LRScaf: Improving Draft Genomes Using Long Noisy Reads

Mao Qin[1,*], mqin@outlook.com

Shigang Wu[1], 495402193@qq.com

Alun Li[1], 343010781@qq.com

Fengli Zhao[1], zw301987@163.com

Hu Feng[1], fenghu01@126.com

Lulu Ding[1], lulu.ding1@outlook.com

Yuxiao Chang[1], changyuxiao@cass.cn

Jue Ruan[1,*], ruanjue@caas.cn

[1] Genome Analysis Laboratory of the Ministry of Agriculture, Agricultural Genomics Institute at Shenzhen, Chinese Academy of Agricultural Sciences, Shenzhen, Guangdong 518124, China.

[*] To whom correspondence should be addressed.

Abstract

Background: The advent of Third Generation Sequencing (TGS) technologies opens the door to improve genome assembly. Long reads are promised to enhance the quality of fragmental draft assemblies constructed from Next Generation Sequencing (NGS) technologies. To date, a few of algorithms, *i.e.*, SSPACE-LongRead, OPERA-LG, SMIS, npScarf, DBG2OLC, Unicycler, and LINKS, have been released that are capable of improving draft assemblies. However, hybrid assembly on large genomes is still challenging.

Results: We develop a scalable and computationally efficient scaffolder, Long Reads Scaffolder (LRScaf), that is capable of boosting assembly contiguity to a large extent using long reads. In our experiment, our method significantly improves the contiguity of human draft assemblies, increasing the NG50 value of CHM1 from 127.5 Kb to 10.4 Mb using 20-fold coverage PacBio dataset and the NG50 value of NA12878 from 115.7 Kb to 17.4 Mb using 35-fold coverage Nanopore dataset. The run time for the scaffolding procedure using LRScaf is the shortest in all cases of our experiment. Compared with the run time of SSPACE-LongRead, LRScaf is faster 300 times for *S. cerevisiae* and 2,300 times for *D. melanogaster*. The peak RAM of LRScaf, by contrast, is more efficient than LINKS in our test. For the rice case, the peak RAM of LINKS (877.72 Gb) is about 196 times higher than LRScaf. For the experiment of human assembly, the peak RAM of LINKS is beyond the capacity of system memory (1 Tb) whereas LRScaf takes 20.28 and 41.20 Gb on CHM1 and NA12878 datasets.

Conclusions: The new method, LRScaf, yields the best or at least moderate contiguity and accuracy of scaffolds in the shortest run time compared with the state-of-the-art methods.

Furthermore, it offers a new opportunity for the hybrid assembly of large genomes.

Keywords: Assembly, Scaffolding, SMRT, ONT, Long Reads,

Background

With the advent of Next Generation Sequencing (NGS) technologies, the genomics community has made significant contributions to *de novo* assembling genomes. Despite that many studies and tools are aimed at reconstructing NGS data into complete *de novo* assemblies of genomes, this goal is difficult to achieve because of intrinsic limitation of NGS data, *i.e.*, read lengths are shorter than most of the repetitive sequences [1]. The existence of repeats makes it difficult to reconstruct complete genomes instead of generating a large set of contiguous sequences (contigs) even when the sequencing coverage is high [2]. Thus, attention is focused on the so-called genomic scaffolding procedure, which aims at reducing the number of contigs by using fragments of moderate lengths whose ends are sequenced (double-barreled data) [3,4]. Nevertheless, major genomic regions still hinder genomic assemblies because of, primarily, large-size repeat and low coverage. In response, Third Generation Sequencing (TGS) technologies have been developed. TGS sheds light on different alternatives to solve genome assembly problems by offering very long reads, *e.g.*, the Single Molecule Real Time (SMRT) sequencing technology of Pacific Biosciences® (PacBio)delivers read lengths of up to 50 Kb [5] and the nanopore sequencing technology of Oxford Nanopore Technologies® (ONT) delivers read lengths which are greater than 800 Kb [6]. These long reads suffer from high sequencing error rates, however, which necessitates high coverage during the genome assembly [7]. In

addition, TGS technologies have a higher cost per base than NGS methods. Consequently, long reads are more commonly used for scaffolding draft assemblies generated from NGS data than for *de novo* assembly [8].

The process of genome assembly is typically divided into two major steps. The first step is to piece overlapping reads together into contigs which is commonly done using the *de Bruijn* or overlap graph [1]. The second step is to assemble scaffolds, consisting of ordered sequences of oriented contigs with estimated distances between them. Scaffolding, which was first introduced by Huson [3], is a critical part of the genome assembly process, especially for NGS data. Yet, scaffolding is a research area that remains largely open because of the NP-hard complexity [9]. By using paired-end and/or mate-pair reads linking information, a number of standalone scaffolders, *e.g.* Bambus [4], MIP [10], Opera [11], SCARPA [12], SOPRA [13], SSPACE [14], BESST [15], and BOSS [16], have been developed. Nevertheless, a recent comprehensive evaluation showed that scaffolding was still computationally intractable and required better quality large insert-size pair read libraries than presently available [17]. As TGS technologies are likely to offer longer reads than the lengths of the most common repeats, these technologies are capable of drastically reducing and solving the complexity caused by repeats. Considered the pros and cons of NGS and TGS data, a hybrid assembly approach that assembled draft genomes using TGS data was proposed [18]. The core strategy of this approach is: 1) long reads are mapped onto the contigs using a long-read mapper (*e.g.* BLASR [19] or minimap [20,21]); 2) examining alignment information, long reads that span more than one contig are identified and their linking relationship is stored in a data structure; 3) the last step is to clean up the structure by removing redundant and error-prone links, calculate distances

between contigs, and build scaffolds using links information.

Based on the hybrid assembly strategy, AHA [18] was the first standalone hybrid scaffolder and was part of the SMRT analysis software suite. As AHA was designed for small genomes and had limitations on the input data, it was not suitable for large genomes. To ensure that scaffolds were as contiguity as possible, AHA performed 6 iterations by default, thus increasing the run time. SSPACE-LongRead [22] produced the final scaffolds in a single iteration and, therefore, had a significantly shorter run time than AHA. Nevertheless, SSPACE-LongRead had somewhat lower assembly accuracy than AHA. Despite being designed for large eukaryotic genomes, SSPACE-LongRead was unpractical because of its intensive run time. LINKS [23] opened a new door to build linking information between contigs. The algorithm used the long interval nucleotide K-mer without computational alignment and reads correction step, but its memory usage was a concern. OPERA-LG [24] provided an exact algorithm for large and repeat-rich genomes. Its main limitation was that it required significant mate-pair information to constrain the scaffold graph and report an optimized result. OPERA-LG was not directly designed for TGS data, and to construct scaffold edges and link contigs together into scaffolds, OPERA-LG needed to be modified by simulated and grouped mate-pair relationship information from long reads. Recent studies, such as SMIS (Available from http://www.sanger.ac.uk/science/tools/smis), npScarf [25], DBG2OLC [26] and Unicycler [27], have been reported based on the hybrid assembly strategy. However, these tools have not been thoroughly assessed for different genome sizes, especially large genomes.

Here we present a Long Reads Scaffolder (LRScaf) to improve draft genomes using TGS data. The input to LRScaf is given by a set of contigs and their alignments over SMRT or ONT

long reads. We compare our method with the state-of-the-art tools on real and synthetic datasets. All the methods tested improve the contiguity of pre-assembled genomes. Our method yields the best assembly metrics and contiguity for pre-assembled genomes of *E. coli*, *S. cerevisiae*, *D. melanogaster*, and *H. sapiens*. More importantly, however, our method consistently returns the most accurate scaffolds and has the shortest run time. Especially, LRScaf significantly improves the contiguity of human draft assemblies, increasing the NG50 value of CHM1 from 127.5 Kb to 10.4 Mb using 20-fold coverage PacBio dataset and the NG50 value of NA12878 from 115.7 Kb to 17.4 Mb using 35-fold coverage Nanopore dataset. We thus show that LRScaf is a valuable tool for improving draft assemblies in a cost-effective way.

Results and discussion

We performed in-depth analysis on five species, *i.e.*, *E. coli*, *S. cerevisiae*, *D. melanogaster*, *O. sativa*, and *H. sapiens*, to test and compare the performance of LRScaf with that of SMIS, npScarf, DBG2OLC, Unicycler, SSPACE-LongRead, LINKS, and OPERA-LG. The details of datasets are provided in Table 1 and in the Methods section. The NGS datasets for *E. coli* and *S. cerevisiae* are real with 600 and 105 -fold coverages respectively, where the NGS datasets for *D. melanogaster* and *O. sativa* were synthesized using pIRS [28] with 50-fold coverage. The real reads for the two small genomes (*E. coli* and *S. cerevisiae*) were first cleaned and then used to construct draft assemblies using SOAPdenovo2 [29] and SPAdes [30]. The synthetic reads for the two large genomes (*D. melanogaster* and *O. sativa*) were directly used to build draft assemblies using SOAPdenovo2. The draft assemblies of two human lines CHM1 [31] and NA12878 [32] were used to test the performances of all scaffolders for large

genomes using the PacBio and Nanopore datasets. The statistics of draft assemblies are shown in Table 2. Results and assembly metrics obtained after the scaffolding procedure are displayed in Tables 3 and 4.

Draft genome assemblies

We used SPAdes to construct draft assemblies for two small genomes (*E. coli* and *S. cerevisiae*) with the "careful" parameter option. Draft assemblies for these two small genomes were also constructed using SOAPdenovo2. In addition, SOAPdenovo2 was used to construct draft assemblies for *D. melanogaster* and *O. sativa*, whose synthetic reads were available. We used the optimal k-mer values for the draft assemblies constructed by SOAPdenovo2 with 51 (*E. coli*), 59 (*S. cerevisiae*), 61 (*D. melanogaster*), and 73 (*O. sativa*). These values were selected based on assembled genome size, number of contigs, and genome contiguity.

The statistics of draft assemblies for *E. coli*, *S. cerevisiae*, *D. melanogaster*, *O. sativa*, and *H. sapiens* are shown in Table 2. For *E. coli*, the draft-genome size obtained using SOAPdenovo2 is 4.6 Mb distributed over 728 contigs, yielding an assembled genome fraction of 98 % and an N50 value of 40.0 Kb. SPAdes yields a draft-assemblies size of 4.6 Mb with 242 contigs, a genome fraction of 98 %, and an N50 value of 133.2 Kb. The draft-assemblies size generated by ABySS is 5.2 Mb with 69 contigs, a genome fraction of 110 %, and an N50 value of 177.6 Kb. For *S. cerevisiae*, the draft-genome size obtained using SOAPdenovo2 is 12.1 Mb with 6,961 contigs, providing a genome fraction of 99 % and an N50 value of 20.0 Kb. SPAdes generates a draft-assemblies size of 11.8 Mb with 2,254 contigs, yielding a genome fraction of 97 % and an N50 value of 107.9 Kb. The draft-assemblies size constructed by Celera Assembly

is 15.0 Mb with 6,953 contigs, a genome fraction of 124 %, and an N50 value of 49.2 Kb. For *D. melanogaster*, the draft-assemblies size constructed by SOAPdenovo2 is 118.1 Mb with 45,480 contigs, a genomic fraction of 98 %, and an N50 value of 111.0 Kb. The draft-genome size for *O. sativa* is 346.2 Mb and it contains 257,801 contigs, yielding a genomic fraction of 92 % and an N50 value of 19.0 Kb. The size of the draft genome of CHM1 is 2.8 Gb distributed over 40,906 contigs, and it has a genomic fraction of 93 % and an N50 value of 140.0 Kb where the draft-assemblies size of NA12878 is 3.1 Gb with 858,918 contigs, yielding a genome fraction of 102 % and an N50 value of 179.8 Kb.

The depth of coverage is an important factor in *de novo* genome assembly. The genome contiguity and completeness obtained are not only determined by the depth of coverage, however, but also by the method's ability to overcome complex genome structures, *e.g.* repetitive regions. *E. coli* is the smallest genome and has the highest coverage (more than 600-fold of NGS reads) among the genomes included in this study. However, the assembly contiguity is still fragmental. As the genome gets larger and more complex, draft assemblies become increasingly fragmental unless auxiliary technologies are included in the assembly process. Consequently, the inclusion of large insert-size mate-pair libraries, Hi-C [33], optical-mapping data [34] and long reads is important to overcome large repeats and to assist the scaffolding procedure.

Scaffolding on SMRT long reads

In this study, we used long reads of SMRT datasets for *E. coli*, *S. cerevisiae*, *D. melanogaster*, *O. sativa*, and *H. sapiens* to assess the performances of seven state-of-the-art scaffolders (*i.e.*,

SSPACE-LongRead, LINKS, OPERA-LG, SMIS, npScarf, Unicycler, and DBG2OLC) and our LRScaf (See Table 1). The median lengths of SMRT long reads for 5 organisms are 8.7 Kb, 4.6 Kb, 19.6 Kb, 3.4 Kb, and 1.6 Kb, respectively. And the longest reads are 41.3 Kb, 27.6 Kb, 33.6 Kb, 24.4 Kb, and 208.6 Kb, respectively. The coverages of SMRT long reads are 20.1-fold (*E. coli*), 20.7-fold (*S. cerevisiae*), 18.9-fold (*D. melanogaster*), 11.7-fold (*O. sativa*), and 20.0-fold (*H. sapiens*). The distributions of read length show that the SMRT long reads approximate normal distributions (See Suppl. Fig. 1). The SMRT long reads of *D. melanogaster* were filtered for the FALCON assembler [35], which resulted in an increased average read length. QUAST [36] was used to assess draft assemblies after the scaffolding procedure. The released version 4.5 of QUAST was failed to assess human assemblies, and, therefore, we used the dev-5.0 version to evaluate the corresponding assembly metrics.

All scaffolders reduce the number of contigs and improve assemblies contiguity (See Table 3 and Suppl. Tables 1 and 2). Whereas SSPACE-LongRead, SMIS, Unicycler, and LRScaf reconstruct the genome for *E. coli* into a complete single chromosome, LINKS, OPERA-LG, npScarf, and DBG2OLC fail to do that. In addition, Unicycler significantly reduces the numbers of contigs. For the 1, 5, and 10 -fold coverages, the performances of scaffolders tested show similar results on the 20-fold coverage where the assemblies contiguity of SSPACE-LongRead, SMIS, Unicycler, and LRScaf are better than that of LINKS, OPERA-LG, npScarf, and DBG2OLC (See Suppl. Tables 1 and 2). For *S. cerevisiae*, the npScarf method yields the best NG50 value (665.8 Kb) and Unicycler generates the best NA50 value (284.1 Kb). SSPACE-LongRead, LINKS, OPERA-LG, npScarf, and LRScaf yield the longest sequence (1.0 Mb). For the 1, 5, and 10 -fold coverages, SSPACE-LongRead yields the best assemblies

contiguity (NG50) in 5 out of 6 cases and OPERA-LG, npScarf, and LRScaf yield the best NG50 in 1 out of 6 cases (See Suppl. Table 1). Based on draft assemblies generated by SOAPdenovo2 using 20-fold coverage, SSPACE-LongRead and LRScaf yield the best NG50 and NA50 value respectively and generate the longest sequence (See Suppl. Table 2). For *D. melanogaster*, SSPACE-LongRead yields the best NG50 value (6.6 Mb) and LRScaf with BLASR produces the best NA50 value (5.2 Mb). SSPACE-LongRead and LRScaf construct the longest sequence of 19.6 Mb. For *O. sativa*, DBG2OLC significantly reduces the number of sequences and produces the best NG50 value (94.5 Kb) and NA50 value (64.9 Kb), and the longest sequence (794.7 Kb). SSPACE-LongRead is excluded from this assessment because it exceeds the 3 weeks' run time limit. For *H. sapiens* CHM1, LRScaf with minimap2 yields the best NG50 value (10.4 Mb) and NA50 value (10.7 Mb,), and the longest sequence (45.0 Mb). The run time of SSPACE-LongRead, SMIS, and npScarf exceeds the time limit, and LINKS exceeds our system's memory capacity of 1 Tb. Thus, these scaffolders are excluded from the test on the *H. sapiens* CHM1 genome. As evident from our experiments, the run time and the memory usage for these scaffolders become significant concerns for the large and complex genomes. DBG2OLC is recommended to use SparseAssembler (Available from: https://github.com/yechengxi/SparseAssembler) to construct draft assemblies for hybrid assembly. This might be the reason for the assembly genome size generated by DBG2OLC is smaller than what the other scaffolders yield, especially for the *H. sapiens*. To summarize, LRScaf yields the best or, at least, moderate assembly metrics when compared with other scaffolders on SMRT long reads.

Scaffolding on ONT long reads

We used the ONT long reads datasets for *E. coli*, *S. cerevisiae*, and *H. sapiens* to assess the performances of scaffolders tested (See Table 4). Because of lack of NGS data, OPERA-LG and Unicycler were excluded from this assessment. For the two small genomes, the ONT long-reads datasets were published in LINKS, including 3 of *E. coli* (FULL, ALL and RAW datasets with 4.7, 34.0, and 66.5 -fold coverages, respectively) and 2 of *S. cerevisiae* (NANOCORR and RAW datasets with 43.6 and 198.2 -fold coverages). We used the *H. sapiens* NA12878 dataset with 35.0-fold coverage as the large genome for this test. The best median and longest length of reads are 6.1 Kb and 1.5 Mb respectively (See Table 1). The distributions of read length show that ONT long reads approximate bimodal distributions with a long tail (See Suppl. Fig. 2). The median length of ONT reads is approximately equal to that of SMRT, but the longest length of ONT reads is significantly longer than that of SMRT datasets. QUAST (Version 4.5) was used to assess draft assemblies and scaffolded assemblies for *E. coli* and *S. cerevisiae*. And QUAST (Dev-5.0 version) was used to evaluate the corresponding assembly metrics for *H. sapiens*.

All scaffolders decrease the number of contigs and improve genome contiguity (See Table 4). The number of contigs for the *E. coli* draft assemblies is 69 with an NG50 value of 179.7 Kb. For the FULL dataset, LRScaf with BLASR yields the best NG50 value (921.6 Kb) and NA50 value (485.2 Kb), and the longest sequence (1.1 Mb). SMIS generates the best NG50 value (992.2 Kb) and NA50 value (618.4 Kb), and the longest sequence (1.2 Mb) for the ALL dataset where LRScaf with BLASR yields similar performance (NG50: 922.5 Kb, NA50: 616.5 Kb, and the longest sequence 1.1 Mb). Whereas SMIS produces the best numbers for the RAW

dataset (NG50: 928.1 Kb, NA50: 879.1 Kb, and the longest sequence 1.2 Mb), LRScaf with BLASR yields very similar metrics. For *S. cerevisiae*, the number of contigs for the draft assemblies is 6,953 with an NG50 value of 58.8 Kb. For the NANOCORR dataset, the npScarf method yields the best NG50 value and the longest sequence (559.4 Kb and 1.5 Mb, respectively), and SMIS produces the best NA50 value (250.7 Kb). The npScarf scaffolder also produces the best metrics (NG50: 578.3 Kb, NA50: 250.0 Kb, and the longest sequencing 1.6 Mb) for the RAW dataset. For the NA12878 dataset, LRScaf with minimap2 significantly improves the contiguity of the draft assemblies and yields the best NG50 and NA50 values (17.4 Mb and 13.6 Mb, respectively). LRScaf with BLASR produces the longest sequence (71.6 Mb). All the other scaffolders are similar to the assessment using the PacBio dataset and exceed either the time limit (3 weeks) or the memory capacity of system (1 Tb). In addition, DBG2OLC is not successful to scaffold draft assemblies generated by DISCOVAR. This is as expected where DBG2OLC is recommended to use SparseAssembler as its NGS Assembler for hybrid assembly. Compared with the results obtained using the SMRT datasets, none of the scaffolders could assemble *E. coli* into a single chromosome and the contiguity of *S. cerevisiae* is more fragmented. Although all scaffolders show certain improvement in our experiment, the application of the ONT data is still challenging. A recent study showed that the NA12878 genome was assembled with an NG50 value of about 6.5 Mb using pure 35-fold ONT data [6]. Our experiments, however, show that it is possible to significantly improve assembly contiguity to 17.4 Mb where it is similar to the PacBio human case. To summarize, LRScaf yields either the best or similar assembly metrics using long reads of ONT compared with the other scaffolders.

Computational performance and accuracy analysis

The assembly metrics are undoubtedly the most concerning matters to biologists and bioinformaticians. Nevertheless, from a practical point of view, the run time limits software applications. SSPACE-LongRead and OPERA-LG use BLASR as their default TGS mapper for construction of joints between contigs. The npScarf software uses BWA [37] as its default mapper. LINKS, SMIS, Unicycler, and DBG2OLC use its built-in algorithms to build joints between contigs. To enable a direct comparison with SSPACE-LongRead and OPERA-LG, our LRScaf supports BLASR. Nevertheless, it also supports a faster TGS mapper minimap (Versions 1 and 2), which enables a significant reduction for the total run time of the scaffolding procedure. LRScaf is the fastest scaffolder for all the cases using SMRT long reads. LRScaf reduces the run time more than 300 times compared with SSPACE-LongRead and more than 3,900 times compared with Unicycler for *S. cerevisiae* (See Table 3). As the genome gets larger, the advantage of shorter run time becomes more important. In *D. melanogaster*, LRScaf is 2,300 times faster than SSPACE-LongRead and 2,550 times faster than SMIS. In *O. sativa*, LRScaf is 1,276 times faster than SIMS. We have no number on how much LRScaf is faster than SSPACE-LongRead because the latter exceeds the time limit (3 weeks). For *H. sapiens*, SSPACE-LongRead, SMIS, and npScarf exceed the time limit (3 weeks). For the ONT datasets, LRScaf is also the fastest scaffolder. LRScaf is more than 131 times faster than SSPACE-LongRead on the FULL dataset for *E. coli*. As the dataset grows larger, the advantage becomes more significant. LRScaf is 714 times faster on the ALL dataset for *E. coli*, 603 times faster on the RAW dataset for *E. coli*, and 1,408 times on the RAW dataset for *S. cerevisiae* than

SSPACE-LongRead. LINKS skips the all-to-all alignment step and is faster than SSPACE-LongRead in all cases. Nevertheless, the memory usage of LINKS is of concern and it might be alleviated by further improvement of the data-structure. Although the peak RAM usage for LRScaf is higher than that of OPERA-LG on small genomes, our experiments show that the memory usage of LRScaf is practical even for large and complex genomes where the peak RAM for LRScaf is not over 30 Gb on CHM1 PacBio dataset and 80 Gb on NA12878 ONT dataset.

Reducing the number of misassemblies is important because misassemblies are likely misinterpreted as true genetic variations [38,39]. For the SMRT datasets, SSPACE-LongRead and LRScaf yield the fewest number of misassemblies (1) among the scaffolders based on draft assemblies for *E. coli* generated by SOAPdenovo2 (See Suppl. Table 2). Unicycler produces the fewest number of misassemblies for *E. coli* (1) and *S. cerevisiae* (17) based on draft assemblies constructed by SPAdes where LINKS and npScarf yields the maximum number of misassemblies for *E. coli* (13) and *S. cerevisiae* (105) respectively (See Table 3). LRScaf yields the fewest number of misassemblies for *D. melanogaster* (15) and *O. sativa* (455) where DBG2OLC and OPERA-LG produce the maximum number of misassemblies for *D. melanogaster* (2,393) and *O. sativa* (2,604) respectively (See Table 3). For *H. sapiens*, we have no number on how many the number of misassemblies for the other scaffolders because all of them are failed to scaffold the draft assemblies. For the ONT datasets, the draft assemblies for *E. coli*, *S. cerevisiae*, and *H. sapiens* contain 5, 19, and 336 misassembled contigs, respectively, and none of the scaffolders significantly increases the number of misassemblies (See Table 4). LRScaf with minimap2 outputs the fewest number of misassemblies on the *E. coli*, *S. cerevisiae*

(RAW data). LRScaf with minimap outputs the fewest number of misassemblies on *H. sapiens*. SMIS yields the fewest number of misassemblies for the *S. cerevisiae* NANOCORR dataset. SSPACE-LongRead yields the maximum number of misassemblies (147) on the RAW dataset for *S. cerevisiae*. In summary, LRScaf introduces a new strategy for keeping valid alignments (See Methods section) and produces fewer misassemblies than most of the other scaffolders. Moreover, LRScaf with minimap2 significantly reduces the run time of scaffolding procedure without increasing the number of misassemblies. Based on the SMRT and ONT performances, we recommend that LRScaf is used with BLASR on small genomes and with minimap on large genomes.

Conclusion

In this work, we present a novel program for scaffolding draft assemblies using noisy TGS long reads information and compare our algorithm with the previous methods. The majority of the draft assemblies constructed using NGS data is fragmented and influenced by repeats. The disadvantage of long reads is that they contain significantly more errors than first- and second-generation sequencing technologies. Nevertheless, we successfully use long reads to build links between contigs, overcome repetitive regions, and improve genome contiguity. We propose a new strategy to filter inaccurate alignments so that these false alignments do not propagate through the scaffolding process. For the assessments on SMRT long-read datasets covering 5 organisms, our method shows significant improvements over the state-of-the-art scaffolders. The primary benefits of LRScaf over these scaffolders are that it yields the fewer number of misassemblies and reduces the run time, yet it retains the best or, at least, average assembly

metrics. These improvements are especially useful for large and complex genomes. For the assessments on ONT long-read datasets for 3 organisms, our method shows significant improvements over the previous algorithms. Our method keeps the best or, at least, average assembly metrics and the shortest run time. In addition, our method has the fewest number of misassemblies in most of the cases. As studied genomes keep getting larger and more complex, the run time and the memory usage for the analysis software are becoming increasingly important to biologists and bioinformaticians. Our method is designed with reduction of the run time and the memory usage in mind and is, thus, much faster than other scaffolders and requires only moderate memory usage. Identification of misassembled contigs is also important, however, because any misassembled sequences are propagated into the next step during biological analysis. Most state-of-the-art scaffolders lack functions for identification of misassembled contigs. In addition, misassemblies might be introduced during the scaffolding procedure. Consequently, to limit the number of misassembled scaffolds, our method incorporates a validation algorithm that checks the links information between contigs. As checking and correcting misassemblies from draft assemblies is important, we are planning to use long read information to achieve and integrate these functions in a future version of LRScaf.

In the past decade, worldwide collaboration has led to several projects, aiming at improving the understanding of species biology and evolution. Examples of such projects are the i5k [40], which provides the genomes of 5,000 species of insects, and the Bird 10,000 Genomes (B10K) [41]. However, a substantial fraction of genomes with short contiguity hinder downstream analysis. Our result shows that TGS data is capable of effectively improving draft assemblies

and LRScaf is a valuable tool for improving draft assemblies in a cost-effective way.

Methods

Alignment of TGS long reads

LRScaf was designed to separate the mapping and scaffolding procedures. Hence, during the mapping procedure, we set the number of processes to 48 and kept the default values for all other parameters using BLASR and minimap (Version 1 and 2). LRScaf supports the default alignment format of these mappers.

Validating alignment

The high error rate is a serious disadvantage of TGS long reads. Thus, a large fraction of the alignments is incorrect and needs to be filtered out. We developed a validation model to validate each alignment (See Figure 1). The model partitioned each long read into three regions (R1, R2, and R3) separated by two points (P1 and P2). Considered the alignment start (S) and end (E) loci in the contig, there were six different combination sets in R, *i.e.*, $R \in \{(S\ in\ R1, E\ in\ R1), (S\ in\ R1, E\ in\ R2), (S\ in\ R1, E\ in\ R3), (S\ in\ R2, E\ in\ R2), (S\ in\ R2, E\ in\ R3), (S\ in\ R3, E\ in\ R3)\}$. We also defined the distal length of a contig to the start or end alignment loci as the over-hang length of the contig. Taken both the alignment region and the over-hang length into account, the valid alignment satisfied: 1) (S in R1, E in R1) with the right over-hang length not exceeding the constraints; 2) (S in R1, E in R2) with the right over-hang length not exceeding the constraints; 3) (S in R2, E in R2) with the two end over-hang length not exceeding the constraints; 4) (S in R2, E in R3) with the left over-hang length not exceeding the

constraints; 5) (S in R3, E in R3) with the left over-hang length not exceeding the constraints. An alignment was filtered out if a long read was entirely covered by a contig (S in R1, E in R3), *i.e.*, the contig contained the long read. After this procedure, the remaining alignments were considered to be valid for the scaffolding procedure.

Repeat identification

Repetitive sequences complicate the genome assembly. Thus, such sequences were masked in our approach. First, based on the uniform coverage of TGS data, we identified and removed repeats by the coverage of reads. In the calculation of reads coverage, long reads that covered the entire contig were counted. Then we computed the mean coverage and the standard deviation among the set of contigs. Any contig coverage that was larger than the threshold coverage, which was set to $\mu_{cov} + 3 \times s.d._{cov}$, was considered to be a repeat and the corresponding contig was removed from the next step of the analysis.

Constructing links and edges

A long read may have multiple mappings because of repeats and high sequencing error rate. Figure 2 describes how links are built between contigs from the validated alignments. This process had two constraints on orientation and distance. Four strand combination sets S were used between contigs to constrain orientation, *i.e.*, $S \in \{s_1: (+,+),\ s_2: (+,-),\ s_3: (-,+),\ s_4: (-,-)\}$. We defined the orientation between contigs as $O(c_i, c_j) = max(s)$. The probability that the internal distance e between two contigs lies outside the range $[\mu_{is} - 3 \times \sigma_{is}, \mu_{is} + 3 \times \sigma_{is}]$ was less than 5%, because e approximately follows a normal distribution

$N(\mu_{is}, \sigma_{is})$. If e lay outside the range $[\mu_{is} - 3 \times \sigma_{is}, \mu_{is} + 3 \times \sigma_{is}]$, it was considered to be abnormal and the linking information was removed. Any long reads linking a contig to itself at different loci were also removed. After validating two constraints on links between contigs, we introduced an edge to represent a bundle of links that jointed two contigs using quadruple parameters $E(c_i, c_j) = (n, \overline{\mu_{is}}, \overline{\sigma_{is}}, o)$. Here, *n* was the number of remaining links considered as the weight of the edge, $\overline{\mu_{is}}$ was the mean internal distance for the remaining links, $\overline{\sigma_{is}}$ was the standard deviation of the internal distances for the remaining links, and *o* was the orientation strand between contigs.

Graph construction and simplification

In this step, LRScaf constructed a scaffold graph $G(V, E)$ similar to the string graph formulation. The vertex set *V* represented the end of the contigs and the edge set *E* represented the linkage implied by long reads between ends of two contigs with weight and orientation function assigned to each edge. The ends of each contig were annotated by their ID with a forward strand (+). Used this node concept, there were 4 types of edges in the graph, *i.e.*, (+, +) joining the forward strands of both contigs, (+, -) joining the forward strand of the first contig with the reversed strand of the second contig, (-, +) joining the reversed strand of the first contig with the forward strand of the second contig, and (-, -) joining the reversed strands of both contigs. After the edges-construction step, we accounted for the majority of the sequencing errors by removing all the edges that had a lower number of long reads than the threshold value. Once the edges were cleaned and filtered, we constructed an assembly graph *G*. We only added an edge to *G* if neither of the two nodes comprising the edge was present in *G*. In some cases, *G*

contained some edges of transitive reduction, error-prone and tips. Thus, such edges were deleted and we got the final scaffold graph which we used for further analysis.

Construction of scaffolds

After the repeats identification and the graph simplification steps, most of the contigs were connected in linear stretches on the assembly graph. There were, however, some complex regions that required addition manipulation. We referred to a contig as a divergent node if it linked more than two nodes in the graph (Figure 3). We searched for unique nodes at the end of this complex region and got through this region if there were any long reads that joined two unique nodes. Otherwise, we stopped travelling the graph in the forward direction and switched to the reverse direction. Similarly, the search along the reverse direction of the graph stopped at the end of a linear stretch or at a divergent node. The process was then repeated using an unvisited node as the starting node. The procedure ended after traversing all the unvisited and unique nodes in the graph and outputted all linear paths. Finally, the gap-size between contigs was calculated. If the gap-size value was negative, the contigs were merged into a combined contig, and if the value was positive, a gap was inserted between the contigs (a gap was represented by one or more undefined 'N' nucleotides, depending on gap-size).

Datasets

All tested data were downloaded from published and released datasets (See Table 1). The NGS data of *E. coli* (EAR000206) and *S. cerevisiae* (SRR527545 and SRR527546) were downloaded from EBI and NCBI, respectively, where the NGS data of *D. melanogaster* and *O.*

sativa were simulated from their latest reference genome using pIRS (version 1.11) with parameters -x 50 and -c 0, respectively. The SMRT long reads datasets for 5 organisms were published by PacBio®: SRX669475 and SRX533603 for *E. coli*, SRX533604 for *S. cerevisiae*, SRX499318 for *D. melanogaster*, SRR3743363 for *O. sativa*, and SAMN02744161 for *H. sapiens* (CHM1). We selected the first 20-fold coverage of each SMRT dataset for comprehensively assessing all scaffolders and we chose 3 different coverages, *i.e.* 1, 5 and 10 -fold, for 2 small genomes (*E. coli* and *S. cerevisiae*) to test all scaffolders performances on lower depths. For the long reads of the ONT dataset, datasets were referred to LINKS and *H. sapiens* (NA12878) with ONT-FULL (ERX708228) for *E. coli*, ONT-ALL (ERX708228) for *E. coli*, ONT-RAW (ERX708228) for *E. coli*, ONT-NANOCORR (SRP055987) for *S. cerevisiae*, ONT-RAW (SRP055987) for *S. cerevisiae* and PRJEB23027 for *H. sapiens*, respectively.

Draft assembly procedure

The draft genomes for *E. coli*, *S. cerevisiae*, *D. melanogaster* and *O. sativa* were constructed using SOAPdenovo2 taking genome size and contiguity into account. We use two subroutines for *E. coli*: 1) pregraph with –k 51 and –R parameters and 2) contig with –R parameter. We used two similar subroutines for *S. cerevisiae*: 1) pregraph with –k 29 and –R parameters and 2) contig with –R parameter. The draft assemblies for *D. melanogaster* was also constructed using two subroutines: 1) pregraph with –k 61 and –R parameters and 2) contig with –R parameter. For *O. sativa*, we used the subroutine all with –K 63 –p 24 –d 1 –R –F. The two small genomes (*E. coli* and *S. cerevisiae*) were also assembled by SPAdes with the “careful” parameter. To assess the performances between LINKS and the other scaffolders on the ONT

long read, the draft assemblies for *E. coli* and *S. cerevisiae* were referred to LINKS. The *H. sapiens* CHM1 and NA12878 draft assemblies were from Steinberg *et al*. [31] and Weisenfeld *et al*. [32]. Table 2 lists the statistics for all of the draft assemblies.

System

All analysis was performed on a 1 Tb memory Linux machine with 48 CPUs incorporating Hyper-threading technology.

Source code

LRScaf is written in Java™ and is capable of running on all platforms including Linux, Windows, and Mac if Java Running Environment (JRE) was installed. The source code is available on GitHub (https://github.com/shingocat/lrscaf). We provide a packaged jar file which could be used straight out of the box and the compilation steps for advanced users.

Additional file

Additional file 1: Long Reads (< 30 Kb) Distribution of Pacific Biosciences® SMRT, and Additional file 2: Long Reads (<30 Kb) Distribution of Oxford Nanopore Technologies® nanopore.

List of abbreviations

BLASR: Basic Local Alignment with Successive Refinement; NGS: Next Generation Sequencing; TGS: Third Generation Sequencing; SMRT: Single Molecule Real Time; ONT:

Oxford Nanopore Technologies; LRScaf: Long Reads Scaffolder

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and materials

The datasets generated and/or analyzed in our study are available in the NCBI repository with accession number listed in Table 1. The datasets synthesized using pIRS are available from the corresponding author on request.

Competing interests

The authors declare that they have no competing interests.

Funding

This research was funded by the Dapeng New District Special Fund for Industrial Development [KY20160204, KY20150113]; and the National Key Research and Development Program of China [2016YFC1200600]; and the National Natural Science Foundation of China [31571353]; and the Fundamental Research Funds for Central Non-profit Scientific Institution

[Y2016PT54]; and the Fund of Key Laboratory of Shenzhen [ZDSYS20141118170111640]; and the Agricultural Science and Technology Innovation Program; and the Shenzhen Science and Technology Research Funding [JSGG20160429104101251]; and the Key Forestry Public Welfare Project [201504105]; and the Agricultural Science and Technology Innovation Program Cooperation and Innovation Mission [CAAS-XTCX2016].

Authors' Contributions

MQ conceived and implemented the method. MQ, ALL, FLZ and HF analyze SMRT dataset characters. MQ, SGW, and LLD analyze ONT dataset characters. MQ wrote the article. YXC and JR supervised the study. All authors read and approved the final manuscript.

Acknowledgments

We would like to thank anonymous reviewers for their comments in revising the manuscript.

Author details

Genome Analysis Laboratory of the Ministry of Agriculture, Agricultural Genomics Institute at Shenzhen, Chinese Academy of Agricultural Sciences, No. 7, Pengfei Road, Dapeng District, Shenzhen, China.

References

1. Miller JR, Koren S, Sutton G. Assembly algorithms for next-generation sequencing data. Genomics. 2010;95:315–27.

2. Gnerre S, MacCallum I, Przybylski D, Ribeiro FJ, Burton JN, Walker BJ, et al. High-quality draft assemblies of mammalian genomes from massively parallel sequence data. Proc. Natl. Acad. Sci. 2011;108:1513–8.

3. Huson DH, Reinert K, Myers EW. The greedy path-merging algorithm for contig scaffolding. J. ACM. 2002;49:603–15.

4. Pop M, Kosack DS, Salzberg SL. Hierarchical scaffolding with Bambus. Genome Res. 2004;14:149–59.

5. Eid J, Fehr A, Gray J, Luong K, Lyle J, Otto G, et al. Real-time DNA sequencing from single polymerase molecules. Science. 2009;323:133–8.

6. 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;36:338–45.

7. Chin CS, Alexander DH, Marks P, Klammer AA, Drake J, Heiner C, et al. Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. Nat. Methods. 2013;10:563–9.

8. 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;7:1–12.

9. Chateau A, Giroudeau R. A complexity and approximation framework for the maximization scaffolding problem. Theor. Comput. Sci. 2015;595:92–106.

10. Salmela L, Mäkinen V, Välimäki N, Ylinen J, Ukkonen E. Fast scaffolding with small independent mixed integer programs. Bioinformatics. 2011;27:3259–65.

11. Sequences HP. Opera□: Reconstructing Optimal Genomic Scaffolds. J. Comput. Biol. 2011;18:1681–91.

12. Donmez N, Brudno M. SCARPA: Scaffolding reads with practical algorithms. Bioinformatics.

2013;29:428–34.

552 13. Dayarian A, Michael TP, Sengupta AM. SOPRA: Scaffolding algorithm for paired reads via statistical optimization. BMC Bioinformatics. 2010;11:345.

554 14. Boetzer M, Henkel C V., Jansen HJ, Butler D, Pirovano W. Scaffolding pre-assembled contigs using SSPACE. Bioinformatics. 2011;27:578–9.

556 15. Sahlin K, Vezzi F, Nystedt B, Lundeberg J, Arvestad L. BESST - Efficient scaffolding of large fragmented assemblies. BMC Bioinformatics. 2014;15:281.

558 16. Luo J, Wang J, Zhang Z, Li M, Wu FX. BOSS: A novel scaffolding algorithm based on an optimized scaffold graph. Bioinformatics. 2017;33:169–76.

560 17. Hunt M, Newbold C, Berriman M, Otto TD. A comprehensive evaluation of assembly scaffolding tools. Genome Biol. 2014;15:R42.

562 18. Bashir A, Klammer AA, Robins WP, Chin C-S, Webster D, Paxinos E, et al. A hybrid approach for the automated finishing of bacterial genomes. Nat. Biotechnol. 2012;30:701–7.

564 19. Chaisson MJ, Tesler G. Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): Application and theory. BMC Bioinformatics. 2012;13:238.

566 20. Li H. Minimap and miniasm: Fast mapping and de novo assembly for noisy long sequences. Bioinformatics. 2016;32:2103–10.

568 21. Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;1–7.

22. Boetzer M, Pirovano W. SSPACE-LongRead: Scaffolding bacterial draft genomes using long read

570 sequence information. BMC Bioinformatics. 2014;15:211.

23. Warren RL, Yang C, Vandervalk BP, Behsaz B, Lagman A, Jones SJM, et al. LINKS: Scalable,

572 alignment-free scaffolding of draft genomes with long reads. Gigascience. 2015;4:35.

24. Gao S, Bertrand D, Chia BKH, Nagarajan N. OPERA-LG: Efficient and exact scaffolding of large, repeat-rich eukaryotic genomes with performance guarantees. Genome Biol. 2016;17:102.

25. Cao MD, Nguyen SH, Ganesamoorthy D, Elliott AG, Cooper MA, Coin LJM. Scaffolding and completing genome assemblies in real-time with nanopore sequencing. Nat. Commun. 2017;8:1–10.

26. Ye C, Hill CM, Wu S, Ruan J, Ma Z. DBG2OLC: Efficient assembly of large genomes using long erroneous reads of the third generation sequencing technologies. Sci. Rep. 2016;6:31900.

27. Wick RR, Judd LM, Gorrie CL, Holt KE. Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. PLoS Comput. Biol. 2017;13:1–22.

28. Hu X, Yuan J, Shi Y, Lu J, Liu B, Li Z, et al. pIRS: Profile-based illumina pair-end reads simulator. Bioinformatics. 2012;28:1533–5.

29. 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;1:18.

30. Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS, et al. SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. J. Comput. Biol. 2012;19:455–77.

31. Steinberg KM, Schneider VA, Graves-Lindsay TA, Fulton RS, Agarwala R, Huddleston J, et al. Single haplotype assembly of the human genome from a hydatidiform mole. Genome Res. 2014;24:2066–76.

32. Weisenfeld NI, Yin S, Sharpe T, Lau B, Hegarty R, Holmes L, et al. Comprehensive variation discovery in single human genomes. Nat. Genet. 2014;46:1350–5.

33. Ghurye J, Pop M, Koren S, Bickhart D, Chin CS. Scaffolding of long read assemblies using long range contact information. BMC Genomics. 2017;18:527.

34. Saha S, Rajasekaran S. Efficient and scalable scaffolding using optical restriction maps. BMC Genomics. 2014;15:S5.

35. Chin CS, Peluso P, Sedlazeck FJ, Nattestad M, Concepcion GT, Clum A, et al. Phased diploid genome assembly with single-molecule real-time sequencing. Nat. Methods. 2016;13:1050–4.

36. Gurevich A, Saveliev V, Vyahhi N, Tesler G. QUAST: Quality assessment tool for genome assemblies. Bioinformatics. 2013;29:1072–5.

37. Li H, Durbin R. Fast and accurate long-read alignment with Burrows-Wheeler transform. Bioinformatics. 2010;26:589–95.

38. Salzberg SL, Yorke JA. Beware of mis-assembled genomes. Bioinformatics. 2005;21:4320–1.

39. Muggli MD, Puglisi SJ, Ronen R, Boucher C. Misassembly detection using paired-end sequence reads and optical mapping data. Bioinformatics. 2015;31:i80–8.

40. Robinson GE, Hackett KJ, Purcell-Miramontes M, Brown SJ, Evans JD, Goldsmith MR, et al. Creating a buzz about insect genomes. Science. 2011;331:1386.

41. Zhang G. Genomics: Bird sequencing project takes off. Nature. 2015;522:34.

618

620

Figure 1. A validating model of alignment. The P1 and P2 are the two points for breaking a long

622 read into 3 regions (R1, R2, and R3).

Figure 2. The construction of link using a long read *lri* and two contigs c_i and c_j. a) a basic

624 schematic for a long read building link between contigs; b) the distance distribution of links.

Figure 3. The schematic illustration for travelling complex region.

626 Table 1. Descriptive statistics of datasets used for the comparative study.

Table 2. The statistics of draft assembly for *E. coli*, *S. cerevisiae*, *D. melanogaster*, *O. sativa*,

628 and *H. sapiens*.

Table 3. The performances of scaffolders tested for *E. coli*, *S. cerevisiae*, *D. melanogaster*, *O.*

630 *sativa*, and *H. sapiens* using PacBio long reads.

Table 4. The performances of scaffolders tested for *E. coli*, *S. cerevisiae*, and *H. sapiens* using

632 ONT long reads.

Supplementary Table 1. The performances for *E. coli* and *S. cerevisiae* based on draft

634 assemblies generated by SOAPdenovo2 and SPAdes using 1, 5, and 10 -fold coverages of PacBio long reads.

636 Supplementary Table 2. The performances for *E. coli* and *S. cerevisiae* based on draft assemblies generated by SOAPdenovo2 using 20-fold coverage of PacBio long reads.

638 Additional file 1: Pacific Biosciences SMRT long reads (< 30 Kb) distribution.

Additional file 2: Oxford Nanopore Technologies long reads (< 30 Kb) distribution.

640

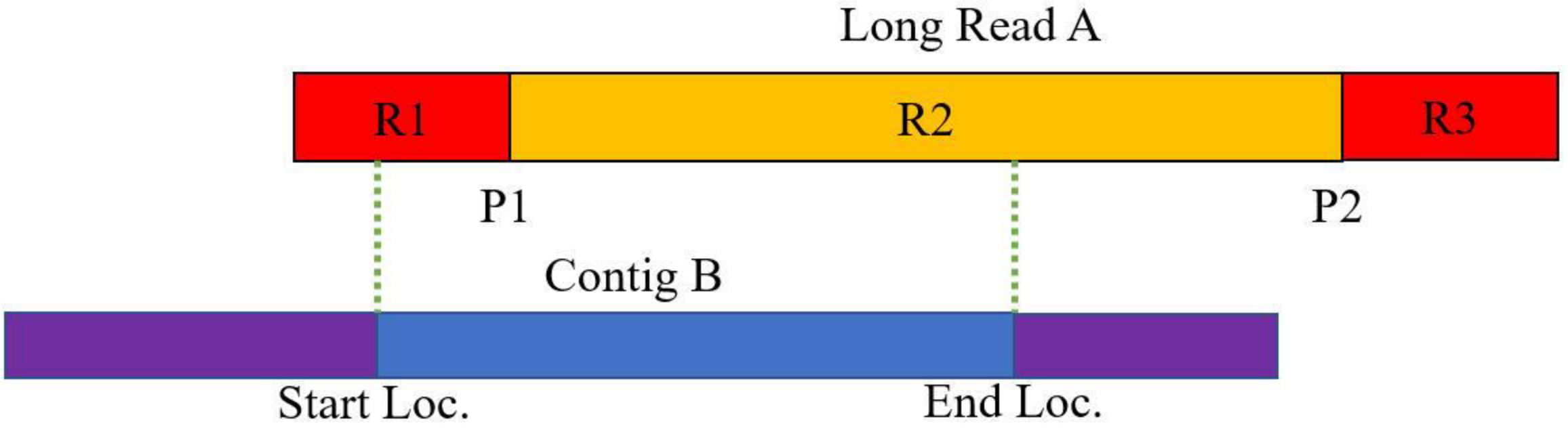

Figure 1. A validating model of alignment. The P1 and P2 are the two points for breaking a long read into 3 regions (R1, R2, and R3).

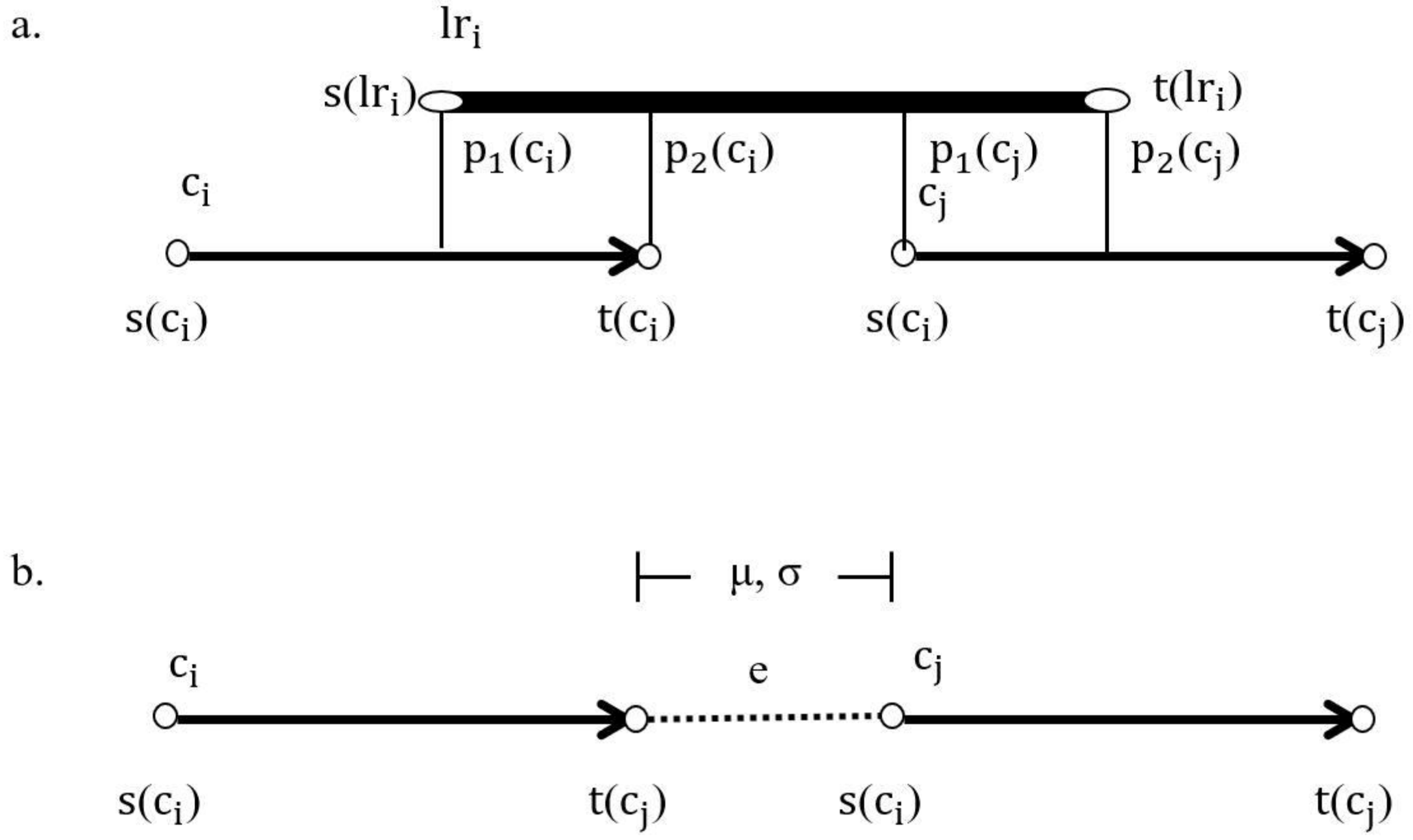

Figure 2. The construction of link using a long read *lri* and two contigs c_i and c_j. a) a basic schematic for a long read building link between contigs; b) the distance distribution of links.

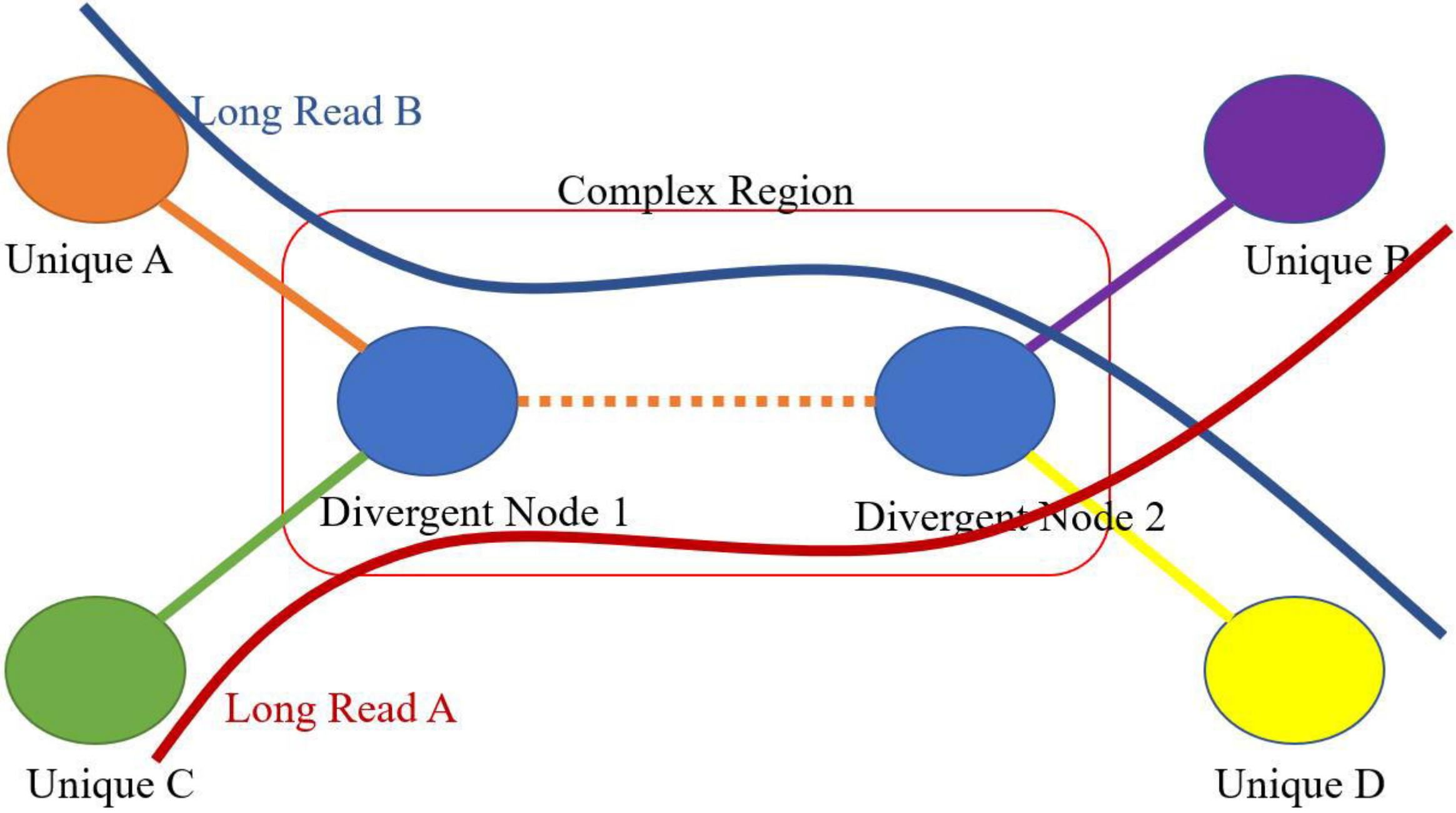

Figure 3. The schematic illustration for travelling complex region.

Table 1. Descriptive statistics of datasets used for the comparative study.

Organism	TYPE	Reads (#)	Total bases (bp)	Coverage	Median (bp)	Longest (bp)	Source
E. coli	Illumina	28,428,648	2,842,864,800	607.2 x	100	100	ERA000206
	PacBio	9,291	93,994,356	20.1 x	8,712	41,331	SRX669475; SRX533603
	ONT-Full[a]	3,471	21,972,483	4.7 x	5,743	47,422	ERX708228
	ONT-All[a]	24,221	158,867,566	34.0 x	6,086	47,422	ERX708228
	ONT-Raw[a]	70,531	311,558,723	66.5 x	3,557	94,116	ERX708228
S. cerevisiae	Illumina	6,801,728	1,268,786,706	105.1 x	202	202	SRR527545; SRR527546
	PacBio	44,786	249,319,042	20.7 x	4,554	27,575	SRX533604
	ONT-Nanocorr[a]	88,218	526,588,732	43.6 x	5,512	72,879	SRP055987
	ONT-Raw[a]	407,761	2,392,848,698	198.2 x	5,059	191,145	SRP055987
D. melanogaster	Illumina	60,190,770	6,019,077,000	50.0 x	100	100	SYNTHESE[b]
	PacBio	127,403	2,271,687,745	18.9 x	19,577	33,581	SRX499318
O. sativa	Illumina	186,622,748	18,662,274,800	50.0 x	100	100	SYNTHESE[b]
	PacBio	1,284,129	4,354,429,905	11.7 x	3,391	24,405	SRR3743363
H. sapiens	PacBio	10,245,649	59,999,995,767	20.0 x	1,569	208,628	SAMN02744161
	ONT	15,599,452	114,380,310,980	35.0 x	4,569	1,537,349	PRJEB23027

Note: [a] refer to LINKS dataset; [b] Synthesized by using pIRS (version 1.11) with parameters -x 50 and -c 0.

Table 2. Draft assembly statistics for *E. coli* , *S. cerevisiae* , *D. melanogaster* , *O. sativa* , and *H. sapiens.*

Organism	Source	Reference Length (bp)	Chr.	Assembled Length (bp)	Fraction	Contigs (#)	N00 (bp)	N50 (bp)	N100 (bp)
E. coli	SOAPdenovo2	4,681,865	1	4,598,322	0.98	728	164,235	40,009	52
	SPAdes			4,579,398	0.98	242	264,985	133,189	56
	ABySS[a]			5,160,631	1.10	69	358,719	177,636	493
S. cerevisiae	SOAPdenovo2	12,071,326	16	12,063,232	0.99	6,961	146,672	19,567	60
	SPAdes			11,754,316	0.97	2,254	451,383	107,906	56
	Celera Assembly[a]			14,910,895	1.24	6,953	257,346	49,258	64
D. melanogaster	SOAPdenovo2	120,381,546	6	118,065,428	0.98	45,480	902,599	111,033	62
O. sativa	SOAPdenovo2	373,245,519	12	346,168,844	0.93	257,801	147,060	18,977	3
H. sapiens	SRPRISM+ARGO[b]	2,996,426,293	23	2,781,084,252	0.93	40,906	1,009,096	140,502	199
	DISCOVAR[c]			3,068,057,564	1.02	858,918	1,380,479	179,783	201

Note: [a] refers to LINKS; [b] refers to [31]; [c] refers to [32].

Table 3. The performances of scaffolders tested for *E. coli* , *S. cerevisiae* , *D. melanogaster* , *O. sativa* , and *H. sapiens* using PacBio long reads.

Organism	Methods	Sequences (#)	Sum	NG50	NA50	Longest Sequence	Misassembles (#)	CPU Time (min)	Peak RAM (Gb)
E. coli	SPAdes	242	4.6 Mb	133.2 Kb	132.9 Kb	264.0 Kb	2	196.28	0.02
	SSPACE-LongRead	164	4.7 Mb	**4.6 Mb**	2.1 Mb	**4.6 Mb**	2	48.50	-
	LINKS	183	4.6 Mb	0.4 Mb	0.2 Mb	1.3 Mb	13	9.40	20.01
	OPERA-LG	176	4.7 Mb	1.5 Mb	1.2 Mb	2.1 Mb	4	361.79	**0.01**
	SMIS	185	4.7 Mb	**4.6 Mb**	**3.6 Mb**	**4.6 Mb**	2	26.51	-
	npScarf	134	4.7 Mb	2.5 Mb	1.6 Mb	1.3 Mb	9	4.49	1.71
	Unicycler	1	4.6 Mb	**4.6 Mb**	2.7 Mb	**4.6 Mb**	**1**	7,378.72	3.28
	DBG2OLC	4	4.2 Mb	1.3 Mb	0.6 Mb	1.6 Mb	4	2.32	0.17
	LRScaf (BLASR)	173	4.8 Mb	**4.6 Mb**	2.6 Mb	**4.6 Mb**	2	1.74	1.80
	LRScaf (minimap)	173	4.8 Mb	**4.6 Mb**	2.7 Mb	**4.6 Mb**	2	**0.17**	0.25
	LRScaf (minimap2)	173	4.8 Mb	**4.6 Mb**	2.7 Mb	**4.6 Mb**	2	0.19	0.25
S. cerevisiae	SPAdes	2,254	11.8 Mb	104.2 Kb	93.5 Kb	451.4 Kb	22	133.31	0.03
	SSPACE-LongRead	2,012	12.1 Mb	510.4 Kb	196.6 Kb	**1.0 Mb**	75	108.05	-
	LINKS	2,057	11.8 Mb	260.2 Kb	161.9 Kb	**1.0 Mb**	43	85.22	45.23
	OPERA-LG	2,078	12.0 Mb	418.6 Kb	247.1 Kb	**1.0 Mb**	41	12.10	**0.01**
	SMIS	2,115	11.9 Mb	416.3 Kb	263.9 Kb	0.9 Mb	32	41.68	-
	npScarf	1,868	11.9 Mb	**665.8 Kb**	202.1Kb	**1.0 Mb**	105	12.45	2.45
	Unicycler	62	11.5 Mb	326.1 Kb	**284.1 Kb**	0.8 Mb	**17**	1,459.92	5.80
	DBG2OLC	38	7.5 Mb	172.2 Kb	174.5 Kb	0.7 Mb	24	16.90	0.42
	LRScaf (BLASR)	2,063	12.7 Mb	440.0 Kb	260.9 Kb	**1.0 Mb**	38	9.27	1.16
	LRScaf (minimap)	2,109	12.3 Mb	421.3 Kb	283.0 Kb	**1.0 Mb**	34	0.39	0.28
	LRScaf (minimap2)	2,111	12.3 Mb	421.2 Kb	283.0 Kb	**1.0 Mb**	33	**0.34**	0.51
D. melanogaster	SOAPdenovo2	45,480	118.1 Mb	107.8 Kb	111.0 Kb	902.6 Kb	0	23.15	43.00
	SSPACE-LongRead	42,136	124.1 Mb	**6.6 Mb**	3.8 Mb	**19.6 Mb**	83	3,703.10	-
	LINKS	42,976	119.0 Mb	0.3 Mb	0.3 Mb	1.4 Mb	480	766.27	675.37
	OPERA-LG	42,543	123.5 Mb	3.7 Mb	2.6 Mb	19.2 Mb	211	130.21	**0.12**
	SMIS	43,387	122.4 Mb	4.0 Mb	3.1 Mb	15.6 Mb	112	4,035.66	-
	npScarf	41,657	120.9 Mb	5.1 Mb	0.3 Mb	11.8 Mb	1,515	37.20	15.26
	DBG2OLC	715	143.3 Mb	5.0 Mb	1.7 Mb	11.3 Mb	2,393	132.20	4.30
	LRScaf (BLASR)	43,116	124.4 Mb	5.4 Mb	**5.2 Mb**	**19.6 Mb**	**15**	32.19	1.46
	LRScaf (minimap)	42,696	124.1 Mb	5.5 Mb	3.7 Mb	15.1 Mb	35	4.08	3.69
	LRScaf (minimap2)	42,675	123.7 Mb	6.1 Mb	3.7 Mb	17.8 Mb	21	**1.61**	3.72
O. sativa	SOAPdenovo2	257,770	346.2 Mb	17.2 Kb	19.0 Kb	147.1 Kb	45	206.92	147.23
	SSPACE-LongRead	TLE[b]	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	LINKS	242,206	351.0 Mb	47.3 Kb	47.7 Kb	424.5 Kb	535	1,272.45	877.72
	OPERA-LG	234,910	357.5 Mb	79.1 Kb	62.8 Kb	684.2 Kb	2,604	391.24	**0.37**
	SMIS	238,851	352.6 Mb	55.1 Kb	50.3 Kb	423.8 Kb	944	8,040.56	-
	npScarf	245,140	347.0 Mb	63.8 Kb	50.1 Kb	553.8 Kb	2,198	87.58	6.50
	DBG2OLC	5,759	331.3 Mb	**94.5 Kb**	**64.9 Kb**	**794.7 Kb**	659	338.11	10.18
	LRScaf (BLASR)	240,136	365.2 Mb	60.5 Kb	54.9 Kb	482.1 Kb	734	117.76	4.09
	LRScaf (minimap)	240,054	362.7 Mb	53.4 Kb	49.6 Kb	426.4 Kb	803	13.46	4.48
	LRScaf (minimap2)	240,857	362.7 Mb	54.3 Kb	51.3 Kb	459.9 Kb	**455**	**6.30**	4.14
H. sapiens (CHM1)[d]	SRPRISM+ARGO	35,120	2.8 Gb	127.5 Kb	140.5 Kb	1.0 Mb	106	-	-
	SSPACE-LongRead	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	LINKS	MLE[c]	MLE	MLE	MLE	MLE	MLE	MLE	MLE
	SMIS	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	npScarf	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	DBG2OLC	3,932	1.2 Gb	-	217.6 Kb	2.3 Mb	169	3,700.02	64.69
	LRScaf (BLASR)	1,319	2.8 Gb	9.5 Mb	9.0 Mb	43.5 Mb	**266**	2,701.48	27.23
	LRScaf (minimap)	1,697	2.8 Gb	5.2 Mb	5.3 Mb	26.0 Mb	371	169.20	23.91
	LRScaf (minimap2)	1,426	2.8 Gb	**10.4 Mb**	**10.7 Mb**	**45.0 Mb**	292	**47.49**	**20.28**

Note: [a] is not available. [b] means that the run time is exceeded 3 weeks' time limit. [c] means that the memory usage is exceeded the capacity of system (1TB). [d] the assembly metrics are computed by QUAST_dev_5.0. The best genomic assembly metrics are highlighted in Bold.

Table 4. The performances of scaffolders tested for *E. coli* , *S. cerevisiae* , and *H. sapiens* using Nanopore long reads.

Organism	Methods	Sequences (#)	Sum	NG50	NA50	Longest Sequence	Misassembles (#)	CPU Time (min)	Peak RAM (Gb)
E. coli FULL	ABySS[a]	69	5.2 Mb	179.7 Kb	146.9 Kb	358.7 Kb	5	-[b]	-
	SSPACE-LongRead	47	5.2 Mb	226.7 Kb	204.3 Kb	628.4 Kb	6	19.77	-
	LINKS	51	5.2 Mb	271.0 Kb	226.3 Kb	633.2 Kb	**5**	1.06	2.10
	SMIS	38	5.2 Mb	638.8 Kb	357.9 Kb	951.2 Kb	**5**	247.62	-
	npScarf	43	5.2 Mb	344.8 Kb	229.6 Kb	632.9 Kb	6	1.01	0.56
	DBG2OLC	3	0.3 Mb	-	188.5 Kb	206.8 Kb	0	4.76	0.13
	LRScaf (BLASR)	30	5.2 Mb	**921.6 Kb**	**485.2 Kb**	**1,054.7 Kb**	6	0.57	**0.28**
	LRScaf (minimap)	53	5.2 Mb	226.7 Kb	204.3 Kb	611.5 Kb	**5**	0.26	0.31
	LRScaf (minimap2)	44	5.1 Mb	358.3 Kb	268.6 Kb	707.7 Kb	**5**	**0.15**	0.36
E. coli ALL	SSPACE-LongRead	45	5.2 Mb	226.7 Kb	226.3 Kb	406.8 Kb	**5**	135.70	-
	LINKS	43	5.2 Mb	294.0 Kb	226.3 Kb	633.2 Kb	**5**	6.78	14.23
	SMIS	27	5.2 Mb	**992.2 Kb**	**618.4 Kb**	**1,152.8 Kb**	**5**	43.08	-
	npScarf	33	5.2 Mb	454.2 Kb	344.7 Kb	838.1 Kb	8	5.18	1.22
	DBG2OLC	4	0.5 Mb	-	59.2 Kb	209.0 kb	1	5.58	0.13
	LRScaf (BLASR)	20	5.0 Mb	987.4 Kb	616.9 Kb	1,147.7 Kb	**5**	1.27	0.36
	LRScaf (minimap)	33	5.0 Mb	487.3 Kb	270.3 Kb	762.7 Kb	**5**	**0.19**	**0.28**
	LRScaf (minimap2)	24	5.2 Mb	693.8 Kb	357.9 Kb	1,147.7 Kb	**5**	0.34	0.46
E. coli RAW	SSPACE-LongRead	44	5.2 Mb	239.0 Kb	226.3 Kb	628.2 Kb	**5**	193.27	-
	LINKS	48	5.2 Mb	267.0 Kb	205.5 Kb	633.2 Kb	**5**	12.26	21.94
	SMIS	26	5.2 Mb	**928.1 Kb**	**879.1 Kb**	**1.152.5 Kb**	6	82.22	-
	npScarf	28	5.2 Mb	762.8 Kb	616.8 Kb	1,146.8 Kb	6	12.65	2.06
	DBG2OLC	3	0.2 Mb	-	9.8 Kb	106. Kb	0	10.49	0.13
	LRScaf (BLASR)	34	5.5 Mb	922.5 Kb	616.5 Kb	1,147.2 Kb	6	4.18	**0.32**
	LRScaf (minimap)	35	5.2 Mb	358.3 Kb	270.3 Kb	610.4 Kb	6	**0.32**	0.35
	LRScaf (minimap2)	36	5.2 Mb	445.1 Kb	357.9 Kb	922.6 Kb	**5**	**0.32**	0.29
S. cerevisiae NANOCORR	Celera Assembly[a]	6,953	14.9 Mb	58.8 Kb	46.4 Kb	257.3 Kb	19	-	-
	SSPACE-LongRead	6,353	15.7 Mb	231.4 Kb	132.9 Kb	733.3 Kb	50	454.82	-
	LINKS	6,651	15.1 Mb	235.5 Kb	110.6 Kb	623.1 Kb	55	50.25	51.73
	SMIS	6,706	15.1 Mb	470.5 Kb	**250.7 Kb**	1,094.4 Kb	**28**	293.07	-
	npScarf	6,649	15.1 Mb	**559.4 Kb**	219.9 Kb	**1,474.2 Kb**	65	7.23	3.26
	DBG2OLC	75	8.4 Mb	139.2 Kb	143.1 Kb	490.8 Kb	76	20.76	0.48
	LRScaf (BLASR)	6,338	15.3 Mb	231.4 Kb	137.7 Kb	761.5 Kb	39	4.95	**0.46**
	LRScaf (minimap)	6,678	15.7 Mb	261.5 Kb	144.7 Kb	741.5 Kb	41	**1.40**	1.34
	LRScaf (minimap2)	6,435	16.3 Mb	445.1 Kb	189.0 Kb	764.9 Kb	39	6.03	0.96
S. cerevisiae RAW	SSPACE-LongRead	5,914	17.8 Mb	239.0 Kb	99.8 Kb	1,086.8 Kb	147	1,563.55	-
	LINKS	6,680	15.0 Mb	231.8 Kb	159.7 Kb	737.2 Kb	26	98.64	153.46
	SMIS	6,696	15.1 Mb	438.2 Kb	205.7 Kb	1,094.8 Kb	35	1,100.18	-
	npScarf	6,629	15.1 Mb	**578.3 Kb**	**250.0 Kb**	**1,566.3 Kb**	48	369.20	8.57
	DBG2OLC	215	13.0 Mb	465.7 Kb	155.0 Kb	1,230.4 Kb	30	111.55	0.48
	LRScaf (BLASR)	6,347	15.6 Mb	318.9 Kb	199.7 Kb	750.8 Kb	29	29.29	**0.37**
	LRScaf (minimap)	6,719	15.8 Mb	375.0 Kb	177.4 Kb	753.2 Kb	37	**1.11**	1.01
	LRScaf (minimap2)	6,498	15.1 Mb	253.8 Kb	168.6 Kb	752.6 Kb	**23**	1.75	0.75
H. sapiens (NA12878)[e]	DISCOVAR	43,541	2.8 Gb	115.7 Kb	127.3 Kb	961.2 Kb	336	-	-
	SSPACE-LongRead	TLE[c]	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	LINKS	MLE[d]	MLE	MLE	MLE	MLE	MLE	MLE	MLE
	SMIS	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	npScarf	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
	DBG2OLC	10	29.3 Mb	-	-	16.6 Mb	0	5,483.93	69.84
	LRScaf (BLASR)	2,412	2.9 Gb	16.5 Mb	11.7 Mb	**71.6 Mb**	856	1,323.26	**41.20**
	LRScaf (minimap)	3,182	2.9 Gb	12.2 Mb	10.1 Mb	49.4 Mb	**720**	377.63	78.89
	LRScaf (minimap2)	2,462	2.9 Gb	**17.4 Mb**	**13.6 Mb**	64.2 Mb	785	**127.07**	62.56

Note: [a] refers to LINKS dataset; [b] is not available. [c] means the run time is exceedd 3 weeks' time limit. [d] means that the memory usage is exceeded the capacity of system (1TB).
[e] the assembly metrics are computed by QUAST_dev_5.0.The best genomic assembly metrics are highlighted in Bold.