PERFORMANCE ANALYSIS OF BACTERIAL GENOME ASSEMBLERS USING ILLUMINA NEXT GENERATION SEQUENCING DATA

NUR ' AIN BINTI MOHD ISHAK

FACULTY OF SCIENCE
UNIVERSITI MALAYA
KUALA LUMPUR

2020

PERFORMANCE ANALYSIS OF BACTERIAL GENOME ASSEMBLERS USING ILLUMINA NEXT GENERATION SEQUENCING DATA

NUR ' AIN BINTI MOHD ISHAK

DISSERTATION SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE

INSTITUTE OF BIOLOGICAL SCIENCES
FACULTY OF SCIENCE
UNIVERSITI MALAYA
KUALA LUMPUR

2020

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: **NUR 'AIN MOHD ISHAK**

Matric No: **SGR110114**

Name of Degree: **MASTER OF SCIENCE**

Title of Thesis:

PERFORMANCE ANALYSIS OF BACTERIAL GENOME ASSEMBLERS USING ILLUMINA NEXT GENERATION SEQUENCING DATA

Field of Study:
BIOINFORMATICS

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation:

PERFORMANCE ANALYSIS OF BACTERIAL GENOME ASSEMBLERS USING ILLUMINA NEXT GENERATION SEQUENCING DATA

ABSTRACT

The advancement of next generation sequencing (NGS) technology has revolutionized the field of genomic and genetic studies. As compared to conventional methods, NGS generate comprehensive genomic data at a fraction of the cost with a higher percentage of accuracy. One of the processing and analyzing NGS data is genome assembly. *De novo* assembly is a process of assembling short reads into contiguous sections of sequence without a reference which is different with conventional mapping technique. *De Bruijn* graph is one of the assembly algorithms that are widely used for short reads sequences produced from NGS platforms. In this study, the performance of four *de novo* assemblers (SPAdes, ABySS, Velvet and MaSuRCA) is reported, in which variants of *de Brujin* graph algorithms are applied, using genomic data generated by the Illumina sequencing platform. The computational performance regarding the assemblers running time were compared. The assembled contigs and scaffolds were also evaluated based on several qualities specifically for their length and the contiguity of the assembly using ABySS-fac. Results showed that on single-end data sets, MaSuRCA, and SPAdes produced generally the best results among all the four assemblers with highest percentage of contigs that were equal or longer than 500 bp, highest total base pairs, highest N50 and the lowest L50 for most assemblers. For paired-end data sets, Velvet are suitable to assemble all the seven bacteria genome sequences. This comparative study will advance the current knowledge of *de novo* genome assembly as it is the first step toward characterizing and revealing whole genomic information. In addition, this work provides a practical guideline that could aid researchers in identifying the appropriate assembler(s) for their research projects.

Keywords: Next generation sequencing (NGS), *de novo* assembly, *de Bruijn* graph, Illumina, whole genome sequencing

ANALISIS PRESTASI PERHIMPUNAN GENOM BAKTERIA MENGGUNAKAN DATA TEKNOLOGI PENJUJUKAN GENERASI AKAN DATANG JENIS ILLUMINA

ABSTRAK

Kemajuan teknologi penjujukan generasi akan datang (NGS) telah membawa satu revolusi dalam bidang kajian genom dan genetik. Berbanding dengan kaedah konvensional, NGS telah dapat menghasilkan data genomik yang komprehensif pada kos yang minimum tetapi peratusan ketepatan yang lebih tinggi. Salah satu proses dan analisis data NGS adalah perhimpunan genom. Perhimpunan secara *de novo* adalah satu proses menyatukan urutan pendek menjadi jujukan bersebelahan (lebih panjang) tanpa rujukan, yang berbeza dengan teknik pemetaan secara konvensional. Graf *de Bruijn* adalah salah satu daripada algoritma penghimpun yang digunakan secara meluas untuk urutan pendek yang dihasilkan dari platform NGS. Dalam kajian ini, prestasi empat penghimpun jujukan secara *de novo* (SPAdes, ABySS, Velvet and MaSuRCA) dilaporkan, yang mana pelbagai algoritma graf *de brujin* diguna pakai bagi data genomik yang dijana oleh platform penjujukan Illumina. Prestasi komputasi mengenai masa yang diperlukan bagi menjalankan kerja-kerja penyatuan dibandingkan. Kontig dan skafold yang terhasil juga dinilai berdasarkan beberapa kualiti khusus untuk panjangnya dan kesinambungan penyatuannya menggunakan ABySS-fac. Hasil kajian menunjukkan pada set data hujung tunggal, MaSuRCA dan SPAdes menghasilkan hasil yang terbaik di antara keempat-empat penghimpun dengan peratusan tertinggi yang sama atau lebih panjang daripada 500 bp, jumlah '*base pairs'* tertinggi, N50 tertinggi dan L50 terendah untuk kebanyakan penghimpun. Untuk set data berpasangan, Velvet sesuai untuk menyusun kesemua tujuh urutan genom bakteria. Kajian perbandingan ini akan dapat memajukan pengetahuan semasa berkenaan perhimpunan genom secara *de novo* seperti yang diketahui bahawa ia adalah langkah pertama ke arah mencirikan dan mendedahkan maklumat keseluruhan

genom. Di samping itu, ia juga dapat menyediakan satu panduan praktikal yang seharusnya membantu para penyelidik mengenal pasti penghimpun yang sesuai untuk projek penyelidikan mereka.

Kata kunci: Penjujukan generasi akan datang (NGS), perhimpunan secara *de novo*, graf *de Bruijn*, Illumina, penjujukan genom keseluruhan

ACKNOWLEDGEMENTS

First of all, I thank Allah the Almighty for all His providence in carrying out this work successfully.

I would like to express my sincere gratitude to my supervisors, Prof Dr Hj Amir Feisal Merican bin Hj Aljunid Merican and Dr Effirul Ikhwan Ramlan, I thank them for providing their invaluable guidance, support, patience and encouragement throughout the course of my master project.

I am grateful and touched for the attentions and support from my loving husband, Abdul Rahman Nordin, my understanding sons, Ariff Aiman Abdul Rahman and Ariff Anas Abdul Rahman. I am blessed to have all of you in my life. I am indebted to my father, Mohd Ishak Hj Masudi, my mother, Jawiah Hj Ishak and my siblings for their constant encouragement throughout the years of my life. Hereby, I place on record to dedicate this thesis solely to my beloved family.

Also, my warm thanks to my fellow friends in University of Malaya (UM) and Malaysian Palm Oil Board (MPOB) who have given me a lot of moral supports and advice during my research journey.

Last but not least, my sense of gratitude to one and all, who directly or indirectly have lent their hand in this venture.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Overview

Genome sequencing has been greatly enhanced by the overwhelming revolution in sequencing technologies (techniques, instruments and software) for the pass forty years. Began with sequencing the genomes of small, simple organisms until more complex with various sizes and shapes of genome involving different numbers of chromosome. Although genome sequencing started more earlier, year 1995 is the time when the first complete genetic catalogue of a free-living organism generated. It was a sequence of *Haemophilus influenzae*, a Gram-negative, pathogenic, facultatively anaerobic bacterium. This bacteria was choosen by the researchers from Johns Hopkins University School of Medicine, USA for their study because its genome size is a common size for a bacteria (1.8 Mb), its G+C base contents (38 percent) is close with human G+C contents and during that time, there is no existance of a *Haemophilus influenzae* physical clone map. Sanger technology had been used to sequence the bacteria (Fleischmann et al., 1995).

However, applying whole genome sequencing (WGS) by Sanger instrument is not an efficient method. This is due to Sanger sequencing needs high costs and it involves time-consuming process. Therefore, some researchers considered to transform to next generation sequencing (NGS) because of the costly-effective and faster when compared with Sanger sequencing (Ahmadloo et al., 2017). Furthermore, NGS capables toproduce very huge amount of data which more than one billion of short reads in a single run (Raza & Ahmad, 2016). The changes that occur in sequencing technology influenced the post-genomic analysis involving small and large genomes (Ni et al., 2018; Ekblom & Wolf, 2014; Li et al., 2009). Although, the data generated from NGS platform is quite short, it

has been used frequently to detect SNP in human and other mammalian genome by the increasing of sequencing depth and coverage. The NGS data also give a lot of information regarding gene fusion, expression and variation especially in disease (Benke et al., 2018; Gioiosa et al., 2018; De Wit et al., 2012; Ozsolak & Milos, 2011).

The pipeline for processing and analyzing data from NGS or is sometimes called 'massively parallel sequencing' platforms is shown in Figure 1.1. It is divided into three stages starting with **primary analysis** in which the sequencing instrument raw signals generates nucleotide base and short-read data. The next stage is **secondary analysis** by aligning the sequences to a reference or *de novo* assembly will be applied to the reads which do not have references. Variant detection is also performed in this stage. Finally, the **tertiary analysis** stage or "interpretation" stage is to determine their biological significance, function and meaning from the genetic data (Oliver, Hart, & Klee, 2015; Moorthie, Hall, & Wright, 2012).

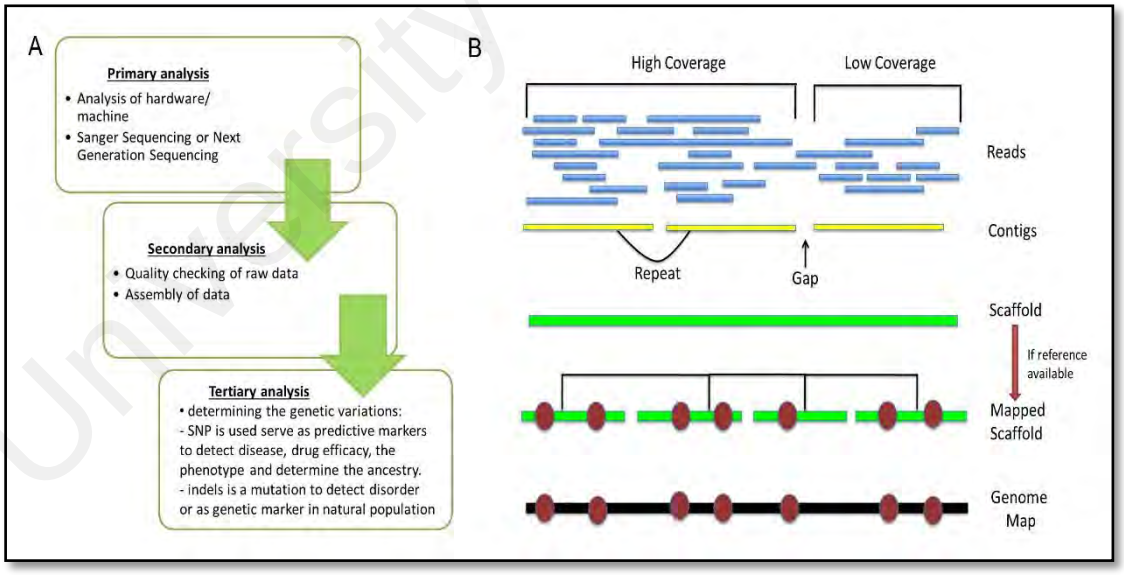


Figure 1.1: (A) The whole genome research in general (B) A trail of assembly process after sequencing.

It is expectable that when the genome is fully sequenced and assembled, it can be produced in a form of full-length chromosomes. However, it is not a straightforward task due data complexity. There are several challenges of NGS output need to handle wisely such as the short reads produced by NGS platform, the gap between existing computational tools to align or assemble these short reads (El-Metwally et al., 2013), repeats can be tricky and make the assembly process more complex (Treangen & Salzberg, 2011), sequencing error and others. All these issues make the genome assembly process harder. The process involved is getting complicated when there is lack or absence reference genome. Thus, *de novo* assembly, a process of assembling short reads unsuitable for conventional mapping technique should be applied. Furthermore, according to Maretty et al. (2017) the *de novo* assembly has a capability to identify a rich information of genomic diversity by looking into the specific organism's genetic and structural variations completely. In addition, with the progressive of a *de novo* assembly method in lower cost would allow the constructing the reference sequences which are really exigency and very important for varies post-genomic analysis such as identify substitutions, insertions, deletions (indel), characterize individual genomes and detect structural and genetic variation especially novel sequences (Sohn & Nam, 2016).

Contiguity, as well as the accuracy of genome assembly can be evaluated with different assembly metrics such as number of contigs (n), number of contigs at least 500 bp (n:500), the number of contigs equal to or longer than N50 reported in the N50 column (L50), smallest contig (min), largest contig (max), N50 contig length (N50), N80 contig length (N80), N20 contig length (N20), the sum of the square of the sequence sizes divided by the assembly size (E-size) and sum of contig lengths (sum).

1.2 Problem Statements

The focus of this study is to recognize and distinguish clearly different type of *de novo* genome assemblies' graphs which are - Greedy extension, overlap-layout-consensus and *de Bruijn* graph. In this study, efforts are also being made to assess the execution of four *de novo* assemblers (SPAdes, ABySS, Velvet and MaSuRCA) which employed *de Bruijn* graph algorithms for bacterial genomes.

1.3 Research Questions

Although the sizes of bacteria genome are small and bacterial sequencing procedure have been started in 1995, in reality many of the sequenced bacterial are still in draft stage. Based on Land et al. (2015), 90% of bacterial genomes in GenBank are incomplete. This situation happened because of the occurrence of repetitive sequences in bacterial genomes, misassembled regions in draft sequence, incorrect gene calls and so forth (Utturkar et al., 2017). Cheung & Kwan (2012) has explained the need to have a genomic analytical workflow to extract the complex bacterial genomes information especially when involved with disease outbreaks cause by bacterial pathogen. In the past few years, several *de novo* assemblers with different types of algorithms have been developed. However, to choose the appropriate assembler for paired-end or single-end data is still a challenging task (Baker, 2012).

1.4 Objectives

1. To evaluate four *de novo de Bruijn* graph assemblers (SPAdes, ABySS, Velvet and MaSuRCA) using bacterial genome sequencing data sets generated by the Illumina platform.
2. To validate the performance of the *de novo* assemblers, on the respective genome sequences using Genome Assembly Gold-Standard Evaluations (GAGE) and gVolante.

1.5 Organization

This thesis comprises of six chapters, which are: Chapter 1-Introduction, Chapter 2-Literature review, Chapter 3-Materials and methods, Chapter 4-Results, Chapter 5-Discussion and Chapter 6-Conclusion. The first chapter describes the overview of genome sequencing using next generation sequencing (NGS) and the objectives of this study. Second chapter contains literature review of entities related to the study. Chapter 3, the materials and methodology chapter describe the software, hardware, parameters and research pipeline adopted in this study. Chapter 4 presents the results of this study and the findings are further discussed in Chapter 5, discussion. The last chapter summarizes the outcome of this study.

Hopefully, this study will advance the current knowledge of *de novo* genome assemblies from different strategies and platforms, as we know that genome assemblies is the first step toward characterizing and revealing whole genomes information. This study also will contribute further to the development of new tools which relevance with the current sequencing platforms.

CHAPTER 2: LITERATURE REVIEW

2.1 Bacterial genome

According to Goldman & Landweber (2016), "genome" of an organism is defined as the entire genetic complement of a living organism. The phrase "entire genetic complement" refers to DNA genomes or Ribonucleic acid (RNA) genomes which comprised genes, gene-related sequences (pseudogenes, introns, gene fragments) and intergenic DNA (repeats, microsatellites). These genomic elements are packaged in chromosomes. In eukaryotic organism, the individual genome consists of several chromosomes with different sizes and shapes while in prokaryotic organism, most of the genome usually exists as a single, circular chromosome (some have linear chromosome and some have more than one circular chromosome) according to the table 2.1.

Table 2.1: The different types of chromosomes for selected prokaryotic organisms.

Species name	Chromosome
Agrobacterium tunefaciens	One linear + one circular
Escherichia coli K-12	One circular
Vibrio cholerae	Two circular
Paracoccus denitrificans	Three circular
Borrelia burgdorferi	One linear

Seventy years ago, it was generally believed that all chromosomes were linear. However, in 1963, Cairns found large circles with a 1300 μm circumferences in *Escherichia coli* cell that he isolated and labelled the DNA using radioactive isotope. It was clear that the bacterium consists of a single circular of molecule DNA. The idea that bacteria have a single circular chromosome by citing *E. coli* as the example, was quickly adopted until the development of new techniques evolved that allowed the separation and analysis of large DNA fragments in early 1980. One of the techniques is pulsed-field gel electrophoresis (PFGE). This technique permitted the study of physical structure of

bacterial genome directly. Thus, several bacterial chromosomes' studies have been conducted and revealed complex structures in some bacteria including *Rhodobacter sphaeroides* had two circular chromosomes and *Borrelia burgdorferi* had a linear chromosome and linear plasmids.

A bacterial genome is unique. Other than its chromosome in its cell, with the function as a governor that keep necessary information for replication and continued life of the cell under normal growth conditions, it is also contain phage genomes and plasmids. These elements sometimes have the ability to integrate into the chromosome and remain there for generation.

2.2 Genome sequencing technique – historical perspective

Genome sequencing has gone through the long history starting mid 1970's, when Walter Fiers and his team sequenced the first genome, the bacteriophage MS2 at the RNA level (Fiers et al., 1976). It was soon followed in 1977, the bacteriophage ΦX174 genome had been sequenced by Frederick Sanger and his team using Sanger sequencing at DNA level (Sanger et al., 1977). This journey of flourishing continued in 1995 when the first free-living organism, *Haemophilus influenzae* was completely sequenced by researchers from Johns Hopkins University School of Medicine, USA. The same team was sequenced *Methanococcus jannaschii*, thermophilic methanogenic (methane producers) archaean. Even during that time, modern computer facilities are not fully ready for this kind of research (Fleischmann et al., 1995).

The rapid advancement of sequencing research discipline is never been stopped. There is a large volume of published studies describing the developing and improving the sequencing technologies including experiment procedures, sequencing instruments and software in determining the precise order of DNA molecules. This is supported by Jay

Shendure et al. (2017) review which described in details of the 40th anniversary of DNA sequencing. It is started with the history of DNA sequencing from early generation of sequencing (the chain termination sequencing method developed by Sanger and Coulson, and the chemical sequencing procedure developed by Maxam and Gilbert) until the improvement of the sequencing methods (including the software) were highlighted in details due to more complex and larger organism involved. The author also explained the application of DNA sequencing and finally the future and hope from these technologies. The authors believe that DNA sequencing still a young technology based on the continuity evolving and arising of the field. It can be comparable with the microscope which is still be applied and upgrade although it has been invented more than 400 years ago.

A number of whole genome sequencing technologies have been developed through three major revolutions: first generation sequencing (Sanger sequencing), second generation sequencing (next generation sequencing) and the third generation of sequencing (single molecule long read sequencing). Sanger sequencing technology is also well known as chain termination sequencing is based on the addition of dideoxynucleotides (ddNTP's) in the normal nucleotides (NTP's) found in DNA. The only difference of ddNTP's and NTP's is the replacement of a hydroxyl group (OH) with a hydrogen group on the 3' carbon (Figure 2.1).

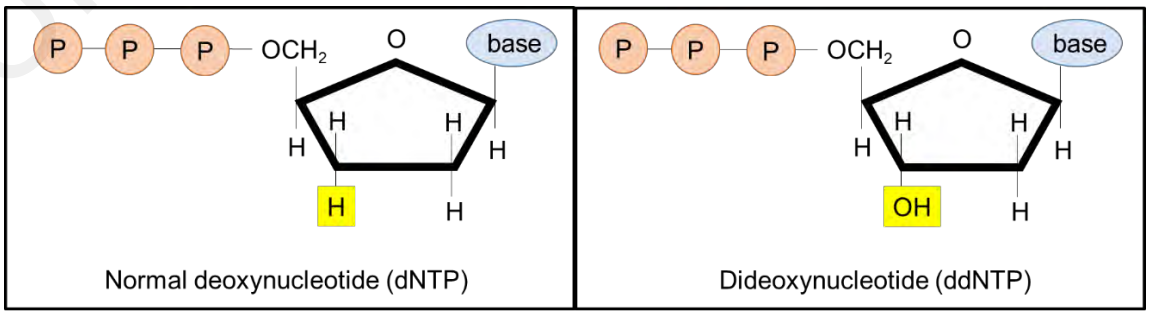


Figure 2.1: Structural comparison of dNTP and ddNTP

This method is faster, reliable and more efficient techniques (less utilization of toxic chemicals and radioisotopes) to sequence DNA compared to Maxam-Gilbert Sequencing.

2.3 Next Generation Sequencing (NGS)

The demand for cost-effective and faster sequencing techniques has increased dramatically especially after the completion of the first human genome. Some of the research community start to shift to NGS technology and it became more widely available. Instead of the factor of time and cost, the innovation of NGS is also give advantageous compared to Sanger sequencing. First, the preparation of NGS libraries in a cell free system. Second, millions of DNA fragments produced in a single reaction (i.e., in parallel) and really suitable for processing complex samples, especially for large-scale studies (van Dijk et al., 2014). NGS sequencing has proven revolutionary, shifting the paradigm of genomics to address biological questions at a genome-wide scale.

The first NGS was introduced to the market by 454 Life Sciences based in Branford, Connecticut in 2005. The sequencer uses pyrosequencing technology that relies on the light detection of pyrophosphate released during the DNA polymerization reaction is occured and used as a marker of DNA incorporation (Fakruddin et al., 2012; Ronaghi, 2001). Later in 2007, 454 Life Sciences acquired by other company, Roche and it was also happened to other NGS founders, Solexa (which invented Genome Analyzer) was purchased by Illumina while Agencourt (which invented SOLiD [Sequencing by Oligo Ligation Detection]) was purchased by Applied Biosystems (Metzker, 2010; Ansorge, 2009; J. Shendure & Ji, 2008; Bentley, 2006). These three NGS platforms have been classified as second-generation sequencing and they shared higher throughput, efficiency and accuracy, instead of it is economically compared with Sanger sequencing (Liu et al., 2012).

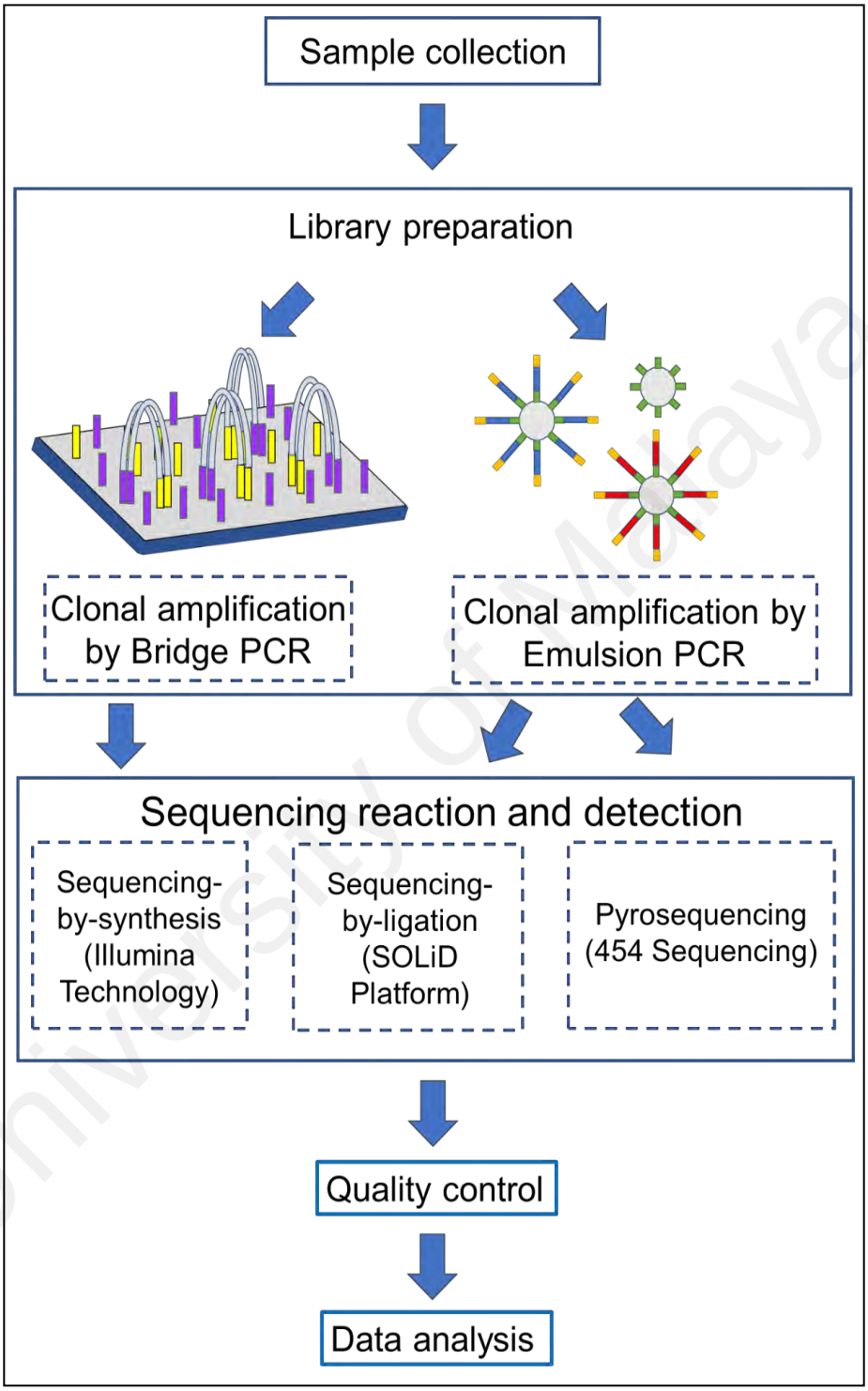


Figure 2.2: General workflow of second-generation sequencing

The general workflow of second-generation sequencing (in Figure 2.2) includes five phases: sample collection, library and template preparation, sequencing reactions and detection, quality control and data analysis. Establishing a high-quality DNA in sufficient quantity is necessary for the first phase and it may originate from different sources such as genomic DNA, reverse-transcribed RNA, cDNA, immunoprecipitated DNA and others. Second, library preparation which involved with converting the sample DNA into a library of sequencing reaction templates by common process including fragmentation, size selection, and adapter ligation. The process of fragmentation involves by randomly breaking the DNA templates into small pieces in which the size is depending on the sequencing platforms. The ligation of platform-specific adapters (which serve as primers) onto the ends of the DNA fragments for amplification and/or sequencing reactions. There are two types of amplification processes that commonly applied in second generation sequencing which are – bridge PCR or emulsion PCR. Third phase involves with sequencing reactions and detection that are vary depending on the sequencing platforms. The Illumina platform is based on sequencing-by-synthesis (SBS), SOLiD platform is based on sequencing-by-ligation (SBL) and Roche/454 platform is based on pyrosequencing.

The last two steps after sequencing is complete which are checking the quality control and analysing of generated raw sequences data. Generally, each platform produces two types of data – the short-read sequences (commonly in FASTQ format) and the generated read quality scores. It is an important step to check and remove poor-quality sequence data including technical sequences (example adapter sequences) before any further analysis conducted. There are several forms of poor-quality sequence generated which are base-call errors (incorrectly identified DNA bases), systematic error of read, sample

contaminants, run-to-run variations, coverage biases and others. Figure 2.3 showed the different types of next generation sequencing platforms.

Figure 2.3: The type of next generation sequencing platforms

NGS technologies revolution have been going through the significant transition from second-generation to third-generation sequencing. This transformation comes out with distinct defining characteristics of the machines which are real-time sequencing with simple divergence (Ambardar et al., 2016). The third-generation sequencing implies the single-molecule sequencing that is PCR-free protocol (directly sequence each of single bases of DNA or RNA molecules without amplification) and cycle-free chemistry that described in Figure 2.4 and Table 2.2. The advantages of this technology are minimizing sample handling and input requirements, increases read length and more sensitive in term of accurate quantitation of nucleic acid molecule. The example of third generation sequencing is single molecule, real time (SMRT) sequencer from Pacific Biosciences and SMRT incorporating nanopore technology from Oxford nanopore technologies.

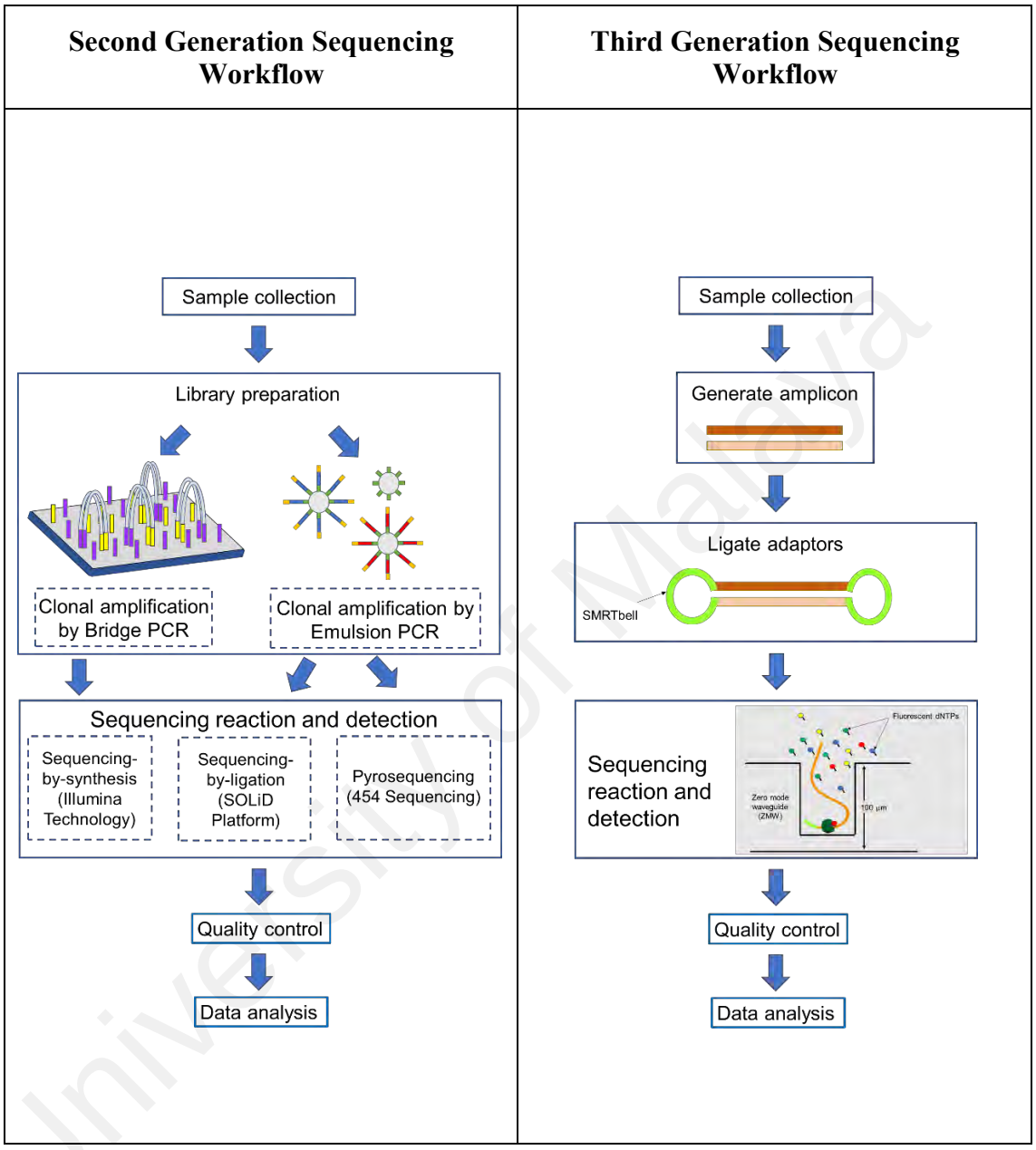


Figure 2.4: The common flow of second-generation and third-generation sequencing

Table 2.2: Comparison of first-, second-, and third-generation sequencing technology

Characteristics	First generation sequencing	Second generation sequencing	Third generation sequencing
Type of platforms (model)	• ABI Sanger (3730xl)	• 454 (GS20, GS FLX, GS FLX Titanium, GS Junior, GS Juniror+) • Illumina (Genome Analyzer II MiniSeq, MiSeq, NextSeq, HiSeq, Hiseq X) • SOLiD (5500 W, 5500xl W)	• Pacific Biosciences – PacBio (PacBio RS) • Ion Torrent System (Ion Torrent Personal Genome Machine (PGM) and Ion Torrent Proton) • Oxford Nanopore (PromethIO, MinION)
Amplification method	• PCR	• Emulsion PCR except Illumina (Bridge PCR)	• Real-time single-molecule template (PacBio) • None (Oxford Nanopore)
Method of sequencing	• Capillary electrophoresis (CE) Sanger sequencing	• Pyrosequencing (454) • Reversible terminator sequencing by synthesis (Illumina) • Sequencing by ligation (SOLiD)	• Real-time single-molecule sequencing (PacBio) • Single molecule sequencing incorporating nanopore technology (Oxford Nanopore)
Method of Detection	• Fluorescence	• Optical (454) • Fluorescence/ Optical (Illumina) • Fluorescence/ Optical (SOLiD)	• Fluorescence/ Optical (PacBio) • Electrical Conductivity (Oxford Nanopore)
Reads per run	< 100	100 – 300,000,000	432 – 50 000
Read length (per base)	400 bp – 1000 bp	35 bp – 800 bp	Up to 60 kbp
Error rate	0.001%	• 1% (454) • 0.4% (Illumina) • 0.1 % (SOLiD)	• 15% (Pac Bio) • 1% (Ion Torrent) • 4% (Oxford Nanopore)
Average time to run	Hours	Days	< 1 day

2.4 Genome assembly

After the sequencing process is done, the reads will be assembled. The read is the output and the most basic element of sequencing. The length of reads is varying, and it depends of the sequencing platforms. For instance, Sanger sequencing produces between 700 to 1000 bp while NGS platforms - pyrosequencing which is only ~800 bp long and Solexa/Illumina from ~100 bp reads (Goodwin et al., 2016; Loman et al., 2012). The sequence assembly is forming a set of contiguous sequences (contigs) from the reads randomly by applying multiple sequence alignment with selected algorithms (Phillippy, 2017). Then, the contigs will form the order and orientation of the DNA strand of scaffolds, either the forward or reverse strand. Scaffolds are also defined as supercontigs or metacontigs (Miller et al., 2010).

As we know, most of reads obtained by NGS platforms is very short length, so assembly process is needed to construct long and contiguous sequences and finally a complete genome. Generally, the raw reads generated by NGS platform is in FASTQ format (compression version "fastq.gz"). It comprises the sequence bases with an associated per base quality score (normally using by Phred). Phred indicates the probability of correct calling of the given base by the equation

$$QPHRED = -10\,x\log 10(Pe) \tag{2.1}$$

Example, Phread quality score of 30 nominally corresponds to a 0.1% error rate equals to a 99.9% base call accuracy (Kanterakis et al., 2018). FASTA format is one of several data file format that is widely accepted for an assembly. In FASTA file, it contains the characters A, C, G, T and other characters with the special meaning based on the assembler.

According to Paszkiewicz & Studholme (2010), the contiguity and accuracy of the

contigs or scaffolds are the important criteria to determine the quality of genome assemblies. The contiguity of the contigs or scaffolds can be defined as the length distributions of these sequences and usually be calculated by various statistical metrics such as number of contigs, number of contigs at least 500 bp, N50 contig length, the number of contigs equal to or longer than N50 contig length reported, smallest contig (min), median contig length, average contig length, maximum contig length and sum of contig lengths. However, N50 length is a metric widely used to assess the contiguity of an assembly which calculated by sorting contigs according to their lengths in descending order then summing their lengths, the length of the shortest contig that represents equal or more than 50% of the sequences. On the other hand, the accuracy which also refers as 'correctness' of an assembly show how well an assembly represent the genome sequenced by aligning with a complete reference genome using different genomic alignment tools to detect misassemblies, including mismatches, indels, and misjoins (Alhakami, H., Mirebrahim, H. & Lonardi, S., 2017). If there is unavailable references genome, the conserved sequences of related organisms may be used to detect conserved sequences in the newly assembled genome (El-Metwally et al., 2013).

There are two methods for assembly which are comparative assembler or *de novo* assembler. Although these two methods are different, yet not exclusive schemes. This is due, during comparative assembly process involved, the *de novo* assembly technique can be applied when there are the areas of the novel genome that differ significantly with the reference genome. The use of assembly depends on biological complexity of the data, computational memory constraints, availability of reference genomes and application. The details about these genome assembly are:

Comparative assembly uses a 'reference' in order to guide the assembly of the target organism. The reference/template can be a closely related organism with the target

organism a different strain of the same genus (Pop et al., 2004) or a different assembly of the same genome. This strategy is used in resequencing applications, for example (Pop et al., 2004) and have many applications such as single nucleotide polymorphisms (SNP) discovery, expression profiling, small RNA discovery and so forth (Nagarajan, N. & Pop, M., 2010). There are two main reference-guided assembly strategies: In the first one, reads are mapped against the reference genome and then used to construct an alternative consensus sequence (Vezzi et al., 2011). In the second approach, the reads are first *de novo* assembled. Then, the resulting contigs/scaffolds are aligned against the reference genome to order and orientate them along chromosomes, to get gene information for genome annotation and to identify potential misassembled contigs or scaffolds (Bao et al., 2014).

***De novo* genome assembly** can be explained as a process of solving a big jigsaw puzzle without knowing the resulting picture. This is due to the absence of a reference sequences or even complete closure of the genome. Although this method may produce errors because of the algorithm will give the best guess during assembly (Horner et al., 2010), reference resource constraints is more crucial. Even, this type of assembly also used for sequence's region that obviously have large different from the reference (Pop, 2009). In addition, this technique is considered much more challenging than comparative assembly. The applications of this technique are exploring unique microbial populations or unique environments, non-model organisms and others.

2.4.1 Challenges in *de novo* genome assembly

Although the implementation of NGS delights some, there are some flaws and challenges that must be addressed. One of them is the existence of repeated sequences (Alkan et al., 2011; Treangen & Salzberg, 2011). As we know, repeated sequences are a common feature from bacteria to eukaryotic DNA. It looks similar or identical with the sequence in the genome and it difficult to detect because it is various in size, multiple sequence and everywhere in the genome. In general, highly repetitive DNA is usually occuring as tandem repeats and organized around centromeres and telomeres, while moderately repetitive DNA is spreaded throughout the euchromatin, chromosome and genome (Biscotti et al., 2015; Primrose & Twyman, 2009). The repeats give a technical challenge during assembly especially the perfect repeats or the repeats are longer than reads (Miller et al., 2010). Thus, the assembly result is reduced or even worse, lost genomic complexity.

The next generation sequencing technique are advantageous in terms of lowering the cost and reducing time needed to produce high-throughput data. However, a problem of these sequencing technologies is the read length produced, which is much shorter than the traditional Sanger sequencing reads. Furthermore, the volume of reads obtained from NGS is three to four greater orders of magnitude when compared to the traditional sequencing method. Some examples are the reads from pyrosequencing (454 sequencing) which is only ~700 bp long and Solexa from 36 to 250 bp reads. However, these lengths of reads cannot compete with those of the traditional Sanger sequencing technologies (500–1,000 bp). This is because, when NGS generates the reads too short, the procedure of repeat masking is disrupted. Therefore, the difficulty to assemble the reads with many repeats will be increased (D. R. Zerbino & E. Birney, 2008).

Another assembly challenge is sequencing error. It happens when one or more bases

are mistakenly called during the sequencing process. Actually, the chance of a sequencing error is generally known, so it is important to ensure that extensive testing and calibration of the sequencing machine is done. For example, the sequencing errors of Illumina sequencing machines is yielded at a rate of $\sim 0.1–1 \times 10^{-2}$ per base sequenced and it is based on the which data-filtering used (Jünemann et al., 2013; Loman et al., 2012). This platform may interpret millions of errors since Illumina sequencing can produces billions of base calls per experiments. There are several types of sequencing errors such as mismatches, indels, ambiguous characters and homopolymer-length errors. Although all of these errors become clear during the alignment of the reads especially with the reference's genome, it invites some confusion if the *de novo* assembly is conducted.

Non-uniform coverage of the target - Coverage variation occurs by chance, when variation in cellular copy number between source DNA molecules, and by compositional bias of sequencing technologies. Very low coverage will bring gaps in assemblies.

Characteristics of *de novo* assembly
• The process of the read's assembly into contigs and scaffolds without the use of previous references • Enable gene discovery (Hirakawa et al., 2019) • Identification of structural and sequence variants (including single nucleotide polymorphisms (SNPs) and small insertions/deletions and alternative splice forms) (M. Li et al., 2017; Chaisson et al., 2015; Pegadaraju et al., 2013) • Estimation of expression abundances • Creation a precise map of highly rearranged genomes and for understanding the associated phenotypes
Challenges and Limitations
• Variety of assembly approach whether greedy algorithm, OLC or *de Bruijn* graph • Complexity and non-randomness of genome sequences such as repeats that cause mis-arrangements or gaps in the assembly, a nonuniform read depth, thus resulting in copy loss or gain in the assembly. • To assemble vast difference in scale of short reads (compared to Sanger read length) generated depending on NGS platform especially big size genome • The rate and types of sequencing errors vary depending to the NGS instruments and library preparation method • Uneven read depth, which results from polymerase chain reaction (PCR), cloning, extreme GC bias, sequencing errors and copy number variations

2.4.2 Algorithms for Genome Assembly

As algorithm is implemented to assemble the reads without the reference, there are various types of assembler algorithms are: greedy approaches, overlap-layout- consensus (OLC) and *de Bruijn* graph (Simpson & Pop, 2015; Boisvert et al., 2010). Figure 2.5 showed different types of algorithm for *de novo* assembly- greedy extension, overlap-layout-consensus and *de Bruijn* graph.

Greedy extension – is the implementation of string-based method. The basic operation of the greedy extension algorithm starts with the joining of individual read or contigs to another read using the highest-scoring overlap. The process is repeated until no more

reads can be connected. This is also applicable when joining contigs to make long scaffolds. Overlap in assembly refers to the prefix of one of the reads sharing sufficient similarity with the suffix of another read (Pop, 2009). The quality score of the overlaps depends on the length of overlaps and the level of identity (matching bases) between overlapping regions in two reads. Although this algorithm is the simplest, most intuitive; solution to the assembly problem (Pop, 2009), this algorithm may lead to misassembled repeats because it drastically simplifies the graph by considering only the high-scoring edges, which only optimizes a local solution. This type of algorithm is used for Sanger data such as, PHRAP (de la Bastide & McCombie, 2007), TIGR Assembler (Granger G. Sutton, 1995) and CAP3 ("HGS- TIGR splits, opportunity knocks," 1997). It is also used for NGS data such as SSAKE (Warren et al., 2007), VCAKE (Jeck et al., 2007) and SHARCGS (Dohm et al., 2007), with minor differences to the greedy approach.

The second approach is **overlap-layout-consensus** or commonly known as OLC. It commonly applied in Sanger sequencing data (Z. Li et al., 2012). The similarity between greedy and OLC techniques is a module called an overlapper. As mentioned before, the overlap refers to the region where the prefix of one of the reads shares sufficient similarity with the suffix of another read (Pop, 2009). The method involves by finding all the overlapping reads in both the forward and reverse complement orientation. Then, the optimal reads are first merged into contigs and next to scaffolds. In the layout phase, the contigs are constructing and manipulating from the overlapping reads to determine the optimal location. Lastly, the consensus sequence of contigs are then created using progressive pair-wise alignments. Although some suggest to use Multiple Sequence Alignment (MSA) to have an accurate layout and consensus sequence, however there is no effective solution to find an optimal MSA (Miller et al., 2010). A few programs that use this algorithm are Newbler (Moore et al., 2006), Arachne (Batzoglou et al., 2002),

Celera Assembler (CABOG) (Miller et al., 2008) and Edena (Hernandez et al., 2008).

The weakness of this approach is it cannot identify clearly the presence of errors and polymorphisms especially indels and structural polymorphisms. Furthermore, it is space and time-consuming process mostly when large data sets involved (Palmer et al., 2010).

***de Bruijn* graph** - In this graph, a node is defined by a sequence of a fixed length of *k* nucleotides ('*k-mer'*, with k considerably shorter than the read length), then form the nodes of the graph (network), if they perfectly overlap by *k – 1* nucleotides, and the sequence data support this connection. This kind of method shows short sequences (k-mers) occurring in reads are only stored once. This algorithm was originally introduced in 1995 by Ramana M. Idury and Michael S. Waterman (Idury & Waterman, 1995) and the first *de Bruijn* assembler was developed by Pavel Pevzner and Michael Waterman in 2001 called EULER (Pevzner, Tang & Waterman, 2001). The beneficial of *de Bruijn* graph is it solves the assembly problem by the properties of the graph itself that having a graph structure representative of the repeat structure of the genome, thus it is not required the storage of pairwise overlaps and provide a solution to the assembly problem concerning excessive computational memory usage caused by the genome length. Examples of *de bruijn* graph assemblers' tools are SPAdes (Bankevich et al., 2012), Velvet (Zerbino & Birney, 2008), ABySS (Simpson et al., 2009), SOAPdenovo and so forth. However, each of these tools have their own uniqueness of graph construction, e.g., bulge/bubble removal in EULER/Velvet while in SPAdes it is applied *multisized de Bruijn* graph.

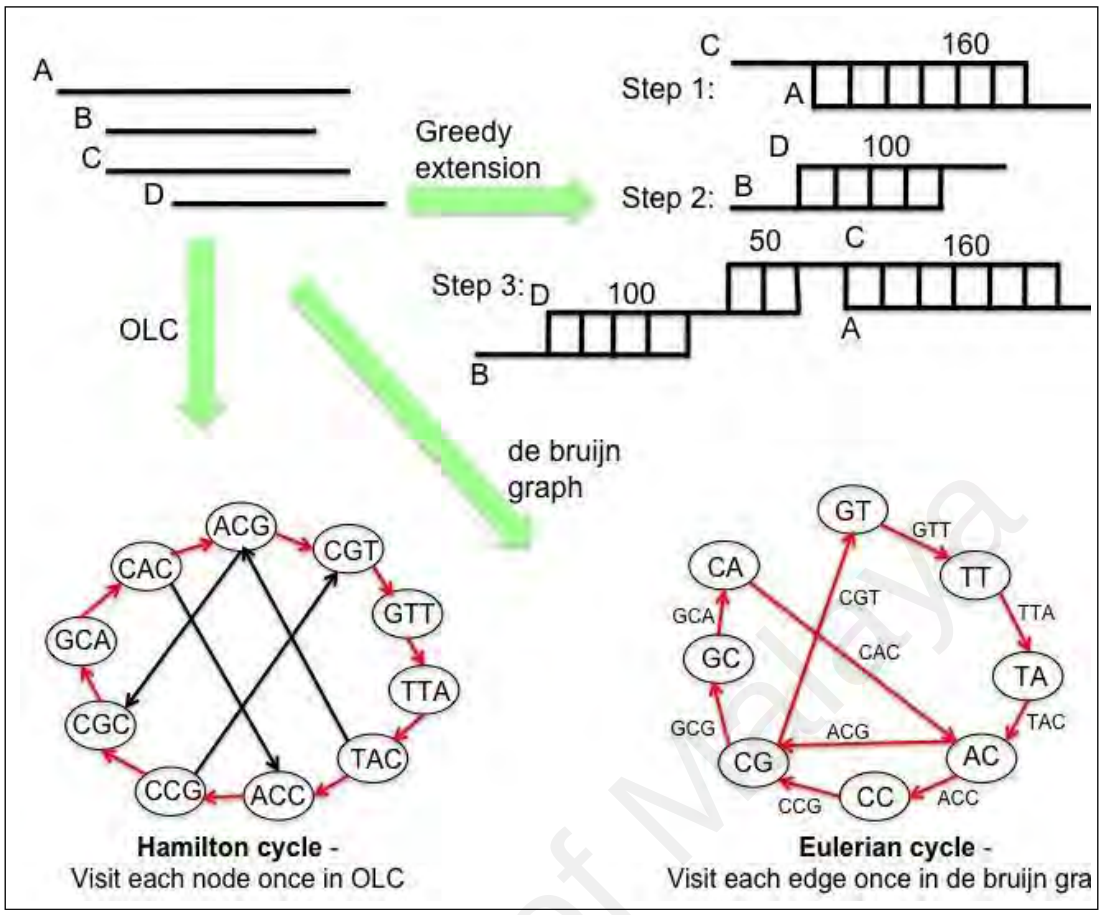


Figure 2.5 : Algorithm for *de novo* assembly- greedy extension, overlap-layout-consensus and *de Bruijn* graph.

CHAPTER 3: MATERIALS AND METHODOLOGY

3.1 Materials

3.1.1 Whole bacterial genomic dataset

Whole genome sequencing data for seven bacterial species in single-end and/or paired-end reads - *Clostridium botulinum, Escherichia coli, Bacillus cereus, Campylobacter jejuni, Salmonella enterica, Streptococcus pneumoniae* and *Listeria monocytogenes* employed in this study. These real data sets were downloaded from European Bionformatics Institute EMBL-EBI (http://www.ebi.ac.uk). The information of these bacterial species including SRA sequence accession number, read length (bp), types of Illumina sequencing platform, read count and base count (bp) summarized in Table 3.1.

The bacteria were chosen for the availability of Illumina sequence data and only applied this platform for this research to standardize the parameter and protocol for each of the data. Illumina is the widely used NGS platform utilized by researchers based on the cost effectiveness of the technology in faster time, high-throughput and short reads with high accuracy when compared with 454 and SOLiD (Verma et al., 2017; Liu et al., 2012). Although it generates the output that is comparable to Illumina, SOLiD platform uses a relatively complicated analysis (Jackman & Birol, 2010).

Table 3.1: Table shows accession numbers and sizes (bp) of every species in European Bionformatics Institute EMBL-EBI

Name	Library	Accession number	Read length (bp)	Sequencing platform	Read count	Base count (bp)
Clostridium botulinum	PAIRED	SRR2075978	2 x 150	Illumina MiSeq	2,377,364	717,963,928
	SINGLE	SRR1190420	200	Illumina MiSeq	1,845,075	516,169,318
Escherichia coli	PAIRED	DRR075676	2 x 150	Illumina HiSeq 2500	4,280,866	1,284,259,800
	SINGLE	ERR2039246	50	Illumina HiSeq 2500	5,723,264	349,119,104
Bacillus cereus	PAIRED	SRR392456	2 x 100	Illumina HiSeq 2000	7,722,767	1,559,998,934
	SINGLE	SRR1118191	200	Illumina HiSeq 2000	8,199,681	1,656,335,562
Campylobacter jejuni	PAIRED	SRR3094442	2 x 100	Illumina HiSeq 2000	4,039,559	807,911,800
	SINGLE	SRR3094490	75	Illumina Genome Analyzer II	3,619,867	260,630,424
Salmonella enterica	PAIRED	SRR3049469	2 x 100	Illumina HiSeq 2500	1,500,491	280,770,533
	SINGLE	ERR000017	35	Illumina Genome Analyzer II	3,191,127	114,880,572
Streptococcus pneumoniae	PAIRED	ERR016715	2 x 50	Illumina Genome Analyzer II	1,989,390	228,779,850
	SINGLE	SRR072214	35	Illumina Genome Analyzer II	2,599,192	93,570,912
Listeria monocytogenes	PAIRED	SRR393537	2 x 100	Illumina Genome Analyzer II	1,267,995	254,866,995
	SINGLE	SRR397563	35	Illumina Genome Analyzer II	6,984,497	251,441,892

3.1.2 Hardware

All the selected assembler programs were run on a server machine equipped with four 2.4GHz Intel(R) Xeon(R) 4 CPU, 4 cores within each CPU, and 32 GB of random-access memory (RAM). The operating system is Ubuntu version 16.04 as the Linux distribution for our interface and architecture 64-bit with an internal storage of 1 TB.

3.1.3 Software

Table 3.2: The list of software used in this study

No.	Software	Function	Reference
1.	FastQC (version 0.11.5)	To identify low quality reads, sequencing biases and adaptors incorporated during library preparation.	(Andrews, 2010)
2.	Trimmomatic (version v0.36)	To trim or eliminate bad quality read and adaptor sequence	(Bolger et al., 2014)
3.	SPAdes (version 3.13.0)	To assemble and reconstruct the read sequences into contigs sequence by varying the kmer size	(Bankevich et al., 2012)
4.	ABySS (version 2.1.2)		(Simpson et al., 2009)
5.	Velvet (version 1.2.10)		(Zerbino & Birney, 2008)
6.	MaSuRCA (version 3.2.6)		(Zimin et al., 2013)
7.	IBM SPSS® Statistics (version V26)	To do statistical analysis	(George & Mallery, 2016)
8.	ABySS-fac (version 2.1.2)	To evaluate the assembly quality and continuity statistics of contigs sequences.	(Simpson et al., 2009)
9.	Genome Assembly Gold-Standard Evaluations (GAGE)	To validate the assembly quality and to assess the genome assembly's completeness	(Salzberg et al., 2012)
10.	gVolante (online tool)		(Nishimura et al., 2017)

A total of ten tools were applied in the analysis and described in Table 3.2. For raw sequencing data quality checking and trimming, FastQC and Trimmomatic were used. SPAdes, ABySS, Velvet and MaSuRCA were performed to assemble the bacterial genome reads (single-end and paired-end) and the quality of the output contigs was assessed by ABySS-fac. Then, GAGE and gVolante employed to validate the assemble contigs completeness. Figure 3.1 showed the workflow of the genome assembly in this study.

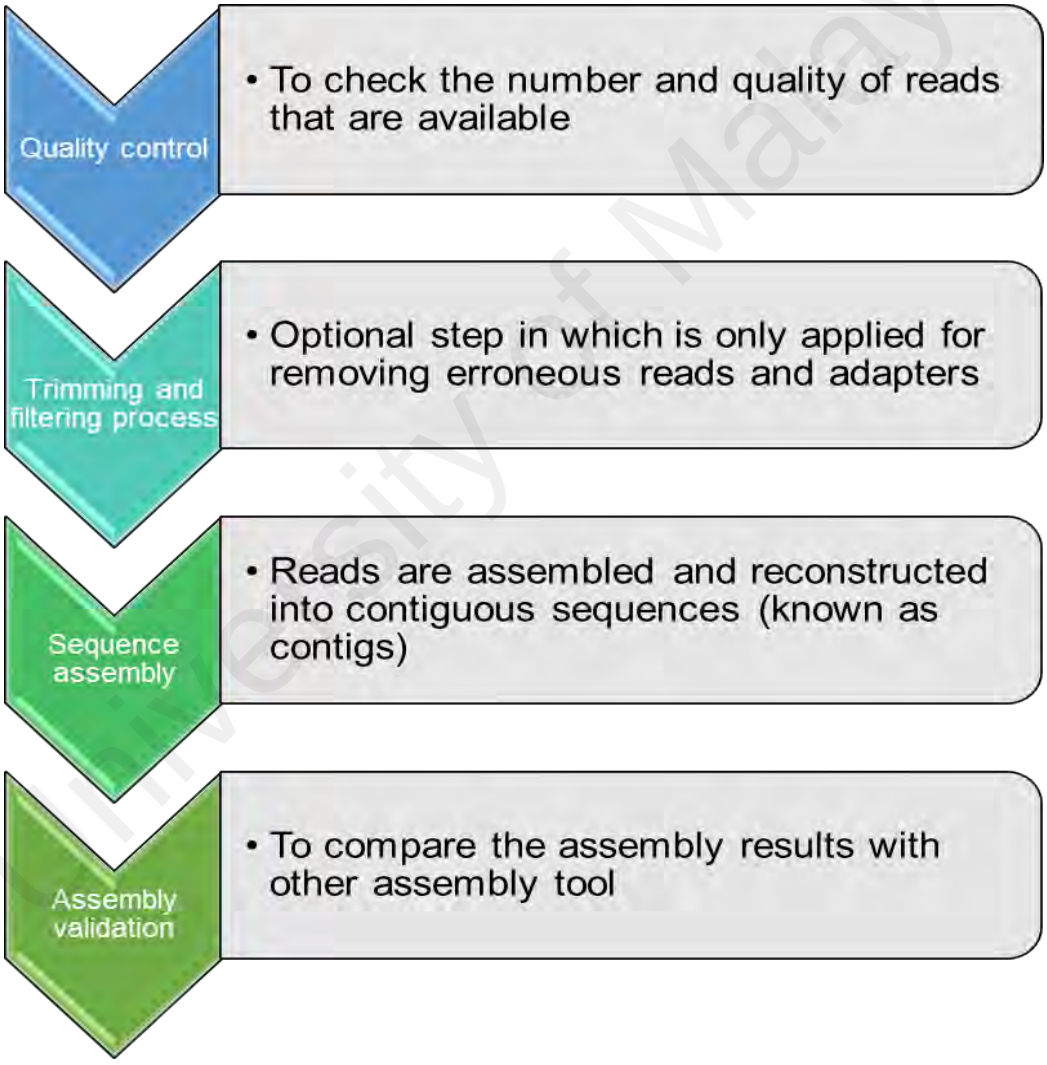


Figure 3.1: Workflow of the genome assembly in whole genome sequencing

3.2 Methodology

3.2.1 Pre-processing filtering and trimming of NGS reads

Data pre-processing represents an important step before any genome analysis conducted. All the real data in this study were verify whether the reads are of good quality. If the reads are considered in the "bad" result category, the reads have to go through the "cleaning up" process before further analysis. Thus, the FastQC (version 0.11.8) program is used to check the quality of the short reads. FastQC program provides a report that contains various metrics of the quality of the reads (Andrews, 2010). If low quality read identified, Trimmomatic (version v0.36) will do trimming to eliminate bases at 3′ end of each read with average quality per base drops below 20 over a 4 bp window and Illumina adapters.

3.2.2 Comparison of *de novo* genome assembly

Four tools, SPAdes, ABySS, Velvet and MaSuRCA were selected for this study for comparative analyses. During the experiment, all default values and parameter were used, and only the the k-mer value was changed. The range of k-mer that used was between 11 to 101 (except MaSuRCA which was automatics compute between k-mer 25 to 127). After that, we chose the four criteria that are useful which are **(I)** the highest percentage of contigs that were equal or longer than 500 bp, **(II)** highest total base pairs, **(III)** highest N50 and **(IV)** the lowest L50.

- **SPAdes** (Bankevich et al., 2012) is a genome assembly algorithm which was designed for single cell and multi-cells bacterial data sets. This tool is created based on Eulerian *de Bruijn* graph assemblers by applying paired *de Bruijn* graph (doubled-layered *de Bruijn* graph). The k-mers from DNA fragment reads build the inner *de Bruijn* graph, which is used for contig assembly. On the other hand, the 'paired k-mers' with large insert size build the outer *de Bruijn* graph, which is used for repeat resolving or scaffolding (Medvedev et al., 2011).
- **ABySS** (**A**ssembly **By** **S**hort **S**equences) assembled a genome usually large genomes by distributing a *de Bruijn* graph (parallel computation) across a cluster of computers. It assembled 3.5 billion pair-end reads from the genome of an African male publicly released by Illumina, Inc. (Simpson et al., 2009).
- **Velvet** has become a standard and very well-known assembler among biologist. It is one of the foremost tools created for assembling short reads data which applied *de Bruijn* graph-based (Jared T Simpson & Durbin, 2012). Similar with SPAdes, Velvet is one of the Eulerian *de Bruijn* graph assembler. However, velvet uses bidirectional *de Bruijn* graph (Zerbino & Birney, 2008).
- **MaSuRCA** (**Ma**ryland **Su**per **R**ead **C**abog **A**ssembler) is whole genome assembly software that can assemble all sizes genomes, from bacteria genomes to mammalian genomes to large plant genomes. It also can assemble data sets containing only short reads from Illumina sequencing or a mixture of short reads and long reads (Sanger, 454, Pacbio and Nanopore). It combines the efficiency and capability of the Overlap-Layout-Consensus (OLC) and the *de Bruijn* graph approaches.

3.2.2.1 Computational performance

The running time consumption metrics has been calculated for computational performance. It is the total time taken by the assembler to complete the assembly process for a given dataset. Time measurements are taken using the Linux utility commands *time*.

3.2.2.2 Assembly quality performance

Several assembly metrics were used for the assembly comparison. There are a number of contigs (n), number of contigs at least 500 bp (n:500), the number of contigs equal to or longer than N50 reported in the N50 column (L50), smallest contig (min), largest contig (max), N50 contig length (N50), N80 contig length (N80), N20 contig length (N20), the sum of the square of the sequence sizes divided by the assembly size (E-size) and sum of contig lengths (sum). All these metrics determined using ABySS-fac to assess the quality between these assemblers.

3.2.3 Evaluation and Validation

The *in-silico* evaluation of assemblies was performed using Genome Assembly Gold-standard Evaluations (GAGE) and gVolante. GAGE is a tool with an objective to evaluate the performance of different assembly tools using standardized data sets. This program is also could be as reference for assisting researchers in planning and managing their sequencing project which as we know that most appropriate criteria of sequencing experimental designs (depending on species of interest) are assembler and parameters values (Salzberg et al., 2012). In addition, gVolante provides a user-friendly interface to the researchers to assess the completeness of their contigs and scaffolds. There are several options can be choose based on the data sets such as sequence type, which pipeline and parameters the researchers preferred to use based on their objectives of studies and so forth. gVolante can generate an analysis reports (zip file) for future work.

CHAPTER 4: RESULTS

Results comprise several analyses which is starting with a pre-processing filtering and trimming of NGS reads output that was generated by FastQC and Trimmomatic software. Then, computational performance results were obtained from the total assembling time of four *de novo* assemblers using Linux *time* command and the differences on the total assembling time according to the types of assemblers also was compared using Kruskal-Wallis test. After that, assembly quality assessments and comparisons of assembled contigs from SPAdes, ABySS, Velvet and MaSuRCA were acquired. Lastly, the assembly quality was validated using Genome Assembly Gold-Standard Evaluation (GAGE) and gVolante.

4.1 Pre-processing filtering and trimming of NGS reads output

There are seven bacteria that had been selected with different length for this study. The first step after the sequencing process is to check the quality of the generated reads. This is due it may affect the following processes such as assembly analysis, incorrect base calling, annotation investigation, downstream applications and others. Adapter and low-quality reads (flaws in librarypreparation and sequencing) were filtered using FastQC to obtain an optimal quality score of 20 or higher at each base. The poor bases (bases in quality score below than 20) that had been identified need to be trimmed and filtered using a standalone trimmer tool, which in our case we used Trimmomatic version 0.36. Figures 4.1 and 4.2 show the output reads (single-end and paired-end) before and after the filtering and trimming processes.

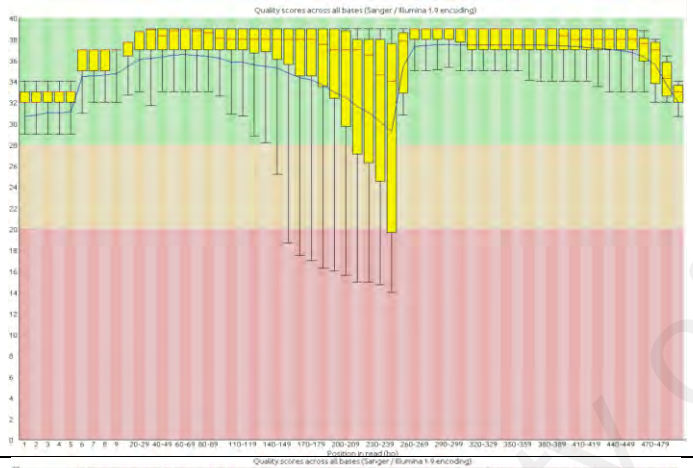
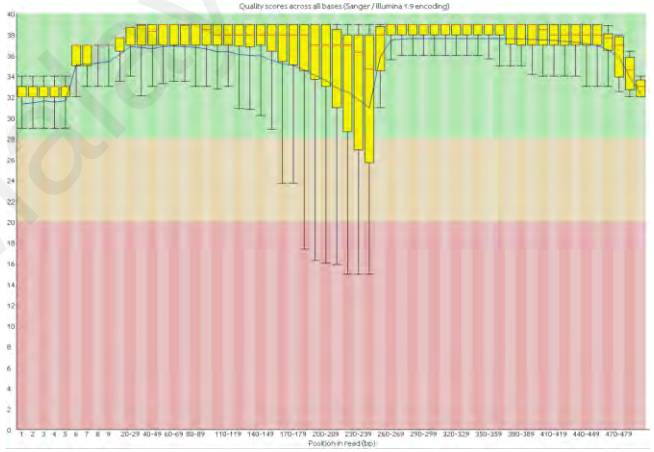
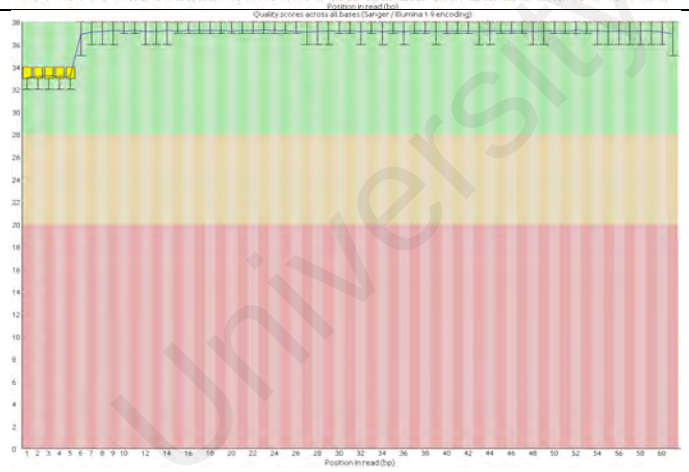
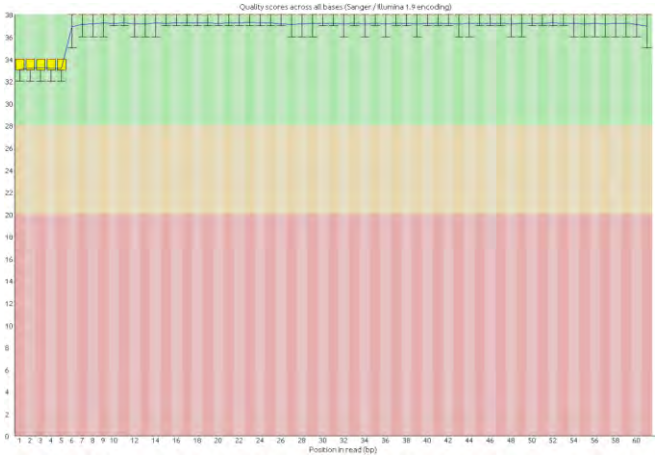
SINGLE-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Clostridium botulinum		
Escherichia coli		

Figure 4.1: The quality control of all selected bacteria (single-end) before and after trimming process.

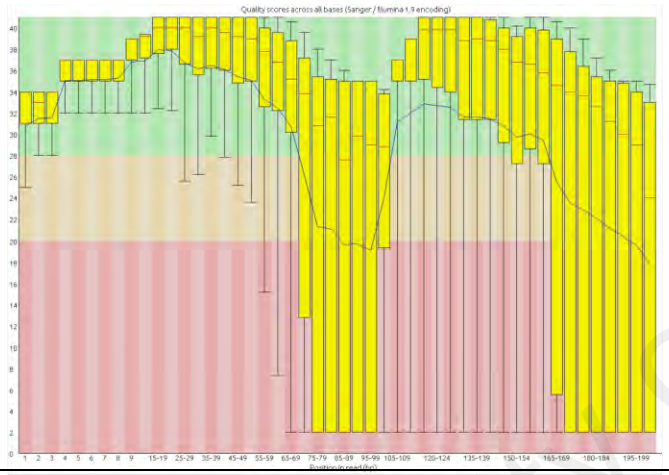
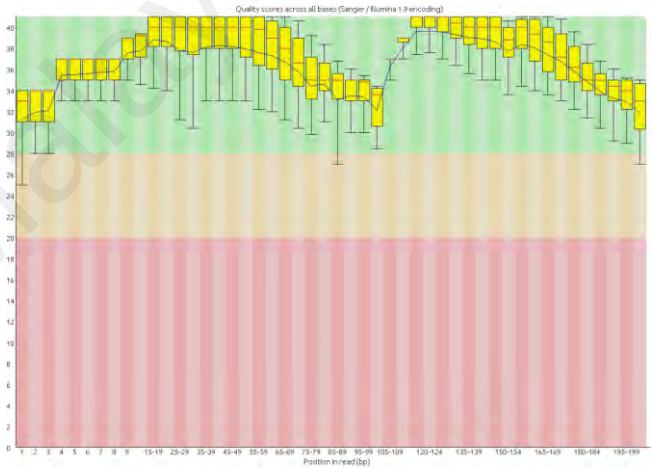
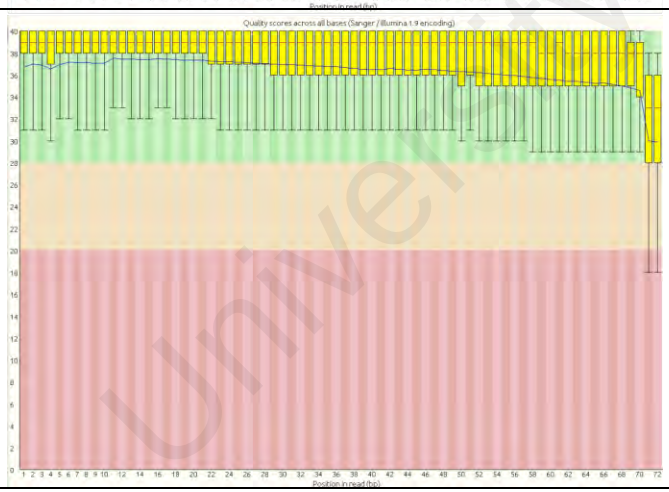
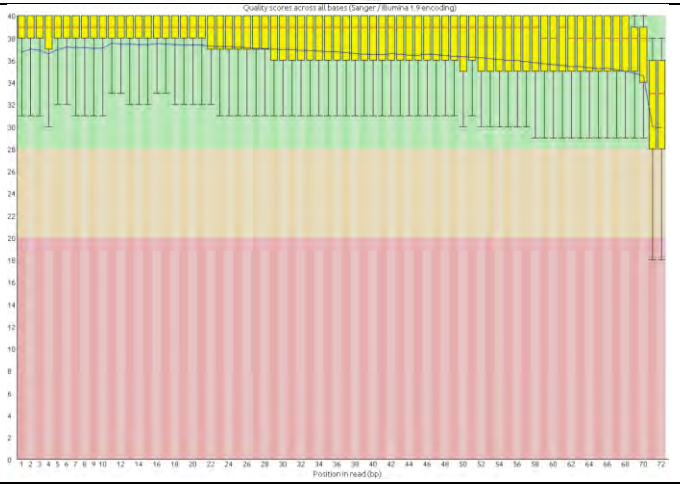
SINGLE-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Bacillus cereus		
Campylobacter jejuni		

Figure 4.1, continued.

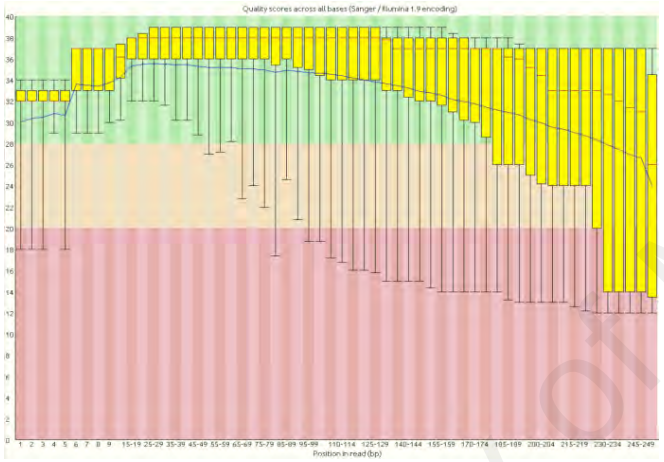
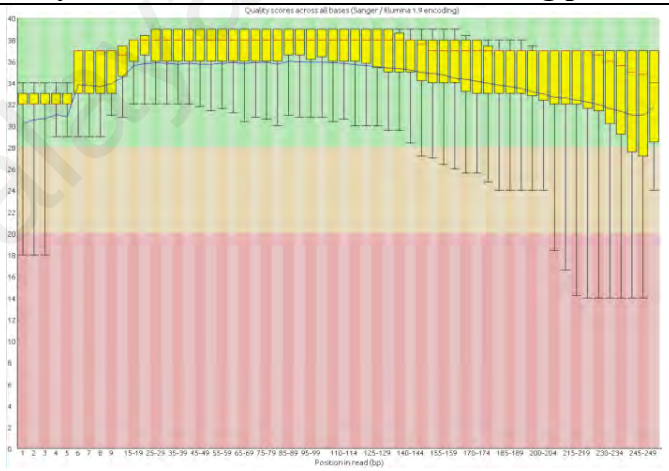
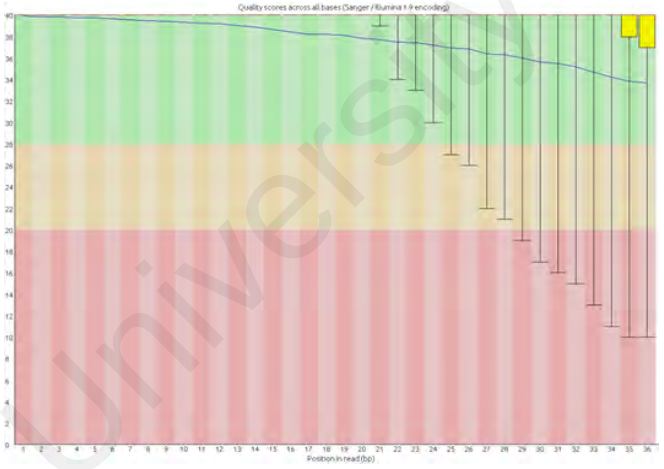
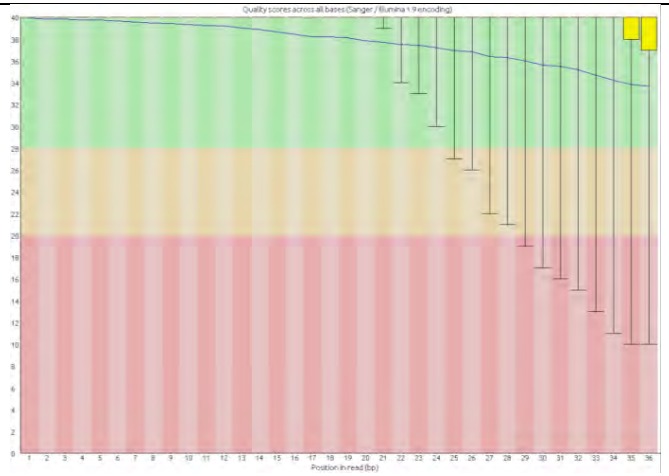
SINGLE-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Salmonella enterica		
Streptococcus pneumoniae		

Figure 4.1, continued.

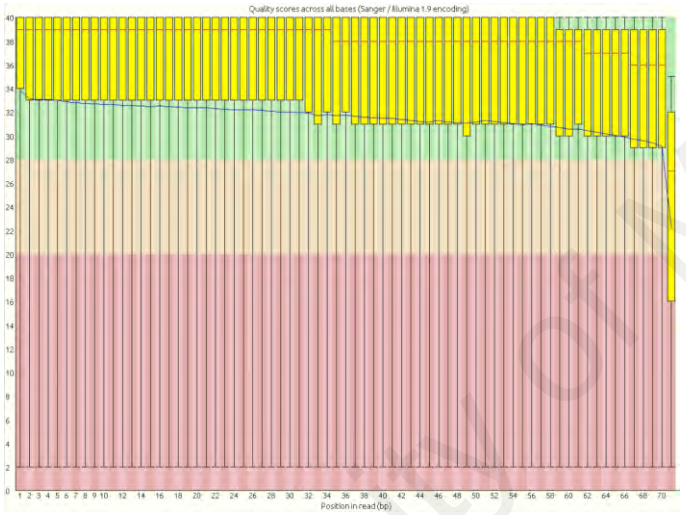
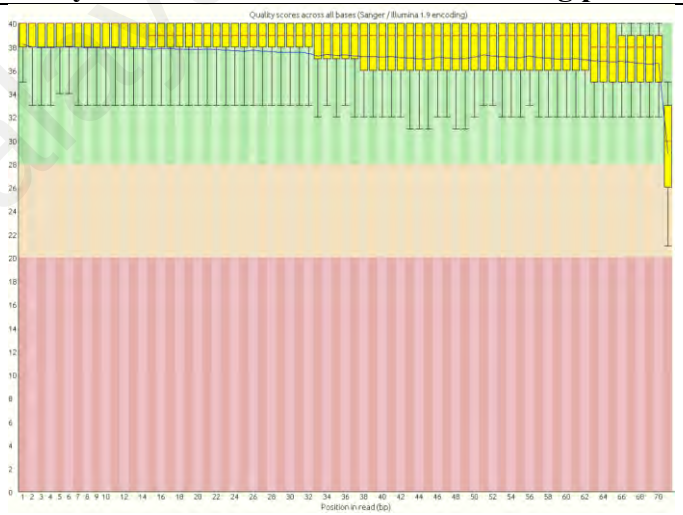
SINGLE-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Listeria monocyto genes		

Figure 4.1, continued.

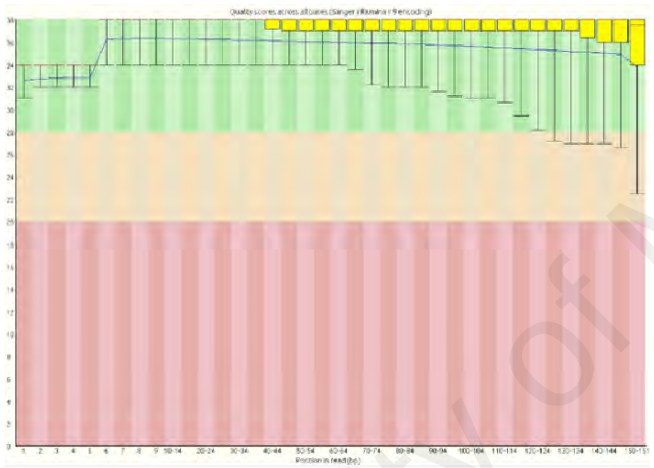
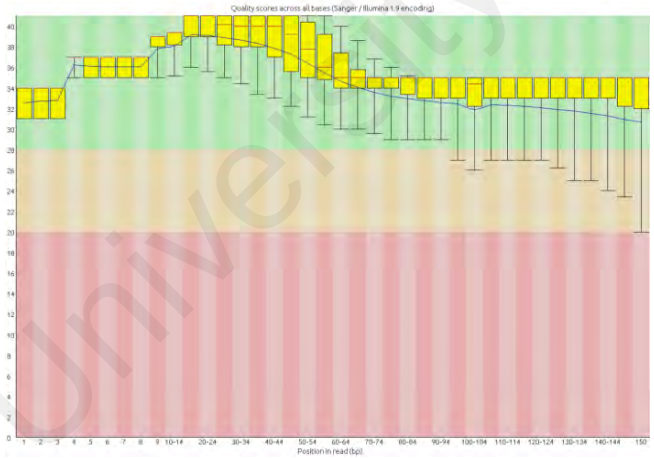
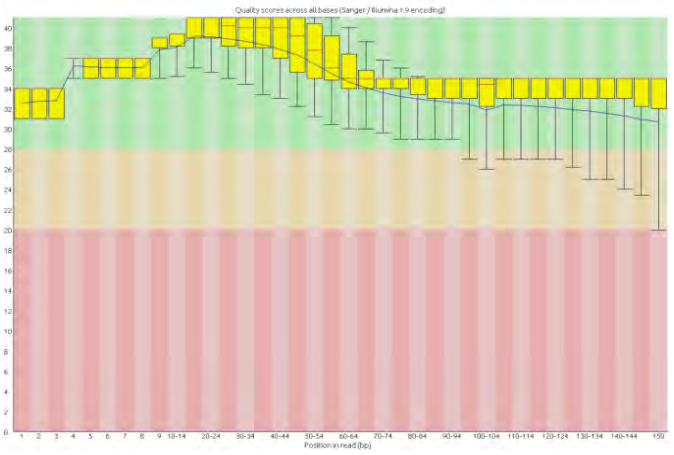
PAIRED-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Clostridium botulinum		
Escherichia coli		

Figure 4.2: The quality control of all selected bacteria (paired-end) before and after trimming process.

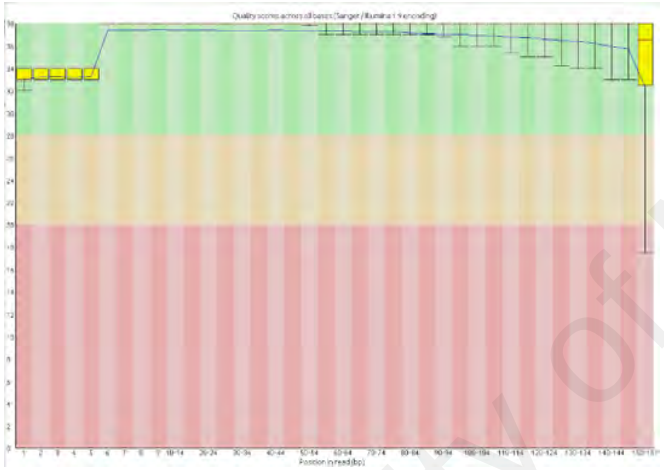
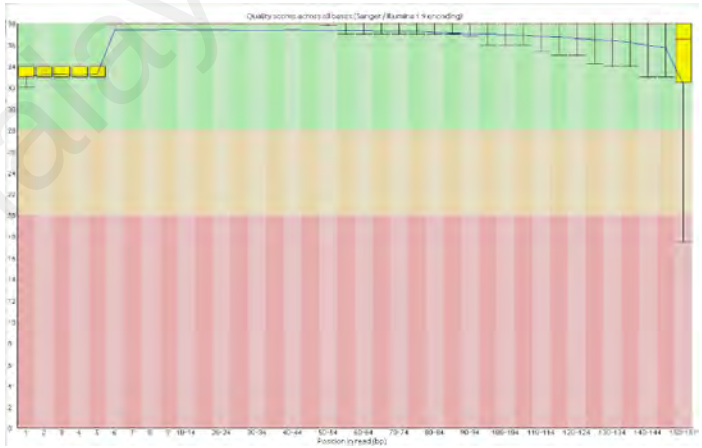
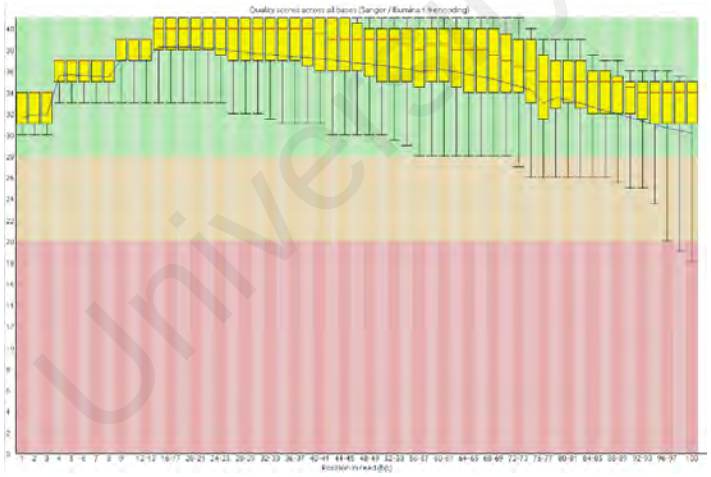
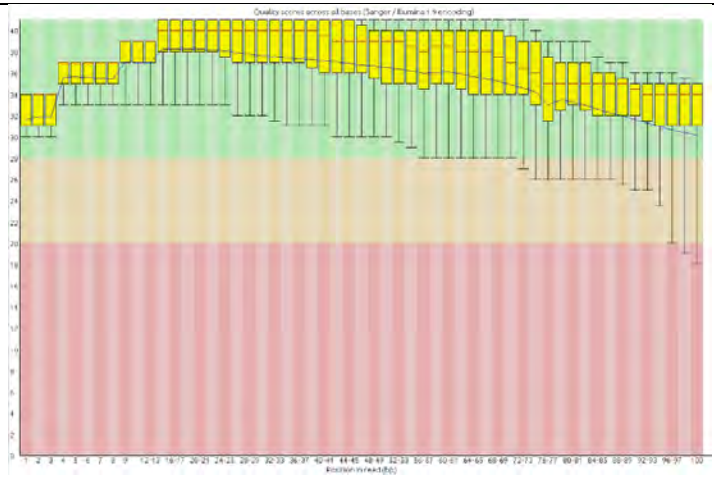
PAIRED-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Bacillus cereus		
Campylobacter jejuni		

Figure 4.2, continued.

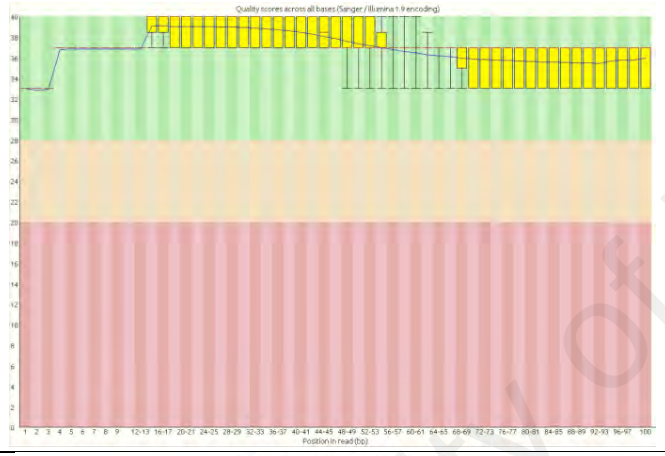
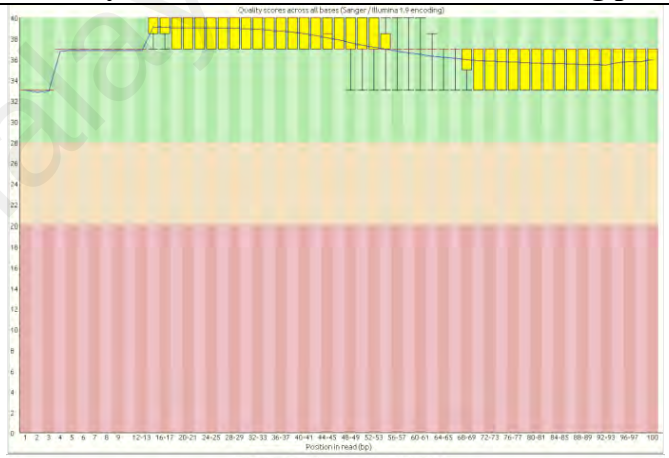
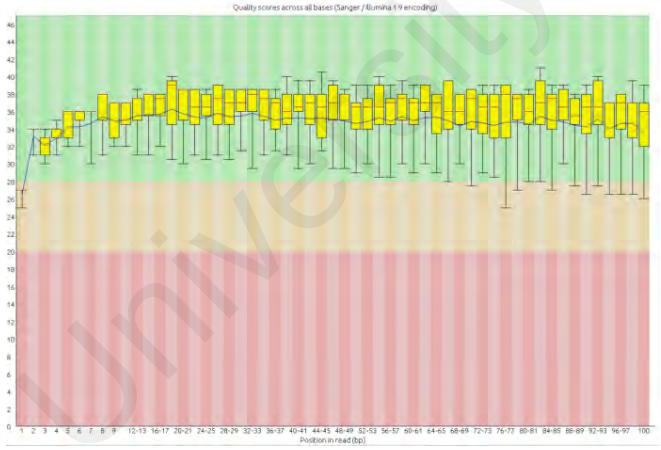
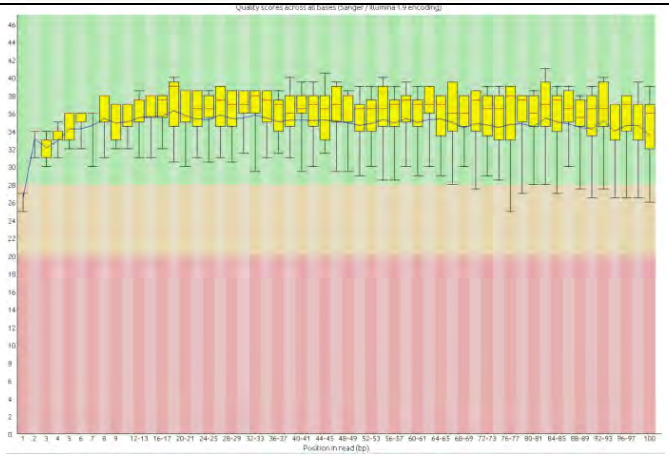
PAIRED-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Salmonella enterica		
Streptococcus pneumoniae		

Figure 4.2, continued.

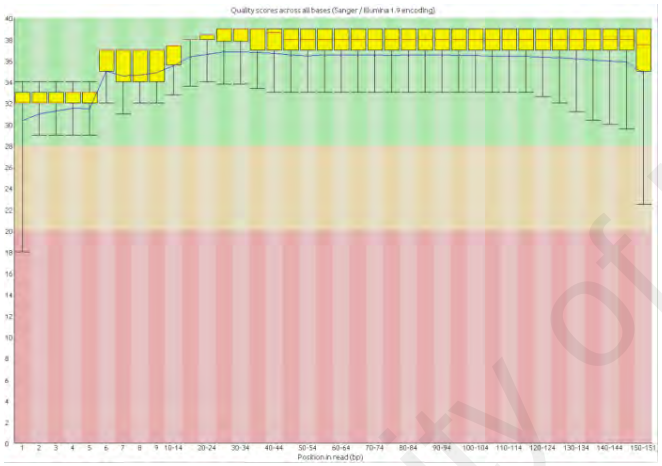
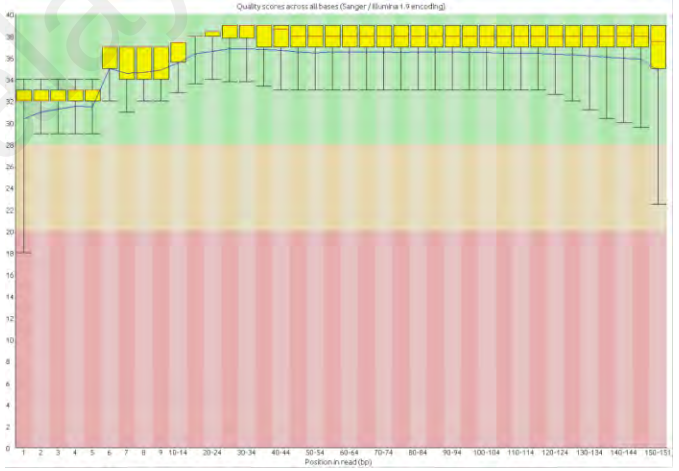
PAIRED-END		
Types of bacteria	**Quality control of the reads before trimming process**	**Quality control of the reads after trimming process**
Listeria monocytogenes		

Figure 4.2, continued.

4.2 Computational performance results

4.2.1 Running time

The total assembling time in seconds was calculated using Linux *time* command and the differences on the total assembling time according to the types of assemblers was compared using Kruskal-Wallis test. For single-end reads data sets, there was no significant difference of total assembling time according to the types of assemblers. This is due the output p-value > 0.05 (Chi square = 5.141, p-value = 0.16, degree of freedom = 3), with a mean rank time (seconds) score of 20.00 for SPAdes, 11.71 for ABySS, 11.14 for Velvet and 15.14 for MaSuRCA.

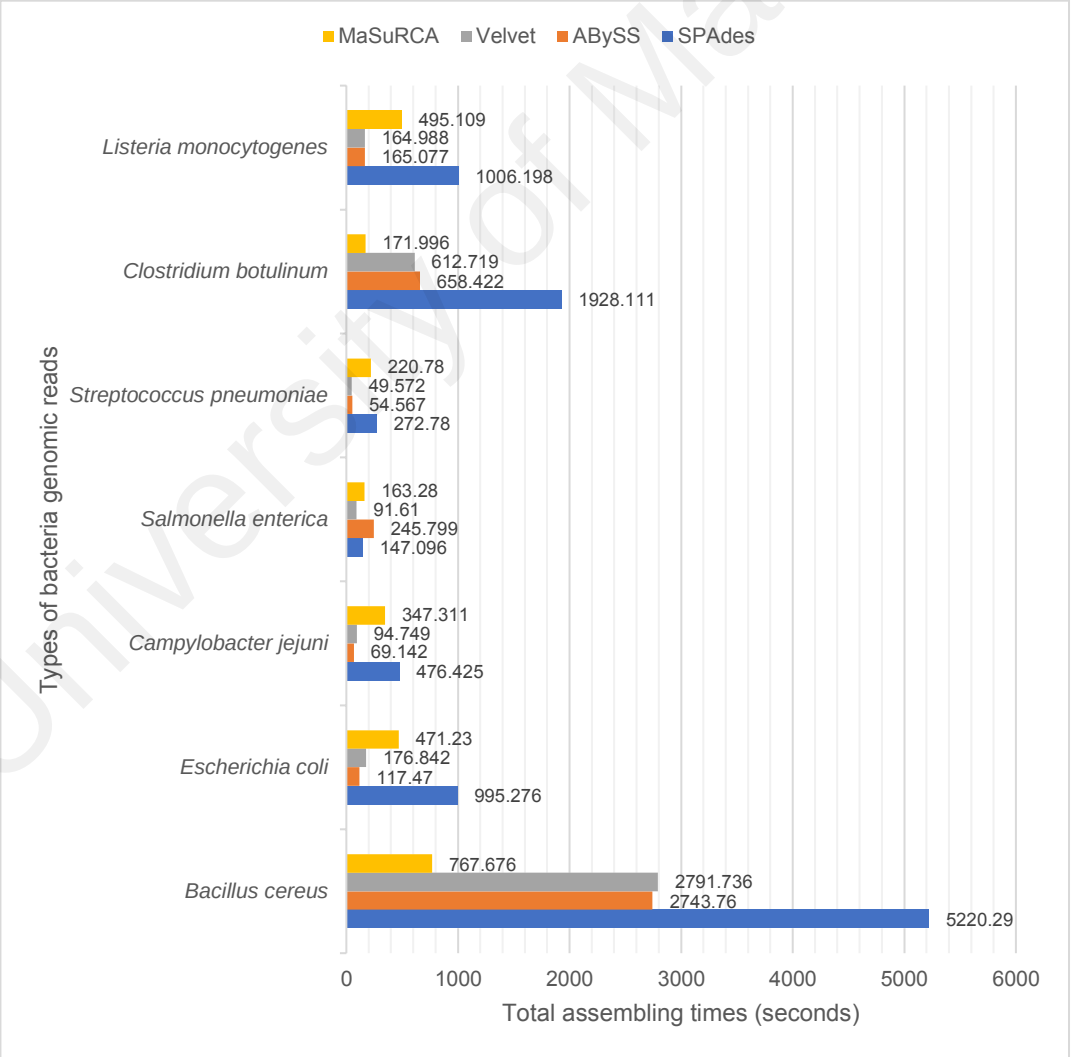


Figure 4.3: The total assembling time of each assembler comparison for single-end data sets.

For paired-end reads data sets, there was a statistically significant difference (Chi square = 11.390, p-value = 0.01, degree of freedom = 3), with a mean rank time (seconds) score of 20.86 for SPAdes, 12.86 for ABySS, 6.86 for Velvet and 17.43 for MaSuRCA. The result showed that Velvet consumed lowest time (in mean) of 6.86 second while SPAdes consumed more time with 20.86 seconds compared to other assemblers. Figures 4.3 and 4.4 showed the total assembling time of each assembler comparison for single-end and paired-end bacterial genomics reads data sets.

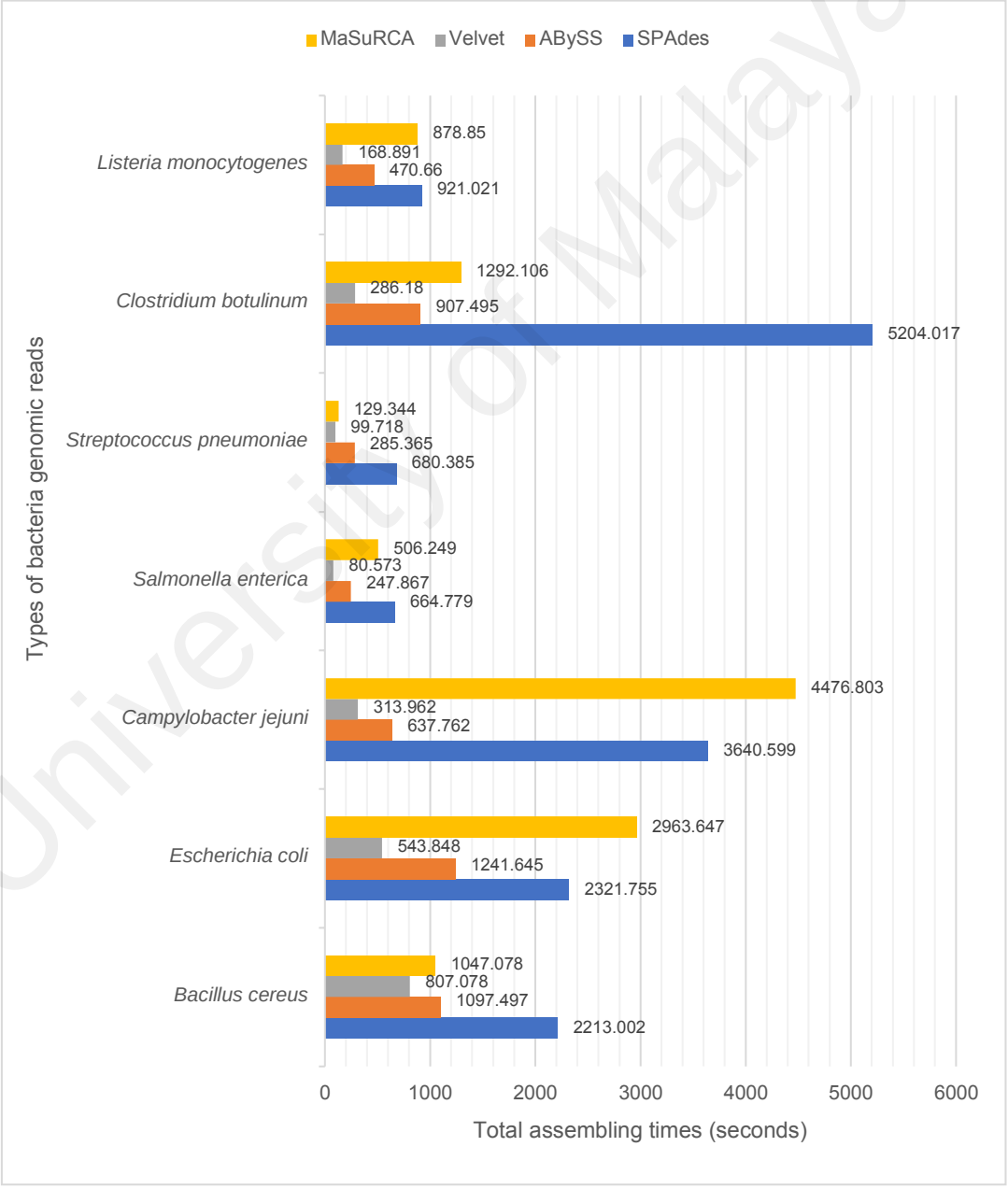


Figure 4.4: The total assembling time of each assembler comparison for paired-end data sets.

4.3 Assembly quality assessments and comparisons of assembled contigs

Each of the bacteria had its own size of reads and based on the Illumina technology. The details had been stated clearly at Table 3.1 at page 25. Each of the bacterial genome were run in single-ends and paired-ends sequences with different number of k-mer starting from 11 until 101 using three different assemblers (SPAdes, ABySS and Velvet) while MaSuRCA was automatics computing k- mer between 25 until 127.

In this study, ABySS-fac is used to compare the contiguity sequences between these assemblers. 'ABySS-fac' is one of the programs in ABySS tools (see Materials and Methodology, subsection 3.1.3, page 26) with the function to calculate the contiguity of the assembly sequences. This program is unrelated with the ABySS assembler. The assembly metrics in ABySS-fac includes number of contigs (n), number of contigs at least 500 bp (n:500), the number of contigs equal to or longer than N50 reported in the N50 column (L50), smallest contig (min), largest contig (max), N50 contig length (N50), N80 contig length (N80), N20 contig length (N20), the sum of the square of the sequence sizes divided by the assembly size (E-size) and sum of contig lengths (sum). The reason of using ABySS-fac is to standardize the calculation of the contigs. Furthermore, SPAdes do not have their own statistic tool. Thus, one program should be used to calculate all assemblers' outputs from all different data and different assembly tools.

Four criteria that are useful to choose the ideal tool for the selected data sets are (I) the lowest number of contigs at the value reported in the L50 column (L50), (II) highest N50 length, (III) the highest percentage of contigs that were longer than 500 bp and (IV) the highest total base pairs obtained. N50 length is calculated by first ordering all contigs (or scaffolds) by length from longest to shortest. Then summing their lengths until the sum exceeds 50% of the total length of all contigs (Blawid et al., 2017). L50 is the number of

contigs (or scaffolds) of the N50 base pair in length location. The total base pairs is the total numbers of nucleotide in particular a strand. Lastly, the percentage of contigs that were longer than 500 bp was calculated by the number of contigs at least 500 bp (n:500) divided by number of contigs (n) times 100. Figures 4.5 until 4.8 showed the statistical results for single- end contigs while figure 4.9 until 4.12 showed the statistical results for paired-end contigs of all the bacteria genomic reads. These stated the percentage of contigs that were equal or longer than 500 bp, L50 value, N50 length and total base pairs.

4.3.1 Single-ends read

The percentage of contigs that were equal or longer than 500 bp for each bacterial genome data (single-end) was calculated and showed in figure 4.5.MaSuRCA produced the highest percentage of contigs that were equal or longer than 500 bp for most bacterial genome data sets which are more than 85.00%. The second highest percentage is SPAdes, followed by Velvet and ABySS.

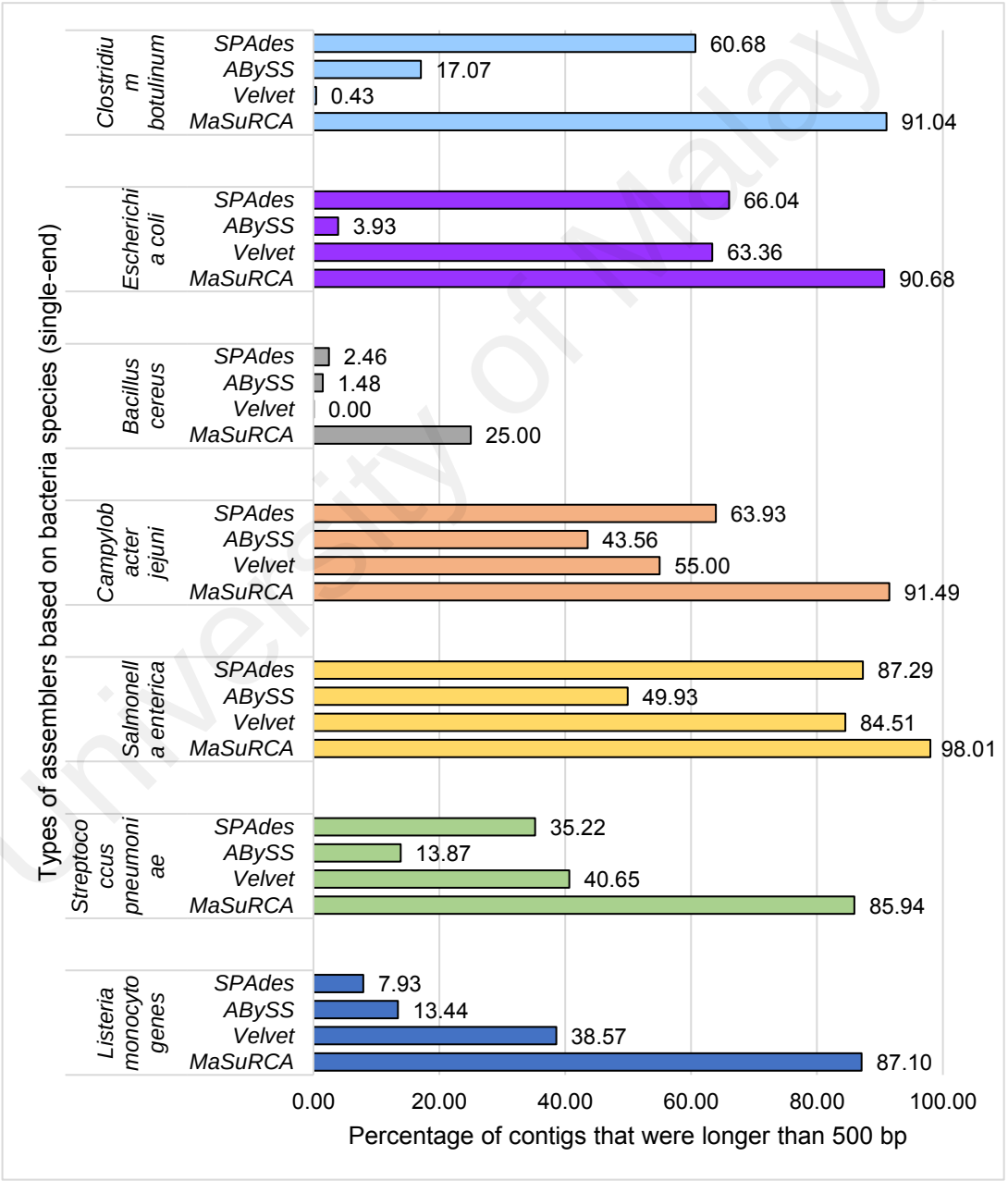


Figure 4.5: Graph of percentage of contigs that were equal or longer than 500 bp vs types of assemblers based on bacteria species (single-ends)

Based on Figure 4.6, MaSuRCA produced the lowest L50 values for most bacterial genome data sets except *Campylobacter jejuni* and *Salmonella enterica*. Furthermore, the L50 values of MaSuRCA and SPAdes (the lowest L50 value) for *Campylobacter jejuni* are high similar, MaSuRCA was 6 while SPAdes was 5. The same situation happened for *Salmonella enterica*, the L50 values of MaSuRCA and SPAdes (the lowest L50 value) are high similar, MaSuRCA was 30 while SPAdes was 28.

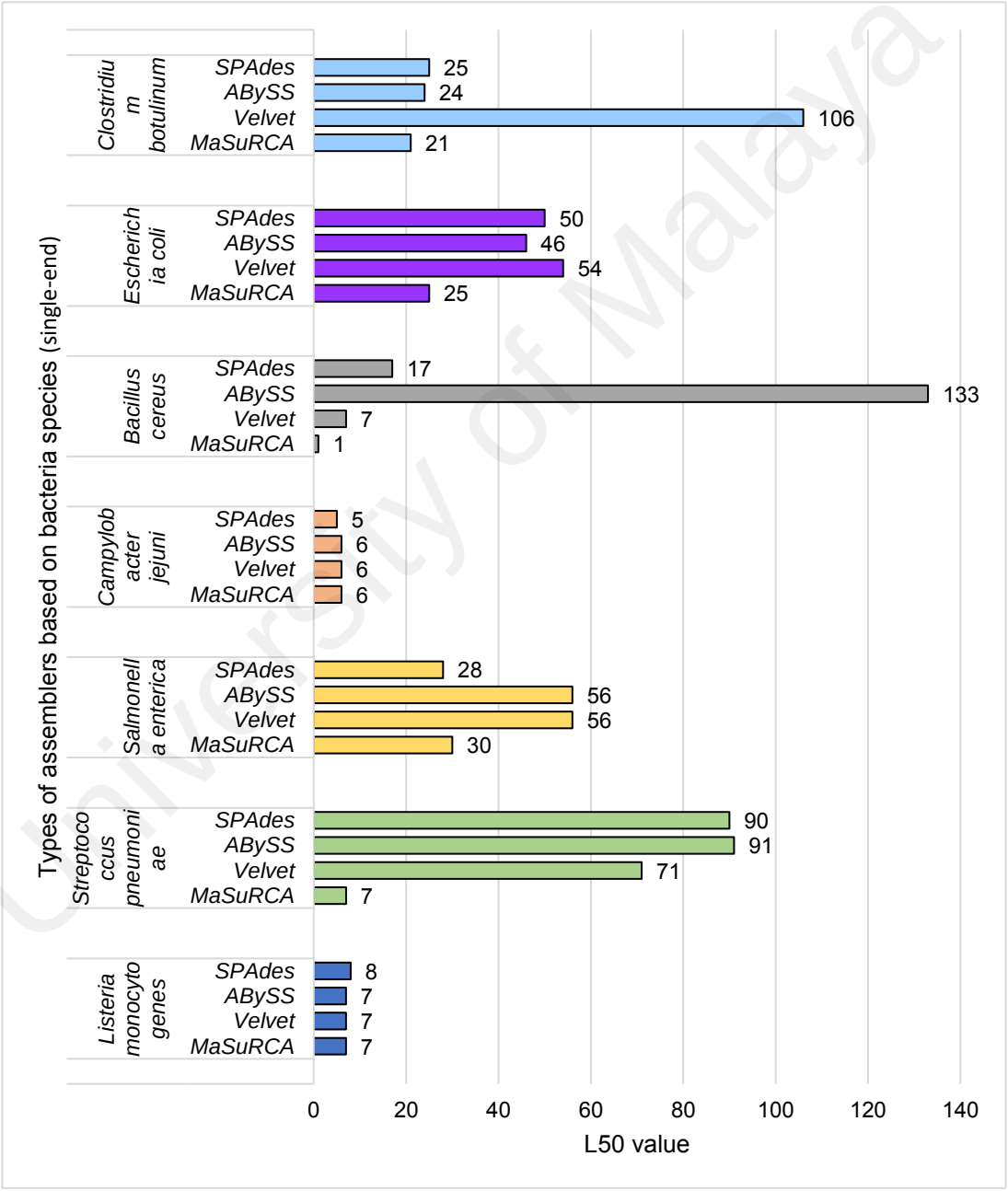


Figure 4.6: Graph of L50 values vs types of assemblers based on bacteria species (single-ends)

Based on Figure 4.7, MaSuRCA and SPAdes generated the highest N50 values for most bacterial genome data sets. The N50 values of MaSuRCA and SPAdes (the highest N50 value) for *Salmonella enterica* are similar, MaSuRCA was 50573 while SPAdes was 53085. However, for *Bacillus cereus, Campylobacter jejuni* and *Listeria monocytogenes* the N50 values of MaSuRCA was not similar with the tools that have the highest N50 values for these bacteria species. The N50 values of MaSURCA was 512 while SPAdes was 124534 for *Bacillus cereus,* the N50 values of MaSURCA was 91767 while SPAdes was 116955 for *Campylobacter jejuni* and The N50 values of MaSURCA was 110830 while ABySS was 143758 for *Listeria monocytogenes*

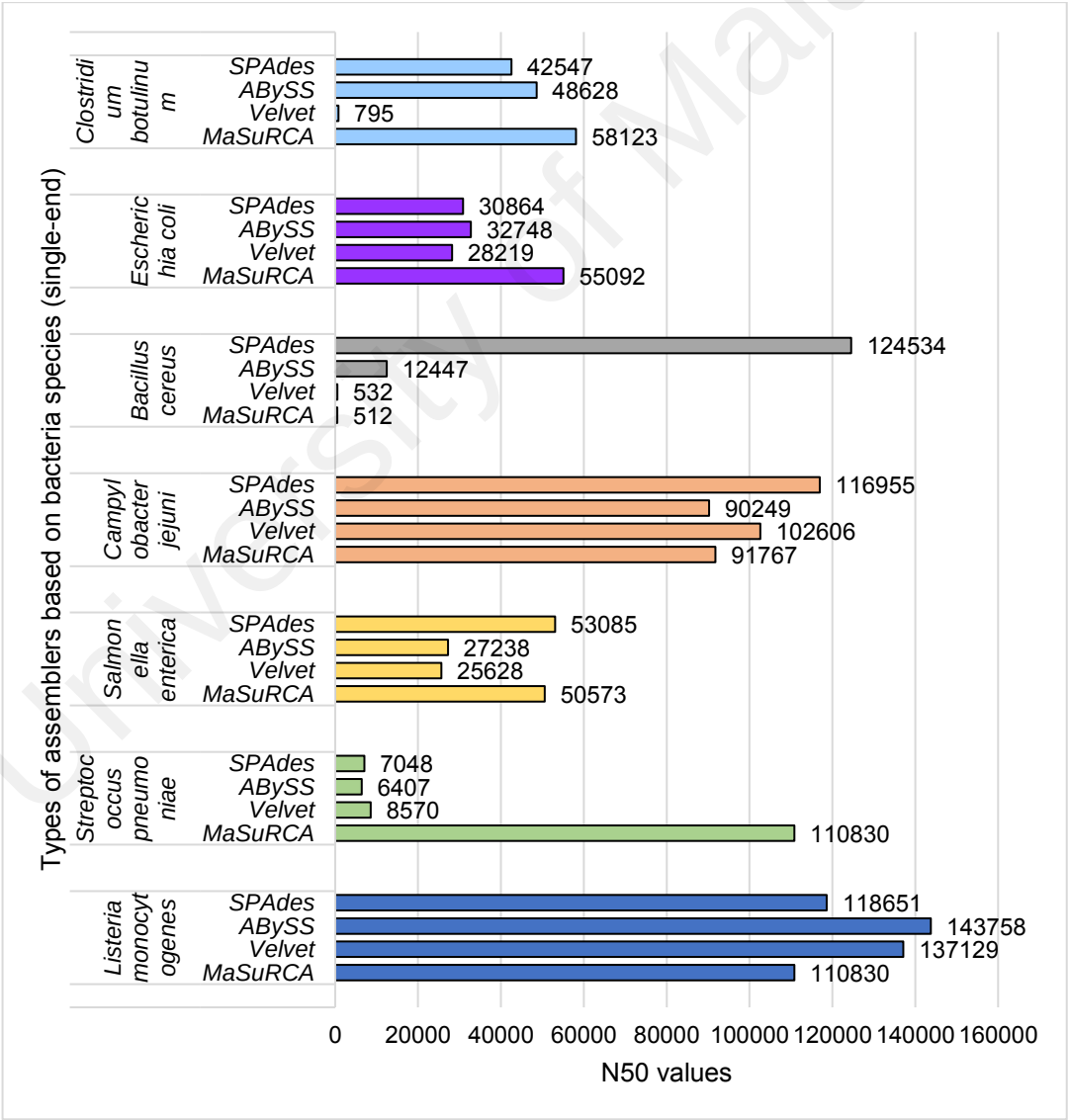


Figure 4.7: Graph of N50 values vs types of assemblers based on bacteria species (single-ends)

The graph of total base pairs vs types of assemblers based on bacteria species (single-ends) at figure 4.8 showed MaSuRCA generated the highest total base pairs for most bacterial genome data sets except *Clostridium botulinum, Bacillus cereus,* and *Salmonella enterica*. The total base pairs of MaSuRCA and SPAdes (the highest total basepairs) for *Salmonella enterica* are similar, MaSuRCA was 4853346 while SPAdes was 4864450. However, for *Clostridium botulinum* and *Bacillus cereus,* the total base pairs of MaSuRCA was not similar with the tools that have the highest total base pairs for these bacteria species. The total base pairs of MaSURCA was 3250712 while SPAdes was 3495662 for *Clostridium botulinum* and the total base pairs of MaSURCA was 512 while SPAdes was 7291165 for *Bacillus cereus.*

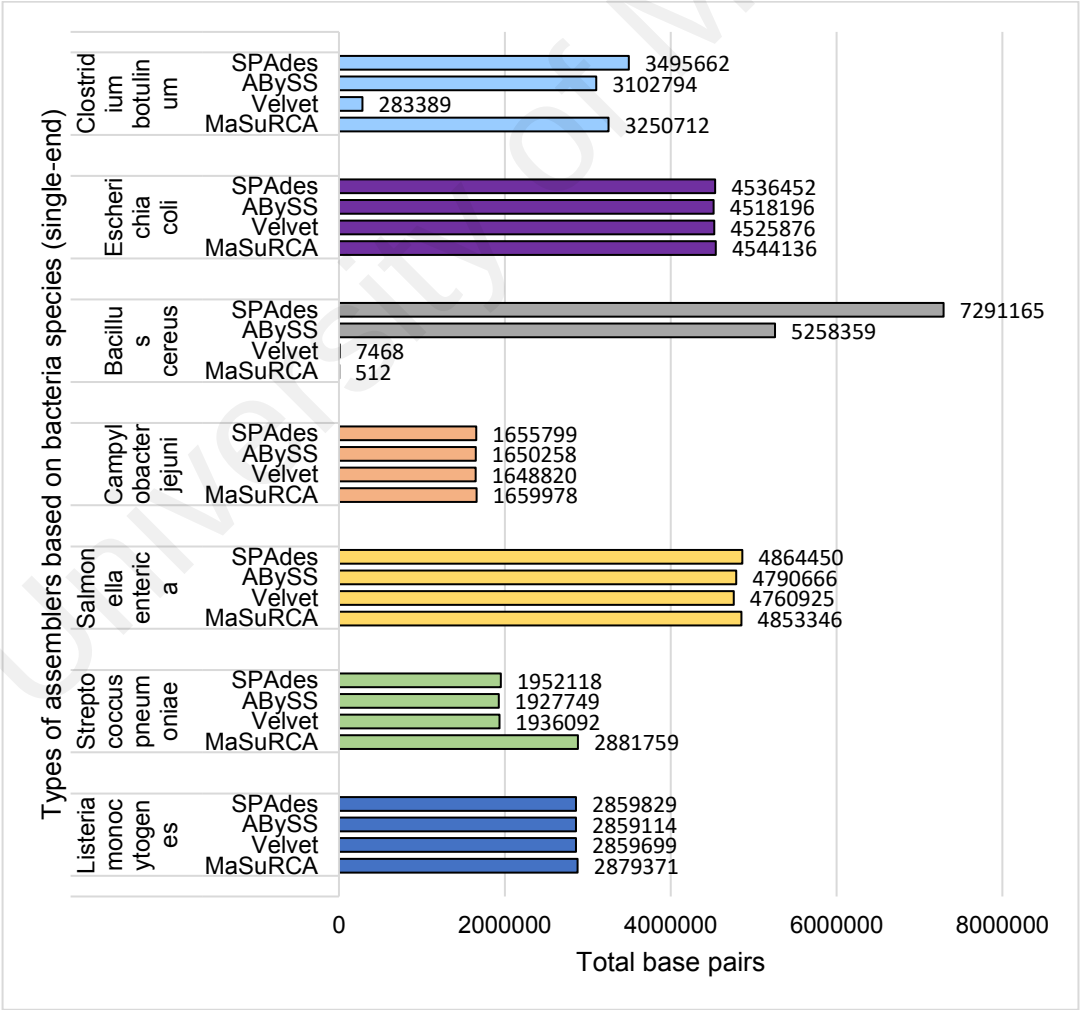


Figure 4.8: Graph of total base pairs vs types of assemblers based on bacteria species (single-ends)

For single-end bacteria genomic data sets, the analysis performed in this study, suggested that MaSuRCA is the best choice for sequencing the single-end bacteria genomic reads regardless of the size of reads that generated by different types of Illumina platforms.

4.3.2 Paired-end read

All selected bacteria were also running in paired-end data with different number of k-mer starting from 11 until 101 using three different assemblers (SPAdes, ABySS and Velvet) while MaSuRCA was automatics computing k-mer between 25 until 127. Figure 4.9 showed the result percentage of contigs that were longer than 500 bp for each assembler based on bacteria species (paired-ends). MaSuRCA produced the highest percentage of contigs that were equal or longer than 500 bp for most bacterial genome data sets which are more than 80.00% except *Bacillus cereus* and *Campylobacter jejuni*. The second highest percentage is SPAdes, followed by ABySS and Velvet.

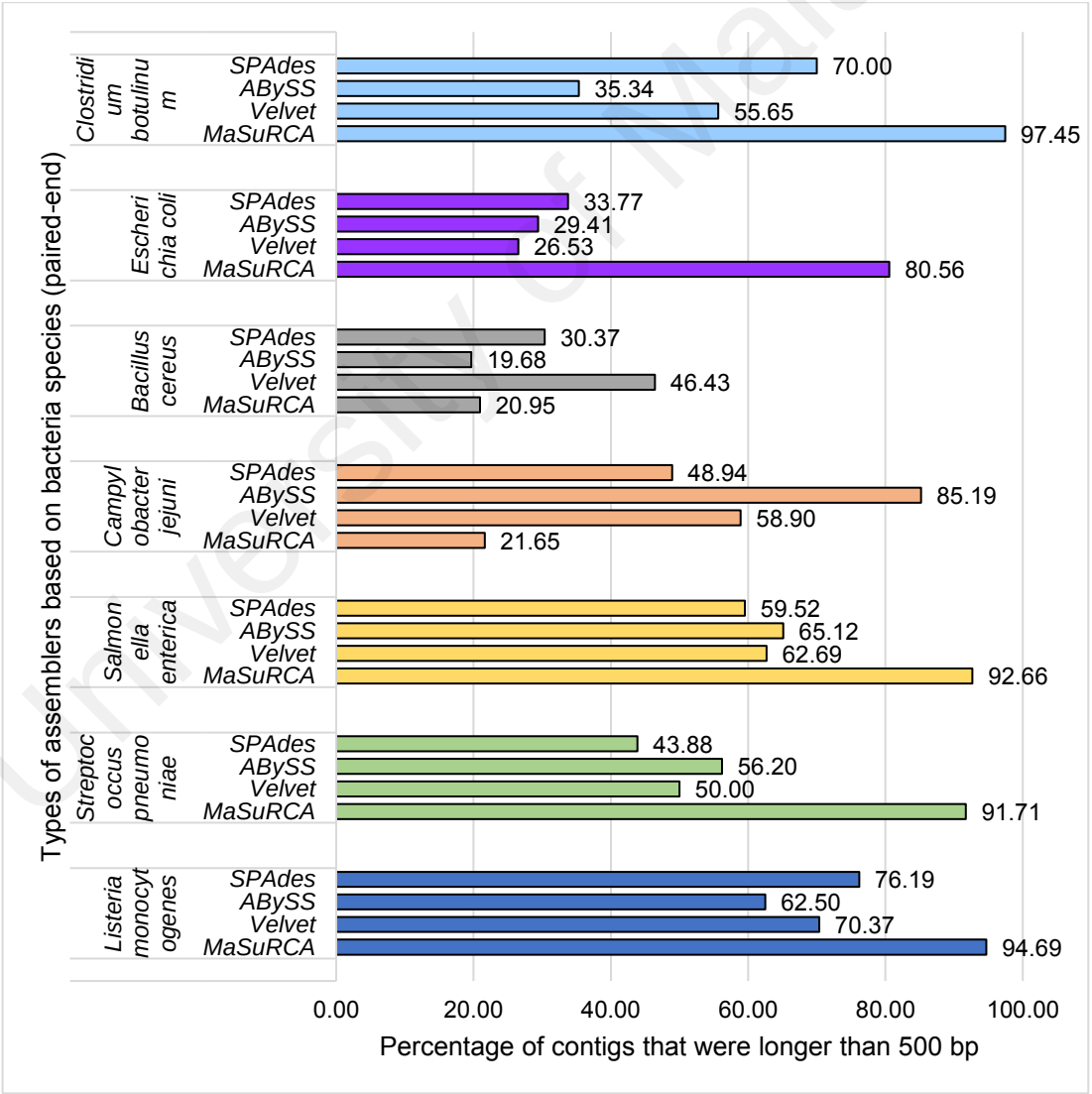


Figure 4.9: Graph of percentage of contigs that were longer than 500 bp vs types of assemblers based on bacteria species (paired-ends)

Based on Figure 4.10, Velvet produced the lowest L50 values for most bacterial genome data sets except *Bacillus cereus* and *Campylobacter jejuni*. Furthermore, the L50 values of Velvet and ABySS (the lowest L50 value) for *Campylobacter jejuni* are high similar, Velvet was 6 while ABySS was 4. However, for *Bacillus cerereus,* the L50 values of Velvet (10) was not similar with the tools that have the lowest L50 values for these bacteria species, ABySS (4).

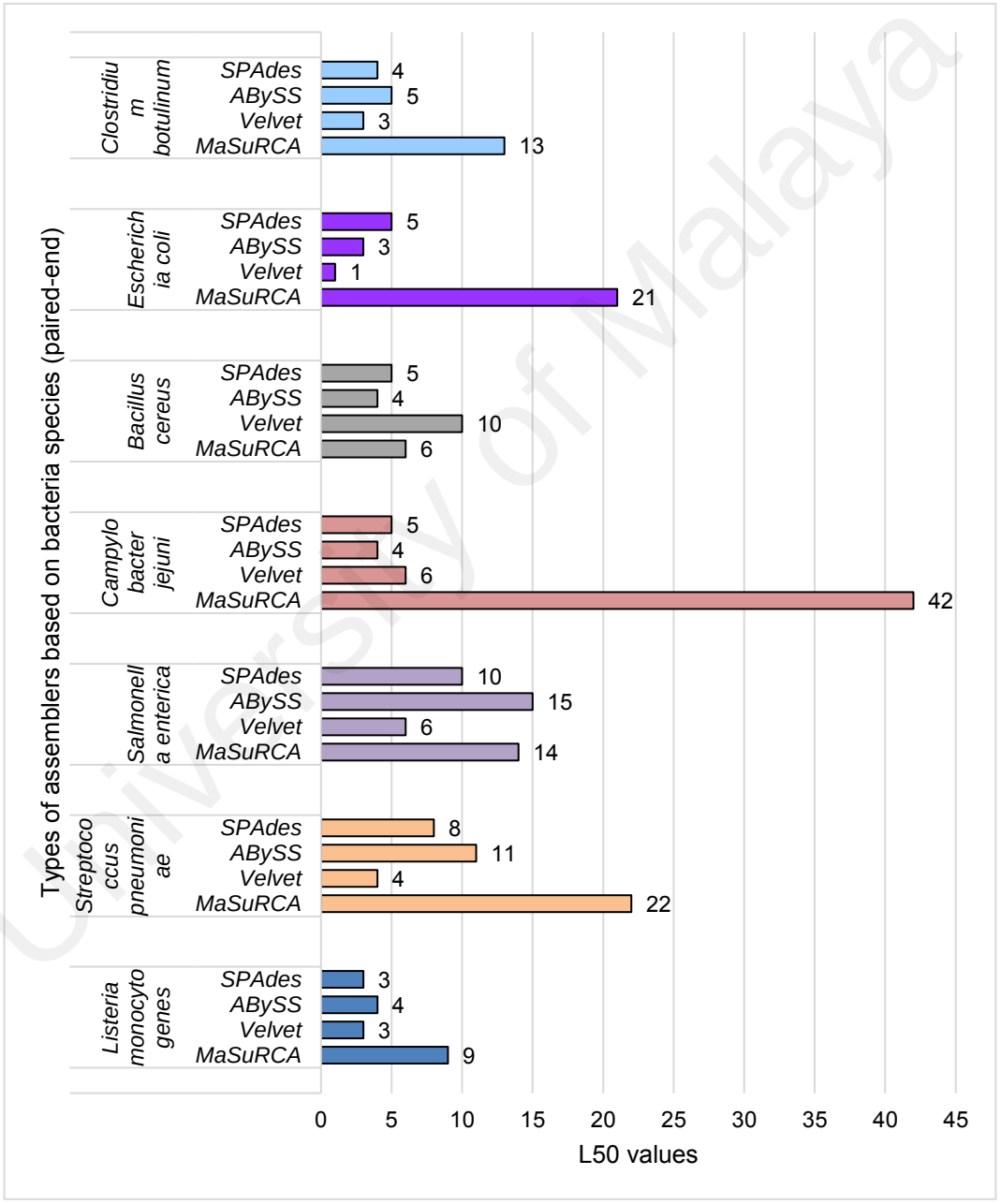


Figure 4.10: Graph of L50 values vs types of assemblers based on bacteria species (paired-ends)

Based on Figure 4.11, Velvet and SPAdes generated the highest N50 values for most bacterial genome data sets equally. Velvet produced the highest N50 values for *Escherichia coli* (2049844), *Salmonella enterica* (227850) and *Streptococcus pneumoniae* (191766) while SPAdes produced the highest N50 values for *Clostridium botulinum* (474479), *Campylobacter jejuni* (154554) and *Listeria monocytogenes* (491195).

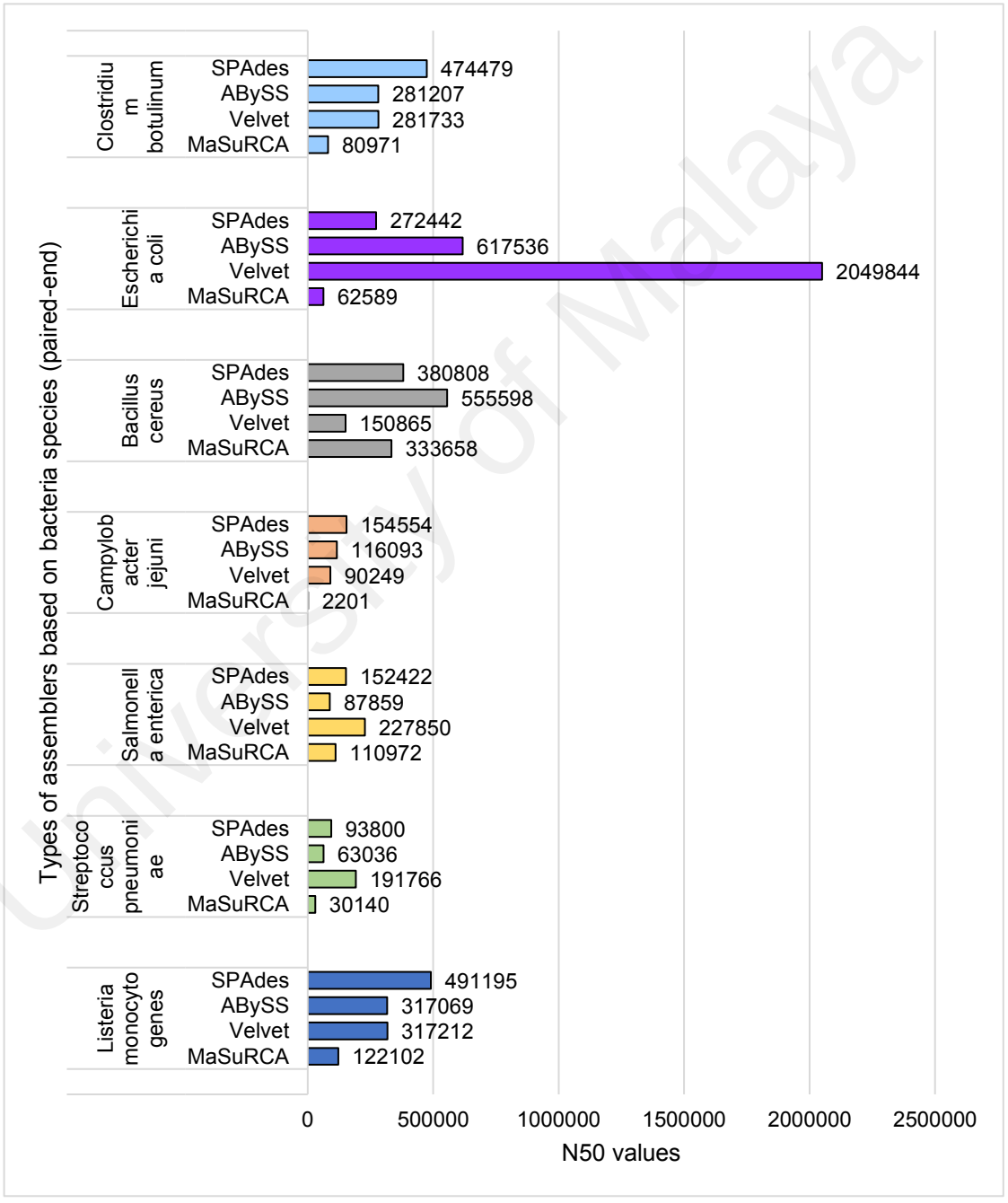


Figure 4.11: Graph of N50 values vs types of assemblers based on bacteria species (paired-ends)

The graph of total base pairs vs types of assemblers based on bacteria species (paired-ends) at figure 4.12 showed MaSuRCA generated the highest total base pairs for most bacterial genome data sets except *Clostridium botulinum, Bacillus cereus,* and *Campylobacter jejuni*. The total base pairs of MaSuRCA and ABySS (the highest total base pairs) for *Clostridium botulinum* and *Bacillus cereus* are similar, MaSuRCA was 3781166 bp while ABySS was 3851901 bp for *Clostridium botulinum* and for *Bacillus cereus* MaSuRCA was 5248746 bp while ABySS was 5252109bp. However, for *Campylobacter jejuni,* the total base pairs of MaSuRCA (409417 bp) was not similar with the tool that has the highest total base pairs for this bacteria species, ABySS (1676333 bp)

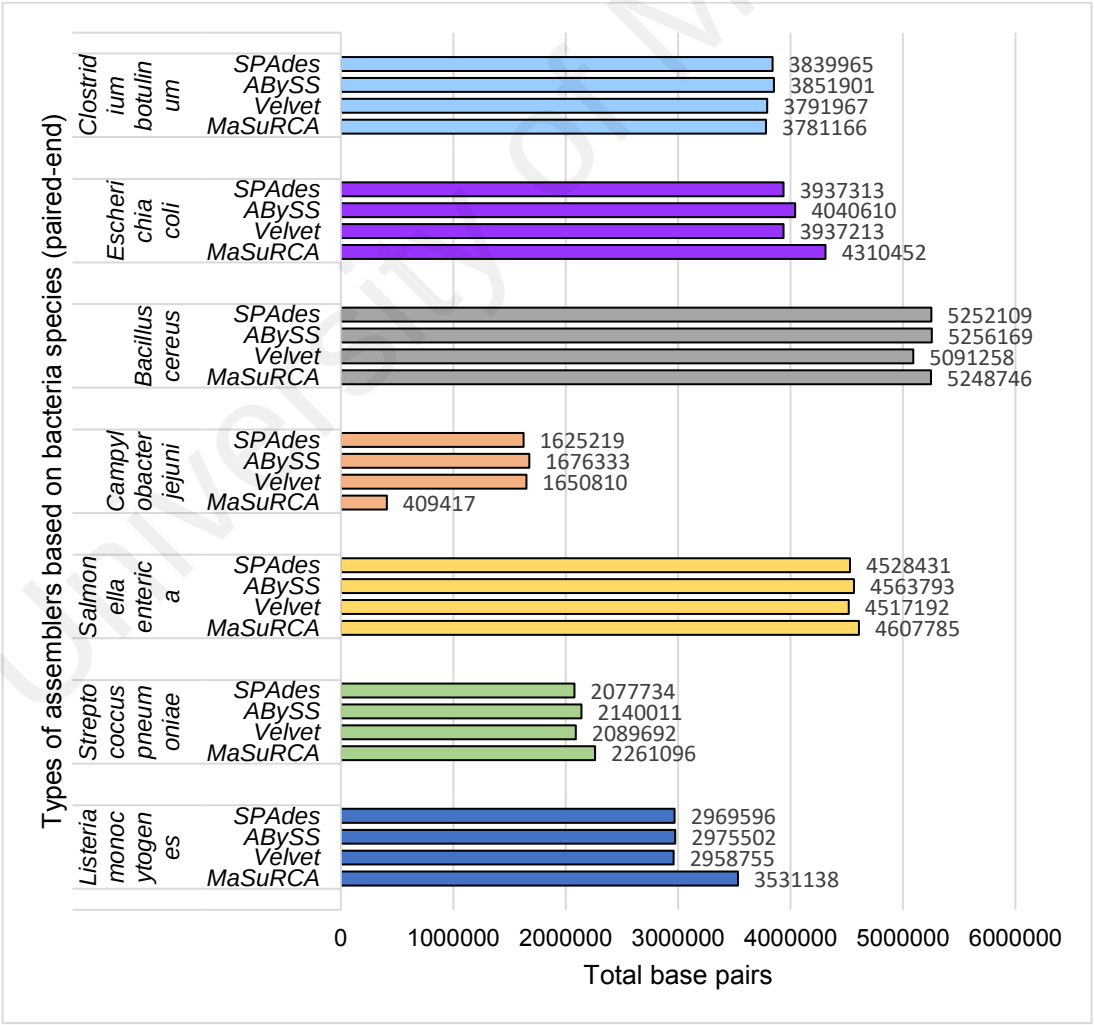


Figure 4.12: Graph of total base pairs vs types of assemblers based on bacteria species (paired-ends)

For paired-end bacteria genomic data sets, the analysis performed in this study, suggested that MaSuRCA and Velvet are the best choice for sequencing the paired-end bacteria genomic reads regardless of the size of reads that generated by different types of Illumina platforms.

4.4 Validation of the assembly quality

In the previous section, we have successfully assembled the bacterial genomic reads (single-ends and paired-ends) with four assemblers tools SPAdes, ABySS, Velvet and MaSuRCA. We also compared and assessed the quality of the contigs by four assembly metrics which were N50, L50 values, percentage of contigs that were longer than 500 bp and total base pairs. As we know, annotation is involved in tertiary analysis with the objective, to determine their biological significance, function and meaning from the genetic data. Thus, assembly analysis is an important and a key step towards successful genome annotation. In this section, we validated the assembly quality using Genome Assembly Gold-Standard Evaluations (GAGE) and gVolante as outlined in objective 2 (see page 5).

4.4.1 GAGE: Genome Assembly Gold-Standard Evaluations

Figure 4.13 and Figure 4.14 showed GAGE e-size value vs. types of bacterial genomic contigs data (single-end and paired-end) graphs. GAGE or Genome Assembly Gold-Standard Evaluations is an evaluation tool which provides a report regarding the quality of data, the degree of contiguity of the data produced by the assembler tools and the correctness of an assembly. E-size in GAGE refers as size of contigs or scaffolds in certain location relatively. E-size is calculated by

$$E = \sum (L)2/G \tag{4.1}$$

where L is the length of contig, and G is the length of genome estimated by the sum of all contig lengths. The same calculation is done to get E-size of scaffolds (Salzberg et al., 2012). The larger the E-size value, the better the assembly.

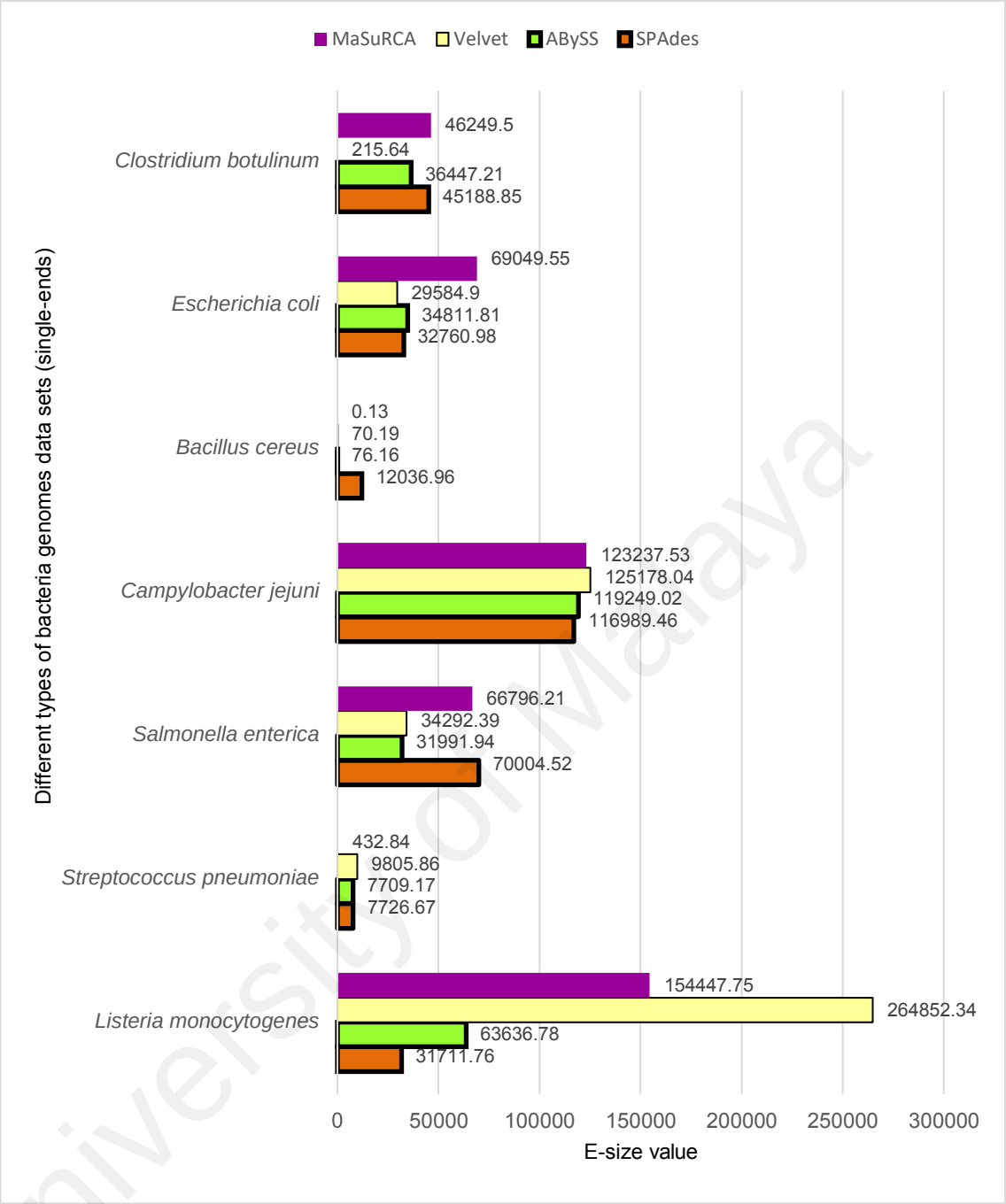


Figure 4.13: E-size value of contigs align for different bacteria genomes data sets (single-ends) using GAGE

Refer to the assembled contigs statistics result of all bacteria single-end; most of the data sets prefer to use MaSuRCA. However, GAGE results of bacteria single-end (Figure 4.13) showed differently when Velvet yielded the highest E-size value of all bacteria single-end. But, if look closely at *Campylobacter jejuni* point, the value of MaSuRCA and Velvet are high similar, MaSuRCA was 123237.53 while Velvet was 125178.04.

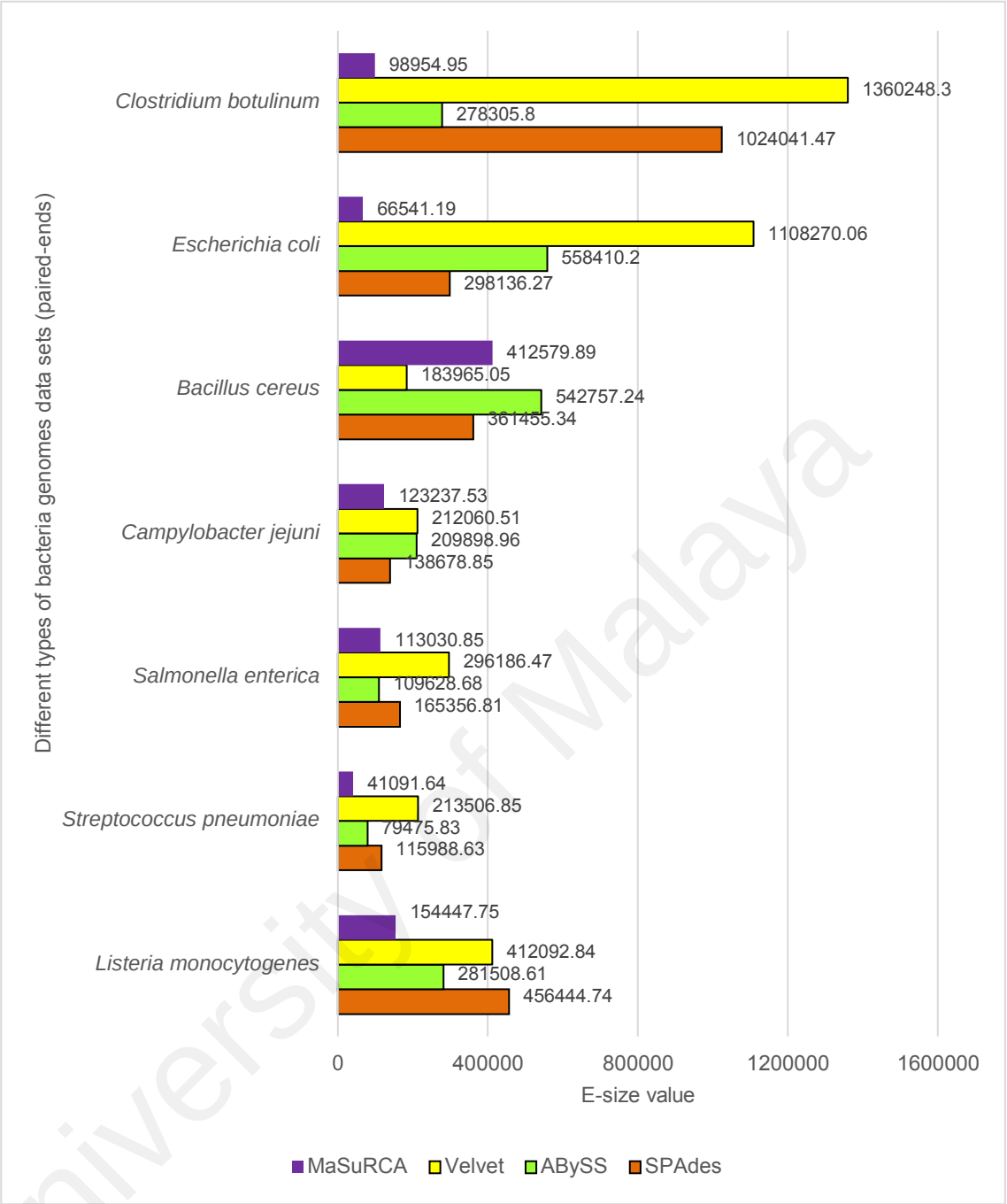


Figure 4.14: E-size value of contigs align for different bacteria genome data sets (paired-ends) using GAGE

Refer to the assembled contigs statistics result of all bacteria paired-end; most of the data sets prefer to use MaSuRCA and Velvet. However, the result occurred in GAGE in figure 4.14 showed only Velvet yielded the highest E-size value of all bacteria genomes (paired-end).

4.4.2 gVolante

Commonly, the performance of a *de novo* assembly is measured by different metrics, such as 'N50'. However, those length-based metrics are inaccurate, and cannot take into account their composition (the coverage of genes) and the accuracy of reconstructed contigs. Thus, gVolante provides an assessment referring to a set of pre-selected conserved genes (a complementary metric of completeness) taking the composition of given sequences (contigs) into account. Figures 4.15 and 4.16 showed the percentage of the bacterial genomic contigs completeness (based on core genes) for single-end and paired-end data sets graphs.

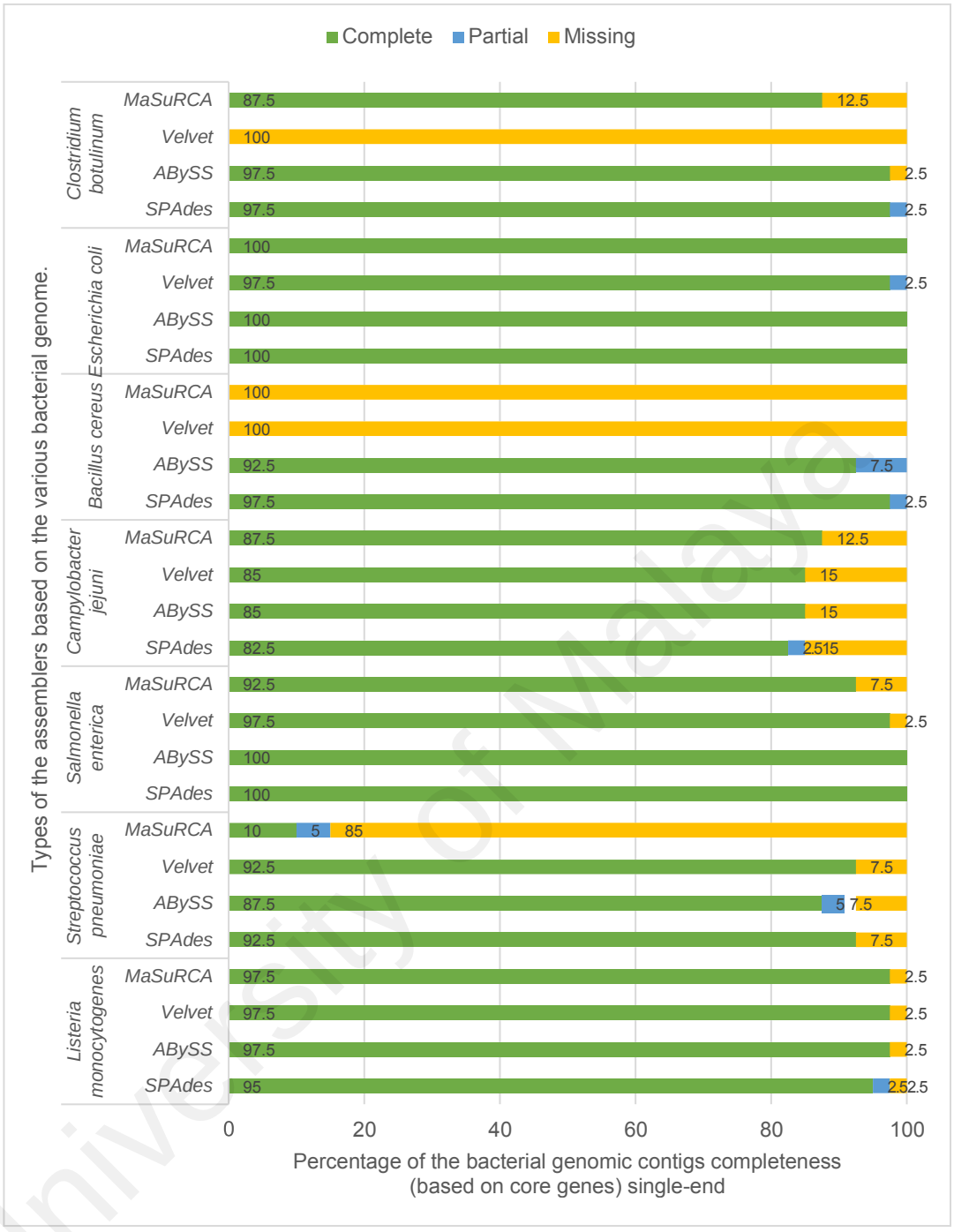


Figure 4.15: The graph of percentage of the bacterial genomic contigs completeness (based on core genes) single-end vs types of assemblers

Refer to the assembled contigs statistics result of all bacteria genomes single-end; most of the data sets prefer to use MaSuRCA while GAGE suggested MaSuRCA and Velvet. However, gVolante results of bacteria genomes single-end (Figure 4.15) showed differently when SPAdes yielded the highest percentage of the bacterial genomic contigs completeness (based on core genes) value of all bacteria single-end.

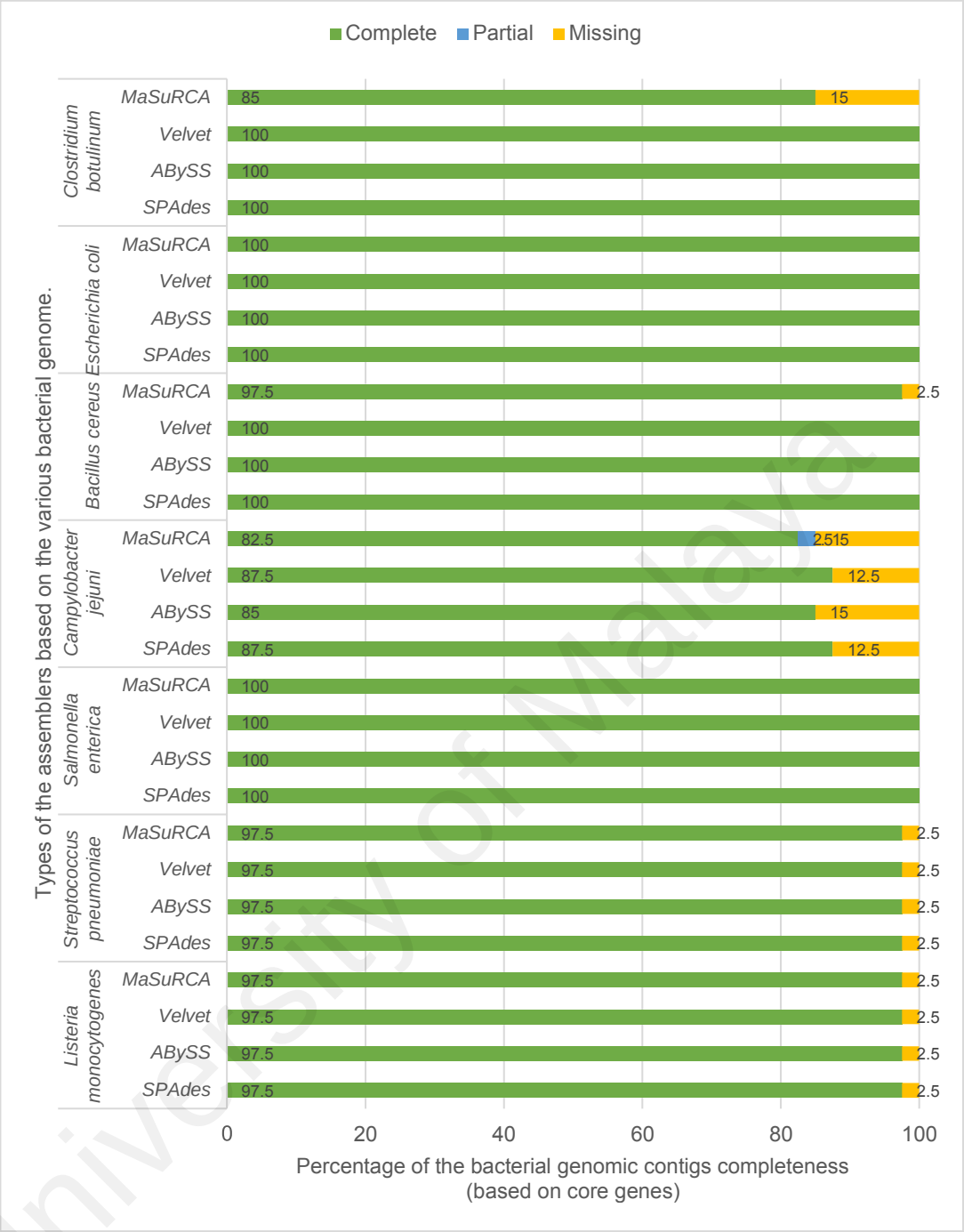


Figure 4.16: The graph of percentage of the bacterial genomic contigs completeness (based on core genes) paired-end vs types of assemblers

Refer to the Figure 4.16, Velvet produced the highest percentage of the bacterial genomics contigs completeness (based on core genes) for paired-end data. The similar result generated by the assembly metrics and GAGE.

CHAPTER 5: DISCUSSION

We evaluated four assemblers, SPAdes, ABySS, Velvet and MaSuRCA using seven types of bacteria single-end and paired-end Illumina-based short reads. All real data of bacteria had been downloaded from European Bionformatics Institute EMBL-EBI. Each data was running with different number of k- mer starting from 11 until 101 using three different assemblers (SPAdes, ABySS and Velvet) while MaSuRCA was automatics computing k-mer between 25 until 127. Several k-mer were applied because many assemblers are lacking robustness with respect to the parameters especially in choosing the suitable k-mer. In *de Bruijn*-based assemblers, the most significant parameter is k, because the k-value chosen for construction influences its structure. When a small k-value applies, the resulting graph would be complicated by spurious edges and vertices especially when it involves with repeats or duplication. It cannot distinguish these repeats or duplication that make the graph tangles and break-up the contigs. On the other hand, when k is too large the higher the chances that a k-mer will have an error in it that makes the graph becomes too sparse and possibly disconnected. Therefore, the choice of k represents a trade-off between several effects and should be chosen appropriately.

In the first analysis of computational performance regarding assemblers running time, the total assembling time was calculated and the differences on the total assembling time according to the types of assemblers was compared using Kruskal-Wallis test. This test is a non-parametric test to compares groups in order to determine if their time variances are different. From this analysis, for single-end reads data sets, there was no significant difference of total assembling time according to the types of assemblers. This is due the output p-value > 0.05 (Chi square = 5.141, p-value = 0.16, degree of freedom= 3), with a mean rank time (seconds) score of 20.00 for SPAdes, 11.71 for ABySS, 11.14 for Velvet

and 15.14 for MaSuRCA. For paired-end reads data sets, there was a statistically significant difference (Chi square = 11.390, p-value = 0.01, degree of freedom = 3), with a mean rank time (seconds) score of 20.86 for SPAdes, 12.86 for ABySS, 6.86 for Velvet and 17.43 for MaSuRCA. The result showed that Velvet consumed lowest time (in median) of 6.86 second while SPAdes consumed more time with 20.86 seconds compared to other assemblers.

Practically, an assembler which produces the highest percentage of contigs that were equal or longer than 500 bp, highest total base pairs, highest N50 and the lowest L50 are ideal (Abnizova et al., 2017). Although there are other metrics to be considered against these four metrics, some researchers and paperwork generally declared that these four criteria are the standard measure of the assembly connectivity. The results from the statistics for assembled contigs showed that MaSuRCA is suitable for bacteria single-end genomic data sets. It produced the highest percentage of contigs that were equal or longer than 500 bp, highest total base pairs, highest N50 and the lowest L50 for most assemblers. MaSuRCA is combined the efficiency and capability of the Overlap-Layout-Consensus (OLC) and the *de Bruijn* graph approaches. Furthermore, this tool has a script which can find optimal MaSuRCA parameters for a library of reads including the optimum k-mer numbers, pre-processing step and so forth. This result had been validated by GAGE, MaSuRCA and Velvet produced the highest E-size value compared to SPAdes and ABySS. The different output has been produced by gVolante when SPAdes yielded the highest percentage of the bacterial genomic contigs completeness (based on core genes) value of all bacteria single-end compared to MaSuRCA, ABySS and Velvet. This is due to the type of read used which is single-end read that have limitation information of relative positions that sometimes having difficulty in identifying gene insertions, deletions, or inversions, worst when involved repetitive regions. However, for this study,

SPAdes and MaSURCA still can be used to assemble bacterial single-end reads because MaSURCA produced the highest percentage of the bacterial genomic contigs completeness (based on core genes) value few bacteria such as *Escherichia coli, Campylobacter jejuni* and *Listeria monocytogenes.* In terms of the types of Illumina platforms that we studied for bacteria single-end data, SPAdes and MaSuRCA are suitable to assemble the datasets regardless of the size of reads from different types of Illumina technology platforms.

On the other hand, for bacteria paired-end reads, most of bacteria paired-end reads data sets were preferring to use MaSuRCA and Velvet for genome assembly based on the statistic. The results produced in GAGE showed Velvet yielded the highest E-size value of all bacteria paired-end contigs. gVolante was also generated the same results with GAGE when Velvet produced the highest percentage of the bacterial genomics contigs completeness (based on core genes) for paired-end data. Thus, Velvet were suitable to assemble the bacterial paired-end read datasets. This is due, paired-end give additional information in term of the directionality of the read, along with the length of the fragment from which the paired end reads were derived, in the assembly process.

The importance of this study can be reviewed by the impact of gene content in the selected organism that have gone through the genome assembly process. According to Florea et al. (2011), they found that inaccuracies in a genome assembly affect a large number of genes, although an extensive post-processing of the genome assembly have been run to improve the sequence, the consequences of assembly errors remain significant, with hundreds of genes left fragmented or incomplete. Furthermore, since there is no gold standard for genome assembly due to the complexity of sequencing generated data some practical considerations for *de novo* assembly, in which assembly

results must be taken several times using different assemblers with different parameter settings to determine their confidence (Bradnam, et al., 2013; El-Metwally et al., 2013). However, at least there is a guideline which appropriate assemblers are suitable for the selected data sets.

CHAPTER 6: CONCLUSION

Two types of analysis have been run in this study which are first, the computational performance in term of assemblers running time and second, assessment of assembly quality of four *de novo de Bruijn* graph sequence assemblers - SPAdes, ABySS, Velvet and MaSuRCA. These tools assembled seven types of bacteria genomic reads (single-end and paired-end) with different types of reads size and different types of Illumina technology platform. Each assembler is capable of assembling the whole bacterial genome sequences.

In the first analysis which is a computational performance in term of assemblers running time, for single-end reads data sets there was no significant difference of total assembling time according to the types of assemblers while for paired-end reads data sets, there was a statistically significant difference. The result showed that Velvet consumed lowest time (in mean) of 6.86 second while SPAdes consumed more time with 20.86 seconds compared to other assemblers. In the second analysis, four *de novo de Bruijn* graph sequence assemblers have been assessed and compared in terms of the assembly quality. On single-end data sets, MaSuRCA, and SPAdes produced generally the best results among all the four assemblers with highest percentage of contigs that were equal or longer than 500 bp, highest total base pairs, highest N50 and the lowest L50 for most assemblers. For paired-end data sets, Velvet are suitable to assemble all the seven bacteria genome sequences. From this study, we concluded that the selection of the best assembler is dependent on the uniqueness of the data sets and the user requirements.

Since genome assembly is the secondary analysis involved in processing and analyzing the NGS data, it will give impact on the tertiary analysis or more known as “genome

annotation and interpretation stage. In this stage, the genomic regions will be attached with biological meaningful and significance information by analyzing their sequence structure, composition and function. Thus, there is important to choose an appropriate assembler tools based on the uniqueness of the data sets. Other than that, this genome assembly study will advance the current knowledge of *de novo* genome assemblies especially the algorithm because the development of assembly algorithms is related closely to the development of sequencing technologies. Furthermore, we are currently towards to third-generation sequencing or long-read sequencing, that routinely generates reads in excess of 10 kb.

In the coming era where more researchers are able to run their own whole-genome sequence data because of the cost is much lower and availability of assembly software and hardware, this study will be further for integration with other biological studies (such as large-scale functional studies or evolutionary genomics investigation) or other omics studies such as metabolomics and proteomics.

REFERENCES

Abnizova, I., te Boekhorst, R., & Orlov, Y. (2017). Computational errors and biases of short read next generation sequencing. *Journal of Proteomics & Bioinformatics, 10*, 1-17.

Ahmadloo, S., Nakaoka, H., Hayano, T., Hosomichi, K., You, H., Utsuno, E., . . . Inoue, I. (2017). Rapid and cost-effective high-throughput sequencing for identification of germline mutations of BRCA1 and BRCA2. *Journal of Human Genetics, 62*, 561-567.

Alkan, C., Sajjadian, S., & Eichler, E. E. (2011). Limitations of next-generation genome sequence assembly. *Nature Methods, 8*(1), 61-65.

Alhakami, H., Mirebrahim, H. & Lonardi, S. (2017). A comparative evaluation of genome assembly reconciliation tools. *Genome Biology*. *18*, 1–14.

Ambardar, S., Gupta, R., Trakroo, D., Lal, R., & Vakhlu, J. (2016). High throughput sequencing: An overview of sequencing chemistry. *Indian Journal of Microbiology, 56*(4), 394-404.

Andrews, S. (2010). FastQC: A quality control tool for high throughput sequence data: Babraham Bioinformatics, Babraham Institute, Cambridge, United Kingdom.

Ansorge, W. J. (2009). Next-generation DNA sequencing techniques. *Nature Biotechnology, 25*(4), 195-203.

Baker, M. (2012). *De novo* genome assembly: What every biologist should know. *Nature Methods*. *9*, 333-337.

Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., . . . Pevzner, P. A. (2012). SPAdes: A new genome assembly algorithm and its applications to single-cell sequencing. *Journal of Computational Biology, 19*(5), 455-477.

Bao, E., Jiang, T., & Girke, T. (2014). AlignGraph: algorithm for secondary *de novo* genome assembly guided by closely related references. *Bioinformatics*, *30*(12), i319-i328.

Batzoglou, S., Jaffe, D. B., Stanley, K., Butler, J., Gnerre, S., Mauceli, E., . . . Lander, E. S. (2002). ARACHNE: A whole-genome shotgun assembler. *Genome Research, 12*(1), 177-189.

Benke, K., Ágg, B., Meienberg, J., Kopps, A. M., Fattorini, N., Stengl, R., . . . Mátyás, G. (2018). Hungarian Marfan family with large FBN1 deletion calls attention to copy number variation detection in the current NGS era. *Journal of Thoracic Disease, 10*(4), 2456-2460.

Bentley, D. R. (2006). Whole-genome re-sequencing. *Current Opinion in Genetics & Development, 16*(6), 545- 552.

Biscotti, M. A., Olmo, E., & Heslop-Harrison, J. S. (2015). Repetitive DNA in eukaryotic genomes. *Chromosome Research, 23*(3), 415-420.

Blawid, R., Silva, J. M. F., & Nagata, T. (2017). Discovering and sequencing new plant viral genomes by next-generation sequencing: description of a practical pipeline. *Annals of Applied Biology, 170*(3), 301-314.

Boisvert, S., Laviolette, F., & Corbeil, J. (2010). Ray: simultaneous assembly of reads from a mix of high-throughput sequencing technologies. *Journal of Computational Biology, 17*(11), 1519-1533.

Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics, 30*(15), 2114-2120.

Bradnam, K. R., Fass, J. N., Alexandrov, A., Baranay, P., Bechner, M., Birol, I., ... & Chitsaz, H. (2013). Assemblathon 2: evaluating *de novo* methods of genome assembly in three vertebrate species. *GigaScience*, *2*(1), 2047-217X.

Chaisson, M. J. P., Wilson, R. K., & Eichler, E. E. (2015). Genetic variation and the *de novo* assembly of human genomes. *Nature Reviews Genetics, 16*, Article#627.

Cheung, M. K., & Kwan, H. S. (2012). Fighting outbreaks with bacterial genomics: case review and workflow proposal. *Public Health Genomics, 15*(6), 341-351.

de la Bastide, M., & McCombie, W. R. (2007). Assembling genomic DNA sequences with PHRAP. *Current Protocols in Bioinformatics, 17*(1), 11-4.

De Wit, P., Pespeni, M. H., Ladner, J. T., Barshis, D. J., Seneca, F., Jaris, H., . . . Palumbi, S. R. (2012). The simple fool's guide to population genomics via RNA-Seq: An introduction to high-throughput sequencing data analysis. *Molecular Ecology Resources, 12*(6), 1058-1067.

Dohm, J. C., Lottaz, C., Borodina, T., & Himmelbauer, H. (2007). SHARCGS, a fast and highly accurate short-read assembly algorithm for *de novo* genomic sequencing. *Genome Research, 17*(11), 1697-1706.

Ekblom, R., & Wolf, J. B. W. (2014). A field guide to whole-genome sequencing, assembly and annotation. *Evolutionary Applications, 7*(9), 1026-1042.

El-Metwally, S., Hamza, T., Zakaria, M., & Helmy, M. (2013). Next-generation sequence assembly: Four stages of data processing and computational challenges. *PLOS Computational Biology, 9*(12), Article#e1003345.

Fakruddin, M., Chowdhury, A., Hossain, M. N., Mannan, K., & Mazumda, R. (2012). Pyrosequencing-principles and applications. *International Journal of Life Science and Pharma Research, 2*, 65-76.

Fiers, W., Contreras, R., Duerinck, F., Haegeman, G., Iserentant, D., Merregaert, J., . . . Ysebaert, M. (1976). Complete nucleotide sequence of bacteriophage MS2 RNA: primary and secondary structure of the replicase gene. *Nature, 260*(5551), 500-

507.

Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., Kirkness, E. F., Kerlavage, A. R., . . . Merrick, J. M. (1995). Whole-genome random sequencing and assembly of *Haemophilus influenzae Rd*. *Science*, 269(5223), 496-512.

Florea, L., Souvorov, A., Kalbfleisch, T. S., & Salzberg, S. L. (2011). Genome assembly has a major impact on gene content: a comparison of annotation in two *Bos taurus* assemblies. *PLOS One*, 6(6), Article#e21400.

George, D., & Mallery, P. (2016). *IBM SPSS statistics 23 step by step: A simple guide and reference*: Routledge.

Gioiosa, S., Bolis, M., Flati, T., Massini, A., Garattini, E., Chillemi, G., . . . Castrignanò, T. (2018). Massive NGS data analysis reveals hundreds of potential novel gene fusions in human cell lines. *GigaScience, 7*(10), Article#giy062.

Goldman, A. D., & Landweber, L. F. (2016). What is a genome? *PLOS Genetics, 12*(7), Article#e1006181.

Goodwin, S., McPherson, J. D., & McCombie, W. R. (2016). Coming of age: Ten years of next-generation sequencing technologies. *Nature Reviews Genetics, 17*, 333-351.

Granger G. Sutton, O. W., Mark D. Adams, and Anthony R. Kerlavage. (1995). TIGR assembler: A new tool for assembling large shotgun sequencing projects. *Genome Science & Technology, 1*(1), Article#19.

Hernandez, D., Francois, P., Farinelli, L., Osteras, M., & Schrenzel, J. (2008). *De novo* bacterial genome sequencing: millions of very short reads assembled on a desktop computer. *Genome Research, 18*(5), 802-809.

HGS-TIGR splits, opportunity knocks. (1997). *Nature Biotechnology, 15*(8), Article#693.

Hirakawa, H., Shirasawa, K., Isobe, S. N., Nagano, S., Yamaguchi, H., Sumitomo, K., . . . Koshioka, M. (2019). *De novo* whole-genome assembly in *Chrysanthemum seticuspe*, a model species of *Chrysanthemums*, and its application to genetic and gene discovery analysis.

Horner, D. S., Pavesi, G., Castrignano, T., De Meo, P. D., Liuni, S., Sammeth, M., . . . Pesole, G. (2010). Bioinformatics approaches for genomics and post genomics applications of next-generation sequencing. *Briefings in Bioinformatics, 11*(2), 181-197.

Jackman, S. D., & Birol, İ. (2010). Assembling genomes using short-read sequencing technology. *Genome Biology, 11*(1), Article#202.

Jeck, W. R., Reinhardt, J. A., Baltrus, D. A., Hickenbotham, M. T., Magrini, V., Mardis, E. R., . . . Jones, C. D. (2007). Extending assembly of short DNA sequences to handle error. *Bioinformatics, 23*(21), 2942-2944.

Jünemann, S., Sedlazeck, F. J., Prior, K., Albersmeier, A., John, U., Kalinowski, J., . . . Stoye, J. (2013). Updating benchtop sequencing performance comparison. *Nature Biotechnology, 31*(4), 294.

Kanterakis, A., Potamias, G., & Patrinos, G. P. (2018). Chapter 4 - An introduction to tools, databases, and practical guidelines for NGS data analysis. In C. G. Lambert, D. J. Baker, & G. P. Patrinos (Eds.), *Human Genome Informatics* (pp. 61-89): Academic Press.

Land, M., Hauser, L., Jun, S.-R., Nookaew, I., Leuze, M. R., Ahn, T.-H., . . . Ussery, D. W. (2015). Insights from 20 years of bacterial genome sequencing. *Functional & Integrative Genomics, 15*(2), 141-161.

Li, Li, Y., Zheng, H., Luo, R., Zhu, H., Li, Q., . . . Wang, J. (2009). Building the sequence map of the human pan-genome. *Nature Biotechnology, 28*, Article#57.

Li, M., Chen, L., Tian, S., Lin, Y., Tang, Q., Zhou, X., . . . Jin, L. (2017). Comprehensive variation discovery and recovery of missing sequence in the pig genome using multiple *de novo* assemblies. *Genome Research, 27*(5), 865-874.

Li, Z., Chen, Y., Mu, D., Yuan, J., Shi, Y., Zhang, H., . . . Fan, W. (2012). Comparison of the two major classes of assembly algorithms: overlap-layout-consensus and *de-bruijn*-graph. *Briefings in Functional Genomics, 11*(1), 25-37.

Liu, L., Li, Y., Li, S., Hu, N., He, Y., Pong, R., . . . Law, M. (2012). Comparison of Next-Generation Sequencing Systems. *Journal of Biomedicine and Biotechnology, 2012*, Article#11.

Loman, Constantinidou, C., Chan, J. Z. M., Halachev, M., Sergeant, M., Penn, C. W., . . . Pallen, M. J. (2012). High-throughput bacterial genome sequencing: an embarrassment of choice, a world of opportunity. *Nature Reviews Microbiology, 10*, Article#599.

Maretty, L., Jensen, J. M., Petersen, B., Sibbesen, J. A., Liu, S., Villesen, P., . . . Schierup, M. H. (2017). Sequencing and *de novo* assembly of 150 genomes from Denmark as a population reference. *Nature, 548*, Article#87.

Medvedev, P., Pham, S., Chaisson, M., Tesler, G., & Pevzner, P. (2011). Paired *de Bruijn* Graphs: A Novel Approach for Incorporating Mate Pair Information into Genome Assemblers. *Journal of Computational Biology, 18*(11), 1625-1634.

Metzker, M. L. (2010). Sequencing technologies - the next generation. *Nature Reviews Genetics, 11*(1), 31-46.

Miller, J. R., Delcher, A. L., Koren, S., Venter, E., Walenz, B. P., Brownley, A., . . . Sutton, G. (2008). Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics, 24*(24), 2818-2824.

Miller, J. R., Koren, S., & Sutton, G. (2010). Assembly algorithms for next-generation sequencing data. *Genomics, 95*(6), 315-327.

Moore, M. J., Dhingra, A., Soltis, P. S., Shaw, R., Farmerie, W. G., Folta, K. M., & Soltis, D. E. (2006). Rapid and accurate pyrosequencing of angiosperm plastid genomes. *BMC Plant Biology, 6*, Article#17.

Nagarajan, N. & Pop, M. (2010). Sequencing and genome assembly using next-generation technologies. In Computational Biology (pp. 1-17). Humana Press, Totowa, NJ.

Ni, P., Bhuiyan, A. A., Chen, J.-H., Li, J., Zhang, C., Zhao, S., . . . Li, K. (2018). *De novo* assembly of mitochondrial genomes provides insights into genetic diversity and molecular evolution in wild boars and domestic pigs. *Genetica*, 146(3), 277-285.

Nishimura, O., Hara, Y., & Kuraku, S. (2017). gVolante for standardizing completeness assessment of genome and transcriptome assemblies. *Bioinformatics, 33*(22), 3635-3637.

Ozsolak, F., & Milos, P. M. (2011). RNA sequencing: advances, challenges and opportunities. *Nature Reviews Genetics, 12*(2), 87-98.

Palmer, L. E., Dejori, M., Bolanos, R., & Fasulo, D. (2010). Improving *de novo* sequence assembly using machine learning and comparative genomics for overlap correction. *BMC Bioinformatics, 11*, 33.

Paszkiewicz, K., & Studholme, D. J. (2010). *De novo* assembly of short sequence reads. *Briefings in Bioinformatics, 11*(5), 457-472.

Pegadaraju, V., Nipper, R., Hulke, B., Qi, L., & Schultz, Q. (2013). *De novo* sequencing of sunflower genome for SNP discovery using RAD (Restriction site Associated DNA) approach. *BMC Genomics, 14*(1), Article#556.

Phillippy, A. M. (2017). New advances in sequence assembly: Cold Spring Harbor Lab.

Pop, M. (2009). Genome assembly reborn: Recent computational challenges. *Briefings in Bioinformatics, 10*(4), 354-366.

Pop, M., Phillippy, A., Delcher, A. L., & Salzberg, S. L. (2004). Comparative genome assembly. *Briefings in Bioinformatics, 5*(3), 237-248.

Pevzner, P. A., Tang, H., & Waterman, M. S. (2001). An Eulerian path approach to DNA fragment assembly. *Proceedings of the National Academy of Sciences*, *98*(17), 9748-9753.

Primrose, S. B., & Twyman, R. (2009). *Principles of Genome Analysis and Genomics*: Wiley.

Raza, K., & Ahmad, S. (2016). Principle, analysis, application and challenges of next-generation sequencing: A review.

Ronaghi, M. (2001). Pyrosequencing sheds light on DNA sequencing. *Genome Research, 11*(1), 3-11.

Salzberg, S. L., Phillippy, A. M., Zimin, A., Puiu, D., Magoc, T., Koren, S., . . . Yorke, J. A. (2012). GAGE: A critical evaluation of genome assemblies and assembly

algorithms. *Genome Research, 22*(3): 557–567.

Sanger, F., Nicklen, S., & Coulson, A. R. (1977). DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences, 74*(12), 5463.

Schatz, M. C., Maron, L. G., Stein, J. C., Wences, A. H., Gurtowski, J., Biggers, E., . . . McCombie, W. R. (2014). Whole genome *de novo* assemblies of three divergent strains of rice, *Oryza sativa*, document novel gene space of aus and indica. *Genome Biology, 15*(11), Article#506.

Shendure, J., Balasubramanian, S., Church, G. M., Gilbert, W., Rogers, J., Schloss, J. A., & Waterston, R. H. (2017). DNA sequencing at 40: Past, present and future. *Nature, 550*, 345-353

Shendure, J., & Ji, H. L. (2008). Next-generation DNA sequencing. *Nature Biotechnology, 26*(10), 1135-1145.

Simpson, & Pop, M. (2015). The theory and practice of genome sequence assembly. *Annual Review of Genomics and Human Genetics, 16*(1), 153-172.

Simpson, Wong, K., Jackman, S. D., Schein, J. E., Jones, S. J., & Birol, I. (2009). ABySS: A parallel assembler for short read sequence data. *Genome Research, 19*(6), 1117-1123.

Sohn, J.-i., & Nam, J.-W. (2016). The present and future of *de novo* whole-genome assembly. *Briefings in Bioinformatics, 19*(1), 23-40.

Treangen, T. J., & Salzberg, S. L. (2011). Repetitive DNA and next-generation sequencing: Computational challenges and solutions. *Nature Reviews Genetics, 13*, 36.

Utturkar, S. M., Klingeman, D. M., Hurt, R. A., & Brown, S. D. (2017). A case study into microbial genome assembly gap sequences and finishing strategies. *Frontiers in Microbiology, 8*, 1272.

van Dijk, E. L., Auger, H., Jaszczyszyn, Y., & Thermes, C. (2014). Ten years of next-generation sequencing technology. *Trends in Genetics, 30*(9), 418-426.

Verma, M., Kulshrestha, S., & Puri, A. (2017). Genome Sequencing. In J. M. Keith (Ed.), *Bioinformatics: Volume I: Data, Sequence Analysis, and Evolution* (pp. 3-33). New York, NY: Springer New York.

Vezzi, F., Cattonaro, F., & Policriti, A. (2011). e-RGA: enhanced reference guided assembly of complex genomes. *EMBnet. journal*, *17*(1), 46-54.

Warren, R. L., Sutton, G. G., Jones, S. J., & Holt, R. A. (2007). Assembling millions of short DNA sequences using SSAKE. *Bioinformatics, 23*(4), 500-501.

Zerbino, D. R., & Birney, E. (2008). Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research, 18*(5), 821-829.

Zimin, A. V., Marçais, G., Puiu, D., Roberts, M., Salzberg, S. L., & Yorke, J. A. (2013). The MaSuRCA genome assembler. *Bioinformatics, 29*(21), 2669-2677.