DESIGN AND EVALUATION OF TWO HYBRID GENOME ASSEMBLY APPROACHES USING ILLUMINA, ROCHE 454, AND PACBIO DATASETS

A Paper
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Liren Sun

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Department:
Computer Science

November 2016

Fargo, North Dakota

North Dakota State University
Graduate School

Title

DESIGN AND EVALUATION TWO HYBRID GENOME ASSEMBLY APPROACHES USING ILLUMINA, ROCHE 454 AND PACBIO DATASETS

By

Liren Sun

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Changhui Yan
Chair

Dr. Anne M. Denton

Dr. Shaobin Zhong

Approved:

12/8/2016
Date

Dr. Brian M. Slator
Department Chair

ABSTRACT

The assembly of next-generation sequencing reads is one of the most challenging and important tasks in bioinformatics. There are many different types of assembly algorithms and programs that have been developed to assemble next-generation sequencing reads. However, the assembly quality of each assembly program may vary. This paper introduces and implements two different assembly approaches that use three types of next-generation sequencing datasets. Both assembly approaches are designed to achieve the same goal, which is to improve assembly quality. The assembly results from the two approaches were compared and evaluated by using some widely used quality metrics. The result shows each approach has advantages and disadvantages.

ACKNOWLEDGEMENTS

First and foremost, I would like to give my sincere gratitude and deepest appreciation to Dr. Changhui Yan, my advisor, with extraordinary patience and consistent encouragement, gave me great help in my paper writing. Then, I am also thankful to all my committee members, Dr. Shaobin Zhong, Dr. and Anne M. Denton for their invaluable time and guidance. Last my thanks would go to my beloved parents, and my girlfriend, for their unconditional support and love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

This paper evaluates the differences in genome assembly quality between two hybrid genome assembly methodologies applied to a sample, which has three different types of datasets generated from Illumina, Roche 454, and PacBio technologies.

The raw files from Illumina and Roche sequencing technologies are composed of many small pieces of DNA which are called reads. The length of reads ranges from 20 to 3000 bases. Each read contains extremely limited biological information because even the simplest chromosome has 130,000 base pairs [1]. Genome assembly is the process to assemble reads to reconstruct the genome sequences.

Some of the sequences assembly programs are designed for processing reads generated by only one type of sequencing platform. In contrast, the hybrid genome assembly programs utilize reads from multiple types of sequencing technologies. It has been shown that hybrid genome assembly methods can improve the assembly quality over methods using a single sequencing technology dataset [5].

Most of the hybrid genome assembly programs required at least one Illumina Paired-end dataset and one Illumina Mate-pair dataset as inputs [3]. Paired-end sequencing and Mate-pair sequencing are two types of sequencing technologies which generate high-quality reads from two sides of a fragment. The major difference between Paired-end and Mate-pair sequencing is the distance between the reads in a pair. In fact, some genome assembly tasks require several separate insert-sizes of Illumina Mate-pair datasets to support the whole assembly process. , Some hybrid genome assembly strategies use a library of long reads, e.g. Roche 454 Single End or PacBio RS datasets, instead of an Illumina Mate-pair dataset [4].

In this paper, two hybrid genome assembly methodologies were designed that used three datasets, Illumina Paired-end, Roche 454 Single End, and PacBio RS, to assemble genome. One was called the Rescaffolding Hybrid Assembly Approach [24, 25, 26], and the other was Cerulean Hybrid Assembly Approach [27]. Compared with the ABySS Hybrid DNA assembly approach, the Cerulean Hybrid Assembly methodology implements the Cerulean assembly tools [27] to extend an assembled sequence file which is assembled using the ABySS assembly tool. The following section of this chapter gives an introduction to the next-generation sequencing technologies and genome assembly. Chapter 2 presents the results of each assembly approach and discussion. Chapter 3 provides a summary about the comparison between two assembly approaches.

1.1. Next-generation Sequencing Technologies

The process of decoding the DNA sequence of an organism is called sequencing [15]. Next-generation sequencing (NGS) technologies are also called "high-throughput" sequencing technologies. The NGSs are a collection of many sequencing technologies developed over the recent decade. The NGS technologies process a large amount of genome sequences in parallel, and generate millions, even billionsof reads at one run [8]. The reads generated by NGS technologies are much shorter than those from traditional Sanger sequencing. But they provide higher coverage which can help overcome sequencing errors. NGS technologies are much faster and less costly than traditional Sanger sequencing [9]. The three types of commonly used NGS technologies which are related to this paper are described as follows:

1.1.1. Illumina Paired-end

The Illumina sequencing platform provides high-throughput sequencing for both DNA and RNA. The Illumina sequencing method is based on Illumina dye-terminators technology

[11]. The accuracy of Illumina sequencing is very high with a 99.9% accuracy rate for a single read. After fragmentation, Illumina sequences each DNA fragment from both 5-end and 3-end, and produces two reads. For each run the read length can be set between 75 to 300 base pair (bp). The reads and their sequencing quality scores are saved in two files in the fastq formats, which one file containing reads from one DNA strand. Some assembly programs need the sequencing quality values to assemble sequence reads to avoid errors from raw data.; Inserted size is another important parameter in Paired-end sequencing. The insert size is the sum of the lengths of the two reads and the length the unsequenced region in between. Insert size is smaller than the fragment size, which is the sum of insert size and the sizes of Illumina adapters. When the insert size is between 200 to 500 bp (short insert size), the Illumina sequencing is commonly called Paired-end sequencing. When the insert size in the range of 2 to 3 kbp (long insert size), the Illumina sequencing is called Mate-pair sequencing. Output dataset from Illumina Mate-pair sequencing is commonly used in the scaffolding process of genome assembly [12].

1.1.2. Roche 454 Single End

Roche 454 [28] Single End has the same accuracy rate as Illumina platform. But Roche 454 Single End only produces one read for each DNA fragment. The length of the reads is around 700 bp. It is also considered to be one of the best input datasets for de novo assembly. In the hybrid assembly, Roche 454 Single End reads are used as long reads to construct contigs and scaffolds [5]. The cost of Roche 454 Single-end sequencing is higher than Illumina sequencing, but it is still an affordable price and much less costly than Sanger method.

1.1.3. PacBio RS

The real name of PacBio sequencing is Single Molecule Real Time sequencing [29]. This sequencing technology belongs to Pacific Biosciences Corporation (PacBio). The maximum read

length produced by PacBio sequencing can be around 40,000 bases, which is the longest read length in all NGS technologies. PacBio sequencing is also as fast and cheap as other NGS technologies. However, the single-read accuracy of PacBio sequencing is only 87% [13]. The long reads can be used in hybrid genome assembly to scaffold, close gaps or extend existing scaffolds [14].

1.2. Genome Assembly

When the reference genome of a species is available, the assembly of individual genomes can be done by aligning NGS reads to the reference genome. This is possible because the difference between two individual genomes of one species is small. For example, the genome of one human being is 99.5% identical to that of another human beings [16]. When the reference genome a species is unavailable, the assembly of individual genomes needs to start from scratch. This type of assembly is called “de novo assembly.” De novo means start from scratch.

If a suffix of a read is identical to the prefix of another read, then there is an overlap between the two reads. The overlap is the glue that assembles sequencing reads into genomes. In some cases, the suffix of a read is not exactly the same as the next prefix, but the two reads are still considered to be overlapped. This is because sequencing errors and polyploidy can cause a difference of one or two bases between reads. The errors can be solved using the quality scores in the fastq files. The coverage is another important parameter in DNA sequencing and genome assembly. The coverage is the number of reads that cover a genome location. It can be calculated as ration of the size of a sequencing-reads dataset to the genome size. For instance, the genome size is 3000 bases, and the sequencing-reads dataset is 9000 bases. The average coverage for this dataset is 3-fold. More coverage leads to more and longer overlaps. But the cost of sequencing will increase too.

There are two commonly used assembly algorithms in genome assembly: Greedy shortest common superstring and Eulerian walks. In the Greedy shortest common superstring, genome assembly is treated as a shortest common superstring problem. The reads and the overlap information will be saved into a direct graph. The nodes are the reads. The edges are the overlapping information between two reads. Without using the Greedy algorithm, the only way to solve the assembly problem is to enumerate all possible orderings of reads and find the shortest superstring. However, it is an NP-complete problem which is computationally intractable. The greedy algorithm addresses this issue very quickly. But the superstring produced by the greedy algorithm may not be the shortest superstring.

The repeats in genome make it very difficult to assemble reads. When the genome has a replicated portion and the length of the sequencing read is the same as the repeated region, it will tend to over collapse the repeats. This approach is also called an Overlap-Layout-Consensus approach. The shortest common superstring is suitable for Sanger sequencing data, but it is not for the next generation sequencing data. Not only is the reads length of next generation sequencing much shorter, but also NGS have a high coverage number which means the number of nodes are too big.

Eulerian walks are good for most of NGS data assembly. Given a graph, an Eulerian walk is a path which passes every node and only goes through each edge once in a graph. When using Eulerian walks, it is necessary to construct a De Bruijn graph to save all overlap information. The De Bruijn graph has a different structure than the overlapping direct graph which is used in the shortest common superstring method. In a De Bruijn graph, the nodes are all possible substrings with a length of "k" in every sequencing read. We call these substrings "k-mers". The choice of the value of k is critical for a genome assembly, and it directly affects the assembly

quality. The sequencing errors can cause dead-ends in the De Bruijn graph, and the polyploidy can produce bubbles in the graph. Therefore, an assembly program will need to refine the De Bruijn graph to remove these abnormal structures. Most genome assembly programs use the Eulerian walks method, such as the Allpaths-LG [30], ABySS [26], Velvet [32], Soap de Novo [33]. After a Eulerian-walk is found on the De Bruijn graph, many contiguous assembled pieces of DNA, called contigs, are generated. Using paired-end or mate-end reads, the relative orientation and distance between some contigs could be defined. A set of contigs for which the relative orientation and distance between them are known is called a "scaffold." Scaffolds are usually the final output of a genome assembly program.

The output file from a genome assembly is called a "draft" assembly. The ideal draft assembly has a low number of fragments and longer contig reads. The maximum, average, and median contig size are parameters commonly used to assess the assembly quality. The N50 is another widely used statistic to assess the assembly quality. The definition of N50 is the shortest reads length at 50% of draft assembly [17]. When the reference genome is available, the draft assembly can be mapped to the reference genome and then evaluation tools can be used to check the overall quality of the assembly. The commonly used evaluation tools are CGAL [34] and Quast [35].

CHAPTER 2. RESULTS AND DISCUSSION

2.1. Input Datasets

The datasets used in this paper are stored in the NCBI SRA database under the Name of CF080- SRP010852 [5]. There are three types of reads which are PacBio RS, Illumina Paired-end, and the Roche 454 Single End reads. The name of the species from which the reads were obtained is Rhizobium sp.CF080, which is a bacterium. Those sequencing files were trimmed and ready to be assembled [7]. The basic dataset information is shown on Table 1. (1 Mb = 1,000,000 bp; 1 Gb = 1,000 Mb)

Table 1. The basic information of three input datasets

CF080- SRP010852	Illumina PE	Roche 454 SE	PacBio RS
Number of Bases	3.9 Gb	437.6 Mb	513.9 Mb
Average Read length	100 bp	670 bp	6749 bp
Sequencing Coverage	553-fold	62-fold	72-fold

The reference genome and gene annotation of Rhizobium_sp.CF080 can be downloaded from www.bacteriaensemble.org, and the name for the reference genome is CF080_Reference.fa. The total base length of reference genome is 7,049,533 bp. Sequencing coverage of the three datasets are ideal for genome assembly. The Illumina PE dataset has two fastq files, named CF080_IlluminaPE_1.fq, and CF080_IlluminaPE_2. fq respectively. The insert size of the Illumina PE dataset is 300 bp. The Roche 454 SE dataset has only one file that is named CF080_454.fq. The file's name for the PacBio RS dataset is CF080_PacBio.fq.

2.2. Rescaffolding Hybrid Assembly Approach

A recent study compared various genome assembly tools showing that the overall quality of hybrid assembly is better than traditional assembly which uses only one type of sequencing dataset [5]. Previous studies also showed that an Illumina Mate-pair dataset was very important in hybrid assembly to achieve a high-quality draft genome [5] [18]. The Illumina Mate-pair dataset is not only used to construct a De Bruijn graph at the early stage of hybrid assembly, but also can be utilized for scaffolding and gap closing in the later stages to improve the assembly's quality. High accuracy long reads, like those from the Roche 454 dataset, can replace the Illumina Mate-pair reads in the latter stages of hybrid assembly, but its assembly quality is lower than using Mate-pair datasets.

The rescaffolding function from ABySS (version 2.0.2) allows high-quality hybrid assembly using long reads.Unlike using a long distance Mate-pair library to do scaffolding, the rescaffolding function allows for a different way to link the contigs toconstruct scaffolds. The rescaffolding function needs to use the BWA-MEM [36], which is a long-read support alignment software. In the rescaffolding stage, the scaffolds from a low-quality assembly are aligned with long reads, such as PacBio and Roche 454 reads. Based on the alignments information the scaffolds arelinked together to produce high-quality draft genome.

There are two steps in the rescaffolding hybrid assembly approach. The first step is to perform a hybrid assembly on the Illumina PE and Roche 454 SE datasets. I tried many different ways to achieve the best assembly result in this step. The SOAP de novo (version 2.01) [37] and ABySS (version 2.0.2) [25] are the only assembly tools that can hybrid assemble Illumina PE and Roche 454 datasets. A demo dataset was successfully tested with both assembly tools in my Ubuntu 14 operating system. However, the SOAP de novo crashed while assembling the

CF080_IlluminaPE_1.fq, and CF080_IlluminaPE_2. fq, and CF080_454.fq. I tried changing several parameters such as the k-mer and cuffoff values of the config file, but it still crashed. SOAP de novo was abandoned at this step due to its unknown reason for crashing.

The ABySS supports multiple datasets and both fasta and fastq format data. It required at least one paired-end dataset. When doing the hybrid assembly on ABySS, the additional dataset can be either a paired-end or single-end dataset.

ABySS required a k-mer value be set to build the De Bruijn graph in the early stages of the genome assembly. The k-mer size can directly affect the resulting quality. A small k value will make it difficult to solve repeats, but it will increase edges (overlapping) [19]. A larger k value will cause the results to have many small-sized contigs, but make it easier to solve the repeat issue. There is no best k-mer value for all assembly tasks. Based on past experiences, the optimized k-mer is between half to two-thirds of the reads length. In this assembly work, the reads to construct a De Bruijn graph are from the CF080 paired-end dataset. The average read length of the CF080 paired-end dataset is 100 bp. So, the optimal k-mer value may be between 33 to 66. The ABySS only allows a k-mer size between 32 to 92. To find the optimal k value, the paired-end dataset was assembled many times using different k-mer sizes between 33 to 66. From each assembly, the contigs files were saved and marked with its own k-mer value. Then the abyss-fac function in ABySS selected an optimized k-mer value through evaluating each contig file. Figure 1 shows the automatic execute for these processes.

```
#!/bin/bash

for k in `seq 33 8 65`; do
    mkdir k$k
    abyss-pe np=8 --enable-maxk name=ABYSS_PE&454 k=$k in= CF080_IlluminaPE_1.fq CF080_IlluminaPE_2.fq '
done
abyss-fac k*/ABYSS_PE_454.fa
```

Figure 1. Scripting code for finding the best k-mer and results

K=49 has been chosen as the optimal k-mer for hybrid assembly of the CF080 PE and Roche 454 datasets. The assembly quality result is shown on Table 2, which also lists two other assembly results using different k values. Compared with the other two k-mer; the k=49 gives the best overall quality. When varying the k value, the N20, N50, N80, and Max Read length reached the maximum when at k=49.When k value increases, the number of contigs decreases. The scaffolds from k-mer=49 hybrid assembly were saved for the next step. The name of the scaffolds file is Abyss_PE_454_k49.fa.

Table 2. The result of three ABySS hybrid assembly by different k-mer

Quality \k-mer	**33**	**49**	**65**
Output File Name	Abyss_PE_454_k33.fa	**Abyss_PE_454_k49.fa**	Abyss_PE_454_k65.fa
N20	616,991bp	**1,117,354bp**	540813bp
N50	427,891bp	**544,947bp**	359,237bp
N80	213,799bp	**228,154bp**	226,814bp
Smallest Read Length	629bp	**526bp**	810bp
Largest Read Length	900,622bp	**1,126,114bp**	615,862bp
Number of contigss	265	**115**	88

The second step is to rescaffold the assembled scaffolds using the long-reads files. ABySS allows multiple long-reads datasets. During rescaffolding, the ABySS will first employ the BWA-MEM to do an alignment for assembled scaffolds and long reads. The BWA-MEM needs to be installed before rescaffolding. The reads from Abyss_PE_454_k49.fa, which has the

best assembly quality from the previous step, served as the low-quality scaffolds in rescaffolding.

Both Roche 454 SE and PacBio can be used as long reads for rescaffolding. Since the PacBio Sequencing technology has lower sequencing accuracy, 87%, in ABySS PacBio dataset can only be used as a long-reads file for rescaffolding, and cannot be used as single reads to build the assembly with paired-end reads during the previous step. To use PacBio reads as long reads in rescaffolding may have a negative effect on alignment due to the lower accuracy rate. I tried three different combinations of long-reads files for rescaffolding. The first combination has a Roche 454 SE dataset (CF080_454.fq). The second combination only has a PacBio dataset (CF080_PacBio.fq). The third one has both Roche 454 SE and PacBio datasets (CF080_454.fq, CF080_PacBio.fq). The quality result for each rescaffolding is listed in Table 3.

The N50 and Max contig Length are two most important quality metrics to evaluate assembly quality. The best overall result quality is from rescaffolding using only PacBio dataset as long read. Rescaffolding with two long-reads files, Roche 454 SE and PacBio did not improve the quality. Figure 1 shows the process of rescaffolding hybrid assembly approach.

Table 3. The result of Rescaffolding Hybrid Assembly Approach

Quality Matrices/long-reads files	**Roche 454 SE**	**PacBio**	**Roche 454 SE & PacBio**
Output File Name	AbyssPE454_Rescff454.fa	**AbyssPE454_RescffPacBio.fa**	AbyssPE454_Rescff454&PacBio.fa.fa
N20	1120672bp	**1120680bp**	1117354bp
N50	577209bp	**589362bp**	544947bp
N80	229,592bp	**230862bp**	228154bp
Smallest Read Length	526bp	**526bp**	526bp
Largest Read Length	1126114bp	**1127415bp**	1126114bp
Number of reads	105	**1299**	114

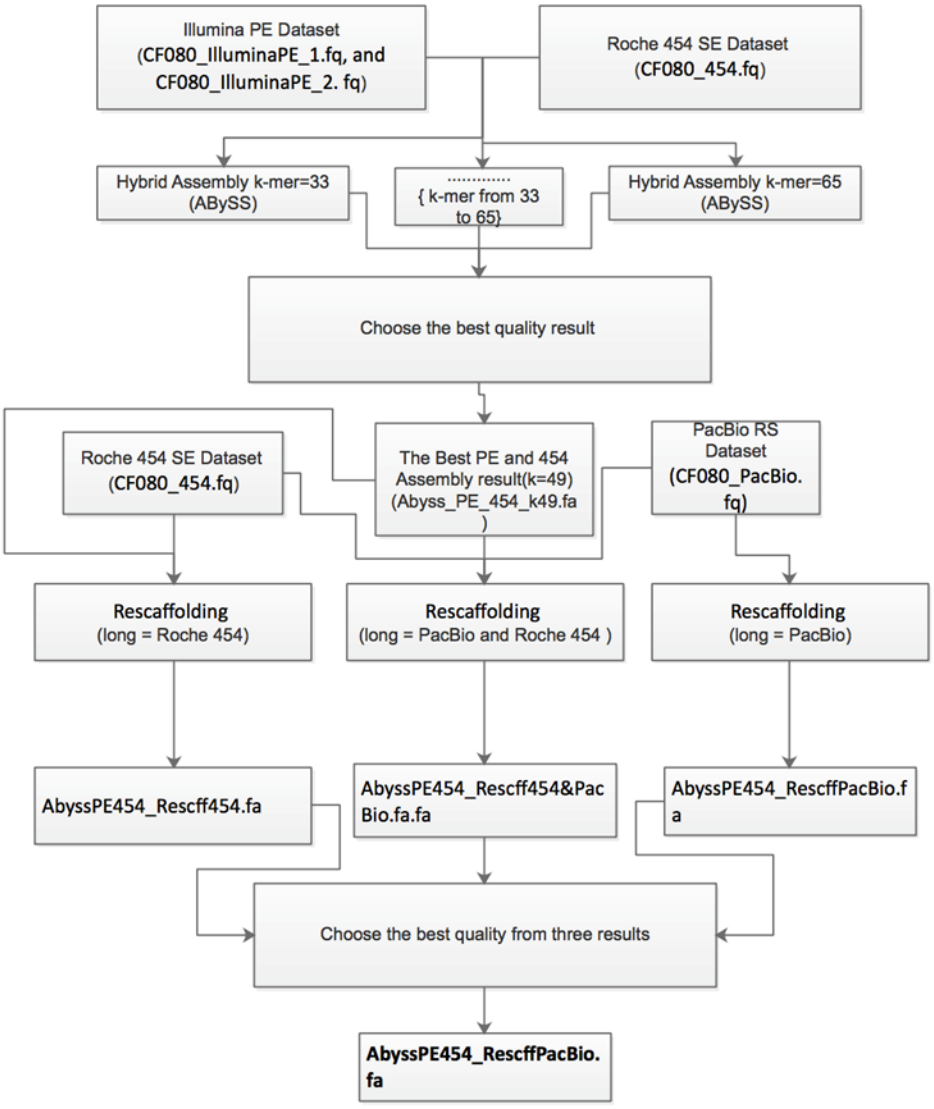

Figure 2. Rescaffolding Hybrid Assembly Approach overview

2.3. Cerulean Hybrid Assembly Approach

Cerulean Hybrid Assembly Approach uses in both Cerulean (Version 0.1) [27] and ABySS (Version 2.0.2) [31] to assemble PacBio long reads, Illumina Paired-end, and Roche 454 SE datasets. At first, ABySS hybrid assembled Illumina Paired-end and Roche 454 SE datasets, and the assembled contigs were mapped with reads from the PacBio RS dataset. Lastly, Cerulean

uses the mapping information to extend the assembled contigs. Cerulean (Version 0.1) is a genome assembly program specified for PacBio datasets. It uses the similar strategy like rescaffolding in ABySS, but Cerulean does not use the De Bruijn Graph. Instead, Cerulean uses a Skeleton Graph which is a simplified De Bruijn graph that ignores all intermediate short contigs[39]. Cerulean uses BLASR (Basic Local Alignment with Successive Refinement) [38] as the alignment tool that was designed for PacBio reads and is more suitable than the BWA-MEM. Cerulean can extend the contigs from ABySS using the PacBio datasets. There are three input files required in the Cerulean Hybrid Assembly approach. The best quality result from ABySS assembly of Illumina Paired-end and Roche 454 SE datasets are used as input (Abyss_PE_454_k49.fa). The Abyss_PE_454_k49.dot file is a graph structure file of Abyss_PE_454_k49.fa, and is also required as an input file for Cerulean. The last required data file is an alignment information file from PacBio reads (CF080_PacBio.fq) map to the assembled contigs (Abyss_PE_454_k49.fa) by BLASR (Basic Local Alignment with Successive Refinement

The high error rate from PacBio reads makes assembly difficult. There are two approaches to handling the PacBio read in assembly. To use the Paired-end short-reads dataset to correct the long reads from PacBio datasets is one approach. The biggest shortcoming of this approach is that it requires an enormous amount of computational resources [20]. Most workstations and desktops cannot handle it. The ALLPATH-LP [30] is a hybrid assembly program that has the best assembly quality for bacterial assemblies. (Required minimum is two Paired-end libraries). ALLPATH-LP implements the first approach which uses short reads to correct the PacBio long reads. The memory requirement for the ALLPATH-LP is 32 GB of memory for small genomes and 512 GB memory for the large genomes [21]. Another approach

is to assemble the short reads first and then to use PacBio long-reads mapping on the assembly graph to solve repetitive regions and extended contigs. Cerulean and rescaffolding apply to the second approach.

Pre-processing is the first step in the Cerulean Hybrid Assembly Approach. In this step, there are two processes. The first process is to assemble the Illumina Paired-end and Roche 454 SE datasets using ABySS, and save its assembled contigs file (Abyss_PE_454_k49.fa) and the graph structure file (Abyss_PE_454_k49.dot). Based on the Cerulean requirement, I rename these two files to AbyssPE454-contigs.fa and AbyssPE454-contigs.dot. The second process is mapping PacBio long reads to assembled contigs using BLASR. Cerulean gives a template BLASR command line, and only data name and number of threads can be modified. The name of output file is AbyssPE454_pacbio_contigs_mapping.fasta.m4. The three files generated from this step need to be saved.

The last step is to run the Cerulean to execute assembly. The resulting name is AbyssPE454_cerulean.fa. Figure 3 provides some command lines which were used in the Cerulean Hybrid Assembly approach.

```
sawriter AbyssPE454-contigs.fa

blasr AbyssPE454_pacbio.fq AbyssPE454-contigs.fa -minMatch 10 -minPctIdentity 70 -bestn 30
-nCandidates 30 -maxScore -500 -nproc 8 -noSplitSubreads -out
AbyssPE454_pacbio_contigs_mapping.fasta.m4

python Cerulean.py --dataname AbyssPE454 --basedir /home/larry/Desktop/CurleanPE/ --nproc 8
```

Figure 3. Command lines in Cerulean Hybrid Assembly approach

To check any possibly to extend the result from rescaffolding hybrid approach, I have also used the final results from the rescaffolding hybrid assembly approach (AbyssPE454_RescffPacBio.fa and AbyssPE454_RescffPacBio.dot) as the input for Cerulean to

perform the Cerulean Hybrid Assembly Approach. The result name is AbyssPE454RescffPacbio-cerulean.fa. The quality metrics of AbyssPE454_cerulean.fa and AbyssPE454RescffPacbio-cerulean.fa is shown in the Table 4.

Table 4. The result of Cerulean Hybrid Assembly Approach

Quality Metrics\File Name	AbyssPE454_cerulean.fa	ABYSSALLPE454_recaffPacBio_ cerulean.fa
N50	4,114,241bp	1,817,394bp
N75	4,107,391bp	731,549bp
Largest reads length	4,346,447bp	1,817,394bp
Number of Reads	10	15

Figure 3 shows whole processes in the Cerulean Hybrid Assembly Approach. The embolden names are the output files in the graph.

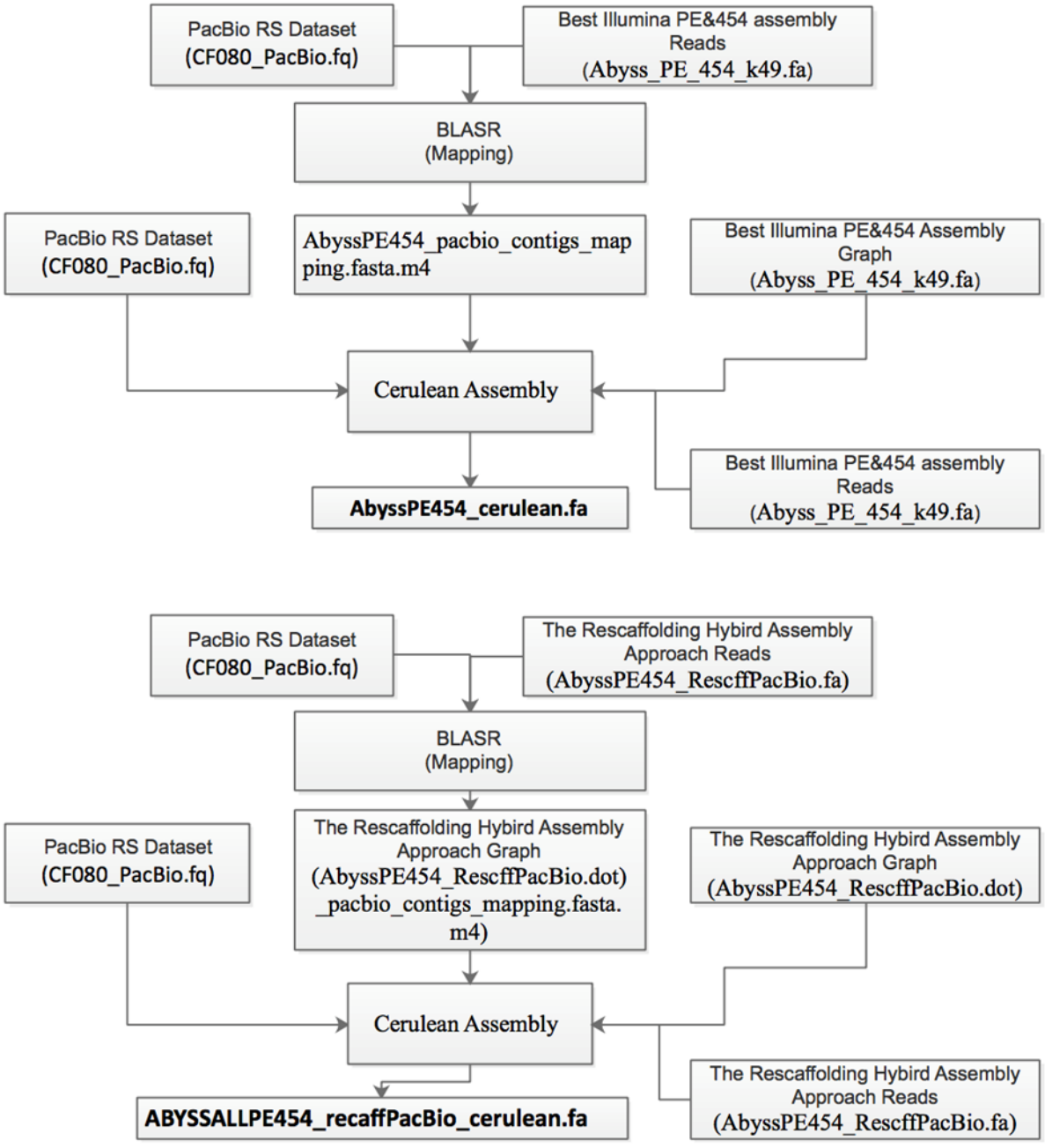

Figure 4. Cerulean Hybrid Assembly Approach overview

2.4. Assembly Quality

The N50, length of largest reads, and number of contigs are the most important quality metrics. Those three are being used to evaluate the quality of the two hybrid assembly

approaches. However, these three metrics only measure the quality in terms of contig length. The assembled cotigss may have some misassemblies caused by the assembly programs. The only method to assess misassembly rates is mapping the assembled contigss to the reference genome. The mapping result shows misassembly locations in the assembled contigs. Quality Assessment Tool (QUAST version 4.3) is a program to evaluate the genome assembly result by computing various metrics which include the number of misassemblies [23]. In the QUAST, misassemblies are classified as relocation, translocation, and inversion. The report only shows a total number of misassemblies, but the detailed information of misassemblies can be found in the alignment viewer which is a virtualized alignment graph provided by QUAST. Figure 6 shows QUAST statistics of four assembled files (AbyssPE454_cerulean.fa, AbyssPE454_RescffPacBio.fa, Abyss_PE_454_k49.fa, and AbyssPE454RescffPacbio-cerulean.fa,).

Genome statistics	AbyssPE454_cerulean	AbyssPE454_RescffPacBio	Abyss_PE_454_k49	AbyssPE454RescffPacbio-cerule...
Genome fraction (%)	99.811	99.834	99.814	99.768
Duplication ratio	3.371	1.004	1.006	2.082
Largest alignment	1 849 234	1 120 680	1 117 962	1 120 682
Total aligned length	23 677 741	7 063 128	7 075 992	14 632 317
NGA50	782 564	258 777	258 696	416 891
LGA50	4	7	7	6
Misassemblies				
# misassemblies	60	19	15	77
Misassembled contigs length	23 461 921	4 034 984	3 575 123	14 642 894
Mismatches				
# mismatches per 100 kbp	7.06	2.13	5.43	2.19
# indels per 100 kbp	10.97	0.36	9.99	0.38
# N's per 100 kbp	240.1	26.67	149.93	49.51
Statistics without reference				
# contigs	10	30	41	15
Largest contig	4 346 447	1 127 416	1 127 664	1 817 394
Total length	23 720 952	7 068 866	7 082 870	14 642 894
Total length (>= 1000 bp)	23 720 952	7 068 340	7 082 344	14 642 894
Total length (>= 10000 bp)	23 720 952	7 032 486	6 995 687	14 642 894
Total length (>= 50000 bp)	23 675 044	6 971 986	6 902 879	14 609 507
Predicted genes				
# predicted genes (unique)	7010	6786	7198	13 680

Figure 5. Screenshot for QUAST results statistics

The two assembled reads files from Cerulean Hybrid Assembly Approaches has fewer contigs numbers and larger contigs, but they also have a large number of misassemblies. The

AbyssPE454_RescffPacBio.fa from Rescaffolding Hybrid Approach has a much lower number of misassemblies and better genome fraction rates (total number of aligned bases in reference genome/reference genome size). Figure 7 shows a piece of alignment view of four assembled reads files aligned to reference genome (CF080_Reference.fa). The image shows two contigs (2034_161306 and 2034_161306_25300181_1939) from Abyss_PE_454_k49.fa that are assembled into one contig without any errors in AbyssPE454_RescffPacBio.fa.

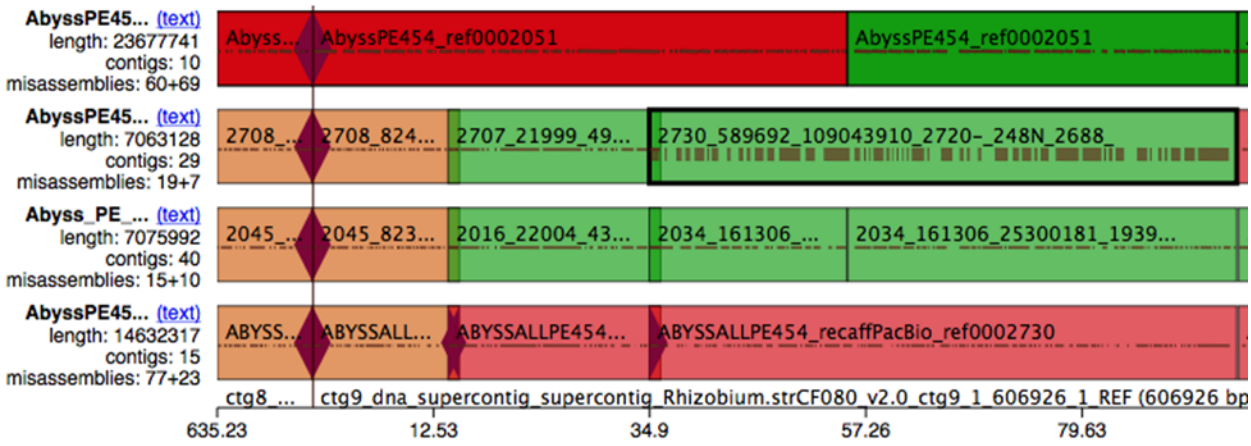

Figure 6. Screenshot for QUAST alignment view

(From top to bottom are AbyssPE454_cerulean.fa, AbyssPE454_RescffPacBio.fa, Abyss_PE_454_k49.fa, and AbyssPE454RescffPacbio-cerulean.fa)

CHAPTER 3. CONCLUSION

This paper presents an overview of two hybrid assembly approaches, Rescaffolding Hybrid Assembly approach, and Cerulean Hybrid Assembly approach, and evaluates the assembly quality for both approaches. In general, Cerulean Hybrid Assembly approach has better assembly quality in N50, the maximum contig length, and contig numbers than the Rescaffolding Hybrid Assembly Approach. But the Rescaffolding Hybrid Assembly has a much lower misassemblies rate. The Cerulean Hybrid approach extends assembled contigs from the Rescaffolding Hybrid Assembly approach which can cause a large number of misassemblies, and N50 and length of the largest read are not significantly changed. Two approaches are designed for assembling three different NGS datasets, Illumina PE, Roche 454 SE, and PacBio RS. These two approaches should help others looking to hybrid assembly those three NGS datasets.

REFERENCES

[1] Wikipedia. "Sequence assembly" https://en.wikipedia.org/wiki/Sequence_assembly#De-novo_vs._mapping_assembly. Accessed 10 Oct. 2016.

[2] Schuster, Stephan C. "Next-Generation Sequencing Transforms Today's Biology." *Nature* vol. 200 no. 8 (2007): 16-18.

[3] Chaisson, Mark J., and Pavel A. Pevzner. "Short Read Fragment Assembly of Bacterial Genomes." *Genome Research* vol. 18 no. 2 (2008): 324-330.

[4] Ikegami, Tsutomu, Toyohiro Inatsugi, Isao Kojima, Myco Umemura, Hiroko Hagiwara, Masayuki Machida, and Kiyoshi Asai. "Hybrid De Novo Genome Assembly Using MiSeq and SOLiD Short Read Data." PLOS ONE PLoS ONE 10.4 (2015): np. Illumina Support. Illumina, Inc., 22 Apr. 2016. Oct.-Nov. 2016. http://www.illumina.com/content/dam/illumina-marketing/documents/products/appnotes/nextera-mate-pair-miseq-application-note-770-2015-048.pdf Accessed 15 Nov. 2016.

[5] Utturkar, Sagar M., et al. "Evaluation and Validation of de Novo and Hybrid Assembly Techniques to Derive High-Quality Genome Sequences." *Bioinformatics* vol. 30 no. 19 (2014): 2709-2716.

[6] "PacificBiosciences/Bioinformatics-Training." GitHub. Ed. Rhall PB. PacificBiosciences, 20 Apr. 2014. https://github.com/PacificBiosciences/Bioinformatics-Training/wiki/Large-Genome-Assembly-with-PacBio-Long-Reads Accessed 15 Nov. 2016.

[7] Brown, Steven D., et al. “Draft Genome Sequence of Rhizobium Sp. Strain PDO1-076, a Bacterium Isolated from Populus Deltoides.” *Journal of Bacteriology* vol. 194 no. 9 (2012): 2383-2384.

[8] Wikipedia, “DNA sequencing” https://en.wikipedia.org/wiki/DNA_sequencing#cite_ note-pmid19900591-55. Accessed 10 Nov. 2016.

[9] de Magalhaes, Joao Pedro, Caleb E. Finch, and Georges Janssens. “Next-generation Sequencing in Aging Research: Emerging Applications, Problems, Pitfalls and Possible Solutions.” *Ageing research reviews* vol. 9 no. 3 (2010): 315-323.

[10] Metzker, Michael L. “Sequencing Technologies—The Next Generation.” *Nature Reviews Genetics* vol. 11 no. 1 (2010): 31-46.

[11] “History of Illumina Sequencing.” *History of Illumina Sequencing*. Illumina, Inc., 12 Feb. 2016. http://www.illumina.com/technology/next-generation-sequencing/solexa-technology.html Accessed 15 Nov. 2016.

[12] Boetzer, Marten, et al. “Scaffolding Pre-assembled Contigs Using SSPACE.” *Bioinformatics* vol. 27.no. 4 (2011): 578-579.

[13] Chin, Chen-Shan, et al. “Nonhybrid, Finished Microbial Genome Assemblies from Long-Read SMRT Sequencing Data.” *Nature Methods* vol. 10 no. 6 (2013): 563-569.

[14] Koren, Sergey, et al. “Hybrid Error Correction and de Novo Assembly of Single-Molecule Sequencing Reads.” *Nature Biotechnology* vol. 30 no. 7 (2012): 693-700.

[15] “Genome Assembly Primer | CBCB.” n.p., n.d. https://www.cbcb.umd.edu/research/ assembly_primer. Accessed 15 Nov. 2016.

[16] Levy, Samuel, et al. “The Diploid Genome Sequence of an Individual Human.” *PLoS Biol* vol. 5 no. 10 (2007): e254.

[17] Wikipedia, “N50, L50, and related statistics” https://en.wikipedia.org/wiki/N50,_L50,_and_related_statistics. Accessed 10 Nov. 2016.

[18] Medvedev, Paul, et al. “Paired de Bruijn Graphs: A Novel Approach for Incorporating Mate Pair Information into Genome Assemblers.” *Journal of Computational Biology* vol. 18 no. 11 (2011): 1625-1634.

[19] Zerbino, Daniel R., and Ewan Birney. “Velvet: Algorithms for de Novo Short Read Assembly Using de Bruijn Graphs.” *Genome Research* vol. 18 no. 5 (2008): 821-829.

[20] Deshpande, Viraj, et al. “Cerulean: A Hybrid Assembly Using High Throughput Short and Long Reads.” International Workshop on Algorithms in Bioinformatics. Springer Berlin Heidelberg, 2013.

[21] ftp://ftp.broadinstitute.org/pub/crd/ALLPATHS/Release-LG/AllPaths-LG_Manual.pdf. Accessed 10 10 2016.

[22] Chaisson, Mark J., and Glenn Tesler. “Mapping Single Molecule Sequencing Reads Using Basic Local Alignment with Successive Refinement (BLASR): Application and Theory.” *BMC Bioinformatics* vol. 13 no. 1 (2012): 238.

[23] http://quast.bioinf.spbau.ru/manual.html#sec3.1.1. Accessed 10 10 2016.

[24] Warren, René L., et al. “LINKS: Scalable, Alignment-free Scaffolding of Draft Genomes with Long Reads.” *GigaScience* vol. 4 no. 1 (2015): 1.

[25] http://www.bcgsc.ca/platform/bioinfo/software/abyss. Accessed 10 11 2016.

[26] Simpson, Jared T. et al. “ABySS: A Parallel Assembler for Short Read Sequence Data.” *Genome Research* vol. 19 no. 6 (2009): 1117-1123.

[27] Deshpande, Viraj, et al. “Cerulean: A Hybrid Assembly Using High Throughput Short and Long Reads.” International Workshop on Algorithms in Bioinformatics. Springer Berlin Heidelberg, 2013.

[28] Voelkerding, Karl V., Shale A. Dames, and Jacob D. Durtschi. “Next-generation Sequencing: From Basic Research to Diagnostics.” *Clinical Chemistry* vol. 55 no. 4 (2009): 641-658.

[29] Jiao, Xiaoli, et al. “A Benchmark Study on Error Assessment and Quality Control of CCS Reads Derived from the PacBio RS.” *Journal of Data Mining in Genomics & Proteomics* vol. 4 no. 3 (2013).

[30] Ribeiro, Filipe J., et al. “Finished Bacterial Genomes from Shotgun Sequence Data.” *Genome Research* vol.22 no. 11 (2012): 2270-2277.

[31] http://www.bcgsc.ca/platform/bioinfo/software/abyss/releases/2.0.2. Accessed 10 07 2016.

[33] Li, Ruiqiang, et al. “De Novo Assembly of Human Genomes with Massively Parallel Short Read Sequencing.” *Genome Research* vol. 20 no. .2 (2010): 265-272.

[34] Rahman, Atif, and Lior Pachter. “CGAL: Computing Genome Assembly Likelihoods.” *Genome Biology* vol. 14 no. 1 (2013): 1.

[35] Gurevich, Alexey, et al. “QUAST: Quality Assessment Tool for Genome Assemblies.” *Bioinformatics* vol. 29 no. 8 (2013): 1072-1075.

[36] Li, Heng. “Aligning Sequence Reads, Clone Sequences and Assembly Contigs with BWA-MEM.” arXiv preprint arXiv:1303.3997 (2013).

[37] http://soap.genomics.org.cn/soapdenovo.html. Accessed 10 11 2016.

[38] Chaisson, Mark J., and Glenn Tesler. “Mapping Single Molecule Sequencing Reads Using Basic Local Alignment with Successive Refinement (BLASR): Application and Theory.” *BMC Bioinformatics* vol. 13 no. 1 (2012): 238.

[39] Chaisson, Mark J., and Glenn Tesler. “Mapping Single Molecule Sequencing Reads Using Basic Local Alignment with Successive Refinement (BLASR): Application and Theory.” *BMC Bioinformatics* vol. 13 no. 1 (2012): 238.