Silvia Madritsch, BSc

Helicobacter pylori Genomics

MASTER'S THESIS

to achieve the university degree of

Master of Science

Master's degree progamm: Biomedical Engineering

submitted to

Graz University of Technology

Supervisor

Dr. Gerhard Thallinger

Institute of Computational Biotechnology

Petersgasse 14, A - 8010 Graz

Institute of Neural Engineering

Stremayrgasse 16/IV, A - 8010 Graz

Graz, October 2017

Abstract

Helicobacter pylori is a gram-negative bacterium that is found in the human gastric mucosa in more than 50% of the world's population. It is a pathogen and infections can lead to chronic gastritis or gastric ulcers, but also a positive effect on asthma in children was observed. Since this bacteria has a high genomic variability it is important to sequence and assemble different *H. pylori* strains to determine intra- and intergenomic variability and the association with diseases.

This Master's thesis is divided into two main parts. First, different assembly tools were compared on a range of different bacterial sequencing data. In particular, the results of the popular GS De Novo Assembler (Newbler) were compared to the results of a previously published assembler benchmark. The performances of the assemblers were quite different depending on the genomic data. In general *MaSuRCA*, *Cabog* and *SPAdes* performed best while *SGA* and *Abyss* got the lowest scores. Newbler achieved good results especially in relation to the reference coverage, less overlapping bases and low rate of mismatches and indels. It obtained better results with Illumina MiSeq than with HiSeq data.

The second and main part of this thesis covers the assembly of four *Helicobacter pylori* strains. These strains were sequenced using Illumina and PacBio sequencing technologies. Assemblies were performed with two different long-read assembly strategies. The first one was a hybrid approach where the long PacBio reads are corrected by mapping of the short Illumina reads before used for assembly. But fully finished and closed genomes were only achieved with the second method, a so called stand-alone-assembly approach of Canu assembler followed by a circularisation step and a consensus building step using Illumina reads. It could be shown that *H. pylori* is a bacterium that has a high genomic variability, including large inversions of about 400 kbp, different copy numbers of the cagA region and a high amount of local variations affecting different genes.

Kurzfassung

Helicobacter pylori ist ein gram-negatives Bakterium welches sich in mehr als 50% der menschlichen Bevölkerung in der Magenschleimhaut befindet. Es ist ein Krankheitserreger und kann chronische Gastritis sowie Magengeschwüre auslösen, aber auch ein positiver Einfluss auf Asthma bei Kindern wurde beobachtet. Da das Bakterium eine hohe genetische Variabilität aufweist, ist es wichtig, unterschiedliche Stämme des Bakteriums zu sequenzieren und zu assemblieren um darauf Rückschlüsse auf Krankheiten und deren Verlauf zu gewinnen.

Diese Masterarbeit ist in zwei Bereiche unterteilt. Zuerst wurden verschiedene Assembly Tools an unterschiedlichen DNA-Sequenzen angewendet und verglichen. Im Speziellen wurden die Ergebnisse vom GS DE Novo Assembler (Newbler) mit den Ergebnissen einer früheren Publikation verglichen. Die Performances der Assembler waren unterschiedlich, abhängig von den verwendeten Daten. Im Allgemeinen haben MaSuRCA, Cabog und SPAdes die besten Ergebnisse geliefert, während SGA und Abyss eher schlechter abschnitten. Newbler hat gut funktioniert, im Speziellen erreichten die Assemblies eine hohe Referenz Coverage, wenige überlappenden Basen und eine geringen Rate an Mismatches und Indels. Newbler lieferte bessere Ergebnisse mit Illumina MiSeq als mit HiSeq Daten.

Der zweite Teil der Arbeit beinhaltet das Assemblieren von vier *H. pylori* Stämmen, welche mit der Sequenziertechnologie von Illumina und zusätzlich mit der von PacBio sequenziert wurden. Die Assemblies wurden mit zwei unterschiedlichen Strategien durchgeführt. Bei der Hybrid Technologie werden die langen PacBio DNA-Fragmente durch Mapping von Illumina Reads korrigiert. Vollständige und geschlossene Assemblies konnten aber nur mit der 2. Methode, einem sogenannten Stand-Alone-Assembly Ansatz mit Canu Assembler, gefolgt von einem Zirkularisierungsschritt und Konsensus-Bildung mit Hilfe der Illumina Reads, erzielt werden. Es wird gezeigt, dass *H. pylori* ein Bakterium mit einer hohen Variabilität ist, welches eine große Inversion von 400 kbp, unterschiedliche Anzahl von cagA Kopien und eine hohe Anzahl an lokalen genetischen Variationen aufweist.

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present Master's thesis.

Date	Signature

Acknowledgement

I would like to express deep gratitude to my supervisor Dr. Gerhard Thallinger for his guidance, encouragement and gracious support throughout my work. I further want to thank Dr. Sabine Kienesberger for providing the data for this thesis.

A big thanks to Dr. Martin Blaser from the NYU Langone Medical Center, USA who accepted me as an intern in his lab and highly supported me in my scientific work. I would also like to acknowledge Dr. Sandra Breum-Anderson of the Blaser's lab group at NYU for reading through my thesis, and I am gratefully indebted for her very valuable comments on this thesis.

Finally, I express gratitude to my parents for providing me with financial support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. Thank you!

Contents

1 Introduction

The aim of genome sequencing is to obtain a digital copy of the organism's DNA bases in the exact order they appear inside the target genome [1,2]. The genome sequencing process can be divided into four steps: At first sequencing, second assembly, third finishing and finally annotation.

First the genomic DNA is isolated from the organism and read by a sequencer. Since current sequencing technologies read at most a few thousand contiguous base pairs, the genome is broken up into small fragments, which are sequenced to yield the reads and through computational algorithms these reads are assembled into a genome [1]. The goal of the final annotation step is to identify functional regions of DNA in the genome [2].

1.1 Genome Sequencing

Until 2005 Sanger technology was solely used for DNA sequencing [3,4]. It creates reads with a length of approximately 800 bp and is very accurate but expensive and slow. To overcome these limitations, over the last years new sequencing technologies, called *next generation sequencing technologies*, were developed. These technologies are much faster and less expensive because they use a massive parallel processing and so it is possible to sequence millions of DNA fragments simultaneously [5]. They have the drawback that they produce shorter reads, which are less accurate making the assembly process much more complicated.

Today there are various DNA sequencers on the market and each sequence DNA in a different manner. NGS reads are in the range of 100 bp to 25,000 bp [6,7].

In the following section Illumina sequencing and PacBio sequencing is described as these technologies are used in this Master thesis.

1.1.1 Illumina Sequencing

Illumina sequencing technologies can sequence thousand of bases in parallel on a flow cell surface. As it is shown in Figure 1, DNA is randomly cut into fragments and adapters are ligated to both ends of the fragments. The DNA strands are denaturated and the single-stranded DNA fragments bind randomly on the surface

of the flow cell channels. On the flow cell a dense arrangement of primers are fixed and the free end of the fragments can now bind to these complementary primers (adapters). By adding nucleotides, polymerase and enzymes the strands are amplified and denaturated again. This process is repeated until million of clusters of DNA sequences are generated in each channel on the flow cell. Now the sequencing process can begin by adding labeled nucleotides with terminators. A laser detects the emitted fluorescence when the first bases of each cluster bind to the sequence. An optical scanner detects the signals from each fragment cluster. The terminator is removed and the sequencing process is repeated until every base of the fragment is sequenced. [8,9].

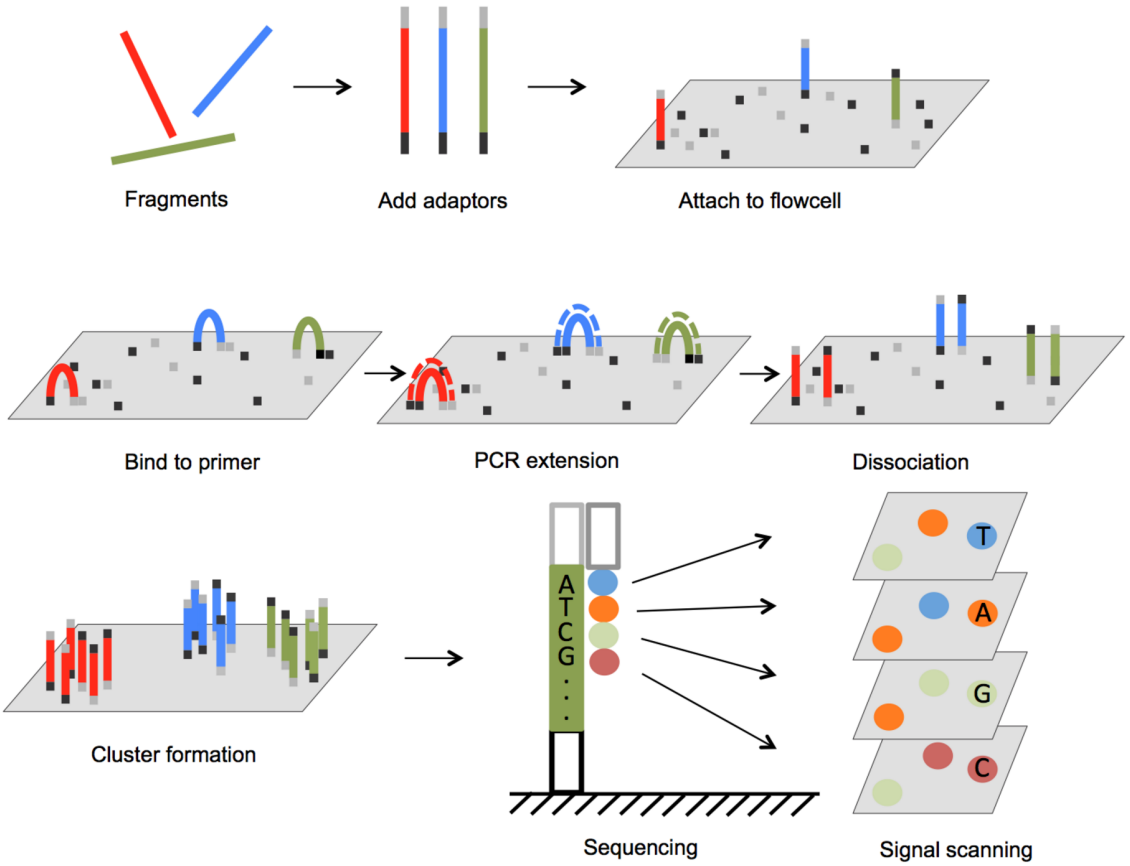


Figure 1: Illumina sequencing process. Adapters are annealed to the ends of the sequence fragments. Fragments bind to primer on the flow cell and build a bridge. Each fragment is amplificated and denaturated until clusters of fragments are produced. For sequencing, fluorescence labeled nucleotides are added to the growing strands. A laser detects the fluorescence signals from all the fragments and the first base of each cluster is sequenced. Then the sequencing terminator is removed and the next sequencing cycle starts (Image taken from [9]).

For the scaffolding process, see Section 1.2.2, the so called paired-end information is necessary. Mate-pair or paired-end information means that segments of DNA of known length are sequenced on both ends. For mate-pair sequencing special libraries are needed. The DNA is circulated, the joint is cut out, the ends are ligated and these ends are sequenced. The final libraries consist of short fragments made up of two DNA segments that were originally separated by several kilo-bases [10]. Paired-end sequencing uses fragments with less than 1kbp. Different adapters are placed at both ends. Sequencing begins at the forward end as described above and in a second step the reverse end is sequenced [11]. Illumina sequenced reads have a length between 150 and 250 bp and the error rate is less than 1% [6].

1.1.2 PacBio Sequencing

Pacific Biosciences [7] introduced Single Molecule Real Time DNA Sequencing (SMRT), a highly parallelized sequencing technology. It is based on two main key technologies, phospholinked nucleotides and zero-mode waveguides (ZMWs).

DNA sequencing is done on a chip that contains many visualisation chambers with ZMVs. These chambers allow visualisation on a single-fluorophore level. A polymerase is attached on the bottom of a ZMV and creates double stranded DNA from the single stranded template DNA (Figure 2). The nucleotides that are used in this process have a fluorescence label, distinguishable by the kind of nucleotide, linked at a phosphor end, which is cleaved off upon incorporation of the nucleotide and optically detected [12].

This technology is a high speed and long read sequencing technology. It produces reads in the length of up to 60 kbp with an average length up to 10 kbp but reads have an error rate of at least 15% [13,14].

1.1.3 Error Correction of PacBio Reads

Before PacBio reads can be used in genome assembly, they have to be corrected for sequencing errors. The expected error rate for insertions is about 10% and up to 5% for deletions [15]. Two main strategies exist, self-error correction or hybrid error correction. Self-error-correction methods are based on aligning the long reads against each other. Currently available tools are for example LoRMA [16] and Canu [17].

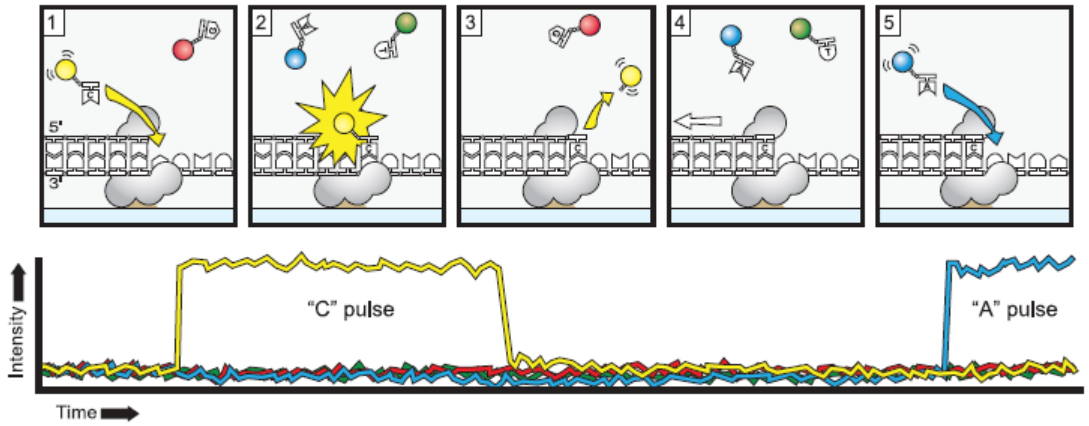


Figure 2: Principle of single-molecule, real-time DNA sequencing. (1) A phospholinked nucleotide bind to the complementary base on the template DNA strand in the polymerase active site. (2) This cause an increasing of the fluorescence on the corresponding color channel. (3) The fluorescent dye is cleaved off while creating a phosphodiester bond and diffuses out of the ZMW. (4) The polymerase moves to the next nucleotide on the template strand and (5) again the next phospholinked nucleotide binds (Image taken from [12]).

Because the aligning process is computationally expensive and a high PacBio read coverage is necessary and long read sequencing leads to high financial costs, hybrid long read error correction methods were developed. They use the much more accurate short read data from next generation sequencing technologies like Illumina to correct the long reads. Pairwise comparisons between long reads is avoided thereby. Most of the hybrid error correction tools like LSC [18], PacBioToCA [19] and proovread [15] rely on mapping short reads to long reads and computing the consensus sequence from the multiple alignment. Recently published error correction software like LoRDEC [20] and Jabba [21] build a de Bruijn graph, see Section 1.2.3, from the short reads and then map the reads on this graph [21]. These two methods achieve similar accuracy as other hybrid error correction methods but they have significantly improved run-times [20,21].

1.1.4 Proovread Error Correction

The proovread error correction pipeline corrects long reads by mapping short reads to the long reads. Therefore a special scoring model specifically for the distribution of PacBio sequencing errors was introduced for the alignment. The work-flow of proovread error correction is shown in Figure 3. It includes iterative pre-correction steps with increasing sensitivity in all three cycles. To improve runtime only a subsample of the short reads (20, 30 and 50%) is used for mapping at each iteration.

Regions with enough coverage (minimum per base coverage of five) are masked. With every cycle the sensitivity and the amount of used short reads are increased. The unmasked regions of the pre-processed long reads act as seeds for the next cycle. In each cycle the consensus is build to correct the reads. In the final cycle all short reads map to regions that are not masked yet at high specificity. In the end, the majority of errors are corrected and chimeric break points (wrong ligation of the read) identified. The resulting reads are then trimmed using a quality cutoff and the chimera annotation.

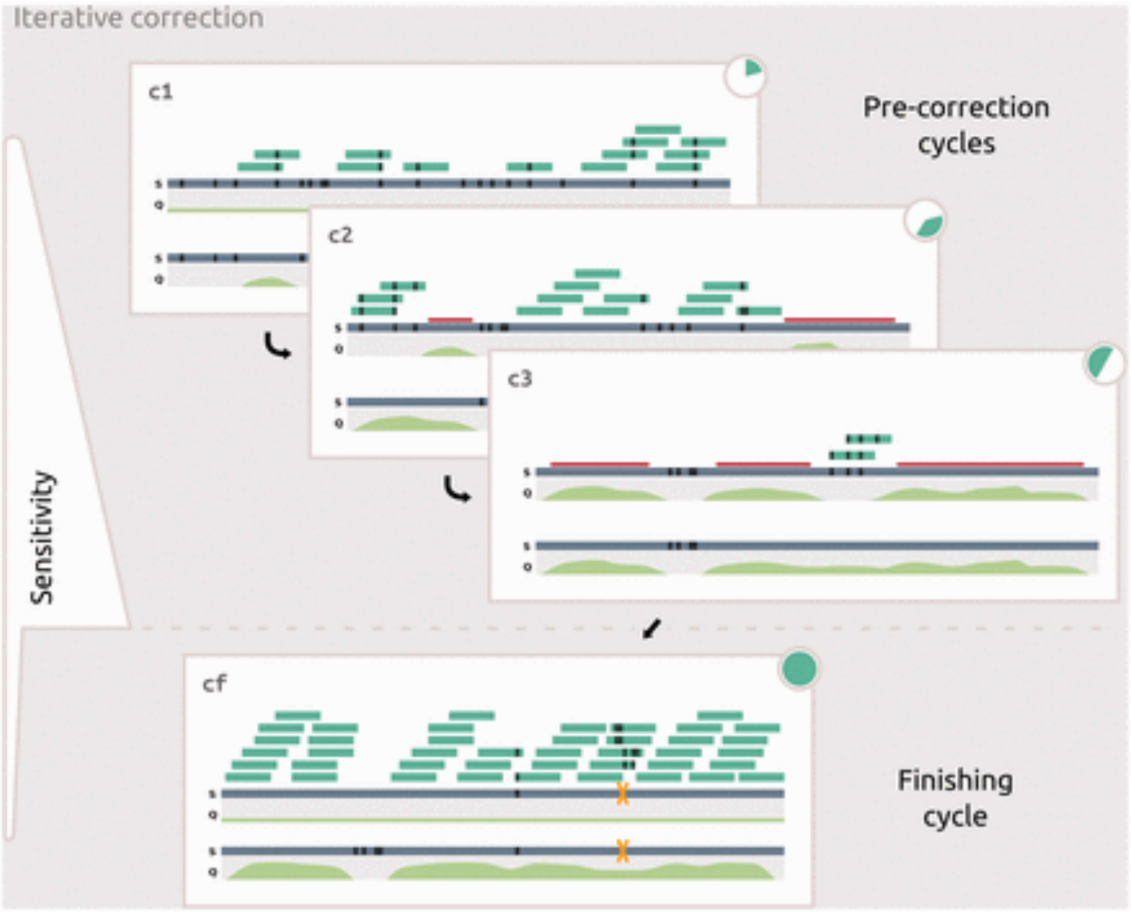


Figure 3: Principle of proovread's error correction pipeline. In the pre-correction cycles (c1-c3) a subsample of short reads (green bars) are mapped against a long read (blue bar). After each iteration the consensus is build. Regions with enough short read coverage are masked (red line). In the final finishing cycle all short reads are mapped onto unmasked preprocessed long reads at high specificity (Image taken from [15]).

1.1.5 Canu Self-Error-Correction

Canu is a recently published hierarchical single-molecule sequence assembler. It does not require a second short-read set. It uses multiple rounds of read overlapping and error correction prior to graph construction and assembly. It selects the best overlaps for correction, estimates the corrected read lengths and generate the corrected reads by building the consensus [17].

1.2 Genome Assembly

With the introduction of next generation sequencing technologies, the focus of sequencing has shifted from data generation to data processing. The assembly and especially the finishing of the short reads to reconstruct the whole genome sequence can be very complicated and time consuming.

1.2.1 Challenges in Genome Assembly

One main challenge in the assembly of genomes are repeat sequences. These lead to reads mapping to multiple locations in the target genome. Repeats that are longer than the read length create gaps in the assembly. Since the read length from NGS is shorter than from Sanger sequencing an assembly is much more fragmented [22].

A possible solution to the repeat problem is to sequence both sides of a longer fragment. The insert size of this fragment is known so the distance between the two sequenced reads can be approximated. Such reads are called mate-paired or paired-end, depending on the library preparation, and are essential for the assembly process [23], see Section 1.1.

1.2.2 Assembly and Scaffolding Process

The first step of the assembly process is grouping the reads that have overlapping regions into longer contigs. In the following scaffolding step these contigs are ordered, orientated as well as the sizes of the gaps between these contigs are computed [24]. For this step the assembler needs the paired-end information of the reads. A schematic sketch of an assembly process is shown in Figure 4.

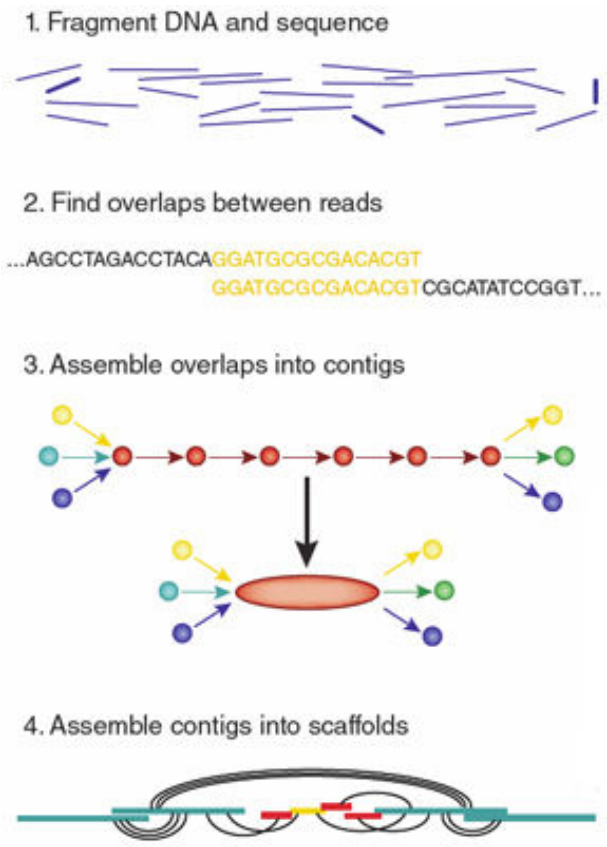


Figure 4: Genome assembly process. Small sequenced DNA fragments (1) are aligned to each other to find overlaps (2). Based on overlaps, reads are combined together to form contigs (3). The assembly of contigs into larger scaffolds is based on the paired-end information of short-reads (4) (Image taken from [25]).

1.2.3 OLC versus de Bruijn Graph Assembler Algorithms

Today all common *de novo* assemblers are based on two main methods: the *Overlap Layout Consensus (OLC)* assembly algorithm and the *de Bruijn Graph* algorithms. Both have in common that they are based on graphs.

Graphs are widely used in computer science. They consist of nodes and edges which connect them. A path is a way that visit nodes in a specific order. In the graph of the OLC algorithm nodes represent the reads and edges represent overlaps between the reads. Paths through the graph are putative contigs [24].

The OLC algorithm starts by computing and building the overlap graph that represents the sequencing reads and their overlaps [26]. It involves all-against-all, pair-wise read alignment. Then the graph is compressed, manipulated and finally the consensus sequence is determined based on the graph generated in the previous two steps [24].

The second method is based on the *de Bruijn Graph* algorithm. It was first developed to represent strings from a finite alphabet [24]. Concerning DNA sequencing nodes represent fix-length subsequences with length k of a read, called k-mer, and the edges represent all the fixed length consecutive overlaps between these subsequences, usually with length k-1. The differences of the two main assembly algorithms are illustrated in Figure 5. The advantage of *de Bruijn Graph* algorithms are they do not have to compute pairwise overlaps and efficient algorithms exist for computing the path through the graph (Eulerian path)[5].

1.2.4 Newbler Assembler

The *GS de novo Assembler* also called *Newbler* is a widely used assembly software distributed by 454 Life Sciences [28]. Newbler is an OLC like algorithm with two OLC cycles. In the first pass it generates so called unitigs, which are small contigs that do not have overlaps with other unitigs [26]. In the second OLC run unitigs are joined to larger contigs based on pair-wise overlaps between unitigs. It could happen that unitigs are split and its prefix and suffix align to different contigs leading to reads placed in multiple contigs. Such reads can be chimera or they are derived from a repeat region [26].

In contrast to other assemblers, *Newbler's* source code is not publicly available [24].

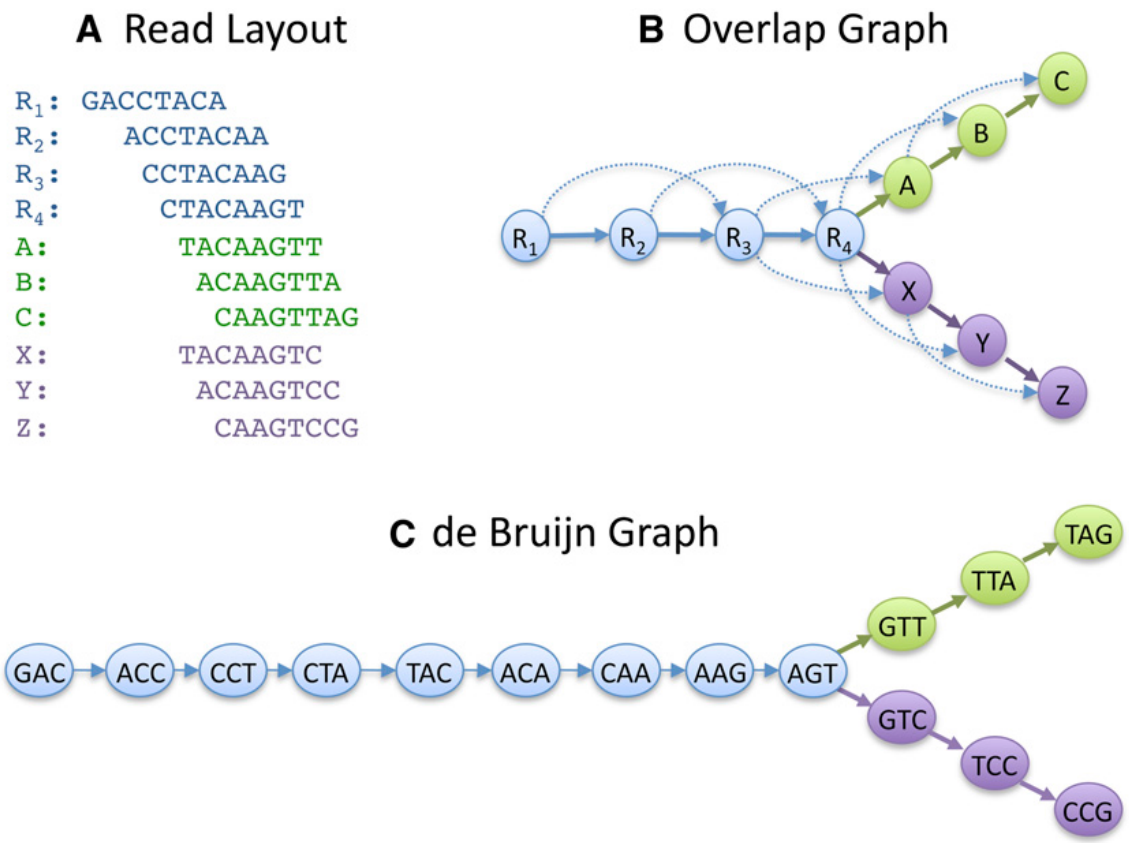


Figure 5: Differences between the OLC and the de Bruijn Graph algorithm.
A: A set of reads is represented. B: based on the reads in A an overlap graph can be build where each read is a node and overlaps with more than 5 base pairs are indicated by edges. C: In a de Bruijn graph, the nodes represent every k-mer in all the reads. Here k has size 3. Edges are drawn where the k-mer overlap by k-1 bases (Image taken from [27]). Such forks that are illustrated in green and violet may indicate a repeat region where the "blue" contig exists two times in the genome.

1.2.5 Canu Assembler

Canu assembler was specifically introduced to assemble the long PacBio reads. The Canu assembly pipeline includes three steps: correction, trimming and assembly. Each step can be performed independently, for example only read error correction or assembly without correction [17].

Canu uses a variant of the greedy best overlap graph (BOG) algorithm from Miller et al. [29] to correct reads and build the assembly. The overlap error rate is defined as the edit distance [30] (minimum number of operations need to transform one string into another) divided by the length of the overlap. Overlaps are filtered to include only overlaps that are within a tolerance of the global median error rate. The longest overlaps are then recomputed with the new subset.

The greedy algorithm can lead to mis-assemblies caused by repeats that are longer than the overlap length. Canu's new "Bogart" algorithm can filter repeat-induced overlaps and inspect the graph for potential errors retrospectively [17].

1.2.6 Performance Comparison of Different Assembly Tools

Evaluating the different genome assemblies can be very challenging, especially when no finished reference genome is available. Recently, two papers have been published that compare the performance of different assemblers on a range of bacterial genomes. One is GAGE-B published by Magoc et al. [31] and the other is GABenchToB published by Jünemann et al. [32]. Evaluating the different assembly results is a complex problem and there does not exist a single parameter to determine the best assembly software. Important metrics to measure the quality of an assembly include the N50 value, the number of mis-assemblies or the total run time. Recently, QUAST, a quality assessment tool for evaluating and comparing genome assemblies was introduced [33]. QUAST can evaluate assemblies both with a reference genome, as well as without a reference. In the studies of GAGE-B [31] and GaBenchToB [32] QUAST was used to measure assembly contiguity and accuracy.

1.3 *Helicobacter pylori*

Helicobacter pylori is a gram-negative bacterium that resides in the epithelium of the human stomach in more than 50 % of the world's population. It stimulates immune and inflammatory cells, which leads to chronic gastritis. Infections generally occur in children but also ulcers and gastric cancer due to a *Helicobacter* infection has been recorded. In these severe cases principally adults are affected [34].

It is of great interest to determine if bacterial, host or environmental factors can influence the disease. A recent study investigated the influence of *H. pylori* on the host's microbiota and immunity [35].

A lot of different *Helicobacter pylori* strains have been isolated from patients. Two main morphotypes of the bacteria exists, bacillary and coccoid. It is believed that the bacillary form is the virulent morphology. In general bacteria cause disease through three different processes: adhesion, invasion or toxin elaboration [34]. *H. pylori* infection is limited to the lumen and causes damage to the gastric mucosa. It has the ability to induce vacuolization in the membrane of epithelial cells [36]. Two major toxins have been purified and studied. The vacuolating toxin VacA and the cytotoxicity-associated immunodominant antigen CagA. CagA is often coexpressed with VacA. CagA has been associated with duodenal ulcers and gastric cancer while VacA can cause epithelial cell damage and gastritis in mice. [34,37,38].

However, there is still a significant lack of understanding of the mechanisms *H. pylori* uses to cause disease. It was also shown that *H. pylori* can have early-life benefits. An positive effect on diseases like asthma, gastro intestinal and systemic infections have been observed [39].

Strain PMSS1 is often used for studies in mice because it is one of the few strains of *H. pylori* that stably infect mice and express the virulence factors vacA and cagA. A patient biopsy was plated, and subcultured. The original strain from the patient is called PMSS1 (pre mouse), the strain after mouse passage SS1 [40].

1.4 Aims of the Thesis

The overall goal of this Master's Thesis is to assemble four strains of the *Helicobacter pylori* genome (SS1, PMSS1, PM21, PM22) where Illumina HiSeq data and PacBio reads are available.

To this end, the following should be achieved:

- alignment of all assembly results from the GAGE-B study as well as the result of Newbler assembler to a standard reference genome available at NCBI [41] to compare important assembly quality parameters like coverage, mismatches or overlapping regions
- alignment of finished contigs of each assembly to Newbler assembly and analysis of differences and similarities such as the amount of identical contigs in both assembly results.
- a general evaluation of the different assembler, especially the Newbler assembler, performed on the GAGE-B data and comparison with the results in the GAGE-B as well as the GaBenchToB paper.
- computation of assemblies of four *Helicobater pylori* strains with a stand-alone assembly tool (self-error-correction of PacBio reads) and with a hybrid approach (correction of PacBio reads using Illumina reads).
- comparison of these two concepts based on the available sequence data
- annotation and alignments of the assemblies against each other and against a reference
- detection of the number of SNPs, affected genes, synonym, non-synonym mutations, hot spots and structural variations
- visualisation of the phylogeny based on the SNPs as a network.

2 Methods

2.1 Assembly Tools for Small Genomes

2.1.1 Data

All the sequence data as well as the finished contigs to analyse the different assembly tools were taken from GAGE-B study [31]. They used Illumina sequence data of

Table 1: The bacteria used in GAGE-B study [31].

Name	Accession	Source	Size (Mbp)	GC (%)	Platform	Read Length
Aeromonas hydrophila	SRR488186	SRA [42]	4.7	65	HiSeq	101
Bacillus cereus	-	Illumina website [43]	5.4	35	MiSeq	250
Bacillus cereus	SRR497464	SRA [42]	5.4	35	HiSeq	101
Bacteroides fragilis	SRR488170	SRA [42]	5.3	43	HiSeq	101
Rhodobacter sphaeroides	SRR522244	SRA [42]	4.6	69	HiSeq	101
Rhodobacter sphaeroides	SRR522246	SRA [42]	4.6	69	MiSeq	251
Staphylococcus aureus	SRR569301	SRA [42]	2.9	33	HiSeq	101
Xanthomonas axonopodis	SRR522415	SRA [42]	2.9	33	HiSeq	101
Mycobacterium abscessus	SRA043447	U. of Maryland [44]	5.1	64	MiSeq	250
Mycobacterium abscessus	SRA043447	U. of Maryland [44]	5.1	64	HiSeq	100
Vibrio cholerae	SRA037376	U. of Maryland [44]	4.0	48	MiSeq	250
Vibrio cholerae	SRA037376	U. of Maryland [44]	4.0	48	HiSeq	100

eight bacteria (Table 1). The genome size ranges from 2.9 to 5.4Mb and the GC content from 33 to 69%. HiSeq and MiSeq data sets were included to compare these technologies. To achieve the same quality of all the data Magoc et al. [31] ran a common set of data cleaning steps for all datasets. They removed adapter sequences and performed Q10 quality trimming using the *ea-utils* package from Aronesty [45]. All the finished assemblies (contigs and scaffolds) as well as the sequence data of the bacteria can be downloaded from the GAGE-B website [46]. They compared eight different assemblers that are presented in Section 2.3 in the GAGE-B paper [31].

Table 2: The assemblers used in GAGE-B study [31] including the GS De Novo Assembler.

Name	Version	Type	Author	Ref.
Abyss	1.3.4	DBG	Simpson et al. 2009	[47]
CABOG	7.0	OLC	Miller et al. 2008	[29]
Mira	3.4.0	OLC	Chevreux et al. 2004	[48]
MaSuRCA	1.8.3	OLC & DBG	Zimin et al. 2013	[49]
SGA	0.9.34	String Graph	Simpson and Durbin 2012	[50]
SoapDenovo2	2.04	DBG	Luo et al. 2012	[51]
SPAdes	2.3.0	DBG	Bankevich et al. 2012	[52]
Velvet	1.2.08	DBG	Zerbino and Birney 2008	[53]
GS De Novo Assembler	2.9	OLC	454 Life Sciences	[28]

2.1.2 Newbler

The GS De Novo Assembler (Newbler) software, Version 2.9, was downloaded from Roche/454 website [28] and installed. To compare the performance of Newbler assembler with the other assemblies from the study, Newbler assembler was invoked with all the trimmed sequence data from GAGE-B [31] using default parameter and a minimum contig length of 1 (`runAssembly -o output_dir -a 1 seq_data.fasta`).

2.1.3 MUMmer Package

MUMmer is an open source software package for the rapid alignment of very large DNA and amino acid sequences. Nucmer is part of the MUMmer package and allows DNA alignment of multiple closely related nucleotide sequences [54]. It starts by finding maximal exact matches of a given length. Then it clusters these matches to larger alignment regions. Finally, it extends alignments outward from each of the matches to join the clusters into a single high scoring pair-wise alignment [54]. To include a high rate of possible alignments, the minimum cluster length was set to 50. For all the other parameters default values were used. The results of Newbler were used as *Query* file and the other assemblies respectively the reference genome from NCBI as *Reference*.

2.2 *Helicobacter pylori* Assemblies

2.2.1 Study Description

Illumina HiSeq and PacBio sequence data was provided by Dr S. Kienesberger, University of Graz [35]. Kienesberger et al. studied the interactions of *H. pylori* with mouse hosts over 6 months. They analysed gastric and pulmonary tissues and investigated an increase in the expression of multiple immune response genes over time in the stomach and in the lungs. Moreover *H. pylori* infection led to significant differences in both the gastric and intestinal microbiota [35]. PMSS1 is a *Helicobacter pylori* strain taken from the stomach of a 42 year-old Greek-born female in Sydney in 1997 [55] and a mouse was infected with this strain. The resulting SS1 strain was reisolated after infection and became a standardized mouse model for compound screening, and studies in pathogenesis. Both strains are publicly available

for research purposes [55]. In the study of Kienesberger and colleagues [35], 3 weeks and 5 weeks old mice were inoculated with strain PMSS1 and strains were isolated every month after infection. SK represents the name of mice group challenged at 4 weeks of age, PM the mice group challenged at 6 weeks of age. All 46 available Illumina sequenced strains are shown in Figure 6. PM21 and PM22 are both isolated after 6 months presence in the mouse.

2.2.2 Data

PacBio read data is available of the *Helicobacter pylori* strains PMSS1 (09), PM21 (45), PM22 (48) and SS1 (01). It has to be mentioned that the PacBio sequenced strain PMSS1 (09) is not from the same colony with whom the mice were inoculated at NYU. Colonies with the number 13 and 14 were used to inoculate the mice (Figure 6). Illumina reads are available for all strains shown in Figure 6. They have a read length of 151 bp and an insert size of 200 bp. As reference genome the NCBI genome assembly PMSS1 with GenBank accession CP018823.1 and the plasmid pHPYLPMSS1 CP018824.1 was used. For comparison of strain SS1 the references CP009259.1 and CP009260.1 was used [40].

2.2.3 Hybrid Approach

The software and tools used for hybrid assembly are shown in Table 3.

Table 3: Software used for hybrid assembly

Software	Version	Usage	Ref.
LoRDEC	0.6	long read error correction	[20]
Jabba		long read error correction	[21]
Proovread	2.13.13	long read error correction	[15]
Cutadapt	1.13	adapter trimming illumina reads	[56]
Trimmomatic	0.36	trimming illumina reads	[57]
Newbler	2.9	assembly	[28]
Canu	1.4	assembly	[17]
MUMmer	3.1	alignment of multiple nucleotide seq.	[54]

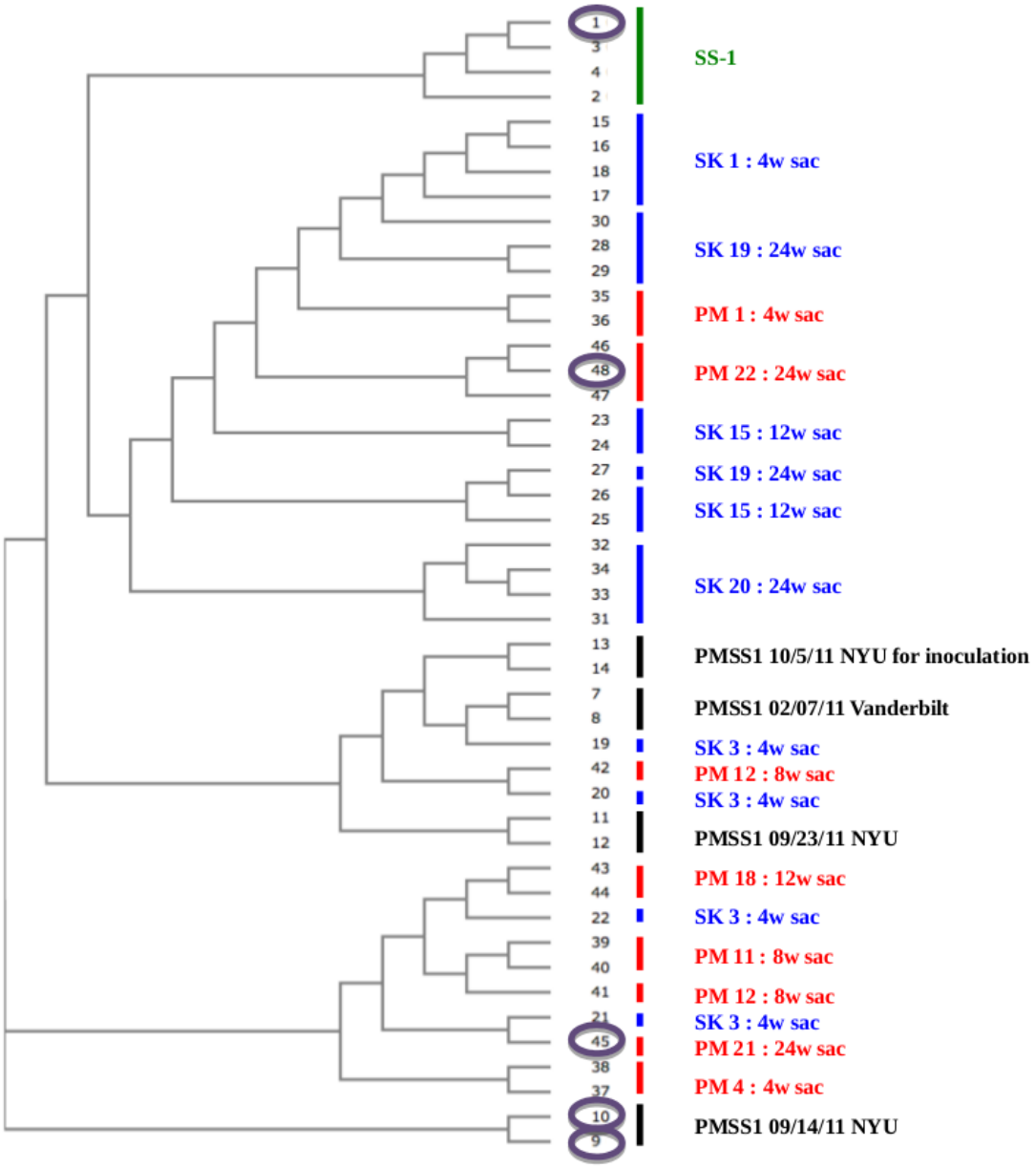


Figure 6: Phylogeny of all available *H. pylori* strains at NYU, including information how long they were present in the mice. W sac denotes the length of time after infection when mice were sacrificed. (provided by Tadasu Iizumi, NYU & NMS based on a mapping of the Illumina data to *H. pylori* 26695 as the reference).

2.2.4 Stand-Alone Approach with Canu Assembler

All software used for the assembly of the stand-alone approach is listed in Table 4.

Table 4: Software used for stand-alone assembly and finishing.

Software	Version	Usage	Ref.
Canu	1.4	assembly	[17]
BWA	0.7.12	reads alignment	[58]
samtools	1.3.1	sort and index sam,bam files	[59]
Circlator	1.4.1	circularisation of assembly	[14]
SPAdes	3.10.0	assembly included in Circlator	[52]
BLAST		alignment	[60]
GATK	3.7	consensus building	[61]
Picard	2.9.0	create dictionary of sequence	[62]
IGV	2.3.91	alignment viewer	[63]
Qualimap	2.2.1	alignment statistics	[64]
Prodigal	2.6.3	protein prediction	[65]
Trimmomatic	0.36	trimming Illumina reads	[57]
BASys		annotation	[66]

2.3 Comparison of the *H. pylori* Genomes

Tools that were used to compare the assembled genomes are shown in Table 5.

Table 5: Software used for the comparison of the genomes.

Software	Version	Usage	Ref.
Mauve	2.4.0	genome alignment, annotation viewer	[67]
snpEff	4.3	variant file annotaion	[68]
RaxML	8.2.11	creating phylogeny	[69]

2.3.1 Creating a Phylogenetic Tree

To create a phylogeny of all the six genomes the output file of the SNPs produced by whole genome alignment with Mauve was converted to an aligned fasta file containing all the SNPs with R, transformed to a special phylip file format using a published perl script and RaxML [69] invoked. RAxML creates a phylogenetic tree using maximum

likelihood methods based on the SNPs. Ten parsimony random seeds and the GTR-Γ model was chosen, (`raxmlHPC -s align_snps.phyl -n out_dir -m GTRGAMMA -p 10`). The tree in the newick file format could be read and plotted in R using the `ape` package [69]. The pairwise distances between the tips of the phylogenetic tree could be computed with the function `cophenetic.phylo` from the `ape` package and the `mst` function was invoked to compute the minimum spanning tree (MST).

2.4 Used R Packages

R is a free software environment for statistical computing and graphics. R-Version 3.2.3 was downloaded and installed [70]. R packages that are used to evaluate different assembly tools of the GAGE-B study are shown in Table 6, those packages for assembly and evaluation of the *H. pylori* strains in Table 7. All the implemented R-functions are described in Section 6.1.

Table 6: R packages that are used to evaluate different assembly tools of GAGE-B study.

Package Name	Version	Comment	Ref.
Biostrings	2.38.4	`IRanges`, `findOverlaps`, ...	[71]
stringr	1.0.0	`str_length` to get sequence lengths `str_locate_all` to find pattern ">" in delta file	[72]
seqinr	3.1-5	`read.fasta` and `write.fasta`	[73]
plyr	1.8.3	`count` number of contigs	[74]
stargazer	5.1	create LATEX code for Tables	[75]

Table 7: R packages that are used to assemble and evaluate *Helicobacter pylori* strains.

Package Name	Version	Comment	Ref.
Biostrings	2.38.4	`XStringSet`, `substring`, ...	[71]
stringr	1.2.0	`str_length`, `str_pad` to format indices	[72]
seqinr	3.3-3	`read.fasta` and `write.fasta`	[73]
ShortRead	1.28.0	FASTQ input and manipulation	[76]
stargazer	5.1	create LATEX code for Tables	[75]
VariantAnnotation	1.16.4	read in a vcf-file	[77]
ape	4.1	read in newick tree file format, `mst`	[78]

3 Results

3.1 Assembly Tools for Small Genomes

3.1.1 R Script to Align and Compare Assemblies

To compare the different assemblies an appropriate R script was implemented. The corresponding flow-diagram is shown in Figure 7.

FASTA files, containing the contigs of the assembly results, are read using the function `read.fasta()` from the package *seqinr*. Also the reference files from the NCBI [41] website are imported. The contigs are sorted by length and assigned numbers (Function `sort_contigs()` in Section 6.1).

Some general parameters of the assemblies are computed directly from these files. These includes the total Contig Number, the Total Length of the assembly, Minimum and Maximum Contig Length, N50 value, the Number of Identical Contigs (more than 99,5% matches in the longer sequence) and the number of contigs, which are shorter than 200 base pairs. The N50 value of a set of contigs is the size of the largest contig for which half the total size is contained in that contigs and those larger.

To compare the assemblies among each other and with the finished reference genomes respectively, *Nucmer* is invoked with parameters `nucmer -maxmatch -l 15 -c 50 <Referencefile.fasta> <QueryFile.fasta>` where `-maxmatch` uses all anchor matches regardless of their uniqueness. The minimum length of a maximal exact match was set to 15 and the minimum cluster length was reduced to 50 (default 65) to get also alignment results matching over a short distance. The output of *nucmer* is a delta file [54] including a header and all matches with information about the names of the query and reference contigs, the length of these contigs, start and the end positions as well as the number of errors including mismatches and indels.

With `read.table()` this delta file is read and sorted in a data fame, see function `create_table_of_delta()`.

The nucmer result table is filtered as follows: Alignment matches, where the Reference region in that contig as well as the Query region is fully covered by other matches are excluded, see function `remove_overlaps(table)`. In the next step a detailed table of the alignment is created, function `detail_table()`. It performs a detailed analysis on each contig compared to all reference contigs. For the computation

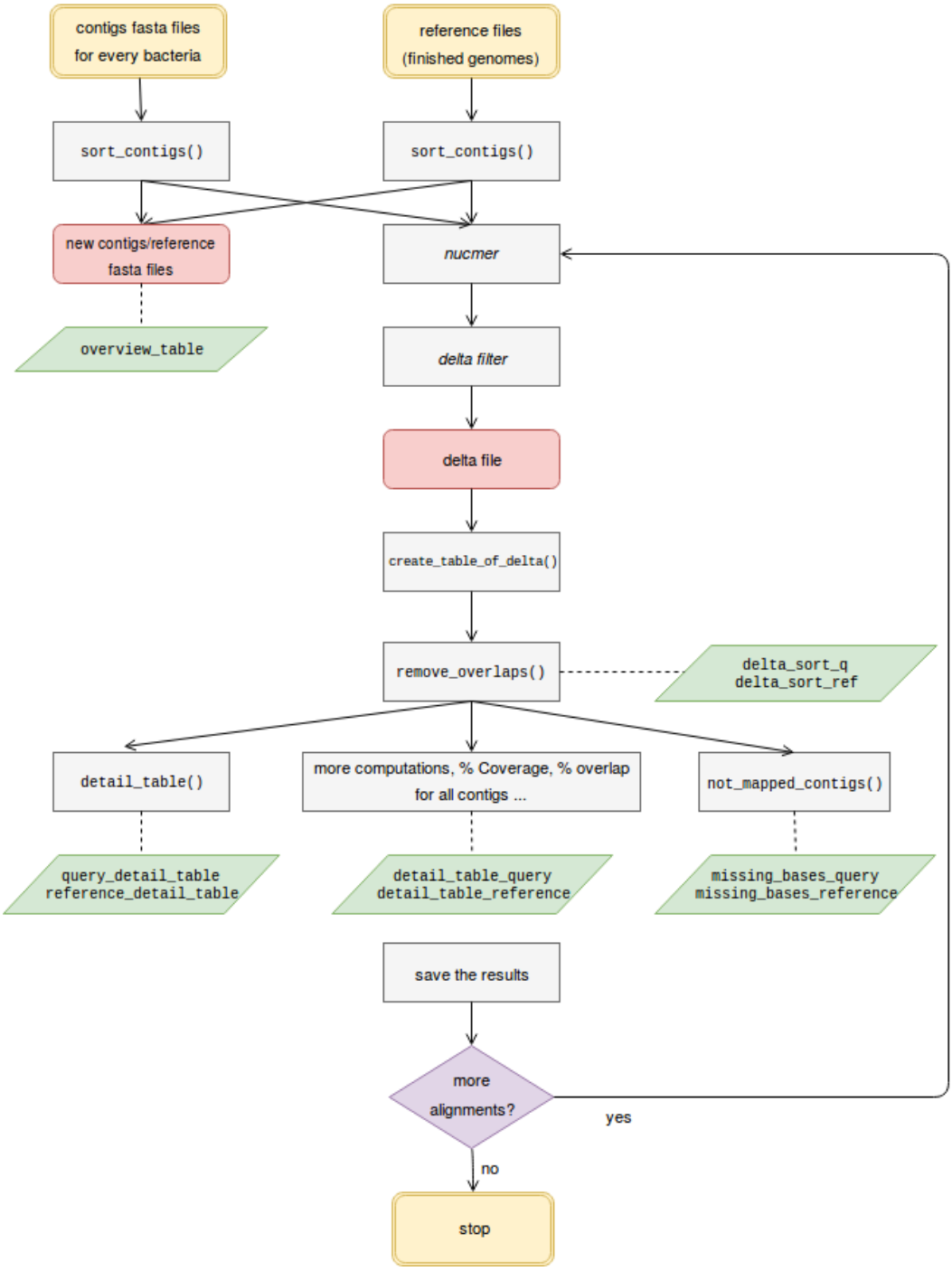


Figure 7: Flow diagram of the implemented R script to compare the different assemblies. Description of the implemented functions see Section 6.1.

of the overlap every base which is aligned more than once to the reference is counted. So more than 100% overlap is possible. This information is stored in the data frames `reference_detail_table` and `query_detail_table` respectively. For detailed information of the output tables see the listing in Section 3.1.2. For every bacteria all the assemblers were matched sequentially with the corresponding reference genome. In the second run every assembler was again matched but now the Newbler assembly was used as reference.

3.1.2 Description of Computed Parameters for Comparison

For every Assembler-Reference and also for every Assembler-Newbler pair, nine data frames were computed and saved as RData-file. In the following the names of these data frames and the computed parameters are listed.

- `delta_sort_ref`/`delta_sort_q`: parameters correspond to the results of Nucmer. `delta_sort_ref` is sort by R_Number while `delta_sort_q` is sort by Q_Number.
  - Q_Number: Number of the query contig that is part of the match
  - R_Number: Number of the reference contig that is part of the match
  - Q_Length: number of bases in the query contig that is part of the match
  - R_Length: number of bases in the reference contig that is part of the match
  - Q_Begin: start coordinate of the alignment match in the query contig
  - Q_End: End coordinate of the alignment match in the query contig
  - R_Begin: start coordinate of the alignment match in the reference contig
  - R_End: End coordinate of the alignment match in the reference contig
  - *Aligned Length*: length of the alignment match
  - *Number of Errors*: Number of bases including mismatches an indels
- `detail_table_query` and `detail_table_reference`. Each of the parameters are computed in four units; Number of Contigs, % of Contigs, Number of Bases and % of total Length of the assembly.
  - *All Contigs*: Number of contigs

  - *Contigs <200 bp* : Number of contigs that are shorter than 200 bp.
  - *Contigs Not Mapping*: Number of contigs that do not map to any range in the reference
  - *Contigs Partially Not Mapping*: Number of contigs that do not map in 100% of their length
  - *Contigs not Mapping or Contigs Partially Not Mapping*: sum of Contigs Not Mapping and Contigs Partially Not Mapping
  - *Reference Covered by Contigs/Contigs Covered by Reference*
  - *Bases Overlapping Query/Reference*
  - *Mismatches and Indels* : Number of bases that have mismatches or indels in the alignments

- `overview_table`: short summary of the main parameters of an assembly.
  - *Contig Number*: Number of contigs
  - *Total Length*: Sum of the length of all contigs
  - *Minimum Contig Length*
  - *Maximum Contig Length*
  - *N50_value*: size of the largest contig for which half the total size is contained in that contigs and those larger
  - *Number of Identical Contigs*: after alignment - number of contigs that share more than 99,5% matches in the longer sequence

- `missing_bases_query` and `missing_bases_reference` show the ranges that do not map to the contigs of the reference and the query respectively.
  - *Contig_Number*
  - *Contig_Length*
  - *Start_Point*
  - *End_Point*
  - *Length*: Length of the range that do not map

- `query_detail_table` and `reference_detail_table` computes to every contig the number of aligned contigs, how many bases cannot be matched and how many bases matched multiple times.
  - *Contig_Number*
  - *Contig_Length*
  - *Number_of_Aligned_Contigs*: How many contigs (partially) matched to the particular contig
  - *Missing_Bases*: How many bases of the particular contig cannot be matched
  - *Overlap_Bases*: How many bases matched more than once. Multiple matches counted every time.
  - *Percentage_Missing_Bases*
  - *Percentage_Overlap_Bases*
  - *Mapped_Contigs*: The number of all the contigs that aligned to the particular *Contig_Number*

3.1.3 Ranking of Assembly Tools

Another R script was implemented to create a summary table with the main parameters of all of the nine data frames (Tables A.1-A.12 in Section 6 Appendix), except for the Bacteria *B. cereus* HiSeq, because the analysis of the results of SGA was not possible due to the high number of contigs and it was not possible to assemble *X. axonopodis* HiSeq data with Newbler. The process always stalled while computing the alignment 428000 of 5501870.

The number of contigs ranges from 173 (MaSuRCA) to 1901 (SGA) for *V. chcolorae* MiSeq and from 130 (MaSuRCA) to 12,186 (SGA) for *R. sphaeroides* HiSeq. The amount of reference that is covered by the contigs ranges from 91.93% (CABOG) up to 99.99% (SPAdes) and the number of bases in the assembly that do not map to the reference from 620 (SPAdes) to 371,652 (CABOG); (Tables 8 and 9). To get an clear overview of the results a ranking was computed of all nine assembley tools, for each parameter, as well as a global ranking. The ranks for each assembler in every bacteria was summed up. MaSuRCA achieved the best overall rank while

SGA the last (Table A.13). In each cell you can see the sum of the ranks that the corresponding assembler achieved for the specific parameter. An additional ranking was determined based on the most important parameters regarding quality and accuracy of the assemblies with the references. In that special case Newbler ranked best while SGA again performed worst (Table A.14).

It is also interesting to see the effect of the different read lengths of MiSeq (250 bp) and HiSeq (100 bp) reads on the individual assemblers. To this end, the three bacteria *R. sphaeroides*, *V. cholerae* and *M. abscessus* where both HiSeq and MiSeq results were available, are compared. Significant differences are observed for Newbler (2nd place for MiSeq and 4th for HiSeq) and for CABOG (5th place for MiSeq and 3rd for HiSeq); (Tables A.15 and A.16).

Further all assemblies of the GAGE-B study were compared with the results of Newbler assembly as reference following the same R script as described before. The assemblies compared to Newbler assembly are quite different, for example, identical contigs with Newbler ranges from 0 to 68.

3.2 *Helicobacter pylori* Genomics

3.2.1 PacBio Reads Statistics

The number of reads available for the four strains ranged from 18 to 122 thousand with an estimated coverage of 48 to 470-fold. The least data was available for PM22. Average read length is between 3.7 and 6.2 kbp. Statistics of the provided raw PacBio reads is shown in Table 10 and the distribution of the read length in Figure 8. After hybrid error correction with proovread, the average read length ranges from 5 to 7 kbp. In all cases the coverage is still higher than 35-fold (Table 11). Proovread can trim and split reads in the last correction step at low quality regions. The average read length drops to 3,251 bp for SS1 up to 4,097 bp for PM22 (Table 12). Reads after using the self-error-correction tool of Canu assembler have an average read length of 5 (PMSS1) to 14 (SS1) kbp and a coverage above 33x (Table 13).

Table 8: *V. cholerae* MiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	564	242	241	501	173	1,901	1,486	209	312	2
Total Length	3,959,969	3,945,419	3,953,298	4,112,140	4,034,988	4,357,240	4,451,154	3,960,394	3,954,533	4,033,464
Minimum Contig Length	63	1	1,007	122	357	87	100	95	101	1,072,315
Maximum Contig Length	178,118	285,908	140,691	450,326	255,146	105,420	741,022	246,346	246,179	2,961,149
N50 Value	60,973	136,901	33,710	112,926	76,131	23,501	246,623	92,036	71,357	2,961,149
Reference Covered by Contigs %	99.62	99.74	98.07	99.75	98.51	99.66	99.76	99.66	99.58	
Bases Overlapping Query %	14.9	5.11	3.72	6.7	3.15	22.77	6.24	5.02	6.36	
Mismatches and Indels %	0.9	0.3	0.22	0.3	0.18	1.16	0.34	0.23	0.41	
Bases not Mapping	15,419	10,656	77,920	10,073	60,192	13,563	9,782	13,834	16,770	
Contigs Covered by Reference %	99.98	99.67	99.85	99.54	99.97	99.83	90.11	99.84	99.79	
Bases Overlapping Reference %	13.18	2.77	3.45	8.58	4.67	32.81	6.56	3.31	4.49	
Identical Contigs with Newbler	10	-	0	11	10	1	3	5	9	

Table 9: *R. sphaeroides* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	1,557	1,014	539	1,581	130	12,186	350	798	7,436	7
Total Length	4,633,465	4,548,770	4,216,057	4,725,855	4,489,204	5,704,978	4,670,077	4,525,236	5,220,141	4,602,977
Minimum Contig Length	31	4	101	45	301	101	56	97	100	37,100
Maximum Contig Length	66,408	41,908	41,321	110,511	358,962	75,323	291,333	59,062	45,522	3,188,524
N50 Value	13,002	10,095	12,421	17,052	176,783	8,907	74,486	13,775	8,960	3,188,524
Reference Covered by Contigs %	99.92	99.87	91.93	99.98	97.83	96.01	99.99	99.02	99.43	
Bases Overlapping Query %	4.11	3.15	2.37	5.15	4.08	24.17	5.04	4.04	5.09	
Mismatches and Indels %	0.31	0.16	0.18	0.33	0.27	1.04	0.25	0.25	0.36	
Bases not Mapping	3,892	5,780	371,652	803	99,810	183,509	620	45,223	26,383	
Contigs Covered by Reference %	99.69	99.97	99.98	99.98	100	99.9	99.44	99.98	99.96	
Bases Overlapping Reference %	4.57	2.03	1.82	7.96	3.67	57.8	6.03	3.24	19.69	
Identical Contigs with Newbler	37	-	68	5	1	1	1	11	6	

Table 10: Provided PacBio reads (Coverage based on a reference length of 1,618,480 bp)

	PM21	PM22	PMSS1	SS1
Number of Reads	39,311	18,021	20,547	122,600
Number of Bases	146,364,886	100,941,738	77,252,921	760,983,670
Average Read Length	3,723	5,601	3,760	6,207
Minimum Read Length	35	35	35	35
Maximum Read Length	36,553	32,949	34,327	35,250
Reference Coverage	90	62	48	470

Table 11: *Untrimmed* PacBio reads after hybrid error correction with proovread. Illumina reads were removed from adapter sequences using Cutadapt.

	PM21	PM22	PMSS1	SS1
Number of Reads	15,936	11,688	11,358	17,437
Number of Bases	96,500,914	80,829,246	57,584,368	115,043,036
Average Read Length	6,056	6,916	5,070	6,598
Minimum Read Length	261	277	258	263
Maximum Read Length	36,553	31,373	33,436	32,919
Reference Coverage	59.62	49.94	35.58	71.08

Table 12: *Trimmed* PacBio reads after hybrid error correction with proovread. Illumina reads were removed from adapter sequences using Cutadapt.

	PM21	PM22	PMSS1	SS1
Number of Reads	22,893	18,611	14,737	32,036
Number of Bases	89,287,563	76,254,408	54,234,516	104,157,835
Average Read Length	3,900	4,097	3,680	3,251
Minimum Read Length	30	27	403	34
Maximum Read Length	34,370	30,914	29,566	25,531
Reference Coverage	55,17	47.11	33.51	64.35

Table 13: PacBio reads after self-error-correction with Canu

	PM21	PM22	PMSS1	SS1
Number of Reads	6,501	8,034	9,728	4,255
Number of Bases	60,496,929	52,637,619	52,964,011	59,247,115
Average Read Length	9,306	6,552	5,444	13,924
Minimum Read Length	1,012	1,001	1,001	1,292
Maximum Read Length	34,312	28,790	33,091	29,696
Reference Coverage	37	33	33	37

3.2.2 Illumina Reads Statistics

To get the best results for hybrid error correction of the PacBio reads, the Illumina reads were trimmed and adapter sequences were removed using either Cutadapt or Trimmomatic. Trimmomatic provides a FASTA file with Illumina specific sequences used in the sequencing process and removes the adapter and further trims the ends

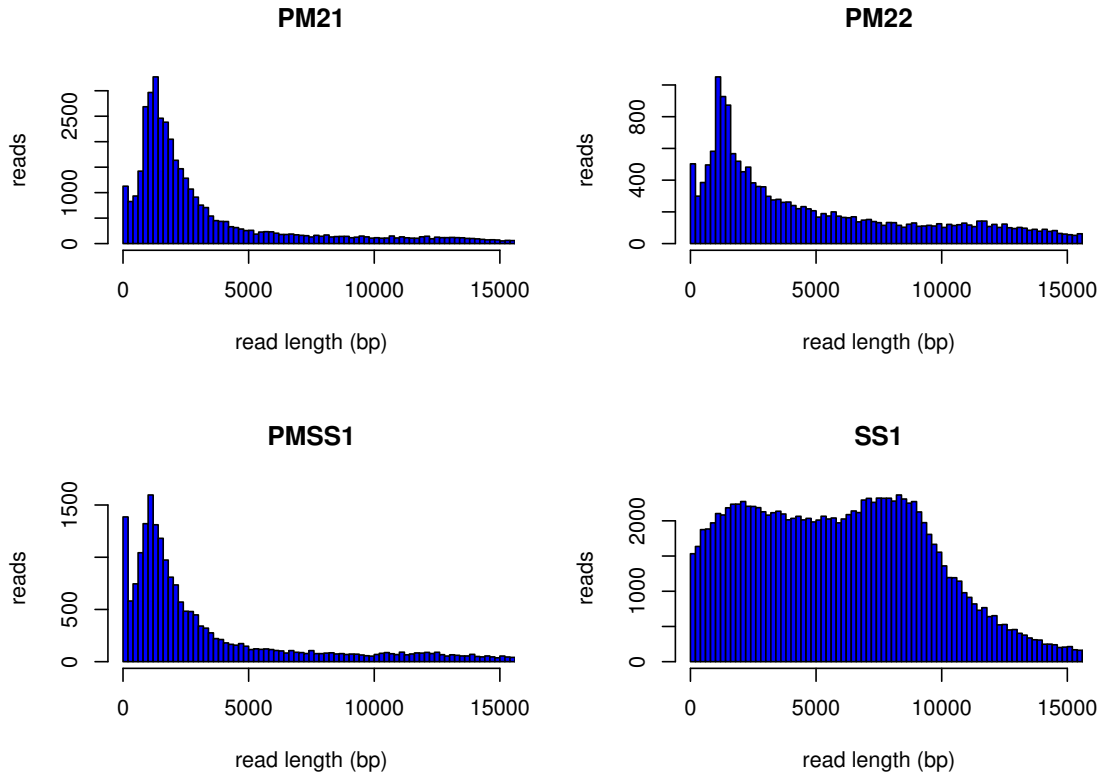


Figure 8: Read length distribution of raw PacBio data of *Helicobacter pylori* strains PM21, PM22, PMSS1 and SS1.

of the reads if the quality falls below 3. The minimum length of the corrected reads is 36 bp. In another run of Illumina read trimming (Trimmomatic modified) the minimum length of the corrected reads and the quality threshold were reduced to 1. The statistic of PacBio reads of strain PMSS1 after proovread, depending on the use of these three methods for Illumina reads adapter removal, does not show significant differences (Table 14). The reference coverage of the Illumina reads drops to around 100 base pairs in all strains after adapter removal with Cutadapt (Table 15).

Table 14: Trimmed PacBio reads for strain PMSS1 after proovread. Illumina reads were removed from adapter sequences using either Cutadapt or Trimmomatic.

	Cutadapt	Trimmomatic	Trimmomatic modified
Number of Reads	14,737	14,890	14,892
Number of Bases	54,234,516	55,023,769	53,964,429
Average Read Length	3,680	3,695	3,624
Minimum Read Length	403	335	500
Maximum Read Length	29,566	29,433	29,565
Reference Coverage	34	34	33

Table 15: Initial Illumina reads (151 bp) statistic of both Illumina paired-end files and reads statistic after adapter removal with Cutadapt

	PM21	PM22	PMSS1	SS1
Number of Reads	7,811,358	17,222,998	8,771,554	15,211,292
Initial				
Number of Bases	1,179,515,058	2,600,672,698	1,324,504,654	2,296,905,092
Reference Coverage	738	1,628	828	1,438
After Cutadapt				
Number of Bases	1,086,163,522	2,463,928,852	1,242,993,451	2,108,718,948
Average Read Length	139	143	142	139
Reference Coverage	671	1,522	768	1,303

3.2.3 Hybrid Approach

Prior to the hybrid assembly, the six subread PacBio files available for each strain were combined with the function `get_one_short_read_set_q()`. Different hybrid error-correction tools were tested on strain PM21. LoRDEC trim corrected reads have an average length of 3.5 kbp and a coverage of 80x while corrected reads with Jabba just had a length of 0.3 kbp and a coverage of 7x. Proovread's results are in the middle with 4.5 kbp average read length and 60-fold coverage (Table 16). Error corrected PacBio reads of strain PM21 were used for assembly with Newbler and Canu. Canu produced 17 contigs with a coverage of 93% using proovread corrected reads and produced 99% coverage as well as 257 contigs with LoRDEC corrected reads. Newbler produced a higher amount of contigs (58 and 1300) and the coverage was 26% and 63% respectivley (Table 17). After evaluation of the three correction tools all further assemblies were executed with proovread.

As the evaluation of the assemblies with Qualimap showed that most of the Illumina reads are clipped during the mapping process, it may be assumed that Illumina reads still have adapters. Remaining adapters were removed with Trimmomatic and Cutadapt using Illumina adapter sequences provided by Trimmomatic to get best hybrid correction and assembly results. Trimming results of Cutadapt and Trimmomatic on the Illumina reads of strain PMSS1 are similar, therefore reads were further trimmed only with Cutadapt using default parameters.

The trimmed Illumina reads and the PacBio reads are the input for proovread. The output are two files, one with all corrected and trimmed long reads (trimmed) and one with all the corrected reads including regions that did not have enough Illumina reads mapped (untrimmed). The statistic of the error corrected PacBio files after proovread and Cutadapt is shown in Tables 11 and 12.

Assemblies were created with both kinds of reads being assembled into slightly fewer contigs with Canu and proovread's trimmed reads and for Newbler with proovread's untrimmed reads, because Newbler has a trimming step included that is adapted for its own assembler algorithm. Newbler produced 22 contigs for PM21 to 32 contigs for SS1 with a reference coverage of 98.5% in each assembly. Canu produced 3 contigs for strain PM21 and up to 9 for strain SS1. In each case the coverage is almost 100% (Tables 18 and 19). Assemblies with Canu and Newbler are also performed with the published reference PMSS1 genome to know the maximum possible contiguity of these assemblers when dealing with error free reads. To this end, the reference GenBank sequence CP018823.1 and the plasmid CP018824.1 was split with random length between 1500 and 1999 bp and coverage 40 using `reference_splitting()`. The split reads are the input for Newbler and Canu, respectively (Tables 18 and 19).

With nucmer [54] the contigs of the assemblies were aligned to the reference assembly. To evaluate these assemblies the R-function `create_table_of_delta()` produces two tables one with all the ranges of overlaps with the reference and one with the ranges not included in the contigs of the assemblies. At two regions of the genome there are missing ranges of more than thousand base pairs for strain PMSS1 in the Canu assembly (Table 20), while Newbler has no missing ranges (Table 21). In strain PM21 there are 5 missing regions in the Newbler assembler, in each less than 50 base pairs, and one missing region with 17 base pairs in the Canu assembly (Tables 22 and 23). All the missing ranges for strain PMSS1 using the different trimming and assembly tools are shown in Tables A.36 to A.50.

Table 16: LoRDEC, Jabba and proovread hybrid error correction reads statistics on strain PM21.

	LoRDEC `split`	LoRDEC `trim`	Jabba	proovread
Number of Reads	180,972	37,319	32,098	20,379
Number of Bases	56,319,064	128,586,429	11,688,414	92,814,063
Average Read Length	311	3,446	364	4,554
Minimum Read Length	100	19	43	376
Maximum Read Length	2,181	35,050	3,485	30,441
Coverage	34	77	7	58

Table 17: Assemblies of strain PM21 after error correction with LoRDEC (L) or proovread (p).

	Newbler (L)	Newbler (p)	Canu (L)	Canu (p)
Number of Contigs	1,358	58	257	17
Number of Bases	1,046,766	1,591,992	434,122	1,491,292
Average Contig Length	771	27,448	1,689	87,723
Minimum Contig Length	100	348	1,001	10,292
Maximum Contig Length	5,338	178,232	5,897	207,150
Coverage	0.63	0.26	0.99	0.93
N50	1,266	1,708	72,359	118,033

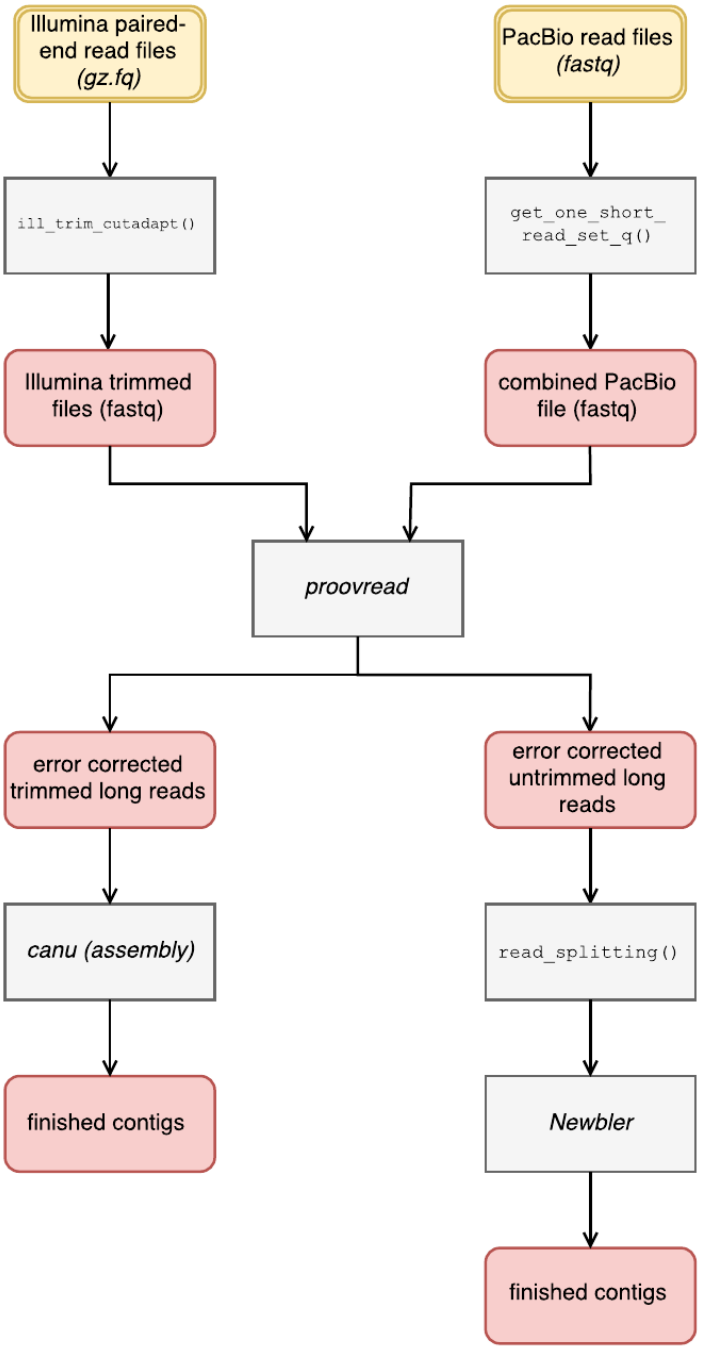


Figure 9: Flow diagram of the implemented R-Script using a hybrid assembly approach.

Table 18: Canu hybrid assembly statistic after proovread (trimmed) and adapter removal of Illumina reads with Cutadapt.

	PM21	PM22	PMSS1	PMSS1 Ref.	SS1
Number of Contigs	3	4	5	11	9
Number of Bases	1,611,458	1,610,425	1,618,840	1,624,016	1,599,252
Average Contig Length	537,153	402,606	323,768	147,638	177,695
Minimum Contig Length	8,492	10,131	10,153	2,692	21,351
Maximum Contig Length	908,828	787,101	849,867	531,590	723,822
Reference Coverage	0.996	0.995	1.000	1.00	0.988
N50	908,828	696,545	849,867	236,000	214,592

Table 19: Newbler assembly statistic after proovread (untrimmed) and adapter removal of Illumina reads with Cutadapt.

	PM21	PM22	PMSS1	PMSS1 Ref.	SS1
Number of Contigs	22	29	23	23	32
Number of Bases	1,594,561	1,594,639	1,592,198	1,595,242	1,594,523
Average Contig Length	72,480	54,988	69,226	69,358	49,829
Minimum Contig Length	303	187	461	462	167
Maximum Contig Length	500,017	499,456	499,602	319,311	500,014
Reference Coverage	0.985	0.985	0.984	0.986	0.985
N50	190,506	189,026	180,015	178,900	178,737

Table 20: Ranges of reference genome that are not covered by PMSS1 Canu contigs.

	GenBank accession	start	end	width
1	CP018823.1	627,520	627,790	271
2	CP018823.1	688,747	689,186	440
3	CP018823.1	689,364	690,365	1,002
4	CP018823.1	691,389	691,508	120
5	CP018823.1	826,509	826,519	11
6	CP018823.1	1,395,521	1,398,498	2,978

Table 21: Ranges of reference genome that are not covered by PMSS1 Newbler contigs.

	GenBank accession	start	end	width
1	CP018823.1	1,438,186	1,438,189	4

Table 22: Ranges of reference genome that are not covered by PM21 Canu contigs.

	GenBank accession	start	end	width
1	CP018823.1	1,398,481	1,398,497	17

Table 23: Ranges of reference genome that are not covered by PM21 Newbler contigs.

	GenBank accession	start	end	width
1	CP018823.1	1,033,069	1,033,113	45
2	CP018823.1	1,162,150	1,162,150	1
3	CP018823.1	1,400,968	1,400,969	2
4	CP018823.1	1,404,779	1,404,822	44
5	CP018823.1	1,409,206	1,409,206	1

3.2.4 Stand-Alone Approach with Canu Assembler

The general work-flow of the implemented R-script to produce a suitable assembly using the whole pipeline of the Canu assembler and following steps to improve the assembly and annotation is shown in Figure 10.

PacBio files belonging to one strain were merged into one fastq-file. This file is the input for the stand-alone long read Canu assembler.

The statistics of the assemblies produced by Canu with default parameter using only the long uncorrected PacBio reads is shown in Table 25. The statistic of the corrected PacBio reads using Canu pipeline is shown in Table 13. The corrected PacBio reads were realigned to the genome sequences created by Canu using BWA-MEM [58] to evaluate the assemblies. It can be observed, that no reads could be mapped over the joined ends of the assembly (Figure 11a). It was necessary to trim and reassemble the ends of the genomes with Circlator [14]. Circlator successfully trimmed and assembled these ends again using the corrected PacBio reads and SPAdes assembler [52], built a circularised genome and present the linear sequence [14]. For the circularised genome reads map over the ends of the genome (Figure 11b).

To compare the sequences with the recently published assembly of strain PMSS1 CP018823.1 [79] the R-function `rearrange_assembly()` (Section 6.1), was implemented. It rearranges the sequence to have the same start point and the same strand orientation as the reference PMSS1 genome.

To evaluate the current assembled sequences, mappings with PacBio or Illumina reads were made. The software IGV [63] and Qualimap [64] were used to evaluate these mappings. Qualimap showed a peak with more than 3 times average coverage in the region of the cagA gene. To control the coverage of the Illumina reads across the assembled sequence in comparison with the number of cagA copies, cagA copies were inserted and the mapping and Qualimap analysis repeated. Using this approach, a drop to average coverage could be observed with four and five times cagA copy numbers (Figure 12).

Comparison of the Canu PMSS1 sequence with the published PMSS1 sequence using the NCBI nucleotide BLAST online tool with default parameters [60] showed that there were still hundreds of indels present in the assembly (Table 24). Mappings of the Illumina and PacBio reads to the stand-alone Canu assembly in two regions are shown in Figures 13 and 14.

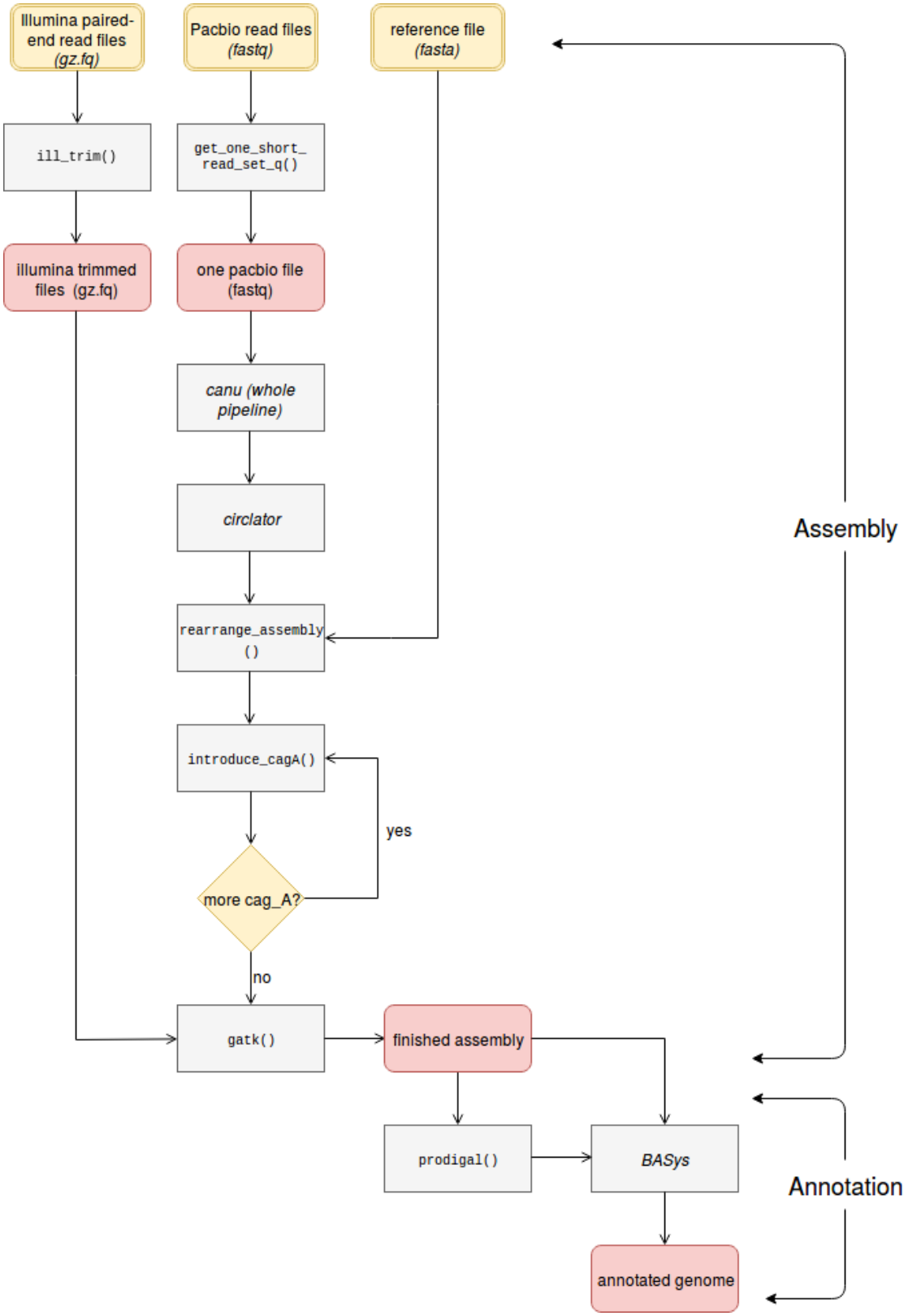
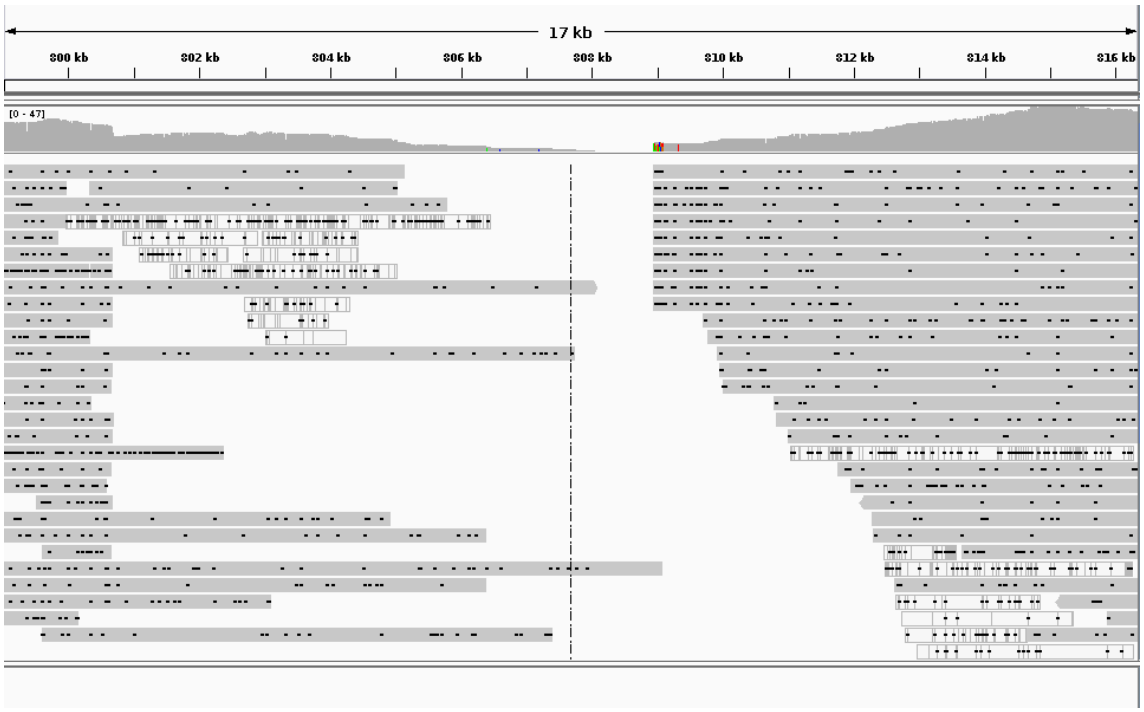


Figure 10: Flow diagram of the implemented R script to gain the assembly using a stand alone approach with Canu assembler.

(a)



(b)

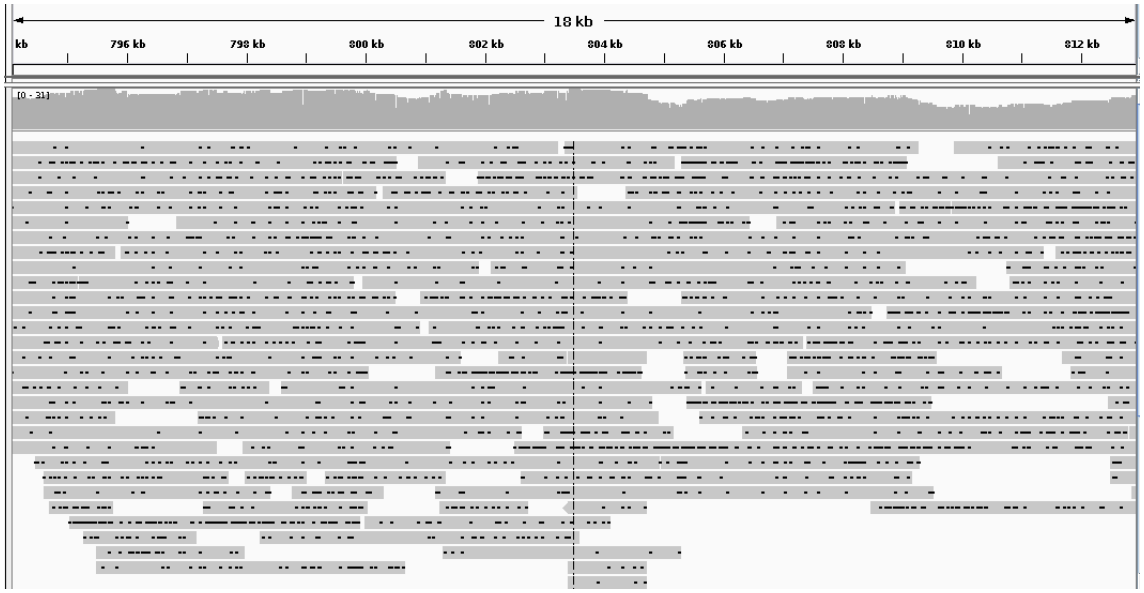


Figure 11: Visualisation of the alignment of the PacBio reads of strain PMSS1 to the corresponding Canu assembly with the software IGV [63]. The ends of the assembly were rearranged in the middle of the genome. On top the visible portion of the genome is shown indicating the location. Underneath the amount of read coverage and on the bottom the aligned reads are presented. Coverage drops down to zero in the middle which indicates an assembly failure at these ends of the assembly. In (a) the mapping before circularisation, and in (b) the mapping after trimming and circularisation with Circlator is shown.

To improve the assemblies the sequences had to be polished using the more accurate Illumina reads for consensus building. To this end, the HaplotypeCaller tool from the software GATK [61] was used. All steps to build the consensus performed within the function `gatk()`. After Circlator around ten thousand bases were trimmed from the Canu assemblies and around thousand bases were introduced after using GATK. There are no changes in the plasmid of PM21 after using GATK (Table 25). BLAST showed that in strain PMSS1 the number of indels compared to the published PMSS1 strain could be reduced from 1468 to 26. In all the assemblies the final number of gap opens is now around 30 (Table 24).

Protein coding genes were predicted using Prodigal. The number of proteins predicted with Prodigal for the published PMSS1 strain is 1,505, the number of CDS that are annotated is 1,535. To evaluate the assembly after GATK and differences between the assembled PMSS1 strain and the recently published PMSS1 assembly in GenBank, the Prodigal results were compared according to the number of identical proteins and mismatched or shorter/longer proteins. The assembled strain PMSS1 has 1484 out of 1510 identical proteins with the reference strain (Table 26). With the finished assembly and the results of Prodigal the CDSs could be annotated using the BASys server [66].

Table 24: Number of mismatches and gap opens compared to the reference PMSS1 genome. Alignments performed with BLAST.

	PM21	PM22	PMSS1	SS1
Before GATK				
# Mismatches	182	73	15	68
# Gap Opens	508	1086	1468	237
After GATK				
# Mismatches	106	69	44	68
# Gap Opens	29	28	26	31

Table 25: Statistic after Canu self-error-correction and assembly, after Circularisation with Circlator and after consensus building with GATK. Canu produced one contig per strain, two for strain PM21, PM22 one including the plasmid.

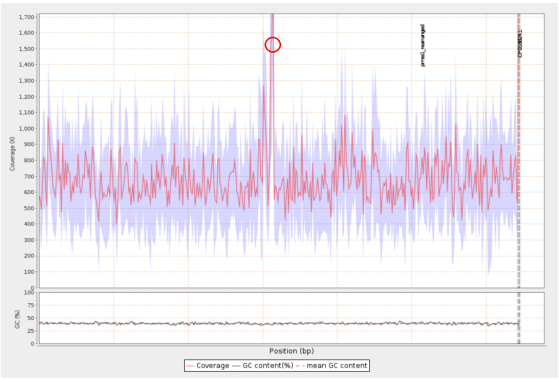
	PM21 Genome	PM21 Plasmid	PM22	PMSS1	SS1
After Canu					
Number of Bases	1,613,639	10,137	1,610,809	1,618,182	1,627,980
Reference Coverage	0.997	1.657	0.995	1.000	1.006
After Circlator					
Number of Bases	1,602,465	6,058	1,602,056	1,607,011	1,613,791
Reference Coverage	0.990	1.000	0.990	0.993	0.997
After GATK					
Number of bases	1,603,014	6,058	1,602,971	1,608,333	1,614,005
Reference Coverage	0.990	1.000	0.990	0.994	0.997

Table 26: Prodigal protein comparison of PMSS1 final assembly with the reference GenBank sequence CP018823.1.

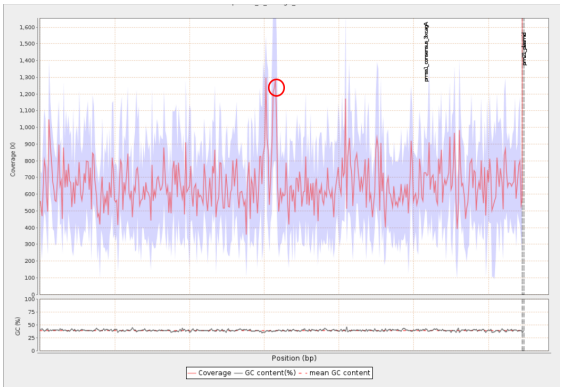
	Own PMSS1	PMSS1 (CP018823)
# Proteins	1,510	1,505
# Identical Proteins*	1,484	1,486
# Proteins with sequence mismatches	5	5
# Shorter/Longer Proteins	21	14

* Identical Proteins: Different results due to multiple occurrences of the same protein in one strain.

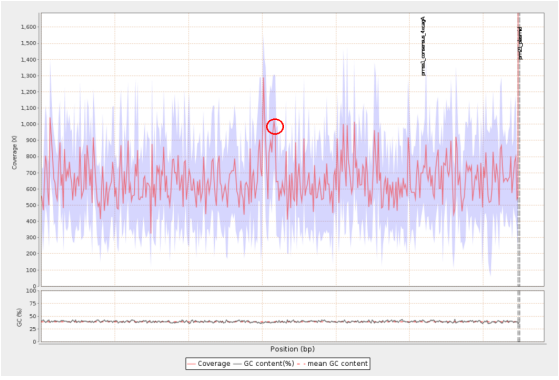
(a) 2 copies of cagA gene



(b) 3 copies of cagA gene



(c) 4 copies of cagA gene



(d) 5 copies of cagA gene

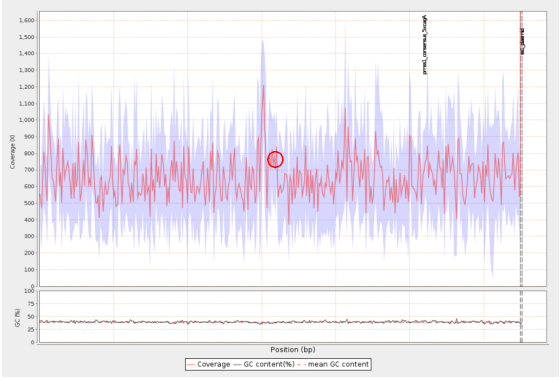


Figure 12: Illumina read coverage across PMSS1 assembly with different copy numbers of the cagA region. Peak (red circle) returns to average with 4 and 5 times cagA copies.

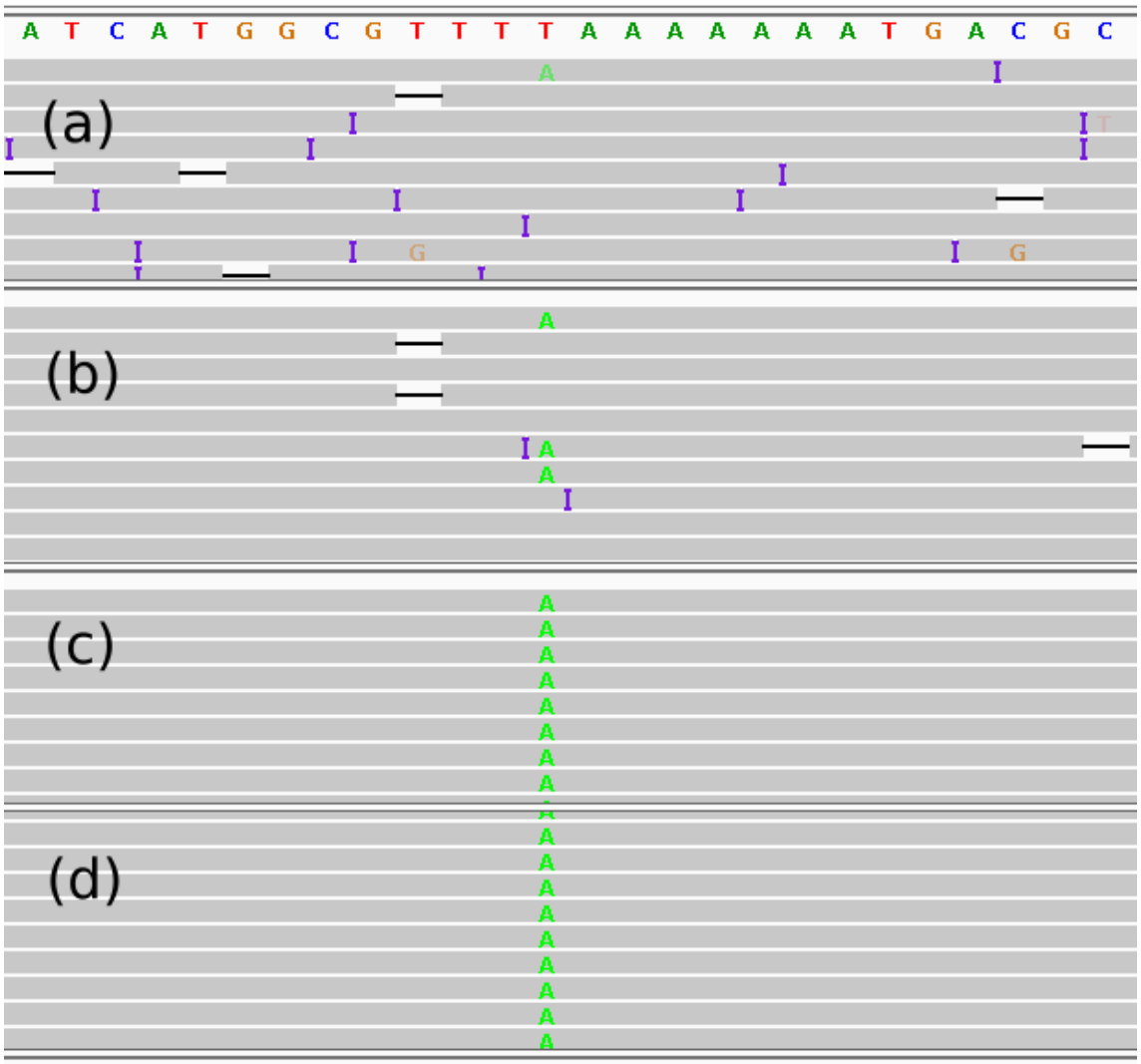


Figure 13: Mapping of reads to the PMSS1 assembly (position 246,195 to 246,220) before GATK, visualised with IGV. Dark gray beams represent the coverage at that point. Light gray lines are the mapped reads. Blue "I"s denote insertions, black lines deletions and letters mismatches. On the top the bases of the assembly in that region are given. (a) mapping of uncorrected PacBio reads, (b) mapping of self-error corrected PacBio reads using Canu, (c) mapping of hybrid error corrected PacBio reads using proovread, (d) mapping of Illumina reads.

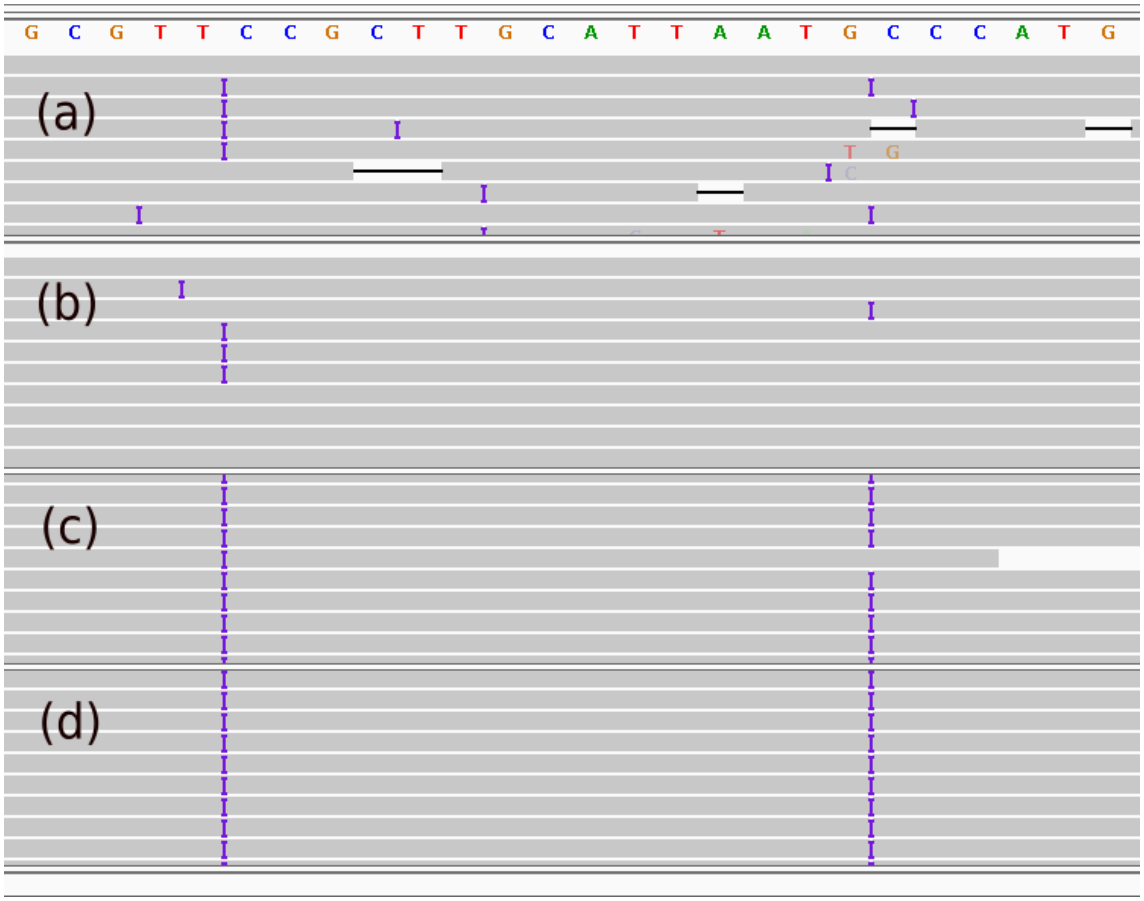


Figure 14: Mapping of reads to the PMSS1 assembly (position 229,281 to 229,306) before GATK, visualised with IGV. Dark gray beams represent the coverage at that point. Light gray lines are the mapped reads. Blue "I"s denote insertions, black lines deletions and letters mismatches. On the top the bases of the assembly in that region are given. (a) mapping of uncorrected PacBio reads, (b) mapping of self-error corrected PacBio reads using Canu, (c) mapping of hybrid error corrected PacBio reads using proovread, (d) mapping of Illumina reads.

3.2.5 Comparison of the Genomes

After executing the `Canu_assembly_pipeline.R` script with all the four assembled genomes, the finished and annotated assemblies as well as the two available reference genomes PMSS1 (CP018823) and SS1 (CP009259) could be aligned with Mauve [67]. The alignment shows differences in length according to the different cagA copy numbers from 1 to 5 (Table 27) and a large inversion of 400 kbp in strain PM22 and in SS1 (CP009259) respectively (Figure 15). The initial assembled cagA copy numbers ranges from 1 for PM21 and PM22, 2 for PMSS1 to 3 for SS1. The estimated number of cagA copies determined by Illumina read coverage changed to 2 for PM22, 4 for PMSS1 and 5 for SS1 (Table 27).

To assess the effect of genetic variants, for example what gene is affected and is it a missense or synonymous mutation, an annotated vcf file has to be created. The Illumina reads of the different strains were mapped against the assembled PMSS1 strain of this project and the HaplotypeCaller of GATK was invoked to find all possible variants. The annotated PMSS1 assembly and the vcf-file was used to annotate the variants with snpEff [68]. The R-function `snps_statistic()` reads in the annotated vcf file and creates a detail and a summary table of all local variants compared to strain PMSS1. The number of variants ranges from 78 for PM21 to 94 for SS1. The majority of these are synonymous mutations. A hot spot with more than 40 SNPs could be observed in the engB gene (Tables 28 and A.51). There are 46 common SNPs in all the strains PM21, PM22 and SS1 compared to the PMSS1 strian. PM21 and PM22 do not share any further SNPs, PM22 and SS1 share 1 further SNP (Figure 16).

In the phylogenetic tree created with RAxML it can be observed that two main clusters are formed, one with the SS1 strains and one with PMSS1, PM21 and PM22 (Figure 17). The minimum spanning tree shows that the published PMSS1 genome is connected to almost all of the other strains (Figure 18).

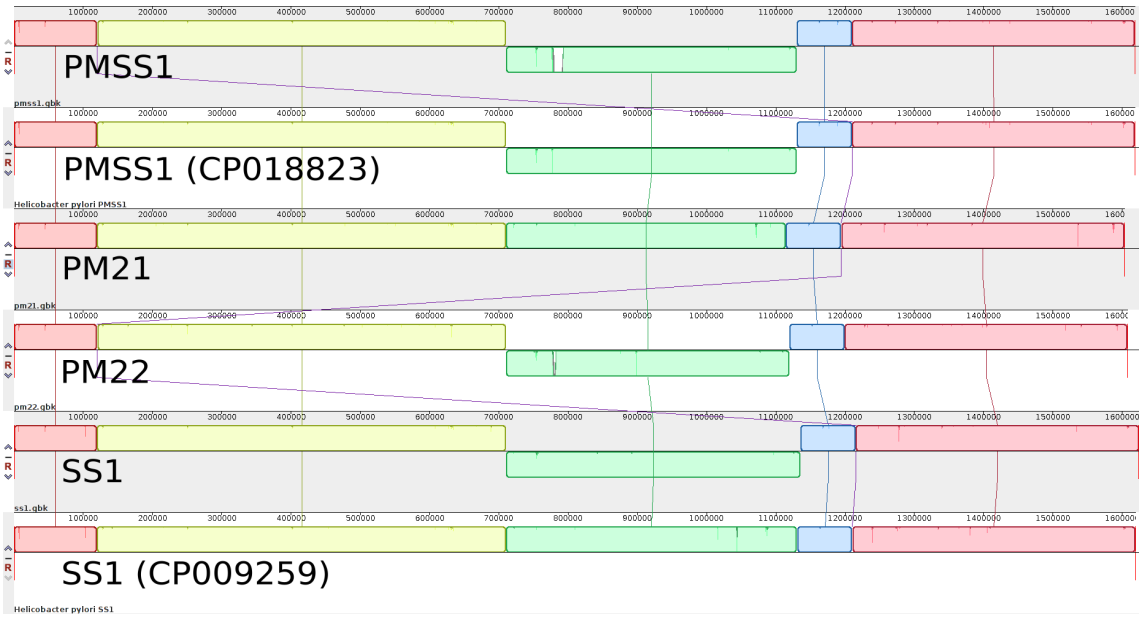


Figure 15: Alignment of the six *H. pylori* genome assemblies with Mauve [67]. Inversion of about 400 kbp in PM21 and in the reference strain SS1 (green bar) compared to the others. Slight differences in the length of the assemblies due to different numbers of cagA copies. Mauve produced alignment errors in this region in PM22 and PMSS1 (white field in the green bar).

Table 27: CagA copy number in the Canu assembly (C) and final number of cagA copies determined by the Illumina read coverage in this region (Q), predicted number of CDS with Prodigal and the final length of the genomes.

	cagA copy number (C)	cagA copy number (Q)	predicted number of CDS	genome length (bp)
PMSS1	2	4	1,510	1,618,489
PMSS1 (CP018823)	–	4	1505	1,618,480
PM21	1	1	1,497	1,602,972
PM22	1	2	1,505	1,608,054
SS1	3	5	1,507	1,624,153
SS1 (CP009259)	–	4	1,507	1,619,098

Table 28: Number of genetic variants (SNPs, Deletions, Insertions) compared to strain PMSS1.

	PM21	PM22	SS1
# Variants	78	99	94
# Intergenic	10	10	16
# SNPs	65	87	77
# Deletions	6	3	6
# Insertions	7	9	11
# Missense	21	25	23
# Synonymous	41	60	46
# Nonsense	1	0	1
Hot Spots	VirB10, rfaI, engB	infC, yejF, acsA, engB	topA, virB10, ssb, engB

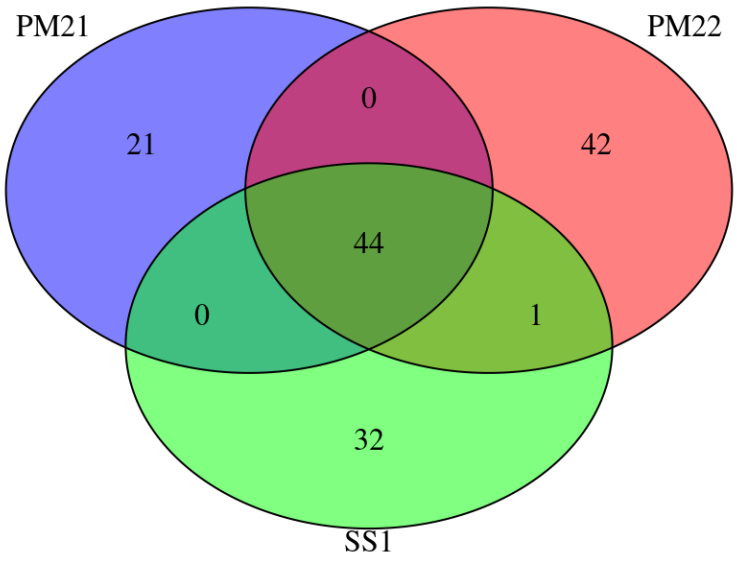


Figure 16: Venn diagram showing the number of shared SNPs of strains PM21, PM22 and SS1.

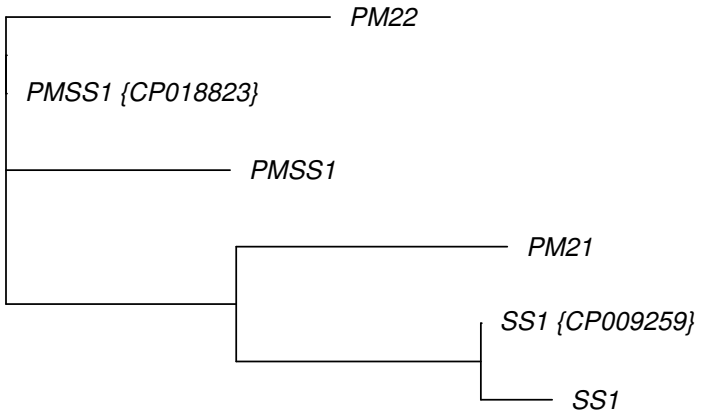


Figure 17: Phylogenetic tree of the six assembled genomes including the reference genomes published on NCBI. Tree created based on the SNPs of the whole genome alignment using a maximum likelihood model with RAxML [69].

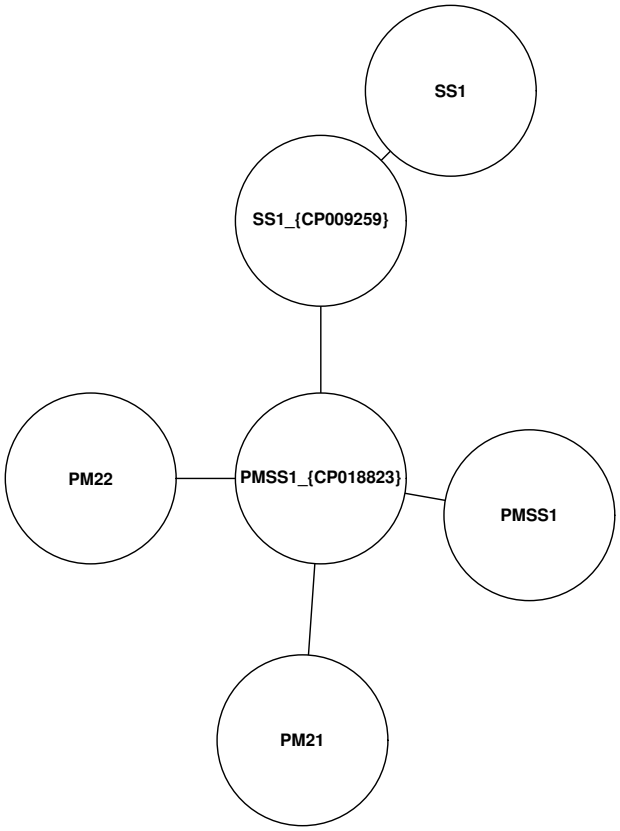


Figure 18: Minimum spanning tree of the six assembled genomes including the reference genomes published on NCBI. MST is based on the phylogenetic tree created in Figure 17.

4 Discussion

In this Master's thesis it could be shown that different assembly tools perform quite differently depending on the bacterial genome and the kinds of reads. Fully finished genomes of 4 *H. pylori* strains could be assembled using the stand-alone-error correction tool Canu, followed by a trimming and circularisation step and a consensus-building step with Illumina reads. Further it could be shown that the genome of *H. pylori* is highly variable.

4.1 Assembly Tools for Small Genomes

4.1.1 GAGE-B and GABenchToB

One of the main goals of this part was to compare the Newbler assembler with the results of GAGE-B and GABenchToB.

The GAGE-B study compared nine assemblers and especially measured metrics like N50 value, number of contigs, global errors (relocations, translocations and inversions) as well as local errors but also the number of proteins fully contained in contigs. The authors came to the overall conclusion that MaSuRCA and SPAdes produced the best assemblies across the twelve bacterial organisms investigated [31].

Newbler, a widely used assembler, was not reviewed in GAGE-B. To assess the performance of Newbler the trimmed Illumina sequence data accessible on the GAGE-B website was used.

In contrast, GaBenchToB compared nine assembler on three different bacterial genomes, *E. coli*, *S. aureus* and *M. tuberculosis*, sequenced either with Illumina MiSeq or with PGM platforms [80]. Six of these assemblers (Abyss, Mira, SoapDenovo2, SPAdes, Velvet and Newbler) are analysed in this project. MaSuRCA, SGA and CABOG were not used in GABenchToB. In particular, NGA50 values and the number of mis-assemblies were analysed. The NGA50 value is like the N50 value but for the total length the real length of a reference genome is used so the values of the different assemblies can be easily compared. Further they also had a focus on the computational cost of an assembly. They came to the conclusion that none of the assembler emerges as the overall winner. The individually assembler performance strongly depends on the nature of the data [32].

4.1.2 Evaluation of the Assemblies

As described in the GAGE-B and GABenchtoB studies, it is hard to find an assembler which performs best on all kinds of bacterial data and organisms. Assembly algorithms are developed to deal with specific types of sequencing platforms as well as genomic data and this fact is reflected in the results of this project too. They all have some benefits but also drawbacks.

Referring to often determined metrics like *Number of Contigs*, *N50 value* and the *Maximum Contig Length*, MaSuRCA performed best for all bacteria in this project. Also, SPAdes got the best total rank for the parameter Maximum Contig Length and made the second place for the N50 value (Table A.14). Referring to the Number of Contigs, SPAdes performs averagely because it sometimes produces a high number of contigs. For all of this three parameters SGA performed worst. It came last place for seven of the ten bacteria concerning the Number of Contigs, in eight concerning the N50 value and in seven concerning the Maximum Contig Length.

Another important parameter is the coverage of the reference genome by the contigs of the assemblies (*Reference Covered by Contigs*). If no contig of the assembly aligns to a specific region in the reference, genome information is missing and this area can never be reproduced without a reference genome. For eleven of the twelve bacterial genomes SPAdes achieved the best results while in eleven CABOG got the worst. It is the only parameter for which MaSuRCA performed the worst, achieving the eighth rank.

Newbler has the fewest *Mismatches and Indels* by far, which means that Newbler produces the least local errors like Single Nucleotide Polymorphisms (SNPs) or small insertions and deletions, which improves the accuracy of the finished genome.

The results of the *Bases Overlapping Query* and *Bases Overlapping Reference* analyses highly depends on the parameters of the achieved alignment. If the alignment produces multiple matches it is hard to exclude non contiguous match-pairs without reducing the coverage of the assembly and the reference, and that is why the Overlapping Bases are rising. MUMmer [54] does not provide a proper filter of the alignment results, therefore an additional filter was implemented.

If one focuses on parameters that indicates the quality and accuracy of the assembly (*Reference Covered by Contigs, Bases Overlapping Query, Mismatches and Indels, Contigs Covered by Reference and Bases Overlapping Reference*) the assembly tools

perform quite similar except Newbler that achieved the best rank. Newbler produces a high number of small contigs, but a very accurate assembly covering most of the reference genome with few multiple regions and local errors. The ranges of the values for the most important parameters are large. For example, while MaSuRCA creates only 130 contigs for *R. sphaeroides* HiSeq, SGA creates 12186 contigs. Also the total length of the assemblies ranges from 4216 kbp for CABOG to 5220 kbp for Soap, while the length of the reference genome is 4603 kbp. Of note, CABOG shares 68 identical contigs with Newbler in the *R. sphaeroides* HiSeq assembly, while CABOG does not share any contig with Newbler in the *V. cholerae* MiSeq assembly. The metric *Reference Covered by Contigs %* in *R. sphaeroides* HiSeq is about 99% in most of the assemblies but CABOG has a bad value with 91,1% (Tables 8 and 9). If one focuses on the parameters that represent the accuracy compared to the reference genome (*Reference Covered by Contigs*, *Bases Overlapping Query*, *Mismatches and Indels*, *Contigs Covered by Reference* and *Bases Overlapping Reference*), Newbler, CABOG and MaSuRCA achieved the best results. In this case the only parameter Newbler performed worse is *Contigs Covered by Reference* (Table A.14). But in all of the 12 data sets this value is only slightly lower than in the other assemblies (Tables A.1 - A.12). A reason could be that the *Minimum Contig Length* for Newbler was set to 1 and contigs smaller than 50 bp are not aligned with Nucmer.

Most of the used assembler perform quite similar when dealing with MiSeq or with HiSeq data (Tables A.15 and A.16). Noticeable is that Newbler in comparison with the other assembler performed two ranks better on the longer MiSeq (2nd) than HiSeq data (4th). The reason could be that Newbler was developed to assemble the longer (∼450 bp) Roche 454 sequenced reads [81]. CABOG performed better on HiSeq data (3rd vs. 5th) in general.

To determine how similar the assemblies to the common Newbler assembly are, the R script was run again with Newbler as reference genome. The highest similarity is observed with CABOG, except for the parameter *Reference Covered by Contigs*, and also Velvet has a high similarity to Newbler, especially concerning the *Number of Identical Contigs* (Tables A.17 - A.21).

4.2 *Helicobacter pylori*

4.2.1 Available Data

The PacBio reads have a length distribution as expected, but the average read length of about 3700 bp for strain PM21 and strain PMSS1 is relatively short compared to the average length of PacBio reads published on the PacBio website [7], where the average read length can be more than 10,000 bp (Table 10). The genome coverage between 48x for strain PMSS1 and 470x for strain SS1 is sufficient for any hybrid approach. The usual recommendation is a minimum long read coverage of about 20x to 30x [82]. For a standard stand-alone approach the read coverage should be around 75x [83] but Canu can deal with much lower coverage of about 30x [84].

4.2.2 Hybrid Approach

A main part of hybrid assembly is the correction of the PacBio reads that have an error rate of at least 15% with the help of the more accurate Illumina reads that have an error rate of less than 1%. A lot of different tools are available (Section 1.1.3), three tools LoRDEC, Jabba and proovread were used on the PacBio data sets. Both LoRDEC and Jabba correct reads by building a de Bruijn Graph of the short reads and threading the long reads through this graph. It has been shown that they are as accurate as mapping error correction tools like proovread [16,21]. For strain PM21, Jabba generated reads with an average length of 300 bp and the remaining coverage was only 7x (Table 16). Reads might be very accurate but most of the reads are filtered during correction, making an assembly impossible. The reads after LoRDEC split, where all the corrected reads are trimmed and split at regions that could not be corrected, are as short as from Jabba and have a coverage of 34x. Reads that are not split (LoRDEC trim) are much longer with 3446 bp average and could be used for assembly with Newbler and Canu. Reads generated with proovread are the longest of about 4500 bp average length and still have enough coverage with 58x.

The assembly for all strains using the corrected long reads were done with Newbler assembler because it achieved best results in the comparison of the assembly tools and with Canu, an assembler especially developed to assemble SMRT-reads. Newbler performed best using proovread's untrimmed reads because Newbler has its own trimming algorithm. Canu achieved better results using the trimmed reads proovread produced. Newbler produced 58 contigs and the coverage is 26%. The best results

produced Canu when using proovread's corrected and trimmed reads, but there are still 17 contigs and missing reference genome ranges.

It was observed, that most of the Illumina reads are clipped when mapped to the reference genome. Therefore we suspected that Illumina reads may still contain adapters. This was confirmed by the more contiguous assemblies resulting from the assemblies of the trimmed reads with either Canu or Newbler (Tables A.32 - A.35). Additionally, most of the ranges of the PMSS1 reference genome are covered for strain PMSS1 (Tables A.36 - A.50). Especially the Newbler assembler produced more contigs with no missing ranges while Canu produced less contigs but the assembly has missing bases in the range of 1000 bp at regions that have low Illumina read coverage for strain PMSS1. Furthermore Canu sometimes does not assemble the plasmid. Canu produced best hybrid result for strain PM21. It assembled only 2 contigs for the genome and one plasmid (Table 18) and there are just 17 bases missing compared to the reference PMSS1 genome (Table 22). As the assemblies still consisted of more than one contig, creating finished and annotated genomes was not possible.

4.2.3 Stand-Alone Assembly

The stand-alone assembly with the recently published Canu assembler [17] was a good choice because it could directly assemble our genomes into a single contig per strain. Mappings of the raw PacBio data for all strains to the reference PMSS1 Plasmid CP018824.1 showed a high coverage indicating that the plasmid is present in the PacBio data, but could be only assembled for strain PM21. In all the other strains the plasmid was removed in the corrected PacBio reads and the assembly. According to the Canu documentation [84] it is recommended to increase in that case the corrected read coverage, because Canu uses only the longest 40x coverage reads for correction. The assembly for the strains PMSS1, PM22 and SS1 was performed again with the *corOutCoverage* parameter set to 100 and a plasmid was now present in the assembly of PM22 too. For PMSS1 and SS1 it was not possible to assemble the plasmid. Canu recommends a coverage above 30 for assembly, which is quite low for a stand-alone assembly approach. After evaluation of the assembly by mapping the corrected PacBio reads to the assembly and visualise the mapping with the software IGV it was evident that the assembly is incorrect at the ends because the coverage falls close to zero at these ends. Furthermore Canu does not recognise the contigs

as circular. To trim and circularise the contigs Canu produced, Circlator was used, which generated a complete circular chromosome (Figure 11b).

One of the important genomic regions of bacteria that cause inflammation of the human's stomach are pathogenicity islands (PAI) that include the cytotoxin associated gene A (cagA), which has a high copy number variation [40]. One cagA region has a size of 5072 bp. In strain PM21 some PacBio reads could be found that span across the whole cagA region of the assembly, indicating that the repeat region is assembled correctly. For all the other strains it was not possible to find such long reads, which is why further investigation of the read coverage with Qualimap was necessary. A coverage peak could be observed in the cagA region indicating that in fact there are more cagA copies present in the genome sequence. The coverage in that area reduces to average with four and five times cagA copies. There is also a second peak slightly before the cagA region that is an unresolved repeat problem in the assembly. (Figures 12a-12d). For final assembly four times cagA for PMSS1 is used. The final number of cagA is determined in all strains through investigating the Illumina read coverage in that region (Table 27).

During the course of this thesis the research group of Draper et al. [40] published a complete assembly of strain PMSS1 (chromosome and plasmid), that could now be used for further evaluation. The plasmid of strain PM21 identified in this thesis after Circlator is identical to the published PMSS1 plasmid, but BLAST showed that there are small indels in the range of hundreds of base pairs between the PMSS1 chromosome and the published one (Table 24). It seems that Canu does not correct all of the insertions and deletions of the PacBio reads to its full extend. Koren et al. report a maximum quality of Q40 (99.99% or one incorrect base in 10,000) for Canu on bacterial genomes [17]. To improve the accuracy an additional polishing step is necessary. Canu recommends Quiver for polishing of the assemblies [17]. The polishing in this thesis is done by mapping the accurate Illumina reads to the assembly and building the consensus with the software GATK. The number of SNPs and indels could be reduced by more than 2 orders of magnitude for all four strains (Table 24).

4.2.4 Comparison of the Genomes

After predicting protein coding genes of PMSS1 with Prodigal, the resulting sequences were compared to those of the published PMSS1 genome. Most of the proteins, 1484 of 1510 are identical (Table 26), 21 of the proteins are shorter or longer due to insertions or deletions of bases. 5 proteins have mismatches in the aminoacid chain due to mismatched bases. Reasons for the differences could be mutations in the sequenced genomes or sequencing errors in homopolymer regions.

After aligning all the six genomes (including the two reference strains) with Mauve structural differences can be determined (Figure 15). Strain PM21 and the reference strain SS1 have an inversion of about 400kp and the strains have different lengths due to different numbers of cagA copies. To analyse the differences in the *Helicobacter pylori* genomes the assembled strain PMSS1, which is the initial strain the mice were inoculated with, was used as the reference strain. The number of SNPs compared to strain PMSS1 is between 65 for PM21 and 87 for PM22 (Table 28). The affected genes vary from strain to strain but hotspots with more than 3 SNPs in one gene were observed for topA, infC, yejF, acsA, virB10, rfaI, ssb and engB (Table 28). Noticeable are an inframe deletion in PM21 and a frameshift insertion in SS1 at the gene virB10 (cagY); (Table A.51). Further, there are 44 mutations in the same way in all strains (PM21, PM22 and SS1); (Figure 16), 41 of these SNPs are in the engB region. Notably, the published PMSS1 strain has these mutations too. The reason could be that the assembled PMSS1 strain (10) is not from the same colony that were used for inoculation of the mice at NYU (Figure 6). The protein EngB is necessary for normal cell division. As concluded in the work of Draper et al. [40], the *H. pylori* strains have a high variability between and within one single colony including large inversions and variations in cagA copy number from 1 to 4. The two SS1 strains (the published one and the assembled one in this project) are closely related (Figure 17) and the published PMSS1 genome has few mutations to most of the other strains (Figure 18). It is difficult to distinguish modifications with time the strains are present in the mouse due to the high variability and the low number of strains that were sequenced with PacBio technology.

4.3 Conclusions

To sum up, the overall performances of the different assembler, MaSuRCA and CABOG are ranked the best but lowest considering reference coverage. Newbler performed very well according to assembly accuracy especially when dealing with Illumina MiSeq data and is an appropriate alternative to MaSuRCA even though Illumina sequence data is used.

In fact it is also important to analyse the computational cost of the assembler but no information of this parameter is available in GAGE-B paper; however, this is in general not relevant if you work with small genomes like bacteria, which have a genome size in the range of a few mega-bases. In GABenchToB DBG-assemblers are much quicker than OLC-assemblers.

The String Graph Assembler (SGA) was developed especially to reduce runtime by using memory efficient data structures to assemble mammalian-sized genomes [50]. SGA is not appropriate for the assembly of small genomes and it is not surprising that it achieved the lowest score.

A drawback in this study is that only Illumina sequencing runs are used and it is possible that the assembler perform quite differently on different data types especially in matters of read length and read coverage. MaSuRCA, for example, can assemble data sets containing a mixture of short reads and long reads (Sanger, 454, PacBio and Nanopore) [49], while Newbler was developed specifically for assembly of sequence data generated by the 454 GS-series of pyrosequencing platforms [28]. In the GABenchToB study [32] it was concluded, that Newbler produces high rates of mis-assemblies when dealing with MiSeq data, but this can not be confirmed in this thesis.

According to the hybrid error correction of the *H. pylori* PacBio reads using Illumina reads it could be shown that the recently published and fast correction tools LoRDEC and Jabba that both build a de Bruijn Graph of the short reads can not produce enough and long enough reads for an assembly. Proovread is a slower alternative and if the Illumina adapter sequences are trimmed, assemblies could be build with Canu and Newbler. Newbler build more contigs (> 20) while Canu produced less contigs (2-9) but the assemblies have more missing regions compared to the reference genome. An accurate, annotated and fully closed assembly using PacBio sequenced long reads of the *Helicobacter pylori* strains could be only achieved with a stand-alone approach

using the new Canu long-read assembler. To achieve a circular genome, Circlator had to be applied to the Canu generated sequences. The Canu assemblies still contain hundreds of SNPs and indels that could be reduced by mapping the accurate short Illumina reads and build the consensus.

The stand-alone assembly with Canu is a very fast and suitable assembly approach because a PacBio read coverage of 30x is sufficient and it is much faster than correcting PacBio reads by mapping of Illumina reads (Hybrid approach), but post-processing steps for polishing including short-read data are necessary and small plasmids are sometimes not assembled.

In this thesis it could be shown that *Helicobacter pylori* is a bacterium that has a high genomic variability. There are different numbers of the cagA region in all the four strains and a huge inversion of 400 kbp in strain PM21. A quite high number of SNPs are present but they do not necessarily increase with the duration that isolates were present in the mouse. As the *H. pylori* strains PMSS1 and SS1 have high genomic variability within one single strain [40], it is hard to determine one representative genome for a specific strain and creating a phylogeny that shows the modification in time is almost impossible. Future analyses should include the determination of the exact cagA copy number in the genomes and the generation of a SNP based phylogeny based on all the 46 strains sequenced with Illumina to get an overview of the mutations of PMSS1 with time in the mouse and the mutations in virulence factors.

5 References

[1] Taylor D, Heyer L, Campbell A, Denham S and Wessner D: **PHAST (Phage Assembly Suite and Tutorial): A Web-Based Genome Assembly Teaching Tool**. Master's thesis, Davidson College, Davidson, North Carolina, USA, 2012.

[2] Campbell A and Heyer L: **How are Genomes Sequenced. In: Discovering Genomics, Proteomics and Bioinformatics.**, volume 2. Benjamin Cummings, San Francisco, USA, 2007.

[3] Perkel J: **Sanger Who? Sequencing the next generation.** *Science* 2009. 324(5924):275–279.

[4] Sanger F, Nicklen S and Coulson A: **DNA sequencing with chain-terminating inhibitors.** *Proceedings of the National Academy of Sciences of the United States of America* 1977. 74(12):5463–5467.

[5] Pop M, Canzar S, Liu X, Su Q, Puiu D, Tallon L and Salzberg S: **Genome assembly reborn: recent computational challenges.** *Briefings in Bioinformatics* 2009. 10(4):354–366.

[6] Buermans H and den Dunnen J: **Next generation sequencing technology: Advances and applications**. *Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease* 2014. 1842(10):1932 – 1941.

[7] **Pacific Bioscience**. 2017. URL `http://www.pacb.com`. Last visited 2017-01-01.

[8] Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG and et al.: **Accurate Whole Human Genome Sequencing using Reversible Terminator Chemistry.** *Nature* 2008. 456(7218):53–59.

[9] 3402 Bioinformatics Group: **Illumina Illustration**. 2017. URL `http://www.3402bioinformaticsgroup.com/service/`. Last visited 2017-06-06.

[10] Illumina Technology: **Mate Pair Sequencing**. 2016. URL `http://www.illumina.com/technology/next-generation-sequencing/mate-pair-sequencing_assay.html`. Last visited 2016-07-25.

[11] Mardis E: **Next-Generation Sequencing Platforms**. *Annual Review of Analytical Chemistry* 2013. 6:287–303.

[12] Eid J, Fehr A, Gray J, Luong K, Lyle J, Otto G and et al.: **Real-time DNA sequencing from single polymerase molecules.** *Science* 2009. 323(5910):133–138.

[13] Pacific Bioscience: **SMRT Sequencing**. 2017. URL `http://www.pacb.com/smrt-science/smrt-sequencing/`. Last visited 2017-01-02.

[14] Hunt M, Silva N, Otto T, Parkhill J, Keane J and Harris S: **Circlator: automated circularization of genome assemblies using long sequencing reads.** *Genome Biology* 2015. 16(294).

[15] Hackl T, Hedrich R, Schultz J and Förster F: **proovread: large-scale high-accuracy PacBio correction through iterative short read consensus.** *Bioinformatics* 2014. 30(21):3004–3011.

[16] Salmela L, Walve R, Rivals E and Ukkonen E: **Accurate self-correction of errors in long reads using de Bruijn graphs**. *Bioinformatics* 2017. 33(6):799–806.

[17] Koren S, Walenz BP, Berlin K, Miller JR, Bergman NH and Phillippy AM: **Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation**. *Genome Research* 2017. 27:722–736.

[18] Au K, Underwood J, Lee L and Wong W: **Improving PacBio Long Read Accuracy by Short Read Alignment**. *PLoS ONE* 2012. 7(10):e46679.

[19] Koren S, Schatz MC, Walenz J, B. P. amd Martin, Howard J, Ganapathy G, Phillippy AM and et al.: **Hybrid error correction and de novo assembly of single-molecule sequencing reads.** *Nature Biotechnology* 2012. 30(7):693–700.

[20] Salmela L and Rivals E: **LoRDEC: accurate and efficient long read error correction.** *Bioinformatics* 2014. 30(24):3506–3514.

[21] Miclotte G, Heydari M, Demeester P, Rombauts S, Van de Peer Y, Audenaert P and Fostier J: **Jabba: hybrid error correction for long sequencing reads.** *Algorithms for Molecular Biology* 2016. 11:10.

[22] Ye L, Hillier LW, Minx P, Thane N, Locke DP, Martin JC, Chen L, Mitreva M, Miller JR, Haub KV *et al.*: **A vertebrate case study of the quality of assemblies derived from next-generation sequences**. *Genome Biology* 2011. 12(3):R31.

[23] Kim S, Tang H and Mardis ER: **Genome sequencing technology and algorithms**. Artech House, MA, USA, 2007.

[24] Miller J, Koren S and Sutton G: **Assembly algorithms for next-generation sequencing data**. *Genomics* 2010. 95(6):315–327.

[25] Baker M: **De novo genome assembly: what every biologist should know**. *Nature Methods* 2015. 9:333–337.

[26] Myers E: **Toward simplifying and accurately formulating fragment assembly**. *Journal of Computational Biology* 1995. 2:275–290.

[27] Schatz MC, Delcher AL and Salzberg SL: **Assembly of large genomes using second-generation sequencing**. *Genome Research* 2010. 20(9):1165–1173.

[28] 454 Life Sciences: **GS De Novo Assembler**. 2016. URL `http://www.454.com/products/analysis-software`. Last visited 2016-07-24.

[29] Miller JR, Delcher AL, Koren S, Venter E, Walenz BP, Brownley A, Johnson J, Li K, Mobarry C and Sutton G: **Aggressive assembly of pyrosequencing reads with mates**. *Bioinformatics* 2008. 24(24):2818–2824.

[30] Atallah M: **Algorithms and Theory of Computation Handbook**. CRC Press LLC, Purdue University, Lafayette, USA, 1999.

[31] Magoc T, Pabinger S, Canzar S, Liu X, Su Q, Puiu D, Tallon LJ and Salzberg SL: **GAGE-B: an evaluation of genome assemblers for bacterial organisms**. *Bioinformatics* 2013. 29(14):1718–1725.

[32] Junemann S, Prior K, Albersmeier A, Albaum S, Kalinowski J, Goesmann A and et al.: **GABenchToB: A Genome Assembly Benchmark Tuned on Bacteria and Benchtop Sequencers.** *PLoS ONE* 2014. 9:e107014.

[33] Gurevich A, Saveliev V, Vyahhi N and Tesler G: **QUAST: quality assessment tool for genome assemblies**. *Bioinformatics* 2013. 29(8):1072–1075.

[34] Ernst P and Gold B: **The disease spectrum of *Helicobacter pylori*: the immunopathogenesis of gastroduodenal ulcer and gastric cancer.** *Annual Review of Microbiolgy.* 2000. 54:615–640.

[35] Kienesberger S, Cox LM, Livanos A, Zhang XS, Chung J, Perez-Perez GI, Blaser MJ and et al.: **Gastric *Helicobacter pylori* infection affects local and distant microbial populations and host responses.** *Cell Reports* 2016. 14(6):1395–1407.

[36] Leunk R, Johnson P, David B, Kraft W and Morgan D: **Cytotoxic activity in broth-culture filtrates of *Campylobacter pylori***. *Jounal Medical Microbiology* 1988. 26(2):93–99.

[37] Blaser M, Perez-Perez G, Kleanthous H, Cover T, Peek R, Chyou P, Stemmermann G and Nomura A: **Infection with *Helicobacter pylori* Strains Possessing cagA Is Associated with an Increased Risk of Developing Adenocarcinoma of the Stomach**. *Cancer Research* 1995. 55:2111–15.

[38] Telford J, Ghiara P, Dellorco M, Comanducci M, Burroni D and et al.: **Gene structure of the *Helicobacter pylori* cytotoxin and evidence of its key role in gastric disease.** *Journal Experimental Medicine* 1994. 179:1653–58.

[39] Arnold IC, Dehzad N, Reuter S, Martin H, Becher B, Taube C and Müller A: ***Helicobacter pylori* infection prevents allergic asthma in mouse models through the induction of regulatory T cells.** *The Journal of Clinical Investigation* 2011. 121(8):3088–3093.

[40] Draper J, Hansen L, Bernick D, Abedrabbo S, Underwood J, Kong N and et al.: **Fallacy of the Unique Genome: Sequence Diversity within Single *Helicobacter pylori* Strains**. *mBio* 2017. 8(1):e02321–16.

[41] **RefSeq: NCBI Reference Sequence Database**. 2016. URL https://www.ncbi.nlm.nih.gov/refseq/. Last visited 2017-06-08.

[42] **NCBI: Sequence Read Archive (SRA)**. 2016. URL http://www.ncbi.nlm.nih.gov/sra. Last visited 2016-08-01.

[43] Illumina Technology: ***B. cereus*** **MiSeq data**. 2016. URL http://www.illumina.com/systems/miseq/scientific_data.html. Last visited 2016-08-01.

[44] University of Maryland, Institute for Genome Sciences: **DRASearch**. 2017. URL https://trace.ddbj.nig.ac.jp/DRASearch/query?center_name=UMIGS. Last visited 2017-09-13.

[45] Aronesty E: **Ea-utils: command-line tools for processing biological sequencing data.** 2011. URL https://expressionanalysis.github.io/ea-utils/. Last visited 2016-08-05.

[46] Magoc T, Pabinger S, Canzar S, Liu X, Su Q, Puiu D, Tallon LJ and Salzberg SL: **GAGE-B website**. 2016. URL ccb.jhu.edu/gage_b. Last visited 2016-08-01.

[47] Simpson JT, Wong K, Jackman SD, Schein JE, Jones SJ and Birol **ABySS: A parallel assembler for short read sequence data**. *Genome Research* 2009. 19(6):1117–1123.

[48] Chevreux B, Pfisterer T, Drescher B, Driesel AJ, Müller WEG, Wetter T and Suhai S: **Using the MiraEST Assembler for Reliable and Automated mRNA Transcript Assembly and SNP Detection in Sequenced ESTs.** *Genome Research* 2004. 14(6):1147–1159.

[49] Zimin A, Marçais G, Puiu D, Roberts M, Salzberg SL and Yorke JA: **The MaSuRCA genome Assembler**. *Bioinformatics* 2013. 29(21):2669–2677.

[50] Simpson JT and Durbin R: **Efficient de novo assembly of large genomes using compressed data structures**. *Genome Research* 2012. 22(3):549–556.

[51] Luo R, Liu B, Xie Y, Li Z and Huang Wea: **SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler.** *GigaScience* 2015. 1:18.

[52] Bankevich A, Nurk S, Antipov D, Gurevich AA and Dvorkin Mea: **SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing.** *Journal of Computational Biology* 2012. 19(5):455–477.

[53] Zerbino D and Birney E: **Velvet: Algorithms for de novo short read assembly using de Bruijn graphs.** *Genome Research* 2008. 18(5):821–829.

[54] Delcher AL, Phillippy A, Carlton J and Salzberg SL: **Fast algorithms for large-scale genome alignment and comparison**. *Nucleic Acids Research* 2002. 30(11):2478–2483.

[55] Lee A, O'Rourke J, De Ungria M, Robertson B, Daskalopoulos G and Dixon M: **A standardized mouse model of *Helicobacter pylori* infection: Introducing the Sydney strain**. *Gastroenterology* 1997. 112(4):1386–1397.

[56] Martin M: **Cutadapt removes adapter sequences from high-throughput sequencing reads**. *EMBnet.journal* 2011. 17(1):10–12.

[57] Bolger AM, Lohse M and Usadel B: **Trimmomatic: A flexible trimmer for Illumina Sequence Data.** *Bioinformatics* 2014. 30(15):2114–2120.

[58] Li H and Durbin R: **Fast and accurate short read alignment with Burrows–Wheeler transform. Bioinformatics**. *Bioinformatics* 2009. 25(14):1754–1760.

[59] Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G and Durbin R: **The Sequence alignment/map (SAM) format and SAMtools.** *Bioinformatics* 2009. 25(16):2078–2079.

[60] Altschul S, Gish W, Miller W, Myers E and Lipman D: **Basic local alignment search tool.** *Journal of Molecular Biology* 1990. 215(3):403–410.

[61] DePristo M, Banks E, Poplin R, Garimella K, Maguire J, Hartl C, Philippakis A, del Angel G and et al.: **A framework for variation discovery and genotyping using next-generation DNA sequencing data**. *Nature Genetics* 2011. 43(5):491–498.

[62] Broad Institute: **Picard**. 2017. URL `https://broadinstitute.github.io/picard/`. Last visited 2017-04-14.

[63] Thorvaldsdóttir H, Robinson JT and Mesirov JP: **Integrative genomics viewer (igv): high-performance genomics data visualization and exploration**. *Briefings in Bioinformatics* 2013. 14(2):178–192.

[64] García-Alcalde F, Okonechnikov K, Carbonell J, Cruz LM, Götz S, Tarazona S, Dopazo J, Meyer TF and Conesa A: **Qualimap: evaluating next-generation sequencing alignment data**. *Bioinformatics* 2012. 28(20):2678–2679.

[65] Hyatt D, Chen GL, LoCascio PF, Land ML, Larimer FW and Hauser LJ: **Prodigal: prokaryotic gene recognition and translation initiation site identification**. *BMC Bioinformatics* 2010. 11(1):119.

[66] Van Domselaar GH, Stothard P, Shrivastava S, Cruz JA, Guo A, Dong X and et al.: **BASys: a web server for automated bacterial genome annotation.** *Nucleic Acids Research* 2005. 33(Web Server issue):W455–W459.

[67] Darling AE, Mau B and Perna NT: **progressiveMauve: Multiple Genome Alignment with Gene Gain, Loss and Rearrangement**. *PLoS ONE* 2010. 5(6):e11147.

[68] Cingolani P, Platts A, Coon M, Nguyen T, Wang L, Land S, Lu X and Ruden D: **A program for annotating and predicting the effects of single nucleotide polymorphisms, snpeff: Snps in the genome of *Drosophila melanogaster* strain w1118; iso-2; iso-3**. *Fly* 2012. 6(2):80–92.

[69] Stamatakis A: **RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies.** *Bioinformatics* 2014. 30(9):1312–1313.

[70] R Core Team: **R: A Language and Environment for Statistical Computing**. R Foundation for Statistical Computing, Vienna, Austria, 2015. URL `http://www.R-project.org/`. Last visited 2016-09-25.

[71] Pages H, Aboyoun P, Gentleman R and DebRoy S: **Biostrings: String objects representing biological sequences, and matching algorithms**, 2016. R package version 2.38.4.

[72] Wickham H: **stringr: Simple, Consistent Wrappers for Common String Operations.**, 2015. URL `http://CRAN.R-project.org/package=stringr`. R package version 1.0.0. Last visited 2016-09-25.

[73] Charif D and Lobry J: **SeqinR 1.0-2: a contributed package to the R project for statistical computing devoted to biological sequences retrieval and analysis.** In U Bastolla, M Porto, H Roman and M Vendruscolo, editors, **Structural approaches to sequence evolution: Molecules, networks, populations**, Biological and Medical Physics, Biomedical Engineering, pages 207–232. Springer Verlag, New York, USA, 2007.

[74] Wickham H: **The Split-Apply-Combine Strategy for Data Analysis**. *Journal of Statistical Software* 2011. 40(1):1–29.

[75] Hlavac M: **stargazer: LaTeX/HTML code and ASCII text for well-formatted regression and summary statistics tables**. Harvard University, Cambridge, USA, 2014. URL `http://CRAN.R-project.org/package=stargazer`. Last visited: 2016-10-06.

[76] Morgan M, Anders S, Lawrence M, Aboyoun P, Pagès H and Gentleman R: **ShortRead: a Bioconductor package for input, quality assessment and exploration of high-throughput sequence data**. *Bioinformatics* 2009. 25:2607–2608.

[77] Obenchain V, Lawrence M, Carey V, Gogarten S, Shannon P and Morgan M: **VariantAnnotation: a Bioconductor package for exploration and annotation of genetic variants**. *Bioinformatics* 2014. 30(14):2076–2078.

[78] Paradis E, Claude J and Strimmer K: **APE: analyses of phylogenetics and evolution in R language**. *Bioinformatics* 2004. 20(2):289–290.

[79] NCBI: ***Helicobacter pylori* PMSS1 assembly**. 2017. URL `https://www.ncbi.nlm.nih.gov/assembly/GCF_001991095.1`. Last visited 2017-04-13.

[80] **Ion PGM™ System for Next-Generation Sequencing**. 2016. URL `https://www.thermofisher.com/at/en/home/life-science/sequencing/next-generation-sequencing/ion-torrent-next-generation-sequencing-workflow/ion-torrent-next-generation-sequencing-run-sequence/ion-pgm-system-for-next-generation-sequencing.html`. Last visited 2016-08-08.

[81] Luo C, Tsementzi D, Kyrpides N, Read T and Konstantinidis KT: **Direct Comparisons of Illumina vs. Roche 454 Sequencing Technologies on the Same Microbial Community DNA Sample**. *PLoS ONE* 2012. 7(12):e30087.

[82] Chakraborty M, Baldwin-Brown J, Long A and Emerson J: **A practical guide to de novo genome assembly using long reads**. *bioRxiv* 2015. doi:https://doi.org/10.1101/029306.

[83] Haeyoung J, Dae-Hee L, Choong-Min R and Seung-Hwan P: **Toward Complete Bacterial Genome Sequencing Through the Combined Use of Multiple Next-Generation Sequencing Platforms**. *Journal of Microbiology and Biotechnology* 2017. 26(1):207–212.

[84] Phillippy A, Koren S and Walenz B: **canu Documentation**. 2017. URL `https://media.readthedocs.org/pdf/canu/latest/canu.pdf`. Last visited 2017-06-14.

6 Appendix

6.1 Implemented R Functions

Following the name of all implemented R functions, as well as a description and input parameter.

- `sort_contigs(reads)`
  - description: sequences of fasta file are sorted by length and assigned with numbers. The longest sequence has the number 1, second number 2 and so on. It returns the sorted `DNAStringSet`.
  - input parameter:
    - reads: a `DNAStringSet` object containing reads/contigs.

- `detail_table(table)`
  - description: the function computes for every contig the `Number_of_Aligned_Contigs` as well as the name/number of them (`Mapped_Contigs`), how many bases cannot be matched (`Missing_Bases`) and how long the overlap is (`Overlap_Bases`). Here every base which is aligned more than once to the reference is counted. So more than 100% overlap is possible.
  - input parameter:
    - table: a Table that was created by `create_table_of_delta()` function

- `remove_overlaps(table)`
  - description: filters nucmer alignment results. Alignment matches, where the Reference region in that contig as well as the Query region is fully covered by other matches are excluded. Therefore the object class *IRanges* from the R package *Biostrings* as well as the function `findOverlaps` was used. It returns the filtered Table in the same format as the input table.

  - input parameter:
    - table: a Table that was created by `create_table_of_delta()` function

- `not_mapped_contigs(table)`
  - description: returns a data frame with the information on the ranges of not mapped contigs. Ranges that are not in the alignment are determined using `setdiff()` from package *Biostrings*.
  - input parameter:
    - table: a Table that was created by `create_table_of_delta()` function

- `read_splitting(input,overlap)`
  - description: split reads if they are longer than 1999 bp into subreads with overlap length specified. Therefore the functions `readFasta` from the package ShortRead and `substring` from the package Biostrings are used. The id of the reads are normalized, starting with an R, following underline, read number, underline, start point, underline end point.
  - input parameter:
    - input: fasta file name
    - overlap: overlap length of split reads (integer)

- `create_dataset(dna_string_set,strain_name)`
  - description: creates a `data.frame()` with basic statistics on the read set. It computes number of reads, number of bases, average read length, minimum and maximum read length.
  - input parameter:
    - dna_string_set: a *DNAStringSet* object
    - strain_name: name of the bacteria (string)

- `histogram(dna_file,strain_name)`
  - description: creates a histogram that shows the read length distribution of a fastq file.
  - input parameter:
    - dna_file: a fastq file name
    - strain_name: name of the title

- `coverage(number_bases, ref_seq)`
  - description: computes the coverage of the reference sequence.
  - input parameter:
    - number_bases: integer
    - ref_seq: reference sequence in fasta format

- `N50(contigs)`
  - description: computes the N50 value of a read set. N50 is the shortest contig length at 50% of the genome size.
  - input parameter:
    - contigs: list of the length of each contig

- `ill_trim(ill_raw_1, ill_raw_2)`
  - description: the raw Illumina paired files are used as input, the software *trimmomatic* is invoked and the trimmed paired files are returned. Reads are trimmed if the quality falls below 3. No adapters are cut off.
  - input parameter:
    - : ill_raw_1: raw Illumina paired-end read file 1.
    - : ill_raw_2: raw Illumina paired-end read file 2.

- `assembly_statistic(assembly_statistics_list,name_list, ref_seq, format)`
  - description: computes a dataframe with standard statistics. The files of `assembly_statistics_list` are read in and function `create_dataset`, as well as `coverage` and `N50` is invoked to compute the statistic.
  - input parameter:
    - assembly_statistics_list: a list of file names used for statistics
    - name_list: names of the strains
    - ref_seq: reference sequence in fasta format
    - format: "f" for FASTA files and "fq" for FASTQ files.

- `get_one_short_read_set(read_set_list)`
  - description: reads in a set of fasta files with `readFasta` and returns one `ShortRead` object.
  - input parameter:
    - read_set_list: a list of fasta file names

- `get_one_short_read_set_q(read_set_list)`
  - description: reads in a set of fastq files with `readFastq` and returns one `ShortReadQ` object.
  - input parameter:
    - read_set_list: a list of fastq file names

- `rearrange_middle(read_list)`
  - description: rearrange assemblies with ends in the middle and writes the assembly in fasta format.
  - input parameter:
    - read_list: list of assembly fasta file names

- `introduce_cagA(input,input_name, ref_file)`
  - description: it uses the cagA region of the PMSS1 ref file and matches this sequence to the input sequence with `matchPattern()` from package *Biostrings*. This function returns all the positions of this pattern with maximum of two mismatches or gaps. It then introduces one more copy of the cagA gene at the end of the first cagA region and writes the result to a file with extension *[cagA number]xcagA.fasta*.
  - input parameter:
    - input: name of the genome file
    - input_name: name of the genome
    - ref_file: reference PMSS1 file in fasta format

- `rearrange_assembly(assembly_name, ref_seq)`
  - description: uses the first 300 bp of the reference assembly to find the start position in the assembly and rearrange it. It uses the `pairwiseAlignment()` function from *Biostrings* package to find this position. If the sequence is in the complementary strain it uses the function `reverseComplement()` from package *ShortRead* to get the same strain as the reference strain.
  - input parameter:
    - assembly_name: name of the assembly file to rearrange
    - ref_seq: reference sequence in fasta format

- `gatk(input, ill_1,ill_2)`
  - description: the gatk function includes
    - creates reference index for bwa, with *bwa index*
    - creates reference dictionary with *picard.jar CreateSequenceDictionary*
    - creates reference index for GATK with *samtools faidx*
    - performs *bwa mem* with defined read groups with the fasta file and the illumina reads for mapping.

    * converts result in bam, sort and index bam files with *bwa view/-sort/index*
    * calls *GATK HaplotypeCaller* with haploid ploidy to find potential variant alleles and creates a vcf file.
    * *GATK FastaAlternateReferenceMaker* is invoced to build the consensus sequence based on the vcf file.
    * returns the consensus sequence.
  - input parameter:
    * input: assembled, rearranged fasta file
    * ill_1: trimmed paired Illumina reads
    * ill_2: trimmed paired Illumina reads

- `prodigal(name_strain, gatk_consensus_file)`
  - description: invokes *prodigal* and saves a protein description gff file, an additional gbk formatted file as well as a gene locus file. The gff file is converted to a text file with the function `convertProdigalToBasys()` that can be used for BASys annotation.
  - input parameter:
    * name_strain: name that should be used in output files
    * gatk_consensus_file: input file used for protein prediction

- `create_table_of_delta(table)`
  - description: creates a demonstrative data frame of a delta file. Columns include the names of the contigs, the length, the start and the end positions and the number of alignment errors including mismatches and gaps. R stands for the Reference genome and Q for the assembly to compare.
  - input parameter:
    * table: a delta file data frame that is read in with `read.table()`

- `nucmer_compare(name_strain, input, ref_file)`
  - description: an assembly is aligned against a reference file and the multiple aligner *nucmer –maxmatch* is invoked. After applying the *delta filter* the delta file is read in and the own function `create_table_of_delta()` is invoked. The obtained data frame shows all the ranges that have overlaps with the reference. The ranges that have no overlaps are determined using the provided function `setdiff()` from the package *Biostrings*. The return value is a list where the first entry is the Table obtained from the `create_table_of_delta` function. The second entry is a Table with the ranges of the reference that are not covered by the query.
  - input parameter:
    - name_strain: name of the strain used to name the delta file
    - input: input fasta file name that should be aligned
    - ref_file: name of the reference file

- `ill_trim_cut(ill_raw_1, ill_raw_2, ill_seq)`
  - description: the Illumina reads are trimmed and adapter sequences provided with the `ill_seq` fasta file are cut from the read with the software *Trimmomatic*. The results are four files, one pair of paired reads and one including unpaired reads.
  - input parameter:
    - ill_raw_1: forward Illumina paired-end file to trim
    - ill_raw_2: reverse Illumina paired-end file to trim
    - ill_seq: all possible adapter sequences in fasta format

- `ill_trim_cutadapt(ill_raw_1, ill_raw_2, ill_seq)`
  - description: the adapter sequences provided with the ill_seq fasta file are cut from the reads with the software *Cutadapt*. The forward and reverse trimmed fasta files are returned.

  - input parameter:
    * ill_raw_1: forward Illumina paired-end file to trim
    * ill_raw_2: reverse Illumina paired-end file to trim
    * ill_seq: all possible adapter sequences in fasta format

- `reference_splitting(input_file)`
  - description: The `DNAString` object of a reference fasta file is split into pieces with a random length between 1500 and 1999 bp. Therefore the functions `sample()` and `substring()` from the package *Biostrings* are used. The whole reference genome is split 40 times to achieve a coverage of 40. This function conduces to control the assembly process of Canu and Newbler.
  - input parameter:
    * input_file: reference fasta file that has to be split

- `snps_statistic(vcf_file,fasta_file,name_strain)`
  - description: reads in an annotated vcf-file using readVcf from package *VariantAnnotation*. The return value is a list with the first entry showing a detail Table with number of variants, effected genes, snps, indels, deletions, missense mutations, synonymous mutations and the number of nonsense mutations. The second Table is a detail Table showing all the variants positions, the reference bases of strain PMSS1, the actual (alternative) bases, the kind of effect and the gene/protein if the variant is in an exon region.
  - input parameter:
    * vcf_file: an annotated vcf file
    * fasta_file: the reference fasta file
    * name_strain: name of the strain

6.2 Assembly Tools for Small Genomes

Following is a summary of the assembly results for all of the twelve data sets used in this study compared with the reference genome (Tables A.1 - A.12).

The performance of each of the assembler compared with Newbler assembler on all of the twelve data sets is summarised in Tables A.17 - A.27. The individual rank of each assembler across all nine bacteria is shown in Figures A.13 - A.16.

6.3 *Helicobacter Pylori*

Detail statistics of LoRDEC and Jabba error correction on PM21 PacBio read files is shown in Table A.28. Statistic of PMSS1 Illumina reads after adapter removal by different tools is shown in Table A.29. PacBio read statistic for strain PMSS1 after hybrid error correction with proovread depending on the different short read adapter removal software tools is shown Tables A.30 and A.31, and the assembly statistics after Canu and Newbler in Tables A.32 - A.35. For missing ranges of the assembly see Tables A.36 - A.50.

Table A.1: *R .sphaeroides* MiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	846	282	148	885	63	1,032	275	655	441	7
Total Length	4,567,617	4,567,635	3,952,878	4,662,776	4,247,935	4,621,116	4,618,855	4,537,774	4,554,476	4,602,977
Minimum Contig Length	49	18	172	85	318	112	66	61	169	37,100
Maximum Contig Length	71,578	118,527	127,844	67,456	241,348	44,874	286,217	71,713	115,051	3,188,524
N50 Value	21,558	36,722	41,794	15,445	142,742	9,086	118,093	24,300	33,829	3,188,524
Reference Covered by Contigs %	99.68	99.94	86.48	99.81	92.57	99.43	99.98	99.25	98.9	
Bases Overlapping Query %	4.63	2.97	2.48	3.45	1.8	5.62	5.5	3.84	3.28	
Mismatches and Indels %	0.28	0.09	0.17	0.2	0.1	0.26	0.23	0.27	0.11	
Bases not Mapping	14,693	2,723	622,292	8,917	342,146	26,338	994	34,355	50,819	
Contigs Covered by Reference %	99.97	99.99	99.98	100	100	99.98	99.7	100	99.86	
Bases Overlapping Reference %	4.11	2.23	1.51	4.98	1.38	6.6	5.61	3.11	3.15	

Table A.2: *R. sphaeroides* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	1,557	1,014	539	1,581	130	12,186	350	798	7,436	7
Total Length	4,633,465	4,548,770	4,216,057	4,725,855	4,489,204	5,704,978	4,670,077	4,525,236	5,220,141	4,602,977
Minimum Contig Length	31	4	101	45	301	101	56	97	100	37,100
Maximum Contig Length	66,408	41,908	41,321	110,511	358,962	75,323	291,333	59,062	45,522	3,188,524
N50 Value	13,002	10,095	12,421	17,052	176,783	8,907	74,486	13,775	8,960	3,188,524
Reference Covered by Contigs %	99.92	99.87	91.93	99.98	97.83	96.01	99.99	99.02	99.43	
Bases Overlapping Query %	4.11	3.15	2.37	5.15	4.08	24.17	5.04	4.04	5.09	
Mismatches and Indels %	0.31	0.16	0.18	0.33	0.27	1.04	0.25	0.25	0.36	
Bases not Mapping	3,892	5,780	371,652	803	99,810	183,509	620	45,223	26,383	
Contigs Covered by Reference %	99.69	99.97	99.98	99.98	100	99.9	99.44	99.98	99.96	
Bases Overlapping Reference %	4.57	2.03	1.82	7.96	3.67	57.8	6.03	3.24	19.69	

Table A.3: *V. cholerae* MiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	564	242	241	501	173	1,901	1,486	209	312	2
Total Length	3,959,969	3,945,419	3,953,298	4,112,140	4,034,988	4,357,240	4,451,154	3,960,394	3,954,533	4,033,464
Minimum Contig Length	63	1	1,007	122	357	87	100	95	101	1,072,315
Maximum Contig Length	178,118	285,908	140,691	450,326	255,146	105,420	741,022	246,346	246,179	2,961,149
N50 Value	60,973	136,901	33,710	112,926	76,131	23,501	246,623	92,036	71,357	2,961,149
Reference Covered by Contigs %	99.62	99.74	98.07	99.75	98.51	99.66	99.76	99.66	99.58	
Bases Overlapping Query %	14.9	5.11	3.72	6.7	3.15	22.77	6.24	5.02	6.36	
Mismatches and Indels %	0.9	0.3	0.22	0.3	0.18	1.16	0.34	0.23	0.41	
Bases not Mapping	15,419	10,656	77,920	10,073	60,192	13,563	9,782	13,834	16,770	
Contigs Covered by Reference %	99.98	99.67	99.85	99.54	99.97	99.83	90.11	99.84	99.79	
Bases Overlapping Reference %	13.18	2.77	3.45	8.58	4.67	32.81	6.56	3.31	4.49	

Table A.4: *V. cholerae* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	338	491	128	2,673	105	2,596	207	325	344	2
Total Length	4,115,424	3,922,846	3,855,763	4,337,203	3,994,617	4,171,725	3,955,126	3,939,751	3,969,747	4,033,464
Minimum Contig Length	51	1	163	33	304	70	100	97	100	1,072,315
Maximum Contig Length	280,277	153,905	256,726	316,666	555,664	108,154	214,711	157,783	425,556	2,961,149
N50 Value	94,508	47,372	61,249	87,069	241,604	22,871	83,518	40,877	135,118	2,961,149
Reference Covered by Contigs %	99.73	99.71	96.93	99.76	99.48	99.58	99.75	99.69	99.75	
Bases Overlapping Query %	11.66	5.09	3.42	6.61	3.79	76.48	7.67	7.67	8.12	
Mismatches and Indels %	0.6	0.36	0.25	0.4	0.23	4.68	0.48	0.44	0.44	
Bases not Mapping	10,903	11,594	124,000	9,548	20,885	16,892	9,967	12,659	9,936	
Contigs Covered by Reference %	99.9	99.88	99.97	99.59	99.97	99.97	99.89	99.98	99.92	
Bases Overlapping Reference %	14.11	2.39	1.92	14.45	3.27	82.89	5.72	5.45	6.57	

Table A.5: *H. fragilis* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	168	588	138	542	119	2,291	151	246	402	1
Total Length	7,041,280	5,300,437	5,214,282	5,433,200	5,396,385	5,508,253	5,332,156	5,289,907	5,314,517	5,373,121
Minimum Contig Length	83	1	141	35	318	56	96	145	100	5,373,121
Maximum Contig Length	419,237	203,398	363,813	378,585	391,846	188,622	534,462	286,976	384,109	5,373,121
N50 Value	106,769	79,346	95,840	134,263	158,716	41,205	157,732	125,214	125,627	5,373,121
Reference Covered by Contigs %	81.94	81.92	81.06	81.94	81.88	81.9	81.94	81.91	81.94	
Bases Overlapping Query %	2.03	2.18	1.4	2.92	1.94	8.9	1.93	2.06	2.18	
Mismatches and Indels %	0.8	0.82	0.86	0.88	0.84	1.1	0.82	0.82	0.82	
Bases not Mapping	970,573	971,486	1,017,555	970,126	973,536	970,573	970,258	972,062	970,368	
Contigs Covered by Reference %	82.9	82.04	83.12	81.38	82.27	81.7	82.09	82.53	82.49	
Bases Overlapping Reference %	29.3	1.17	1	3.31	2.7	10.9	1.44	1.39	1.82	

Table A.6: *A. hydrophila* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	174	458	113	1,314	32	2,186	339	103	216	1
Total Length	4,985,428	4,948,435	5,009,370	5,269,026	4,915,618	5,123,704	5,011,698	4,895,627	4,915,552	4,744,448
Minimum Contig Length	65	1	69	35	302	70	96	95	100	4,744,448
Maximum Contig Length	654,090	565,723	710,684	290,325	1,124,684	199,964	592,124	526,254	455,879	4,744,448
N50 Value	237,457	234,867	278,382	244,066	828,647	64,647	379,681	180,361	243,851	4,744,448
Reference Covered by Contigs %	89.16	89.08	88.09	89.17	88.76	89.1	89.17	89.1	89.15	
Bases Overlapping Query %	4.27	1.92	1.12	4.03	2.41	13.4	4.68	2.15	3.33	
Mismatches and Indels %	5.86	5.72	5.83	5.7	5.87	6.1	5.8	5.83	5.88	
Bases not Mapping	514,113	518,215	564,881	513,783	533,169	515,160	513,826	517,025	514,639	
Contigs Covered by Reference %	85.1	84.12	85.5	80.6	85.1	84.3	84	85.3	85.2	
Bases Overlapping Reference %	4.9	0.71	3.4	4.9	2	16.5	4.5	1.1	2.6	

Table A.7: *S. aureus* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	132	379	56	256	52	1,921	72	82	278	3
Total Length	3,040,616	2,825,765	2,796,508	2,890,706	2,896,852	3,024,925	2,861,974	2,829,571	2,856,225	2,903,081
Minimum Contig Length	65	1	1,278	35	300	69	96	153	76	3,125
Maximum Contig Length	244,957	173,047	285,328	379,101	449,789	173,010	350,174	336,190	364,183	2,872,915
N50 Value	64,963	51,397	111,165	132,448	221,821	35,817	187,080	122,507	146,332	2,872,915
Reference Covered by Contigs %	93.35	93.24	92.18	93.35	92.66	93.3	93.36	93.35	93.35	
Bases Overlapping Query %	5.69	4	4.87	6.51	4.33	19	4.56	3.86	7.13	
Mismatches and Indels %	1.46	1.41	1.56	1.79	1.45	2.1	1.47	1.49	1.5	
Bases not Mapping	193,023	196,264	227,080	192,970	213,091	193,883	192,901	193,006	192,972	
Contigs Covered by Reference %	91.9	94.1	95	93.6	92.85	94.3	94.2	94.5	94.4	
Bases Overlapping Reference %	8.9	2.2	4.1	6.3	4.31	24.9	4.1	2.5	6.5	

Table A.8: *X. axonopodis* HiSeq overview table

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	2,372	189	2,969	155	85,148	453	258	2,056	1
Total Length	5,086,459	4,848,222	6,588,629	4,935,414	13,567,484	5,014,024	4,913,541	5,187,613	4,967,469
Minimum Contig Length	49	66	56	307	100	86	125	100	4,967,469
Maximum Contig Length	196,813	312,406	352,831	322,669	132,038	312,376	307,295	196,882	4,967,469
N50 Value	87,649	105,803	66,966	117,883	102	117,467	83,025	72,978	4,967,469
Reference Covered by Contigs %	89.1	88.8	89.12	89.1	89	89.12	89.08	89.1	
Bases Overlapping Query %	2.5	1.51	15.23	2.8	45.9	2.59	1.46	8.6	
Mismatches and Indels %	5.4	5.63	5.98	5.7	5.9	5.53	5.52	6	
Bases not Mapping	543,642	556,307	540,468	541,377	545,104	540,273	542,603	540,581	
Contigs Covered by Reference %	86.8	90.9	71.9	89.7	38	88.1	89.47	85.6	
Bases Overlapping Reference %	2.4	1.4	26.4	2.7	139	2.3	0.84	9.3	

Table A.9: *M. abscessus* MiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	328	368	857	2,167	326	1,230	910	286	197	2
Total Length	5,149,082	5,167,890	5,116,733	5,922,392	5,418,490	5,380,709	5,412,372	5,161,555	5,152,802	5,090,491
Minimum Contig Length	58	1	423	54	382	92	100	193	100	23,319
Maximum Contig Length	245,660	182,170	40,668	279,306	104,386	59,847	498,363	226,629	286,460	5,067,172
N50 Value	70,424	55,820	8,655	81,728	36,211	12,516	215,400	47,327	131,561	5,067,172
Reference Covered by Contigs %	99.38	99.43	96.4	99.44	98.43	99.38	99.43	99.13	99.42	
Bases Overlapping Query %	3.69	0.63	0.39	1.1	0.25	2.34	0.55	1.18	2.63	
Mismatches and Indels %	0.09	0.02	0.05	0.07	0.03	0.13	0.03	0.06	0.05	
Bases not Mapping	31,401	29,206	183,126	28,767	79,862	31,587	29,025	44,250	29,646	
Contigs Covered by Reference %	98.3	97.96	98.54	97.66	98.45	98.4	93.8	98.08	98.22	
Bases Overlapping Reference %	3.79	0.65	3.04	15.47	6.63	7.12	0.88	1.51	2.67	

Table A.10: *M. abscessus* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	147	169	127	16,036	66	679	99	189	143	2
Total Length	5,134,453	5,132,927	5,116,091	7,760,947	5,146,013	5,183,701	5,139,836	5,136,597	5,150,322	5,090,491
Minimum Contig Length	57	5	1,196	25	327	74	100	97	116	23,319
Maximum Contig Length	610,290	653,368	238,341	338,337	375,149	101,254	627,055	191,465	474,277	5,067,172
N50 Value	119,446	125,716	81,416	104,848	246,830	28,533	150,258	60,955	148,639	5,067,172
Reference Covered by Contigs %	99.38	99.42	98.98	99.44	99.42	99.35	99.44	99.38	99.42	
Bases Overlapping Query %	3.2	2.83	0.86	1.4	0.18	2.65	1.53	2.49	2.12	
Mismatches and Indels %	0.1	0.04	0.09	0.18	0.07	0.13	0.06	0.06	0.04	
Bases not Mapping	31,694	29,650	52,120	28,742	29,673	33,028	28,703	31,723	29,255	
Contigs Covered by Reference %	98.47	98.39	98.49	97.27	98.49	98.49	98.45	98.45	98.27	
Bases Overlapping Reference %	3.19	2.65	0.88	51.01	0.33	3.65	1.52	2.48	2.14	

Table A.11: *B. cereus* MiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap	Reference
Contig Number	617	364	78	164	90	3,344	50,090	444	123	2
Total Length	5,431,491	5,360,290	5,419,995	5,422,635	5,440,292	6,180,355	17,888,992	5,325,180	5,352,193	5,432,652
Minimum Contig Length	49	1	425	35	300	149	86	58	130	208,369
Maximum Contig Length	430,487	151,588	258,489	311,769	766,822	106,317	346,945	91,844	606,530	5,224,283
N50 Value	130,570	51,065	155,352	116,480	246,697	22,044	251	24,577	246,346	5,224,283
Reference Covered by Contigs %	99.97	99.95	99.51	99.99	99.87	99.96	99.99	99.28	99.18	
Bases Overlapping Query %	6.49	2.13	1.37	4.31	5.32	101.08	16.19	3.45	3.13	
Mismatches and Indels %	0.34	0.07	0.04	0.11	0.1	0.89	5.22	0.16	0.14	
Bases not Mapping	1,681	2,457	26,649	771	7,096	2,263	778	39,266	44,568	
Contigs Covered by Reference %	99.88	99.9	99.96	99.96	99.96	99.94	80.85	99.95	99.96	
Bases Overlapping Reference %	6.43	0.72	1.59	4.1	5.58	128.8	219.69	2.07	2.36	

Table A.12: *B. cereus* HiSeq overview table

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SPAdes	Velvet	Soap	Reference
Contig Number	894	1,212	164	687	259	1,127	527	3,161	2
Total Length	5,879,341	5,614,401	4,797,470	5,670,388	5,640,375	5,962,659	5,016,338	5,991,251	5,432,652
Minimum Contig Length	65	1	476	75	96	76	62	100	208,369
Maximum Contig Length	257,581	184,282	387,299	524,448	397,830	459,337	267,159	268,115	5,224,283
N50 Value	38,893	28,984	79,405	43,373	97,165	90,150	42,292	48,403	5,224,283
Reference Covered by Contigs %	73.22	73.06	59.1	71.88	71.46	73.32	66.5	73.3	
Bases Overlapping Query %	6.22	2.18	2.7	1.88	3	4.14	3.89	29.8	
Mismatches and Indels %	7.63	7.64	7.7	7.79	7.89	7.58	7.83	8	
Bases not Mapping	1,454,867	1,463,431	2,223,123	1,527,588	1,550,507	1,449,234	1,819,713	1,452,551	
Contigs Covered by Reference %	68.9	70.3	67.8	70.6	69.9	67.5	72	69.6	
Bases Overlapping Reference %	7.9	1.7	3.1	3.6	4.1	5.2	3.5	36.2	

Table A.13: Sum of the ranks of all bacteria for each assembler for all parameters

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Contig Number	54	59	27	73	13	86	46	42	50
Maximum Contig Length	45	58	61	45	26	83	26	64	42
N50 Value	54	58	58	43	20	88	28	61	40
Reference Covered by Contigs %	41	45	88	18	74	60	14	58	47
Bases Overlapping Query %	70	36	18	58	26	86	53	43	59
Mismatches and Indels %	63	18	40	58	35	86	50	46	51
Contigs Covered by Reference %	53	65	21	66	27	47	76	32	49
Bases Overlapping Reference %	70	20	25	72	41	86	52	31	51
Sum of the Ranks	450	359	338	433	262	622	345	377	389
Global Rank	8	4	2	7	1	9	3	5	6

Table A.14: Sum of the ranks of all bacteria for each assembler for the most improtant quality parameters

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	41	45	88	18	74	60	14	58	47
Bases Overlapping Query %	70	36	18	58	26	86	53	43	59
Mismatches and Indels %	63	18	40	58	35	86	50	46	51
Contigs Covered by Reference %	53	65	21	66	27	47	76	32	49
Bases Overlapping Reference %	70	20	25	72	41	86	52	31	51
Sum of the Ranks	297	184	192	272	203	365	245	210	257
Global Rank	8	1	2	7	3	9	5	4	6

Table A.15: Sum of the ranks of the three MiSeq data (*R. sphaeroides*, *V. cholerae* and *M. abscessus*) for each assembler for all parameters

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Contig Number	18	13	11	23	5	26	18	10	11
Maximum Contig Length	18	13	20	13	13	26	3	16	13
N50 Value	18	11	20	14	13	26	4	16	13
Reference Covered by Contigs %	15	8	27	6	24	15	4	18	18
Bases Overlapping Query %	24	11	6	17	3	25	16	15	18
Mismatches and Indels %	25	6	11	17	6	25	14	17	14
Contigs Covered by Reference %	12	18	9	17	5	13	27	11	19
Bases Overlapping Reference %	20	5	10	23	13	26	16	9	13
Sum of the Ranks	150	85	114	130	82	182	102	112	119
Global Rank	8	2	5	7	1	9	3	4	6

Table A.16: Sum of the ranks of the three HiSeq data (*R. sphaeroides*, *V. cholerae* and *M .abscessus*) for each assembler for all parameters

	Abyss	Newbler	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Contig Number	16	18	8	25	3	25	7	15	18
Maximum Contig Length	12	17	21	12	7	22	10	21	13
N50 Value	13	18	19	13	3	27	9	20	13
Reference Covered by Contigs %	13	13	27	4	20	23	6	19	10
Bases Overlapping Query %	22	13	4	15	7	25	15	15	19
Mismatches and Indels %	21	5	10	20	11	26	15	11	16
Contigs Covered by Reference %	18	20	6	20	4	10	21	9	19
Bases Overlapping Reference %	19	10	4	24	8	26	14	12	18
Sum of the Ranks	134	114	99	133	63	184	97	122	126
Global Rank	8	4	3	7	1	9	2	5	6

Table A.17: *H. fragilis* HiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.03	98.02	99.6	99	99.08	99.22	99.01	99.06
Bases Overlapping Query %	1.48	1.31	1.49	1.19	3.17	1.34	1.23	1.44
Mismatches and Indels %	0.17	0.2	0.18	0.18	0.4	0.17	0.1	0.15
Bases not Mapping Query	3,104	1,911	17,526	1,996	20,142	2,656	1,368	3,809
Bases not Mapping Reference	51,428	104,828	21,265	53,266	48,754	41,224	52,511	50,040
Contigs Covered by Reference %	99.96	99.96	99.68	99.96	99.63	99.95	99.97	99.93
Bases Overlapping Reference %	35.73	1.66	4.1	3.99	7.76	2.67	2	2.59
Number of Identical Contigs	4	2	7	1	23	6	25	20

Table A.18: *A. hydrophila* HiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	98.69	98.52	99.77	98.66	98.73	99.11	98.68	98.71
Bases Overlapping Query %	3.36	1.8	2.3	1.77	3.41	2.21	2.96	1.64
Mismatches and Indels %	0.05	0.06	0.1	0.1	0.14	0.04	0.04	0.04
Bases not Mapping Query	4,828	1,281	224,831	1,858	29,209	55,073	2,866	4,227
Bases not Mapping Reference	64,585	73,183	11,215	66,372	62,887	44,009	65,483	63,993
Contigs Covered by Reference %	99.9	99.97	95.73	99.96	99.43	98.9	99.94	99.91
Bases Overlapping Reference %	5.35	4.54	4.62	2.4	7.77	3.29	3.14	2.18
Number of Identical Contigs	7	3	8	0	9	3	9	5

Table A.19: *S. aureus* HiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.57	97.91	99.83	98.59	99.59	99.68	99.47	99.61
Bases Overlapping Query %	1.73	1.66	1.34	0.87	3.96	1.25	1.25	1.2
Mismatches and Indels %	0.14	0.23	0.2	0.18	0.39	0.16	0.15	0.15
Bases not Mapping Query	3,201	1,527	10,064	1,092	27,382	4,839	953	2,264
Bases not Mapping Reference	12,080	58,986	4,749	39,874	11,619	8,960	15,068	10,956
Contigs Covered by Reference %	99.89	99.95	99.65	99.96	99.09	99.83	99.97	99.92
Bases Overlapping Reference %	9.77	2.66	3.46	4.77	10.73	2.68	1.88	2.59
Number of Identical Contigs	4	0	1	0	23	0	5	5

Table A.20: *M. abscessus* MiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.5	96.39	99.97	98.47	99.51	99.69	99.3	99.59
Bases Overlapping Query %	2.61	0.43	0.53	0.29	1.52	0.26	0.75	1.37
Mismatches and Indels %	0.06	0.05	0.06	0.02	0.1	0.02	0.03	0.04
Bases not Mapping Query	5,379	777	25,926	628	2,240	245,054	13,559	9,077
Bases not Mapping Reference	25,744	186,630	1,439	79,220	25,507	15,945	35,950	21,050
Contigs Covered by Reference %	99.9	99.98	99.56	99.99	99.96	95.47	99.74	99.82
Bases Overlapping Reference %	2.64	3.03	14.73	6.67	6.16	0.57	1.06	1.3
Number of Identical Contigs	11	0	52	4	4	19	4	8

Table A.21: *M. abscessus* HiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.86	99.46	99.98	99.89	99.83	99.91	99.85	99.92
Bases Overlapping Query %	1.91	1.98	1.71	0.46	1.9	2.37	2	1.91
Mismatches and Indels %	0.07	0.06	0.15	0.08	0.08	0.02	0.04	0.03
Bases not Mapping Query	1,688	880	98,988	779	4,043	2,467	4,592	13,832
Bases not Mapping Reference	7,257	27,758	1,205	5,572	8,827	4,571	7,487	4,104
Contigs Covered by Reference %	99.97	99.98	98.72	99.98	99.92	99.95	99.91	99.73
Bases Overlapping Reference %	2.06	2.17	51.87	0.8	3	2.54	2.13	2.07
Number of Identical Contigs	15	3	5	9	3	16	13	8

Table A.22: *B. cereus* MiSeq comparison Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.85	99.88	99.94	99.91	99.89	99.96	99.19	99.79
Bases Overlapping Query %	2.05	0.45	0.6	0.51	2.67	1.41	1.49	1.16
Mismatches and Indels %	0.17	0.06	0.08	0.04	0.26	3.83	0.08	0.08
Bases not Mapping Query	15,183	1,521	3,952	1,731	27,183	3,516,008	2,541	1,870
Bases not Mapping Reference	8,234	6,232	3,438	4,791	5,754	2,345	43,699	11,196
Contigs Covered by Reference %	99.72	99.97	99.93	99.97	99.56	80.35	99.95	99.97
Bases Overlapping Reference %	3.3	1.64	1.76	2.07	17.97	172.91	1.6	1.18
Number of Identical Contigs	1	7	9	1	46	9	17	9

Table A.23: *B. cereus* HiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.43	99.88	98.84	98.08	99.59	99.83	88.57	99.68
Bases Overlapping Query %	4.79	0.45	3.61	3.73	7.24	3.76	4.15	3.37
Mismatches and Indels %	0.5	0.06	0.41	0.48	0.96	0.44	0.45	0.4
Bases not Mapping Query	15,853	1,521	6,991	8,550	188,126	216,293	7,993	27,664
Bases not Mapping Reference	31,767	6,232	65,141	107,592	22,929	9,347	641,649	18,035
Contigs Covered by Reference %	99.73	99.97	99.88	99.85	97.86	96.37	99.84	99.54
Bases Overlapping Reference %	10	1.64	5.66	5.95	64.59	6.48	4.33	10.12
Number of Identical Contigs	8	7	6	7	40	0	18	14

Table A.24: *R. sphaeroides* MiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.67	86.85	99.81	92.57	99.43	99.99	99.24	99.07
Bases Overlapping Query %	2.9	1.91	2.55	2.71	2.58	3.08	2.91	2.94
Mismatches and Indels %	0.23	0.14	0.15	0.15	0.16	0.14	0.25	0.14
Bases not Mapping Query	8,181	1,033	3,366	1,636	3,799	18,947	3,644	7,693
Bases not Mapping Reference	14,878	600,441	8,712	339,401	26,097	554	34,691	42,579
Contigs Covered by Reference %	99.82	99.97	99.93	99.96	99.92	99.59	99.92	99.83
Bases Overlapping Reference %	3.05	1.31	4.81	2.92	4.27	3.84	2.92	3.41
Number of Identical Contigs	29	3	8	2	9	10	24	9

Table A.25: *R. sphaeroides* HiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.86	92.32	99.96	97.78	95.95	99.96	98.97	99.38
Bases Overlapping Query %	1.92	1.91	2.32	1.86	4.03	3.07	2.33	2.56
Mismatches and Indels %	0.23	0.15	0.2	0.23	0.48	0.17	0.16	0.27
Bases not Mapping Query	26,373	3,656	11,626	5,166	71,962	34,193	10,767	12,702
Bases not Mapping Reference	6,364	349,164	1,797	100,955	184,120	1,705	46,736	28,343
Contigs Covered by Reference %	99.43	99.91	99.75	99.88	98.74	99.27	99.76	99.76
Bases Overlapping Reference %	3.37	2.06	6.08	2.63	32.95	5.11	2.59	18.04
Number of Identical Contigs	37	68	5	1	1	1	11	6

Table A.26: *V. cholerae* MiSeq comparison with Newbler

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.94	99.99	99.91	99.98	99.82	90.41	99.98	99.85
Bases Overlapping Query %	6.56	2.5	7.47	5.13	18.55	4.89	2.31	2.46
Mismatches and Indels %	0.35	0.14	0.37	0.23	0.7	0.28	0.18	0.18
Bases not Mapping Query	17,843	33,355	4,979	30,155	10,528	946	10,126	12,741
Bases not Mapping Reference	2,278	489	3,768	730	7,942	426,822	970	6,032
Contigs Covered by Reference %	99.55	99.16	99.87	99.24	99.73	99.98	99.74	99.68
Bases Overlapping Reference %	5.78	1.47	3.09	2.08	7.29	2.55	1.7	2.05
Number of Identical Contigs	10	0	11	10	1	3	5	9

Table A.27: *V. cholerae* HiSeq comparison with Newber

	Abyss	CABOG	Mira	MaSuRCA	SGA	SPAdes	Velvet	Soap
Reference Covered by Contigs %	99.69	99.91	99.49	99.93	98.66	99.87	99.76	99.81
Bases Overlapping Query %	6.92	1.68	12.61	3.73	8.99	2.76	1.82	3.11
Mismatches and Indels %	0.24	0.14	0.38	0.33	0.48	0.22	0.21	0.23
Bases not Mapping Query	5,220	75,074	2,420	7,339	11,004	2,879	7,275	3,003
Bases not Mapping Reference	12,729	3,367	22,262	2,614	56,105	5,306	9,395	7,569
Contigs Covered by Reference %	99.87	98.09	99.94	99.81	99.72	99.93	99.81	99.92
Bases Overlapping Reference %	2.1	1.59	2.32	1.75	3.58	1.99	1.44	2.02
Number of Identical Contigs	12	1	1	0	23	10	11	10

Table A.28: LoRDEC and Jabba error correction statistics on PM21 strain and subsequent data.

	Number of Reads	Number of Bases	Average Read Length	Minimum Read Length	Maximum Read Length	Coverage
PM21_corr	39,311	143,755,371	3,657	34	36,462	86.191
PM21_split	180,972	56,319,064	311	100	2,181	33.767
PM21_trim	37,319	128,586,429	3,446	19	35,050	77.096
PM21_jabba	32,098	11,688,414	364	43	3,485	7.008
PM21_1_X01_corr	5,990	22,344,166	3,730	35	27,577	13.397
PM21_1_X02_corr	6,754	24,792,935	3,671	35	31,893	14.865
PM21_1_X03_corr	7,982	30,166,041	3,779	34	35,335	18.087
PM21_2_X01_corr	5,829	19,145,856	3,285	35	33,181	11.479
PM21_2_X02_corr	6,522	22,301,080	3,419	35	32,963	13.371
PM21_2_X03_corr	6,234	25,005,293	4,011	35	36,462	14.992
PM21_1_X01_split	28,122	8,719,368	310	100	2,178	5.228
PM21_1_X02_split	31,546	9,864,231	313	100	2,181	5.914
PM21_1_X03_split	38,169	11,838,801	310	100	2,180	7.098
PM21_2_X01_split	23,894	7,343,891	307	100	2,156	4.403
PM21_2_X02_split	28,307	8,922,547	315	100	2,178	5.350
PM21_2_X03_split	30,934	9,630,226	311	100	2,178	5.774
PM21_1_X01_trim	5,740	20,105,402	3,503	19	26,894	12.055
PM21_1_X02_trim	6,375	22,232,101	3,487	19	31,247	13.330
PM21_1_X03_trim	7,625	27,134,092	3,559	19	35,050	16.269
PM21_2_X01_trim	5,522	16,853,537	3,052	19	32,503	10.105
PM21_2_X02_trim	6,167	19,795,122	3,210	19	32,814	11.869
PM21_2_X03_trim	5,890	22,466,175	3,814	19	34,706	13.470
PM21_1_X01_jabba	4,974	1,821,403	366	45	2,880	1.092
PM21_1_X02_jabba	5,424	1,903,477	351	43	3,012	1.141
PM21_1_X03_jabba	6,770	2,551,445	377	44	3,001	1.530
PM21_2_X01_jabba	4,582	1,662,146	363	44	2,179	0.997
PM21_2_X02_jabba	5,235	1,901,774	363	44	2,514	1.140
PM21_2_X03_jabba	5,113	1,848,169	361	45	3,485	1.108

Table A.29: Illumina reads statistic after adapter removal for strain PMSS1

	Cutadapt	Trimmomatic	Trimmomatic modified
Number of Reads	8,771,554	7,261,142	7,329,607
Number of Bases	1,242,993,451	1,004,083,398	1,005,824,793
Average Read Length	142	138	137
Minimum Read Length	1	36	2
Maximum Read Length	151	151	151
Reference Coverage	768	620	621

Table A.30: Trimmed PacBio reads after proovread for strain PMSS1.

	Cutadapt	Trimmomatic	Trimmomatic modified
Number of Reads	14,737	14,890	14,892
Number of Bases	54,234,516	55,023,769	53,964,429
Average Read Length	3,680	3,695	3,624
Minimum Read Length	403	335	500
Maximum Read Length	29,566	29,433	29,565
Reference Coverage	34	34	33

Table A.31: PacBio reads after proovread with untrimmed reads included for strain PMSS1.

	Cutadapt	Trimmomatic	Trimmomatic modified
Number of Reads	11,358	11,428	11,384
Number of Bases	57,584,368	57,522,225	57,623,746
Average Read Length	5,070	5,033	5,062
Minimum Read Length	258	223	245
Maximum Read Length	33,436	33,442	33,422
Reference Coverage	36	36	36

Table A.32: Canu hybrid assembly statistic on strain PMSS1.

	Cutadapt	Trimmomatic	Trimmomatic mod.	Canu	Ref. Canu
Number of Contigs	5	4	3	1	11
Number of Bases	1,618,840	1,612,244	1,608,236	1,618,182	1,624,016
Average Contig Length	323,768	403,061	536,079	1,618,182	147,638
Minimum Contig Length	10,153	132,723	132,725	1,618,182	2,692
Maximum Contig Length	849,867	697,030	1,218,223	1,618,182	531,590
Reference Coverage	1.00	1.00	0.99	1.00	1.00
N50	849,867	487,513	1,218,223	1,618,182	236,000

Table A.33: Canu hybrid assembly statistic with untrimmed long reads included for strain PMSS1.

	Cutadapt	Trimmomatic	Trimmomatic modified
Number of Contigs	9	13	17
Number of Bases	1,720,798	1,781,498	1,804,749
Average Contig Length	191,200	137,038	106,162
Minimum Contig Length	11,473	15,680	11,474
Maximum Contig Length	470,219	409,372	464,730
Reference Coverage	1.1	1.1	1.1
N50	418,706	304,083	240,898

Table A.34: Newbler hybrid assembly statistic for strain PMSS1

	Cutadapt	Trimmomatic	Trimmomatic modified	Ref. Newbler
Number of Contigs	26	28	30	23
Number of Bases	1,592,469	1,592,870	1,592,200	1,595,242
Average Contig Length	61,249	56,888	53,073	69,358
Minimum Contig Length	461	461	212	462
Maximum Contig Length	500,015	295,061	292,685	319,311
Reference Coverage	0.98	0.98	0.98	0.99
N50	180,012	149,354	180,015	178,900

Table A.35: Newbler hybrid assembly statistic with untrimmed reads included for strain PMSS1.

	Cutadapt	Trimmomatic	Trimmomatic modified
Number of Contigs	23	23	23
Number of Bases	1,592,198	1,592,250	1,592,294
Average Contig Length	69,226	69,228	69,230
Minimum Contig Length	461	461	461
Maximum Contig Length	499,602	499,655	499,651
Reference Coverage	0.98	0.98	0.98
N50	180,015	180,015	180,013

Table A.36: PMSS1: Ranges of reference genome that are not covered by contigs using *Cutadapt*, proovread and Canu.

	GenBank accession	start	end	width
1	CP018823.1	627,520	627,790	271
2	CP018823.1	688,747	689,186	440
3	CP018823.1	689,364	690,365	1,002
4	CP018823.1	691,389	691,508	120
5	CP018823.1	826,509	826,519	11
6	CP018823.1	1,395,521	1,398,498	2,978

Table A.37: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic*, proovread and Canu.

	GenBank accession	start	end	width
1	CP018823.1	200,903	201,354	452
2	CP018823.1	826,507	826,519	13
3	CP018823.1	1,523,463	1,524,403	941
4	CP018824.1	1	6,058	6,058

Table A.38: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic modified*, proovread and Canu

	GenBank accession	start	end	width
1	CP018823.1	426,830	431,440	4,611
2	CP018823.1	826,505	827,082	578
3	CP018824.1	1	6,058	6,058

Table A.39: PMSS1: Ranges of reference genome that are not covered by contigs using *Cutadapt*, proovread with untrimmed reads included and Canu

	GenBank accession	start	end	width
1	CP018823.1	402,365	406,894	4,530
2	CP018823.1	1,461,723	1,461,833	111
3	CP018823.1	1,462,318	1,462,446	129
4	CP018823.1	1,469,338	1,469,851	514

Table A.40: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic*, proovread with untrimmed reads included and Canu

	GenBank accession	start	end	width
1	CP018823.1	198,793	199,165	373
2	CP018823.1	402,764	411,376	8,613
3	CP018823.1	1,043,425	1,045,247	1,823
4	CP018823.1	1,406,665	1,407,165	501
5	CP018823.1	1,509,145	1,509,209	65
6	CP018823.1	1,510,017	1,510,748	732
7	CP018824.1	1	6,058	6,058

Table A.41: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic modified*, proovread with untrimmed reads included and Canu

	GenBank accession	start	end	width
1	CP018823.1	162,745	166,000	3,256
2	CP018823.1	876,443	892,557	16,115
3	CP018823.1	892,825	892,992	168
4	CP018823.1	1,149,454	1,149,819	366
5	CP018823.1	1,609,359	1,616,704	7,346

Table A.42: PMSS1: Ranges of reference genome using stand-alone Canu assembly

	GenBank accession	start	end	width
1	CP018824.1	1	6,058	6,058

Table A.43: PMSS1: Ranges of reference genome that are not covered by contigs using splitted reads of the reference (1500-1999 bp) with coverage 40 and Canu

	GenBank accession	start	end	width

Table A.44: PMSS1: Ranges of reference genome that are not covered by contigs using *Cutadapt*, proovread and Newbler.

	GenBank accession	start	end	width
1	CP018823.1	321,342	321,342	1
2	CP018823.1	691,444	691,456	13
3	CP018823.1	691,499	691,508	10
4	CP018823.1	826,505	826,519	15
5	CP018823.1	1,409,206	1,409,206	1

Table A.45: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic*, proovread and Newbler.

	GenBank accession	start	end	width
1	CP018823.1	200,903	200,910	8
2	CP018823.1	826,510	826,519	10
3	CP018823.1	1,409,206	1,409,206	1
4	CP018823.1	1,524,299	1,524,322	24

Table A.46: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic modified*, proovread and Newbler.

	GenBank accession	start	end	width
1	CP018823.1	826,509	826,519	11
2	CP018823.1	1,033,073	1,033,073	1

Table A.47: PMSS1: Ranges of reference genome that are not covered by contigs using *Cutadapt*, proovread with untrimmed reads included and Newbler.

	GenBank accession	start	end	width
1	CP018823.1	1,438,186	1,438,189	4

Table A.48: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic*, proovread with untrimmed reads included and Newbler.

	GenBank accession	start	end	width

Table A.49: PMSS1: Ranges of reference genome that are not covered by contigs using *Trimmomatic modified*, proovread with untrimmed reads included and Newbler.

	GenBank accession	start	end	width
1	CP018823.1	707,790	707,790	1
2	CP018823.1	1,033,073	1,033,073	1
3	CP018823.1	1,409,206	1,409,206	1
4	CP018823.1	1,438,138	1,438,138	1

Table A.50: PMSS1: Ranges of reference genome that are not covered by contigs using splitted reads (1500-1999 bp) of the Reference with coverage 40 and Newbler

	GenBank accession	start	end	width
1	CP018823.1	707,790	707,790	1
2	CP018823.1	796,972	796,973	2
3	CP018823.1	978,052	978,053	2
4	CP018823.1	1,033,073	1,033,073	1
5	CP018823.1	1,409,206	1,409,206	1

Table A.51: Genetic variants (SNPs and indels) compared to strain PMSS1. Numbers in Gene column indicate that mutation is in a region of a hypothetical protein.

Position	Ref.	Alt.	Type	Gene	Strains
5,928	C	T	missense_variant	1007	PM21
6,327	G	A	missense_variant	1007	PM22
6,328	G	A	missense_variant	1007	PM21
7,165	GTT	G	intergenic		SS1
7,165	GT	G	intergenic		PM21, PM22
101,464	C	T	missense_variant	mcp4_[H]	SS1
102,853	A	AC	intergenic		SS1
116,556	A	C	intergenic		SS1
116,558	T	G	intergenic		SS1
116,595	T	G	intergenic		SS1
116,597	A	G	intergenic		SS1
116,601	T	C	intergenic		SS1
117,183	C	T	synonymous_variant	topA_[H]	SS1
117,552	T	C	synonymous_variant	topA_[H]	SS1
117,564	C	G	synonymous_variant	topA_[H]	SS1
118,800	T	C	synonymous_variant	topA_[H]	SS1
127,408	C	T	synonymous_variant	infC_[H]	PM22
127,429	G	A	synonymous_variant	infC_[H]	PM22
127,465	G	A	synonymous_variant	infC_[H]	PM22
127,486	G	A	synonymous_variant	infC_[H]	PM22
127,501	G	A	synonymous_variant	infC_[H]	PM22
127,507	T	C	synonymous_variant	infC_[H]	PM22
127,522	A	G	synonymous_variant	infC_[H]	PM22
127,525	C	T	synonymous_variant	infC_[H]	PM22
127,630	A	G	synonymous_variant	infC_[H]	PM22
127,775	T	C	synonymous_variant	infC_[H]	PM22
134,530	A	G	stop_lost&splice_region_variant	sdaC_[H]	SS1
204,567	C	T	stop_gained	1195	PM21
249,916	C	T	missense_variant	yejF_[H]	PM22
249,924	C	T	synonymous_variant	yejF_[H]	PM22
249,951	A	G	synonymous_variant	yejF_[H]	PM22
249,984	A	C	missense_variant	yejF_[H]	PM22
249,985	A	G	missense_variant	yejF_[H]	PM22
249,986	T	C	missense_variant	yejF_[H]	PM22
249,990	T	A	synonymous_variant	yejF_[H]	PM22
250,028	A	AT	frameshift_variant	yejF_[H]	SS1
250,049	A	G	synonymous_variant	yejF_[H]	PM22
250,059	A	G	synonymous_variant	yejF_[H]	PM22
250,068	C	T	synonymous_variant	yejF_[H]	PM22
250,185	C	T	synonymous_variant	yejF_[H]	PM22
250,212	T	C	synonymous_variant	yejF_[H]	PM22
250,215	A	G	synonymous_variant	yejF_[H]	PM22
343,489	CA	C	intergenic		PM21
343,489	C	CA	intergenic		PM22
400,924	T	C	synonymous_variant	acsA_[H]	PM22
401,068	C	T	synonymous_variant	acsA_[H]	PM22
401,074	A	G	synonymous_variant	acsA_[H]	PM22
401,086	C	T	synonymous_variant	acsA_[H]	PM22
401,113	A	G	synonymous_variant	acsA_[H]	PM22
401,119	G	A	synonymous_variant	acsA_[H]	PM22

401,206	C	T	synonymous_variant	acsA_[H]	PM22
401,294	A	G	missense_variant	acsA_[H]	PM22
401,299	G	A	synonymous_variant	acsA_[H]	PM22
401,319	T	G	missense_variant	acsA_[H]	PM22
401,369	C	T	missense_variant	acsA_[H]	PM22
401,416	G	A	synonymous_variant	acsA_[H]	PM22
419,034	T	G	missense_variant	fur_[H]	PM22, SS1
476,442	C	CA	intergenic		PM22
477,181	CA	C	frameshift_variant	katA_[H]	SS1
477,326	T	G	missense_variant	katA_[H]	PM21, PM22, SS1
538,772	CA	C	intergenic		SS1
584,930	CG	C	intergenic		PM21, SS1
607,693	G	A	missense_variant	mrdB_[H]	PM21
627,017	A	G	synonymous_variant	1605	PM21
627,446	T	C	synonymous_variant	1605	SS1
628,851	A	AC	frameshift_variant&start_lost	dcuA_[H]	PM21
630,278	C	A	synonymous_variant	1608	PM22
630,281	A	C	synonymous_variant	1608	PM22
630,283	A	C	missense_variant	1608	PM22
630,479	C	T	synonymous_variant	1608	SS1
630,482	G	A	synonymous_variant	1608	SS1
630,488	G	A	synonymous_variant	1608	SS1
630,506	G	A	synonymous_variant	1608	SS1
630,656	C	T	synonymous_variant	1608	SS1
632,165	C	CA	intergenic		PM22, SS1
634,124	A	G	missense_variant	yggA_[H]	SS1
667,520	ATT	A	intergenic		PM22
691,443	C	CT	intergenic		PM21, PM22
721,372	G	GT	frameshift_variant	pldA_[C]	SS1
733,089	G	A	missense_variant	1696	SS1
750,920	C	CA	frameshift_variant	1713	PM22
753,543	C	T	synonymous_variant	virB10_[H]	SS1
753,564	T	C	synonymous_variant	virB10_[H]	SS1
753,567	T	A	synonymous_variant	virB10_[H]	SS1
753,628	A	AGA...	frameshift_variant	virB10_[H]	SS1
753,987	G	A	synonymous_variant	virB10_[H]	PM21
754,019	C	A	missense_variant	virB10_[H]	PM21
754,020	T	C	synonymous_variant	virB10_[H]	PM21
754,274	AAA...	A	conservative_inframe_deletion	virB10_[H]	PM21
810,173	G	A	missense_variant	ykgB	PM22
835,966	G	GA	intergenic		SS1
885,411	C	CT	intergenic		SS1
1,009,698	C	A	missense_variant	czcA	SS1
1,086,712	G	GA	frameshift_variant	2000	PM21
1,086,881	G	A	synonymous_variant	2000	PM21
1,086,893	G	A	synonymous_variant	2000	PM21
1,086,920	G	A	synonymous_variant	2000	PM21
1,086,937	A	G	synonymous_variant	2000	PM21
1,090,839	G	T	synonymous_variant	hemH	PM21
1,119,323	A	G	synonymous_variant	rfaI	PM21
1,119,336	A	G	synonymous_variant	rfaI	PM21
1,119,469	T	C	missense_variant	rfaI	PM21
1,119,476	A	C	missense_variant	rfaI	PM21
1,119,523	C	CAAA	intergenic		PM21

1,119,567	C	T	intergenic		PM21
1,129,510	T	G	synonymous_variant	2040	PM21
1,129,513	A	G	synonymous_variant	2040	PM21
1,170,423	T	C	intergenic		PM22
1,188,487	G	A	stop_gained	cstA	SS1
1,196,576	A	G	synonymous_variant	2103	PM22
1,238,320	TTA...	T	intergenic		SS1
1,242,314	G	A	missense_variant	2138	SS1
1,271,704	C	CAA...	intergenic		PM21
1,272,186	A	C	missense_variant	ssb	SS1
1,272,187	A	G	synonymous_variant	ssb	SS1
1,272,195	C	G	missense_variant	ssb	SS1
1,272,199	A	G	synonymous_variant	ssb	SS1
1,272,203	G	T	missense_variant	ssb	SS1
1,272,241	A	G	synonymous_variant	ssb	SS1
1,274,668	G	A	synonymous_variant	2168	SS1
1,279,530	A	AT	frameshift_variant	yejA	SS1
1,319,161	T	C	missense_variant	rplQ	PM21
1,333,645	ATT...	A	intergenic		PM22
1,350,996	CA	C	intergenic		PM21
1,379,250	T	TG	frameshift_variant&stop_lost	2286	PM21, PM22, SS1
1,414,470	CGA	C	frameshift_variant	cptA	PM21
1,414,470	C	CGA	frameshift_variant	cptA	PM22
1,528,405	A	AC	frameshift_variant	2422	PM21, PM22, SS1
1,551,199	CTTT	C	intergenic		SS1
1,601,213	G	A	missense_variant	mrdA	PM21, PM22, SS1
1,601,222	G	GT	intergenic		PM21, PM22, SS1
1,601,227	G	T	intergenic		PM21, PM22, SS1
1,601,232	T	C	stop_retained_variant	engB	PM21, PM22, SS1
1,601,236	G	A	missense_variant	engB	PM21, PM22, SS1
1,601,370	C	A	missense_variant	engB	PM21, PM22, SS1
1,601,385	C	T	synonymous_variant	engB	PM21, PM22, SS1
1,601,408	T	A	missense_variant	engB	PM21, PM22, SS1
1,601,463	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,601,481	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,601,517	T	A	synonymous_variant	engB	PM21, PM22, SS1
1,601,577	T	A	synonymous_variant	engB	PM21, PM22, SS1
1,601,691	A	C	synonymous_variant	engB	PM21, PM22, SS1
1,601,721	C	T	synonymous_variant	engB	PM21, PM22, SS1
1,601,751	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,601,774	T	C	missense_variant	engB	PM21, PM22, SS1
1,601,775	G	C	synonymous_variant	engB	PM21, PM22, SS1
1,601,792	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,601,796	T	C	synonymous_variant	engB	PM21, PM22, SS1
1,601,799	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,601,814	C	T	synonymous_variant	engB	PM21, PM22, SS1
1,602,704	C	T	missense_variant	engB	PM21, PM22, SS1
1,602,712	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,602,738	C	T	missense_variant	engB	PM21, PM22, SS1
1,602,748	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,602,751	G	C	missense_variant	engB	PM21, PM22, SS1
1,602,752	C	T	missense_variant	engB	PM21, PM22, SS1
1,602,760	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,602,775	T	C	synonymous_variant	engB	PM21, PM22, SS1

1,602,793	C	T	synonymous_variant	engB	PM21, PM22, SS1
1,602,796	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,602,826	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,602,840	C	T	missense_variant	engB	PM21, PM22, SS1
1,602,901	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,602,928	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,602,933	A	G	synonymous_variant	engB	PM21, PM22, SS1
1,602,952	C	T	synonymous_variant	engB	PM21, PM22, SS1
1,602,998	T	C	missense_variant	engB	PM21, PM22, SS1
1,603,070	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,603,340	G	A	synonymous_variant	engB	PM21, PM22, SS1
1,603,359	G	A	missense_variant	engB	PM21, PM22, SS1
1,603,425	A	G	missense_variant	engB	PM21, PM22, SS1
1,603,427	C	T	synonymous_variant	engB	PM21, PM22, SS1
1,603,433	T	C	synonymous_variant	engB	PM21, PM22, SS1