GigaScience

TGS-GapCloser: A fast and accurate gap closer for large genomes with low coverage of error-prone long reads

--Manuscript Draft--

Manuscript Number:	GIGA-D-20-00014R2
Full Title:	TGS-GapCloser: A fast and accurate gap closer for large genomes with low coverage of error-prone long reads
Article Type:	Technical Note
Funding Information:	Shenzhen Municipal Government of China Peacock Plan (KQTD2015033017150531) — Dr. Yongwei Zhang National Key Research and Development Program of China (2018YFD0900301-05) — Dr. Guangyi Fan Qingdao Applied Basic Research Projects (19-6-2-33-cg) — Dr. Mengyang Xu
Abstract:	Background: Analyses that use genome assemblies are critically affected by the contiguity, completeness, and accuracy of those assemblies. Recently, single molecule sequencing techniques generating long read information have become available and enabled substantial improvement in contig length and genome completeness, especially for large genomes (>100Mb), although bioinformatic tools for these applications are still limited. Findings: We developed a software tool to close sequence gaps in genome assemblies, TGS-GapCloser, that uses low-depth (~10×) long single molecule reads. The algorithm extracts reads that bridge gap regions between two contigs within a scaffold, error corrects only the candidate reads, and assigns the best sequence data to each gap. As a demonstration, we used TGS-GapCloser to improve the scaftig NG50 value of three human genome assemblies by 24-fold on average with only ~10× coverage of Oxford Nanopore or Pacific Biosciences reads, covering with sequence data up to 94.8% gaps with 97.7% positive predictive value. These improved assemblies achieve 99.998% (Q46) single-base accuracy with final inserted sequences having 99.97% (Q35) accuracy, despite the high raw error rate of single molecule reads, enabling high quality downstream analyses, including up to a 31-fold increase in the scaftig NGA50 and up to 13.1% more complete BUSCO genes. Additionally, we show that even in ultra large genome assemblies, such as the ginkgo (~12Gb), TGS-GapCloser is able to cover 71.6% of gaps with sequence data. Conclusions: TGS-GapCloser can close gaps in large genome assemblies using raw long reads in a fast and cost-effective way. The final assemblies generated by TGS-GapCloser have improved contiguity and completeness while maintaining high accuracy. The software is available at https://github.com/BGI-Qingdao/TGS-GapCloser .
Corresponding Author:	Mengyang Xu BGI CHINA
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	BGI
Corresponding Author's Secondary Institution:	
First Author:	Mengyang Xu
First Author Secondary Information:	
Order of Authors:	Mengyang Xu Lidong Guo Shengqiang Gu

	Ou Wang
	Rui Zhang
	Brock A. Peters
	Guangyi Fan
	Xin Liu
	Xun Xu
	Li Deng
	Yongwei Zhang
Order of Authors Secondary Information:	
Response to Reviewers:	Dear Editor, We are pleased that the manuscript entitled “TGS-GapCloser: A fast and accurate gap closer for large genomes with low coverage of error-prone long reads.” (GIGA-D-20-00014R1) is potentially acceptable in GigaScience. Again, we would like to thank you and the reviewers for the time spent on reviewing our manuscript and the comments helping us to improve the article. As suggested, we mainly focused on improving the language in the manuscript. The structure remains unchanged. We then address comments specific to each reviewer below. Response to Reviewer #1: Comment: The authors have solved all my previous concerns, and I don't have further comments for the manuscript. I would recommend for an "Accept". Response: We again appreciate the reviewer’s positive comments. It does not require any further revision. Response to Reviewer #2: Comment: Most of my technical concerns were addressed well and the authors have done a good job in this regard. At this stage, consider the paper in a state of minor revision but it is on the borderline because I cannot overlook the number of issues I have found in the writing. Though some of the revisions to the paper have improved the writing, the paper still has some issues that indicate more proof-reading is needed. Again, the paper is written well enough that most readers who have an understanding of assembly algorithms can figure out the intent of what you are trying to say in most cases, but it overall comes off as far too sloppy for publication. To be fair, I have seen many other manuscripts with much worse writing, but there are too many errors to overlook. Based on the type of errors I see, there are hallmarks of correction via grammar checking software, albeit almost a blind acceptance of what the program was spitting out. Grammar checking software is a good tool but not a substitute for proper proofreading, as I imagine is the case for any language. Here are some examples of erroneous or poor writing as well as the possible correction (sequentially from the start of the paper): - continuity, completeness -> contiguity, completeness (substitutions of contiguity for continuity occur multiple times in the paper, most grammar checking programs would think the use of "contiguity" is an error as it is generally an assembly specific term) - The development of genome sequencing techniques has been reducing the cost and improving the throughput at a speed beyond the Moore's Law over the last decade -> Genome sequencing techniques have been reducing in cost and improving in throughput at a speed beyond the Moore's Law over the last decade ("the cost/throughput" of what? The use of the preposition "in" ties the subject with these terms) - progressively increasing focuses move from small bacterial and fungal genomes to large eukaryotes. -> progressively increasing a focus from smaller bacterial and fungal genomes to larger eukaryotes genomes. (this was just wrong, but I do understand the intent) - BioNano physical map[10], provides -> BioNano physical maps [10], provide - relative to the NGS-based assembly -> relative to pure NGS-based assemblies.

	- the limitation of sequencing platform -> limitations of sequencing platforms - and the trade-off of algorithms -> and algorithms trade-offs - The first effort to finish gaps in draft genome assemblies was made -> The first efforts to finish gaps in draft genome assemblies were made - The NGS technologies -> NGS technologies - overcame the financial problem -> overcame this financial problem - of large CPU and memory consuming -> of large CPU and memory consumption I have only provided only corrections for the first few pages of the paper (up to page 4). Given the number of errors in such a short span of the manuscript, I think you can see why I am concerned. Reviewers are not copy-editors but these errors are quite minor and if they only occurred a few times I would have accepted this paper and have just provided corrections for all of them. Please consider having someone with a good grasp of the English language (ideally with an understanding of assembly) edit the work. Structurally the organization of ideas of the paper is done well; the authors clearly have an understanding of how to communicate science but it is unfortunate English can be such a frustrating language to use, yet is also the de-facto language of science. Response: We would like to thank the reviewer for raising this writing quality issue. All the errors mentioned by the reviewer have been corrected. To further improve the English, we have invited two bioinformatics experts to carefully go through the manuscript: Dr. Brock Peters is a native English speaker in the US, and Dr. Yongwei Zhang has been working in the US for more than 20 years. Both of them have made extensive polishing, and thus are listed as co-authors. We hope that the current writing quality can meet the requirement of publishing. We look forward to hearing from you and would like to respond to any further questions and comments you or reviewers may have. Sincerely, Mengyang Xu
Additional Information:	
Question	**Response**
Are you submitting this manuscript to a special series or article collection?	No
Experimental design and statistics Full details of the experimental design and statistical methods used should be given in the Methods section, as detailed in our Minimum Standards Reporting Checklist. Information essential to interpreting the data presented should be made available in the figure legends. Have you included all the information requested in your manuscript?	Yes
Resources A description of all resources used, including antibodies, cell lines, animals	Yes

and software tools, with enough information to allow them to be uniquely identified, should be included in the Methods section. Authors are strongly encouraged to cite Research Resource Identifiers (RRIDs) for antibodies, model organisms and tools, where possible. Have you included the information requested as detailed in our Minimum Standards Reporting Checklist?	
Availability of data and materials All datasets and code on which the conclusions of the paper rely must be either included in your submission or deposited in publicly available repositories (where available and ethically appropriate), referencing such data using a unique identifier in the references and in the “Availability of Data and Materials” section of your manuscript. Have you have met the above requirement as detailed in our Minimum Standards Reporting Checklist?	Yes

TGS-GapCloser: A fast and accurate gap closer for large genomes with low coverage of error-prone long reads

Mengyang Xu[1,2,4,#], Lidong Guo[3,1,#], Shengqiang Gu[3,1,#], Ou Wang[4,6], Rui Zhang[1], Brock A. Peters[4,7], Guangyi Fan[1,4], Xin Liu[1,2,4,5], Xun Xu[4,5], Li Deng[1,2,4,*] & Yongwei Zhang[4,7,*]

[1]BGI-Qingdao, BGI-Shenzhen, Qingdao 266555, China

[2]State Key Laboratory of Agricultural Genomics, BGI-Shenzhen, Shenzhen 518083, China

[3]BGI Education Center, University of Chinese Academy of Sciences, Shenzhen 518083, China

[4]BGI-Shenzhen, Shenzhen 518083, China

[5]China National GeneBank, BGI-Shenzhen, Shenzhen 518120, China

[6]MGI, BGI-Shenzhen, Shenzhen 518083, China

[7]Complete Genomics Inc., 2904 Orchard Pkwy, San Jose, California, 95134, USA

[#]These authors contributed equally to this work.

*Corresponding authors: Li Deng (dengli1@genomics.cn) and Yongwei Zhang (zhangyongwei@genomics.cn)

ORCIDs:

Mengyang Xu, 0000-0002-4487-7088;

Ou Wang, 0000-0001-8673-6497;

Brock A Peters, 0000-0002-5137-3902;

Guangyi Fan, 0000-0001-7365-1590;

Xin Liu, 0000-0003-3256-2940;

Xun Xu, 0000-0002-5338-5173;

Abstract

Background: Analyses that use genome assemblies are critically affected by the contiguity, completeness, and accuracy of those assemblies. In recently years single molecule sequencing techniques generating long read information have become available and enabled substantial improvement in contig length and genome completeness, especially for large genomes (>100Mb), although bioinformatic tools for these applications are still limited.

Findings: We developed a software tool to close sequence gaps in genome assemblies, TGS-GapCloser, that uses low-depth (~10×) long single molecule reads. The algorithm extracts reads that bridge gap regions between two contigs within a scaffold, error corrects only the candidate reads, and assigns the best sequence data to each gap. As a demonstration, we used TGS-GapCloser to improve the scaftig NG50 value of three human genome assemblies by 24-fold on average with only ~10× coverage of Oxford Nanopore or Pacific Biosciences reads, covering with sequence data up to 94.8% gaps with 97.7% positive predictive value. These improved assemblies achieve 99.998% (Q46) single-base accuracy with final inserted sequences having 99.97% (Q35) accuracy, despite the high raw error rate of single molecule reads, enabling high quality downstream analyses, including up to a 31-fold increase

in the scaftig NGA50 and up to 13.1% more complete BUSCO genes. Additionally, we show that even in ultra large genome assemblies, such as the ginkgo (~12Gb), TGS-GapCloser is able to cover 71.6% of gaps with sequence data.

Conclusions: TGS-GapCloser can close gaps in large genome assemblies using raw long reads in a fast and cost-effective way. The final assemblies generated by TGS-GapCloser have improved contiguity and completeness while maintaining high accuracy. The software is available at https://github.com/BGI-Qingdao/TGS-GapCloser.

Keywords: gap-closure, third-generation sequencing, genome assembly, ginkgo, MHC

Findings

Introduction

The cost and time necessary to a megabase of DNA has been decreasing at a speed beyond Moore's Law over the last decade[1]. Databases of genetic sequences have been growing dramatically with the size of completed genomes increasing from small bacterial and fungal genomes to very large eukaryotic genomes. In addition to short read second generation sequencing technologies (NGS) that have enabled this dramatic increase in genome sequencing, recent state-of-the-art techniques, such as, third generation single molecule long reads (TGS)[2, 3], synthetic long read (SLR) libraries[4-6], Hi-C[7], and BioNano physical maps[8], have provided long range

genome information to help increase contiguity of genome assemblies. However, the finished assemblies for most large genomes (> 100 Mb) remain imperfect and contain numerous gaps of unknown nucleic acids (represented by N's)[9, 10]. These gaps are often due to repetitive or difficult DNA sequences, polymorphisms between individual genomes of the same species, limitations of sequencing platforms, and algorithmic trade- offs. The process of gap closure or gap filling can recover these unknown bases and extend scaftigs (contigs within a scaffold without N's)[11] to completely or partially bridge these gaps and there is a need for tools to enable this on existing assemblies, especially for large highly complex eukaryotic genomes.

The first efforts to close gaps in genome assemblies were made using Fosmid and BAC libraries combined with Sanger sequencing[12]. But cost and labor associated with this manual to semi-automated gap-closing process were very high[10] and practically limited to only very well-funded genome programs (e.g., the Human Genome Project). As NGS technologies lowered sequencing costs, new paired-end and mate-pair libraries made processes and several bioinformatics tools were designed to help improve the gap-closing process [13-17]. These tools were based on *k*-mer-extension or local reassembly algorithms, but suffered from large CPU and memory consumption. In addition, these strategies rarely spanned repetitive DNA regions and tended to cause more misassemblies due to the short read lengths of NGS.

Current single molecule TGS technologies, such as Pacific Biosciences (PacBio) and Oxford Nanopore (ONT) have the potential to break through these limitations as their reads can exceed 100 kb and are typically longer than most DNA repeats[18].

Although the *de novo* genome assembly using TGS reads alone is possible, the lower raw read accuracy relative to NGS platforms generally requires sufficient sequencing coverage and high computational costs for error correction of the assembly[19]. This correction is necessary as these base-calling errors may cause frameshifts and other changes in the gene-coding or regulatory regions and thus cause inaccurate interpretation of the genome[20].

Recently, there have been several hybrid assemblers designed to take advantage of the combination of both TGS and NGS read data. Most construct a final assembly graph by mixing NGS contigs and TGS long reads based on the Overlap-Layout-Consensus or string graph algorithm[21], or connect the contigs generated by NGS with their alignments against long reads[22-24]. In contrast, the gap-closing algorithms provide a direct way to reduce the computing complexity and costs through improvements only in the missing regions and preservation of the majority of the existing assembly information. PBJelly[10] is the first tool to use PacBio reads to close gaps through local assembly of the long reads in gap regions. FGAP[25] selects the best matched pre-assembled contig to fill gaps based on BLAST[26] alignments. GMcloser[27] tries to increase the accuracy of gap closure using likelihood-based classifiers. Cobbler[28] uses new aligners to accelerate the buildup of the relationship between long high-quality sequences (usually scaftigs/contigs from other assemblies) and input scaffolds, and patches the gaps if the alignment of long sequence to the assembly meets a threshold score. Finally, LR_Gapcloser[29] reduces the computational costs of alignments by fragmenting long reads into tags and aligning the short tags against

scaffolds instead of the whole long reads. These tools have been widely used to close gaps with TGS long reads, but their efficiencies and accuracies are very much dependent on the quality of the long reads used. PBJelly improves the quality of inserted long reads through local assembly, but requires sufficient coverage. Other tools bypass the limitation of input quality and require or recommend pre-error-corrected long reads or pre-assembled contigs. However, the additional assembly or correction for all reads prior to gap closure needs adequate coverage of expensive long reads or additional short NGS reads. This requires extra time and memory consumption, especially for large genomes. In addition, the correction algorithms might trim ambiguous segments[30] and split long reads into short fragments[31] due to the undetermined bases, thus losing valuable length information.

Three key factors should be considered to develop a TGS gap-closing algorithm. First, use TGS data as little as possible. Although the cost has been decreasing[32], the gap-closing efficiency is still the first priority, particularly for small labs or small projects. As such, local reassembly or pre-error correction based on the long-read overlaps is not preferable. Another important factor is the accuracy and precision in the selection of long reads to fill the gaps. It has been demonstrated that the number of assembly errors caused by gap-closing tools is higher than that of *de novo* assembled scaftigs[27]. The misalignments of long reads against the scaffolds caused by base-calling errors or repeats may increase the probability of large misassembly events. An effective scoring mechanism can prevent the gap-closing tools from making some of these incorrect selections of reads. Finally, the filled sequences should not diminish

the single-base level accuracy of the whole assembly and thus impact the quality of downstream analyses. There is still a need for error correction for the inserted raw long-read segments. It should be noted that recently PacBio improved its base-calling accuracy to 99.8%[33], which may simplify the problem, however this improved accuracy comes at a significant cost throughput and read length.

In this work, we describe a software tool, TGS-GapCloser, that uses low coverage error-prone long reads to close gaps in large genomes more efficiently and accurately than other current gap-closing tools. Using only 10× coverage of ONT or PacBio long reads[34, 35] applied to three *de novo* assembled human genomes we demonstrate an increase in the scaftig NG50 by 11.0 to 45.0-fold and an increase in the scaftig NGA50 by 6.8 to 30.6-fold. Further, we show that 71.6% of gaps in the ultra large genome assembly of ginkgo can be closed using just 10.5× coverage of corrected PacBio reads, increasing the scaftig N50 from 57.1kb to 364.8kb. A hybrid strategy of updating a draft *de novo* genome assembly with TGS-GapCloser is an efficient and accurate strategy for improving the quality of gene annotation and structure variation detection. Ultimately this will help lead to high quality downstream analyses of ontogeny, phylogeny, and evolution.

Data description

Three datasets from two species containing large genomes were used to examine the gap-closing results by TGS-GapCloser: human, human Chr19, and ginkgo. We sequenced *Homo sapiens* (HG001/NA12878, Coriell Cat# GM12878,

RRID:CVCL_7526)) using the MGIEasy stLFR Library Prep Kit on the DNBSEQ-G50 platform (formerly known as BGISEQ-500, RRID:SCR_017979) generating a total 660 Gb of read data. Reads mapped to the Chr19 reference were also extracted for comparisons and further analysis. These short reads were assembled using MaSuRCA[23] (version 3.3.1; MaSuRCA, RRID:SCR_010691) or Mercedes (in-house tool) to obtain short but highly accurate contigs, and the SLR long-range (co-barcode/read cloud) and short-range (paired-end) information provided by the stLFR technique were exploited to do further scaffolding by SLR-superscaffolder[36] (version 1.0.0). In addition, Supernova[37] (version 2.1.1) (Supernova assembler, RRID:SCR_016756) was used to obtain draft scaffolds despite being originally designed to assemble 10X Genomics data. To test the potential application of TGS-GapCloser, we used newly generated data from both long-read platforms (ONT and PacBio) to close gaps in human genome assemblies: ONT MinION Rel3 dataset (Rel3)[34] and PacBio CCS HiFi dataset (HiFi)[35].

The genome assembly of a female *Ginkgo biloba* (estimated genome size about 12 Gb) used in this study was obtained from [38] and was initially assembled with SOAPdenovo2 (SOAPdenovo2, RRID:SCR_014986) [13] and updated using Hi-C data[38]. The PacBio reads for ginkgo were sequenced on a PacBio Sequel using a Sequel Sequencing Kit 3.0 Bundle (4 rxn). A total of 256 Gb of read data with an average read length of 38,623 bp was generated. Error correction by Canu (Canu, RRID:SCR_015880) [30] reduced the data size to 126 Gb, with an average read length of 10,722 bp. Statistics for input assemblies and sequencing reads can be found

in Tables S1 and S4, respectively.

Algorithm and implementation of TGS-GapCloser

TGS-GapCloser can accept as input any type of TGS long reads or other pre-assembled contigs to fill gaps in a draft assembly in the four steps as shown in Figure 1: (i) identification of gap regions in the draft assembly; (ii) acquisition of candidates from the alignments of long reads against gaps; (iii) base-level error correction of alternative sub-long reads; and (iv) gap closure using the error-corrected candidates with the highest score for each gap or linkage of the neighboring scaftigs with overlaps.

Figure 1. A schematic of TGS-GapCloser workflow. (A) A flow chart of the overall algorithm, (B) a schematic description on how gap regions are identified, the acquisition of candidate long-read fragments, and the error correction of alternative sub-long reads, (C) a detailed flow chart for gap filling or scaftig merging in a gap region with the most appropriate medium/long-range information provided by long reads.

The input scaffolds were first split into fragments called scaftigs from the observed N positions in the scaffolds, and each pair of neighboring scaftigs based upon their positions in the shared scaffold were defined as a gap to be filled. TGS-GapCloser retains the input scaffold information as the base-level accuracy and the order and orientation of scaftigs, but not the estimated gap size. This is caused by the lack of

sufficient resolution in the long-range information provided by SLR, Hi-C, or BioNano to accurately predict the size of gaps below approximately 10 kb.

We used minimap2 (Minimap2, RRID:SCR_018550) [39] to align long reads against each gap to obtain the corresponding candidate fragments. A candidate for a specific gap is defined as the segment truncated from the aligned long reads in the N region between two neighboring scaftigs plus 2kb-long of aligned sequence on both sides of the gap. Each long read might provide several candidate sequences depending on the length spanned and base-calling accuracy, but is limited to give at most one candidate for the same gap. This is to avoid redundant alignments induced by the alignment algorithm and the high error rate of TGS reads.

The quantity and quality of candidate reads determines the efficiency and accuracy of gap closure. Thus, we designed a scoring system of candidates for quality control and filtration based on the length and identity ratio (matched bases/ aligned bases) of the alignment between a long-read candidate and flanking scaftig sequence next to the gap. The score *QS* is given by

$$QS = a \cdot logA_i + b \cdot logI_i + a \cdot logA_{i+1} + b \cdot logI_{i+1}$$

where letter *A* refers to the alignment length, letter *I* refers to the identity ratio for the *i*th and *i+1*th scaftigs, respectively; letter *a* and *b* are two arbitrary coefficients to distinguish *A* and *I*'s weights on the score, and have been tuned to 1:6 for the ONT dataset as default. For each gap, a maximum of ten candidates with the highest *QS* were chosen for error correction in order to limit the size of data for further analysis. To further reduce the complexity and requirements on computational resources, the

overlapped candidates in the same long read were clipped and merged prior to the correction. Either Pilon (Pilon, RRID:SCR_014731) [31] or Racon (Racon, RRID:SCR_017642) [40] were used to enhance the base-level accuracy of merged sequences. Pilon is capable of correcting individual base errors, small indels, and local misassemblies with short but accurate NGS reads, while Racon corrects sequencing errors by constructing a SIMD-accelerated partial-order alignment graph from the overlap of long reads. The short reads were aligned to candidates by minimap2 with the option *-k14 -w5 -n2 -m20 -s40 --sr --frag yes*.

The corrected candidates were realigned to the gap and scored again, and finally the one with the highest *QS* was selected to fill the gap. The correction not only increased the single-base accuracy but also helped to find the best final candidate. We hypothesized that the *QS* of a candidate with higher-quality alignments would be increased due to the more precise mapping to the gap region after error correction, while the candidates with relatively lower-quality alignments tends to fail to be mapped. After final alignment to the gap region, those 2 kb sequences aligning to the scaftigs on either side of the gap were removed and only the bases filling the gap from the highest scoring candidate were retained.

If the highest scoring candidate resulted in a reduction in bases within the gap, then the gap would collapse to a single scaftig according to the alignment. A portion of scaftigs could have overlaps with other scaftigs because of incorrect paths during the initial assembly graph or over-aggressive scaftig extension. However, a TGS read spanning the gap has the ability to solve the overlap if two scaftigs can be mapped to

the correct positions. Candidates resulting in a reduction in bases were selected only with more stringent criteria as large indels or homopolymeric repeats in long reads tend to cause incorrect overlaps. Gaps lacking any candidates could not be closed, in some cases this could be due to misassemblies in the draft assembly.

TGS-GapCloser is coded in the C++ programing language (requires GCC 4.4+). It uses minimap2 to obtain alignments, and Pilon (requires Java runtime 1.7+) or Racon (requires GCC 4.8+) to correct candidate fragments. The algorithm automatically identifies gaps and tries to find the best matched long read fragments to close gaps or merge adjacent scaftigs. To accelerate the gap closure without losing efficiency and accuracy, TGS-GapCloser only selects a limited number of fragmented long reads as candidates for subsequent error correction and competition. This also reduces the computational complexity and improves the accuracy through a straightforward but efficient scoring system (Table S3) and correction-enhanced mapping (Table S4). In addition, the aligner, minimap2 shows noticeable improvements in speed and mapping accuracy for error-prone long reads[39]; helping to shorten the time of sequence alignment and improve the overall quality of the final gap closed sequence. The details of each step of this process, including gap identification, mapping, candidate identification, error correction, and final candidate selection are recorded. The final output is reported in FASTA format, with a log file describing the detailed insertion/merging information.

Gap closure in the human genome

Three assemblies and two TGS datasets were used to benchmark the utility of TGS-GapCloser in gap closure and scaftig merging in the human genome. Using the same co-barcoded short read stLFR library, the whole genome was assembled by: (1) MaSuRCA assembled contigs + scaffolds from SLR-superscaffolder, (2) Mercedes assembled contigs + scaffolds from SLR-superscaffolder, and (3) contigs and scaffolds assembled by Supernova using all of the barcoded long-range information. Although MaSuRCA itself can scaffold the contigs, the assembler does not utilize the SLR information, and generates relatively short scaffolds. As such, it is necessary to employ SLR-superscaffolder to obtain a scaffold NG50 comparable to Supernova.

To assess the efficiency of TGS-GapCloser, we used ~10× coverage of long reads from an ONT Rel3 dataset with a claimed mean read identity of 82.73%[34] and a PacBio HiFi dataset with the claimed average read concordance of 99.8%[33]. The long-read fragments from ONT Rel3 were corrected by Pilon with NGS short reads while those from HiFi were corrected by Racon using the long reads themselves. Figure 2 describes the improvements in the assembly evaluation given by QUAST (QUAST, RRID:SCR_001228) [41] after gap closure. Up to 91.8% of a total of 191,189, 94.8% of a total of 129,408, and 86.8% of a total of 42,359 gaps were successfully closed by TGS-GapCloser for three assemblies. The scaftig NG50 increased from 13.6 kb to 610.6 kb with the ONT Rel3 reads and to 243.7 kb with the PacBio HiFi reads for Assembly #1, 15.8 kb to 682.4 kb with the ONT Rel3 reads and to 173.7 kb with the PacBio HiFi reads for Assembly #2, and 113.0 kb to 1,229.2 kb with the ONT Rel3 reads and to 1,566.1 kb with the PacBio HiFi reads for Assembly

#3. Additionally, the corresponding scaftig NGA50 was also improved from 13.4 kb to 411.1 kb and 205.9 kb for Assembly #1, 15.7 kb to 418.2 kb and 153.2 kb for Assembly #2, and 108.5 kb to 734.2 kb and 849.7 kb for Assembly #3 with ONT Rel3 and PacBio HiFi reads, respectively. Note that our current algorithm does not split or merge input scaffolds. But the scaffold NG50 and NGA50 may change as a result of the replacement of N's and the combining of scaftigs. As listed in Table S2, the genome fraction against the reference also increased by 1.4%, 3.2% and 0.4% with ONT Rel3 reads and 1.2%, 1.9% and 0.4% with PacBio HiFi reads for Assemblies #1-3, respectively, indicating that many of the sequence filled gaps in each assembly are mapped to the human reference assembly. The application of ONT Rel3 read dataset increased the large-scale misassemblies (>1kb) created by the filled sequences by 22.2% and 6.3% in Assemblies #2 and #3, but decreased misassemblies by 9.5% in Assembly #1 as a result of the updated scaffolds mapping more precisely to the reference. In addition, local misassemblies (<1kb) increased by 1.2-fold, 7.4-fold, and 1.1-fold for assemblies #1-3, respectively, despite the ONT Rel3 reads having undergone error correction. The PacBio HiFi dataset, with higher initial read accuracy, resulted in fewer induced misassemblies and local misassemblies: -6.1% and 0.3-fold for Assembly #1, 13.1% and 1.3-fold for Assembly #2, 13.9% and 0.5-fold for Assembly #3. Overall, ONT Rel3 reads closed more gaps resulting in better contiguity than PacBio HiFi reads with the tradeoff of inducing more assembly errors. This is because the ONT Rel3 dataset is composed of single long reads (the longest >500kb) while the PacBio HiFi dataset produces ~10-fold coverage of each single read

followed by a read consensus process resulting in ~13-kb final reads with higher single-base accuracy (Figure S3). The performance of TGS-GapCloser is substantially dependent on both the length and the accuracy of input long reads, which are current balancing factors for single-molecule sequencing techniques.

After gap closure of the assemblies, BUSCO[42] (version 3.0.2) (BUSCO, RRID:SCR_015008) analysis indicated there are possible improvements for bioinformatics analysis such as gene annotation. The assemblies were compared against the vertebrata_odb9 database. It revealed that 90.5%, 89.7% and 94.1% of the expected vertebrate genes are complete for Assemblies #1-3, respectively, with ONT Rel3, and 90.4%, 85.3% and 94.0% for Assemblies #1-3, respectively, with PacBio HiFi. A substantial improvement was observed from the original 86.2%, 76.6% and 90.7% for Assemblies #1-3, respectively.

Figure 2. Gap filling improvements and effects on the draft assemblies produced by TGS-GapCloser. (A) scaftig NG50, (B) scaftig NGA50, (C) number of remaining gaps, (D) genome fraction, (E) misassemblies and (F) local misassemblies for the human genome were calculated by direct counting or reported by QUAST.

Gap closure in the ultra large genome of ginkgo

The *Ginkgo biloba* is considered a living fossil with its form and structure essentially unchanged for over 270 million years. This makes it very unique in the evolutionary tree of life[43]. We applied TGS-GapCloser to the chromosomal-level assembly of

Ginkgo biloba[38] using ~10.5× coverage of Canu corrected PacBio reads. The input assembly has been assigned to 13 chromosomes totaling 9,570,195,624 bp of sequence interrupted by 613,821 gaps. TGS-GapCloser filled 71.6% of the gaps in the assembly and replaced N containing regions by 411,608,879 bp of sequence. This resulted in the scaftig N50 increasing from 57.1 kb to 364.8 kb. Previously most gap-closing tools had only been used for bacterial and fungal genomes or small eukaryotic genomes [25, 27, 44]. This is the first example of using a gap closing tool on an ultra large genome with reasonable computational resources.

Validation of gap-closing sequences

As a sanity check, we mapped the gaps in input scaffolds to the human reference assembly, generated filled sequences based on the reference assembly, and compared these to the filled long-read fragments created by TGS-GapCloser. Note that the statistics for the filled gaps described here are different from those given by direct counting (Figure 2 (C)) because the gaps closed with scaftig overlapping are not counted. The evaluation (Table 1) consists of two parts: long read accuracy and single base level accuracy. For the selection of fragments inserted by TGS-GapCloser, the validated PPV ranges from 98.1% to 62.0% and the sensitivity from 96.4% to 51.2% for the three assemblies. Overall, gap-closing results with PacBio HiFi reads show relatively higher PPV due to its higher read accuracy, but lower sensitivity due to its shorter read length. The accuracy of Assemblies #1 and #2 is better than that of Assembly #3, which has more small gaps. This result implies that TGS-GapCloser

tends to fill large gaps.

In terms of single-base level accuracy, we calculated the Phred-like concordance QV by the method described in [33]. The QV of the inserted long-read fragments was improved after error correction. However, the overall QV of the assembly decreased: the scaftig QV was reduced from 45.8 to 40.8 with ONT Rel3 reads and to 42.1 with PacBio HiFi reads on average. Accuracy decline was less obvious with PacBio HiFi reads after error correction, which was consistent with the higher PPV in the long-read selection. That said, the final assemblies had >Q40 single-base quality making them comparable to or even better than most *de novo* TGS assemblies with pre-error correction and polishing[33, 34].

Table 1. Gap-closing accuracy statistics and computational consumptions for TGS-GapCloser.

Performance of TGS-GapCloser for large genomes

TGS-GapCloser is relatively fast and accurate. For the human genome, it consumed as little as 155 CPU hours in total and 32 GB of peak memory. The algorithm design substantially reduced the time for read mapping and error correction. Gap closure using the NGS-based error correction for the inserted sequences (~189 hours on average) was much slower than that with the TGS-based correction (~15 hours on average). As a comparison, the *de novo* assembly for 30× coverage of long reads requires ~40K CPU hours for ONT and ~62K CPU hours for PacBio[34]. The

computation can be further reduced by not using error correction. It only took 541 CPU hours for the ginkgo genome using pre-corrected PacBio reads. TGS-GapCloser requires low coverage of expensive long reads without pre-error correction, making this approach more cost-effective and suitable for research projects with limited budgets.

Comparison with other gap-closing tools

We did not compare TGS-GapCloser to NGS gap-closing tools because the utilization of TGS read information can span the repetitive or other complicated regions in the assembly that *k*-mer-based extension approaches cannot. In this paper, we used a variety of published long-read gap closers, including PBJelly (PBJelly, RRID:SCR_012091) [10], FGAP[25], GMcloser (GMcloser, RRID:SCR_000646) [27], Cobbler[28] and LR_Gapcloser[29], on the same Chr19 Mercedes+SLR-superscaffolder assembly with ONT Rel3 reads, and systematically compared their performances.

The comparison shows that TGS-GapCloser has the best overall performance among six tools with this combination of inputs (Table 2). Its gap-closing efficiency was considerably higher than that of other tools, reducing the number of gaps from 2,600 to 288, and increasing the scaftig NG50 from 9.6 kb to 194.5 kb. LR_Gapcloser, the next best performing tool for total gaps filled, was able to increase the scaftig NG50 to 157.2 kb. FGAP, closed a similar number of gaps to LR_Gapcloser, leaving 458 gaps unfilled, and was able to increase the scaftig NG50 to 127.4 kb. The remaining

tools, PBJelly, GMcloser, and Cobbler left more than 1,000 gaps unresolved and did not show much increase in scaftig lengths.

In terms of accuracy, TGS-GapCloser led to the largest increase (> 5.2X) in the scaftig NGA50 (16.0 folds to input) with fewer misassemblies. Although FGAP and LR_Gapcloser extended the scaftig NG50 longer than 100 kb, both generated more misassemblies resulting in a shorter scaftig NGA50. Most gap-closing tools were originally designed for error-corrected long reads or high-quality pre-assembled contigs, and as a result, their performances are mostly unsatisfactory with low-coverage raw ONT reads.

Table 2. Gap filling statistics for TGS-GapCloser and other gap-closing tools.

Input data	Unfilled gaps	Misassembly	Local misassembly	Scaffold NG50 (bp)	Scaffold NGA50 (bp)	Scaftig NG50 (bp)	Scaftig NGA50 (bp)	Runtime (min)	Peak memory (GB)
Draft Assemblies	2,600	176	126	1,561,142	196,307	9,687	9,464	/	/
TGS-GapCloser	288	187	324	1,426,438	383,995	194,512	149,166	12	16.37
PBJelly	1,730	664	741	1,240,439	83,803	29,715	19,247	3,137	9.93
FGAP	458	867	684	1,871,611	44,244	127,982	28,615	2,687	35.06
GMcloser	2,600	175	125	1,561,142	195,886	9,570	9,335	17,140	11.39

Cobbler	1,475	230	516	1,522,592	176,960	24,072	18,217	24	9.43
LR_Gapcloser	447	1,064	1,076	1,561,028	27,211	157,181	18,216	74	2.90

All datasets were run with 16 threads on the same computer. Note that QUAST accepts <10 continuous N's in the scaftig.

In addition, we analyzed the running time and memory consumption for each tool under the same operating conditions. TGS-GapCloser ran approximately 261-, 224-, 1428-, 2- and 6-fold faster than PBJelly, FGAP, GMcloser, Cobbler and LR_Gapcloser, respectively. The BLAST[26]-based GMcloser and FGAP, and the BLASR[45]-based PBJelly were the most time-consuming. The relatively higher memory requirement of TGS-GapCloser was due to the error correction needs. LR_Gapcloser employed short-tag comparisons to avoid long-read alignments, and thus required less memory than others.

Effects of long-read coverage

It is worth noting the effects of long-read coverage on the gap closure. We randomly extracted 1×, 5×, 10×, 20× and 29× coverages of mapped ONT Rel3 reads against the Chr19 reference, and individually applied them to the same Chr19 MaSuRCA +SLR-superscaffolder assembly by TGS-GapCloser using the same default parameters. As shown in Figure S1 (A), the number of closed gaps and the total filled bases grew with the increasing coverage, but saturated at ~10× coverage, close to the level of theoretically filled gap numbers and bases. Surprisingly, the total time usage did not

change much with the increasing coverage, but the peak memory showed an approximately linear growth (Figure S1, B). With more long reads, the sensitivity of inserted sequences increased from 22.1% to 87.4% while the PPV remained similar (Figure S1, C). In terms of single-base level accuracy, the average concordance QV of inserted sequences dropped as more gaps were closed, but had a negligent effect on that of scaftigs (Figure S1, D). The result indicates that TGS-GapCloser closes a considerable number of gaps with high quality sequence while using low-coverage error-prone long reads. In contrast, a high-quality long-read assembly requires at least 30× sequencing coverage[19].

Improvements in the MHC region

TGS read data has been shown to be useful in the assembly of the human MHC region. This ~6 Mb region in Chr6 is difficult to assemble with short reads only due to high repetition and polymorphism[34]. It contains class I and II human leukocyte antigen genes, important to cancer and immunity studies[46]. We analyzed three assemblies before and after gap closure to investigate the contiguity and accuracy in this region as shown in Table 3. For Assembly #3, a portion of a single long scaffold (>29 Mb) completely covered the MHC region, while several portions of two or three scaffolds (0.6-27 Mb) covered the region for Assemblies #1 and #2. Gap closure with the ONT Rel3 dataset reduced the number of scaftigs in those scaffolds from 339, 271, and 76 to 31, 26, and 12 in Assemblies #1, #2, and #3, respectively. In addition, TGS-GapCloser reduced the percent of N bases from 15.2% of the total assembly

down to 3.7% on average while increasing the genome fraction mapped to the reference assembly from 81.52% to 91.17%. As a result, the scaftig NG50 and NGA50 improved from 46.7 kb to 585.1 kb, and 41.0 kb to 300.4 kb. Importantly, this result would be expected to improve the gene annotations, structural variation detection, and single nucleotide polymorphism calling in this region. Although TGS long reads resolved the MHC locus into one or several contigs, the relatively short contig NGA50 (52.6 kb), low genome fraction (59.86%), and numerous local misassemblies indicated that improving the accuracy in short-range information was still a challenge for TGS applications.

Table 3. Improved assemblies in the MHC region by TGS-GapCloser.

Future direction

There are potential future improvements to consider for TGS-GapCloser. The selection of inserted sequences largely depends on the performance of the aligner. Although minimap2 performs well in most cases, the alignment results in errors if the pairwise sequences share small overlaps. We believe this can be solved by using other aligners or additional parameter optimization. In addition, the computational consumption by error correctors or polishers is still significant, even with our efforts to reduce the input data size as much as possible. As error-correction tools are updated and ideally become more efficient TGS-GapCloser performance will benefit from

these improvements. In addition, as long read error rates continue to fall, as promised by ONT and PacBio, it may be possible to eliminate this extra step of error correction. Finally, we use the input scaffolds including the orientation and order relations of scaftigs to retain the existing assembly information, ignoring possible assembly errors. As a future update we plan to use the information provided by TGS reads to correct scaftig errors within the same scaffold and link different scaffolds if sufficient overlapping is present. Nonetheless, in its current form, TGS-Gapcloser enables the combination of different genetic information with different lengths and resolutions and makes it possible to complete high-quality (ultra) large genome assemblies.

Methods

Gap closing with other tools

We compared the performance of TGS-GapCloser with that of five TGS gap-closing tools, including PBJelly (version PBSuite_15.8.24) (PBJelly, RRID:SCR_012091), FGAP (version 1.8.1), GMcloser (version 1.6.2) (GMcloser, RRID:SCR_000646), Cobbler (version 0.6.1) and LR_Gapcloser (no version information available) (LR_Gapcloser, RRID:SCR_016194). Some were unable to close gaps using the default parameters on low coverage raw TGS reads. As a result, we needed to tune FGAP to be able to close large gaps (<100kb, default <500). For GMcloser, we used the example parameters for long reads from the manual. In addition, the parameters for Cobbler were tuned according to the authors' guidance on GitHub. All other tools

were run using the default parameters for ONT data.

Validation of gap-closing results

We evaluated the gap-closing accuracy at two levels: the selection of long reads and the single-base level. The former is determined by whether the algorithm can capture the best long read to close the corresponding gap. This will affect the detection of chromosomal variations, large relocations, and inversions. The quality of error correction and the size of inserted long-read bases determines the single-base level accuracy. This affects single-nucleotide polymorphisms and small insertion/deletion calls.

QUAST[41] (version 5.0.2) (QUAST, RRID:SCR_001228) was used to determine length statistics for the assembly such as total length, scaffold NG50, and scaftig NG50, as well as alignment to the reference, including scaffold NGA50, scaftig NGA50, genome fraction, misassemblies, and local misassemblies. To further assess the efficiency and accuracy of TGS-GapCloser, we aligned the reference assembly against the input scaffolds to generate theoretically filled gap sequences using QUAST intermediate files, and compared them to the filled sequences by TGS-GapCloser with minimap2 (*-x map-ont*). Gaps that were capable of being filled by the reference were chosen to evaluate the sensitivity and PPV. Note that gaps smaller than 100 bp were filtered out. The sensitivity is defined as the ratio of the number of TGS-GapCloser filled gaps that the reference also successfully fills to the total number of gaps that the reference can fill. The PPV is defined as the ratio of the number of TGS-

GapCloser filled gaps that can be uniquely matched to the reference-filled gaps to the total number of filled gaps by both. Note that TGS-GapCloser also completes gaps that the reference cannot fill and as such the accuracy of these cannot be easily determined. The single-base level accuracy was quantified by mapping the scaftigs in the assembly to the GIAB high-confidence regions in the reference genome GRCh37 to calculate the concordance QV with the method in [33], where the scaftigs were split into bins of 100 kb, and those bins with >50% mapped length at >50% identity ratio were used to calculate the average concordance quality value. The QVs were expressed in Phred format.

BUSCO

To quantify the possible improvements for downstream bioinformatics analyses, we ran BUSCO on all the human assemblies against the vertebrata_odb9 and the gingko assemblies against the embryophyta_odb9 database. Note that we directly input the whole human assemblies, but split gingko ultra-long scaffolds (>1.1 Gb) into several portions at the position of large gaps (>1 kb) as the aligner tblastn[47] in BUSCO could not handle such long sequences. The additional random breakpoints in the original scaffolds would decrease the contiguity and affect the BUSCO benchmarking.

Availability of source code and requirements

Project name: TGS-GapCloser

Project home page: https://github.com/BGI-Qingdao/TGS-GapCloser

Operating system(s): Linux

Programming language: C++, shell

Other requirements: Racon, or SAMtools and Pilon are required to be pre-installed

License: GPLv3

RRID: SCR_017633

biotools ID: TGS-GapCloser

Conda Access: conda install -c bioconda tgsgapcloser

Availability of supporting data and materials

The stLFR sequencing data for the human sample (HG001/NA12878) has been deposited in the CNGB under accession number CNP0000066. We downloaded the ONT long reads of human from [48], and PacBio reads from GIAB[49]. The PacBio long reads for ginkgo genome has been deposited in the CNGB under accession number CNP0000796 (PRJNA656117). All the evaluated assemblies of human and ginkgo generated by us can be obtained in the CNGB under accession number CNP0000796. The genome assemblies and all supporting data can be accessed at the *GigaScience* GigaDB database[50].

Additional files

Supplementary information contains the following information:

Figure S1: Effects of long read coverage on gap closure. (A) the number of filled gaps and bases, (B) wall-clock time and peak memory, (C) accuracy in long-read selection, and (D) accuracy at single-base level. All datasets were run with 16 threads.

Figure S2. Length distribution of gaps in draft scaffolds and that of TGS-GapCloser filled gap sequences.

Figure S3. Read length distribution for the input ONT Rel3 and PacBio HiFi reads.

Table S1. Summary of the input assemblies in this work.

Table S2. Summary of the updated assemblies in this work.

Table S3. The effect of the scoring system on the candidate selection and the gap-closing performance.

Table S4. The effect of error correction on the candidate selection and the gap-closing performance.

Table S5. The effect of long-read coverage on the TGS assemblies and gap-closing results.

Table S6. Genomics dataset source.

Table S7. Control parameters used for different software tools.

Abbreviations

TGS: third-generation sequencing; GIAB: Genome in a Bottle; SLR: synthetic long reads; NGS: next-generation sequencing; PacBio: Pacific Biosciences; ONT: Oxford Nanopore Techniques; OLC: Overlap-Layout-Consensus; PPV: positive predictive value; MHC: major histocompatibility complex; stLFR: single tube Long Fragment Reads; Chr19: Chromosome 19; Chr6: Chromosome 6; QS: quality score; SIMD: single-instruction-multiple-data; BUSCO: Benchmarking Universal Single-Copy Orthologs; QV: quality value; CNSA: CNGB Nucleotide Sequence Archive.

Competing interests

All authors of the article are employees of BGI or its subsidiaries.

Funding

This research was supported by the Shenzhen Municipal Government of China Peacock Plan (No. KQTD2015033017150531), the National Key Research and Development Program of China (Grant No. 2018YFD0900301-05) and the Qingdao Applied Basic Research Projects (Grant No. 19-6-2-33-cg).

Authors' contributions

M.X., L.G., and L.D. performed software design and implementation. M.X., L.G., S.G., O.W., and R.Z. contributed to data modeling, data curation, and assembler

benchmarking. M.X. wrote the draft manuscript, and L.G., B.P., G.F., L.D. and Y.Z. contributed to manuscript editing. L.D. and Y.Z. supervised the project. M.X., G.F. and Y.Z. secured funding. All authors read and approved the final manuscript.

ACKNOWLEDGEMENTS

The authors are grateful for the advice from Hongmei Zhu, the support of Mercedes Assembler from Yinlong Xie, and many other BGI-Shenzhen employees in the development of TGS-GapCloser.

References

1. KA. W. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP). 2014.
2. Branton D, Deamer DW, Marziali A, Bayley H, Benner SA, Butler T, et al. The potential and challenges of nanopore sequencing. Nat Biotechnol. 2008;26 10:1146-53. doi:10.1038/nbt.1495.
3. Schadt EE, Turner S and Kasarskis A. A window into third-generation sequencing. Hum Mol Genet. 2010;19 R2:R227-40. doi:10.1093/hmg/ddq416.
4. Peters BA, Kermani BG, Sparks AB, Alferov O, Hong P, Alexeev A, et al. Accurate whole-genome sequencing and haplotyping from 10 to 20 human cells. Nature. 2012;487 7406:190-5. doi:10.1038/nature11236.
5. Kaper F, Swamy S, Klotzle B, Munchel S, Cottrell J, Bibikova M, et al. Whole-genome haplotyping by dilution, amplification, and sequencing. Proc Natl Acad Sci U S A. 2013;110 14:5552-7. doi:10.1073/pnas.1218696110.
6. Zheng GX, Lau BT, Schnall-Levin M, Jarosz M, Bell JM, Hindson CM, et al. Haplotyping germline and cancer genomes with high-throughput linked-read sequencing. Nat Biotechnol. 2016;34 3:303-11. doi:10.1038/nbt.3432.
7. Belton JM, McCord RP, Gibcus JH, Naumova N, Zhan Y and Dekker J. Hi-C: a comprehensive technique to capture the conformation of genomes. Methods. 2012;58 3:268-76. doi:10.1016/j.ymeth.2012.05.001.
8. Shelton JM, Coleman MC, Herndon N, Lu N, Lam ET, Anantharaman T, et al. Tools and pipelines for BioNano data: molecule assembly pipeline and FASTA super scaffolding tool. BMC Genomics. 2015;16:734. doi:10.1186/s12864-015-1911-8.

9. Eichler EE, Clark RA and She X. An assessment of the sequence gaps: unfinished business in a finished human genome. Nat Rev Genet. 2004;5 5:345-54. doi:10.1038/nrg1322.
10. 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 11:e47768. doi:10.1371/journal.pone.0047768.
11. Li Y, Hu Y, Bolund L and Wang J. State of the art de novo assembly of human genomes from massively parallel sequencing data. Hum Genomics. 2010;4 4:271-7. doi:10.1186/1479-7364-4-4-271.
12. Adams MD FC, Venter JC. Automated DNA Sequencing and Analysis Techniques. Academic Press; 1994.
13. Luo R, Liu B, Xie Y, Li Z, Huang W, Yuan J, et al. SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. Gigascience. 2012;1 1:18. doi:10.1186/2047-217x-1-18.
14. Boetzer M and Pirovano W. Toward almost closed genomes with GapFiller. Genome Biol. 2012;13 6:R56. doi:10.1186/gb-2012-13-6-r56.
15. Tsai IJ, Otto TD and Berriman M. Improving draft assemblies by iterative mapping and assembly of short reads to eliminate gaps. Genome Biol. 2010;11 4:R41. doi:10.1186/gb-2010-11-4-r41.
16. Gao S, Bertrand D and Nagarajan N. FinIS: Improved in silico Finishing Using an Exact Quadratic Programming Formulation. In: Berlin, Heidelberg, 2012, pp.314-25. Springer Berlin Heidelberg.
17. Puranik R, Quan G, Werner J, Zhou R and Xu Z. A pipeline for completing bacterial genomes using in silico and wet lab approaches. BMC Genomics. 2015;16 Suppl 3 Suppl 3:S7. doi:10.1186/1471-2164-16-s3-s7.
18. Catasti P, Chen X, Mariappan SV, Bradbury EM and Gupta G. DNA repeats in the human genome. Genetica. 1999;106 1-2:15-36. doi:10.1023/a:1003716509180.
19. Ou S, Liu J, Chougule KM, Fungtammasan A, Seetharam AS, Stein JC, et al. Effect of sequence depth and length in long-read assembly of the maize inbred NC358. Nature Communications. 2020;11 1:2288. doi:10.1038/s41467-020-16037-7.
20. Watson M and Warr A. Errors in long-read assemblies can critically affect protein prediction. Nat Biotechnol. 2019;37 2:124-6. doi:10.1038/s41587-018-0004-z.
21. Ye C, Hill CM, Wu S, Ruan J and Ma ZS. DBG2OLC: Efficient Assembly of Large Genomes Using Long Erroneous Reads of the Third Generation Sequencing Technologies. Sci Rep. 2016;6:31900. doi:10.1038/srep31900.
22. Boetzer M and Pirovano W. SSPACE-LongRead: scaffolding bacterial draft genomes using long read sequence information. BMC Bioinformatics. 2014;15:211. doi:10.1186/1471-2105-15-211.
23. Zimin AV, Marçais G, Puiu D, Roberts M, Salzberg SL and Yorke JA. The MaSuRCA genome assembler. Bioinformatics. 2013;29 21:2669-77. doi:10.1093/bioinformatics/btt476.
24. Luo J, Lyu M, Chen R, Zhang X, Luo H and Yan C. SLR: a scaffolding algorithm based on long reads and contig classification. BMC Bioinformatics. 2019;20 1:539. doi:10.1186/s12859-019-3114-9.
25. Piro VC, Faoro H, Weiss VA, Steffens MB, Pedrosa FO, Souza EM, et al. FGAP: an automated gap closing tool. BMC Res Notes. 2014;7:371. doi:10.1186/1756-0500-7-371.

26. McGinnis S and Madden TL. BLAST: at the core of a powerful and diverse set of sequence analysis tools. Nucleic Acids Res. 2004;32 Web Server issue:W20-5. doi:10.1093/nar/gkh435.
27. Kosugi S, Hirakawa H and Tabata S. GMcloser: closing gaps in assemblies accurately with a likelihood-based selection of contig or long-read alignments. Bioinformatics. 2015;31 23:3733-41. doi:10.1093/bioinformatics/btv465.
28. Warren RL. RAILS and Cobbler: Scaffolding and automated finishing of draft genomes using long DNA sequences. Journal of Open Source Software. 2016;1 7:116. doi:10.21105/joss.00116.
29. Xu G-C, Xu T-J, Zhu R, Zhang Y, Li S-Q, Wang H-W, et al. LR_Gapcloser: a tiling path-based gap closer that uses long reads to complete genome assembly. GigaScience. 2018;8 1 doi:10.1093/gigascience/giy157.
30. Koren S, Walenz BP, Berlin K, Miller JR, Bergman NH and Phillippy AM. Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. Genome Res. 2017;27 5:722-36. doi:10.1101/gr.215087.116.
31. 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 11:e112963. doi:10.1371/journal.pone.0112963.
32. Weirather JL, de Cesare M, Wang Y, Piazza P, Sebastiano V, Wang XJ, et al. Comprehensive comparison of Pacific Biosciences and Oxford Nanopore Technologies and their applications to transcriptome analysis. F1000Res. 2017;6:100. doi:10.12688/f1000research.10571.2.
33. Wenger AM, Peluso P, Rowell WJ, Chang PC, Hall RJ, Concepcion GT, et al. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. Nat Biotechnol. 2019; doi:10.1038/s41587-019-0217-9.
34. Jain M, Koren S, Miga KH, Quick J, Rand AC, Sasani TA, et al. Nanopore sequencing and assembly of a human genome with ultra-long reads. Nat Biotechnol. 2018;36 4:338-45. doi:10.1038/nbt.4060.
35. Zook JM, Chapman B, Wang J, Mittelman D, Hofmann O, Hide W, et al. Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. Nat Biotechnol. 2014;32 3:246-51. doi:10.1038/nbt.2835.
36. Deng L, Guo L, Xu M, Wang W, Gu S, Zhao X, et al. SLR-superscaffolder: a de novo scaffolding tool for synthetic long reads using a top-to-bottom scheme. bioRxiv. 2019:762385. doi:10.1101/762385.
37. Weisenfeld NI, Kumar V, Shah P, Church DM and Jaffe DB. Direct determination of diploid genome sequences. Genome Res. 2017;27 5:757-67. doi:10.1101/gr.214874.116.
38. Guan R, Zhao Y, Zhang H, Fan G, Liu X, Zhou W, et al. Updated genome assembly of Ginkgo biloba. GigaScience Database. 2019; http://dx.doi.org/10.5524/100613.
39. Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;34 18:3094-100. doi:10.1093/bioinformatics/bty191.
40. Vaser R, Sovic I, Nagarajan N and Sikic M. Fast and accurate de novo genome assembly from long uncorrected reads. Genome Res. 2017;27 5:737-46. doi:10.1101/gr.214270.116.
41. Gurevich A, Saveliev V, Vyahhi N and Tesler G. QUAST: quality assessment tool for genome assemblies. Bioinformatics. 2013;29 8:1072-5. doi:10.1093/bioinformatics/btt086.
42. Simao FA, Waterhouse RM, Ioannidis P, Kriventseva EV and Zdobnov EM. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs.

Bioinformatics. 2015;31 19:3210-2. doi:10.1093/bioinformatics/btv351.

43. Guan R, Zhao Y, Zhang H, Fan G, Liu X, Zhou W, et al. Draft genome of the living fossil Ginkgo biloba. Gigascience. 2016;5 1:49. doi:10.1186/s13742-016-0154-1.
44. de Sa PH, Miranda F, Veras A, de Melo DM, Soares S, Pinheiro K, et al. GapBlaster-A Graphical Gap Filler for Prokaryote Genomes. PLoS One. 2016;11 5:e0155327. doi:10.1371/journal.pone.0155327.
45. Chaisson MJ and Tesler G. Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory. BMC Bioinformatics. 2012;13:238. doi:10.1186/1471-2105-13-238.
46. Brandt DY, Aguiar VR, Bitarello BD, Nunes K, Goudet J and Meyer D. Mapping Bias Overestimates Reference Allele Frequencies at the HLA Genes in the 1000 Genomes Project Phase I Data. G3 (Bethesda). 2015;5 5:931-41. doi:10.1534/g3.114.015784.
47. Gertz EM, Yu YK, Agarwala R, Schäffer AA and Altschul SF. Composition-based statistics and translated nucleotide searches: improving the TBLASTN module of BLAST. BMC Biol. 2006;4:41. doi:10.1186/1741-7007-4-41.
48. Nanopore Whole Human Genome Sequencing Project. https://github.com/nanopore-wgs-consortium/NA12878/blob/master/nanopore-human-genome/rel_3_4.md. Accessed August 2020.
49. GIAB NA12878 PacBio_SequelII_CCS_11kb. ftp://ftp.ncbi.nlm.nih.gov/giab/ftp/data/NA12878/PacBio_SequelII_CCS_11kb/HG001.SequelII.pbmm2.hs37d5.whatshap.haplotag.RTG.trio.bam. Accessed August 2020.
50. Xu M, Guo L, Gu S, Wang O, Zhang R, Peters BA, et al. Supporting data for "TGS-GapCloser: A fast and accurate gap closer for large genomes with low coverage of error-prone long reads.". *GigaScience* Database. 2020; http://dx.doi.org/10.5524/100773.

Table 1. Gap-closing accuracy statistics and computational consumptions for TGS-GapCloser.

Accuracy in long-read selection						
Input data	**No. of closed gaps**	**No. of closed gaps in theory**	**PPV (%)**	**Sensitivity (%)**	**Runtime (hours)**	**Peak memory (GB)**
MaSuRCA+SLR-superscaffolder+TGS-GapCloser (ONT)	75,629	74,353	96.6	96.3	259	50
MaSuRCA+SLR-superscaffolder+TGS-GapCloser (PacBio)	74,321	74,353	98.2	89.8	13	33
Mercedes+SLR-superscaffolder+TGS-GapCloser (ONT)	58,938	61,267	97.7	93.4	145	51
Mercedes+SLR-superscaffolder+TGS-GapCloser (PacBio)	52,116	61,267	98.4	75.6	11	32
Supernova+TGS-GapCloser	22,563	24,760	62.0	51.2	163	74

(ONT)						
Supernova+TGS-GapCloser (PacBio)	26,919	24,760	76.1	61.2	20	38

Accuracy in single-base level

Input data	No. of filled bases (bp)	No. of filled bases in theory (bp)	Input QV (Phred)		Output QV (Phred)	
			Raw long reads	Scaftigs	Filled long reads	Scaftigs
MaSuRCA+SLR-superscaffolder+TGS-GapCloser (ONT)	335,541,557	353,352,038	7.63	40.51	23.24	36.06
MaSuRCA+SLR-superscaffolder+TGS-GapCloser (PacBio)	198,327,815	353,352,038	26.99	40.51	35.52	37.64
Mercedes+SLR-superscaffolder+TGS-GapCloser (ONT)	352,316,717	497,208,670	7.63	48.09	23.23	40.19
Mercedes+SLR-superscaffolder+TGS-GapCloser (PacBio)	146,148,151	497,208,670	26.99	48.09	36.25	42.29

Supernova+TGS-GapCloser (ONT)	49,669,581	38,276,270	7.63	48.72	23.15	46.11
Supernova+TGS-GapCloser (PacBio)	22,178,115	38,276,270	26.99	48.72	34.82	46.48

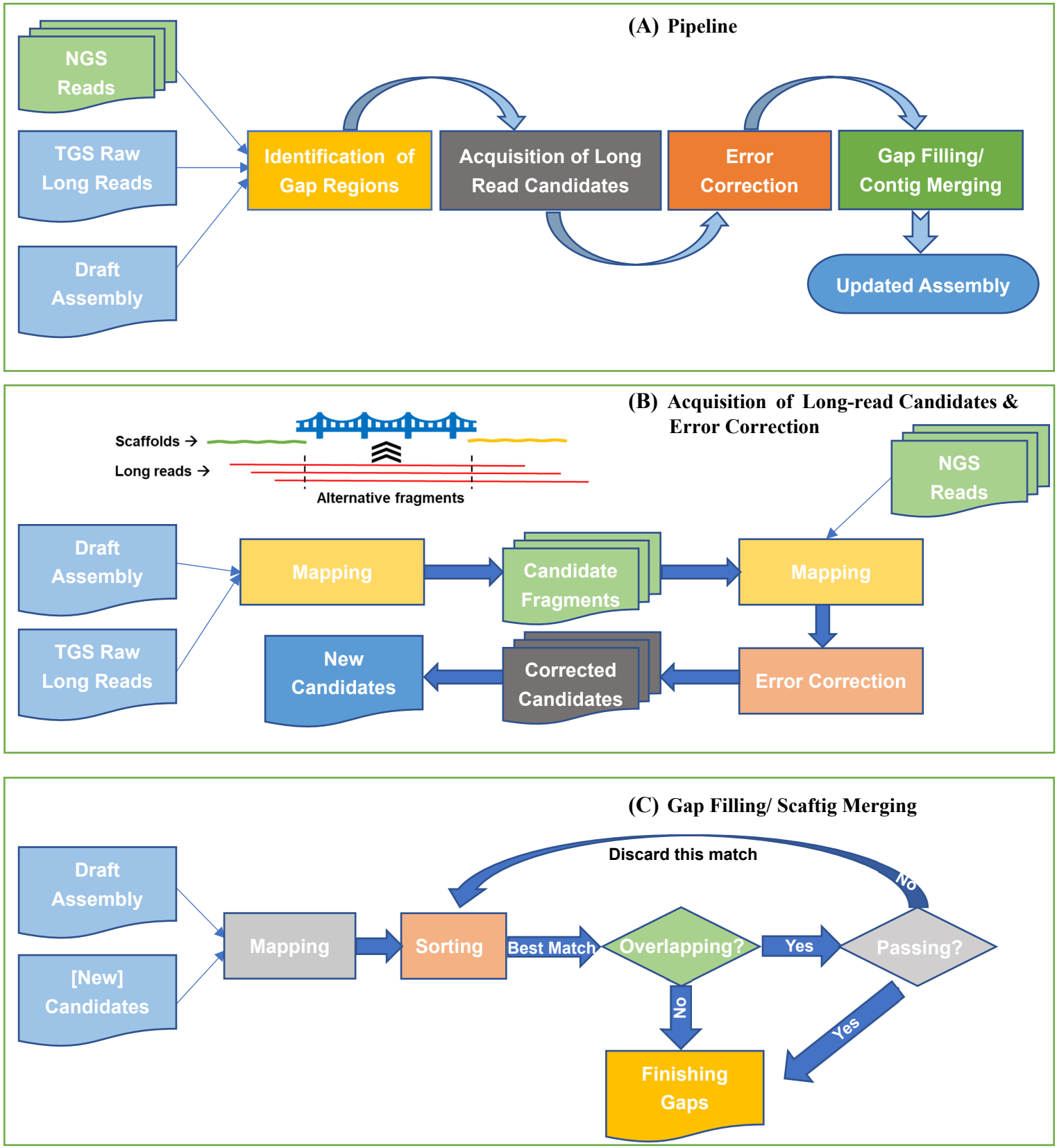
All datasets were run with 42 threads. Note that the peak memory consumption by Pilon or Racon is not counted. The higher speed of runs using the PacBio HiFi dataset mainly originates from the usage of Racon to correct fragments with long reads. Note that QUAST accepts <10 continuous N's in the scaftig.

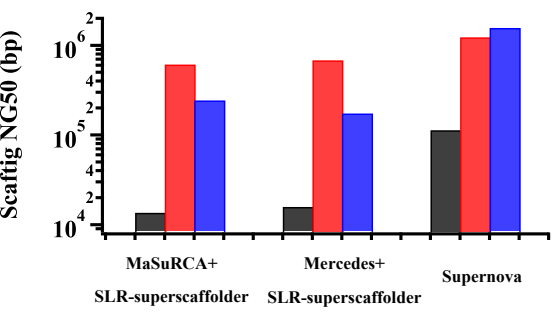
Table 3. Improved assemblies in the MHC region by TGS-GapCloser.

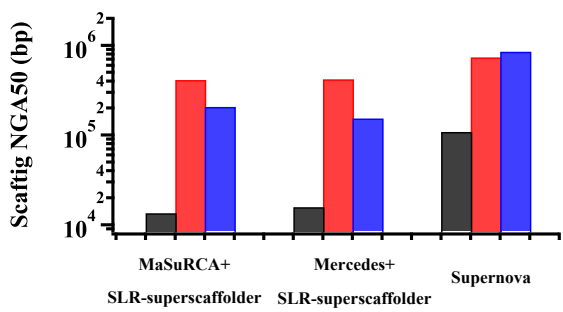
	MaSuRCA+SLR-superscaffolder+TGS-GapCloser		**Mercedes+SLR-superscaffolder+TGS-GapCloser**		**Supernova+TGS-GapCloser**		**Ref.（33）**	
	draft	updated	draft	updated	draft	updated	Rel3	Rel5
No. of scaffolds (>1kb)	2	2	3	3	1	1	/	/
No. of Scaftigs/contigs (>1kb)	339	31	271	26	76	12	7	1

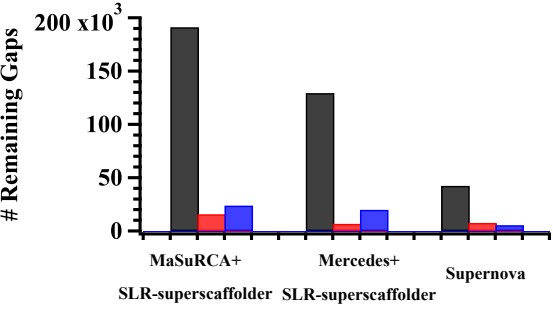
Non-N bases (bp)	5,293,785	5,907,069	4,134,156	5,445,373	5,831,980	5,988,090	5,739,339	5,628,041
No. of gaps	343	31	268	23	81	16	/	/
Scaffold NG50 (bp)	3,400,000	3,400,000	4,400,000	4,400,000	6,000,000	6,000,000	/	/
Scaffold NGA50 (bp)	232,462	396,537	182,662	429,613	649,591	534,616	/	/
Scaftig/contig NG50 (bp)	17,483	324,807	12,244	450,213	110,320	980,326	3,007,673	5,628,041
Scaftig/contig NGA50 (bp)	16,630	199,405	11,901	321,624	94,556	380,102	49,485	52,555
Genome Fraction (%)	82.801	92.623	67.869	85.609	93.887	95.292	62.521	59.855
No. of misassemblies	11	25	13	22	15	17	20	53
No. of local misassemblies	34	101	11	122	29	42	546	484

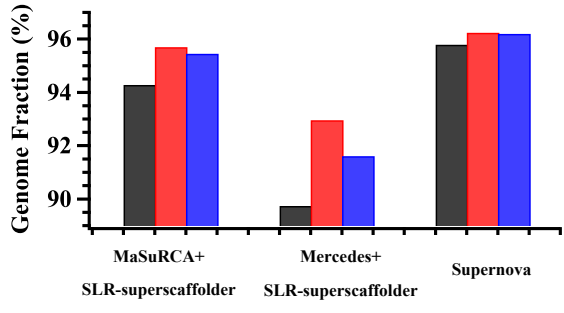
The statistical results were generated by QUAST. Note that QUAST accepts <10 continuous N’s in the scaftig/contig.

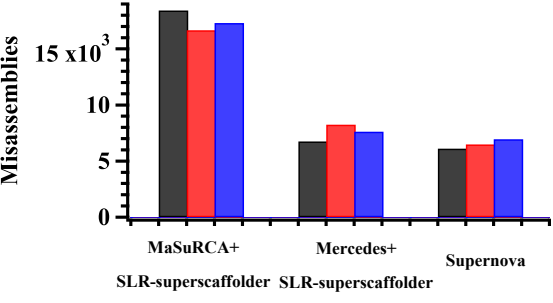

(A) Pipeline
NGS Reads
TGS Raw Long Reads
Draft Assembly
Identification of Gap Regions
Acquisition of Long Read Candidates
Error Correction
Gap Filling/ Contig Merging
Updated Assembly
(B) Acquisition of Long-read Candidates & Error Correction
Scaffolds →
Long reads →
Alternative fragments
NGS Reads
Draft Assembly
TGS Raw Long Reads
Mapping
Candidate Fragments
Mapping
New Candidates
Corrected Candidates
Error Correction
(C) Gap Filling/ Scaftig Merging
Draft Assembly
[New] Candidates
Mapping
Sorting
Best Match
Overlapping?
Yes
Passing?
Discard this match
No
No
Yes
Finishing Gaps

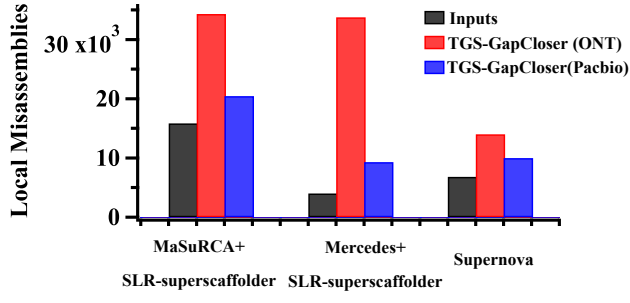

Scaftig NG50 (bp)
MaSuRCA+ SLR-superscaffolder
Mercedes+ SLR-superscaffolder
Supernova


Scaftig NGA50 (bp)
MaSuRCA+ SLR-superscaffolder
Mercedes+ SLR-superscaffolder
Supernova


Remaining Gaps
200 x10^3
150
100
50
0
MaSuRCA+ SLR-superscaffolder
Mercedes+ SLR-superscaffolder
Supernova


Genome Fraction (%)
96
94
92
90
MaSuRCA+ SLR-superscaffolder
Mercedes+ SLR-superscaffolder
Supernova


Misassemblies
15 x10^3
10
5
0
MaSuRCA+ SLR-superscaffolder
Mercedes+ SLR-superscaffolder
Supernova


Local Misassemblies
30 x10^3
20
10
0
MaSuRCA+ SLR-superscaffolder
Mercedes+ SLR-superscaffolder
Supernova
Inputs
TGS-GapCloser (ONT)
TGS-GapCloser(Pacbio)

February 2nd, 2020

Dear *GigaScience* Editor,

It is our great pleasure to submit the enclosed manuscript for your consideration of publishing on *GigaScience*. The brief of our submission is:

Title: TGS-GapCloser: A fast and accurate gap closer for large genomes with low coverage of error-prone long reads

Authors: Mengyang Xu, Lidong Guo, Shengqiang Gu, Ou Wang, Rui Zhang, Guangyi Fan, Xun Xu, Li Deng & Xin Liu

Manuscript type: Technical Notes

The application of third-generation sequencing technology has brought a revolution in life and biomedical fields, but suffers the problem of expense and accuracy. We developed a gap-closing software tool, TGS-GapCloser that utilizes only low depth of single molecule sequencing long reads to discover the complicated areas in large genomes that short reads cannot reach. We demonstrate that TGS-GapCloser improves the continuity, completeness of human genome and gingko ultra large genome without loss of accuracy. Comparing with mainstream long-read gap-closing tools, it can complete more gaps in input assemblies, but run incredibly faster. We believe that the TGS-GapCloser-based hybrid assembly strategy comprehensively employs assembly information to the utmost extent from various sequencing platforms, and improves the quality of downstream analysis of gene annotation. The low-depth requirement of expensive long reads makes this approach more costly effective and suitable for the community with small budgets, and readily enlarges the “big data” database.

All authors have declared that they have no competing interests, approved the contents of the manuscript and agreed with the submission to *GigaScience*. This manuscript is not under consideration for publication elsewhere and has been preprinted in bioRxiv only. We look forward to hearing from you soon. Your kind assistance on this is greatly appreciated!

Sincerely yours,

Mengyang Xu, Ph.D.

BGI-Research

BGI-Qingdao, BGI-SZ, Qingdao 266555, China