GALA: gap-free chromosome-scale assembly with long reads

Mohamed Awad & Xiangchao Gan*

Max Planck Institute for Plant Breeding Research, Carl-von-Linné-Weg 10, 50829 Köln, Germany

* Authors for correspondence: e-mail address (gan@mpipz.mpg.de) phone: +49 (221) 5062158, fax: +49 (221) 5062674

Abstract

High-quality genome assembly has wide applications in genetics and medical studies. However, it is still very challenging to achieve gap-free chromosome-scale assemblies using current workflows of long-read platforms. Here we propose a chromosome-by-chromosome assembly strategy implemented through the multiple-layer computer graph which identifies mis-assemblies within preliminary assemblies or chimeric raw reads and partitions the data into chromosome-scale linkage groups. The subsequent independent assembly of each linkage group generates gap-free assembly free from the mis-assembly errors which usually plague existing workflows. This flexible framework also allows us to integrate data from various technologies, such as Pacbio, Nanopore, Hi-C, and the genetic map, to generate gap-free chromosome-scale assembly. We *de novo* assembled *C. elegans* and *A. thaliana* genomes using GALA with combined Pacbio and Nanopore sequening data from publicly available datasets. We also demonstrated its applicability with a gap-free assembly of two chromosomes in the human genome. In addition, GALA showed promising performance for Pacbio high-fidelity long reads. Our method enables straightforward assembly of genomes with multiple data sources and multiple computational tools, overcoming barriers that at present restrict the application of de novo genome assembly technology.

Keywords

genome assembly, multiple-layer computer graph, long reads, Pacbio sequencing, Nanopore sequencing

Main

De novo genome assembly has wide applications in plant, animal and human genetics. It is still impossible for long-read platforms, such as Nanopore and Pacbio, to provide chromosome-scale sequences [1, 2]. Plenary de novo assembly tools have been developed to obtain longer and more accurate representative sequences from raw sequencing data [3-5]. However, assemblies by these tools comprise hundreds or even thousands of contigs in most of studies. To produce chromosome-scale assembly, various technologies, such as Hi-C, Bionano and the genetic map, have been increasingly used to anchor contigs into big scaffolds[6, 7]. As a consequence, the final genome assembly usually contains numerous gaps, and sometimes could be plagued with mis-assemblies [8, 9].

Gaps and mis-assemblies in a genome assembly could seriously undermine genomic studies. For example, a lot of sequence alignment tools have much lower performances when query sequences contain gaps [10, 11]. In intraspecific genome comparison, large gaps not only significantly increase the possibility of failure to detect long structure variants, but also produce inaccurate results of gene annotation [12, 13]. It has been reported that gaps and mis-assembly accounts for a large number of gene model error in the existing genome assembly studies [14, 15].

We developed a scalable method, GALA, for gap-free chromosome-scale assembly (**Fig. 1**). GALA separates two steps: firstly, identifying multiple linkage groups in the genome, each representing a single chromosome, and describing chromosome structure with raw reads and assembled contigs from multiple *de novo* assembly

tools; secondly, assembly of each linkage group by integrating results from multiple assembly tools and inferring from the raw reads. Our method can also exploit the information from Hi-C data to obtain chromosome-scale linkage groups in the studies where straightforward application of GALA failed to achieve it usually due to complicated genome structure conjugated with low sequencing quality. The straightforward utility of Hi-C information can be easily extended to other source of information, such as genetic map or even genome synteny information.

We demonstrate the utility of GALA by gap-free and chromosome-scale assemblies of Pacbio or Nanopore sequencing data from two publicly available datasets where the original assembly contains large gaps and a number of unanchored scaffolds. Our approach performs way better than existing algorithms in both datasets. We also demonstrate the application of our method to refine an existing gapped assembly with a gap-free assembly of two human chromosomes.

Results

Overview of GALA framework

GALA exploits the information from multiple *de novo* assembly tools and raw reads, and potential other sources of information, such as Hi-C, Bionano and genetic map if exist. In GALA, a batch of de novo assembly tools are selected to create preliminary assemblies at first. The multiple assemblies and raw reads are then aligned against each other. We use a multiple-layer computer graph to model the GALA, with each assembly encoded as one layer, together with an extra layer representing the raw reads. Inside each layer, a contigs (or a read in raw-read layer) is encoded as a graph node. GALA browses through the reciprocal alignments and creates two types of edges. Any contradictory information between multiple assemblies or raw reads will be recorded as a cross-layer edge. Inside each layer, if two nodes both partially overlap with the same node inside a different layer, a within-layer edge were created between them (**Fig. 2**).

Depending on the raw read quality and complexity of the genome structure, existing assembly software tools usually exhibit different performances in terms of the number of misassembled contigs and N50. To prevent the spread of errors, we developed a mis-assembly detection module (MDM). It is achieved through cutting out the contradictory cross-layer edges by splitting nodes (Online Methods). After removing contradictory cross-layer links, the contig-clustering module (CCM) pools the linked nodes within different layers and those inside the same layer into different linkage groups, usually each representing a chromosome (Online Methods). In several experiments, we identified orphan contigs. Interestingly, most of them come from external sources such as bacterial, or sample contamination.

The successful partitioning of existing preliminary assemblies and raw reads into separate linkage groups allows us essentially a chromosome-by-chromosome assembly. The raw reads from each linkage group are extracted and assembled with multiple assembly tools and merged together if necessary. For those tools which take corrected reads as input, we correct reads using existing tools. Interestingly, we found that chromosome-by-chromosome assembly always provide better performance, especially for the repetitive fragments, while the improvement of read-correction with chromosome-by-chromosome analysis is trivial. GALA also provides a fast mode, where the consensus assembly for each chromosome is obtained by merging the assembled contigs within the linkage group without working on raw reads. However, in many cases, the fast mode generated gapped

assemblies, indicating a big advantage of the chromosome-by-chromosome assembly strategy over existing tools.

***Caenorhabditis elegans* genome assembly:**

We used a publicly available dataset for *Caenorhabditis elegans* VC2010. The dataset was generated on Pacbio platform with a coverage 290X, and with extra 32X coverage of Nanopore sequence [16]. As no assembly tools support pooled sequencing data from Pacbio and nanopore platforms, we used both datasets separately to generate preliminary assemblies (**Supplementary Fig. 1**). Preliminary assemblies were generated using Canu, Flye, Miniasm+Minimap2/Minimap, Wtdbg2, Mecat/Necat (**Supplementary Table 1** and Online Methods). Among all our preliminary assemblies, the one produced by Pacbio-Flye showed the smallest number of contigs, with 41 contigs for 102Mbp of overall sequence.

We rans GALA on the raw reads and 10 of the preliminary assemblies. The numbers of mis-assemblies in each preliminary assembly reported by MDM algorithm ranged from 0 to 19. After removing the mis- scaffolds through node-splitting operation, GALA modelled the input into 14 independent linkage groups, 7 of them are from short continuous contigs and the others represent individual chromosomes or chromosome arms. After removing the 4 contigs from bacterial contamination or organelle DNA, we were able to merge the graph further into 6 linkage groups through telomeric motif analyses (**Supplementary Fig. 2** and Online Methods). The integrative assembly of each linkage group generated gap-free complete sequences for all 6 chromosomes.

We performed additional analysis to test the performance of our assembly using the Hi-C dataset generated by the same group. By aligning the Hi-C data against our assembly using BWA-MEM, then detecting the potential mis-assemblies using Salsa [17], no any mis-assemblies were reported. Salsa also supported the merging of two linkage groups suggested by the telomeric motif based analyses in our assembly. For comparison, we also applied Salsa with Hi-C data to the best preliminary assembly from Pacbio-Flye for comparison. This Flye/Hi-C assembly contains eight scaffolds and 14 unanchored contigs after excluding those from sample contamination. We observed 13 spanned gaps in Flye/Hi-C assembly with the two largest gaps is 4.95 Mb and 1.59 Mb (**Fig 3**).

We polished our assembly using Pacbio and Illumina short reads and compared it to the published assembly. The evaluation from Busco 3.0.0 indicated our assembly successfully assembled two more genes. The alignment of Illumina short reads against our assembly shows better alignment rate and much less variants (**Table 1** and **Supplementary Fig. 3**).

Arabidopsis *thaliana* genome assembly:

We assembled *Arabidopsis thaliana* accession KBS-Mac-74 by combinatory analysis of two publicly available datasets using GALA, one is from Pacbio with a coverage 58X, and the other is from Nanopore with 28X coverage [18]. We used both datasets separately to generate preliminary assemblies. Both raw reads and corrected reads by Canu and Mecat/Necat (**Supplementary Fig. 4**) were used as input for Canu, Flye, Miniasm+Minimap2/Minimap, Wtdbg2, Mecat/Necat assemblers (**Supplementary Table 2** and Online Methods).

GALA analysis on the raw reads and 13 of the preliminary assemblies reported a number of mis-assemblies for each preliminary assembly, ranging from 1-18. GALA modelled the input into 14 independent linkage groups.

Among them, one from the mitochondrial genome, one from the Chloroplast genome and three are continuous fragments (2.1 Kb, 7.6 kb and 18.7 kb irrespectively). The remaining 10 linkage groups represent a chromosome arm each. Previous studies indicated that Pacbio and Nanopore platform seldomly sequence through centromeric regions in *A. thaliana* [18, 19], so we only aim to assembly each chromosome arms in this study. Our algorithm assembled each of linkage group into a continuous sequence. We were able to identify telomer motif in 8 assembled chromosome-arm sequences. Interestingly, only 2 telomer motifs have been observed in the Col-0 reference genome, indicating possible missing sequences in the reference genome.

Our final assembly contains 10 complete chromosome-arm sequences and 3 unanchored contigs. We also further analysed the three unanchored contigs and all of them were mapped to the centromeric region in in reference genome. The successful assembly of *A. thaliana* genome by combining Pacbio and Nanopore sequencing data indicated that GALA provide a sound framework for integrated assembly of multiple sequencing platforms.

Assembly of the human genome

We assembled the reference human genome using a publicly available Nanopore sequencing dataset [20]. In the original assembly from the publication, there are numerous gaps. We tested whether GALA can be applied to improve the existing assembly. The raw reads were partitioned into independent linkage groups by aligning to the reference genome. GALA then assembled the linkage group representing Chromosome 19 by merging results from Flye, Canu and Minasm/Minmap. We successfully assembled 55 Mbp of sequence into a single chromosome free of gaps. Further analysis indicated a successful assembly of the telomeric region at the end of positive strand, which is lost in current reference genome GRCh38.p13. The successful assembly of Chromosome 19 indicates that GALA algorithm can efficiently applied to fill gap and enhance an existing assembly.

We also tested GALA on high-fidelity (HiFi) long reads generated by Pacbio using the circular consensus sequencing (CCS) mode. A publicly available Pacbio HiFi sequencing dataset [21] was used. The raw reads were partitioned into independent linkage groups by aligning to the reference genome. Each linkage group is then assembled with GALA. In the reference genome GRCh38.p13 and also the original de novo assembly, Chromosome 16 has several gaps and un-anchored contigs. GALA successfully assembled the chromosome into single contig free of gaps of length 88 Mbp. The assembled Chromosome 16 have a telomeric region at the both ends, which is missing in GRCh38.p13. The successful assembly of Chromosome 16 indicates that GALA algorithm can efficiently applied to Pacbio HiFi data.

Effect of chromosome-by-chromosome assembly on assembly graph

We investigated how GALA achieve complete assembly while existing tools had failed. We postulated that the chromosome-by-chromosome assembly strategy had played a role, thus compared our assembly of *C. elegans* to this from Minasm/Minimap. We noticed a much simpler computer graph in chromosome-by-chromosome assembly. In terms of the number of overlaps between reads (graph edges) in the assembly of *C. elegans*, whole genome assembly generate 190,936,281 edges while chromosome-by-chromosome assembly only generated 138,678,842 edges (27.37% less). A comparison between chromosome-by chromosome assembly and the whole genome assembly is demonstrated in **Fig. 4**.

The advantage of chromosome-by chromosome assembly is more obvious in the regions which contain the highly similar sequence, but still with unique markers, for example regions with ancient transposons. In addition, the regions which contain repetitive sequences but expanded by long reads usually allow a complete assembly by overlap-graph based algorithms, such as Canu or Mecat, but turn out to be too challenging to *de Brujin* graph based algorithms, such as Wtdbg2. In both scenarios, the GALA can obtain the best results.

Discussion

We reported GALA, a scalable method for gap-free chromosome-scale assembly. In many settings, GALA improved the contiguity and completeness of genome assembly on existing state-of-art solutions, which previously required separate datasets and analyses. The approach is highly modular: mis-assembly detection module (MDM) should be applicable for error correction regardless of the specific algorithm used for assembly; and the contig-clustering module (CCM) could have wide application in generating consensus assembly from multiple sequences. Although we have focused on de novo assembly, GALA should work equally well in other applications.

GALA shows big advantage over the existing assembly workflows. As a fact, we generated chromosome-scale gap-free assemblies in most of our experiments. In certain circumstances, we failed to assemble a challenging region, such as centromere in *A. thaliana*. The failure is mainly due to the absence of raw sequencing data in that region and the failure occurred in most of other tools too [19, 22-24]. The strength of GALA comes from the multiple-layer computer graph model, which is highly flexible in incorporating heterogenous information. As proved in the assembly of *C. elegans* and *A. thaliana* genome, combinatory analyses of Pacbio and Nanopore sequencing data were achieved. We believe studies incorporating other sources of information, such as Bionano data, genetic map or even genome synteny information, to create independent linkage groups, will yield new insights.

The performance of our algorithm also reflects the advantage of chromosome-by-chromosome assembly. The concept of chromosome-by-chromosome assembly has been justified through physical separation of chromosomes in wheat genome assembly with expensive devices and time-consuming procedures [25, 26]. GALA provide a computational solution to achieve the same effect. The same strategy can also be applied to existing computational tools to refine an existing assembly.

Acknowledgements: We thank M. Tsiantis and R. Mott for the comments on the work. M. A. is supported by International Max Planck Research Schools program. This work was supported by the Max Planck Society core grants.

Code Availability: The source code of GALA is available from github at https://github.com/ganlab/gala.

Reference:

1. Cao, M.D., et al., *Scaffolding and completing genome assemblies in real-time with nanopore sequencing.* Nat Commun, 2017. **8**: p. 14515.
2. Li, C., et al., *Genome Sequencing and Assembly by Long Reads in Plants.* Genes (Basel), 2017. **9**(1).
3. Xiao, C.L., et al., *MECAT: fast mapping, error correction, and de novo assembly for single-molecule sequencing reads.* Nat Methods, 2017. **14**(11): p. 1072-1074.

4. Koren, S., et al., *Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation.* Genome Res, 2017. **27**(5): p. 722-736.
5. Kolmogorov, M., et al., *Assembly of long, error-prone reads using repeat graphs.* Nat Biotechnol, 2019. **37**(5): p. 540-546.
6. Ellison, C.E. and W. Cao, *Nanopore sequencing and Hi-C scaffolding provide insight into the evolutionary dynamics of transposable elements and piRNA production in wild strains of Drosophila melanogaster.* Nucleic Acids Res, 2020. **48**(1): p. 290-303.
7. Jiao, W.B., et al., *Improving and correcting the contiguity of long-read genome assemblies of three plant species using optical mapping and chromosome conformation capture data.* Genome Res, 2017. **27**(5): p. 778-786.
8. Jibran, R., et al., *Chromosome-scale scaffolding of the black raspberry (Rubus occidentalis L.) genome based on chromatin interaction data.* Hortic Res, 2018. **5**: p. 8.
9. Stankova, H., et al., *BioNano genome mapping of individual chromosomes supports physical mapping and sequence assembly in complex plant genomes.* Plant Biotechnol J, 2016. **14**(7): p. 1523-31.
10. Song, B., R. Mott, and X. Gan, *Recovery of novel association loci in Arabidopsis thaliana and Drosophila melanogaster through leveraging INDELs association and integrated burden test.* PLoS Genet, 2018. **14**(10): p. e1007699.
11. Chen, X. and M. Tompa, *Comparative assessment of methods for aligning multiple genome sequences.* Nat Biotechnol, 2010. **28**(6): p. 567-72.
12. BSong B, S.Q., Wang H, Pei H, Gan X and Wang F, *Complement Genome Annotation Lift Over Using a Weighted Sequence Alignment Strategy.* Front. Genet, 2019. **10**.
13. Bickhart, D.M. and G.E. Liu, *The challenges and importance of structural variation detection in livestock.* Front Genet, 2014. **5**: p. 37.
14. Denton, J.F., et al., *Extensive error in the number of genes inferred from draft genome assemblies.* PLoS Comput Biol, 2014. **10**(12): p. e1003998.
15. Zhang, X., J. Goodsell, and R.B. Norgren, Jr., *Limitations of the rhesus macaque draft genome assembly and annotation.* BMC Genomics, 2012. **13**: p. 206.
16. Yoshimura, J., et al., *Recompleting the Caenorhabditis elegans genome.* Genome Res, 2019. **29**(6): p. 1009-1022.
17. Ghurye, J., et al., *Integrating Hi-C links with assembly graphs for chromosome-scale assembly.* PLoS Comput Biol, 2019. **15**(8): p. e1007273.
18. Michael, T.P., et al., *High contiguity Arabidopsis thaliana genome assembly with a single nanopore flow cell.* Nat Commun, 2018. **9**(1): p. 541.
19. Pucker, B., et al., *A chromosome-level sequence assembly reveals the structure of the Arabidopsis thaliana Nd-1 genome and its gene set.* PLoS One, 2019. **14**(5): p. e0216233.
20. Miga, K.H., et al., *Telomere-to-telomere assembly of a complete human X chromosome.* 2019: p. 735928.
21. Nurk, S., et al., *HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads.* bioRxiv, 2020.
22. Arabidopsis Genome, I., *Analysis of the genome sequence of the flowering plant Arabidopsis thaliana.* Nature, 2000. **408**(6814): p. 796-815.
23. Zapata, L., et al., *Chromosome-level assembly of Arabidopsis thaliana Ler reveals the extent of translocation and inversion polymorphisms.* Proc Natl Acad Sci U S A, 2016. **113**(28): p. E4052-60.
24. Jiao, W.B. and K. Schneeberger, *Chromosome-level assemblies of multiple Arabidopsis genomes reveal hotspots of rearrangements with altered evolutionary dynamics.* Nat Commun, 2020. **11**(1): p. 989.
25. Paux, E., et al., *A physical map of the 1-gigabase bread wheat chromosome 3B.* Science, 2008. **322**(5898): p. 101-4.
26. Holusova, K., et al., *Physical Map of the Short Arm of Bread Wheat Chromosome 3D.* Plant Genome, 2017. **10**(2).

Table 1 The assembly performance evaluation with Illumina short reads.

	REF(N2)	VC2010	Our assembly
Mapped reads	130,604,410	130,639,345	130,652,108
Unmapped reads	4,568,540	4,533,605	4,520,842
Variants	17,385	14,839	14,169
SNPs	16,179	14,167	13,701
Deletions	412	282	124
Insertions	794	390	344
Indels	1,206	672	468

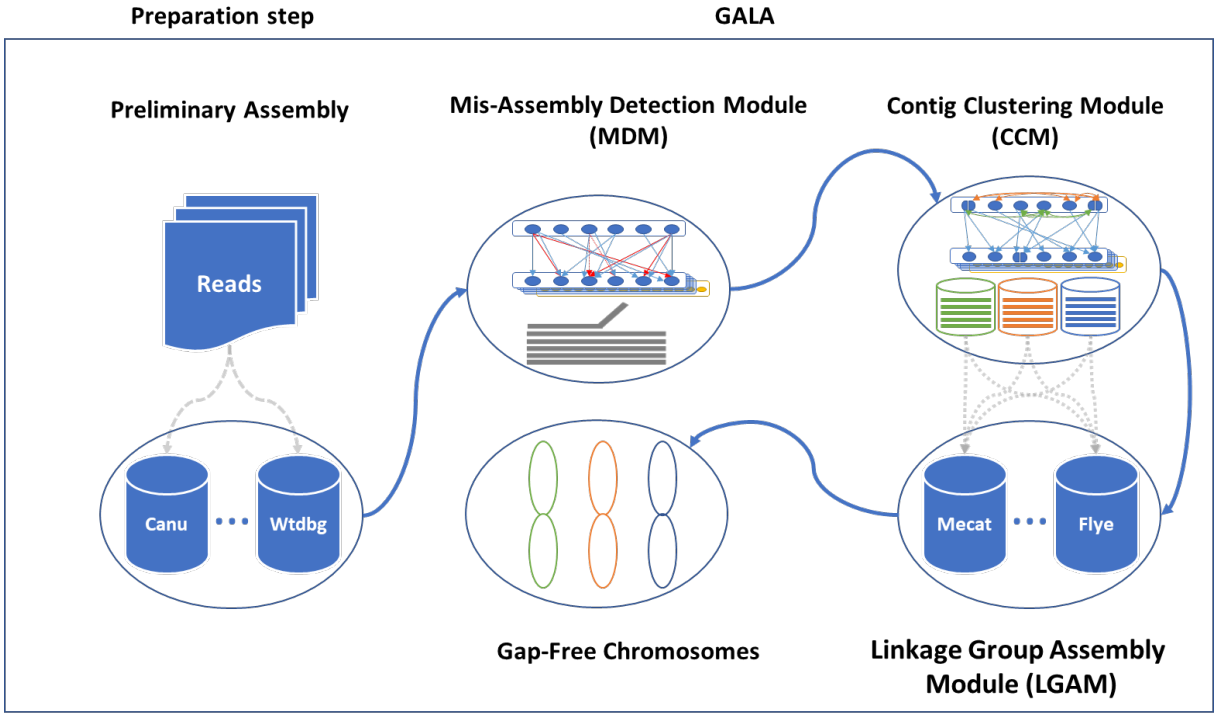

Figure 1. Overview of GALA. After *de novo* assembling with various tools, preliminary assemblies and raw reads are encoded into a multiple-layer computer graph. (ii) Mis-assemblies are identified with MDM by browsing through the inter-layer information. (III) The split nodes are clustered into multiple linage groups by CCM. (iv) Each linkage group is assembled independently to achieve (v) the final gap-free assembly.

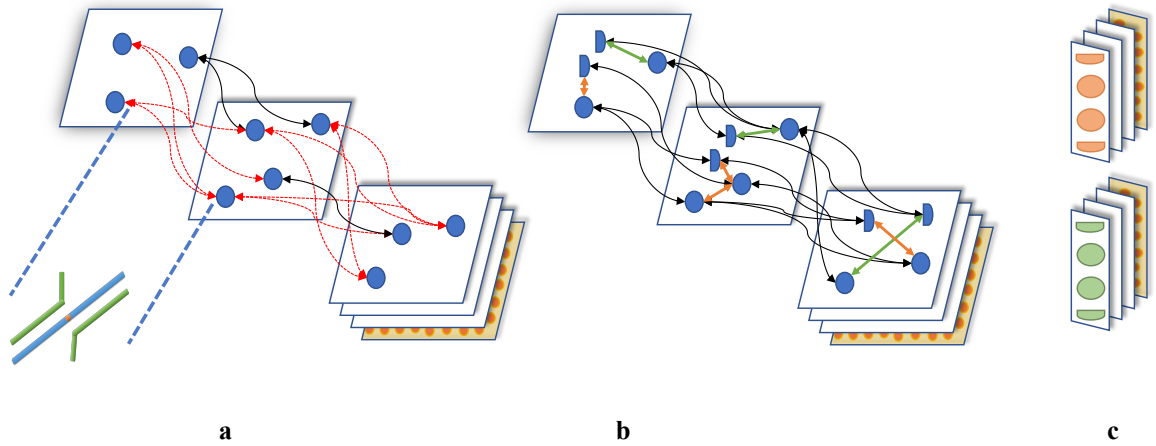

Figure 2. Illustration of multiple-layer computer graph in GALA. (a) The preliminary assemblies and raw reads are aligned against each other and encoded into a multiple-layer graph. Conflicted alignments are indicated in red. (b) The conflicts are removed iteratively by splitting the nodes involved and new edges are assigned accordingly. (c) Nodes are clustered into different linkage groups.

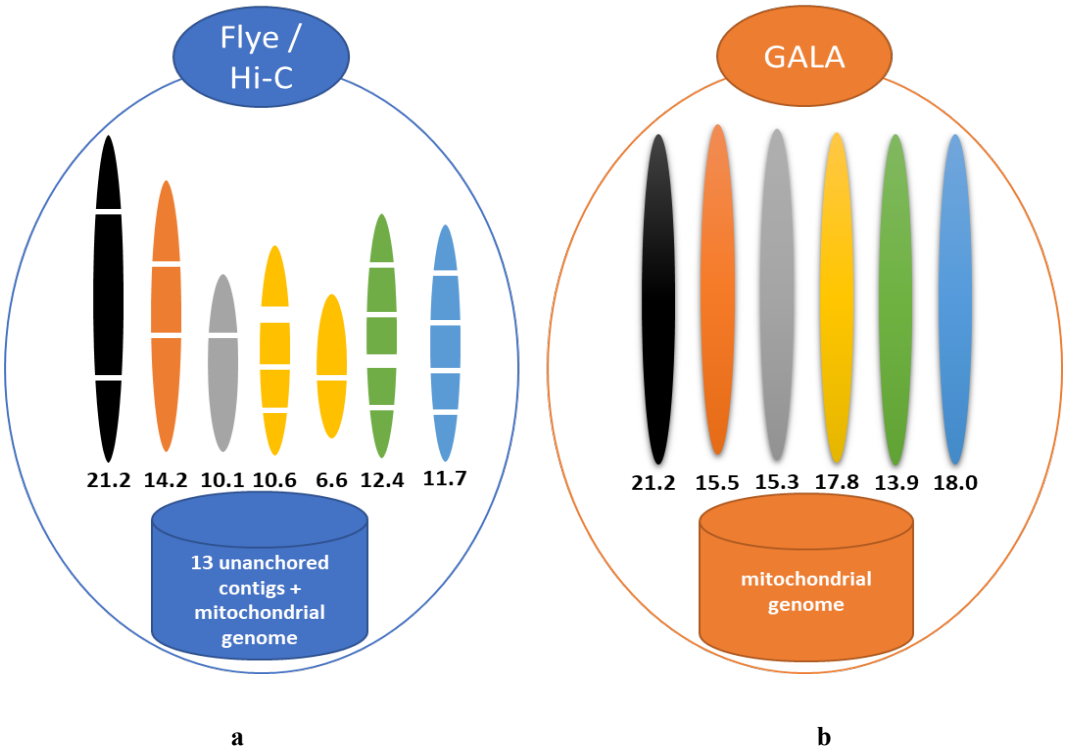

Figure 3. Comparison of Flye assembly with Hi-C scaffolding and GALA assembly of long reads of the *C. elegans* genome. (a) The former contains numerous gaps and 13 unanchored contigs in the assembly. (b) GALA produces gap-free assembly for each chromosome. Note this is not a fair comparison as GALA did not use Hi-C data in this assembly.

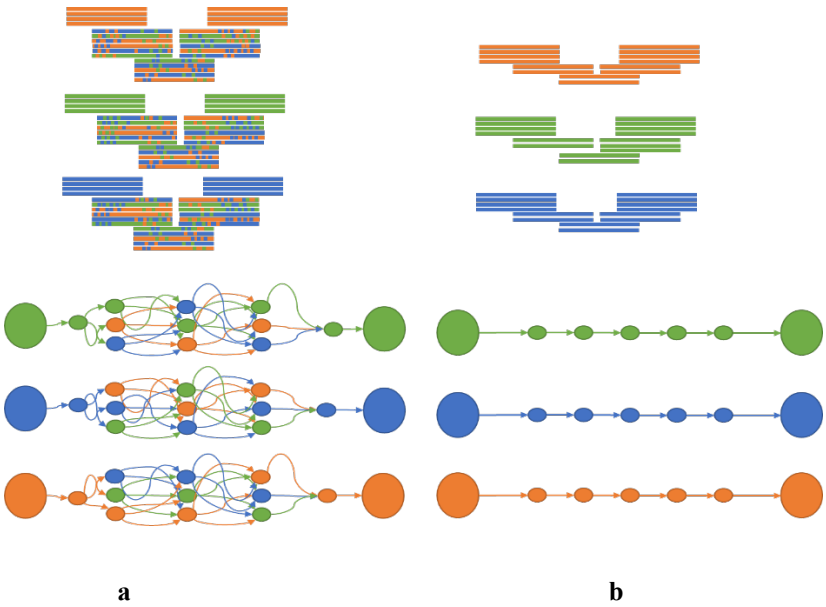

Figure 4. Comparison of the overlap graphs used by Minasm during assembling a region in the *C. elegans* genomes when chromosome-by-chromosome mode is applied or not. (a) In the whole-genome assembly mode, the overlap graph used by Minasm contains numerous edges, with extra effort needed to collapse edges. (b) The chromosome-by-chromosome assembly allows a linear overlap graph for Minasm in the same region

.

Online Methods:

Reciprocal alignment between preliminary assemblies:

Minimap2 [27] (-x asm5) was used to map preliminary assemblies against each other. The raw and corrected reads were aligned to an assembly using BWA-MEM [28] with default parameters.

Mis-assembly detection module (MDM):

We built a multiple-layer graph by encoding the information from various preliminary assemblies D_n. Each preliminary assembly D_x represented a layer consists of a set of nodes C_m, each node representing an assembled contigs. The starting point of the MDM was reciprocal alignment of D_n, which produced $(n * (n-1))$ PAF mapping files. We filtered the mapping files based on four criteria; I) Mapping quality (default 20), II) Contig length (default 5kb), III) Alignment block size (default 5kb), and IV) Sequence similarity percentage (default 70%). All parameters are tunable in GALA. A simple merging procedure was performed to merge nodes within the same layer if they are overwhelmingly support by the reciprocal alignment to lowered down the burden to computational resources.

We then linked the nodes between different layers by retrieving the information from reciprocal alignment. Assuming that a contig in node C in query layer D_x , denoted as C^{D_x} , is mapped to a set of nodes in layer $(D_{1..n})$, denoted as $\{C_1^{D_1}, \dots, C_i^{D_1}, \dots C_i^{D_n}\}$, A mis-assembly at region M occurs if and only if contig $C_i^{D_k} \in \{C_1^{D_1}, \dots, C_i^{D_1}, \dots C_i^{D_n}\}$ is partially mapped to C^{D_x}.

Let L be the length of contig C^{D_x} , and N_A be the number of contigs partially mapped to M, and N_B the number of contigs with complete alignment, and N_S be the number of contigs starting or ending at M. We considered M as a genuine mis-assembled locus if:

$$N_A \geq (n/2) \quad (1)$$

$$N_B = 0 \ \& \ N_A \geq 2 \quad (2)$$

$$N_A \geq 2 \ \& \ \left(\frac{N_B}{N_A}\right) \leq 0.5 \quad (3)$$

$$N_S > 0 \ \& \left(\frac{N_B - N_S}{N_A}\right) \leq 0.6 \quad (4)$$

If a mis-assembly is identified, the node is split into two from the region M. This procedure iterates until the whole graph is free of mis-assemblies.

Contigs clustering module CCM:

The multiple-layer computer graph output by MDM was expanded by adding into an extra layer representing the raw reads. So far, within each layer, nodes were separate from each other and no intra-layer edge existed. We first built intra-layer edges by browsing through the existing the cross-layer edges. For node C^{D_x} and its linked cross-layer node $\{C_0^{D_1}, \dots, C_i^{D_1}, \dots C_i^{D_n}\}$, the CCM starts by visiting all $\{C_0^{D_1}, \dots, C_i^{D_1}, \dots C_i^{D_n}\}$. An intra-layer edge was

built up if more than one node in the same layer was linked to the same cross-layer node. CCM pooled all connected nodes into a linkage group.

In previous MDM, only contigs with length larger than a certain threshold value, 5kb at default, were encoded into our computer graph. Those with smaller size were not suggested for mis-assembly detection to avoid the situation where unique sequences could be dumped by accident, we kept them and classified them into existing linkage groups for further analysis.

If Hi-C information or a genetic map is available, extra links can be created between internal nodes. This would essentially lead to the merge of multiple independent linkage groups. CCM could also be performed in an iterative mode together with the linkage group assembly module as demonstrated in the examples below.

Linkage group assembly module (LGAM):

The reads within a linkage group were assembled using different assembly tools, e.g., Mecat, Flye, and Miniasm. We mapped the newly assembled contigs against one of the chosen preliminary assemblies recorded in the same linkage group (free of mis-assembly) and removed those absent or those with multiple copies. A simplified version of overlap-graph-based merging algorithm was used to merge two contigs if necessary.

***Caenorhabditis elegans* assembly:**

Pacbio dataset were generated in three different runs and there was a clear batch effect with sequence quality and the amount of data between runs. We used all runs (290X in coverage), and the biggest run alone (240X in coverage) for the assembly. For the Pacbio dataset we used reads-correcting-and-trimming module from Canu 1.8 [4] to correct the raw reads. Preliminary assemblies were generate using Canu 1.8, Flye 2.4 [5], Miniasm 0.3-r179 [29] +Minimap2/Minimap, Wtdbg2 [30], Mecat/Necat [3] from Pacbio corrected and raw reads and nanopore raw reads. By comparing the summary statistics of preliminary assemblies (**Supplementary Table 1**), 10 preliminary assemblies are chosen for GALA.

GALA modelled the preliminary assemblies and raw reads into 14 independent linkage groups, 7 of them were short continuous contigs and the others represented individual chromosomes or chromosome arms. Further analyses by blasting the 7 short contigs in NCBI database indicated that 3 of them were from *E. coli* contamination and one from *C. elegans* mitochondrial genome, and thus excluded from the subsequent analyses. The remaining 3 short contigs, 2 of them can be reliably put into the 7 previously created linkage groups with the help of the assembly of Nanopore reads with Miniasm/Minimap2 (**Supplementary Fig. 1**).

We assembled 7 linkage groups with LGAM, each into a continuous sequence. Among the 7 continuous sequences and 1 unanchored short contig, 4 of them show clear telomere repetitive motif at both terminals, indicating they are complete assemblies of single chromosomes. One chromosome-scale sequence show telomere repetitive motif at one end, and its missing telomeric repetitive motif can be identified in the unanchored short contig, indicating they both should be merged as a single linkage group. The remaining two had telomer repetitive motif at either side and their sizes clearly indicated they were two arms from a single chromosome. We thus pooled their linkage groups together. At last, we re-assembled the two newly created linkage groups, and we were able to create complete sequences for the two chromosomes with telomeric repetitive motif at both terminals. Further analyses

indicated the split of this single chromosome into two linkage groups in the first run was mainly due to several transposon sequences.

***Caenorhabditis elegans* genome assembly polishing and quality control:**

For a fair comparison, we polished our assembly with Pacbio and Illumina sequencing data. We ran racon [31] with corrected Pacbio reads first. The assembly were then polished using quiver 2.3.2 [32] with Pbmm2 1.1.0 as aligner. Finally, we ran pilon 1.23 [33] using Illumina sequencing data to correct short errors, especially those in homomorphic regions.

We evaluated the completeness of our assembly with Busco 3.0.0, and compared to the published assembly and the reference genome (**Supplementary Fig. 5**). We also aligned the Illumina short reads to our assembly using BWA-MEM and called the variants using BCFtools 1.9. We collected the statistics and compared them to those from the published assembly as a benchmark for the precision of the assembly.

Arabidopsis *thaliana* assembly:

Pacbio and Nanopore raw reads datasets were corrected using Canu 1.8 correcting and trimming module. We noticed that Mecat/Necat had shown good performance of reads correction in this dataset, so we additionally generated corrected Pacbio reads with Mecat with the minimal length of reads parameter as 5000 or 3000 separately, and corrected Nanopore reads with Necat with the minimal length of reads parameter 500. The preliminary assembly were generate using Canu, Flye, Miniasm+Minimap2/Minimap, Wtdbg2, Mecat/Necat from corrected and raw reads in both datasets. 13 preliminary assemblies are chosen for GALA with default parameters (**Supplementary Table 2**). In LGAM, we used Wtdbg2, Flye, Miniasm+Minimap2/Minimap, and Mecat/Necat tools.

References:

27. Li, H., *Minimap2: pairwise alignment for nucleotide sequences.* Bioinformatics, 2018. **34**(18): p. 3094-3100.
28. Li, H. and R. Durbin, *Fast and accurate short read alignment with Burrows-Wheeler transform.* Bioinformatics, 2009. **25**(14): p. 1754-60.
29. Li, H., *Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences.* Bioinformatics, 2016. **32**(14): p. 2103-10.
30. Ruan, J. and H. Li, *Fast and accurate long-read assembly with wtdbg2.* 2019: p. 530972.
31. Vaser, R., et al., *Fast and accurate de novo genome assembly from long uncorrected reads.* Genome Res, 2017. **27**(5): p. 737-746.
32. Chin, C.S., et al., *Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data.* Nat Methods, 2013. **10**(6): p. 563-9.
33. Walker, B.J., et al., *Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement.* PLoS One, 2014. **9**(11): p. e112963.