* (ants)	15622	300103
GCA_003063865.1	AZXXN	*Aphaenogaster picea* (ants)	15430	322984
GCF_001594045.1	Acol1.0	*Atta colombica* (ants)	15290	2037154
GCA_004329405.1	Cnig_gn1	*Cataglyphis niger* (desert ant)	15276	17950
GCF_001594115.1	Tsep1.0	*Trachymyrmex septentrionalis* (ants)	14962	2520094
GCF_000143395.1	Attacep1.0	*Atta cephalotes* (ants)	14798	5154485
GCA_000143395.1	Attacep1.0	*Atta cephalotes* (ants)	14798	5154485
GCF_000188075.1	Si_gnG	*Solenopsis invicta* (red fire ant)	14677	558018
GCF_000648655.1	Cflo_1.0	*Copidosoma floridanum* (wasps, ants, and bees)	14521	1037125
GCF_000572035.2	Mdem2	*Microplitis demolitor* (wasps, ants, and bees)	14116	1139389
GCA_003595255.1	Sf_gnA	*Solenopsis fugax* (ants)	13777	14463
GCF_000572035.1	Mdem1	*Microplitis demolitor* (wasps, ants, and bees)	13540	318766
GCA_000980195.1	M.pharaonis_V1.0	*Monomorium pharaonis* (pharaoh ant)	13470	16239
GCA_900475205.1	Ebru	*Eurytoma brunniventris* (wasps, ants, and bees)	12988	13596
GCA_900490025.1	Mdor	*Megastigmus dorsalis* (wasps, ants, and bees)	12960	18748
GCA_001276565.1	ASM127656v1	*Melipona quadrifasciata* (bees)	12520	1864352
GCA_002201625.1	Edil_v1.0	*Euglossa dilemma* (dilemma orchid bee)	12398	143590
GCA_004307685.1	ASM430768v1	*Ceratina australensis* (bees)	12363	14575l
GCA_004195275.1	ASM419527v1	*Cataglyphis hispanica* (ants)	11959	13064
GCF_000187915.1	Pbar_UMD_V03	*Pogonomyrmex barbatus* (red harvester ant)	11605	819605
GCA_002249905.1	ASM224990v1	*Trichomalopsis sarcophagae* (wasps, ants, and bees)	9960	22350
GCA_900474315.1	Taur	*Torymus auratus* (wasps, ants, and bees)	9699	9797
GCA_003121605.1	ASM312160v1	*Leptopilina boulardi* (wasps, ants, and bees)	9374	15354
GCA_009602685.1	ASM960268v1	*Leptopilina heterotoma* (wasps, ants, and bees)	9278	11848
GCF_000469605.1	Apis dorsata 1.3	*Apis dorsata* (giant honeybee)	8422	732052
GCA_001855655.1	ASM185565v1	*Leptopilina clavipes* (wasps, ants, and bees)	8276	13761
GCA_900490015.1	Msti	*Megastigmus stigmatizans* (wasps, ants, and bees)	7394	9143
GCA_004480015.1	UT_Atex_0.2	*Atta texana* (ants)	6806	10773

GCA_002806875.1	ASM280687v1	*Lepidotrigona ventralis hoosana* (bees)	6124	6644
GCA_003575265.1	Mp_gnA	*Monomorium pharaonis* (pharaoh ant)	5266	5718
GCA_900474355.1	Tger	*Torymus geranii* (wasps, ants, and bees)	4982	4996
GCA_009026005.1	ASM902600v1	*Leptopilina heterotoma* (wasps, ants, and bees)	4476	5051
GCA_009025955.1	ASM902595v1	*Leptopilina heterotoma* (wasps, ants, and bees)	4420	5024
GCA_900474235.1	Csem	*Cecidostiba semifascia* (wasps, ants, and bees)	3407	6094
GCA_000004775.1	Ngir_1.0	*Nasonia giraulti* (wasps, ants, and bees)	1971	759431
GCA_000004795.1	Nlon_1.0	*Nasonia longicornis* (wasps, ants, and bees)	1876	758407
GCA_001675545.1	ASM167554v1	*Cotesia vestalis* (diamondback moth parasitoid)	1100	1100
GCA_900490065.1	Neuqba	*Neuroterus quercusbaccarum* (wasps, ants, and bees)	1019	1664
GCA_900474215.1	Andinfl	*Andricus inflator* (wasps, ants, and bees)	980	1665
GCA_900474265.1	Andqra	*Andricus quercusramuli* (wasps, ants, and bees)	977	1138
GCA_900474195.1	Andgro	*Andricus grossulariae* (wasps, ants, and bees)	799	1511
GCA_009394715.1	ASM939471v1	*Diadromus collaris* (wasps, ants, and bees)	695	695
GCA_900474205.1	Andcurv	*Andricus curvator* (wasps, ants, and bees)	635	1116
GCA_900490055.1	Neusal	*Pseudoneuroterus saliens* (wasps, ants, and bees)	536	970
GCA_000819425.1	Ami_v1	*Apis mellifera intermissa* (honey bee)	504	527

Table A3.S2: Microsatellite primers details

Average allele per locus: 27.5. Primers from (Fournier, Aron and Milinkovitch, 2002).

	Size range		Number of alleles	
Locus	**Expected**	**Observed**	**Expected**	**Observed**
Ppalo1T	156–194	144-202	11	23
Ppal33	106–122	98-154	9	32
Ppal84	104–128	100-139	8	31
Ppalo3	94–122	78-120	6	31
Ppal73	136–150	121-159	8	29
Ppal12	95–124	90-125	9	19

Table A3.S3: Sample details – geography and social form

P = polygynous, multiple-queen; M = monogynous, single-queen

Sample	Gyny	Country	Region	Locality	Latitude	Longitude	Elevation (m)
A01	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A02	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A03	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A04	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A05	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A06	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A07	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A08	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A09	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A11	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A12	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A13	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A14	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A15	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A17	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A18	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A19	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A20	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A21	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A22	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A23	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A24	P	France	Occitanie	Bruniquel	44.050628	1.677367	220

A25	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A26	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A27	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A28	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A29	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A30	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A31	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A32	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A33	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A34	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A35	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A36	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A37	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A38	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A39	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A40	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A41	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A43	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A44	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A45	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A46	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A47	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A48	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A49	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A50	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A51	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A52	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A53	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A54	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A55	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A56	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A57	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
A58	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A59	P	France	Occitanie	Bruniquel	44.050628	1.677367	220
A60	M	France	Occitanie	Bruniquel	44.050628	1.677367	220
CAH1	P	France	Occitanie	Cahones	42.38892	2.4965	79
CMR01	P	Spain	Catalonia	Girona	42.187349	2.487893	505
CU121	P	Spain	Andalucia	Jaén	37.788168	-3.773779	427

E01	P	France	Occitanie	Perpignan	42.653533	2.924267	25
E02	M	France	Occitanie	La Vall	42.50741	3.00662	245
E03	M	France	Occitanie	La Vall	42.50733	3.00696	245
E04	P	France	Occitanie	La Vall	42.505117	3.0085	248
E05	P	France	Occitanie	La Vall	42.65355	2.924283	286
E112	M	France	Occitanie	Bruniquel	44.050281	1.655848	248
E114	P	France	Occitanie	Bruniquel	44.050325	1.656046	245
E120	P	France	Occitanie	Bruniquel	44.050203	1.656265	249
E121	P	France	Occitanie	Bruniquel	44.050046	1.6562	249
E129	M	France	Occitanie	Bruniquel	44.05018	1.65608	250
E130	P	France	Occitanie	Bruniquel	44.050295	1.65607	246
E133	P	France	Occitanie	Bruniquel	44.05032	1.656043	242
E14	P	France	Occitanie	St-Cyprien	42.639533	3.0146	0
E16	M	France	Occitanie	Perpignan	42.646417	2.8569	62
E19	M	France	Occitanie	St-Cyprien	42.646567	2.853683	66
E24	P	France	Occitanie	St-Cyprien	42.639533	3.0146	0
E25	M	France	Occitanie	Banyuls	42.457228	3.08627	77
E28	P	France	Occitanie	Perpignan	42.649033	2.8506	27
E37	P	France	Occitanie	Perpignan	42.646417	2.8569	62
E55	P	France	Occitanie	Perpignan	42.648633	2.848333	63
E73	M	France	Occitanie	Bruniquel	44.05016	1.65587	245
E87	P	France	Occitanie	Bruniquel	44.050341	1.655911	244
E88	P	France	Occitanie	Bruniquel	44.050246	1.656186	251
E90	M	France	Occitanie	Bruniquel	44.050083	1.656265	249
E95	P	France	Occitanie	Bruniquel	44.050043	1.656522	248
I02	M	Italy	Tuscany	Scandicci	43.745767	11.1315	100
I03	M	Italy	Tuscany	Scandicci	43.745953	11.131233	106
I04	P	Italy	Tuscany	Scandicci	43.745942	11.131283	107
I05	P	Italy	Tuscany	Scandicci	43.745925	11.131286	102
I06	M	Italy	Tuscany	Scandicci	43.745823	11.132031	90
I07	P	Italy	Tuscany	Scandicci	43.74613	11.1323	90
I08	M	Italy	Tuscany	Scandicci	43.74613	11.132264	89
I10	M	Italy	Tuscany	Scandicci	43.74618	11.13127	107
I11	M	Italy	Tuscany	Scandicci	43.745972	11.13122	109
I12	M	Italy	Tuscany	Scandicci	43.746028	11.130442	117
I13	M	Italy	Tuscany	Scandicci	43.745945	11.130439	114
I14	M	Italy	Tuscany	Scandicci	43.746033	11.130528	114
I15	P	Italy	Tuscany	Scandicci	43.746342	11.131203	106

I16	P	Italy	Tuscany	Scandicci	43.7466	11.131181	111
I18	M	Italy	Tuscany	Scandicci	43.746537	11.131283	111
I19	M	Italy	Tuscany	Scandicci	43.746042	11.130414	115
I20	M	Italy	Tuscany	Scandicci	43.745381	11.129961	119
I21	M	Italy	Tuscany	Scandicci	43.74444	11.128992	133
I22	P	Italy	Tuscany	Scandicci	43.743661	11.128736	145
I23	M	Italy	Tuscany	Scandicci	43.743758	11.127064	148
I24	M	Italy	Tuscany	Scandicci	43.743733	11.126486	153
I25	M	Italy	Tuscany	Scandicci	43.74355	11.126069	145
I27	P	Italy	Tuscany	Scandicci	43.74587	11.130128	113

Table A3.S4: BLASTN hits of significant SNPs to *S. invicta* genome

35 out 46 significant SNPs have BLASTN hits (-evalue 1e-5 -max_target_seqs 1), including 2 in supergene (in bold).

qseqid	sseqid	evalue	identity	length	Location in gnG
Ppal_E.contig_2068:294164-295164	NW_011800493.1	0	93.701	1016	recombining
Ppal_E.contig_1245:234310-235310	NW_011795970.1	0	88.596	947	not in genetic map
Ppal_E.contig_1136:149827-150827	NW_011797448.1	0	83.887	993	recombining
Ppal_E.contig_1561:331475-332475	NW_011796802.1	0	86.724	693	recombining
Ppal_E.contig_1050:84432-85432	NW_011847194.1	0	93.952	463	not in genetic map
Ppal_E.contig_1562:744702-745702	**NW_011795053.1**	**0**	**79.922**	**1031**	**supergene**
Ppal_E.contig_100:367779-368779	**NW_011794844.1**	**0**	**88.151**	**557**	**supergene**
Ppal_E.contig_1734:422761-423761	NW_011800113.1	0	83.631	727	recombining
Ppal_E.contig_1250:93795-94795	NW_011794565.1	0	79.041	1064	recombining
Ppal_E.contig_2165:567683-568683	NW_011794869.1	2.25E-177	90.968	465	recombining
Ppal_E.contig_413:53098-54098	NW_011796802.1	8.10E-177	88.528	523	recombining
Ppal_E.contig_1226:101449-102449	NW_011798146.1	5.05E-149	77.905	964	recombining
Ppal_E.contig_1808:410483-411483	NW_011794668.1	1.83E-143	88.759	427	recombining
Ppal_E.contig_2581:111745-112745	NW_011795719.1	1.83E-143	77.143	1015	recombining
Ppal_E.contig_1327:605293-606293	NW_011796896.1	8.52E-142	85.221	521	recombining
Ppal_E.contig_1909:787554-788554	NW_011800113.1	8.52E-142	79.67	787	recombining
Ppal_E.contig_123:763844-764844	NW_011796690.1	1.43E-139	78.954	822	recombining
Ppal_E.contig_1096:326434-327434	NW_011794959.1	1.45E-129	84.294	503	recombining
Ppal_E.contig_1399:157121-158121	NW_011796848.1	3.20E-111	84.828	435	recombining
Ppal_E.contig_506:39428-40428	NW_011847869.1	4.22E-95	78.269	566	not in genetic map
Ppal_E.contig_16:792877-793877	NW_011801990.1	1.52E-94	87.261	314	recombining

Ppal_E.contig_1327:605293-606293	NW_011796896.1	3.31E-86	85.993	307	recombining
Ppal_E.contig_1018:243769-244769	NW_011798105.1	3.33E-81	84.356	326	recombining
Ppal_E.contig_1916:428803-429803	NW_011802221.1	4.34E-75	79.121	455	recombining
Ppal_E.contig_1096:326434-327434	NW_011794959.1	1.57E-69	85.053	281	recombining
Ppal_E.contig_261:349222-350222	NW_011798377.1	1.25E-50	82.52	246	recombining
Ppal_E.contig_132:0-670	NW_011800878.1	5.09E-33	92.929	99	recombining
Ppal_E.contig_1096:104519-105519	NW_011797741.1	9.95E-32	78.667	225	not in genetic map
Ppal_E.contig_1734:410266-411266	NW_011800113.1	2.15E-28	84.444	135	recombining
Ppal_E.contig_39:141224-142224	NW_011796746.1	1.69E-19	79.006	181	recombining
Ppal_E.contig_144:1312842-1313842	NW_011799552.1	1.31E-15	81.818	110	recombining
Ppal_E.contig_1100:250823-251823	NW_011803501.1	2.20E-13	82.292	96	recombining
Ppal_E.contig_1399:230526-231526	NW_011802208.1	2.20E-13	85.897	78	not in genetic map
Ppal_E.contig_2082:146022-147022	NW_011799280.1	1.02E-11	85.915	71	recombining
Ppal_E.contig_261:348035-349035	NW_011798377.1	4.76E-10	75.776	161	recombining
Ppal_E.contig_1474:55129-56129	NW_011796384.1	7.97E-08	100	33	recombining
Ppal_E.contig_261:348035-349035	NW_011798377.1	7.97E-08	97.222	36	recombining
Ppal_E.contig_261:348035-349035	NW_011798377.1	7.97E-08	97.222	36	recombining
Ppal_E.contig_530:3757-4757	NW_011795475.1	1.03E-06	97.059	34	recombining

Table A3.S5: Regions unique to single- and multiple-queen genomes

	Number of regions	**Regions length range (bp)**	**Blast results**
Genomic regions unique to all 35 single-queen samples	13	1,016 - 2,373	8 regions without any match 5 regions with PREDICTED ant mRNA (either uncharacterized; or associated with cell energy, cell recognition)
Genomic regions unique to all 73 multiple-queen samples	161	1,002 - 10,499	

Table A3.S6: Illumina sequencing summary

We received 213G of data from Genewiz (size of directory with 115 fastq.gz files), with a total of 2,762,930,432 raw paired-end sequences.

	number of samples	**mean number of sequences representing a sample**	**minimum number of sequences representing a sample**	**maximum number of sequences representing a sample**
Total	115	12,012,741	6,411,726	63,391,150
Polygynous samples	74	11,802,151	6,411,726	63,391,150
Monogynous samples	39	12,378,667	7,040,495	20,799,984

Table A3.S7: Nanopore sequencing summary

Sample Origin	**Flow cell**	**Run name**	**Content**	**Chemistry**	**Total sequencing yield (bp)**	**Average sequence length (bp)**	**Genome coverage* (x)**
colony 129**	1	Phei2	1 male	2D	230 M	2.6 K	0.77
	2	Phei3	1 male	2D	141 M	2.0 K	0.47
	3	Phei4	1 male + 35 workers	2D	2.7 G	2.6 K	9
colony 12***	4	ppal4	100 workers	1D ligation	35 M	1.7 K	0.12
	5	ppal15	150 workers	1D ligation	409 M	2.1 K	1.4
	6	ppal16	200 workers	1D ligation	1.6 G	4.6 K	5.31

* estimated Pheidole genome size = 300Mb

** From Bruniquel, France (latitude 44.05018, longitude 1.65608)

*** From Perpignan, France

References

Alexa, A. and Rahnenführer, J. (2009) "Gene set enrichment analysis with topGO," *Bioconductor Improv*. bioconductor.riken.jp, 27. Available at: http://bioconductor.riken.jp/packages/3.0/bioc/vignettes/topGO/inst/doc/topGO.pdf.

Alkan, C., Sajjadian, S. and Eichler, E. E. (2011) "Limitations of next-generation genome sequence assembly," *Nature methods*, 8(1), pp. 61–65.

Altschul, S. F. *et al.* (1990) "Basic local alignment search tool," *Journal of molecular biology*. Elsevier, 215(3), pp. 403–410.

Ammann, P. and Offutt, J. (2016) *Introduction to Software Testing*. Cambridge University Press.

Arguello, J. R. and Connallon, T. (2011) "Gene duplication and ectopic gene conversion in Drosophila," *Genes*. mdpi.com, 2(1), pp. 131–151.

Aron, S. *et al.* (1999) "Social structure and split sex ratios in the ant Pheidole pallidula," *Ethology Ecology & Evolution*, pp. 209–227. doi: 10.1080/08927014.1999.9522824.

Ascunce, M. S. *et al.* (2011) "Global invasion history of the fire ant Solenopsis invicta," *Science*, 331(6020), pp. 1066–1068.

Avril, A. *et al.* (2019) "Asymmetric assortative mating and queen polyandry are linked to a supergene controlling ant social organization," *Molecular Ecology*, pp. 1428–1438. doi: 10.1111/mec.14793.

Avril, A. *et al.* (2020) "Maternal effect killing by a supergene controlling ant social organization," *Proceedings of the National Academy of Sciences of the United States of America*. doi: 10.1073/pnas.2003282117.

Ballouz, S., Dobin, A. and Gillis, J. A. (2019) "Is it time to change the reference genome?," *Genome biology*. BioMed Central, 20(1), pp. 1–9.

Bankevich, A. *et al.* (2012) "SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing," *Journal of computational biology: a journal of computational molecular cell biology*, 19(5), pp. 455–477.

Bastian, F. *et al.* (2008) "Data Integration in the Life Sciences," *Lecture notes in computer science*. Springer Berlin Heidelberg Berlin, Heidelberg, pp. 124–131.

Benjamini, Y. and Hochberg, Y. (1995) "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *Journal of the Royal Statistical Society: Series B (Methodological)*, pp. 289–300. doi: 10.1111/j.2517-6161.1995.tb02031.x.

Benson, D. A. *et al.* (2018) "GenBank," *Nucleic acids research*. pdfs.semanticscholar.org, 46(D1), pp. D41–D47.

Benton, R. *et al.* (2009) "Variant ionotropic glutamate receptors as chemosensory receptors in Drosophila," *Cell*. Elsevier, 136(1), pp. 149–162.

Berlin, K. *et al.* (2015) "Assembling large genomes with single-molecule sequencing and locality-sensitive hashing," *Nature biotechnology*, 33(6), pp. 623–630.

Bhatkar, A. and Whitcomb, W. H. (1970) "Artificial Diet for Rearing Various Species of Ants," *The Florida Entomologist*, p. 229. doi: 10.2307/3493193.

Blanchard, B. D. and Moreau, C. S. (2017) "Defensive traits exhibit an evolutionary trade-off and drive diversification in ants," *Evolution*, pp. 315–328. doi: 10.1111/evo.13117.

Blanchoud, S. *et al.* (2018) "De novo draft assembly of the Botrylloides leachii genome provides further insight into tunicate evolution," *Scientific reports*. nature.com, 8(1), p. 5518.

Boratyn, G. M. *et al.* (2019) "Magic-BLAST, an accurate RNA-seq aligner for long and short reads," *BMC Bioinformatics*. doi: 10.1186/s12859-019-2996-x.

Boulay, R. *et al.* (2014) "The ecological benefits of larger colony size may promote polygyny in ants," *Journal of evolutionary biology*, 27(12), pp. 2856–2863.

Bourke, A. F. G. and Franks, N. R. (1995) *Social Evolution in Ants*. Princeton, New Jersey: Princeton University Press.

Bourke, A. F. G. and Heinze, J. (1994) "The ecology of communal breeding: the case of multiple-queen leptothoracine ants," *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*. Royal Society, 345(1314), pp. 359–372.

Bray, N. L. *et al.* (2016a) "Erratum: Near-optimal probabilistic RNA-seq quantification," *Nature biotechnology*. nature.com, 34(8), p. 888.

Bray, N. L. *et al.* (2016b) "Near-optimal probabilistic RNA-seq quantification," *Nature biotechnology*, 34(5), pp. 525–527.

Brelsford, A. *et al.* (2020) "An Ancient and Eroded Social Supergene Is Widespread across Formica Ants," *Current biology: CB*, 30(2), pp. 304-311.e4.

Buchfink, B., Xie, C. and Huson, D. H. (2015) "Fast and sensitive protein alignment using DIAMOND," *Nature methods*, 12(1), pp. 59–60.

Buechel, S. D., Wurm, Y. and Keller, L. (2014) "Social chromosome variants differentially affect queen determination and the survival of workers in the fire ant *Solenopsis invicta*," *Molecular ecology*, 23(20), pp. 5117–5127.

Buels, R. *et al.* (2016) "JBrowse: a dynamic web platform for genome visualization and analysis," *Genome biology*, 17(1), p. 66.

Calkins, T. L. *et al.* (2018) "Brain gene expression analyses in virgin and mated queens of fire ants reveal mating-independent and socially regulated changes," *Ecology and evolution*, 8(8), pp. 4312–4327.

Camacho, C. *et al.* (2009) "BLAST+: architecture and applications," *BMC bioinformatics*. Springer, 10, p. 421.

Cantarel, B. L. *et al.* (2007) "MAKER: An easy-to-use annotation pipeline designed for emerging model organism genomes," *Genome Research*, pp. 188–196. doi: 10.1101/gr.6743907.

Capella-Gutiérrez, S., Silla-Martínez, J. M. and Gabaldón, T. (2009) "trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses," *Bioinformatics* , 25(15), pp. 1972–1973.

Chang, C. C. *et al.* (2015) "Second-generation PLINK: rising to the challenge of larger and richer datasets," *GigaScience*, 4, p. 7.

Charif, D. *et al.* (2007) "Structural approaches to sequence evolution: molecules, networks, populations," *Biological and Medical Physics, Biomedical Engineering*.

Cheng, H. *et al.* (2022) "Haplotype-resolved assembly of diploid genomes without parental data," *Nature biotechnology*, 40(9), pp. 1332–1335.

Chevreux, B., Wetter, T. and Suhai, S. (1999) "Computer science and biology. Proceedings of the German Conference on Bioinformatics, GCB'99," *Genome sequence assembly using trace signals and additional sequence information. [Google Scholar]*.

Chevreux, Bastien, Wetter, T. and Suhai, S. (1999) "Genome sequence assembly using trace signals and additional sequence information," in *Computer Science and Biology: Proceedings of the German Conference on Bioinformatics GCB'99*. Heidelberg, pp. 45–56.

Chin, C.-S. *et al.* (2013) "Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data," *Nature methods*, 10(6), pp. 563–569.

Conesa, A. *et al.* (2005) "Blast2GO: a universal tool for annotation, visualization and analysis in functional genomics research," *Bioinformatics* , 21(18), pp. 3674–3676.

Conte, M. A. *et al.* (2017) "A high quality assembly of the Nile Tilapia (*Oreochromis niloticus*) genome reveals the structure of two sex determination regions," *BMC genomics*, 18(1), p. 341.

Corona, M. *et al.* (2013) "Vitellogenin underwent subfunctionalization to acquire caste and behavioral specific expression in the harvester ant Pogonomyrmex barbatus," *PLoS genetics*. journals.plos.org, 9(8), p. e1003730.

Cui, Y. *et al.* (2016) "BioCircos.js: an interactive Circos JavaScript library for biological data visualization on web applications," *Bioinformatics* , *32*(11), pp. 1740–1742.

Danecek, P. *et al.* (2011) "The variant call format and VCFtools," *Bioinformatics* , *27*(15), pp. 2156–2158.

DeHeer, C. J. (2002) "A comparison of the colony-founding potential of queens from single- and multiple-queen colonies of the fire antSolenopsis invicta," *Animal behaviour*. Elsevier, 64(4), pp. 655–661.

DeHeer, C. J., Goodisman, M. A. D. and Ross, K. G. (1999) "Queen Dispersal Strategies in the Multiple-Queen Form of the Fire Ant Solenopsis invicta," *The American naturalist*. The University of Chicago Press, 153(6), pp. 660–675.

Denton, J. F. *et al.* (2014) "Extensive Error in the Number of Genes Inferred from Draft Genome Assemblies," *PLoS computational biology*, 10(12), p. e1003998.

Dillies, M.-A. *et al.* (2013) "A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis," *Briefings in bioinformatics*. academic.oup.com, 14(6), pp. 671–683.

Drăgan, M.-A. *et al.* (2016) "GeneValidator: identify problems with protein-coding gene predictions," *Bioinformatics* , *32*(10), pp. 1559–1561.

Eliyahu, D. *et al.* (2011) "Venom alkaloid and cuticular hydrocarbon profiles are associated with social organization, queen fertility status, and queen genotype in the fire ant Solenopsis invicta," *Journal of chemical ecology*. Springer, *37*(11), pp. 1242–1254.

Fan, J.-B. *et al.* (2002) "Paternal origins of complete hydatidiform moles proven by whole genome single-nucleotide polymorphism haplotyping," *Genomics*, 79(1), pp. 58–62.

Finn, R. D. *et al.* (2014) "Pfam: the protein families database," *Nucleic acids research*. academic.oup.com, 42(Database issue), pp. D222-30.

Flanagan, D. and Matsumoto, Y. (2008) *The Ruby Programming Language: Everything You Need to Know*. "O'Reilly Media, Inc."

Florea, L. *et al.* (2011) "Genome assembly has a major impact on gene content: a comparison of annotation in two *Bos taurus* assemblies," *PloS one*, 6(6), p. e21400.

Fontana, S. *et al.* (2019) "The fire ant social supergene is characterized by extensive gene and transposable element copy number variation," *Molecular ecology*, 29(1), pp. 105–120.

Forêt, S. and Maleszka, R. (2006) "Function and evolution of a gene family encoding odorant binding-like proteins in a social insect, the honey bee (Apis mellifera)," *Genome research*. genome.cshlp.org, 16(11), pp. 1404–1413.

Fournier, D., Aron, S. and Milinkovitch, M. C. (2002) "Investigation of the population genetic structure and mating system in the ant Pheidole pallidula," *Molecular ecology*, 11(9), pp. 1805–1814.

Franch-Gras, L. *et al.* (2018) "Genomic signatures of local adaptation to the degree of environmental predictability in rotifers," *Scientific reports*, 8(1), p. 16051.

Gadau, J. (2009) "DNA isolation from ants," *Cold Spring Harbor protocols*, 2009(7), p. db.prot5245.

Gaj, T., Gersbach, C. A. and Barbas, C. F., 3rd (2013) "ZFN, TALEN, and CRISPR/Cas-based methods for genome engineering," *Trends in biotechnology*. Elsevier, 31(7), pp. 397–405.

Garrett, J. J. (2010) *The Elements of User Experience: User-Centered Design for the Web and Beyond*. Pearson Education.

Garrison, E. *et al.* (2018) "Variation graph toolkit improves read mapping by representing genetic variation in the reference," *Nature biotechnology*, 36(9), pp. 875–879.

Garrison, E. and G., M. (2012) "Haplotype-based variant detection from short-read sequencing," *arXiv preprint*. doi: arXiv:1207.3907 [q-bio.GN].

Garrison, E. and Marth, G. (2012) "Haplotype-based variant detection from short-read sequencing," *arXiv [q-bio.GN]*. Available at: http://arxiv.org/abs/1207.3907.

Gómez, J. *et al.* (2013) "BioJS: an open source JavaScript framework for biological data visualization," *Bioinformatics* . academic.oup.com, 29(8), pp. 1103–1104.

Goodisman, M. A. D., DeHeer, C. J. and Ross, K. G. (2000) "Unusual behavior of polygyne fire ant queens on nuptial flights," *Journal of insect behavior*. Springer, 13(3), pp. 455–468.

Goto, N. *et al.* (2010) "BioRuby: bioinformatics software for the Ruby programming language," *Bioinformatics* . academic.oup.com, 26(20), pp. 2617–2619.

Gotoh, O. (2008) "Direct mapping and alignment of protein sequences onto genomic sequence," *Bioinformatics* , 24(21), pp. 2438–2444.

Gotzek, D. *et al.* (2011) "Odorant Binding Proteins of the Red Imported Fire Ant, Solenopsis invicta: An Example of the Problems Facing the Analysis of Widely Divergent Proteins," *PloS one*, 6(1), p. e16289.

Gotzek, D. and Ross, K. G. (2007) "Genetic regulation of colony social organization in fire ants: an integrative overview," *The Quarterly review of biology*. journals.uchicago.edu, 82(3), pp. 201–226.

Gotzek, D. and Ross, K. G. (2009) "Current status of a model system: the gene Gp-9 and its association with social organization in fire ants," *PloS one*. journals.plos.org, 4(11), p. e7713.

Gurevich, A. *et al.* (2013) "QUAST: quality assessment tool for genome assemblies," *Bioinformatics* , 29(8), pp. 1072–1075.

Hahn, M. W., Zhang, S. V. and Moyle, L. C. (2014) "Sequencing, Assembling, and Correcting Draft Genomes Using Recombinant Populations," *G3: Genes|Genomes|Genetics*, 4(4), pp. 669–679.

Harris, A. and Haase, K. (2011) *Sinatra: Up and Running: Ruby for the Web, Simply*. "O'Reilly Media, Inc."

Helmkampf, M., Cash, E. and Gadau, J. (2015) "Evolution of the insect desaturase gene family with an emphasis on social Hymenoptera," *Molecular biology and evolution*. academic.oup.com, 32(2), pp. 456–471.

Hickey, G. *et al.* (2020) "Genotyping structural variants in pangenome graphs using the vg toolkit," *Genome biology*, 21(1), p. 35.

Hoff, K. J. *et al.* (2016) "BRAKER1: Unsupervised RNA-Seq-Based Genome Annotation with GeneMark-ET and AUGUSTUS," *Bioinformatics* , 32(5), pp. 767–769.

Hölldobler, B. and Wilson, E. O. (1977) "The number of queens: An important trait in ant evolution," *Naturwissenschaften*, pp. 8–15. doi: 10.1007/bf00439886.

Holt, C. and Yandell, M. (2011) "MAKER2: an annotation pipeline and genome-database management tool for second-generation genome projects," *BMC bioinformatics*, 12(1), p. 491.

Huang, Y.-C. and Wang, J. (2014) "Did the fire ant supergene evolve selfishly or socially?," *BioEssays: news and reviews in molecular, cellular and developmental biology*. Wiley Online Library, 36(2), pp. 200–208.

Hughes, W. O. H., Ratnieks, F. L. W. and Oldroyd, B. P. (2008) "Multiple paternity or multiple queens: two routes to greater intracolonial genetic diversity in the eusocial Hymenoptera," *Journal of Evolutionary Biology*, pp. 1090–1095. doi: 10.1111/j.1420-9101.2008.01532.x.

Hunt, A. and Thomas, D. (2000) *The pragmatic programmer: from journeyman to master*. USA: Addison-Wesley Longman Publishing Co., Inc.

Hunt, G. J. and Page, R. E., Jr (1992) "Patterns of inheritance with RAPD molecular markers reveal novel types of polymorphism in the honey bee," *Theoretical and applied genetics*. Springer Science and Business Media LLC, 85(1), pp. 15–20.

Hunt, G. J. and Page, R. E., Jr (1995) "Linkage map of the honey bee, Apis mellifera, based on RAPD markers," *Genetics*, 139(3), pp. 1371–1382.

Hysi, P. G. *et al.* (2018) "Genome-wide association meta-analysis of individuals of European ancestry identifies new loci explaining a substantial fraction of hair color variation and heritability," *Nature genetics*, 50(5), pp. 652–656.

Ingram, K. K. *et al.* (2012) "The molecular clockwork of the fire ant Solenopsis invicta," *PloS one*. journals.plos.org, 7(11), p. e45715.

Iovinella, I. *et al.* (2011) "Differential expression of odorant-binding proteins in the mandibular glands of the honey bee according to caste and age," *Journal of proteome research*. ACS Publications, 10(8), pp. 3439–3449.

Iqbal, Z. *et al.* (2012) "De novo assembly and genotyping of variants using colored de Bruijn graphs," *Nature genetics*, 44(2), pp. 226–232.

Jarvis, E. D. *et al.* (2022) "Semi-automated assembly of high-quality diploid human reference genomes," *Nature*. doi: 10.1038/s41586-022-05325-5.

Jasper, W. C. *et al.* (2016) "Large-Scale Coding Sequence Change Underlies the Evolution of Postdevelopmental Novelty in Honey Bees," *Molecular biology and evolution*. academic.oup.com, 33(5), p. 1379.

Jeffries, D. L. *et al.* (2018) "A rapid rate of sex-chromosome turnover and non-random transitions in true frogs," *Nature communications*, 9(1), p. 4088.

Jiang, H. *et al.* (2014) "Skewer: a fast and accurate adapter trimmer for next-generation sequencing paired-end reads," *BMC bioinformatics*, 15, p. 182.

Johnson, B. R., Atallah, J. and Plachetzki, D. C. (2013) "The importance of tissue specificity for RNA-seq: highlighting the errors of composite structure extractions," *BMC genomics*. bmcgenomics.biomedcentral.com, 14, p. 586.

Johnson, B. R. and Linksvayer, T. A. (2010) "Deconstructing the superorganism: social physiology, groundplans, and sociogenomics," *The Quarterly review of biology*. journals.uchicago.edu, 85(1), pp. 57–79.

Jombart, T. (2008) "adegenet: a R package for the multivariate analysis of genetic markers," *Bioinformatics*, pp. 1403–1405. doi: 10.1093/bioinformatics/btn129.

Joron, M. *et al.* (2011) "Chromosomal rearrangements maintain a polymorphic supergene controlling butterfly mimicry," *Nature*, pp. 203–206. doi: 10.1038/nature10341.

Käfer, S. *et al.* (2019) "Re-assessing the diversity of negative strand RNA viruses in insects," *PLoS pathogens*, 15(12), p. e1008224.

Katoh, K. and Toh, H. (2008) "Recent developments in the MAFFT multiple sequence alignment program," *Briefings in bioinformatics*. academic.oup.com, 9(4), pp. 286–298.

Keilwagen, J. *et al.* (2018) "Combining RNA-seq data and homology-based gene prediction for plants, animals and fungi," *BMC bioinformatics*, 19(1), p. 189.

Keller, L. and Ross, K. G. (1995) "Gene by Environment Interaction: Effects of a Single Gene and Social Environment on Reproductive Phenotypes of Fire Ant Queens," *Functional ecology*. [British Ecological Society, Wiley], 9(4), pp. 667–676.

Keller, L. and Ross, K. G. (1998) "Selfish genes: a green beard in the red fire ant," *Nature*. Nature Publishing Group, 394(6693), pp. 573–575.

Keller and Ross (1999) "Major gene effects on phenotype and fitness: the relative roles of Pgm-3 and Gp-9 in introduced populations of the fire ant Solenopsis invicta," *Journal of evolutionary biology*. Wiley, 12(4), pp. 672–680.

Kelley, D. R. and Salzberg, S. L. (2010) "Detection and correction of false segmental duplications caused by genome mis-assembly," *Genome biology*, 11(3), p. R28.

Kent, W. J. (2002) "BLAT—The BLAST-Like Alignment Tool," *Genome research*, 12(4), pp. 656–664.

Kent, W. J. *et al.* (2003) "Evolution's cauldron: Duplication, deletion, and rearrangement in the mouse and human genomes," *Proceedings of the National Academy of Sciences*, 100(20), pp. 11484–11489.

Khelik, K. *et al.* (2020) "NucBreak: Location of structural errors in a genome assembly by using paired-end Illumina reads," *BMC bioinformatics*, 21(1), p. 393488.

Kim, D. *et al.* (2013) "TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions," *Genome biology*, 14(4), p. R36.

Kim, D. *et al.* (2019) "Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype," *Nature biotechnology*, 37(8), pp. 907–915.

King, T., Butcher, S. and Zalewski, L. (2017) "Apocrita - High Performance Computing Cluster for Queen Mary University of London."

Kolmogorov, M. *et al.* (2019) "Assembly of long, error-prone reads using repeat graphs," *Nature biotechnology*, 37(5), pp. 540–546.

Kondrashov, F. A. (2012) "Gene duplication as a mechanism of genomic adaptation to a changing environment," *Proceedings. Biological sciences / The Royal Society*. royalsocietypublishing.org, 279(1749), pp. 5048–5057.

Koren, S. *et al.* (2013) "Reducing assembly complexity of microbial genomes with single-molecule sequencing," *Genome biology*, 14(9), p. R101.

Koren, S. *et al.* (2017) "Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation," *Genome research*, 27(5), pp. 722–736.

Korf, I. (2004) "Gene finding in novel genomes," *BMC bioinformatics*, 5(1), p. 59.

Krieger, M. J. B. (2005) "To b or not to b: a pheromone-binding protein regulates colony social organization in fire ants," *BioEssays: news and reviews in molecular, cellular and developmental biology*. Wiley, 27(1), pp. 91–99.

Krieger, M. J. B. and Ross, K. G. (2002) "Identification of a major gene regulating complex social behavior," *Science*, 295(5553), pp. 328–332.

Krieger, M. J. B. and Ross, K. G. (2005) "Molecular evolutionary analyses of the odorant-binding protein gene Gp-9 in fire ants and other Solenopsis species," *Molecular biology and evolution*. academic.oup.com, 22(10), pp. 2090–2103.

Kronenberg, Z. N. *et al.* (2018) "High-resolution comparative analysis of great ape genomes," *Science*, 360(6393), p. eaar6343.

Kuhn, R. M., Haussler, D. and Kent, W. J. (2013) "The UCSC genome browser and associated tools," *Briefings in bioinformatics*, 14(2), pp. 144–161.

Kulmuni, J., Wurm, Y. and Pamilo, P. (2013) "Comparative genomics of chemosensory protein genes reveals rapid evolution and positive selection in ant-specific duplicates," *Heredity*. nature.com, 110(6), pp. 538–547.

Langfelder, P. and Horvath, S. (2008) "WGCNA: an R package for weighted correlation network analysis," *BMC bioinformatics*. Springer, 9, p. 559.

Langmead, B. *et al.* (2009) "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome," *Genome biology*, 10(3), p. R25.

Langmead, B. and Salzberg, S. L. (2012) "Fast gapped-read alignment with Bowtie 2," *Nature methods*, 9(4), pp. 357–359.

Lassance, J.-M. *et al.* (2010) "Allelic variation in a fatty-acyl reductase gene causes divergence in moth sex pheromones," *Nature*. nature.com, 466(7305), pp. 486–489.

Lawrence, M. *et al.* (2013) "Software for computing and annotating genomic ranges," *PLoS computational biology*, 9(8), p. e1003118.

Lawson, L. P., Vander Meer, R. K. and Shoemaker, D. (2012) "Male reproductive fitness and queen polyandry are linked to variation in the supergene Gp-9 in the fire ant Solenopsis invicta," *Proceedings. Biological sciences / The Royal Society*. royalsocietypublishing.org, 279(1741), pp. 3217–3222.

Leal, W. S. (2013) "Odorant reception in insects: roles of receptors, binding proteins, and degrading enzymes," *Annual review of entomology*. annualreviews.org, 58, pp. 373–391.

Lee, E. *et al.* (2013) "Web Apollo: a web-based genomic annotation editing platform," *Genome biology*. Springer Nature, 14(8), p. R93.

Leek, J. T. *et al.* (2012) "The sva package for removing batch effects and other unwanted variation in high-throughput experiments," *Bioinformatics* , 28(6), pp. 882–883.

Lehman, M. M. (1980) "Programs, life cycles, and laws of software evolution," *Proceedings of the IEEE*. ieeexplore.ieee.org, 68(9), pp. 1060–1076.

Lewin, H. A. *et al.* (2018) "Earth BioGenome Project: Sequencing life for the future of life," *Proceedings of the National Academy of Sciences of the United States of America*, 115(17), pp. 4325–4333.

Li, H. *et al.* (2009) "The Sequence Alignment/Map format and SAMtools," *Bioinformatics* , 25(16), pp. 2078–2079.

Li, H. (2011) "Tabix: fast retrieval of sequence features from generic TAB-delimited files," *Bioinformatics* , 27(5), pp. 718–719.

Li, H. (2013) "Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. 2013. arXiv:1303.3997."

Li, H. (2018) "Minimap2: pairwise alignment for nucleotide sequences," *Bioinformatics* , 34(18), pp. 3094–3100.

Li, H. and Durbin, R. (2009) "Fast and accurate short read alignment with Burrows-Wheeler transform," *Bioinformatics*, pp. 1754–1760. doi: 10.1093/bioinformatics/btp324.

Li, S. *et al.* (2008) "Multiple functions of an odorant-binding protein in the mosquito Aedes aegypti," *Biochemical and biophysical research communications*. Elsevier, 372(3), pp. 464–468.

Liew, Y. J., Aranda, M. and Voolstra, C. R. (2016) "Reefgenomics.org - a repository for marine genomics data," *Database: the journal of biological databases and curation*. academic.oup.com, 2016. doi: 10.1093/database/baw152.

Linksvayer, T. A., Busch, J. W. and Smith, C. R. (2013) "Social supergenes of superorganisms: do supergenes play important roles in social evolution?," *BioEssays: news and reviews in molecular, cellular and developmental biology*. Wiley Online Library, 35(8), pp. 683–689.

Liu, B. *et al.* (2015) "Structural variation discovery in the cancer genome using next generation sequencing: Computational solutions and perspectives," *Oncotarget*, 6(8), pp. 5477–5489.

Liu, D., Hunt, M. and Tsai, I. J. (2018) "Inferring synteny between genome assemblies: a systematic evaluation," *BMC bioinformatics*, 19(1), p. 26.

Logsdon, G. A., Vollger, M. R. and Eichler, E. E. (2020) "Long-read human genome sequencing and its applications," *Nature reviews. Genetics*, 21(10), pp. 597–614.

Lorite, P. and Palomeque, T. (2010) "Karyotype evolution in ants (Hymenoptera: Formicidae), with a review of the known ant chromosome numbers," *Myrmecological news / Osterreichische Gesellschaft fur Entomofaunistik*.

Love, M. I., Huber, W. and Anders, S. (2014) "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2," *Genome biology*. Springer, 15(12), p. 550.

Löytynoja, A. and Goldman, N. (2005) "An algorithm for progressive multiple alignment of sequences with insertions," *Proceedings of the National Academy of Sciences of the United States of America*. National Acad Sciences, 102(30), pp. 10557–10562.

Martin, M. (2011) "Cutadapt removes adapter sequences from high-throughput sequencing reads," *EMBnet.journal*, 17(1), pp. 10–12.

Martin, R. C. (2009) *Clean code: a handbook of agile software craftsmanship*. Pearson Education.

Martinez-Ruiz, C. *et al.* (2020) "Genomic architecture and evolutionary antagonism drive allelic expression bias in the social supergene of red fire ants," *eLife*, 9, p. e55862.

Mather, K. (1950) "The Genetical Architecture of Heterostyly in Primula sinensis," *Evolution*, p. 340. doi: 10.2307/2405601.

McCormick, R. F., Truong, S. K. and Mullet, J. E. (2016) "3D Sorghum Reconstructions from Depth Images Identify QTL Regulating Shoot Architecture," *Plant physiology*. Am Soc Plant Biol, 172(2), pp. 823–834.

Miga, K. H. *et al.* (2020) "Telomere-to-telomere assembly of a complete human X chromosome," *Nature*, 585(7823), pp. 79–84.

Mikheenko, A. *et al.* (2018) "Versatile genome assembly evaluation with QUAST-LG," *Bioinformatics* , 34(13), pp. i142–i150.

Minio, A. *et al.* (2019) "Diploid genome assembly of the wine grape Carménère," *G3* , 9(5), p. g3.400030.2019.

Minoche, A. E. *et al.* (2015) "Exploiting single-molecule transcript sequencing for eukaryotic gene prediction," *Genome biology*, 16(1), p. 184.

Mohr, S. E. *et al.* (2014) "RNAi screening comes of age: improved techniques and complementary approaches," *Nature reviews. Molecular cell biology*. nature.com, 15(9), pp. 591–600.

Montgomery, S. H. and Mank, J. E. (2016) "Inferring regulatory change from gene expression: the confounding effects of tissue scaling," *Molecular ecology*. Wiley Online Library, 25(20), pp. 5114–5128.

Morandin, C. *et al.* (2016) "Comparative transcriptomics reveals the conserved building blocks involved in parallel evolution of diverse phenotypic traits in ants," *Genome biology*, 17(1), p. 43.

Moreau, C. S. and Bell, C. D. (2013) "Testing the museum versus cradle tropical biological diversity hypothesis: phylogeny, diversification, and ancestral biogeographic range evolution of the ants," *Evolution; international journal of organic evolution*. Wiley Online Library, 67(8), pp. 2240–2257.

Myers, E. W. *et al.* (2000) "A Whole-Genome Assembly of Drosophila," *Science*, 287(5461), pp. 2196–2204.

Nachman, M. W. (2001) "Single nucleotide polymorphisms and recombination rate in humans," *Trends in genetics: TIG*. Elsevier, 17(9), pp. 481–485.

Nadeau, N. J. (2016) "Genes controlling mimetic colour pattern variation in butterflies," *Current opinion in insect science*, 17, pp. 24–31.

Nadeau, N. J. *et al.* (2016) "The gene cortex controls mimicry and crypsis in butterflies and moths," *Nature*, 534(7605), pp. 106–110.

Nei, M. (1987) *Molecular evolutionary genetics*. Columbia university press.

Nei, M. and Rooney, A. P. (2005) "Concerted and birth-and-death evolution of multigene families," *Annual review of genetics*. annualreviews.org, 39, pp. 121–152.

Niehuis, O. *et al.* (2013) "Behavioural and genetic analyses of Nasonia shed light on the evolution of sex pheromones," *Nature*. nature.com, 494(7437), pp. 345–348.

Nipitwattanaphon, M. *et al.* (2013) "A simple genetic basis for complex social behaviour mediates widespread gene expression differences," *Molecular ecology*, 22(14), pp. 3797–3813.

Nipitwattanaphon, M. *et al.* (2014) "Effects of ploidy and sex-locus genotype on gene expression patterns in the fire ant Solenopsis invicta," *Proceedings. Biological sciences / The Royal Society*. royalsocietypublishing.org, 281(1797). doi: 10.1098/rspb.2014.1776.

Noor, M. A. F. *et al.* (2001) "Chromosomal inversions and the reproductive isolation of species," *Proceedings of the National Academy of Sciences*, pp. 12084–12088. doi: 10.1073/pnas.221274498.

Nurk, S. *et al.* (2020) "HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads," *Genome research*. Cold Spring Harbor Laboratory, p. gr.263566.120.

Nurk, S. *et al.* (2022) "The complete sequence of a human genome," *Science*, 376(6588), pp. 44–53.

Pedersen, B. S. and Quinlan, A. R. (2018) "Mosdepth: quick coverage calculation for genomes and exomes," *Bioinformatics* , 34(5), pp. 867–868.

Pelosi, P. *et al.* (2006) "Soluble proteins in insect chemical communication," *Cellular and molecular life sciences: CMLS*. Springer, 63(14), pp. 1658–1676.

Pelosi, P. *et al.* (2014) "Soluble proteins of chemical communication: an overview across arthropods," *Frontiers in physiology*. frontiersin.org, 5, p. 320.

Pfeifer, B. *et al.* (2014) "PopGenome: An Efficient Swiss Army Knife for Population Genomic Analyses in R," *Molecular Biology and Evolution*, pp. 1929–1936. doi: 10.1093/molbev/msu136.

Phillippy, A. M., Schatz, M. C. and Pop, M. (2008) "Genome assembly forensics: finding the elusive mis-assembly," *Genome biology*, 9(3), p. R55.

Pracana, R., Levantis, I., *et al.* (2017) "Fire ant social chromosomes: differences in number, sequence and expression of odorant binding proteins," *Evolution Letters*, 1(4), pp. 199–210.

Pracana, R., Priyam, A., *et al.* (2017) "The fire ant social chromosome supergene variant Sb shows low diversity but high divergence from SB," *Molecular ecology*, 26(11), pp. 2864–2879.

Privman, E., Wurm, Y. and Keller, L. (2013) "Duplication and concerted evolution in a master sex determiner under balancing selection," *Proceedings. Biological sciences / The Royal Society*, 280(1758), p. 20122968.

Priyam, A. *et al.* (2019) "Sequenceserver: a modern graphical user interface for custom BLAST databases," *Molecular biology and evolution*, p. 033142.

Prlić, A. and Procter, J. B. (2012) "Ten simple rules for the open development of scientific software," *PLoS computational biology*. journals.plos.org, 8(12), p. e1002802.

Purcell, J. *et al.* (2014) "Convergent genetic architecture underlies social organization in ants," *Current biology: CB*, 24(22), pp. 2728–2732.

Quinlan, A. R. and Hall, I. M. (2010) "BEDTools: a flexible suite of utilities for comparing genomic features," *Bioinformatics* , 26(6), pp. 841–842.

R Core Team (2014) *R: A language and environment for statistical computing*. Available at: http://www.R-project.org/.

Rahman, A. and Pachter, L. (2013) "CGAL: computing genome assembly likelihoods," *Genome biology*, 14(1), p. R8.

Raymond, E. S. (2003) *The Art of UNIX Programming*. Pearson Education.

Raymond, O. *et al.* (2018) "The Rosa genome provides new insights into the domestication of modern roses," *Nature genetics*, 50(6), pp. 772–777.

Reese, W. (2008) "Nginx: the high-performance web server and reverse proxy," *Linux Journal*. Houston, TX: Belltown Media (2), 2008(173), p. 2.

Reichler, S. J. *et al.* (2018) "Pseudomonas fluorescens group bacterial strains are responsible for repeat and sporadic postpasteurization contamination and reduced fluid milk shelf life," *Journal of dairy science*. Elsevier, 101(9), pp. 7780–7800.

Rhoads, A. and Au, K. F. (2015) "PacBio Sequencing and Its Applications," *Genomics, proteomics & bioinformatics*, 13(5), pp. 278–289.

Riba-Grognuz, O. *et al.* (2011) "Visualization and quality assessment of de novo genome assemblies," *Bioinformatics* , 27(24), pp. 3425–3426.

Roach, M. J., Schmidt, S. A. and Borneman, A. R. (2018) "Purge Haplotigs: allelic contig reassignment for third-gen diploid genome assemblies," *BMC bioinformatics*, 19(1), p. 460.

Robertson, H. M., Warr, C. G. and Carlson, J. R. (2003) "Molecular evolution of the insect chemoreceptor gene superfamily in Drosophila melanogaster," *Proceedings of the National Academy of Sciences of the United States of America*. National Acad Sciences, 100 Suppl 2, pp. 14537–14542.

Robinson, G. E., Grozinger, C. M. and Whitfield, C. W. (2005) "Sociogenomics: social life in molecular terms," *Nature reviews. Genetics*. nature.com, 6(4), pp. 257–270.

Robinson, S. W. *et al.* (2013) "FlyAtlas: database of gene expression in the tissues of Drosophila melanogaster," *Nucleic acids research*. academic.oup.com, 41(Database issue), pp. D744-50.

Rochette, N. C., Rivera-Colón, A. G. and Catchen, J. M. (2019) "Stacks 2: Analytical methods for paired-end sequencing improve RADseq-based population genomics," *Molecular ecology*, 28(21), pp. 4737–4754.

Roff, D. A., Stirling, G. and Fairbairn, D. J. (1997) "The Evolution Of Threshold Traits: A Quantitative Genetic Analysis Of The Physiological And Life-history Correlates Of Wing Dimorphism In The Sand Cricket," *Evolution; international journal of organic evolution*, 51(6), pp. 1910–1919.

Ross, K. G. (1997) "Multilocus evolution in fire ants: effects of selection, gene flow and recombination," *Genetics*. Genetics Soc America, 145(4), pp. 961–974.

Ross, K. G. and Keller, L. (1995) "Ecology and evolution of social organization: Insights from fire ants and other highly eusocial insects," *Annual review of ecology and systematics*. Annual Reviews, 26(1), pp. 631–656.

Ross, K. G. and Keller, L. (1998) "Genetic control of social organization in an ant," *Proceedings of the National Academy of Sciences of the United States of America*. National Acad Sciences, 95(24), pp. 14232–14237.

Ross, K. and Keller, L. (2002) "Experimental conversion of colony social organization by manipulation of worker genotype composition in fire ants (Solenopsis invicta)," *Behavioral ecology and sociobiology*. Springer, 51(3), pp. 287–295.

Ross, M. G. *et al.* (2013) "Characterizing and measuring bias in sequence data," *Genome biology*, 14(5), p. R51.

Ruan, J. and Li, H. (2020) "Fast and accurate long-read assembly with wtdbg2," *Nature methods*, 17(2), pp. 155–158.

Rubenstein, D. R. *et al.* (2019) "Coevolution of Genome Architecture and Social Behavior," *Trends in ecology & evolution*, 34(9), pp. 844–855.

Ruby, S., Copeland, D. B. and Thomas, D. (2020) *Agile Web Development with Rails 6*. Pragmatic Bookshelf.

Salzberg, S. L. *et al.* (2012) "GAGE: A critical evaluation of genome assemblies and assembly algorithms," *Genome research*, 22(3), pp. 557–567.

Sametinger, J. (1997) *Software Engineering with Reusable Components*. Springer Science & Business Media.

Schatz, M. C., Delcher, A. L. and Salzberg, S. L. (2010) "Assembly of large genomes using second-generation sequencing," *Genome research*, 20(9), pp. 1165–1173.

Schirmer, M. *et al.* (2016) "Illumina error profiles: resolving fine-scale variation in metagenomic sequencing data," *BMC bioinformatics*, 17, p. 125.

Schneider, V. A. *et al.* (2017) "Evaluation of GRCh38 and de novo haploid genome assemblies demonstrates the enduring quality of the reference assembly," *Genome research*, 27(5), pp. 849–864.

Schnoes, A. M. *et al.* (2009) "Annotation error in public databases: misannotation of molecular function in enzyme superfamilies," *PLoS computational biology*. journals.plos.org, 5(12), p. e1000605.

Schwander, T., Libbrecht, R. and Keller, L. (2014) "Supergenes and complex phenotypes," *Current biology: CB*. Elsevier, 24(7), pp. R288-94.

Seifert, B. (2016) "Inconvenient hyperdiversity – the traditional concept of 'Pheidole pallidula' includes four cryptic species (Hymenoptera: Formicidae)," *Soil organisms*, 88(1), pp. 1–17.

Seim, I. *et al.* (2017) "Whole-Genome Sequence of the Metastatic PC3 and LNCaP Human Prostate Cancer Cell Lines," *G3* . g3journal.org, 7(6), pp. 1731–1741.

Shen, X.-X. *et al.* (2016) "Reconstructing the Backbone of the Saccharomycotina Yeast Phylogeny Using Genome-Scale Data," *G3* . g3journal.org, 6(12), pp. 3927–3939.

Shields, E. J. *et al.* (2018) "High-Quality Genome Assemblies Reveal Long Non-coding RNAs Expressed in Ant Brains," *Cell reports*, 23(10), pp. 3078–3090.

Shore, J. and Warden, S. (2007) *The Art of Agile Development: Pragmatic Guide to Agile Software Development*. "O'Reilly Media, Inc."

Shumate, A. and Salzberg, S. L. (2020) "Liftoff: an accurate gene annotation mapping tool," *Cold Spring Harbor Laboratory*. doi: 10.1101/2020.06.24.169680.

Simão, F. A. *et al.* (2015) "BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs," *Bioinformatics* , 31(19), pp. 3210–3212.

Skinner, M. E. *et al.* (2009) "JBrowse: a next-generation genome browser," *Genome research*. Cold Spring Harbor Laboratory, 19(9), pp. 1630–1638.

Slater, G. S. C. and Birney, E. (2005) "Automated generation of heuristics for biological sequence comparison," *BMC bioinformatics*, 6, p. 31.

Smith, J. M. and Haigh, J. (1974) "The hitch-hiking effect of a favourable gene," *Genetical research*. cambridge.org, 23(1), pp. 23–35.

Soneson, C., Love, M. I. and Robinson, M. D. (2015) "Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences," *F1000Research*, 4, p. 1521.

Srivastava, A. *et al.* (2019) "Genome assembly and gene expression in the American black bear provides new insights into the renal response to hibernation," *DNA research: an international journal for rapid publication of reports on genes and genomes*, 26(1), pp. 37–44.

Stamatakis, A. (2006) "RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models," *Bioinformatics* . academic.oup.com, 22(21), pp. 2688–2690.

Stanke, M. *et al.* (2006) "Gene prediction in eukaryotes with a generalized hidden Markov model that uses hints from external sources," *BMC bioinformatics*, 7(1), p. 62.

Stanke, M. *et al.* (2008) "Using native and syntenically mapped cDNA alignments to improve de novo gene finding," *Bioinformatics* , 24(5), pp. 637–644.

Stearns, F. W. (2010) "One hundred years of pleiotropy: a retrospective," *Genetics*, 186(3), pp. 767–773.

Steijger, T. *et al.* (2013) "Assessment of transcript reconstruction methods for RNA-seq," *Nature methods*, 10(12), pp. 1177–1184.

Stolle, E. *et al.* (2019) "Degenerative Expansion of a Young Supergene," *Molecular biology and evolution*, 36(3), pp. 553–561.

Suen, G. *et al.* (2011) "The genome sequence of the leaf-cutter ant Atta cephalotes reveals insights into its obligate symbiotic lifestyle," *PLoS genetics*, 7(2), p. e1002007.

Sun, Y.-L. *et al.* (2012) "Expression in antennae and reproductive organs suggests a dual role of an odorant-binding protein in two sibling Helicoverpa species," *PloS one*. journals.plos.org, 7(1), p. e30040.

Suzek, B. E. *et al.* (2015) "UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches," *Bioinformatics* , 31(6), pp. 926–932.

Tan, J. A. and Mikheyev, A. S. (no date) "A scaled-down workflow for Illumina shotgun sequencing library preparation: lower input and improved performance at small fraction of the cost." doi: 10.7287/peerj.preprints.2475.

Tang, H. *et al.* (2015) "ALLMAPS: robust scaffold ordering based on multiple maps," *Genome biology*, 16(1), p. 3.

Tange, O. (2011) "GNU parallel - the command-line power tool," *The USENIX Magazine*, 36, pp. 42–47.

The Uniprot Consortium (2015) "UniProt: a hub for protein information," *Nucleic acids research*, 43(Database issue), pp. D204-12.

The Uniprot Consortium (2017) "UniProt: the universal protein knowledgebase," *Nucleic acids research*, 45(D1), pp. D158–D169.

Thomas, G. W. C. and Hahn, M. W. (2019) "Referee: Reference Assembly Quality Scores," *Genome biology and evolution*, 11(5), pp. 1483–1486.

Thompson, M. J. and Jiggins, C. D. (2014) "Supergenes and their role in evolution," *Heredity*. nature.com, 113(1), pp. 1–8.

Thorvaldsdóttir, H., Robinson, J. T. and Mesirov, J. P. (2013) "Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration," *Briefings in bioinformatics*, 14(2), pp. 178–192.

Trible, W. and Ross, K. G. (2016) "Chemical communication of queen supergene status in an ant," *Journal of evolutionary biology*. Wiley Online Library, 29(3), pp. 502–513.

Tschinkel, W. R. (2006) *The Fire Ants*. Harvard University Press.

Tsutsui, N. D. *et al.* (2008) "The evolution of genome size in ants," *BMC evolutionary biology*, 8, p. 64.

Venthur, H. and Zhou, J.-J. (2018) "Odorant Receptors and Odorant-Binding Proteins as Insect Pest Control Targets: A Comparative Analysis," *Frontiers in physiology*, 9, p. 1163.

Vieira, F. G. and Rozas, J. (2011) "Comparative genomics of the odorant-binding and chemosensory protein gene families across the Arthropoda: origin and evolutionary history of the chemosensory system," *Genome biology and evolution*. academic.oup.com, 3, pp. 476–490.

Vieira, F. G., Sánchez-Gracia, A. and Rozas, J. (2007) "Comparative genomic analysis of the odorant-binding protein family in 12 Drosophila genomes: purifying selection and birth-and-death evolution," *Genome biology*. Springer, 8(11), p. R235.

Vollger, M. R. *et al.* (2019) "Long-read sequence and assembly of segmental duplications," *Nature methods*, 16(1), pp. 88–94.

Walker, B. J. *et al.* (2014) "Pilon: An Integrated Tool for Comprehensive Microbial Variant Detection and Genome Assembly Improvement," *PloS one*, 9(11), p. e112963.

Wang, J. *et al.* (2007) "An annotated cDNA library and microarray for large-scale gene-expression studies in the ant Solenopsis invicta," *Genome biology*. Springer, 8(1), p. R9.

Wang, J. *et al.* (2013) "A Y-like social chromosome causes alternative colony organization in fire ants," *Nature*, 493(7434), pp. 664–668.

Wang, J., Ross, K. G. and Keller, L. (2008) "Genome-wide expression patterns and the genetic architecture of a fundamental social trait," *PLoS genetics*. journals.plos.org, 4(7), p. e1000127.

Ward, P. S., Brady, Sean G., *et al.* (2015) "The evolution of myrmicine ants: phylogeny and biogeography of a hyperdiverse ant clade (H ymenoptera: F ormicidae)," *Systematic entomology*. Wiley Online Library, 40(1), pp. 61–81.

Ward, P. S., Brady, Seán G., *et al.* (2015) "The evolution of myrmicine ants: phylogeny and biogeography of a hyperdiverse ant clade (Hymenoptera: Formicidae)," *Systematic Entomology*, pp. 61–81. doi: 10.1111/syen.12090.

Warr, A. *et al.* (2015) "Identification of Low-Confidence Regions in the Pig Reference Genome (Sscrofa10.2)," *Frontiers in genetics*, 6, p. 338.

Weirather, J. L. *et al.* (2017) "Comprehensive comparison of Pacific Biosciences and Oxford Nanopore Technologies and their applications to transcriptome analysis," *F1000Research*, 6, p. 100.

Weisfeld, M. (2013) *The Object-Oriented Thought Process*. 4th ed. Addison-Wesley Professional.

West, S. A. and Gardner, A. (2010) "Altruism, spite, and greenbeards," *Science*. science.sciencemag.org, 327(5971), pp. 1341–1344.

Wicker, T. *et al.* (2018) "Impact of transposable elements on genome structure and evolution in bread wheat," *Genome biology*, 19(1), p. 363192.

Wilson, G. *et al.* (2014) "Best practices for scientific computing," *PLoS biology*. journals.plos.org, 12(1), p. e1001745.

Winnenburg, R. *et al.* (2006) "PHI-base: a new database for pathogen host interactions," *Nucleic acids research*. academic.oup.com, 34(Database issue), pp. D459-64.

Wintersinger, J. A. and Wasmuth, J. D. (2015) "Kablammo: an interactive, web-based BLAST results visualizer," *Bioinformatics* . academic.oup.com, 31(8), pp. 1305–1306.

Wood, D. E., Lu, J. and Langmead, B. (2019) "Improved metagenomic analysis with Kraken 2," *Genome biology*, 20(1), p. 257.

Wu, Y. *et al.* (2008) "Efficient and Accurate Construction of Genetic Linkage Maps from the Minimum Spanning Tree of a Graph," *PLoS genetics*, 4(10), p. e1000212.

Wurm, Y. *et al.* (2009) "Fourmidable: a database for ant genomics," *BMC genomics*, 10, p. 5.

Wurm, Y. *et al.* (2011) "The genome of the fire ant Solenopsis invicta," *Proceedings of the National Academy of Sciences*, 108(14), pp. 5679–5684.

Wurm, Y. (2015) “Avoid having to retract your genomics analysis,” *The Winnower*, 2, p. e143696.

Xu, P. X., Zwiebel, L. J. and Smith, D. P. (2003) “Identification of a distinct family of genes encoding atypical odorant-binding proteins in the malaria vector mosquito, Anopheles gambiae,” *Insect molecular biology*. Wiley, 12(6), pp. 549–560.

Yan, Z. *et al.* (2020) “Evolution of a supergene that regulates a trans-species social polymorphism,” *Nature ecology & evolution*, 4(2), pp. 240–249.

Yandell, M. and Ence, D. (2012) “A beginner’s guide to eukaryotic genome annotation,” *Nature reviews. Genetics*, 13(5), pp. 329–342.

Yang, X. *et al.* (2013) “HTQC: a fast quality control toolkit for Illumina sequencing data,” *BMC bioinformatics*, 14, p. 33.

Yao, E. *et al.* (2020) “JBrowse Connect: A server API to connect JBrowse instances and users,” *PLoS computational biology*, 16(8), p. e1007261.

Ye, L. *et al.* (2011) “A vertebrate case study of the quality of assemblies derived from next-generation sequences,” *Genome biology*, 12(3), p. R31.

Zhang, H., Jain, C. and Aluru, S. (2019) “A comprehensive evaluation of long read error correction methods,” *bioRxiv*. doi: 10.1101/519330.

Zhang, S. V., Zhuo, L. and Hahn, M. W. (2016) “AGOUTI: improving genome assembly and annotation using transcriptome data,” *GigaScience*, 5(1), p. 31.

Zhang, W. *et al.* (2016) “Tissue, developmental, and caste-specific expression of odorant binding proteins in a eusocial insect, the red imported fire ant, Solenopsis invicta,” *Scientific reports*. nature.com, 6, p. 35452.

Zhao, H. *et al.* (2014) “CrossMap: a versatile tool for coordinate conversion between genome assemblies,” *Bioinformatics* , 30(7), pp. 1006–1007.

Zhou, X. *et al.* (2012) “Phylogenetic and transcriptomic analysis of chemosensory receptors in a pair of divergent ant species reveals sex-specific signatures of odor coding,” *PLoS genetics*. journals.plos.org, 8(8), p. e1002930.