



FACULTÉ
DES SCIENCES



UNIVERSITÉ LIBRE DE BRUXELLES

Improving genome assemblies of non-model non-vertebrate animals with long reads and Hi-C

Thesis presented by Nadège GUIGLIELMONI

in fulfillment of the requirements of the

PhD degree in "Docteur en Sciences"

Année académique 2020-2021

Supervisor: Jean-François FLOT
Co-supervisor: Romain KOSZUL

Thesis jury :

Patrick Mardulyn (Université libre de Bruxelles, Chair)
Guillaume Smits (Université libre de Bruxelles, Secretary)
Michael Eitel (Ludwig-Maximilians-Universität)
Ann McCartney (National Institutes of Health)



Acknowledgements

The first individual I would like to thank is not a human, but my dog Curie. He became a part of my life on 16 April 2019, thanks to the charity Vivre Libre, and he became pivotal in the completion of my PhD, as he made sure that I would get two walks per day and made every day easier with his cuteness. He interacted with several collaborators and followed me to Munich, at a time when I was so stressed to get valuable data for my PhD. During the coronavirus pandemic, he was the only one sharing my office. Thanks to him, I met at the Wolvendael Park an incredible group: Alain Bourguignon, Benoit Dethiège, Caroline Gréant, Chloé Mertens, Gwen Legein, Julie Declerck, Ludovic Berghmans, Nadia Swaelens, and last but not least, Valentine Legrand. We stuck together when the coronavirus changed all our lives and we became a tight group. I will cherish the memories of our Friday drinks together and the birthdays we celebrated. Our group made Uccle feel like a home that I would be happy to return to.

I would like to thank my supervisors Jean-François Flot and Romain Koszul. Jean-François Flot welcomed me into his team and gave me the means to complete my PhD. I do hope that I lived up to his expectations, and even surpassed them. Funnily, Jean-François and I have opposite opinions on many subjects, which, I think, strengthened our work together. He made me benefit from his wide network and made sure that I would stay busy for many years to come. Romain Koszul welcomed me in his team full-time during the first three months of my PhD. I benefited from his huge knowledge of Hi-C with Aurèle Piazza, Axel Cournac, Cyril Matthey-Doret, Lyam Baudry, Martial Marbouty and Vittore Scolari. More specifically, Cyril Matthey-Doret started his PhD a little after I did, and I always saw him as a comrade-in-arms. I also want to thank Agnès Thierry and Christophe Chopard who took care of my Hi-C libraries and made my projects move forward.

I was also well surrounded at the Université libre de Bruxelles and in the EEG team, in particular with Ana Rodriguez-Jimenez, Catalina Ramírez-Portilla and Claire Chauveau. I had the opportunity to su-

pervise Roland Faure during a Master degree internship, and to develop GraphUnzip together; I have no doubt he will succeed in his upcoming PhD too. I want to thank everybody in EBE, Serge Aron, Patrick Mardulyn, Olivier Hardy, Claire Baudoux, Arthur Boom, Nicolas Fontaine, Tania d'Haijere, Nicolas Kaczamrek, Svitlana Lukicheva, Katarina Matvijev, Jérémy Migliore, Florence Rodriguez, Maeva Sorel, for the lunch breaks, and the sadly few game nights and evenings at the Tavernier. I owe many thanks to Laurent Grumiau, without whom I would not have been able to get any experiment done.

I was lucky to be part of the Innovative Training Network IGNITE. I remember how intimidated I was during the first Network-wide Training Event, as I had so little knowledge of non-vertebrates due to my bachelor degree in Molecular Biology. I tried to make up for it during my PhD, and I hope that I succeeded. In this program, Michael Eitel was an incredible resource to rely on. I gave him the nickname of the "Babysitter", as his job of Project Manager made him in charge of all IGNITE students. Michi was an anchor through this whole project, as he gave me advice about Nanopore DNA sequencing and RNA sequencing, and I hope that more people will be lucky to receive his advice in the future. Besides, Ramon Rivera-Vicéns and Ferenc Kagan have been my closest friends in the program. One night, after some partying in Split, they took me each by one arm to celebrate together being members of IGNITE, and since we have called ourselves the "Party People". I hope we will have more parties together, whatever the time or the place.

I had many collaborations during my PhD. I remember meetings in my living room with Anne Guichard, Kathryn Stankiewicz and Ksenia Juravel at the beginning of March 2020, shortly before COVID-19 disturbed everybody's life. Among all this madness, we were able with Kathryn to make our genome assembly of *Astrangia poculata* work and reach chromosome level, although I regret that Kathryn was not able to try all the sorts of cheese she should have. I was also fortunate to work on sponges with Antonio Ruiz and Kenneth Sandoval, on cones with Yihe Zhao, and on mollusks with Zeyuan Chen.

I want to thank my friends Lola Champesme and Quentin Schumacher, who did not abandon me even when I was not the best person, and my unusual friend Antoine Régnier, who is my secret IT support. I certainly do not call the three of you enough, but I never stop caring about you. Finally, I thank my whole family, even the ones that disappeared along the way, and particularly my grandparents, Josette and André Maury, who welcomed me into their home during the first years of my college studies. I dedicate this thesis to my father, Jean-Claude Guiglielmoni, who has always been a quiet, strong and

loving presence.

Abstract

The corpus of reference genomes is rapidly expanding as more and more genome assemblies are released for a wide variety of species. The constant progress in sequencing technologies has led to the release in 2021 of a first complete, telomere-to-telomere, gap-less assembly of a human genome, yet a myriad of eukaryote species still lack genomic resources. For animals, genomic projects have focused on species closely related to humans (vertebrates) and those with an impact on health and agriculture. By contrast, there is still a dearth of non-vertebrate genomes that poorly represents their tremendous diversity (about 95% of animal diversity).

Haploid chromosome-level genome assemblies using long reads and chromosome conformation capture (such as Hi-C) have become a standard in recent publications. To provide a haploid representation of diploid and polyploid genomes, assemblers collapse haplotypes into a single sequence, yet they are sensitive to high levels of heterozygosity and often yield fragmented assemblies with artefactual duplications. I tackled these shortcomings with two strategies: improving collapsed assemblies with a comprehensive long-read assembly methodology tuned for highly heterozygous genomes; and separating haplotypes to obtain phased assemblies using long reads and Hi-C. The assemblies were finally brought to chromosome-level scaffolds with a new Hi-C scaffolder, which demonstrated its efficiency on genomes of non-model organisms.

These methods were applied to generate chromosome-level assemblies of three species for which none or few assemblies of closely related species were available: the bdelloid rotifer *Adineta vaga*, the coral *Astrangia poculata*, and the chaetognath *Flaccisagitta enflata*. These high-quality assemblies contribute to filling the current gaps in non-vertebrate genomics and pave the way for future sequencing initiatives aiming to generate such reference assemblies for all the species on Earth.

Scientific communications

Peer-reviewed publications

Nadège Guiguelmoni, Antoine Houtain, Alessandro Derzelle, Karine Van Doninck, Jean-François Flot. Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms. *BMC Bioinformatics* (2021).

Lyam Baudry, **Nadège Guiguelmoni**, Hervé Marie-Nelly, Alexandre Cormier, Martial Marbouty, Komlan Avia, Yann Loe Mie, Olivier Godfroy, Lieven Sterck, J Mark Cock, Christophe Zimmer, Susana M Coelho, Romain Koszul. instaGRAAL: chromosome-level quality scaffolding of genomes using a proximity ligation-based scaffold. *Genome Biology* (2020).

Preprints

Roland Faure, **Nadège Guiguelmoni**, Jean-François Flot. GraphUnzip: unzipping assembly graphs with long reads and Hi-C. *bioRxiv* (2021).

Zeyuan Chen, Özgül Doğan, **Nadège Guiguelmoni**, Anne Guichard, Michael Schrödl. The *de novo* genome of the "Spanish" slug *Arion vulgaris* Moquin-Tandon, 1855 (Gastropoda: Panpulmonata): massive expansion of transposable elements in a major pest species. *bioRxiv* (2020).

Paul Simion, Jitendra Narayan, Antoine Houtain, Alessandro Derzelle, Lyam Baudry, Emilien Nicolas, Rohan Arora, Marie Cariou, Corinne Cruaud, Florence Rodriguez Gaudray, Clément Gilbert, **Nadège Guiguelmoni**, Boris Hespeels, Djampa Kozłowski, Karine Labadie, Antoine Limasset, Marc Llrós, Martial Marbouty, Matthieu Terwagne, Julie Virgo, Richard Cordaux, Etienne GJ Danchin, Bernard Hallet, Romain Koszul, Jean-François Flot, Karine Van Doninck. Homologous chromosomes in asexual

rotifer *Adineta vaga* suggest automixis. *bioRxiv* (2020).

Talks

Roland Faure, **Nadège Guiglielmoni**, Jean-François Flot. GraphUnzip: unzipping assembly graphs with long reads and Hi-C. *Journées Ouvertes en Biologie, Informatique et Mathématiques* (2021).

Nadège Guiglielmoni, Roland Faure, Jean-François Flot. hic2gfa: unzipping assembly graphs with chromosome conformation capture. *3D Genomics* (2020).

Nadège Guiglielmoni, Jean-François Flot. How to crack the genomes of non-model invertebrates: lessons from coral and rotifer genome projects. *Biodiversity Genomics* (2020).

Nadège Guiglielmoni, Antoine Houtain, Alessandro Derzelle, Karine Van Doninck, Jean-François Flot. Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms. *Journées Ouvertes en Biologie, Informatique et Mathématiques* (2020). Best talk award.

Posters

Nadège Guiglielmoni, Antoine Houtain, Alessandro Derzelle, Karine Van Doninck, Jean-François Flot. Overcoming uncollapsed haplotypes in Nanopore assemblies. *London Calling* (2021).

Nadège Guiglielmoni, Romain Koszul, Jean-François Flot. Benchmarking Hi-C scaffolders. *Journées Ouvertes en Biologie, Informatique et Mathématiques* (2019).

Contents

1	Introduction	1
1.1	Sequencing	4
1.2	Genome assembly	6
1.3	Assembly pre and post-processing	11
1.4	Assemblies evaluation	15
1.5	Phasing assemblies	19
1.6	Outline of the thesis	20
2	Benchmark of long-read assemblers on the genome of the bdelloid rotifer <i>Adineta vaga</i>	22
3	Unzipping assembly graphs with long reads and Hi-C	77
3.1	Introduction	77
3.2	Methods	79
3.3	Results	82
3.4	Conclusion	84
4	Scaffolding assemblies with Hi-C	85

5	Hi-C scaffolding of the bdelloid rotifer <i>Adineta vaga</i>	120
6	Genome assembly of the coral <i>Astrangia poculata</i>	172
6.1	Introduction	172
6.2	Material & Method	173
6.3	Results	174
6.4	Discussion	177
7	Genome assembly of a chaetognath	180
7.1	Introduction	180
7.2	Material & Method	181
7.3	Results	183
7.4	Discussion	188
8	Discussion & Conclusion	189
8.1	Genome assemblies of non-vertebrate animals	189
8.2	Combining long reads and Hi-C for chromosome-level assemblies	190
8.3	Defining a new benchmark dataset: <i>Adineta vaga</i>	191
8.4	Decreasing long-read sequencing depth	192
8.5	Phasing assemblies	192
8.6	Reproducibility in genome projects	193

Chapter 1

Introduction

This introduction is from a review of genomes assemblies of non-vertebrate animals, in preparation with Ramon E. Rivera-Vicéns, Romain Koszul and Jean-François Flot.

The field of genomics is presently thriving, with new genomes of all kind of organisms becoming available every day. For Metazoa, efforts have unsurprisingly focused on human's closest relatives (i.e., vertebrates) so far [1]: out of 5,994 metazoan assemblies available in the NCBI database (accessed on April 21st, 2021) [2], $\sim 67.5\%$ (3,809) belong to the subphylum Vertebrata. However, from the currently ~ 2.1 million described metazoan species, only $\sim 73,000$ (3.5%) belong to vertebrates [3]. The remaining metazoan phyla, hereafter called "non-vertebrate animals", are thus underinvestigated and lack genetic resources.

Non-vertebrate animals are found in nearly all known terrestrial and aquatic ecosystems (both marine and freshwater), and represent the diverse branches of the metazoan tree of life (among which vertebrates are just a twig that originated about 600 millions years ago [4]). Characterizing the genome structure and gene content of non-vertebrate animals is therefore pivotal for expanding our knowledge regarding the evolution, ecology and biodiversity of metazoans.

In recent years, important sequencing efforts have started to tackle the dearth of genomic data for non-vertebrate animals, with a strong focus on arthropods (1,279 assemblies on NCBI). The phylum Arthropoda is very diverse: it consists of more than 1.3 million species, the majority of which belong to the class Insecta (~ 1 million species) [5]. Insects have a significant impact on agriculture (e.g. as

crop pests) and on the transmission of diseases (e.g. malaria and dengue) [6]. They also play important beneficial and regulatory roles in natural ecosystems, through pollination and decomposition of organic matter [7]. Genome sequencing yields invaluable insights into species that are key in the aforementioned processes. For example, various genome projects have targeted insects such as *Bemisia tabaci*, a common crop pest [8], and the mosquitoes *Aedes aegypti* (vector of yellow fever, dengue and chikungunya) [9] and *Anopheles darlingi* (vector of malaria) [10]. These studies unveiled, among other findings, expansions of genes involved in insecticide resistance. The genomes of these species are so important for human health and food security that many have actually been sequenced multiple times, either because of the availability of newer sequencing methods or to compare different strains (for instance, three versions of the genome of *Aedes aegypti* [11, 12, 9] were successively published). Many phyla with less direct human implications, however, do not even have a single good-quality genome assembly available to date (e.g., chaetognaths).

Many other non-vertebrates (and their symbionts) have also shown tremendous importance and relevance with respect to socio-economic impact. Snails, sponges and corals all produce metabolites with biological activities such as anticancer, anti-inflammatory, antibacterial, among others [13, 14, 15]. Terpenoid metabolites have been found in more than 70 gastropods species [16]. In sponges, compounds such as polyketides, terpenoids and alkaloids have also been found in species of the genera *Haliclona*, *Petrosia*, and *Discodemia*, these three genera being the richest among sponges in terms of bioactive compounds [17]. Thus, genome assemblies are essential to identify and better understand the genes, pathways and sources of these compounds. Among mollusks, several species valued as food resources are studied for their impact in aquaculture [18]. Moreover, non-vertebrates are important model systems to understand processes such as adaptation to climate change, ocean acidification, biomineralization [19, 20, 21, 22]. Various species of corals [23, 24, 25, 26] have been sequenced to study the effects of increasing seawater temperatures and to understand how these species may survive in changing environments.

Some genome projects are motivated by more theoretical questions, to improve species classification and elucidate specific traits. Genome assemblies provide abundant sets of genes to build robust phylogenetic trees, opening the field of phylogenomics [27]. New genome resources bring novel insights into difficult phylogenetic positions: a large analysis based on genomes and transcriptomes confirmed that myxozoans belonged to Cnidaria [28]; the sequence of *Hoilungia hongkongiensis* placed placozoans as a sister group to cnidarians and bilaterians [29].

The lack of non-vertebrate genomic resources may be blamed to the difficulty to collect individuals or extract pure, high-molecular-weight DNA, as well as to their frequently large genomes characterized by high repetitive contents and high heterozygosity. However, sequencing technologies now offer cost-effective solutions and wide applicability to solve some of these problems. Reducing the current unbalance in genomic resources between vertebrates and non-vertebrate animals will increase the precision of future tools and studies. Indeed, genome data is often used as the foundation for different genomic and protein databases. The program BUSCO (Benchmarking Universal Single-Copy Orthologs) [30], used to measure the completeness of a genome assembly, relies on genomic data to build reference gene sets that are used for scoring. It uses hidden Markov models to detect orthologs that are shared by $\geq 90\%$ of the species in a given clade. Thus, results from under-sampled groups could change drastically when more species are added to the gene sets. These could also have major effects in analyses such as phylogenomics, protein families studies and of gene duplication events. Another consequence of the current dearth of genomic resources for non-vertebrate animals is that BLAST [31] searches for these organisms most often recover vertebrate and arthropod hits, even though the target species is distant from these phyla, making difficult the identification of sequences from a species lacking a reference or closely related genome.

It is therefore imperative to explore thoroughly the diversity of metazoans, specifically from non-vertebrates species. International consortia such as the Global Invertebrate Genomics Alliance (GIGA) [32, 33] have been put in place to overcome some of the aforementioned limitations. Other consortia such as the Earth BioGenome Project [34], the Darwin Tree of Life [35], the Aquatic Symbiosis Genomics Project [36] and the European Reference Genome Atlas [37] are also expected to significantly boost the genomic resources of non-vertebrates in the near future. Undoubtedly, these projects will benefit from the drastic improvements in sequencing technologies over the last years.

This chapter introduces the current state of genome assemblies of non-vertebrate species. The first part presents a summary of available sequencing technologies, followed by a description of common assembly algorithms and an inventory of tools for assembly pipelines. Assembly evaluation methods are then detailed to identify correct assemblies. The last section opens on strategies for phased assemblies.

1.1 Sequencing

Sequencing technologies have dramatically evolved over the last two decades, providing researchers with various options when it comes to tackling a genome project (Table 1.1). Sanger sequencing, the widely used sequencing method with chain-terminating inhibitors published in 1977, produces reads around 1,000 basepair long (bp) with an error rate of about 1% [38]. The principle is to synthesize complementary strands of DNA from a single strand with a mixture of regular nucleotides and dideoxynucleotides, the latter stopping the polymerase when incorporated. Four reactions are performed for each type of base, and the resulting oligonucleotides are migrated by electrophoresis to identify the correct base at every position and generate a read. This method laid the foundations for DNA sequencing and was used extensively in several genome assemblies projects, which were at that time typically ran by large international consortia: the budding yeast *Saccharomyces cerevisiae* [39] was the first eukaryote sequenced, whereas the nematode *Caenorhabditis elegans* was the first metazoan [40]. Sanger sequencing is a relatively low-throughput method in terms of the number of sequences generated, and is costly as well [41]. Although it is almost not used in genome projects anymore, the technology was pivotal for the generation of the first assembly of the human genome published in 2001, a monumental effort by 20 sequencing centers, to an estimated cost of 300 million US dollars [42].

Second-generation sequencing technologies, initially called next-generation sequencing (NGS), are characterized by a strong increase in sequencing throughputs compared to the Sanger method, with millions of DNA fragments sequenced simultaneously. NGS reads are much smaller than Sanger reads (from 110 bp in for the first 454 machine in 2005 up to to 350 bp for MiSeq Illumina machine nowadays), resulting in the need for new analysis algorithms and programs[43]. Nevertheless, the arrival of NGS sequencing democratized genome assembly projects, broadening the scope of investigated species beyond well-studied model organisms. Several second-generation sequencing methods have emerged through the years, some of which have since then been discontinued: 454 pyrosequencing [44], Ion Torrent [45], SOLiD [46], and Solexa (for a comparison on the approaches, see [47]). Among these methods, Solexa, subsequently purchased by Illumina [48], became and remains the most widely used approach to this day. This approach consists in amplifying short DNA molecules bound on a flow cell, and sequencing them by the sequential addition of fluorescently tagged nucleotides. This protocol generates highly accurate single or paired-end reads with a length up to a few hundred bases. The recent NovaSeq system further increased the output from a single run and abated the cost (up to 3 Terabases per flowcell). Short reads stimulated the whole field of genomics, and led to a large production of assemblies for all sorts of organisms, up to this day

(Figure 1.1). These short-reads based assemblies resulted in a tremendous increase of genomic resources, which remained typically quite fragmented (with N50s below 1 Megabase (Mb)).

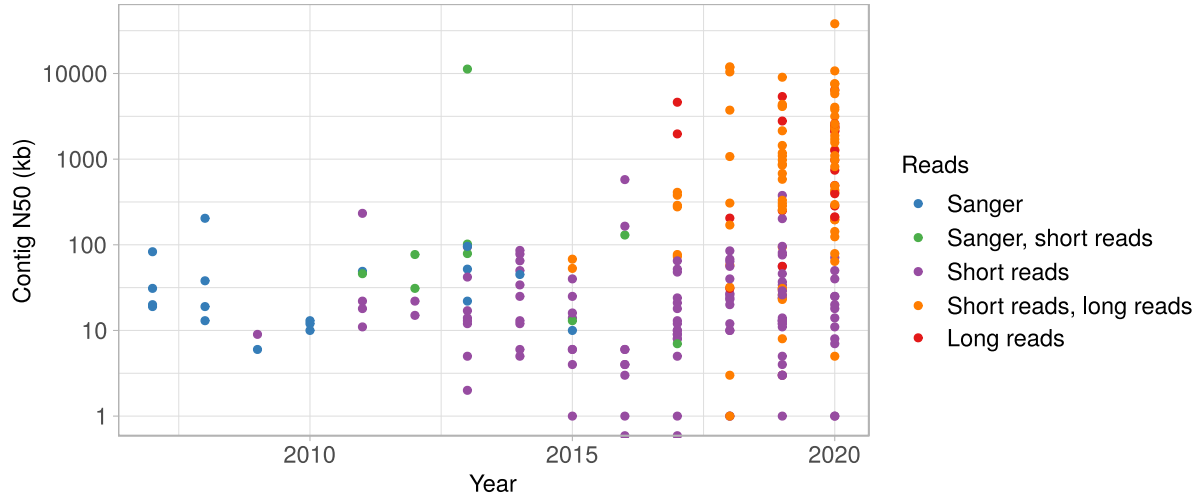


Figure 1.1: Contig N50 of non-vertebrate genome assemblies over time. The N50 represents the contiguity of an assembly and is defined as the length of the largest contig for which at least 50% of the assembly size is contained in contigs equal or greater in length.

Third-generation sequencing has brought a whole new range of sequencing data, with the sequencing of long DNA molecules extending up to hundreds of thousands of bases [49]. The two main players in the field, Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (Nanopore), use two different kinds of technologies. PacBio developed Single Molecule Real-Time (SMRT) sequencing, where a complementary strand of DNA is produced from a single strand by addition of fluorescently labeled nucleotides. The fluorescent tag is released and the luminescence is interpreted as a base [50]. The resulting reads have a length around twenty kilobases (kb) and a high error rate, an issue recently addressed by the introduction of an extra step called Circular Consensus Sequencing (CCS). In CCS, the DNA polymerase passes multiple times on the same base on a circularized strand to produce High Fidelity (HiFi) reads that can achieve an accuracy over 99% [51].

Nanopore sequencing uses a membrane with protein pores, through which an electrical current is flowing. DNA strands are pulled through the pores, with each passing nucleotide generating a distinct disruption signature in the current that can be inferred as a specific base [52]. The firm has specifically oriented its strategy toward a "do it yourself" approach, enabling sequencing in any lab and even directly in the field via a small portable device [53]. Researchers can control how they generate their sequencing data, contribute to protocol development, and develop their own basecalling [54] to increase the yield and im-

prove the quality and length of the reads. Although Nanopore reads still exhibit a high error rate, their length keeps increasing to attain hundreds of kilobases to 1 Mb [55]. Besides, the error rate has also been decreasing with the release of the new flow cells and the development of more accurate basecallers such as Bonito [56].

Long reads are now routinely included in genome assembly projects and have led to N50 lengths much larger than short-read only assemblies (Figure 1.1). A current limitation lies in the amount of DNA required to prepare long-read libraries. Still, long-read sequencing remains inaccessible for certain species: whereas Illumina sequencing can handle small DNA amounts, with a poor quality, long-read protocols require high-molecular weight DNA [57]. PacBio and Nanopore sequencing remain difficult when one animal is too small to provide a sufficient amount of DNA, especially when the organism requires extraction protocols that lead to overly fragmented DNA (for example, with coral skeletons). In addition, secondary metabolites associated to DNA molecules, or branched DNA structures, can also disturb the sequencing reaction.

1.2 Genome assembly

A variety of programs have been developed to assemble sequencing reads *de novo*, taking advantage of different sequencing technologies while considering their limitations. Genome assembly aims to correctly reconstruct the original chromosome sequences from short or long, and accurate or error-prone fragments. Assemblers are typically based on one of the following paradigms: greedy, Overlap-Layout-Consensus, de Bruijn graphs.

The assembly problem can be represented as a linear puzzle where the pieces are the reads. Reads match together when they have overlapping sequences. This puzzle could be intuitively solved by iteratively putting together the overlapping pieces that match best: this greedy approach is an efficient heuristic to find the shortest common superstring of the set of reads (i.e., the shortest sequence that includes all the reads as substrings) [114]. Greedy algorithms have been implemented for first-generation sequencing reads, for instance in TIGR [69], and were further applied in short-read assemblers like PERGA [82], SSAKE [91] and VCAKE [93]. However, they cannot resolve complex, repetitive genomes: for this reason, greedy assemblers are mostly used nowadays to assemble small organelle genomes such as chloroplasts and mitochondria [80].

Table 1.1: Sequencing approaches and associated assemblers.

Sequencing	Length	Accuracy	Methods	Assemblers
First generation	1 kb	High	Sanger	ARACHNE [58], Atlas [59], CAP3 [60], Celera [61], Euler [62], JAZZ [63], Minimus [64], MIRA [65], phrap [66], Phusion [67], SUTTA [68], TIGR [69]
Second generation	25-300 bp	High	454, IonTorrent, Solexa, SOLiD	ABYSS [70, 71], ALLPATHS [72], CABOG [73], Edena [74], Euler-SR [75], Gossamer [76], IDBA [77], JR-Assembler [78], Meraculous [79], MIRA [65], Newbler, NOVOPlasty [80], PCAP [81], PERGA [82], Platanus [83], QSRA [84], Ray [85], Readjoiner [86], SGA [87], SOAPdenovo [88], SOAPdenovo2 [89] SPAdes [90], SSAKE [91], SUTTA [68], Taipan [92], VCAKE [93], Velvet [94]
Third generation	10-100.000+ kb	Low	PacBio CLR, Nanopore	Canu [95], FALCON [96], Flye [97], HINGE [98], MECAT [99], MECAT2 [99], miniasm [100], NECAT [101], NextDenovo [102], Ra [103], Raven [104], Shasta [105], SMARTdenovo [106], wtdbg [107], wtdbg2 [108]
	20 kb	High	PacBio HiFi	Flye [97], HiCanu [109], hifiasm [110], IPA [111], mdBG [112], MIRA [65], Peregrine [113]

The Overlap-Layout-Consensus (OLC) paradigm was first described in 1979 by Rodger Staden [115] and is based on an overlap graph (Figure 1.2). The Overlap step consists in finding overlaps above a certain quality threshold between all the reads and building a directed graph, where the nodes are the reads and the edges represent the overlaps between them. The Layout step removes redundant edges that can be inferred from other edges. Finally, the Consensus step finds the shortest generalized Hamiltonian path through the graph, i.e. returns the shortest path (or set of disconnected paths) that visit each contig of the assembly at least once. The OLC paradigm has thrived with the program Celera [61], which was used to assemble a human genome from a Sanger shotgun dataset [116].

De Bruijn Graphs (DBGs) (Figure 1.3) are a well studied structure in graph theory, described by Nicolaas Govert de Bruijn in 1946 [117] and before him by Camille Flye Sainte-Marie [118]. DBG-based assemblers require highly accurate reads in which errors are only substitutions, with no indels. They start by indexing all the different sequences of a given k length (k -mers) found in the reads. In node-centric DBGs, the k -mers present in the reads are represented as nodes and are connected in the graph when they have an

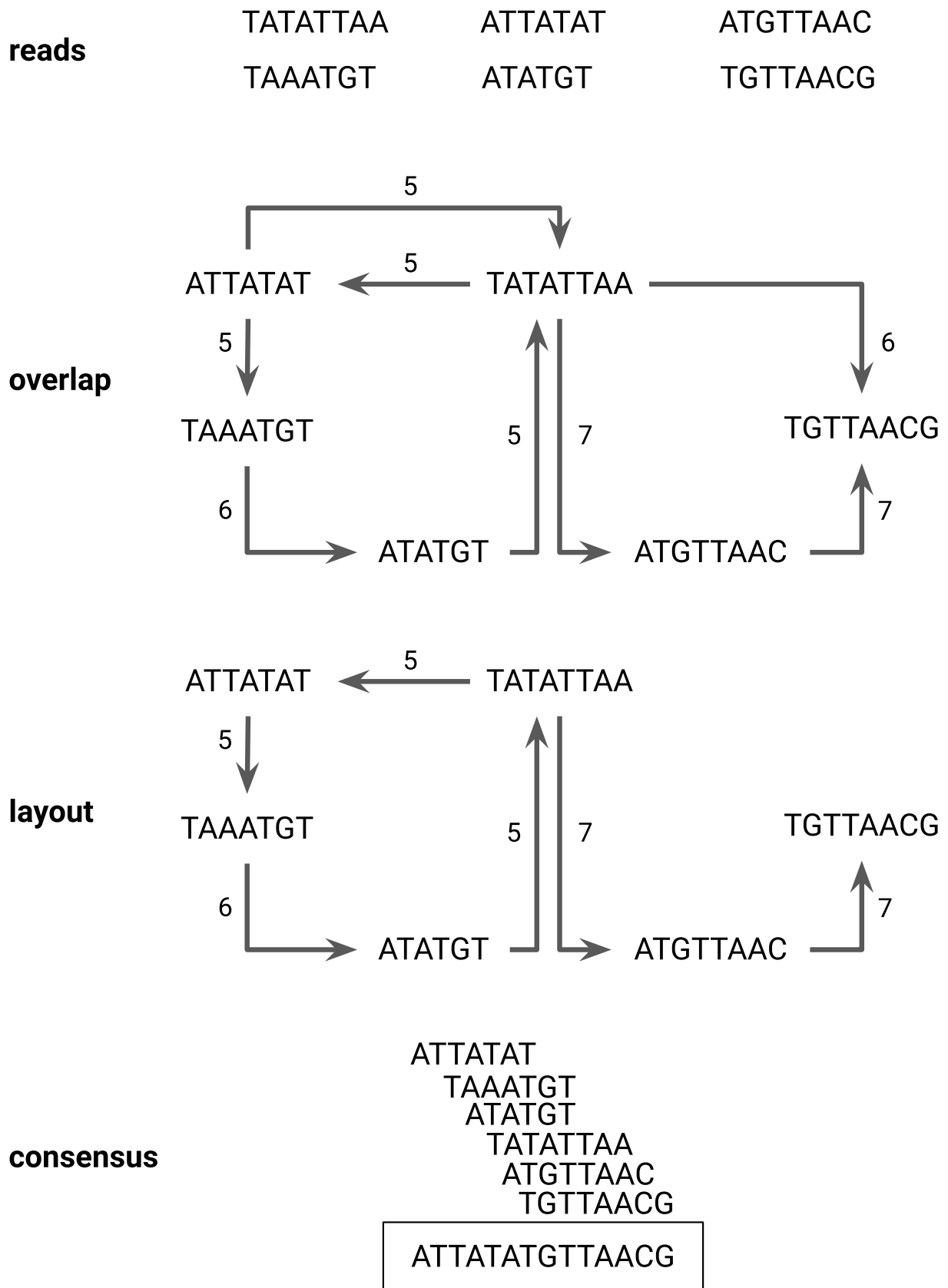


Figure 1.2: Overview of Overlap-Layout-Consensus assembly. The graph was built with all overlaps of at least 5 bases with a tolerance of 1 mismatch.

overlap of a $k-1$ length. In edge-centric DBGs, the k -mers present in the reads are represented as edges connecting their left and right $(k-1)$ -mers. Once the graph is constructed, DBG assemblers look for a generalized Eulerian (in the case of edge-centric DBGs) or Hamiltonian (in the case of node-centric DBGs) path through the graph, i.e. returns the shortest path (or set of disconnected paths) that visits each k -mer of the assembly at least once. This approach was first used for genome assembly of first-generation sequencing datasets [119] and was quickly implemented in multiple popular short-read assemblers, e.g. ABySS [70, 71], IDBA [77], SOAPdenovo [88] and SOAPdenovo2 [89], SPAdes [90], Velvet [94].

With the advent of third-generation sequencing, OLC assemblers have benefited from a renewed interest whereas DBG-based ones are poorly suited for long, low-accuracy reads, containing many erroneous k -mers. Numerous assemblers have implemented the OLC approach to produce *de novo* assemblies from error-prone long-read datasets: Flye [97], Ra [103], Raven [104], Shasta [105], wtdbg2 [108]. Now that HiFi reads bring a new type of high-accuracy long reads, assemblers have been adapted to better handle these sequences, such as Flye (with adapted parameters), HiCanu [109] and hifiasm [110], and we can expect the development of new DBG assemblers adapted for large k -mer values [120, 112].

From sequencing reads, assemblers build contiguous sequences called contigs. A perfectly assembled genome should have one contig representing each chromosome, but this is rarely achieved for eukaryotes. Assemblers need to find unambiguous paths in the assembly graph to reconstitute the chromosomes, but they often fail to do so due to the genomic structure: size, heterozygosity, repetitive content. Large genomes require a high amount of sequencing data in order to reach a sufficient depth to represent every loci. Genome sizes have a high variability (Figure 1.4): in the phylum Cnidaria, some myxozoans have a genome size of only some tens of Megabases (Mb) (*Kudoa iwatai*: 22.5 Mb, *Myxobolus squamalis*: 53.1 Mb, *Henneguya salminicola*: 60.0 Mb [121]), while the hydrozoan *Hydra oligactis* (1.3 Gigabases (Gb)) [122] has a genome size two orders of magnitude larger. Heterozygous regions constitute a major cause for breaks in assemblies of non-model animal genomes, as they generally have higher levels of heterozygosity than model species [123]. Most assemblers try to build a haploid representation of all genomes, even for multiploid (i.e. diploid or polyploid) genomes. To this end, heterozygous regions are collapsed in order to keep a single sequence for every region in the genome. In an assembly graph, these heterozygous regions will appear as bubbles, where one contig (a homozygous region) can be connected to several other contigs (the alternative haplotypes of a heterozygous region). When the assembler is unable to select one path, the homozygous region is not joined with any of the haplotypes, leading to a break in the assembly.

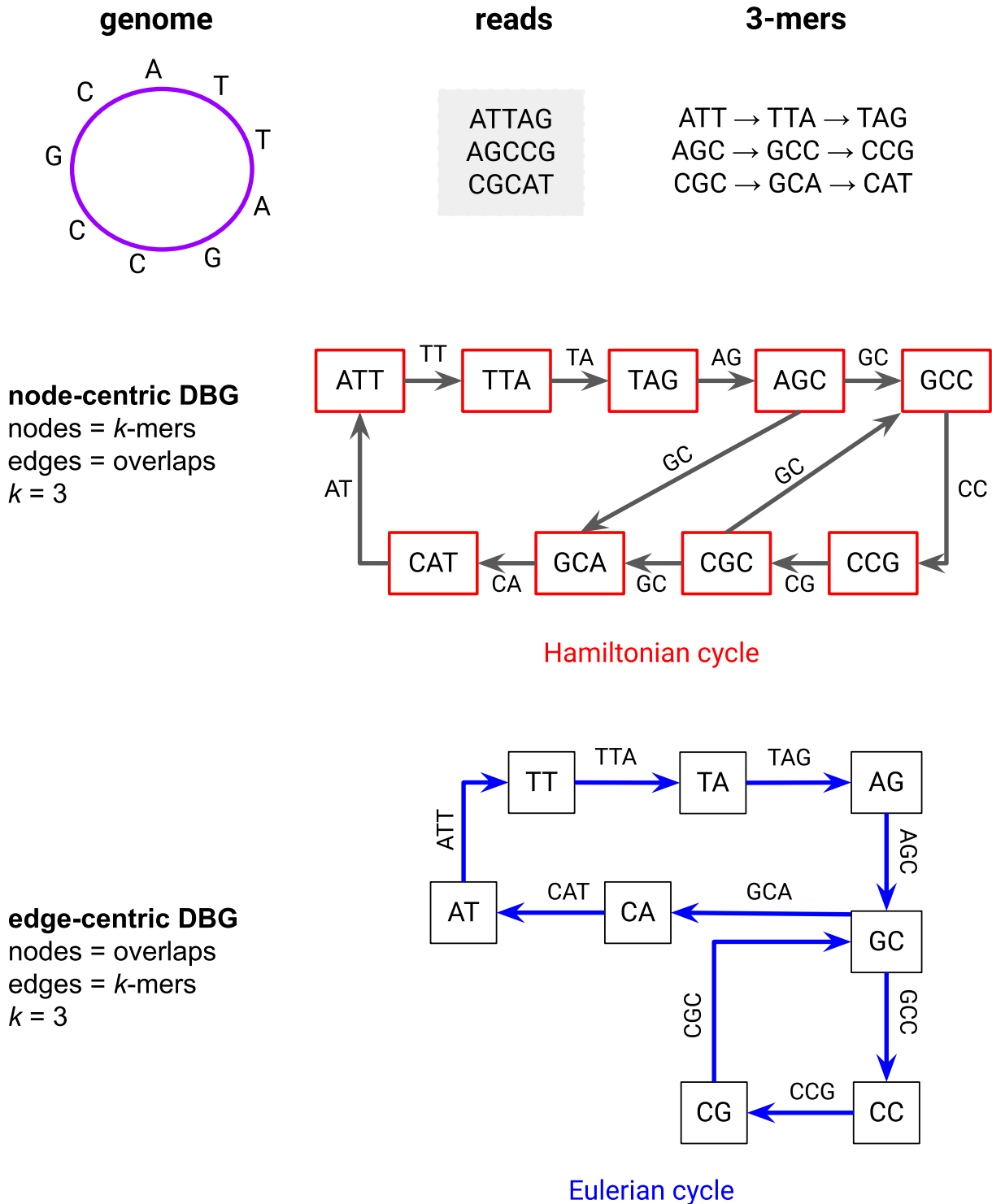


Figure 1.3: Overview of genome assembly using de Bruijn graphs. A circular genome is assembled based on three reads using node-centric and edge-centric DBGs with $k = 3$. The node-centric DBG is searched for a Hamiltonian cycle (visiting all nodes), and the edge-centric DBG for an Eulerian cycle (visiting all edges). These cycles are represented in blue in the graphs.

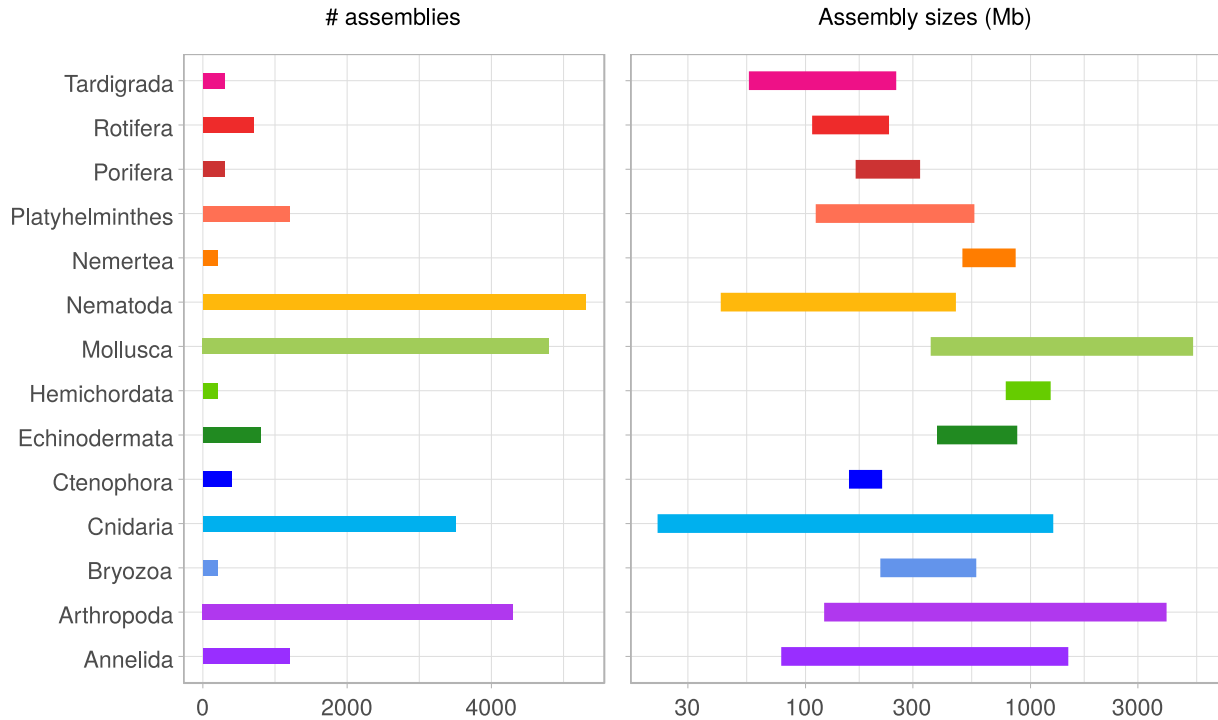


Figure 1.4: Assembly sizes. The left graph shows the number of assemblies included for each phylum and the right part shows the corresponding assembly-size ranges.

1.3 Assembly pre and post-processing

As obtaining high-quality chromosome-level contigs still remains challenging, upstream and downstream tools have been developed in conjunction with assemblers (Table 1.2). Researchers can test numerous combinations of these tools to devise the pipeline that will yield the best assembly.

Long reads have the advantage over short reads that they result in more contiguous assemblies. Nevertheless, assemblies of PacBio Continuous Long Reads (CLR) or Nanopore reads can have remaining errors due to their low accuracy; while errors in PacBio CLR are random and are compensated with a high coverage, Nanopore reads have systematic errors in homopolymeric regions. Assemblies of error-prone long reads often necessitate additional processes to increase the quality. There are two possible strategies: correct the long reads prior to assembly, and polish the contigs after assembly. Correcting long reads can be done using only the long reads or by adding high-accuracy short reads. Many tools have been developed for both scenarios and have been thoroughly reviewed on multiple datasets [198]. When tested on *Caenorhabditis elegans* Nanopore reads, the error rate decreased from 28.93% to less than 1% (using Canu [95], CONSENT [131], FLAS [133], Jabba [127], LORMA [129] or MECAT [99]). Some assemblers include a self-correction step in their pipeline, namely Canu [95], MECAT [99], NECAT

Table 1.2: Assembly pre and post-processing tools for haploid assemblies.

Step	Sequencing data	Tools
Reads filtering	Long reads	Filtlong [124]
Long reads error correction	Short reads	CoLoRMAP [125], Hercules [126], Jabba [127], LoRDEC [128], LoRMA [129], proovread [130]
	Long reads	Canu [95], CONSENT [131], Daccord [132], FLAS [133], NextDenovo [102], MECAT [99], MECAT2 [99], NECAT [101]
Polishing	Short reads	ntEdit [134], Pilon [135], POLCA [136]
	Short & long reads	Apollo [137], HyPo [138], Racon [139]
	Long reads	Arrow [140], CONSENT [131], Medaka [141], NextPolish [142], Nanopolish [143], Quiver [140]
Haplotigs purging	Long reads	HaploMerger2 [144], purge_dups [145], Purge Haplotigs [146]
Scaffolding	Short reads Mate pairs	Bambus [147], BATISCAF [148], BESST [149], BOSS [150], GRASS [151], MIP [152], Opera [153], ScaffMatch [154], ScaffoldScaffolder [155], SCARPA [156], SCOP [157], SLIQ [158], SOPRA [159], SSPACE [160], WiseScaffolder [161]
	Long reads	LINKS [162], LRScaf [163], npScarf [164], PBJelly [165], RAILS [166], SLR [167], SMIS [168], SMSC [169], SSPACE-LongRead [170]
	Genetic maps	ALLMAPS [171]
	Optical maps	AGORA [172], BiSCoT [173], OMGS [174], SewingMachine [175], SOMA [176]
	Linked reads	ARBitR [177], Architect [178], ARCS [179], ARKS [180], fragScaff [181], Scaff10X [182]
	3C/Hi-C	3D-DNA [12], dnaTri [183], GRAAL [184], HiCAssembler [185] instaGRAAL [186], Lachesis [187], SALSA [188], SALSA2 [189]
Gap filling	Short reads	GapFiller [190], GAPPadder [191], Sealer [192]
	Long reads	Cobbler [166], FGAP [193], GMcloser [194], LR_Gapcloser [195], PBJelly [165], PGcloser [196], TGS-GapCloser [197]

[101], NextDenovo [102]. Assembling corrected reads is expected to yield contigs with higher quality and contiguity. Alternatively, or additionally, the contigs can be polished to reduce errors, using long reads and/or short reads. Polishing can be a more computationally efficient strategy: the reads are mapped solely to the draft assembly, while correction is usually based on an all-versus-all read mapping.

Assemblers are generally tested on model-organism datasets, and are ill-suited for non-model genomes with variable levels of heterozygosity. They often fail to collapse highly divergent haplotypes, causing artefactually duplicated regions that hinder subsequent analyses [199]. Some long-read assemblers, Ra and wtdbg2, have been identified as less prone to retain uncollapsed haplotypes [200]. Contigs can also be post-processed to remove these duplications with dedicated tools such as HaploMerger2 [144], purge_dups [145] and Purge Haplotigs [146]. HaploMerger2 detects uncollapsed haplotypes based on sequence similarities, while purge_dups and Purge Haplotigs also rely on coverage depth.

To improve the contiguity of an assembly, contigs can be grouped, ordered and oriented into scaffolds. These scaffolds may contain gaps, when the sequence that should connect two contigs cannot be retrieved, represented as a sequence of Ns, and these gaps can be reduced post-scaffolding with gap-filling tools. Chromosome-level scaffolds have become a standard in genome assembly publications: unlike fragmented assemblies, they can be used for synteny analysis, finding rearrangements, and to separate chromosomes from different species. Several sequencing techniques have been used to scaffold assemblies: mate pairs, long reads, genetic maps, optical mapping, linked reads, and proximity ligation [201]. Mate pairs are short reads with a large insert size (more than several kb), and have been widely used in next-generation assemblies. Among the 237 assemblies we surveyed, 78 included a mate-pair scaffolding step (Figure 1.5). Both genetic maps [202] and optical maps [203] provide information on the linkage and relative position of a set of markers, spread over the genome, thus they can be used to anchor contigs. Genetic maps were used for the genome assemblies of the flatworm *Schistosoma mansoni* [204], the copepod *Tigriopus japonicus* [205] and the coral *Acropora millepora* [206]. Although existing genetic maps provide precious resources, building one is particularly difficult as it requires breeding [202], making it hardly accessible for wild species, and impossible for asexual species. Markers of optical maps are motifs in the sequence that are labeled and detected by a fluorescent signal. Companies such as Bionano or Nabsys propose this service to scaffold assemblies [207], and this method was included in some non-vertebrate genome projects: several nematodes including *Onchocerca volvulus* [208], *Ascaris suum* and *Parascaris univalens* [209], the tapeworms *Echinococcus multilocularis* [210] and *Hymenolepis microstoma* [211], and the chiton *Acanthopleura granulata* [212].

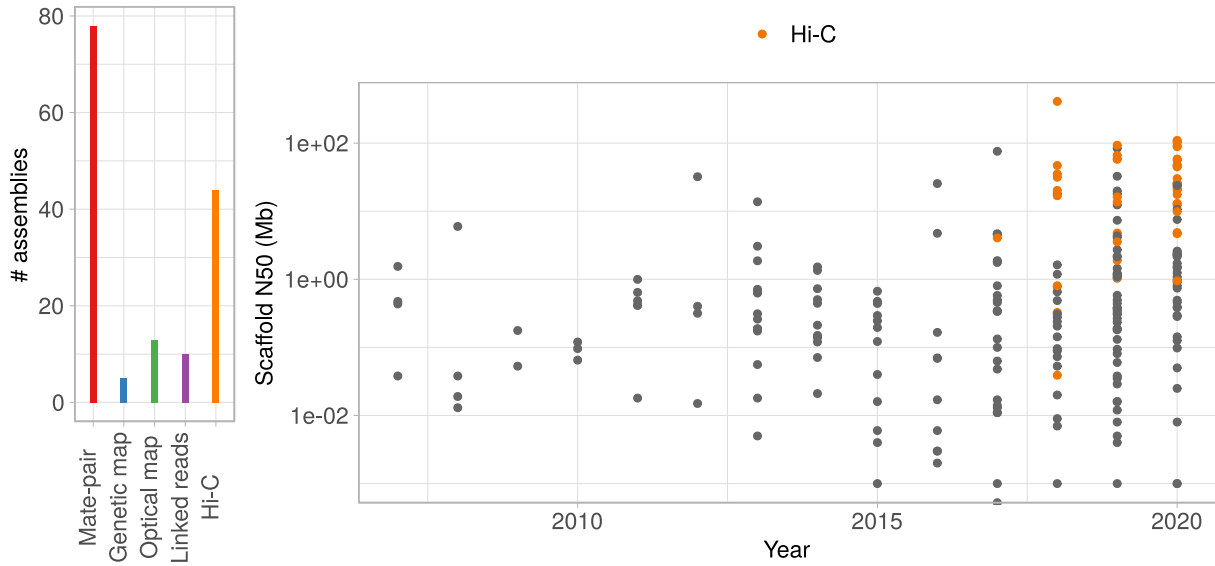


Figure 1.5: Assemblies scaffolding. Left: number of assemblies that included each scaffolding method. Right: scaffold N50 of non-vertebrate genome assemblies over time. The assemblies that included a Hi-C scaffolding step are highlighted in red; they form a cluster with a scaffold N50 over 1 Mb.

Linked reads and proximity ligation are based on short-read sequencing, preceded by a specific library preparation. For linked reads, also called cloud reads, long fragments of DNA are barcoded and then sequenced. The company 10X Genomics was a leader of this technology, but they chose to discontinue its commercialization in June 2020. Linked reads have been used to scaffold the genomes of the coral *Acropora millepora* [206] and the bee *Lasioglossum albipes* [213]. As linked reads are also shotgun Illumina reads, these reads are sometimes used for assembly (using Architect [178] or Supernova [214]) or polishing, as was done for the mosquito *Anopheles funestus* [215].

Proximity ligation techniques, based on capture of chromosome conformation [216], were not originally developed with genome sequencing applications in mind. Instead, they aimed at investigating the interplay between chromosome 3D organization and DNA processes [217]. A popular genomic derivative of 3C, Hi-C [218] documents the average conformation of the genomes of a population of cells. Briefly, the approach consists in freezing the chromosome folding of each individual cell using chemical fixation by formaldehyde, which generates bonds between proteins and proteins, and proteins and DNA. Then, the genome is cut into fragments using a restriction enzymes, that are then ligated in dilute conditions. As a consequence, fragments that were trapped together by the crosslinking step are more prone to be ligated with each other, rather than with a fragment belonging to a different crosslinked complex. This results in chimeric fragments with respect to the original genome agencement, reflective of their 3D contacts *in vivo*. The relative proportions of ligation events between all restriction fragments of a genome can then

be quantified, in theory, through high-throughput sequencing. On average, and because of the polymer nature and physical properties of DNA, the frequency of contacts between a pair of loci reflects either their 1D *cis* disposition along a chromosome, or their *trans* disposition on two independent chromosomes [219, 220]. Hi-C scaffolders have been developed following these principles: some follow a graph approach and use Hi-C links to join contigs (3D-DNA [12], SALSA2 [189]), whereas others exploit Markov Chain Monte Carlo (MCMC) sampling and Bayesian statistics to reorganize DNA segments into the scaffolds most likely to explain the observed interaction frequencies (GRAAL [184] and its later improved version instaGRAAL [186]).

The Hi-C protocol itself is becoming more and more accessible as commercial kits are now available (e.g. Arima Hi-C, Phase Genomics, or Dovetails Genomics). Besides, Dovetails Genomics uses both regular Hi-C and its own protocol for *in vitro* proximity ligation, dubbed CHICAGO. Hi-C scaffolding proved efficient at bringing highly fragmented draft assemblies to chromosome-level scaffolds (Figure 1.5), and is now included in many genome projects for all sorts of non-vertebrates: the arthropods *Varroa destructor* [221] and *Carcinoscorpius rotundicauda* [222], the cnidarians *Xenia* sp. [223] and *Rhopilema esculentum* [224], the echinoderms *Lytechinus variegatus* [225] and *Pisaster ochraceus* [226], the molluscs [227] and *Chrysomallon squamiferum* [228], the nematods *Caenorhabditis remanei* [229] and *Heterodera glycines* [230], the platyhelminthe *Schistosoma haematobium* [231], the poriferan *Ephydatia muelleri* [232], the rotifer *Adineta vaga* [233], the xenacoelomorph *Hofstenia miamia* [234], and more. A compelling advantage of Hi-C scaffolding over other scaffolding methods is its ability to discriminate different organisms in a draft assembly: DNA from different organisms belong to distinct nuclei, thus they have no 3D interactions. This feature is especially useful for non-vertebrates with symbionts, that can hardly be eliminated from the host prior to sequencing, and are often targets for genome assembly as well.

1.4 Assemblies evaluation

A critical step in genome assembly is to estimate the quality of draft assemblies, and choose the best one for subsequent analysis. The first metric to assess is the assembly size and its adequacy with an estimated genome size. The size can be estimated experimentally with flow cytometry or Feulgen densitometry [235], but these methods require a reference species for which the genome size is already well known, exposing them to errors induced by the reference genome size. Reference-free genome size estimation tools are typically *k*-mer based approaches and use high-accuracy reads (e.g. Illumina, PacBio

Table 1.3: Assembly evaluation of *Achatina fulica* and *Xenia* sp..

		<i>Achatina fulica</i>	<i>Xenia</i> sp.
Basic statistics	Assembly size	1.86 Gb	222.7 Mb
	N50	59.6 Mb	14.8 Mb
	N90	44.1 Mb	6.9 Mb
	Largest scaffold	116.6 Mb	22.5 Mb
	Number of scaffolds	1500	168
	Number of scaffolds larger than 1 Mb	32	17
	N count	3,600,500	194,000
BUSCO completeness	Complete and single-copy BUSCOs	84.4%	86.0%
	Complete and duplicated BUSCOs	3.6%	2.2%
	Fragmented BUSCOs	3.5%	3.5%
	Missing BUSCOs	8.5%	8.3%
Reads mapping	Short reads	96.2%	87.8%
	Long reads	81.62%	99.5%
	Hi-C	70.2%	65.7%

HiFi). These tools, such as BBtools [236], GenomeScope [237] and KAT [238], build a k -mer spectrum representing the number of k -mers with a certain frequency of occurrence. When the sequencing depth is sufficient, the k -mer spectrum should display one or more peaks depending on the ploidy. For a haploid organism, there should be only one peak, whereas a diploid organism should have two peaks. The plot may also show a peak of k -mers with a frequency of occurrence close to zero, corresponding to erroneous k -mers. Another recent tool called MGSE [239] estimates genome size based on reads mapping to a highly continuous assembly of the same genome; this method can be used as a post-hoc analysis.

N50 is a popular metric that reflects the contiguity of an assembly: it is defined as the length of the largest contig for which 50% of the assembly size is contained in contigs of equal or greater length. Some tools provide in addition the N75, N90, N99, computed in a similar fashion. The NG50 is a variant of N50 that refers to an estimated genome size instead of the assembly size. The target assembly can further be mapped against a reference assembly to detect misassemblies and break them: the N50 and NG50 of the resulting fragments are called NA50 and NGA50. All these metrics can be computed using QUAST [240]. For genome assemblies of non-model non-vertebrates, reference assemblies are seldom available, or they have a poor quality or contiguity that the new assembly aspires to improve. Therefore we will focus on reference-free evaluation methods. Table 1.3 and Figure 1.6 present an example of assembly evaluation for the recently published snail *Achatina fulica* [241] and coral *Xenia* sp. [223].

Another feature to optimize is the completeness of the genome, usually based on orthologs or k -mers. BUSCO [30] searches for orthologs in a user-provided lineage; the current Metazoa lineage (designated as Metazoa odb10) contains 954 features. Assemblies are evaluated based on the proportion of orthologs

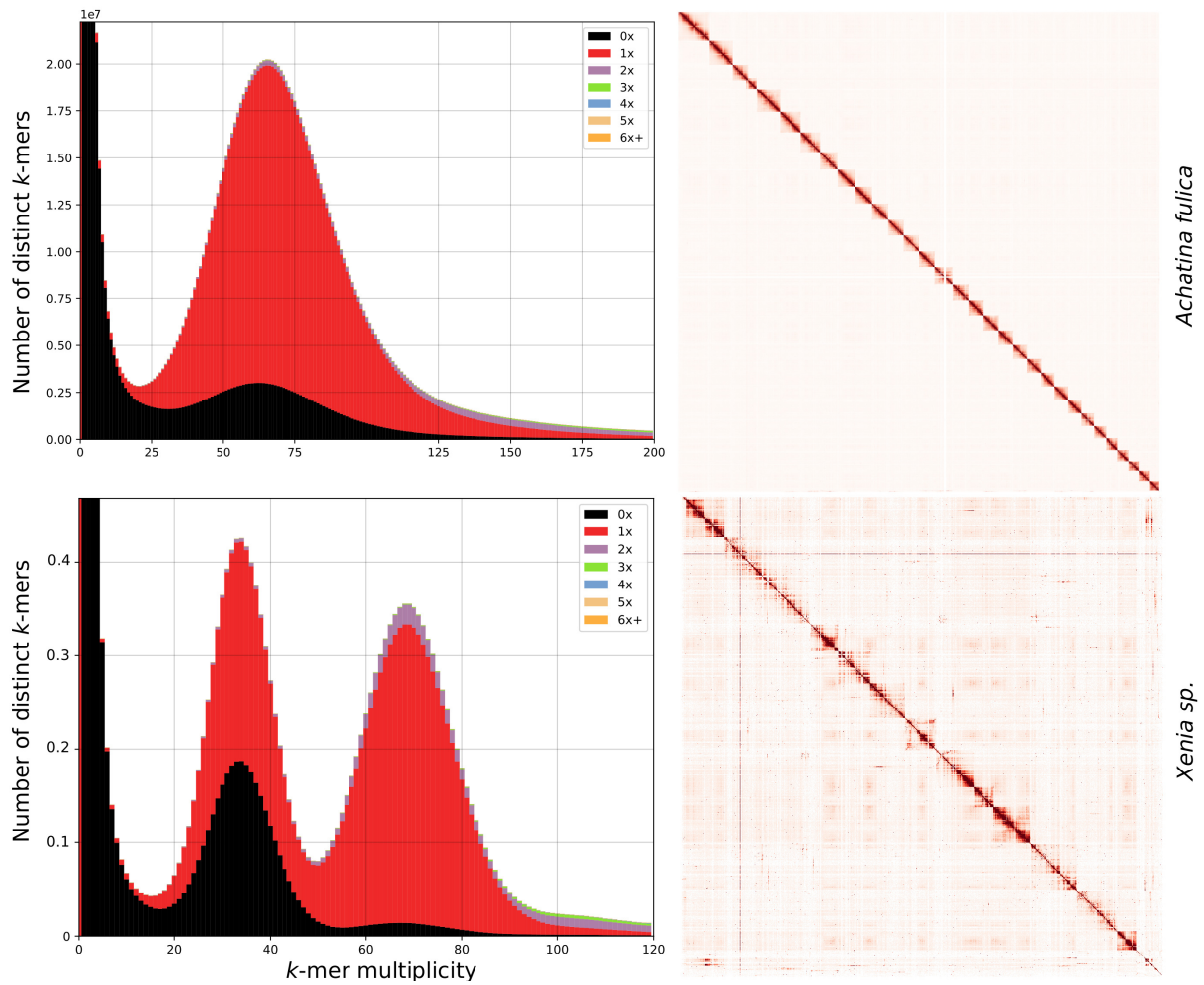


Figure 1.6: Assembly evaluation of *Achatina fulica* and *Xenia sp.*. Left: KAT comparison of the k -mers in the Illumina datasets v. the assembly. Right: Hi-C contact maps, with a binning of 300 for *Achatina fulica*, 30 for *Xenia sp.*.

to these 954 genes that can be retrieved into them; yet, some features are systematically missing in some genomes as they are absent from these species. More specific lineages are available for arthropods, insects, vertebrates, mammals, as many assemblies are available for these groups, but other metazoan phyla suffer from their lack of resources. Consequently, BUSCO is most powerful when comparing several draft assemblies for one genome. BUSCO scores provide information on complete single-copy and duplicated features, and the latter can be used to detect improperly duplicated regions in a haploid assembly. However, BUSCO scores are limited to genomic regions and cannot report for non-coding ones.

k -mer completeness scores do not present such limitations: KAT assesses the completeness of a whole assembly based on its representation of k -mers from a high accuracy sequencing dataset. The k -mer spectrum should display one or several peaks depending on the ploidy of the genome: one peak for a haploid genome; two peaks for a diploid genome, the first depicting heterozygous k -mers, and the second for homozygous k -mers. Depending on the ploidy of the genome, every k -mers should be represented in the assembly as many times as they actually are in the genome.

Both *Achatina fulica* and *Xenia* sp. have high BUSCO scores (against the lineage Metazoa odb10), yet slightly below 90%, and they have few duplicated BUSCO features. The k -mer spectrum of *Achatina fulica* only shows one peak around 70X (Figure 1.6, top left). These k -mers are expected to be represented exactly once, which is the case for the majority; there are almost no k -mers that appear twice in the assembly (in purple), but there is a noteworthy amount of missing k -mers (in black). For *Xenia* sp., the k -mer spectrum has two peaks with a k -mer multiplicity around 35X and 70X (Figure 1.6, bottom left). The first peak, representing heterozygous k -mers, shows that a portion is represented once in the assembly, while the rest is missing, as expected in a collapsed assembly. The second peak, for homozygous k -mers has a majority of k -mers represented once, and some k -mers either absent or duplicated. These assemblies seem overall properly collapsed and complete.

KAD, for k -mer abundance difference [242], proposes an alternative k -mer-based evaluation. This tool does not compute an overall completeness score, but instead classifies k -mers based on their abundance in the assembly and the sequencing dataset: good k -mers, erroneous k -mers (absent from the dataset), overrepresented k -mers (duplications), and underrepresented k -mers (collapsed repetitions).

Assemblies need to be screened for contaminants, to tell apart the sequences coming from the target and

from other species. Contaminants may originate from the environment, the symbiont, or be artificially introduced by the sequencing process. Blobtools [243] and BlobToolKit [244] aim to identify them with GC content, coverage depth and taxonomy assignment using the NCBI TaxID. Discriminating bacteria in metazoan assemblies is usually straightforward based on their distinct GC percentage. The task is more challenging when the target metazoan genome is mixed with other eukaryotes or even metazoans, especially when these species are absent from databases. Chromosome-level assemblies reduce the risk of contamination, as downstream analysis can be run exclusively on sequences that were anchored to the main scaffold. In addition, with Hi-C data, sequences from different species can be separated based on their absence of *trans* interactions. Contamination can lead to false conclusions: for instance, a study on a highly fragmented genome assembly ($N50 = 16$ kb) of the tardigrade *Hypsibius dujardini* assumed that about 17% of its genome derived from horizontal gene transfers [245], when these sequences were in fact contaminants [246].

When Hi-C data are available, contact maps, i.e. the representation of the paired-end reads from the Hi-C library aligned on the resulting scaffold, procure another evaluation asset to search for misassemblies. The contact map is expected to show heightened frequencies for each chromosome, in a chromosome-level assembly, and these interaction frequencies should decrease with increased distances separating loci on the sequence, based on the distance law. For *Achatina fulica*, 30 chromosome-level scaffolds (out of 31) display relatively consistent and regular contact patterns, representing well individualized entities in the contact map (Figure 1.6, top right). By contrast, the contact map of *Xenia* sp. does not display such patterns, with multiple *trans* contacts appearing between the scaffolds and most likely corresponding to scaffolding errors.

1.5 Phasing assemblies

As collapsing multiploid genomes can be difficult for highly divergent regions and frequently causes breaks in the assembly, an intuitive solution would be to phase genomes to retrieve all haplotypes. Phased assemblies represent a whole different challenge as they necessitate to correctly associate alleles, i.e. different versions of a heterozygous region [247]. A first approach, called trio-binning, is to assemble one individual using sequencing data from the individual itself and its parents [248]; yet this method is only adapted when the parents can be identified, and is inapplicable on asexual species. Some tools are able to reconstruct haplotypes from collapsed assemblies using long reads, namely HapCUT2 [249] and WhatsHap [250]. Ideally, genomes should be uncollapsed, as can be done with Bwise [251] and Platanus-Allée [252]

using short reads and FALCON-Unzip [96] using PacBio CLR or HiFi. FALCON-Unzip uses the output from the FALCON assembler, that includes both a haploid assembly and alternative haplotigs for heterozygous regions, to associate haplotypes based on long reads. Phased assemblies of low-accuracy long reads are limited, as small heterozygous regions were confused with errors; this led to haplotypes being erroneously collapsed.

HiFi reads have made a disruption in the fields of genomics: they are especially well-suited for phased assemblies thanks to their length and low error rate, and they have already been used to produce phased assemblies of a human [253] and the potato *Solanum tuberosum* [254]. Nevertheless, sequencing HiFi reads can remain inaccessible for non-model organisms as pure DNA is necessary.

Many organisms have already been assembled using low-accuracy long reads and high-accuracy short reads, thus an alternative is to correct long reads with short reads using a tool that conserves haplotypes such as Ratanosk [255]. Phased long-read assemblies can be further polished with adequate programs (e.g. Hapo-G [256]). As Hi-C has already demonstrated its efficiency to scaffold haploid assemblies, the principles were further exploited in ALLHiC [257] and FALCON-Phase [258] to phase assemblies while increasing their contiguity: as alleles from one haplotype belong to one chromosome, these alleles have higher Hi-C interaction frequencies together than with alleles from alternative haplotypes.

Phasing-specific evaluation methods are still scarce, and publications of phased assembly rely on various datasets to prove their correctness (e.g. parental assemblies [253]). Merqury [259] proposes a k -mer-based approach, inspired by KAT, and computes plots and scores to assess phasing completeness and find haplotype switches. However, similarly to trio-binning, it requires parental data.

1.6 Outline of the thesis

This first chapter introduced the principles of DNA sequencing, genome assembly, and the difficulties specific to non-vertebrate animals. This thesis is divided into two main sections: Chapters 2, 3 and 4 describe methodologies and tools for genome assembly, whereas Chapters 5, 6 and 7 present applications of these methods to past and ongoing genome projects to which I contributed.

Chapter 1 evaluates the performances of seven long-read assemblers, and more specifically their behavior on non-model genomes with variable levels of heterozygosity, based on the example of the rotifer *Adineta vaga*. The end goal of this study is to produce high-quality collapsed contigs from multiploid genomes. In Chapter 2, the strategy is the opposite, as the aim is to obtain uncollapsed, phased assemblies. This part presents the tool GraphUnzip, which takes advantage of assembly graphs, long reads and Hi-C reads to yield phased gap-less supercontigs with a high contiguity. Chapter 3 takes contigs to scaffolds with the Hi-C scaffolder instaGRAAL, a new version of GRAAL. Chapter 4, 5 and 6 describe the genome assemblies of *Adineta vaga*, *Astrangia poculata* and *Flaccisagitta enflata*, using the strategies identified in Chapter 2 for long-read assembly, and instaGRAAL for Hi-C scaffolding.

Chapter 2

Benchmark of long-read assemblers on the genome of the bdelloid rotifer

Adineta vaga

Long reads have made highly contiguous assemblies accessible for all genomes, but most long-read assemblers aim to produce a haploid assembly, regardless of the actual degree of ploidy (haploid, diploid or polyploid) of the genome being assembled. To obtain haploid assemblies from diploid or polyploid genomes, homologous chromosomes need to be collapsed into a single sequence. This process is straightforward for homozygous regions, but more challenging for heterozygous regions as the assembler needs to find a consensus between haplotypes or select one to represent the region. Collapsing haplotypes is especially challenging for non-model diploid or polyploid genomes, as they often display variable levels of heterozygosity across their genomes.

I designed a benchmark of seven long-read assemblers, namely Canu [95], Flye [97], NextDenovo [102], Ra [103], Raven [104], Shasta [105] and wtdbg2 [108]. I tested these assemblers on the genome of a non-model diploid organism, *Adineta vaga*, for which high-coverage sequencing datasets of both PacBio and Nanopore low-accuracy long reads were available. I investigated the improvement of haplotype-collapsing when combining these assemblers with pre-assembly read filtering and post-assembly haplotig purging. I defined a thorough evaluation strategy to identify the best haploid assemblies, based on assembly size, contiguity, completeness, and a new metric of haploidy that was implemented in the tool HapPy. I also

evaluated the impact of sequencing depth on haplotype collapsing and overall assembly quality, and found that most assemblers were optimized for a sequencing depth of 40X. A higher sequencing depth would not necessarily improve the assemblies but would rather lead to more uncollapsed haplotypes.

I initiated this benchmark to evaluate how long-read assemblers behaved on a small non-model eukaryote genome, and applied these strategies in several assembly projects for larger genomes, including *Astrangia poculata* and *Flaccisagitta enflata*.

RESEARCH ARTICLE

Open Access



Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms

Nadège Guiguelmoni^{1*}, Antoine Houtain², Alessandro Derzelle², Karine Van Doninck^{2,3} and Jean-François Flot^{1,4}

*Correspondence:
nadege.guiguelmoni@ulb.be
¹ Service Evolution
Biologique et Ecologie,
Université libre de Bruxelles
(ULB), Avenue Franklin D.
Roosevelt 50, 1050 Brussels,
Belgium
Full list of author information
is available at the end of the
article

Abstract

Background: Long-read sequencing is revolutionizing genome assembly: as PacBio and Nanopore technologies become more accessible in technicity and in cost, long-read assemblers flourish and are starting to deliver chromosome-level assemblies. However, these long reads are usually error-prone, making the generation of a haploid reference out of a diploid genome a difficult enterprise. Failure to properly collapse haplotypes results in fragmented and structurally incorrect assemblies and wreaks havoc on orthology inference pipelines, yet this serious issue is rarely acknowledged and dealt with in genomic projects, and an independent, comparative benchmark of the capacity of assemblers and post-processing tools to properly collapse or purge haplotypes is still lacking.

Results: We tested different assembly strategies on the genome of the rotifer *Adineta vaga*, a non-model organism for which high coverages of both PacBio and Nanopore reads were available. The assemblers we tested (Canu, Flye, NextDenovo, Ra, Raven, Shasta and wtdbg2) exhibited strikingly different behaviors when dealing with highly heterozygous regions, resulting in variable amounts of uncollapsed haplotypes. Filtering reads generally improved haploid assemblies, and we also benchmarked three post-processing tools aimed at detecting and purging uncollapsed haplotypes in long-read assemblies: HaploMerger2, purge_haplotigs and purge_dups.

Conclusions: We provide a thorough evaluation of popular assemblers on a non-model eukaryote genome with variable levels of heterozygosity. Our study highlights several strategies using pre and post-processing approaches to generate haploid assemblies with high continuity and completeness. This benchmark will help users to improve haploid assemblies of non-model organisms, and evaluate the quality of their own assemblies.

Keywords: Genome assembly, Long reads, Haplotype collapsing



© The Author(s), 2021. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

Background

With the advent of third-generation sequencing, high-quality assemblies are now commonly achieved for all types of organisms. The rise of two main long-read sequencing companies, Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (Nanopore), has prompted an increase in output as well as a decrease in cost, making their technologies more accessible to research teams and more applicable to any genome, including the more challenging ones with large genome sizes and high repetitive content. The primary advantage of long reads over short reads (such as those generated by Illumina sequencing platforms) is their typical length, which is two or three orders of magnitude greater [1]. As a result, long reads facilitate genome assembly into contigs and scaffolds as they can span repetitive regions [2] and resolve haplotypes [3].

However, long reads typically have a much higher error rate than Illumina data, and these errors are mainly insertions and deletions (vs. substitutions for Illumina reads). PacBio data have a random error pattern that can be compensated with high coverage, and recent developments have aimed to increase accuracy by generating circular consensus sequences [4], where one DNA fragment is read multiple times. Nanopore reads, on the other hand, have systematic errors in homopolymeric regions and therefore Nanopore contigs generally require further correction using Illumina or PacBio reads, in a process called “polishing” [5, 6]. Despite this disadvantage, Nanopore reads are currently much longer than PacBio reads, with runs attaining N50s over 100 kilobases (kb) and longest reads spanning over 1 Megabase (Mb) [7, 8].

This progress has prompted the development of programs dedicated to producing *de novo* assemblies from long reads, all of which follow the Overlap Layout Consensus (OLC) paradigm [9]. Briefly, OLC methods start by building an overlap graph (the “O” step), then simplify it and clean it by applying various heuristics (the “L” step), which typically include the removal of transitively inferable overlaps, and finally compute the consensus sequence of each contig (the “C” step). Some long-read assemblers follow strictly this paradigm, such as Flye [10], Ra [11], Raven [12] (a further development of Ra by the same authors), Shasta [13] and wtdbg2 [14]; whereas other assemblers such as Canu [15] and NextDenovo [16] add a preliminary correction step based on an all-versus-all alignment of the reads (see Table 1). At present, most assemblers aim to generate a haploid assembly, in which each region of the genome is represented exactly once. For diploid or polyploid genomes, haploid assemblies include only one version of each heterozygous region and are therefore reduced representations of the actual complexity of the genome. Haplotypes can be reconstructed from a reference collapsed assembly using

Table 1 Assemblers included in this study

Assembler	Reads correction	Version used	Access
Canu	✓	1.9	github.com/marbl/canu
Flye	×	2.5	github.com/fenderglass/Flye
NextDenovo	✓	2.2	github.com/Nextomics/NextDenovo
Ra	×	0.2.1	github.com/lbcb-sci/ra
Raven	×	0.0.7	github.com/lbcb-sci/raven
Shasta	×	0.3.0	github.com/chanzuckerberg/shasta
wtdbg2	×	2.5	github.com/ruanjue/wtdbg2

long reads with tools such as HapCUT2 [17] and WhatsHap [3]. PacBio reads can also be used to generate haplotype-aware *de novo* assemblies with Falcon-Unzip [18]. Nevertheless, haploid assemblies provide a precious resource for genome analysis as they make it possible to compare easily genome structures, gene sets across species, identify orthologs for phylogenomic analysis, and detect variants across individuals.

Long-read assemblers were recently benchmarked on real and simulated PacBio and Nanopore bacterial datasets [19], and all assemblers tested proved their efficiency at reconstructing small haploid genomes within 1 h and with a low RAM usage. For eukaryotic genomes, the wtdbg2 publication [14] included an evaluation of Canu, Flye, Ra and wtdbg2 on several model organisms (*Caenorhabditis elegans*, *Drosophila melanogaster*, *Arabidopsis thaliana* and *Homo sapiens*), all of which had low levels of heterozygosity (at most 1% for *A. thaliana*). The assemblies of *Caenorhabditis elegans* and *Homo sapiens* had small variations in assembly size, as these genomes have a low heterozygosity and are properly collapsed. For *Arabidopsis thaliana*, the Canu assembly was oversized (196.5 Mb) compared to the expected size of 144 Mb, suggesting that the higher heterozygosity of this species led to artefactual duplications. By contrast, all the other assemblies were shorter than the expected size, thus they were likely incomplete and it is unclear how well these assemblers can handle divergent haplotypes. This comparison suggested that wtdbg2 produces fewer artefactual duplications, but no attempt was made to pre-filter reads or post-process assemblies to improve the result. Hence, a comprehensive evaluation of strategies to generate a structurally correct haploid assembly of a non-model, heterozygous diploid organism is still lacking.

To fill this gap, we present here a quantitative and qualitative assessment of seven long-read assemblers on the relatively small eukaryotic genome of the bdelloid rotifer *Adineta vaga*, for which a draft assembly based on short reads was published some years ago [20]. As with most non-model organisms, *Adineta vaga*'s genome presents a mid-range heterozygosity of ca. 2% with a mix of highly heterozygous and low-heterozygosity regions, making such genome more challenging to assemble than those of model organisms, which often exhibit very low levels of polymorphism [21].

In addition to assessing the ability of these seven assemblers to collapse highly heterozygous regions, we investigated whether adding a pre-assembly read-filtering step (selecting reads based on their length and quality) or removing uncollapsed haplotypes post-assembly (using existing tools HaploMerger2 [22], purge_dups [23] and purge_haplotigs [24]) improved the assembly. HaploMerger2 detects uncollapsed haplotypes in assemblies based on sequence similarity alone and can process both low and high-heterozygosity genomes. Along with sequence similarity, purge_dups and purge_haplotigs take into account the coverage depth obtained by mapping short or long reads to the contigs. Coverage depth represents the number of reads covering a position in a contig (computed after mapping reads on the assembly). The contigs are then aligned to select duplicates accurately and remove them. While purge_dups sets its coverage thresholds automatically, purge_haplotigs requires user-provided values. As the focus of the present benchmark was on dealing with uncollapsed haplotypes and not on polishing assemblies (a step for which many tools are available and that would represent a benchmark topic in itself), we did not perform polishing of our contigs.

Assemblies were evaluated using several metrics quantifying their level of continuity and the correctness of their haploid representation of a diploid genome: namely their assembly size, N50, BUSCO completeness, k -mer completeness, and coverage distribution. The assembly size represents the sum of the lengths of all the contigs in the assemblies. The N50 is a popular metrics that reflects directly the continuity of the assembly but does not account for possible structural errors; it is defined as the length of the largest contig for which 50% of the assembly size is contained in contigs of equal or greater length. The BUSCO completeness [25] assesses the number of orthologs retrieved completely from the assembly in one or several copies: a high-quality, properly collapsed haploid assembly should exhibit a high number of complete single-copy BUSCO features and a low number of duplicated BUSCO features. The k -mer completeness is the percentage of solid (i.e., frequently observed and therefore probably correct) k -mers in the set of reads present in the assembly. In the case of a haploid assembly of a diploid genome, all homozygous k -mers (i.e., k -mers that are shared by the two haplotypes) should be represented in the assembly, whereas only half of the heterozygous k -mers (i.e., k -mers that are found in only one haplotype) should be represented. To detect both underpurging and overpurging, we focused in our benchmark on the k -mer completeness of heterozygous k -mers: as we expect only half of them to be present in a haploid assembly, a well-collapsed assembly should exhibit a k -mer completeness of about 50%, whereas a lower value indicates that too many k -mers were lost (overpurging) and a higher value indicates that too many k -mers were retained (underpurging).

We also investigated the coverage-depth distribution of each assembly. In an ideal haploid assembly, all positions should be equally covered, hence we would expect a single peak in the coverage distribution. Based on our analysis of the coverage distribution, we developed a new metric to evaluate the haploidy, or proper collapsing, of assemblies of diploid genomes. The haploidy score is based on the identification of two peaks in the per-base coverage depth distribution: a high-coverage peak that corresponds to bases in collapsed haplotypes (hereafter called “collapsed peak”), and a peak at about half-coverage of the latter that corresponds to bases in uncollapsed haplotypes (“uncollapsed peak”). The haploidy score represents the fraction of collapsed bases in the assembly, and is equal to $C/(C+U/2)$, i.e. the ratio of the area of the collapsed peak (C) divided by the sum of the area of the collapsed peak (C) and half of the area of the uncollapsed peak ($U/2$). This metric reaches its maximum of 1.0 when there is no uncollapsed peak, in a perfectly collapsed assembly, whereas it returns 0.0 when the assembly is not collapsed at all (as in the case of a phased diploid assembly). We implemented this metric in a new tool called HapPy [26] available at github.com/AntoineHo/HapPy.

Finally, as computational resources can be a limiting factor in genome assembly, we compared the CPU time and RAM usage for the different assemblers tested by running them on the same machine under the same conditions. Canu and NextDenovo were not included in this comparison, as they required significantly higher resources and had to be run on different machines.

Results

Preliminary observations

The genome size of our benchmark organism was estimated as 102 Mb based on k -mer frequencies (see Methods). The k -mer spectrum shows two peaks (Additional file 1: Figure S1): the first one, around 45X, for heterozygous k -mers (found in only one haplotype); and the second peak, around 90X, for homozygous k -mers (identical in both haplotypes). As *Adineta vaga* has a mid-range heterozygosity, the number of distinct heterozygous k -mers is higher than the number of homozygous k -mers, making a haploid assembly more difficult to obtain than with low-heterozygosity genomes.

We initially ran each assembler (Table 1) five times on our complete and filtered Nanopore and PacBio datasets, as we had observed that there were some discrepancies (assembly size, N50) in the outputs when running several times Flye, NextDenovo, Shasta and wtdbg2. We found that the assembly size, N50 and BUSCO scores of the resulting assemblies were very similar to each other (Additional file 1: Figures S2–S3). As a result, we chose randomly one replicate assembly from each assembler and used this replicate for the subsequent haplotype-purging step.

To represent assembly statistics in a comprehensive manner, we defined four scores (see Methods): size, that is 1 minus the distance of the assembly size to the expected genome size; N50; completeness, a combined metric that includes both the single-copy BUSCO score and the distance of the observed k -mer completeness to the expected value of 50%; haploidy, computed using HapPy.

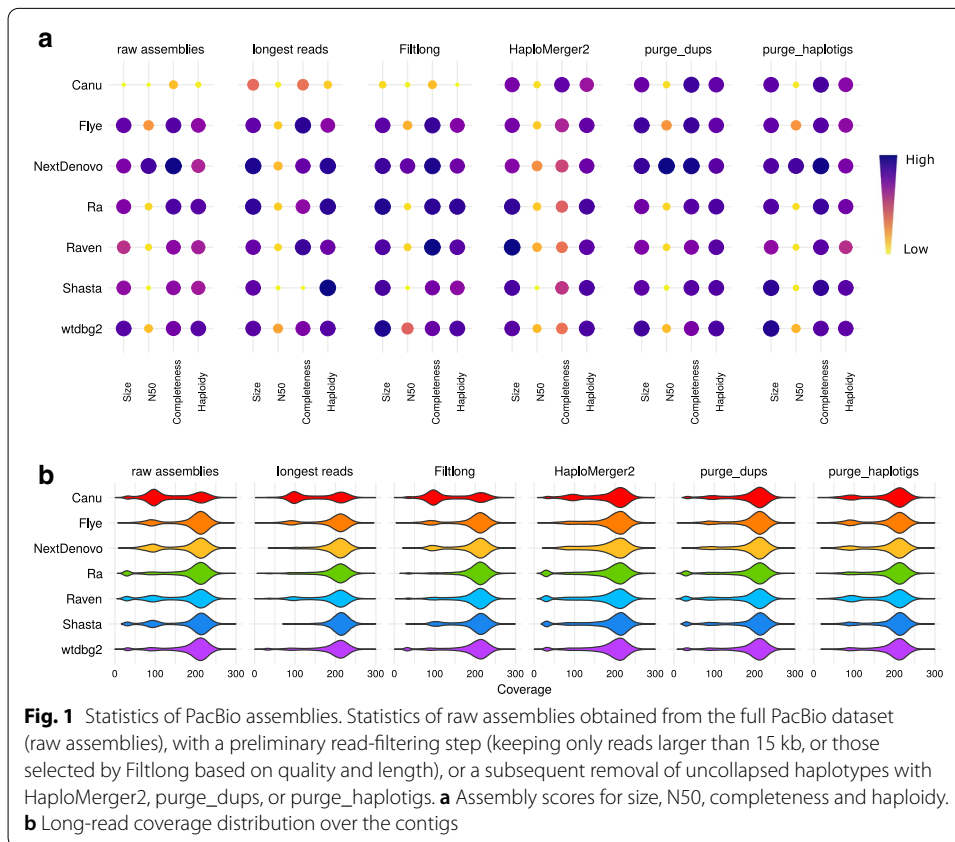
Assemblies of PacBio reads

Full assembly statistics are available in Additional file 1: Figure S4 and Additional file 1: Table S1–S3, whereas a summary of the results is presented in Fig. 1.

Raw assemblies

All raw assemblies of the full PacBio datasets were oversized compared to the estimated genome size of 102 Mb, ranging from 114 Mb (wtdbg2) to 169 Mb (Canu) (Fig. 1a, raw assemblies). The NextDenovo assembly obtained the highest N50 score with a value of 8.9 Mb, while other assemblies had an N50 ranging from 301 kb (Canu) to 2.7 Mb (Flye). The Canu assembly had the lowest completeness score, as its k -mer completeness greatly exceeded the expected value of 50% (73.5%) and its number of single-copy BUSCOs was only 538, compared to a highest value of 687 features (NextDenovo). The wtdbg2 and Ra assemblies scored the highest according to the haploidy metrics (0.90), while Canu obtained the lowest score (0.59).

Larger assembly sizes correlated with highly bimodal coverage distributions (Fig. 1b, raw assemblies). This was particularly the case with Canu assemblies, which exhibited two large peaks plus a smaller, low-coverage peak. The high-coverage peak, around 210X, was the collapsed peak C whereas the 100X peak, at about half-coverage of the C peak, corresponded to the uncollapsed peak U. In the case of the Canu assembly, the U peak was larger than the C peak, revealing a poor collapsing of highly heterozygous regions. The Flye, NextDenovo, Raven and Shasta assemblies also exhibited two peaks in their coverage distribution, although their U peak was smaller than the one of Canu. The



Ra, Raven, Shasta, and wtdbg2 assemblies exhibited an additional low-coverage peak identified as contaminants (see Additional file 1: Figures S5–S11).

Read filtering

We filtered PacBio reads using two strategies: either keeping reads longer than 15 kb; or filtering reads with Filtlong [27] based on quality (in priority) and length. These filtered datasets resulted in assemblies closer to the expected size than assemblies of all reads for Canu, NextDenovo, Ra, Raven and Shasta (Fig. 1a, read filtering). In the case of NextDenovo, filtering based on length made the N50 assembly drop to a value comparable with other assemblies (from 8.9 to 1.8 Mb), while the N50 did not decrease for the Filtlong dataset. Most assemblies maintained their completeness score with both strategies, and it even increased for some (Canu, Flye, Raven). As for the coverage distribution (Fig. 1b, read filtering), the Ra assembly no longer showed a low-coverage contaminant peak. For the NextDenovo assembly the U peak was absent when selecting the longest reads, but remained with the Filtlong dataset. The contaminant peak was also removed for the Raven assembly, and the U peak was reduced. The Shasta assembly had no U peak with the longest reads, but the assembly was shorter than expected (89 Mb) and had a poor completeness score.

Haplotig purging

When adding a post-assembly haplotig-purging step, we observed strikingly different results depending on the combination of assembler and post-assembly purging tool, namely HaploMerger2, purge_dups, purge_haplotigs. While HaploMerger2 reduced the size of all assemblies (resulting in higher size scores on Fig. 1a), it also led to a decrease of the completeness score of all assemblies, except for Canu (Fig. 1a, HaploMerger2). Nevertheless, the haploidy scores all increased (with a minimum of 0.84 for Canu and a maximum of 0.92 for Ra and wtdbg2) as U peaks decreased drastically in all coverage distributions (Fig. 1b, HaploMerger2). Assemblies purged with purge_dups were all closer to the expected size of 102 Mb, and the N50 and completeness scores were maintained or even improved (Fig. 1a, purge_dups). The haploidy scores were also improved as they ranged from 0.89 (Canu and Flye) to 0.91 (Ra and wtdbg2). The coverage distributions showed that the U peaks were removed or at least reduced, but the low-coverage peaks were not (Fig. 1b, purge_dups). After purging with purge_haplotigs, all assembly sizes were closer to the expected size except for Flye, and the N50 and completeness scores were maintained or even improved (Fig. 1a, purge_haplotigs). The haploidy scores were improved for Canu, NextDenovo and Shasta. The coverage distributions showed a reduction of the U peak for the Canu, NextDenovo and Shasta assemblies, explaining their higher haploidy scores (Fig. 1b, purge_haplotigs). The low-coverage peaks were removed for Ra, Raven, Shasta and wtdbg2, explaining their smaller assembly size.

Combination of read filtering and haplotig purging

The combination of read filtering and haplotig purging resulted in assemblies that almost all had a unimodal coverage distribution (Additional file 1: Figure S12–S13). As observed with assemblies of all reads, HaploMerger2 seemed to overpurge assemblies of the filtered datasets. Only the Canu assembly remained above the expected size, but its statistics were similar to those obtained with the Canu assembly of all reads. Assemblies purged with purge_dups or purge_haplotigs were closer to the estimated size and exhibited high numbers of single-copy BUSCO features. The combination of pre-assembly read filtering and post-assembly purge_dups seemed beneficial for most assemblers with the exception of Shasta, as the resulting assemblies ranged in haploidy from 0.90 (Canu and Flye) to 0.97 (NextDenovo). By contrast, the improvements observed when using a combination of read filtering and purge_haplotigs were similar to those obtained with either one of the two. NextDenovo, Ra and wtdbg2 assemblies had satisfying assembly sizes (from 99 to 107 Mb), high haploidy scores (from 0.85 to 0.96) (Additional file 1: Table S2) and unimodal coverage distributions, but similar scores were also obtained using only read selection for NextDenovo and Ra and using only purge_haplotigs for wtdbg2.

Combination of haplotig-purging tools

The combination of purge_dups or purge_haplotigs with HaploMerger2 resulted in problems similar to those observed on assemblies purged only with HaploMerger2: except for Canu, assemblies were shorter than expected and the number of single-copy BUSCO features dropped (Additional file 1: Figure S14). Assemblies purged with both purge_dups and purge_haplotigs were all reduced in size, but none went below the

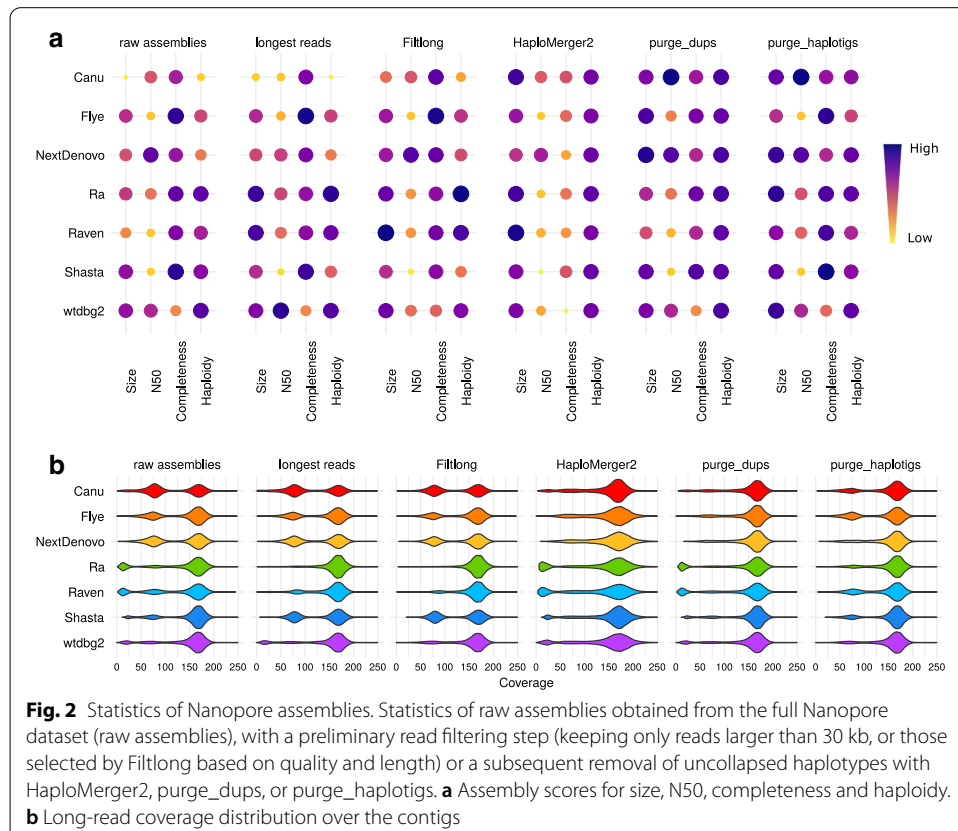
expected size. The BUSCO score and the k -mer completeness were stable, and the haploidy scores ranged from 0.88 (Canu and Raven) to 0.92 (NextDenovo) (Additional file 1: Table S3). The coverage distributions were all close to a unimodal distribution, as there were no low-coverage contigs and the U peaks had mostly disappeared.

Assemblies of Nanopore reads

Full assembly statistics are provided in Additional file 1: Figure S15 and Additional file 1: Table S4–S6.

Raw assemblies

Similarly to the PacBio assemblies, Nanopore assembly sizes exceeded the expected size of 102 Mb, ranging from 118 Mb (Shasta, wtdbg2) to 154 Mb (Canu) (Fig. 2a, raw assemblies). Nanopore assemblies achieved a much higher continuity than PacBio assemblies, as PacBio assemblies had a lowest N50 of 301 kb while Nanopore assemblies had a lowest N50 of 1.6 Mb. Canu, NextDenovo, Ra and wtdbg2 achieved a N50 over 5 Mb using Nanopore reads. The number of complete single-copy BUSCOs was lower in Nanopore assemblies (up to 559) than in PacBio assemblies (up to 699). The k -mer completeness was also usually lower for Nanopore assemblies, around 38.6–54.8%, compared to 47.7–73.5% for PacBio assemblies. These lower values for k -mer completeness in Nanopore assemblies were likely not due to a better collapsing but rather to systematic errors. The haploidy scores were higher for Nanopore assemblies produced by Canu, Ra, Raven,



Shasta and wtdbg2, in comparison with PacBio assemblies, but this score was lower for Flye and NextDenovo assemblies (Additional file 1: Table S1, Additional file 1: Table S4). The coverage distribution of the Canu assembly exhibited two distinct U (uncollapsed) and C (collapsed) peaks respectively around 75X and 160X, indicating that many haplotypes were not collapsed and were therefore represented twice in the assemblies (Fig. 2b, raw assemblies). This was also the case for the Flye, NextDenovo, Raven and Shasta assemblies, albeit their U peak was smaller than the one of the Canu assembly. The Ra, Raven, Shasta and wtdbg2 assemblies had an additional low-coverage peak identified as contaminants (see Additional file 1: Figures S16–S22).

Read filtering

When assembling a subset of either the longest Nanopore reads (over 30 kb) or reads filtered with Filtrong, the Ra and Raven assemblies had sizes closer to the estimated size and did not exhibit a contaminant peak in their coverage distribution, whereas other assemblies were generally unmodified (Fig. 2, read filtering). The Raven assembly of filtered reads had a smaller U peak compared to the raw assembly, but still present, whereas the U peak of the Ra assembly disappeared. The wtdbg2 assembly of the Filtrong dataset no longer shows a contaminant peak. These improvements came along with increased haploidy scores: from 0.90 to 0.95–0.97 for Ra, and from 0.83 to 0.89–0.92 for Raven. The result produced by Shasta with read filtering was the opposite, as the size and haploidy scores became lower and the U peak became larger with both filtering strategies. This increase in size of the U peak explained the higher k -mer completeness obtained due to a higher percentage of uncollapsed homozygous and heterozygous k -mers (Additional file 1: Figure S23–S24). Overall, read filtering did not affect completeness scores.

Haplotig purging

As we observed with PacBio reads, all assembly sizes were reduced by HaploMerger2, which resulted in higher size scores, and the U peaks were removed from the coverage distributions, but this also led to lower scores for completeness (Fig. 2, HaploMerger2). `purge_dups` improved size scores while maintaining or increasing completeness scores and removing U peaks (Fig. 2, `purge_dups`). These improved coverage distributions resulted in higher haploidy scores, ranging from 0.90 (Flye and Raven) to 0.93 (Ra and Raven). `purge_haplotigs` improved size scores for all assemblies except Flye and also kept high completeness scores (Fig. 2). Haploidy scores were higher for assemblies produced by Canu and NextDenovo, and the coverage distribution shows that U peak were indeed reduced or removed for Canu and NextDenovo assemblies, while the contaminant peaks were removed for Ra, Raven, Shasta and wtdbg2 assemblies. We observed again that HaploMerger2 generally decreased the continuity and quality metrics of the assemblies, while `purge_dups` and `purge_haplotigs` did not. Interestingly, the Canu assembly obtained the highest N50 of all the assemblies presented in this paper (12.4 Mb) when raw assemblies were purged with either `purge_dups` or `purge_haplotigs`.

Combination of read filtering and haplotig purging

The read-filtered Nanopore assemblies after HaploMerger2 were too short, except for the Canu assembly (Additional file 1: Figure S25–S26). All assemblies of the longest reads after `purge_dups` were close to the expected genome size (96–109 Mb), except for the `wtdbg2` assembly (114 Mb), and the coverage distributions were almost all unimodal, with a haploidy score ranging from 0.87 (Canu) to 0.96 (Ra) (Additional file 1: Table S5). Notably, the Shasta assembly of filtered reads had a strong U peak that was completely removed by `purge_dups`, which brought the assembly to a size close to our estimation (102 Mb) and resulted in a high number of single-copy BUSCO features (519 features). `purge_haplotigs` was apparently less efficient, as the U peak was reduced for the Canu and Raven assemblies but remained high for the Flye, NextDenovo and Shasta assemblies. The BUSCO scores of assemblies purged with `purge_dups` or `purge_haplotigs` were similarly high (up to 539 single-copy BUSCO features, Ra).

Combination of haplotig-purging tools

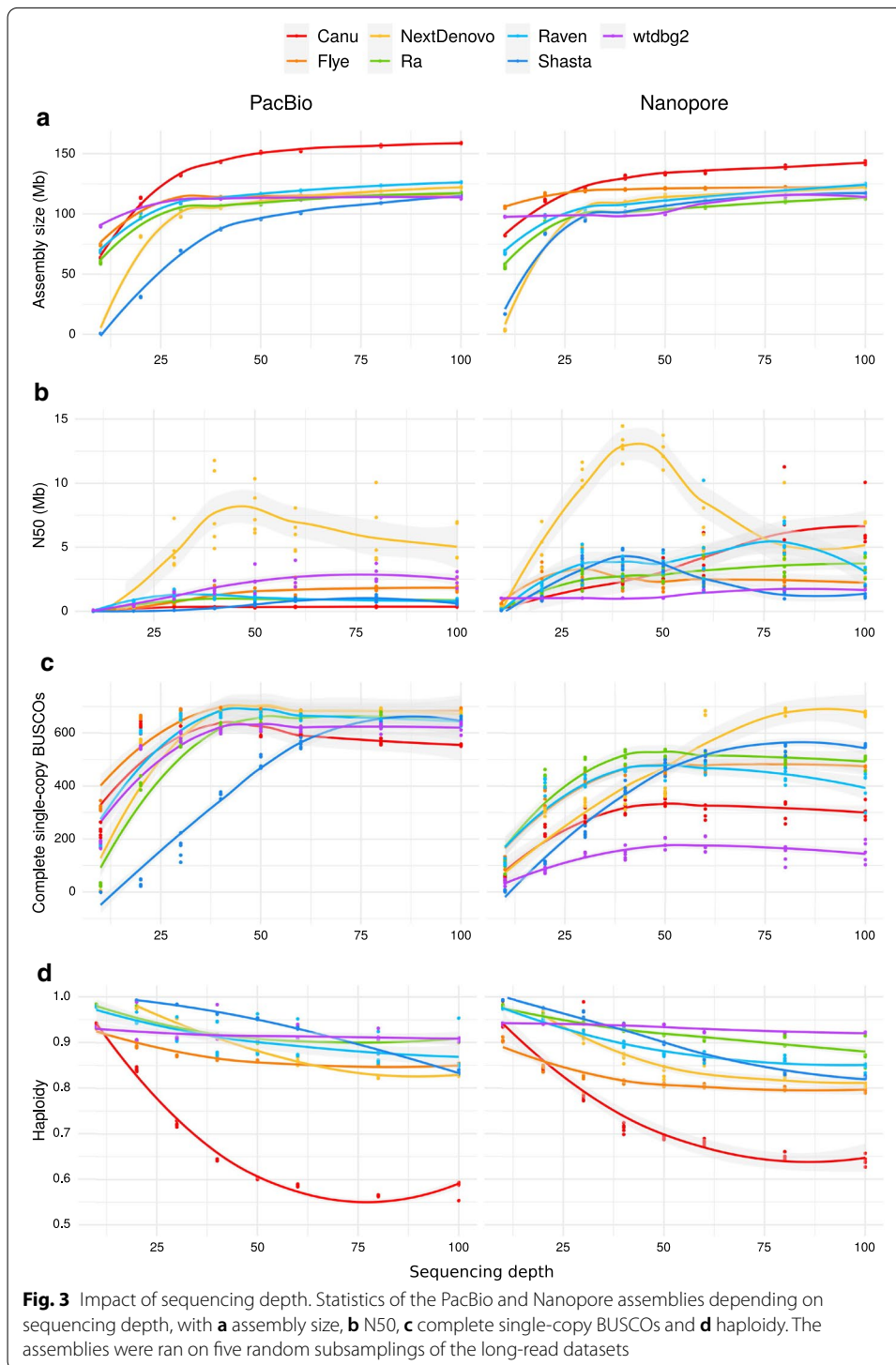
The combination of HaploMerger2 with either `purge_dups` or `purge_haplotigs` led to excessively shorter assemblies, except for Canu, as HaploMerger2 tended to overpurge (Additional file 1: Figure S27). The assemblies purged with both `purge_dups` and `purge_haplotigs` were all close to the expected size (100–110 Mb), their single-copy BUSCO score was maintained (up to 559 features, Flye assembly), they had a k -mer completeness below 50%, their haploidy ranged from 0.90 (Flye and Raven) to 0.94 (NextDenovo) (Additional file 1: Table S6) and their coverage distribution was unimodal or close to it.

Impact of sequencing depth

We further evaluated the impact of sequencing depths ranging from 10X to 100X on the size, N50, BUSCO score, and haploidy metrics of PacBio and Nanopore assemblies (Fig. 3).

With a 10X sequencing depth, almost all assemblers produced outputs excessively small compared to the expected genome size, and NextDenovo and Shasta performed the worst. However, with `wtdbg2` on PacBio and Nanopore reads, and with Flye on PacBio reads, the assembly size was close to the expected size. At 20X, Canu, Flye, Ra, Raven and `wtdbg2` reached at least 93 Mb. Assembly sizes increased sharply up to 40X, except for Canu for which the assemblies kept increasing in size with sequencing depth.

While there was no variation in assembly size among replicates, N50s were highly variable for PacBio assemblies with NextDenovo, Flye and `wtdbg2`, and for almost all Nanopore assemblies other than `wtdbg2`. For NextDenovo, there was a clear optimal coverage at about 40X in terms of N50 with PacBio as well as Nanopore reads. Other assemblers also showed a decrease in N50 above a certain sequencing depth: for PacBio assemblies, this happened for Raven over 30X and for Shasta and `wtdbg2` over 80X; for Nanopore assemblies, this happened for Flye over 30X, for Shasta over 40X, and for Raven over 80X.



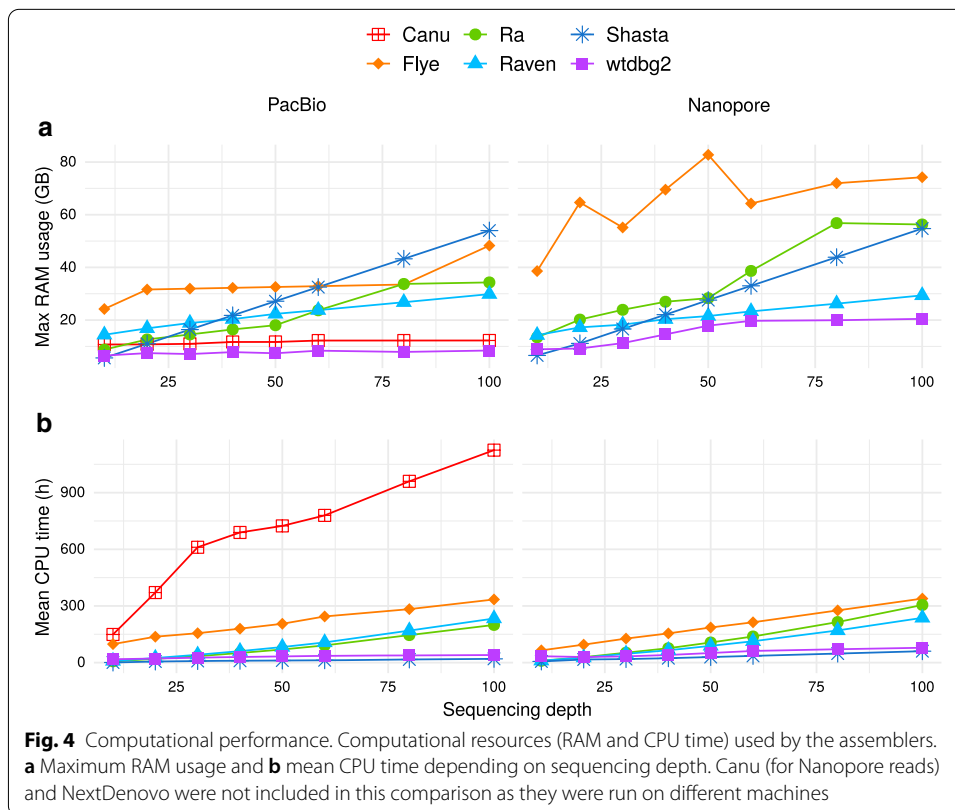
The number of single-copy BUSCO features did not vary much among replicate subsamplings; only NextDenovo exhibited some variability in that regard. Over 40X, the number of single-copy BUSCOs became strikingly stable for most assemblers with both PacBio and Nanopore reads, except for Shasta that is tuned for a 60X

depth. For Canu assemblies, the number of complete single-copy BUSCOs decreased when sequencing depth exceeded 30X. For wtdbg2 the complete single-copy BUSCOs remained low for the Nanopore dataset at the different sequencing depths.

For most assemblers, haploidy decreased when sequencing depth increased, and this was especially drastic for Canu assemblies. Only the assemblies produced by wtdbg2 had a stable haploidy value, whereas the Ra assemblies of PacBio reads also exhibited limited variations in haploidy.

Computational performance

We evaluated the computational performance of Flye, Ra, Raven, Shasta and wtdbg2. Canu was only evaluated on PacBio reads as the CPU time was too high for Nanopore reads, and we had to run Canu on a cluster. For the full Nanopore dataset, with 32 threads, Canu ran in 7 days 4 h 41 min and used 99 GB. NextDenovo was not included in this comparison as it required high RAM resources and was therefore run on a different machine; we were not able to measure properly its RAM usage or CPU time. RAM usage and CPU time (Fig. 4) increased with sequencing depth, except for wtdbg2 and Canu that did not display much change. wtdbg2 was the assembler that required the lowest amount of RAM (less than 20 GB) and was the second fastest on PacBio reads (less than 53 h) and Nanopore reads (less than 192 h). Flye had the highest RAM usage for Nanopore assemblies (Canu and NextDenovo excluded), with high variability. Notably, Shasta generally required a lot of RAM and had the highest memory footprint for high-coverage PacBio data, but it ran the fastest on PacBio



as well as on Nanopore reads. The most recent versions of Flye (2.8) and Raven (1.5) have decreased the CPU time. Raven was already an efficient assembler, and its latest improvements have halved the RAM usage and CPU time, when tested on the full PacBio and Nanopore datasets. Shasta 0.7.0 also decreased its RAM usage and CPU time.

Discussion

Comparison of PacBio and Nanopore assemblies

While PacBio assemblies were superior in terms of completeness, the continuity of Nanopore assemblies was far greater for most assemblers, probably due to the greater length of Nanopore reads. The lower completeness scores of Nanopore assemblies likely resulted from the lower accuracy along with the non-random error pattern of Nanopore reads producing errors (mostly indels) in the consensus sequences produced by the assemblers. These systematic errors in Nanopore reads may be improved with more recent protocols [28] and basecallers [29].

However, there was no striking difference regarding the efficiency of haplotype collapsing when assembling PacBio or Nanopore reads. The results in terms of coverage distribution and haploidy were similar, and it appears therefore that both technologies can be used to produce properly collapsed assemblies. An important finding was that filtering reads based on length and quality improved in many cases the quality of haploid assemblies, as it led to a decrease in coverage depth and to a lower support for both haplotypes, thereby favoring the collapse of the region. We further observed that read filtering did not lead to a decrease in assembly quality as long as the sequencing depth remained sufficient, and for some assemblers read filtering resulted in an increase in N50 and in completeness.

Successful combinations for haploid assemblies

We found that Canu poorly collapsed alleles and yielded oversized assemblies. The program did not seem able to collapse highly divergent regions on its own. Post-processing assemblies using haplotype-purging tools greatly improved haploidy. All three tools tested reduced the assembly size and the size of the U peak, but purging was most efficient using `purge_dups`. Purging Canu assemblies with `purge_dups` or `purge_haplotigs` improved greatly their haploidy. Besides, Canu assemblies of Nanopore reads purged with `purge_dups` or `purge_haplotigs` yielded the highest N50s among all our tests. However, the computational resources required to run Canu may be a limiting factor: although the RAM usage can be limited with the parameter `maxMemory`, this reduced the number of CPUs used and increased the running time.

Flye assemblies exhibited uncollapsed haplotypes too; selecting the longest reads did not help, but `purge_dups` improved collapsing, brought the assembly size close to the expectation and kept a good quality.

NextDenovo produced the assemblies with the highest N50s before post-processing, but with poorly collapsed haplotypes. This problem was alleviated for PacBio assemblies when selecting the longest reads, and uncollapsed haplotypes were efficiently removed by haplotig-purging tools. The best haploidy values were achieved when combining NextDenovo with read filtering, `purge_dups` and `purge_haplotigs`. These assemblies also

reached high values of continuity and completeness. However, although NextDenovo runs quickly, it requires a large amount of RAM.

Ra was more efficient at collapsing haplotypes than most assemblers, and its oversized assemblies were rather due to contaminants. Ra assemblies proved even better when using only the longest reads, which led to a better continuity, equal completeness and improved collapsing. Although Ra was not the most computationally efficient assembler, its RAM usage and CPU time remained low up to a sequencing depth of 50X; thus it appeared even more desirable to use only a subset of the longest reads for this assembler.

Raven is a further development of Ra, yet it exhibited a different behavior: Raven was more computationally efficient but produced less collapsed haplotypes compared to Ra. Read filtering and haplotig-purging reduced these uncollapsed haplotypes, resulting in high-quality assemblies when combining read filtering with `purge_dups`, or with `purge_dups` and `purge_haplotigs`. The low RAM usage and runtime of the newest version make it a compelling assembler.

We observed singular results with Shasta. The assembly of filtered Nanopore reads was less collapsed than the assembly of all reads. `purge_dups` efficiently purged the assemblies of all PacBio and Nanopore reads but, surprisingly, the best haploid Shasta assembly was obtained from filtered Nanopore reads purged with `purge_dups`. Shasta assemblies generally achieved a good completeness, but their continuity was lower than with other programs, as the developers explicitly aimed for quality over continuity.

`wtdbg2` performed well on PacBio data, but less on Nanopore reads, for which it obtained the lowest completeness scores. This program did not seem to have difficulties with heterozygous regions, but low-coverage contigs identified as contaminants remained in the final assemblies. Read selection on size did not significantly improve the assemblies, but `purge_haplotigs` removed contaminant contigs, therefore improving the output. Short-read polishing would certainly improve the low completeness of Nanopore assemblies. Users may want to test this assembler as it collapses genomes well and runs fast using a moderate amount of RAM.

Based on the above, we recommend users interested in generating the best haploid assembly of a diploid genome to try all or some of the solutions described in Table 2, depending on the size of the genome they want to assemble, the technology of reads they have, and their available computational resources.

Table 2 Recommended strategies for generating high-quality haploid assemblies

Assemblers	Recommended strategies	Advantages
Canu	<code>purge_dups</code> and/or <code>purge_haplotigs</code>	Highest N50 (Nanopore)
Flye	<code>purge_dups</code>	
NextDenovo	read filtering, <code>purge_dups</code> and/or <code>purge_haplotigs</code>	High continuity
Ra	read filtering	Low RAM usage and CPU time
Raven	read filtering + <code>purge_dups</code> or <code>purge_dups</code> + <code>purge_haplotigs</code>	Low RAM usage and CPU time
Shasta	<code>purge_dups</code>	Low CPU time
<code>wtdbg2</code>	<code>purge_haplotigs</code>	Lowest RAM usage and CPU time

Impact of sequencing depth

Our study of the impact of sequencing depth on the assemblies showed that deeper sequencing usually did not result in higher continuity or improved haploidy of the assemblies. Most programs reached the expected assembly size between 10X and 30X, while the BUSCO scores and k -mer completeness plateaued around 40X. Depending on the assembler, N50s also decreased when sequencing depth went beyond a specific threshold. A deeper coverage may lead to erroneous low-coverage contigs and provide more support to both haplotypes in highly heterozygous regions, promoting incomplete collapsing of haplotypes. We observed in our benchmark that a deeper sequencing led to a lower haploidy (as computed using HapPy). The combination of the continuity and quality metrics show that a sequencing depth of 40X is sufficient for generating a high-quality haploid assembly. Besides, a counter-intuitive finding is that a larger amount of reads does not improve the assembly and can even make it worse in terms of continuity and haploidy. Most assemblers seem optimized for sequencing depths around 30 to 40X and therefore did not appear to benefit from more data, except Shasta that is optimized for 60X.

Assembly evaluation

We propose here a set of metrics to evaluate thoroughly genome assemblies and identify uncollapsed haplotypes. The N50, BUSCO score and k -mer completeness are commonly used to estimate the continuity and completeness of assemblies, but do not discriminate properly collapsed haploid assemblies. It is possible to combine the completeness and continuity with a comparison of assembly size vs. expected genome size and an examination of the coverage distribution to identify the best assemblies. We further described a new metric of the haploidy of an assembly and implemented it in HapPy. We used the haploidy metric to systematically evaluate haploid assemblies. HapPy gives an accurate numerical representation of the coverage distribution.

These metrics have their limits, and not one of them is sufficient to identify the best assembly. The N50 is the most popular metric to describe contigs as it represents the continuity, yet high N50s can be achieved with efficient scaffolding methods. Therefore, we should aim for high-quality contigs that can be later turned into high-quality scaffolds. Comparing the assembly size to an estimated genome size depends on the reliability of the estimation itself. Genome size can be estimated computationally with a k -mer spectrum [30, 31], or experimentally using flow cytometry and Feulgen densitometry [32]. The BUSCO score only represents orthologs and does not account for the completeness of non-coding regions. Besides, the k -mer completeness is not sufficient as a value, as it could reach 50% with an assembly that has a balanced combination of missing regions and uncollapsed ones. To better estimate the k -mer completeness, it is necessary to examine plots provided by KAT. Collapsed repeated regions appear in the coverage plot along the contigs as localized peaks of elevated coverage. Since most assemblers included in our benchmark can produce an assembly graph, we strongly advise readers to investigate this file using dedicated tools [33]: uncollapsed haplotypes are usually observed as bubbles in the graph. Ideally, the contigs should be evaluated with all metrics available to find, if not the perfect assembly, the best assembly, while assessing its limitations.

Genome papers present only the most successful assembly strategy, but researchers usually try more than one method to obtain the best result. They rarely report details of all the tested approaches in publications, although such negative results could help the community and could guide developers in how to improve their tools. To improve on this situation, we suggest reporting alternative assembly results as additional file or separately on a preprint platform.

Comparison with other studies

Few comparisons of long-read assemblers are currently available. A thorough study was conducted on simulated and real bacterial datasets [19], and showed that all these assemblers can achieve assemblies of varying quality. Flye and Raven were also tested in this study and emerged among the most reliable assemblers. We also found that these assemblers reached high single-copy BUSCO scores, but when processing data from a diploid organism, they are not the most efficient at collapsing haplotypes. Besides, the benchmark on bacterial datasets also showed that Canu required more computational time and memory than most assemblers. Although this benchmark gave essential information on the performance of these assemblers, eukaryotic genomes represent a completely different challenge. Recently, a publication compared different long-read sequencing technologies to assemble a plant genome, *Macadamia jansanii* [34]. They included statistics for different assemblers and obtained, depending on the tool, oversized assemblies combined with heightened numbers of duplicated BUSCO features on an 80X PacBio dataset, while they did not observe such differences on a 30X Nanopore dataset. These results agree with our observations on the impact of sequencing depth, as the 30X Nanopore dataset was not problematic, while the 80X PacBio dataset was, likely because of a deeper sequencing.

Toward high-quality diploid and polyploid assemblies

Haploid assemblies of multiploid (i.e. diploid or polyploid) organisms provide a partial representation of their genomes as only one version of all heterozygous regions is included in the assembly. Ideally, we would prefer to generate phased multiploid assemblies. Low-accuracy long reads can separate haplotypes for highly heterozygous regions, but their high error rates do not allow the identification and separation of small heterozygous regions. Furthermore, phased assemblies bring an extra challenge, as alleles from different heterozygous regions need to be correctly associated. A protocol for high-accuracy long reads (above 99%) has been released recently, called PacBio HiFi [4], and brings new possibilities for phased multiploid assemblies. To better accommodate these high-accuracy long reads, new versions of assemblers have been released such as HiCanu [35] (a development of Canu), hifiasm [36], or Flye's new option `--pacbio-hifi`. Fully phased assemblies will provide complete representations of multiploid genomes.

Conclusion

We tested seven long-read assemblers on PacBio and Nanopore for a non-model eukaryote genome. As this genome has variable levels of heterozygosity, including highly heterozygous regions, we found that most assemblers had difficulties collapsing divergent haplotypes, resulting in oversized assemblies. Ra and wtdbg2 emerged as the most efficient programs to obtain haploid assemblies. We identified several assembly strategies combining assemblers, pre-assembly read filtering and post-assembly haplotig purging tools (either `purge_dups` or `purge_haplotigs`) that led to properly collapsed haploid assemblies, and also improved continuity and completeness. To guide users when filtering datasets, we tested assemblers with different sequencing depths, and found that aiming for a sequencing depth of 40X could optimize haplotype collapsing and continuity. In addition, we also conducted a thorough evaluation of these assemblies using N50, BUSCO and k -mer completeness, coverage distribution, and a new metric of haploidy that we implemented in HapPy. We recommend to reproduce these evaluations for any haploid assembly of a multiploid genome to ensure proper collapsing and avoid artefactual duplications.

We believe that benchmarks such as ours are essential to help researchers working on non-model organisms select a long-read sequencing technology and an assembly method suitable for their project. It will also help them better understand the results they obtain thereby improving the rapidly evolving field of genomics.

Methods

All command lines are provided in Additional file 1: Table S7.

Genome size estimation

The genome size of *Adineta vaga* was estimated using KAT v2.4.2 [30] on an Illumina dataset of 25 millions paired-end 250 basepairs (bp) reads (see Additional file 1: Table S2). The diploid size was estimated to 204.6 Mb, therefore a haploid assembly should have a length around 102.3 Mb.

Long-read assemblies

Canu, Flye, NextDenovo, Ra, Raven, Shasta and wtdbg2 were tested on two *Adineta vaga* long-read datasets: PacBio reads totalling 23.5 Gb with a N50 of 11.6 kb; and Nanopore reads totalling 17.5 Gb with a N50 of 18.8 kb (after trimming using Porechop v0.2.4, github.com/rrwick/Porechop). All assemblers were used with default parameters, except for Shasta for which the minimum read length was set to zero (instead of the default 10 kb setting). To run Shasta on PacBio reads, we used the recommended parameters `--Assembly.consensusCaller Modal --Kmers.k 12`. When assemblers required an estimated size, the value 100 Mb was provided. PacBio assemblies were run on all reads and on reads > 15 kb (4.7 Gb). Nanopore assemblies were run on all reads and on reads > 30 kb (5.7 Gb). For both datasets, we tested several length thresholds to find the optimal one. For more details on the long-read datasets we used, see the publication by Simion et al. [37] and Additional file 1: Table S8. To test for reproducibility, all assemblers were run five times.

Read filtering

Reads were filtered following two strategies: keeping only the reads larger than 15 kb (PacBio reads) or 30 kilobases (Nanopore reads); using `Filtlong v0.2.0`. `Filtlong` was run with the parameters `--target_bases 4092000000 -mean_q_weight 10` to keep about 40X of data and give a priority to quality over length.

Purging duplicated regions

Reads were mapped on assemblies using `minimap2 v2.17-r941` [38]. For each assembly, we ran `purge_haplotigs hist` [24] to compute coverage histograms that we used to set low, mid and high cutoffs; these values were then used by `purge_haplotigs cov` to detect suspect contigs. Finally, we ran `purge_haplotigs purge` to eliminate duplicated regions.

`purge_dups` [23] was run following instructions by first generating the configuration file and then purging the assembly. `HaploMerger2` [22] was run by sequentially running the modules `BuildDatabase`, `RepeatModeler`, `RepeatMasker` and finally the main script of `HaploMerger2` to purge the assembly.

Impact of sequencing depth

To find out the impact of sequencing depth, five replicate subsets were randomly sampled from the long-read datasets using the script `reformat.sh` from `BBTools v38.79`, available at sourceforge.net/projects/bbmap/, by providing the desired number of bases. The assemblers were run on these subsets with the same parameters as previously, with the exception of `Canu`: the parameter `stopOnLowCoverage` was set to 1 to allow runs on low-depth datasets. We tested subsets with a sequencing depth of 10X, 20X, 30X, 40X, 50X, 60X, 80X, 100X.

Assembly evaluation

To evaluate the assemblies, we ran `BUSCO v4` [25] against `metazoa odb10` (954 features) without the parameter `--long`. We ran `KAT comp v 2.4.2` [30] to calculate k -mer completeness by reference to the same `Illumina 2*250 bp` dataset used to estimate the genome size. To compute coverage, long reads were mapped on one replicate assembly per assembler using `minimap2` and the coverage was computed with `tinycov`, available at github.com/cmdoret/tinycov, with a window size of 20 kb.

Haploidy evaluation

To evaluate the collapsing of assemblies based on the coverage distribution, we developed a script, available at github.com/AntoineHo/HapPy. `HapPy` estimates the haploidy of an assembly (a measure of how well it is collapsed) by analyzing the per-base coverage histogram obtained after mapping reads to the assembly. For a well-collapsed haploid assembly, this histogram should consist of one peak around the theoretical average depth of coverage.

`HapPy` takes a raw coverage frequency histogram as input. First, to avoid problems due to local variations and noise in the dataset, two filters are applied before attempting to find peaks in the dataset. The first filter avoids misdetections due to potential

peaks at very large coverage. These can be due to repetitive DNA for instance. These peaks are eliminated from the curve by using a filter on the cumulative sum of the frequency of each coverage value. For each increasing coverage value, the total frequency sum is incremented by the frequency of this coverage value. When this sum exceeds 99% of the sum of frequencies of all coverage values, then the remaining coverage values are discarded. Effectively, this discards the very large coverage bins containing low information peaks. The second filter is applied on the resulting histogram curve. It smoothes the curve using a Savitzky-Golay filter [39] from the `SciPy` package. This filter uses convolution to smooth the curve by fitting a low-degree polynomial function (using linear least squares) to consecutive subsets of adjacent data points. The window length used is 41 and the polynomial degree used is 3.

Then `SciPy` is used to detect local maxima in the curve. A local maximum is defined as a sample in the input array for which any neighbouring sample has a smaller amplitude. Local peaks detected are filtered by `SciPy` using different parameters such as the minimum height to be considered a peak and the prominence of a peak compared to its neighboring local maxima. Finally, to determine peak widths, `SciPy` uses an algorithm described in its documentation: https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.peak_widths.html#scipy.signal.peak_widths.

Actual coverage curves rarely match the theoretical expectation of a single peak for a haploid assembly because of several factors:

- contaminant contigs (e.g. bacteria and viruses that were sequenced along the organism of interest) can appear as additional peaks at unexpected coverage values (usually lower than the actual sequencing depth);
- some contigs or contig regions in the assembly may actually correspond to uncollapsed haplotigs;
- large and/or abundant hemizygous deletions, as well as haploid chromosomes (e.g. the Y chromosome of male mammals), can result in a half-coverage peak; in such case, this peak is a biological signal that will prevent reaching a perfect haploidy score.

`HapPy` expects up to three peaks in the per-base coverage histogram: a low-coverage contaminant peak (if any), an uncollapsed peak at around half of the expected coverage and a collapsed peak around the expected coverage.

After peak detection, `HapPy` determines whether peaks are contaminant, collapsed or uncollapsed based on the thresholds given by the user in input parameters and their relative positions in the curve. Then, it finds the actual limits between contaminants, collapsed and uncollapsed regions of the curve using the computed peak widths. The peak area corresponds to the number of bases in the specific range of coverage values that was attributed to each peak.

We defined a haploidy score using the following Equation 1, in which U is the area of the uncollapsed peak and C the area of the collapsed peak. It describes the proportion between bases that are collapsed and the total number of bases that we expect in a perfectly haploid assembly. A perfectly haploid assembly is theoretically an assembly of a diploid organism for which all bases from one haplotype have an homologous representation on the other haplotype. For such an assembly, the haploidy score would be equal to 1.0. However, this

equation does not take into account insertions and deletions (indels). If indels are numerous or large, then the best scores that the assembly could reach would be lower than 1.0 because all bases do not have an homologous equivalent on the alternative haplotype. For instance, an insertion will not have, by definition, a homologous counterpart; thus this homologous region is not present to be sequenced, which will result in a halved coverage over the insertion region.

$$Haploidy = \frac{C}{C + \frac{U}{2}} \quad (1)$$

Note that Eq. 1 does not take into account bases that were attributed to contaminant sequences based on the coverage curve. Ideally, the contigs should be pre-filtered for obvious contaminant taxa before using HapPy.

Scoring

We defined four scores to evaluate the assemblies in Figs. 1 and 2: size, N50, completeness and haploidy. The N50 score corresponds to the regular N50 value, and the haploidy score is computed using HapPy. The size score reflects the distance of the assembly size to the estimated haploid genome size and is computed following Eq. 2, in which s is the assembly size and G the estimated haploid genome size (102 Mb).

$$S = 1 - \frac{Abs(s - G)}{G} \quad (2)$$

The completeness score includes both the number of single-copy BUSCO features and a measure of the distance of the observed k -mer completeness compared to the expected one. This metrics is computed using Eq. 3, in which k_{obs} is the observed k -mer completeness, k_{exp} the expected k -mer completeness and $k_{exp} = 50$.

$$K = 1 - Abs(k_{obs} - k_{exp}) / k_{exp} \quad (3)$$

The number of single-copy BUSCOs and the value K are normalized on a 0 to 1 scale following Eq. 4, in which x_i is the initial value, x_{min} the minimum value for all PacBio or Nanopore assemblies, and x_{max} the maximum value for all PacBio or Nanopore assemblies.

$$x_f = (x_i - x_{min}) / (x_{max} - x_{min}) \quad (4)$$

The final completeness score is computed using Eq. 5, in which B_{norm} is the normalized single-copy BUSCO score and K_{norm} is the normalized k -mer completeness value computed previously.

$$Comp = (B_{norm} + K_{norm}) / 2 \quad (5)$$

Performance evaluation

For Flye, Ra, Raven, Shasta and wtdbg2, maximal RAM usage and mean CPU time were measured using the command `time` with 14 threads on a computer with an i9-9900X 3.5 Ghz processor and 128 GB RAM. NextDenovo was run on a computer with an Intel

Xeon E5-2650 with 256 GB of RAM, while Canu was run on a cluster with an AMD Epyc 7551P and 256 Gb of RAM. NextDenovo ran in a few hours, while Canu runs each required several days.

Abbreviations

kb: kilobase; Mb: Megabase; Nanopore: Oxford Nanopore Technologies; OLC: Overlap Layout Consensus; PacBio: Pacific Biosciences.

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12859-021-04118-3>.

Additional file 1. This file includes all Supplementary Figures S1–27 and Tables S1–S8.

Acknowledgements

We thank Antoine Limasset and Paul Simion for their useful advice. We also thank Michael Eitel for prompting us to initiate this benchmark of long-read assemblers. Nanopore reads were generated at Genoscope as part of the France Génomique project 'ALPAGA' coordinated by Etienne Danchin (www.france-genomique.org/projet/alpaga/). Part of this analysis was performed on computing clusters of the Leibniz-Rechenzentrum (LRZ) and the Consortium des Équipements de Calcul Intensif (CÉCI) funded by the Fonds de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under Grant No. 2.5020.11.

Authors' contributions

NG and JFF jointly devised the study. NG, AH and AD ran the assemblies. NG evaluated the assemblies. AH conceived and implemented HapPy. NG and JFF wrote the manuscript. KVD and JFF provided the sequencing data. All authors read and approved the final manuscript.

Funding

This project was funded by the Horizon 2020 research and innovation program of the European Union under the Marie Skłodowska-Curie grant agreement No. 764840 (ITN IGNITE, www.itn-ignite.eu) for NG and JFF, and under the European Research Council (ERC) grant agreement No. 725998 (RHEA) to KVD. AH and AD are Research Fellows of the Fonds de la Recherche Scientifique – FNRS. These funding sources had no role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials

The datasets analyzed in this study were published in [37].

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare no competing interests.

Author details

¹Service Evolution Biologique et Ecologie, Université libre de Bruxelles (ULB), Avenue Franklin D. Roosevelt 50, 1050 Brussels, Belgium. ²Laboratoire d'Ecologie et Génétique Evolutive, Université de Namur, Rue de Bruxelles 61, 5000 Namur, Belgium. ³Département de Biologie des Organismes, Université libre de Bruxelles (ULB), Avenue Franklin D. Roosevelt 50, 1050 Brussels, Belgium. ⁴Interuniversity Institute of Bioinformatics in Brussels - (IB)², Avenue Franklin D. Roosevelt 50, 1050 Brussels, Belgium.

Received: 18 January 2021 Accepted: 2 April 2021

Published online: 05 June 2021

References

1. Sedlazeck FJ, Lee H, Darby CA, Schatz MC. Piercing the dark matter: bioinformatics of long-range sequencing and mapping. *Nat Rev Genet.* 2018;19(6):329–46. <https://doi.org/10.1038/s41576-018-0003-4>.
2. Pollard MO, Gurdasani D, Mentzer AJ, Porter T, Sandhu MS. Long reads: their purpose and place. *Hum Mol Genet.* 2018;27(R2):234–41. <https://doi.org/10.1093/hmg/ddy177>.

3. Patterson MD, Marschall T, Pisanti N, Van Iersel L, Stougjie L, Klau GW, Schönhuth A. WhatsHap: weighted haplotype assembly for future-generation sequencing reads. *J Comput Biol.* 2015;22(6):498–509. <https://doi.org/10.1089/cmb.2014.0157>.
4. Wenger AM, Peluso P, Rowell WJ, Chang P-C, Hall RJ, Concepcion GT, Ebler J, Functammasan A, Kolesnikov A, Olson ND, Töpfer A, Alonge M, Mahmoud M, Qian Y, Chin C-S, Phillippy AM, Schatz MC, Myers G, Depristo MA, Ruan J, Marschall T, Sedlazeck FJ, Zook JM, Li H, Koren S, Carroll A, Rank DR, Hunkapiller MW. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. *Nat Biotechnol.* 2019;37(10):1155–62. <https://doi.org/10.1038/s41587-019-0217-9>.
5. Kundu R, Casey J, Sung W-K. HyPo: super fast & accurate polisher for long read assemblies. *bioRxiv.* 2019. <https://doi.org/10.1101/2019.12.19.882506>.
6. Zimin AV, Salzberg SL. The genome polishing tool POLCA makes fast and accurate corrections in genome assemblies. *PLoS Comput Biol.* 2020;16(6):1007981. <https://doi.org/10.1371/journal.pcbi.1007981>.
7. Jain M, Koren S, Miga KH, Quick J, Rand AC, Sasani TA, Tyson JR, Beggs AD, Dilthey AT, Fiddes IT, Malla S, Marriott H, Nieto T, O’Grady J, Olsen HE, Pedersen BS, Rhie A, Richardson H, Quinlan AR, Snutch TP, Tee L, Paten B, Phillippy AM, Simpson JT, Loman NJ, Loose M. Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nat Biotechnol.* 2018;36(4):338–45. <https://doi.org/10.1038/nbt.4060>.
8. Miga KH, Koren S, Rhie A, Vollger MR, Gershman A, Bzikadze A, Brooks S, Howe E, Porubsky D, Logsdon GA, Schneider VA, Potapova T, Wood J, Chow W, Armstrong J, Fredrickson J, Pak E, Tigyi K, Kremitzki M, Markovic C, Maduro V, Dutra A, Bouffard GG, Chang AM, Hansen NF, Thibaud-Nissen F, Schmitt AD, Belton JM, Selvaraj S, Dennis MY, Soto DC, Sahasrabudhe R, Kaya G, Quick J, Loman NJ, Holmes N, Loose M, Surti U, Risques RA, Graves Lindsay TA, Fulton R, Hall I, Paten B, Howe K, Timp W, Young A, Mullikin JC, Pevzner PA, Gerton JL, Sullivan BA, Eichler EE, Phillippy AM. Telomere-to-telomere assembly of a complete human X chromosome. *Nature.* 2020;585(7823), 79–84. <https://doi.org/10.1038/s41586-020-2547-7>.
9. Miller JR, Koren S, Sutton G. Assembly algorithms for next-generation sequencing data. *Genomics.* 2010;95(6):315–27. <https://doi.org/10.1016/j.ygeno.2010.03.001>.
10. Kolmogorov M, Yuan J, Lin Y, Pevzner PA. Assembly of long, error-prone reads using repeat graphs. *Nat Biotechnol.* 2019;37(5):540–6. <https://doi.org/10.1038/s41587-019-0072-8>.
11. Vaser R, Šikić M. Yet another *de novo* genome assembler. In: International symposium on image and signal processing and analysis, ISPA. 2019. p. 147–51. <https://doi.org/10.1109/ISPA.2019.8868909>.
12. Vaser R, Šikić M. Raven: a *de novo* genome assembler for long reads. *bioRxiv.* 2020. <https://doi.org/10.1101/2020.08.07.242461>.
13. Shafin K, Pesout T, Lorig-roach R, Haukness M, Olsen HE, Bosworth C, Armstrong J, Tigyi K, Maurer N, Koren S, Sedlazeck FJ, Marschall T, Mayes S, Costa V, Zook JM, Liu KJ, Kilburn D, Sorensen M, Munson KM, Vollger MR, Monlong J, Garrison E, Eichler EE, Salama S, Haussler D, Green RE, Akeson M, Phillippy A, Miga KH, Carnevali P, Jain M, Paten B. Nanopore sequencing and the Shasta toolkit enable efficient *de novo* assembly of eleven human genomes. *Nat Biotechnol.* 2020;38(9):1044–53. <https://doi.org/10.1038/s41587-020-0503-6>.
14. Ruan J, Li H. Fast and accurate long-read assembly with wtdbg2. *Nat Methods.* 2020;17(2):155–8. <https://doi.org/10.1038/s41592-019-0669-3>.
15. Koren S, Walenz BP, Berlin K, Miller JR, Bergman NH, Phillippy AM. Canu: scalable and accurate long-read assembly via adaptive *k*-mer weighting and repeat separation. *Genome Res.* 2017;25(2):1–11. <https://doi.org/10.1101/gr.215087.116>.
16. NextOmics: NextDeNovo. 2019. <https://github.com/Nextomics/NextDeNovo>.
17. Edge P, Bafna V, Bansal V. HapCUT2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Res.* 2017;27(5):801–12.
18. Chin C-S, Peluso P, Sedlazeck FJ, Nattestad M, Concepcion GT, Clum A, Dunn C, O’Malley R, Figueroa-Balderas R, Morales-Cruz A, et al. Phased diploid genome assembly with single-molecule real-time sequencing. *Nat Methods.* 2016;13(12):1050–4.
19. Wick RR, Holt KE. Benchmarking of long-read assemblers for prokaryote whole genome sequencing. *F1000Research* 8, 2019;2138. <https://doi.org/10.12688/f1000research.21782.1>.
20. Flot J-F, Hespels B, Li X, Noel B, Arkhipova I, Danchin EGJ, Hejnol A, Henrissat B, Koszul R, Aury J-M, Barbe V, Barthélémy RM, Bast J, Bazykin GA, Chabrol O, Couloux A, Da Rocha M, Da Silva C, Gladyshev E, Gouret P, Hallatschek O, Hecox-Lea B, Labadie K, Lejeune B, Piskurek O, Poulain J, Rodriguez F, Ryan JF, Vakhrusheva OA, Wajnberg E, Wirth B, Yushenova I, Kellis M, Kondrashov AS, Welch DBM, Pontarotti P, Weissenbach J, Wincker P, Jaillon O, Van Doninck K. Genomic evidence for ameiotic evolution in the bdelloid rotifer *Adineta vaga*. *Nature.* 2013;500(7463):453–7. <https://doi.org/10.1038/nature12326>.
21. Leffler EM, Bullaughey K, Matute DR, Meyer WK, Ségurel L, Venkat A, Andolfatto P, Przeworski M. Revisiting an old riddle: what determines genetic diversity levels within species? *PLoS Biol.* 2012;10(9):1001388. <https://doi.org/10.1371/journal.pbio.1001388>.
22. Huang S, Kang M, Xu A. HaploMerger2: rebuilding both haploid sub-assemblies from high-heterozygosity diploid genome assembly. *Bioinformatics.* 2017;33(16):2577–9. <https://doi.org/10.1093/bioinformatics/btx220>.
23. Guan D, McCarthy SA, Wood J, Howe K, Wang Y, Durbin R. Identifying and removing haplotypic duplication in primary genome assemblies. *Bioinformatics.* 2020;36(9):2896–8. <https://doi.org/10.1093/bioinformatics/btaa025>.
24. Roach MJ, Schmidt SA, Borneman AR. Purge Haplotigs: allelic contig reassignment for third-gen diploid genome assemblies. *BMC Bioinform.* 2018;19(1):1–10. <https://doi.org/10.1186/s12859-018-2485-7>.
25. Simão FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics.* 2015;31(19):3210–2. <https://doi.org/10.1093/bioinformatics/btv351>.
26. Houtain A, Guiguelmoni N, Flot J-F. AntoineHo/HapPy: v0.1 (version v0.1.2zen). *Zenodo.* 2020. <https://doi.org/10.5281/zenodo.4292076>.
27. Wick RR. *Filtlong.* 2017. <https://github.com/rwick/Filtlong>.
28. Van der Verren SE, Van Gerven N, Jonckheere W, Hambley R, Singh P, Kilmour J, Jordan M, Wallace EJ, Jayasinghe L, Remaut H. A dual-constriction biological nanopore resolves homonucleotide sequences with high fidelity. *Nat Biotechnol.* 2020. <https://doi.org/10.1038/s41587-020-0570-8>.

29. Wick RR, Judd LM, Holt KE. Performance of neural network basecalling tools for Oxford Nanopore sequencing. *Genome Biol.* 2019;20(1):129. <https://doi.org/10.1186/s13059-019-1727-y>.
30. Mapleson D, Garcia Accinelli G, Kettleborough G, Wright J, Clavijo BJ. KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies. *Bioinformatics.* 2016;33(4):574–6. <https://doi.org/10.1093/bioinformatics/btw663>.
31. Ranallo-Benavidez TR, Jaron KS, Schatz MC. GenomeScope 2.0 and Smudgeplot for reference-free profiling of polyploid genomes. *Nat Commun.* 2020;11(1):1432. <https://doi.org/10.1038/s41467-020-14998-3>.
32. Mulligan KL, Hiebert TC, Jeffery NW, Gregory TR. First estimates of genome size in ribbon worms (phylum Nemertea) using flow cytometry and Feulgen image analysis densitometry. *Can J Zool.* 2014;92(10):847–51. <https://doi.org/10.1139/cjz-2014-0068>.
33. Wick RR, Schultz MB, Zobel J, Holt KE. Bandage: interactive visualization of de novo genome assemblies. *Bioinformatics.* 2015;31(20):3350–2. <https://doi.org/10.1093/bioinformatics/btv383>.
34. Murigneux V, Rai SK, Furtado A, Bruxner TJ, Tian W, Harliwong I, Wei H, Yang B, Ye Q, Anderson E, et al. Comparison of long-read methods for sequencing and assembly of a plant genome. *GigaScience.* 2020;9(12):146. <https://doi.org/10.1093/gigascience/giaa146>.
35. Nurk S, Walenz BP, Rhie A, Vollger MR, Logsdon GA, Grothe R, Miga KH, Eichler EE, Phillippy AM, Koren S. HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads. *Genome Res.* 2020;30(9):1291–305. <https://doi.org/10.1101/gr.263566.120>.
36. Cheng H, Concepcion GT, Feng X, Zhang H, Li H. Haplotype-resolved de novo assembly with phased assembly graphs. *Nat. Methods.* 2021;18(2), 170–5. <https://doi.org/10.1038/s41592-020-01056-5>.
37. Simion P, Narayan J, Houtain A, Derzelle A, Baudry L, Nicolas E, Cariou M, Guiglielmoni N, Kozłowski DKL, Gaudray FR, Terwagne M, Virgo J, Noel B, Wincker P, Danchin EGJ, Marbouty M, Hallet B, Koszul R, Limasset A, Flot J-F, Van Doninck K. Homologous chromosomes in asexual rotifer *Adineta vaga* suggest automixis. *bioRxiv.* 2020. <https://doi.org/10.1101/2020.06.16.155473>.
38. Li H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics.* 2018;34(18):3094–100. <https://doi.org/10.1093/bioinformatics/bty191>.
39. Savitzky A, Golay MJ. Smoothing and differentiation of data by simplified least squares procedures. *Anal Chem.* 1964;36(8):1627–39. <https://doi.org/10.1021/ac60214a047>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.



Supplementary data

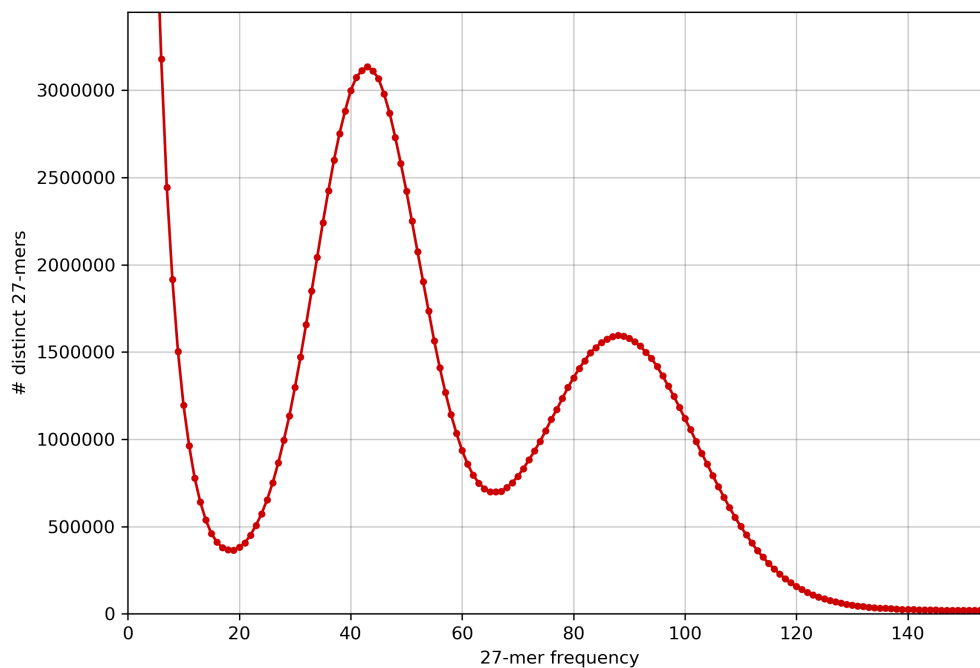


Figure S1: k -mer spectrum of *Adineta vaga* using Illumina reads and KAT v2.4.2. The first peak corresponds to heterozygous k -mers (around $45\times$) and the second peak corresponds to homozygous k -mers.

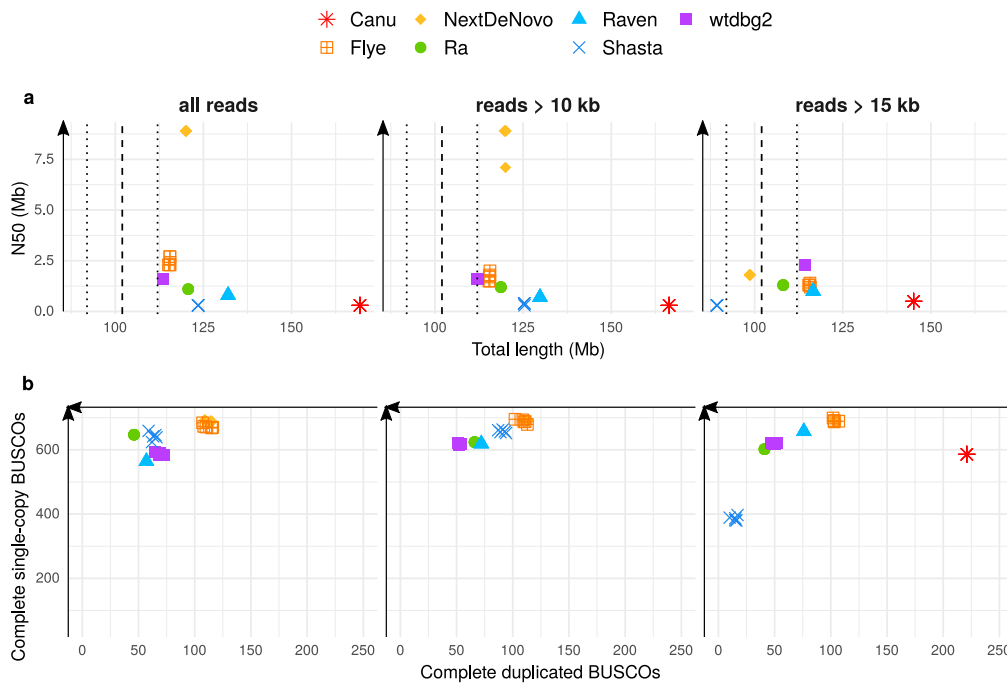


Figure S2: Statistics of PacBio assemblies obtained from the full PacBio dataset or with a read-filtering step prior to assembly based on read length exclusively, using different thresholds: 10 kb, 15 kb. All assemblies were run five times to assess the reproducibility of the output produced by each assembler. a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs.

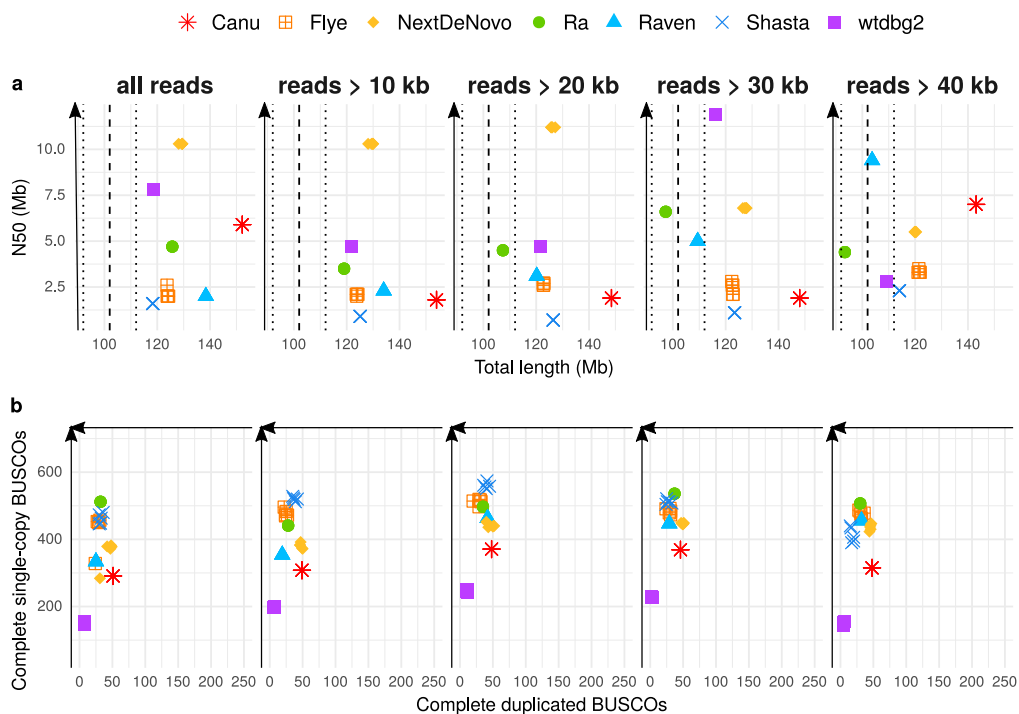


Figure S3: Statistics of Nanopore assemblies obtained from the full Nanopore dataset or with a read-filtering step prior to assembly based on read length exclusively, using different thresholds: 10 kb, 20 kb, 30 kb, 40 kb. All assemblies were run five times to assess the reproducibility of the output produced by each assembler. a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs.

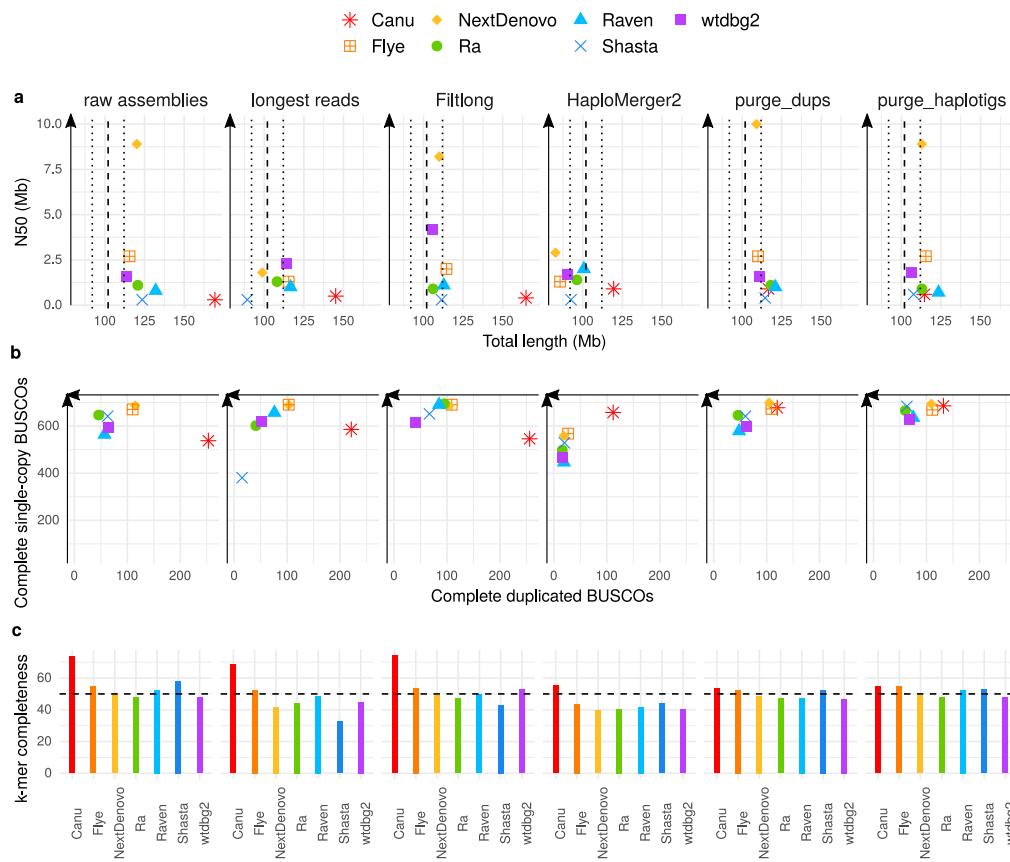


Figure S4: Statistics of raw assemblies obtained from the full PacBio dataset (raw assemblies), with a preliminary read filtering step (keeping only reads larger than 15 kb, or those selected by Filtlong based on quality and length) or a subsequent removal of uncollapsed haplotypes with HaploMerger2, purge_dups, or purge_haplotigs. a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) *k*-mer completeness. The dashed line indicates the expected 50% completeness.

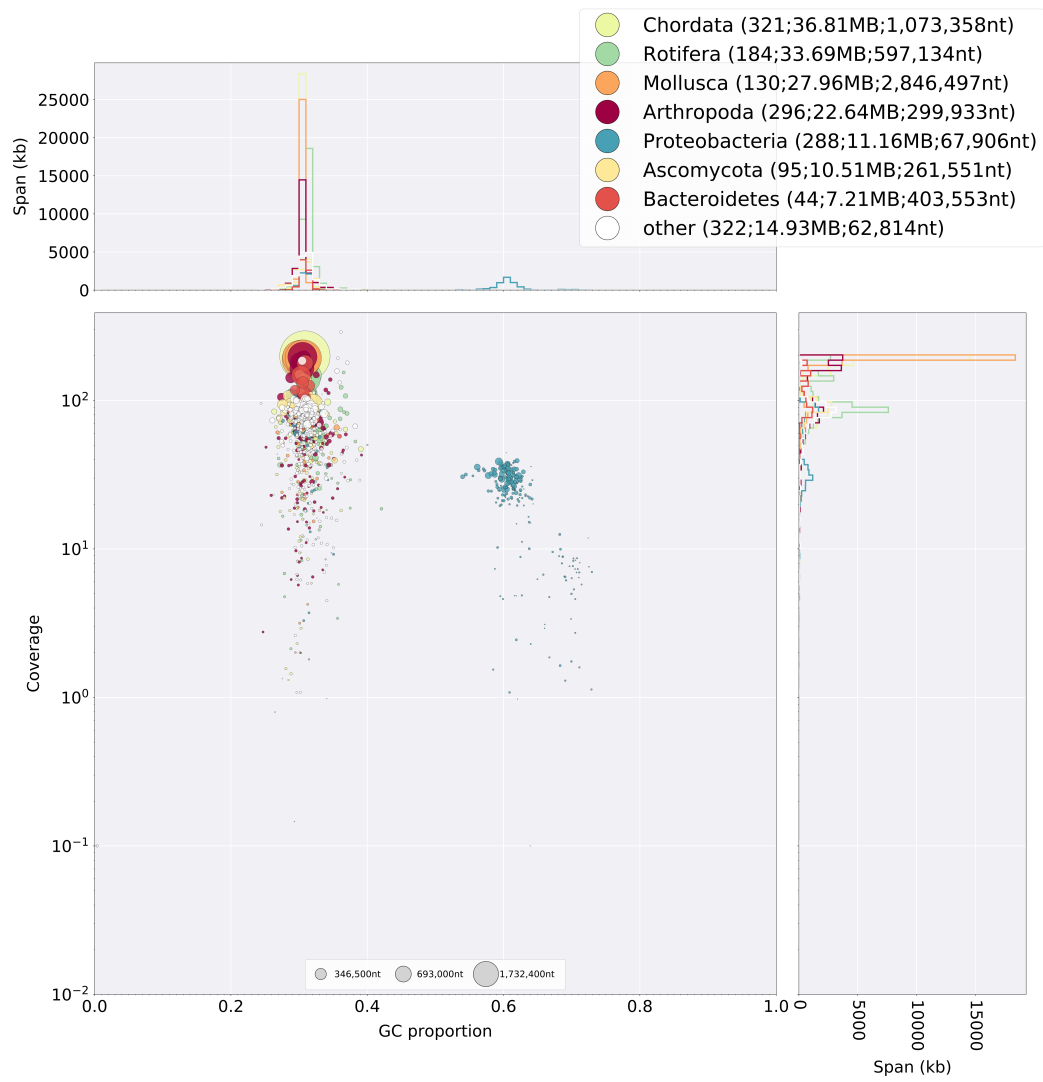


Figure S5: Blobtools v1.0 analysis of a Canu assembly of the full PacBio dataset.

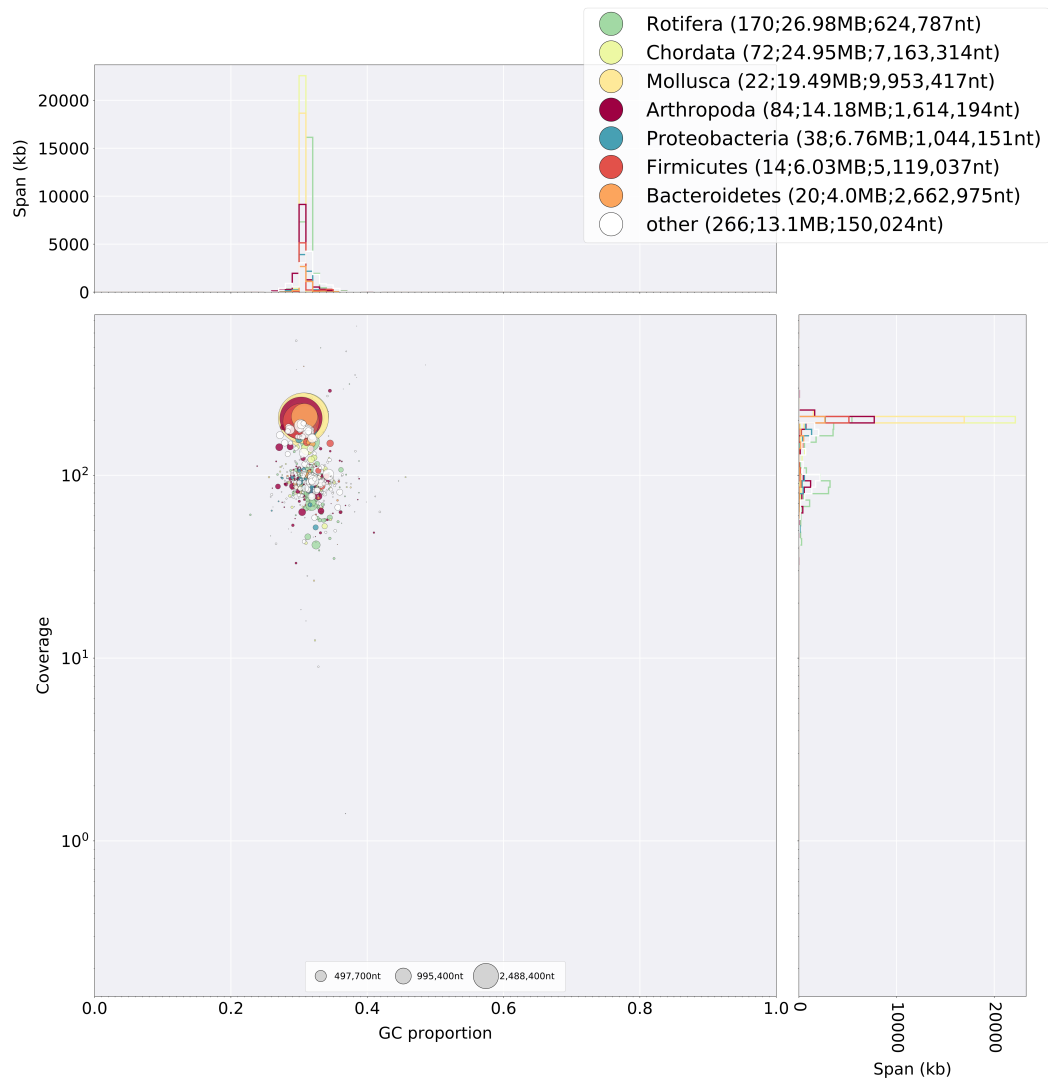


Figure S6: Blobtools v1.0 analysis of a Flye assembly of the full PacBio dataset.

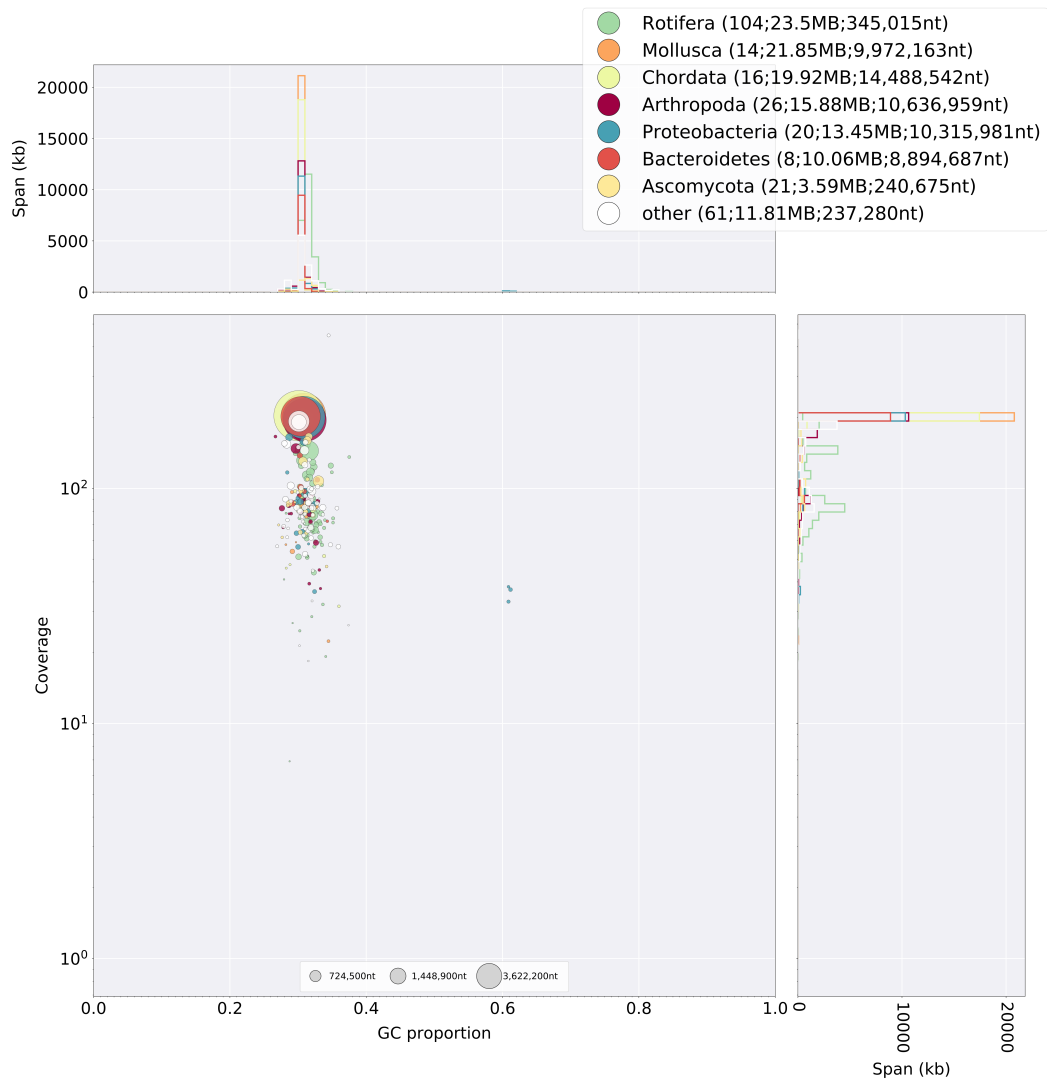


Figure S7: Blobtools v1.0 analysis of a NextDenovo assembly of the full PacBio dataset.

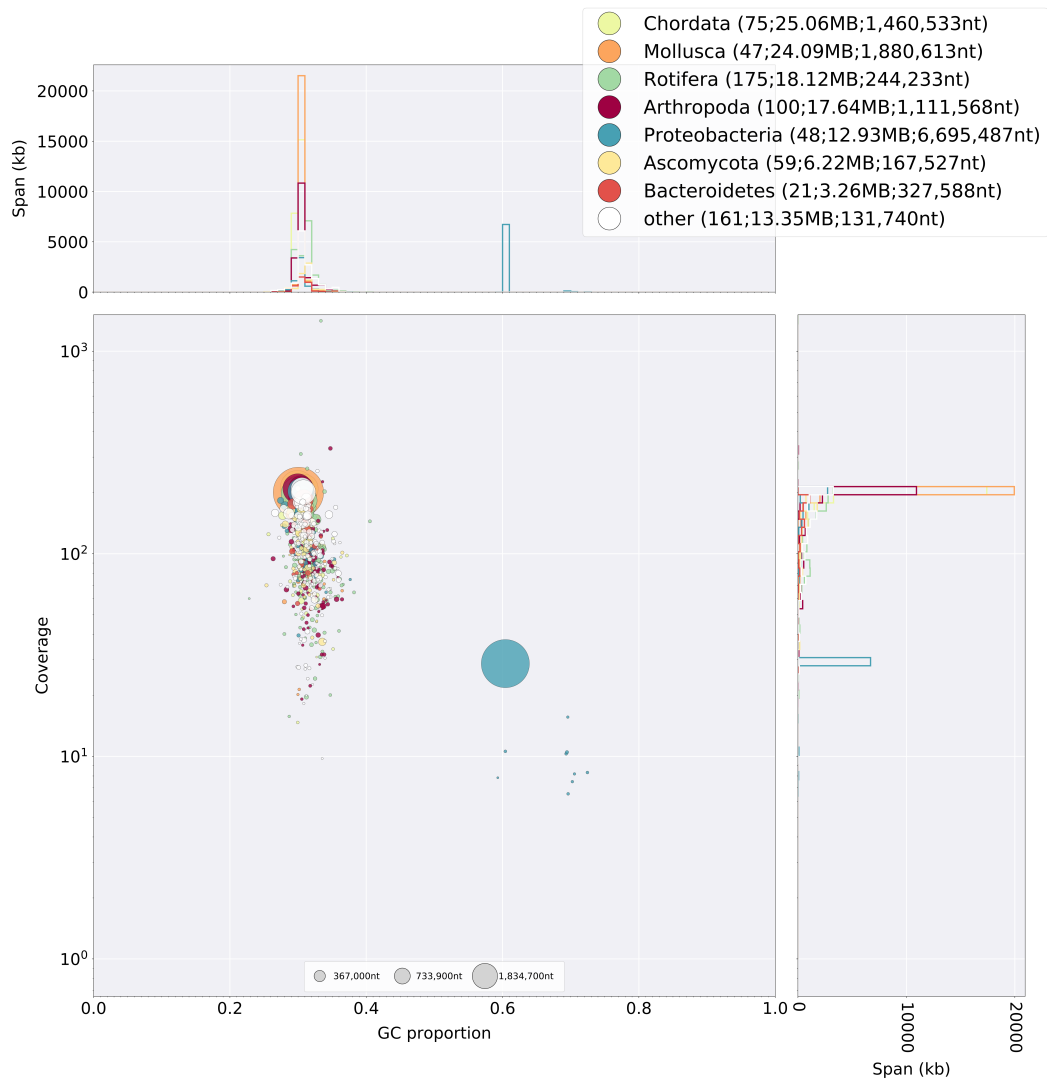


Figure S8: Blobtools v1.0 analysis of a Ra assembly of the full PacBio dataset.

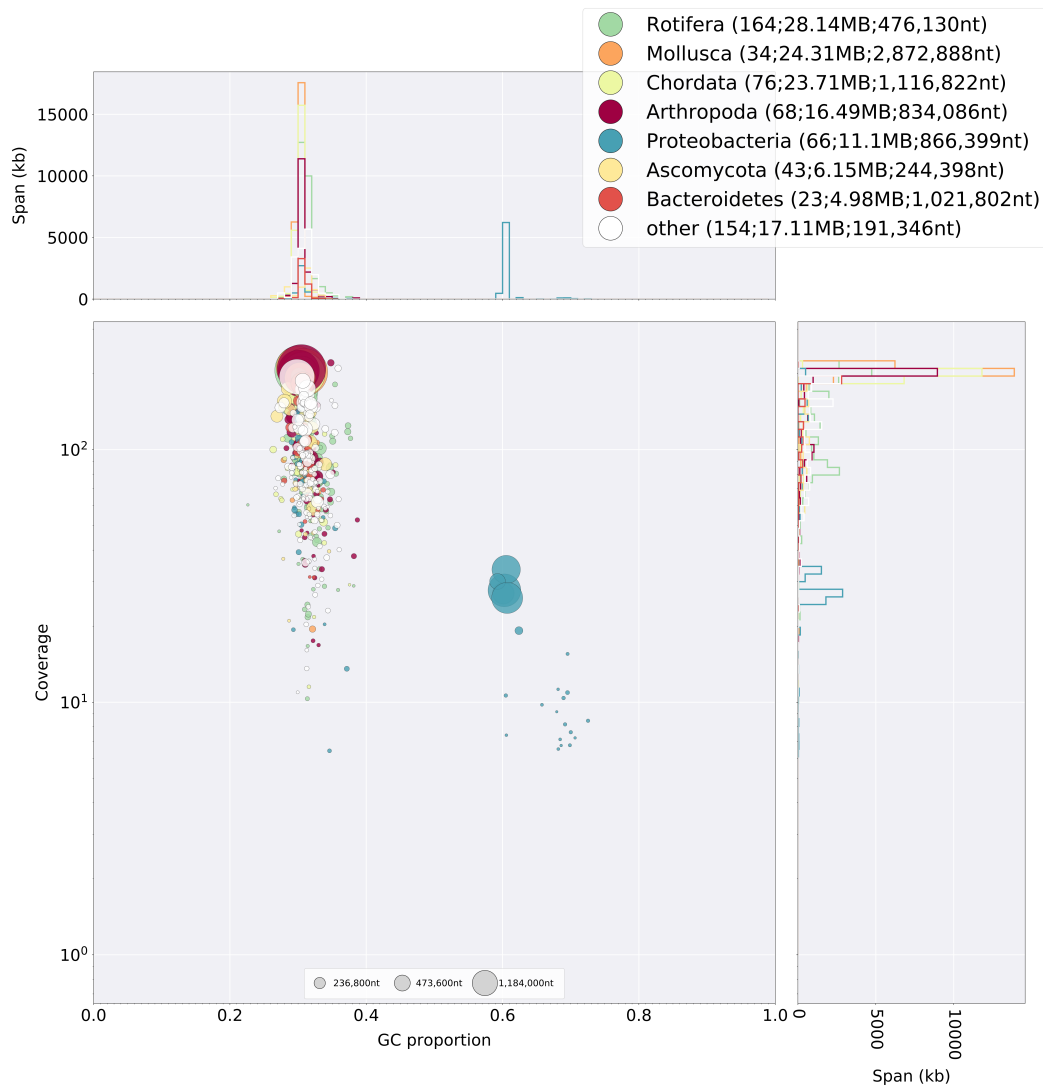


Figure S9: Blobtools v1.0 analysis of a Raven assembly of the full PacBio dataset.

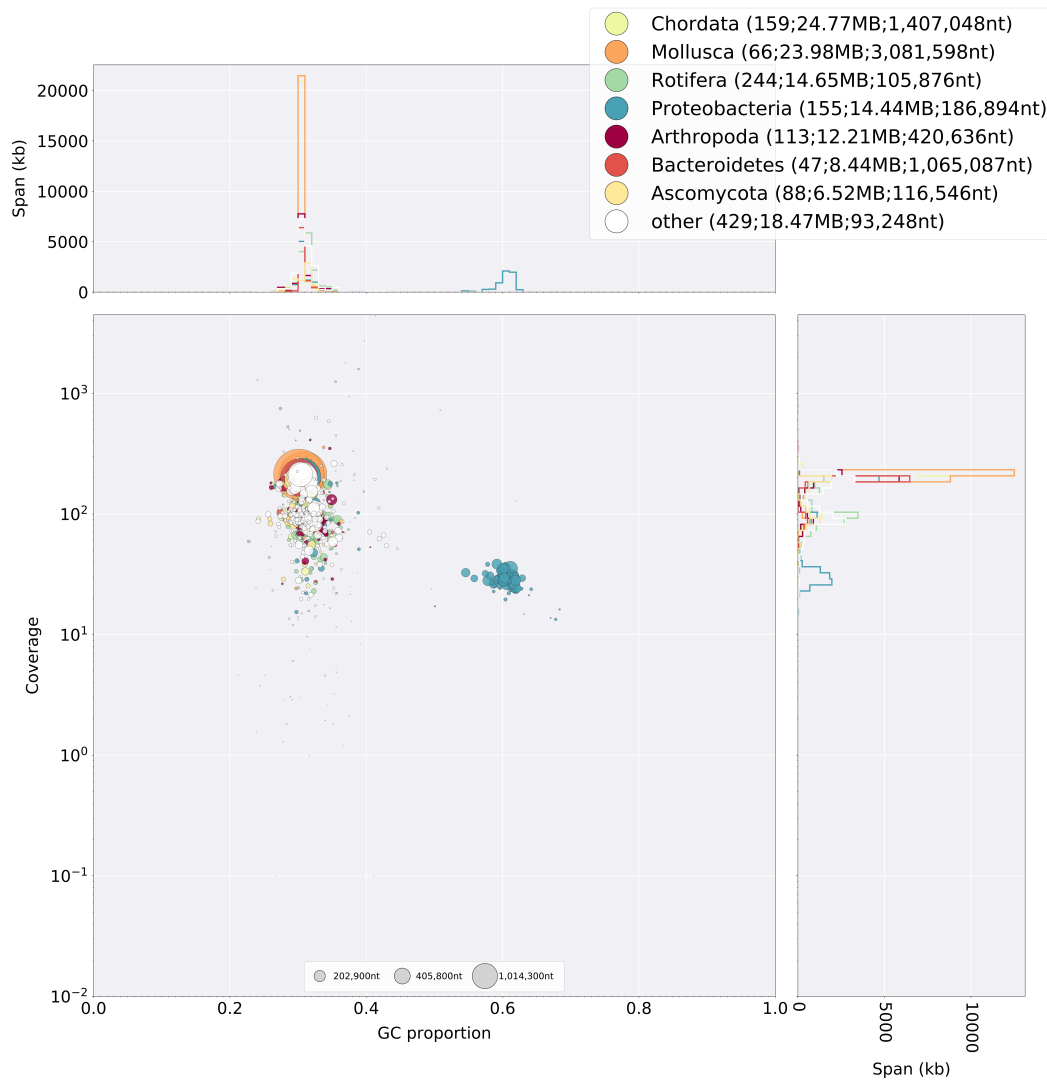


Figure S10: Blobtools v1.0 analysis of a Shasta assembly of the full PacBio dataset.

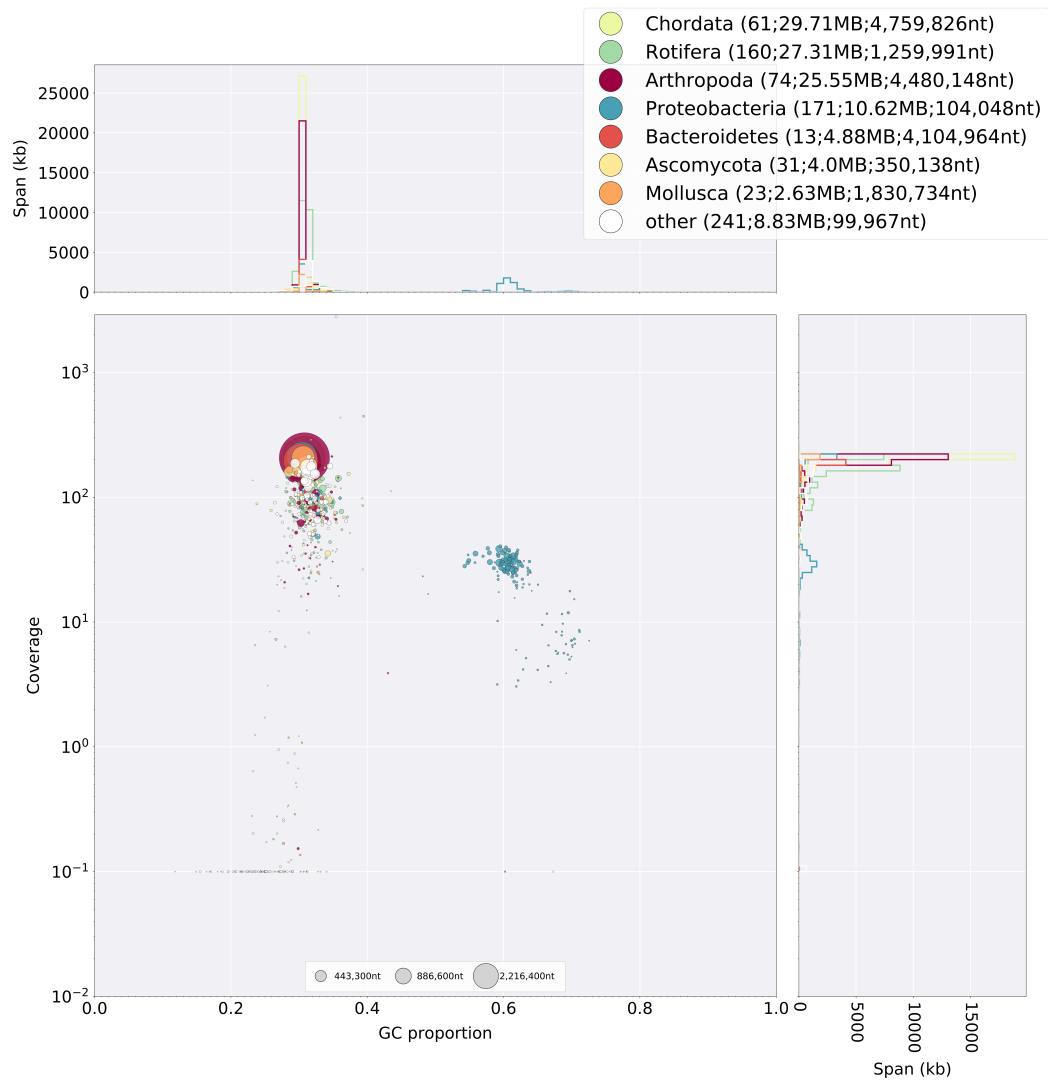


Figure S11: Blobtools v1.0 analysis of a wtdbg2 assembly of the full PacBio dataset.

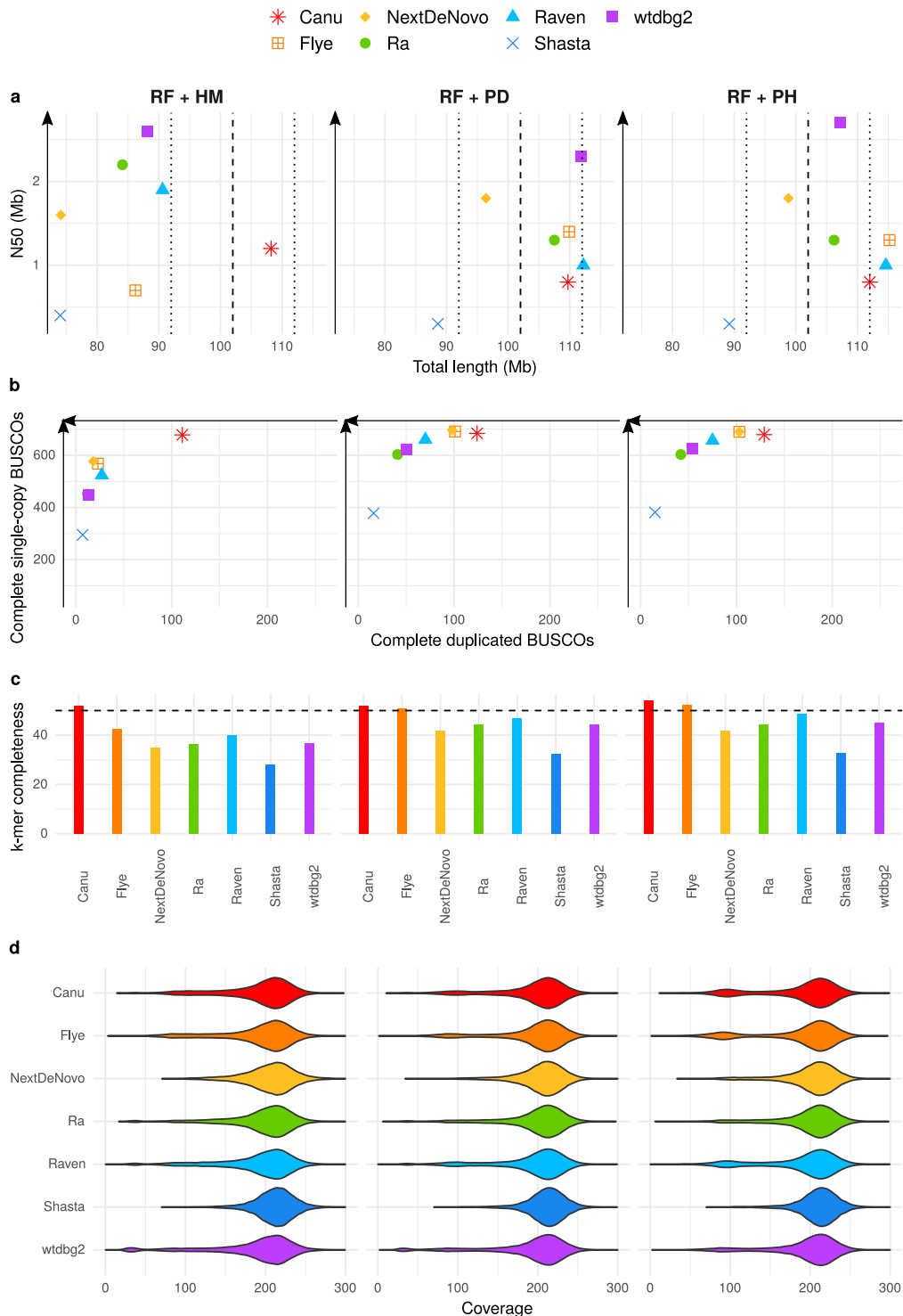


Figure S12: Statistics of PacBio assemblies obtained from the filtered PacBio dataset of reads longer than 15 kb, with a subsequent removal of uncollapsed haplotypes with HaploMerger2 (HM), purge_dups (PD), or purge_haplotigs (PH). a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a ± 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) k -mer completeness. The dashed line indicates the expected 50% completeness. d) Long-read coverage distribution over the contigs.

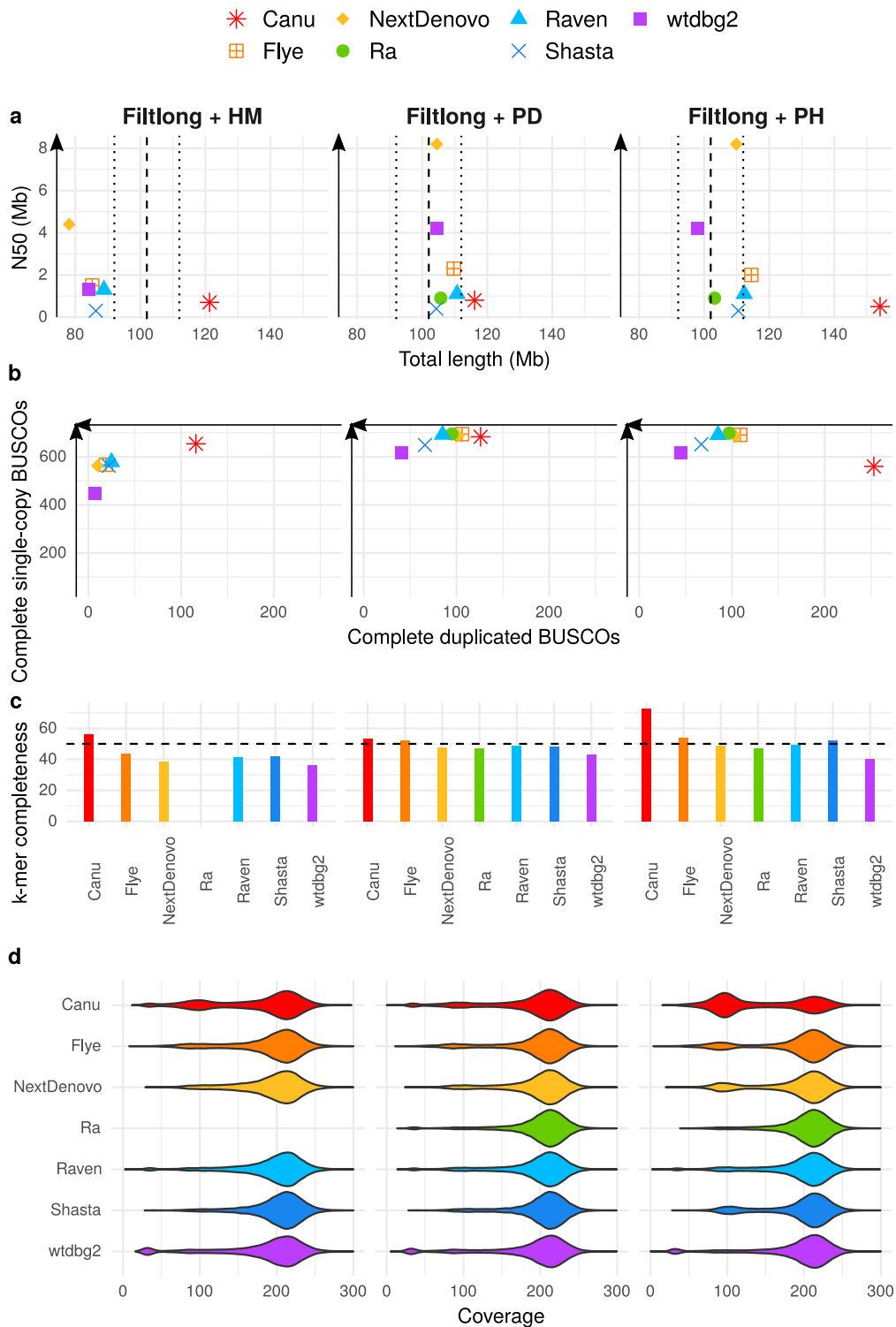


Figure S13: Statistics of PacBio assemblies obtained from the PacBio dataset filtered with Filtrlong, with a subsequent removal of uncollapsed haplotypes with HaploMerger2 (HM), purge_dups (PD), or purge_haplotigs (PH). a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) k -mer completeness. The dashed line indicates the expected 50% completeness. d) Long-read coverage distribution over the contigs.

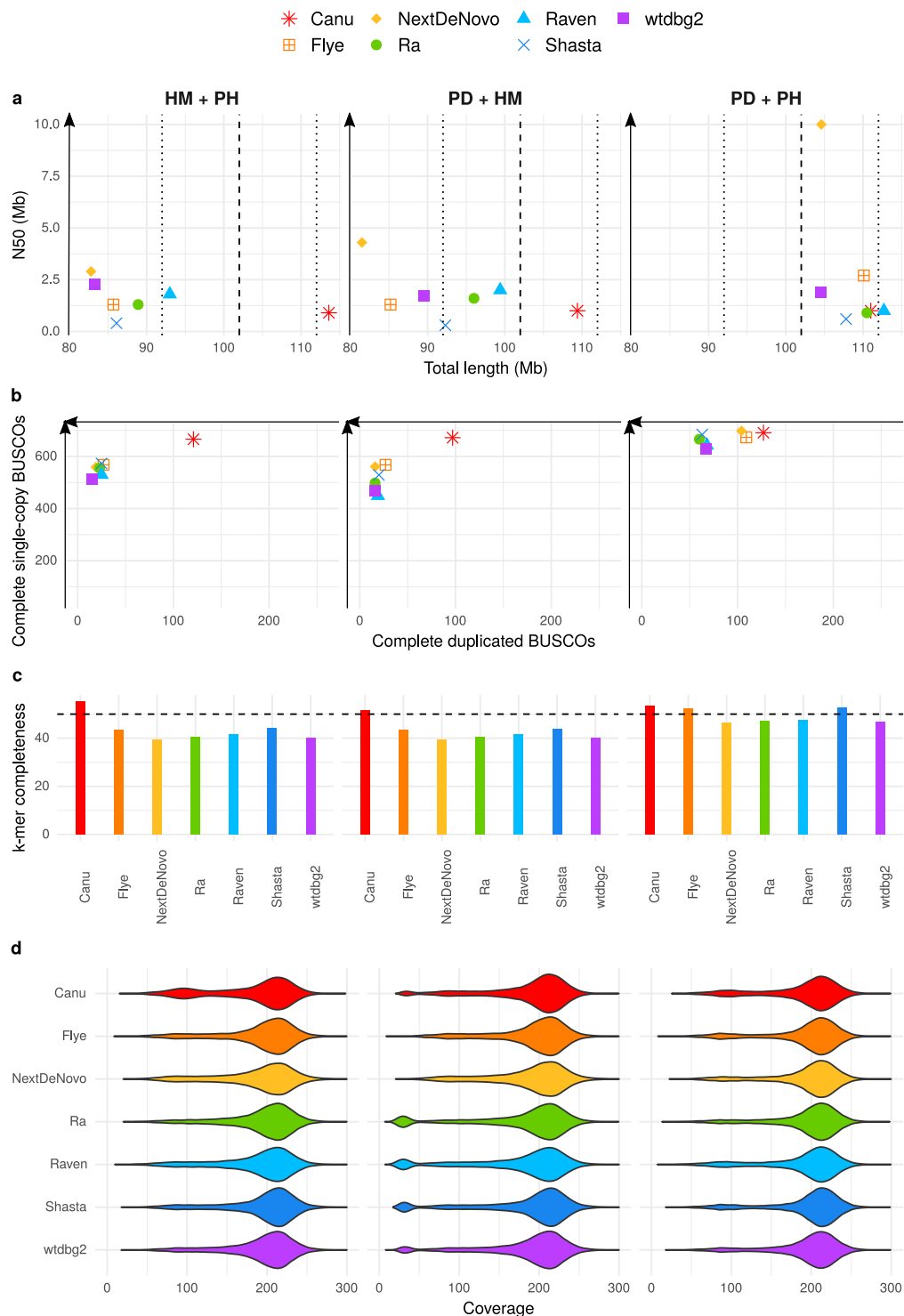


Figure S14: Statistics of PacBio assemblies obtained from the full PacBio dataset with a subsequent removal of uncollapsed haplotypes with combinations of HaploMerger2 (HM), purge_dups (PD), and purge_haplotigs (PH). a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a ± 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) k -mer completeness. The dashed line indicates the expected 50% completeness. d) Long-read coverage distribution over the contigs.

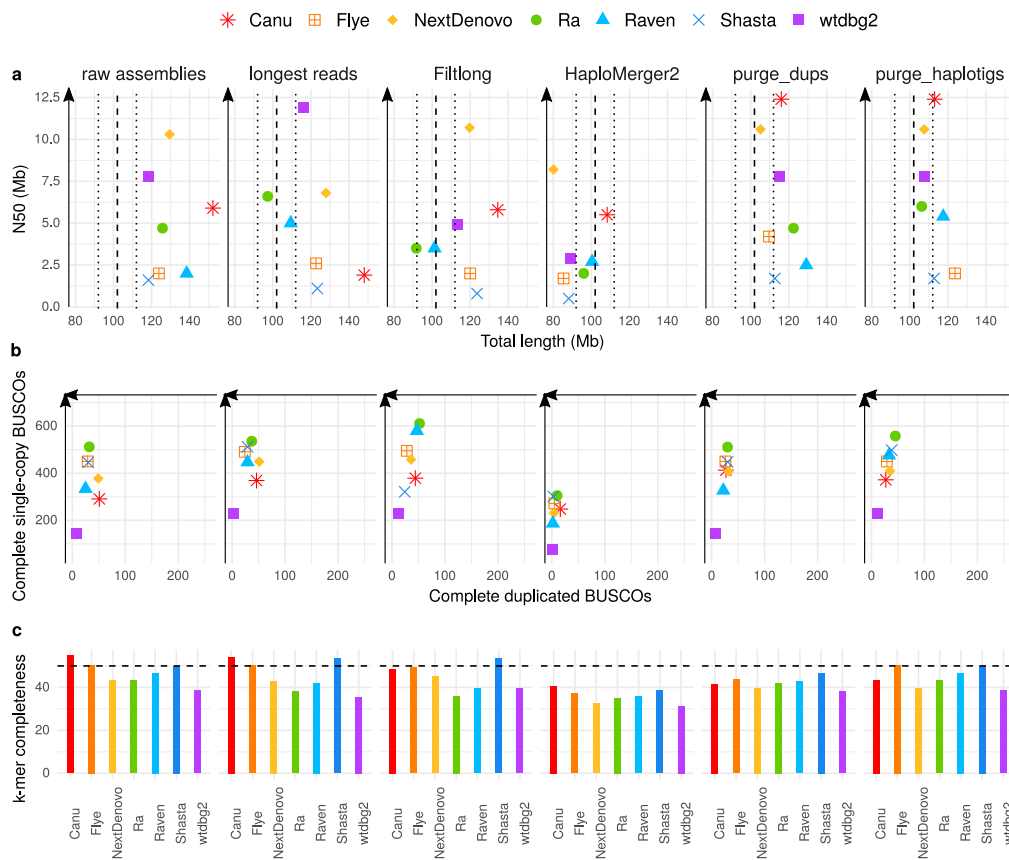


Figure S15: Statistics of raw assemblies obtained from the full Nanopore dataset (raw assemblies), with a preliminary read filtering step (keeping only reads larger than 30 kb, or those selected by Filtlong based on quality and length) or a subsequent removal of uncollapsed haplotypes with HaploMerger2, `purge_dups`, or `purge_haplotigs`. a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) k -mer completeness. The dashed line indicates the expected 50% completeness.

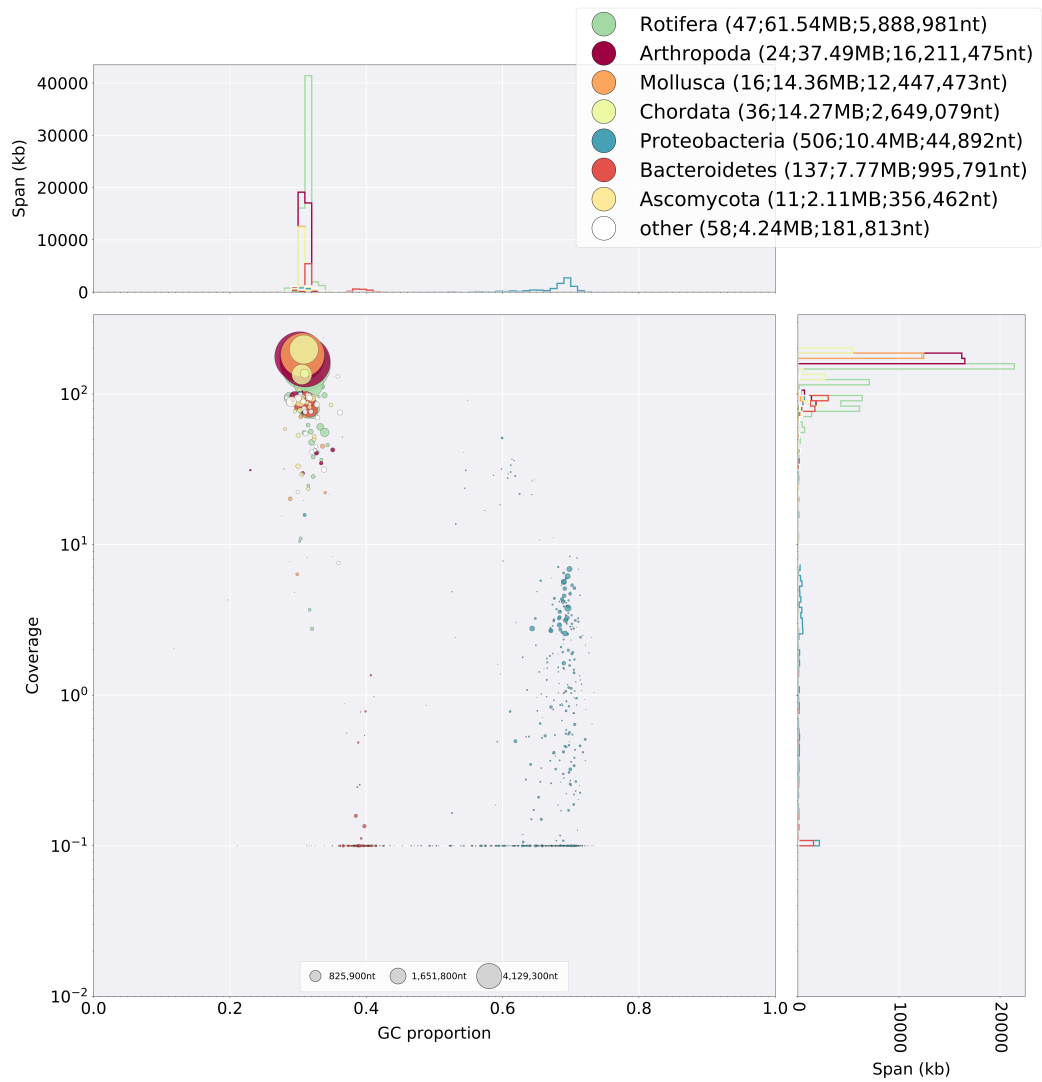


Figure S16: Blobtools v1.0 analysis of a Canu assembly of the full Nanopore dataset.

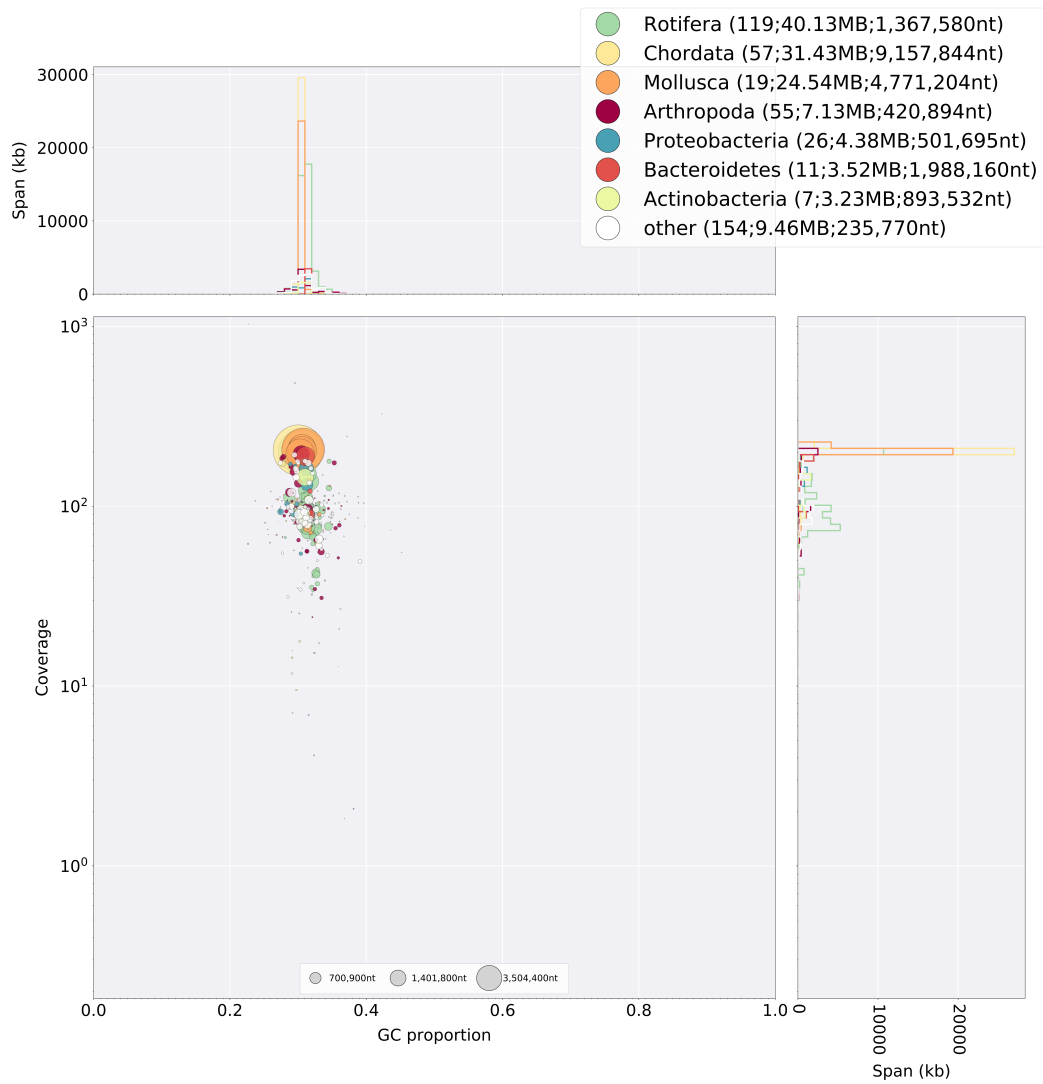


Figure S17: BlobsTools v1.0 analysis of a Flye assembly of the full Nanopore dataset.

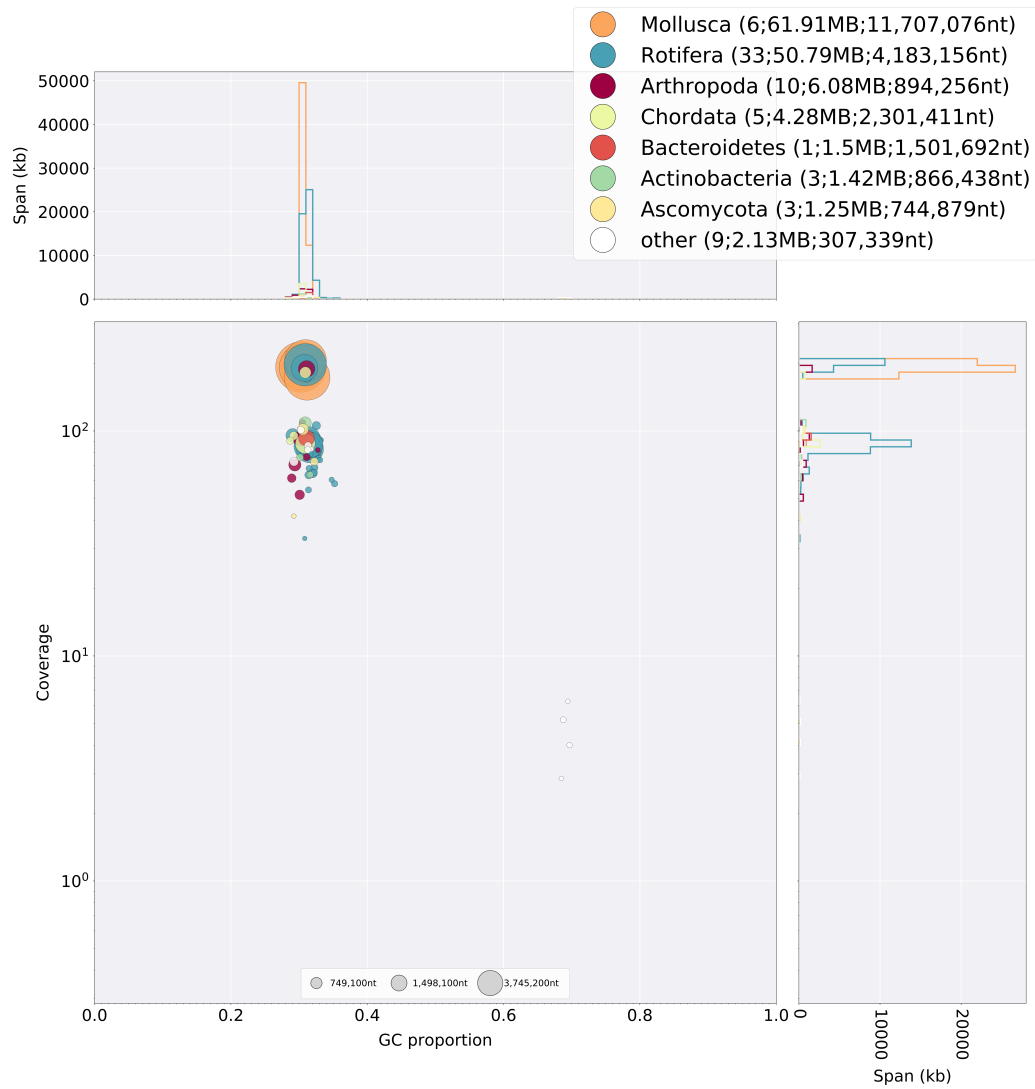


Figure S18: Blobtools v1.0 analysis of a NextDenovo assembly of the full Nanopore dataset.

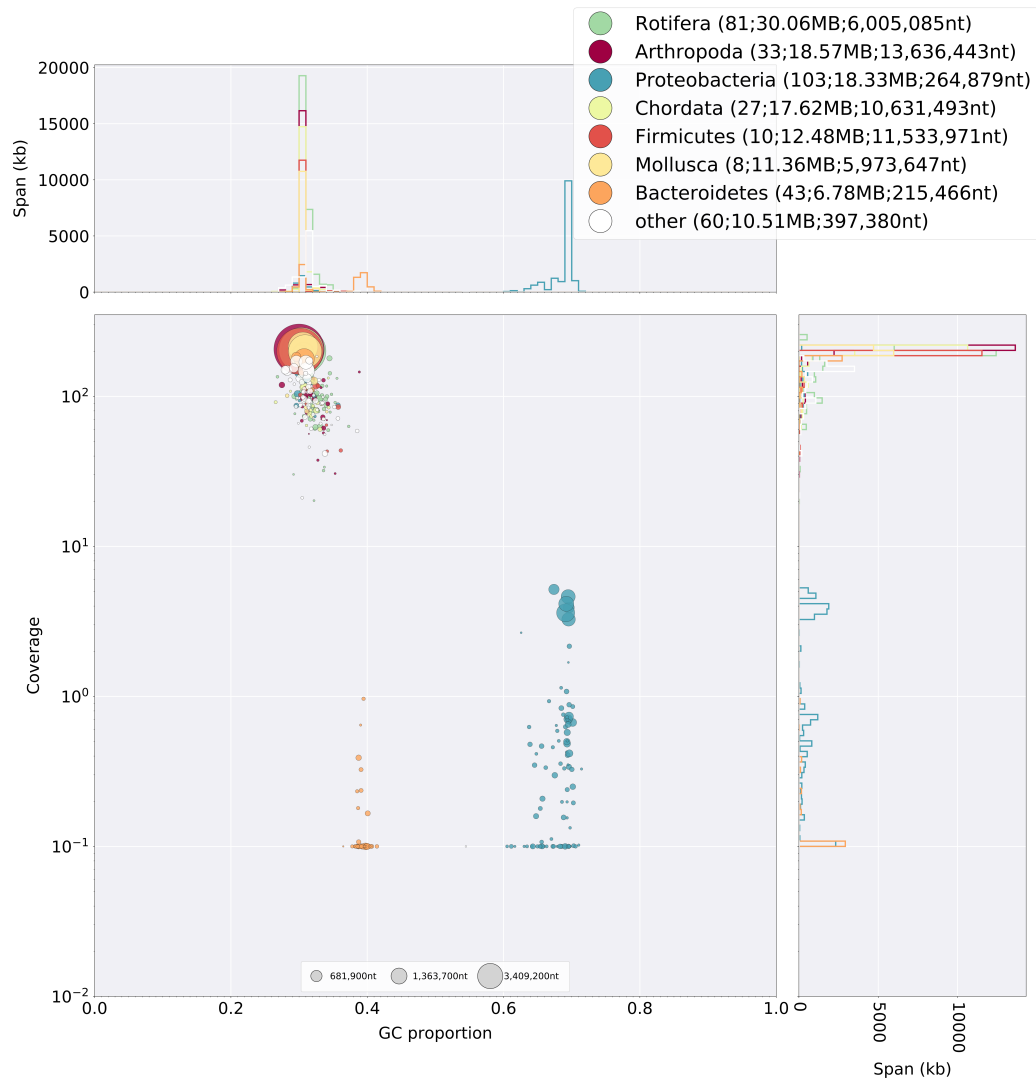


Figure S19: BlobsTools v1.0 analysis of a Ra assembly of the full Nanopore dataset.

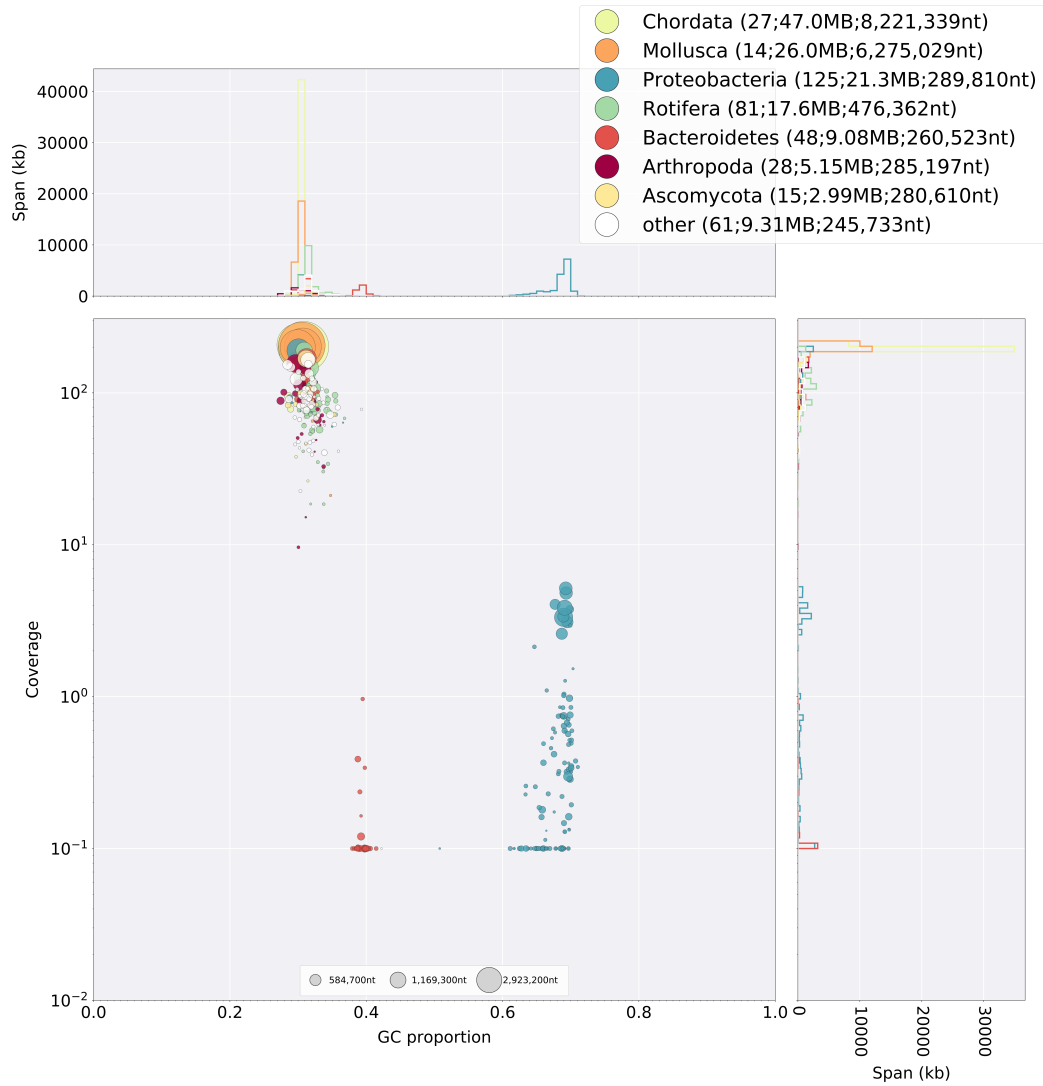


Figure S20: Blobtools v1.0 analysis of a Raven assembly of the full Nanopore dataset.

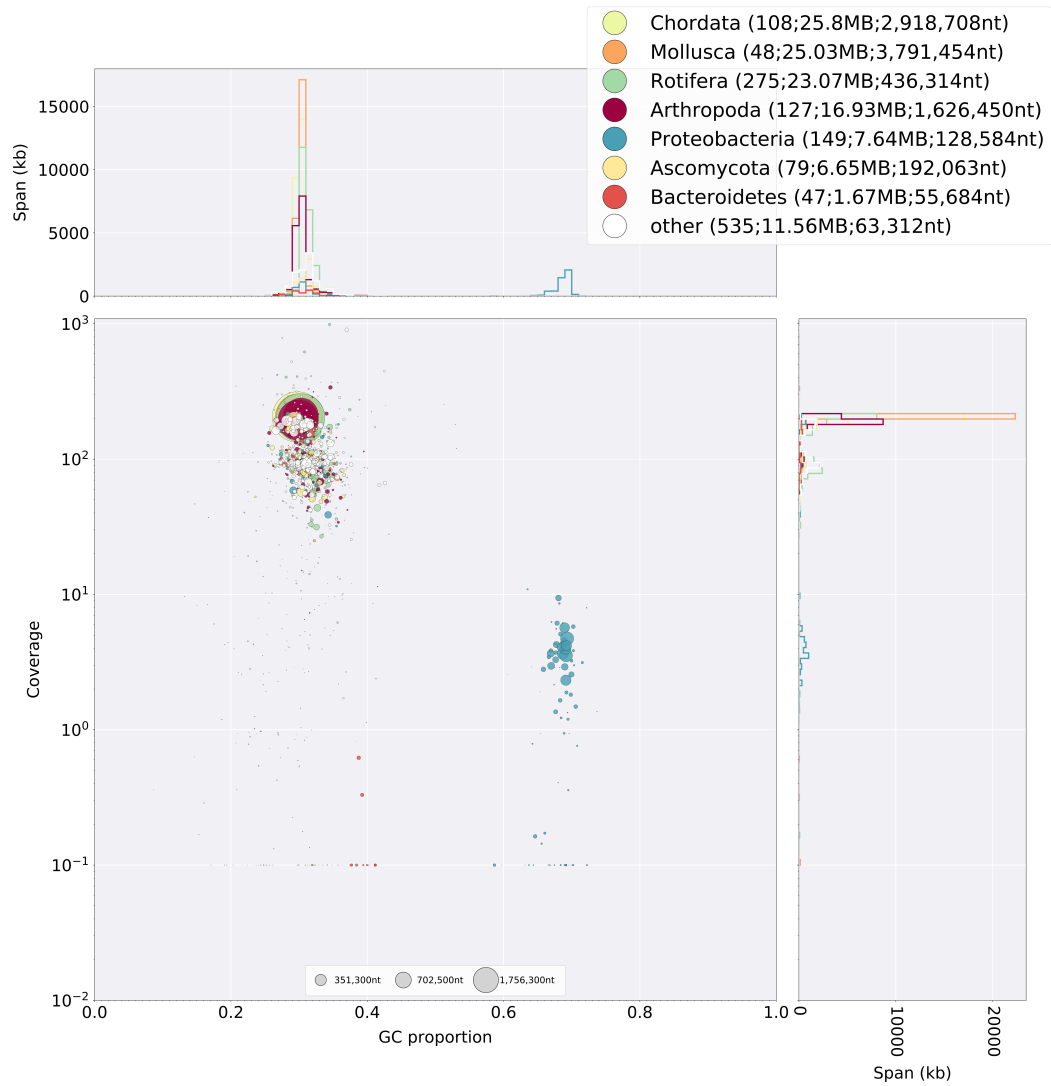


Figure S21: Blobtools v1.0 analysis of a Shasta assembly of the full Nanopore dataset.

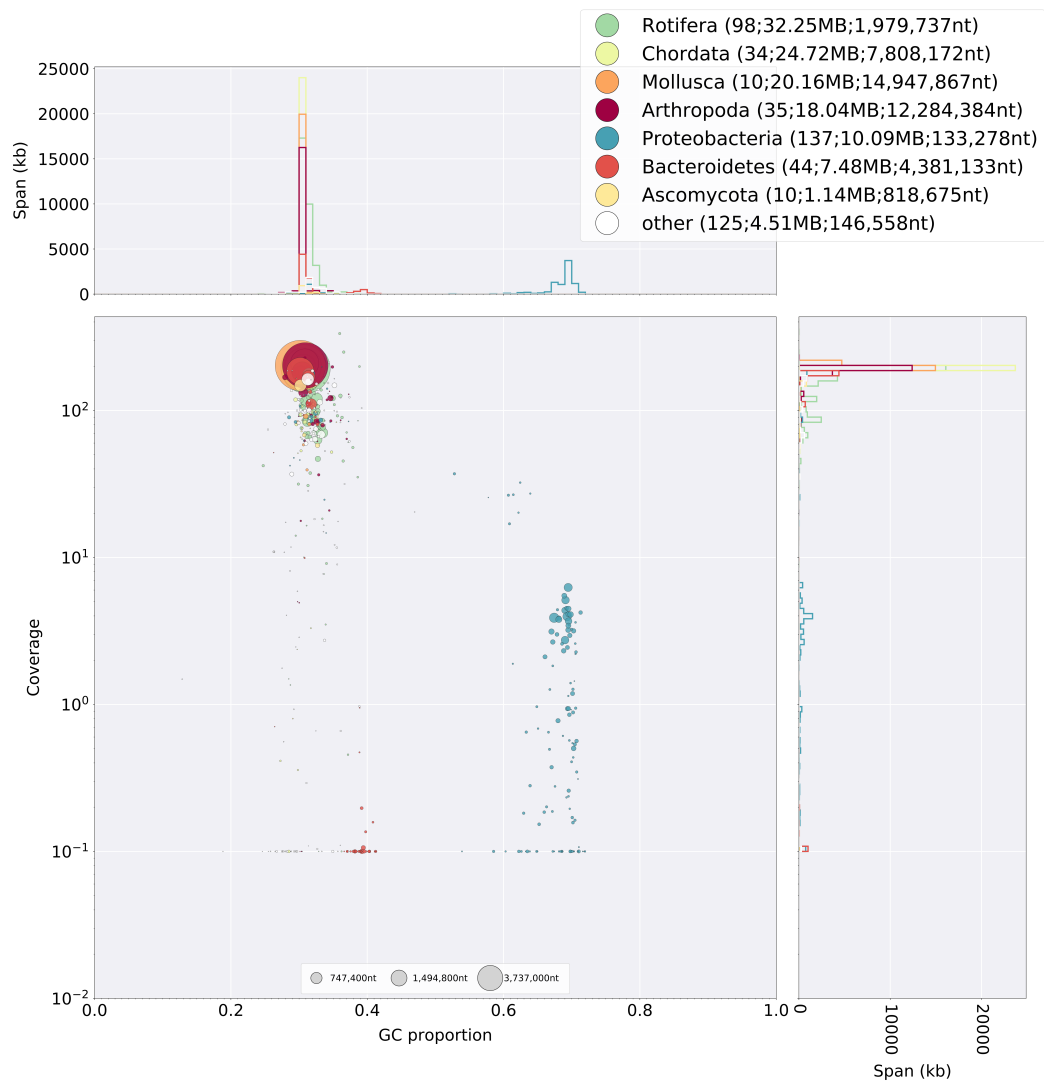


Figure S22: Blobtools v1.0 analysis of a wtdbg2 assembly of the full Nanopore dataset.

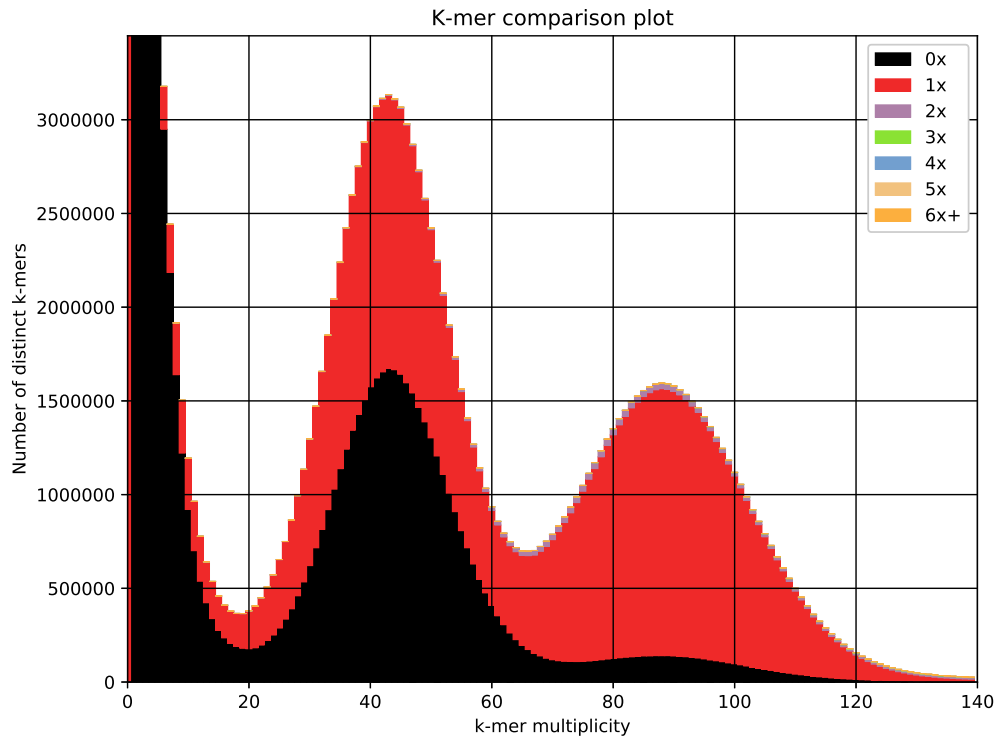


Figure S23: *k*-mer spectrum of the Shasta assembly of the full Nanopore dataset obtained with KAT v2.4.2.

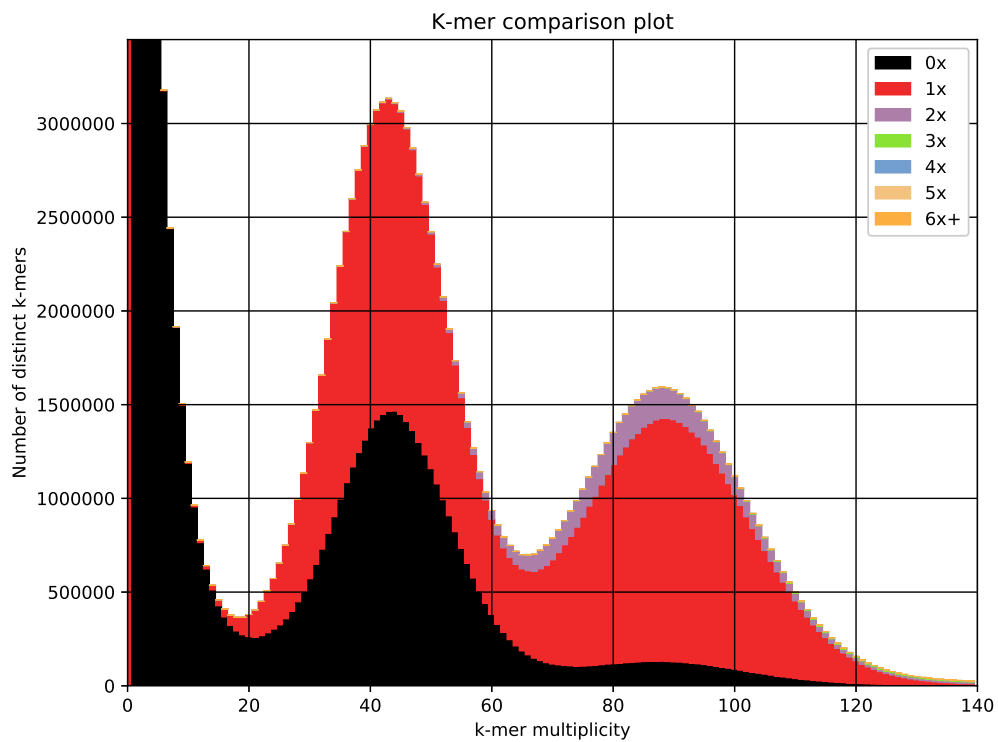


Figure S24: *k*-mer spectrum of the Shasta assembly of the longest Nanopore reads obtained with KAT v2.4.2.

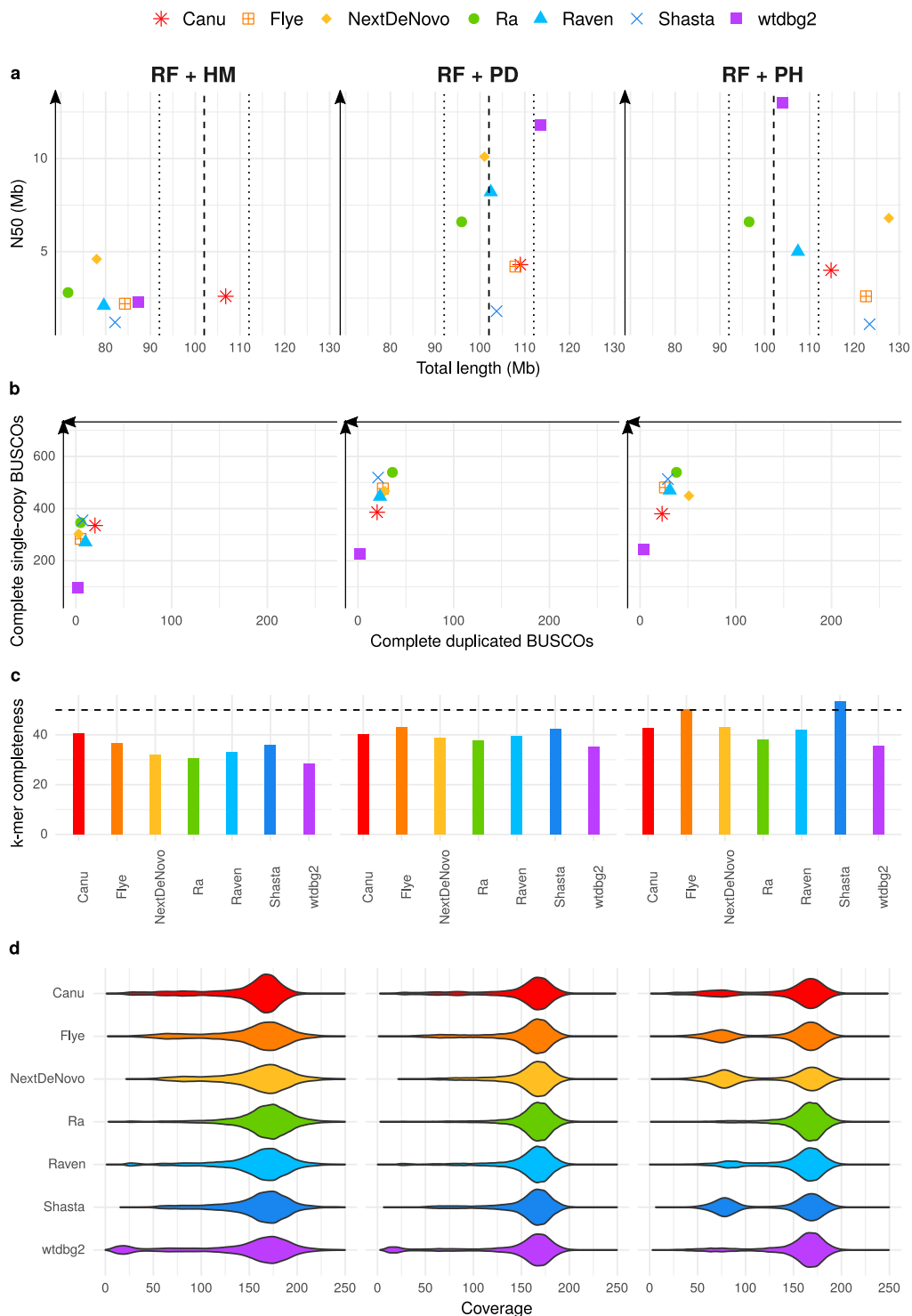


Figure S25: Statistics of Nanopore assemblies obtained from the filtered Nanopore dataset of reads longer than 30 kb, with a subsequent removal of uncollapsed haplotypes with HaploMerger2 (HM), purge_dups (PD), or purge_haplotigs (PH). a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) *k*-mer completeness. The dashed line indicates the expected 50% completeness. d) Long-read coverage distribution over the contigs.

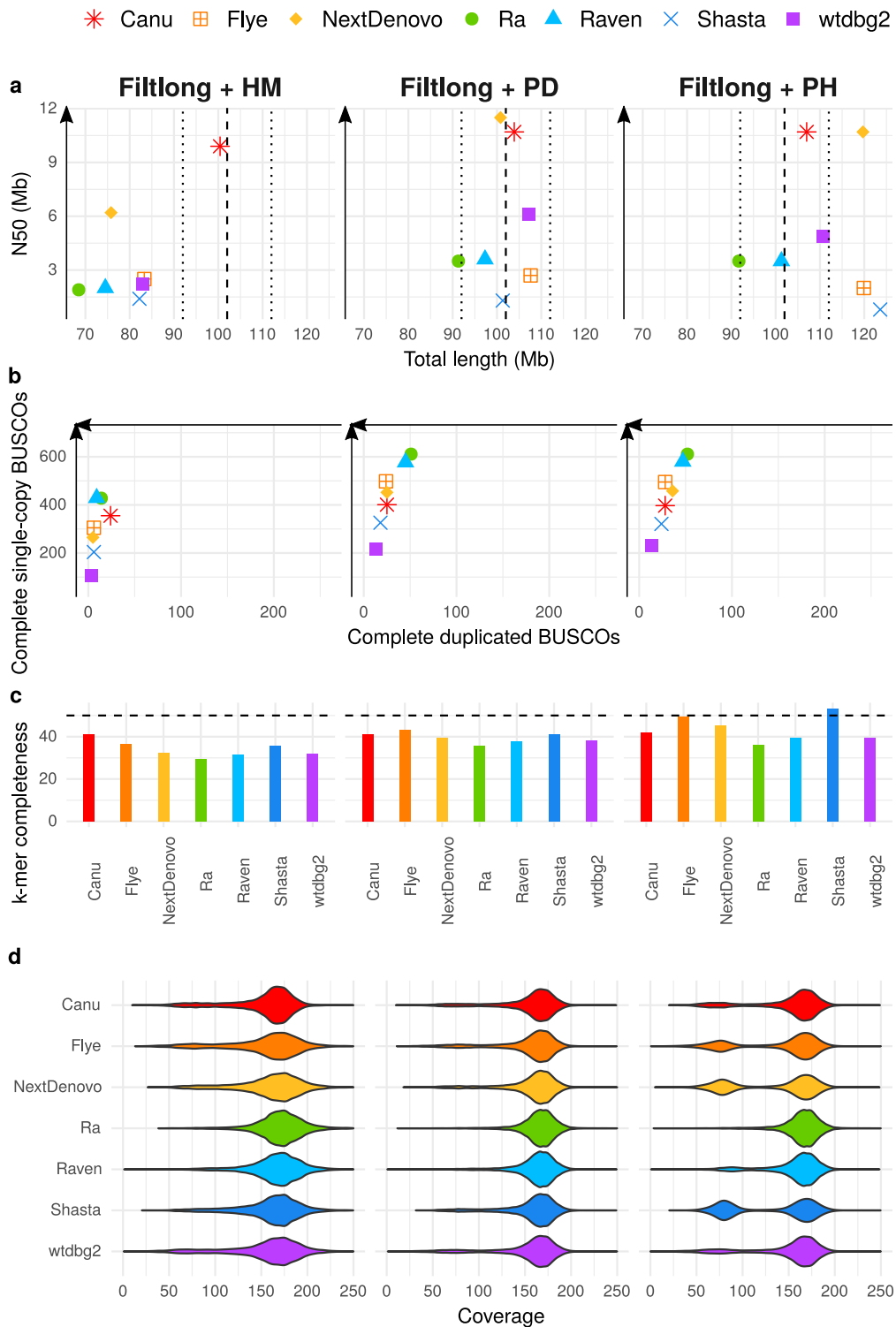


Figure S26: Statistics of Nanopore assemblies obtained from the Nanopore dataset filtered with Filtlong, with a subsequent removal of uncollapsed haplotypes with HaploMerger2 (HM), purge_dups (PD), or purge_haplotigs (PH). a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a +/- 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) k -mer completeness. The dashed line indicates the expected 50% completeness. d) Long-read coverage distribution over the contigs.

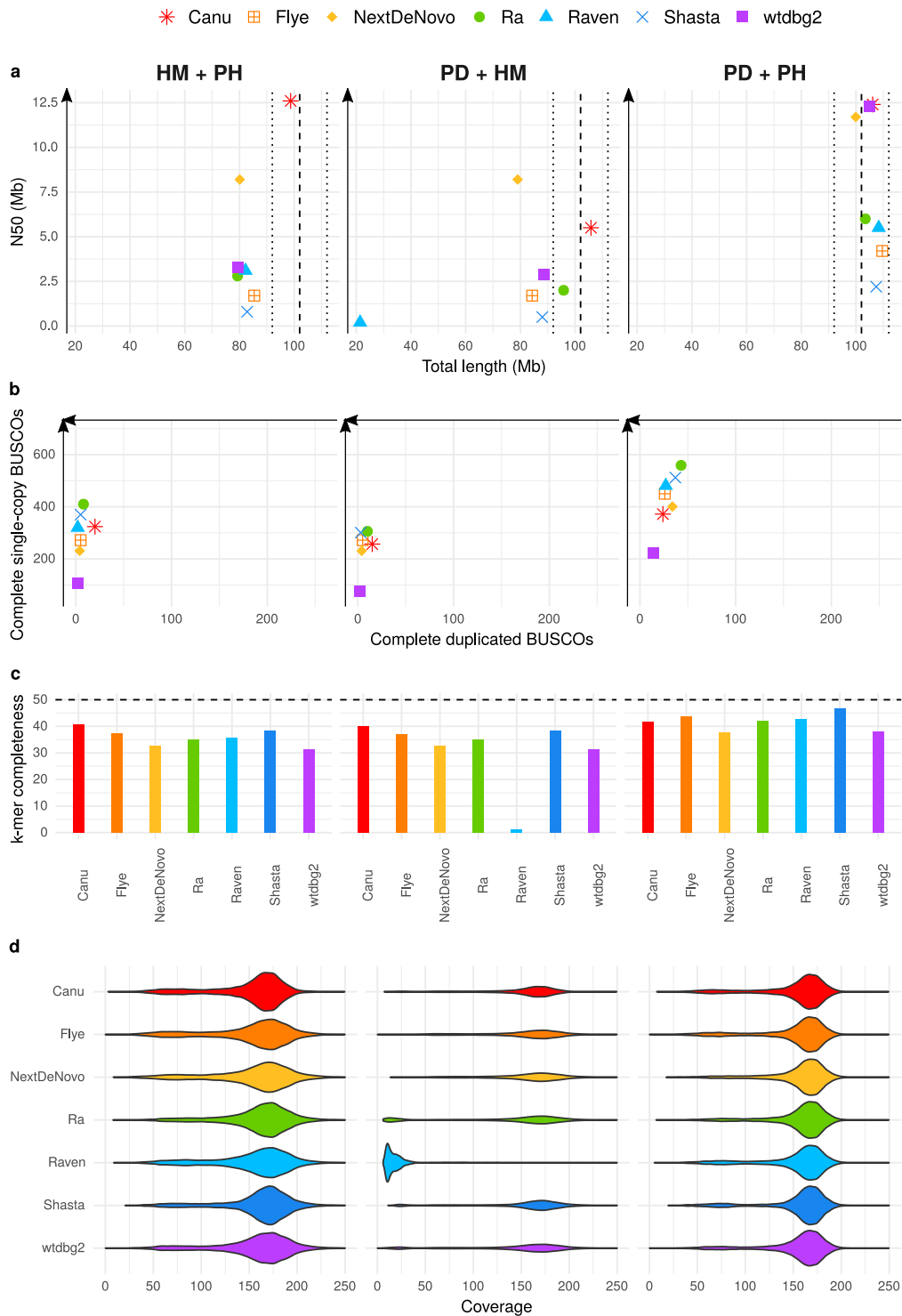


Figure S27: Statistics of Nanopore assemblies obtained from the full Nanopore dataset with a subsequent removal of uncollapsed haplotypes with combinations of HaploMerger2 (HM), purge_dups (PD), and purge_haplotigs (PH). a) N50 plotted against total assembly length. The dashed line indicates the expected genome size, with a ± 10 Mb margin delimited by the dotted lines. b) Number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, from a total of 954 orthologs. c) k -mer completeness. The dashed line indicates the expected 50% completeness. d) Long-read coverage distribution over the contigs.

Table S1: Haploidy values computed by HapPy v0.1 for PacBio assemblies.

Assembler	Processing	Haploidy
Canu	raw assemblies	0.59
Flye	raw assemblies	0.85
NextDenovo	raw assemblies	0.81
Ra	raw assemblies	0.90
Raven	raw assemblies	0.82
Shasta	raw assemblies	0.83
wtdbg2	raw assemblies	0.90
Canu	longest reads	0.62
Flye	longest reads	0.85
NextDenovo	longest reads	0.94
Ra	longest reads	0.94
Raven	longest reads	0.88
Shasta	longest reads	0.96
wtdbg2	longest reads	0.90
Canu	Filtlong	0.58
Flye	Filtlong	0.86
NextDenovo	Filtlong	0.88
Ra	Filtlong	0.94
Raven	Filtlong	0.90
Shasta	Filtlong	0.85
wtdbg2	Filtlong	0.91
Canu	HaploMerger2	0.84
Flye	HaploMerger2	0.89
NextDenovo	HaploMerger2	0.88
Ra	HaploMerger2	0.92
Raven	HaploMerger2	0.90
Shasta	HaploMerger2	0.91
wtdbg2	HaploMerger2	0.92
Canu	purge_dups	0.89
Flye	purge_dups	0.89
NextDenovo	purge_dups	0.90
Ra	purge_dups	0.91
Raven	purge_dups	0.90
Shasta	purge_dups	0.90
wtdbg2	purge_dups	0.91
Canu	purge_haplotigs	0.86
Flye	purge_haplotigs	0.85
NextDenovo	purge_haplotigs	0.87
Ra	purge_haplotigs	0.88
Raven	purge_haplotigs	0.80
Shasta	purge_haplotigs	0.90
wtdbg2	purge_haplotigs	0.90

Table S2: Haploidy values computed by HapPy v0.1 for PacBio assemblies.

Assembler	Processing	Haploidy
Canu	longest reads + purge_haplotigs	0.87
Flye	longest reads + purge_haplotigs	0.85
NextDenovo	longest reads + purge_haplotigs	0.94
Ra	longest reads + purge_haplotigs	0.92
Raven	longest reads + purge_haplotigs	0.87
Shasta	longest reads + purge_haplotigs	0.96
wtdbg2	longest reads + purge_haplotigs	0.90
Canu	longest reads + purge_dups	0.91
Flye	longest reads + purge_dups	0.90
NextDenovo	longest reads + purge_dups	0.97
Ra	longest reads + purge_dups	0.95
Raven	longest reads + purge_dups	0.91
Shasta	longest reads + purge_dups	0.97
wtdbg2	longest reads + purge_dups	0.92
Canu	Filtlong + purge_haplotigs	0.56
Flye	Filtlong + purge_haplotigs	0.86
NextDenovo	Filtlong + purge_haplotigs	0.88
Ra	Filtlong + purge_haplotigs	0.93
Raven	Filtlong + purge_haplotigs	0.90
Shasta	Filtlong + purge_haplotigs	0.85
wtdbg2	Filtlong + purge_haplotigs	0.94
Canu	Filtlong + purge_dups	0.90
Flye	Filtlong + purge_dups	0.90
NextDenovo	Filtlong + purge_dups	0.93
Ra	Filtlong + purge_dups	0.94
Raven	Filtlong + purge_dups	0.92
Shasta	Filtlong + purge_dups	0.92
wtdbg2	Filtlong + purge_dups	0.91

Table S3: Haploidy values computed by HapPy v0.1 for PacBio assemblies.

Assembler	Processing	Haploidy
Canu	HaploMerger2 + purge_haplotigs	0.82
Flye	HaploMerger2 + purge_haplotigs	0.89
NextDenovo	HaploMerger2 + purge_haplotigs	0.88
Ra	HaploMerger2 + purge_haplotigs	0.88
Raven	HaploMerger2 + purge_haplotigs	0.83
Shasta	HaploMerger2 + purge_haplotigs	0.88
wtdbg2	HaploMerger2 + purge_haplotigs	0.84
Canu	purge_dups + HaploMerger2	0.91
Flye	purge_dups + HaploMerger2	0.90
NextDenovo	purge_dups + HaploMerger2	0.90
Ra	purge_dups + HaploMerger2	0.92
Raven	purge_dups + HaploMerger2	0.93
Shasta	purge_dups + HaploMerger2	0.92
wtdbg2	purge_dups + HaploMerger2	0.92
Canu	purge_dups + purge_haplotigs	0.88
Flye	purge_dups + purge_haplotigs	0.89
NextDenovo	purge_dups + purge_haplotigs	0.92
Ra	purge_dups + purge_haplotigs	0.89
Raven	purge_dups + purge_haplotigs	0.88
Shasta	purge_dups + purge_haplotigs	0.90
wtdbg2	purge_dups + purge_haplotigs	0.91

Table S4: Haploidy values computed by HapPy v0.1 for Nanopore assemblies.

Assembler	Processing	Haploidy
Canu	raw assemblies	0.63
Flye	raw assemblies	0.79
NextDenovo	raw assemblies	0.72
Ra	raw assemblies	0.90
Raven	raw assemblies	0.83
Shasta	raw assemblies	0.86
wtdbg2	raw assemblies	0.92
Canu	longest reads	0.59
Flye	longest reads	0.79
NextDenovo	longest reads	0.72
Ra	longest reads	0.95
Raven	longest reads	0.89
Shasta	longest reads	0.75
wtdbg2	longest reads	0.92
Canu	Filtlong	0.67
Flye	Filtlong	0.81
NextDenovo	Filtlong	0.77
Ra	Filtlong	0.97
Raven	Filtlong	0.92
Shasta	Filtlong	0.72
wtdbg2	Filtlong	0.87
Canu	HaploMerger2	0.89
Flye	HaploMerger2	0.87
NextDenovo	HaploMerger2	0.89
Ra	HaploMerger2	0.91
Raven	HaploMerger2	0.88
Shasta	HaploMerger2	0.90
wtdbg2	HaploMerger2	0.89
Canu	purge_dups	0.92
Flye	purge_dups	0.90
NextDenovo	purge_dups	0.92
Ra	purge_dups	0.93
Raven	purge_dups	0.90
Shasta	purge_dups	0.91
wtdbg2	purge_dups	0.93
Canu	purge_haplotigs	0.86
Flye	purge_haplotigs	0.79
NextDenovo	purge_haplotigs	0.90
Ra	purge_haplotigs	0.90
Raven	purge_haplotigs	0.83
Shasta	purge_haplotigs	0.86
wtdbg2	purge_haplotigs	0.91

Table S5: Haploidy values computed by HapPy v0.1 for Nanopore assemblies.

Assembler	Processing	Haploidy
Canu	longest reads + purge_haplotigs	0.85
Flye	longest reads + purge_haplotigs	0.79
NextDenovo	longest reads + purge_haplotigs	0.72
Ra	longest reads + purge_haplotigs	0.95
Raven	longest reads + purge_haplotigs	0.89
Shasta	longest reads + purge_haplotigs	0.75
wtdbg2	longest reads + purge_haplotigs	0.91
Canu	longest reads + purge_dups	0.89
Flye	longest reads + purge_dups	0.91
NextDenovo	longest reads + purge_dups	0.95
Ra	longest reads + purge_dups	0.96
Raven	longest reads + purge_dups	0.95
Shasta	longest reads + purge_dups	0.93
wtdbg2	longest reads + purge_dups	0.92
Canu	Filtlong + purge_haplotigs	0.90
Flye	Filtlong + purge_haplotigs	0.81
NextDenovo	Filtlong + purge_haplotigs	0.77
Ra	Filtlong + purge_haplotigs	0.97
Raven	Filtlong + purge_haplotigs	0.92
Shasta	Filtlong + purge_haplotigs	0.72
wtdbg2	Filtlong + purge_haplotigs	0.89
Canu	Filtlong + purge_dups	0.93
Flye	Filtlong + purge_dups	0.91
NextDenovo	Filtlong + purge_dups	0.94
Ra	Filtlong + purge_dups	0.97
Raven	Filtlong + purge_dups	0.96
Shasta	Filtlong + purge_dups	0.94
wtdbg2	Filtlong + purge_dups	0.91

Table S6: Haploidy values computed by HapPy v0.1 for Nanopore assemblies.

Assembler	Processing	Haploidy
Canu	HaploMerger2 + purge_haplotigs	0.89
Flye	HaploMerger2 + purge_haplotigs	0.87
NextDenovo	HaploMerger2 + purge_haplotigs	0.89
Ra	HaploMerger2 + purge_haplotigs	0.91
Raven	HaploMerger2 + purge_haplotigs	0.92
Shasta	HaploMerger2 + purge_haplotigs	0.90
wtdbg2	HaploMerger2 + purge_haplotigs	0.90
Canu	purge_dups + purge_haplotigs	0.91
Flye	purge_dups + purge_haplotigs	0.90
NextDenovo	purge_dups + purge_haplotigs	0.94
Ra	purge_dups + purge_haplotigs	0.93
Raven	purge_dups + purge_haplotigs	0.90
Shasta	purge_dups + purge_haplotigs	0.91
wtdbg2	purge_dups + purge_haplotigs	0.92
Canu	purge_dups + HaploMerger2	0.90
Flye	purge_dups + HaploMerger2	0.88
NextDenovo	purge_dups + HaploMerger2	0.90
Ra	purge_dups + HaploMerger2	0.91
Raven	purge_dups + HaploMerger2	0.51
Shasta	purge_dups + HaploMerger2	0.90
wtdbg2	purge_dups + HaploMerger2	0.89

Table S7: List of command lines used for each tool. Values L, M, H for `purge_haplotigs cov` were selected for each assembly according to the histogram produced by `purge_haplotigs hist`.

Program	Dataset	Command lines
Filllong	-	<code>filllong --target_bases 4092000000 --mean_q_weight 10 long_read_data</code>
Canu	PacBio	<code>canu -d out -p out genomeSize=100m useGrid=false -pacbio-raw pb_data</code>
Canu	Nanopore	<code>canu -d out -p out genomeSize=100m useGrid=false -nanopore-raw ont_data</code>
Flye	PacBio	<code>flye -o out -g 100m --pacbio-raw pb_data</code>
Flye	Nanopore	<code>flye -o out -g 100m --nano-raw ont_data</code>
NextDenovo	PacBio	<code>echo pb_data > input.fofn seq_stat input.fofn -g 100Mb -d 150 > stats.txt NextDenovo run.cfg</code>
NextDenovo	Nanopore	<code>echo ont_data > input.fofn seq_stat input.fofn -g 100Mb -d 150 > stats.txt NextDenovo run.cfg</code>
Ra	PacBio	<code>ra -x pb pb_data > assembly.fasta</code>
Ra	Nanopore	<code>ra -x ont ont_data > assembly.fasta</code>
Raven	-	<code>raven long_read_data > assembly.fasta</code>
Shasta	PacBio	<code>shasta --input pb_data --Reads.minReadLength 0 --assemblyDirectory out --Assembly.consensusCaller Modal --Kmers.k 12</code>
Shasta	Nanopore	<code>shasta --input ont_data --Reads.minReadLength 0 --assemblyDirectory out</code>
wtdbg2	PacBio	<code>wtdbg2 -x rs -g 100m -i pb_data -fo out wtpoa-cns -i out.ctg.lay.gz -o out.ctg.fa minimap2 -x map-pb -a out.ctg.fa pb_data samtools sort > out.ctg.bam samtools view out.ctg.bam wtpoa-cns -d out.ctg.fa -i - -fo assembly.fasta</code>
wtdbg2	Nanopore	<code>wtdbg2 -x ont -g 100m -i ont_data -fo out wtpoa-cns -i out.ctg.lay.gz -o out.ctg.fa minimap2 -x map-ont -a out.ctg.fa ont_data samtools sort > out.ctg.bam samtools view out.ctg.bam wtpoa-cns -d out.ctg.fa -i - -fo assembly.fasta</code>
HaploMerger2	-	<code>samtools faidx assembly.fasta BuildDatabase -name asm.db -engine ncbi assembly.fasta RepeatModeler -engine ncbi -database asm.db RepeatMasker -e ncbi -lib consensi.fa -xsmall assembly.fasta run_all.batch</code>
purge_dups	PacBio	<code>echo pb_data > input.fofn pd_config.py assembly.fasta input.fofn run_purge_dups.py config.json purge_dups_bin species_id</code>
purge_dups	Nanopore	<code>echo ont_data > input.fofn pd_config.py assembly.fasta input.fofn run_purge_dups.py config.json purge_dups_bin species_id</code>
purge_haplotigs	PacBio	<code>minimap2 -ax map-pb assembly.fasta pb_data --secondary=no > aligned.bam samtools sort -o ali.sorted.bam -T tmp.ali aligned.bam samtools index ali.sorted.bam samtools faidx assembly.fasta purge_haplotigs hist -b ali.sorted.bam -g assembly.fasta purge_haplotigs cov -i ali.sorted.bam -l L -m M -h H -o cov_stats.csv purge_haplotigs purge -g assembly.fasta -c cov_stats.csv -o assembly.purged.fasta</code>
purge_haplotigs	Nanopore	<code>minimap2 -ax map-ont assembly.fasta ont_data --secondary=no > aligned.bam samtools sort -o ali.sorted.bam -T tmp.ali aligned.bam samtools index ali.sorted.bam samtools faidx assembly.fasta purge_haplotigs hist -b ali.sorted.bam -g assembly.fasta purge_haplotigs cov -i ali.sorted.bam -l L -m M -h H -o cov_stats.csv purge_haplotigs purge -g assembly.fasta -c cov_stats.csv -o assembly.purged.fasta</code>
BBtools	-	<code>reformat.sh in=long_reads_data out=subset_data samplebasestarget=number_of_bases</code>
BUSCO	-	<code>busco -i assembly.fasta -o busco_output -l metazoa_odb10 -m genome</code>
KAT	Illumina	<code>kat comp -o kat_output 'end1.fastq end2.fastq' assembly.fasta</code>
tinycov	Nanopore	<code>minimap2 -x map-ont -a assembly.fasta ont_data samtools sort > aligned.bam tinycov covplot -r 20000 -t cov.txt aligned.bam</code>
tinycov	PacBio	<code>minimap2 -x map-pb -a assembly.fasta pb_data samtools sort > aligned.bam tinycov covplot -r 20000 -t cov.txt aligned.bam</code>
HapPy	Nanopore	<code>minimap2 -x map-ont -a assembly.fasta ont_data samtools sort > aligned.bam HapPy.py depth aligned.bam out_dir HapPy.py estimate out_dir/aligned.bam.hist</code>
HapPy	PacBio	<code>minimap2 -x map-pb -a assembly.fasta pb_data samtools sort > aligned.bam HapPy.py depth aligned.bam out_dir HapPy.py estimate out_dir/aligned.bam.hist</code>
time	-	<code>/usr/bin/time -v -o time_output.txt</code>

Table S8: Long-read and short-read datasets used in the study.

Data type	Minimum length	Total data	N50
PacBio	-	23.5 Gb	11.6 kb
	15 kb	4.7 Gb	17.6 kb
Nanopore	-	17.5 Gb	18.8 kb
	30 kb	5.7 Gb	51.8 kb
Illumina 2*250 bp	30 bp	11.4 Gb	250 bp

Chapter 3

Unzipping assembly graphs with long reads and Hi-C

This chapter is a paper in preparation with Roland Faure (co-first author) and Jean-François Flot.

3.1 Introduction

The field of genomics is thriving and chromosome-level assemblies are now commonly achieved for all types of organisms, thanks to the combined improvements of sequencing and assembly methods. Chromosome-level assemblies are generally haploid, regardless of the ploidy of the genome. To obtain a haploid assembly of a multiploid (i.e. diploid or polyploid) genome, homologous chromosomes are collapsed into one sequence. However, assemblers often struggle to collapse highly heterozygous regions, which leads to breaks in the assembly and duplicated regions [200]. Furthermore, haploid assemblies provide a partial representation of multiploid genomes: ideally, multiploid genomes should be phased rather than collapsed if the aim is to grasp their whole complexity [247].

The combination of low-accuracy long reads, such as Oxford Nanopore Technologies (ONT) reads and Pacific Biosciences (PacBio) Continuous Long Reads (CLRs), with proximity ligation (Hi-C) reads has made chromosome-level assemblies accessible for all types of organisms. The latest development of PacBio, high-accuracy long circular consensus sequencing (CCS) reads (a.k.a. HiFi), is now starting to deliver highly contiguous phased assemblies [97, 110, 109]. Hi-C scaffolding is commonly used in genome assem-

bly projects to obtain chromosome-level scaffolds. This approach relies on the interaction frequency in the genome and these interactions are heightened between closer loci belonging to the same chromosome [219]. Based on this principle, alleles can be associated using their interaction frequencies.

A first approach to phase assemblies is called trio-binning and uses sequencing data from the individual and its parents to retrieve haplotypes [248]; yet this method is unavailable when the parents cannot be identified, or for asexual species. Existing tools are able to use either long reads (Falcon-Unzip [96], WhatsHap [250]) or Hi-C reads (Falcon-Phase [258], ALLHiC [257]) for phasing assemblies, but they are limited to phasing local variants or well-identified haplotypes and are not suited for complex, highly heterozygous genomes. WhatsHap takes as input a collapsed assembly and searches for alternative haplotypes. As collapsing haplotypes can be too difficult for highly heterozygous regions, it seems more intuitive to phase these assemblies *de novo*. FALCON-Unzip and FALCON-Phase offer this alternative, yet they are dependant on the output of the FALCON assembler and cannot be combined with other assemblers.

We present GraphUnzip, a new tool to phase assemblies using long reads and/or Hi-C. GraphUnzip implements a radically new approach to phasing that starts from an assembly graph instead of a set of linear sequences. In an assembly graph, heterozygous regions result in bubbles every time the assembler is unable to collapse the haplotypes or to choose one of them. GraphUnzip "unzips" the graph, meaning that it separates the haplotypes by duplicating homozygous regions that have been collapsed and partitioning heterozygous regions into haplotypes. This tool is based on a simple principle that was implemented in many scaffolders since SSPACE [160]: long-range data (mate-pair reads, long reads, linked reads, proximity ligation...) provide information on the linkage between contigs that can be used to group and orient them into scaffolds. As GraphUnzip takes as input and produces as output assembly graphs, it only connects contigs that are actually adjacent in the genome and yields gap-less scaffolds, i.e. supercontigs. GraphUnzip is compatible with any assembler that produces an assembly graph. We tested GraphUnzip on the genomes of the human HG00733 and the potato *Solanum tuberosum*. GraphUnzip is available at github.com/nadegeguiglielmoni/GraphUnzip.

3.2 Methods

3.2.1 Inputs

GraphUnzip requires an assembly graph in GFA (Graphical Fragment Assembly) format. The Hi-C input is a sparse matrix, such as the one obtained when processing the reads with hicstuff [260]. hicstuff also provides a module to convert other file formats (e.g. cool, a common Hi-C format) to a sparse matrix. The long reads are mapped to the assembly graph using GraphAligner [261].

3.2.2 Overview of GraphUnzip

In an assembly graph, contigs that are inferred to be adjacent or to overlap in the assembly are connected with edges. However, some of these connections between contigs may be artefacts. To discriminate correct edges from erroneous ones, GraphUnzip relies on long reads and/or Hi-C data. These data are translated into interactions between contigs: the strength of interaction between two contigs is defined as the number of long reads bridging both contigs when using long reads as input; and as the number of Hi-C contacts between the two contigs when using Hi-C as input. In both cases, a strong interaction is a sign of proximity on the genome.

GraphUnzip first builds one or two interaction matrices containing all pairwise interactions between contigs, depending on whether long-read data, Hi-C data or both are provided (Figure 3.1). In the next step, GraphUnzip iteratively reviews all contigs and their edges. The strength of an edge i is computed based on the strength of interaction between the contigs it connects. A high strength supports the reality of the link, while a low strength may signal an artefactual edge. When a contig has several edges at one of its extremities, these edges are compared in a pairwise fashion. This comparison uses two user-provided thresholds: the rejection threshold T_R and the acceptance threshold T_A , where $T_R < T_A$. Considering two edges X and Y and their respective strengths $i(X)$ and $i(Y)$, if $i(X) < i(Y)$, Y is considered strong; if $i(X)/i(Y) < T_R$, then X is considered weak, else, if $T_R \leq i(X)/i(Y) < T_A$, X is flagged as dubious. X is labelled as strong when $i(X)/i(Y) \geq T_A$. The algorithm thereafter considers weak edges as artefacts that do not actually exist in the genome, whereas strong edges represent true connections. If both long reads and Hi-C input data are provided, strengths based on long reads are used first because they are more reliable locally, and strengths based on Hi-C are only used if some edges are flagged as dubious.

Edges identified as weak in the previous calculation are removed. Then, every contig that has more than one strong edge and no dubious edge at one end is duplicated as many times as the number of these strong edges. Such contigs are typically collapsed homozygous regions that need to be present in several copies to be included in every haplotypes. All the copies retain the edges of the original contig at its other end. This entails that the duplication of contigs creates many new (and potentially artefactual) edges. Contigs that are unambiguously linked are merged in supercontigs that will be handled as regular contigs thereafter.

When assessing the strength of two putative edges (S_1, S_2) and (S_1, S_3) connecting the supercontigs S_1 , S_2 , and S_3 , the strength of these edges are calculated as the strength of interaction between contigs in S_1 and contigs present in S_2 but not in S_3 (and vice versa). For example, in the third step of Figure 3.1, when trying to associate supercontig a-b to either d-e or d'-f, only the interactions between the supercontig a-b and the contigs e and f are considered. Interactions between the supercontig a-b and the contigs d and d' are not considered in the calculation because d and d' actually originate from the duplication of a collapsed region.

All contigs and edges are iteratively processed s times to phase the assembly, where s is a user-provided parameter. Because extremely long contigs tend to share a significant number of Hi-C contacts even if they are not adjacent, we observed that in extreme cases the algorithm could join two chromosomes by their telomeric ends. The Hi-C matrix is used at the end of the process to detect such chimeric connections in the assembly graph, based on low Hi-C interactions, and break them.

3.2.3 *Homo sapiens* HG00733 assemblies

We used HiFi, ONT and Hi-C reads from [253]. HiFi reads were assembled using hifiasm with the parameter `-l 0`, and the resulting `p_utg` assembly graph was used for downstream analyses. All HiFi reads and the ONT reads longer than 30 kb were mapped to the assembly using GraphAligner with the parameter `-x vg`. Hi-C reads were processed with hicstuff using the parameters `--aligner bowtie2 --enzyme 200 --iterative`. GraphUnzip was run with parameters `-accept 0.10 -reject 0.05 --exhaustive --whole_match --minimum_match 0.8`. All non-ambiguous paths in the GFA were merged using Bandage. The assemblies were compared to the DipAsm reference [262] using QUASt v5.0.2 [240] with the parameters `-m 0 -eukaryote -large -min-identity 99.9`.

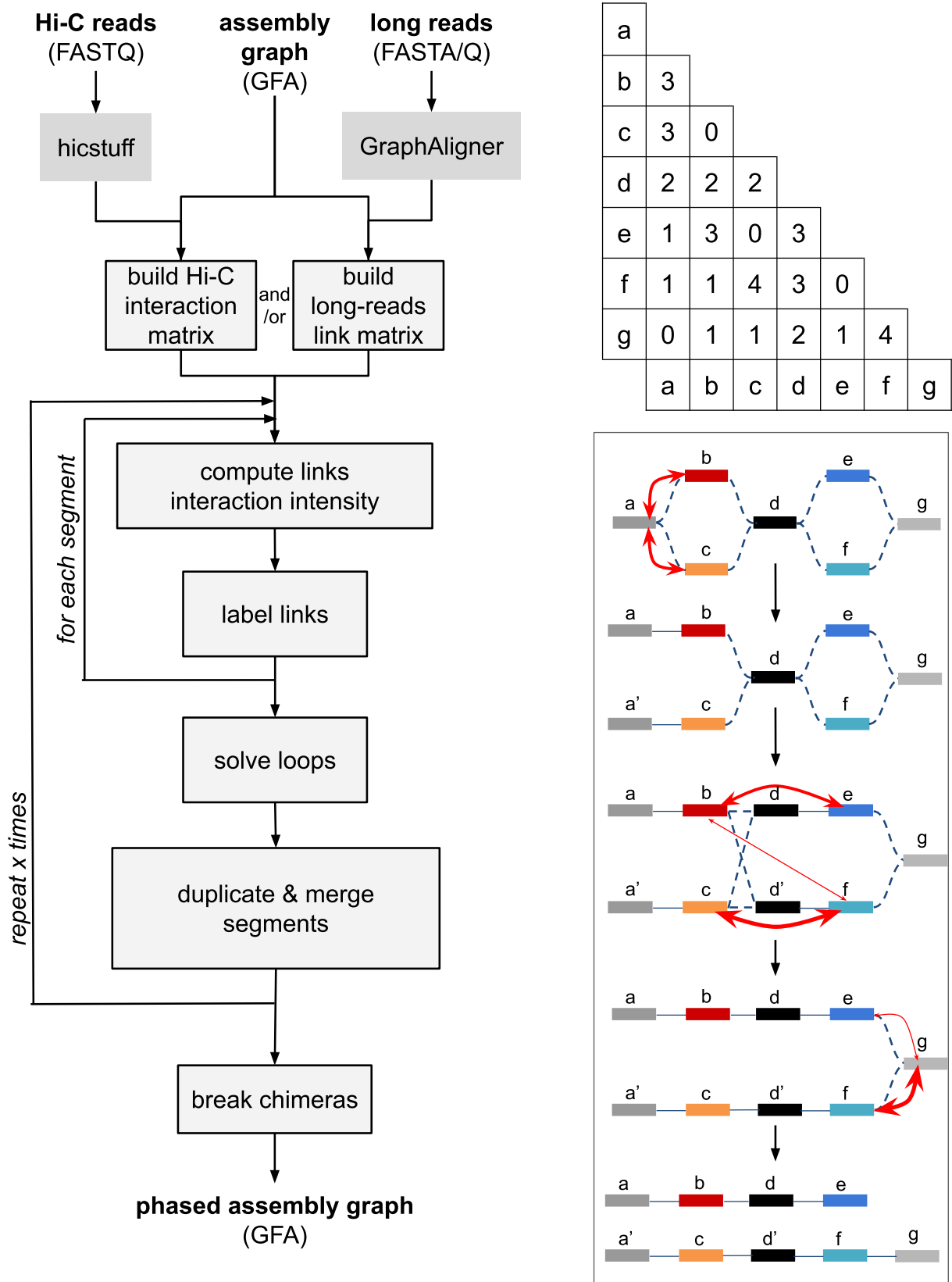


Figure 3.1: Description of GraphUnzip: workflow of the program (left), interaction matrix (top right), and overview of the algorithm to discriminate links (bottom right). This example algorithm analyzes the potential links between the segments a, b, c, d, e, f, g. The red arrows represent the intensity of interactions between the segments, computed based on the values in the matrix.

3.2.4 *Solanum tuberosum* assemblies

HiFi, ONT and Hi-C reads published in [254] were retrieved from the NCBI Sequence Read Archive with the Bioproject accession number PRJNA573826. The HiFi reads were assembled using hifiasm with the parameter `-l 0`, and the `p_utg` assembly graph was used for downstream analyses. All HiFi reads and the ONT reads longer than 25 kb were mapped to the assembly using GraphAligner with the parameter `-x vg`. Hi-C reads were processed with hicstuff using the parameters `--aligner bowtie2 --enzyme MboI --iterative`. GraphUnzip was run with parameters `-accept 0.40 -reject 0.10 --exhaustive --whole_match --minimum_match 0.8`. All non-ambiguous paths in the GFA were merged using Bandage. To check the output of GraphUnzip, we mapped the published assembly to the assembly graph using GraphAligner. We used calN50 (available at github.com/lh3/calN50) to compute the NG50 against the published assembly size of 1.67 Gb [254]. BUSCO v4 [30] was run with parameters `-m genome -long` against the dataset `viridiplantae odb10`.

3.2.5 Computational performance

RAM usage and CPU time were measured with the command `/usr/bin/time -v` on a desktop computer with 128 GB of RAM and a i9-9900X 3.5 GHz processor.

3.3 Results

3.3.1 *Homo sapiens* HG00733

Table 3.1: Assembly metrics of *Homo sapiens* HG00733 compared with the DipAsm reference.

Assembly	GraphUnzip	Size	N50	NA50	Misassemblies	CPU	RAM
Reference	-	5.9 Gb	27.8 Mb	27.8 Mb	84	-	-
hifiasm	-	5.5 Gb	397 kb	343 kb	9146	-	-
	ONT + Hi-C	6.2 Gb	1.5 Mb	1.2 Mb	8091	33min 46s	23.5 GB

We compared the hifiasm + GraphUnzip assembly of the human HG00733 genome with a published reference obtained using DipAsm, based on the N50, the NA50 and the number of misassemblies. The N50 represents the contiguity of the assembly: it is defined as the length of the largest contig for which 50% of the assembly size is contained in contigs of equal or greater length. The NA50 is the N50 of the

assembly broken at every misassembly (compared to a reference). GraphUnzip increased the size of the hifiasm assembly (from 5.5 Gb to 6.2 Gb), and the N50 rose as well (from 397 kb to 1.2 Mb) (Table 3.1). The NA50 was improved while the number of misassemblies decreased in the GraphUnzip supercontigs. Notably, the reference assembly size is only 5.9 Gb, while the GraphUnzip assembly reaches 6.2 Gb, which is the expected size for a phased human genome.

We also tried an assembly of the HiFi reads with Flye, but the draft assembly was only 2.9 Gb, little below half the expected size, which indicates that the haplotypes were nearly completely collapsed. A good candidate assembly for GraphUnzip should have uncollapsed heterozygous regions, as GraphUnzip is not able to retrieve a missing haplotype in collapsed heterozygous regions and can only duplicate the collapsed region, leading in that case to a suboptimal result.

3.3.2 *Solanum tuberosum*

Table 3.2: Assembly metrics of *Solanum tuberosum*. The NG50 values were computed based on an estimated genome size of 1.67 Gb.

Assembly	GraphUnzip	Size	NG50	BUSCO		CPU	RAM
				Single	Dup.		
Reference	-	1.67 Gb	66.1 Mb	21.6%	76.9%	-	-
hifiasm	-	1.51 Gb	2.2 Mb	21.2%	77.9%	-	-
	HiFi	1.69 Gb	3.7 Mb	7.1%	91.5%	16s	0.2 GB
	ONT	1.67 Gb	3.4 Mb	6.8%	92.2%	52s	0.2 GB
	Hi-C	1.69 Gb	5.6 Mb	7.8%	91.5%	38min 27s	11.5 GB
	HiFi + Hi-C	1.69 Gb	4.9 Mb	9.4%	89.4%	39min 59s	11.5 GB
	ONT + Hi-C	1.73 Gb	5.9 Mb	7.3%	91.8%	39min 10s	11.5 GB

We tested GraphUnzip on the diploid genome of the potato *Solanum tuberosum* RH89-039-16, for which a phased assembly of 1.67 Gb [254] was recently published. We assembled the HiFi reads with hifiasm and then ran GraphUnzip using the HiFi, ONT and/or Hi-C reads. The draft assembly was 1.51 Gb, and after phasing with GraphUnzip, the assembly size rose to 1.67-1.73 Gb (Table 3.2). In this case, we compared the NG50s, a value similar to N50 but based on a reference genome size rather than the assembly size. GraphUnzip increased the contiguity: from 2.2 Mb, the NG50 reached 3.4 to 5.9 Mb. The combination of both ONT and Hi-C reads yielded the highest NG50. Hi-C reads improved the contiguity better than long reads. The overall BUSCO completeness of the GraphUnzip supercontigs was slightly improved compared to the reference: 98.6-99.3% against 98.5% for the reference, and the number of duplicated BUSCO features was higher as well (89.4-92.2% against 76.9%). We mapped the published assembly to the GraphUnzip assembly graph obtained when using Hi-C and ONT reads. We found that

there were no differences in phasing between the two assemblies. However, some regions that were phased by hifiasm and GraphUnzip were collapsed in the published assembly. This result, in conjunction with the higher number of duplicated features, indicates that GraphUnzip led to an improved phased assembly.

3.3.3 Computational performance

For both the human and *Solanum tuberosum* genomes, GraphUnzip required limited computational resources as it ran in less than 1 hour on a single thread and used up to 23.5 GB of memory. For *Solanum tuberosum*, the run time was also shorter when using only long reads (less than a minute). The longer run time when using Hi-C reads was due to the building of the interaction matrix. As this interaction matrix is outputted by the program, this file can be reused for other runs, which will consequently finish faster. Therefore, users can try several sets of parameters to optimize the result, with short runtimes.

3.4 Conclusion

GraphUnzip is a flexible tool that can phase assemblies of high-accuracy long reads with long reads and/or Hi-C. A limitation of GraphUnzip is that it does not necessarily reach chromosome-level assemblies like most Hi-C scaffolders, but it aims instead to produce more contiguous gap-less supercontigs by fully exploiting assembly graphs. As genome projects now usually include long reads and Hi-C to obtain chromosome-level assemblies, GraphUnzip can easily be integrated in assembly projects after assembly to obtain *de novo* phased assemblies for non-model organisms.

Chapter 4

Scaffolding assemblies with Hi-C

Recent genome assembly projects generally aim to achieve chromosome-level scaffolds, and Hi-C scaffolding is a major step to reach this goal. This method has been included successfully in many studies for bacteria, yeasts, plants, animals, and is part of assembly pipelines for several consortia, such as the Vertebrate Genome Project [263] and the Darwin Tree of Life [35]. instaGRAAL is an overhauled, improved version of GRAAL [184], a Hi-C scaffolder inspired by Gibbs sampling, a MCMC-based approach, to iteratively test and ponder arrangements of DNA fragments until the resulting organization converges towards an assembly with a higher likelihood based on contact frequencies. Two main aspects were improved: first, instaGRAAL, through the use of sparse contact maps, is more computationally efficient, enabling it to handle larger genomes (over 1 Gb); second, it introduces a module to automatically refine the scaffolds based on the input contigs, and reduce local misassemblies. In the following paper, instaGRAAL was used to assemble the brown algae *Desmarestia herbacea* and *Ectocarpus* sp., and benchmarked against SALSA2 [189] on a human; it systematically yielded chromosome-level scaffolds.


I contributed to testing instaGRAAL, in particular regarding its application to the human genome. I also improved the documentation to make it more accessible for new users, and contribute since then to the maintenance of the program on the github account of Romain Koszul's lab github.com/koszullab/instaGRAAL.

SOFTWARE

Open Access



instaGRAAL: chromosome-level quality scaffolding of genomes using a proximity ligation-based scaffolder

Lyam Baudry^{1,2}, Nadège Guiguelmoni^{1,3}, Hervé Marie-Nelly^{1,2}, Alexandre Cormier⁴, Martial Marbouty¹, Komlan Avia^{4,5}, Yann Loe Mie⁶, Olivier Godfroy⁴, Lieven Sterck^{7,8}, J. Mark Cock⁴, Christophe Zimmer⁹, Susana M. Coelho^{4*} and Romain Koszul^{1*} 

* Correspondence: coelho@sb-roscoff.fr; romain.koszul@pasteur.fr

⁴Sorbonne Université, Laboratory of Integrative Biology of Marine Models, Algal Genetics, UMR 8227, Roscoff, France

¹Institut Pasteur, Unité Régulation Spatiale des Génomes, CNRS, UMR 3525, C3BI USR 3756, F-75015 Paris, France

Full list of author information is available at the end of the article

Abstract

Hi-C exploits contact frequencies between pairs of loci to bridge and order contigs during genome assembly, resulting in chromosome-level assemblies. Because few robust programs are available for this type of data, we developed instaGRAAL, a complete overhaul of the GRAAL program, which has adapted the latter to allow efficient assembly of large genomes. instaGRAAL features a number of improvements over GRAAL, including a modular correction approach that optionally integrates independent data. We validate the program using data for two brown algae, and human, to generate near-complete assemblies with minimal human intervention.

Keywords: *Ectocarpus*, Hi-C scaffolding, Hi-C, genome assembly, MCMC, GPU, *Desmarestia herbacea*

Background

Continuous developments in DNA sequencing technologies aim at alleviating the technical challenges that limit the ability to assemble sequence data into full-length chromosomes [1–3]. Conventional assembly programs and pipelines often encounter difficulties to close gaps in draft genome assemblies introduced by regions enriched in repeated elements. These assemblers efficiently generate overlapping sets of reads (i.e., contiguous sequences or contigs) but encounter difficulties linking these contigs together into scaffolds. At the chromosome level, these programs often incorrectly orient DNA sequences or predict incorrect numbers of chromosomes [4]. The development of long-read sequencing technology and accompanying assembly programs has considerably alleviated these difficulties, but some gaps remain nevertheless in genome scaffolds, notably at the level of long repeated/low-complexity DNA sequences. In addition, long-read-based assemblies are associated with increased error rate among long reads, which can result in misassemblies [3]. Consequently, many currently



© The Author(s). 2020 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

available genomes still contain structural errors, as well as gaps that need to be bridged to reach a chromosome-level structure.

These limitations have been partially addressed thanks to active support from the community and competitions such as GAGE [5] or the Assemblathon [6]. However, there is as yet no systematic, reliable workflow of producing near-perfect genome assemblies of guaranteed optimal best quality without a considerable amount of empiric parameter adjustment and manual post-processing evaluation and correction [7].

Recent sequencing projects have typically relied on a combination of independently obtained data such as optical mapping, long-read sequencing, and chromosomal conformation capture (3C, Hi-C) to obtain large genome assemblies of high accuracy. The latter procedure derives from techniques aiming at recovering snapshots of the higher-order organization of a genome [8, 9]. When applied to genomics, Hi-C-based methods are sometimes referred to as proximity ligation approaches, as they quantify and exploit physical contacts between pairs of DNA segments in a genome to assess their collinearity along a chromosome, and the distance between the segments [10]. Early studies using control datasets demonstrated that Hi-C can be used to scaffold and/or correct a wide range of eukaryotic DNA regions [11–14], i.e. stretches of bp, whether they be small-scale contigs or full chromosomes. The Hi-C scaffolder GRAAL (Genome Re-Assembly Assessing Likelihood from 3D) is a probabilistic program that uses a Markov Chain Monte Carlo (MCMC) approach. This tool was able to generate the first chromosome-level assembly of an incomplete eukaryote genome [13] by permuting DNA segments according to their contact frequencies until the most likely scaffold was reached (see also [15]). Since these proof of concept studies, the assemblies of many genomes of various sizes from eukaryotes [16–18] and prokaryotes [19] have been significantly improved using scaffolding approaches exploiting Hi-C data.

Although GRAAL was effective on medium-sized or small (< 100 Mb) eukaryotic genomes such as that of the fungus *Trichoderma reesei* [20], scalability limitations were encountered when tackling genomes whose complexity and size required significant computer calculation capacity. Furthermore, as was also observed with other Hi-C-based scaffolders, the raw output of GRAAL includes a number of caveats that need to be corrected manually to obtain a finished genome assembly. To overcome these limitations, we developed instaGRAAL, an enhanced, open-source program optimized to reduce the computational load of chromosome scaffolding and that includes a misassembly “correction” module installed alongside the scaffolder. Moreover, instaGRAAL can optionally exploit available genetic linkage data.

We applied instaGRAAL to three genomes of increasing size: in the first two runs, and in order to demonstrate its added value, we applied the program to the 214-Mb and 500-Mb haploid genomes of the brown alga *Ectocarpus* sp. [21, 22] and *Desmarestia herbacea* (unpublished), respectively. Brown algae are a group of complex multicellular eukaryotes that have been evolving independently from animal and land plants for more than a billion years. *Ectocarpus* sp. was the first species within the brown algal group to be sequenced (reference v1 assembly [22]), as a model organism to investigate multiple aspects of brown algal biology including the acquisition of multicellularity, sex determination, life cycle regulation, and adaptation to the intertidal [22–25]. A range of genetic and genomic resources have also been established for *Ectocarpus* sp. including a dense genetic map generated with 3588 SNP markers (v2 assembly) [26], which was

used to comprehensively validate both a GRAAL (v3) and the instaGRAAL (v4) assemblies. In a third run, we benchmarked instaGRAAL using the human genome, to confirm that our software readily scales to larger (Gb-sized) and more complex assemblies, an important requirement to tackle the next era of assembly projects.

Results

From GRAAL to instaGRAAL

The core principles of GRAAL and instaGRAAL are similar: both exploit a MCMC approach to perform a series of permutations (insertions, deletions, inversions, swapping, etc.) of genome fragments (referred to here as “bins,” see the “Material and methods” section) based on an expected contact distribution [13]. The parameters (A , α , and δ) that describe this contact distribution are first initialized using a model inspired by polymer physics [27]. This model describes the expected contact frequency $P(s)$ between two loci separated by a genomic distance s (when applicable):

$$P(s) = \begin{cases} \max(A \cdot s^{-\alpha}, \delta) & : \text{intracontacts} \\ \delta & : \text{intercontacts} \end{cases}$$

The parameters are then iteratively updated directly from the real scaffolds once their sizes increase sufficiently [13]. Each bin is tested in several positions relative to putative neighboring fragments. The likelihood of each arrangement is assessed from the simulated or computed contact distribution, and the arrangement is either accepted or rejected [13]. This analysis is carried out in cycles, with a cycle being completed when all the bins of the genome have been processed in this way. Any number of cycles can be run iteratively, and the process is usually continued until the genome structure ceases to evolve, as measured by the evolution of the parameters of the model. The core functions of the program use Python libraries, as well as the CUDA programming language, and therefore necessitate a NVIDIA graphics card with at least 1 Gb of memory.

The technical limitations of GRAAL were (1) high memory usage when handling Hi-C data for large genomes (i.e. over 100 Mb), (2) difficulties when installing the software, and (3) the need to adjust multiple ad hoc parameters to adapt to differences in genome size, read coverage, Hi-C contact distribution, specific contact features, etc. instaGRAAL (<https://github.com/koszullab/instaGRAAL>) addresses all these shortcomings. First, we rewrote the memory-critical parts of the program, such as permutation sampling and likelihood calculation, so that they are computed using sparse contact maps. We reduced the software’s dependency footprint and added detailed documentation, deployment scripts, and containers to ease its installation. Finally, we opened up multiple hard-coded parameters to give more control for end-users while improving the documentation on each of them and selecting relevant default parameters that can be implemented for a wide range of applications (see options online and the “Discussion” section). Overall, these upgrades result in a program that is lighter in resources, more flexible, and more user-friendly.

Other problems encountered with the original GRAAL program included (1) the presence of potential artifacts introduced by the permutation sampler, such as spurious permutations (e.g. local inversions) or incorrect junctions between bins; (2) difficulties with the correct integration of other types of data such as long reads; and (3) difficulties

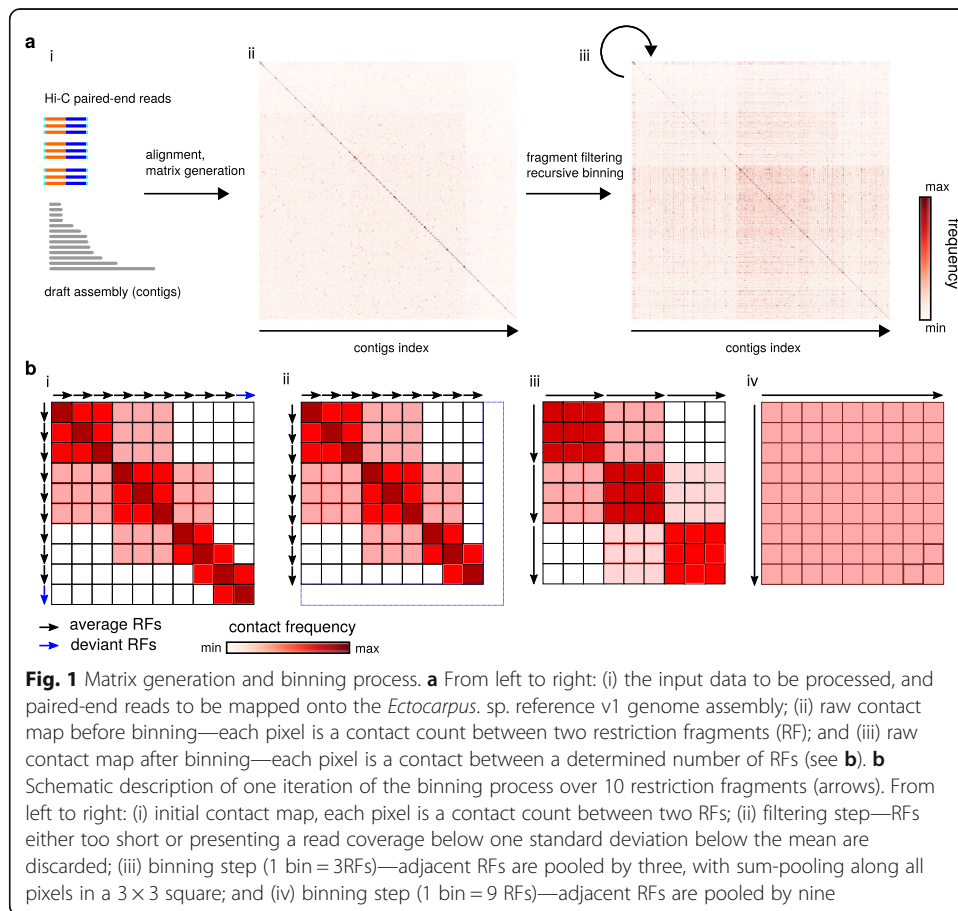
with handling sequences that were either too short, highly repeated, or with low coverage. We addressed these points by identifying and putting aside these problematic sequences during a filtering step. These sequences are subsequently reinserted into the final scaffolds, whenever possible (see the “Material and methods” section), with the help of linkage data when available. Overall, when compared to the raw GRAAL output, the resulting “corrected” instaGRAAL assemblies were significantly more complete and more faithful to the actual chromosome structure.

Scaffolding of the *Ectocarpus* sp. chromosomes with instaGRAAL

To test and validate instaGRAAL, we generated an improved assembly of the genome of the model brown alga *Ectocarpus* sp. A v1 genome consisting of 1561 scaffolds generated from Sanger sequence data is available [22]. A Hi-C library was generated from a clonal culture of a haploid partheno-sporophyte carrying the male sex chromosome using a GC-neutral restriction enzyme (DpnII). The library was paired-end sequenced (2×75 bp—the first ten bases were used as a tag and to remove PCR duplicates) on a NextSeq apparatus (Illumina). Of the resulting 80,521,968 paired-end reads, 41,288,678 read pairs were aligned unambiguously along the v1 genome using bowtie2 (quality scores below 30 were discarded), resulting in 2,554,639 links bridging 1,806,386 restriction fragments (Fig. 1a) (see the “Material and methods” section for details on the experimental and computational steps). The resulting contact map in sparse matrix format was then used to initialize instaGRAAL along with the restriction fragments (RFs) of the reference genome (Fig. 1a, b) (see Additional file 1: Table S1 for an example of sparse file matrix).

Given the probabilistic nature of the algorithm, we evaluated the program’s consistency by running it three times with different resolutions. Briefly, we filtered out RFs that were shorter than 50 bp and/or whose coverage was one standard deviation below the mean coverage. Then, we sum-pooled (or binned) the sparse matrix by groups (or bins) of three RFs five times, recursively (Fig. 1a, b). Each recursive instance of the sum-pooling is subsequently referred to as a level of the contact map. A level determines the resolution at which permutations are being tested: the higher the level, the lower the resolution, the longer the sequences being permuted and, consequently, the faster the computation. The binning process is shown in Fig. 1b. Regarding *Ectocarpus* sp., we found that level 4 (bins of 81 RFs) was an acceptable balance between high resolution and fast computation on a desktop computer with a GeForce GTX TITAN Z graphics card. Moreover, whether instaGRAAL was run at level 4, 5, or 6 (equivalent to bins of 81, 243, and 729 RFs, respectively), all assemblies quickly (~ 6 h) converged towards similar genome structures (Fig. 2a).

We plotted the evolution of the log-likelihood and of model parameters as a function of the number of arrangements performed (iterations) (Fig. 2b). The interquartile ranges (IQR, used to indicate stability in Marie-Nelly et al. [13]) of all parameters decreased to near-zero values at the end of each scaffolding run, indicating that they all stably converged and that the final structures oscillated near the final values in negligible ways. More qualitatively, each run led to the formation of 27 main scaffolds (Fig. 2a) with the 27th largest scaffold being more than a hundred times longer than the 28th largest one (Fig. 3, Additional file 1: movie S1). Each of the 27 scaffolds was

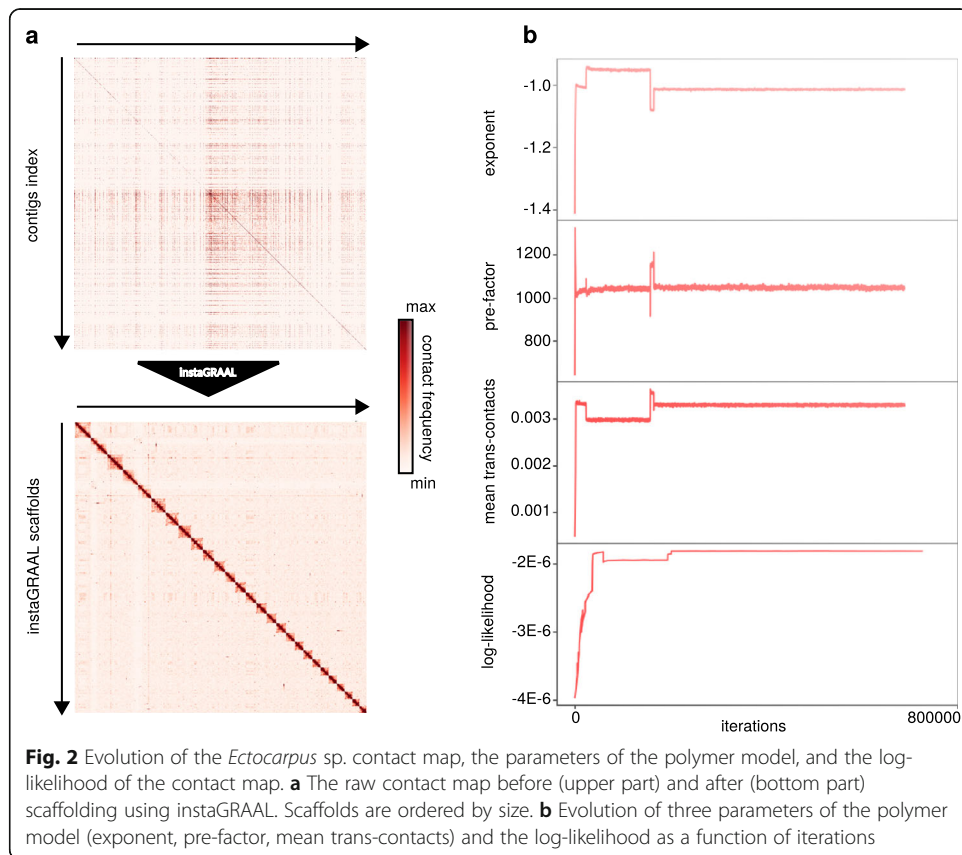


between four and ten times longer than the combined length of the remaining sequences (Fig. 3). This strongly suggests that the 27 scaffolds correspond to chromosomes, a number consistent with karyotype analyses [28]. Taken together, these results indicate that instaGRAAL successfully assembled the *Ectocarpus* sp. genome into chromosome-level scaffolds. As the supplementary movie suggests, scaffold-level convergence is visible after only a few cycles, indicating that instaGRAAL is able to quickly determine the global genome structure most likely to fit the contact data. The remainder of the cycles is devoted to intra-chromosomal refinement.

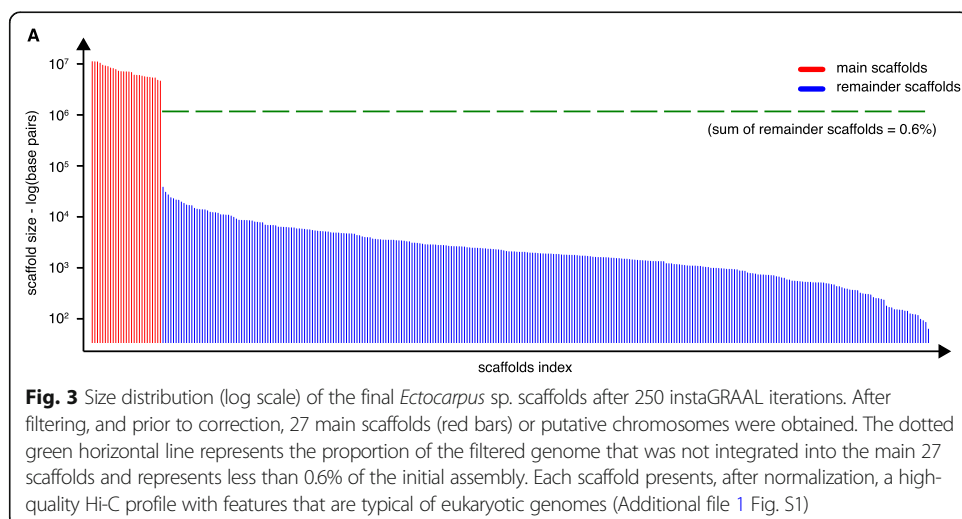
Correcting the chromosome-level instaGRAAL assembly of the *Ectocarpus* sp. genome

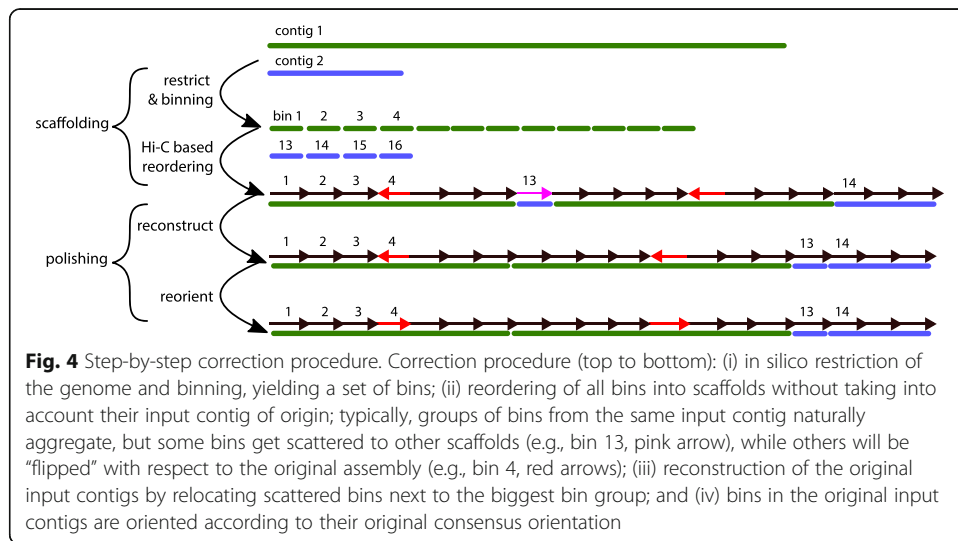
instaGRAAL also includes a number of procedures that aim to correct some of the modifications introduced into the input contigs from the original assembly by the Hi-C scaffolding (Fig. 4). We implemented it as a separate “correction” module that is automatically installed alongside the scaffolder.

These modifications principally involve discrete inversions or insertions of DNA segments (typically corresponding to single bins or RFs) (see also [13]). Such alterations are inherent to the statistical nature of instaGRAAL, which will occasionally improperly permute neighboring bins because of the high density of contacts between them. However, we reasoned that input contigs from the original assembly, especially those



generated for *Ectocarpus* sp. with Sanger sequencing, were unlikely to contain misassemblies. Therefore, we decided to favor input contigs' structure whenever local conflicts arose. These are part of a broader set of assembly errors that we detected by aligning the v1 assembly on the instaGRAAL scaffolds and analyzing the mapping results using QUAST. The v1 assembly was used as a reference by QUAST to identify potential errors introduced by instaGRAAL when scaffolding the v1 assembly. We corrected these errors as follows: first, all bins processed by instaGRAAL that





belonged to the same input contig were constrained to their original orientation (Fig. 4). If an input contig was split across multiple scaffolds, the smaller parts of this contig were relocated to the largest one, respecting the original order and orientation of the bins. Then, we reinserted whenever possible sequences that had been filtered out prior to instaGRAAL processing (e.g., contig extremities with poor read coverage; see the “Material and methods” section and Marie-Nelly et al. [13]) into the chromosome-level scaffold at their original position in the original input contig. 3,832,980 bp were reinserted into the assembly this way. These simple steps alleviated artificial truncations of input contigs observed with the original GRAAL program.

Some filtered bins had no reliable region to be associated with post-scaffolding, because their initial input contig had been completely filtered before scaffolding. These sequences, which were left as-is and appended at the end of the genome, were included into 543 scaffolds spanning 3,141,370 bp, i.e., <2% of the total DNA. Together, these steps removed all the misassemblies detected by QUAST.

To further validate the assembly, we exploited an assembly generated by combining genetic recombination data and the Sanger assembly [21, 26] (“linkage group [LG] v2 assembly”) as well as an assembly generated by running the original GRAAL program on the original reference v1 genome assembly (“GRAAL v3 assembly”).

We searched for potential translocations between scaffold extremities between the linkage group v2 assembly and the v3 or v4 assemblies. This comparison, which was implemented as a separate module installed alongside the scaffolder, detected such events in the uncorrected v3 GRAAL assembly but none in the corrected v4 instaGRAAL assembly. The corrected instaGRAAL v4 assembly is therefore fully consistent with the genetic recombination map data, confirming the efficiency of the approach.

Comparisons with previous *Ectocarpus* sp. assemblies and validation of the instaGRAAL assembly

We compared the corrected instaGRAAL v4 assembly with the three earlier assemblies of the *Ectocarpus* sp. genome mentioned above (Table 1 and Additional file 1: Table

S2): (1) the original v1 genome assembly generated using Sanger sequencing data [22], which was assumed to be highly accurate but fragmented (1561 scaffolds); (2) the linkage group [LG] v2 assembly; and (3) the original GRAAL program v3 assembly.

We aligned the corrected instaGRAAL (v4), LG (v2), and GRAAL (v3) assemblies onto the original v1 assembly to detect misassemblies and determine whether the genome annotations (362,919 features) were conserved. We then validated each assembly using genetic linkage data (see the “Material and methods” section). For each assembly, we computed the following metrics: the number of misassemblies, ortholog completeness, and cumulative length/Nx distributions (Table 1). These assessments were carried out using BUSCO [29] for ortholog completeness (Additional file 1: Fig. S1) and QUAST-LG’s validation pipeline [30] to search for misassemblies introduced in the scaffolds. QUAST-LG is an updated version of the traditional QUAST pipeline specifically designed for large genomes and is a state-of-the-art software for assembly evaluation and comparison. We used QUAST to verify that annotations transferred successfully from the reference v1 assembly to the instaGRAAL v4 assembly and that no structural discrepancy (a.k.a. misassemblies) was found in the instaGRAAL v4 assembly with respect to the reference v1 assembly. We followed the terminology used by both programs, such as the BUSCO definition of ortholog and completeness, as well as QUAST’s classification system of contig and scaffold misassemblies.

The corrected instaGRAAL assembly was of better quality than both the LG v2 and GRAAL v3 assemblies (Table 1 and Additional file 1: Fig. S2). The corrected assembly incorporated 795 of the v1 genome scaffolds (96.8% of the sequence data) into the 27 chromosomes based on the high-density genetic map [21], compared to 531 for the LG v2 assembly (90.5% of the sequence data). Moreover, this assembly contained fewer misassemblies and was more complete in terms of BUSCO ortholog content. For some metrics, the differences were marginal, but always in favor of the corrected instaGRAAL v4 assembly. BUSCO completeness was similar (76.2%, 76.9%, and 77.6% for the GRAAL v3 assembly, LG v2, and corrected instaGRAAL v4 assemblies, respectively) (Additional file 1: Fig. S2) and an improvement over the 75.9% of the v1 assembly. These absolute numbers remain quite low, presumably because of the lack of a set of orthologs well adapted to brown algae.

Table 1 Comparison of Nx, NGx (i.e., Nx with respect to the original reference v1 genome assembly; in bp), and BUSCO completeness for the different assemblies (linkage group v2, GRAAL v3, and corrected instaGRAAL v4) of the *Ectocarpus* sp. genome

	Reference v1 assembly	Linkage group v2 assembly	v3 GRAAL	v4 corrected instaGRAAL
N50	497,380	6,528,661	6,867,074	6,813,345
NG50	497,380	6,528,661	6,725,743	6,813,345
N75	233,412	5,613,161	5,693,784	5,686,617
NG75	233,412	5,613,161	5,672,622	5,686,617
L50	118	12	11	11
LG50	118	12	12	11
L75	258	19	18	19
LG75	258	19	19	19
BUSCO completeness (%)	75.9	76.9	76.24	77.56

All quantitative metrics, such as N50, L50, and cumulative length distribution, increased dramatically when compared with the reference genome v1 assembly (Table 1). N50 increased more than tenfold, from 496,777 bp to 6,867,074 bp after the initial scaffolding and to 6,942,903 bp after the correction steps. 99.4% of the sequences in the 1018 contigs were integrated into the 27 largest scaffolds after instaGRAAL processing. Overall, the analysis indicated that many of the rearrangements found in the LG v2 assembly were potentially errors and that both GRAAL and instaGRAAL were efficient at placing large regions where they belong in the genome, albeit less accurately for GRAAL and in the absence of correction. These statistics underline the importance of the post-scaffolding correction steps and the usefulness of a program that automates these steps.

Comparison between the *Ectocarpus* sp. instaGRAAL and linkage group assemblies

Compared to the LG v2 assembly, the corrected instaGRAAL v4 assembly lost 23 scaffolds but gained 287 that the genetic map had been unable to anchor to chromosomes (Additional file 1: Table S2). We observed few conflicts between the two assemblies, and the linkage markers are globally consistent with the instaGRAAL scaffolds (Additional file 1: Fig. S3). One major difference is that instaGRAAL was able to link the 4th and 28th linkage groups (LG) that were considered to be separate by the genetic map [26] because of the limited number of recombination events observed. The fusion in the instaGRAAL v4 assembly is consistent with the fact that the 28th LG is the smallest, with only 54 markers over 41.8 cM and covering 3.8 Mb. The 28th LG has a very large gap which might reflect uncertainty in the ordering of the markers. Interestingly, this gap is located at one end of the group, precisely where instaGRAAL now detects a fusion with the 4th LG. In addition, the fact that there is no mix between the 4th and 28th LGs on the merged instaGRAAL (pseudo) chromosome but rather a simple concatenation suggests that the genetic map was unsuccessful in joining those two LGs, but that instaGRAAL correctly assembled the two LGs (see Additional file 1: Table S3 for correspondences between LGs and instaGRAAL super scaffolds).

instaGRAAL was also more accurate than the genetic map in orienting scaffolds (Additional file 1: Table S2). Among the scaffolds that were oriented in the LG v2 assembly, about half of the “plus” orientated were actually “minus” and vice versa. The limited number of markers detected in the scaffolds anchored to the genetic map was likely the reason for this high level of incorrect orientations.

Scaffolding of the *Desmarestia herbacea* genome

To test and validate instaGRAAL on a second, larger genome, we generated an assembly of the haploid genome of *D. herbacea*, a brown alga that had not been sequenced before. We set up the assembly pipeline and subsequent scaffolding from raw sequencing reads to assess the robustness of instaGRAAL with de novo, non-curated data. The pipeline proceeded as follows: first, we acquired 259,556,174 short paired-end shotgun reads (Illumina HiSeq2500 and 4000) as well as 1,353,202 long reads generated using PacBio and Nanopore (about 150× short reads and 15× long reads). Sequencing reads were processed using the hybrid MaSuRCA assembler (v3.2.9) [31], yielding 7743 contigs representing 496 Mb (Table S4). We generated Hi-C data following a protocol

similar to that used for *Ectocarpus* sp. (see the “Material and methods” section). Briefly, 101,879,083 reads were mapped onto the hybrid assembly, yielding 7,649,550 contacts linking 1,359,057 fragments. We then ran instaGRAAL using similar default parameters to that used for *Ectocarpus* sp., for the same number of cycles. We corrected the resulting scaffolds. The scaffolding process resulted in 40 scaffolds larger than 1 Mb (Additional file 1: Fig. S4, S5, S6), representing 98.1% of the initial, filtered scaffolding and 89.3% of the total initial genome after correction and reintegration. The exact number of chromosomes in *D. herbacea* is unknown but was estimated to be ~ 23 , and possibly up to 29, based on cytological observations [32]. Most (35) of the scaffolds generated by instaGRAAL were syntenic with the 27 *Ectocarpus* sp. scaffolds. Among the remaining five scaffolds, one corresponded to the genome of an associated bacterium, and two to large regions with highly divergent GC content (37 and 40% vs. 48% for the rest of the genome) and no predicted *D. herbacea* genes. Overall, instaGRAAL successfully scaffolded the *D. herbacea* genome, although the final number of scaffolds remained slightly higher than the estimated number of chromosomes in this species.

Comparisons with existing methods

To date, only a limited number of Hi-C-based scaffolding programs are publicly available, and as far as we can tell, no detailed comparison has been performed between the existing programs to assess their respective qualities and drawbacks. In an attempt to benchmark instaGRAAL, we ran SALSA2 [33] and 3D-DNA on the same *Ectocarpus* sp. v1 and *Desmarestia herbacea* reference genome and Hi-C reads. 3D-DNA is a scaffold folder that was hallmarked with the assembly of *Aedes aegypti*, and SALSA2 is a recent program with a promising approach that directly integrates Hi-C weights into the assembly graph. For *Ectocarpus* sp., SALSA2 ran for nine iterations and yielded 1042 scaffolds, with an N50 of 6,552,506 (L50 = 11). Its BUSCO completeness was 77.6%, a level identical to that obtained with instaGRAAL. Overall, the metrics were satisfactory but SALSA2 was outperformed by instaGRAAL post-correction. The contact map of the resulting SALSA2 assembly displayed noticeably unfinished scaffolds (Additional file 1 Fig. S7 and S8). This, coupled with a lower N50 value, suggests that instaGRAAL is more successful at merging scaffolds when appropriate.

We computed similar size and completeness statistics for the final instaGRAAL *D. herbacea* assembly and compared these to the values obtained with SALSA2 and 3D-DNA. We also mapped the Hi-C reads onto all three final assemblies in order to qualitatively assess the chromosome structure. The results are summarized in Table S4.

Briefly, statistics across assemblies were similar; the corrected instaGRAAL assembly had 73% BUSCO completeness, consistent with the values of 73.6% and 70.3% obtained for SALSA2 and 3D-DNA, respectively. However, the Lx/Nx metrics diverged significantly; the instaGRAAL assembly N50 was 12.4 Mb, similar to SALSA2 (12.8) and much larger than 3D-DNA (0.2 Mb). However, visual inspection of the contact maps indicated that neither SALSA2 nor 3D-DNA succeeded in fully scaffolding the genome of *Desmarestia herbacea* (Additional file 1: Fig. S7). Notably, SALSA2 created a number of poorly supported junctions to generate chromosomes, whereas 3D-DNA failed to converge towards any kind of structure. In contrast, although the instaGRAAL final assembly still contains input contigs that are incorrectly positioned, a coherent

structure corresponding to 40 scaffolds (including contaminants) emerged (Additional file 1: Fig. S4). One possibility is that the de novo MaSuRCA assembly was low quality, likely due to the low coverage of long reads, which would have resulted in alignment errors that disrupted the contact distribution and subsequent Hi-C scaffolding. Another possible explanation for these differences is that it remains difficult to dissect all the options and tunable parameters of these scaffolders, and therefore that we did not find the optimal combination with respect to the *D. herbacea* draft assembly. Nevertheless, these results highlight the robustness of instaGRAAL which was able to scaffold the *D. herbacea* genome using default parameters.

Scaffolding the human genome

To confirm that instaGRAAL scaffolds larger (Gb scale) genomes in a reasonable time, we ran it on the GRCh38 human genome sliced into 300-kb segments (artificial assembly), using a Hi-C dataset generated with an Arima Genomics Hi-C kit (see the “Material and methods” section). instaGRAAL was run for 15 cycles, with the parameter `--levels` set to 5, and the scaffolds were subsequently corrected with instaGRAAL-polish. We obtained a total of 1302 scaffolds, out of which 24 have a length ranging from 18 to 239 Mb. These 24 chromosome-level scaffolds are represented in the contact map in Additional file 1: Fig. S9. These scaffolds have an N50 and an NGA50 of 143 Mb, close to the 145 Mb obtained for the reference genome (Table 2; the results from [33] using SALSA2 are included). The dot plot similarity map between the instaGRAAL scaffolds and reference genome assembly (Additional file 1: Fig. S10) shows that the 22 autosomes and the X chromosome were recovered by instaGRAAL (although a few relocations and inversions remain visible). In addition, a 24th scaffold is visible composed of sequences also in contacts with the other scaffolds, corresponding to repeated sequences clustering together. instaGRAAL produced scaffolds with a lower contiguity than those of SALSA2: while their N50 are comparable, the N75 of instaGRAAL is significantly lower. However, the number of complete genomic features in the instaGRAAL scaffolds is largely improved compared to the input fragments, while SALSA2 only slightly increased this score. These results suggest that although the scaffolds of instaGRAAL are less contiguous, they are of better quality. Since these scaffolds were obtained after only 15 cycles, increasing the number of cycles is very likely to improve the N75. All in all, and though additional work is needed to polish such an output as with all assembly projects, these results confirm that instaGRAAL can efficiently scaffold large genomes.

Benchmarking of the system requirements

To quantify the improvements made over the original GRAAL program, we ran both GRAAL and instaGRAAL over the *Ectocarpus* sp. v1 genome separately and measured the peak memory load, the graphics card memory load taken by the contact maps, and the per-cycle runtime as averaged from 20 cycles. The results are summarized in Table S5. As expected, the memory load on the graphics card is an order of magnitude smaller for instaGRAAL, while the peak RAM and runtime are several times smaller. The shrinkage of memory requirements is predicted by the use of sparse data structures and the fact that our original dataset for *Ectocarpus* sp. is relatively lean when

Table 2 Comparison of Nx, NGx (i.e., Nx with respect to the original human reference genome assembly; in bp), and other QUASt statistics for the different assemblies (artificial assembly, corrected instaGRAAL, and SALSA2) of the *Homo sapiens* genome

	Reference genome assembly	Artificial assembly	instaGRAAL	SALSA2
N50	145,138,636	300,000	143,373,745	152,389,473
NG50	145,138,636	300,000	143,373,745	152,389,473
N75	107,043,718	300,000	89,477,166	130,103,422
NG75	107,043,718	300,000	82,128,910	103,672,000
L50	9	5165	9	9
LG50	9	5454	9	9
L75	15	7747	15	15
LG75	15	8181	17	17
No. of genomic features	3,625,295 + 305 part	3,411,473 + 44,299 part	3,456,227 + 3836 part	3,415,115 + 44,127 part
Genome fraction (%)	100.0	94.6	94.6	94.5
No. of misassemblies	9	0	776	438

compared to the size of the genome. The origin of the accelerated runtime is less clear and could be due to multiple contributions to the program, including the use of sparse data structures but also external contributions (e.g., porting to Python 3, upgraded libraries, or more recent CUDA versions).

It is important to note, however, that these results are highly specific to the hardware and data used here, and due to the many different factors involved, any comparison should stick to orders of magnitude. Nevertheless, this confirms that instaGRAAL’s improvements over GRAAL are very substantial and make it suitable for modern, large genome assembly projects.

Discussion

instaGRAAL is a Hi-C scaffolding program that can process large eukaryotic genomes. Below, we discuss the improvements made to the program, its remaining limitations, and the steps that will be needed to tackle them.

Refinement/correction step

An important improvement of instaGRAAL compared to GRAAL relates to post-scaffolding corrections. Local misassemblies, e.g., local bin inversions or disruptive insertions of small scaffolds within larger ones, are an inevitable consequence of the algorithm’s most erratic random walks. These small misassemblies are retained because flipping a bin does not markedly change the relative distance of an RFs relative to its neighbors, and because small scaffolds typically carry less signal and therefore exhibit a greater variance in terms of acceptable positions. Depending on the trust put in the initial set of contigs, one may be unwilling to tolerate these changes as well as “partial translocations,” i.e., the splitting of an original contig into two scaffolds. The prevalence of such mistakes can be estimated by comparing the orientation of bins relative to their neighbors in the instaGRAAL v4 assembly vs. the original assembly (v1 assembly). Our assumption is that if a single bin was flipped or split by instaGRAAL, this was likely a

mistake that needed to be corrected. Consequently, we chose to remain faithful to the input contigs of the original v1 assembly, given that the initial *Ectocarpus sp.* v1 (reference) genome sequence was based on Sanger reads. Our correction therefore aims at reinstalling the initial contig structure and orientation while preserving to a maximum extent the overall instaGRAAL scaffold structure.

In addition, our correction reintegrates into the assembly the bins removed during the initial filtering process according to their position along the original assembly contigs. Most filtered bins corresponded to the extremities of the original contigs, because their size depended on the position of the restriction sites within the contig, or because they consisted of repeated sequences with little or no read coverage. The tail filtering correction step inserts these bins back at the extremities of these contigs in the instaGRAAL assembly.

The combination of a probabilistic algorithm with a deterministic correction step provides robustness to instaGRAAL. First, the MCMC step identifies, with few prior assumptions, a high-likelihood family of genome structures, almost always very close to the correct global scaffolding. The correction step combines this result with prior assumptions made about the initial contig structures generated through robust, established assembly programs, refining the genomic structure within each scaffold. To give the user a fine-grained degree of control over our correction procedures, the implementation into instaGRAAL is split into independent modules that each assume about the initial contig structure necessary to perform the correction: the “reorient” module assumes that the initial contigs do not display inversions, and the “rearrange” module assumes that there are no relocations within contigs.

We underline that despite the improvements brought about by these new procedures, instaGRAAL assemblies remain perfectible, notably because of the reliance on the quality of the input contigs used for correction. For instance, the *D. herbacea* genome heavily relies on contigs generated from a de novo hybrid assembly, and the contact maps in Additional file 1: Fig. S4, S5, and S6 show some extraneous signal that may point at misassemblies. Analogous observations may be made with respect to *Ectocarpus sp.* in Additional file 1: Fig. S11. In addition, inherent limits to Hi-C technology such as the restriction fragment size mean that there are going to be false junctions between fragments or bins. This is only a problem if one chooses not to reconstruct every input contig within a newly formed scaffold with our correction procedure, i.e., one is distrustful of the initial input contigs. This was not the case for *Ectocarpus sp.* but could be argued for *D. herbacea*, where the de novo contigs generated from 15× coverage may be of poor quality.

Sparse data handling

The implementation of a sparse data storage method in instaGRAAL allows much more intense computation than with GRAAL. Because the majority of map regions are devoid of contacts, instaGRAAL essentially halves the order of magnitude of both algorithm complexity and memory load, i.e., they increase roughly linearly with the size of the genome instead of geometrically. This improvement potentially allows the assembly of Gb-sized genomes in 4 to 5 days using a laptop (i.e., much faster with more computational resources).

Filtering

Variations in GC% along the genome, and/or other genomic features, can lead to variation in Hi-C read coverage and impair interpretation of the Hi-C data. Correction and attenuation procedures that alleviate these biases are therefore commonly used in Hi-C studies [34–36]. However, these procedures are not compatible with instaGRAAL’s estimation of the contact distribution (for more details, see [37]). A subset of bins will therefore diverge strongly from the others, displaying little if no coverage. A filtering step is needed to remove these bins as they would otherwise impact the contact distribution and the model parameter estimation. These disruptive bins represent a negligible fraction of the total genome (< 3% of the total genome size of *Ectocarpus* sp., for instance) and are reincorporated into the assembly during correction. On the other hand, a subset of bins representing small, individual scaffolds are not reinserted during correction and are added to the final assembly as extra-scaffolds (as in all sequencing projects). Additional analyses and new techniques such as long or linked reads are needed to improve the integration of these scaffolds into the genome.

Resolution

The binning procedure will influence the structure of the final assembly as well as its quality. For example, low-level binning (e.g., one bin = three RFs) will lead to an increased number of bins and a large, sparse contact map with a low signal-to-noise ratio, where many of the bins display poor read coverage as on average they will have fewer contacts with their immediate neighbors. Because of the resulting low signal-to-noise ratio, an invalid prior model will be generated, and when referring to this model, the algorithm will fail to scaffold the bins properly, if at all. Moreover, due to its probabilistic nature, the algorithm will generate a number of false positive structural modifications such as erroneous local inversions or permutations of bins. The numerous bins will create more genome structures to explore to handle all the potential combinations, and exploring this space until convergence will take longer and be computationally demanding.

On the other hand, one of the advantages of instaGRAAL is its ability to scaffold fragments or bins instead of contigs themselves. This has two main effects: First, it dodges the size bias issue whereby larger contigs will feature more contacts and will need to be normalized. Second, it allows for greater flexibility when exploring genome space, potentially uncovering misassemblies within input contigs. This is more relevant in the case of large contigs generated with long reads. And even if we assume that the initial contigs are completely devoid of misassemblies, this flexibility is useful when the contact distribution is disrupted by extraneous signals and the scaffolder needs to decide between two regions of similar affinity. The correction tool subsequently reconstructs the initial contigs from these rough arrangements, as discussed above (reference-based correction).

An optimal resolution is therefore a compromise between the bin size, the coverage, and the quality of the input contigs from the original assembly. Although a machine powerful enough operating on an extremely contact-rich matrix would be successful at any level, it is unclear whether such resources are necessary. Our present assemblies (e.g., 1 bin = 81 RFs for both; see the “[Material and methods](#)” section) had good quality

metrics after a day's worth of calculation on a standard desktop computer for *Ectocarpus* sp. and *D. herbacea*. Moreover, convergence was qualitatively obvious after a few cycles. This suggests that more computational power yields diminishing returns and therefore that appropriate correction procedures are a more efficient approach for remaining misassemblies.

Binning

The fragmentation of the original assembly used to generate the initial contact map has a substantial effect on the quality of the final scaffolding. Because binning cannot be performed beyond the resolution of individual input contigs, however small they may be, there is a fixed upper limit to the scale at which a given matrix can be binned. A highly fragmented genome with many small input contigs will necessarily generate a high-noise, high-resolution matrix. Attempts to reassemble a genome based on such a matrix will run into the problems discussed above (resolution). This limitation can be alleviated, to some extent, by discarding the smallest contigs, with the hope that the remaining contigs will cover enough of the genome. The input contigs that are removed can be reintegrated into the final scaffold during the correction steps. This ensures an improved Nx metric while retaining genome completeness. It should be noted, however, that the size of the input contigs is important as they need to contain sufficient restriction sites, and each of the restriction fragments must have sufficient coverage. The choice of enzyme and the frequency of its corresponding site are thus crucial. For instance, with an average of one restriction site every 600 to 1000 bp for *DpnII*, input contigs as short as 10 kb may contain enough information to be correctly reassembled. The restriction map therefore strongly influences both the minimum limit on N50 and genome fragmentation.

Benchmarking

In order to test our tool against existing programs, we ran two scaffolders available online (SALSA2 and 3D-DNA) on our two genomic datasets. In all instances, instaGRAAL proved more successful at scaffolding both genomes. However, we have not extensively tested all the combinations of parameters of both programs, and acknowledge the difficulty in designing and implementing Hi-C scaffolding pipelines with extensive dependencies that compound the initial complexity of the task and add yet more configurable options to know in advance. Finding the correct combination of CUDA and Python dependencies to install instaGRAAL on a given machine can be challenging as well. Therefore, our benchmarking attempt should be rather seen as a way to stress the importance of implementing sensible default parameters that readily cover as many use cases as possible for the end user. There is almost no doubt that both 3D-DNA and SALSA2, with the appropriate parameters and correction steps, would produce satisfying scaffolding; on the other hand, knowing which input parameters has to be specified in advance is a non-trivial task, especially given the computational resources needed for a single scaffolding run. With instaGRAAL, we wish to combine the simplicity of a default configuration that works in most instances, with the flexibility offered by the power of MCMC methods.

Choosing your parameters

In the benchmarking, we have discussed why some parameters are crucial and why we took care, through trial-and-error, to implement sensible defaults for future similar assembly projects. On the other hand, it is crucial that such defaults be not the result of overfitting for the assemblies we tested. However, none of what we outlined previously assumes anything specific about the genomes at hand beyond very broad metrics such as their total size or N50. The parameters of the program scale intuitively with such metrics. For larger genomes, one may simply increase the size of the bins so that the contact map does not grow too large, which is what we did for the human genome. The N50 sets the resolution limit in that it is often desirable to be able to break down contigs into many bins of roughly equal size so as not to run into the aforementioned size bias and also to be able to give more flexibility to the program. For instance, an N50 close to 100 kb should not feature bins larger than 50–60 kb. Oftentimes, however, such minutiae is not necessary, and for most genome projects ranging across 10^7 – 10^9 bp, instaGRAAL will typically work out of the box with default parameters. For instance, we kept the same parameters for both algae and only switched to a lower resolution (higher bin size) for the human genome to scale with its size. When needed, through these simple rules of thumb, one may adapt the defaults to other genomes with more extreme metrics.

Handling diploid genomes

As assembly projects have grown more complex and exhaustive, expectations have increased as well. Assembling diploid, if not polyploid, genomes with well-characterized haplotypes is a stumbling block in the field. Moreover, such problems are more likely to be encountered as the low-hanging fruit gets picked. Typical projects involve assembling many individual complete human genomes with haplotypes, or the sequencing and scaffolding of even larger and more complex genomes such as that of plants. In this context, instaGRAAL in particular (and Hi-C in general) is relatively agnostic, as its success or failure will hinge on the reference genome being properly haplotyped in the first place. While it may prove intractable to phase haplotypes directly from only Hi-C data, instaGRAAL will conserve such information when provided in the first place. This is because the scaffolder is robust to local disruptions like haplotype-induced mapping artifacts. It has been shown that GRAAL and by extension instaGRAAL will eventually resolve such disruptions even when the distribution is noisy, as long as the general three-parameter model (and power law) still holds globally [13, 19, 20]. In other words, even though instaGRAAL cannot “guess” whether a given reference sequence is homologous or heterozygous without considerable difficulty, it can still cleanly scaffold chromosome pairs from clear contig pairs because the global 3D intra-signature from a given contig is too strong to be confused with mapping artifacts in a pair. Should such information be missing, the scaffolder will likely interlace all regions into a giant linkage group. In that respect, instaGRAAL could interface well with diploid classical assemblers and is suitable for any pipeline integration involving diploid genomes. More work is needed in that direction so that the scaffolder does not rely that strongly on the quality of the input contigs to work out haplotypes.

Integrating information from the Hi-C analysis with other types of data

Aggregating data from multiple sources to construct a high-quality genome sequence remains a challenging problem with no systematic solution. As long-read technologies become more affordable, there is an increasing demand to reconcile the scaffolding capabilities of Hi-C-based methods with the ability of long reads to span regions that are difficult to assemble, such as repeated sequences. The most intuitive approach would be to perform Hi-C scaffolding on an assembly derived from high-coverage and corrected long reads, as was done for several previous assembly projects [16, 38]. Alternative approaches also exist, such as generating Hi-C- and long-read-based assemblies separately and merging them using programs such as CAMSA [39] or Metassembler [40]. Pipelines such as PBJelly [41] have proven successful at filling existing gaps in draft genomes, regardless of their origin, with the help of long reads. Lastly, with assembly projects involving both long and short reads, hybrid assemblies and hybrid polishing have become an important focus. Polishers such as Racon [42] or Pilon [43] are widely used, and new tools such as HyPo are also emerging [44]. Yet the question of which kind of pipeline to use (e.g., Racon to Hi-C scaffolding to Pilon, or Racon to Pilon to Hi-C scaffolding, etc.) along with which hybrid assembler (Masurca, Alpaca, hybridSPAdes, etc.) [31, 45, 46] can prove cumbersome, and often finding the process yielding the most satisfying output in terms of metrics involves much trial-and-error with different configurations. InstaGRAAL shows that high-quality metrics can still be attained without the help of long reads, but long-read polishing may still be necessary in order to get rid of the lingering errors we mentioned. Long reads are not the only type of data that can be used to improve assemblies. Linkage maps, RNA-seq, optical mapping, and 10X technology all provide independent data sources that can help improve genome structure and polish specific regions. The success of future assembly projects will hinge on the ability to process these various types of data in a seamless and efficient manner.

Material and methods**Preparation of the Hi-C libraries**

The Hi-C library construction protocol was adapted from [8, 47]. Briefly, parthenosporophyte material was chemically cross-linked for 1 h at RT using formaldehyde (final concentration, 3% in 1× PBS; final volume, 30 ml; Sigma-Aldrich, St. Louis, MO). The formaldehyde was then quenched for 20 min at RT by adding 10 ml of 2.5 M glycine. The cells were recovered by centrifugation and stored at -80°C until use. The Hi-C library was then prepared as follows. Cells were resuspended in 1.2 ml of 1× *DpnII* buffer (NEB, Ipswich, MA), transferred to a VK05 tubes (Precellys, Bertin Technologies, Rockville, MD), and disrupted using the Precellys apparatus and the following program ([20 s—6000 rpm, 30 s—pause] 9× cycles). The lysate was recovered (around 1.2 ml) and transferred to two 1.5-ml tubes. SDS was added to a final concentration of 0.3%, and the 2 reactions were incubated at 65°C for 20 min followed by an incubation of 30 min at 37°C . A volume of 50 μl of 20% Triton-X100 was added to each tube, and incubation was continued for 30 min. *DpnII* restriction enzyme (150 units) was added to each tube, and the

reactions were incubated overnight at 37 °C. Next morning, reactions were centrifuged at 16,000×g for 20 min. The supernatants were discarded, and the pellets were resuspended in 200 µl of NE2 1× buffer and pooled (final volume = 400 µl). DNA extremities were labeled with biotin using the following mix (50 µl NE2 10× buffer, 37.5 µl 0.4 mM dCTP-14-biotin, 4.5 µl 10 mM dATP-dGTP-dTTP mix, 10 µl Klenow 5 U/µl) and an incubation of 45 min at 37 °C. The labeling reaction was then split in two for the ligation reaction (ligation buffer—1.6 ml, ATP 100 mM—160 µl, BSA 10 mg/ml—160 µl, ligase 5 U/µl—50 µl, H₂O—13.8 ml). The ligation reactions were incubated for 4 h at 16 °C. After addition of 200 µl of 10% SDS, 200 µl of 500 mM EDTA, and 200 µl of proteinase K 20 mg/ml, the tubes were incubated overnight at 65 °C. DNA was then extracted, purified, and processed for sequencing as previously described (Lazar-Stefanita et al. [47]). Hi-C libraries were sequenced on a NextSeq 550 apparatus (2 × 75 bp, paired-end Illumina NextSeq with the first ten bases acting as barcodes; Marbouty et al. [15]).

Contact map generation

Contact maps were generated from reads using the hicstuff pipeline for processing generic 3C data, available at <https://github.com/koszullab/hicstuff>. The back-end uses the bowtie2 (version 2.2.5) aligner run in paired-end mode (with the following options: --maxins 5 --very-sensitive-local). Alignments with mapping quality lower than 30 were discarded. The output was in the form of a sparse matrix where each fragment of every chromosome was given a unique identifier and every pair of fragments was given a contact count if it was non-zero.

Fragments were then filtered based on their size and total coverage. First, fragments shorter than 50 bp were discarded. Then, fragments whose coverage was less than one standard deviation below the mean of the global coverage distribution were removed from the initial contact map. A total of 6,974,350 bp of sequences was removed this way. An initial contact distribution based on a simplified a polymer model [27] with three parameters was first computed for this matrix. Finally, the instaGRAAL algorithm was run using the resulting matrix and distribution.

For the *Ectocarpus* sp. genome, instaGRAAL was run at levels 4 ($n = 81$ RFs), 5 ($n = 243$ RFs), and 6 ($n = 729$ RFs). Levels 5 and 6 were only used to check for genome stability and consistency in the final chromosome count. Level 4 was used for all subsequent analyses. All runs were performed for 250 cycles. The starting fragments for the analysis were the reference genome scaffolds split into restriction fragments. The same parameters were used for the *D. herbacea* genome. The same parameters were used for the human genome, except we used level 6 instead of 4.

Correcting genome assemblies

The assembled genome generated by instaGRAAL was corrected for misassemblies using a number of simple procedures that aimed to reinstate the local structure of the input contigs of the original assembly where possible. Briefly, bins belonging to the same input contig were juxtaposed in the same relative positions as in the original assembly. Small groups of bins were preferentially moved to the location of larger groups when several such groups were present in the assembly. The orientations of sets of bins

that had been regrouped in this manner were modified so that orientation was consistent and matched that of the majority of the group, re-orientating minority bins when necessary. Both steps are illustrated in Fig. 4. Finally, fragments that had been removed during the filtering steps were reincorporated if they had been adjacent to an already integrated bin in the original assembly. The remaining sequences that could not be reintegrated this way were appended as non-integrated scaffolds.

Validation metrics

Original and other assembly metrics (Nx, GC distribution) were obtained using QUAST-LG [30]. Misassemblies were quantified using QUAST-LG with the minimap2 aligner in the backend. Ortholog completeness was computed with BUSCO (v3) [29]. Assembly completeness was also assessed with BUSCO. The evolution of genome metrics between cycles was obtained using instaGRAAL's own implementation.

Validation with the genetic map

The validation procedure with respect to linkage data was implemented as part of instaGRAAL. Briefly, the script considers a set of linkage group where regions are separated by SNP markers and a set of Hi-C scaffolds where regions are bins separated by restriction sites. It then finds best-matching pairs of linkage groups/scaffolds by counting how many of these regions overlap from one set to the other. Then, for each pair, the bins in the Hi-C scaffold are rearranged so that their order is consistent with that of the corresponding linkage group. Such rearrangements are parsimonious and try to alter as little as possible. Since there is not a one-to-one mapping from restriction sites to SNP markers, some regions in the Hi-C scaffolds are not present in the linkage groups, in which case they are left unchanged. When the Hi-C scaffolds are altered this way, as was found in the case of the raw GRAAL v3 assembly, the script acts as a correction. When the scaffolds are unchanged, as was the case with the instaGRAAL corrected v4 assembly, the script acts as a validation.

Benchmarking with other assemblers

For each genome, the 3D-DNA program was run using the run-assembly-pipeline.sh entry point script with the following options: `-i 1000 --polisher-input-size 10000 --splitter-input-size 10000`. The Hi-C data was prepared with the Juicer pipeline as recommended by 3D-DNA's documentation. The SALSA2 program was run with the `-cutoff=0` option, and misassembly correction with the `-clean=yes` option. No expected genome size was provided. The program halted after 9 iterations for *Ectocarpus* sp. and 18 iterations for *D. herbacea*. Hi-C data was prepared with the Arima pipeline as recommended by SALSA2's documentation. The similarity dot plot between corrected instaGRAAL and SALSA scaffolds was generated with minimap2.

Benchmarking with the human genome

We followed a procedure similar to the benchmark analysis detailed in [33]. Briefly, the GRCh38 reference genome was cut into 300-kb fragments. The Hi-C library generated using an Arima Genomics kit was aligned against the genome (SRA: SRR6675327). instaGRAAL was run on the resulting contact map, using the same default parameters

as for the algae genomes, except we increased the resolution level to 6 (from 4). The similarity dot plot between instaGRAAL and SALSA scaffolds was generated with `mini-map2`, with the options `-DP -k19 -w19 -m200`.

Software tool requirements

The instaGRAAL software is written in Python 3 and uses CUDA for the computationally intensive parts. It requires a working installation of CUDA with the `pycuda` library. CUDA is a proprietary parallel computing framework developed by NVIDIA and requires a NVIDIA graphics card. The scaffolder also requires a number of common scientific Python libraries specified in its documentation. The instaGRAAL website lists computer systems onto which the program was successfully installed and run.

Supplementary information

Supplementary information accompanies this paper at <https://doi.org/10.1186/s13059-020-02041-z>.

Additional file 1. Supplementary tables and figures.

Additional file 2. Review history.

Acknowledgements

We thank our colleagues from the team, especially Cyril Matthey-Doret, as well as Hugo Darras, Heather Marlow, Francois Spitz, Jitendra Narayan, Jean-François Flot, Jérémy Gauthier, Jean-Michel Drezen, and all Github users and contributors for valuable feedback and comments.

Review history

The review history is available as Additional file 2.

Peer review information

Andrew Cosgrove was the primary editor on this article and managed its editorial process and peer review in collaboration with the rest of the editorial team.

Authors' contributions

LB rewrote and updated the GRAAL program originally designed by HMN, CZ, and RK. MM and AC performed the experiments. LB and NG performed and ran the scaffoldings. LB, NG, and RK analyzed the assemblies, with contributions from AC, KA, LS, JMC, and SMC. LM and RK wrote the manuscript, with contributions from NG, MM, JMC, MC, and SMC. LB, MM, SMC, and RK conceived the study. The authors read and approved the final manuscript.

Authors' information

Twitter handle: @rkozul (Romain Koszul).

Funding

This research was supported by funding to R.K. and S.M.C. from the European Research Council under the Horizon 2020 Program (ERC grant agreements 260822 and 638240, respectively). This project has also received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 764840.

Availability of data and materials

The datasets generated and analyzed in the present work are available in the SRA repository, SRR8550777 [48]. The instaGRAAL software and its documentation are freely available under the GPL-3.0 license at <https://github.com/koszulab/instaGRAAL> [49]. Assemblies, contact maps, and relevant materials for the reproduction of the main results and figures are available at https://github.com/koszulab/ectocarpus_scripts [50].

Ethics approval and consent to participate

No ethical approval was required.

Competing interests

instaGRAAL is owned by the Institut Pasteur. The entire program and its source code are freely available under a free software license.

Author details

¹Institut Pasteur, Unité Régulation Spatiale des Génomes, CNRS, UMR 3525, C3BI USR 3756, F-75015 Paris, France. ²Sorbonne Université, Collège Doctoral, F-75005 Paris, France. ³Evolutionary Biology & Ecology, Université Libre de Bruxelles, 1050 Brussels, Belgium. ⁴Sorbonne Université, Laboratory of Integrative Biology of Marine Models, Algal

Genetics, UMR 8227, Roscoff, France. ⁵Present Address: Université de Strasbourg, INRA, SVQV UMR-A 1131, Colmar, France. ⁶Institut Pasteur, Center of Bioinformatics, Biostatistics and Integrative Biology (C3BI), USR3756, CNRS, Paris, France. ⁷Department of Plant Biotechnology and Bioinformatics, Ghent University, B-9052 Ghent Ghent, Belgium. ⁸VIB Center for Plant Systems Biology, Technologiepark 927, B-9052 Ghent, Belgium. ⁹Institut Pasteur, Imaging and Modeling Unit, CNRS, UMR 3691, C3BI USR 3756, F-75015 Paris, France.

Received: 31 July 2019 Accepted: 11 May 2020

Published online: 18 June 2020

References

1. Khan AR, Pervez MT, Babar ME, Naveed N, Shoaib M. A comprehensive study of de novo genome assemblers: current challenges and future prospective. *Evol Bioinforma Online*. 2018;14. Available from: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5826002/>. Accessed 12 Dec 2019.
2. Sedlazeck FJ, Lee H, Darby CA, Schatz MC. Piercing the dark matter: bioinformatics of long-range sequencing and mapping. *Nat Rev Genet*. 2018;19:329.
3. Rice ES, Green RE. New approaches for genome assembly and scaffolding. *Annu Rev AnimBiosci*. 2019;7:17–40.
4. Alkan C, Coe BP, Eichler EE. Genome structural variation discovery and genotyping. *Nat Rev Genet*. 2011;12:363–76.
5. Salzberg SL, Phillippy AM, Zimin A, Puiu D, Magoc T, Koren S, et al. GAGE: a critical evaluation of genome assemblies and assembly algorithms. *Genome Res*. 2012;22:557–67.
6. Bradnam KR, Fass JN, Alexandrov A, Baranay P, Bechner M, Birol I, et al. Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species. *GigaScience* [Internet]. 2013 [cited 2018 Nov 2];2. Available from: <https://academic.oup.com/gigascience/article/2/1/2047-217X-2-10/2656129>.
7. Alhakami H, Mirebrahim H, Lonardi S. A comparative evaluation of genome assembly reconciliation tools. *Genome Biol*. 2017;18:93.
8. Lieberman-Aiden E, van Berkum NL, Williams L, Imakaev M, Ragoczy T, Telling A, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*. 2009;326:289–93.
9. Dekker J, Rippe K, Dekker M, Kleckner N. Capturing chromosome conformation. *Science*. 2002;295:1306–11.
10. Flot J-F, Marie-Nelly H, Koszul R. Contact genomics: scaffolding and phasing (meta) genomes using chromosome 3D physical signatures. *FEBS Lett*. 2015;589:2966–74.
11. Burton JN, Adey A, Patwardhan RP, Qiu R, Kitzman JO, Shendure J. Chromosome-scale scaffolding of de novo genome assemblies based on chromatin interactions. *Nat Biotechnol*. 2013;31:1119–25.
12. Kaplan N, Dekker J. High-throughput genome scaffolding from in vivo DNA interaction frequency. *Nat Biotechnol*. 2013;31:1143–7.
13. Marie-Nelly H, Marbouty M, Cournac A, Flot J-F, Liti G, Parodi DP, et al. High-quality genome (re) assembly using chromosomal contact data. *Nat Commun*. 2014;5:5695.
14. Marie-Nelly H. A probabilistic approach for genome assembly from high-throughput chromosome conformation capture data [Doctoral dissertation]. Université Pierre et Marie Curie – Paris 6. 2013.
15. Marbouty M, Cournac A, Flot J-F, Marie-Nelly H, Mozziconacci J, Koszul R. Metagenomic chromosome conformation capture (meta3C) unveils the diversity of chromosome organization in microorganisms. *eLife*. 2014;3:e03318.
16. Bickhart DM, Rosen BD, Koren S, Sayre BL, Hastie AR, Chan S, et al. Single-molecule sequencing and chromatin conformation capture enable *de novo* reference assembly of the domestic goat genome. *Nat Genet*. 2017;49:643–50.
17. Dudchenko O, Batra SS, Omer AD, Nyquist SK, Hoeger M, Durand NC, et al. De novo assembly of the *Aedes aegypti* genome using Hi-C yields chromosome-length scaffolds. *Science*. 2017;356:92–5.
18. Putnam NH, O’Connell BL, Stites JC, Rice BJ, Blanchette M, Calef R, et al. Chromosome-scale shotgun assembly using an in vitro method for long-range linkage. *Genome Res*. 2016;26:342–50.
19. Marbouty M, Baudry L, Cournac A, Koszul R. Scaffolding bacterial genomes and probing host-virus interactions in gut microbiome by proximity ligation (chromosome capture) assay. *Sci Adv*. 2017;3:e1602105.
20. Jourdain E, Baudry L, Poggi-Parodi D, Vicq Y, Koszul R, Margeot A, et al. Proximity ligation scaffolding and comparison of two *Trichoderma reesei* strains genomes. *BiotechnolBiofuels*. 2017;10:151.
21. Cormier A, Avia K, Sterck L, Derrien T, Wucher V, Andres G, et al. Re-annotation, improved large-scale assembly and establishment of a catalogue of noncoding loci for the genome of the model brown alga *Ectocarpus*. *New Phytol*. 2017;214:219–32.
22. Cock JM, Sterck L, Rouzé P, Scornet D, Allen AE, Amoutzias G, et al. The *Ectocarpus* genome and the independent evolution of multicellularity in brown algae. *Nature*. 2010;465:617–21.
23. Coelho SM, Godfroy O, Arun A, Corguillé GL, Peters AF, Cock JM. OUROBOROS is a master regulator of the gametophyte to sporophyte life cycle transition in the brown alga *Ectocarpus*. *Proc Natl Acad Sci*. 2011;108:11518–23.
24. Ahmed S, Cock JM, Pessia E, Luthringer R, Cormier A, Robuchon M, et al. A haploid system of sex determination in the brown alga *Ectocarpus* sp. *Curr Biol*. 2014;24:1945–57.
25. Arun A, Coelho SM, Peters AF, Bourdareau S, Pérès L, Scornet D, et al. Convergent recruitment of TALE homeodomain life cycle regulators to direct sporophyte development in land plants and brown algae. McCormick S, Hardtke CS, editors. *eLife*. 2019;8:e43101.
26. Avia K, Coelho SM, Montecinos GJ, Cormier A, Lerck F, Mauger S, et al. High-density genetic map and identification of QTLs for responses to temperature and salinity stresses in the model brown alga *Ectocarpus*. *Sci Rep*. 2017;7:43241.
27. Rippe K. Making contacts on a nucleic acid polymer. *Trends Biochem Sci*. 2001;26:733–40.
28. Müller DG. Untersuchungen zur Entwicklungsgeschichte der Braunalge *Ectocarpus siliculosus* Aus Neapel. *Planta*. 1966;68:57–68.
29. Simão FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics*. 2015;31:3210–2.
30. Mikheenko A, Pribelski A, Saveliev V, Antipov D, Gurevich A. Versatile genome assembly evaluation with QAST-LG. *Bioinformatics*. 2018;34:i142–50.

31. Zimin AV, Marçais G, Puiu D, Roberts M, Salzberg SL, Yorke JA. The MaSuRCA genome assembler. *Bioinformatics*. 2013; 29:2669–77.
32. Ramirez ME, Müller DG, Peters AF. Life history and taxonomy of two populations of ligulate Desmarestia (Phaeophyceae) from Chile. *Can J Bot*. 1986;64:2948–54.
33. Ghurye J, Rhie A, Walenz BP, Schmitt A, Selvaraj S, Pop M, et al. Integrating Hi-C links with assembly graphs for chromosome-scale assembly. *PLoS Comput Biol*. 2019;15:e1007273.
34. Cournac A, Marie-Nelly H, Marbouty M, Koszul R, Mozziconacci J. Normalization of a chromosomal contact map. *BMC Genomics*. 2012;13:436.
35. Imakaev M, Fudenberg G, McCord RP, Naumova N, Goloborodko A, Lajoie BR, et al. Iterative correction of Hi-C data reveals hallmarks of chromosome organization. *Nat Methods*. 2012;9:999–1003.
36. Yaffe E, Tanay A. Probabilistic modeling of Hi-C contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nat Genet*. 2011;43:1059–65.
37. Muller H, Scolari VF, Agjer N, Piazza A, Thierry A, Mercy G, et al. Characterizing meiotic chromosomes' structure and pairing using a designer sequence optimized for Hi-C. *Mol Syst Biol*. 2018;14:e8293.
38. Consortium (IWGSC) TIWGS, Investigators IR principal, Appels R, Eversole K, Feuillet C, Keller B, et al. Shifting the limits in wheat research and breeding using a fully annotated reference genome. *Science*. 2018;361:eaar7191.
39. Aganezov SS, Alekseyev MA. CAMSA: a tool for comparative analysis and merging of scaffold assemblies. *BMC Bioinformatics*. 2017;18:496.
40. Wences AH, Schatz MC. Metassembler: merging and optimizing de novo genome assemblies. *Genome Biol*. 2015;16:207.
41. English AC, Richards S, Han Y, Wang M, Vee V, Qu J, et al. Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology. *PLoS One*. 2012;7:e47768.
42. Vaser R, Sović I, Nagarajan N, Šikić M. Fast and accurate de novo genome assembly from long uncorrected reads. *Genome Res*. 2017;27:737–46.
43. Walker BJ, Abeel T, Shea T, Priest M, Abouelliel A, Sakthikumar S, et al. Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. *PLoS One*. 2014;9:e112963.
44. Kundu R, Casey J, Sung W-K. HyPo: super fast accurate polisher for long read genome assemblies. *bioRxiv*. 2019;2019.12.19.882506.
45. Antipov D, Korobeynikov A, McLean JS, Pevzner PA. hybridSPAdes: an algorithm for hybrid assembly of short and long reads. *Bioinformatics*. 2016;32:1009–15.
46. Miller JR, Zhou P, Mudge J, Gurtowski J, Lee H, Ramaraj T, et al. Hybrid assembly with long and short reads improves discovery of gene family expansions. *BMC Genomics*. 2017;18:541.
47. Lazar-Stefanita L, Scolari VF, Mercy G, Muller H, Guérin TM, Thierry A, et al. Cohesins and condensins orchestrate the 4D dynamics of yeast chromosomes during the cell cycle. *EMBO J*. 2017;36(18):2684–97.
48. Baudry L, Guiguelmoni N, Marie-Nelly H, Cormier A, Marbouty M, Avia K, Mie YL, Godfroy O, Sterck L, Cock JM, Zimmer C, Coelho SM, Koszul R. Large genome reassembly based on Hi-C data, continuation of GRAAL. *Sequence Read Archive Datasets*. 2020. <https://www.ncbi.nlm.nih.gov/sra/?term=SRR8550777>.
49. Lyam Baudry, Nadège Guiguelmoni, Hervé Marie-Nelly, Romain Koszul. Large genome reassembly based on Hi-C data, continuation of GRAAL. 2019. <https://github.com/koszullab/instagraal> <https://doi.org/10.5281/zenodo.3753965>. Accessed 16 Apr 2020.
50. Lyam Baudry, Nadège Guiguelmoni, Alexandre Cormier, Komlan Avia, Mark Cock, Susana Coelho, Romain Koszul. Large genome reassembly based on Hi-C data, continuation of GRAAL. 2019. https://github.com/koszullab/ectocarpus_scripts <https://doi.org/10.5281/zenodo.3753973>. Accessed 16 Apr 2020.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions



Table S1: Example of a sparse matrix.

id_frag_a	id_frag_b	n_contacts
0	0	1368
0	1	21
0	2	7
0	3	3
0	4	5
0	7	5
0	8	1
0	9	1
0	12	2
0	15	1
0	22	1
0	23	1
0	26	1
0	27	1
0	33	2
0	36	2
0	37	1
0	51	1
0	69	1
0	74	2
0	76	1
0	97	1
0	99	1
0	107	1

Table S2: Comparison of the integrated sequences between the different assemblies and the v1 assembly for *Ectocarpus* sp.

	v1 assembly	linkage group v2 assembly	corrected instaGRAAL v4 assembly
Scaffolds integrated into linkage groups (out of 1561)	325	531	793
Percent sequence data integrated into linkage groups	70.10%	90.50%	96.80%
Integrated oriented scaffolds in the linkage groups	12%	49%	100%
Number of linkage groups	34	28	27

Table S3: Correspondences between instaGRAAL super scaffolds and linkage groups from the v2 assembly for the *Ectocarpus* sp. genome.

instaGRAAL v4 assembly	Linkage group v2 assembly
1	1
2	21
3	4 & 28
4	5
5	13
6	6
7	12
8	7
9	27
10	26
11	3
12	2
13	8
14	14
15	10
16	11
17	19
18	16
19	9
20	15
21	18
22	20
23	24
24	23
25	17
26	25
27	22

Table S4: Metrics of *Desmarestia herbacea* assemblies using three different programs.

	<i>De novo</i> original assembly	3D-DNA	SALSA2	instaGRAAL
N50 (bp)	184,092	175,000	12,780,148	12,444,485
L50	697	545	11	17
Contig count	7,743	5,385	4,827	4,304
BUSCO %	72.6	70.7	73.6	73.0

Table S5: Performance of GRAAL and instaGRAAL at scaffolding the *Ectocarpus* sp. genome.

	GRAAL	instaGRAAL
Peak memory load (Gb)	2.5	1.1
Memory used in graphic card	113 (Mb)	11
Per-cycle runtime (avg. over 20 min)	13	4

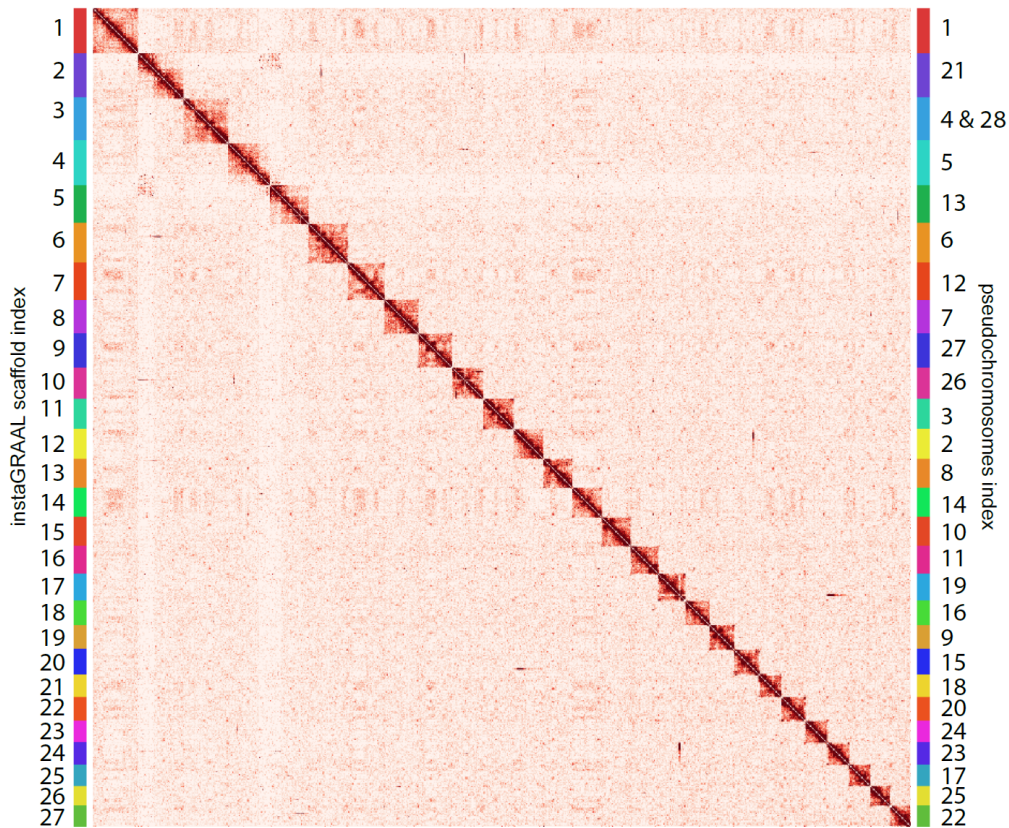


Figure S1: Normalized contact map of the *Ectocarpus* sp. genome scaffolded using instaGRAAL (bin = 200 kb). The colour scale represents the normalized interaction frequencies. No large-scale rearrangements are clearly apparent in the interchromosomal contacts. On the right the linkage groups indices from the v2 assembly are indicated.

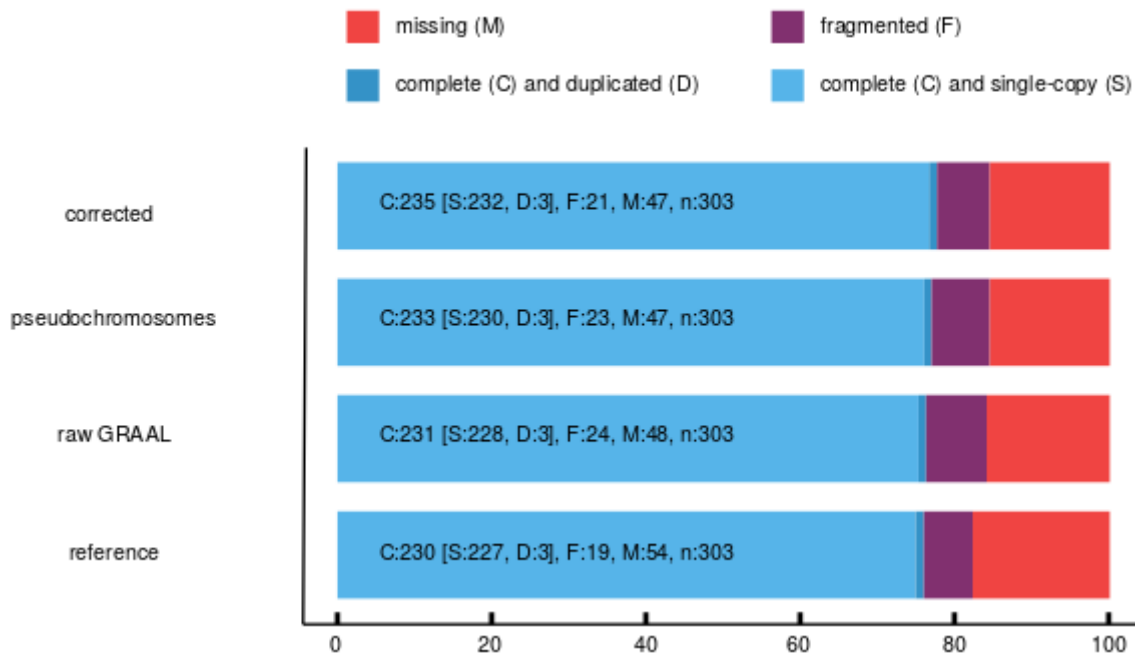


Figure S2: Estimates of BUSCO completeness for the three *Ectocarpus* sp. assemblies and the reference genome v1 assembly.

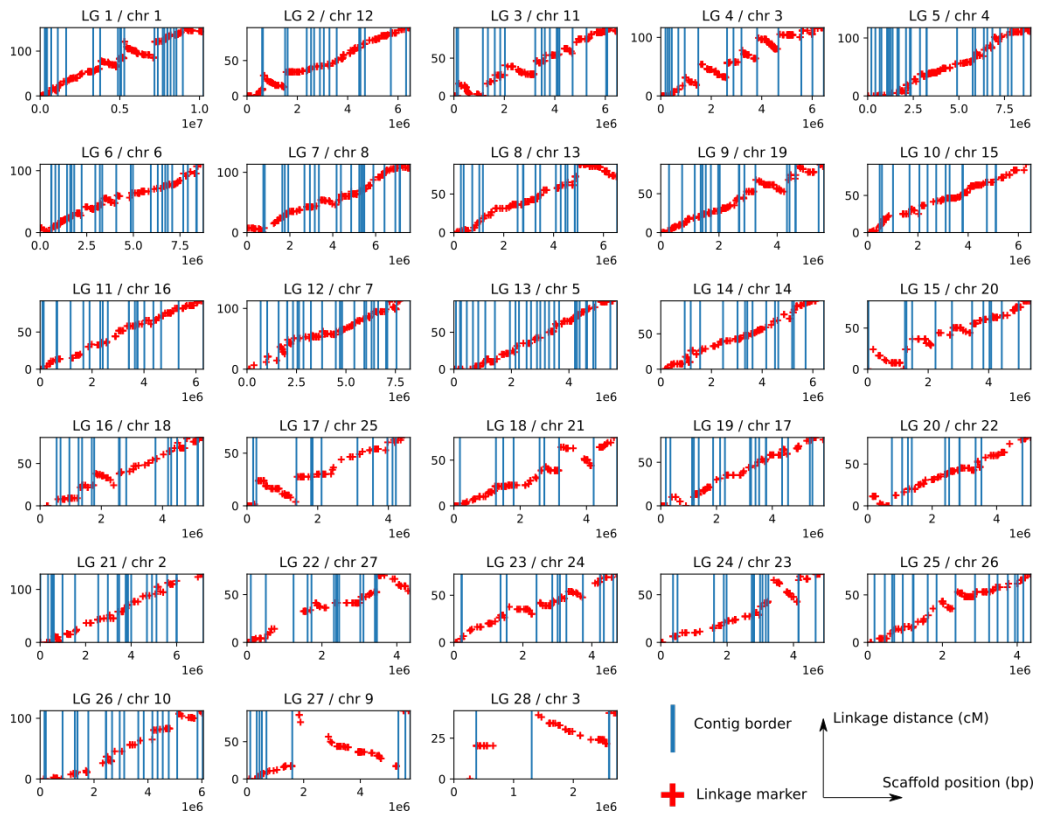


Figure S3: Linkage markers vs. scaffold positions for all linkage groups/chromosomes (chromosome 3 is made up of linkage groups 4 and 28). The initial contig borders within each chromosome have been underlined. Linkage marker positions are always monotonous (only increasing, or only decreasing) within an initial contig.

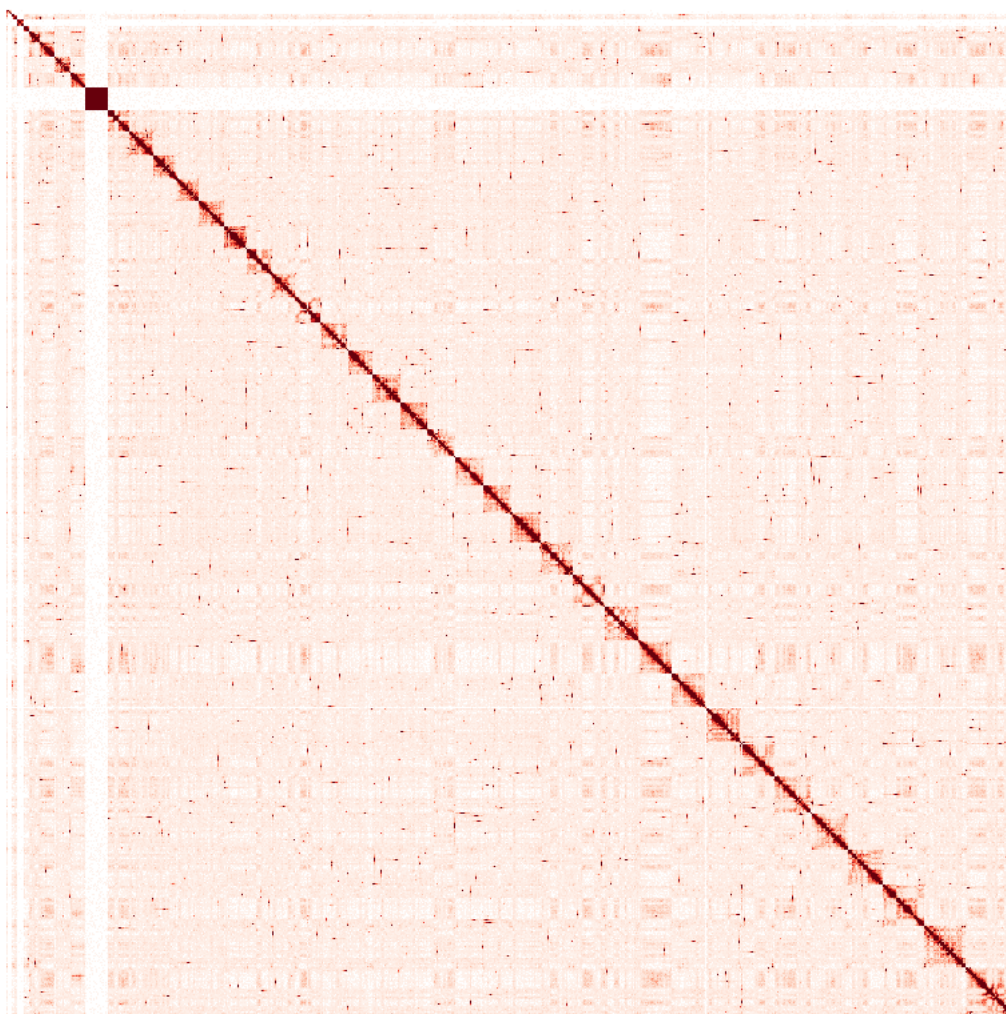


Figure S4: The 40 main scaffolds of *Desmarestia herbacea* after instaGRAAL scaffolding.

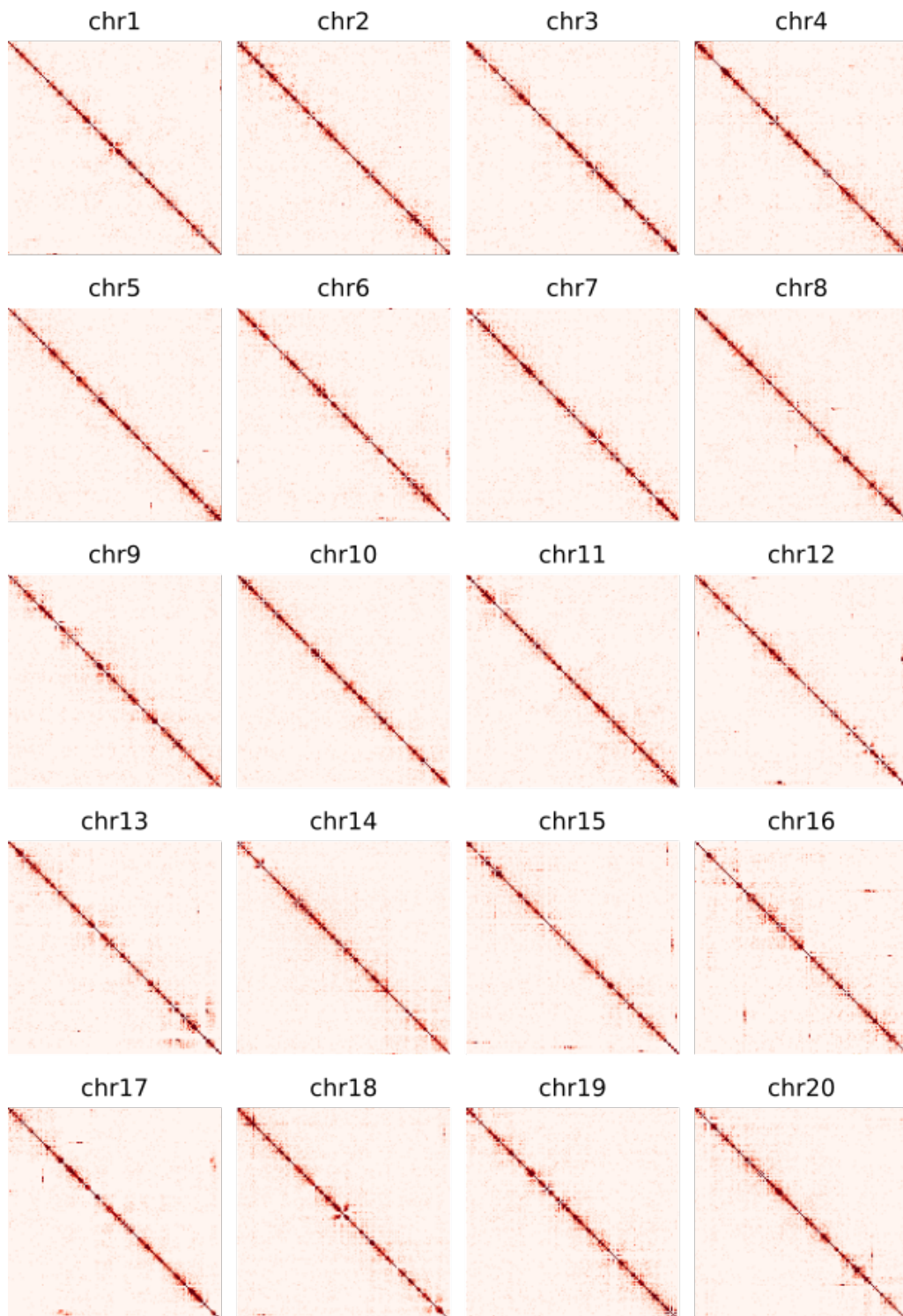


Figure S5: Contact maps of the first twenty newly formed scaffolds/putative chromosomes of *Desmarestia herbacea*, generated after scaffolding at a 20-kb resolution.

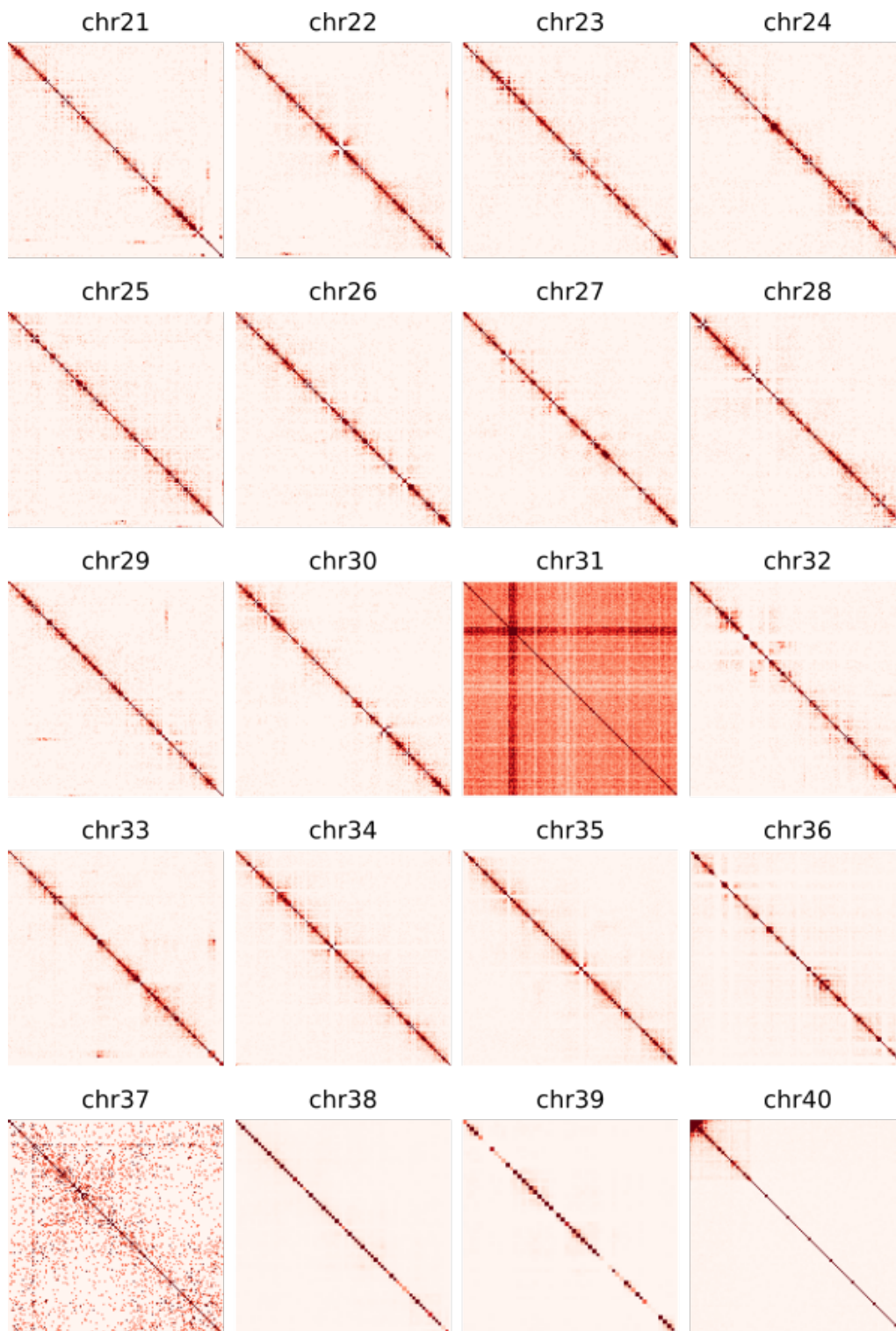


Figure S6: The last twenty newly formed scaffolds/putative chromosomes of *Desmarestia herbacea* post-scaffolding at a 20-kb resolution.

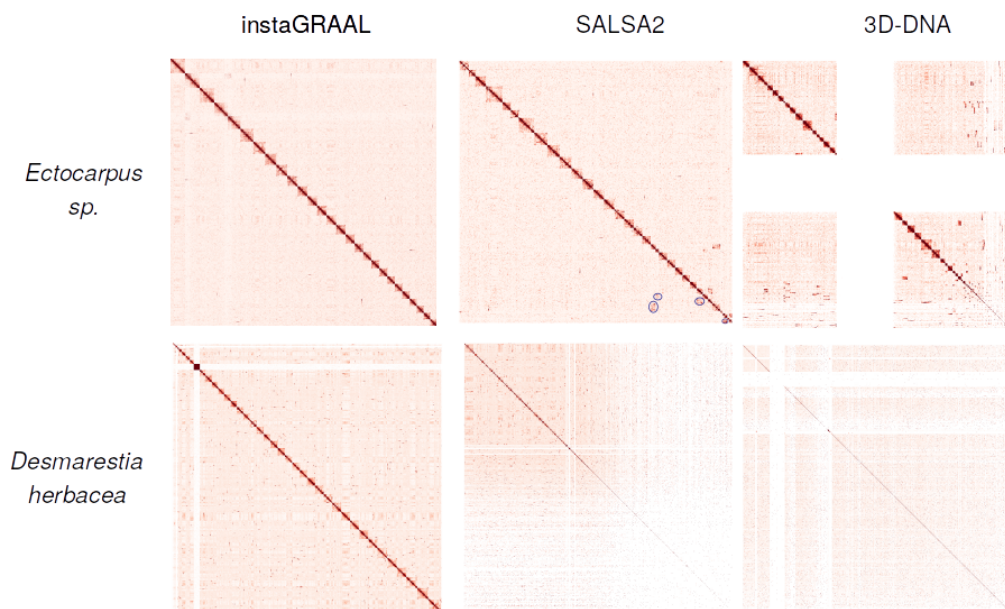


Figure S7: Normalized contact map of the *Ectocarpus* sp. genome scaffolded using instaGRAAL (bin = 200 kb). The colour scale represents the normalized interaction frequencies. No large-scale rearrangements are clearly apparent in the interchromosomal contacts. On the right the linkage groups indices from the v2 assembly are indicated.

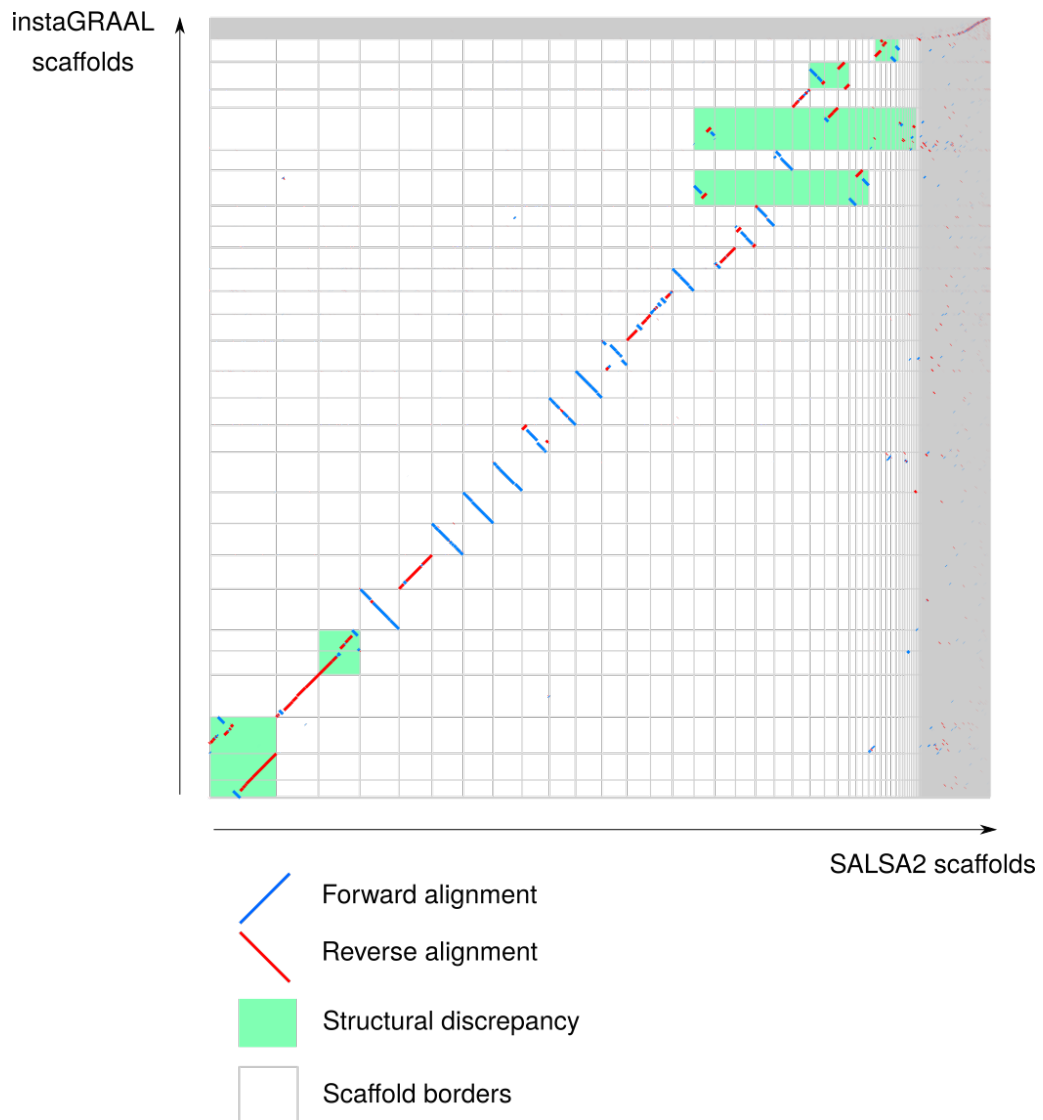


Figure S8: Similarity dotplot of the SALSA2 vs. instaGRAAL 27 scaffolds for *Ectocarpus* sp. large-scale structural discrepancies have been underlined in green. The contact maps suggest instaGRAAL solutions are more likely.

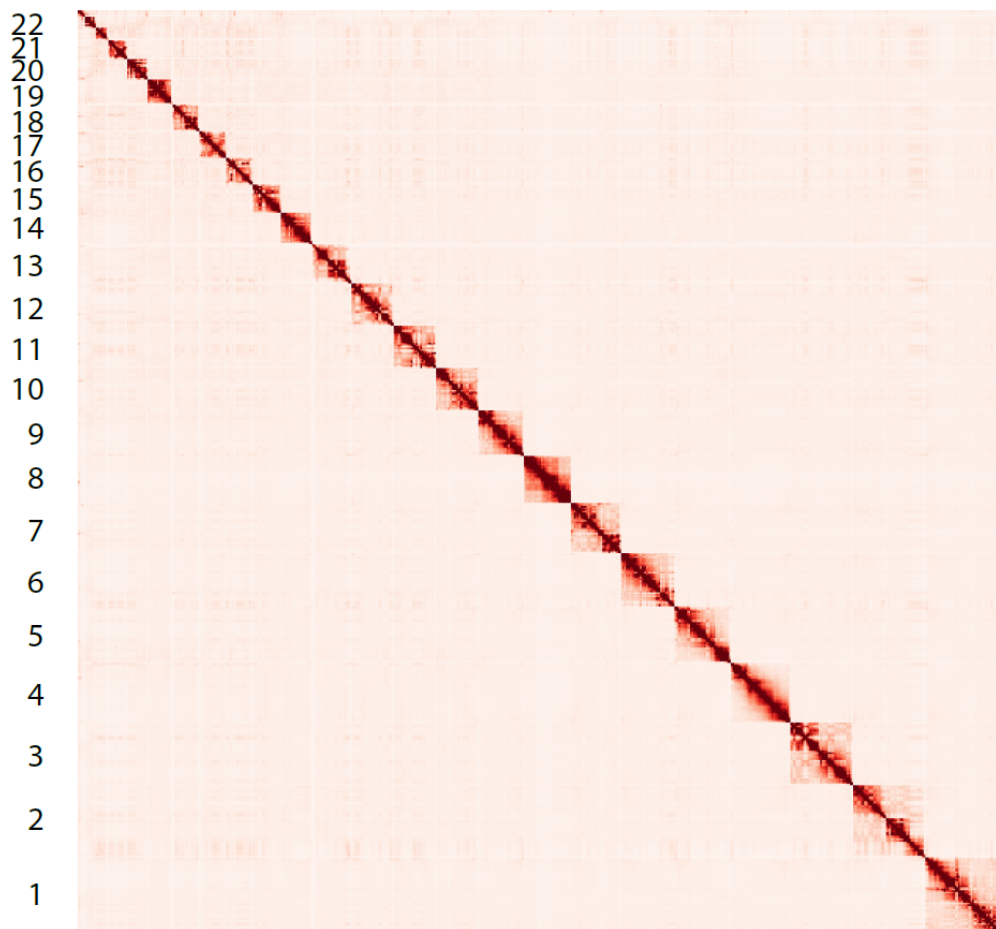


Figure S9: Contact map of the *Homo sapiens* genome, fragmented in 300 kb sequences, after scaffolding with instaGRAAL, at 5-Mb resolution.

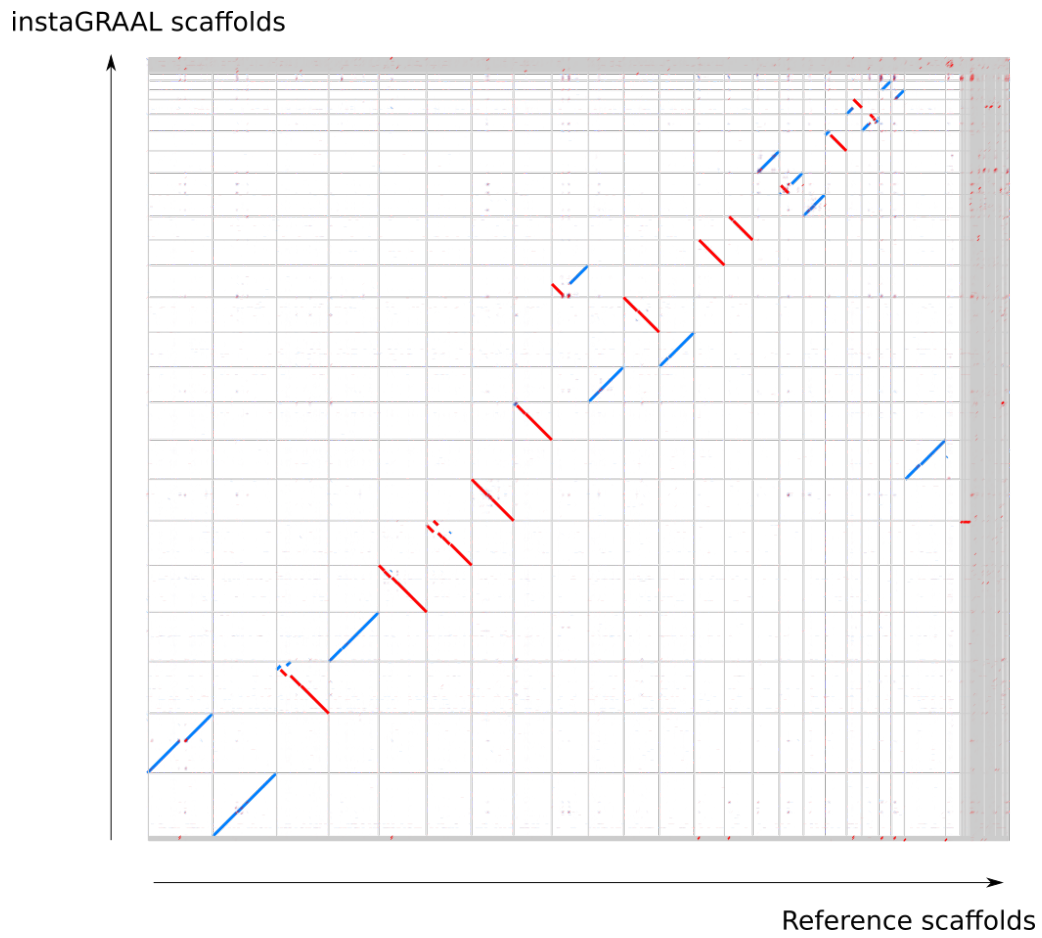


Figure S10: Similarity dotplot of the instaGRAAL vs. reference scaffolds for the GRCh38 human genome. Relocations are visible but the one-to-one mapping between the 23 first scaffolds is preserved.

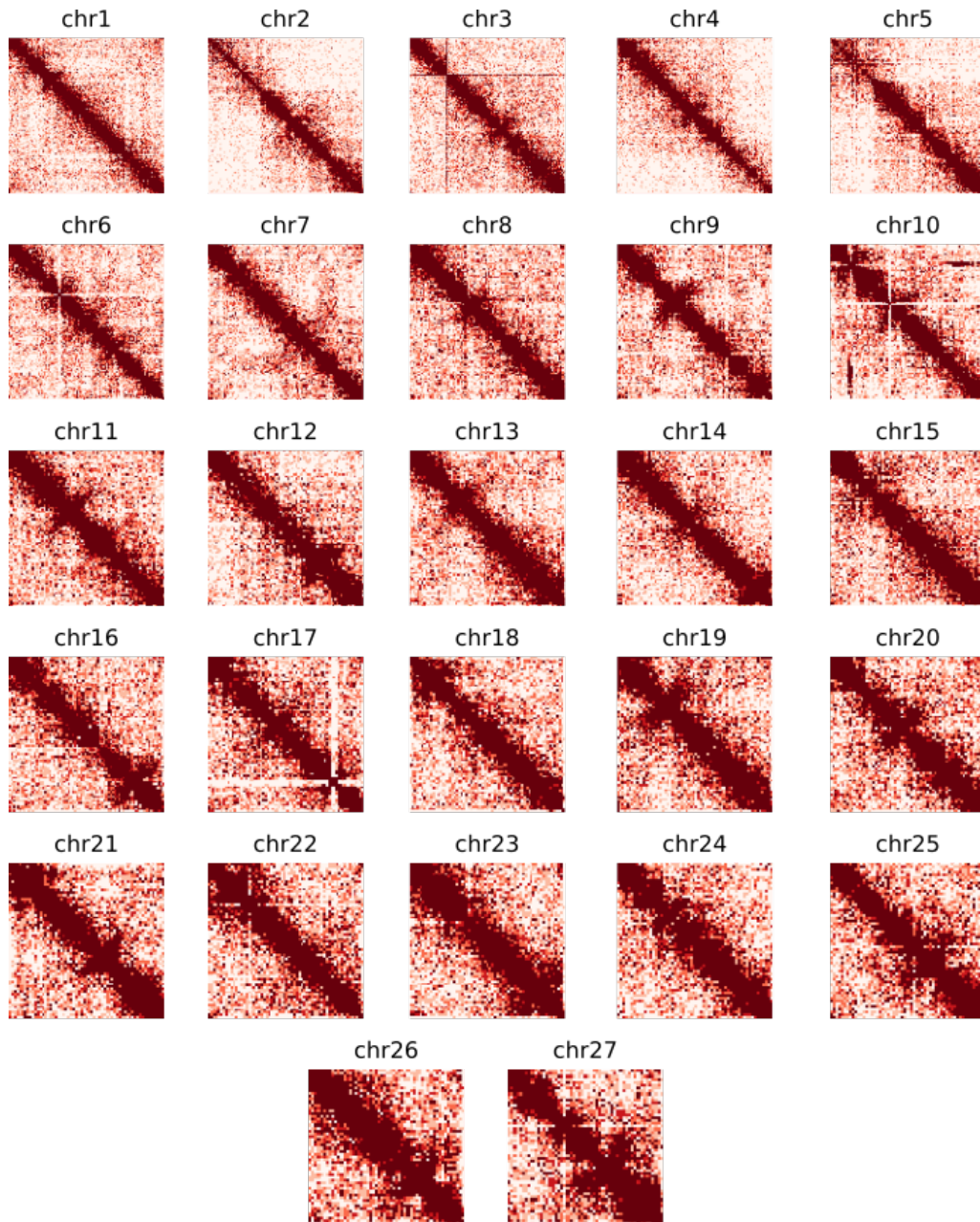


Figure S11: All 27 newly formed scaffolds/putative chromosomes of *Ectocarpus* sp. post-scaffolding at a 50-kb resolution. Centromere patterns are clearly apparent in all chromosomes, but some errors (potentially due to mapping issues) linger, such as chromosome 10 or 17.

Chapter 5

Hi-C scaffolding of the bdelloid rotifer

Adineta vaga

Bdelloid rotifers have been drawing interest due to their suspected unusual ancient asexuality. A first diploid assembly of *Adineta vaga* was published in 2013 [264], with a total length of 218 Mb and a N50 of 260 kb. The genome was described at the time as "incompatible with conventional meiosis", for the surprising non-colinear structure of homologous sequences. In the following paper, the genome of *Adineta vaga* was assembled *de novo* using PacBio CLR, Nanopore, Illumina and Hi-C reads. Three assemblies were produced: a collapsed haploid assembly using all types of reads; and two diploid assemblies, one using PacBio CLR obtained with FALCON and FALCON-Unzip, and the second one using Illumina reads with Bwise. All these assemblies were scaffolded using Hi-C by the program instaGRAAL [186], and converged to 6 haploid chromosomes (collapsed assembly) or 12 phased chromosomes (FALCON-Unzip and Bwise assemblies). These results show that the genome of *Adineta vaga* is: diploid, with 6 pairs of chromosomes; a paleotetraploid, as it has homoeologous, colinear chromosomes (pairs 1, 2 and 3 are homoeologous to pairs 4, 5, 6 respectively); and thus compatible with meiosis.

I contributed to this study in the assembly and Hi-C scaffolding of the collapsed haploid assembly.

Chromosome-level genome assembly reveals homologous chromosomes and recombination in asexual rotifer *Adineta vaga*

Paul Simion,^{1+*} Jitendra Narayan,¹⁺ Antoine Houtain¹ Alessandro Derzelle¹
Lyam Baudry^{2,3} Emilien Nicolas^{1,4} Rohan Arora^{1,4} Marie Cariou⁵
Corinne Cruaud⁶ Florence Rodriguez Gaudray⁷ Clément Gilbert⁸
Nadège Guiglielmoni⁷ Boris Hespeels¹ Djampa KL Kozlowski⁹
Karine Labadie⁶ Antoine Limasset¹⁰ Marc Lirós¹¹ Martial Marbouty²
Matthieu Terwagne¹ Julie Virgo¹ Richard Cordaux¹²
Etienne GJ Danchin⁹ Bernard Hallet¹³ Romain Koszul²
Thomas Lenormand¹⁴ Jean-François Flot^{7,15*} Karine Van Doninck^{1,4*}

¹Université de Namur, LEGE, URBE, Namur, 5000, Belgium

²Institut Pasteur, Unité Régulation Spatiale des Génomes, UMR 3525, CNRS, Paris, F-75015, France

³Sorbonne Université, Collège Doctoral, F-75005 Paris, France

⁴Université libre de Bruxelles (ULB), Molecular Biology and Evolution, Brussels, 1050, Belgium

⁵CIRI, Centre International de Recherche en Infectiologie, Univ Lyon, Inserm, U1111,

Université Claude Bernard Lyon 1, CNRS, UMR5308, ENS de Lyon, F-69007, Lyon, France.

⁶Genoscope, Institut François Jacob, CEA, CNRS, Univ Evry, Université Paris-Saclay, 91057 Evry, France

⁷Université libre de Bruxelles (ULB), Evolutionary Biology and Ecology, Brussels, 1050, Belgium

⁸Université Paris-Saclay, CNRS, IRD, UMR Évolution, Génomes, Comportement et Écologie, 91198, Gif-sur-Yvette, France

⁹INRAE, Université Côte-d'Azur, CNRS, Institut Sophia Agrobiotech, Sophia Antipolis, 06903, France

¹⁰Université de Lille, CNRS, UMR 9189 - CRIStAL, 59655 Villeneuve-d'Ascq, France

¹¹Institut d'Investigació Biomèdica de Girona, Malalties Digestives i Microbiota, 17190 Salt, Spain

¹²Université de Poitiers, UMR CNRS 7267 Ecologie et Biologie des interactions, 5 rue Albert Turpain, 86073 Poitiers, France

¹³Université Catholique de Louvain (UCLouvain), LIBST, Croix du Sud 4/5, Louvain-la-Neuve, 1348, Belgium

¹⁴CEFE, Univ Montpellier, CNRS, Univ Paul Valéry Montpellier 3, EPHE, IRD, Montpellier, France

¹⁵Interuniversity Institute of Bioinformatics in Brussels - (IB)², Brussels, 1050, Belgium

+ these authors contributed equally to this work

*To whom correspondence should be addressed; E-mail: karine.vandinck@unamur.be, jflot@ulb.ac.be, polo.simion@gmail.com

1 **Bdelloid rotifers are notorious because they represent a speciose and ancient**
2 **clade comprising only asexual lineages. In addition, most bdelloid species**
3 **withstand complete desiccation and ionizing radiation, being able to repair**
4 **their highly fragmented DNA. Producing a well-assembled reference genome**
5 **is a critical step to unlock the understanding of the effects of long-term asexu-**
6 **ality and DNA breakage on genome evolution. Here, we present the first high-**
7 **quality chromosome-level genome assemblies for the bdelloid species *Adineta***
8 ***vaga*, composed of six pairs of homologous chromosomes (i.e. diploid), with**
9 **a footprint of paleotetraploidy. The observed large-scale losses of heterozy-**
10 **gosity are signatures of recombination between homologous chromosomes, ei-**
11 **ther during mitotic DNA double-strand break repair or when resolving pro-**
12 **grammed DNA breaks during a modified meiosis. Dynamic subtelomeric re-**
13 **gions harbor more structural diversity (e.g. chromosome rearrangements,**
14 **transposable elements, haplotypic divergence). Our results trigger the reap-**
15 **praisal of potential meiotic processes in bdelloid rotifers and help unravel their**
16 **long-term asexual evolutionary success.**

17 **Introduction**

18 Sexual reproduction and recombination are prevalent throughout the eukaryotes, despite the
19 substantial evolutionary costs such as the two-fold cost of males or the cost of recombination
20 that breaks up co-adapted gene combinations (1, 2)). Several eukaryotic species appear to have
21 evolved adaptations that reduce these costs of males, for example by producing males only fac-
22 ultatively as in cyclical parthenogens (e.g. *Brachionus plicatilis* (3)), or by retaining a modified
23 meiosis rescuing diploidy without fertilization by males (e.g. *Diploscapter pachys* (4)). Very
24 few however appear to have renounced to sex and recombination completely, in which males

25 and meiosis are abolished. Theory predicts that in the absence of recombination during meio-
26 sis, the physical linkage among loci reduces the effectiveness of selection upon individual loci,
27 resulting in a decreased rate of adaptation and the accumulation of mildly deleterious muta-
28 tions (5). Obligate asexuals are therefore suitable model systems to gain general insights into
29 the long term consequences of the lack of recombination and sexual reproduction.

30 Bdelloid rotifers are notorious ancient asexual animals. Indeed, the longevity (>60 My) of
31 the bdelloid rotifer clade and their diversity (>400 morphospecies) challenge the expectation
32 that obligatory asexual animal lineages, in which recombination and outcrossing are absent,
33 are evolutionary dead-ends. Historical observations (or lack thereof) had produced a consen-
34 sus that bdelloid rotifers do not produce male or hermaphrodite individuals (6), that they are
35 strictly parthenogenetic without any meiosis (7, 8) and that the initial description of the struc-
36 ture of *Adineta vaga* genome, lacking colinear homologous scaffolds, was irreconcilable with
37 meiosis (9). A draft genome assembly of the closely-related bdelloid species *Adineta ricciae*
38 found colinearity between homologous regions but could not verify it at chromosome-scale (10),
39 which was also the case for previous studies based on a handful of genomic regions (11–13).
40 The presence or the absence of an ameiotic structure in bdelloids therefore remained unresolved
41 and a chromosome-scale assembly appeared critical.

42 Besides its asexual evolution, the bdelloid rotifer *A. vaga* also became a model species for its
43 extreme resistance to desiccation, freezing and ionizing radiation, with implications for space
44 research (14, 15). Both prolonged desiccation, encountered in their ephemeral limno-terrestrial
45 habitats, and ionizing radiation induce oxidative stress and massive genome breakage that *A.*
46 *vaga* seems to handle, maintaining high survival and fecundity rates while efficiently repairing
47 DNA damage (15–17). Maintaining such long-term survival and genome stability following
48 DNA fragmentation likely requires the use of homologous recombination (HR) at least in the
49 germ cells. Given the supposed absence of homologous chromosomes in *A. vaga* (9), the exact

50 nature of their double-strand break (DSB) repair mechanism remains elusive.

51 Recent studies have provided evidence for recombination in bdelloid rotifers. These in-
52 clude a drop of linkage disequilibrium (LD) with increasing distance between genomic loci in
53 *A. vaga* (13), signatures of gene conversion (9, 12), heterozygosity levels within the range of
54 those reported for sexual metazoans (9, 10, 18), and reports of allele sharing between bdelloid
55 individuals from the wild (13, 19–21). While recombination likely takes place in bdelloid ro-
56 tifers, its underlying mechanisms remain unknown. Recombination might theoretically occur
57 in a mitotic or meiotic cellular context, involve short genomic regions or canonical chromo-
58 some pairing, and take place between homologous or non-homologous loci (i.e. ectopic). The
59 interpretation of these recombination events have yet to be reconciled with the long-standing
60 apparent absence of males and meiosis in bdelloid rotifers (6) and to account for their ubiquity
61 in semi-terrestrial habitats where frequent desiccation occurs, inducing DNA DSBs (14, 15, 17).

62 Here, we present a high quality chromosome-level genome assembly of *A. vaga*. This new
63 genome is pivotal to tackle these contradictions between its putative ameiotic structure and the
64 footprints of recombination, possibly associated to DSB repair and desiccation. We combined
65 the use of short reads (Illumina), long reads (ONT and PacBio) and chromosome conformation
66 capture data (Hi-C) with three assembly methods, to successfully assemble *A. vaga* genome.
67 We provide the first telomere-to-telomere assemblies of a parthenogenetic lineage, both haploid
68 and phased, paving the way to study genome evolution in an asexual clade. Using a newly devel-
69 oped and publicly available tool, Alienomics, we annotated candidate horizontal gene transfers
70 (HGTc) and confirmed that *A. vaga* possesses the highest number of HGTc across all animals.
71 Interestingly, *A. vaga* has a diploid genome made of six pairs of homologous chromosomes,
72 refuting the ameiotic structure previously described for this genome (9) and challenging the
73 complete absence of meiosis in one of the most striking asexual animal clade. In addition, by
74 observing large tracks of heterozygosity losses (LOH), we show that large-scale recombination

75 between homologous chromosomes occurs in *A. vaga*. The possibility of chromosome pairing
76 in *A. vaga*, during a mitotic or meiotic-like process, allows for the re-interpretation of the sig-
77 natures of LD decay and allele sharing. Until now, the lack of chromosome-scale assemblies
78 of parthenogenetic genomes hampered the investigation of the impact of meiosis, recombina-
79 tion, outcrossing, or their absence, on entire genomes. Moreover, characterizing homologous
80 chromosomes as potential templates for DNA repair through HR in *A. vaga* is an important land-
81 mark in the understanding of bdelloid extreme resistance. This high-quality genome assembly
82 of *A. vaga* (AV20) is also timely for comparative biology within rotifers and protostomians,
83 extending the list of chromosome-level genomes in overlooked phyla.

84 **Results and discussion**

85 **A diploid genome with a tetraploid past** Distinct independent genome assembly procedures,
86 relying on different assumptions regarding ploidy levels (Bwise (22), NextDenovo (23) and
87 Falcon (24)), were first used on a combination of short and long sequencing reads. These
88 assemblies were then scaffolded using Hi-C data and instaGRAAL (25), revealing similar
89 chromosome-level assemblies and genome size estimations, consistent with flow cytometry
90 measurements (Fig. 1A and Supp. Figs. 1 and 2). All pairwise alignments of the three inde-
91 pendent assemblies (referred to as "phased" without ploidy assumption, "haploid" and phased
92 "diploid", see Fig. 1A) confirmed chromosome-level synteny and converged towards identical
93 genome structure with the six longest scaffolds from the haploid assembly (hereafter named
94 "AV20") being each colinear to exactly two long scaffolds from the phased assembly (Fig. 1B,
95 see also Supp. Figs. 3, 4 and 5). In order to validate these assemblies, we performed fluorescent
96 *in situ* hybridization (FISH) analyses with three pairs of fluorescent probe libraries complemen-
97 tary to separate parts of chromosomes 2, 5 and 6 from the AV20 assembly (Figure 1B, right
98 side). For each pair of probes (one green and one red) two individual chromosomes were la-

99 belled with little or no overlap between both signals (Figure 1C). Chromosome painting on the
100 karyotype of 12 chromosomes of *A. vaga* (26) was consistent with our chromosome-scale as-
101 semblies showing that the *A. vaga* genome is diploid, being composed of six pairs of colinear
102 homologous chromosomes.

103 We compared our new AV20 assembly to the previously published draft genome assembly
104 (hereafter named "AV13" (9)). None of the previously described colinearity breakpoints and
105 palindromes were retrieved in the new AV20 genome, indicating that these were likely assembly
106 artefacts resulting from erroneous scaffolding of uncollapsed haplotypes (Supp. Figs. 6 and 7).
107 Chromosome-level colinearity, albeit weaker than between homologous chromosomes, was also
108 observed between pairs of homoeologous (or ohnologous) chromosomes in the AV20 genome,
109 a signature confirming the previously reported paleotetraploidy of *A. vaga* (9, 10, 12) (grey links
110 on Figure 1B). The three chromosome pairs 1, 2 and 3 are homoeologous to the three pairs
111 4, 5 and 6, respectively. *A. vaga* is thus a diploid, paleotetraploid species in which the level
112 of synteny between homoeologous chromosomes is high. Notably, 30.8% of the genes have a
113 homoeologous copy within conserved synteny blocks (see Materials & Methods section) and
114 with an average nucleotide divergence of about 13% (Supp. Fig. 8).

115 **Recombination between homologous chromosomes causes loss of heterozygosity** The dis-
116 covery of homologous chromosomes in the oldest known asexual animal clade represents a
117 major shift for studies of ancient asexuals and leads us to reconsider the possibility for homol-
118 ogous recombination in *A. vaga*. One potential genetic consequence of recombination between
119 homologous chromosomes is large-scale loss of heterozygosity (LOH). We measured and com-
120 pared heterozygosity along the chromosomes of three *A. vaga* samples cultured from a same
121 ancestral laboratory strain that never underwent stresses causing recombinogenic DSBs and
122 that were sequenced at three distinct timepoints (2009, 2015 and 2017, Fig. 2A, Supp. Ta-

ble S1). Mean single-nucleotide polymorphism (SNP) heterozygosity (i.e. divergence between homologous chromosomes) was around 1.7% (horizontal line on Fig. 2A, similar to previous reports (9, 10)). Interestingly, we observed large regions (from 100 kb to 4.5 Mb) that were fully homozygous, except for a few SNPs, in specific isolates while heterozygous in others (numbered tokens in Fig. 2A). Note that a few homozygous tracks are associated with coverage variation and could have been caused by a hemizygous deletion (when coverage drops by approximately 50%, e.g. event 5 on Fig. 2A) or by the high density of repeated sequences (e.g. event 13 on Fig. 2A). Given the genealogy of these laboratory lines, we argue that the large homozygous tracks that are associated with homogeneous median coverage are signatures of allelic recombination events causing LOH (Fig. 2B).

These LOH appeared to accumulate through time as some are shared by two samples (e.g. event 12 on Fig. 2) while others appeared in only one of these two samples (e.g. event 2 on Fig.2). Noteworthy, no ancestral LOH was found that would be shared by all of the strains. This is likely because large LOH events increase the chance to expose recessive deleterious mutations and are thus likely selectively eliminated in nature, maintaining the relatively homogeneous heterozygosity level in the ancestral laboratory strain (Fig. 2A). Observing LOH tracks on all six chromosome pairs in the three laboratory samples over a relatively short period of time (i.e. several years, Fig. 2B) might be due to the culturing conditions allowing for possible bottlenecks and relaxed selection. Recombination occurring along the entire chromosomes, instead of being restricted to the telomeres only (27), invalidates the hypothesis that an *Oenothera*-like meiosis underlies their reproductive mode (in agreement with a recent study (13)). Overall, these LOH tracks combined with the recently reported LD decay (13) represent a clear footprint that molecular processes involving recombination between homologous chromosomes occur in the germline of *A. vaga*.

147 **Recombination could be accidental or programmed** Theoretically, recombination between
148 homologous chromosomes resulting in inheritable LOH can occur in the germline during mi-
149 totic repair of accidental DSBs or when handling programmed DSBs during meiosis (potentially
150 induced by Spo11 protein (9, 28)). DSBs can be repaired by different recombination pathways
151 but LOH of large chromosome regions without coverage reduction (e.g. events 1, 6-9, 11, 12,
152 Fig. 2) can primarily arise from two processes, break-induced replication (BIR) and the for-
153 mation of crossing-over (CO). BIR is a mechanism of mitotic recombination characterized by
154 replication fork progression over hundreds of kilobases on the repair template (29). When in-
155 volving allelic loci, it causes LOH of the segment extending from the breakpoint site until the
156 end of the chromosome. If a double BIR (dBIR) occurs, switching templates from the homolo-
157 gous chromosome back to the sister or the original chromatid, a LOH tract, possibly long, that
158 does not encompass the telomere is produced (30). Such LOH could also be generated by the
159 recombinational repair of respectively one or two DSBs leading to CO (i.e. a reciprocal genetic
160 exchange between chromosomes). Compared to BIR, CO is however a minor pathway in mitot-
161 ically cycling cells (31) that preferentially takes place between sister chromatids and therefore
162 remains genetically silent (32).

163 Alternatively, programmed DSBs during meiosis can produce large LOH tracks by favoring
164 CO formation between homologous chromosomes (31). LOH signatures in *A. vaga* genome
165 could therefore be acquired through meiotically-induced recombination instead of during mi-
166 tosis. Several mechanisms of meiotic parthenogenesis, globally referred to as automixis, have
167 been described in various species such as in *Daphnia pulex* (33), *Artemia parthenogenetica* (34)
168 or *Apis mellifera capensis* (35). If automixis occurs in *A. vaga*, the heterozygosity patterns ob-
169 served here (Fig. 2) in which the maternal heterozygosity is conserved along chromosomes due
170 to the non-segregation of homologous chromosomes while large LOH tracks (likely counter-
171 selected in nature) could result from their CO recombination, is genetically equivalent to what

172 is referred to as central fusion automixis (34). Nevertheless, no cytological evidence of any
173 meiotic process has been described so far in bdelloid rotifers. Whether recombination is a key
174 feature of the reproductive mode of bdelloids (through programmed DSBs during a modified
175 meiosis) or whether it is mainly driven by desiccation resistance mechanisms (through acciden-
176 tal DSB repair in the germline), or both, remains an open question. Whichever mechanism is
177 involved, recombination likely plays a major role in the long-term evolution of *A. vaga* genome.

178 **Dynamic subtelomeric regions** We found a low amount of transposable elements (TEs) in
179 *A. vaga* (Fig. 3, Supp. Fig. 9). By combining two approaches to annotate both TE-like el-
180 ements, including repeated sequences, as well as canonical TEs (i.e. the EDTA and REPET
181 pipelines) we detected 6.6% of TE-like elements and 1.9% of canonical TEs, predominantly
182 located at subtelomeric regions (Fig. 3). In addition, rotifer-specific telomeric repeats (i.e.
183 (TGTGGG)_n (36)) were detected at the extremities of every scaffold of the AV20 assembly,
184 indicating that they indeed correspond to telomeric and subtelomeric regions and that AV20
185 reached a chromosome-level assembly (Supp. Fig. 10). Most consensus TE sequences were
186 found at low copy numbers (i.e. 96% of canonical TEs consensus sequences are present in
187 ($\leq 5x$) copies in AV20, see Supp. Fig. 9). Notably, terminal inverted repeats (TIRs) DNA trans-
188 posons (i.e. Class-II) were quantitatively dominant (48% of all TEs) among the low amount of
189 TEs in *A. vaga* genome (Supp. Figs. 9 and 11). These results are in line with previous stud-
190 ies of TEs in bdelloids (9, 10, 37, 38). Using sequence similarity between a TE copy and their
191 consensus as a proxy for how recent this copy is, we found that Class-II TIRs and Class-I
192 LINEs and LTRs had high average similarity to their consensus sequences suggesting that they
193 have been at least recently active in *A. vaga* genome (Supp Fig. 12). Investigating putative
194 endogenous viral elements (EVEs) in *A. vaga* revealed very few viral-like sequences (i.e. 94
195 loci scattered along the 6 chromosomes, Fig. 3) with potential donor candidates belonging to

196 the group of large double stranded (ds) DNA viruses. None of these EVE candidates however
197 had definitive viral origins as their similarity was not restricted to viral sequences and there was
198 no conservation of viral gene synteny.

199 Syntenic HGTc regions among non-homoeologous chromosomes are visible, mainly at sub-
200 telomeric regions (violet links on Fig. 3) and may suggest chromosomal re-arrangements. Sub-
201 telomeric regions are also the regions on which almost all divergent haplotypes (i.e. haplotigs
202 corresponding to uncollapsed haplotypes during genome assembly process) were located (grey
203 links on Fig. 3). Overall, these subtelomeric regions in *A. vaga* are enriched in canonical TEs,
204 TE-like elements, HGTc, viral-like sequences but also retain a higher haplotypic divergence
205 (i.e. uncollapsed haplotigs) and most chromosomal re-arrangements. When accounting for only
206 coding sequences (CDS), no distinct increase or decrease of heterozygosity could however be
207 observed at subtelomeric regions (Supp. Fig. 13). At this stage, it is therefore unclear whether
208 homologous recombination rate covaries with telomeric proximity in *A. vaga*. Nevertheless, our
209 results suggest that subtelomeric regions seem more prone to chromosomal re-arrangements, in-
210 corporation of foreign DNA (TEs and HGTs) and structural variations such as putative allelic
211 deletions (see LOH events 3 and 10 in Fig. 2), evolving faster than the rest of the genome.

212 **Horizontal gene transfers in *A. vaga* genome** The acquisition of foreign DNA has been
213 hypothesized to play an important role in bdelloid evolution (20). HGTs could be a way to
214 circumvent some deleterious effects of the lack of sexual outcrossing, and the occasional inte-
215 gration of foreign DNA could trigger adaptation (10,17,20,39,40). No automated tool existed to
216 detect HGTc, therefore we developed Alienomics, an innovative pipeline to detect both HGTc
217 and contaminants in a genome assembly. Alienomics combines several genomic parameters
218 such as gene taxonomy, GC content, sequencing depth, but also taking into account gene inte-
219 gration into the genome using synteny and expression data, to detect HGTc from non-metazoan

220 species. In contrast with the overall low amount of TEs, many candidate HGTc (2,679, about
221 8.3% of all genes) were detected in the *A. vaga* AV20 genome assembly, confirming previous
222 reports of the highest HGTc content among metazoans (9, 10, 41, 42). HGTc were enriched in
223 subtelomeric regions as previously reported (41), although many HGTc were distributed along
224 the chromosomes and two visible local hotspots were detected outside the subtelomeric regions
225 (pink stars on Figure 3). Interestingly, one HGT hotspot is associated with a slight increase of
226 interstitial telomeric repeats (Supp. Fig. 10) that could represent a signature of ancient chro-
227 mosome fusion. Overall, the heterogeneity in HGTc density between subtelomeric regions and
228 the rest of the genome could be explained either by varying rates of HGTs incorporation along
229 the chromosomes or by varying successful integration of HGTs within the genome through
230 selection.

231 Using both MCScanX and Alienomics outputs, we measured that 257 foreign genes (9.6%
232 of all HGTc) conserved their synteny across homoeologous chromosomes, including the HGTc
233 hotspots notably visible in homoeologous chromosomes 1 and 4 (stars on Figure 3). These
234 horizontal transfer events therefore occurred before the ancestral tetraploidization of modern
235 bdelloids. This amount of ancient HGTc is however likely underestimated as any loss or translo-
236 cation of an ancient HGTc copy would break the ancestral synteny. When looking specifically
237 at these HGTc that occurred prior to the tetraploidization, we observed an enrichment of genes
238 involved DNA recombination and DNA ligation involved in DNA DSB repair, among other
239 enriched functional categories (see Supp. Table S2). These HGTc might have set bdelloids up
240 to resist and overcome massive DNA breakage. When analyzing all HGTc, we found that they
241 are enriched in genes involved in oxidation-reduction and carbohydrate metabolic processes (9)
242 as well as in the response to nitrosative stress (see Supp. Table S2). Acquisition of HGTs
243 might therefore be central in their resistance to extreme desiccation and towards more efficient
244 homeostasis. Overall, these results are in line with previous studies suggesting that HGTs have

245 been continuously acquired within bdelloid rotifers, even before their tetraploidization (10, 42).
246 However, if bdelloids have the same low rate of HGT acquisition from other individuals of
247 the same species than from non-metazoans (12.8 HGT/Myr), HGT is possibly insufficient to
248 compensate for the plausible lack of outcrossing in bdelloid rotifers (40). Actually, a high rate
249 of HGT acquisition from distinct species might be deleterious for *A. vaga*, which appears to
250 rely on recombination between homologous chromosomes to maintain heterozygosity and/or
251 genome structure.

252 **Reasoning on bdelloid rotifer reproductive mode** Bdelloid rotifer species are both suppos-
253 edly devoid of males and prone to integrate foreign DNA (through HGTs) into their genome. In
254 this context, several reports of allele sharing between bdelloid individuals sampled from the wild
255 triggered a debate whether they could exchange genetic content at all and whether this might be
256 done through HGT or through sexual reproduction (13, 19–21, 43–45). At a first glance, show-
257 ing that homologous chromosomes exist and recombine in *A. vaga* could be viewed as a support
258 to the hypothesis that bdelloids might undergo meiotic sexual reproduction (13). However, this
259 hypothesis has yet to be reconciled with the absence of both males and canonical meiosis in
260 bdelloid rotifers and here we speculate on the mechanisms of homologous recombination in
261 *A. vaga*. The three *A. vaga* lineages analyzed here (Fig. 2) were kept in hydrated conditions,
262 leaving few opportunities for desiccation-induced, accidental DNA DSBs. Moreover, a much
263 lower heterozygosity than for *A. vaga* has been observed in two obligate aquatic bdelloid rotifer
264 species (i.e. *Rotaria*, upper limit of homologous divergence ranged between 0.033 and 0.075),
265 also described as asexual and never experiencing dessication. Both these observations are com-
266 patible with the hypothesis that homologous recombination in bdelloids could be caused by
267 programmed DNA DSBs during a meiotic-like process. Frequent and programmed recombina-
268 tion would cause LOH in *A. vaga* (Fig. 2) and would have lowered heterozygosity even in the

269 obligately aquatic *Rotaria* species.

270 Whatever the underlying mechanism, the observed recombination signatures in bdelloid
271 rotifers are compatible with the three hypotheses proposed to explain the previous reports of
272 allele sharing patterns in bdelloid rotifers: i) allele sharing may be due to undetected contam-
273 ination between cultures, either during colony culture itself or during sample preparation for
274 sequencing (46–50); ii) allele sharing is the result of horizontal genetic transfers between bdel-
275 loid individuals through unknown molecular mechanisms, possibly associated with desiccation
276 (but not for the non-desiccating species) and potentially linked to the high propensity of bdel-
277 loids to retain non-metazoan genes into their genomes (13, 17, 20, 40, 43, 51); iii) allele sharing
278 is caused by cryptic sexual reproduction (52), with sex events being rare enough so that males,
279 sperm, fertilization and meiosis were never observed, but sufficient to leave a distinctive foot-
280 print in every population sample studied so far (13, 19, 21, 45). The mechanism behind the
281 observed signatures of genetic exchanges between bdelloid individuals remains puzzling and
282 therefore the significance of outcrossing in this asexual lineage remains unclear. We anticipate
283 the chromosome-level genome assembly of *A. vaga* presented here will stimulate future popu-
284 lation genomics studies that will help to determine the cause of these allele sharing patterns.

285 **Long-term asexual evolution** This high-quality telomere-to-telomere assembly firmly estab-
286 lishes *A. vaga* as a model system to study long-term asexual evolution. Homologous chromo-
287 somes are present in the bdelloid species *A. vaga* and might well occur in all bdelloid rotifers,
288 as colinear pairs of sequenced fosmids were found in two distinct bdelloid species *A. vaga* and
289 *Philodina roseola*, with each colinear pair in one species resembling the colinear pair in the
290 other species (12). The observed long LOH tracks indicate the existence of long-range homol-
291 ogous recombination (Fig. 4), whether this occurs during a meiotic-like parthenogenetic mode
292 of reproduction or in a mitotic context during frequent repair of accidental DNA DSBs remains

293 speculative. Recombination (mitotic or meiotic) could increase the rate of gene conversion in
294 asexual lineages, a signature previously observed in *A. vaga* (9). Gene conversion, particularly
295 when slightly biased, could correct deleterious mutations and reduce the rate of clonal dete-
296 rioration (53), or even speed up the fixation of beneficial mutations (54). However, besides
297 signatures of LOH representing allelic recombination, we also observed LOH via deletions in
298 the genome of *A. vaga*. The random accumulation of LOH events could expose deleterious
299 recessive mutations in asexuals through loss of complementation (55). This chromosome-scale
300 genome assembly of asexual *A. vaga* therefore is a critical tool to be able to evaluate the rela-
301 tive benefits of these recombination events on their long-term evolution and paves the way for
302 studies on genome dynamics in *A. vaga*.

303 In general, asexual populations suffer from the absence of gene shuffling with other individ-
304 uals and the long-term evolutionary success of bdelloid rotifers in the absence of outcrossing
305 therefore remains puzzling. It is important to try to discriminate between the consequences of
306 the two aspects underlying sex: recombination and outcrossing. Theoretical work on popula-
307 tion genetics showed that selection could be at least as efficient in automictic lineages than in
308 sexuals under certain circumstances (e.g. effective population size, recombination rates (56)). It
309 is therefore conceivable that the combination of potentially large populations, a relatively high
310 level of heterozygosity (or mutation rate) and specific recombination rates might explain how *A.*
311 *vaga* maintains a delicate balance between losing and accumulating heterozygosity, and how it
312 adapts and persists in the long term. Unfortunately, critical knowledge about bdelloid biology is
313 still missing (e.g. mutation and recombination rates) to determine whether they might circum-
314 vent the lack of outcrossing through a genetic equivalent of automixis. Outcrossing through
315 sexual reproduction might speed up adaptation by allowing the combination of independently
316 evolved alleles within the same individual, but might not be essential for bdelloid rotifers, es-
317 pecially if a high frequency of HGT is also taking place. Despite presenting the highest amount

318 of HGTc among animals, our results also suggest that bdelloid rotifers might have to balance
319 the acquisition rate of HGTs, a source of functional novelties, with the maintenance of faithful
320 homology between chromosomes for homologous recombination. Overall, our work reinforces
321 the hypothesis that recombination is critical for lineage longevity. Ancient asexual animals
322 without a minimal rate of recombination, programmed through meiotic processes and/or acci-
323 dental through their life-style, might not exist at all.

324 **Materials and Methods**

325 Complete description of materials and methods can be found in Supplementary Materials.

326 **Genome sequencing and assembly** Continuous cultures of *A. vaga* AD008 lab strain were
327 processed in order to obtain the following sequencing data: about 350x coverage of WGS 250-
328 bp paired-end Illumina reads, 200x coverage of PacBio RSII long reads, 125x coverage of
329 ONT long reads and 75-bp paired-end Illumina reads of Hi-C libraries. Three independant
330 genomes were assembled using Bwise (on Illumina short reads), NextDenovo (on ONT long
331 reads) and Falcon (on Pacbio long reads). Uncollapsed haplotypes in the ONT-based assembly
332 were then detected and discarded using `purge_dups`. The resulting assembly was polished based
333 on Illumina short reads and Pacbio long reads using HyPo, and is here referred to as "AV20"
334 genome assembly. All assemblies were then scaffolded using instaGRAAL (on Hi-C data).
335 Ploidy level and genome size was confirmed using k-mers spectra using KAT, synteny analyses
336 using MCScanX, nucmer and D-GENIES, flow cytometry measurements and FISH using three
337 pairs of oligo datasets designed on three chromosomes.

338 **Genome annotation** TE-like elements and canonical TEs consensus were build from the
339 AV20 genome assembly using EDTA and TEdenovo pipeline, and AV20 genome was then

340 annotated using TEannot (part of the REPET pipeline). Genes were annotated using funanno-
341 tate. For this, repeated elements previously annotated were masked using bedtools, part of the
342 available RNA-Seq reads were mapped onto the genome while the other part of RNA-Seq reads
343 were used to produce a de novo assembled transcriptome which was subsequently aligned onto
344 the genome as part of the PASA pipeline. Then a combination of PASA annotations, de novo as-
345 sembled transcriptome, metazoan BUSCO database and the proteic Uniprot database within fu-
346 nannotate predict function. This first produced ab initio predictions using Genemark-ES, which
347 were then used along with transcripts and proteic data to train Augustus to generate a second set
348 of annotations. Lastly, it used Evidence Modeler as a weighted approach to combine annota-
349 tions from PASA, Genemark and high quality predictions from Augustus into an integrated gene
350 annotation set. We then used InterProScan5 in order to produce functional annotations to the
351 predicted genes which were then used in combination with busco metazoan database using the
352 funannotate annotate function with default parameters. We used Alienomics, a newly designed
353 pipeline, in order to detect HGTc. This approach combines GC content, coverage, blasts, taxo-
354 nomic information, expression level and synteny information in order to detect both HGTc (i.e.
355 alien genes integrated into host scaffolds) and potential contaminants (i.e. alien genes present
356 on alien scaffold). Note that our approach can only detect transfers from alien source outside
357 of a given clade (i.e. here, metazoa). Viral-like genes were detected by performing a diamond
358 blastx search on AV20 scaffolds using all viral proteins extracted from the nr database of NCBI
359 (February 2020) to the exception of Retroviridae and Hepadnaviridae.

360 **Genome analyses** Coverage along AV20 scaffolds was computed using read mapping with
361 bwa mem, and these alignments were used for genotyping three samples (i.e. GC047403,
362 BXQF, ERR321927) using GATK (HaplotypeCaller function with -ERC GVCF option). The
363 resulting gvcf files were combined (CombineGVCFs function) and were then jointly genotyped

364 (GenotypeGVCFs function). The variants were then filtered in order to only retain SNPs using
365 custom bash and perl scripts. Divergence between homoeologous chromosomes was obtained
366 by the production of a self-alignment of AV20 genome using nucmer, which was then filtered
367 to only retain genomic alignments between homoeologous regions ranging from 500 to 10,000
368 bp. MCScanX was used to detect synteny among HGTc, and custom scripts were used to
369 detect strictly homoeologous HGTc synteny blocks stemming from the paelotetraploidization
370 of bdelloids (i.e. ancient HGTc). GO terms from functional annotation were extracted for the
371 2,422 recent HGTc and the 257 ancient HGTc were respectively compared to the entire gene set
372 of the AV20 genome containing 32,378 proteins. Enrichment analyses were performed using
373 topGO package with a fisher test and the "elim" algorithm.

374 **Reappraisal of AV13 genome assembly** ONT reads were trimmed with porechop and were
375 mapped onto the previous AV13 genome assembly using NGMLR. The AV20 and AV13 genome
376 assemblies were aligned together using Sibelia. This alignment was used to detect putative
377 breakpoint location which were then inspected using Tablet in order to evaluate whether ONT
378 reads confirmed one of the assemblies. Synteny blocks from the Sibelia alignment between
379 AV20 and AV13 were screened using a custom perl script (available at <https://github.com/jnarayan81/huntPalindromes>)
380 and circos plots in order to evaluate the existence of palindromes in AV20 genome. No break-
381 points or palindromes detected in the AV13 genome assembly could be found in the AV20
382 genome assembly, nor be confirmed by ONT reads.

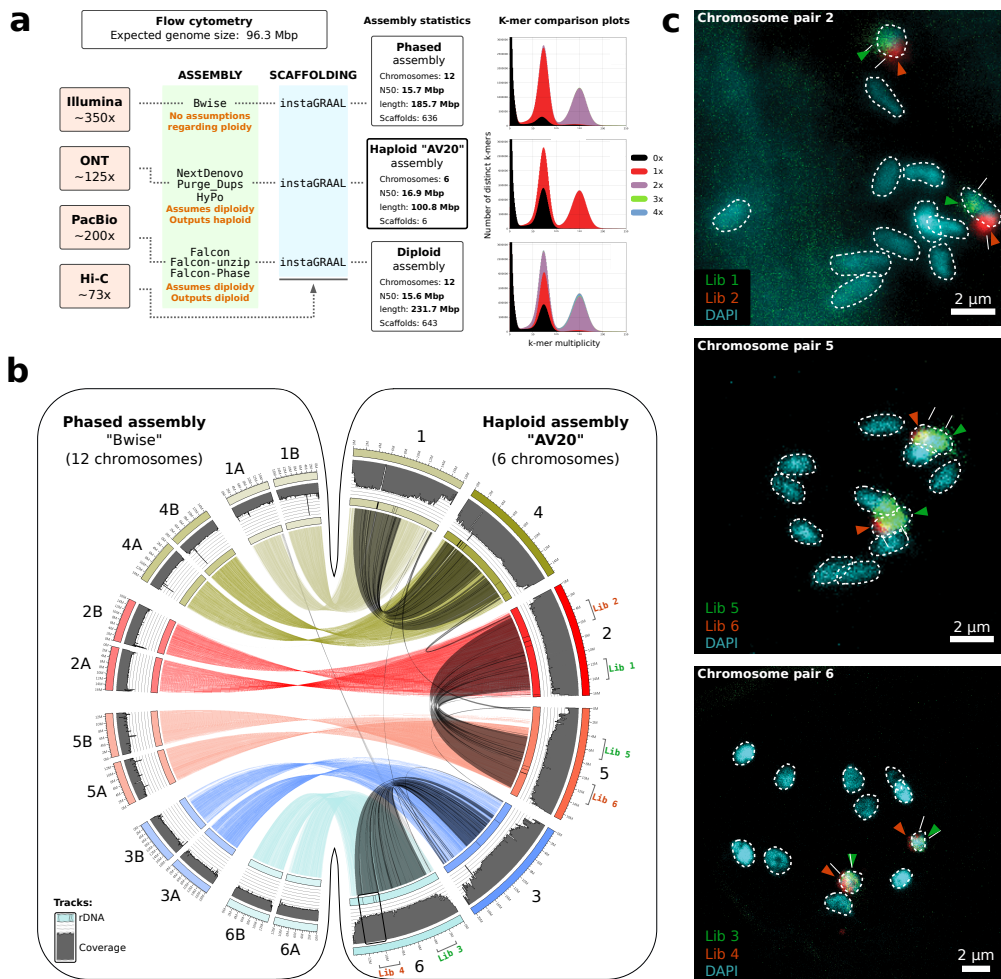


Figure 1: **Genome structure of *Adineta vaga* is diploid.** a) Outline of the three genome assembly approaches underlined by different assumptions on genome ploidy with median read coverage for all sequencing technologies indicated on the left and estimated with respect to the AV20 haploid genome assembly. The haploid genome size estimate of *A. vaga* obtained by flow cytometry is given (under the assumption that the genome is diploid) as well as the summary statistics of the genome assemblies. Number of chromosomes corresponds to the number of scaffolds longer than 10 Mbp. Ploidy levels of assemblies is indicated by the KAT plots of k-mers distribution (first and second peaks corresponds to heterozygous and homozygous k-mers, respectively; red and purple indicates haploidy and diploidy, respectively). b) Circos plot of the pairwise colinearity between the haploid AV20 and the phased Bwise genome assemblies, depicted by colored links and obtained using nucmer. Synteny blocks within AV20 genome (between homoeologous copies) are depicted as grey links and were obtained using MCScanX. Coverage along scaffolds of both AV20 and the phased assembly are depicted as grey histograms and were computed based on illumina reads from sample GC047403. Thin black bars on the scaffold ideograms correspond to rRNA genes. Schematic position of the FISH probe libraries on chromosome pair 2, 5 and 6 is indicated on the corresponding AV20 chromosomes. c) Karyotype of the 12 chromosomes of *A. vaga* (DAPI staining) with chromosome pairs 2, 5 and 6 highlighted by oligo painting using the FISH probe libraries depicted in panel b.

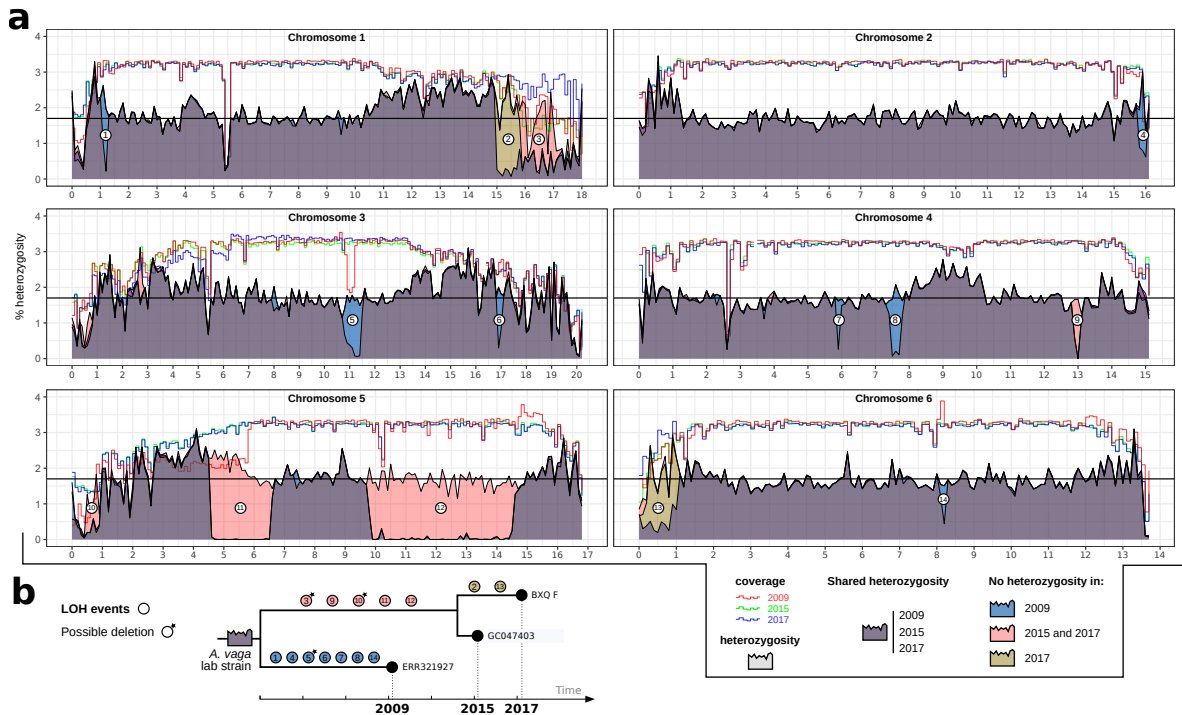


Figure 2: **Heterozygosity dynamics in *Adineta vaga*.** a) Heterozygosity and coverage distributions of three independent *A. vaga* samples from the same laboratory strain along the six chromosomes. Samples are labeled by the date of the extraction of their DNA (i.e. 2009, 2015 and 2017). Data from 2009 were used to assemble the previous version of *A. vaga* genome (9). Lines indicate short read coverage (normalized) and filled areas indicate the percentage of heterozygosity (y-axis). Chromosome lengths (x-axis) are in Mb. Mean SNP heterozygosity (1.7%) is depicted by the horizontal black line. b) Schematic reconstruction of heterozygosity evolution among 3 samples from the same initial *A. vaga* lab strain. Note that each sample had its own independent evolution and the exact sequence and timing of LOH events is unknown. LOH events noted with a small asterisk might correspond to deletions given the drop of coverage associated with the absence of heterozygosity.

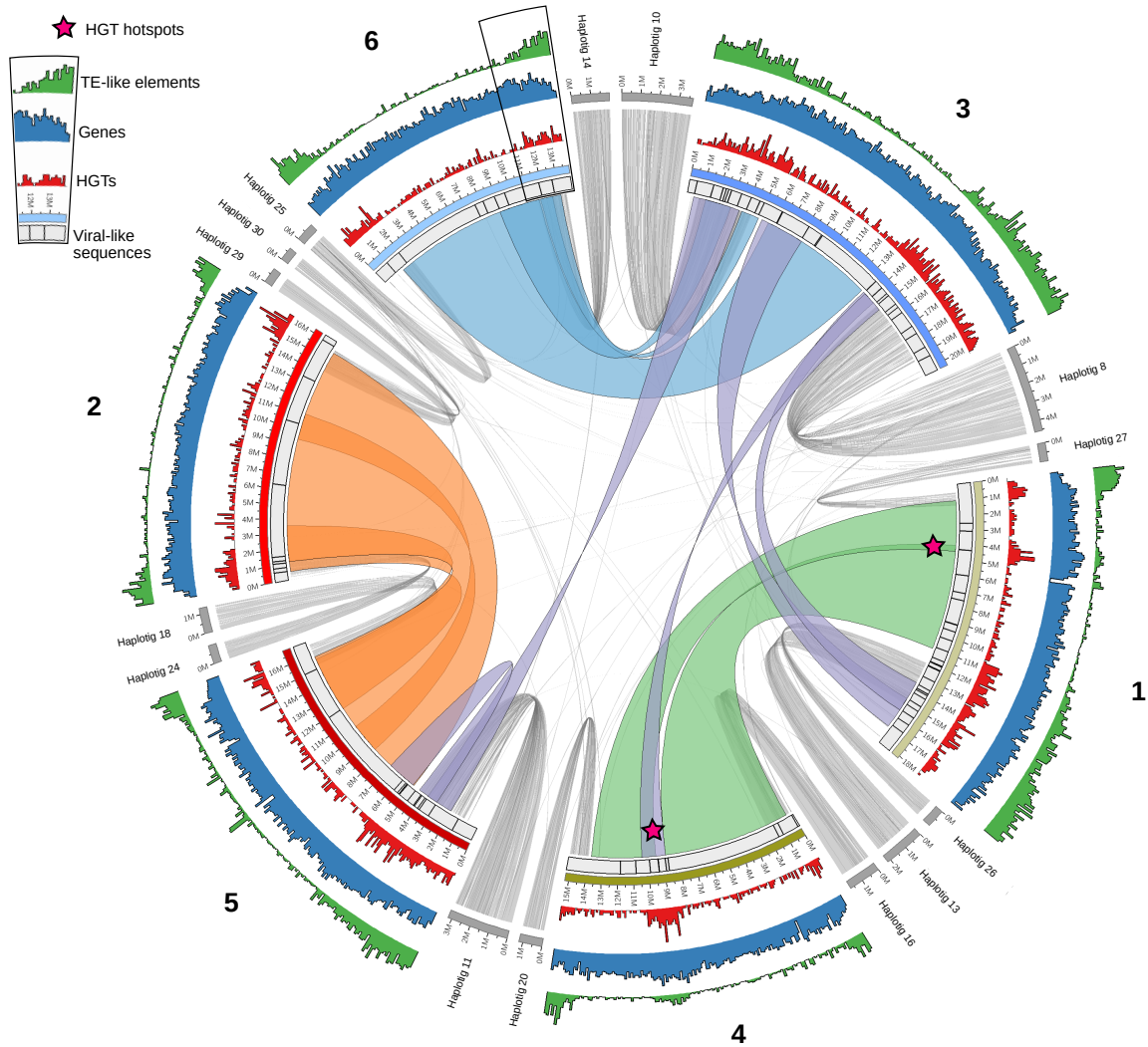


Figure 3: **DNA content of haploid *Adineta vaga* genome AV20.** Synteny of HGTc is depicted by colored links between the 6 chromosome pairs. Violet links correspond to synteny block of HGTc between non-homoeologous chromosomes. Localization of alternative haplotigs, removed prior to genome scaffolding, are depicted by grey links. Distribution of repeated elements, genes, HGTc and viral-like sequences are depicted in green, blue, red and black bars, respectively. Ancient HGTc hotspots are indicated by pink stars.

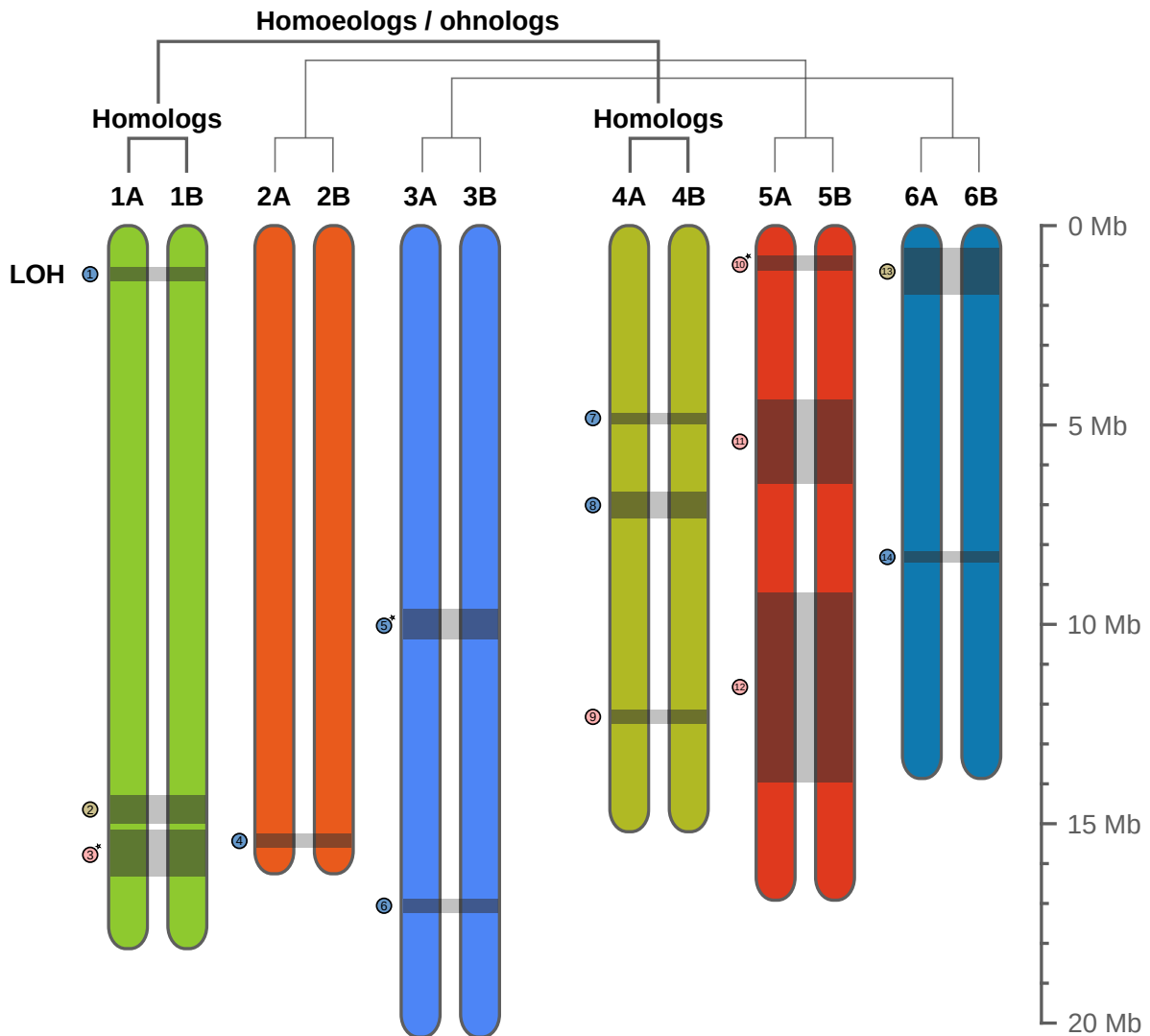


Figure 4: **Schematic representation of *A. vaga* karyotype.** The 12 chromosomes correspond to 6 pairs of homologous chromosomes (i.e. diploidy) sharing the same colour. Ancestral genome hybridization (or duplication) led to the existence of pairs of homoeologs (or ohnologs), represented in different but similar colours. Grey blocks linking homologous chromosomes and their corresponding numbered tokens depict loss of heterozygosity events produced by homologous recombination (see also Fig. 2). Length of chromosomes (in Mb) is indicated by the scale on the right.

References and Notes

- 383
- 384 1. J. Maynard Smith, *The evolution of sex* (Cambridge University Press Cambridge, 1978).
 - 385 2. J. Lehtonen, M. D. Jennions, H. Kokko, *Trends Ecol Evol* **27**, 172 (2012).
 - 386 3. O. Seudre, E. Vanhoenacker, S. Mauger, J. Coudret, D. Roze, *Journal of Evolutionary*
387 *Biology* **33**, 112 (2020).
 - 388 4. H. Fradin, *et al.*, *Current Biology* **27**, 2928 (2017).
 - 389 5. W. G. Hill, A. Robertson, *Genetics Research* **8**, 269 (1966).
 - 390 6. C. W. Birky, *Journal of Heredity* **101**, S42 (2010).
 - 391 7. W. S. Hsu, *La Cellule Section* **57**, 283 (1956).
 - 392 8. W. S. Hsu, *The Biological Bulletin* **111**, 364 (1956).
 - 393 9. J.-F. Flot, *et al.*, *Nature* **500**, 453 (2013).
 - 394 10. R. W. Nowell, *et al.*, *PLOS Biology* **16**, e2004830 (2018).
 - 395 11. D. B. M. Welch, J. L. M. Welch, M. Meselson, *Proceedings of the National Academy of*
396 *Sciences* **105**, 5145 (2008).
 - 397 12. J. H. Hur, K. Van Doninck, M. L. Mandigo, M. Meselson, *Molecular Biology and Evolu-*
398 *tion* **26**, 375 (2009).
 - 399 13. O. A. Vakhrusheva, *et al.*, *Nature Communications* **11**, 6421 (2020).
 - 400 14. D. Fontaneto, N. Bunnefeld, M. Westberg, *Astrobiology* **12**, 863 (2012).
 - 401 15. B. Hespeels, *et al.*, *Frontiers in Microbiology* **11**, 1792 (2020).
 - 402 16. E. Gladyshev, M. Meselson, *Proceedings of the National Academy of Sciences* **105**, 5139
403 (2008).
 - 404 17. B. Hespeels, *et al.*, *Journal of Evolutionary Biology* **27**, 1334 (2014).
 - 405 18. J. Romiguier, *et al.*, *Nature* **515**, 261 (2014).
 - 406 19. A. Signorovitch, J. Hur, E. Gladyshev, M. Meselson, *Genetics* **200**, 581 (2015).
 - 407 20. N. Debortoli, *et al.*, *Current Biology* **26** (2016).
 - 408 21. A. Signorovitch, J. Hur, E. Gladyshev, M. Meselson, *Current Biology* **26**, R754 (2016).

- 409 22. A. Limasset, Novel approaches for the exploitation of high throughput sequencing data,
410 PhD thesis, Université Rennes 1 (2017).
- 411 23. Nextdenovo (2020). <https://github.com/Nextomics/NextDenovo>.
- 412 24. C.-S. Chin, *et al.*, *Nature Methods* **13**, 1050 (2016).
- 413 25. L. Baudry, *et al.*, *bioRxiv* p. 2019.12.22.882084 (2019).
- 414 26. J. L. Mark Welch, M. Meselson, *Hydrobiologia* **387**, 403 (1998).
- 415 27. H. Golczyk, A. Massouh, S. Greiner, *The Plant Cell* **26**, 1280 (2014).
- 416 28. B. J. Hecox-Lea, D. B. Mark Welch, *BMC Evolutionary Biology* **18**, 177 (2018).
- 417 29. C. J. Sakofsky, A. Malkova, *Critical Reviews in Biochemistry and Molecular Biology* **52**,
418 395 (2017).
- 419 30. E. Yim, K. E. O’Connell, J. St. Charles, T. D. Petes, *Genetics* **198**, 181 (2014).
- 420 31. M. Bzymek, N. H. Thayer, S. D. Oh, N. Kleckner, N. Hunter, *Nature* **464**, 937 (2010).
- 421 32. L. C. Kadyk, L. H. Hartwell, *Genetics* **132**, 387 (1992).
- 422 33. C. Hiruta, C. Nishida, S. Tochinai, *Chromosome Research* **18**, 833 (2010).
- 423 34. O. Nogu , *et al.*, *Journal of Evolutionary Biology* **28**, 2337 (2015).
- 424 35. F. Goudie, B. P. Oldroyd, *Apidologie* **45**, 306 (2014).
- 425 36. J. M. Mason, T. A. Randall, R. Capkova Frydrychova, *Chromosoma* **125**, 65 (2016).
- 426 37. I. Arkhipova, M. Meselson, *BioEssays* pp. 76–85 (2005).
- 427 38. R. W. Nowell, *et al.*, *eLife* **10**, e63194 (2021).
- 428 39. C. Boschetti, *et al.*, *PLOS Genetics* **8**, e1003035 (2012).
- 429 40. I. Eyres, *et al.*, *BMC biology* **13**, 90 (2015).
- 430 41. E. A. Gladyshev, M. Meselson, I. R. Arkhipova, *Science* **320**, 1210 (2008).
- 431 42. B. Hespeels, J.-F. Flot, A. Derzelle, K. Van Doninck, *Evolutionary Biology: Genome Evo-*
432 *lution, Speciation, coevolution and Origin of Life*, P. Pontarotti, ed. (Springer International
433 Publishing, 2014), pp. 207–225.
- 434 43. J.-F. Flot, N. Debortoli, B. Hallet, K. V. Doninck, *Current Biology* **26**, R755 (2016).

- 435 44. T. Schwander, *Current Biology* **26**, R233 (2016).
- 436 45. V. N. Laine, T. Sackton, M. Meselson, *bioRxiv* p. 2020.08.06.239590 (2020).
- 437 46. M. Ballenghien, N. Faivre, N. Galtier, *BMC Biology* **15**, 25 (2017).
- 438 47. P. Simion, *et al.*, *BMC Biology* **16**, 28 (2018).
- 439 48. C. G. Wilson, R. W. Nowell, T. G. Barraclough, *Current Biology* **28**, 2436 (2018).
- 440 49. M. Prous, K. M. Lee, M. Mutanen, *Molecular Phylogenetics and Evolution* **143**, 106670
441 (2020).
- 442 50. P. Simion, F. Delsuc, H. Philippe, *Phylogenetics in the Genomic Era*, C. Scornavacca,
443 F. Delsuc, N. Galtier, eds. (<https://hal.inria.fr/PGE/>, 2020), pp. 2.1:1–2.1:34.
- 444 51. J.-F. Flot, N. Debortoli, B. Hallet, J. Narayan, K. Van Doninck, *BioRxiv* p. 368209 (2018).
- 445 52. L. Boyer, R. Jabbour-Zahab, M. Mosna, C. R. Haag, T. Lenormand, *Evolution Letters* **5**,
446 164 (2021).
- 447 53. O. Khakhlova, R. Bock, *The Plant Journal* **46**, 85 (2006).
- 448 54. M. A. Mandegar, S. P. Otto, *Proceedings. Biological Sciences* **274**, 1301 (2007).
- 449 55. M. Archetti, *Journal of Evolutionary Biology* **17**, 1084 (2004).
- 450 56. J. Engelstädter, *Genetics* **206**, 993 (2017).
- 451 57. L. Lazar-Stefanita, *et al.*, *The EMBO Journal* **36**, 2684 (2017).
- 452 58. E. Lieberman-Aiden, *et al.*, *Science* **326**, 289 (2009).
- 453 59. D. B. Mark Welch, M. Meselson, *Biological Journal of the Linnean Society* **79**, 85 (2003).
- 454 60. M. D. Bennett, I. J. Leitch, H. J. Price, J. S. Johnston, *Annals of Botany* **91**, 547 (2003).
- 455 61. J. L. Mark Welch, D. B. Mark Welch, M. Meselson, *Proceedings of the National Academy*
456 *of Sciences* **101**, 1618 (2004).
- 457 62. B. J. Beliveau, *et al.*, *Proceedings of the National Academy of Sciences* **109**, 21301 (2012).
- 458 63. B. J. Beliveau, *et al.*, *Proceedings of the National Academy of Sciences* **115**, E2183 (2018).
- 459 64. B. J. Beliveau, *et al.*, *Nature Communications* **6**, 7147 (2015).
- 460 65. B. D. Fields, S. C. Nguyen, G. Nir, S. Kennedy, *eLife* **8**, e42823 (2019).

- 461 66. R. Chikhi, A. Limasset, P. Medvedev, *Bioinformatics* **32**, i201 (2016).
- 462 67. A. Limasset, J.-F. Flot, P. Peterlongo, *Bioinformatics* **36**, 1374 (2020).
- 463 68. A. Limasset, B. Cazaux, E. Rivals, P. Peterlongo, *BMC Bioinformatics* **17**, 237 (2016).
- 464 69. A. V. Zimin, *et al.*, *Bioinformatics* **29**, 2669 (2013).
- 465 70. D. Guan, *et al.*, *Bioinformatics* **36**, 2896 (2020).
- 466 71. H. Li, *Bioinformatics* **34**, 3094 (2018).
- 467 72. F. Cabanettes, C. Klopp, *PeerJ* **6**, e4958 (2018).
- 468 73. R. Kundu, J. Casey, W.-K. Sung, *bioRxiv* p. 2019.12.19.882506 (2019).
- 469 74. H. Li, *arXiv:1303.3997 [q-bio]* (2013).
- 470 75. H. Li, *et al.*, *Bioinformatics* **25**, 2078 (2009).
- 471 76. A. Tarasov, A. J. Vilella, E. Cuppen, I. J. Nijman, P. Prins, *Bioinformatics* **31**, 2032 (2015).
- 472 77. S. Koren, *et al.*, *Genome Research* **27**, 722 (2017).
- 473 78. C. Matthey-Doret, *et al.*, hicstuff (2020). <https://github.com/koszullab/hicstuff>.
- 474 79. B. Langmead, S. L. Salzberg, *Nature Methods* **9**, 357 (2012).
- 475 80. K.-K. Lam, K. LaButti, A. Khalak, D. Tse, *Bioinformatics* **31**, 3207 (2015).
- 476 81. A. L. Delcher, S. L. Salzberg, A. M. Phillippy, *Current Protocols in Bioinformatics* **00**,
477 10.3.1 (2003).
- 478 82. S. Ou, *et al.*, *Genome Biology* **20**, 275 (2019).
- 479 83. T. Flutre, E. Duprat, C. Feuillet, H. Quesneville, *PLOS ONE* **6**, e16526 (2011).
- 480 84. H. Quesneville, *et al.*, *PLOS Computational Biology* **1**, e22 (2005).
- 481 85. J. Palmer, J. Stajich, *nextgenusfs/funannotate: funannotate v1.5.3* (Zenodo, 2019).
- 482 86. A. R. Quinlan, I. M. Hall, *Bioinformatics* **26**, 841 (2010).
- 483 87. A. M. Bolger, M. Lohse, B. Usadel, *Bioinformatics* **30**, 2114 (2014).
- 484 88. M. G. Grabherr, *et al.*, *Nature Biotechnology* **29**, 644 (2011).
- 485 89. B. J. Haas, *et al.*, *Nucleic Acids Research* **31**, 5654 (2003).

- 486 90. A. Lomsadze, V. Ter-Hovhannisyanyan, Y. O. Chernoff, M. Borodovsky, *Nucleic Acids Re-*
487 *search* **33**, 6494 (2005).
- 488 91. P. Jones, *et al.*, *Bioinformatics* **30**, 1236 (2014).
- 489 92. T. Seemann, Barrnap 0.9 (2018). <https://github.com/tseemann/barrnap>.
- 490 93. B. Buchfink, C. Xie, D. H. Huson, *Nature Methods* **12**, 59 (2015).
- 491 94. D. Mapleson, G. Garcia Accinelli, G. Kettleborough, J. Wright, B. J. Clavijo, *Bioinform-*
492 *atics* **33**, 574 (2017).
- 493 95. S. Kurtz, *et al.*, *Genome Biology* **5**, R12 (2004).
- 494 96. A. McKenna, *et al.*, *Genome Research* **20**, 1297 (2010).
- 495 97. R. Poplin, *et al.*, *bioRxiv* p. 201178 (2018).
- 496 98. Y. Wang, *et al.*, *Nucleic Acids Research* **40**, e49 (2012).
- 497 99. A. Alexa, J. Rahnenführer, T. Lengauer, *Bioinformatics* **22**, 1600 (2006).
- 498 100. R. R. Wick, L. M. Judd, C. L. Gorrie, K. E. Holt, *Microbial Genomics*, **3**, e000132 (2017).
- 499 101. F. J. Sedlazeck, *et al.*, *Nature Methods* **15**, 461 (2018).
- 500 102. I. Minkin, A. Patel, M. Kolmogorov, N. Vyahhi, S. Pham, *Algorithms in Bioinformatics*,
501 A. Darling, J. Stoye, eds. (Springer, 2013), pp. 215–229.
- 502 103. I. Milne, *et al.*, *Bioinformatics* **26**, 401 (2010).

503 **Data availability**

504 All data and genome assembly are available under the project accession number PRJNA680543.

505 **Acknowledgments**

506 We thank Alexandre Mayer for his help with scores computation and transformation within
507 Alienomics, Mathilde Colinet for her help with FISH experiments as well as Nathalie Ver-
508 bruggen for kindly providing *Arabidopsis* plantlets for genome size estimation. The Morphim
509 imaging platform of UNamur is acknowledged for his technical help with the FISH image anal-
510 yses.

511 **Funding**

512 This project received funding from the Horizon 2020 research and innovation program un-
 513 der the European Research Council (ERC) grant agreement 725998 (RHEA) and from BEL-
 514 SPO PRODEX for the ESA selected ILSRA-2014-0106 Project to KVD; from the Fédération
 515 Wallonie-Bruxelles via an 'Action de Recherche Concertée' (ARC) grant to KVD and BH;
 516 under the Marie Skłodowska-Curie grant agreement 764840 to JFF (ITN IGNITE, www.itn-
 517 ignite.eu); from the Fédération Wallonie-Bruxelles via an 'Action de Recherche Concertée'
 518 (ARC) grant to JFF; from the European Research Council under the Horizon 2020 Program
 519 (ERC grant agreement 260822) and JPI-EC-AMR STARCS ANR-16-JPEC-0003-05 grant to
 520 RK; through funding by the French Government "Investissement d'Avenir" program FRANCE
 521 GENOMIQUE (ANR-10-INBS-09); EN obtained a FSR UNamur fund; AH, AD and RA are
 522 Research Fellows of the Fonds de la Recherche Scientifique – FNRS.

523 **Author contributions**

524 Conceptualization : PS, JN, TL, JFF, EN, KVD, AH, AD, MC, RK, AL, MT.
 525 Data curation: PS, JN, AH, AD, RA.
 526 Formal Analysis: PS, JN, TL, JFF, EN, AH, CG, FR, AD, MC, LB, DK.
 527 Funding acquisition: JFF, KVD.
 528 Investigation: PS, JN, TL, JFF, KVD, AH, EN, MC, ML, AD, LB, RK, ED, DK, RA, AL.
 529 Methodology: PS, JN, TL, JFF, EN, AH, CG, RC, ML, AD, MC, LB, RK, DK, MM, NG, AL.
 530 Project administration: JFF, KVD.
 531 Resources: LB, CC, MM, BH, KL, JV, JFF, KVD.
 532 Software: JN, PS, JFF, AH, AD, MC, DK, NG, AL.
 533 Supervision: TL, JFF, KVD, RK, ED, RC, BH.
 534 Validation: JN, PS, JFF, KVD, AH.
 535 Visualization: PS, JN, JFF, EN, KVD, AH, DK.
 536 Writing – original draft: PS, JN, JFF, KVD, AH, MT, ED, TL.
 537 Writing – review & editing: all authors.

538 **Competing Interests**

539 All authors declare that they have no competing interests.

Supplementary Materials

Data generation

Strain culture, library preparation and DNA sequencing We continuously cultivated *A. vaga* individuals from AD008 strain (i.e. same strain as in (9), COI sequence accession number is KM043184) since 2007 in Petri dishes using Spa water, feeding them with sterile extract of lettuce juice and stocking well-grown cultures at -80°C . *A. vaga* individuals were thawed before proceeding to DNA extraction using QIAGEN Genra Puregene Tissue Kit. Genomics Core (UZLeuven) produced PCR-free 250-bp paired-end Illumina reads that were sequenced with a depth of approximately 350x on a HiSeq 2500 sequencing platform. The same procedure was followed in order to obtain high molecular weight DNA using Macherey-Nagel NucleoBond HMW procedure that was subsequently sent to the Genomics Core (UZLeuven) to generate a depth of 200x of PacBio RSII sequencing data. Around 30 μg of high molecular weight DNA was also extracted from living *A. vaga* individuals using the QIAGEN Genra Puregene Tissue Kit and then sent to the Genoscope sequencing center (François Jacob Institute of Biology) which produced 5 ONT libraries, each starting from 2 to 5 μg of DNA, using the 1D ligation sequencing kit (SQ-LSK108) and R9.4 (or R9.4.1) flowcells. This resulted in a sequencing depth of 125x long-reads using Oxford Nanopore Technology (ONT). All samples ID and SRA accession numbers are detailed in Supp. Table S1.

Chromosome conformation capture: Hi-C The Hi-C library construction protocol was adapted from (57, 58). Briefly, individuals from the *A. vaga* AD008 strain were chemically cross-linked for 20 min at room temperature and 30 min at 4°C (with gentle stirring) using formaldehyde (final concentration: 5% in milliQ water; final volume: 50 ml). After fixation the sample was centrifuged for 10 min at 4000 rpm at 4°C . The formaldehyde was then quenched for 5 min at RT and 15 min at 4°C (with gentle stirring) by adding 50 ml of 250mM glycine. The cells were recovered by centrifugation for 10 min at 4000rpm at 4°C , supernatant was removed and pellet stored at -80°C until use. The Hi-C library was then prepared as follows. Cells were resuspended in 1.2 mL of 1X DpnII buffer (NEB), transferred to a VK05 tubes (Precellys) and disrupted using the Precellys apparatus and the following program ([20 sec – 6000 rpm, 30 sec – pause] 9x cycles). The lysate was recovered (around 1.2 mL) and transferred to two 1.5 mL tubes. SDS was added to a final concentration of 0.3% and the 2 reactions were incubated at 65°C for 20 minutes followed by an incubation of 30 minutes at 37°C . A volume of 50 μL of 20% triton-X100 was added to each tube and incubation was continued for 30 minutes. DpnII restriction enzyme (150 units) was added to each tube and the reactions were incubated overnight at 37°C . Next morning, reactions were centrifuged at $16,000 \times g$ for 20 minutes. The supernatants were discarded and the pellets were resuspended in 200 μL of NE2 1X buffer and pooled (final volume = 400 μL). DNA extremities were labelled with biotin using the following mix (50 μL NE2 10X buffer, 37.5 μL 0.4 mM dCTP-14-biotin, 4.5 μL 10mM dATP-dGTP-dTTP mix, 10 μL Klenow 5 U/ μL) and an incubation of 45 minutes at 37°C . The labelling reaction

578 was then split in two for the ligation reaction (ligation buffer – 1.6 mL, ATP 100 mM – 160
579 μ L, BSA 10 mg/mL – 160 μ L, ligase 5 U/ μ L – 50 μ L, H₂O – 13.8 mL). The ligation reactions
580 were incubated for 4 hours at 16°C. After addition of 200 μ L of 10% SDS, 200 μ L of 500 mM
581 EDTA and 200 μ L of proteinase K 20 mg/mL, the tubes were incubated overnight at 65°C. DNA
582 was then extracted, purified and processed for sequencing as previously described (57). Hi-C
583 libraries were sequenced on a NextSeq 550 sequencer (2 \times 75 bp, paired-end Illumina NextSeq
584 with the first ten bases acting as barcodes).

585 **Genome size estimation** The genome assemblies produced by all three methods (Bwise, Flye
586 and Falcon) were markedly smaller than expected based on the generally admitted genome size
587 of 0.25 pg per (non-reduced) oocyte (http://www.genomesize.com/result_species.php?id=5369),
588 equivalent to 244 Mbp for a diploid assembly or 122 Mbp for a haploid assembly. As there is
589 considerable confusion in the literature considering the genome size of *Adineta vaga* (e.g. re-
590 port of a nuclear DNA content of about 0.7 pg (59), nearly 3 times higher than in the Animal
591 Genome Size database although the entry there refers to this article), we decided to perform
592 an independent assessment of the genome size of *Adineta vaga* using flow cytometry, with
593 *Arabidopsis thaliana* ecotype Colombia (for which a haploid genome size of 157 Mbp was pre-
594 viously measured (60) as a genome-size standard for comparison. Nuclei from both species
595 were isolated according to the protocol from the Cystain Pi absolute T (SYSMEX #05- 5023)
596 kit. Briefly, we chopped them together in the same extraction buffer (500 μ L), after which the
597 material was filtered through a 30 μ m nylon membrane. After RNase treatment (80 μ g/ml),
598 the DNA was labeled for 1h in the dark with 2 ml of staining buffer containing 120 μ L of
599 propidium iodide. The labeled nuclei were then analyzed on the CyFlow Space flow cytome-
600 ter (Sysmex) of the research unit "Evolutionary Biology & Ecology" of the Université libre de
601 Bruxelles (ULB). We used a blue laser with an excitation wavelength of 488 nm. The whole
602 procedure was performed three times on different days, using different batches of rotifers and
603 leaves from different *A. thaliana* plants every time, and the .FCS files were analyzed using the
604 FlowJo v10.6.2 software. The estimated haploid genome size is presented in Supp. Fig. 1.

605 **Chromosome painting (FISH)** To assess the colinearity between two chromosomal mark-
606 ers, FISH experiments were performed on samples containing well resolved condensed chro-
607 mosomes. As bdelloids are eutelic, such condensed chromosomes are only found in embryos
608 undergoing nuclear divisions. Particularly, young embryos containing only few nuclei usually
609 exhibit the nicest karyotypes (61). To collect young, ideally one-cell, embryos, about 200 ro-
610 tifers bearing a single egg were first isolated in a petri dish containing a 1% agarose pad and
611 ice-cold Spa® spring water. The agarose pad avoids the embryos to stick at the bottom of the
612 plate and ease their isolation. The rotifers were starved for 24 hours at 4°C and, the next day,
613 about half of the water was removed and replaced by the same volume of fresh water at RT
614 containing lettuce filtrate. Rotifers were incubated at 25°C and, about 3 hours later, all individ-
615 uals were laying eggs almost synchronously. Immediately after laying, the eggs were collected
616 and fixed in methanol (Merck Millipore®, 1070182511): acetic acid glacial (VWR™, 20104-

243) (3:1) solution on ice. After isolation of all eggs, they were collected by centrifugation (14,000 rpm, 2 min, RT), fixed again with methanol: acetic acid glacial (3:1) and stored at 4°C until slide preparation. About 100 embryos bearing one or few nuclei can be collected by this method. For the FISH probe synthesis, we used the Oligopaint strategy that consists in the use of libraries of short single-stranded oligonucleotides (oligos) that are fluorescently labeled to visualize megabases (Mbs) of genomic regions (62). The design of the probes was performed using the OligoMiner pipeline (63) that selects for oligos having similar parameters such as melting temperature (T_m) or the absence of secondary structures. The selected oligos have a 30-42 nt region of genomic homology with a T_m of 42°C flanked by constant nongenic sequences at the 5' end (5'-ccc-gcg-tta-acc-ata-cac-cg-3') and at 3' end (5'-ggt-agc-cac-acg-ctt-cga-tg-3'). These sequences are necessary for the labeling and the amplification of the libraries by PCR (see below). We ordered 6 libraries from GenScript®: (i) library 1 (9.2k oligos) targets the chromosomes 2a/b from 13 to 16 Mbs; (ii) library 2 (7.7k oligos) targets the chromosomes 2a/b from 2 to 6 Mbs; (iii) library 3 (7.7k oligos) targets the chromosomes 6a/b from 2 to 6 Mbs; (iv) library 4 (8.0k oligos) targets the chromosomes 6a/b from 8 to 12 Mbs; (v) library 5 (7.9k oligos) targets the chromosomes 5a/b from 3 to 7 Mbs; and (vi) library 6 (7.8k oligos) targets the chromosomes 5a/b from 9 to 13 Mbs. The probes were labeled and amplified according to the 'One-day' probe synthesis protocol using lambda exonuclease described in (64) (<https://oligopaints.hms.harvard.edu/protocols>). The oligo libraries were first amplified and labeled by PCR. Twenty-four PCR reactions (24 x 50 µl) were performed with 1 U of Q5 high-fidelity polymerase (New England Biolabs®, M0491), 200 µM dNTPs, 0.5 µM of fluorescently labeled forward primer (5'-Fluo/ccc-gcg-tta-acc-ata-cac-cg-3'), 0.5 µM of phosphorylated reverse primer (5'-Phos/cat-cga-agc-gtg-tgg-cta-cc-3'), and 1.25 ng of Oligopaint library. The primers were ordered from IDT®. To perform the two-color FISH experiments, libraries 1, 3 and 5 were labeled with 5Atto488N (green) and the libraries 2, 4, and 6 were labeled with Atto565N (red). The PCR reactions were incubated at 98°C for 5 min, followed by 40 cycles of 30 sec at 98°C, 30 sec at 56°C, and 15 sec at 72°C, and a final extension at 72°C for 5 min. The PCR reactions were then collected and concentrated using the Zymo DNA clean concentrator kit (Zymo research®, D4032). The concentration was performed according to the manufacturer protocol and the libraries were eluted in 2,800 µl of RNase/DNase free water. Lambda exonuclease (New England Biolabs®, M0262) was then used to hydrolyze the 5'-phosphorylated strand of the double-stranded amplicons. DNA eluant (2,200 µl) was processed by 250 U of lambda exonuclease at 37°C for 30 min, and then stopped by incubation at 75°C for 10 min. The single-stranded labeled probes were finally cleaned up using the Monarch PCR & DNA cleanup kit (New England Biolabs®, T1030) following the oligonucleotide cleanup protocol. Probes were eluted in 20 µl of RNase/DNase free water and stored protected from light at -20°C until use. The hybridization of the probes on embryos was adapted from previous protocols (61, 65). At least 100 embryos stored in methanol: acetic acid glacial were dropped onto an uncoated and clean microscope slide (VWR™, 631-1550) and let dry on a wet paper for 30 min. Then, a cover slip (VWR™, 631-1572) was placed over the embryos and they were squashed by gentle pressure on the slide. All following treatments of embryos on slides were conducted

658 in Coplin jars. Embryos were permeabilized in 0.1% saponin (Sigma-Aldrich®, 47036)/0.1%
659 triton X-100 (Sigma-Aldrich®, T8787) in PBS (Lonza®, 17516Q) for 10 min, followed by 2
660 washes of 5 min in PBS. Samples were incubated for 20 min in PBS containing 20% of glycerol
661 (Carl Roth®, 7530.1) and washed again 2 times in PBS. Slides were incubated for 5 min
662 in 2x SSC (SSC 20X, Invitrogen 15557-036) supplemented with 0.1% of Tween-20 (Sigma-
663 Aldrich®, P1379) (i.e., 2x SSCT), and then for 5 min in 2x SSCT supplemented with 50% of
664 formamide (Sigma-Aldrich®, 47671). The slides were then put on top of a thermoblock at 92°C
665 for 2.5 min and transferred in a Coplin jar containing 2x SSCT-50% formamide at 60°C for 20
666 min. The jar was then removed from 60°C and placed at RT for 1 hour. The hybridization mix-
667 ture (50 μ l) composed of 2x SSC, 50% formamide, 1 ul of RNase A (Sigma-Aldrich®, R4642),
668 10% dextran sulfate (Sigma-Aldrich®, S4030), and 10 μ l of each labeled oligo libraries, was
669 placed on a clean cover slip and the slide containing the embryos was inverted onto this cocktail
670 of hybridization. For the two-colors FISH, the oligo library 1 (green) was mixed with the oligo
671 library 2 (red), the oligo library 3 (green) was mixed with the oligo library 4 (red), and the oligo
672 library 5 (green) was mixed with the oligo library 6 (red). The cover slip was sealed with rubber
673 cement and let dry for 5 min at RT. The mounted slide was denatured at 92°C for 2.5 min on a
674 thermoblock, transferred to a dark humidified chamber, and incubated O/N at 37°C. The next
675 day, the cover slip was removed carefully from the slides. The slides were then washed in 2x
676 SSCT at 60°C for 15 min, and in 2x SSCT at RT for 10 min. Chromosomes were counterstained
677 for 20 minutes with 1 μ g/ml DAPI (4',6-diamidino-2-phenylindole; ThermoFisher Scientific,
678 D3571) in 2x SSC. Slides were washed twice in 2x SSC for 10 min, and mounted under a 24
679 \times 32 mm cover slip in Mowiol 40-88 (Sigma-Aldrich®, 324590). Chromosomes and FISH
680 signals were observed under a Leica TCS SP5 fluorescence confocal microscope using the 488
681 nm laser to capture the green signal, the 561 nm laser for the red signal and the 405 nm laser
682 line for the DAPI signal. Images were captured in Z-stacks with the LAS AF software and they
683 were finally processed and analyzed with Fiji (ImageJ, version 2.0.0).

684 **Chromosome-level genome assemblies**

685 **Phased assembly: Bwise** The Bwise assembler v0.1 (<https://github.com/Malfoy/BWISE>)
686 was used on high-coverage Illumina data (sample ID GC047403, see Supp. Table 1) to pro-
687 duce a draft phased genome assembly. We selected a Kmer size parameter of 63 (-k 63) as this
688 produced the most contiguous assembly over the range of tested Kmer sizes: 63, 73, 101, 201.
689 Other parameters were left as default. Bwise rests on a different paradigm than most assem-
690 blers: it starts by generating a de Bruijn graph from the reads to assemble (66), then cleans
691 the graph by removing tips caused by sequencing errors (67), remaps the initial reads on this
692 corrected de Bruijn graph (68), transforming them in super-reads (69). Finally, the resulting
693 super-reads are assembled in a greedy fashion whenever they overlap unambiguously by one
694 or several unitigs. This approach was devised in order to produce an assembly that reflects
695 faithfully the unknown ploidy level of the organism sequenced. Therefore, Bwise will produce
696 haploid assemblies whenever the organism sequenced is haploid, diploid assemblies whenever

697 the organism sequenced is diploid, triploid... etc.

698 **AV20 Haploid assembly: NextDeNovo** Reads quality control was performed with FastQC
699 v0.11.8, seqkit v0.11.0 stats and MultiQC v1.7. PacBio reads (GC032883
700 $\sim 235x$) were scanned for SmartBells adapters that remained in the .fastq files with the mod-
701 ule `removesmartbells.sh` from `bbmap v38.73`. Then modules `rmdup` and `rename`
702 from `seqkit v0.11.0` were used to rename duplicated read names. PacBio reads were sep-
703 arated in two sets: **Assembly (AS)** and **Polishing (PS)**. AS reads were filtered with `filtlong`
704 v0.2.0 on length and cropped on quality. Length threshold was set at 42Kbps, quality
705 thresholds ($min_mean_q \leq 9$ and $min_window_q \leq 8$) were chosen based on the FastQC
706 v0.11.8 stats. PS reads were filtered and cropped using the same parameters except for the
707 length threshold that was set at 5 Kbps. In both AS and PS, the worst 10% reads were dis-
708 carded. After filtering: 0.3x remained in AS and 208.5x in PS. Oxford Nanopore minion reads
709 (BXQ_F $\sim 175x$) were used only for assembly. Reads were filtered with `filtlong v0.2.0`
710 on length and cropped on quality. Length threshold was set at 42Kbps, quality thresholds
711 ($min_mean_q \leq 16$ and $min_window_q \leq 15$) were chosen based on the FastQC v0.11.8
712 stats. Since $\sim 114x$ Illumina HiSeq 2x240bp was also available for this biosample, the 16-mer
713 spectrum from these reads was used to crop and split long reads using `filtlong` parameters
714 `split 250` and `trim`. The worst 10% reads were discarded. After filtering: 38.3x remained.
715 Illumina HiSeq2500 reads (GC047403 $\sim 357x$) were not filtered nor cropped. The sequencing
716 facility (GenomicsCore KUL) provided pre-processed sequences without adapters. Assem-
717 bly was performed with <https://github.com/Nextomics/NextDenovo>, using AS reads (PacBio
718 and ONT). Many parameters were adjusted empirically, see the provided configuration file for
719 details. After this step, all assembly files between each step were sorted and renamed us-
720 ing the `funannotate v1.5.0-12dd8c7 sort` module. The raw assembly contained
721 uncollapsed diploid sequences called haplotigs. Haplotigs were removed with `purge_dups`
722 v1.0.0 (70) using PacBio PS (GC032883 $\sim 208x$). The configuration file was modified to
723 skip the facultative BUSCO and KCM steps. The obtained purged assembly was self aligned
724 with `minimap2 v2.17-r941 -DP` (71) and then visualized with `dgenies` (72) to find
725 haplotype alignments. Haplotigs were not visible compared to the unpurged assembly. The
726 purged assembly was then polished using `HyPo v1.0.3` (73) with Illumina (GC047403 \sim
727 $357x$) and PacBio reads from the polishing set (GC032883 $\sim 208x$). PacBio long reads align-
728 ments were performed using `minimap2 v2.17-r941`. Illumina reads were mapped using
729 `bwa mem v0.7.17-r1188` (74) and `bam` indexing and sorting was done with `samtools`
730 v1.10 (75). Illumina coverage estimates for HyPo was done using the average coverage of the
731 mapping file computed with `sambamba v0.6.8 depth base` (76) (106x) and expected
732 haploid genome size (96m) was set based on the last available flow cytometry genome size
733 estimations results.

734 **Diploid assembly: FALCON** The *de novo* assembly of *Adineta vaga* genome was carried out
735 with diploid-aware long-read assembler FALCON version 0.7.0, FALCON-Unzip and partial

736 FALCON-Phase (only FALCON-Phase Workflow steps 1, 2 and 3) (24). Prior to the assem-
737 bly, Canu error correction module (77) was used for read error correction based on raw PacBio
738 reads. The FALCON software is highly optimised for eukaryote genomes, and uses hierarchical
739 genome assembly process (HGAP). More specifically, reads longer than 15 kb were selected by
740 Falcon as "seed" reads to generate consensus sequences with high accuracy. The pre-assembly
741 steps in FALCON uses DALigner to do all-by-all alignments of the corrected PacBio reads.
742 Long reads were then trimmed at regions of low coverage with FALCON sense parameters (-
743 minidt 0.70 -mincov 4 -maxnread 200) and sensitive DALigner parameters were selected (-h60
744 -e.96 -1500 -s1000) for pre-assembly process. The FALCON pre-assembly resulted in 331 pri-
745 mary contigs of total length 125 Mb, contig N50 of 6 Mb and an additional 36 Mb of "associate
746 contigs" that represent divergent haplotypes in the genome. FALCON-unzip was then used to
747 phase the pre-assembly, producing contiguous leading contigs (named "primary") and associ-
748 ated contigs (i.e. phased, alternate haplotypes). The genome assembly was polished as part of
749 the FALCON-Unzip pipeline using haplotype-phased reads. The haplotigs contain one of the
750 two allelic copies of the heterozygous regions; in this respect, the haplotigs serve as phasing in-
751 formation for the haploid representation. The FALCON-Unzip assembly had 241 primary con-
752 tigs and 999 haplotigs. FALCON-Phase (<https://github.com/phasegenomics/FALCON-Phase>)
753 was developed to resolve haplotype switching in diploid genome assemblies. The FALCON-
754 Phase haplotig placement defines phased blocks in the FALCON-Unzip assembly. The Falcon-
755 Phase Workflow steps 1 and 2 were used to place the haplotigs along primary contigs. Once the
756 haplotig placement file and phase block pairings are done, the primary contigs are cut up into
757 very small pieces at phase block boundaries with Falcon-phase workflow step 3.

758 **Assemblies scaffolding: instaGRAAL** Hi-C contact maps were generated from paired-end
759 reads using the hicstuff pipeline (78) for processing generic 3C data, available at
760 <https://github.com/koszullab/hicstuff>. The backend uses bowtie2 (79) in paired-end mode (with
761 the following options: `-{}-maxins 5 -{}-very-sensitive-local`). We discarded alignments with
762 mapping quality lower than 30. The remainder was converted to a sparse matrix representing
763 contacts between each pair of DpnII restriction fragments. The instaGRAAL program (25)
764 was used in conjunction with the contact maps to scaffold the genomes. Prior to running it,
765 restriction fragments are filtered based on their size and total coverage. Fragments shorter than
766 fifty base pairs are discarded. Then, fragments with coverage lesser than one standard deviation
767 below the mean of the global coverage distribution are also removed from the initial contact
768 map. These fragments were reintegrated later after the scaffolding step. The instaGRAAL
769 scaffolder uses a Markov Chain Monte Carlo (MCMC) method: briefly, the contact data is
770 fitted on a simple three-parameter polymer model. The 3D contacts are exploited and used by
771 the program to infer the relative 1D positions of the sequences and thus the genome structure.
772 To do so, the program attempts to perform a number of operations between each sequence and
773 one of its neighbours (*e.g.* flipping, swapping, merging or splitting contigs) and the operation
774 is either accepted or rejected with a certain probability depending on the likelihood shift. The
775 model parameters are then also updated and a new iteration begins. A set of computations

776 whereby every sequence of the genome has been iterated over this way is called a *cycle*. The
777 scaffolder was run for 100 cycles on the phased and the diploid genome and was run for 50
778 cycles on the AV20 haploid genome, after which convergence in both genome structure and
779 model parameters was evidently apparent. The scaffolded assemblies were then refined using
780 instaGRAAL's instaPolish module, with the aim of correcting the small artefactual inversions
781 sometimes produced by instaGRAAL. The resulting contact map can be seen in Supp. Fig. 2.

782 **Post-treatment of scaffolded assemblies** Post-treatment of the diploid assembly (Falcon):
783 we used the repeat-aware finisherSC tool (80) to upgrade the *de novo* phased genome assem-
784 bly of *Adineta vaga*. Final round of polishing were performed with the Pilon corrector using
785 Illumina data (sample ID GC047403, see Supp. Table 1). Post-treatment of the phased as-
786 sembly (Bwise): : to resolve a remaining fragmentation of one single chromosome (i.e. chro-
787 mosome 5B) after scaffolding with instaGRAAL based on Hi-C data, we established a novel
788 comparative approach that incorporates computational methods to transform fragmented con-
789 tigs into near-chromosome fragments. First, Bwise contigs were aligned against themselves
790 using NUCmer v4.0 (81). Ploidy pairing was evaluated using the online visualization tool,
791 DOT (<https://dnanexus.github.io/dot/>) and we were able to anchor fragmented contigs into a
792 single chromosome using its homologous template (i.e. chromosome 5A).

793 AV20 genome annotation

794 **Transposable elements annotation** TE-like elements, including transposable elements (TEs),
795 were predicted using a combination of two complementary tools: EDTA v1.7.8 (82) and TEde-
796 novo (part of the REPET pipeline) (83, 84). The former relies on structure-based programs
797 allowing for the detection of even single-copy elements, while the latter relies on sequence re-
798 peatedness. The TE-like elements consensus sequences they both produced were then merged
799 and subsequently filtered by performing a basic annotation of the genome with TEannot from
800 the REPET pipeline, and retrieving only consensus sequences with at least one full length copy
801 annotated and discarding sequences corresponding to potential host genes. The 521 retained
802 consensus sequences (293 from EDTA, 124 from TEdenovo) were then used as input for the
803 subsequent genome annotation with TEannot. This resulted in a draft annotation of 8,590 TE-
804 like elements covering 6.57% of the genome. A series of filters were then applied to these
805 annotations using in-house script: i) conserving only retro-transposons and DNA-transposons;
806 ii) with minimal copy length of 250 bp; iii) with minimum identity with consensus of 85%;
807 iv) with a minimal proportion of the consensus overlapped of 33%; v) resolving overlapping
808 annotation. These filtering steps resulted in a final annotation of 841 putative canonical TEs
809 covering 1.98% of the genome. Proportions of TE-like sequences and TEs are shown in Supp.
810 Fig. 9.

811 **Gene annotation** Gene prediction and annotation of AV20 genome were done according to
812 current integrative approaches based on several independent lines of evidence. We first dis-

813 carded scaffolds shorter than 1000 bp using funannotate clean function (85). Repeats in the
814 genome were then soft-masked using bedtools (86) maskfasta function using the draft annota-
815 tion of repeated elements as described above. RNA-Seq data from several cultured clones (see
816 Supp. Table 1) were used to produce *de novo* a transcriptomic assembly with trimmomatic (87)
817 and trinity (88) both under default parameters. This transcriptomic assembly as well as addi-
818 tional RNA-Seq data directly mapping on the genome (see Supp. Table 2) were used as input
819 for the funannotate train function that wrap the PASA pipeline (89) which relies on RNA-Seq
820 to produce high quality annotations. Then, we used a combination of PASA annotations, *de*
821 *nov*o assembled transcriptome, metazoan BUSCO database and the proteic Uniprot database
822 within funannotate predict function. This first produced *ab initio* predictions using Genemark-
823 ES (90), which were then used along with transcripts and proteic data to train Augustus to
824 generate a second set of annotations. Lastly, it used Evidence Modeler as a weighted approach
825 to combine annotations from PASA, Genemark and high quality predictions from Augustus into
826 an integrated gene annotation set. We then used InterProScan5 in order to produce functional
827 annotations to the predicted genes (91) which were then used in combination with busco meta-
828 zoan database using the funannotate annotate function with default parameters. In addition,
829 Ribosomal RNA genes were predicted from the AV20 genome assembly using barrnap (92)
830 (parameters: -kingdom euk). Note that the number of genes annotated differs greatly from
831 the number of genes annotated previously (9). This is mainly due to the structure of the two
832 genome assemblies: the AV13 genome was phased (many pairs of annotated genes correspond
833 to alleles) while the genome assembly we present here is haploid.

834 **Detecting HGTc with a new tool: Alienomics** We used a newly developed tool, named
835 Alienomics, in order to detect Horizontal gene transfers candidate (HGTc). This tool is be-
836 ing submitted and described in detail elsewhere. Briefly, its approach first integrates several
837 lines of quantitative evidence into a score for every predicted gene. This gene score is based
838 on several blast results (i.e. against Uniref50 database, a user-defined set of closely-related
839 reference genomes, bacterial rRNA database, BUSCO database) as well as on read coverage
840 and GC content. It represents how "alien" or "self" a given gene is. Note that when con-
841 sidering blast results the taxonomy of multiple best-hits are parsed and evaluated in order to
842 determine whether the query origin belong to "self" or to "alien". We then superimpose this
843 qualitative synteny information to the quantitative gene score in order to discriminate if alien
844 genes stemmed from contaminant or from HGT. For this, scaffolds are being given a score
845 based on the integration of all the gene scores, slightly modified using expression level based
846 on RNA-Seq (in order not to penalize scaffolds including many true HGTs). This scaffold score
847 represents whether it originated from a contaminant or from the genome under study. Syn-
848 teny is then taken into account by comparing gene scores to their respective scaffold scores to
849 validate a HGTc. For example, an "alien" gene on a "self" scaffold corresponds to a HGTc
850 while an "alien" gene on an "alien" scaffold is a contaminant. Alienomics is available here:
851 <https://github.com/jnarayan81/Alienomics>. Within Alienomics, results for each criteria (e.g.
852 blast bitscores, GC content, coverage) are transformed into criteria scores ranging from -1 to

853 +1. Criteria scores from blast results are turned into negative values if the taxon id from the best
854 representative match among all hits do not belong to a user-defined clade (such as "metazoa").
855 Gene scores result from the combination of criteria scores and correspond to the hyperbolic tan-
856 gent of the sum of criteria scores multiplied by a ratio that depends on the number of informative
857 criteria (e.g. number of criteria for which the value is different from "0"). Scaffold scores result
858 from the combination of gene scores (with the addition of expression score based on RNA-Seq
859 data) and correspond to the hyperbolic tangent of the sum of gene scores multiplied by the
860 square root of the number of genes and normalized by gene lengths. Coverage information
861 was computed from raw ONT reads using minimap2 (parameters as follows: -ax map-ont -c
862 -Y). Alienomics was run here under the following parameters: level_upto = metazoa; gc_filter
863 = 26:38 ; value = 1e-01; qcover = 0; bitscoreCutoff = 150; coverage = 100; ignoretaxid =
864 104782—10195—96448—249248—1813166—104781—4513—112509—9606—7574—42192—29159—28
865 HGTc were categorized as such under the following default thresholds: genescore = 0.5; scaf-
866 foldscore = 0.5.

867 **Endogenous viral elements detection** Sequences showing similarity to viral genes were
868 searched in the AV20 genome assembly by performing a diamond (93) blastx search (options: -
869 max-target-seqs 1 -range-culling -min-score 40 -more-sensitive -F 15) using AV20 scaffolds as
870 queries all viral proteins extracted from the nr database of NCBI (February 2020). Proteins from
871 two viral families were excluded from this database (i.e. Retroviridae and Hepadnaviridae) to
872 avoid false-positive blast hits corresponding to the reverse-transcriptases of *A. vaga* retrotrans-
873 posons. All *A. vaga* sequences showing similarity to a viral sequence were then used as queries
874 to perform a reciprocal diamond blastx search against the entire NCBI nr protein database. All
875 sequences aligning with a higher score to a viral sequence than to a non-viral sequence were
876 annotated as viral-like sequence.

877 AV20 genome analyses

878 **Ploidy, synteny and colinearity among the three *A. vaga* genome assemblies** Genome as-
879 sembly tools rely on various assumptions including the ploidy level of the organism under
880 study. In order to circumvent potential impact of such ploidy assumptions on genome struc-
881 ture, we compared our three new genome assemblies. First we evaluated the classical genome
882 assembly statistics using in-house script (see Figure 1A). We then used the illumina reads (i.e.
883 GC047403, see Supplementary Table 1, as input for the *comp* function of the KAT software (94)
884 which uses k-mers distribution in order to explore ploidy levels of *A. vaga* genomes (see 1A).
885 Genomes were aligned pairwise using nucmer 3.1 (using -maxmatch option) (95), the results
886 of which were converted into paf format using minimap2 pafutils script (71). We then used the
887 online tool D-GENIES (72) to visualise the three pairwise alignment as dotplots (see Supp. Fig.
888 3, 4 and 5).

889 **Read depth and heterozygosity** Average coverage on AV20 genome assembly was computed
890 independently for the Illumina PE reads, ONT reads and PacBio reads, on 100 Kbs windows.
891 The mapping was performed with bwa mem 0.7.17 (74) on default settings for the short-reads
892 reads. Heterozygosity analysis was performed using GATK 4.1.0.0 (96) on Illumina PE reads
893 for genotyping all sites (HaplotypeCaller function with -ERC GVCF option). This was done
894 for all samples analyzed (i.e. GC047403, BXQF, ERR321927). The resulting gvcf files were
895 combined (CombineGVCFs function) and were then jointly genotyped (GenotypeGVCFs func-
896 tion) (97). Distribution of heterozygous sites are shown on Figure 2A.

897 **Homoeologous divergence** The self alignment of AV20 genomic sequences obtained using
898 nucmer 3.1 (-maxmatch option, see methods on synteny and colinearity above) (95) was re-
899 used in order to evaluate the genomic divergence between homoeologous chromosomes. The
900 paf alignment file was filtered using custom script in order to only retain genomic alignments
901 between homoeologous regions ranging from 500 to 10,000 bp (see Supp. Fig 8). Note that
902 the divergence between homoeologues (and alleles) measured here are much lower than the
903 measures previously reported (9). This is because we aligned genomic regions at the nucleotide
904 level using nucmer. On the contrary, previous study aligned CDS at the proteic level using MC-
905 ScanX which then guided corresponding alignment at the nucleotide level, producing additional
906 indels due to the existence of frameshifts.

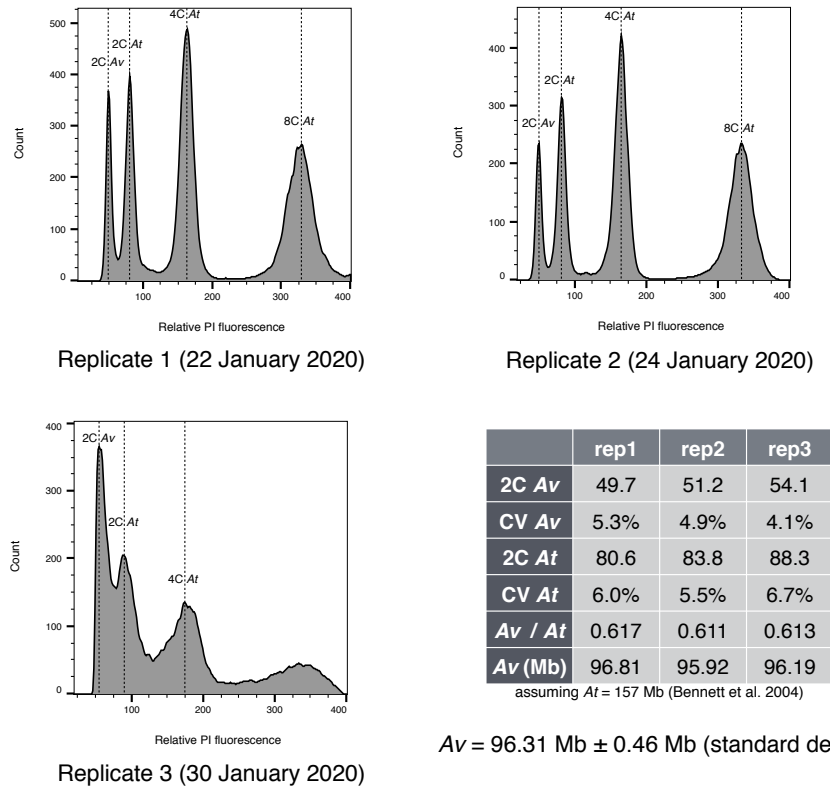
907 **Detecting gene synteny** Protein sequences from annotated genes were used as input for MC-
908 ScanX (98) in order to detect blocks of gene synteny (parameters: -s 5 -b 1). Home-made
909 script was used in order to only retain genes that composed syntenic blocks between homoe-
910 ologous chromosomes. Among the 31,582 annotated proteins in *A. vaga*, 9,726 of them had
911 a proteic counterpart in a synteny block on their respective homoeologous chromosome (i.e.
912 30.79% of proteins). Note that the number of gene existing prior to the tetraploidyza-
913 tion of the genome is very likely larger than this estimate, as any gene loss, translocation or structural
914 re-arrangements would break gene synteny. All synteny blocks are depicted as grey links on
915 Figure 1B. The same procedure was followed to detect colinear blocks of synteny using only
916 the 2,679 HGTc (corresponding synteny blocks are depicted as colored links on Figure 3).

917 **Gene enrichment analyses** GO terms from functional annotation of the haploid genome
918 were extracted from gene annotation (see gene annotation section above). The 2,422 recent
919 HGTc and the 257 ancient HGTc were respectively compared to the entire gene set of the AV20
920 genome containing 32,378 proteins. Enrichment analyses were performed using topGO pack-
921 age with a fisher test and the "elim" algorithm (99). Results are presented in Supplementary
922 Table 2.

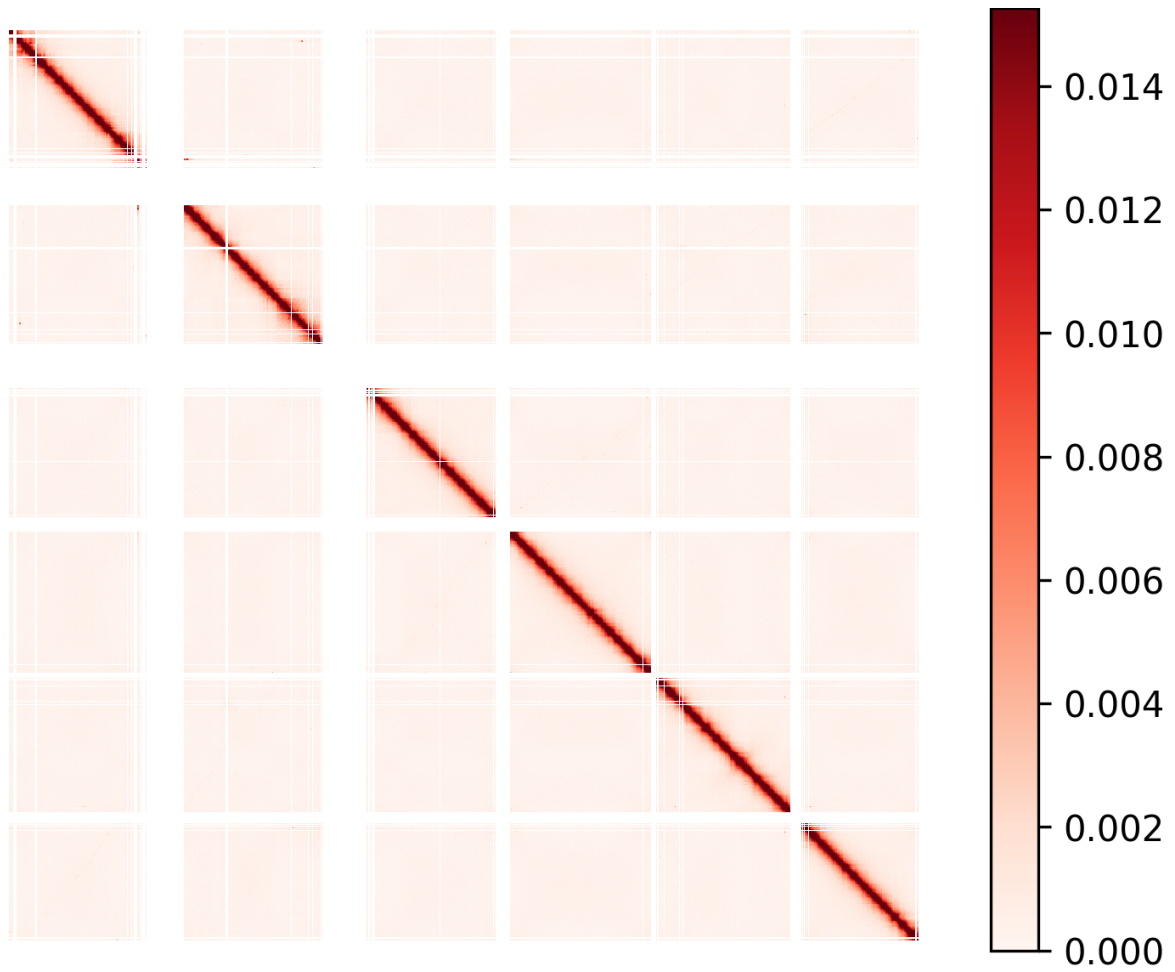
923 **Re-appraisal of the AV13 Genome**

924 **Investigating AV13 Breakpoints** The previously identified synteny breakpoints in the genome
925 of *A. vaga* 2013 (AV13 (9)) were verified by mapping the ONT reads (median size: 4,149 kb;
926 max size: 353,147 kb) produced in this study onto the AV13 genome according to the following
927 procedure: i) ONT reads were filtered with Porechop (100) to discard long reads containing
928 adapters. This discarded 1,202 out of the 1,634,477 reads; ii) Reads were mapped onto the
929 AV13 genome using NGMLR (101) with default parameters. This tool was selected for its
930 accuracy when aligning long reads in a context of structural variation; iii) The scaffold of inter-
931 est (i.e. scaffold1 from AV13) was aligned against the rest of the AV13 genome using Sibelia
932 v3.0.7 (102) with the following parameters: `'-s loose -m 10000 --gff'`. iv) The new
933 AV20 haploid genome assembly was aligned against the AV13 genome using the same proce-
934 dure as in the previous step; v) Synteny block from Sibelia were used to determine the genomic
935 windows containing the putative breakpoints described previously (9). These regions were man-
936 ually screened using Tablet (103) to visualize the alignment of ONT reads. We notably checked
937 for the presence of clipped regions. Every window contained at least one clipped region (i.e. a
938 position that is not supported by a single long read) which we reported as screenshots in Supp.
939 Fig. 6.

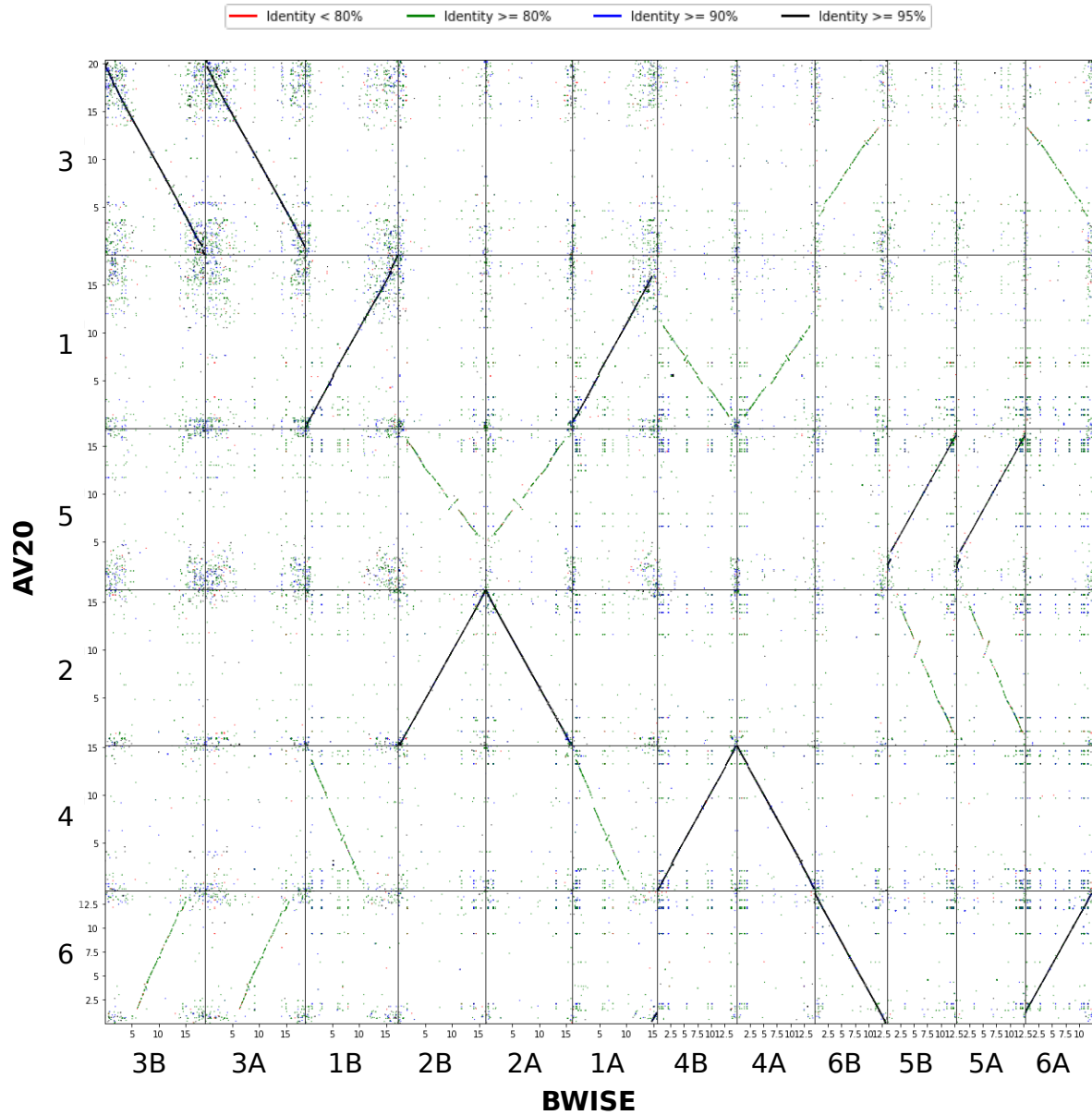
940 **Investigating AV13 Palindromes** Palindromes previously reported in the AV13 genome were
941 investigated in the light of our new AV20 assembly. We first *de novo* determined the location
942 of palindromes in AV13 by filtering ONT long reads, mapping them onto AV13 genome using
943 NGMLR (101) with default parameters and subsequently detecting the palindromic breakpoints
944 (PBR) using a in-house tool, huntPalindrome (available at
945 <https://github.com/jnarayan81/huntPalindrome>). Each PBR location was extended by 2.5 kbp
946 on both sides to produce PBR windows within which we checked for clipped long reads us-
947 ing in-house script. Additionally, we used the alignment between AV13 and AV20 genomes (as
948 described in the previous paragraph) to show how these 20 palindromes from AV13 were assem-
949 bled in AV20 (see Supp. Fig. 7). All these palindromes were collapsed into non-palindromic
950 regions in the new AV20 genome assembly.



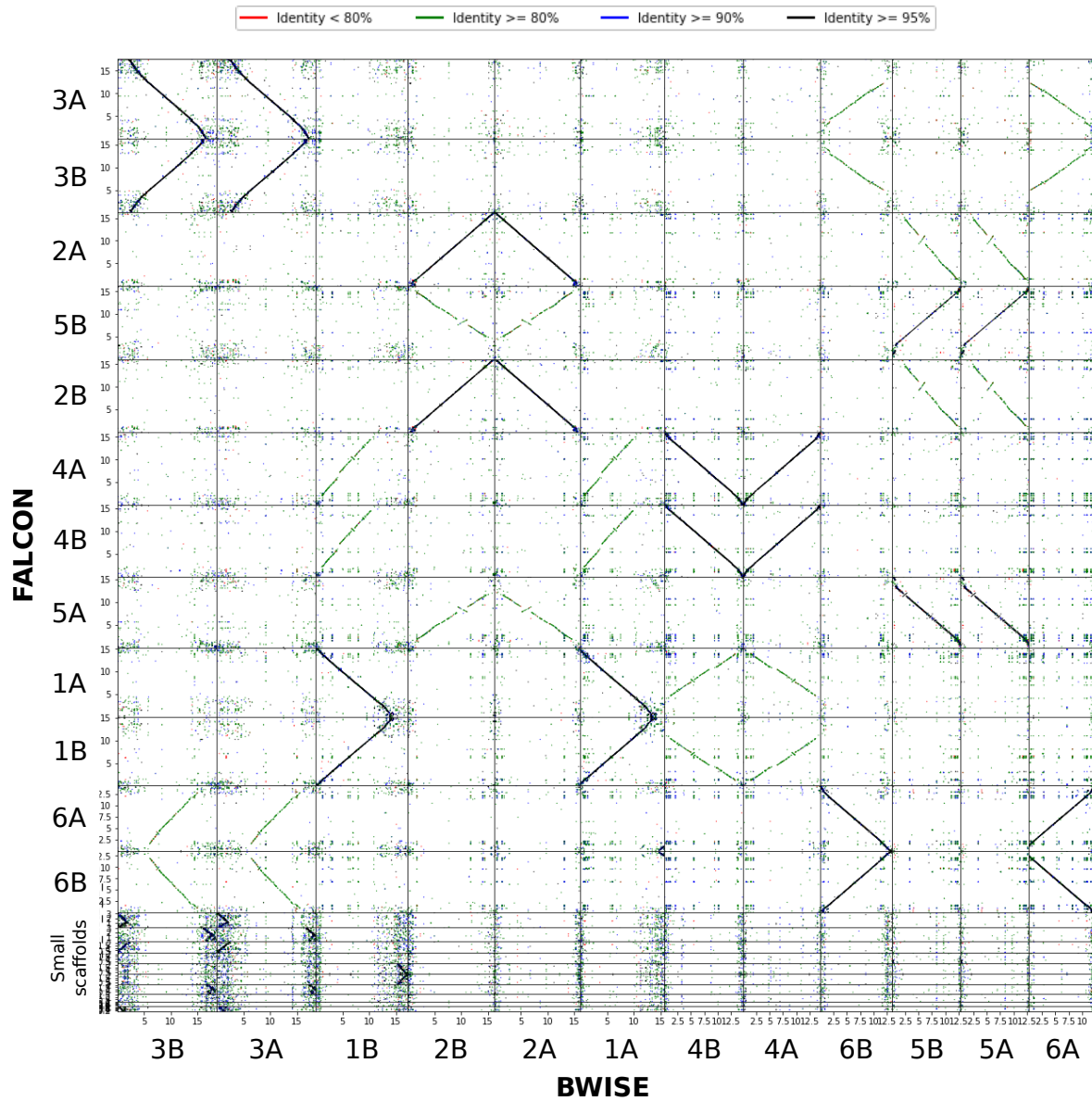
Supplementary Figure S1: *Genome size estimation*. Flow cytometry measurement of the genome size of *Adineta vaga* (*Av*) by comparison to *Arabidopsis thaliana* cultivar Colombia (*At*). Genome size length of *A. thaliana* 1C is 157 Mbp. Assuming *A. vaga* is diploid, the ratio between the two species (i.e. *Av/At*) is about 0.61, leading to the estimation that *A. vaga* 1C genome size is 96.3 Mbp



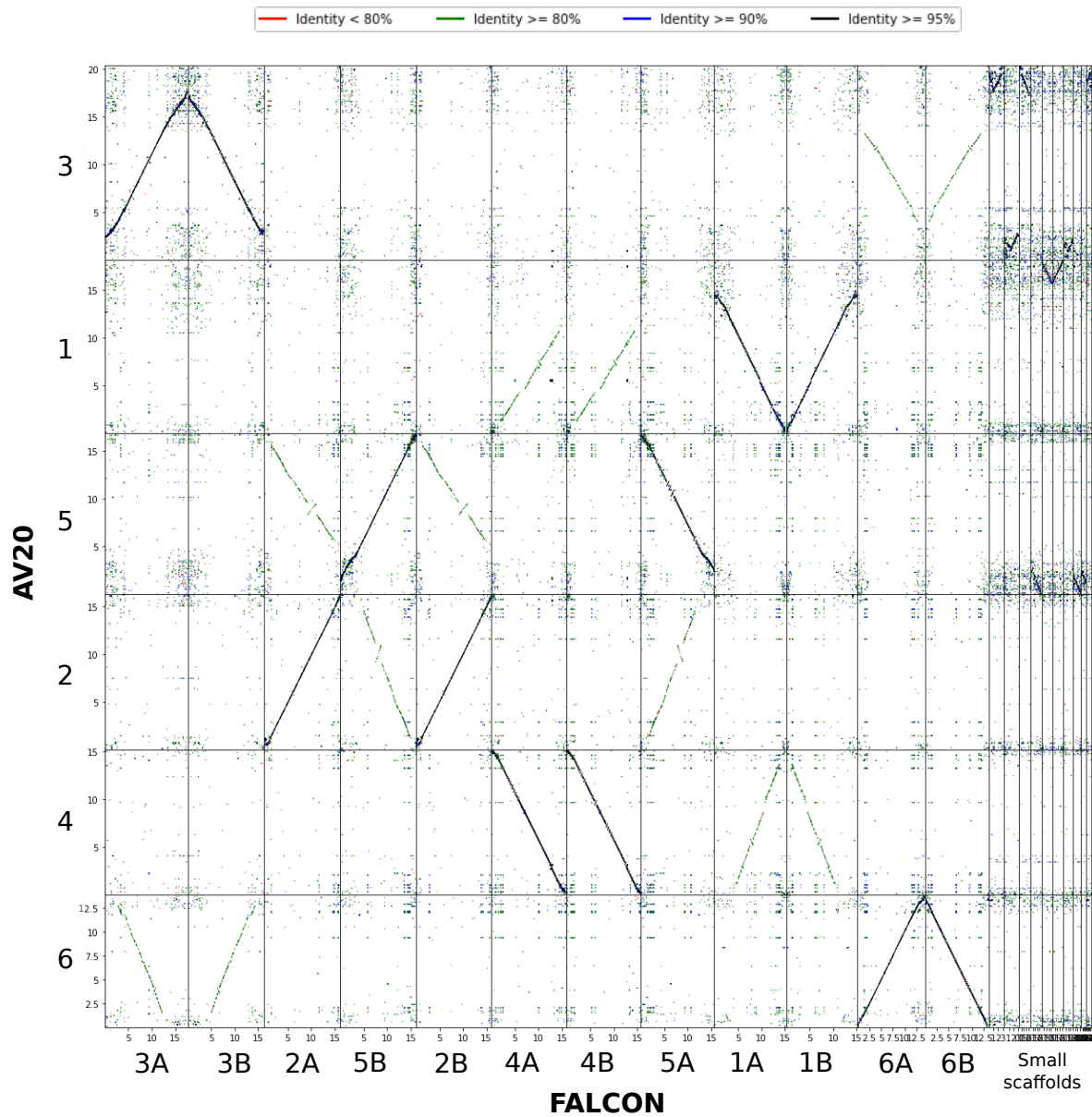
Supplementary Figure S2: *AV20 contact map*. Proximity ligation sequencing data (Hi-C) contact map on AV20 assembly after scaffolding using instaGRAAL and instapolish.



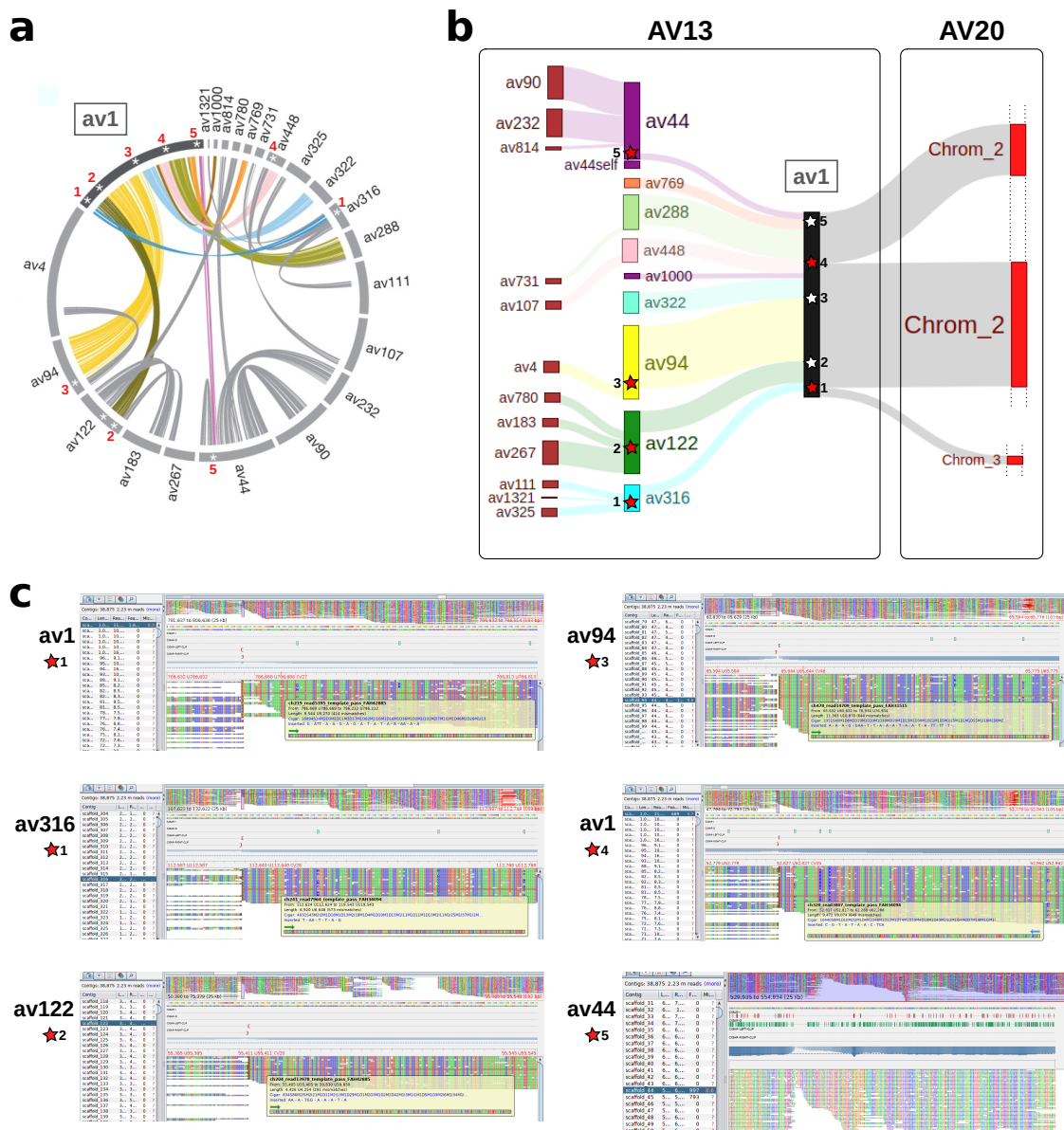
Supplementary Figure S3: *Haploid versus phased genome dotplot*. Pairwise alignment of the haploid assembly (AV20) against the phased assembly (B-WISE), visualised using D-GENIES.



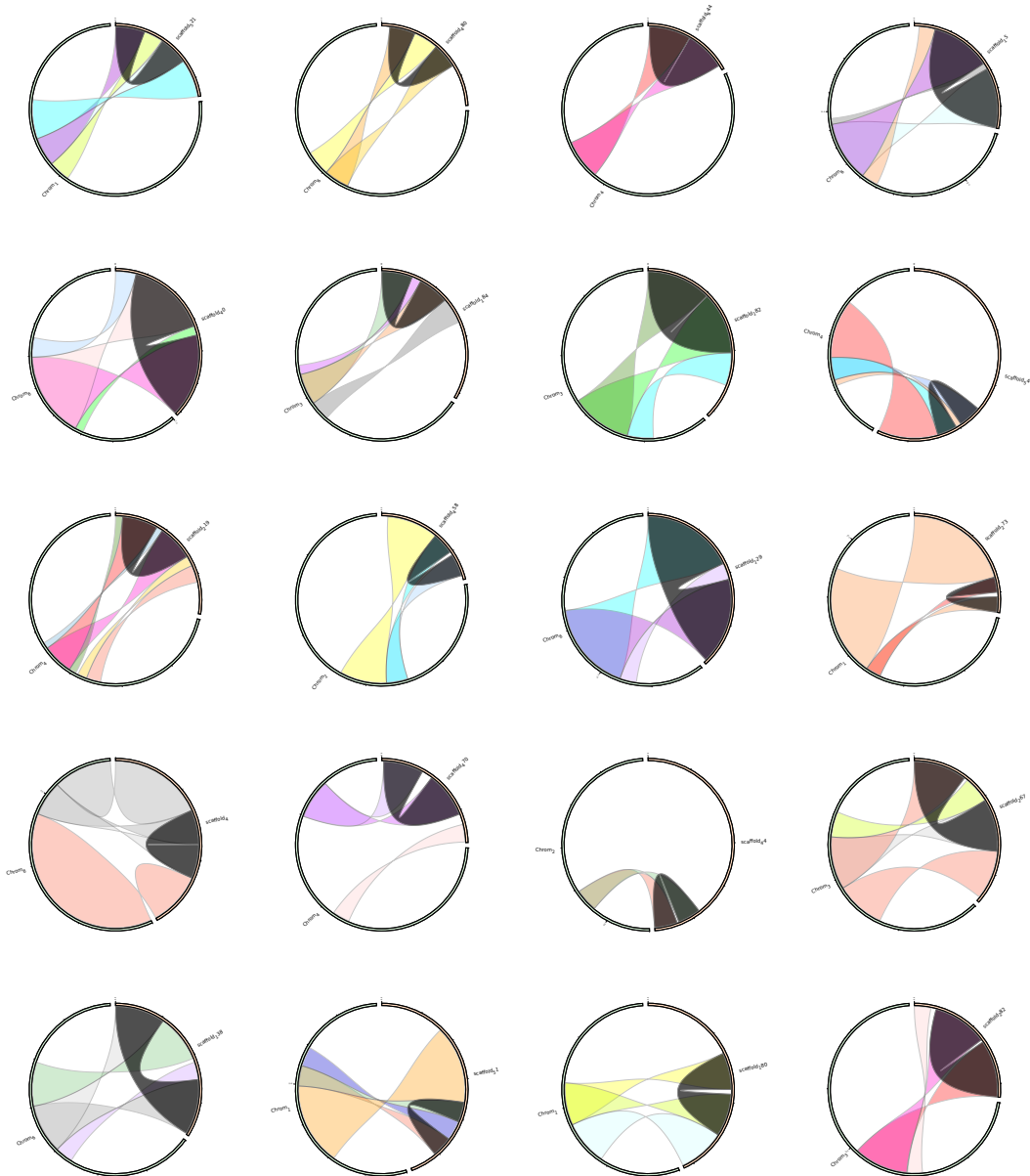
Supplementary Figure S4: *diploid versus phased genome dotplot*. Pairwise alignment of the diploid assembly (Falcon) against the phased assembly (BWISE), visualized using D-GENIES.



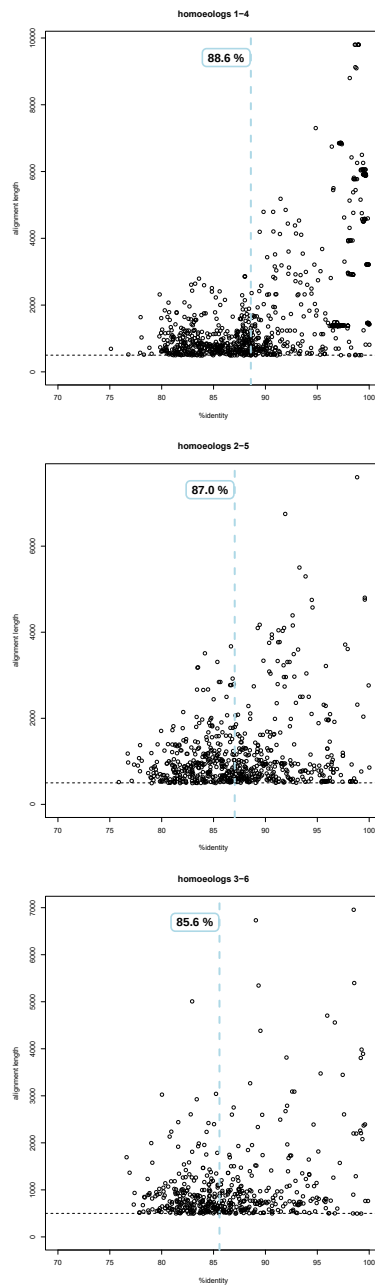
Supplementary Figure S5: *Haploid versus diploid genome dotplot*. Pairwise alignment of the haploid assembly (AV20) against the diploid assembly (FALCON), visualized using D-GENIES.



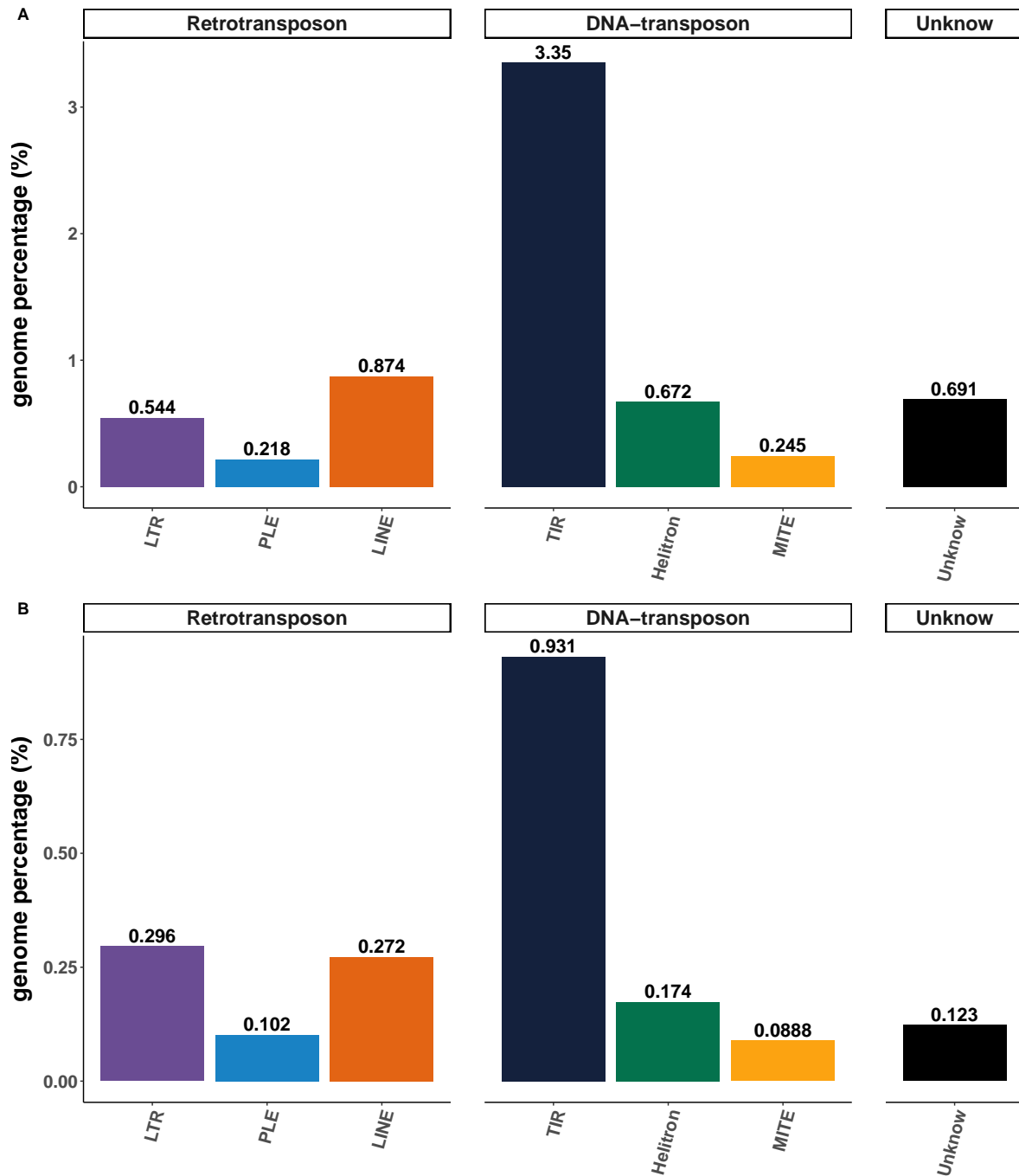
Supplementary Figure S6: *Invalidating AV13 breakpoints.* a) Schematic view of AV13 genome syntenicity depicting five putative colinear breakpoints on the scaffold av1 and its homologous counterparts (adapted from (9)). b) Schematic view of syntenicity alignment between the scaffold av1 from AV13 and the new AV20 genome. The 5 putative colinear breakpoints corresponding to panel a are also depicted. Red stars indicate genomic region in AV13 assembly that are not supported by long reads, while white stars indicate regions supported by long reads. Note that regions supported by long-reads in scaffold av1 (white stars 2, 3 and 5) systematically corresponded to a red star in their homologous counterparts in AV13 (red stars on other AV13 scaffolds), indicating that the colinear breakpoint was, in fact, not supported. c) Screenshots of the alignment of long-reads on AV13 assembly (using Tablet) depicting clipped regions at the location of putative colinearity breakpoints. These cases correspond to the red stars depicted on panel b.



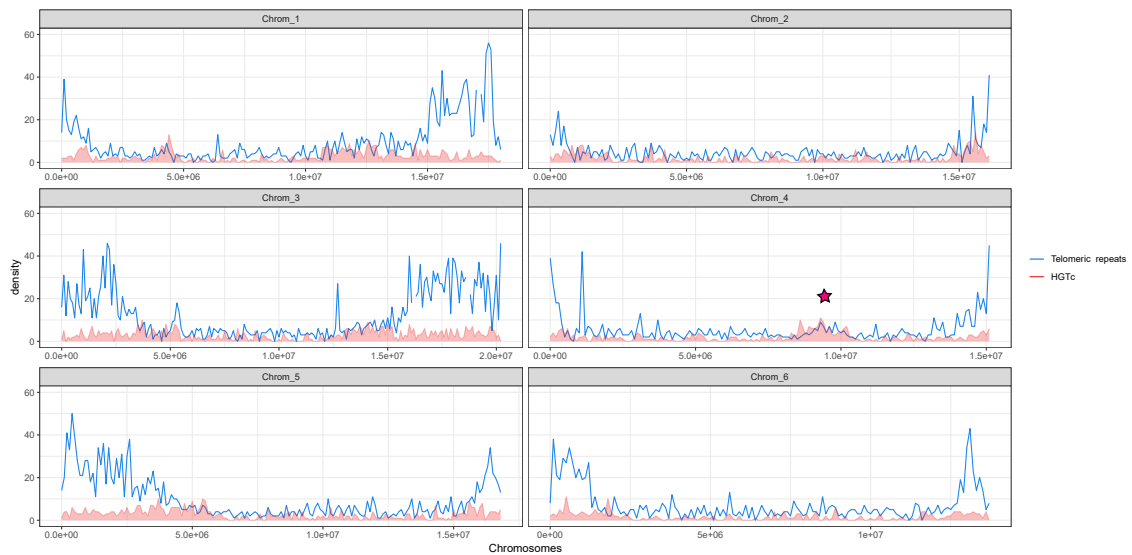
Supplementary Figure S7: *Invalidating AV13 palindromes.* Alignment of the AV13 genome assembly (9) against the new AV20 genome assembly shows the total absence of previously reported palindromes. Orange bars represent scaffolds from 2013 assembly and green bars represents chromosomes assembled in the present study. Palindromic regions in 2013 assembly are shown in dark grey.



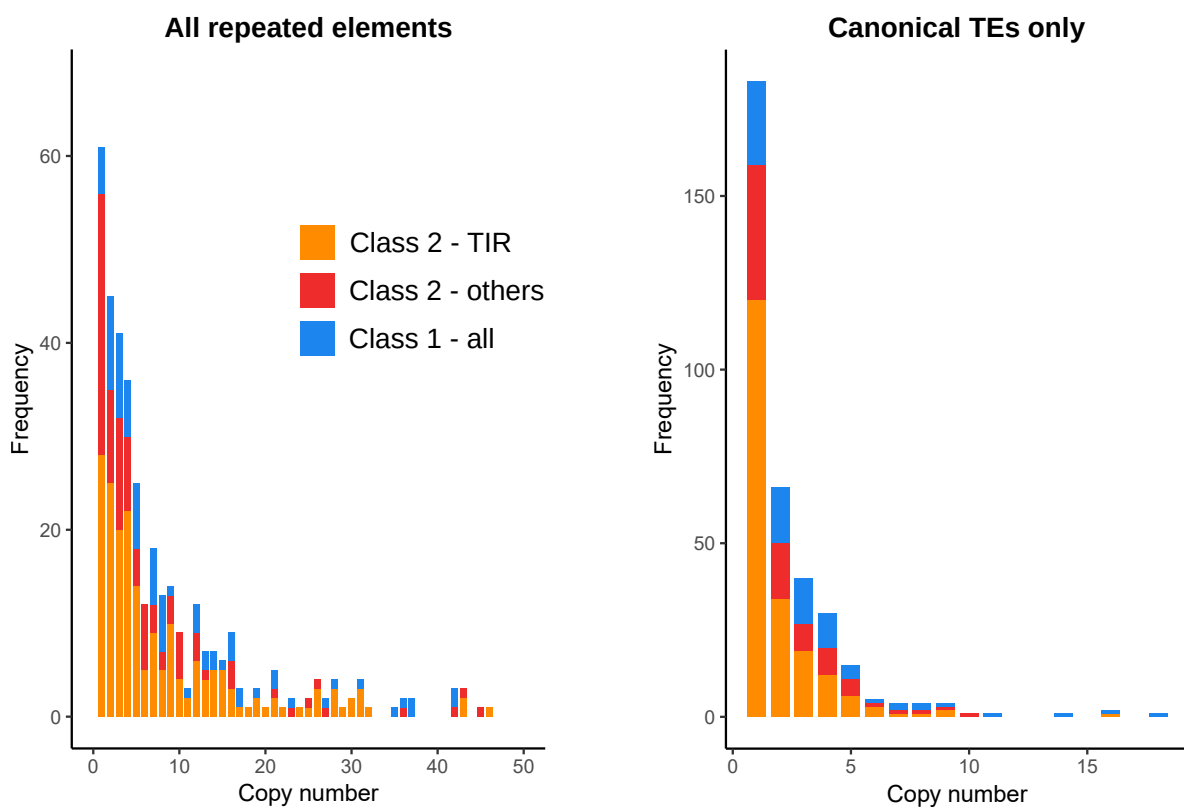
Supplementary Figure S8: *Genomic divergence between homoeologous chromosomes.* Chromosome pairwise alignments length and identity percentage for every pair of homoeologous chromosomes (i.e. chromosomes 1 and 4, 2 and 5, 3 and 6). Median identity percentage between homoeologous chromosome is indicated on each plot (i.e. vertical dotted blue line). Alignments shorter than 500 bp (i.e. horizontal dotted black line) and longer than 10,000 bp were discarded.



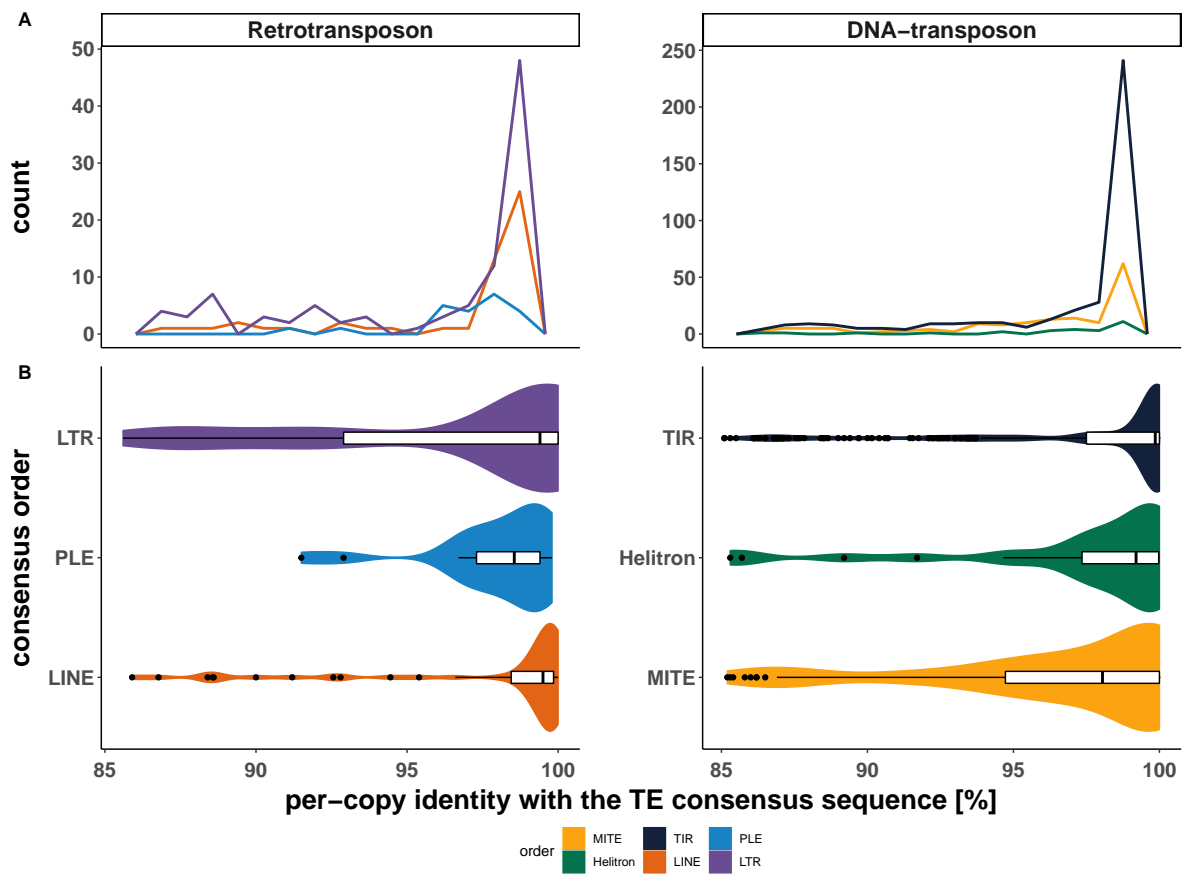
Supplementary Figure S9: *Repeated and transposable elements*. Proportion of the genome covered by each TE order for: a) draft annotation including all repeated elements likely to be related to transposable elements; b) filtered annotation, including only putative canonical TEs.



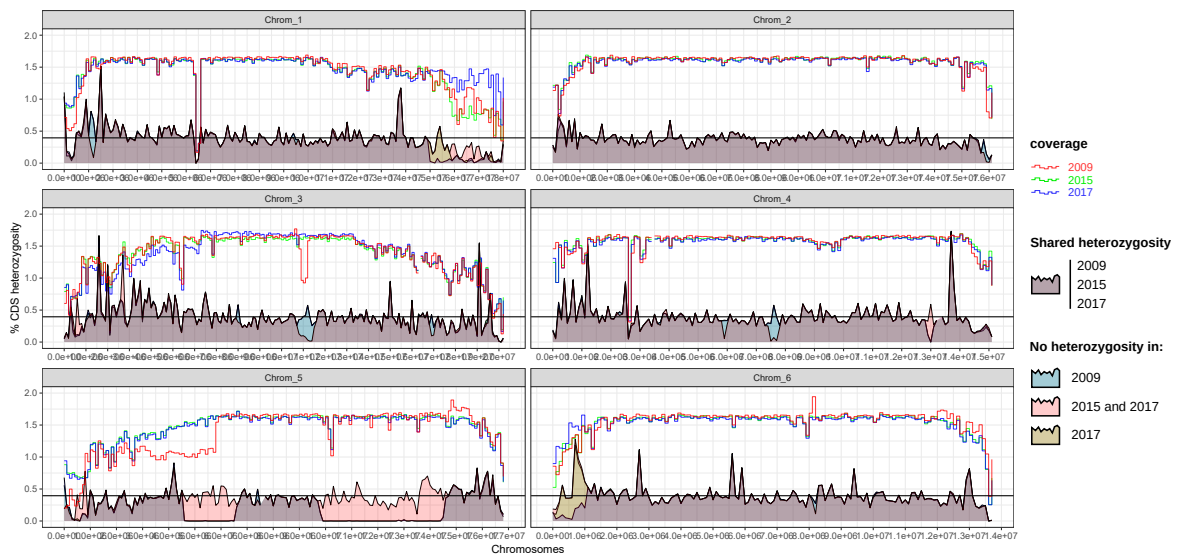
Supplementary Figure S10: *Telomeric repeats and HGTc distribution.* Telomeric repeats (blue) are mostly found in telomeres and subtelomeric regions. Note the small increase of telomeric repeats colocalizing with local hotspot of HGTc (red) on chromosome 4.



Supplementary Figure S11: *Transposable elements per-consensus copy number distribution.*



Supplementary Figure S12: *Transposable elements per-copy identity with consensus.*



Supplementary Figure S13: *CDS Heterozygosity distribution*. Heterozygosity level was normalized by the density of CDS per windows. Sample names and colors as in Figure 2.

Chapter 6

Genome assembly of the coral

Astrangia poculata

6.1 Introduction

The species *Astrangia poculata* [265], also called the Northern star coral, is a temperate hard coral distributed across a wide range of latitudes in the western Atlantic ocean [266]. It belongs to the class Anthozoa, a division of cnidarians that includes hard corals, soft corals, and sea anemones. Along with its adaptation to temperature variations, this coral has a facultative symbiosis with algae from the family Symbiodiniaceae, making it a compelling model to study coral response to environmental changes. To this end, we assembled its genome which will constitute a resource for downstream analysis. The genome had previously been assembled with a combination of Illumina and Hi-C reads; although this first version was highly contiguous, its size was excessively small compared to the expected genome size, and the draft had a poor completeness. I assembled the genome *de novo* with newly sequenced Nanopore reads, which I combined with Illumina and Hi-C reads to produce an improved reference sequence.

6.2 Material & Method

6.2.1 Sequencing data

High-molecular-weight DNA was extracted by Dovetails Genomics. The sample was further purified with AMPure XP beads and fragments were selected on their size with Circulomics Short Reads Eliminator XS.

A Nanopore library was prepared with the Ligation Sequencing Kit LSK109, starting with 2.1 μg of DNA, and yielded 1.4 μg of DNA. The library was sequenced with a MinION on a R9.4 flowcell, with fast Guppy v4 basecalling. The flowcell was washed and reloaded three times (281 ng of DNA for the first load, 187 ng for subsequent loads) and ran for 89 hours. A total output of 6.79 Gb was obtained with an N50 of 18 kb and an N90 of 5 kb (Table 6.1). Adaptors were removed using Porechop [267] with default parameters. After trimming, the dataset reached 6.77 Gb.

Dovetails Genomics produced two shotgun Illumina datasets of paired-end 150 bp reads: one with 414 million reads and an estimated insert size of 395 bp, and the second with 235 million reads and an estimated insert size of 484 bp (Table 6.1). Adaptors were removed using cutadapt github.com/marcelm/cutadapt.

Dovetails Genomics also provided three Hi-C libraries with 198 million, 266 million and 257 million paired-end 150 bp reads (Table 6.1). The reads were trimmed of the adaptors with cutadapt.

Table 6.1: *Astrangia poculata* sequencing datasets.

Reads	Length	N50	Size
Hi-C	2*150 bp	-	217 Gb
Illumina	2*150 bp	-	195 Gb
Nanopore	-	18 kb	7 Gb

6.2.2 Genome size estimation

Dovetails Genomics estimated the genome size to 462 Mb. I estimated the genome size using the second shotgun Illumina dataset of 235 million reads and the module `kmercount.sh` from BBtools [236]; the tool predicted a haploid size of 453 Mb, a ploidy of 2, and 40.95% of repeats.

6.2.3 Genome assembly

Five assemblers were tested with default parameters: Canu [95], Ra [103], Raven [104], Flye [97], wtdbg2 [108]. Purge Haplotigs [146] was run on the wtdbg2 assembly with default parameters, using the full shotgun Illumina datasets mapped with bowtie2 [268]. The wtdbg2 assembly was polished with HyPo [138], using the full shotgun Illumina datasets mapped with bowtie2. Hi-C reads were mapped to the wtdbg2 assembly and processed using hicstuff [260], available at github.com/koszullab/hicstuff, with the parameters `--enzyme DpnII --iterative --aligner bowtie2`. The draft assembly was then scaffolded using instaGRAAL [186], with default parameters (`--levels 4 --cycles 100 --coverage-std 1, --neighborhood 5`). The output was refined with the module `instaGRAAL-polish`.

6.2.4 Assembly evaluation

BUSCO v4 [30] was run against metazoa odb10 (954 features) without the parameter `--long`. k -mer completeness was calculated by running KAT comp v2.4.2 [238] with the full shotgun Illumina datasets. The contact map was built using the hicstuff pipeline, with the three Hi-C libraries, and `hicstuff view` with the parameter `--binning 200`.

6.3 Results

The assembly provided by Dovetails Genomics had chromosome-level scaffolds, but one of its major flaws was that its total size only reached 252 Mb (Table 6.2) whereas the genome size was estimated to 462 Mb by Dovetails Genomics and to 453 Mb by BBtools.

Table 6.2: Basic statistics of *Astrangia poculata* assemblies, presenting the strategies that were used for each assembly (purging haplotigs, assembly polishing, scaffolding), assembly size, number of contigs, N50, number of BUSCO single complete features and BUSCO duplicate complete features.

Assembler	Purging	Polishing	Scaffolding	Assembly size	# contigs	N50	BUSCO	
							single	dup.
Dovetails	-	-	-	252 Mb	7848	16.8 Mb	60.0%	0.2%
Canu	×	×	×	597 Mb	8317	97 kb	56.0%	7.9%
Flye	×	×	×	719 Mb	7259	167 kb	67.9%	8.9%
Ra	×	×	×	271 Mb	2851	115 kb	43.9%	0.3%
Raven	×	×	×	400 Mb	2912	172 kb	52.3%	0.4%
wtdbg2	×	×	×	476 Mb	4423	439 kb	55.1%	0.3%
	✓	×	×	452 Mb	2995	475 kb	54.6%	0.3%
	✓	✓	×	458 Mb	2995	480 kb	87.7%	2.4%
	✓	✓	✓	458 Mb	488	31.0 Mb	89.1%	1.2%

Ra and Raven both produced smaller assemblies than expected, with the Ra assembly size close to the one of Dovetails Genomics. Canu and Flye both produced assemblies quite larger than expected, which is likely due to uncollapsed haplotypes, as is shown by the increased percentages of duplicated BUSCO complete features. wtdbg2 produced the most convincing draft, with an assembly size close to expectations and the highest N50. Purge Haplotigs reduced the number of contigs from 4423 to 2995 and slightly increased the N50 from 439 kb to 475 kb. After polishing, the overall number of complete BUSCOs went from 54.9% to 90.1%. Scaffolding with instaGRAAL yielded 14 chromosome-level scaffolds, with sizes ranging from 21.1 Mb to 54.8 Mb. The final assembly contains 14 scaffolds and, after removing small sequences, has a size of 455 Mb and a BUSCO completeness of 90.4%. The KAT plot shows two peaks, as the species is diploid 6.2. Low multiplicity (or erroneous) k -mers are absent from the assembly, as expected. A part of heterozygous k -mers are represented once in the assembly and the rest are not, as only one haplotype is represented for heterozygous regions in collapsed haploid assemblies. The majority of homozygous k -mers are represented once, although there are some missing and duplicated k -mers.

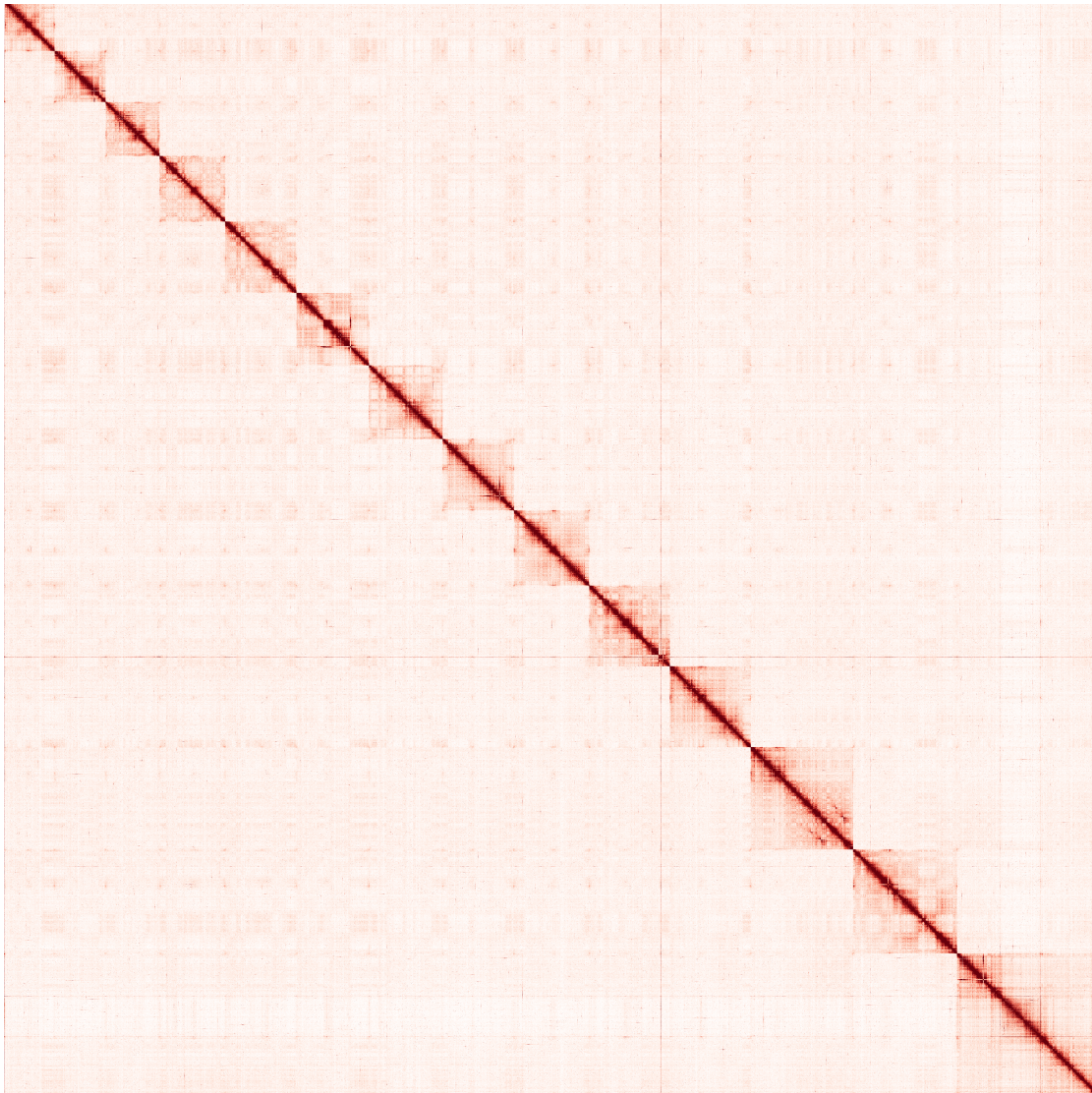


Figure 6.1: Contact map representing the 14 chromosome-level scaffolds of the final assembly (combining wtdbg2, Purge Haplotigs, HyPo and instaGRAAL).

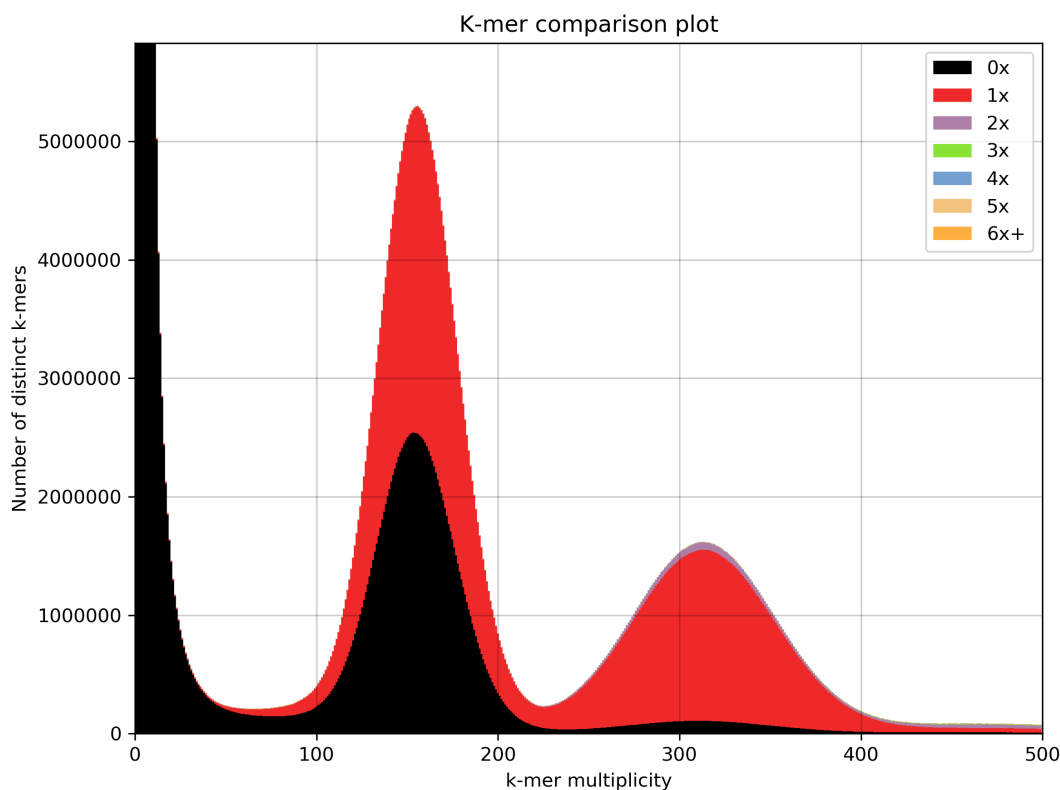


Figure 6.2: *k*-mer analysis of the chromosome-level scaffolds of *Astrangia poculata*.

6.4 Discussion

The initial assembly of *Astrangia poculata* reached a high contiguity, but its small size and completeness indicated that the assembly was incomplete. The new assembly has a size and a number of chromosome-level scaffolds within the expected range, as well as high BUSCO and *k*-mer completeness. This demonstrates that, although Hi-C scaffolding is a robust method to achieve chromosome-level assemblies, the quality of the input contigs is crucial. In this case, the small size of the initial assembly may result from the high repetitive content (estimated to 40.95%) which is typically poorly handled by short reads; repeats are better resolved by long reads as their length can cover full repetitive regions [49]. Interestingly, a low coverage of long reads (about 15X) was sufficient to yield an assembly with a size close to the estimated genome size, and polishing with a high-coverage short read dataset further improved the completeness.

Among assemblies of anthozoan genomes, only a few reached chromosome-level scaffolds: *Acropora millepora*, *Xenia* sp., and the assembly of *Astrangia poculata* presented here (Table 6.3). The most recent version of *Acropora millepora* combines long reads, linked reads and genetic maps, while *Xenia* sp. and

Astrangia poculata were both obtained with long reads, short reads, and Hi-C. One genome was scaffolded with an alternative *in vitro* Hi-C protocol, called CHICAGO, but this approach led to a poor contiguity, and regular Hi-C should be favored for chromosome-level assemblies. Many genomes of hard corals (Scleractinia) were assembled with Illumina reads and scaffolded with mate pairs, and particularly for a large genomic analysis of their adaptation to elevated temperatures [269]. All these assemblies have a size around 400 Mb, an overall BUSCO completeness over 88% with few duplicated BUSCO features, and several have an N50 over 1 Mb. These assemblies, although obtained with short reads only, do not have the same flaws as the initial assembly of *Astrangia poculata*, suggesting that short reads are still relevant to yield high-quality draft assemblies when combined with efficient assemblers (Platanus, in this case). Besides, the contiguity and completeness of the short-read only assemblies are comparable or higher compared to assemblies that included long reads. Hi-C scaffolding could be highly beneficial for the study of these genomes, as species of the genus *Acropora* have an endosymbiosis with zooxanthellae and Hi-C scaffolding can tell apart these different genomes.

The quality of the assembly of *Astrangia poculata* makes it a new reliable reference among anthozoan genomes for downstream analysis and comparison with other species.

Table 6.3: Comparison of assembly statistics with other genomes of the class Anthozoa.

Subclass	Order	Species	Reads technology	Assembly size	N50	BUSCO			
						single	dup.		
Hexacorallia	Scleractinia	<i>Astrangia poculata</i>	Illumina, Nanopore, Hi-C	455 Mb	31 Mb	89.2%	1.2%		
		<i>Acropora acuminata</i> [269]	Illumina, mate pair	395 Mb	1.0 Mb	93.3%	0.7%		
		<i>Acropora aui</i> [269]	Illumina, mate pair	429 Mb	1.1 Mb	89.0%	0.3%		
		<i>Acropora cytherea</i> [269]	Illumina, mate pair	426 Mb	1.1 Mb	88.6%	2.9%		
		<i>Acropora digitifera</i> [270]	454, Illumina, mate pair	447 Mb	484 kb	67.7%	5.0%		
		<i>Acropora digitifera</i> [269]	Illumina, PacBio	416 Mb	1.9 Mb	91.6%	0.6%		
		<i>Acropora echinata</i> [269]	Illumina, mate pair	401 Mb	1.9 Mb	88.2%	0.3%		
		<i>Acropora florida</i> [269]	Illumina, mate pair	443 Mb	751 kb	89.1%	1.7%		
		<i>Acropora gemmifera</i> [269]	Illumina, mate pair	401 Mb	1.1 Mb	87.3%	0.7%		
		<i>Acropora hyacinthus</i> [269]	Illumina, mate pair	447 Mb	1.6 Mb	91.4%	1.6%		
		<i>Acropora intermedia</i> [269]	Illumina, mate pair	417 Mb	577 kb	90.6%	1.8%		
		<i>Acropora microphthalmia</i> [269]	Illumina, mate pair	384 Mb	1.1 Mb	88.6%	1.4%		
		<i>Acropora millepora</i> [271]	Illumina, mate pair	387 Mb	495 kb	92.3%	0.7%		
		<i>Acropora millepora</i> [272]	Illumina, mate pair, Hi-C	387 Mb	22.6 Mb	91.7%	0.7%		
		<i>Acropora millepora</i> [206]	PacBio, linked reads, genetic map	475 Mb	19.8 Mb	91.9%	1.5%		
		<i>Acropora muricata</i> [269]	Illumina, mate pair	421 Mb	575 kb	87.4%	1.7%		
		<i>Acropora nasuta</i> [269]	Illumina, mate pair	416 Mb	1.1 Mb	89.4%	2.5%		
		<i>Acropora selago</i> [269]	Illumina, mate pair	393 Mb	657 kb	87.8%	1.3%		
		<i>Acropora tenuis</i> [269]	Illumina, mate pair	403 Mb	1.2 Mb	91.6%	0.8%		
		<i>Acropora yongei</i> [269]	Illumina, mate pair	438 Mb	3.0 Mb	89.9%	1.2%		
		<i>Montipora cactus</i> [269]	Illumina, mate pair	653 Mb	899 kb	89.4%	0.9%		
		<i>Montipora capitata</i> [273]	PacBio	886 Mb	541 kb	75.7%	16.9%		
		<i>Montipora capitata</i> [274]	Linked reads	615 Mb	186 kb	79.7%	0.5%		
		<i>Montipora efflorescens</i> [269]	Illumina, mate pair	643 Mb	1.1 Mb	88.4%	0.9%		
		<i>Orbicella faveolata</i> [275]	Illumina, mate pair	486 Mb	1.6 Mb	82.7%	2.3%		
		<i>Pocillopora damicornis</i> [276]	Illumina, CHICAGO	234 Mb	326 kb	88.5%	0.4%		
		<i>Stylophora pistillata</i> [277]	Illumina, mate pair	400 Mb	457 kb	87.6%	0.5%		
		Actiniaria		<i>Actinia equina</i> [278]	PacBio	409 Mb	493 kb	65.1%	29.5%
				<i>Actinia tenebrosa</i> [279]	Illumina, mate pair	238 Mb	189 kb	91.4%	0.6%
				<i>Exaiptasia pallida</i> [280]	Illumina, mate pair	256 Mb	442 kb	84.0%	2.6%
				<i>Nematostella vectensis</i> [281]	Sanger	357 Mb	473 kb	91.7%	1.8%
		Corallimorpharia		<i>Amplexidiscus fenestrafer</i> [282]	Illumina, mate pair	370 Mb	510 kb	84.4%	0.5%
<i>Discosoma</i> sp. [282]	Illumina, mate pair			444 Mb	772 kb	85.2%	2.1%		
Octocorallia	Alcyonacea	<i>Dendronephthya gigantea</i> [283]	Illumina, PacBio	286 Mb	1.4 Mb	84.3%	8.3%		
		<i>Paramuricea clavata</i> [284]	Illumina, Nanopore	607 Mb	24 kb	72.5%	1.3%		
		<i>Xenia</i> sp. [223]	Illumina, Nanopore, Hi-C	223 Mb	14.8 Mb	85.1%	2.1%		
	Pennatulacea	<i>Renilla muelleri</i> [285]	Illumina, PacBio	172 Mb	71 kb	85.2%	3.1%		

Chapter 7

Genome assembly of a chaetognath

7.1 Introduction

Chaetognaths, commonly known as arrow worms, are transparent marine predators widely distributed at various depths in all oceans, though low depth remains the favorite habitat of most species [286]. They are characterized by a transparent and elongated body, with one or two pairs of lateral fins, a caudal fin, a head with hooks, and range in size from a few millimeters to several centimeters [287]. Current chaetognath species are divided into two orders depending on the presence of transversal muscles, or phragms: Phragmophora and Aphragmophora [288]. The whole phylum now encompasses about 150 species. They form an enigmatic clade whose phylogenetic position is still discussed. Chaetognaths were initially considered as deuterostomians, due to their development, but analyses of 18S rDNA rejected this hypothesis [289, 290] and further brought support to the Phragmophora and Aphragmophora branches [291]. Later, Nielsen suggested that chaetognaths belonged to the clade Gnathifera [292], which gathers Gnathostomulida, Micrognathozoa and Rotifera. This hypothesis was supported by a recent transcriptome analysis of ten chaetognath species [293].

To this day, there is no genome assembly available for the whole phylum Chaetognatha, despite the fact that such a resource could help resolve and refine their phylogenetic position. Within the framework of the IGNITE consortium and my PhD project, I therefore tackled the assembly of a chaetognath genome, provisionally identified as *Flaccisagitta enflata* (see below). This species was first described by Grassi in 1881; it belongs to the order Aphragmophora and the family Sagittidae. It is epipelagic and present

across all oceans in warmer waters [294]. This species was selected based on specimen sizes (up to 2.5 cm), to avoid pooling individuals for sequencing and running into more haplotyping complexity, and its moderate genome size, estimated at 0.71 pg [295].

7.2 Material & Method

7.2.1 Collection and fixation

Chaetognaths were collected by Mark Vermeij around the island of Curaçao after sunset from October 29th to November 2nd 2019. A total of 30 individuals were sampled: eleven were crosslinked in 3% formaldehyde for 30 to 45 minutes, quenched in 250 mM glycine, then frozen at -80°C ; twelve were preserved in absolute ethanol and kept at 4°C ; seven were preserved in RNAlater and kept at 4°C . All samples used for DNA, RNA and Hi-C sequencing were collected on November 2nd 2019 in Snake Bay.

7.2.2 High-molecular-weight DNA extraction

One 2-cm individual preserved in ethanol was incubated in 180 μL of CTAB buffer (described in Table 7.1) and 25 μL of proteinase K for 3 hours at 60°C and 300 rpm. The lysed sample was purified with phenol-chloroform-isoamyl alcohol 25:24:1, chloroform-isoamyl alcohol 24:1 and with AMPure XP beads. I obtained 2.5 μg of DNA with $\text{OD}_{260/280} = 1.95$ and $\text{OD}_{260/230} = 1.95$.

Table 7.1: Composition of the cetyltrimethylammonium bromide (CTAB) buffer.

Solution	Stock concentration	Volume for 2.45 mL
Polyvinylpyrrolidone (PVP)	10%	500 μL
Tris(hydroxymethyl)aminomethane-HCl	1 M	250 μL
Ethylenediaminetetraacetic acid (EDTA)	500 mM	125 μL
NaCl	5 M	1 mL
H ₂ O	-	75 μL
CTAB	10%	500 μL
β -mercaptoethanol	-	25 μL

7.2.3 Whole-genome sequencing

The library for Nanopore sequencing was prepared with the Nanopore SQK-LSK109 Ligation sequencing kit. This library was loaded four times in the PromethION flow cell with nuclease flushes in between. The flow cell (with pore proteins R9.4.1) ran for 72 hours and gave an output of 40.5 Gb with an N50 = 6.1 kb. Basecalling was done with Guppy v4. 168 Gb of paired-end 150-bp Illumina reads were also sequenced by Novogene.

7.2.4 Hi-C sequencing

One 1.5-cm individual crosslinked in 3% formaldehyde was used to prepare a Hi-C library with the Arima Hi-C kit (including the restriction enzymes DpnII and HinfI), and resulted in 466 ng of DNA. The DNA was fragmented with a Covaris (300 bp) and biotinylated fragments were selected with streptavidin beads. The library was prepared for Illumina sequencing using Invitrogen TM Colibri TM PS DNA Library Prep Kit and following manufacturer instructions. Sequencing by Novogene resulted in 489 millions pairs of reads of 150 bp.

7.2.5 Pre-assembly analysis

The genome size was estimated with BBtools [236] using the script `kmercountexact.sh` and the shotgun Illumina reads. As this tool is based on k -mers, three values of k were tested with $k = \{27, 29, 31\}$. A k -mer histogram of the Illumina dataset was built using KAT hist v2.4.2 (with $k = 27$).

7.2.6 Genome assembly

The Nanopore reads were trimmed using Porechop [267] with default parameters, then they were assembled *de novo* with several assemblers: Canu [95], Flye [97], Ra [103], Raven [104] and wtdbg2 [108]. The assemblies were polished with HyPo [138] and remaining uncollapsed haplotypes were purged with `purge_dups` [145] and `Purge Haplotigs` [146]. I also tried preprocessing the reads by filtering them using a length threshold or the tool `Filtlong` with the parameters `--keep_percent 52.0 --min_length 2000`. Assembly pipelines were defined following the strategies identified in Chapter 2. Two assemblies were selected as candidates for Hi-C scaffolding:

- FEv1: Ratatosk + Canu + `purge_dups` x2
- FEv2: Raven + HyPo + `purge_dups` + `Purge Haplotigs`

7.2.7 Hi-C scaffolding

Hi-C reads were mapped to the draft assemblies using hicstuff [260] with the parameters `--enzyme DpnII,HinfI --aligner bowtie2 --iterative`. instaGRAAL was run with the parameters `--level 5 --cycles 150`. The scaffolds were post-processed with instaGRAAL-polish to reduce misassemblies and 10 Ns were added as gaps.

7.2.8 Gap filling

Gaps in the scaffolded assemblies were filled by TGS-GapCloser [197] using the Ratatosk-corrected Nanopore reads for FEv1 and the Nanopore reads for FEv2. FEv1 was further polished using HyPo.

7.2.9 Assembly evaluation

Assemblies were assessed using BUSCO v4 against the lineage metazoa odb10 without the parameter `--long` and using KAT comp 2.4.2 against the Illumina dataset (with $k = 27$). Contact maps were built for scaffolded assemblies using the hicstuff pipeline as described previously and hicstuff view with the parameter `--binning 2000`.

7.3 Results

7.3.1 Species identification

Morphological characteristics were registered in living and fixed individuals in order to identify the species:

- length up to 2.5 cm;
- body transparent, soft and inflated-looking;
- 8-10 hooks (Figure 1.A1);
- no collarette;
- anterior position of the ganglion (Figure 1.A2);
- pair of lateral fins;
- short rounded fins;
- ovaries not extending till the anterior fins (Figure 1.B3);

– round vesicles close to the tail (Figure 1.B4).

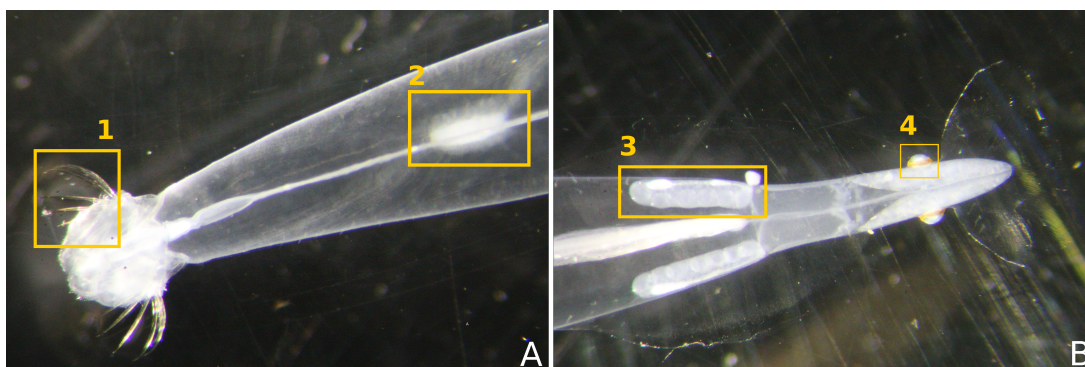


Figure 7.1: A chaetognath specimen fixed in 3% formaldehyde. This individual was used for Hi-C sequencing.

The specimens were all provisionally identified as *Flaccisagitta enflata* using the identification key provided in Michel [294].

7.3.2 Whole-genome assembly

The haploid genome size was estimated to 695-699 Mb, which matches the estimation available on the Animal Genome Size Database [295] of 0.71 picograms (approximately 694 Mb), measured with Feulgen densitometry. The k -mer histogram shows two distinct peaks, one around 85X, for heterozygous k -mers, and a second around 172X, for homozygous k -mers, thus the genome is diploid (Figure 7.2). The homozygous peak is strikingly small compared to the heterozygous peak, indicating a high level of heterozygosity.

Most initial assemblies had a size larger than the expected genome size (Table 7.2), due to the high heterozygosity of the genome that leads to artefactual duplications, as described in Chapter 2. Canu and Flye tend to retain uncollapsed haplotypes; Flye yielded an assembly about twice the expected haploid size when using all raw Nanopore reads (1.45 Gb), suggesting a diploid assembly, but surprisingly Canu produced a smaller assembly than expected (614 Mb). These behaviors were reversed with the Ratatosk-corrected Nanopore dataset: the Canu assembly was likely diploid (1.48 Gb), while the Flye assembly was close to the haploid genome size (849 Mb). However, the poor BUSCO score of the Ratatosk-Flye assembly suggested that it was not a good candidate. Two rounds of `purge_dups` greatly improved the Canu-Ratatosk and Flye-HyPo assemblies, as the number of duplicated BUSCO features were reduced in favor of single-copy BUSCO features. The Ratatosk-Canu-`purge_dups` assembly (designated as FEv1) was selected for scaffolding based on its high BUSCO score, low duplicated features and contiguity. The Ra assembly of raw Nanopore reads longer than 5 kb was the closest to the expected genome

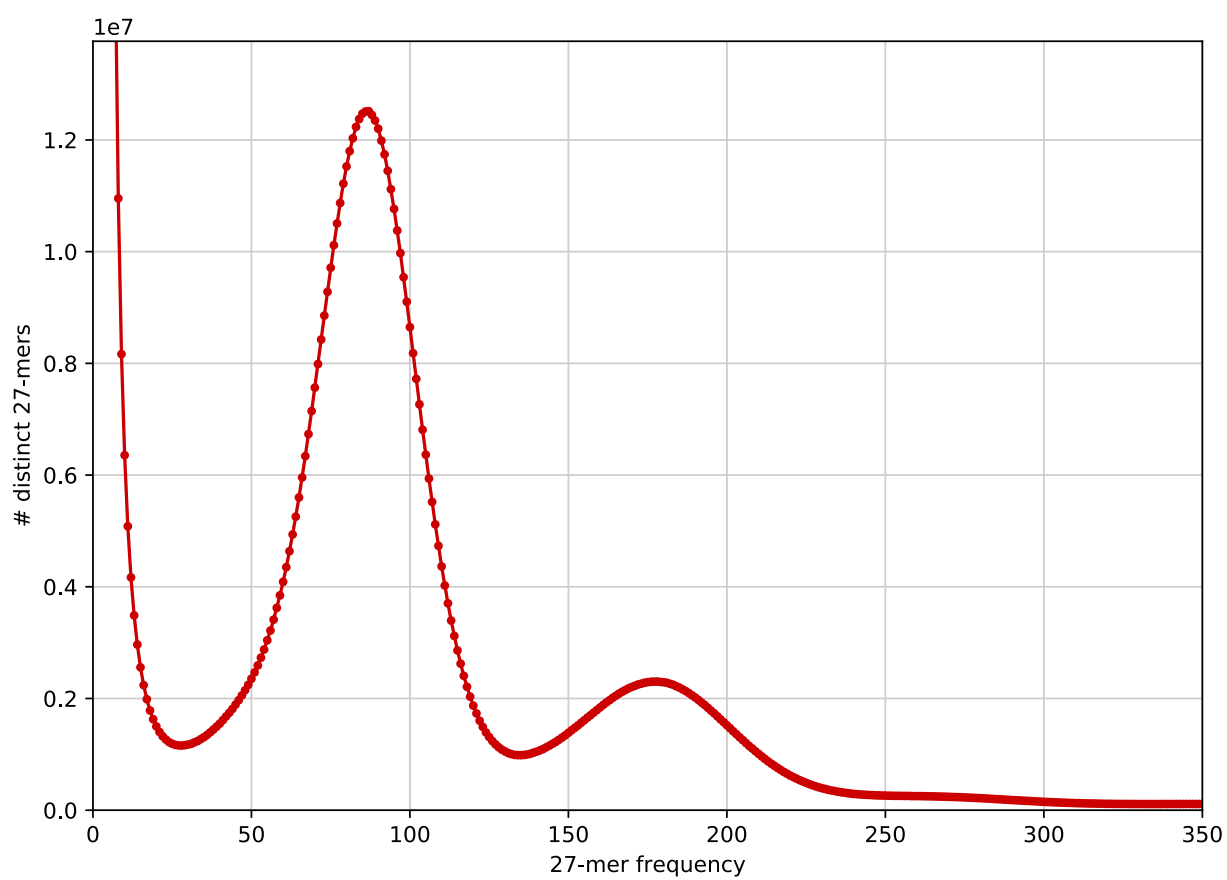
Figure 7.2: k -mer analysis of the Illumina dataset.

Table 7.2: Assembly statistics. purge_dups and Purge Haplotigs are respectively designated as PD and PH.

Assembly	Read selection	Hybrid correction	Polishing	Haplotig purging	Assembly size	# contigs	N50	Largest contig	BUSCO	
									single	dup.
Canu	5 kb	×	×	×	614 Mb	24038	44 kb	497 kb	27.6%	2.7%
	×	Ratatosk	×	×	1.48 Gb	13385	195 kb	1.2 Mb	28.3%	66.0%
	×	Ratatosk	×	PD x2	946 Mb	9288	215 kb	1.2 Mb	85.4%	7.1%
Flye	×	×	×	×	1.45 Gb	31205	264 kb	1.6 Mb	61.1%	12.3%
	×	Ratatosk	×	×	849 Mb	40273	50 kb	967 kb	43.9%	47.1%
	×	×	HyPo	×	1.45 Gb	31205	263 kb	1.6 Mb	35.0%	58.8%
	×	×	HyPo	PD x2	985 Mb	9508	277 kb	1.6 Mb	71.2%	19.8%
Ra	5 kb	×	×	×	728 Mb	7706	110 kb	509 kb	42.9%	1.3%
	5 kb	×	HyPo	×	730 Mb	7706	111 kb	510 kb	68.8%	11.6%
	Filtlong	×	×	×	684 Mb	7507	106 kb	498 kb	39.7%	0.6%
Raven	×	×	×	×	1.10 Gb	7656	185 kb	1.2 Mb	60.0%	6.0%
	5 kb	×	×	×	1.01 Gb	6901	180 kb	864 kb	37.8%	0.9%
	Filtlong	×	×	×	1.01 Gb	6741	186 kb	1.0 Mb	36.3%	1.3%
	×	×	HyPo	×	1.10 Gb	7656	186 kb	1.2 Mb	61.1%	30.9%
	×	×	HyPo	PD + PH	929 Mb	6612	191 kb	1.2 Mb	78.9%	11.6%
wtdbg2	×	×	×	×	965 Mb	13558	219 kb	2.0 Mb	19.3%	0.2%
	×	×	HyPo	×	987 Mb	13558	224 kb	2.1 Mb	71.5%	13.1%

size (728 Mb), yet the BUSCO score after polishing is low (80.4% single-copy and duplicated features). The Raven assembly of all raw reads was oversized, and after polishing the BUSCO score pointed at a large amount of duplications (30.9% duplicated features). A combination of purge_dups and Purge Haplotigs diminished the number of duplicated features (11.6%) and increased the single-copy BUSCO score (78.9%); this assembly was selected for scaffolding as FEv2. The wtdbg2 assembly had a moderate amount of duplications, but its BUSCO completeness after polishing was still lower than FEv1 and FEv2.

The mapping rate of Hi-C reads was low for both FEv1 and FEv2: only 37% of the reads aligned unambiguously. Scaffolding with instaGRAAL yielded 9 chromosome-level scaffolds (Table 7.3, Figure 7.3) that were retained for the final assemblies, to discard contamination from bacteria and plankton in the digestive tube. The cumulative sizes of the 9 scaffolds are close to the expected genome size, yet still slightly higher (794 Mb for FEv1, 745 Mb for FEv2). The scaffolds are notably larger in FEv1 than in FEv2, but the contact maps are similar. FEv1 has the lowest number of Ns in gaps, and its BUSCO score is higher as well. As for the k -mer spectra (Figure 7.4), both FEv1 and FEv2 have some remaining duplicated k -mers in the homozygous peak. Nevertheless, most homozygous k -mers are represented once, part of heterozygous k -mers are represented once, and the rest are not included in the assembly, as is expected for a collapsed assembly of a diploid genome. There is yet fewer missing homozygous k -mers in FEv1 than in FEv2.

Table 7.3: Comparison of scaffolded assemblies.

Assembly	Assembly size	Chromosomes lengths	N count	BUSCO		
				overall	single	dup.
FEv1	794 Mb	71-112 Mb	23,555	93.3%	86.9%	6.4%
FEv2	745 Mb	59-105 Mb	84,572	87.6%	83.3%	4.3%

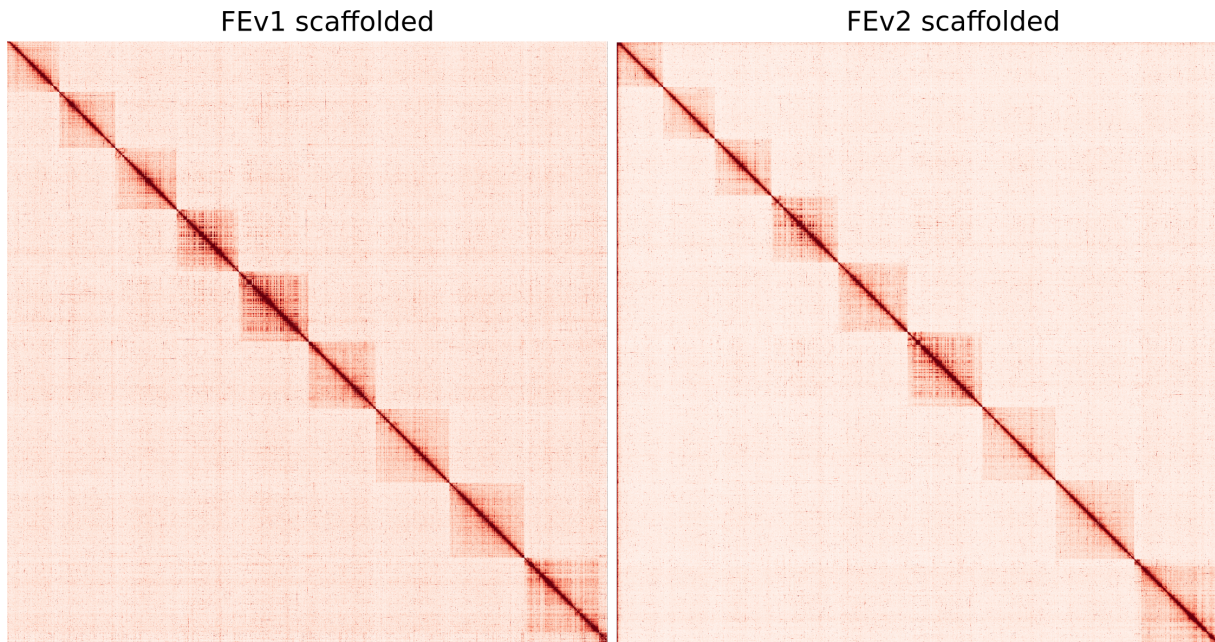


Figure 7.3: Contact maps of chromosome-level scaffolds for the two Hi-C scaffolded assemblies.

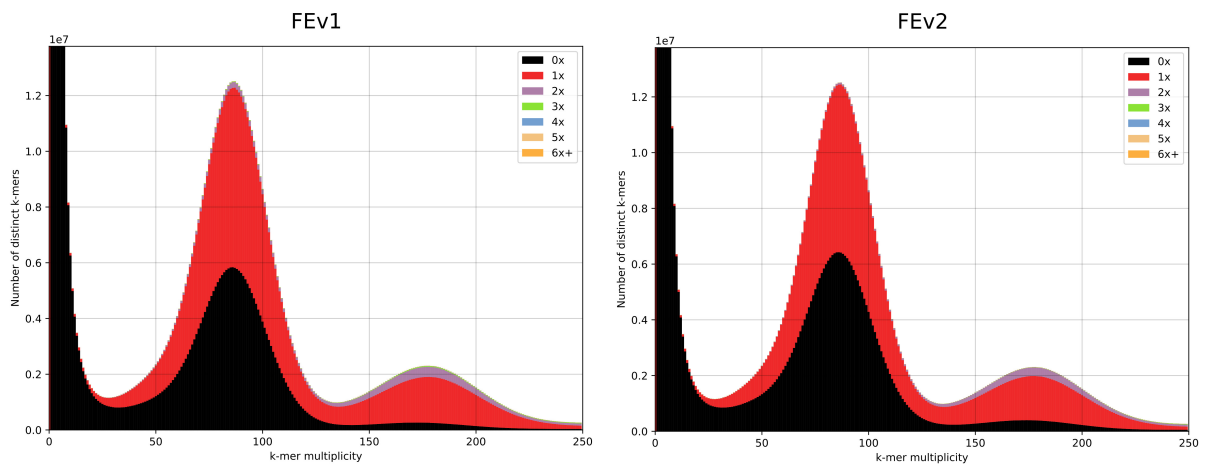


Figure 7.4: *k*-mer analysis of chromosome-level scaffolds for the two Hi-C scaffolded assemblies.

7.4 Discussion

The genome of *Flaccisagitta enflata* posed a challenge on several aspects. The amount of DNA that could be extracted from one specimen was too low to run all analysis on a single individual, but it was however sufficient to avoid pooling individuals for long-read sequencing or for Hi-C sequencing. First analysis showed that it was even more desirable to assemble the contigs from only one individual as the genome has a high heterozygosity; thus, a pool of highly divergent individuals would have further hindered the assembly. Using a second individual for Hi-C sequencing is a likely cause for the low mapping rate, due to the variability from one specimen to another. Besides, the collapsed assembly only represents one haplotype for all heterozygous regions; as this genome has a high heterozygosity, many Hi-C reads may fail to map to the missing haplotypes. Considering the level of heterozygosity, a diploid assembly would have been a more complete representation. Such assembly is possible when combining short and long reads, as for the Ratatosk-Canu assembly, but phasing with Hi-C is impossible in this case as the datasets were generated from different specimens.

Despite these difficulties, the FEv1 and FEv2 assemblies reached high contiguity and completeness. The draft long-read assemblies had a relatively small N50, which may be to the numerous heterozygous regions causing unresolved bubbles in the assembly graphs and leading to breaks. Still, Hi-C scaffolding with instaGRAAL brought these assemblies to chromosome level; although no karyotype is available for this species, FEv1 and FEv2 both converged towards 9 chromosome-level scaffolds. The assembly is currently under annotation, and will subsequently be analyzed to bring insights into the phylogeny of chaetognaths.

This study will serve as a basis for chaetognath genomics, since this is the first chaetognath assembly. Future phylogenomic analyses may shed light on the puzzling position of this clade, and bring insights into its evolution. In addition, the methods used for this project should facilitate new chaetognaths assemblies as it provides resources for high-molecular-weight DNA extraction, Nanopore sequencing, Hi-C sequencing, and assembly.

Chapter 8

Discussion & Conclusion

8.1 Genome assemblies of non-vertebrate animals

Genomic resources are constantly growing, however, animal genome projects have been biased towards vertebrates. The wide diversity of non-vertebrate animals brings equal possibilities and difficulties, as protocols and assembly strategies need to be adapted for each project. The genome assemblies presented in this thesis contribute to filling the gap in genomic resources for several clades of non-vertebrate animals. The new genome assembly of *Adineta vaga* is the first chromosome-level assembly of a rotifer species, and is additionally a first instance of Nanopore and Hi-C sequencing for this group; the genome of *Adineta ricciae* [296] was assembled with PacBio CLR but remained heavily fragmented. As for the chaetognath *Flaccisagitta enflata*, there is not yet any nuclear genome assembly available for the whole phylum. Coral genomes were published recently, which included long reads and Hi-C (the hard coral *Acropora millepora* [206, 272] and the soft coral *Xenia* sp. [223]), and the genome of *Astrangia poculata* will further enrich coral genomics. As these assemblies have reached chromosome-level scaffolds using Hi-C data, the main scaffolds should be devoid of contaminations, making them robust references for downstream analysis. Annotated genome assemblies represent complete gene sets which can be compared between species to identify orthologs and specific genes. The coral *Astrangia poculata* may have genes that exempt it from symbiosis and make it adaptable to a wide range of temperatures, unlike other corals from the genus *Acropora*. These chromosome-level assemblies also enable structural analyses of these genomes. The rotifer *Adineta vaga* was already identified as a paleotetraploid, and the genome of *Flaccisagitta enflata* may have similar features, as a prior of study of the transcriptome of the chaetognath *Spadella cephaloptera* suspected a whole-genome duplication event [297].

8.2 Combining long reads and Hi-C for chromosome-level assemblies

Long reads and Hi-C technologies became in recent years the winning combination, with short-read sequencing, to reach chromosome-level assemblies with high completeness. Long-read sequencing is still more laborious than short reads, due to its requirement for high-molecular-weight DNA, yet long-read assemblies are generally favored for their higher contiguity and better resolution of repeats. However, long reads are often not sufficient to assemble eukaryote genomes into chromosome-level contigs, and a scaffolding step often remains necessary. Increasing the sequencing depth may improve the contiguity to a certain extent, but long-read assemblers do not seem able to take advantage of huge sequencing depths (up to 230X of PacBio CLR and 170X of Nanopore reads in the case of *Adineta vaga*) to fully solve assemblies. Scaffolding is therefore needed, and Hi-C has emerged as the most robust method to bring assemblies to chromosome level. The popularity of Hi-C has stimulated the release of protocols, commercial kits and programs, providing researchers with a variety of options to adapt to their genome projects. The genomes presented here, *Adineta vaga*, *Astrangia poculata* and *Flaccisagitta enflata*, were assembled with a mix of short reads, long reads and Hi-C, and all reached chromosome-level scaffolds with high completeness. The genome assembly of *Flaccisagitta enflata* was the most challenging out of the three due to: its moderate size (694-699 Mb); its high heterozygosity; the low N50 of Nanopore reads; the poor Hi-C mapping rate. Nevertheless, the quality of the final assembly further demonstrates the robustness of the combination of long reads and Hi-C.

The amount of Hi-C data and the mapping rates are highly variable among projects (Table 8.1). The differences in mapping rates cannot be attributed to read length as the Hi-C reads for *Adineta vaga* are only 66 bp-long (against 150 bp for *Astrangia poculata* and *Flaccisagitta enflata*), but *Adineta vaga* has the highest mapping rate (83%). In addition, most Hi-C reads of *Adineta vaga* (72%) were mapped in the first round of iterative mapping, using only 20 bases. The low mapping rate for *Flaccisagitta enflata* may be attributed to the high heterozygosity of the genome and to the use of a different individual rather than the one used for Illumina and Nanopore sequencing. It is unclear what would be the necessary amount of reads for Hi-C scaffolding to obtain chromosome-level scaffolds. The company Arima Genomics recommends 200 millions pairs of Hi-C reads for a \sim 1-Gb genome. This raw estimation does not take into account the mapping rate nor the fragmentation of the genome, and a thorough review of Hi-C scaffolding

Species	# Hi-C pairs	Mapping rate
<i>Adineta vaga</i>	55 millions	83%
<i>Astrangia poculata</i>	723 millions	67%
<i>Flaccisagitta enflata</i>	489 millions	37%

Table 8.1: Overview of Hi-C datasets.

should consider these factors to find optimal Hi-C sequencing depths depending on the genome projects.

Furthermore, Hi-C reads are generated for scaffolding in genome projects, but they can also be used to explore the 3D architecture of the corresponding genome. As chromosome-level assemblies and Hi-C datasets are accumulating for a wide variety of species, these resources could be compiled into an evolutionary analysis based on the 3D genomes. For instance, the tool Chromosight was used to detect chromatin 3D structures in bacteria, yeasts, and 11 animals [298]. Furthermore, a recent study investigated the mechanisms underlying genome folding in 27 species of animals, fungi and plants [272]. This analysis targeted eukaryotes in general; it surveyed 20 animals, including 6 vertebrates, and disregarded several metazoan phyla. Recently published non-vertebrate genomes with Hi-C data, such as the sponge *Ephydatia muelleri* [232], the echinoderm *Lytechinus variegatus* [225], the nematode *Caenorhabditis remanei* [229], the slug *Arion vulgaris* [299], and the ones presented here, could be integrated in a large study of the 3D genomes of animals.

8.3 Defining a new benchmark dataset: *Adineta vaga*

New assembly tools are typically benchmarked against the genomes of bacteria or model organisms with a low heterozygosity, such as *Drosophila melanogaster*, *Caenorhabditis elegans*, *Homo sapiens*, and up to a heterozygosity of 1% for *Arabidopsis thaliana*. Testing new programs on the human genome is however often a requirement for publication (as was the case for GRAAL [184] and instaGRAAL [186]), as large sequencing datasets of all types are available and this genome is the closest to a perfect assembly, evermore since the release of a gap-less reference [300]. Therefore, these programs are often tuned for low-heterozygosity genomes and can only poorly handle higher levels of heterozygosity. The benchmark of long-read assemblers (Chapter 2) shed light on the limitations of these assemblers on a non-model genome, *Adineta vaga*, with a mixture of low-heterozygosity and high-heterozygosity regions. Long-read assemblers showed distinct behaviors on the same long-read datasets. wtdbg2 yields contigs with few duplications as it eliminates alternative haplotypes in the assembly graph by identifying and removing bubbles, i.e. regions where one homozygous sequence can be connected to several sequences, correspond-

ing to the different haplotypes for one heterozygous region. By contrast, Canu has a more conservative approach to separate repetitions and haplotypes, leading to uncollapsed assemblies. However, all these assemblers can produce high-quality haploid assemblies when combined with pre-assembly filtering or post-assembly haplotig purging.

The genome of *Adineta vaga* now has a reliable reference and large PacBio CLR, Nanopore, Illumina, and Hi-C datasets, making it a compelling example for benchmarks of new assembly tools on a mid-heterozygosity genome. These assembly strategies are not exclusive to *Adineta vaga*, and some were used for *Astrangia poculata* and *Flaccisagitta enflata*. Implementing this methodology into an assembly pipeline, including evaluation steps to identify well-collapsed candidate assemblies, would facilitate assembly projects.

8.4 Decreasing long-read sequencing depth

wtdbg2 emerged as the most cost-effective assembler, reaching a size close to the expected genome size with as little as a 10X long-read dataset, and with small variations with increasing sequencing depth. This result on *Adineta vaga* was further confirmed with the genome of *Astrangia poculata*: wtdbg2 yielded the best draft assembly using a 15X Nanopore dataset. Besides, wtdbg2 requires small computational resources compared to other assemblers. The capacity of wtdbg2 to accommodate low-depth long-read datasets demonstrates that the limiting factor is not sequencing depth but long-read assemblers. The cost of sequencing and assembly is crucial as it will determine the feasibility of a genome project, thus adapting assemblers to low sequencing depth (around 10 to 20X) would decrease the financial burden of genome assembly and increase accessibility for any research team.

8.5 Phasing assemblies

Since chromosome-level assemblies have become the target of sequencing projects, the challenge of genome assembly is now moving on to another step: phasing assemblies. This goal brings new difficulties: phasing is incompatible with pooling individuals (unless they are clones), and the necessary sequencing depth is multiplied by the ploidy of the genome at hand. GraphUnzip offers an approach for phasing genomes with long reads and Hi-C. It requires an uncollapsed assembly rather than trying to call variants from

a collapsed assembly, thus it is adequate for non-model genomes, with large heterozygous genomes and sometimes hemiploidy. However, GraphUnzip needs additional tests to evaluate the correct association of haplotypes from one heterozygous region to another. PacBio HiFi reads are opening new possibilities for phased assemblies thanks to their low error rate, hence we can expect full haplotype-resolved assemblies to become common in the next years.

8.6 Reproducibility in genome projects

As more and more genomes are being released, protocols are published in parallel which circumvent the challenges posed by a given species, and these methods can be applied for genome projects of similar species. Many genome projects have recourse to private companies, providing a service for high-molecular-weight DNA extraction, long-read sequencing, and Hi-C sequencing. As a result, the protocols are not publicly available; while having these chromosome-level assemblies is essential, they do not bring clues for new genome projects. This is the case for the genome of *Astrangia poculata*, as we performed Nanopore sequencing, but high-molecular-weight DNA extraction and Hi-C sequencing were done by Dovetails Genomics, and other stony coral projects cannot build upon our work. For *Adineta vaga* and *Flaccisagitta enflata* however, the methods are publicly available and can be reproduced. In addition, sequencing platforms use in-house tools; some are open source (e.g. the PacBio CLR assembler Falcon, the PacBio HiFi assembler IPA for Pacific Biosciences), whereas others are not (the Hi-C assembler HiRise, for Dovetails Genomics, only has a non-user-friendly and outdated version online). These programs may often result in high-quality assemblies, yet, depending on the genome, some open-source alternatives such as instaGRAAL may be more adequate.

Genome assemblies presented in Chapter 1 were surveyed on the National Center for Biotechnology Information (NCBI) database [2], which provides information on genome size, contig N50 and scaffold N50. Not all assemblies found in publications were available on this website, and in some cases, the assembly statistics in the paper did not match the ones on the database. For most projects, sequencing datasets were associated with a BioProject number, but these datasets were sometimes incomplete or missing. The absence of these sequencing datasets prevents reusability, reproducibility, or independent analyses. The metadata also lack information on the tools used for assembly; comprehensive records of assembly methods would help evaluate assembly programs.

Bibliography

- [1] Edward S Rice and Richard E Green. New approaches for genome assembly and scaffolding. *Annual review of animal biosciences*, 7:17–40, 2019.
- [2] National Center for Biotechnology Information. NCBI, <https://www.ncbi.nlm.nih.gov/>, 2021.
- [3] International Union for Conservation of Nature. Red List, www.iucnredlist.org/resources/summary-statistics, Accessed on May 4th, 2021.
- [4] S Blair Hedges and Sudhir Kumar. *The timetree of life*. OUP Oxford, 2009.
- [5] Zhi-Qiang Zhang. Animal biodiversity: An update of classification and diversity in 2013. In *Animal Biodiversity: An Outline of Higher-level Classification and Survey of Taxonomic Richness (Addenda 2013)*, volume 3703, pages 1–82. Magnolia Press, 2013.
- [6] Fei Li, Xianxin Zhao, Meizhen Li, Kang He, Cong Huang, Yuenan Zhou, Zekai Li, and James R Walters. Insect genomes: progress and challenges. *Insect Molecular Biology*, 28(6):739–758, 2019.
- [7] Jorge Ari Noriega, Joaquín Hortal, Francisco M Azcárate, Matty P Berg, Núria Bonada, Maria JI Briones, Israel Del Toro, Dave Goulson, Sébastien Ibanez, Douglas A Landis, et al. Research trends in ecosystem services provided by insects. *Basic and Applied Ecology*, 26:8–23, 2018.
- [8] Wenbo Chen, Daniel K Hasegawa, Navneet Kaur, Adi Kliot, Patricia Valle Pinheiro, Junbo Luan, Marcus C Stensmyr, Yi Zheng, Wenli Liu, Honghe Sun, et al. The draft genome of whitefly *bemisia tabaci* meam1, a global crop pest, provides novel insights into virus transmission, host adaptation, and insecticide resistance. *BMC Biology*, 14(1):1–15, 2016.
- [9] Vishvanath Nene, Jennifer R. Wortman, Daniel Lawson, Brian Haas, Chinnappa Kodira, Zhi-jian Tu, Brendan Loftus, Zhiyong Xi, Karyn Megy, Manfred Grabherr, Quinghu Ren, Evgeny M. Zdobnov, Neil F. Lobo, Kathryn S. Campbell, Susan E. Brown, Maria F. Bonaldo, Jingsong Zhu, Steven P. Sinkins, David G. Hogenkamp, Paolo Amedeo, Peter Arensburger, Peter W. Atkinson,

- Shelby Bidwell, Jim Biedler, Ewan Birney, Robert V. Bruggner, Javier Costas, Monique R. Coy, Jonathan Crabtree, Matt Crawford, Becky DeBruyn, David DeCaprio, Karin Eiglmeier, Eric Eisenstadt, Hamza El-Dorry, William M. Gelbart, Suely L. Gomes, Martin Hammond, Linda I. Hannick, James R. Hogan, Michael H. Holmes, David Jaffe, J. Spencer Johnston, Ryan C. Kennedy, Hean Koo, Saul Kravitz, Evgenia V. Kriventseva, David Kulp, Kurt LaButti, Eduardo Lee, Song Li, Diane D. Lovin, Chunhong Mao, Evan Mauceli, Carlos F.M. Menck, Jason R. Miller, Philip Montgomery, Akio Mori, Ana L. Nascimento, Horacio F. Naveira, Chad Nusbaum, Sinéad O'Leary, Joshua Orvis, Mihaela Pertea, Hadi Quesneville, Kyanne R. Reidenbach, Yu Hui Rogers, Charles W. Roth, Jennifer R. Schneider, Michael Schatz, Martin Shumway, Mario Stanke, Eric O. Stinson, Jose M.C. Tubio, Janice P. VanZee, Sergio Verjovski-Almeida, Doreen Werner, Owen White, Stefan Wyder, Qiandong Zeng, Qi Zhao, Yongmei Zhao, Catherine A. Hill, Alexander S. Raikhel, Marcelo B. Soares, Dennis L. Knudson, Norman H. Lee, James Galagan, Steven L. Salzberg, Ian T. Paulsen, George Dimopoulos, Frank H. Collins, Bruce Birren, Claire M. Fraser-Liggett, and David W. Severson. Genome sequence of *Aedes aegypti*, a major arbovirus vector. *Science*, 316(5832):1718–1723, 2007.
- [10] Osvaldo Marinotti, Gustavo C Cerqueira, Luiz Gonzaga Paula De Almeida, Maria Inês Tiraboschi Ferro, Elgion Lucio da Silva Loreto, Arnaldo Zaha, Santuza MR Teixeira, Adam R Wespiser, Alexandre Almeida e Silva, Aline Daiane Schlindwein, et al. The genome of anopheles darlingi, the main neotropical malaria vector. *Nucleic Acids Research*, 41(15):7387–7400, 2013.
- [11] Benjamin J. Matthews, Olga Dudchenko, Sarah B. Kingan, Sergey Koren, Igor Antoshechkin, Jacob E. Crawford, William J. Glassford, Margaret Herre, Seth N. Redmond, Noah H. Rose, Gareth D. Weedall, Yang Wu, Sanjit S. Batra, Carlos A. Brito-Sierra, Steven D. Buckingham, Corey L. Campbell, Saki Chan, Eric Cox, Benjamin R. Evans, Thanyalak Fansiri, Igor Filipović, Albin Fontaine, Andrea Gloria-Soria, Richard Hall, Vinita S. Joardar, Andrew K. Jones, Raissa G.G. Kay, Vamsi K. Kodali, Joyce Lee, Gareth J. Lycett, Sara N. Mitchell, Jill Muehling, Michael R. Murphy, Arina D. Omer, Frederick A. Partridge, Paul Peluso, Aviva Presser Aiden, Vidya Ramasamy, Gordana Rašić, Sourav Roy, Karla Saavedra-Rodriguez, Shruti Sharan, Atashi Sharma, Melissa Laird Smith, Joe Turner, Allison M. Weakley, Zhilei Zhao, Omar S. Akbari, William C. Black, Han Cao, Alistair C. Darby, Catherine A. Hill, J. Spencer Johnston, Terence D. Murphy, Alexander S. Raikhel, David B. Sattelle, Igor V. Sharakhov, Bradley J. White, Li Zhao, Erez Lieberman Aiden, Richard S. Mann, Louis Lambrechts, Jeffrey R. Powell, Maria V. Sharakhova, Zhijian Tu, Hugh M. Robertson, Carolyn S. McBride, Alex R. Hastie, Jonas Korlach, Daniel E. Neafsey, Adam M. Phillippy, and Leslie B. Vosshall. Improved reference genome of *Aedes aegypti* informs arbovirus vector control. *Nature*,

- 563(7732):501–507, 2018.
- [12] Olga Dudchenko, Sanjit S. Batra, Arina D. Omer, Sarah K. Nyquist, Marie Hoeger, Neva C. Durand, Muhammad S. Shamim, Ido Machol, Eric S. Lander, Aviva Presser Aiden, and Erez Lieberman Aiden. *De novo* assembly of the *Aedes aegypti* genome using Hi-C yields chromosome-length scaffolds. *Science*, 356(6333):92–95, 2017.
- [13] Anthony R Carroll, Brent R Copp, Rohan A Davis, Robert A Keyzers, and Michèle R Prinsep. Marine natural products. *Natural Product Reports*, 2021.
- [14] Shaden AM Khalifa, Nizar Elias, Mohamed A Farag, Lei Chen, Aamer Saeed, Mohamed-Elamir F Hegazy, Moustafa S Moustafa, Abd El-Wahed, Saleh M Al-Mousawi, Syed G Musharraf, et al. Marine natural products: A source of novel anticancer drugs. *Marine Drugs*, 17(9):491, 2019.
- [15] Tzi Bun Ng, Randy Chi Fai Cheung, Jack Ho Wong, Adnan A Bekhit, and Alaa El-Din Bekhit. Antibacterial products of marine organisms. *Applied Microbiology and Biotechnology*, 99(10):4145–4173, 2015.
- [16] Conxita Avila. Terpenoids in marine heterobranch molluscs. *Marine Drugs*, 18(3):162, 2020.
- [17] Bing-Nan Han, Li-Li Hong, Bin-Bin Gu, Yang-Ting Sun, Jie Wang, Jin-Tang Liu, and Hou-Wen Lin. Natural products from sponges. In *Symbiotic Microbiomes of Coral Reefs Sponges and Corals*, pages 329–463. Springer Netherlands, 2019.
- [18] Takeshi Takeuchi. Molluscan genomics: implications for biology and aquaculture. *Current Molecular Biology Reports*, 3(4):297–305, 2017.
- [19] Chelse M Prather, Shannon L Pelini, Angela Laws, Emily Rivest, Megan Woltz, Christopher P Bloch, Israel Del Toro, Chuan-Kai Ho, John Kominoski, TA Scott Newbold, et al. Invertebrates, ecosystem services and climate change. *Biological Reviews*, 88(2):327–348, 2013.
- [20] Andre Gomes-dos Santos, Manuel Lopes-Lima, L Filipe C Castro, and Elsa Froufe. Molluscan genomics: the road so far and the way forward. *Hydrobiologia*, 847(7):1705–1726, 2020.
- [21] Nicola Conci, Sergio Vargas, and Gert Wörheide. The Biology and Evolution of Calcite and Aragonite Mineralization in Octocorallia. *Frontiers in Ecology and Evolution*, 9:81, 2021.
- [22] Melody S Clark. Molecular mechanisms of biomineralization in marine invertebrates. *Journal of Experimental Biology*, 223(11), 2020.

- [23] Chuya Shinzato, Eiichi Shoguchi, Takeshi Kawashima, Mayuko Hamada, Kanako Hisata, Makiko Tanaka, Manabu Fujie, Mayuki Fujiwara, Ryo Koyanagi, Tetsuro Ikuta, et al. Using the *Acropora digitifera* genome to understand coral responses to environmental change. *Nature*, 476(7360):320–323, 2011.
- [24] Yafei Mao, Evan P Economo, and Noriyuki Satoh. The roles of introgression and climate change in the rise to dominance of acropora corals. *Current Biology*, 28(21):3373–3382, 2018.
- [25] Zachary L Fuller, Veronique JL Mocellin, Luke A Morris, Neal Cantin, Jihanne Shepherd, Luke Sarre, Julie Peng, Yi Liao, Joseph Pickrell, Peter Andolfatto, et al. Population genetics of the coral acropora millepora: Toward genomic prediction of bleaching. *Science*, 369(6501), 2020.
- [26] Chuya Shinzato, Konstantin Khalturin, Jun Inoue, Yuna Zayasu, Miyuki Kanda, Mayumi Kawamitsu, Yuki Yoshioka, Hiroshi Yamashita, Go Suzuki, and Noriyuki Satoh. Eighteen coral genomes reveal the evolutionary origin of acropora strategies to accommodate environmental changes. *Molecular Biology and Evolution*, 38(1):16–30, 2021.
- [27] Paschalia Kapli, Ziheng Yang, and Maximilian J Telford. Phylogenetic tree building in the genomic age. *Nature Reviews Genetics*, 21(7):428–444, 2020.
- [28] E Sally Chang, Moran Neuhof, Nimrod D Rubinstein, Arik Diamant, Hervé Philippe, Dorothée Huchon, and Paulyn Cartwright. Genomic insights into the evolutionary origin of Myxozoa within Cnidaria. *Proceedings of the National Academy of Sciences*, 112(48):14912–14917, 2015.
- [29] Michael Eitel, Warren R Francis, Frederique Varoquaux, Jean Daraspe, Hans-Jürgen Osigus, Stefan Krebs, Sergio Vargas, Helmut Blum, Gray A Williams, Bernd Schierwater, et al. Comparative genomics and the nature of placozoan species. *PLoS Biology*, 16(7):1–36, 2018.
- [30] Felipe A Simão, Robert M Waterhouse, Panagiotis Ioannidis, Evgenia V Kriventseva, and Evgeny M Zdobnov. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics*, 31(19):3210–3212, 2015.
- [31] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.
- [32] GIGA Community of Scientists. The Global Invertebrate Genomics Alliance (GIGA): developing community resources to study diverse invertebrate genomes. *Journal of Heredity*, 105(1):1–18, 2014.
- [33] Christian R. Voolstra, GIGA Community of Scientists (COS), Gert Wörheide, and Jose V. Lopez. Advancing genomics through the global invertebrate genomics alliance (GIGA). *Invertebrate Systematics*, 31(1):1, 2017.

- [34] Harris A Lewin, Gene E Robinson, W John Kress, William J Baker, Jonathan Coddington, Keith A Crandall, Richard Durbin, Scott V Edwards, Félix Forest, M Thomas P Gilbert, et al. Earth biogenome project: Sequencing life for the future of life. *Proceedings of the National Academy of Sciences*, 115(17):4325–4333, 2018.
- [35] Darwin Tree of Life. Darwin Tree of Life, <https://www.darwintreeoflife.org>, 2021.
- [36] Aquatic Symbiosis Genomics Project. Aquatic Symbiosis Genomics Project, www.sanger.ac.uk/collaboration/aquatic-symbiosis-genomics-project, 2021.
- [37] European Reference Genome Atlas. European Reference Genome Atlas, www.erga-biodiversity.eu, 2021.
- [38] Frederick Sanger, Steven Nicklen, and Alan R Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 1977.
- [39] André Goffeau, Bart G Barrell, Howard Bussey, Ronald W Davis, Bernard Dujon, Heinz Feldmann, Francis Galibert, Jörg D Hoheisel, Claude Jacq, Michael Johnston, et al. Life with 6000 genes. *Science*, 274(5287):546–567, 1996.
- [40] *C. elegans* Sequencing Consortium. Genome sequence of the nematode *C. elegans*: a platform for investigating biology. *Science*, 282(5396):2012–2018, 1998.
- [41] Bilal Wajid, Muhammad U. Sohail, Ali R. Ekti, and Erchin Serpedin. The A, C, G, and T of genome assembly. *BioMed Research International*, 2016:1–10, may 2016.
- [42] International Human Genome Sequencing Consortium and others. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.
- [43] Mihai Pop and Steven L Salzberg. Bioinformatics challenges of new sequencing technology. *Trends in Genetics*, 24(3):142–149, 2008.
- [44] Marcel Margulies, Michael Egholm, William E Altman, Said Attiya, Joel S Bader, Lisa A Bemben, Jan Berka, Michael S Braverman, Yi-Ju Chen, Zhoutao Chen, et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, 2005.
- [45] Jonathan M Rothberg, Wolfgang Hinz, Todd M Rearick, Jonathan Schultz, William Mileski, Mel Davey, John H Leamon, Kim Johnson, Mark J Milgrew, Matthew Edwards, et al. An integrated semiconductor device enabling non-optical genome sequencing. *Nature*, 475(7356):348–352, 2011.

- [46] Kevin Judd McKernan, Heather E. Peckham, Gina L. Costa, Stephen F. McLaughlin, Yutao Fu, Eric F. Tsung, Christopher R. Clouser, Cisyla Duncan, Jeffrey K. Ichikawa, Clarence C. Lee, Zheng Zhang, Swati S. Ranade, Eileen T. Dimalanta, Fiona C. Hyland, Tanya D. Sokolsky, Lei Zhang, Andrew Sheridan, Haoning Fu, Cynthia L. Hendrickson, Bin Li, Lev Kotler, Jeremy R. Stuart, Joel A. Malek, Jonathan M. Manning, Alena A. Antipova, Damon S. Perez, Michael P. Moore, Kathleen C. Hayashibara, Michael R. Lyons, Robert E. Beaudoin, Brittany E. Coleman, Michael W. Laptewicz, Adam E. Sannicandro, Michael D. Rhodes, Rajesh K. Gottimukkala, Shan Yang, Vineet Bafna, Ali Bashir, Andrew MacBride, Can Alkan, Jeffrey M. Kidd, Evan E. Eichler, Martin G. Reese, Francisco M. De La Vega, and Alan P. Blanchard. Sequence and structural variation in a human genome uncovered by short-read, massively parallel ligation sequencing using two-base encoding. *Genome Research*, 19(9):1527–1541, 2009.
- [47] Michael L Metzker. Sequencing technologies — the next generation. *Nature Reviews Genetics*, 11(1):31–46, 2010.
- [48] David R. Bentley, Shankar Balasubramanian, Harold P. Swerdlow, Geoffrey P. Smith, John Milton, Clive G. Brown, Kevin P. Hall, Dirk J. Evers, Colin L. Barnes, Helen R. Bignell, Jonathan M. Boutell, Jason Bryant, Richard J. Carter, R. Keira Cheetham, Anthony J. Cox, Darren J. Ellis, Michael R. Flatbush, Niall A. Gormley, Sean J. Humphray, Leslie J. Irving, Mirian S. Karbelashvili, Scott M. Kirk, Heng Li, Xiaohai Liu, Klaus S. Maisinger, Lisa J. Murray, Bojan Obradovic, Tobias Ost, Michael L. Parkinson, Mark R. Pratt, Isabelle M.J. Rasolonjatovo, Mark T. Reed, Roberto Rigatti, Chiara Rodighiero, Mark T. Ross, Andrea Sabot, Subramanian V. Sankar, Aylwyn Scally, Gary P. Schroth, Mark E. Smith, Vincent P. Smith, Anastassia Spiridou, Peta E. Torrance, Svilen S. Tzonev, Eric H. Vermaas, Klaudia Walter, Xiaolin Wu, Lu Zhang, Mohammed D. Alam, Carole Anastasi, Ify C. Aniebo, David M.D. Bailey, Iain R. Bancarz, Saibal Banerjee, Selena G. Barbour, Primo A. Baybayan, Vincent A. Benoit, Kevin F. Benson, Claire Bevis, Phillip J. Black, Asha Boodhun, Joe S. Brennan, John A. Bridgham, Rob C. Brown, Andrew A. Brown, Dale H. Buermann, Abass A. Bundu, James C. Burrows, Nigel P. Carter, Nestor Castillo, Maria Chiara E. Catenazzi, Simon Chang, R. Neil Cooley, Natasha R. Crake, Olubunmi O. Dada, Konstantinos D. Diakoumakos, Belen Dominguez-Fernandez, David J. Earnshaw, Ugonna C. Egbujor, David W. Elmore, Sergey S. Etchin, Mark R. Ewan, Milan Fedurco, Louise J. Fraser, Karin V. Fuentes Fajardo, W. Scott Furey, David George, Kimberley J. Gietzen, Colin P. Goddard, George S. Golda, Philip A. Granieri, David E. Green, David L. Gustafson, Nancy F. Hansen, Kevin Harnish, Christian D. Haudenschild, Narinder I. Heyer, Matthew M. Hims, Johnny T. Ho, Adrian M. Horgan, Katya Hoschler, Steve Hurwitz, Denis V. Ivanov, Maria Q. Johnson, Terena James, T. A. Huw

- Jones, Gyoung Dong Kang, Tzvetana H. Kerelska, Alan D. Kersey, Irina Khrebtukova, Alex P. Kindwall, Zoya Kingsbury, Paula I. Kokko-Gonzales, Anil Kumar, Marc A. Laurent, Cynthia T. Lawley, Sarah E. Lee, Xavier Lee, Arnold K. Liao, Jennifer A. Loch, Mitch Lok, Shujun Luo, Radhika M. Mammen, John W. Martin, Patrick G. McCauley, Paul McNitt, Parul Mehta, Keith W. Moon, Joe W. Mullens, Taksina Newington, Zemin Ning, Bee Ling Ng, Sonia M. Novo, Michael J. O'Neill, Mark A. Osborne, Andrew Osnowski, Omead Ostadan, Lambros L. Paraschos, Lea Pickering, Andrew C. Pike, Alger C. Pike, D. Chris Pinkard, Daniel P. Pliskin, Joe Podhasky, Victor J. Quijano, Come Raczy, Vicki H. Rae, Stephen R. Rawlings, Ana Chiva Rodriguez, Phyllida M. Roe, John Rogers, Maria C. Rogert Bacigalupo, Nikolai Romanov, Anthony Romieu, Rithy K. Roth, Natalie J. Rourke, Silke T. Ruediger, Eli Rusman, Raquel M. Sanches-Kuiper, Martin R. Schenker, Josefina M. Seoane, Richard J. Shaw, Mitch K. Shiver, Steven W. Short, Ning L. Sizto, Johannes P. Sluis, Melanie A. Smith, Jean Ernest Sohna Sohna, Eric J. Spence, Kim Stevens, Neil Sutton, Lukasz Szajkowski, Carolyn L. Tregidgo, Gerardo Turcatti, Stephanie Vandevondele, Yuli Verhovsky, Selene M. Virk, Suzanne Wakelin, Gregory C. Walcott, Jingwen Wang, Graham J. Worsley, Juying Yan, Ling Yau, Mike Zuerlein, Jane Rogers, James C. Mullikin, Matthew E. Hurles, Nick J. McCooke, John S. West, Frank L. Oaks, Peter L. Lundberg, David Klenerman, Richard Durbin, and Anthony J. Smith. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218):53–59, 2008.
- [49] Martin O. Pollard, Deepti Gurdasani, Alexander J. Mentzer, Tarryn Porter, and Manjinder S. Sandhu. Long reads: their purpose and place. *Human Molecular Genetics*, 27(R2):R234–R241, 2018.
- [50] John Eid, Adrian Fehr, Jeremy Gray, Khai Luong, John Lyle, Geoff Otto, Paul Peluso, David Rank, Primo Baybayan, Brad Bettman, Arkadiusz Bibillo, Keith Bjornson, Bidhan Chaudhuri, Frederick Christians, Ronald Cicero, Sonya Clark, Ravindra Dalal, Alex DeWinter, John Dixon, Mathieu Foquet, Alfred Gaertner, Paul Hardenbol, Cheryl Heiner, Kevin Hester, David Holden, Gregory Kearns, Xiangxu Kong, Ronald Kuse, Yves Lacroix, Steven Lin, Paul Lundquist, Congcong Ma, Patrick Marks, Mark Maxham, Devon Murphy, Insil Park, Thang Pham, Michael Phillips, Joy Roy, Robert Sebra, Gene Shen, Jon Sorenson, Austin Tomaney, Kevin Travers, Mark Trulson, John Vieceli, Jeffrey Wegener, Dawn Wu, Alicia Yang, Denis Zaccarin, Peter Zhao, Frank Zhong, Jonas Korlach, and Stephen Turner. Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–138, 2009.
- [51] Aaron M Wenger, Paul Peluso, William J Rowell, Pi-chuan Chang, Richard J Hall, Gregory T Concepcion, Jana Ebler, Arkarachai Fungtammasan, Alexey Kolesnikov, Nathan D Olson, Armin

- Töpfer, Michael Alonge, Medhat Mahmoud, Yufeng Qian, Chen-shan Chin, Adam M Phillippy, Michael C Schatz, Gene Myers, Mark A Depristo, Jue Ruan, Tobias Marschall, Fritz J Sedlazeck, Justin M Zook, Heng Li, Sergey Koren, Andrew Carroll, David R Rank, and Michael W Hunkapiller. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. *Nature Biotechnology*, 37(October), 2019.
- [52] David Deamer, Mark Akeson, and Daniel Branton. Three decades of Nanopore sequencing. *Nature Biotechnology*, 34(5):518–524, 2016.
- [53] Miten Jain, Hugh E. Olsen, Benedict Paten, and Mark Akeson. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. *Genome Biology*, 17(1):1–11, 2016.
- [54] Ryan R. Wick, Louise M. Judd, and Kathryn E. Holt. Performance of neural network basecalling tools for Oxford Nanopore sequencing. *Genome Biology*, 20(1):1–10, 2019.
- [55] Miten Jain, Sergey Koren, Karen H. Miga, Josh Quick, Arthur C. Rand, Thomas A. Sasani, John R. Tyson, Andrew D. Beggs, Alexander T. Dilthey, Ian T. Fiddes, Sunir Malla, Hannah Marriott, Tom Nieto, Justin O’Grady, Hugh E. Olsen, Brent S. Pedersen, Arang Rhie, Hollian Richardson, Aaron R. Quinlan, Terrance P. Snutch, Louise Tee, Benedict Paten, Adam M. Phillippy, Jared T. Simpson, Nicholas J. Loman, and Matthew Loose. Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nature Biotechnology*, 36(4):338–345, 2018.
- [56] Bonito, <https://github.com/nanoporetech/bonito>.
- [57] Pacharaporn Angthong, Tanaporn Uengwetwanit, Wirulda Pootakham, Kanchana Sittikankaew, Chutima Sonthirod, Duangjai Sangsrakru, Thippawan Yoocha, Intawat Nookaew, Thidathip Wongsurawat, Piroon Jenjaroenpun, et al. Optimization of high molecular weight DNA extraction methods in shrimp for a long-read sequencing platform. *PeerJ*, 8:e10340, 2020.
- [58] Serafim Batzoglou, David B Jaffe, Ken Stanley, Jonathan Butler, Sante Gnerre, Evan Mauceli, Bonnie Berger, Jill P Mesirov, and Eric S Lander. ARACHNE: a whole-genome shotgun assembler. *Genome Research*, 12(1):177–189, 2002.
- [59] Paul Havlak, Rui Chen, K James Durbin, Amy Egan, Yanru Ren, Xing-Zhi Song, George M Weinstock, and Richard A Gibbs. The Atlas genome assembly system. *Genome Research*, 14(4):721–732, 2004.
- [60] Xiaoqiu Huang and Anup Madan. CAP3: a DNA sequence assembly program. *Genome Research*, 9(9):868–877, 1999.

- [61] Gennady Denisov, Brian Walenz, Aaron L. Halpern, Jason Miller, Nelson Axelrod, Samuel Levy, and Granger Sutton. Consensus generation and variant detection by Celera Assembler. *Bioinformatics*, 24(8):1035–1040, 2008.
- [62] Pavel A Pevzner, Haixu Tang, and Michael S Waterman. An Eulerian path approach to DNA fragment assembly. *Proceedings of the National Academy of Sciences*, 98(17):9748–9753, 2001.
- [63] Samuel Aparicio, Jarrod Chapman, Elia Stupka, Nik Putnam, Jer-ming Chia, Paramvir Dehal, Alan Christoffels, Sam Rash, Shawn Hoon, Arian Smit, et al. Whole-genome shotgun assembly and analysis of the genome of *Fugu rubripes*. *Science*, 297(5585):1301–1310, 2002.
- [64] Daniel D Sommer, Arthur L Delcher, Steven L Salzberg, and Mihai Pop. Minimus: a fast, lightweight genome assembler. *BMC Bioinformatics*, 8(1):1–11, 2007.
- [65] Bastien Chevreux, Thomas Wetter, Sándor Suhai, et al. Genome sequence assembly using trace signals and additional sequence information. In *German conference on bioinformatics*, volume 99, pages 45–56. Citeseer, 1999.
- [66] Brent Ewing and Phil Green. Base-calling of automated sequencer traces using Phred. II. error probabilities. *Genome Research*, 8(3):186–194, 1998.
- [67] James C Mullikin and Zemin Ning. The Phusion assembler. *Genome Research*, 13(1):81–90, 2003.
- [68] Giuseppe Narzisi and Bud Mishra. Scoring-and-unfolding trimmed tree assembler: concepts, constructs and comparisons. *Bioinformatics*, 27(2):153–160, 11 2010.
- [69] Anthony R. Sutton, Granger and White, Owen and Adams, Mark D. and Kerlavage. TIGR Assembler: A new tool for assembling large shotgun projects. *Genome Science and Technology*, 1(1):9–19, 1995.
- [70] Jared T. Simpson, Kim Wong, Shaun D. Jackman, Jacqueline E. Schein, Steven J.M. Jones, and Inanç Birol. ABySS: A parallel assembler for short read sequence data. *Genome Research*, 19(6):1117–1123, 2009.
- [71] Shaun D Jackman, Benjamin P Vandervalk, Hamid Mohamadi, Justin Chu, Sarah Yeo, S Austin Hammond, Golnaz Jahesh, Hamza Khan, Lauren Coombe, Rene L Warren, and Inanc Birol. ABySS 2.0: resource-efficient assembly of large genomes using a Bloom filter effect of Bloom filter false positive rate. *Genome Research*, 27:768–777, 2017.

- [72] Jonathan Butler, Iain MacCallum, Michael Kleber, Ilya A Shlyakhter, Matthew K Belmonte, Eric S Lander, Chad Nusbaum, and David B Jaffe. ALLPATHS: *de novo* assembly of whole-genome shotgun microreads. *Genome Research*, 18(5):810–820, 2008.
- [73] Jason R. Miller, Arthur L. Delcher, Sergey Koren, Eli Venter, Brian P. Walenz, Anushka Brownley, Justin Johnson, Kelvin Li, Clark Mobarry, and Granger Sutton. Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics*, 24(24):2818–2824, 2008.
- [74] David Hernandez, Patrice François, Laurent Farinelli, Magne Østerås, and Jacques Schrenzel. *De novo* bacterial genome sequencing: millions of very short reads assembled on a desktop computer. *Genome Research*, 18(5):802–809, 2008.
- [75] Mark J. Chaisson and Pavel A. Pevzner. Short read fragment assembly of bacterial genomes. *Genome Research*, 18(2):324–330, 2008.
- [76] Thomas Conway, Jeremy Wazny, Andrew Bromage, Justin Zobel, and Bryan Beresford-Smith. Gossamer — a resource-efficient *de novo* assembler. *Bioinformatics*, 28(14):1937–1938, 2012.
- [77] Yu Peng, Henry C.M. Leung, S. M. Yiu, and Francis Y.L. Chin. IDBA - a practical iterative De Bruijn graph *de novo* assembler. *Research in Computational Molecular Biology*, 6044 LNBI:426–440, 2010.
- [78] Te-Chin Chu, Chen-Hua Lu, Tsunglin Liu, Greg C Lee, Wen-Hsiung Li, and Arthur Chun-Chieh Shih. Assembler for *de novo* assembly of large genomes. *Proceedings of the National Academy of Sciences*, 110(36):E3417–E3424, 2013.
- [79] Jarrod A Chapman, Isaac Ho, Sirisha Sunkara, Shujun Luo, Gary P Schroth, and Daniel S Rokhsar. Meraculous: *de novo* genome assembly with short paired-end reads. *PloS One*, 6(8):e23501, 2011.
- [80] Nicolas Dierckxsens, Patrick Mardulyn, and Guillaume Smits. NOVOPlasty: *de novo* assembly of organelle genomes from whole genome data. *Nucleic Acids Research*, 45(4):e18–e18, 2017.
- [81] Xiaoqiu Huang, Jianmin Wang, Srinivas Aluru, Shiaw-Pyng Yang, and LaDeana Hillier. PCAP: a whole-genome assembly program. *Genome Research*, 13(9):2164–2170, 2003.
- [82] Xiao Zhu, Henry CM Leung, Francis YL Chin, Siu Ming Yiu, Guangri Quan, Bo Liu, and Yadong Wang. PERGA: a paired-end read guided *de novo* assembler for extending contigs using SVM and look ahead approach. *PloS One*, 9(12):e114253, 2014.

- [83] Rei Kajitani, Kouta Toshimoto, Hideki Noguchi, Atsushi Toyoda, Yoshitoshi Ogura, Miki Okuno, Mitsuru Yabana, Masayuki Harada, Eiji Nagayasu, Haruhiko Maruyama, et al. Efficient *de novo* assembly of highly heterozygous genomes from whole-genome shotgun short reads. *Genome Research*, 24(8):1384–1395, 2014.
- [84] Douglas W. Bryant, Weng-Keen Wong, and Todd C. Mockler. QSRA – a quality-value guided *de novo* short read assembler. *BMC Bioinformatics*, 10(1):1–6, 2009.
- [85] Sébastien Boisvert, François Laviolette, and Jacques Corbeil. Ray: simultaneous assembly of reads from a mix of high-throughput sequencing technologies. *Journal of Computational Biology*, 17(11):1519–1533, 2010.
- [86] Giorgio Gonnella and Stefan Kurtz. Readjoinder: a fast and memory efficient string graph-based sequence assembler. *BMC Bioinformatics*, 13(1):1–19, 2012.
- [87] Jared T Simpson and Richard Durbin. Efficient *de novo* assembly of large genomes using compressed data structures. *Genome Research*, 22(3):549–556, 2012.
- [88] Ruiqiang Li, Hongmei Zhu, Jue Ruan, Wubin Qian, Xiaodong Fang, Zhongbin Shil, Yingrui Lil, Shengting Li, Gao Shan, Karsten Kristiansen, Songgang Li, Huanming Yang, Jian Wang, and Jun Wang. *De novo* assembly of human genomes with massively parallel short read sequencing. *Genome Research*, 20(2):265–272, 2010.
- [89] Ruibang Luo, Binghang Liu, Yinlong Xie, Zhenyu Li, Weihua Huang, Jianying Yuan, Guangzhu He, Yanxiang Chen, Qi Pan, Yunjie Liu, et al. SOAPdenovo2: an empirically improved memory-efficient short-read *de novo* assembler. *GigaScience*, 1(1):2047–217X, 2012.
- [90] Anton Bankevich, Sergey Nurk, Dmitry Antipov, Alexey A. Gurevich, Mikhail Dvorkin, Alexander S. Kulikov, Valery M. Lesin, Sergey I. Nikolenko, Son Pham, Andrey D. Prjibelski, Alexey V. Pyshkin, Alexander V. Sirotkin, Nikolay Vyahhi, Glenn Tesler, Max A. Alekseyev, and Pavel A. Pevzner. SPAdes: A new genome assembly algorithm and its applications to single-cell sequencing. *Journal of Computational Biology*, 19(5):455–477, 2012.
- [91] René L. Warren, Granger G. Sutton, Steven J.M. Jones, and Robert A. Holt. Assembling millions of short DNA sequences using SSAKE. *Bioinformatics*, 23(4):500–501, 2007.
- [92] Bertil Schmidt, Ranjan Sinha, Bryan Beresford-Smith, and Simon J Puglisi. A fast hybrid short read fragment assembly algorithm. *Bioinformatics*, 25(17):2279–2280, 2009.

- [93] William R. Jeck, Josephine A. Reinhardt, David A. Baltrus, Matthew T. Hickenbotham, Vincent Magrini, Elaine R. Mardis, Jeffery L. Dangl, and Corbin D. Jones. Extending assembly of short DNA sequences to handle error. *Bioinformatics*, 23(21):2942–2944, 2007.
- [94] Daniel R. Zerbino. Using the Velvet *de novo* assembler for short-read sequencing technologies. *Current Protocols in Bioinformatics*, Chapter 11:1–12, 2010.
- [95] Sergey Koren, Brian P. Walenz, Konstantin Berlin, Jason R. Miller, Nicholas H. Bergman, and Adam M. Phillippy. Canu: scalable and accurate long-read assembly via adaptive k -mer weighting and repeat separation. *Genome Research*, 25(2):1–11, 2017.
- [96] Chen-Shan Chin, Paul Peluso, Fritz J Sedlazeck, Maria Nattestad, Gregory T Concepcion, Alicia Clum, Christopher Dunn, Ronan O’Malley, Rosa Figueroa-Balderas, Abraham Morales-Cruz, et al. Phased diploid genome assembly with single-molecule real-time sequencing. *Nature Methods*, 13(12):1050–1054, 2016.
- [97] Mikhail Kolmogorov, Jeffrey Yuan, Yu Lin, and Pavel A. Pevzner. Assembly of long, error-prone reads using repeat graphs. *Nature Biotechnology*, 37(5):540–546, 2019.
- [98] Govinda M Kamath, Ilan Shomorony, Fei Xia, Thomas A Courtade, and N Tse David. HINGE: long-read assembly achieves optimal repeat resolution. *Genome Research*, 27(5):747–756, 2017.
- [99] Chuan Le Xiao, Ying Chen, Shang Qian Xie, Kai Ning Chen, Yan Wang, Yue Han, Feng Luo, and Zhi Xie. MECAT: fast mapping, error correction, and *de novo* assembly for single-molecule sequencing reads. *Nature Methods*, 14(11):1072–1074, 2017.
- [100] Heng Li. Minimap and miniasm: fast mapping and *de novo* assembly for noisy long sequences. *Bioinformatics*, 32(14):2103–2110, 2016.
- [101] Ying Chen, Fan Nie, Shang-Qian Xie, Ying-Feng Zheng, Thomas Bray, Qi Dai, Yao-Xin Wang, Jian-feng Xing, Zhi-Jian Huang, De-Peng Wang, et al. Fast and accurate assembly of nanopore reads via progressive error correction and adaptive read selection. *bioRxiv*, 2020.
- [102] NextOmics. NextDeNovo, <https://github.com/Nextomics/NextDenovo>, 2019.
- [103] Robert Vaser and Mile Šikić. Yet another *de novo* genome assembler. *International Symposium on Image and Signal Processing and Analysis, ISPA*, pages 147–151, 2019.
- [104] Robert Vaser and Mile Šikić. Time-and memory-efficient genome assembly with raven. *Nature Computational Science*, pages 1–5, 2021.

- [105] Kishwar Shafin, Trevor Pesout, Ryan Lorig-roach, Marina Haukness, Hugh E Olsen, Colleen Bosworth, Joel Armstrong, Kristof Tigyi, Nicholas Maurer, Sergey Koren, Fritz J Sedlazeck, Tobias Marschall, Simon Mayes, Vania Costa, Justin M Zook, Kelvin J Liu, Duncan Kilburn, Melanie Sorensen, Katy M Munson, Mitchell R Vollger, Jean Monlong, Erik Garrison, Evan E Eichler, Sofie Salama, David Haussler, Richard E Green, Mark Akeson, Adam Phillippy, Karen H Miga, Paolo Carnevali, Miten Jain, and Benedict Paten. Nanopore sequencing and the Shasta toolkit enable efficient *de novo* assembly of eleven human genomes. *Nature Biotechnology*, 38(9):1044–1053, 2020.
- [106] Hailin Liu, Shigang Wu, Alun Li, and Jue Ruan. SMARTdenovo: A *de novo* assembler using long noisy reads. *Preprints*, 2020.
- [107] Jue Ruan. wtdbg, <https://github.com/ruanjue/wtdbg>, 2016.
- [108] Jue Ruan and Heng Li. Fast and accurate long-read assembly with wtdbg2. *Nature Methods*, 17(2):155–158, 2020.
- [109] Sergey Nurk, Brian P Walenz, Arang Rhie, Mitchell R Vollger, Glennis A Logsdon, Robert Grothe, Karen H Miga, Evan E Eichler, Adam M Phillippy, and Sergey Koren. HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads. *bioRxiv*, 2020.
- [110] Haoyu Cheng, Gregory T Concepcion, Xiaowen Feng, Haowen Zhang, and Heng Li. Haplotype-resolved *de novo* assembly using phased assembly graphs with hifiasm. *Nature Methods*, pages 1–6, 2021.
- [111] Pacific Biosciences. IPA, <https://github.com/PacificBiosciences/pbbioconda>, 2018.
- [112] Barış Ekim, Bonnie Berger, and Rayan Chikhi. Minimizer-space de bruijn graphs. *bioRxiv*, 2021.
- [113] Chen-Shan Chin and Asif Khalak. Human genome assembly in 100 minutes. *bioRxiv*, 2019.
- [114] Jorma Tarhio and Esko Ukkonen. A greedy approximation algorithm for constructing shortest common superstrings. *Theoretical Computer Science*, 57(1):131–145, 1988.
- [115] Rodger Staden. A strategy of DNA sequencing employing computer programs. *Nucleic Acids Research*, 6(7):2601–2610, 1979.
- [116] J Craig Venter, Mark D Adams, Eugene W Myers, Peter W Li, Richard J Mural, Granger G Sutton, Hamilton O Smith, Mark Yandell, Cheryl A Evans, Robert A Holt, et al. The Sequence of the Human Genome. *Science*, 291(5507):1304–1351, 2001.
- [117] Nicolass Govert de Bruijn. A combinatorial problem. *Koninklijke Nederlandse Akademie v. Wetenschappen*, 49:758–764, 1946.

- [118] Camille Flye Sainte-Marie. 48. *L'Intermédiaire des Mathématiciens*, 1:107–110, 1894.
- [119] Phillip E.C. Compeau, Pavel A. Pevzner, and Glenn Tesler. How to apply de Bruijn graphs to genome assembly. *Nature Biotechnology*, 29(11):987–991, 2011.
- [120] Anton Bankevich, Andrey Bzikadze, Mikhail Kolmogorov, and Pavel A Pevzner. Assembling long accurate reads using de Bruijn graphs. *bioRxiv*, 2020.
- [121] Dayana Yahalomi, Stephen D Atkinson, Moran Neuhof, E Sally Chang, Hervé Philippe, Paulyn Cartwright, Jerri L Bartholomew, and Dorothée Huchon. A cnidarian parasite of salmon (Myxozoa: Henneguya) lacks a mitochondrial genome. *Proceedings of the National Academy of Sciences*, 117(10):5358–5363, 2020.
- [122] Matthias C. Vogg, Leonardo Beccari, Laura Iglesias Ollé, Christine Rampon, Sophie Vrız, Chrystelle Perruchoud, Yvan Wenger, and Brigitte Galliot. An evolutionarily-conserved Wnt3/ β -catenin/Sp5 feedback loop restricts head organizer activity in Hydra. *Nature Communications*, 10(1):1–15, 2019.
- [123] Ellen M. Leffler, Kevin Bullaughey, Daniel R. Matute, Wynn K. Meyer, Laure Ségurel, Aarti Venkat, Peter Andolfatto, and Molly Przeworski. Revisiting an old riddle: what determines genetic diversity levels within species? *PLoS Biology*, 10(9):e1001388, 2012.
- [124] Ryan R. Wick. Filtlong, <https://github.com/rrwick/Filtlong>, 2017.
- [125] Ehsan Haghshenas, Faraz Hach, S. Cenk Sahinalp, and Cedric Chauve. CoLoRMap: correcting long reads by mapping short reads. *Bioinformatics*, 32(17):i545–i551, 2016.
- [126] Can Firtina, Ziv Bar-Joseph, Can Alkan, and A Ercument Cicek. Hercules: a profile HMM-based hybrid error correction algorithm for long reads. *Nucleic Acids Research*, 46(21):e125–e125, 2018.
- [127] Gilles Miclotte, Mahdi Heydari, Piet Demeester, Stephane Rombauts, Yves Van de Peer, Pieter Audenaert, and Jan Fostier. Jabba: hybrid error correction for long sequencing reads. *Algorithms for Molecular Biology*, 11(1):1–12, 2016.
- [128] Leena Salmela and Eric Rivals. LoRDEC: Accurate and efficient long read error correction. *Bioinformatics*, 30(24):3506–3514, 2014.
- [129] Leena Salmela, Riku Walve, Eric Rivals, Esko Ukkonen, and Cenk Sahinalp. Accurate self-correction of errors in long reads using de Bruijn graphs. *Bioinformatics*, 33(6):799–806, 2017.
- [130] Thomas Hackl, Rainer Hedrich, Jörg Schultz Schultz, and Frank Förster. proovread: large-scale high-accuracy PacBio correction through iterative short read consensus. *Bioinformatics*, 30(21):3004–3011, 2014.

- [131] Pierre Morisse, Camille Marchet, Antoine Limasset, Thierry Lecroq, and Arnaud Lefebvre. Scalable long read self-correction and assembly polishing with multiple sequence alignment. *Scientific Reports*, 11(1):1–13, 2021.
- [132] German Tischler and Eugene W. Myers. Non hybrid long read consensus using local de Bruijn graph assembly. *bioRxiv*, 2017.
- [133] Ergude Bao, Fei Xie, Changjin Song, and Dandan Song. FLAS: fast and high-throughput algorithm for PacBio long-read self-correction. *Bioinformatics*, 35(20), 2019.
- [134] René L Warren, Lauren Coombe, Hamid Mohamadi, Jessica Zhang, Barry Jaquish, Nathalie Isabel, Steven JM Jones, Jean Bousquet, Joerg Bohlmann, and Inanç Birol. ntEdit: scalable genome sequence polishing. *Bioinformatics*, 35(21):4430–4432, 2019.
- [135] Bruce J Walker, Thomas Abeel, Terrance Shea, Margaret Priest, Amr Abouelliel, Sharadha Sakthikumar, Christina A Cuomo, Qiandong Zeng, Jennifer Wortman, Sarah K Young, et al. Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. *PloS one*, 9(11):e112963, 2014.
- [136] Aleksey V. Zimin and Steven L. Salzberg. The genome polishing tool POLCA makes fast and accurate corrections in genome assemblies. *bioRxiv*, 2019.
- [137] Can Firtina, Jeremie S Kim, Mohammed Alser, Damla Senol Cali, A Ercument Cicek, Can Alkan, and Onur Mutlu. Apollo: a sequencing-technology-independent, scalable, and accurate assembly polishing algorithm. *Bioinformatics*, 2020.
- [138] Ritu Kundu, Joshua Casey, and Wing-kin Sung. HyPo : super fast & accurate polisher for long read assemblies. *bioRxiv*, 2019.
- [139] Robert Vaser, Ivan Sović, Niranjan Nagarajan, and Mile Šikić. Fast and accurate *de novo* genome assembly from long uncorrected reads. *Genome Research*, 27(5):737–746, 2017.
- [140] PacificBiosciences. GenomicConsensus, <https://github.com/PacificBiosciences/GenomicConsensus>, 2014.
- [141] Oxford Nanopore Technologies. Medaka, <https://github.com/nanoporetech/medaka>, 2017.
- [142] Jiang Hu, Junpeng Fan, Zongyi Sun, and Shanlin Liu. NextPolish: a fast and efficient genome polishing tool for long-read assembly. *Bioinformatics*, 2019.
- [143] Jared Simpson. Nanopolish, <https://github.com/jts/nanopolish>, 2014.

- [144] Shengfeng Huang, Mingjing Kang, and Anlong Xu. HaploMerger2: rebuilding both haploid sub-assemblies from high-heterozygosity diploid genome assembly. *Bioinformatics*, 33(16):2577–2579, 2017.
- [145] Dengfeng Guan, Shane A. McCarthy, Jonathan Wood, Kerstin Howe, Yadong Wang, and Richard Durbin. Identifying and removing haplotypic duplication in primary genome assemblies. *Bioinformatics*, 36(9):2896–2898, 2020.
- [146] Michael J Roach, Simon A Schmidt, and Anthony R Borneman. Purge Haplotigs: allelic contig reassignment for third-gen diploid genome assemblies. *BMC Bioinformatics*, 19(1):1–10, 2018.
- [147] Mihai Pop, Daniel S Kosack, and Steven L Salzberg. Hierarchical scaffolding with Bambus. *Genome Research*, 14(1):149–159, 2004.
- [148] Igor Mandric and Alex Zelikovsky. Solving scaffolding problem with repeats. *bioRxiv*, 2018.
- [149] Kristoffer Sahlin, Francesco Vezzi, Björn Nystedt, Joakim Lundeberg, and Lars Arvestad. BESST - Efficient scaffolding of large fragmented assemblies. *BMC Bioinformatics*, 15(1):1–11, 2014.
- [150] Junwei Luo, Jianxin Wang, Zhen Zhang, Min Li, and Fang-Xiang Wu. BOSS: a novel scaffolding algorithm based on an optimized scaffold graph. *Bioinformatics*, 33(2):169–176, 2017.
- [151] Alexey A Gritsenko, Jurgen F Nijkamp, Marcel JT Reinders, and Dick de Ridder. GRASS: a generic algorithm for scaffolding next-generation sequencing assemblies. *Bioinformatics*, 28(11):1429–1437, 2012.
- [152] Leena Salmela, Veli Mäkinen, Niko Välimäki, Johannes Ylinen, and Esko Ukkonen. Fast scaffolding with small independent mixed integer programs. *Bioinformatics*, 27(23):3259–3265, 2011.
- [153] Song Gao, Wing-Kin Sung, and Niranjan Nagarajan. Opera: reconstructing optimal genomic scaffolds with high-throughput paired-end sequences. *Journal of Computational Biology*, 18(11):1681–1691, 2011.
- [154] Igor Mandric and Alex Zelikovsky. ScaffMatch: scaffolding algorithm based on maximum weight matching. *Bioinformatics*, 31(16):2632–2638, 2015.
- [155] Paul M Bodily, M Stanley Fujimoto, Quinn Snell, Dan Ventura, and Mark J Clement. Scaffold-Scaffolder: solving contig orientation via bidirected to directed graph reduction. *Bioinformatics*, 32(1):17–24, 2016.
- [156] Nilgun Donmez and Michael Brudno. Scarpa: scaffolding reads with practical algorithms. *Bioinformatics*, 29(4):428–434, 2013.

- [157] Min Li, Li Tang, Fang-Xiang Wu, Yi Pan, and Jianxin Wang. SCOP: a novel scaffolding algorithm based on contig classification and optimization. *Bioinformatics*, 35(7):1142–1150, 2019.
- [158] Rajat S Roy, Kevin C Chen, Anirvan M Sengupta, and Alexander Schliep. SLIQ: Simple Linear Inequalities for Efficient Contig Scaffolding. *Journal of Computational Biology*, 19(10):1162–1175, 2012.
- [159] Adel Dayarian, Todd P Michael, and Anirvan M Sengupta. SOPRA: Scaffolding algorithm for paired reads via statistical optimization. *BMC Bioinformatics*, 11(1):1–21, 2010.
- [160] Marten Boetzer, Christiaan V Henkel, Hans J Jansen, Derek Butler, and Walter Pirovano. Scaffolding pre-assembled contigs using sspace. *Bioinformatics*, 27(4):578–579, 2011.
- [161] Gregory K Farrant, Mark Hoebeke, Frédéric Partensky, Gwendoline Andres, Erwan Corre, and Laurence Garczarek. WiseScaffolder: an algorithm for the semi-automatic scaffolding of next generation sequencing data. *BMC Bioinformatics*, 16(1):1–13, 2015.
- [162] René L Warren, Chen Yang, Benjamin P Vandervalk, Bahar Behsaz, Albert Lagman, Steven JM Jones, and Inanç Birol. LINKS: scalable, alignment-free scaffolding of draft genomes with long reads. *GigaScience*, 4(1):s13742–015, 2015.
- [163] Mao Qin, Shigang Wu, Alun Li, Fengli Zhao, Hu Feng, Lulu Ding, and Jue Ruan. LRScf: improving draft genomes using long noisy reads. *BMC Genomics*, 20(1):1–12, 2019.
- [164] Minh Duc Cao, Son Hoang Nguyen, Devika Ganesamoorthy, Alysha G Elliott, Matthew A Cooper, and Lachlan JM Coin. Scaffolding and completing genome assemblies in real-time with nanopore sequencing. *Nature Communications*, 8(1):1–10, 2017.
- [165] Adam C English, Stephen Richards, Yi Han, Min Wang, Vanesa Vee, Jiabin Qu, Xiang Qin, Donna M Muzny, Jeffrey G Reid, Kim C Worley, et al. Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology. *PloS One*, 7(11):e47768, 2012.
- [166] Rene L Warren. RAILS and Cobbler: Scaffolding and automated finishing of draft genomes using long DNA sequences. *Journal of Open Source Software*, 1(7):116, 2016.
- [167] Junwei Luo, Mengna Lyu, Ranran Chen, Xiaohong Zhang, Huimin Luo, and Chaokun Yan. SLR: a scaffolding algorithm based on long reads and contig classification. *BMC Bioinformatics*, 20(1):1–11, 2019.
- [168] Wellcome Sanger Institute. SMIS, <https://www.sanger.ac.uk/tool/smis/>, 2015.

- [169] Shenglong Zhu, Danny Z Chen, and Scott J Emrich. Single molecule sequencing-guided scaffolding and correction of draft assemblies. *BMC Genomics*, 18(10):51–59, 2017.
- [170] Marten Boetzer and Walter Pirovano. SSPACE-LongRead: scaffolding bacterial draft genomes using long read sequence information. *BMC Bioinformatics*, 15(1):1–9, 2014.
- [171] Haibao Tang, Xingtian Zhang, Chenyong Miao, Jisen Zhang, Ray Ming, James C Schnable, Patrick S Schnable, Eric Lyons, and Jianguo Lu. ALLMAPS: robust scaffold ordering based on multiple maps. *Genome Biology*, 16(1):1–15, 2015.
- [172] Henry C Lin, Steve Goldstein, Lee Mendelowitz, Shiguo Zhou, Joshua Wetzell, David C Schwartz, and Mihai Pop. AGORA: assembly guided by optical restriction alignment. *BMC Bioinformatics*, 13(1):1–14, 2012.
- [173] Benjamin Istace, Caroline Belser, and Jean-Marc Aury. BiSCoT: improving large eukaryotic genome assemblies with optical maps. *PeerJ*, 8:e10150, 2020.
- [174] Weihua Pan, Tao Jiang, and Stefano Lonardi. OMGS: optical map-based genome scaffolding. *Journal of Computational Biology*, 27(4):519–533, 2020.
- [175] Jennifer M Shelton, Michelle C Coleman, Nic Herndon, Nanyan Lu, Ernest T Lam, Thomas Anantharaman, Palak Sheth, and Susan J Brown. Tools and pipelines for BioNano data: molecule assembly pipeline and FASTA super scaffolding tool. *BMC Genomics*, 16(1):1–16, 2015.
- [176] Niranjana Nagarajan, Timothy D Read, and Mihai Pop. Scaffolding and validation of bacterial genome assemblies using optical restriction maps. *Bioinformatics*, 24(10):1229–1235, 2008.
- [177] Markus Hiltunen, Martin Ryberg, and Hanna Johannesson. ARBitR: an overlap-aware genome assembly scaffolder for linked reads. *Bioinformatics*, 2020.
- [178] Volodymyr Kuleshov, Michael P Snyder, and Serafim Batzoglou. Genome assembly from synthetic long read clouds. *Bioinformatics*, 32(12):i216–i224, 2016.
- [179] Sarah Yeo, Lauren Coombe, René L Warren, Justin Chu, and Inanç Birol. ARCS: scaffolding genome drafts with linked reads. *Bioinformatics*, 34(5):725–731, 2018.
- [180] Lauren Coombe, Jessica Zhang, Benjamin P Vandervalk, Justin Chu, Shaun D Jackman, Inanç Birol, and René L Warren. ARKS: chromosome-scale scaffolding of human genome drafts with linked read kmers. *BMC Bioinformatics*, 19(1):1–10, 2018.

- [181] Andrew Adey, Jacob O Kitzman, Joshua N Burton, Riza Daza, Akash Kumar, Lena Christiansen, Mostafa Ronaghi, Sasan Amini, Kevin L Gunderson, Frank J Steemers, et al. In vitro, long-range sequence information for de novo genome assembly via transposase contiguity. *Genome Research*, 24(12):2041–2049, 2014.
- [182] High Performance Assembly Group at the Wellcome Sanger Institute. Scaff10X, <https://github.com/wtsi-hpag/Scaff10X>, 2018.
- [183] Noam Kaplan and Job Dekker. High-throughput genome scaffolding from *in vivo* DNA interaction frequency. *Nature biotechnology*, 31(12):1143–1147, 2013.
- [184] Hervé Marie-Nelly, Martial Marbouty, Axel Cournac, Jean-François Flot, Gianni Liti, Dante Poggi Parodi, Sylvie Syan, Nancy Guillén, Antoine Margeot, Christophe Zimmer, et al. High-quality genome (re) assembly using chromosomal contact data. *Nature Communications*, 5(1):1–10, 2014.
- [185] Gina Renschler, Gautier Richard, Claudia Isabelle Keller Valsecchi, Sarah Toscano, Laura Arrigoni, Fidel Ramírez, and Asifa Akhtar. Hi-C guided assemblies reveal conserved regulatory topologies on x and autosomes despite extensive genome shuffling. *Genes & development*, 33(21-22):1591–1612, 2019.
- [186] Lyam Baudry, Nadège Guiglielmoni, Hervé Marie-Nelly, Alexandre Cormier, Martial Marbouty, Komlan Avia, Yann Loe Mie, Olivier Godfroy, Lieven Sterck, J Mark Cock, et al. instaGRAAL: chromosome-level quality scaffolding of genomes using a proximity ligation-based scaffold. *Genome Biology*, 21(1):1–22, 2020.
- [187] Joshua N Burton, Andrew Adey, Rupali P Patwardhan, Ruolan Qiu, Jacob O Kitzman, and Jay Shendure. Chromosome-scale scaffolding of *de novo* genome assemblies based on chromatin interactions. *Nature Biotechnology*, 31(12):1119–1125, 2013.
- [188] Jay Ghurye, Mihai Pop, Sergey Koren, Derek Bickhart, and Chen-Shan Chin. Scaffolding of long read assemblies using long range contact information. *BMC Genomics*, 18(1):1–11, 2017.
- [189] Jay Ghurye, Arang Rhie, Brian P Walenz, Anthony Schmitt, Siddarth Selvaraj, Mihai Pop, Adam M Phillippy, and Sergey Koren. Integrating Hi-C links with assembly graphs for chromosome-scale assembly. *PLoS Computational Biology*, 15(8):e1007273, 2019.
- [190] Marten Boetzer and Walter Pirovano. Toward almost closed genomes with GapFiller. *Genome Biology*, 13(6):1–9, 2012.
- [191] Chong Chu, Xin Li, and Yufeng Wu. GAPPadder: a sensitive approach for closing gaps on draft genomes with short sequence reads. *BMC Genomics*, 20(5):1–10, 2019.

- [192] Daniel Paulino, René L Warren, Benjamin P Vandervalk, Anthony Raymond, Shaun D Jackman, and Inanç Birol. Sealer: a scalable gap-closing application for finishing draft genomes. *BMC Bioinformatics*, 16(1):1–8, 2015.
- [193] Vitor C Piro, Helisson Faoro, Vinicius A Weiss, Maria BR Steffens, Fabio O Pedrosa, Emanuel M Souza, and Roberto T Raittz. FGAP: an automated gap closing tool. *BMC Research Notes*, 7(1):1–5, 2014.
- [194] Shunichi Kosugi, Hideki Hirakawa, and Satoshi Tabata. GMcloser: closing gaps in assemblies accurately with a likelihood-based selection of contig or long-read alignments. *Bioinformatics*, 31(23):3733–3741, 2015.
- [195] Gui-Cai Xu, Tian-Jun Xu, Rui Zhu, Yan Zhang, Shang-Qi Li, Hong-Wei Wang, and Jiong-Tang Li. LR_Gapcloser: a tiling path-based gap closer that uses long reads to complete genome assembly. *GigaScience*, 8(1):giy157, 2019.
- [196] Peng Lu, Jingjing Jin, Zefeng Li, Yalong Xu, Dasha Hu, Jiajun Liu, and Peijian Cao. PGcloser: fast parallel gap-closing tool using long-reads or contigs to fill gaps in genomes. *Evolutionary Bioinformatics*, 16, 2020.
- [197] Mengyang Xu, Lidong Guo, Shengqiang Gu, Ou Wang, Rui Zhang, Brock A Peters, Guangyi Fan, Xin Liu, Xun Xu, Li Deng, et al. TGS-GapCloser: a fast and accurate gap closer for large genomes with low coverage of error-prone long reads. *GigaScience*, 9(9):giaa094, 2020.
- [198] Pierre Morisse, Thierry Lecroq, and Arnaud Lefebvre. Long-read error correction: a survey and qualitative comparison. *bioRxiv*, 2020.
- [199] Byung June Ko, Chul Lee, Juwan Kim, Arang Rhie, DongAhn Yoo, Kerstin Howe, Jonathan Wood, Seoae Cho, Samara Brown, Giulio Formenti, et al. Widespread false gene gains caused by duplication errors in genome assemblies. *bioRxiv*, 2021.
- [200] Nadège Guiguelmoni, Antoine Houtain, Alessandro Derzelle, Karine Van Doninck, and Jean-François Flot. Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms. *BMC Bioinformatics*, 22(1):1–23, 2021.
- [201] Jay Ghurye and Mihai Pop. Modern technologies and algorithms for scaffolding assembled genomes. *PLoS computational biology*, 15(6):e1006994, 2019.
- [202] Janna L Fierst. Using linkage maps to correct and scaffold *de novo* genome assemblies: methods, challenges, and computational tools. *Frontiers in Genetics*, 6:220, 2015.

- [203] David C Schwartz, Xiaojun Li, Luis I Hernandez, Satyadarshan P Ramnarain, Edward J Huff, and Yu-Ker Wang. Ordered restriction maps of *Saccharomyces cerevisiae* chromosomes constructed by optical mapping. *Science*, 262(5130):110–114, 1993.
- [204] Anna V. Protasio, Isheng J. Tsai, Anne Babbage, Sarah Nichol, Martin Hunt, Martin A. Aslett, Nishadi de Silva, Giles S. Velarde, Tim J.C. Anderson, Richard C. Clark, Claire Davidson, Gary P. Dillon, Nancy E. Holroyd, Philip T. LoVerde, Christine Lloyd, Jacquelline McQuillan, Guilherme Oliveira, Thomas D. Otto, Sophia J. Parker-Manuel, Michael A. Quail, R. Alan Wilson, Adhemar Zerlotini, David W. Dunne, and Matthew Berriman. A systematically improved high quality genome and transcriptome of the human blood fluke *Schistosoma mansoni*. *PLoS Neglected Tropical Diseases*, 6(1), 2012.
- [205] Chang Bum Jeong, Bo Young Lee, Beom Soon Choi, Min Sub Kim, Jun Chul Park, Duck Hyun Kim, Minghua Wang, Heum Gi Park, and Jae Seong Lee. The genome of the harpacticoid copepod *Tigriopus japonicus*: potential for its use in marine molecular ecotoxicology. *Aquatic Toxicology*, 222:105462, 2020.
- [206] Zachary L Fuller, Veronique JL Mocellin, Luke A Morris, Neal Cantin, Jihanne Shepherd, Luke Sarre, Julie Peng, Yi Liao, Joseph Pickrell, Peter Andolfatto, et al. Population genetics of the coral *Acropora millepora*: Toward genomic prediction of bleaching. *Science*, 369(6501), 2020.
- [207] Yuxuan Yuan, Claire Yik-Lok Chung, and Ting-Fung Chan. Advances in optical mapping for genomic research. *Computational and Structural Biotechnology Journal*, 2020.
- [208] James A Cotton, Sasisekhar Bennuru, Alexandra Grote, Bhavana Harsha, Alan Tracey, Robin Beech, Stephen R Doyle, Matthew Dunn, Julie C Dunning Hotopp, Nancy Holroyd, et al. The genome of *Onchocerca volvulus*, agent of river blindness. *Nature Microbiology*, 2(2):1–12, 2016.
- [209] Jianbin Wang, Shenghan Gao, Yulia Mostovoy, Yuanyuan Kang, Maxim Zagoskin, Yongqiao Sun, Bing Zhang, Laura K. White, Alice Easton, Thomas B. Nutman, Pui Yan Kwok, Songnian Hu, Martin K. Nielsen, and Richard E. Davis. Comparative genome analysis of programmed DNA elimination in nematodes. *Genome Research*, 27(12):2001–2014, 2017.
- [210] Isheng J. Tsai, Magdalena Zarowiecki, Nancy Holroyd, Alejandro Garcarrubio, Alejandro Sanchez-Flores, Karen L. Brooks, Alan Tracey, Raúl J. Bobes, Gladis Fragoso, Edda Sciutto, Martin Aslett, Helen Beasley, Hayley M. Bennett, Jianping Cai, Federico Camicia, Richard Clark, Marcela Cucher, Nishadi De Silva, Tim A. Day, Peter Deplazes, Karel Estrada, Cecilia Fernández, Peter W.H. Holland, Junling Hou, Songnian Hu, Thomas Huckvale, Stacy S. Hung, Laura Kamenetzky, Jacqueline A. Keane, Ferenc Kiss, Uriel Koziol, Olivia Lambert, Kan Liu, Xuenong Luo, Yingfeng Luo,

- Natalia MacChiaroli, Sarah Nichol, Jordi Paps, John Parkinson, Natasha Pouchkina-Stantcheva, Nick Riddiford, Mara Rosenzvit, Gustavo Salinas, James D. Wasmuth, Mostafa Zamanian, Yadong Zheng, Xuepeng Cai, Xavier Soberon, Peter D. Olson, Juan P. Laclette, Klaus Brehm, Matthew Berriman, Enrique Morett, Tobias Portillo, Marco V. Jose, Julio C. Carrero, Carlos Larralde, Jorge Morales-Montor, Jorge Limon-Lason, Miguel A. Cevallos, Victor Gonzalez, Adrian Ochoa-Leyva, Abraham Landa, Lucia Jimenez, and Victor Valdes. The genomes of four tapeworm species reveal adaptations to parasitism. *Nature*, 496(7443):57–63, 2013.
- [211] Peter D Olson, Alan Tracey, Andrew Baillie, Katherine James, Stephen R Doyle, Sarah K Buddenborg, Faye H Rodgers, Nancy Holroyd, and Matt Berriman. Complete representation of a tapeworm genome reveals chromosomes capped by centromeres, necessitating a dual role in segregation and protection. *BMC Biology*, 18(1):1–16, 2020.
- [212] Rebecca M Varney, Daniel I Speiser, Carmel McDougall, Bernard M Degnan, and Kevin M Kocot. The iron-responsive genome of the chiton *Acanthopleura granulata*. *Genome Biology and Evolution*, 13(1):evaa263, 2021.
- [213] Sarah D Kocher, Ricardo Mallarino, Benjamin ER Rubin, W Yu Douglas, Hopi E Hoekstra, and Naomi E Pierce. The genetic basis of a social polymorphism in halictid bees. *Nature Communications*, 9(1):1–7, 2018.
- [214] SuperNova. SuperNova, <https://github.com/10XGenomics/supernova>, 2016.
- [215] Jay Ghurye, Sergey Koren, Scott T Small, Seth Redmond, Paul Howell, Adam M Phillippy, and Nora J Besansky. A chromosome-scale assembly of the major african malaria vector *Anopheles funestus*. *GigaScience*, 8(6):giz063, 2019.
- [216] Job Dekker, Karsten Rippe, Martijn Dekker, and Nancy Kleckner. Capturing chromosome conformation. *Science*, 295(5558):1306–1311, 2002.
- [217] Job Dekker, Marc A Marti-Renom, and Leonid A Mirny. Exploring the three-dimensional organization of genomes: interpreting chromatin interaction data. *Nature Reviews Genetics*, 14(6):390–403, 2013.
- [218] Erez Lieberman-Aiden, Nynke L van Berkum, Louise Williams, Maxim Imakaev, Tobias Ragozy, Agnes Telling, Ido Amit, Bryan R Lajoie, Peter J Sabo, Michael O Dorschner, Richard Sandstrom, Bradley Bernstein, M A Bender, Mark Groudine, Andreas Gnirke, John Stamatoyannopoulos, Leonid A Mirny, Eric S Lander, and Job Dekker. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–93, 2009.

- [219] Jean-François Flot, Hervé Marie-Nelly, and Romain Koszul. Contact genomics: scaffolding and phasing (meta) genomes using chromosome 3d physical signatures. *FEBS Letters*, 589(20):2966–2974, 2015.
- [220] Sivan Oddes, Aviv Zelig, and Noam Kaplan. Three invariant Hi-C interaction patterns: applications to genome assembly. *Methods*, 142:89–99, 2018.
- [221] Maeva A Techer, Rahul V Rane, Miguel L Grau, John Mk Roberts, Shawn T Sullivan, Ivan Liachko, Anna K Childers, Jay D Evans, and Alexander S Mikheyev. Divergent evolutionary trajectories following speciation in two ectoparasitic honey bee mites. *Communications Biology*, 2(1):1–16, 2019.
- [222] Prashant Shingate, Vydianathan Ravi, Aravind Prasad, Boon-Hui Tay, Kritika M Garg, Balaji Chattopadhyay, Laura-Marie Yap, Frank E Rheindt, and Byrappa Venkatesh. Chromosome-level assembly of the horseshoe crab genome provides insights into its genome evolution. *Nature Communications*, 11(1):1–13, 2020.
- [223] Minjie Hu, Xiaobin Zheng, Chen-Ming Fan, and Yixian Zheng. Lineage dynamics of the endosymbiotic cell type in the soft coral *Xenia*. *Nature*, 582(7813):534–538, 2020.
- [224] Yunfeng Li, Lei Gao, Yongjia Pan, Meilin Tian, Yulong Li, Chongbo He, Ying Dong, Yamin Sun, and Zunchun Zhou. Chromosome-level reference genome of the jellyfish *Rhopilema esculentum*. *GigaScience*, 9(4):giaa036, 2020.
- [225] Phillip L Davidson, Haobing Guo, Lingyu Wang, Alejandro Berrio, He Zhang, Andrew L Soborowski, David R McClay, Guangyi Fan, and Gregory A Wray. Chromosomal-level genome assembly of the sea urchin *Lytechinus variegatus* substantially improves functional genomic analyses. *Genome Biology and Evolution*, 12(7):1080–1086, 2020.
- [226] Dannise V Ruiz-Ramos, Lauren M Schiebelhut, Katharina J Hoff, John P Wares, and Michael N Dawson. An initial comparative genomic autopsy of wasting disease in sea stars. *Molecular Ecology*, 29(6):1087–1102, 2020.
- [227] Chang Ming Bai, Lu Sheng Xin, Umberto Rosani, Biao Wu, Qing Chen Wang, Xiao Ke Duan, Zhi Hong Liu, and Chong Ming Wang. Chromosomal-level assembly of the blood clam, *Scapharca (Anadara) broughtonii*, using long sequence reads and Hi-C. *GigaScience*, 8(7):1–8, 2019.
- [228] Jin Sun, Chong Chen, Norio Miyamoto, Runsheng Li, Julia D. Sigwart, Ting Xu, Yanan Sun, Wai Chuen Wong, Jack C.H. Ip, Weipeng Zhang, Yi Lan, Dass Bissessur, Tomo o. Watsuji, Hiromi Kayama Watanabe, Yoshihiro Takaki, Kazuho Ikeo, Nobuyuki Fujii, Kazutoshi Yoshitake,

- Jian Wen Qiu, Ken Takai, and Pei Yuan Qian. The Scaly-foot Snail genome and implications for the origins of biomineralised armour. *Nature Communications*, 11(1), 2020.
- [229] Anastasia A Teterina, John H Willis, and Patrick C Phillips. Chromosome-level assembly of the *Caenorhabditis remanei* genome reveals conserved patterns of nematode genome organization. *Genetics*, 214(4):769–780, 2020.
- [230] Yun Lian, He Wei, Jinshe Wang, Chenfang Lei, Haichao Li, Jinying Li, Yongkang Wu, Shufeng Wang, Hui Zhang, Tingfeng Wang, et al. Chromosome-level reference genome of x12, a highly virulent race of the soybean cyst nematode *Heterodera glycines*. *Molecular Ecology Resources*, 19(6):1637–1646, 2019.
- [231] Andreas J. Stroehlein, Pasi K. Korhonen, Teik Min Chong, Yan Lue Lim, Kok Gan Chan, Bonnie Webster, David Rollinson, Paul J. Brindley, Robin B. Gasser, and Neil D. Young. High-quality *Schistosoma haematobium* genome achieved by single-molecule and long-range sequencing. *GigaScience*, 8(9):1–12, 2019.
- [232] Nathan J Kenny, Warren R Francis, Ramón E Rivera-Vicéns, Ksenia Juravel, Alex de Mendoza, Cristina Díez-Vives, Ryan Lister, Luis A Bezares-Calderón, Lauren Grombacher, Maša Roller, et al. Tracing animal genomic evolution with the chromosomal-level assembly of the freshwater sponge *Ephydatia muelleri*. *Nature Communications*, 11(1):1–11, 2020.
- [233] Paul Simion, Jitendra Narayan, Antoine Houtain, Alessandro Derzelle, Lyam Baudry, Emilien Nicolas, Marie Cariou, Nadège Guiglielmoni, Djampa Kozłowski, Florence Rodriguez Gaudray, et al. Homologous chromosomes in asexual rotifer *Adineta vaga* suggest automixis. *bioRxiv*, 2020.
- [234] Andrew R Gehrke, Emily Neverett, Yi-Jyun Luo, Alexander Brandt, Lorenzo Ricci, Ryan E Hulet, Annika Gompers, J Graham Ruby, Daniel S Rokhsar, Peter W Reddien, et al. Acoel genome reveals the regulatory landscape of whole-body regeneration. *Science*, 363(6432), 2019.
- [235] Kelly L. Mulligan, Terra C. Hiebert, Nicholas W. Jeffery, and T. Ryan Gregory. First estimates of genome size in ribbon worms (phylum Nemertea) using flow cytometry and Feulgen image analysis densitometry. *Canadian Journal of Zoology*, 92(10):847–851, 2014.
- [236] Joint Genome Institute. BBtools, <https://sourceforge.net/projects/bbmap/>, 2014.
- [237] T. Rhyker Ranallo-Benavidez, Kamil S. Jaron, and Michael C. Schatz. GenomeScope 2.0 and Smudgeplot for reference-free profiling of polyploid genomes. *Nature Communications*, 11(1):1432, 2020.

- [238] Daniel Mapleson, Gonzalo Garcia Accinelli, George Kettleborough, Jonathan Wright, and Bernardo J. Clavijo. KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies. *Bioinformatics*, 33(4):574–576, 2016.
- [239] Boas Pucker. Mapping-based genome size estimation. *bioRxiv*, 2019.
- [240] Alexey Gurevich, Vladislav Saveliev, Nikolay Vyahhi, and Glenn Tesler. QUASt: quality assessment tool for genome assemblies. *Bioinformatics*, 29(8):1072–1075, 2013.
- [241] Yunhai Guo, Yi Zhang, Qin Liu, Yun Huang, Guangyao Mao, Zhiyuan Yue, Eniola M. Abe, Jian Li, Zhongdao Wu, Shizhu Li, Xiaonong Zhou, Wei Hu, and Ning Xiao. A chromosomal-level genome assembly for the giant African snail *Achatina fulica*. *GigaScience*, 8(10):1–8, 2019.
- [242] Cheng He, Guifang Lin, Hairong Wei, Haibao Tang, Frank F White, Barbara Valent, and Sanzhen Liu. Factorial estimating assembly base errors using k -mer abundance difference (KAD) between short reads and genome assembled sequences. *NAR Genomics and Bioinformatics*, 2(3):lqaa075, 2020.
- [243] Dominik R Laetsch and Mark L Blaxter. BlobTools: Interrogation of genome assemblies. *F1000Research*, 6(1287):1287, 2017.
- [244] Richard Challis, Edward Richards, Jeena Rajan, Guy Cochrane, and Mark Blaxter. BlobToolKit—Interactive quality assessment of genome assemblies. *G3: Genes, Genomes, Genetics*, 10(4):1361–1374, 2020.
- [245] Thomas C Boothby, Jennifer R Tenlen, Frank W Smith, Jeremy R Wang, Kiera A Patanella, Erin Osborne Nishimura, Sophia C Tintori, Qing Li, Corbin D Jones, Mark Yandell, et al. Evidence for extensive horizontal gene transfer from the draft genome of a tardigrade. *Proceedings of the National Academy of Sciences*, 112(52):15976–15981, 2015.
- [246] Georgios Koutsovoulos, Sujai Kumar, Dominik R Laetsch, Lewis Stevens, Jennifer Daub, Claire Conlon, Habib Maroon, Fran Thomas, Aziz A Aboobaker, and Mark Blaxter. No evidence for extensive horizontal gene transfer in the genome of the tardigrade *Hypsibius dujardini*. *Proceedings of the National Academy of Sciences*, 113(18):5053–5058, 2016.
- [247] Xingtian Zhang, Ruoxi Wu, Yibin Wang, Jiaxin Yu, and Haibao Tang. Unzipping haplotypes in diploid and polyploid genomes. *Computational and Structural Biotechnology Journal*, 18:66–72, 2020.

- [248] Sergey Koren, Arang Rhie, Brian P Walenz, Alexander T Dilthey, Derek M Bickhart, Sarah B Kingan, Stefan Hiendleder, John L Williams, Timothy PL Smith, and Adam M Phillippy. *De novo* assembly of haplotype-resolved genomes with trio binning. *Nature Biotechnology*, 36(12):1174–1182, 2018.
- [249] Peter Edge, Vineet Bafna, and Vikas Bansal. HapCUT2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Research*, 27(5):801–812, 2017.
- [250] Murray D. Patterson, Tobias Marschall, Nadia Pisanti, Leo Van Iersel, Leen Stougie, Gunnar W. Klau, and Alexander Schönhuth. WhatsHap: weighted haplotype assembly for future-generation sequencing reads. *Journal of Computational Biology*, 22(6):498–509, 2015.
- [251] Antoine Limasset. *Novel approaches for the exploitation of high throughput sequencing data*. PhD thesis, Université Rennes 1, 2017.
- [252] Rei Kajitani, Dai Yoshimura, Miki Okuno, Yohei Minakuchi, Hiroshi Kagoshima, Asao Fujiyama, Kaoru Kubokawa, Yuji Kohara, Atsushi Toyoda, and Takehiko Itoh. Platanus-allee is a *de novo* haplotype assembler enabling a comprehensive access to divergent heterozygous regions. *Nature Communications*, 10(1):1–15, 2019.
- [253] David Porubsky, Peter Ebert, Peter A Audano, Mitchell R Vollger, William T Harvey, Pierre Marijon, Jana Ebler, Katherine M Munson, Melanie Sorensen, Arvis Sulovari, et al. Fully phased human genome assembly without parental data using single-cell strand sequencing and long reads. *Nature Biotechnology*, pages 1–7, 2020.
- [254] Qian Zhou, Dié Tang, Wu Huang, Zhongmin Yang, Yu Zhang, John P Hamilton, Richard GF Visser, Christian WB Bachem, C Robin Buell, Zhonghua Zhang, et al. Haplotype-resolved genome analyses of a heterozygous diploid potato. *Nature Genetics*, pages 1–6, 2020.
- [255] Guillaume Holley, Doruk Beyter, Helga Ingimundardottir, Peter L Møller, Snædis Kristmundsdottir, Hannes P Eggertsson, and Bjarni V Halldorsson. Ratatosk: hybrid error correction of long reads enables accurate variant calling and assembly. *Genome Biology*, 22(1):1–22, 2021.
- [256] Jean-Marc Aury and Benjamin Istace. Hapo-G, haplotype-aware polishing of genome assemblies. *NAR Genomics and Bioinformatics*, 2021.
- [257] Xingtian Zhang, Shengcheng Zhang, Qian Zhao, Ray Ming, and Haibao Tang. Assembly of allele-aware, chromosomal-scale autopolyploid genomes based on Hi-C data. *Nature Plants*, 5(8):833–845, 2019.

- [258] Zev N Kronenberg, Arang Rhie, Sergey Koren, Gregory T Concepcion, Paul Peluso, Katherine M Munson, David Porubsky, Kristen Kuhn, Kathryn A Mueller, Wai Yee Low, et al. Extended haplotype-phasing of long-read de novo genome assemblies using hi-c. *Nature Communications*, 12(1):1–10, 2021.
- [259] Arang Rhie, Brian P Walenz, Sergey Koren, and Adam M Phillippy. Merqury: reference-free quality, completeness, and phasing assessment for genome assemblies. *Genome Biology*, 21(1):1–27, 2020.
- [260] Cyril Matthey-Doret, Lyam Baudry, Amaury Bignaud, Axel Cournac, Rémi Montagne, Nadège Guiguelmoni, Théo Foutel-Rodier, and Vittore F. Scolari. koszullab/hicstuff, October 2020.
- [261] Mikko Rautiainen and Tobias Marschall. GraphAligner: rapid and versatile sequence-to-graph alignment. *Genome Biology*, 21(1):1–28, 2020.
- [262] Shilpa Garg, Arkarachai Fungtammasan, Andrew Carroll, Mike Chou, Anthony Schmitt, Xiang Zhou, Stephen Mac, Paul Peluso, Emily Hatas, Jay Ghurye, et al. Chromosome-scale, haplotype-resolved assembly of human genomes. *Nature Biotechnology*, 39(3):309–312, 2021.
- [263] Vertebrate Genomes Project. Vertebrate Genomes Project, <https://vertebratengenomesproject.org>, 2021.
- [264] Jean-François Flot, Boris Hespeels, Xiang Li, Benjamin Noel, Irina Arkhipova, Etienne GJ Danchin, Andreas Hejnol, Bernard Henrissat, Romain Koszul, Jean-Marc Aury, et al. Genomic evidence for ameiotic evolution in the bdelloid rotifer *Adineta vaga*. *Nature*, 500(7463):453–457, 2013.
- [265] Esther C Peters, Stephen D Cairns, MEQ Pilson, JW Wells, WC Jaap, JC Lang, CE Vasleski, and L St Pierre Gollahon. Nomenclature and biology of *Astrangia poculata* (= *a. danae*, = *a. astreiformis*)(cnidaria: Anthozoa). *Proceedings of the Biological Society of Washington*, 1988.
- [266] JL Dimond, AH Kerwin, R Rotjan, K Sharp, FJ Stewart, and DJ Thornhill. A simple temperature-based model predicts the upper latitudinal limit of the temperate coral *Astrangia poculata*. *Coral Reefs*, 32(2):401–409, 2013.
- [267] Ryan R. Wick. Porechop, <https://github.com/rrwick/Porechop>, 2017.
- [268] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, 2012.
- [269] Chuya Shinzato, Konstantin Khalturin, Jun Inoue, Yuna Zayasu, Miyuki Kanda, Mayumi Kawamitsu, Yuki Yoshioka, Hiroshi Yamashita, Go Suzuki, and Noriyuki Satoh. Eighteen coral

-
- genomes reveal the evolutionary origin of *Acropora* strategies to accommodate environmental changes. *Molecular Biology and Evolution*, 38(1):16–30, 2021.
- [270] Chuya Shinzato, Eiichi Shoguchi, Takeshi Kawashima, Mayuko Hamada, Kanako Hisata, Makiko Tanaka, Manabu Fujie, Mayuki Fujiwara, Ryo Koyanagi, Tetsuro Ikuta, et al. Using the *Acropora digitifera* genome to understand coral responses to environmental change. *Nature*, 476(7360):320–323, 2011.
- [271] Hua Ying, David C Hayward, Ira Cooke, Weiwen Wang, Aurelie Moya, Kirby R Siemering, Susanne Sprungala, Eldon E Ball, Sylvain Forêt, and David J Miller. The whole-genome sequence of the coral *Acropora millepora*. *Genome Biology and Evolution*, 11(5):1374–1379, 2019.
- [272] Claire Hoencamp, Olga Dudchenko, Ahmed MO Elbatsh, Sumitabha Brahmachari, Jonne A Raaijmakers, Tom van Schaik, Ángela Sedeño Cacciatore, Vinícius G Contessoto, Roy GHP van Heesbeen, Bram van den Broek, Aditya N Mhaskar, Hans Teunissen, Brian Glenn St Hilaire, David Weisz, Arina D Omer, Melanie Pham, Zane Colaric, Zhenzhen Yang, Suhas SP Rao, Namita Mitra, Christopher Lui, Weijie Yao, Ruqayya Khan, Leonid L Moroz, Andrea Kohn, Judy St. Leger, Alexandria Mena, Karen Holcroft, Maria Cristina Gambetta, Fabian Lim, Emma Farley, Nils Stein, Alexander Haddad, Daniel Chauss, Ayse Sena Mutlu, Meng C Wang, Neil D Young, Evin Hildebrandt, Hans H Cheng, Christopher J Knight, Theresa LU Burnham, Kevin A Hovel, Andrew J Beel, Pierre-Jean Mattei, Roger D Kornberg, Wesley C Warren, Gregory Cary, José Luis Gómez-Skarmeta, Veronica Hinman, Kerstin Lindblad-Toh, Federica Di Palma, Kazuhiro Maeshima, Asha S Multani, Sen Pathak, Liesl Nel-Themaat, Richard R Behringer, Parwinder Kaur, René H Medema, Bas van Steensel, Elzo de Wit, José N Onuchic, Michele Di Pierro, Erez Lieberman Aiden, and Benjamin D Rowland. 3D genomics across the tree of life reveals condensin II as a determinant of architecture type. *Science*, 372(6545):984–989, 2021.
- [273] Alexander Shumaker, Hollie M Putnam, Huan Qiu, Dana C Price, Ehud Zelzion, Arye Harel, Nicole E Wagner, Ruth D Gates, Hwan Su Yoon, and Debashish Bhattacharya. Genome analysis of the rice coral *Montipora capitata*. *Scientific Reports*, 9(1):1–16, 2019.
- [274] Martin Helmkampf, M Renee Bellinger, Scott M Geib, Sheina B Sim, and Misaki Takabayashi. Draft genome of the rice coral *Montipora capitata* obtained from linked-read sequencing. *Genome Biology and Evolution*, 11(7):2045–2054, 2019.
- [275] Carlos Prada, Bishoy Hanna, Ann F Budd, Cheryl M Woodley, Jeremy Schmutz, Jane Grimwood, Roberto Iglesias-Prieto, John M Pandolfi, Don Levitan, Kenneth G Johnson, et al. Empty niches
-

- after extinctions increase population sizes of modern corals. *Current Biology*, 26(23):3190–3194, 2016.
- [276] Ross Cunning, RA Bay, P Gillette, AC Baker, and N Traylor-Knowles. Comparative analysis of the pocillopora damicornis genome highlights role of immune system in coral evolution. *Scientific Reports*, 8(1):1–10, 2018.
- [277] Christian R Woolstra, Yong Li, Yi Jin Liew, Sebastian Baumgarten, Didier Zoccola, Jean-François Flot, Sylvie Tambutté, Denis Allemand, and Manuel Aranda. Comparative analysis of the genomes of *Stylophora pistillata* and *Acropora digitifera* provides evidence for extensive differences between species of corals. *Scientific Reports*, 7(1):1–14, 2017.
- [278] Craig S Wilding, Nicola Fletcher, Eleanor K Smith, Peter Prentis, Gareth D Weedall, and Zac Stewart. The genome of the sea anemone *Actinia equina* (L.): Meiotic toolkit genes and the question of sexual reproduction. *Marine Genomics*, 53:100753, 2020.
- [279] Joachim M Surm, Zachary K Stewart, Alexie Papanicolaou, Ana Pavasovic, and Peter J Prentis. The draft genome of *Actinia tenebrosa* reveals insights into toxin evolution. *Ecology and Evolution*, 9(19):11314–11328, 2019.
- [280] Sebastian Baumgarten, Oleg Simakov, Lisl Y Esherrick, Yi Jin Liew, Erik M Lehnert, Craig T Michell, Yong Li, Elizabeth A Hambleton, Annika Guse, Matt E Oates, et al. The genome of *Aiptasia*, a sea anemone model for coral symbiosis. *Proceedings of the National Academy of Sciences*, 112(38):11893–11898, 2015.
- [281] Nicholas H Putnam, Mansi Srivastava, Uffe Hellsten, Bill Dirks, Jarrod Chapman, Asaf Salamov, Astrid Terry, Harris Shapiro, Erika Lindquist, Vladimir V Kapitonov, et al. Sea anemone genome reveals ancestral eumetazoan gene repertoire and genomic organization. *Science*, 317(5834):86–94, 2007.
- [282] Xin Wang, Yi Jin Liew, Yong Li, Didier Zoccola, Sylvie Tambutte, and Manuel Aranda. Draft genomes of the corallimorpharians *Amplexidiscus fenestrafer* and *Discosoma* sp. *Molecular ecology resources*, 17(6):e187–e195, 2017.
- [283] Yeonsu Jeon, Seung Gu Park, Nayun Lee, Jessica A Weber, Hui-Su Kim, Sung-Jin Hwang, Seonock Woo, Hak-Min Kim, Youngjune Bhak, Sungwon Jeon, et al. The draft genome of an octocoral, *Dendronephthya gigantea*. *Genome Biology and Evolution*, 11(3):949–953, 2019.
- [284] Jean-Baptiste Ledoux, Fernando Cruz, Jèssica Gómez-Garrido, Regina Antoni, Julie Blanc, Daniel Gómez-Gras, Silvija Kipson, Paula López-Sendino, Agostinho Antunes, Cristina Linares, et al. The

- genome sequence of the octocoral *Paramuricea clavata* - a key resource to study the impact of climate change in the Mediterranean. *G3: Genes, Genomes, Genetics*, 10(9):2941–2952, 2020.
- [285] Justin B Jiang, Andrea M Quattrini, Warren R Francis, Joseph F Ryan, Estefania Rodriguez, and Catherine S McFadden. A hybrid *de novo* assembly of the sea pansy (*Renilla muelleri*) genome. *GigaScience*, 8(4):giz026, 2019.
- [286] Angeles Alvarino. Bathymetric distribution of chaetognaths. *Pacific Science*, XVIII, 1964.
- [287] Elvezio Ghirardelli. Some aspects of the biology of the chaetognaths. *Advances in Marine Biology*, 6:271–375, 1969.
- [288] Takasi Tokioka. The taxonomical outline of chaetognatha. *Publications of the Seto Marine Biological Laboratory*, 12(5):335–357, 1965.
- [289] Maximilian J Telford and Peter WH Holland. The Phylogenetic Affinities of the Chaetognaths: A Molecular Analysis. *Molecular Biology and Evolution*, 10(3):660–676, 1993.
- [290] Hiroshi Wada and Noriyuki Satoh. Details of the evolutionary history from invertebrates to vertebrates, as deduced from the sequences of 18S rDNA. *Proceedings of the National Academy of Sciences*, 91(5):1801–1804, 1994.
- [291] Maximilian J Telford and Peter WH Holland. Evolution of 28S Ribosomal DNA in Chaetognaths: Duplicate Genes and Molecular Phylogeny. *Journal of Molecular Evolution*, 44(2):135–144, 1997.
- [292] Claus Nielsen. *Animal evolution: interrelationships of the living phyla*. Oxford University Press on Demand, 2011.
- [293] Ferdinand Marlétaz, Katja TCA Peijnenburg, Taichiro Goto, Noriyuki Satoh, and Daniel S Rokhsar. A new spiralian phylogeny places the enigmatic arrow worms among gnathiferans. *Current Biology*, 29(2):312–318, 2019.
- [294] Harding B Michel. *Chaetognatha of the Caribbean Sea and adjacent areas*, volume 15. National Oceanic and Atmospheric Administration, National Marine Fisheries, 1984.
- [295] T Ryan Gregory. Animal Genome Size Database, <http://www.genomesize.com>, unpublished data, 2021.
- [296] Reuben W Nowell, Pedro Almeida, Christopher G Wilson, Thomas P Smith, Diego Fontaneto, Alastair Crisp, Gos Micklem, Alan Tunnacliffe, Chiara Boschetti, and Timothy G Barraclough. Comparative genomics of bdelloid rotifers: insights from desiccating and nondesiccating species. *PLoS Biology*, 16(4):e2004830, 2018.

- [297] Ferdinand Marlétaz, André Gilles, Xavier Caubit, Yvan Perez, Carole Dossat, Sylvie Samain, Gabor Gyapay, Patrick Wincker, and Yannick Le Parco. Chaetognath transcriptome reveals ancestral and unique features among bilaterians. *Genome Biology*, 9(6):1–18, 2008.
- [298] Cyril Matthey-Doret, Lyam Baudry, Axel Breuer, Rémi Montagne, Nadège Guiguelmoni, Vittore Scolari, Etienne Jean, Arnaud Campeas, Philippe Henri Chanut, Edgar Oriol, et al. Computer vision for pattern detection in chromosome contact maps. *Nature communications*, 11(1):1–11, 2020.
- [299] Zeyuan Chen, Özgül Doğan, Nadège Guiguelmoni, Anne Guichard, and Michael Schrödl. The *de novo* genome of the "spanish" slug *arion vulgaris* moquin-tandon, 1855 (gastropoda: Panpulmonata): massive expansion of transposable elements in a major pest species. *bioRxiv*, 2020.
- [300] Sergey Nurk, Sergey Koren, Arang Rhie, Mikko Rautiainen, Andrey V. Bzikadze, Alla Mikheenko, Mitchell R. Vollger, Nicolas Altemose, Lev Uralsky, Ariel Gershman, Sergey Aganezov, Savannah J. Hoyt, Mark Diekhans, Glennis A. Logsdon, Michael Alonge, Stylianos E. Antonarakis, Matthew Borchers, Gerard G. Bouffard, Shelise Y. Brooks, Gina V. Caldas, Haoyu Cheng, Chen-Shan Chin, William Chow, Leonardo G. de Lima, Philip C. Dishuck, Richard Durbin, Tatiana Dvorkina, Ian T. Fiddes, Giulio Formenti, Robert S. Fulton, Arkarachai Fungtammasan, Erik Garrison, Patrick G. S. Grady, Tina A. Graves-Lindsay, Ira M. Hall, Nancy F. Hansen, Gabrielle A. Hartley, Marina Haukness, Kerstin Howe, Michael W. Hunkapiller, Chirag Jain, Miten Jain, Erich D. Jarvis, Peter Kerpedjiev, Melanie Kirsche, Mikhail Kolmogorov, Jonas Korlach, Milinn Kremitzki, Heng Li, Valerie V. Maduro, Tobias Marschall, Ann M. McCartney, Jennifer McDaniel, Danny E. Miller, James C. Mullikin, Eugene W. Myers, Nathan D. Olson, Benedict Paten, Paul Peluso, Pavel A. Pevzner, David Porubsky, Tamara Potapova, Evgeny I. Rogaev, Jeffrey A. Rosenfeld, Steven L. Salzberg, Valerie A. Schneider, Fritz J. Sedlazeck, Kishwar Shafin, Colin J. Shew, Alaina Shumate, Yumi Sims, Arian F. A. Smit, Daniela C. Soto, Ivan Sovic, Jessica M. Storer, Aaron Streets, Beth A. Sullivan, Francoise Thibaud-Nissen, James Torrance, Justin Wagner, Brian P. Walenz, Aaron Wenger, Jonathan M. D. Wood, Chunlin Xiao, Stephanie M. Yan, Alice C. Young, Samantha Zarate, Urvashi Surti, Rajiv C. McCoy, Megan Y. Dennis, Ivan A. Alexandrov, Jennifer L. Gerton, Rachel J. O'Neill, Winston Timp, Justin M. Zook, Michael C. Schatz, Evan E. Eichler, Karen H. Miga, and Adam M. Phillippy. The complete sequence of a human genome. *bioRxiv*, 2021.