Briefings in Functional Genomics, 18(1), 2019, 1–12

doi: 10.1093/bfgp/ely037
Advance Access Publication Date: 21 November 2018
Review paper

The bioinformatics tools for the genome assembly and analysis based on third-generation sequencing

YongKiat Wee, Salma Begum Bhyan, Yining Liu, Jiachun Lu, Xiaoyan Li and Min Zhao

Corresponding authors: Xiaoyan Li, Beijing Anzhen Hospital, Capital Medical University, Beijing, China. Tel.: +86010-64456199; Fax: +86010-64456169; E-mail: xiaoyanli82@163.com, Min Zhao, School of Science and Engineering, Faculty of Science, Health, Education and Engineering, University of the Sunshine Coast, Queensland 4558, Australia. Tel.: +61 (0)423791085; Fax: 61 7 5456 3402; E-mail: mzhao@usc.edu.au

Abstract

The application of third-generation sequencing (TGS) technology in genetics and genomics have provided opportunities to categorize and explore the individual genomic landscapes and mutations relevant for diagnosis and therapy using whole genome sequencing and *de novo* genome assembly. In general, the emerging TGS technology can produce high quality long reads for the determination of overlapping reads and transcript isoforms. However, this technology still faces challenges such as the accuracy for the identification of nucleotide bases and high error rates. Here, we surveyed 39 TGS-related tools for *de novo* assembly and genome analysis to identify the differences among their characteristics, such as the required input, the interaction with the user, sequencing platforms, type of reads, error models, the possibility of introducing coverage bias, the simulation of genomic variants and outputs provided. The decision trees are summarized to help researchers to find out the most suitable tools to analyze the TGS data. Our comprehensive survey and evaluation of computational features of existing methods for TGS may provide a valuable guideline for researchers.

Key words: third-generation sequencing; genome assembly; mapping; genome sequencing

Introduction

The advent of next-generation sequencing (NGS) technologies created a revolutionary impact on human genomics study. Since the 1st market launch in 2005, these technologies accelerated genome mining through dramatic reduction of overall cost of sequencing [1]. NGS technologies are distinct in their approaches and are of high-throughput in nature providing millions of sequencing reactions simultaneously. Current NGS platform includes Roche/454 [2], Illumina/Solexa [3, 4], Ion torrent, Sequencing by Oligonucleotide Ligation and Detection (SOLiD), etc. All these technologies have some advantages and disadvantages over each other. However, the common disadvantages of NGS technologies in genome assembly and analysis are (1) small read lengths (<300 bp), which create difficulties in *de novo* assembly; (2) regions such as high/low G+C regions, tandem repeat regions and interspersed repeat regions, which are hard to sequence using the NGS platforms; and (3) *de novo* genome assemblies lacking entire portions of genomes and missing vital genes, which could be due to fragmentation [5]. The missing genome regions can generate genome assemblies that lack adequate robustness to examine the whole genome organization and chromosome architecture [6].

Third-generation sequencing (TGS) technologies came out with a new insight in sequencing and produce highly accurate *de novo* assemblies in different genomes *de novo* [7]. These technologies improved the sequencing efficiency through rapid sample preparation and real-time signaling. The major platforms using TGS technology are Pacific Biosciences (PacBio) single-molecule real-time (SMRT) sequencing,

YongKiat Wee is a PhD student at the University of the Sunshine Coast.
Salma Begum Bhyan is a PhD student at the University of the Sunshine Coast.
Yining Liu is a research fellow at the Guangzhou Medical University.
Jiachun Lu is a Professor at The School of Public Health, The First Affiliated Hospital, Guangzhou Medical University.
Xiaoyan Li is a research fellow and experts in clinical sequencing at Beijing Anzhen Hospital, Capital Medical University.
Min Zhao is a senior research fellow at the University of the Sunshine Coast.

Oxford Nanopore Technologies (ONT) sequencing and BioNano Genomics (BioNano) sequencing [8]. These platforms have advantages over first- and second-generation platforms: (1) long read lengths (half of the data in reads >20 kb, maximum read length >60 kb), (2) high percentage of consensus accuracy (>99.999% at 30× coverage depth, free of systematic errors), (3) low bias of G+C content and (4) simultaneous epigenetic classification (direct identification of DNA base modifications at one-base resolution) [9]. Finally, TGS is also competing with alternative technologies that can perform similar analyses often at a lower cost. In summary, comparing the NGS platforms, there are three important improvements in TGS platforms: (1) increase in read length from tens of bases to tens of thousands of bases per read, (2) reduction in the sequencing time from days to hours (or to minutes for real-time applications) and (3) reduction or elimination of sequencing bias introduced by polymerase chain reaction (PCR) amplification [10]. The ONT sequencing technologies continue to evolve and improve these past years. A new device called PromethION (Oxford Nanopore Technologies, Oxford, United Kingdom) has been introduced recently; it's a bigger version compared to another nanopore device, MinION (Oxford Nanopore Technologies, Oxford, United Kingdom), that is created for portability and accessibility of its workflow [11]. The PromethION is a standalone high-throughput benchtop instrument which offers the flexibility to load up to 192 libraries across the whole instrument in an asynchronous approach. In comparison to MinION that performs with a single 512-channel flow cell, PromethION possess a higher capacity and larger scale with 48 individual flow cells each with 3000 pores (equivalent to 48 MinIONs) running at 500 base pairs per second that is adequately powerful to achieve high accuracy and high coverage for a larger genome such as human genome [12]. In addition, users can execute or stop the analysis as requested or increase the speed by utilizing the numerous flow cells onto one single analysis. Real-time base calling and further analysis can be performed in the integrated compute module [13]. The nanopore data generated by the MinION and the PromethION are integrated into a cloud-based analytics company, Metrichor. Metrichor is powered by its EPI2ME platform [11]. Metrichor allows the automation of data analysis workflows to aid in tracking, predicting and interpreting biological data on a real-time basis. ONT develop and offer several different types of analysis software tools such as MinKNOW (Oxford Nanopore Technologies, Oxford, United Kingdom), Albacore and Guppy. The nanopore data generated from these two instruments could be utilized for detecting the complex structural variants (SVs), for uncovering the highly repetitive sequences and for examining the biological structure of larger genomes in different species such as mammalian genomes [12].

Because of the advantage of longer reads, TGS technologies have been implemented as a powerful tool for studying the evolution and genomic diversity of an organism [7]. Data developed through TGS have been widely applied in resequencing analyses, creating detailed maps of structural variations and phasing variants across large regions of human chromosomes. TGS have also been widely recognized as useful tools for studying transcriptomics and discovering thousands of novel isoforms including alternative splicing detection and gene fusions that were not identified using second-generation short read sequencing [7]. A combined approach of TGS and mapping technologies could enhance the analysis of structural variations by forming super-contigs ('scaffolds') that can span almost the entire arm of a chromosome. Although long-read sequencing in TGS overcomes the length limitation of NGS, it remains considerably more expensive and has lower throughput than other platforms, limiting the widespread adoption of this technology in favor of less expensive approaches.

Here, we present our systematic and comprehensive review of available software tools for the *de novo* and whole genome sequencing analyses of TGS data. We review a total of 39 TGS analysis tools, which were either recently published or developed (Table S1). We discuss their various characteristics, such as the required input, interaction with the user, sequencing platforms, type of reads, error models, possibility of introducing coverage bias, simulation of genomic variants and output provided. This is done within the framework of potential applications, providing readers with guidelines for the identification of the TGS *de novo* software applications that are best suited for their purposes. This review evaluates various tools applied in three main TGS platforms on genome assembly and further analysis. Details of each approach along with its benefits and drawbacks are discussed. Lastly, two distinct decision trees are presented to guide researchers for selecting a suitable TGS *de novo* and genome-based sequencing analysis tools.

De novo assembly using TGS technologies

In recent years, there has been a major transformation in the way of extracting genomic information from organisms. *De novo* long-read genome assembly involves in several steps including raw read mapping, read error correction, assembly of corrected reads and assembly polishing. Long-read genome assemblers normally use overlap-based procedures such as overlap–layout–consensus (OLC) algorithms to assemble the long reads [14]. It first generates the alignments between long reads. After that it calculates the best overlap graph and then the consensus sequence of the contigs is generated from the graph. There are two approaches for long reads error correction. The 1st approach involves in aligning the long reads against themselves while the 2nd approach uses short reads to correct long reads [14]. Even though error correction stage may have been part of the assembly process, errors can still be found in the assembly, specifically in long-read assemblies. This can be improved by polishing the assemblies with short or long reads such as the accuracy of the base calls [15]. The development of high-throughput sequencing technologies including TGS has been instrumental in advancing research in all scientific areas. As shown by the impressive increase in genomic data output, TGS tools have been developed to allow for the rapid and easy annotation, prioritization and navigation of large variant data sets from various platforms. Figure 1 highlights the major development of the TGS tools for the past 5 years. There are more than five different tools that have been developed in 2015 and 2016. Most of these tools are involved in *de novo* and genome-based sequencing analysis such as RefAligner, Canu and Nanopore Synthetic-Long (NaS).

The specialized assembly of sequencing reads is an essential process in *de novo* sequencing and assembling the novel genome for the first time. The read length in TGS provides a great advantage for the genome assembly process. Second-generation platforms include MiSeq, HiSeq and NextSeq from Illumina. These platforms use massively parallel sequencing to achieve high throughput and have high base-calling accuracy; however, the sequencing reads are short and this can result in split contigs in repetitive regions during sequence assembly. Third-generation sequencers including PacBio from Pacific Biosciences and MinION from ONT can generate a very long read length at high throughput by sequencing single-molecule templates. TGS

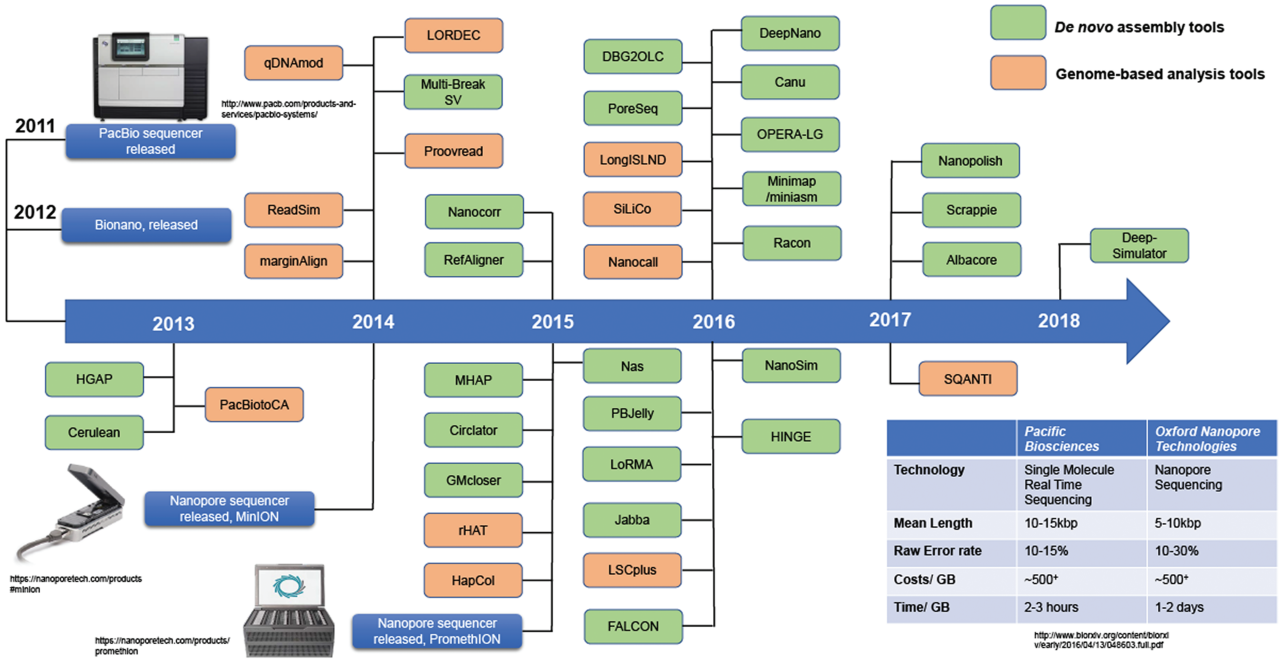

Figure 1. Milestones in TGS analysis software development. The green box refers to the *de novo* assembly tool while the orange box refers to the genome-based analysis tool.

platforms can address the problems inherent to short sequence reads by sequencing long single molecules in real time. Basically, assemblers are developed based on various types of algorithms, including OLC, *de Bruijn* graphs (DBG) and string graphs [16]. To make a *de novo* OLC-based assembly, three essential steps are involved in the process: preassembly, consensus build-up and consensus polishing. The main purpose of preassembly data processing is to produce long and accurate sequences by correcting base errors. Seed reads (a subset of the sequencing reads) are selected based on the read length distribution. Each single read is then mapped to the seed reads to generate a consensus sequence for the mapped reads, resulting in long and accurate fragments of the target genome. The computation in this step is very intensive as it involves all-versus-all raw read mapping and base error correction. The next step is the consensus building from the overlapping reads. A few options are available when selecting assembly algorithms, but OLC assemblers offer clear advantages for *de novo* assembly using multi-kb long reads. For genomes with repeats of any length, a single long error-corrected read could simply bridge the gaps among unique sequences and ensure that the consensus building process continues without interruption. When designing a *de novo* genome sequencing project, reasonable read coverage (50–60×) is needed to generate sufficient coverage of reads that uniquely anchor the longest repeat regions in the genome assembly. For preassembled reads, there could be base errors in the repetitive regions, where raw base errors are coupled with repeats. Errors such as indels and substitutions in the preassembled reads could also be easily passed on to the consensus. Therefore, there is a need for consensus polishing for assemblies produced from the TGS data.

Genome alignment and assembly tools for TGS platform

Several tools are being used in assembling long-read sequence developed through TGS platforms and a decision tree has been presented in Figure 2 for guiding the researchers to choose the suitable sequencing tools based on different *de novo* sequencing analysis in three different platforms including ONT, SMRT and BioNano sequencing platform. MinHash Alignment Process (MHAP), PBJelly, Hierarchical Genome-Assembly Process (HGAP), FALCON and HINGE utilize long reads from SMRT platforms. In the BioNano platform, the alignment tool known as RefAligner uses a dynamic programming algorithm to align each molecule map to the reference maps by identifying the best matching region in the sequence genome. In the BioNano platform, the alignment tool known as RefAligner implements a dynamic programming algorithm to align each molecule and map to the reference by determining the best matching region in a sequence genome [17]. The match score is then recorded from the *in silico* nicking sites in the region on the reference sequence and the distribution of fluorescent labels on the molecule. For the ONT platform, PoreSeq and Nanocorr are available for *de novo* sequencing analysis. Another two software programs - Minimap/miniasm and Circlator, which utilize long reads from both ONT and SMRT platforms for *de novo* assembly and circularization genome assembly analysis.

MHAP is developed to identify all overlaps among noisy long reads using a probabilistic hashing algorithm. MinHash sketches are implemented in this software for better alignment filtering. The algorithm works by estimating the Jaccard similarity based on the min-mers (minimum k-mers). The required time to index, store, hash and compare k-mers is proportional to the sketch size; hence, it is recommended that the sketches be kept at smaller size [18]. Even though NGS technologies can perform the sequencing in a faster and more cost-effective way, decoding a complete genome remains one of the important challenges in bioinformatics, particularly for complex genomes. Fragments or 'gaps' in *de novo* genome assembly can result from the short-read length, repetitive components and low sequence coverage [18].

PBJelly is a software program that uses scaffolding approach for gap closing in genome assembling [19]. The reads are first aligned to the contigs in establishing a scaffold and then reads that span numerous contigs are applied as links to build

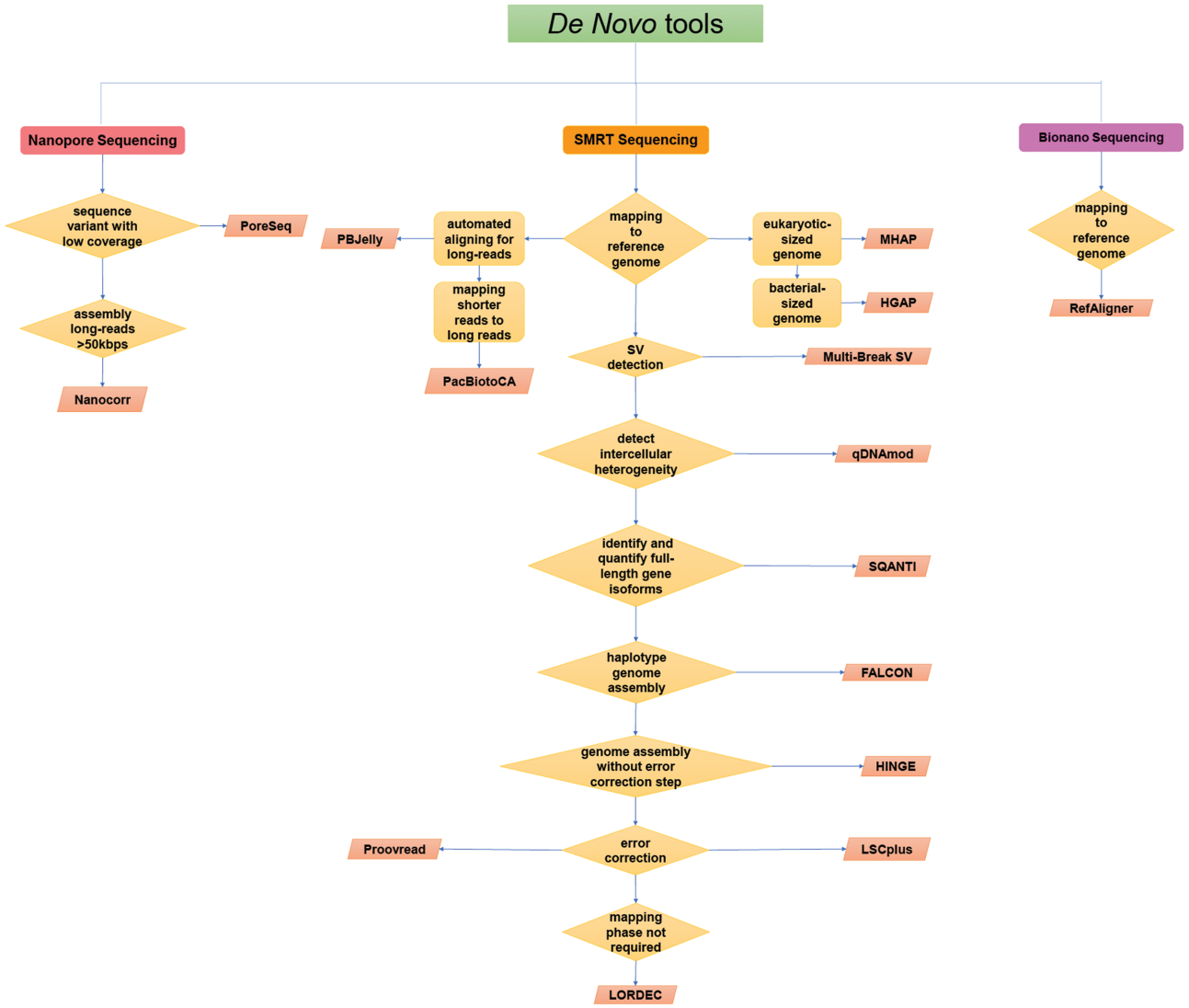

Figure 2. Decision tree for the selection of a suitable TGS *de novo* sequencing analysis tools in nanopore, SMRT and BioNano sequencing platform. The selection of a TGS tool requires a set of sequential decisions. First, one must decide on the reads from the main TGS platforms: nanopore, ONT; SMRT sequencing, PacBio technologies; and BioNano sequencing technologies. Then, in SMRT sequencing, one must determine whether the genome read is from a eukaryote or bacterium and whether it is automated aligning for long reads. In addition, one must decide which analysis involved in *de novo* sequencing including detection of SV or intercellular heterogeneity, identification and quantification of isoforms, haplotype genome assembly, genome assembly with or without error correction method and whether mapping phase is required. In BioNano platform, only one software is available for genome mapping. For nanopore sequencing, one must perform *de novo* sequencing analysis for sequence variants with low coverage or with long-reads assembly faster than 50 kbps.

a scaffold graph. It utilizes SMRT long reads instead of NGS short reads for gap closing [19]. HGAP is used for *de novo* assembly and it applies the longest reads for assembling a sequence genome [20]. The principle behind the hierarchical genome assembly process involves using long-insert-size DNA shotgun template libraries with SMRT sequencing. The longest reads are selected as 'seed' reads to which all other reads are sequenced and mapped. Both HGAP and PBJelly are applicable for bacterial-sized genomes and MHAP is used in eukaryotic-sized genomes. HGAP has a higher assembly quality and the ability to resolve repetitive regions, while MHAP has a higher sensitivity in overlapping processes, as it integrates another software known as Celera Assembly. PBJelly is designed to be automated for finishing the genome assembly process and it needs only FASTA format sequences as an input data; thus, it can perform the task faster than HGAP. In addition, MHAP has another great advantage over HGAP and PBJelly, as it can improve the telomere assemblies by reconstructing the repeat-rich heterochromatic regions of eukaryotic assemblies. FALCON is one of the hierarchical haplotype genome assembly tools that follows the design of HGAP but utilizes more computationally optimized elements [21]. It is applied on the long-read data from SMRT platform. Daligner is implemented to split the sequence data into blocks for comparison. Firstly, it collates a list of k-mers with their respective identified variables and read coordinates and subsequently organizes them lexicographically. The similar k-mers from each individual block are incorporated into a new list including both the query identifiers and the corresponding coordinates [22]. The sorting approach is implemented to generate the overlap candidates by locating the neighbouring matches adjacent to each other. Based on the alignments of the overlaps, a directed string graph consists of heterozygosity information is build [21]. HINGE is another assembler that implements the OLC paradigm with the absent of error correction step [23].

Dalinger is applied for detecting the overlaps. The principal behind this assembler is the repeat regions which are not spanned by longer reads will be replaced and marked with hinges [23]. The boundaries of unbridged repeats such as in-hinge and out-hinge are marked on the reads and the coverage gradients of the alignments are used to determine the repeats [24]. The overlapping reads will not be considered for hinge placing when a repeat is spanned by a completely bridged read and resulted in separate bridged repeats. Before acquiring a consensus, hinge-aided greedy graphs are useful in resolving the repeat junctions [23]. For the ONT platform, PoreSeq is the only available open source software and it uses Python for consensus, variant calling and *de novo* sequencing [25]. To acquire the *de novo* reads with higher accuracy and more uniform coverage, a novel algorithm that applies statistical models is proposed in this software. The base-calling algorithm works by using the discretized ionic current data from an autonomous number of nanopore reads of the same area of DNA, including reverse or partial accompaniment reads. PoreSeq is designed ideally for sequences with low coverage as it produces higher accuracy in classifying the sequence variants at low coverage than other methods [25]. Nanocorr is built as a novel open-source hybrid error correction algorithm using complementary MiSeq data and generating a *de novo* assembly that is notably high in accuracy [26].

There are three software programs, Minimap/miniasm, Circlator and Canu, which utilize long reads from both SMRT and ONT. Minimap/miniasm is a *de novo* assembly software program that functions as a mapper for mapping and assembling the SMRT and ONT reads with higher accuracy than other available tools [27]. Miniasm applies the 'O' and 'L' approaches in the OLC assembly paradigm [27]. It discovers long noisy reads that can be assembled without an error correction stage, and without this stage, the assembly process can be significantly improved, while attaining similar contiguity and large-scale accuracy to current developed pipelines, at least for genomes without excessive repetitive sequences. Despite the fact that these new technologies emphasize the automated completion of genome sequencing, the existing assembly software still presumes that the end products including contigs they generate are linear. The genomes in various species contain at least one circular DNA structure including bacterial chromosomes and plasmids and the plastid and mitochondrial genomes of eukaryotes. Correct completion and circularization of these molecules are

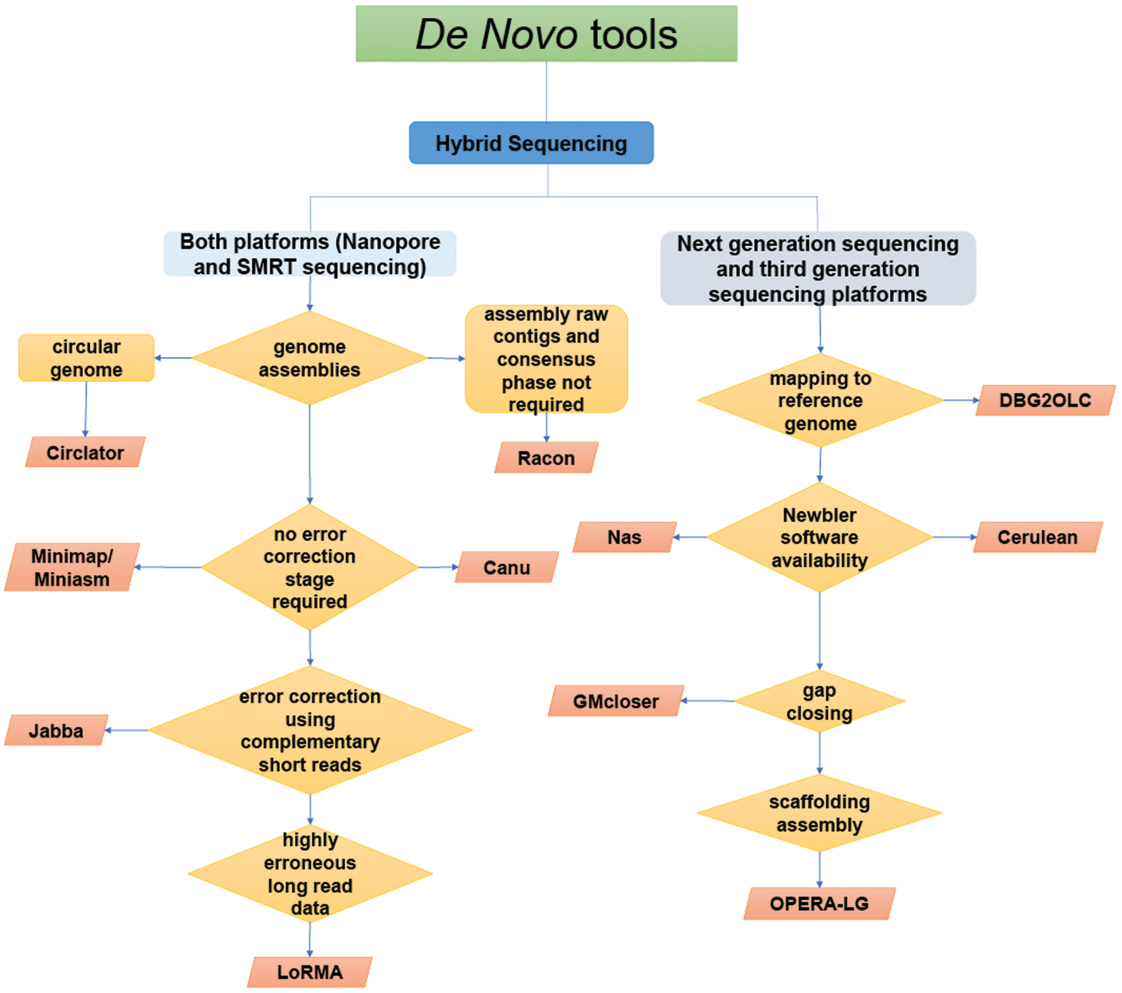

Figure 3. Decision tree for the selection of a suitable TGS *de novo* sequencing analysis tools in hybrid sequencing platform. The selection of the *de novo* sequencing analysis software packages used in hybrid sequencing platform. If the read used both platforms (nanopore and SMRT sequencing), one must decide whether the reads are from a circular genome and if no error correction is required. If the reads are applied from both NGS and the TGS platform, one must identify whether the reads are from AB eukaryotic-sized or bacterial-sized genome and determine the availability of the Newbler software. In addition, one must determine whether the reads from both the NGS and TGS platform need gap closing or scaffolding assembly.

important if the technology is to be applied frequently in clinical practice. Hence, Circlator is the first tool created to automate the assembly and produce accurate linear representations of circular sequences using both SMRT and ONT long reads [28]. The contigs are circularized using the local assemblies of corrected long reads at contig ends, preventing the search for common sequences between low-quality contig ends and this enabling the process of circularization even when overlaps do not exist [28]. Although Minimap/miniasm performs the sequencing faster than other software, it has lower accuracy in assembling the sequences as no error correction stage is required. As a result, it is difficult to identify the cause of low identity matching between two long noisy reads. Furthermore, a larger space and RAM are needed to execute this software; hence, it is not memory efficient. Circlator yields higher quality assemblies than other existing approaches, but the assembly of long reads without consideration for the circularization process can be problematic, particularly for small plasmids whose lengths are shorter than the length of the reads used to assemble it. This can cause the generated contigs to compose the entire sequence of the plasmid two or more times. Canu is developed to address the noisy read data of single-molecule sequences. This software support both PacBio and Oxford Nanopore data [29]. In comparison with the first two software, it has lower runtime and it requires only low coverage as little as 20× single-molecule coverage. Three stages—correction, trimming and assembly—are included in the Canu pipeline where each stage can perform independently. Canu has another great advantage over Miniasm as Miniasm lacks correction step, thus it could not resolve the repeat or the error rates. This means that Miniasm is less continuous than Canu assemblies on large genomes [29]. The assemblies generated by the Miniasm can be difficult to discard during filtering and polishing as it contains higher frequency of large insertions and deletions and low bas accuracy (<90%). Hence, a few rounds of polishing have to be processed in Miniasm before the assembly quality converges while a single round of polishing is only required by Canu [29]. However, Canu is not the fastest tool to generate a polished assembly read. As we discussed earlier, assembly with Miniasm followed by Racon performs faster than Canu itself.

Sequencing tools for hybrid technologies of both NGS and TGS

To date, many researchers have adopted a hybrid strategy by using both NGS and TGS to perform genome sequencing, thereby producing higher coverage and accuracy. A decision tree for hybrid strategy using both NGS and TGS technologies for sequence reads is presented in Figure 3 and this combined approach is common in *de novo* assembling, and these software programs include OPERA-LG, DBG2OLC, GMCloser, NaS and Cerulean. The average per base identity of the ONT reads can be greatly improved from 65% across all flow cell iterations to greater than 97% using this approach. Hence, it produces highly contiguous and complete assemblies given sufficient read lengths and sequence coverage [26].

OPERA-LG is used in assembly scaffolding ideally for larger and repeat-rich genomes [30]. It generates a framework for the scaffolding of repetitive sequences and a structured approach for incorporating the sequencing data from both second-generation and TGS technologies. To generate a scaffold with higher accuracy in genomes with larger sizes and more repeats, OPERA-LG combines some novel characteristics and improvements, including (a) optimized data structures to enhance its scalability, (b) improved edge-length estimation and the capability to utilize numerous libraries to enhance scaffolding accuracy and (c) extensions that allow for the scaffolding of repeat sequences [30]. One of the greatest advantages of OPERA-LG is that it has faster performance and it takes a few seconds and a few hundred megabytes of memory (largely for storing read mapping information) and thus it needs notably less memory.

DBG2OLC is a hybrid assembly approach that simultaneously utilizes NGS and TGS data to address both high error rates and the excessive cost of sequencing [31]. The software is designed based on the following fundamental principles: (i) compact representation of the long reads leads to efficient alignments; (ii) base-level errors can be ignored, structural errors need to be detected and corrected; (iii) structurally correct TGS reads are assembled and polished [31]. Furthermore, since NGS and TGS data can compensate for each other, the utilization of NGS data also lowers the required sequencing depth of TGS and leads to a reduced cost of sequencing. A base-level correction-free assembly pipeline is developed by directly analyzing and exploiting overlap information in the long reads. It utilizes NGS assemblies to lower the computational burden of aligning TGS sequences rather than just polishing the TGS data. This enables users to take advantage of the cheap and easily accessible NGS reads, while avoiding the issues associated with existing hybrid approaches [31].

Both NaS and Cerulean are used to assemble microbial genome. NaS is a hybrid method, which enables the sequencing of microbial genomes using the MinION® [32]. Cerulean is another hybrid assembler that uses both short (Illumina) and long (SMRT) reads [33]. This software program does not use short reads directly; however, it incorporates an assembly graph structure produced from short read data using other existing assembly tools. The algorithm works with a simplified version of the assembly graph consisting only of long contigs; then, in each iteration, smaller contigs are added slowly to improve the assembly process. In contrast to the state-of-the-art long-read error correction method, which needs high computational resources and a long run time on a supercomputer, the software can produce a similar assembly using only a standard desktop in a short running time [34]. In summary, DBG2LOC is applicable for eukaryotic-sized genomes and both NaS and Cerulean are used for bacterial-sized genomes. However, Cerulean has many benefits over NaS as it is completely automated, has short running time, is a stand-alone software, lowers the usage of computer resources and has high accuracy in assembling.

Tools for genome-based sequence analysis

Sequence alignment tools

The genome-based sequence alignment is involved in several steps including data processing and quality control. However, this review has mainly focused on the tools that are applied to TGS data analysis. Figure 4 shows a list of tools that allow users to perform whole genome sequence analysis based on certain requirements and parameters. In total, there are 14 tools for two TGS platforms.

For the SMRT data, one of the popular read alignment tool called regional Hashing-based Alignment Tool (rHAT) that is applicable for long reads only. rHAT uses a seed-and-extension-based read alignment method for noisy long reads [35]. A regional hash table (RHT) is implemented for indexing the reference genome by reporting the short tokens within local windows of a reference genome. During the seeding stage,

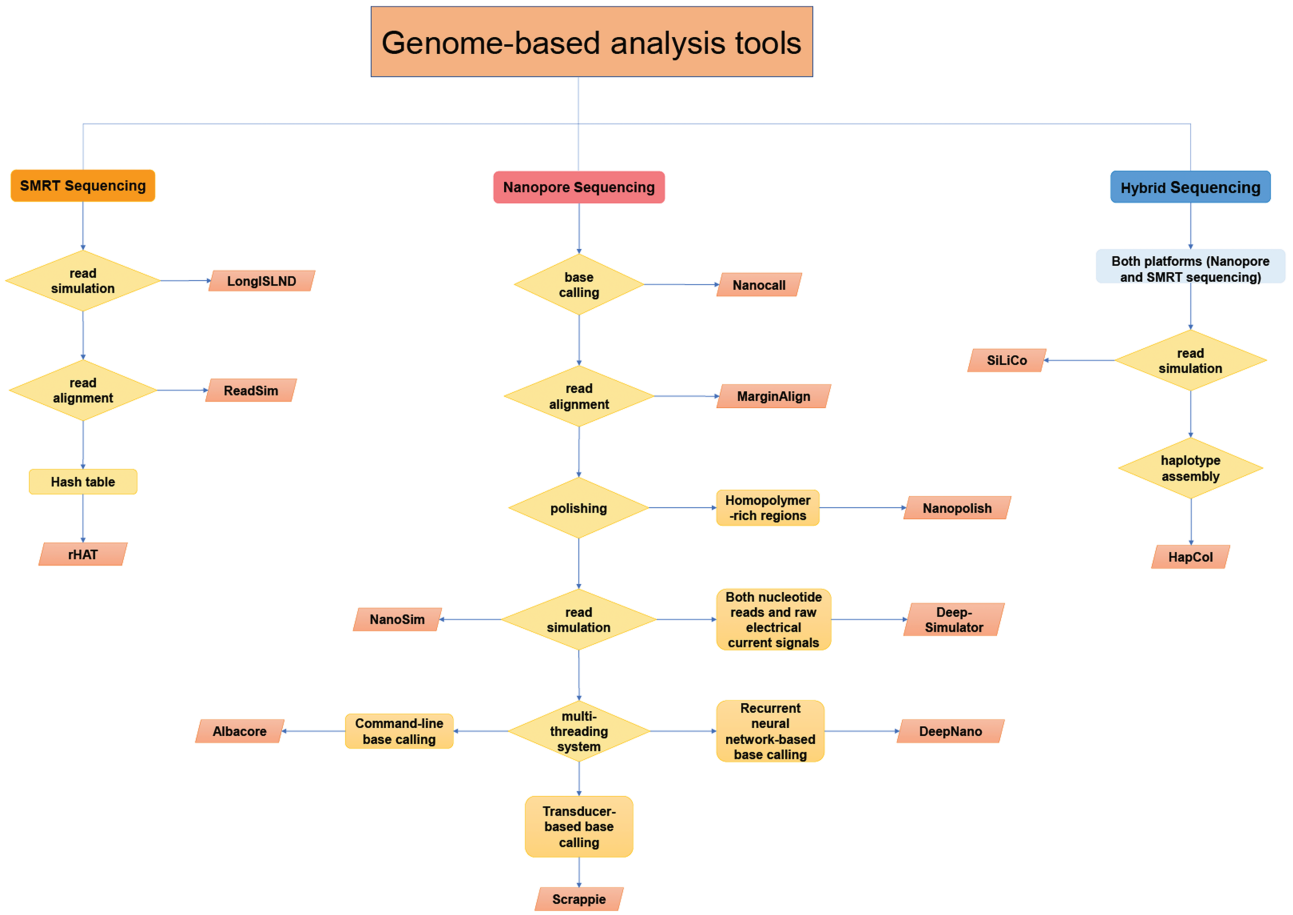

Figure 4. Decision tree for the selection of a suitable TGS genome-based sequencing analysis tool in SMRT, nanopore and hybrid sequencing platform. A set of sequential decisions has to be made when performing genome-based sequencing analyses. First, one must determine whether the reads are generated from the two TGS platforms nanopore or ONT and SMRT sequencing or PacBio technologies and whether it is a hybrid or non-hybrid read. If the reads from the SMRT platform are used for read alignment, one must identify the length of the reads and whether a hash table is required. For the reads from the nanopore platform, one must decide whether the reads should be performed for base calling or read alignment. For the hybrid reads that utilize both platforms, one must decide whether the analyses should be carried out for read simulation or haplotype assembly.

rHAT deploys RHT to determine the potential candidate sites by calculating the number of short token matches between the fragmented reads and local genomic windows in a genome sequence. One of the advantages of using rHAT is that it can lower the cost of aligning reads by implementing a sparse dynamic programming base heuristic approach in the extension step [35]. On the other hand, marginAlign used Oxford Nanopore long reads for read alignment. The rates of deletions, substitutions and insertions in MiNION reads are deduced by the maximum likelihood estimates using an expectation–maximization (EM) algorithm [36]. This EM is applied on the hidden Markov model (HMM) for the robust interpretation of the error sources into several classes of genetic mutations, including insertions, deletions and mismatches [36]. Overall, marginAlign generates high-quality sequence alignments that allow users to call single-nucleotide variant accurately with its built-in software, marginCaller. In addition, it also enables users to characterize the unresolved part with the repetitive sequence.

Base calling and polishing tools

Another tool that uses Oxford Nanopore long reads for base calling is Nanocall. It divides the sequence of events into strands based on several heuristic approaches. First, it measures the basic current level using a heuristic approach and identifies the islands with five or more consecutive abasic current estimations. Next, Nanocall chooses the island that is located nearest to the middle of the event sequence. If the selected island is positioned within the middle third of the whole event sequence, it is used to divide the events corresponding to the two strands [37]. Nanocall will stop executing if the island is identified outside of the middle third of the event sequence. An HMM is also applied on these events where the states are the k-mers being sequenced, the pore model emissions are then voluntarily scaled using several rounds of EM based on posteriors calculated with forward–backward algorithm, and the base calls are generated by running Viterbi [37]. Generally, the main advantage of using Nanocall is its double-strand pore model scaling performs better than the single-strand, implying that the latter might cause model overfitting. In addition, another base caller tool for nanopore data is Albacore (can be downloaded from ONT user community) [38]. It was a command-line base caller and implemented for the ultra-long reads. Albacore is a memory-efficient tool as it can directly base call the FASTQ file; hence, it saves more disk spaces [38]. This advantage makes Albacore quite practical and convenience compared to Nanopolish. Scrappie is the latest C-based local base caller software tool developed by ONT [39]. It conducts a transducer-based base calling method in order

to define the accurate length of homopolymers. It is known as the 1st base caller which resolves the base calling issue of homopolymer sequencing errors. Furthermore, the base calling with the raw current signal can be executed in Scrappie with the absence of event detection [39]. DeepNano is a freely available base caller software tool which utilizes a recurrent neural network-based method to conduct base calling for the MinION nanopore sequencing platform [40]. It was developed in Python. Both Scrappie and Albocore are considered better options for base calling compared to DeepNano as they support multi-threading [41]. The assembly reads can be improved through polishing process such as post-assembly error correction. There are two state-of-the-art sequence polishing tools—Nanopolish and Racon. Nanopolish implemented a Hidden Markov method to enhance the base quality by estimating the probabilities for each base from the raw signal data of reads [42]. Furthermore, the accuracy of the draft genome can also be increased using Nanopolish as it corrects the homopolymer-rich regions of the genome [42]. Racon is a consensus model that works independently to correct the raw contigs produced by the assembling approaches and it does not require consensus phase [43]. It attempts to identify the best alignment in order to improve the accuracy and quality of the assembly reads [43]. It supports the data generated by both Pacific Biosciences and ONT. In terms of accuracy, Nanopolish has higher accuracy results for polishing compared to Racon. However, Nanopolish is computationally expensive, and hence time-consuming.

Simulation tools: testing sequence alignment software

To test the software, a list of simulation software programs was developed to produce a synthetic TGS read. There are two distinct tools that allow users to apply SMRT or Oxford Nanopore long reads for read simulation: SiLiCo and ReadSim. SiLiCo is among the first *in silico* tool to generate high-quality sequencing reads from both TGS platforms [44]. It simulates sequencing results from the two sequencing technologies by randomly generating genomic coordinates and acquiring the corresponding nucleotide sequences from a reference assembly. An *in silico* simulator has been established to quantify the patterns of nick sites in sequencing libraries by generating an empirical distribution of terminal nucleotides in ideal long-read sequencing libraries. The scalability of SiLiCo enables the end user to build an empirical distribution of various genomic characteristics, as it can scale up to a Monte Carlo simulation [44]. ReadSim is developed based on a new data-driven model using support vector regression which can accurately estimate the assembly performance of a sequence [45]. It produces long reads imitating the read length distribution that exists in an input file by choosing a stochastic starting location in the genome and producing a read of the following observed length [45]. Another simulation tool known as Noisy Datatypes (LongISLND). LongISLND applies another approach which is known as learn-and-stimulate [35]. An empirical model is established through the learning process by capturing the samples for a specific set of real data, setting up an empirical model. Generally, LongISLND understands the alignment data by recording the base calls, with and without errors which correspond to the sequencing structure of the reference genome [46]. To examine these alignment records, a non-parametric model is implemented including the error profile such as genome sequence [46]. Overall, LongISLND has an advantage over rHAT in that it allows users to customize the output formats.

Recently, two new simulator tools have been introduced in nanopore sequencing platform—NanoSim and DeepSimulator. NanoSim is developed based on Python for simulation purpose and analyzing the read length [47]. It examines the experimental ONT reads to model read specifications including the sequence length distributions and error profiles. It then implements these characteristics to produce *in silico* reads that serves as an input reference [47]. DeepSimulator can imitate the sequence reads from the statistical models of the data including both nucleotide reads and raw electrical current signals [48]. It can be applied to generate a guideline to examine the recently designed approaches for nanopore sequencing data analysis. This tool is comprised of different frameworks: sequence generation and formation of the simulated current raw signals [48]. The main difference between these two software programs is that ReadSim can perform both tasks including read simulation and performance prediction of genome sequence assembly, but SiLiCo is only capable of read simulation. However, SiLiCo has the great advantage of ensuring all the nucleotides have similar likelihoods of being chosen in a simulated read; it selects the start and end of a genomic coordinate using a buffer and this eventually preventing the occurrence of end-selection bias. In summary, SiLiCo is more user-friendly, as it allows users to supply the corresponding parameters for the desired genome coverage, the standard deviation and the mean of read length, compared to ReadSim.

Haplotype assembly tools

Haplotype assembly is one of the computational difficulties of rebuilding the haplotypes, which are the two parental copies in a diploid genome. Haplotype assembly also has an important role in measuring the allele-specific expression [49]. The complexity includes sequencing errors, which require higher usage of computer resources, and uneven coverage across transcripts, which may introduce false variants or cause true heterozygous variants to be discarded from the analysis [49]. Hence, a method with faster performance and memory-efficiency, called HapCol, was designed to overcome these problems in haplotype assembly. HapCol applies a fixed-parameter algorithm to the k-constrained minimum error correction problem, a recently developed variant of the weighted Minimum Error Correction (wMEC) issue that takes into consideration the important features of future high-throughput sequencing technologies, including the increased read lengths and the constant distributions of sequencing errors. HapCol can be applied to the long reads in both platforms. It is important to have reads that are long enough to span numerous distinct heterozygous locations in order to accurately assemble the haplotype reads [50]. HapCol excludes the traditional all-heterozygous assumption and hence it phases the data sets with a much higher coverage. There are a few other methods that cannot process long reads or coverage greater than 20×, however, this software is capable of performing the task with data sets including both long reads (exceed 100 000 bp long) and coverage up to 25×, on standard workstations/small servers [50]. According to the error model, users can apply different error distributions by choosing the maximum number (k) of errors per position. Furthermore, it can also be effortlessly adapted by setting a higher value for k until a useful solution is obtained. Even if the average error rate is low, for example in data from the existing Illumina sequencing technologies, this approach greatly reduces the impact of systematic sequencing errors on the performance when processing the data sets [50]. In summary,

HAPCOL overcomes the traditional heterozygous assumption and processes data sets with coverage of 25× on standard workstations/small servers and the value of k-mers can be adjusted easily based on the user's requirement and thus it is more flexible.

Error correction tools

The error-correction stage is an essential step in numerous analyses including sequence assembly, haplotype interference and single nucleotide variant calling. It is important that the errors produced by common high-throughput sequencing platforms be categorized in order to generate high-quality reads. Error correction approaches belong to two main categories that include *de novo* and hybrid methods. Hybrid approaches use short and long reads data, while non-hybrid methods such as *de novo* self-correct reads by exploiting overlap of high-coverage data. We included five hybrid correctors—LORDEC, proovread, Jabba, LSC and PacBioToCA—and two *de novo* correctors—PacBioToCA and LORMA—in this review. PacBiotoCA, proovread and LORDEC utilized the long reads only from the SMRT platform for error correcting in whole genome sequence analysis. In addition, there are two tools, LoRMA and Jabba, which are used in both SMRT and ONT long reads for error correction. MultiBreak-SV can be used in different platforms for error correction.

PacBioToCA is part of the module in the Celera Assembler software package that functions by mapping shorter, high accuracy reads onto the long reads [51]. The strategy consists of two stages: a long-read correction phase and an assembly phase. Both are implemented as part of the Celera Assembler, but the output of the correction phase can be used as input to any other analysis or assembler capable of utilizing long FastA sequences. The outline of the correction algorithm is as follows: (1) high-identity short-read sequences are simultaneously mapped to all long-read sequences; (2) repeats are resolved by placing each short-read sequence in its highest identity repeat copy; (3) chimera and trimming problems are identified and corrected within the long-read sequences; and (4) based on a multiple alignment of the short-read sequences, a consensus sequence is calculated for each long-read sequence [51].

proovread is a hybrid correction pipeline and mapping-based approach for SMRT reads. They correct the long reads first by mapping the short reads on long reads and correct them based on a consensus built on the mapped short reads. proovread-corrected sequences were longer, and the throughput was higher. Thus, proovread combines the most accurate correction results with an excellent adaptability to the available hardware. Therefore, it will enhance the performance of SMRT sequencing [52]. LoRDEC is a hybrid error correction method that builds a succinct DBG representing the short reads and seeks a corrective sequence for each erroneous region in the long reads by traversing chosen paths in the graph [53]. In comparison, LoRDEC is at least six times faster and requires at least 93% less memory or disk space than available tools, while achieving similar accuracy [53]. Compared with other correction algorithms, LoRDEC offers a novel graph-based approach. Path searching in a DBG allows for handling higher error rates. However, this search can fail if either no path or too many paths exist between the source and target k-mers. PacBioToCA generates higher quality assemblies with fewer errors and gaps than proovread as the goal of proovread is not generating high accuracy reads or reducing the cost of sequencing; instead, it was developed to run on standard computers as well as computer grids/independent of the computing infrastructure, and it can be easily adapted to various use cases. proovread is more flexibly adapted than PacBioToCA and LoRDEC on existing hardware infrastructure from a laptop to a high-performance computing cluster. However, LoRDEC has great advantages over the other two software programs, as it allows trimming processes, is less bias when correcting SMRT reads and uses memory efficiently.

Jabba is a hybrid approach for error correction in long third-generation reads by mapping them on a corrected DBG that was constructed from second generation data [54]. The main difference is this software uses a pseudo alignment approach with a seed-and-extend methodology, using maximal exact matches (MEMs) as seeds. It applies a pseudo alignment approach based on a seed-and-extend methodology. The seeds are MEM between an individual read and a node of the graph. New algorithms, based on DBG, were specifically designed to efficiently integrate with the assembly of huge amounts of NGS data [54]. Overlap between short reads is then established in linear time between reads that share a k-mer. Overall, the pseudo alignment with MEMs is a fast and reliable method to map long highly erroneous sequences on a DBG. Jabba is faster, is highly reliable on the generated aligned reads, generates higher accuracy as many of the aligned reads are error-free and has lower usage of CPU time.

LSCplus was specifically designed to apply a hybrid sequencing approach that combines NGS and SMRT data which improves long reads accuracy by short read alignment [55]. LSCplus is designed for RNA-seq analysis. Due to the high error rate in PacBio long reads, hybrid sequencing is required. The original algorithm in the error correction step of LSC was optimized in LSCplus [55]. During the error correction process, if a specific position only covers a few different bases, the program cannot decide which one is real. By increasing the coverage depth, the number of true positives is increased, increasing the true positive rate. Overall, LSCplus is applicable for both long and short reads; however, LoRMA and Jabba are only suitable for long reads.

MultiBreak-SV is an algorithm to identify SVs from single molecule sequencing data, paired read sequencing data or a combination of sequencing data from different platforms [56]. MultiBreak-SV applies a probabilistic approach to reduce the error rates in sequencing. A study also showed that MultiBreak-SV can determine SVs with high sensitivity and specificity by applying to PacBio data from four human fosmids [56]. LoRMA is used for error correction in long reads only. There are two steps involved in LoRMA: first, an iterative alignment-free correction method is used based on DBG with increasing length of k-mers; and second, the long-distance dependencies determined using multiple alignments are used to further improve the corrected reads [57]. The method demonstrates that efficient alignment free methods can be implemented on highly erroneous long-read data [57].

Discussion

Three main advantages of TGS technologies: it is fast, is easy and produces much longer reads. The availability of long reads will have a major impact on genomics studies involving the process of assembling. Assembling genomes solely based on short reads, without any available reference genome remains a challenge [58]. The long reads have proved invaluable for achieving high-quality assemblies because they span proportionally more of the repeats present in a genome.

Long-read assemblers implement an overlap graph or string graph approach that begins by comparing the entire long reads to

each other. In Cerulean, long reads are used to find the best path in the DBG that bridges the gaps between large contigs. Although these software packages have achieved important advances for TGS genome assembly, resolving intricate ambiguities is inherently difficult. Furthermore, the underlying graph search algorithms usually have exponential complexity with respect to the search depth; highly repeating regions (such as long repeats of simple sequences) will lead to large search depths and are not resolvable. In addition, the more powerful read overlap graph structure (of the long reads) was not fully explored in all these approaches. Generally, these algorithms depend on heuristics such as contig lengths and iterations are required.

Compared to the string graph approach, hybrid strategies that associate with NGS data are more effective when a limited amount of long-read coverage is available, especially below 30× coverage, whereas self-correction is better suited to higher sequencing coverage because more reliable alignments can be made between the long reads. For example, HGAP was developed using a non-hybrid strategy to assemble SMRT sequencing data, which does not require the usage of NGS short reads. HGAP contains a consensus algorithm that generates long and highly accurate overlapping sequences by correcting errors on the longest reads using shorter reads from the same library. This correction approach was proposed earlier in the hybrid setting and is widely implemented in assembly pipelines. However, this non-hybrid, hierarchical assembly technique needs relatively high sequencing coverage (50–100×) and substantial error correction time to acquire adequate results. It is notable to mention that most of the algorithms we reviewed in HGAP were originally designed for bacterial-sized genomes. Though recent advancements in aligning erroneous long reads have also shortened the computational time of TGS assembly, running these programs on large genomes, particularly mammalian-sized genomes, normally requires a huge computational burden more appropriate to large computational clusters.

To help scientists in choosing the appropriate TGS tool(s) for genomic studies using TGS, we summarize our discussion for whole genome sequencing analyses and *de novo* assembly analyses tools collected in this paper based on different TGS platforms. For the whole genome sequencing analyses tools used in the SMRT platform, both rHAT and LongISLND are applicable for read alignment. LongISLND is more ideal for read alignment if the hash table is not required. In the ONT platform, Nanocall is an effective tool for base calling and MarginAlign is applicable for read alignment using ONT long reads. HapCol, SiLiCo and readSim are applied using the long reads from both platforms. HapCol is relevant to haplotype assembly, while SiLiCo and readSim are applicable for read simulation. ReadSim is suited to both long and short reads; however, SiLiCo is only used for long reads. Furthermore, for the *de novo* assembly tools using the SMRT platform, HGAP, PBJelly and HINGE are suitable for assembling bacterial-sized genomes, while MHAP is used for assembling mammalian genomes. FALCON is the only SMRT diploid-aware assembler which allows researchers to investigate of haplotype structure and heterozygous structural variation of the genome sequence. In the ONT platform, PoreSeq is an efficient tool for assembling the genomes if the sequence variants contain low-coverage regions. Minimap/miniasm is suitable for assembling the long reads from both platforms without any correction stage. Circlator is another tool that applies the SMRT and ONT long reads for circular genome assemblies. Numerous tools including GMcloser, OPERA-LG, Nanocorr, DBG2LOC, NaS and Cerulean implement a hybrid-approach to assemble the reads from both TGS and NGS. GMcloser works for gap closing while OPERA-LG is developed for scaffolding assembly. Nanocorr is designed to generate a *de novo* assembly with a built-in error correction algorithm. DBG2LOC is specific for assembling eukaryotic-sized genomes while both NaS and Cerulean are more suitable for bacterial-sized genomes.

The error correction stage is one of the challenges in TGS genome assembly. Mapping phase of the error correction method is generally involved in processing the sequencing reads through mapping them to a reference genome or aligning the reads to other sequence to form a potential overlap. Bad alignments are normally caused by the noise presented in the error reads. Bad alignments are normally caused by noise introduced from errors in the reads. These low low-quality alignments may then be removed from the downstream analysis and thus result in loss of important information. This can be difficult specifically when examining the low low-quality reads in low-coverage genomic regions. Therefore, error correction methods can be implemented to overcome all these difficulties. For example, the tools implemented in TGS include proovread, PacBioToCA and LSC. proovread is more flexible than these other two software programs. Although LSC was developed mainly for the correction of (human) transcriptomic data, PacBioToCA can handle different data sets, but is part of the Celera WGS pipeline and requires the installation of the complete package. LSC does not trim the data but both PacBioToCA and proovread can trim the data. To give the user maximum flexibility, proovread also reports the untrimmed corrected reads. Furthermore, the trimming step is independent of the correction, thereby enabling the user to easily optimize the trimming parameters for the given data set. All the alignments from the sequencing can be optimized by correcting the errors in the reads, resulting in higher accuracy and quality of the alignments, and ultimately leading to better downstream analysis. Currently, a new generation of technologies is pushing the limit even further, producing the resolution of single nucleic acid molecules in shorter time. For example, an ultra-fast mapping, error correction and *de novo* assembly tool MECAT was developed for single-molecule sequencing reads, which could be deployed in personal computers [59]. By integrating both novel computational algorithms and new technological characteristics, this is an iterative procedure of establishing high resolution TGS computational tools. However, all these procedures need committed efforts from collaborations between industrial and academic scientists, which may aid in performing the sequencing with much higher efficiency and accuracy.

Key points

- TGS technologies hold the promise of longer read lengths; they have been implemented to generate highly accurate *de novo* assemblies of hundreds of species of genomes, providing new insights into evolution and sequence diversity.
- We evaluated various tools applied in three main TGS platforms: PacBio, SMRT sequencing, ONT sequencing and BioNano sequencing. We discussed their various characteristics, such as the required input, interaction with the user, sequencing platforms, type of reads, error models, possibility of introducing coverage bias, simulation of genomic variants and output provided. This was done within the framework of potential

applications, providing readers with guidelines for the identification of the TGS *de novo* software applications that are best suited for their purposes.
- We presented two distinct decision trees to guide researchers for selecting a suitable TGS *de novo* and whole-genome sequencing analysis tools.

Supplementary Data

Supplementary data are available online at https://academic.oup.com/bfg.

Acknowledgement

This work was supported by the research start-up fellowship of University of Sunshine Coast to M.Z.

References

1. Reuter JA, Spacek DV, Snyder MP. High-throughput sequencing technologies. *Mol Cell* 2015;**58**:586–97.
2. Margulies M, Egholm M, Altman WE, *et al*. Genome sequencing in microfabricated high-density picolitre reactors. *Nature* 2005;**437**:376.
3. Bennett S. Solexa Ltd. *Pharmacogenomics* 2004;**5**:433–8.
4. Shen R, Fan JB, Campbell D, *et al*. High-throughput SNP genotyping on universal bead arrays. *Mutat Res* 2005;**573**:70–82.
5. Alkan C, Sajjadian S, Eichler EE. Limitations of next-generation genome sequence assembly. *Nat Methods* 2011;**8**:61–5.
6. Denton JF, Lugo-Martinez J, Tucker AE, *et al*. Extensive error in the number of genes inferred from draft genome assemblies. *PLoS Comput Biol* 2014;**10**:e1003998.
7. Pareek CS, Smoczynski R, Tretyn A. Sequencing technologies and genome sequencing. *J Appl Genet* 2011;**52**:413–35.
8. Stankova H, Hastie AR, Chan S, *et al*. BioNano genome mapping of individual chromosomes supports physical mapping and sequence assembly in complex plant genomes. *Plant Biotechnol J* 2016;**14**:1523–31.
9. Nakano K, Shiroma A, Shimoji M, *et al*. Advantages of genome sequencing by long-read sequencer using SMRT technology in medical area. *Hum Cell* 2017;**30**:149–61.
10. Lu H, Giordano F, Ning Z. Oxford Nanopore MinION sequencing and genome assembly. *Genomics Proteomics Bioinformatics* 2016;**14**:265–79.
11. Jain M, Olsen HE, Paten B, *et al*. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. *Genome Biol* 2016;**17**:239.
12. Magi A, Semeraro R, Mingrino A, *et al*. Nanopore sequencing data analysis: state of the art, applications and challenges. *Brief Bioinform* 2017. doi.org/10.1093/bib/bbx062.
13. Leggett RM, Clark MD. A world of opportunities with nanopore sequencing. *J Exp Bot* 2017;**68**:5419–29.
14. de Lannoy C, de Ridder D, Risse J. The long reads ahead: *de novo* genome assembly using the MinION. *F1000Res* 2017;**6**:1083.
15. Sohn JI, Nam JW. The present and future of de novo whole-genome assembly. *Brief Bioinform* 2018;**19**:23–40.
16. Henson J, Tischler G, Ning Z. Next-generation sequencing and large genome assemblies. *Pharmacogenomics* 2012;**13**:901–15.
17. Mak AC, Lai YY, Lam ET, *et al*. Genome-wide structural variation detection by genome mapping on nanochannel arrays. *Genetics* 2016;**202**:351–62.
18. Berlin K, Koren S, Chin CS, *et al*. Assembling large genomes with single-molecule sequencing and locality-sensitive hashing. *Nat Biotechnol* 2015;**33**:623–30.
19. English AC, Richards S, Han Y, *et al*. Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology. *PLoS One* 2012;**7**:e47768.
20. Chin CS, Alexander DH, Marks P, *et al*. Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. *Nat Methods* 2013;**10**:563–9.
21. Chin CS, Peluso P, Sedlazeck FJ, *et al*. Phased diploid genome assembly with single-molecule real-time sequencing. *Nat Methods* 2016;**13**:1050–4.
22. Zimin AV, Puiu D, Luo MC, *et al*. Hybrid assembly of the large and highly repetitive genome of Aegilops tauschii, a progenitor of bread wheat, with the MaSuRCA mega-reads algorithm. *Genome Res* 2017;**27**:787–92.
23. Kamath GM, Shomorony I, Xia F, *et al*. HINGE: long-read assembly achieves optimal repeat resolution. *Genome Res* 2017;**27**:747–56.
24. Jayakumar V, Sakakibara Y. Comprehensive evaluation of non-hybrid genome assembly tools for third-generation PacBio long-read sequence data. *Brief Bioinform* 2017. doi.org/10.1093/bib/bbx147.
25. Szalay T, Golovchenko JA. De novo sequencing and variant calling with nanopores using PoreSeq. *Nat Biotechnol* 2015;**33**:1087–91.
26. Goodwin S, Gurtowski J, Ethe-Sayers S, *et al*. Oxford Nanopore sequencing, hybrid error correction, and de novo assembly of a eukaryotic genome. *Genome Res* 2015;**25**:1750–6.
27. Li H. Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. *Bioinformatics* 2016;**32**:2103–10.
28. Hunt M, Silva ND, Otto TD, *et al*. Circlator: automated circularization of genome assemblies using long sequencing reads. *Genome Biol* 2015;**16**:294.
29. Koren S, Walenz BP, Berlin K, *et al*. Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome Res* 2017;**27**:722–36.
30. Gao S, Bertrand D, Chia BK, *et al*. OPERA-LG: efficient and exact scaffolding of large, repeat-rich eukaryotic genomes with performance guarantees. *Genome Biol* 2016;**17**:102.
31. Ye C, Hill CM, Wu S, *et al*. DBG2OLC: efficient assembly of large genomes using long erroneous reads of the third generation sequencing technologies. *Sci Rep* 2016;**6**:31900.
32. Madoui MA, Engelen S, Cruaud C, *et al*. Genome assembly using nanopore-guided long and error-free DNA reads. *BMC Genomics* 2015;**16**:327.
33. Deshpande V, Fung ED, Pham S, *et al*. Cerulean: a hybrid assembly using high throughput short and long reads. *Springer* 2013;**8126**:349–63.
34. Bao S, Jiang R, Kwan W, *et al*. Evaluation of next-generation sequencing software in mapping and assembly. *J Hum Genet* 2011;**56**:406–14.
35. Liu B, Guan D, Teng M, *et al*. rHAT: fast alignment of noisy long reads with regional hashing. *Bioinformatics* 2016;**32**:1625–31.

36. Jain M, Fiddes IT, Miga KH, *et al*. Improved data analysis for the MinION nanopore sequencer. *Nat Methods* 2015;**12**: 351–6.
37. David M, Dursi LJ, Yao D, *et al*. Nanocall: an open source basecaller for Oxford Nanopore sequencing data. *Bioinformatics* 2017;**33**:49–55.
38. Technologies ON. *Albacore*. https://github.com/Albacore/albacore. (5 May 2018, date last accessed).
39. Technologies ON. *Scrappie*. https://github.com/nanoporetech/scrappie. (15 April 2018, date last accessed).
40. Boza V, Brejova B, Vinar T. DeepNano: deep recurrent neural networks for base calling in MinION nanopore reads. *PLoS One* 2017;**12**:e0178751.
41. Senol Cali D, Kim JS, Ghose S, *et al*. Nanopore sequencing technology and tools for genome assembly: computational analysis of the current state, bottlenecks and future directions. *Brief Bioinform* 2018. doi.org/10.1093/bib/bby017.
42. Loman NJ, Quick J, Simpson JT. A complete bacterial genome assembled de novo using only nanopore sequencing data. *Nat Methods* 2015;**12**:733–5.
43. Vaser R, Sovic I, Nagarajan N, *et al*. Fast and accurate de novo genome assembly from long uncorrected reads. *Genome Res* 2017;**27**:737–46.
44. Baker EAG, Goodwin S, McCombie WR, *et al* . SiLiCO: a simulator of long read sequencing in PacBio and Oxford Nanopore. *bioRxiv* 2016:1–3. doi.org/10.1101/076901.
45. Lee H, Gurtowski J, Yoo S, *et al*. Error correction and assembly complexity of single molecule sequencing reads. *bioRxiv* 2014. doi:10.1101/006395.
46. Lau B, Mohiyuddin M, Mu JC, *et al*. LongISLND: in silico sequencing of lengthy and noisy datatypes. *Bioinformatics* 2016;**32**:3829–32.
47. Yang C, Chu J, Warren RL, *et al*. NanoSim: nanopore sequence read simulator based on statistical characterization. *Gigascience* 2017;**6**:1–6.
48. Li Y, Han R, Bi C, *et al*. DeepSimulator: a deep simulator for nanopore sequencing. *Bioinformatics* 2018;**34**:2899–2908.
49. Cao H, Wu H, Luo R, *et al*. *De novo* assembly of a haplotype-resolved human genome. *Nat Biotechnol* 2015;**33**: 617–22.
50. Pirola Y, Zaccaria S, Dondi R, *et al*. HapCol: accurate and memory-efficient haplotype assembly from long reads. *Bioinformatics* 2016;**32**:1610–7.
51. Koren S, Schatz MC, Walenz BP, *et al*. Hybrid error correction and de novo assembly of single-molecule sequencing reads. *Nat Biotechnol* 2012;**30**:693–700.
52. Hackl T, Hedrich R, Schultz J, *et al*. proovread: large-scale high-accuracy PacBio correction through iterative short read consensus. *Bioinformatics* 2014;**30**:3004–11.
53. Salmela L, Rivals E. LoRDEC: accurate and efficient long read error correction. *Bioinformatics* 2014; **30**:3506–14.
54. Miclotte G, Heydari M, Demeester P, *et al*. Jabba: hybrid error correction for long sequencing reads. *Algorithms Mol Biol* 2016;**11**:10.
55. Hu R, Sun G, Sun X. LSCplus: a fast solution for improving long read accuracy by short read alignment. *BMC Bioinformatics* 2016;**17**:451.
56. Ritz A, Bashir A, Sindi S, *et al*. Characterization of structural variants with single molecule and hybrid sequencing approaches. *Bioinformatics* 2014;**30**:3458–66.
57. Salmela L, Walve R, Rivals E, *et al*. Accurate self-correction of errors in long reads using de Bruijn graphs. *Bioinformatics* 2017;**33**:799–806.
58. Lee H, Gurtowski J, Yoo S, *et al*. Third-generation sequencing and the future of genomics. *bioRxiv* 2016. doi:10.1101/048603.
59. Xiao C, Chen Y, Xie SQ, *et al*. MECAT: fast mapping, error correction, and de novo assembly for single-molecule sequencing reads. *Nat Methods* 2017;**14**:1072–4.