GALA: a computational framework for de novo chromosome-by-chromosome assembly with long reads

Mohamed Awad
Max Planck Institute for Plant Breeding Research
Xiangchao Gan (gan@mpipz.mpg.de)
Max Planck Institute for Plant Breeding Research https://orcid.org/0000-0001-6398-5191

Article

Keywords: genome assembly, multi-layer computer graph, long reads, Pacbio sequencing, Nanopore sequencing, gap-free assembly, T2T assembly

Posted Date: March 2nd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1339386/v1

GALA: a computational framework for *de novo* chromosome-by-chromosome assembly with long reads

Mohamed Awad[1] & Xiangchao Gan[1,2*]

[1]*Max Planck Institute for Plant Breeding Research, Department of Comparative Development and Genetics, Carl-von-Linné-Weg 10, 50829 Köln, Germany*

[2]*State Key Laboratory for Crop Genetics and Germplasm Enhancement, Academy for Advanced Interdisciplinary Studies, Nanjing Agricultural University, Nanjing 210095, China*

[*] Author for correspondence: e-mail address (gan@mpipz.mpg.de)

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 for long-read platforms. Here we report on GALA (**Ga**p-free **l**ong-read **a**ssembler), a computational framework for *de novo* chromosome-by-chromosome assembly implemented through a multi-layer graph that identifies discordances within preliminary assemblies and partitions the data into chromosome-scale linkage groups. The subsequent independent assembly of each linkage group generates a gap-free assembly likely free from the mis-assembly errors which usually hamper existing workflows. This flexible framework also allows us to integrate data from various technologies, such as Hi-C, genetic maps, and even motif analyses to generate gap-free chromosome-scale assemblies. As proof of principle we *de novo* assembled the *C. elegans* genome using combined Pacbio and Nanopore sequencing data and a *rice* cultivar genome using Nanopore sequencing data from publicly available datasets. We also demonstrated the new method's applicability with a gap-free assembly of the human genome using Pacbio high-fidelity (HiFi) long reads. Thus, our method enables straightforward assembly of genomes with multiple data sources and overcomes barriers that at present restrict the application of *de novo* genome assembly technology.

Keywords

genome assembly, multi-layer computer graph, long reads, Pacbio sequencing, Nanopore sequencing, gap-free assembly, T2T assembly

Introduction

De novo genome assembly has wide applications in plant, animal, and human genetics. However, it is still very challenging for long-read platforms, such as Nanopore and Pacbio, to provide chromosome-scale sequences [1, 2]. To date, numerous *de novo* assembly tools have been developed to obtain longer and more accurate representative sequences from raw sequencing data [3-5]. In most studies, however, assemblies by these tools comprise hundreds

or even thousands of contigs. To produce chromosome-scale assembly, various information sources, such as Hi-C, genetic maps, or a reference genome, 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, is also plagued with mis-assemblies, as reported in [8].

Gaps and mis-assemblies in a genome assembly can seriously undermine genomic studies. For example, a lot of sequence alignment tools have much lower performances when query sequences contain gaps [9, 10]. In intraspecific genome comparisons, large gaps not only significantly increase the possibility of failure to detect long structure variants, but also produce inaccurate results of gene annotation [11, 12]. Moreover, gaps and mis-assemblies have been reported to account for a large number of gene model errors in existing genome assembly studies [13, 14].

In this study, we report on GALA (**Ga**p-free **l**ong-read **a**ssembler), a scalable chromosome-by-chromosome assembly method implemented through a multi-layer computer graph. (**Fig. 1**). GALA separates two steps: firstly, it identifies multiple linkage groups in the genome, each representing a single chromosome (sometimes a chromosome arm) and it also describes chromosome structure with raw reads and assembled contigs from multiple *de novo* assembly tools; secondly, it assembles each linkage group by integrating results from multiple assembly tools and inference from raw reads. Moreover, our method can also exploit the information derived from Hi-C data to obtain chromosome-scale linkage groups in studies even with a complicated genome structure or those with low sequencing quality. Of note is that our method can be easily extended to incorporate other sources of information such as genetic maps or even a reference genome. Here, we show the utility of GALA by gap-free and chromosome-scale assemblies of Pacbio or Nanopore sequencing data from two publicly available datasets for which the original assembly contains large gaps and a number of unanchored scaffolds. Notably, our new method significantly outperforms existing algorithms in both datasets. Finally, we also demonstrate the application of our method to assemble a human genome with the help of a reference genome using Pacbio high-fidelity (HiFi) long reads.

Results

Overview of the GALA framework

GALA exploits information from multiple *de novo* assembly tools and raw reads, as well as other information sources, such as Hi-C, genetic maps, or even a reference genome, if they exist. In GALA, various *de novo* assembly tools are selected first to create preliminary assemblies. These preliminary assemblies and raw reads are then

aligned against each other. We use a multi-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 contig (or a read in the 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 is 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 is created between them (**Fig. 2**).

Depending on the sequencing quality and complexity of the genome structure, existing assembly 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). This module works by estimating the probability of mis-assemblies based on the contradictory cross-layer edges, and splitting those nodes containing highly likely mis-assemblies to resolve discordances in the computer graph (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 (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 to essentially perform 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 suggested methods. Interestingly, we found that chromosome-by-chromosome assembly always provides better performance, especially for the repetitive fragments in terms of contiguity. In contrast, the improvement of read correction with chromosome-by-chromosome analysis is negligible. We also tested GALA in 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, thereby highlighting the distinct 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 the Pacbio platform with a 290X coverage along with an extra 32X coverage of Nanopore sequences [15]. As no current 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, Mecat2/Necat, Miniasm, and Wtdbg2 (Methods). Among all our preliminary assemblies, the one produced by Pacbio-Flye showed the smallest number of contigs, with 41 contigs for 102 Mbp of overall sequences.

We applied GALA to the raw reads and the preliminary assemblies. The numbers of discordances in each preliminary assembly derived by the MDM algorithm ranged from 0 to 19. After resolving the discordances through the node-splitting operation, GALA modelled the input into 14 independent linkage groups. Seven of them contain a very small amount of sequencing data and apparently come from short continuous contigs. Among them, four contigs are from bacterial contamination or organelle DNA and two of them can be pooled into seven large linkage groups using Nanopore sequencing data. The remaining one contains a telomeric repetitive motif. We then performed telomeric motif analyses for the seven large linkage groups. Four of them contain complete chromosomes. Two groups contain the telomeric repetitive motif at one end and apparently come from two arms of the same chromosome and one group misses the telomeric repetitive motif at one end. We thus were able to merge 14 linkage groups further into six ones (**Supplementary Fig. 2** and Methods). Of note is that the integrative assembly of each linkage group generated gap-free T2T complete sequences for all six chromosomes.

We polished our assembly using Pacbio and Illumina short reads and then compared it to the published VC2010 assembly and the N2 reference genome. Note that the VC2010 sample is derived from the N2 reference sample and their assemblies are supposed to be very close. The evaluation from Busco 3.0.0 indicated that our assembly successfully assembled two more genes. Furthermore, the alignment of Illumina short reads against our assembly also reveals a better alignment rate as well as fewer variants (**Table 1** and **Supplementary Fig. 3**).

We performed additional analyses to test the performance of our assembly using the Hi-C dataset generated by the same research group. No discordances were revealed by aligning the Hi-C data against our assembly using BWA-MEM, then detecting the discordances using Salsa [16]. Salsa also supported the merging of two linkage groups suggested by the telomeric motif analyses in our assembly. For comparison, we also applied Salsa with Hi-C data to the best preliminary assembly from Flye with Pacbio data. This Flye/Hi-C assembly contains seven scaffolds and 14 unanchored contigs after excluding those from sample contamination. We observed 17 spanned gaps in the Flye/Hi-C assembly, with the two largest gaps being 495 Kbp and 159 Kbp (**Fig 3**). Furthermore, we aligned the raw Pacbio reads to different assemblies and examined the distribution of the depth-of-coverage across the genome

(**Supplementary Fig. 4**). Apart from being free of gaps, the GALA assembly shows comparable performance to the VC2010 assembly in terms of assembly error in repetitive regions.

Oryza sativa genome assembly

We assembled *Oryza sativa* circum-basmati landrace Dom Sufid (sadri) using a publicly available dataset with GALA. The dataset contains 42.7 GB Nanopore sequencing data, equivalent to 56X coverage of the rice genome with 12 chromosomes [17]. Firstly, we used the Canu self-correction and trimming module to correct the raw reads, and produced a preliminary assembly with corrected reads. Flye, Miniasm and Wtdbg2 were used to generate six preliminary assemblies using raw and corrected reads respectively. In addition, Necat produced a preliminary assembly from the raw reads.

GALA analyses on the preliminary assemblies highlighted a number of discordances in each preliminary assembly, which ranged from 0-2. The input was rectified and separated into 16 independent linkage groups. Among them, one was from the mitochondrial genome and one from the chloroplast genome. The remaining 14 linkage groups represent ten chromosomes and four chromosome-arms. For these four chromosome-arm scale linkage groups which represent two chromosomes (Chr2 and Chr11), there are only three possible combinations. We run linkage group assembly module (LGAM) on each of combination, only one produced continuous telomere-to-telomere pseudomolecules for both chromosomes. The linkage group assembly on the 10 chromosome-scale linkage groups generated 10 gap-free complete contigs. In total, our final assembly produced 12 gap-free complete chromosome-sequences.

Interestingly, our assembly showed an inversion on Chr6 compared to the reference genome of Nipponbare. This inversion was reported in the circum basmati genome study but the previous Dom Sufid assembly cannot produce a gap-free complete sequence for this region [17]. The *de novo* assembly of *Oryza sativa* using GALA significantly improved the previous Dom Sufid assembly which was generated through a reference guided scaffolding method [17] (**Supplementary Fig. 5**).

Human genome assembly

We next assembled a human genome using high-fidelity (HiFi) long reads generated by Pacbio using the circular consensus sequencing (CCS) mode [18]. For simplicity, we used the published preliminary *de novo* assembly by HiCanu [18] (3.28GB overall) and the current human reference genome GRCh38.p13 as input for GALA. The raw reads and the input HiCanu preliminary assembly are partitioned by the contig-clustering module (CCM) of GALA.

Here, the CCM only serves as a raw-read separation tool to enable subsequent chromosome-by-chromosome *de novo* assembly. Both information from the input reference genome, which could be from a close relative thus different from the query genome, and information from the preliminary assembly of the query genome, were used for raw-read separation. GALA revealed 23 independent linkage groups and assembled them one-by-one. Interestingly, when assembling linkage groups, we used two softwares, namely HiCanu and Hifiasm, and they provided significantly different assemblies in terms of the length of sequences. Taking Chromosome 17 as an example, HiCanu assembled its linkage group into three contigs with a total length of 83.2 Mb (40 Mb, 24.7 Mb, and 18.5 Mb). In contrast, Hifiasm produced one single telomere-to-telomere contig of a total length of 82.1 Mb. To resolve this, we aligned the raw HiFi reads to both assemblies and examined the distribution of the depth-of-coverage. We selected the better genome assembly by taking into account the number of assembly errors as well as gaps. The comparison between our GALA assembly and the published assembly can be found in **Fig.4a** and **Supplementary Fig. 6**. Overall, our assembly comprised of 38 continuous contigs, including seven telomere-to-telomere gap-free pseudomolecular sequences (3, 7, 10, 11, 16, 17, and 20), four near-complete chromosomes (5, 8, 12, and 19) each with a small telomeric fragment unanchored, and four chromosomes (4, 6 9, and 18) with gapped centromeric regions. Note that we only assembled the long arms of the five acrocentric chromosomes (13, 14, 15, 21 and 22) since the sequencing and assembly of their *p* arms are too challenging as they are almost all missing in both the reference genome and the published assembly.

Our human genome assembly is depicted chromosome-by-chromosome in Fig. 4b. Here, two chromosomes are of key interest: chromosome 11 and chromosome X. In the reference genome GRCh38.p13 and also the published HiCanu assembly, Chromosome 11 has several gaps and unanchored contigs. Interestingly, it is considered as one of the chromosomes with the highest density of genes linked with genetic diseases [19]. GALA successfully assembled this chromosome into a single contig free of gaps of a total length of 134.9 Mbp (**Supplementary Fig. 6**). The assembled chromosome 11 has two telomeric regions at both ends; however, one of them is missing in GRCh38.p13. The second example is chromosome X, whose assembly is regarded as highly challenging and extra effort has been devoted to this in a recent paper [20]. Our assembly only contains two short gaps (about 0.75Kbp and 1.8Kbp) compared to the published one. The successful assembly of the human genome indicates that GALA can efficiently be applied to Pacbio HiFi data.

In the above assembly of CHM13 by GALA, the reference genome was used to help separate raw-read into linkage groups. One might wonder whether this would lead to a vulnerability that plagues traditional *reference-guided* assemblies or scaffolding. It has been reported that traditional *reference-guided* assemblies suffer from short-

length assembly errors and mis-scaffolding because of reference biases and chromosomal rearrangements among different strains and cell lines, as well as errors of sequence alignment [21-23]. In addition, *reference-guided* assembly leads to missing sequences in highly divergent regions [22]. Fortunately, GALA can avoid both problems. Firstly, GALA only uses the reference genome to cluster contigs from the preliminary assembly and raw reads, so in this respect the reference functions more like a genetic map, and is largely insensitive to sequence variation between the query genome and the reference. Moreover, the subsequent *de novo* assembly of linkage groups prevents assembly errors and mis-scaffolds. For example, if raw reads are mistakenly placed into the same linkage group, this leads to assembly fragmentation rather than other types of errors. Secondly, GALA's linkage groups contain contigs from the preliminary assembly, so unique and highly divergent regions absent from the reference would not be missed out when aligning raw reads to linkage groups. For comparison, we performed the *reference-guided* scaffolding of the HiCanu preliminary assembly using Ragoo [24] and gap-filled it using PBJelly [25]. Ragoo scaffolded ~ 12 Mbp of centromeric and pre-centromeric sequences of Chr9 to Chr4 (**Supplementary Fig. 7**) with big gaps. In contrast, GALA clustered and assembled the reads from highly similar centromeric regions and constructed two continuous contigs in the two regions.

Effect of the sequencing depth on the performance of GALA

We next investigated how the performance of GALA changes depending on the sequencing depth. We subsampled the original *C. elegans* Pacbio sequencing data using software Fastq-sample to 20X, 30X, 40X, 50X, 60X, 70X, 80X, 90X, 100X, and 150X coverage, together with Hi-C data, and performed *de novo* assembly independently. Preliminary assemblies were generated using Canu, Flye, Mecat2, Miniasm, and Wtdbg2 with raw and corrected reads. A detailed comparison between the resulting assemblies can be found in **Fig. 5** and **Supplementary Table 1**. This study revealed two interesting findings. Firstly, the gap-free *de novo* assembly is not a suitable option when the data coverage is less than 40X due to the limitation of current *de novo* assembly tools. As a consequence, GALA switches to gapped assembly for this scenario. Secondly, without Hi-C for scaffolding, Flye and GALA reach the performance curve plateau at 60X and 40X coverage, respectively, regarding the number of scaffolds and N50 of their assemblies. When Hi-C data are applied, the performance curve plateau starts from 40X for Flye and GALA (**Fig. 5a, b**). The higher coverage leads to better assembly for Flye with or without Hi-C data by lowering down the number of big gaps and mis-assemblies; however, no notable effects on N50 and the number of scaffolds are observed (**Fig. 5c**). Thus, the higher coverage of data has no notable effect on GALA assembly in general.

The performance of GALA, as well as almost all assembly software tools, changes significantly with raw read length and sequencing error. Note that the above analyses are based on the Pacbio sequencing data generated with Pacbio RSII. Consequently, the lengths of the raw reads are notably smaller and sequencing error is significantly higher than the current Pacbio Sequel II. In practice, the sequencing length distribution often varies significantly between different sequencing platforms, genome centers, and sample preparation. Therefore, it is difficult to set a straightforward threshold value for the minimum coverage of data for GALA assembly. As a rule of thumb, GALA can produce gap-free assembly from 25X coverage of Pacbio Sequel II data or Nanopore MinION data if N50 of the raw data is larger than 20 Kbp. For Pacbio HiFi, 20X coverage works well for GALA due to its low sequencing error rate.

Effect of chromosome-by-chromosome assembly on the assembly graph

We investigated why GALA achieved complete assembly while existing assembly software tools had failed. We postulated that the chromosome-by-chromosome assembly strategy had played a role, and thus, we compared our assembly of *C. elegans* to that from Miniasm. This comparison revealed a much simpler computer graph in the chromosome-by-chromosome assembly. In terms of the number of overlaps between reads (graph edges) in the assembly of *C. elegans*, the whole genome assembly generated 190,936,281 edges, whereas the chromosome-by-chromosome assembly only generated 138,678,842 edges (27.37% less). A comparison between the whole genome and the chromosome-by-chromosome assembly is depicted in **Fig. 6**.

The advantage of chromosome-by-chromosome assembly is more obvious in the regions which contain highly similar sequences, but still have unique markers, e.g., regions with ancient transposons (**Fig. 6**). In addition, the regions which contain repetitive sequences, but are expanded by long reads, usually allow for a complete assembly by overlap graph-based algorithms, such as Canu or Mecat. However, such assembly is too challenging for *de Bruijn* graph-based algorithms like Wtdbg2. In both scenarios, the GALA method can obtain superior results (**Supplementary Fig. 8**).

Discussion

Here, we have presented GALA, a scalable chromosome-by-chromosome assembly method implemented through a multi-layer computer graph. Compared to existing state-of-art assembly workflows and computational tools, GALA improved the contiguity and completeness of genome assembly significantly. Furthermore, our new method is highly modular. In detail, the 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) can be widely applied for generating consensus assembly from multiple sequences. Although we have focused on *de novo* assembly in this paper, the modules in GALA should also work equally well in other applications, such as polishing an existing assembly.

In this study, we generated chromosome-scale gap-free assemblies in our experiments. In certain circumstances, we failed to assemble challenging regions such as certain regions in the human genome. This failure is mainly due to the absence of raw sequencing data in these regions (**Supplementary Fig. 9**), and thus, also occurred in most of the other commonly used computational tools [26-29]. The strength of GALA comes from the multi-layer computer graph model, which is highly flexible in incorporating heterogenous information. As clearly demonstrated in the assembly of the *C. elegans* genomes, combinatory analyses of Pacbio and Nanopore sequencing data were achieved.

The performance of our new GALA method also reflects the advantage of chromosome-by-chromosome assembly Notably, the concept of chromosome-by-chromosome assembly was successfully tested on genome assembly in wheat, for which expensive devices and time-consuming procedures have had to be applied [30, 31]. GALA is the first method to demonstrate that this can be achieved computationally. The concept of chromosome-by-chromosome assembly can also be applied to existing computational tools to refine an existing assembly. In addition, linkage group-based assembly provides a flexible framework for GALA to support haplotype assembly in the future. This can be achieved by updating the linkage group assembly module (LGAM) to support haplotype assembly tools.

Finally, there is still room to improve GALA's assembly quality. Specifically, GALA assembly sometimes collapses long repetitive regions **(Supplementary Figs. 4** and **6)**. In this context, we compared the raw reads aligned to Chromosome X of the T2T v1.0 assembly and the reads in GALA's Chromosome X linkage group. Interestingly, only a single read aligned to the Chromosome X of the T2T v1.0 assembly is missing from GALA's Chromosome X linkage group, indicating that a bottleneck for the performance of GALA is the linkage group assembly module (LGAM) which relies on existing assembly tools. Thus, a new tool that can fully exploit the chromosome structure and depth-of-coverage, similar to centroFlye [32] but applicable to all long repetitive fragments, would be helpful in the future.

Methods

Reciprocal alignment between preliminary assemblies:

Minimap2 [33] (-x asm5) was used to map preliminary assemblies against each other. The raw and corrected reads were aligned to an assembly using BWA-MEM [34] with default parameters.

Mis-assembly detection module (MDM):

We built a multi-layer graph by encoding the information from various preliminary assemblies D_n. Each preliminary assembly D_x represented a layer that consists of a set of nodes C_m, each node representing an assembled contig. The starting point of the MDM was the reciprocal alignment of D_n, which produced $n * (n - 1)$ mapping results. We filtered the mapping results based on four criteria: (I) mapping quality (default 20), (II) contig length (default 5 Kbp), (III) alignment block length (default 5 Kbp), and (IV) sequence identity percentage (default 70%). All parameters are tunable in GALA. A simple merging procedure was performed to merge nodes within the same layer if they satisfy these four criteria to reduce the burden on 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 discordance 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} as exemplified in Fig. 2a. Two sequences are partially mapped if they cannot be merged together but their substrings, usually from one end, can be merged together according to the above four criteria.

Let *L* be the length of the contig C^{D_x}, N_A be the number of contigs partially mapped to M, 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 nodes from the region M. This procedure iterates until the whole graph is free of mis-assemblies.

Contigs clustering module (CCM):

The multi-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 cross-layer edges. For node C^{D_x} and its linked cross-layer node $\{C_0^{D_1}, \ldots, C_i^{D_1}, \ldots C_i^{D_n}\}$, CCM starts by traversing all $\{C_0^{D_1}, \ldots, C_i^{D_1}, \ldots C_i^{D_n}\}$. An intra-layer edge was built up if more than one node in the same layer was linked to the same cross-layer node. Then, CCM pooled all connected nodes into a linkage group.

In the previous step of MDM, only contigs with a length larger than a certain threshold value, 5 Kbp at default, were encoded into our computer graph. Thus, those with smaller sizes were not used for mis-assembly detection. To avoid the situation where unique sequences could be missed out 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 approach would essentially lead to the merging of multiple independent linkage groups. CCM could also be performed in an iterative mode together with the linkage group assembly module (LGAM) as demonstrated in the examples below.

Linkage group assembly module (LGAM):

The reads within a linkage group were assembled using assembly tools, e.g., Flye, Mecat, and Miniasm. In most cases, the assembly tool can produce a gap-free chromosome-scale assembly. We noticed that when a single continuous contig cannot be achieved for a linkage group, the breakpoint usually contains a very long repetitive sequence (most of the time in centromeric regions). LGAM provides a simplified version of the overlap graph-based merging algorithm to merge two contigs if necessary. However, this procedure sometimes causes collapsing of repetitive regions.

The long repetitive regions could also confuse existing assembly tools in a similar way. When assemblies from multiple software tools are significantly different in terms of length of sequence, we suggest the user to align the raw reads to different assemblies and examine the distribution of the depth-of-coverage. The user should select the best assembly by taking into account the number of assembly errors as well as gaps.

***Caenorhabditis elegans* assembly:**

The Pacbio dataset contains three different runs and there was a clear batch effect with the sequencing quality and the amount of data between runs. We thus tested the assembly tools with either all runs (290X in coverage), or the biggest run alone (240X in coverage). We also used the reads-correcting-and-trimming module from Canu 1.8 [4] to correct the raw reads if the assembly tools take corrected reads as input. Preliminary assemblies were generated using Canu 1.8, Mecat2/Necat [3], Flye 2.4 [5], Miniasm 0.3-r179 [35], and Wtdbg2 [36], from Pacbio raw and corrected reads as well as Nanopore raw reads. By comparing the summary statistics of preliminary assemblies, ten preliminary assemblies were chosen for GALA.

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

We assembled seven linkage groups with LGAM, each into a continuous sequence. Among the seven continuous sequences and one unanchored short contig, four of them revealed the telomere repetitive motif at both terminals, indicating they are complete assemblies of single chromosomes. One chromosome-scale sequence had a 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 a telomere 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. Finally, we re-assembled the two newly created linkage groups and were able to create complete sequences for the two chromosomes with a telomeric repetitive motif at both terminals. Further analyses indicated that the split of this single chromosome into two linkage groups in the first run was mainly due to several tandem repeats.

Caenorhabditis elegans genome assembly polishing and quality control:

For a more accurate comparison, we polished our assembly with Pacbio and Illumina sequencing data. For this purpose, we first ran racon [37] with corrected Pacbio reads. The assembly was then polished using quiver 2.3.2 [38] with Pbmm2 1.1.0 as an aligner. Finally, we ran pilon 1.23 [39] using Illumina sequencing data to correct short errors, especially those in homomorphic regions.

We evaluated the completeness of our polished assembly with Busco 3.0.0, and compared it to the published assembly, which is also polished using the same Illumina sequencing data as well as the reference genome (Table 1 and Supplementary **Fig. 10**). We also aligned the Illumina short reads to our assembly using BWA-MEM and called the variants using BCFtools 1.9. Finally, we collected the statistics and compared them to those from the published assembly as a benchmark for the precision of the assembly.

Oryza sativa assembly:

Nanopore raw reads was corrected using the Canu 1.8 correcting and trimming module. We found that Necat showed good assembly performance using its own read correction module with the minimal length of reads parameter set at 5000. Eight preliminary assemblies were generated using Canu, Flye, Necat, Miniasm, and Wtdbg2 from corrected and raw reads. In LGAM analysis, reads were mapped to Canu and Necat preliminary assemblies.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and materials

Data availability: The Pacbio, Nanopro sequencing data, and Illumina reads of *C. elegans* are available at. Oryza sativa Dom Sufid Nanopore sequencing data are available at the European Nucleotide Archive (PRJEB32431). We downloaded the human dataset from

https://obj.umiacs.umd.edu/marbl_publications/hicanu/index.html, https://obj.umiacs.umd.edu/marbl_publications/hicanu/chm13_20k_hicanu_hifi.fasta.gz. Genome assemblies that were generated by GALA in this study are available at https://doi.org/10.5281/zenodo.6008862.

Code availability: The source code of GALA is available from github at https://github.com/ganlab/gala under the MIT license. The version of the source code of GALA used in the study is available at https://doi.org/10.5281/zenodo.4674388. External software used in the current study were downloaded from the following URLs: Bcftools1.9, https://github.com/samtools/bcftools/releases; Busco 3.0.0, https://busco-archive.ezlab.org/v3; BWA 0.7.15-r1140, https://github.com/lh3/bwa; Canu 1.8, https://github.com/marbl/canu; Fastq-sample b9a7f71 https://github.com/fplaza/fastq-sample; Flye 2.4, https://github.com/fenderglass/Flye; Hifiasm 0.5-dirty-r247, https://github.com/chhylp123/hifiasm; MECAT2, https://github.com/xiaochuanle/MECAT2; Miniasm 0.3-r179, https://github.com/lh3/miniasm; Minimap 0.2-r124-dirty https://github.com/lh3/minimap; Minimap2 2.17-r941, https://github.com/lh3/minimap2; NECAT, https://github.com/xiaochuanle/NECAT; PBJelly, https://github.com/esrice/PBJelly; pbmm2 1.1.0, https://github.com/PacificBiosciences/pbmm2; pilon 1.23, https://github.com/broadinstitute/pilon/releases; quiver 2.3.2, https://github.com/PacificBiosciences/GenomicConsensus; Racon 1.3.1, https://github.com/lbcb-sci/racon; Ragoo, https://github.com/malonge/RaGOO ;SALSA2, https://github.com/marbl/SALSA; and Wtdbg2 2.5, https://github.com/ruanjue/wtdbg2.

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported in part by a Max Planck Society core grant to the Department of Comparative Development and Genetics, a grant from the National Natural Science Foundation of China (Grant No. 3217040347), and grants from the National Science Foundation of Jiangsu Province in China (Grant No. JSSCRC2021508 and BK20212010). XG is supported by Jiangsu Collaborative Innovation Center for Modern Crop Production. MA is supported by the International Max Planck Research Schools programme.

Authors' contributions

XG conceived the project and interpreted the data. MA developed the GALA program and analyzed the data. XG and MA wrote the manuscript. The authors read and approved the final manuscript.

Acknowledgements

We thank M. Tsiantis, R. Mott and D. Megahed for their helpful comments on the work and Yuxia He for technical support. We also wish to acknowledge S. Morishita for sharing the original *C. elegans* Pacbio data with us.

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. Muggli, M.D., et al., *Misassembly detection using paired-end sequence reads and optical mapping data.* Bioinformatics, 2015. **31**(12): p. i80-8.
9. 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.
10. Chen, X. and M. Tompa, *Comparative assessment of methods for aligning multiple genome sequences.* Nat Biotechnol, 2010. **28**(6): p. 567-72.
11. 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**.
12. Bickhart, D.M. and G.E. Liu, *The challenges and importance of structural variation detection in livestock.* Front Genet, 2014. **5**: p. 37.
13. 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.
14. 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.
15. Yoshimura, J., et al., *Recompleting the Caenorhabditis elegans genome.* Genome Res, 2019. **29**(6): p. 1009-1022.
16. Ghurye, J., et al., *Integrating Hi-C links with assembly graphs for chromosome-scale assembly.* PLoS Comput Biol, 2019. **15**(8): p. e1007273.
17. Choi, J.Y., et al., *Nanopore sequencing-based genome assembly and evolutionary genomics of circum-basmati rice.* Genome Biol, 2020. **21**(1): p. 21.
18. Nurk, S., et al., *HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads.* Genome Res, 2020. **30**(9): p. 1291-1305.

19. Taylor, T.D., et al., *Human chromosome 11 DNA sequence and analysis including novel gene identification.* Nature, 2006. **440**(7083): p. 497-500.
20. Miga, K.H., et al., *Telomere-to-telomere assembly of a complete human X chromosome.* Nature, 2020.
21. Ekblom, R. and J.B. Wolf, *A field guide to whole-genome sequencing, assembly and annotation.* Evol Appl, 2014. **7**(9): p. 1026-42.
22. Lischer, H.E.L. and K.K. Shimizu, *Reference-guided de novo assembly approach improves genome reconstruction for related species.* BMC Bioinformatics, 2017. **18**(1): p. 474.
23. Schneeberger, K., et al., *Reference-guided assembly of four diverse Arabidopsis thaliana genomes.* Proc Natl Acad Sci U S A, 2011. **108**(25): p. 10249-54.
24. Alonge, M., et al., *RaGOO: fast and accurate reference-guided scaffolding of draft genomes.* Genome Biol, 2019. **20**(1): p. 224.
25. English, A.C., et al., *Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology.* PLoS One, 2012. **7**(11): p. e47768.
26. Arabidopsis Genome, I., *Analysis of the genome sequence of the flowering plant Arabidopsis thaliana.* Nature, 2000. **408**(6814): p. 796-815.
27. 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.
28. 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.
29. 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.
30. Paux, E., et al., *A physical map of the 1-gigabase bread wheat chromosome 3B.* Science, 2008. **322**(5898): p. 101-4.
31. Holusova, K., et al., *Physical Map of the Short Arm of Bread Wheat Chromosome 3D.* Plant Genome, 2017. **10**(2).
32. Bzikadze, A.V. and P.A. Pevzner, *Automated assembly of centromeres from ultra-long error-prone reads.* Nat Biotechnol, 2020. **38**(11): p. 1309-1316.
33. Li, H., *Minimap2: pairwise alignment for nucleotide sequences.* Bioinformatics, 2018. **34**(18): p. 3094-3100.
34. Li, H. and R. Durbin, *Fast and accurate short read alignment with Burrows-Wheeler transform.* Bioinformatics, 2009. **25**(14): p. 1754-60.
35. Li, H., *Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences.* Bioinformatics, 2016. **32**(14): p. 2103-10.
36. Ruan, J. and H. Li, *Fast and accurate long-read assembly with wtdbg2.* 2019: p. 530972.
37. Vaser, R., et al., *Fast and accurate de novo genome assembly from long uncorrected reads.* Genome Res, 2017. **27**(5): p. 737-746.
38. Chin, C.S., et al., *Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data.* Nat Methods, 2013. **10**(6): p. 563-9.
39. 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.

Table 1. The assembly performance evaluation of GALA with Busco scores and statistics of alignment of Illumina short reads. The Busco scores are computed using Busco V.3.0.0 with nematoda odb9 database. The QV scores are calculated using merqury reference free assessment tool.

	N2 reference genome	VC2010 assembly	GALA assembly
Assembly length	100,286,401	102,092,263	102,301,025
Number of contigs	7	7	7
Busco complete	968/982	968/982	970/982
Busco duplicated	6/982	6/982	6/982
Busco fragmented	8/982	8/982	6/982
Busco Missing	6/982	6/982	6/982
QV	36.4155	36.0716	36.2818
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

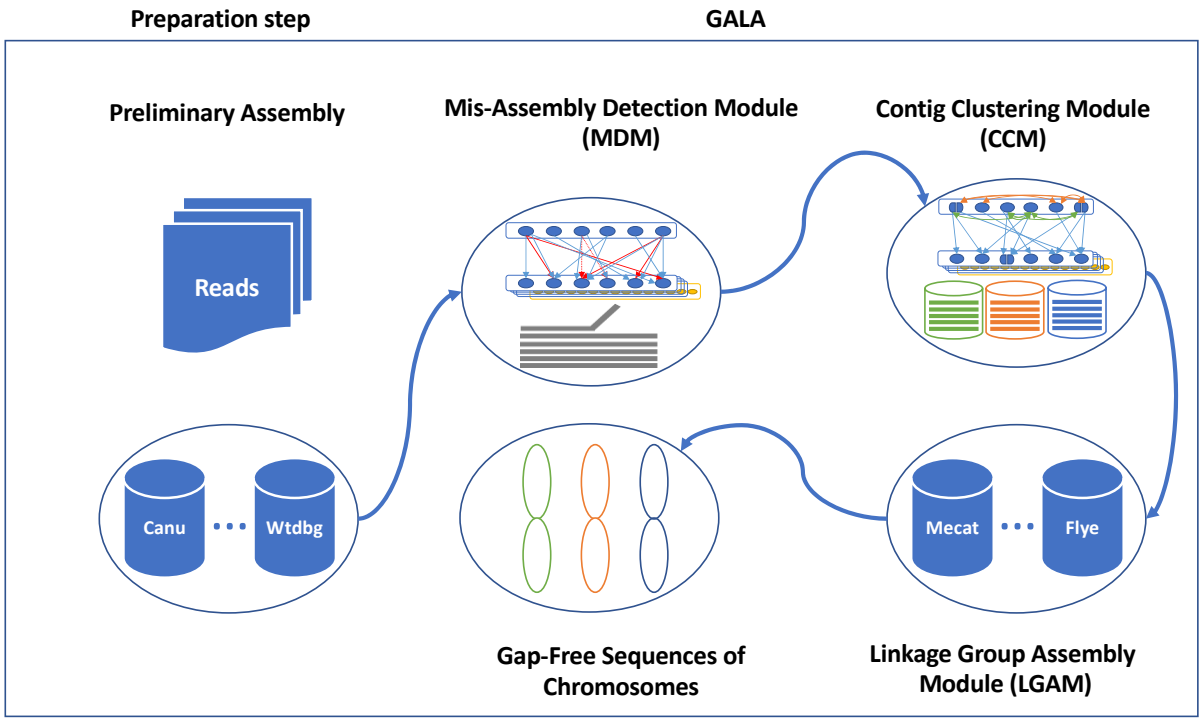
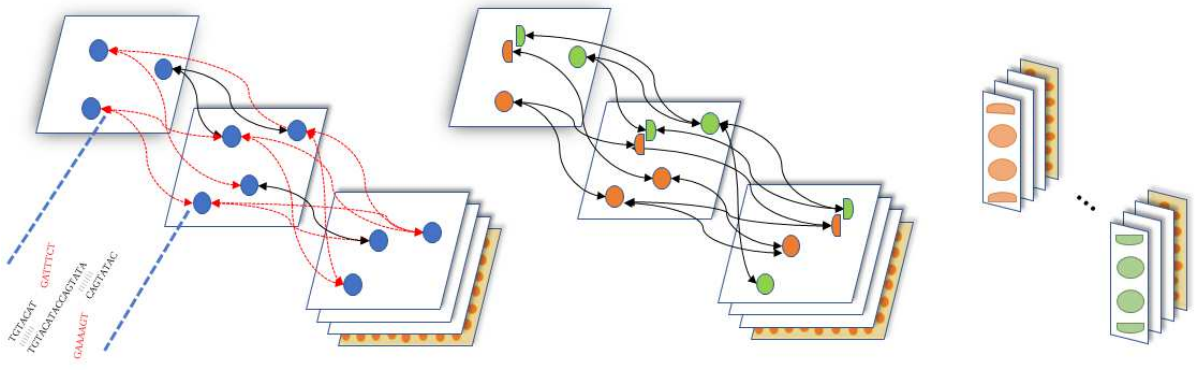
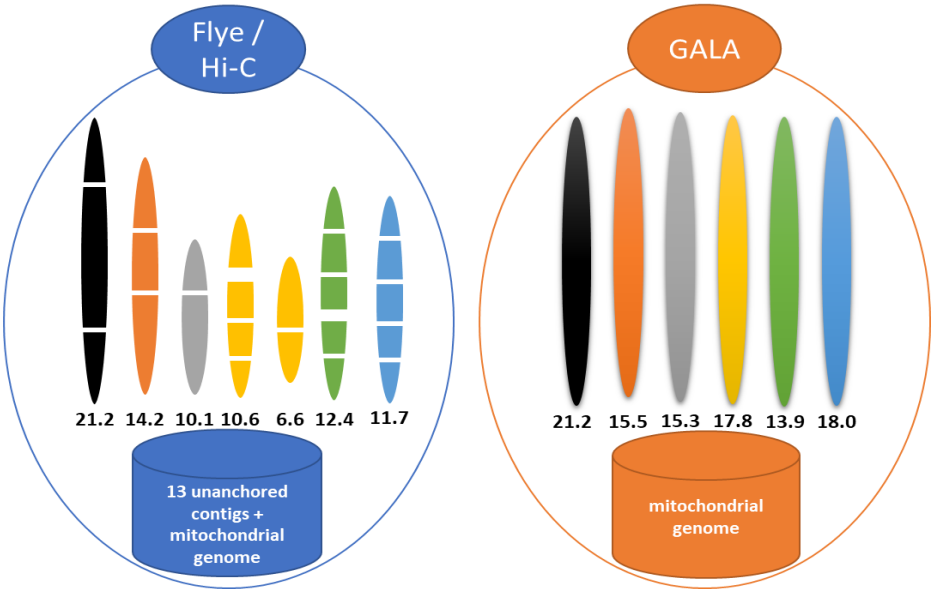


Figure 1. Overview of GALA. After *de novo* assembling with various tools, preliminary assemblies and raw reads are encoded into a multi-layer computer graph. Mis-assemblies are identified with MDM by browsing through the inter-layer information. The split nodes are clustered into multiple linage groups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



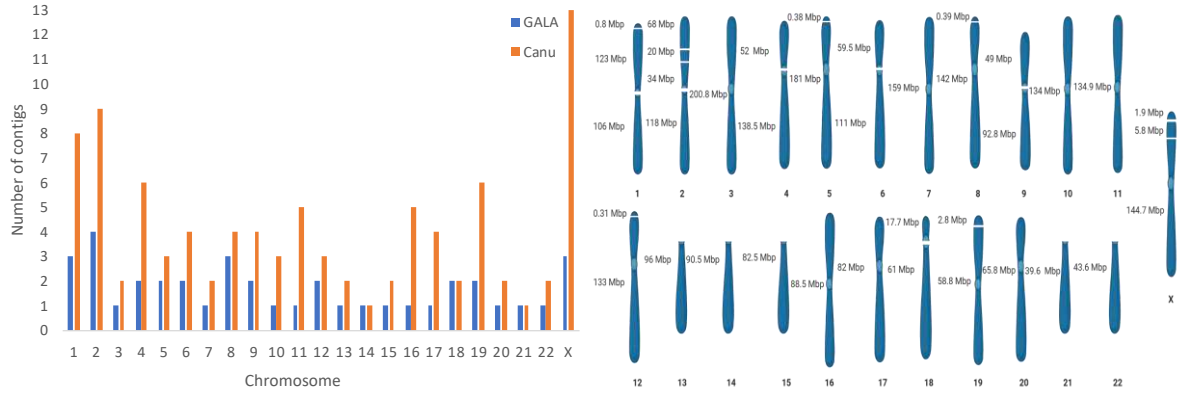
a **b** **c**

Figure 2. Illustration of a multi-layer computer graph in GALA. (a) The preliminary assemblies and raw reads are aligned against each other and encoded into a multi-layer graph. Conflicted alignments are encoded with edges in red. (b) The conflicted alignments are removed iteratively by splitting the nodes involved and new edges are assigned accordingly. The procedure stops only after all conflicted alignments in the system have been resolved. (c) Nodes connected by edges are clustered into linkage groups.



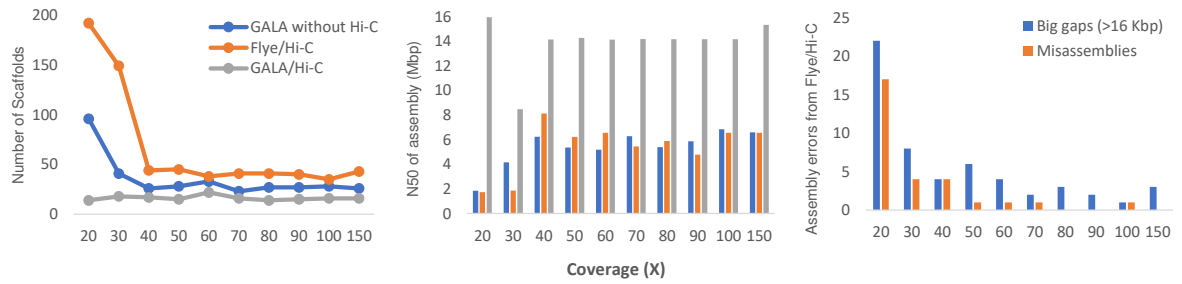
a **b**

Figure 3. Comparison of Flye assembly with Hi-C scaffolding and GALA assembly of long reads of the *C. elegans* genome. (a) The Flye assembly with Hi-C scaffolding 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 since GALA did not use Hi-C data in this assembly.



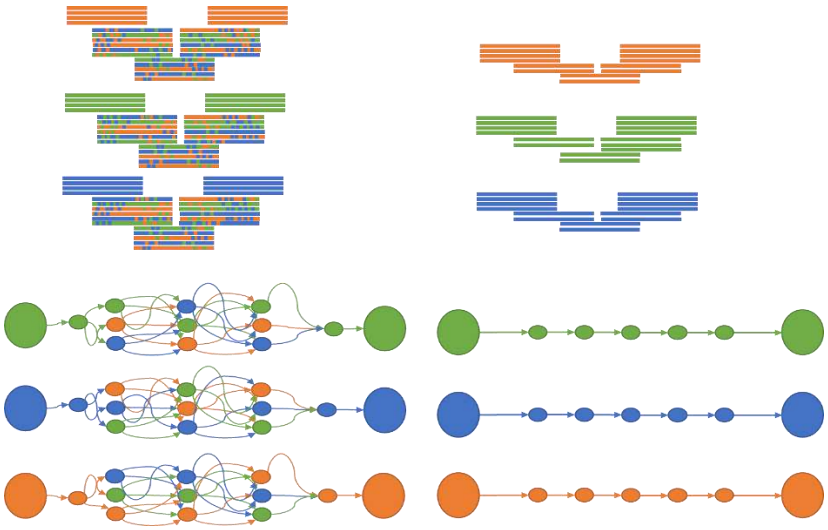
a **b**

Figure 4. Human genome assembly by GALA. (a) Comparison of the number of contigs in assemblies by Canu and GALA. (b) A cartoon presentation of each chromosome assembled by GALA with the lengths of contigs labelled.



a **b** **c**

Figure 5. The assembly performances of GALA and Flye with Pacbio sequencing data at various coverages. Three assembly procedures have been tested: GALA without Hi-C data, Flye/Hi-C, and GALA/Hi-C. The assembly performances are evaluated in terms of (a) the number of scaffolds, (b) N50, and (c) the number of big gaps (>16Kbp) and mis-assemblies. For simplicity, only the number of gaps and mis-assemblies for Flye/Hi-C have been shown, as only one mis-assembly has been identified in the assembly by GALA using 30X coverage sequencing data without the application of Hi-C data.



a **b**

Figure 6. Comparison of the overlap graphs used by Miniasm during assembly of a region in the *C. elegans* genomes when the chromosome-by-chromosome strategy is applied or not. (a) In the whole genome assembly mode, the overlap graph used by Miniasm contains numerous edges and extra effort is needed to collapse edges. (b) The chromosome-by-chromosome assembly allows a linear overlap graph to be derived by Miniasm in the same region.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- supplementaryinformation20220208.pdf