

Computational Analysis of Viruses in Metagenomic Data

Saima Sultana Tithi

Dissertation submitted to the faculty
of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Computer Science & Application

Liqing Zhang, Chair
Na Meng
Sharath Raghvendra
Roderick V. Jensen
Linshu Liu

September 13, 2019
Blacksburg, Virginia

Keywords: Metagenomics, Virus, Phage, Viral Read Classification, Viral Genome
Assembly, Improvement of Virus Assembly
Copyright 2019, Saima Sultana Tithi

Computational Analysis of Viruses in Metagenomic Data

Saima Sultana Tithi

(ABSTRACT)

Viruses have huge impact on controlling diseases and regulating many key ecosystem processes. As metagenomic data can contain many microbiomes including many viruses, by analyzing metagenomic data we can analyze many viruses at the same time. The first step towards analyzing metagenomic data is to identify and quantify viruses present in the data. In order to answer this question, we developed a computational pipeline, FastViromeExplorer. FastViromeExplorer leverages a pseudoalignment based approach, which is faster than the traditional alignment based approach to quickly align millions/billions of reads. Application of FastViromeExplorer on both human gut samples and environmental samples shows that our tool can successfully identify viruses and quantify the abundances of viruses quickly and accurately even for a large data set.

As viruses are getting increased attention in recent times, most of the viruses are still unknown or uncategorized. To discover novel viruses from metagenomic data, we developed a computational pipeline named FVE-novel. FVE-novel leverages a hybrid of both reference-based and de novo assembly approach to recover novel viruses from metagenomic data. By applying FVE-novel to an ocean metagenome sample, we successfully recovered two novel viruses and two different strains of known phages.

Analysis of viral assemblies from metagenomic data reveals that viral assemblies often contain assembly errors like chimeric sequences which means more than one viral genomes are incorrectly assembled together. In order to identify and fix these types of assembly errors, we developed a computational tool called VirChecker. Our tool can identify and fix assembly errors due to chimeric assembly. VirChecker also extends the assembly as much as possible to complete it and then annotates the extended and improved assembly. Application of VirChecker to viral scaffolds collected from an ocean metagenome sample shows that our tool successfully fixes the assembly errors and extends two novel virus genomes and two strains of known phage genomes.

Computational Analysis of Viruses in Metagenomic Data

Saima Sultana Tithi

(GENERAL AUDIENCE ABSTRACT)

Virus, the most abundant micro-organism on earth has a profound impact on human health and environment. Analyzing metagenomic data for viruses has the benefit of analyzing many viruses at a time without the need of cultivating them in the lab environment. Here, in this dissertation, we addressed three research problems of analyzing viruses from metagenomic data. To analyze viruses in metagenomic data, the first question needs to answer is what viruses are there and at what quantity. To answer this question, we developed a computational pipeline, FastViromeExplorer. Our tool can identify viruses from metagenomic data and quantify the abundances of viruses present in the data quickly and accurately even for a large data set. To recover novel virus genomes from metagenomic data, we developed a computational pipeline named FVE-novel. By applying FVE-novel to an ocean metagenome sample, we successfully recovered two novel viruses and two strains of known phages. Examination of viral assemblies from metagenomic data reveals that due to the complex nature of metagenome data, viral assemblies often contain assembly errors and are incomplete. To solve this problem, we developed a computational pipeline, named VirChecker, to polish, extend and annotate viral assemblies. Application of VirChecker to virus genomes recovered from an ocean metagenome sample shows that our tool successfully extended and completed those virus genomes.

Dedication

This dissertation is dedicated to my beloved seven month old son, Saadid.

Acknowledgments

First of all, I would like to thank my PhD supervisor, Dr. Liqing Zhang for her tremendous support and guidance from the first day of my PhD throughout my final defense day. I would also like to thank all my committee members, Dr. Na Meng, Dr. Sharath Raghvendra, Dr. Roderick V. Jensen, and Dr. Linshu Liu for their great suggestions to improve this dissertation. Special thanks to Dr. Roderick V. Jensen and Dr. Frank Aylward for sharing their biology knowledge and insights which was critical for the successful completion of my projects.

I would like to thank my lab mates Hong, Gustavo, Shaohua, Vinaya, Min, Dhoha, Suraj for their questions, comments, and suggestions during our day-to-day discussions and during our lab meetings which was a great help to improve all my research projects.

I would like to thank all my beloved family members, my mom and dad, my mother-in-law and father-in-law, my sister-in-laws for their continuous support and encouragement throughout my PhD. I would like to thank my husband, Mohammad Shabbir Hasan, to whom I am forever indebted for his enduring love and support throughout the ups and downs of my PhD journey, who also finished his PhD from Virginia Tech and despite his PhD study always made time for his family. Last of all, I would like to mention my seven month old son, Saadid, whose smile has become the source of greatest joy of our life, who was the reason to never give up during the challenging days of my PhD.

Contents

Chapter 1 Introduction	1
1.1 Research Objectives	2
1.1.1 Objective 1: Virus and phage identification and abundance profiling in metagenomic data	2
1.1.2 Objective 2: Discovery of draft genomes of novel viruses and phages in metagenomic data	2
1.1.3 Objective 3: Development of a computational pipeline for error-correction, extension, and annotation of viral scaffolds	3
Chapter 2 FastViromeExplorer: A Pipeline for Virus and Phage Identifi- cation and Abundance Profiling in Metagenomic Data	4
2.1 Introduction	4
2.2 Methods	5
2.3 Results and Discussion	10
2.4 Conclusion	16
Chapter 3 FVE-novel: Discovery of Draft Genomes of Novel Viruses and Phages in Metagenomic Data	17
3.1 Introduction	17
3.2 Methods	18
3.2.1 Algorithm Overview	18
3.2.2 Preprocessing the Reference Database	18
3.2.3 Step 1. Read Mapping and Generating Seed Scaffolds	19
3.2.4 Step 2. Extending Seed Scaffolds using Iterative Assembly	20
3.2.5 Step 3: Analysis of New Contigs and Comparison to Reference Sequences	21
3.2.6 Benchmarking on Real Data	21
3.3 Results	22
3.3.1 Length Distribution of the FVE-novel Scaffolds	22
3.3.2 Comparison Between the FVE-novel Scaffolds and Their Reference Sequences	23
3.3.3 Comparison of the FVE-novel Scaffolds Against Several Databases . .	24
3.3.4 Comparison Within the FVE-novel Scaffolds	28
3.3.5 Function Annotation of the Four Complete Scaffolds	36

3.4	Discussion	38
3.5	Conclusion	39
Chapter 4	VirChecker: An Integrated Pipeline for Error-Correction, Extension, and Annotation of Viral Scaffolds	40
4.1	Introduction	40
4.2	Methods	41
4.3	Results and Discussion	44
4.3.1	Group 1 Scaffolds (S0, S1, and S6)	44
4.3.2	Group 2 Scaffold, S2	48
4.3.3	Group 3 Scaffold, S3	50
4.3.4	Group 4 Scaffold, S8	51
4.3.5	Limitations of VirChecker and Usage Recommendations	54
4.4	Conclusion	56
Chapter 5	Conclusion and Future Prospects	57
	Bibliography	59
	Appendix A Supplementary Material of Chapter 2	67
	Appendix B Supplementary Material of Chapter 3	70
	Appendix C Supplementary Material of Chapter 4	79

List of Figures

2.1	Kallisto’s indexing time for five reference databases, NCBI RefSeq eukaryotic viruses (99 MB), NCBI RefSeq phages (148 MB), All NCBI RefSeq viruses and phages (247 MB), 62,921 mVCs (992 MB), and 125,842 mVCs (2 GB).	11
2.2	Comparison of running time among FastViromeExplorer, ViromeScan, and Blastn for seven data sets with 1, 3, 5, 10, 20, 30, and 40 million reads, respectively (a) against a reference database containing 8,957 NCBI RefSeq viruses, (b) against a reference database containing 125,842 mVCs	12
2.3	F1 score of FastViromeExplorer, ViromeScan, and Blastn when using NCBI eukaryotic viruses as the reference database and four simulated data sets of 1 million reads each with mutation frequency 3%, 5%, 7%, and 10% respectively.	13
2.4	Number of viruses from ViromeScan result before applying any filter, after applying criterion 1, after applying criteria 1 and 2, and after applying all three criteria.	14
2.5	Relative abundance of host bacteria at Order level in the samples from FastViromeExplorer result using the 125,842 mVCs, where abundance is normalized by the total abundance of viruses in the sample.	15
3.1	Overview of the FVE-novel pipeline, where the inputs are single-end or paired-end reads and reference database and the output is a set of final extended scaffolds along with ANI and depth of coverage of the output scaffolds.	19
3.2	Length distribution of the 268 scaffolds generated by FVE-novel for the ocean metagenome sample.	22
3.3	The length comparison of the 59 scaffolds against their corresponding references (GOV database). These 59 scaffolds are a subset of the total 268 scaffolds which are extended by FVE-novel tool. The dotted line represents 1:1 ratio between the x and y axis. The color of the dots represents the ANI between the scaffolds produced by FVE-novel and their references.	23
3.4	Alignment of the ten scaffolds to their corresponding references in GOV database. Here the red and blue lines represent the forward and reverse alignment respectively and the intensity of the color is proportional to the percent identity of the alignment.	27
3.5	Percentage of similarity between each pair of the longest ten scaffolds of the 268 scaffolds generated by applying FVE-novel to the ocean metagenome sample.	28

3.6	The <i>log2</i> -scaled depth of coverage of (a) <i>S0</i> (193 kb) and (b) 153 kb scaffold representing the dominant strain of the novel virus (recovered from <i>S0</i>) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).	30
3.7	The <i>log2</i> -scaled depth of coverage of <i>S1</i> (155 kb) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).	31
3.8	The <i>log2</i> -scaled depth of coverage of (a) <i>S2</i> (136 kb) and (b) 151 kb scaffold representing the extended and complete version of <i>S2</i> across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).	33
3.9	The <i>log2</i> -scaled depth of coverage of (a) <i>S3</i> (133 kb) and (b) 177 kb scaffold representing the dominant strain of Prochlorococcus phage P-SSM4 (recovered from pieces of <i>S3</i>) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).	34
3.10	The <i>log2</i> -scaled depth of coverage of (a) <i>S8</i> (73 kb) and (b) 183 kb scaffold representing the dominant strain of Prochlorococcus phage P-HM2 (recovered from pieces of <i>S8</i>) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).	35
3.11	Protein annotation of the two novel scaffolds (a) 153 kb scaffold of dominant strain of a novel virus, (b) 151 kb scaffold of dominant strain of a novel virus.	36
3.12	Protein annotation of the two strains of Prochlorococcus phage (a) 177 kb scaffold of dominant strain of Prochlorococcus phage P-SSM4 and (b) 183 kb scaffold of dominant strain of Prochlorococcus phage P-HM2.	37
4.1	Overview of the VirChecker pipeline, where input is a draft assembly of virus and corresponding reads and output is an extended and improved assembly.	42
4.2	Checking the circularity of the scaffold by (a) dividing the scaffold into two parts and aligning them against each other, (b) find a highly similar region, (c) extend the similar region as much as possible and trim one of the similar regions	42
4.3	The depth of coverage of scaffold, <i>S0</i> from previous study (length 193 kb) where the non-uniform coverage regions or suspicious regions are marked.	45
4.4	The <i>log2</i> -scaled depth of coverage of the improved version of scaffold <i>S0</i> (length 158,259 bp) after applying VirChecker.	45
4.5	The depth of coverage of scaffold, <i>S1</i> from previous study (length 153 kb) where the non-uniform coverage regions or suspicious regions are marked.	46
4.6	The <i>log2</i> -scaled depth of coverage of the improved version of scaffold <i>S1</i> (length 153,133 bp) after applying VirChecker.	47
4.7	The protein annotation of scaffold <i>S1</i> from VirChecker.	47
4.8	The <i>log2</i> -scaled depth of coverage of the improved version of scaffold <i>S6</i> (length 152,967 bp) after applying VirChecker.	48
4.9	Percentage of similarity of 153 kb dominant strain with the improved scaffolds <i>S0</i> , <i>S1</i> , and <i>S6</i> obtained from VirChecker tool.	49

4.10	The \log_2 -scaled depth of coverage of the improved version of scaffold S_2 (length 149,414 bp) after applying VirChecker.	49
4.11	The protein annotation of scaffold S_2 from VirChecker.	50
4.12	Percentage of similarity of 151 kb dominant strain with the improved scaffold S_2 obtained from VirChecker.	51
4.13	The depth of coverage of scaffold, S_3 from previous study (length 132 kb) where the non-uniform coverage regions or suspicious regions are marked. . .	52
4.14	The \log_2 -scaled depth of coverage of the improved version of scaffold S_3 (length 178,118 bp) after applying VirChecker.	52
4.15	The protein annotation of scaffold S_3 from VirChecker.	53
4.16	Percentage of similarity of 177 kb dominant strain of Prochlorococcus phage with the improved scaffold S_3 obtained from VirChecker.	53
4.17	The depth of coverage of scaffold, S_8 from previous study (length 73 kb) where the non-uniform coverage regions or suspicious regions are marked.	54
4.18	The \log_2 -scaled depth of coverage of the improved version of scaffold S_8 (length 179,890 bp) after applying VirChecker.	55
4.19	Percentage of similarity of 183 kb dominant strain of Prochlorococcus phage with the improved scaffold S_8 obtained from VirChecker.	55
4.20	The protein annotation of scaffold S_8 from VirChecker.	56
A.1	Visualization of the reads mapped to the repeat region of BeAn 58058 virus.	67
A.2	Visualization of the reads mapped to the several repeat regions of Pandoravirus dulcis.	67
A.3	Visualization of the reads mapped to the repeat region of Encephalomyocarditis virus from ViromeScan result.	68
B.1	Comparison of S_0 with 153 kbp scaffold representing the dominant strain of the novel virus recovered from S_0	70
B.2	Comparison of S_1 with 153 kbp scaffold representing the dominant strain of the novel virus recovered from S_0	71
B.3	Comparison of S_6 with 153 kbp scaffold representing the dominant strain of the novel virus recovered from S_0	72
B.4	The \log_2 -scaled depth of coverage of S_6 (80 kbp) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).	72
B.5	Comparison of S_2 with 151 kbp scaffold representing the extended and complete version of S_2	73
B.6	Comparison of S_3 with 177 kbp dominant strain of Prochlorococcus phage P-SSM4 (recovered from pieces of S_3).	74
B.7	Comparison of 177 kbp dominant strain with Prochlorococcus phage P-SSM4.	75
B.8	Comparison of S_8 with 183 kbp dominant strain of Prochlorococcus phage P-HM2 (recovered from pieces of S_8).	76
B.9	Comparison of 183 kbp dominant strain with Prochlorococcus phage P-HM2.	77

C.1	High coverage region of 151 kb dominant strain with length about 2 kb. . . .	79
C.2	Percentage of similarity of 151 kb dominant strain with the improved scaffold S2 obtained from VirChecker tool without applying Pilon tool.	80
C.3	Visualization of the low coverage region of $S1$	80
C.4	Visualization of the low coverage region of manually curated 177 kb dominant strain.	81
C.5	Visualization of the low coverage region of $S3$	81

List of Tables

3.1	The longest ten scaffolds of the 268 scaffolds generated by applying FastViromeExplorer-novel to the ocean metagenome sample collected from Aylward et al. showing ANI and aligned nucleotide percentage between the reference (GOV database) scaffold and the extended scaffold.	25
3.2	The ten scaffolds are compared with the 483 scaffolds assembled in the original Aylward et al. study and blast nr nucleotide database using BLAST, the best hit from BLAST are selected, and ANI and aligned nucleotide percentage for the best BLAST hit result are calculated using MUMmer tool	26
A.1	Recall and precision of the three tools for four simulated datasets with mutation rate 3%, 5%, 7%, and 10% respectively.	68
B.1	Description of the 12 viral-metagenomic samples collected from the study Aylward et al. Along with the time points when these samples were collected, as the sample at station 6 was collected first, we considered the time for collecting this sample as 0 hour.	71
B.2	Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 153 kbp dominant strain of the novel virus recovered from <i>S0</i> . Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.	73
B.3	Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 151 kbp scaffold representing the extended and complete version of <i>S2</i> . Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.	74
B.4	Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 177 kbp dominant strain of Prochlorococcus phage P-SSM4 (recovered from pieces of <i>S3</i>). Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.	76

B.5 Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 183 kbp dominant strain Prochlorococcus phage P-HM2 (recovered from pieces of *S8*). Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample. 78

Chapter 1

Introduction

Identifying the kinds of viruses that infect eukaryotes and prokaryotes (phages) and understanding their functions are important because they are the most abundant entities on Earth [47]. In the human body, there are estimated 100 times more viral particles than eukaryotic cells [17]. Studies have shown that there are connections between human gut microbiome (viruses and bacteria) and diseases such as inflammatory bowel disease (IBD), colorectal cancer [38, 39, 24]. Moreover, recent emerging viral outbreaks including Zika outbreak in Brazil [9], Ebola in West Africa [11, 21], Middle East respiratory syndrome coronavirus (MERS-CoV) [22], SARS, influenza-A caused tens of thousands of human deaths. Viruses can also be used in the treatment of some diseases [33]. Bacteriophages (viruses that attack bacteria), which can act as antibacterial agents, can be used in the treatment of diseases such as Cystic Fibrosis lung infections [40], tuberculosis [10], and multidrug-resistant bacterial infections [74]. Viruses like retroviruses and adenoviruses are used in gene therapy because these viruses can be used as vehicles to carry good genes into a human cell [27]. In addition, viruses are fundamental drivers of many vital ecosystem processes including ocean nutrient cycling, horizontal gene transfer, global biogeochemistry [55]. Viruses have been shown to play important roles in shaping the composition and function of environmental microbiomes [54]. To better understand the viruses and eventually prevent viral outbreaks, it is critical to have timely identification and annotation of viruses. Traditional techniques of virus identification rely on isolation and culturing, which is not only time-consuming but often infeasible as many viruses and their hosts are difficult to cultivate in laboratories. Thanks to the fast development of biotechnology, it is now easy and quick to produce metagenomics data for a direct analysis of genetic materials to identify viruses and their abundances in various environments [23].

1.1 Research Objectives

1.1.1 Objective 1: Virus and phage identification and abundance profiling in metagenomic data

With the ease of metagenomic data generation also comes the challenge of downstream data analysis, including the computational identification of viral species and their abundances in a fast yet accurate manner from hundreds of millions/billions of short sequences. To provide fast and accurate virus detection and quantification of metagenomics data, we developed a stand-alone pipeline, FastViromeExplorer [67]. FastViromeExplorer has two steps, (1) read mapping step, and (2) filtering step. For the mapping step, FastViromeExplorer calls kallisto [8], a pseudoalignment based approach originally developed for alignment and quantification of RNA-Seq data, as a subprocess to map the input reads against the virus reference database. Our tool leverages kallisto for its ultrafast speed and then the results are filtered using three criteria, introduced to ensure the quality of virus detection and especially to reduce the number of false positive viruses from the result. As FastViromeExplorer can process millions of reads within minutes while having similar virus detection accuracy to the gold standard tool BLASTN [2], it empowers researchers, that have limited computing power, to process large metagenomic data within a reasonable time and without using computer clusters. Results show that FastViromeExplorer is applicable to a variety of metagenomics datasets including human microbiome and environmental samples. The detailed workflow of FastViromeExplorer and results from applying this tool to human gut samples and ocean environmental samples are given in chapter 2.

1.1.2 Objective 2: Discovery of draft genomes of novel viruses and phages in metagenomic data

After finding the known viruses present in a metagenomic dataset, the next step is to find the presence of novel viruses in the dataset and recover the full-length genome of the new virus. For all the viral outbreaks like Ebola, Zika virus outbreak, the first step taken towards finding a medication is to identify the pathogen from the sequencing data and then recover the full genome of the pathogen [60, 11]. Moreover, as most of the research on microbiomes in the last decade focused on bacteria, it is only recently that viruses and phages have gained increased attention. Hence our understanding of viruses is greatly limited by the vast unknown of viral sequence space, with an estimated 99% of the viruses still uncharacterized at the genomic level and thus considered as “viral dark matter” [75]. The main step towards filling this gap is to recover and categorize novel viruses. The traditional approach for recovering novel viral genomes suffers from high demand for computational resources for de novo assembly. To mitigate this problem, we are developing a tool, FVE-novel, a computational pipeline for uncovering draft genomes of novel viruses and phages from metagenomic samples. FVE-

novel leverages FastViromeExplorer (FVE), a pipeline developed for rapid identification of viruses in metagenomic samples, to efficiently map metagenomic reads to viral reference genomes or contigs, performs de novo assembly of the mapped reads to generate scaffolds, and then extends those generated scaffolds via iterative assembly to produce final novel viral scaffolds. This methodology is a hybrid of both reference-based and de novo assembly approaches that represents a novel approach for the identification of novel viral genomes. We applied FVE-novel to an ocean metagenome sample and obtained 268 viral scaffolds that potentially come from novel viruses, and through manual examination and validation of the ten longest scaffolds obtained from our tool we successfully recovered four complete virus genomes. Two are novel as they were not found in the existing databases and the other two are related to known phages. This hybrid reference-based and de novo assembly approach used by FVE-novel represents a powerful new approach for exploring viral diversity in metagenomic data. The detailed description of the methods of this tool along with the results from applying this tool to the ocean sample is given in chapter 3.

1.1.3 Objective 3: Development of a computational pipeline for error-correction, extension, and annotation of viral scaffolds

Because of the advancement in high throughput sequencing technologies metagenomic data now yields millions of reads. The complex nature of metagenomic data and the vast amount of sequencing reads make the analysis of metagenomic data difficult. As a result, instead of complete genomes, analysis of metagenomic data often yields draft assemblies or fragmented assemblies of viruses. These draft assemblies need to be polished and extended further to recover complete genomes and in order to do that computational tools are needed. Here, we developed a computational pipeline, VirChecker, to polish, extend, and annotate draft assemblies or scaffolds of viruses. Application of our tool shows a successful discovery of four virus genomes from an ocean metagenomic dataset. The detailed description of this tool along with the results is given in chapter 4.

Chapter 2

FastViromeExplorer: A Pipeline for Virus and Phage Identification and Abundance Profiling in Metagenomic Data

2.1 Introduction

With the increase in availability of metagenomic data generated by next generation sequencing, there is an urgent need for fast and accurate tools for identifying viruses in host-associated and environmental samples. Strategies for identification and abundance quantification of viruses vary among different tools, ranging from analyzing marker genes, binning sequences or reads into taxonomic groups, assembling sequences into contigs and then annotating the genes from the contigs for taxonomy, to directly aligning short reads to a reference database and inferring virus types and abundances based on the alignment results. The most straightforward and fastest approach for virus taxonomic annotation is to align short reads to a marker gene database and identify viruses based on the alignments, for example, MetaPhlAn [62] and its updated version MetaPhlAn2 [68] use this approach. However, marker gene analysis strategy does not work well when the input data contain species that do not have known marker genes. Comparatively, assembling reads into longer contigs and then performing the taxonomic analysis with contigs tend to produce more accurate results [56]. This type of virus analysis pipelines normally requires the users to assemble the reads using an independent assembler and then annotates the assembled contigs (e.g., VirSorter [57], VirFinder [53], Metavir [58], Metavir2 [59], and Virome [73]). Another assembly-based workflow for identifying viral elements from metagenomic reads is FRAP (fragment recruitment, assembly, purification) [13]. Understandably the assembly of short reads into contigs gives longer sequences including longer coding regions with more informative content, which

leads to improved annotation and downstream analysis. However, read assembly can be very time-consuming for large metagenomics data and can also generate chimeras (i.e., sequences from different genomes that are incorrectly assembled together due to their similarity) that mislead downstream annotation [70, 69]. Finally, tools such as MG-RAST [37], ViromeScan [52], VIP [36], and HoloVir [30] directly align short reads to a reference database of whole genomes for taxonomy annotation. Many of these tools were initially developed for bacteria but adapted later for viruses and tend to work poorly due to the much smaller reference databases available for viruses than for bacteria [17]. In addition, as many virus annotation tools (i.e., Metavir [58], Metavir2 [59], Virome [73], MG-RAST [37]) are web-based, users need to upload their data to the website and wait for a long time to get results.

To provide fast and accurate virus detection and quantification on metagenomics data, we developed a stand-alone pipeline, FastViromeExplorer [67]. Instead of the traditional read alignment tools such as BLAST [2] or Bowtie2 [31], FastViromeExplorer uses kallisto [8], a pseudoalignment based approach originally developed for alignment and quantification of RNA-seq data. Kallisto has also been used to map metagenome reads to a database of bacterial genomes [61]. Here we first use kallisto to rapidly map short metagenome reads to a reference virus database. Then FastViromeExplorer filters the alignment results based on minimal coverage criteria and reports virus types and abundances along with taxonomic annotation. To test the performance of FastViromeExplorer, we used simulated datasets of a known mixture of viral, phage, and bacterial genomes with different error/mutation rates. We also applied FastViromeExplorer to real metagenome datasets generated from a Fecal Microbiota Transplantation (FMT) experiment [32] and from environmental ocean samples [3]. FastViromeExplorer is directly compared with blastn and ViromeScan [52], a recently developed read based annotation tool for eukaryotic viruses.

FastViromeExplorer is freely available at <https://code.vt.edu/saima5/FastViromeExplorer>.

2.2 Methods

FastViromeExplorer, written in Java, has two main steps, (1) the read mapping step where all reads are mapped to a reference database, and (2) the filtering step where the mapping results are subjected to three major filters (detailed later) for output of the final results on virus types and abundances. The input of the read alignment step is raw reads (single-end or paired-end) in fastq format. FastViromeExplorer uses the reference database downloaded from NCBI containing 8,957 RefSeq viral genomes as default but can also use any updated or customized databases as reference. FastViromeExplorer incorporates the reference database as an input parameter, so that user can use any database of his choice as input. A precomputed kallisto index file, generated for the 8,957 genomes is distributed here: <http://bench.cs.vt.edu/FastViromeExplorer/>.

First, FastViromeExplorer calls kallisto [8] as a subprocess to map the input reads against the

reference database. Kallisto was developed to map RNA-seq data to a reference transcriptome (all the transcripts for a genome) leveraging the pseudoalignment process and estimate the abundance of the transcripts using the Expectation-Maximization (EM) algorithm [14]. As there is no actual sequence alignment of the entire read over the reference sequences, the pseudoalignment process enables read mapping to be both lightweight and superfast. Essentially, kallisto searches for exact matches for a short k-mer (default size 31 bp) between the metagenomic reads and the sequences in the virus/phage database. For example, kallisto was able to map and quantify 30 million paired-end RNA-seq reads from a human transcriptome sample in less than 10 minutes on a small laptop computer with a 1.3-GHz processor [8]. In addition to the ultrafast speed, kallisto also gives accurate estimation of abundance of each transcript or reference sequence [61, 66]. Consequently, kallisto could provide an ideal tool for detection and quantification of viruses in metagenomic samples that commonly have tens of millions of reads, mapping of which using commonly used programs such as BLAST can be time-consuming and often infeasible without computer clusters. Therefore, FastViromeExplorer deploys kallisto for the purpose of read mapping and abundance estimation of the viruses. Since kallisto searches for exact matches for a short k-mer (default size 31 bp) between the metagenomic reads and the sequences in the virus/phage database, if a 31 bp match is found then the virus is detected. If multiple hits occur, then kallisto uses an EM algorithm to help resolve the redundancy and quantify the abundances of the detected viruses. The k-mer size in kallisto can be altered depending on user’s need. For example, if the sample is expected to contain viral sequences that are divergent from those in the reference database the k-mer size can be reduced to improve detection sensitivity.

After the first alignment step, FastViromeExplorer takes the output of kallisto that includes information of the aligned reads together with estimated abundances or estimated read counts of all the identified viruses for the processing of the second step. The second step filters the output of the first step using three criteria, introduced to ensure the quality of virus detection and especially to reduce the number of false positive viruses from the result. In detail, the first criterion, hereafter referred to as “ R ”, is based on the ratio of the observed extent of genome coverage with the expected extent of genome coverage, computed as

$$R = \frac{C_o}{C_e}, \quad (2.1)$$

C_o is the observed extent of genome coverage by the mapped reads, computed as

$$C_o = \frac{L_s}{L_g}, \quad (2.2)$$

where L_s is the actual length of the genome that is supported or covered by the mapped reads and L_g is the length of the genome. C_e is the expected extent of genome coverage, assuming a Poisson distribution for the mapped reads along the genome, and therefore,

$$C_e = 1 - e^{-\frac{N * L_r}{L_g}}, \quad (2.3)$$

where N is the number of mapped reads to the genome, L_r is the read length, and L_g is the length of the genome. If a virus has $R < 0.3$, FastViromeExplorer discards the virus. This criterion is motivated by the observation that some viruses detected by our tool only have reads mapped to the repeat regions of their genomes. For example, while analyzing the fecal samples from Lee et al. [32], we found that for the BeAn 58058 virus (NC_032111.1), all the reads were mapped to one particular region of its genome, from 8,200 bp to 8,700 bp (see Appendix A, Figure A.1). Analyzing this region using RepeatMasker [63] revealed that it is a simple repeat region and falls into the class of Alu elements. If the virus is truly present in the sample, we expect reads to be mapped to not only the repeat region but also other regions of the genome. Therefore, finding this virus is likely an artifact caused by the prevalence of repeat regions instead of real biological signals. If the reads are all mapped to a repeat region, the observed coverage of the virus genome C_o is expected to be much lower than C_e , as a result, R is low and by imposing a cutoff of 0.3 (determined based on our empirical analyses), viruses that have reads mapped to only repeat regions get filtered out.

The second criterion requires $C_o \geq 0.1$, that is, a virus that has $C_o < 0.1$ is discarded. This criterion requires that the mapped reads should cover at least 10% of the viral genome. Manual inspection of the results of our tool reveals that very large viruses may have several repeat regions in their genomes and as a result, though all the reads are mapped to the repeat regions, they are mapped to different repeat regions. In these cases, the difference between C_o and C_e may be small and therefore R can be high enough to pass the first filter. However, it is very likely that the result is simply an artifact of repetitive sequences. For example, while analyzing the fecal samples [32], we found that Pandoravirus dulcis (NC_021858.1), a very large virus with 1,908,524 bp, has several repeat regions, and all the reads were mapped only to the repeat regions (see Appendix A, Figure A.2). Hence, to alleviate this artifact, $C_o \geq 0.1$ is used as the second filter. As repeat regions of a virus usually cover less than 10% of the genome [50], if any virus is covered by more than 10% by the reads, it is reasonable to assume that the reads are not merely from repeat regions and thus the virus should be considered in the result.

The third criterion is based on the number of mapped reads N . Extensive empirical analysis and inspection of the results of our tool show that for very small viruses, only a few reads are enough to cover a good portion of the viral genome, resulting in high R and C_o that pass criteria 1 and 2. For example, in the fecal samples [32] that we analyzed, four reads were mapped to Rose rosette virus RNA3 (NC_015300.1). As the viral sequence has only 1,544 bp, four reads of length 150 bp were enough to pass criteria 1 and 2. But as only a handful of reads are mapped, it is likely that the virus is false positive. To be more stringent, FastViromeExplorer applies the third filter requiring the number of mapped reads to be greater than 10, and therefore discards the ones with $N < 10$.

After applying all the filters, FastViromeExplorer outputs the final result that contains a list of identified viruses in the given sample along with the estimated read count or abundance and taxonomy of the viruses. The output list is sorted by the abundance with the most

abundant viruses on the top of the list.

It is worth noting that the three criteria are introduced to improve the virus detection specificity by alleviating artifacts caused by factors such as repeat sequences and low genome coverage. The actual cutoff values for R , C_o , and N are based on our empirical experience and literature observation. However, depending on the specific studies and the need of users, the cutoff values used here might not be suitable. To allow flexibility and customization, FastViromeExplorer incorporates these three filters as parameters so that users can easily adjust the values to adapt to their own studies. For example, users can deploy more stringent criteria by setting higher values for R , C_o , and N than the default, to get a “high confidence” set of viruses or can lower these values to increase sensitivity to detect divergent viruses or viral reads in metagenomic data where coverage may be expected to not be uniform [65].

FastViromeExplorer was run on both simulated and real data to examine its running time and accuracy. FastViromeExplorer used kallisto (version 0.43.1) with default settings and generated pseudoalignment results in sam format and filtered abundance results in a tab-delimited file. The abundance results contain identified virus names, NCBI accession numbers, NCBI taxonomic path, and estimated read counts. FastViromeExplorer was run on two different reference databases, the default database distributed together with FastViromeExplorer, that is, the NCBI RefSeq database containing 8,957 genomes of eukaryotic viruses and phages, and the set of sequences collected from the JGI “earth virome” study [47] containing 125,842 metagenomic viral contigs (mVCs). The taxonomic annotation and host information for these mVCs were collected from the IMG/VR database [46].

In addition to the challenge of mapping 10s or 100s of millions of metagenomic reads, tools for the accurate identification and quantification of viral genomes must also be capable of handling ever-growing reference databases of viral sequences. In order to measure how the indexing step of kallisto scales with reference databases of different sizes, kallisto was applied to index five different databases. Three databases were generated from NCBI RefSeq viral database, one containing only phages (2,187 phage genomes), one containing only eukaryotic viruses (6,770 eukaryotic virus genomes), and one containing both phages and eukaryotic viruses (8,957 viral genomes). The other two databases were created from sequences collected from Paez-Espino et al. [47], one containing all the 125,842 mVCs and the other containing half of the mVCs. The time analysis of kallisto’s indexing step was produced on a Linux based cluster with 64 CPUs and 128 GB RAM. The indexing step was run using default k-mer size 31 and default number of threads 1. The precomputed kallisto index file for the full 125,842 mVCs from JGI is available here: <http://bench.cs.vt.edu/FastViromeExplorer/>.

To evaluate the performance of FastViromeExplorer, we compared speed and accuracy with ViromeScan, a recently developed virus annotation pipeline that calls Bowtie2 as a subprocess for read mapping, that was shown to be 1,000 times faster than previous tools [52]. ViromeScan was run with default settings and with the eukaryotic DNA/RNA virus database containing 4,370 genome sequences, the largest reference database provided by ViromeScan, and with a custom database consisting of the 125,842 mVCs from JGI. ViromeScan gener-

ated alignment results and abundances of viruses at family, genus, and species level. We also ran `blastn` (version `ncbi-blast-2.6.0+`) using both the NCBI RefSeq viral database and the large JGI database. `Blastn` only generated the alignment result in text format. All the time analyses were calculated using elapsed real time from Unix’s time command.

To examine the virus detection and quantification accuracy of FastViromeExplorer, simulated metagenomic data were used. A randomly selected collection of genomes containing 4,000 virus genomes and 2,000 bacteria genomes were obtained from NCBI RefSeq database. Four paired-end read datasets, each containing one million reads of length 100 bp, were generated from these genomes using the read simulator WGSIM (<https://github.com/lh3/wgsim>). For all the datasets, 49% reads were from viruses and 51% from bacteria. The four datasets were generated using 1% sequencing error rate and 3%, 5%, 7%, or 10% mutation frequencies respectively. ViromeScan and `blastn` were also applied to these four datasets. As ViromeScan uses eukaryotic viruses as the reference database, for comparison, both FastViromeExplorer and `blastn` were run on a reference database containing only NCBI RefSeq eukaryotic viruses. ViromeScan was run with the eukaryotic virus database provided by ViromeScan. Under the default setting, ViromeScan removed all the mapped reads during its quality filtering and trimming step (`trimBWastyle.pl` script) and did not produce any results. Therefore, it was run without ViromeScan’s quality filtering and trimming step. With the ground truth for the alignment of the reads, recall, precision, and F1 score were calculated using the following formula:

$$Recall = \frac{TP}{TP + FN}, \quad (2.4)$$

$$Precision = \frac{TP}{TP + FP}, \quad (2.5)$$

$$F1\ score = \frac{2 * Recall * Precision}{Recall + Precision}. \quad (2.6)$$

To examine the running time and performance of FastViromeExplorer in detecting viruses on real data, the fecal metagenomics datasets described in Lee et al. [32] were downloaded from NCBI under the accession number SRP093449 and annotated with both FastViromeExplorer and ViromeScan. The study tracked bacteria colonization in a fecal microbiota transplantation (FMT) experiment through the analysis of metagenomic data. To examine how the viruses/bacteriophages were affected by the transplantation, we reanalyzed the four fecal metagenomic samples collected from a healthy donor and three samples from a recipient patient suffering mild/moderate ulcerative colitis. The three samples for the recipient were collected prior to FMT, four weeks after FMT, and eight weeks after FMT, respectively. All the reads were Illumina paired-end reads with 150 bp read length. Seven data sets of different sizes (1, 3, 5, 10, 20, 30, and 40 million reads) were also generated from the samples and annotated by FastViromeExplorer and ViromeScan to compare their running time on large datasets. To examine the effect of the reference database on results, FastViromeExplorer

was applied to the samples using two different reference databases, FastViromeExplorer’s default reference database and the set of 125,842 mVCs collected from the study [47]. While using the NCBI RefSeq database as reference, a Linux based laptop with Intel core i5-3230M CPU @ 2.60 GHz * 4 processors and 12 GB RAM was used to produce the results, and while using the 125,842 mVCs as reference, a Linux based cluster with 64 CPUs and 128 GB RAM was used to produce the results. While using the cluster, only 1 thread was used to run the tools.

To examine the applicability of FastViromeExplorer on environmental samples, an ocean water metagenome file described in Aylward et al. [3] was downloaded from NCBI SRA under the accession number SRX2912986 and analyzed with FastViromeExplorer. The metagenome sequencing file had around 18 million paired-end reads and the 125,842 mVCs collected from the study [47] was used as reference database. As the original study focused on ocean virome, a viral contig set collected from Global Ocean Virome (GOV) study [55] was also used as reference database. The GOV contig set contains 298,383 epipelagic and mesopelagic viral contigs and a precomputed kallisto index file for this viral contig set is available here: <http://bench.cs.vt.edu/FastViromeExplorer/>.

2.3 Results and Discussion

We applied kallisto to index five databases of different sizes and calculated the running time of the indexing step. Figure 2.1 shows that indexing time increases linearly with the size of the reference databases, and for the largest reference database of 2 GB, kallisto took 3 hours and 38 minutes to generate the index file.

To examine how running time changes with sample size, we created seven data sets with 1, 3, 5, 10, 20, 30, and 40 million reads respectively from the data described in Lee et al. [32] and applied FastViromeExplorer, ViromeScan, and blastn. As blastn took too long to run on large data sets, we run blastn on only three data sets of size 1, 3, and 5 million reads respectively. Two databases, one containing all NCBI RefSeq viral genomes and the other containing 125,842 mVCs from Paez-Espino et al. [47], were used as the reference databases, to also examine the effect of reference databases on running time. Figure 2.2a shows the running time using the NCBI database as reference. FastViromeExplorer has the shortest running time for all the seven data sets. For the data set with 5 million reads FastViromeExplorer took only seven minutes, compared to 12 minutes for ViromeScan, 31 minutes for blastn. The speedup of FastViromeExplorer compared to ViromeScan became much more pronounced when a larger reference database was used. Figure 2.2b shows that when we used the larger reference database, for a data set with 5 million reads, FastViromeExplorer took 17 minutes, compared to 53 minutes for ViromeScan, and 4 hours and 40 minutes for blastn. So FastViromeExplorer ran 3 times faster than ViromeScan and 16 times faster than blastn. For the largest data set with 40 million reads, FastViromeExplorer took 2 hours and 27 minutes, a 2.5x speedup compared to ViromeScan that took 6 hours and 23 minutes.

Taken together, when using NCBI virus and phage database as reference, FastViromeExplorer takes on average about 1 minute to process one million reads; when using a larger database (125,842 mVCs, 2GB), FastViromeExplorer takes 3–4 minutes to process one million reads, a 2–3x speed up compared to ViromeScan. Note that the indexing time (for both FastViromeExplorer and ViromeScan) was not counted in the running time shown (Figure 2.2) as indexing needs to be computed only once. Once the index file is generated, it can be used to analyze any metagenomic data.

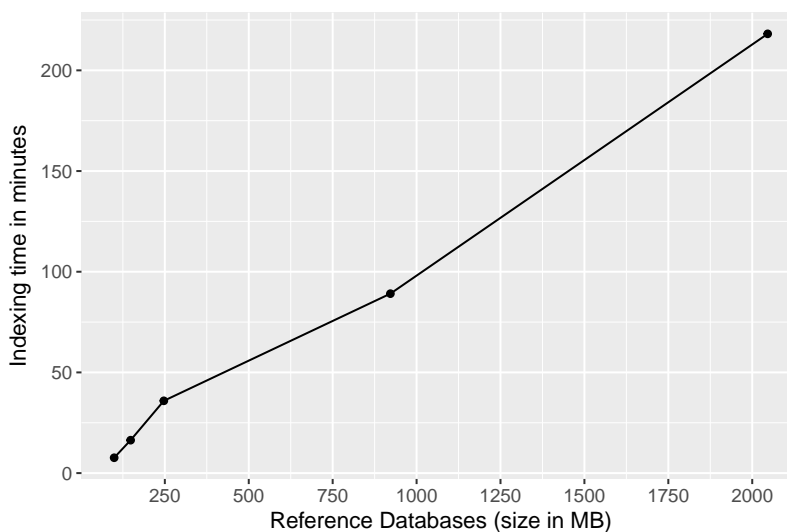


Figure 2.1: Kallisto’s indexing time for five reference databases, NCBI RefSeq eukaryotic viruses (99 MB), NCBI RefSeq phages (148 MB), All NCBI RefSeq viruses and phages (247 MB), 62,921 mVCs (992 MB), and 125,842 mVCs (2 GB).

Simulated datasets were initially used to compare the annotation performance of FastViromeExplorer with ViromeScan and blastn. Since viruses mutate fast, even if it is the same viral species, the viral sequences in the metagenomic data might not be exactly the same as their sequences in the reference database, it is therefore important to examine the performance of a virus detection tool taking into account virus’s high mutation rate. We therefore simulated four data sets with different mutation frequencies (3%, 5%, 7%, and 10%) from the references and applied FastViromeExplorer, ViromeScan, and blastn. Figure 2.3 shows the F1 score (*Recall* and *Precision* are given in Appendix A, Table A.1). All the tools have had high *Precision* (99%) across all the data sets. But as mutation frequency becomes higher, the number of mapped reads is reduced and *Recall* becomes lower for all the tools. In terms of F1 score, blastn has the best score, FastViromeExplorer has similar but slightly lower score, and ViromeScan has the lowest score. For the data set with the highest mutation frequency (10%), the F1 scores for blastn, FastViromeExplorer and ViromeScan are 0.79, 0.7, and 0.43 respectively. But FastViromeExplorer took 2 minutes compared to blastn which took 8 minutes. Therefore, for these simulated data sets and using all eukaryotic viruses as the reference database, FastViromeExplorer runs four times faster than blastn while maintaining

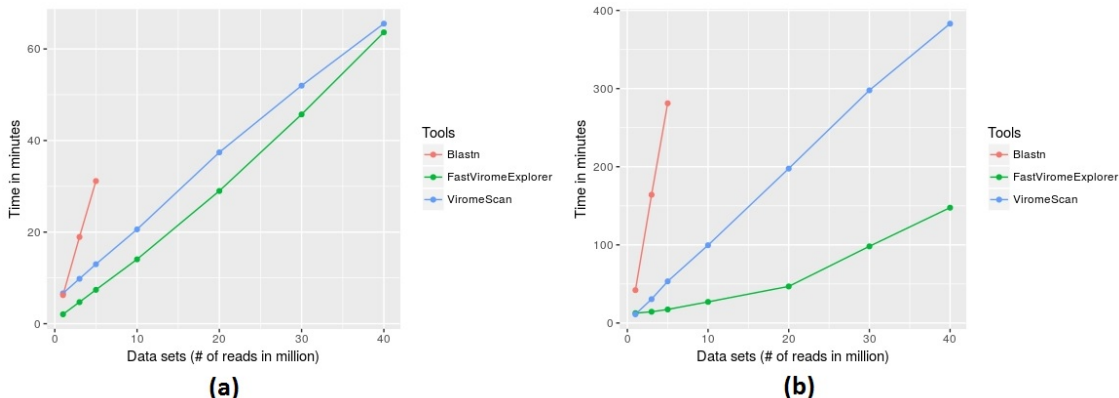


Figure 2.2: Comparison of running time among FastViromeExplorer, ViromeScan, and Blastn for seven data sets with 1, 3, 5, 10, 20, 30, and 40 million reads, respectively (a) against a reference database containing 8,957 NCBI RefSeq viruses, (b) against a reference database containing 125,842 mVCs

a similar F1 score to blastn.

To examine the performance of FastViromeExplorer in detecting and quantifying viruses on real data, we applied FastViromeExplorer to the fecal metagenomic samples collected from Lee et al. [32]. Lee et al. followed the dynamics and consequence of fecal microbiota transplantation (FMT) by examining the metagenomics data from a donor’s and recipients’ preFMT and postFMT samples. They constructed 92 bacterial metagenome-assembled genomes (MAGs) from reads of the donor samples and examined the occurrence of the MAGs in the recipient samples. They found that the bacterial MAGs that were present in the donor samples and also colonized the recipient samples after FMT mostly belonged to the order *Bacteroidales*. Here we examined the dynamics of viruses/phages to see whether it is consistent with the finding of Lee et al. [32].

From the result of FastViromeExplorer using the 8,957 NCBI RefSeq viral genomes as reference, we observed that only three viruses (Human endogenous retrovirus K113, Glypta fumiferanae ichnovirus segment C10, and Lactococcus prophage bIL311) were found in all four donor samples, with human endogenous retrovirus K113 being the most abundant for samples 1, 3, and 4, and Lactococcus prophage bIL311 the most abundant in sample 2. For the recipient, 30 viruses were found in the preFMT sample whereas only five were found in the two postFMT samples. Among the five viruses, only Lactococcus prophage was also found in one donor sample. But as this prophage was also present in the preFMT sample, we cannot conclude that the virus was transferred from the donor to the recipient. Overall, using the NCBI RefSeq database as the reference, we only detected 38 viruses in the FMT samples, and this result reveals no clear evidence of virus/phage transfer from the donor to the recipient. This result indicates that as our tool is a reference-based virus detection tool, having a suitable and/or complete reference database is important for performance.

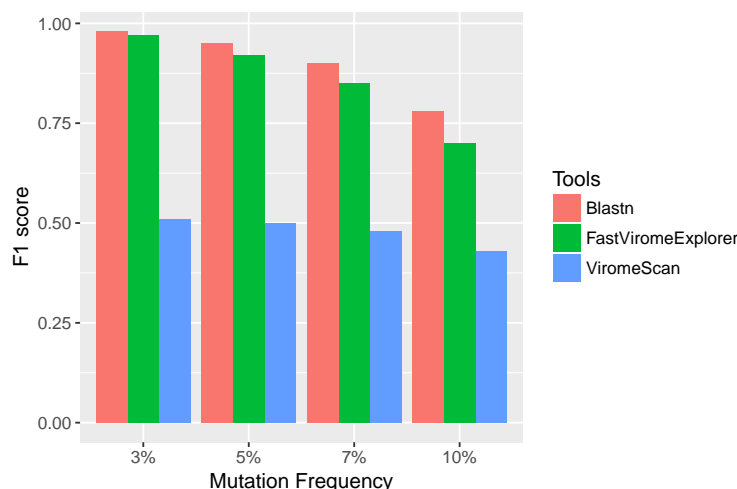


Figure 2.3: F1 score of FastViromeExplorer, ViromeScan, and Blastn when using NCBI eukaryotic viruses as the reference database and four simulated data sets of 1 million reads each with mutation frequency 3%, 5%, 7%, and 10% respectively.

We also applied ViromeScan to the fecal samples with its default reference database containing 4,370 eukaryotic DNA/RNA viruses. ViromeScan identified 847 viruses in all the samples. Compared to ViromeScan’s reference database, ours is two times bigger and it is thus surprising that ViromeScan identified a lot more viruses than FastViromeExplorer. Analysis of the ViromeScan result shows that the most abundant virus, Encephalomyocarditis virus, has all the reads mapped to a repeat region of its genome (see Appendix A, Figure A.3), indicating that the annotation is likely false positive. In fact, Encephalomyocarditis virus was also present in the initial result produced by FastViromeExplorer, but was discarded after the first filtering step. To further examine the effect of our three filtering criteria, we applied them to the ViromeScan result. Figure 2.4 shows that most of the viruses were filtered out and only Human endogenous retrovirus K113 and Glypta fumiferanae ichnovirus remained, both of which were also present in the final result of FastViromeExplorer. The finding here shows the importance of the filtering criteria in removing viruses that might be annotation artifacts caused by repeats, low coverage, and small genome sizes.

Since the analysis of the fecal samples using the default NCBI viral database did not reveal anything meaningful about fecal microbiota transplantation from the donor to the recipient, we tried FastViromeExplorer again using the 125,842 metagenomic viral contigs (mVCs) collected from Paez-Espino et al. [47] as reference. These mVCs are mostly unknown partial or complete viral genomes but have been predicted/annotated for their possible hosts and the host information of the mVCs is made available through the IMG/VR website [47, 46]. Therefore, the predicted host information of the mVCs, collected from the IMG/VR website, can be used to examine the result. Using these mVCs as reference, our tool detected 3,479 viral contigs in the FMT samples. Figure 2.5 shows the relative abundance of host

bacteria across all donor and recipient samples. The order *Bacteroidales* is more abundant than the order *Clostridiales* in all donor samples. For the recipient, prior to FMT, the order *Clostridiales* clearly dominated the microbiota, however, after the transplantation, the abundance of phages infecting the order *Bacteroidales* increased dramatically and the abundance of the order *Clostridiales* decreased greatly. This result indicates that phages with host bacteria from the order *Bacteroidales* were either successfully transferred or greatly enriched as a result of the microbiota transplantation from the donor to the recipient. For example, in donor samples, “SRS049900_LANL_scaffold_14438” is one of the most abundant mVC, being the most abundant in donor samples 1 and 2, and the second most abundant in samples 3 and 4. This mVC was not present in the recipient’s preFMT sample but was highly abundant in the postFMT samples, suggesting either the successful transferring of the mVC from the donor to the recipient or the great enrichment of the mVCs in the recipient as a result of FMT. As the host of this mVC is from the order *Bacteroidales*, this suggests the successful colonization of bacteria from the order *Bacteroidales* from the donor to the recipient. Therefore, our result on phage transfer following the FMT is consistent with the observation on bacterial colonization following the FMT shown in the original study [32]. The detailed annotation result is given in Appendix A, Table S2.

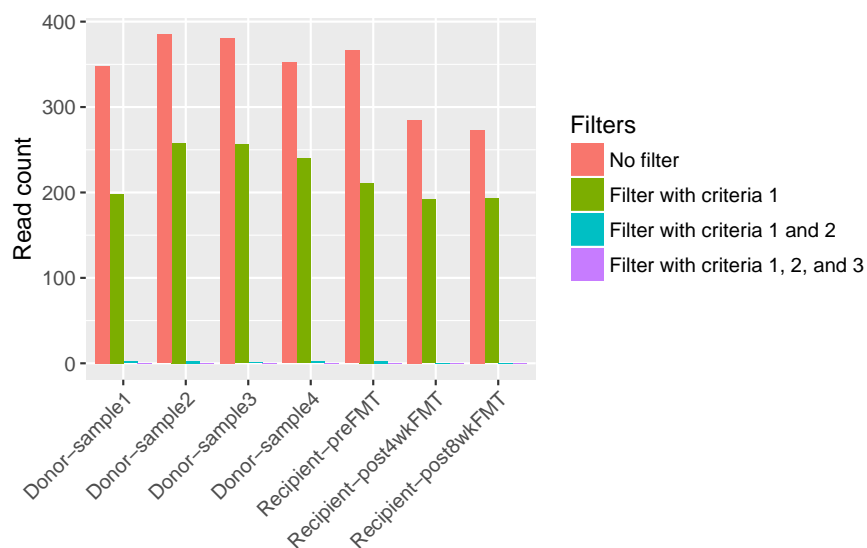


Figure 2.4: Number of viruses from ViromeScan result before applying any filter, after applying criterion 1, after applying criteria 1 and 2, and after applying all three criteria.

Consequently, when we applied FastViromeExplorer to the samples using a larger reference database, our tool detected 3,479 viral contigs which was much greater than the number of viruses detected using NCBI RefSeq database (38 viruses). Using a larger reference database, a much clearer correlation between our results and the biological results reported in the original paper emerges, highlighting the importance of having larger and more complete reference databases.

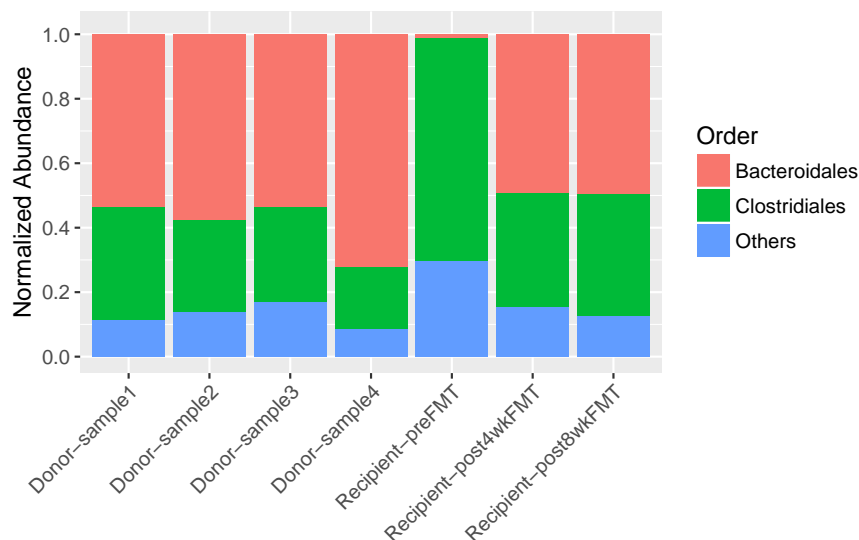


Figure 2.5: Relative abundance of host bacteria at Order level in the samples from FastViromeExplorer result using the 125,842 mVCs, where abundance is normalized by the total abundance of viruses in the sample.

We also applied FastViromeExplorer to ocean microbiome samples collected at multiple time points from Aylward et al. [3]. This study assembled 483 viral scaffolds (NCBI accession numbers NTLX01000001.1 to NTLX01000483.1) from the metagenome reads from 44 ocean samples. In our study, we tried to find if FastViromeExplorer could rapidly identify dominant viral components directly from the read files using both the JGI 125,842 mVC data set collected from Paez-Espino et al. [47] as well as the GOV dataset containing 298,383 epipelagic and mesopelagic viral contigs collected from [55]. Results show that FastViromeExplorer was able to successfully identify some contigs that resemble the original 483 scaffolds. For example, while using the 125,842 mVCs as reference, the most abundant mVC was “GOS2241_1000284”, which according to blastn aligns with scaffold “NTLX01000031.1” with identity 76.33% and alignment length 562 bp. Another example was mVC with id “JGI25127J35165_1001802” which has similarity with the longest scaffold “NTLX01000001.1”. In addition while using the GOV contig set [55] as reference, we identified an abundant contig with id “GOV_bin.1783_contig-100.1” with 98.35% identity and 2,004 bp alignment length with scaffold “NTLX01000307.1”. These successful hits could subsequently be used to assemble the actual viral sequences found in the samples. Taken together, our results show that FastViromeExplorer can also be applied to detect and quantify viruses and phages in metagenomic samples taken from environmental samples, and the results are accurate if given a sufficiently complete reference database.

2.4 Conclusion

Here, we develop a new tool FastViromeExplorer for detecting and quantifying viruses in metagenomic data. Worth emphasizing is that FastViromeExplorer can detect both viruses and phages depending on the reference database users deploy. As FastViromeExplorer can process millions of reads within minutes while having similar virus detection accuracy to the gold standard tool blastn, it empowers researchers, that have limited computing power, to process large metagenomic data within reasonable time. Similar to all other reference database based tools, the limitation of FastViromeExplorer is that it cannot identify a virus or phage if a similar sequence is not present in the reference database, so our tool cannot be used to identify or recover novel viruses that have no similarity to sequences in the reference database. Our preliminary results for the human microbiome and ocean environmental data highlight the pressing issue of building and/or extending the current viral sequence database for improving virus/phage detection and quantification in metagenomic data.

Chapter 3

FVE-novel: Discovery of Draft Genomes of Novel Viruses and Phages in Metagenomic Data

3.1 Introduction

Although viruses are the most abundant biological entities and a critical component of ecosystems around the globe, around 99% of viruses remain unknown [75]. Compared to prokaryotes the diversity of viruses in the biosphere has been relatively underexplored until recently. As a result, the number of complete virus genomes present in NCBI RefSeq database is much less than the number of complete bacteria genomes [44].

Recovering virus or phage genomes from metagenomic samples generally requires the identification of viral reads in the sample and then assembling those viral reads. We recently developed FastViromeExplorer (FVE), a computational pipeline that uses a reference-based approach for quick and accurate identification of viral reads [67]. FastViromeExplorer uses pseudoalignment approach to quickly align reads to a database of viral sequences to identify the presence and relative abundance of known viral groups in a metagenomic sample. Here we expand upon FastViromeExplorer and include a complimentary *de novo* assembly step which allows for viral sequences to be recovered even if only a small fragment is present in the reference database used.

Although a combination of strategies have been used to recover novel virus genomes (e.g., [15, 42]), a readily usable pipeline that integrates both reference-based read mapping and *de novo* assembly to recover novel virus genomes is lacking. Here we developed FVE-novel for this purpose. FVE-novel leverages FastViromeExplore [67] to quickly identify viral reads from the metagenomic data. Then it assembles the viral reads to generate short viral scaffolds. As FVE-novel starts with only viral reads, it avoids assembling bacteria or archaea genomes

present in the sample, making the assembly faster than the traditional approaches that start the assembly using all the reads in the sample. FVE-novel grows each scaffold as much as possible using local iterative assembly and generates viral scaffolds. FVE-novel uses SPAdes tool to do the assembly step and to generate scaffolds [4, 43]. Those scaffolds are then compared to the input references and the average nucleotide identity (ANI) is used to determine whether the viral scaffolds are potential novel viruses. Finally, FVE-novel reports potential novel viral scaffolds together with their per base depth of coverage, which can be used to examine the quality of the generated scaffolds and whether they are chimeric. FVE-novel is freely available at <https://github.com/saima-tithi/FVE-novel>.

3.2 Methods

3.2.1 Algorithm Overview

The inputs of FVE-novel are reads and reference genomes/contigs. As reference databases often contain genomes/contigs similar to each other, our tool needs to preprocess the reference database by binning the genomes/contigs based on similarity. After processing the input database, our tool starts generating and extending viral scaffolds. This task can be divided into three main steps, (1) the read mapping and scaffold generation step, (2) the scaffold extension step where scaffolds generated in previous step or seed scaffolds are extended through iterative assembly, and (3) generating ANI and coverage statistics step for all extended scaffolds. Figure 3.1 shows a comprehensive outline of all the steps of FVE-novel tool.

3.2.2 Preprocessing the Reference Database

In this step, the reference database is processed by binning the genomes or contigs in the reference based on sequence similarity. FVE-novel was tested on two reference databases, 125,842 metagenomic viral contigs (mVCs) collected from JGI “earth virome” study [47] and 24,411 viral contigs collected from Global Ocean Virome (GOV) study [55]. For these two reference databases, files with binning information are generated and distributed with FVE-novel. The reference database containing 125,842 mVCs is binned using Mash [45]. The Mash distance between all pairs of mVCs are calculated using k-mer size 11 and sketch size 10,000. Here small k-mer size is used as many viral contigs in the database are short and large sketch size is used to increase sensitivity of the result. Mash distance 0.15 is used as cutoff value to pair the genomes, which equates to about 85% ANI. Thus all the genomes that have pairwise mash distances less than or equal to 0.15 are paired together. Then all the genome pairs are used to form maximal cliques using Bron-Kerbosch algorithm [16]. All maximal cliques are again grouped together if they share any common genome. The 125,842

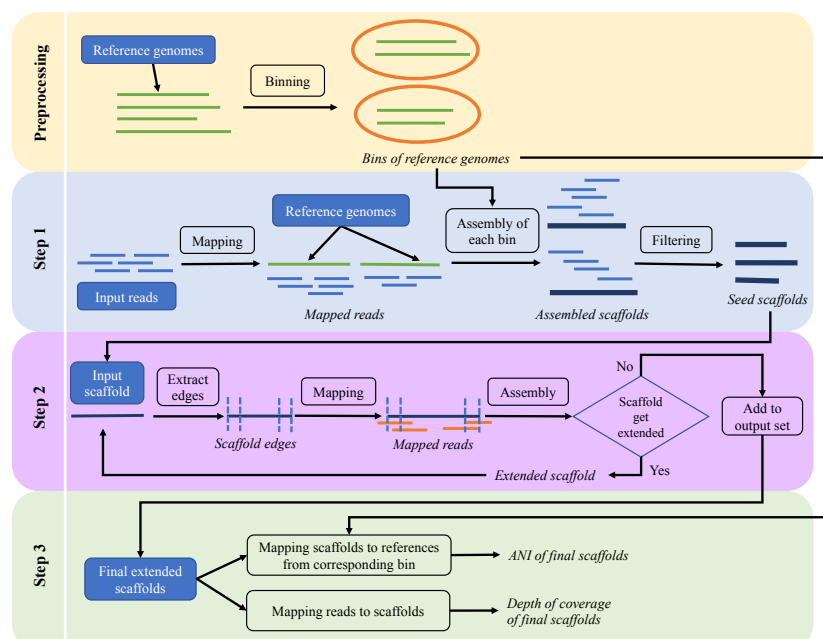


Figure 3.1: Overview of the FVE-novel pipeline, where the inputs are single-end or paired-end reads and reference database and the output is a set of final extended scaffolds along with ANI and depth of coverage of the output scaffolds.

mVCs are grouped into 21,409 bins, where 81,624 mVCs belong to these bins and 44,218 mVCs remain unbinned, and the largest bin contains 2,763 mVCs and the smallest one contains two mVCs. For the GOV reference database, the binning from the original study [55] is used: 24,411 viral contigs are grouped into 15,280 bins, with 15,222 bins corresponding to epipelagic and mesopelagic viral populations and 58 bins corresponding to bathypelagic viral populations.

3.2.3 Step 1. Read Mapping and Generating Seed Scaffolds

In the first step, FVE-novel takes single-end or paired-end reads and a reference database as input and invokes FastViromeExplorer for read mapping/alignment. FastViromeExplorer outputs the viruses present in the sample and their abundances, and the reads mapped to the viruses. As metagenomic data contains reads from various organisms, using FastViromeExplorer, all the reads coming from bacteria, archaea, and other hosts are quickly filtered out and only viral reads are retained for the next assembly step. In this way, assembling genomes other than viruses can be avoided, making assembly much faster and more efficient. After getting the mapping result from FastViromeExplorer, all the mapped reads are binned based on the binning of reference genomes. Here all the reads mapped to the reference contigs/genomes from the same bin are binned together.

After read mapping, depths of coverage for the contigs/genomes that have mapped reads are calculated. For each bin, if all the contigs/genomes in the bin have $<3x$ coverage the bin is discarded. The rationale for the threshold is that the genomes that have low coverage tend to have reads sparsely aligned to the genomes and assembling the reads often leads to a few short scaffolds that are useless for recovering the original genomes. Therefore, to speed up the process, we applied this filter. However, realizing that this may eliminate the low abundance genomes in the sample, FVE-novel allows users to adjust the coverage cutoff to suit their own studies. After filtering the bins based on coverage, reads belonging to the same bin are assembled together. Based on the type of input reads, either SPAdes (version 3.10.1, for single-end reads) or metaSPAdes (option: `-meta`, for paired-end reads) is used with default settings and with the assembly mode (option: `-only-assembler`). Here, SPAdes break the reads into fixed-length k-mers, build a de bruijn graph using the overlap of k-mers, and traverse the graph to produce longer sequences or genome fragments. After the initial assembly step, we have added another assembly step to extend each scaffold as much as possible. This extension step is required because assemble the reads once is usually not enough for recovering whole genome. Theoretically, if a virus is fully covered by the reads, then its full-length genome sequence can be recovered by the overlapping of the k-mers generated from the reads. However studies show that the heterogeneous nature of metagenomic data with many organisms present in highly uneven coverages and the presence of many strains of the same virus make the recovery of full virus genomes difficult [20]. As a result, de novo assemblers often produce fragments of a genome instead of a full-length genome [64, 20, 70]. That's why a second iterative assembly step or scaffold extension step is implemented in FVE-novel to recover whole genome sequences.

After the initial assembly step, all the generated scaffolds are examined and only the ones longer than 2 kb are kept. Similar to the coverage parameter, the 2 kb cutoff is implemented so that the pipeline does not attempt to extend every short segment that can be time consuming yet contribute little to the overall scaffolds. Similarly, the 2kb cutoff can also be changed tailoring to the specific research need. After removing scaffolds shorter than the cutoff length, the remaining scaffolds are clustered using CD-HIT with 95% global average nucleotide identity and word size $n = 10$ (i.e., “cd-hit-est” program with options: `-c 0.95 -n 10`) [19]. With the setting, CD-HIT clusters all sequences with more than 95% identity into a group and outputs one representative sequence for the group. Only the representative sequences are kept by FVE-novel and therefore if some scaffolds are very similar to one another, all except the longest one will be discarded. This step also aims to eliminate redundant scaffolds to speed up the process. Then the remaining scaffolds are used as seed scaffolds or input scaffolds for the next extension step.

3.2.4 Step 2. Extending Seed Scaffolds using Iterative Assembly

In the second step or extension step of FVE-novel tool, for each seed scaffold, start and end edges are extracted using BEDTools [51]. Read length * 1.5 is used as default edge length.

Then for each seed scaffold, all the reads are mapped to the two edges of that scaffold using Salmon [48]. Salmon uses the pseudoalignment approach and does not align the entire read, so it can report reads that are only partially mapped to the edges of a scaffold. It is found that assembling the original scaffold along with the overhanging reads mapped to the edges of that scaffold can effectively extend the scaffold in one or both ends. For this purpose, SPAdes is used for the assembly step (with option: `-only-assembler`) using the original scaffold as `-trusted-contigs` parameter and the reads mapped to the edges of the scaffold as input reads. The mapping-and-then-assembly step is run iteratively for each scaffold until the scaffold stops growing. After getting all the extended scaffolds, they are clustered again using CD-HIT with 95% global average nucleotide identity so that for scaffolds with high sequence identity, only the longest one is kept. Finally FVE-novel outputs the remaining scaffolds as the final extended scaffolds.

3.2.5 Step 3: Analysis of New Contigs and Comparison to Reference Sequences

The third and final step of FVE-novel involves generating some statistics for the scaffolds including ANI, percentage of aligned nucleotides, and per base depth of coverage. As each scaffold is assembled from the reads mapped to a set of reference genomes belonging to a bin, the reference genomes in the bin are used for calculating ANIs of the corresponding scaffold using MUMmers “dnadiff” program [29]. The output of FVE-novel contains ANI and the percentage of aligned nucleotides between the extended scaffold and each of its reference genomes. All the reads are mapped back to each scaffold using Bowtie2 [31] and then depth of coverage is calculated for each base pair of the respective scaffold using Samtools [35]. The summary statistics can be used to evaluate the quality and the novelty of the scaffolds generated in step 2. For example, a high percentage of ANI and aligned nucleotides with any of the reference genomes indicates that the generated scaffold might be an existing viral genomic sequence. On the other hand, a low percentages of ANI and/or low percentages of aligned nucleotides with all the reference genomes indicate that the scaffold might be a novel viral genomic sequence. Also if the depth of coverage is fairly uniform along the scaffold, it is less likely to be chimeric.

3.2.6 Benchmarking on Real Data

To demonstrate and evaluate the performance of FVE-novel, we downloaded the 12 samples from the original Aylward et al. [3] (NCBI SRA accession numbers: SRX2912986, SRX2912968, SRX2912972, SRX2912964, SRX2912992, SRX2912996, SRX2912975, SRX2912979, SRX2912983, SRX2912998, SRX2913002, and SRX2912985), corresponding to the metagenomic samples for the same location/station but at different time points (Detailed description of all stations is given in Appendix B, table B.1). We applied FVE-novel to the sample

SRX2912986 that contains 18,471,506 paired-end reads with read length 151 bps and used the GOV database as reference. Other samples were used to evaluate the scaffolds generated by FVE-novel. A Linux based cluster with 64 CPUs and 128 GB RAM was used to generate all the results.

3.3 Results

3.3.1 Length Distribution of the FVE-novel Scaffolds

We applied the FVE-novel pipeline to the ocean virome sample collected from Aylward et al. [3] under the accession number SRX2912986 (HOE Legacy II diel viral-size metagenome: Station 70). This sample contains 18,471,506 paired-end reads with read length 151 bp. By using the GOV database containing 24,411 contigs as reference and the reads from the station 70 as input, our tool generated 268 scaffolds. Figure 3.2 shows the length distribution of the scaffolds, which ranges from 2,026 bp to 193,112 bp, with a median length of 4,561 bp. There are 66 scaffolds with lengths greater than 10 kb, which could potentially be nearly complete or complete viral genomes.

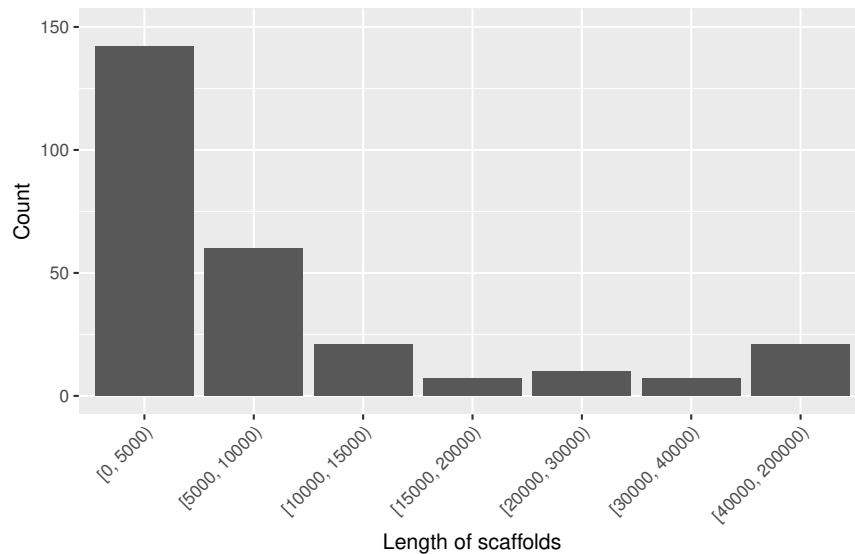


Figure 3.2: Length distribution of the 268 scaffolds generated by FVE-novel for the ocean metagenome sample.

3.3.2 Comparison Between the FVE-novel Scaffolds and Their Reference Sequences

As each scaffold is assembled from a set of reads mapped to a set of contigs or genomes from the same bin, the set of contigs or genomes can be considered as the “reference genomes” of the scaffold. Because the reads used for assembly were originally obtained by read mapping, it is important to assess how similar the newly assembled scaffolds are to the original references used. Among the 268 scaffolds generated by our tool, 59 are longer than their reference genomes in the GOV database. Figure 3.3 shows the lengths of the 59 scaffolds with respect to those of their reference genomes (here the reference genome with the highest ANI is selected), indicating that FVE-novel can generate considerably longer scaffolds than their original references. Moreover, the observation that some scaffolds have high ANIs (e.g., >95%) to their references suggests that parts of these scaffolds are present in the reference database and FVE-novel successfully extended those partial references. On the other hand, some scaffolds have low ANIs to their reference genomes, indicating that these scaffolds are potential novel viral genomes that are only similar to reference sequences over short stretches of their genome. Interestingly, some scaffolds have different lengths and ANI even though originated from the same reference genome, which could be the product of different parts of the same genomes.

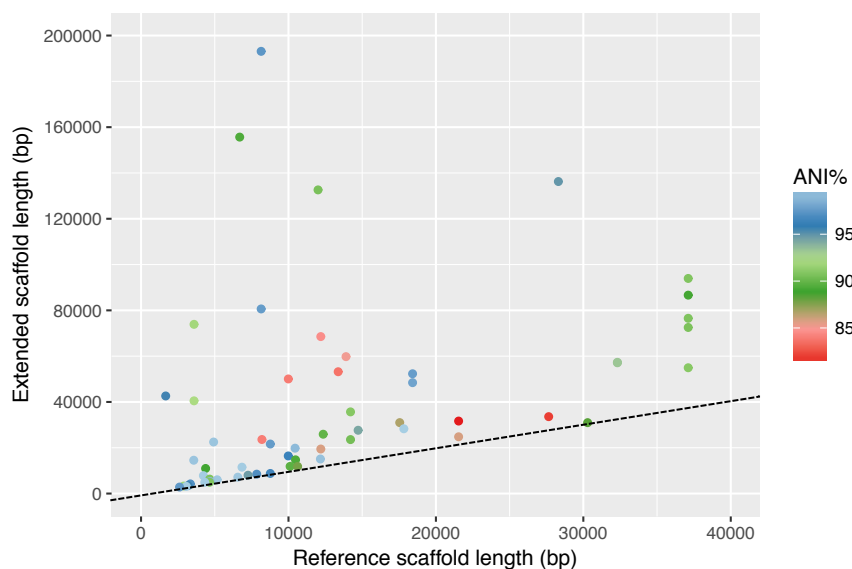


Figure 3.3: The length comparison of the 59 scaffolds against their corresponding references (GOV database). These 59 scaffolds are a subset of the total 268 scaffolds which are extended by FVE-novel tool. The dotted line represents 1:1 ratio between the x and y axis. The color of the dots represents the ANI between the scaffolds produced by FVE-novel and their references.

3.3.3 Comparison of the FVE-novel Scaffolds Against Several Databases

We were interested in the possibility of obtaining complete viral genomes from metagenomic data, so we analyzed the longest ten scaffolds, *S0*, *S1*, *S2*, *S3*, *S4*, *S5*, *S6*, *S7*, *S8*, and *S9* produced by FVE-novel in more detail. To examine how similar the scaffolds are from existing viral sequences, we used BLASTN [2] to compare the scaffolds against the nr nucleotide database and the 483 assembled scaffolds produced by Aylward et al. [3], in addition to the GOV database based on which the scaffolds are assembled. Table 3.1 shows the comparison of the ten scaffolds against the GOV database and table 3.2 shows the comparison against the nr nucleotide database and the 483 assembled scaffolds produced by Aylward et al. The scaffolds range from 72,532 bps to 193,112 bps, with average coverage ranging from 48.5x to 338x. Their ANIs (%) to the GOV database sequences range from 89.21 to 97.78, alignment percentage 6.25-99.98; their ANIs to the 483 scaffold sequences range from 95.25 to 99.73, alignment percentage 38.69-100; their ANIs to the nr nucleotide database sequences range from 84.89 to 92.68, alignment percentage 14.36-51.53.

Name	Length (bp)	Avg coverage	GOV database			
			Name	Length (bp)	ANI%	Aligned nucleotide%
S0	193,112	93.0	GOV_bin_1296_contig-100_1	8,146	97.51	97.51
S1	155,659	103.6	GOV_bin_892_contig-100_6	6,690	89.5	6.25
S2	136,254	48.5	Tp1_32_SUR_0-0d2_scaffold20227_1	28,306	95.06	99.98
S3	132,604	338.0	Tp1_18_DCM_0-0d2_scaffold48213_1	12,004	90.73	86.78
S4	93,939	67.4	Tp1_18_DCM_0-0d2_scaffold31884_1	37,113	90.95	92.4
S5	86,648	327.2	Tp1_18_DCM_0-0d2_scaffold31884_1	37,113	89.21	37.41
S6	80,620	102.9	GOV_bin_1296_contig-100_1	8,146	97.78	93.09
S7	76,528	312.4	Tp1_18_DCM_0-0d2_scaffold31884_1	37,113	90.88	38.5
S8	73,892	123.2	GOV_bin_5370_contig-100_47	3,592	92.03	99.81
S9	72,532	223.1	Tp1_18_DCM_0-0d2_scaffold31884_1	37,113	90.78	55.00

Table 3.1: The longest ten scaffolds of the 268 scaffolds generated by applying FastViromeExplorer-novel to the ocean metagenome sample collected from Aylward et al. showing ANI and aligned nucleotide percentage between the reference (GOV database) scaffold and the extended scaffold.

Name	483 scaffolds				BLAST nr nucleotide database			
	Name	Length (bp)	ANI%	Aligned nucleotide %	Name	Length (bp)	ANI%	Aligned nucleotide %
S0	NTLX01000256.1	8,327	98.08	64.33	No significant hit			
S1	NTLX01000256.1	8,327	98.19	64.33	No significant hit			
S2	NTLX01000174.1	10,777	99.73	100.00	No significant hit			
S3	NTLX01000012.1	37,975	99.25	100.00	Prochlorococcus phage P-SSM4 (AY940168.2)	178,249	86.69	23.34
S4	NTLX01000030.1	27,949	97.63	100.00	Prochlorococcus phage P-SSM4 (AY940168.2)	178,249	92.68	51.53
S5	NTLX01000006.1	43,974	99.49	58.77	Prochlorococcus phage P-SSM4 (AY940168.2)	178,249	86.46	14.54
S6	No hit found				No significant hit			
S7	NTLX01000006.1	43,974	99.36	38.69	Prochlorococcus phage P-SSM4 (AY940168.2)	178,249	87.39	14.36
S8	NTLX01000263.1	8,048	99.40	100.00	Prochlorococcus phage P-HM2 (GU075905.1)	183,806	84.89	17.00
S9	NTLX01000012.1	37,975	95.25	49.32	Prochlorococcus phage P-SSM4 (AY940168.2)	178,249	91.09	25.38

Table 3.2: The ten scaffolds are compared with the 483 scaffolds assembled in the original Aylward et al. study and blast nr nucleotide database using BLAST, the best hit from BLAST are selected, and ANI and aligned nucleotide percentage for the best BLAST hit result are calculated using MUMmer tool

As the scaffolds are assembled using the GOV database as the input reference database, it is expected that the scaffolds have sequence similarity to the GOV database sequences. Figure 3.4, generated using Artemis [12] shows how the scaffolds align to their corresponding references in the GOV database and the sequence similarity, further supporting that FVE-novel successfully extended the reference genomes.

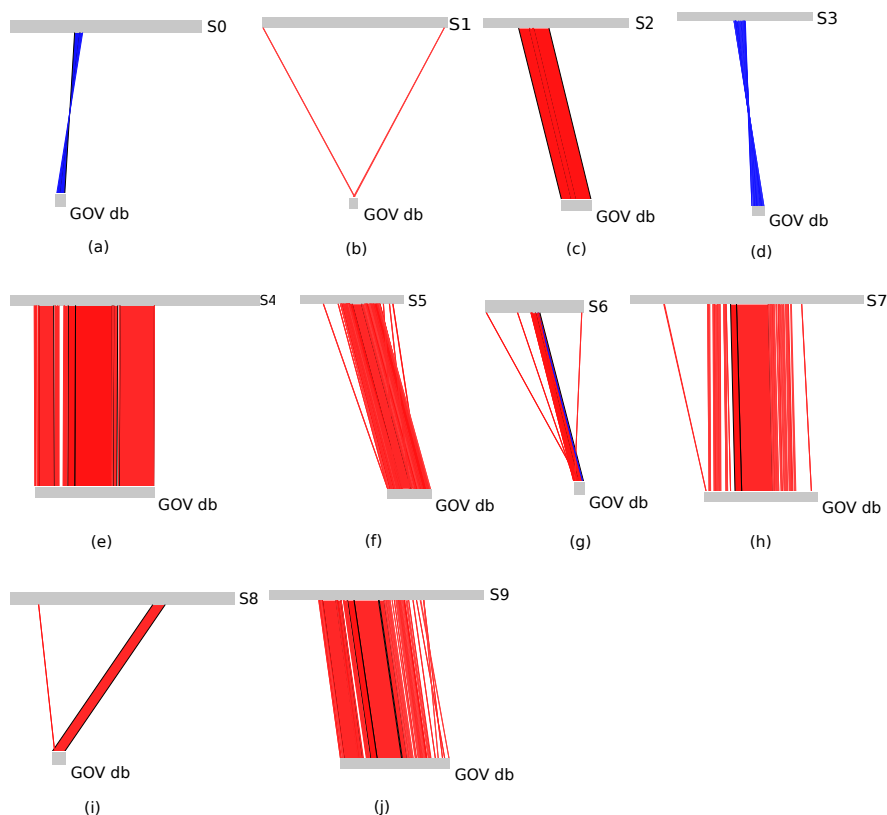


Figure 3.4: Alignment of the ten scaffolds to their corresponding references in GOV database. Here the red and blue lines represent the forward and reverse alignment respectively and the intensity of the color is proportional to the percent identity of the alignment.

To see how the FVE-novel scaffolds compared to the 483 scaffolds reported in the Aylward et al study, we used BLASTN [2] to compare the scaffolds against the 483 assembled scaffolds. Table 3.2 shows the best hit result. Nine FVE-novel scaffolds have very high ANIs to their best hits (95.25 to 99.73%). This is not surprising as both sets of scaffolds are assembled from the same read samples. Scaffold *S6* does not have any hits in the 483 scaffolds, but has a best hit to the GOV database with 97.78% ANI and 93.09% alignment length. Therefore, our tool not only generates but also extends successfully the original references. For all the nine scaffolds, our tool successfully generated longer scaffolds than the scaffolds from the original study.

We also used BLASTN to compare the ten scaffolds against the nr nucleotide database and

found that four scaffolds (S_0 , S_1 , S_2 , and S_6) have no significant hits and the other six scaffolds (S_3 , S_4 , S_5 , S_7 , S_8 , and S_9) have similarity to *Prochlorococcus phage P-SSM4* (ANIs ranging from 86.46 to 92.68%) and *Prochlorococcus phage P-HM2* (ANI 84.89%). This result is consistent with the original study [3] where parts of the *Prochlorococcus* phage genomes were also recovered from the data.

3.3.4 Comparison Within the FVE-novel Scaffolds

The observation that some of the ten scaffolds have similar BLAST hits suggests that their sequences might be similar. Therefore we also compared the scaffolds against each other (Figure 3.5). Four groups of scaffolds are observed: group 1 containing S_0 , S_1 , and S_6 , group 2 containing only S_2 with no significant similarity to any other nine scaffolds, group 3 containing S_3 , S_4 , S_5 , S_7 , and S_9 , and group 4 containing only S_8 . As our tool filtered out all scaffolds with identity greater than 95%, these scaffolds have similarity less than 95% and they can be different strains of the same virus. Group 3 scaffolds (S_3 , S_4 , S_5 , S_7 , and S_9) all have similarity to *Prochlorococcus phage P-SSM4* (Table 3.2), thus likely represent part of different strains of *Prochlorococcus phage P-SSM4*. In the following, we analyzed the four groups of scaffolds in detail.

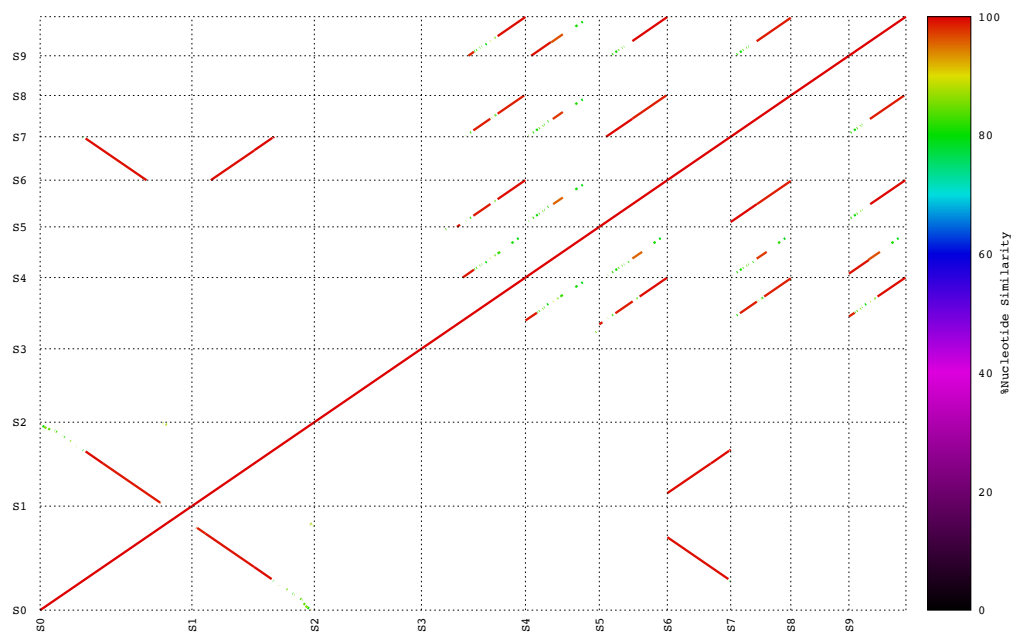


Figure 3.5: Percentage of similarity between each pair of the longest ten scaffolds of the 268 scaffolds generated by applying FVE-novel to the ocean metagenome sample.

Group 1 Scaffolds (S0, S1, S6)

S0, the longest scaffold in group 1, has a length 193,112 bp and is generated from the station 70 sample, thus we checked if this scaffold is also present in the other 11 viral metagenomic samples. Figure 3.6a shows the depth of coverage of *S0* in all 12 samples. The average depth of coverage of *S0* in station 6, 14, 18, 22, 28, 32, 37, 52, 56, 61, 67, and 70 samples are 95.26, 28.39, 32.24, 94.78, 52.08, 73.75, 17.71, 20.18, 82.55, 148.61, 85.54, and 93.02 respectively, with station 37 having the lowest coverage (17.71) and station 61 (148.61) the highest coverage. The pairwise Pearson correlation coefficient of per base coverage of *S0* between any two samples ranges from 0.77 to 0.94, and therefore even though the abundances of *S0* differ among stations, the per base read coverages along the scaffold are highly correlated.

Figure 3.6a also shows that coverage dropped greatly after 150 kb for all samples, which means fewer reads got mapped to this region. We further analyzed this region, but could not find any homopolymer or short repeats which can cause lower mapping rate. Another possible cause of coverage drop can be that this part of the scaffold got contaminated by another strain of the same virus or another virus that have sequence similarity in this region. Here, we explored this possibility by reassembling the scaffold and used the “Map to Reference” algorithm implemented in Geneious 11.0.4 [28] to identify whether there are multiple phage strains and to see whether we can generate a complete assembly of the dominant strain. Specifically, all of the metagenome reads in the station 70 sample were aligned to scaffold *S0* using the “Low sensitivity/Fastest” settings allowing for 10% mismatches. The alignment revealed a large number of Single Nucleotide Polymorphisms (SNPs), indicating two or more separate strains for this phage as well as distinct regions of high and low coverage associated with the initial chimeric assembly of these strains. Then the consensus sequence from the alignment was segmented into contigs with the highest coverage > 40X. These contigs were binned into lists of contigs with similar coverage for further assembly. Next, the contigs in each bin were iteratively grown using Geneious by mapping reads to the ends with high stringency. Specifically, all of the phage metagenome paired-end reads were aligned to these high coverage contigs using “Map to Reference” with stringent “Custom Sensitivity” settings allowing no more than 1% “Mismatches per Read” and 1% “Gaps per Read” and requiring that both of the paired-end reads map to the new consensus sequence. This process was iterated for each contig using the Geneious “Fine Tuning” settings up to “100 times”. This process was continued until the extended contigs merged together, maintaining approximately uniform coverage, and could no longer be extended or closed into a circular genome sequence. Using Geneious, we recovered a 153 kb scaffold from scaffold *S0* with uniform coverage across all 12 samples (Figure 3.6b). Comparison of this 153 kb scaffold with *S0* reveals that the middle 90 kb of *S0* (starting at 60 kb and ending at 150 kb) are exactly the same as the 153 kbp strain curated by Geneious (Appendix B, Figure B.1). But the first 60 kb of *S0* has around 80% similarity with 153 kb strain, so this part of *S0* could be from a different strain and the last 40 kb of *S0* (starting at 150 kb and ending at 193 kb) is the result of assembly artifact (Appendix B, Figure B.1). By comparing the 153 kb

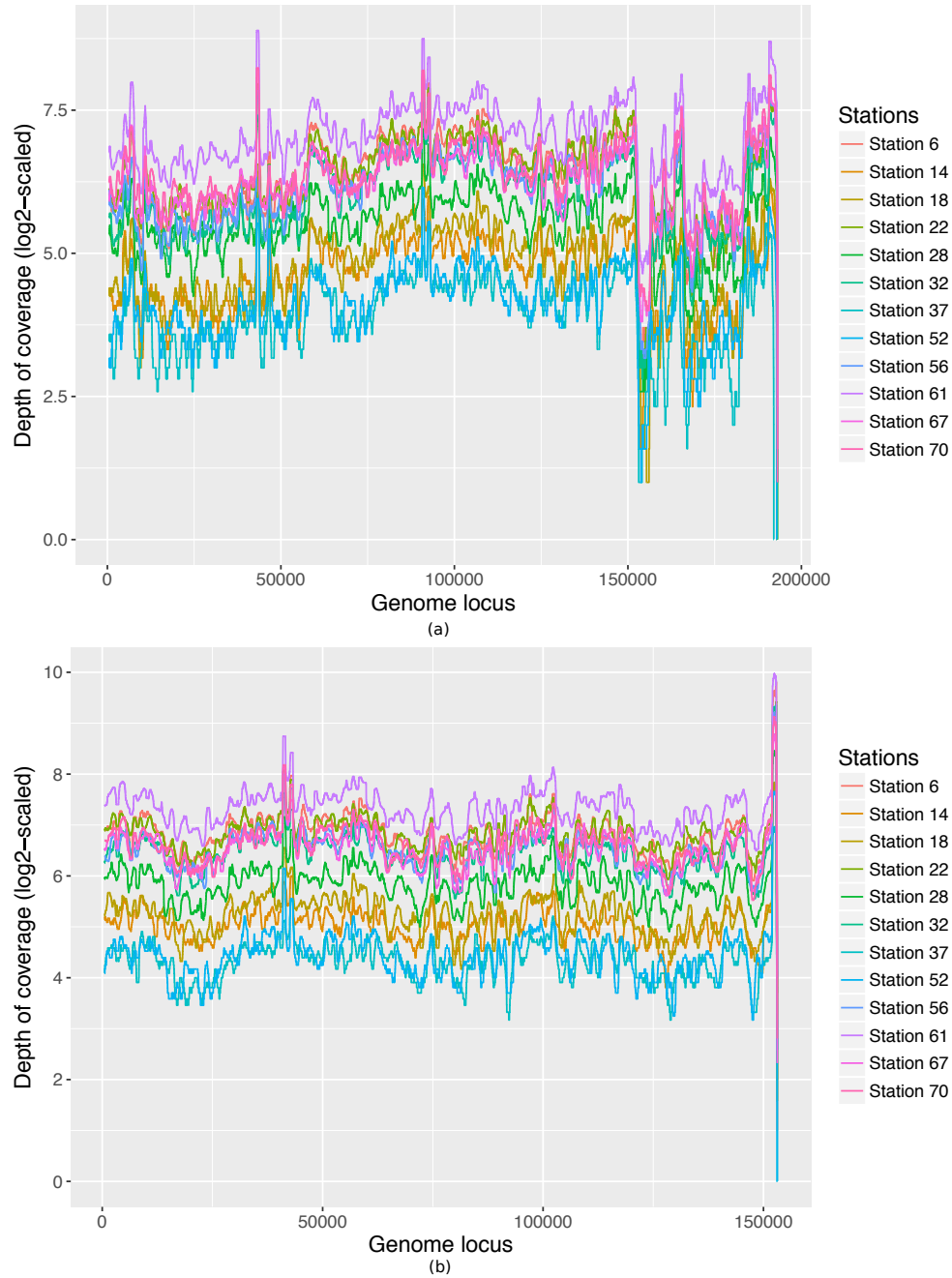


Figure 3.6: The \log_2 -scaled depth of coverage of (a) S_0 (193 kb) and (b) 153 kb scaffold representing the dominant strain of the novel virus (recovered from S_0) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).

strain with $S1$ (155 kb), we found that our tool recovered this dominant strain successfully in $S1$, with an extra 2 kb at the end of the scaffold due to misassembly (Appendix B, Figure B.2). As scaffold $S1$ is a correct representation of the dominant strain of this novel virus, from figure 3.7, we can see that $S1$ has a uniform coverage across all 12 samples. In scaffold $S6$, our tool captured the first 80 kbp of the 153 kbp dominant strain (Appendix B, Figure B.3) and $S6$ also has a uniform depth of coverage across all 12 samples (Appendix B, Figure B.4). In order to find out if this virus has multiple strains present in station 70 sample, we sub-sampled the 153 kb strain into 10 pieces where each piece is 20 kb long and applied TenSQR [1], a viral quasispecies reconstruction tool, to each piece. For each piece, TenSQR reported two or three strains, consistent with our observation that multiple strains of this virus are present in the station 70 sample (Appendix B, Table B.2).

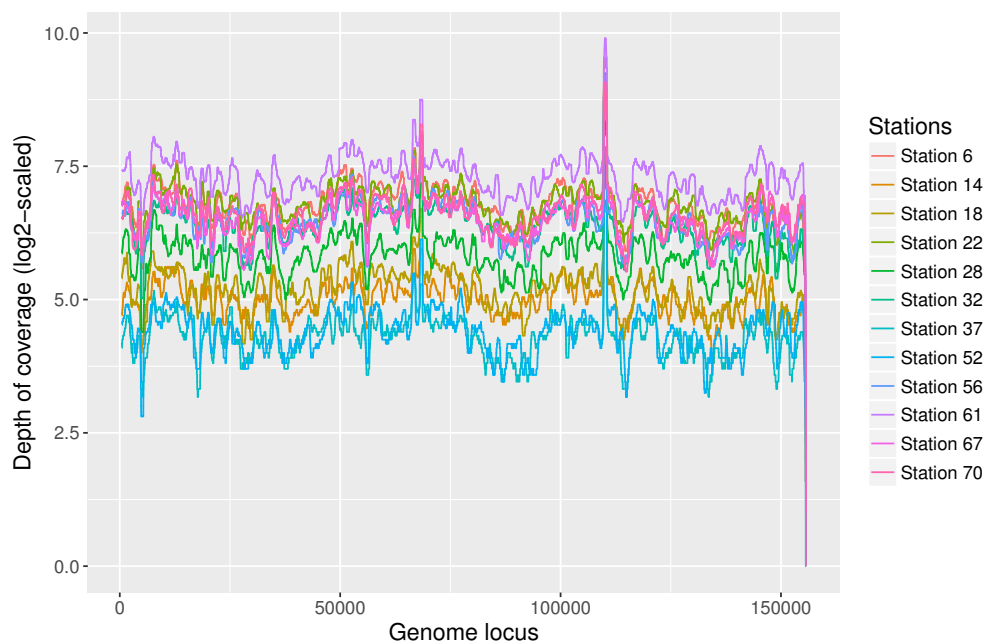


Figure 3.7: The \log_2 -scaled depth of coverage of $S1$ (155 kb) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).

Group 2 Scaffold (S2)

We then checked the presence of scaffold $S2$ in all the 12 samples and found that it is also present in all samples with varying abundances but highly correlated depth of coverage among the samples (Figure 3.8a). For this scaffold, we also tried to extend it more to get the complete genome of this novel virus using Geneious. With the same procedure, we generated a 151 kb scaffold that most likely represents a complete novel virus genome recovered from $S2$. Figure 3.8b shows that this 151 kb scaffold has a uniform depth of coverage across all 12 samples. Comparison of this scaffold to $S2$ shows that our tool successfully recovered most

of this virus except a 15 kb long portion of it (Appendix B, Figure B.5). We also checked if multiple strains of this virus are present in the station 70 sample using TenSQR. Results show that multiple strains of this virus are also present in this sample (Appendix B, Table B.3).

Group 3 Scaffolds (S3, S4, S5, S7, S9)

As scaffolds *S3*, *S4*, *S5*, *S7*, and *S9* have similarity to *Prochlorococcus phage P-SSM4*, from here we analyzed the longest scaffold *S3* with length 132,604 bp. Figure 3.9a shows consistent per base depth of coverage of *S3* across all 12 samples, but coverage dropped dramatically in all samples from 50 kb to 65 kb. We then aligned all the reads from station 70 sample to scaffold *S3* using Geneious and observed the presence of multiple strains. Using the same procedure as above, we were able to recover a 177 kb scaffold representing the dominant strain of *Prochlorococcus phage P-SSM4* present in station 70 sample, with a uniform coverage across all 12 samples (Figure 3.9b). Comparison of this 177 kb scaffold to *S3* shows that *S3* matches the dominant strain with about 100% similarity for all except the 15 kb segment (the 50-65 kb part in *S3*) where the similarity dropped to 80% (Appendix B, Figure B.6). This result implies that our tool successfully captured about 117 kb of the dominant strain except from a piece of length 15 kb where the assembly switched to a different strain with lower coverage. Consistently, analysis with TenSQR suggests the presence of three to seven strains in the sample (Appendix B, Table B.4). Interestingly, alignment of the 177 kb scaffold to *Prochlorococcus phage P-SSM4* (length 178,249 bp) shows similar as well as some dissimilar regions (ANI 82.9%) (Appendix B, Figure B.7).

Group 4 Scaffold (S8)

The depth of coverage of the scaffold *S8* is quite consistent across all 12 samples along the region except the beginning 6 kb and the last 10 kb where coverage is higher than the remaining region, which can be caused by a different strain (Figure 3.10a). According to the blast result, *S8* has similarity to *Prochlorococcus phage P-HM2*. We then used Geneious to recover the dominant strain from *S8* and obtained a 183 kb scaffold. Figure 3.10b shows that the 183 kb scaffold has uniform coverage across all 12 samples. Comparison to *S8* shows that in *S8* we recovered about 60 kb of this dominant strain (Appendix B, Figure B.8). Analysis with TenSQR suggests the presence of two or three strains in this sample Appendix B, Table B.5. Alignment of the 183 kb scaffold to *Prochlorococcus phage P-HM2* (length 183,806 bp) also shows many similar regions and some dissimilar regions (ANI 86.56%) (Appendix B, Figure B.9).

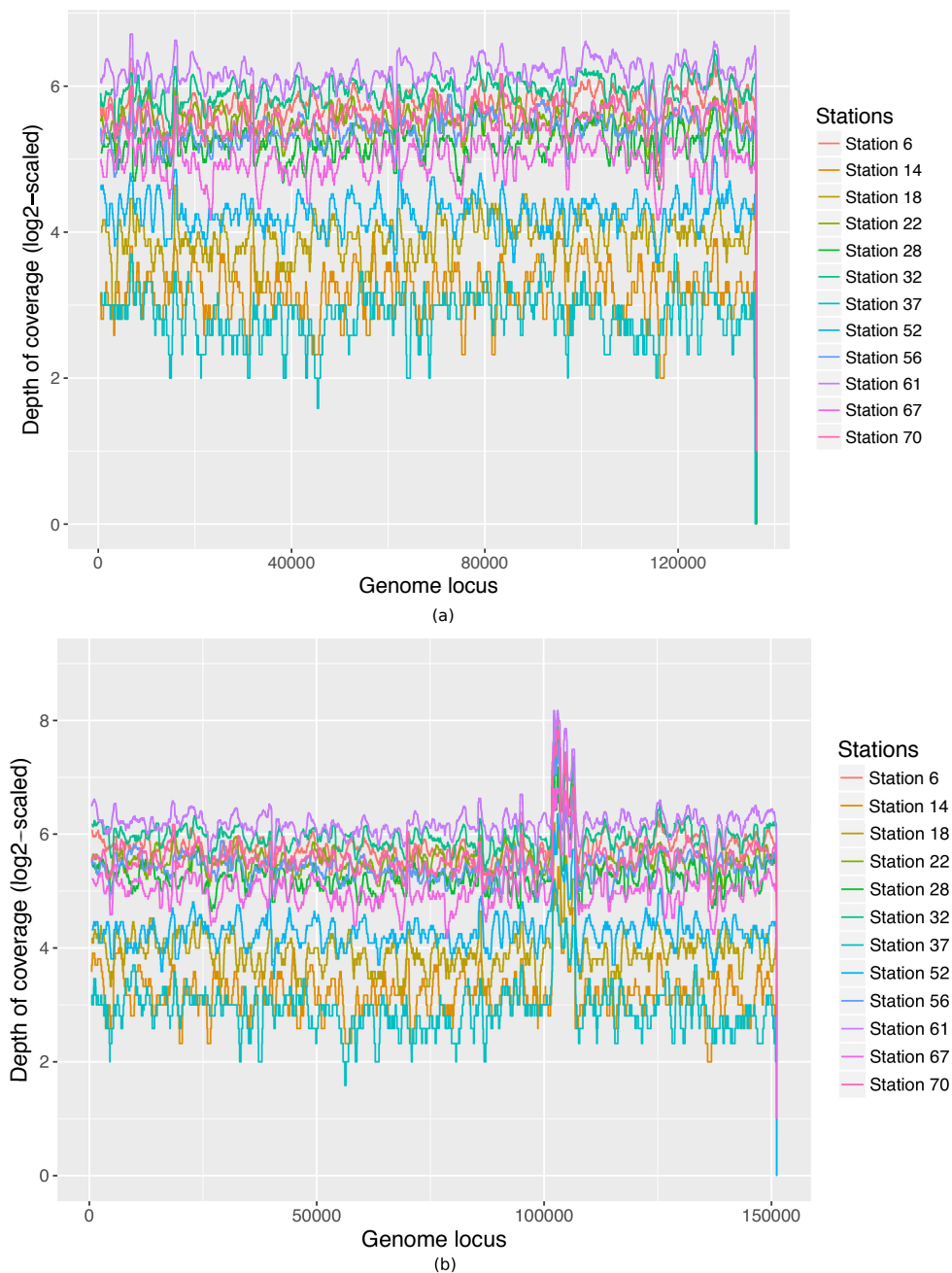


Figure 3.8: The \log_2 -scaled depth of coverage of (a) *S2* (136 kb) and (b) 151 kb scaffold representing the extended and complete version of *S2* across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).

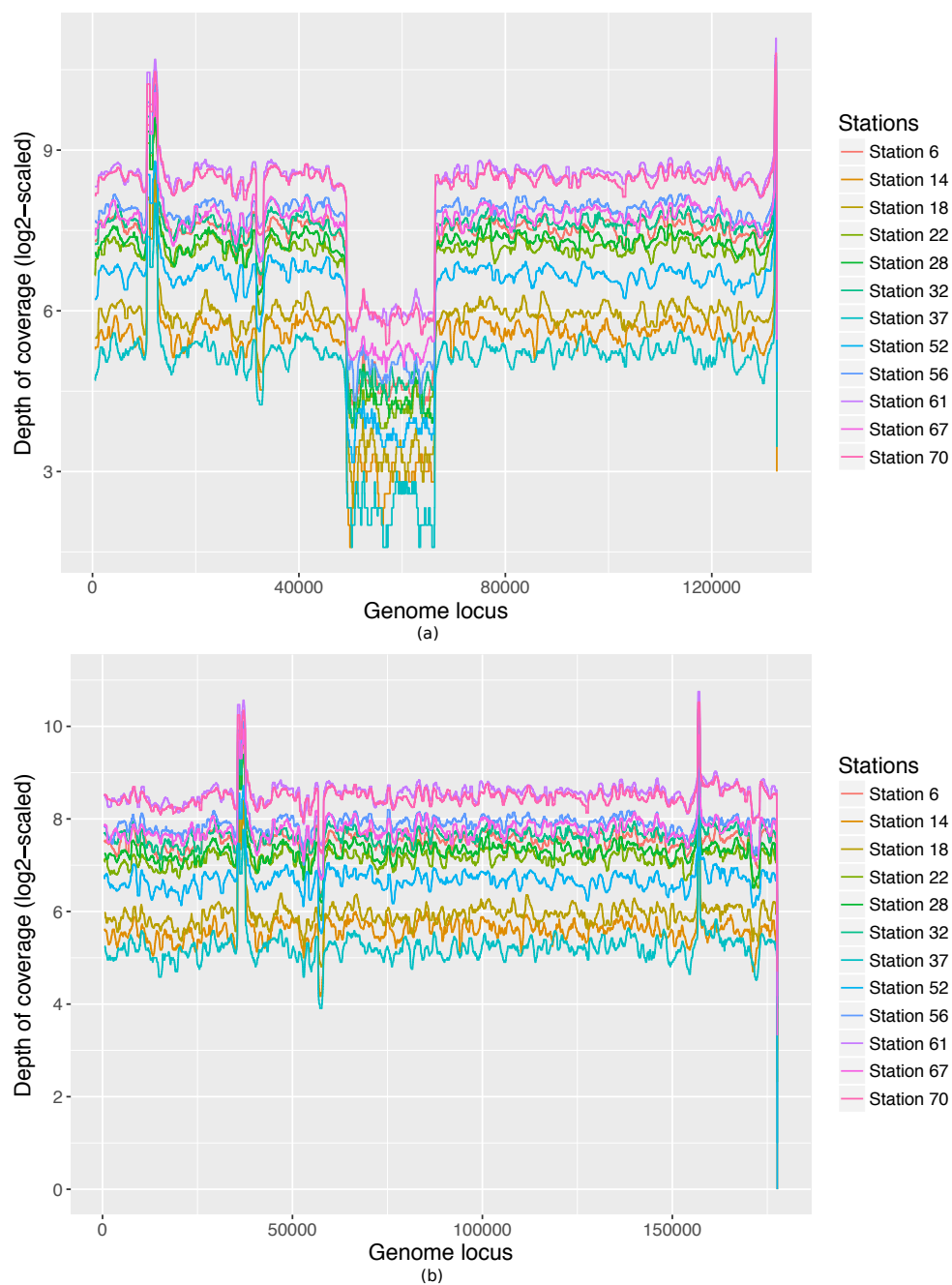


Figure 3.9: The \log_2 -scaled depth of coverage of (a) *S3* (133 kb) and (b) 177 kb scaffold representing the dominant strain of *Prochlorococcus* phage P-SSM4 (recovered from pieces of *S3*) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).

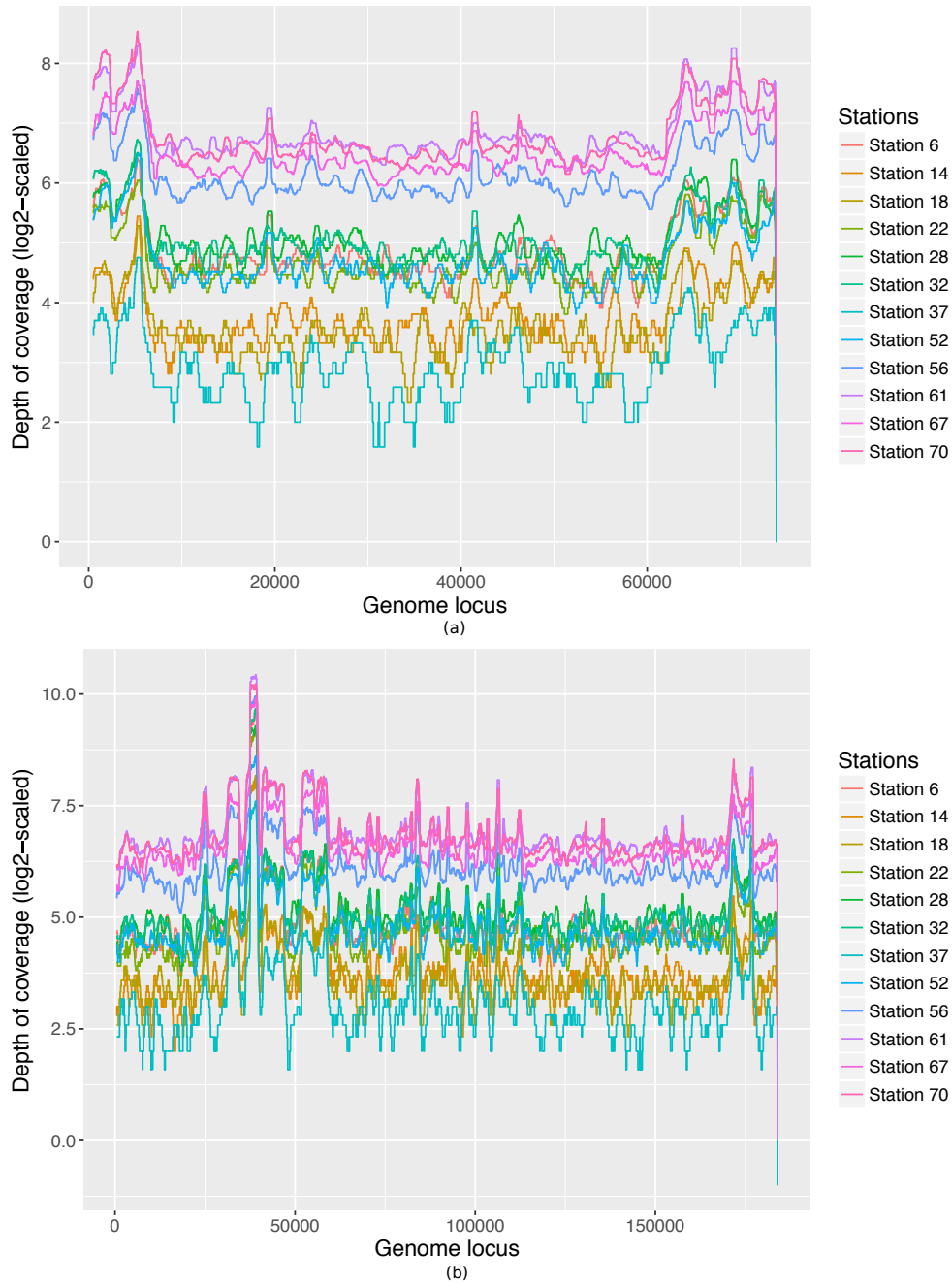


Figure 3.10: The \log_2 -scaled depth of coverage of (a) *S8* (73 kb) and (b) 183 kb scaffold representing the dominant strain of *Prochlorococcus* phage P-HM2 (recovered from pieces of *S8*) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).

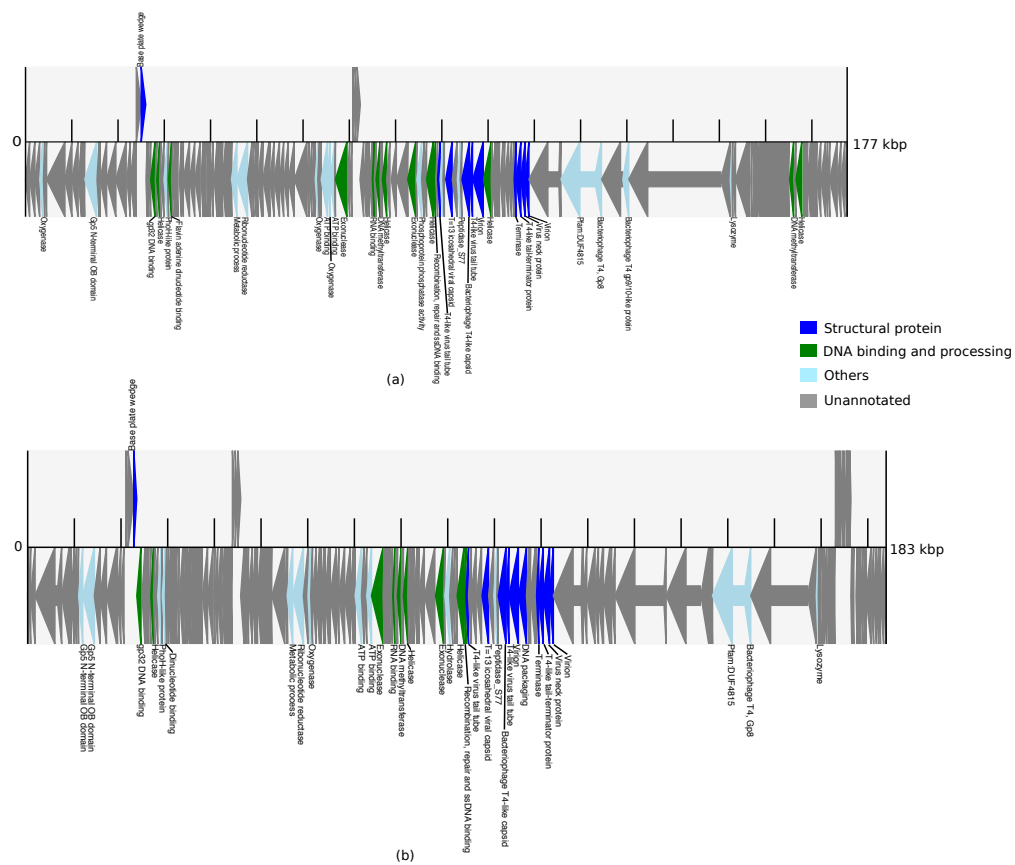


Figure 3.12: Protein annotation of the two strains of Prochlorococcus phage (a) 177 kb scaffold of dominant strain of Prochlorococcus phage P-SSM4 and (b) 183 kb scaffold of dominant strain of Prochlorococcus phage P-HM2.

3.4 Discussion

Here we discovered four complete genomes of viruses that were recovered from the ten longest scaffolds generated by FVE-novel. In total our tool generated 268 scaffolds and out of 18,471,506 paired-end reads from station 70 sample, 1,240,164 reads got mapped to these 268 scaffolds. 8.63% of these mapped reads got mapped to the 153 kb novel virus recovered from *S0*. For the other three complete viruses, 1) 151 kb novel virus recovered from *S2*, 2) 177 kb dominant strain of *Prochlorococcus phage P-SSM4*, and 3) 183 kb strain of *Prochlorococcus phage P-HM2*, 4.2%, 35.32%, and 13.93% reads got mapped to these three viruses respectively. This result implies that these four viruses were very abundant in station 70 sample.

Our assembly of abundant marine viruses revealed a high incidence of microdiversity in naturally-occurring bacteriophage populations in the environment. Genomic chimeras often have anomalous coverage since they represent distinct genotypes that have been incorrectly merged, and most metagenome assemblers use this coverage information to break contigs or scaffolds into smaller pieces to avoid mis-assembly. Given that the recovery of complete bacteriophage genomes from metagenome assemblies is rare, it is plausible that microdiversity may be a prevalent cause of assembly fragmentation. Moreover, given that the genomic microdiversity is difficult to identify and may not always lead to contig breakage by assembler, it is possible that many viral contigs or scaffolds present in publicly-available repositories are in fact chimeras of multiple viral strains. The prevalence of this is difficult to ascertain given the paucity of high-quality bacteriophage genomic references to use for benchmarking. Future studies should therefore prioritize the rigorous identification of strain-level microdiversity in bacteriophage populations to identify the nature and extent of this genome-wide microdiversity in nature.

FVE-novel is freely available online and can be downloaded from <https://github.com/saima-tithi/FVE-novel>. FVE-novel can be used to generate draft virus genomes as well as draft phage genomes. The three steps of FVE-novel are implemented as separate modules so that users can use each module separately if needed. For example, for any metagenomic sequencing data, users can run the entire pipeline, i.e., all three steps, to obtain viral scaffolds. Alternatively, if a user has a read sample and some viral sequences generated from those reads (e.g., using a different assembly pipeline), the user can directly run the second step or seed extension step to extend the input viral sequences.

For each scaffold generated, FVE-novel reports all of its reference genomes along with the ANI and percentage of aligned nucleotide between the scaffold and each reference. FVE-novel also reports per base depth of coverage along the scaffolds, which can be used similarly to what we demonstrated above to determine the quality of the scaffold and identify “problematic regions” that might be the result of misassembly. As our detailed analyses show, having multiple strains of the same viruses can create great challenges to assembly algorithms and it is thus paramount that users examine the report on depth of coverage for the generated

scaffolds to ensure the quality of the putative novel viral scaffolds.

3.5 Conclusion

In this chapter, we presented FVE-novel, a new computational pipeline, for recovering novel viral scaffolds based on reference-based mapping and iterative assembly. By applying our tool to an ocean metagenome sample, we assembled 268 viral scaffolds. Some of these viral scaffolds are quite long, which can be potential near-complete viral genomes. Manual curation and validation of the ten longest scaffolds lead to successful recovery of four complete viral genomes. Among these four viral genomes, two of them are novel genomes as they were not found in the existing databases, one of them represents strain of *Prochlorococcus phage P-SSM4* and another one represents strain of *Prochlorococcus phage P-HM2*. We also noted substantial microdiversity in the phage genomes present in the metagenomic data, which should be considered further in future work given it is a potential complication to the recovery of full-length viral genomes. Overall, FVE-novel implements a novel strategy to recover viral genomes and will serve as a powerful tool for future studies that will continue to enhance existing viral databases.

Chapter 4

VirChecker: An Integrated Pipeline for Error-Correction, Extension, and Annotation of Viral Scaffolds

4.1 Introduction

Analyzing metagenomic data is a great way to analyze hundreds of microbial organisms at the same time without cultivating them in the lab environment. But this presence of many organisms make it computationally difficult to recover the complete genomes of those organisms. Many assembly tools are developed to assemble microbial genomes from metagenomic samples, i.e., MetaVelvet [41], metaSPAdes [43], Ray Meta [7], IDBA-UD [49], and MEGAHIT [34]. But because of the complex nature of metagenomic data, for example the presence of hundreds of organisms, highly uneven coverage of organisms present, presence of multiple strains of the same species, assemblers often face difficulty in recovering complete genomes [20]. As a result, assemblers often produce only part of the genomes from metagenomic datasets [64, 20, 70]. Due to the presence of closely related species and presence of multiple strains of the same species, assemblers sometimes produce chimeric sequences (sequences where genomes from multiple organisms are incorrectly assembled together [70]). Tools are developed to correct assembly errors like chimeric sequences and to improve the quality of draft assemblies by correcting single nucleotide polymorphisms, insertions and deletions [71, 72]. There are also tools to extend draft genomes and to fill up the gaps of draft genomes [6, 5, 18]. But no single tool is available which can perform all of these steps together. Here we developed a computational pipeline, VirChecker, which can do error-correction, extension, and annotation of draft assemblies of viral genomes all together in the same tool.

VirChecker takes a draft assembly of viral genome and the corresponding read sample as

input. Then error-correction and extension steps are applied iteratively to grow the assembly as much as possible while making sure that the extended assembly is error free. As non-uniform coverage is an indication that the assembly can be a chimeric one, in the error-correction step, VirChecker checks the uniformity of the coverage of the assembly and keeps only the uniform coverage part of the assembly for the next extension step. In the extension step, VirChecker leverages FVE-novel, a tool previously developed by us for identifying and growing viral genomes from metagenomic data. After the iterative error-correction and extension steps are done, the final extended viral genome is annotated. The output of our tool consists of extended viral genome/scaffold along with the annotation of the scaffold. Our tool is freely available at <https://github.com/saima-tithi/VirChecker>.

4.2 Methods

The inputs of VirChecker are one viral scaffold/contig and the read sample from which the input scaffold was assembled. The workflow of VirChecker can be divided into three main steps, the error-correction step, the extension step, and the annotation step. The error-correction step contains checking of the circularity of the scaffold and checking of the uniformity of the coverage of the scaffold. The error-correction and extension steps are done iteratively until the scaffold can not be extended anymore. Then the final extended scaffold is annotated. The output of our tool contains the extended scaffold along with the protein annotation of the extended scaffold. Figure 4.1 shows a comprehensive outline of all three steps of VirChecker.

During the error-correction step of VirChecker, at first the circularity of the scaffold is checked. If the assembly of a scaffold is circular or the assembly repeats itself from the beginning, it indicates that the assembly of the genome is complete. If VirChecker finds that a scaffold is circular, it goes to the third step or annotation step, trims the redundant part of the scaffold, and outputs the scaffold as final scaffold. On the other hand, if VirChecker finds that the scaffold is not circular, it goes to the next steps which are the checking of the coverage of the scaffold and extension steps. In order to check the circularity of a scaffold, VirChecker divides the sequence into two parts. Let assume, L_r is the read length and L_s is the length of the scaffold. VirChecker divides the sequence into two parts, G_a and G_b , where G_a starts from $(L_s - 2 * L_r)$ bp to $(L_s - L_r)$ bp and G_b starts from the beginning or from 1 bp to the beginning of G_a or to $(L_s - 2 * L_r)$ bp. G_a is aligned against G_b using BLAST tool (Figure 4.2a). If any part of G_a aligns with G_b with 95% identity and 95% alignment length, then the scaffold is marked as circular sequence (Figure 4.2b) and VirChecker goes to the third step and trims the circular part of the scaffold. In order to identify the circular part, the originally identified similar region which is a region with 95% identity and 95% alignment length, got extended on both sides to get the similar region with maximum length (Figure 4.2c). Then one of the similar region is trimmed because having the same region twice will be redundant.

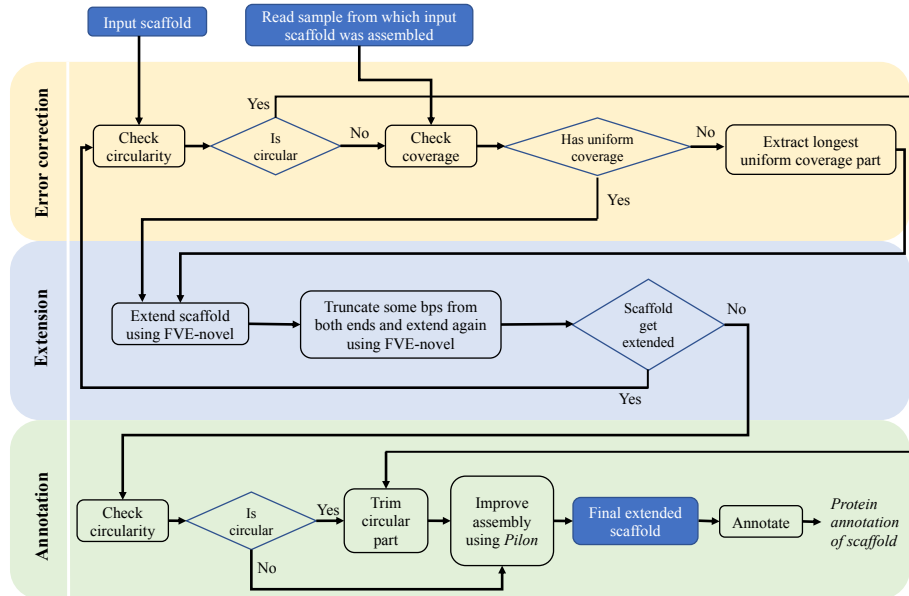


Figure 4.1: Overview of the VirChecker pipeline, where input is a draft assembly of virus and corresponding reads and output is an extended and improved assembly.

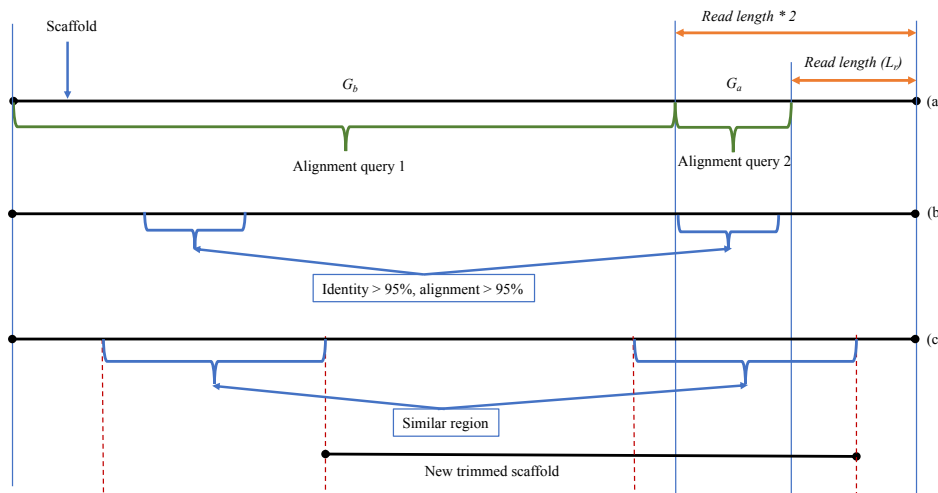


Figure 4.2: Checking the circularity of the scaffold by (a) dividing the scaffold into two parts and aligning them against each other, (b) find a highly similar region, (c) extend the similar region as much as possible and trim one of the similar regions

After checking the circularity of the scaffold, comes checking the uniformity of the coverage of the scaffold. As non-uniform depth of coverage along the scaffold indicates that it may be a chimeric one, in order to get rid of the chimeric part of the scaffold, our tool checks the uniformity of the coverage of the scaffold and keeps an uniform coverage part of the scaffold for next steps. At first, per base depth of coverage of the scaffold is calculated using Samtools [35]. If for any base pair, the coverage is within the 15 to 85th percentile, it is considered within the normal range. On the other hand, for any base pair, if the coverage is outside of this normal range or outside of 15 to 85th percentile, then it is marked as suspicious base. All bases marked as suspicious form suspicious regions, R_s . For any suspicious region, R_s , if the length is > 1000 bp, then it is considered as true suspicious region, R_{ts} . All the regions other than R_{ts} are flagged as true non-suspicious region, R_{tns} . Then VirChecker keeps the longest R_{tns} and extends this part of the scaffold during the extension step.

During the extension step, VirChecker leverages the FVE-novel tool, a tool based on a novel iterative assembly approach to effectively extend a virus genome/scaffold as much as possible. After the scaffold can not be extended anymore, VirChecker trims some bps from both ends of the scaffold and try to extend the trimmed scaffold again using FVE-novel tool. The logic behind trimming some bases from the ends is from our empirical study we found that assemblers often do misassembly in one or both ends of the sequence, and then because of the misassembly in one end that end can not be extended anymore. Trimming some bases from both ends often helps the assembler to continue the assembly in the right direction. After trimming and extending the scaffold using FVE-novel, if the scaffold can be extended then the new extended scaffold goes back to the error-correction step. But if it can not be extended even after trimming then the extension step is ended and the scaffold goes to the third step or annotation step.

During the last step or annotation step of VirChecker, the scaffold is checked for circularity and if it is circular then the scaffold is trimmed to get rid of the redundant part. Then Pilon [71] is applied to the scaffold to improve the assembly by correcting single nucleotide polymorphisms or SNPs, insertions, and deletions. The inputs of Pilon tool are a genome/scaffold in FASTA format and reads mapped to the genome in BAM format. From the alignment information, Pilon creates a pileup structure and then based on the frequency of each nucleotide in a base corrects the base. During the base correction step, Pilon also considers if the reads are properly paired or not and also the mapping quality of the base. Based on the pileup information, Pilon sometimes inserts or deletes some bases. If the alignment of read pairs indicates a discrepancy in the assembly, then Pilon tries to fix the assembly by doing a local reassembly in those places. After applying Pilon, the improved scaffold is the final output of our tool. Then in order to annotate the final scaffold, the proteins of the scaffold are predicted using Prodigal [26] with metagenome option (Prodigal option: -p meta). After that all the predicted proteins are annotated using eggno-mapper [25] using virus database and HMMER option. The final output of our tool contains the extended and improved assembly and the annotation of the assembly.

In previous chapter, by applying FVE-novel to an ocean metagenome sample collected from

the study of Aylward et al. [3], we obtained 268 viral scaffolds. Manual examination of the ten longest scaffolds from those 268 scaffolds reveals that those ten scaffolds are potential virus scaffolds. But error-correction and extension are needed to generate complete virus genome from those scaffolds. Here we applied VirChecker to those scaffolds to obtain an improved and extended assembly of those scaffolds.

4.3 Results and Discussion

We applied VirChecker pipeline to the six scaffolds obtained from the results of FVE-novel described in the previous chapter. These six scaffolds are *S0*, *S1*, *S2*, *S3*, *S6*, and *S8*. From the previous study, we found that these six scaffolds can be divided into four groups, where group 1 contains *S0*, *S1*, and *S6*. These three scaffolds have high similarity with each other and they are potential novel viral scaffolds as they were not found in the existing viral databases. Group 2 contains scaffold *S2* which is also a potential novel virus. Group 3 contains *S3* and group 4 contains *S8*. These two scaffolds were strains of Prochlorococcus phage. Here we applied VirChecker to these six scaffolds to obtain an improved assembly of these six scaffolds. In the following, we discussed the results of applying VirChecker to these scaffolds in detail.

4.3.1 Group 1 Scaffolds (*S0*, *S1*, and *S6*)

The longest scaffold obtained from FVE-novel result was scaffold *S0* with length 193,112 bp. By examining the per base depth of coverage of *S0*, we found that around 24,500 bp to 25,500 bp coverage is lower than the average coverage. And also after 150 kb coverage varied a lot. VirChecker checked the uniformity of the depth of coverage of this scaffolds and flagged some regions as suspicious regions (Figure 4.3). VirChecker flagged region 24,562 bp to 25,566 bp as suspicious region and many regions after 152,990 bp as suspicious regions. Then it extracted the longest non-suspicious region or region with uniform coverage, which was a region with length 127,423 bp from 25,567 bp to 152,989 bp. This longest non-suspicious region went through the iterative extension and error-correction steps. After the iterative extension step was done, the output scaffold, which was a scaffold with length 158,743 bp, was checked for circularity. As this scaffold was not circular, then pilon was applied to the output scaffold in order to improve assembly. Pilon corrected 548 snps, 257 ambiguous bases, 2 insertions, and 1 deletions. The improved scaffold (length 158,259 bp) from Pilon was the final output of VirChecker. From figure 4.4 we can see that the improved scaffold *S0* has a uniform depth of coverage along the length.

Similarly, after checking the depth of coverage of the scaffold *S1* from previous study, we found that there was a non-uniform coverage region or suspicious region from 133,376 bp to 134,543 bp (Figure 4.5). VirChecker then extracted the longest non-suspicious region

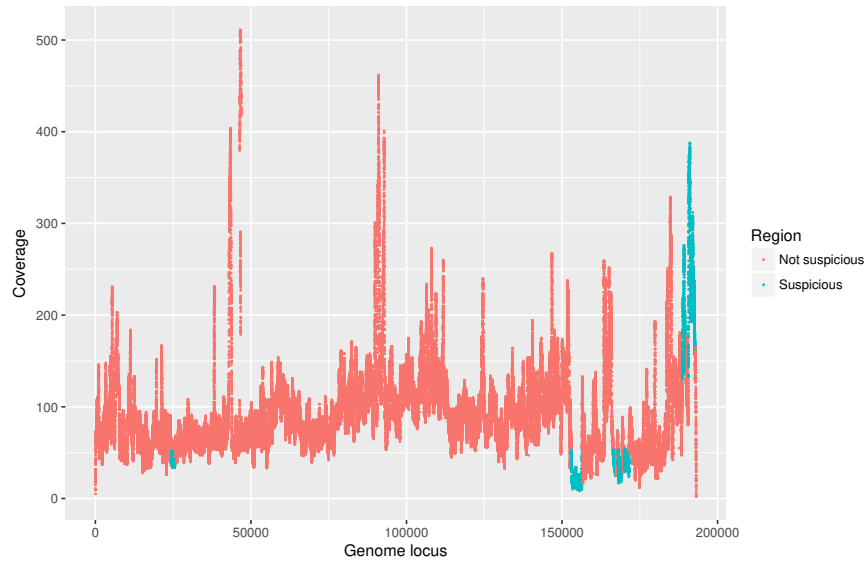


Figure 4.3: The depth of coverage of scaffold, *S0* from previous study (length 193 kb) where the non-uniform coverage regions or suspicious regions are marked.

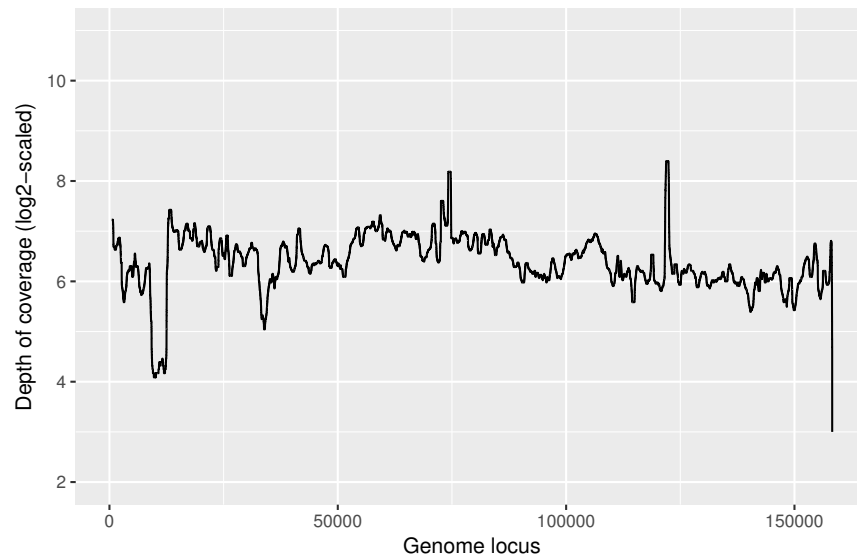


Figure 4.4: The \log_2 -scaled depth of coverage of the improved version of scaffold *S0* (length 158,259 bp) after applying VirChecker.

which was a region of length 133,375 bp from 1 bp to 133,375 bp and then the iterative extension and error-correction steps were applied to generate an extended scaffold. After the iterative extension step was done, VirChecker generated a scaffold with length 152,630 bp. VirChecker then checked the circularity of this scaffold and found that it is a circular scaffold, so it had some redundant regions due to assembling the same region twice. Then the redundant part of the scaffold was trimmed resulting in an scaffold with length 152,291 bp. Then Pilon was applied and the final output was a scaffold with length 153,133 bp. Figure 4.6 shows the depth of coverage of this improved scaffold *S1*. The protein annotation of scaffold *S1* is shown in Figure 4.7.

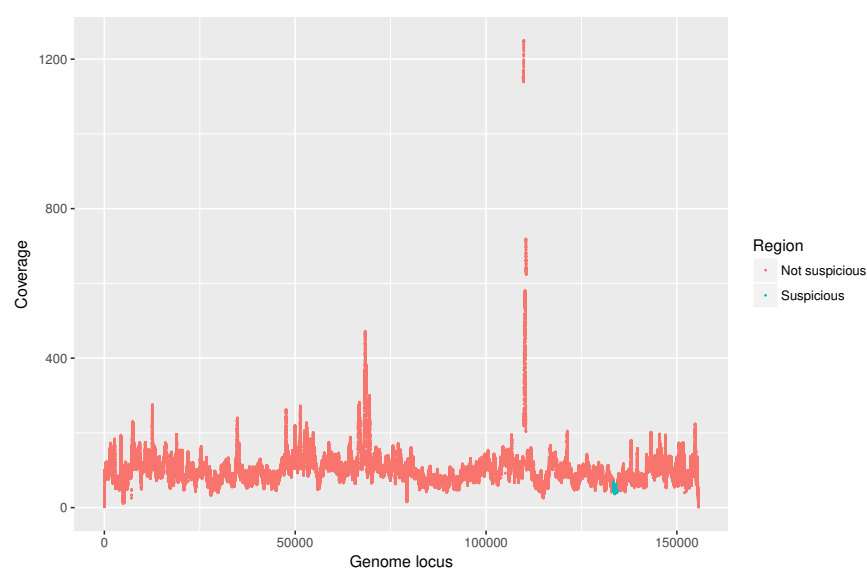


Figure 4.5: The depth of coverage of scaffold, *S1* from previous study (length 153 kb) where the non-uniform coverage regions or suspicious regions are marked.

For the last scaffold of group 1, scaffold *S6* from the previous study, the per depth of coverage was checked using VirChecker and according to VirChecker it had no non-uniform coverage region or suspicious region. This scaffold then went through the extension and error-correction steps iteratively and generated a scaffold with length 152,906 bp. Then after checking the circularity using VirChecker, we found that it is not circular. After that, pilon was applied and an improved assembly with length 152,967 bp was generated. The per base depth of coverage of the improved *S6* scaffold shows that it has a uniform depth of coverage along the length (Figure 4.8).

In the previous study, from manual examination and curation of scaffold *S0*, we obtained a 153 kb dominant strain of a novel virus. Here after improving scaffolds *S0*, *S1*, and *S6*, the improved versions of *S0*, *S1*, and *S6* were compared with the manually curated 153 kb strain (Figure 4.9). From figure 4.9, we can see that *S0* is a bit different than the 153 kb dominant strain recovered in the previous study, but *S1* and *S6* are almost identical to the manually curated scaffolds. We further compared *S1* (length 153,210 bp) with the manually curated

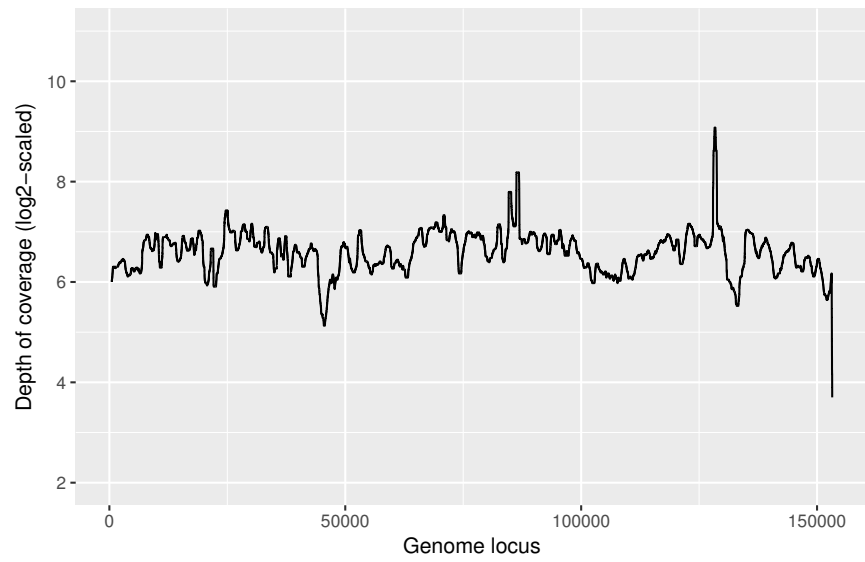


Figure 4.6: The \log_2 -scaled depth of coverage of the improved version of scaffold *S1* (length 153,133 bp) after applying VirChecker.

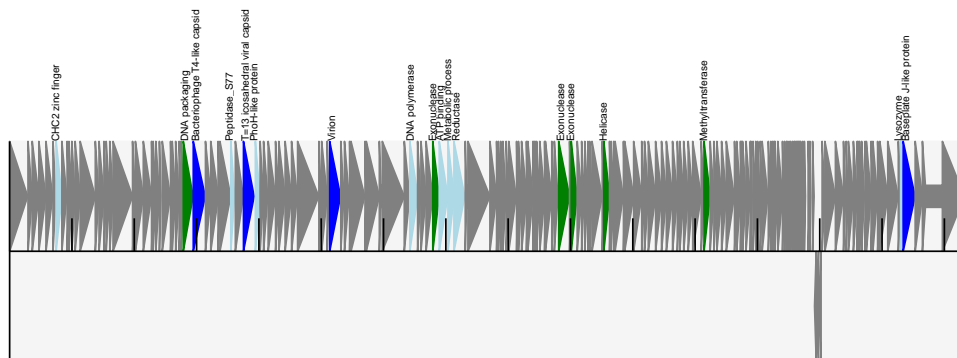


Figure 4.7: The protein annotation of scaffold *S1* from VirChecker.

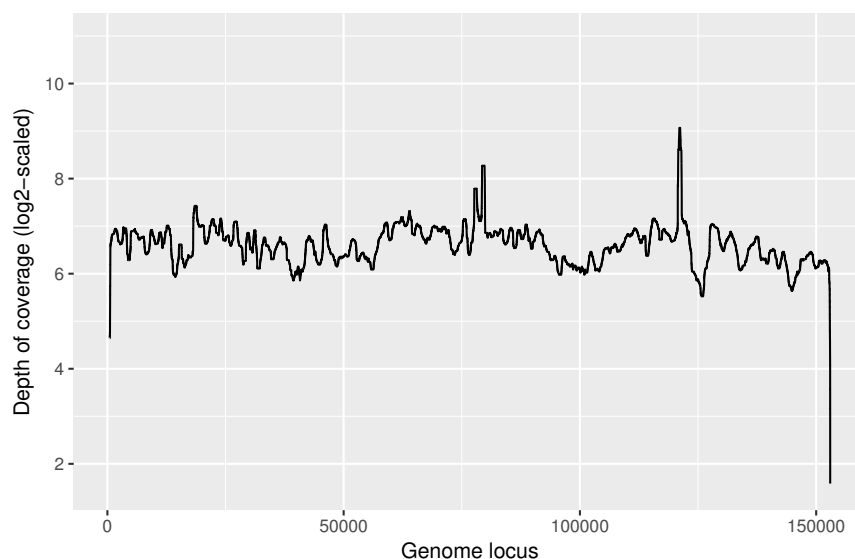


Figure 4.8: The \log_2 -scaled depth of coverage of the improved version of scaffold *S6* (length 152,967 bp) after applying VirChecker.

153 kb dominant strain (length 153,113 bp). From the comparison we found that in *S1*, a 109 bp region, from 14,745 bp to 14,853 bp does not match with the manually curated 153 kb dominant strain and for this region depth of coverage is around 30x, where the average coverage of the scaffold is 100x (Figure C.3).

4.3.2 Group 2 Scaffold, *S2*

The scaffold, *S2* with length 136,254 bp was collected from the previous study and VirChecker tool was applied to this 136 kb scaffold. After checking the coverage of *S2*, we found that there was no non-uniform coverage region or suspicious region. This 136 kb scaffold was then extended and corrected iteratively using VirChecker and generated a scaffold of length 151,189 bp. After checking circularity of this scaffold it was found as not circular. Then Pilon was applied to improve the scaffold and it generated a scaffold of length 149,414 bp. Figure 4.10 shows that per base depth of coverage of improved and extended scaffold *S2* are uniform along the length. The protein annotation of *S2* is shown in Figure 4.11.

The comparison of improved scaffold *S2* with the 151 kb dominant strain collected from the previous study shows that these two scaffolds are very similar except for a high coverage region of 2 kb, which is present in the 151 kb dominant strain, but missing in the improved *S2* from VirChecker (Figure 4.12). Figure C.1 shows the high coverage region present in the 151 kb dominant strain. Further examination of the different steps of VirChecker reveals that this 2kb high coverage region was present in the result of VirChecker before applying Pilon. Before applying Pilon, VirChecker generated a scaffold with length 151,189 bp. Figure C.2

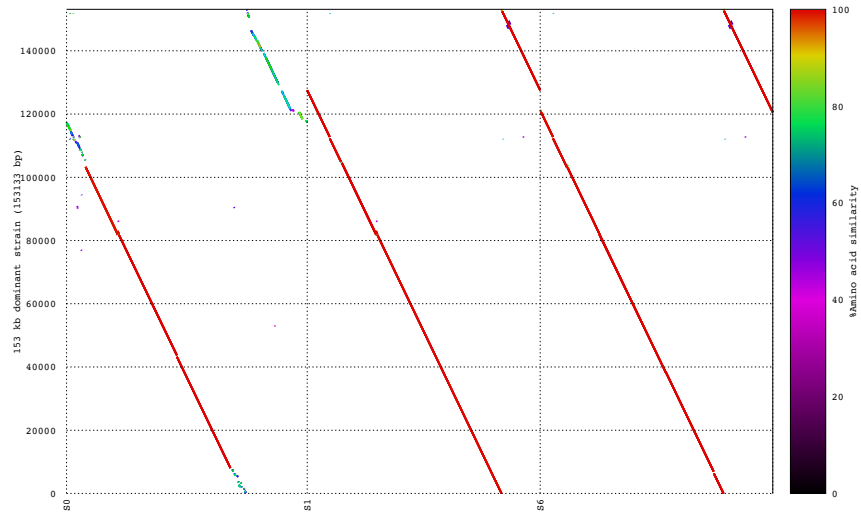


Figure 4.9: Percentage of similarity of 153 kb dominant strain with the improved scaffolds S_0 , S_1 , and S_6 obtained from VirChecker tool.

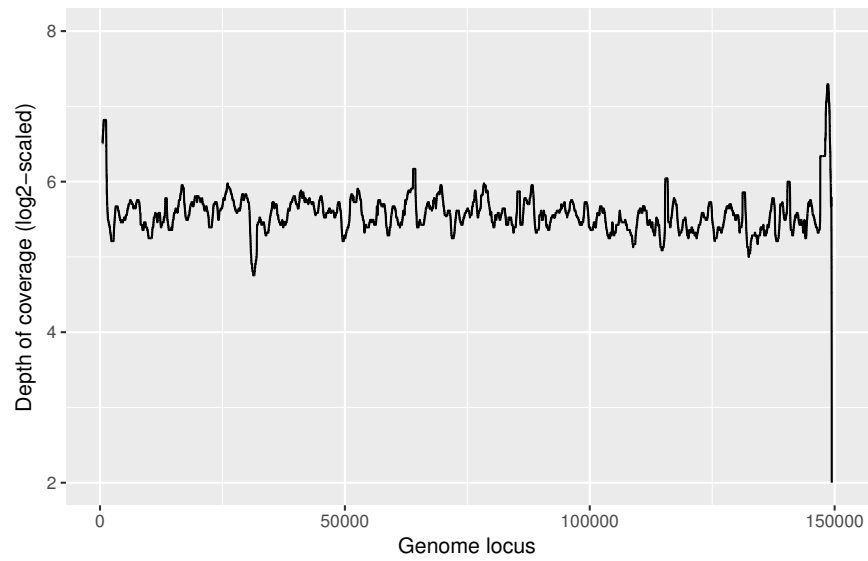


Figure 4.10: The \log_2 -scaled depth of coverage of the improved version of scaffold S_2 (length 149,414 bp) after applying VirChecker.

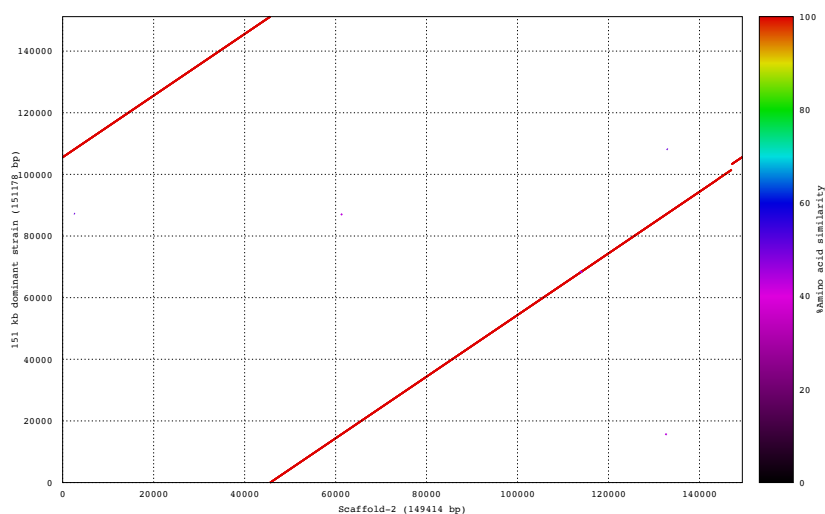


Figure 4.12: Percentage of similarity of 151 kb dominant strain with the improved scaffold S2 obtained from VirChecker.

Prochlorococcus phage from the previous study and we can see that the improved *S3* is very similar to the 177 kb dominant strain. We further compared *S3* (length 178,118 bp) with the 177 kb dominant strain (length 177,631 bp) to find out what are the differences between these two scaffolds. Comparison of these two scaffolds reveal that in manually curated 177 kb dominant strain, 1,282 bp, from 56,831 bp to 58,112 bp does not have any match with *S3*. In this part instead of this 1,282 bp, *S3* contains 1,342 bp from 20,853 to 22,194 bp. Analysis of the depth of coverage of these two scaffolds in the area where they are different reveals that those areas have the lower depth of coverage (about 150x) then the average depth of coverage (300x) (Figure). This lower depth of coverage areas in these two scaffolds indicate that they may be potential different strains of same phage.

4.3.4 Group 4 Scaffold, S8

From the previous study, we obtained scaffold *S8* with length 73,892 bp and checked the uniformity of the depth of coverage by applying VirChecker. From figure 4.17 we can see that *S8* had non-uniform coverage regions or suspicious regions on both ends of the scaffold. VirChecker then extracted the longest uniform coverage region with length 58,560 bp from 5,168 bp to 63,727 bp. Then this 58 kb scaffold got extended using iterative extension and error-correction steps and an extended scaffold with length 179,893 bp was generated. By checking circularity, we found that it is not circular. After that Pilon was applied to this scaffold and an improved scaffold with length 179,890 bp was generated, which was the final

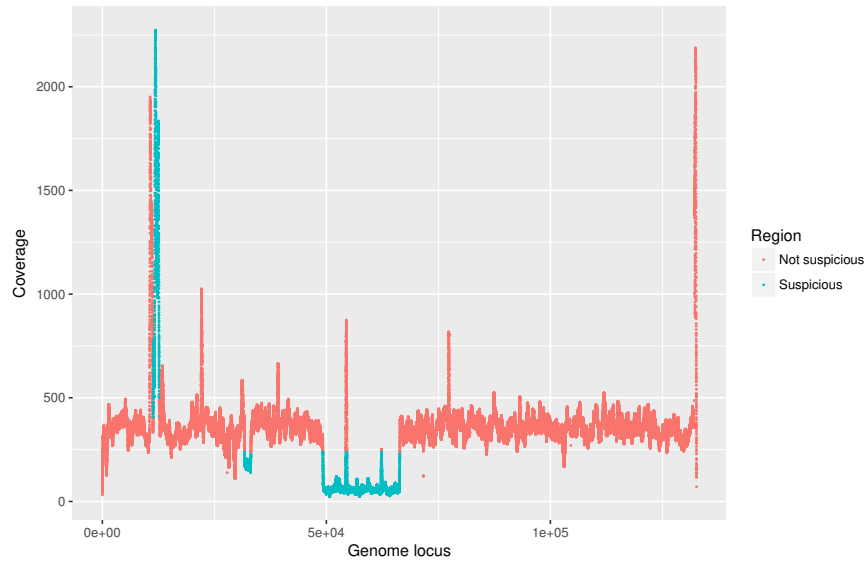


Figure 4.13: The depth of coverage of scaffold, *S3* from previous study (length 132 kb) where the non-uniform coverage regions or suspicious regions are marked.

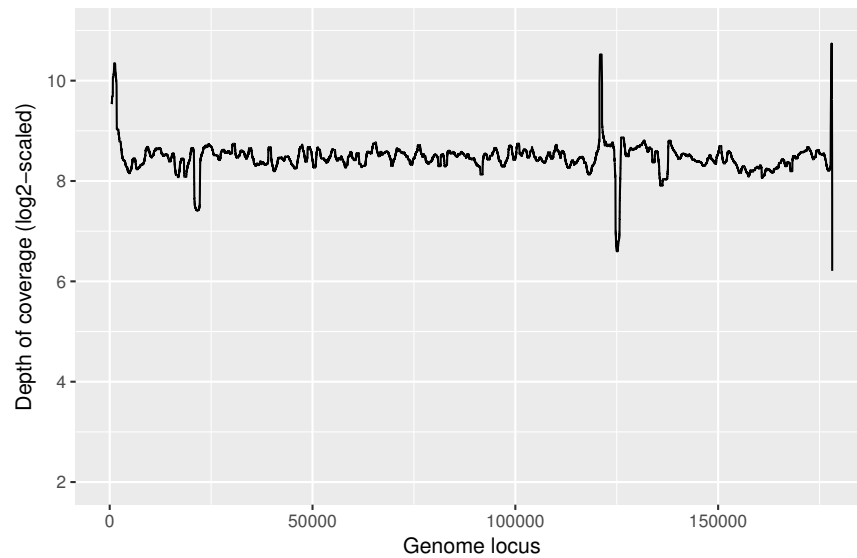


Figure 4.14: The \log_2 -scaled depth of coverage of the improved version of scaffold *S3* (length 178,118 bp) after applying VirChecker.

output of VirChecker. Figure 4.18 shows the depth of coverage of improved and extended *S8*. From this figure we can see that *S8* has some non-uniform coverage regions around 110 kb to 140 kb had coverage higher than the average coverage. By comparing 179 kb scaffold from VirChecker with the manually curated 183 kb dominant strain from previous study, we found that they have some differences from 110 kb to 140 kb (Figure 4.19). Figure 4.20 shows the protein annotation of scaffold *S8*.

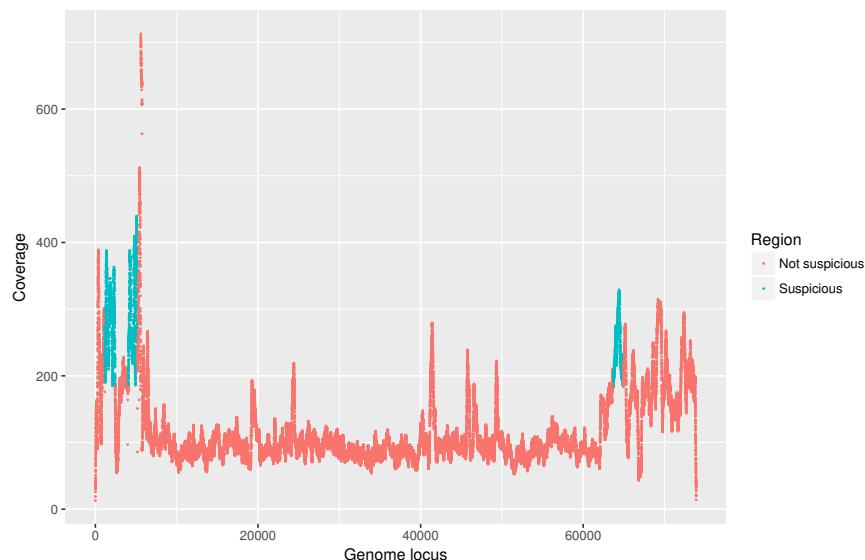


Figure 4.17: The depth of coverage of scaffold, *S8* from previous study (length 73 kb) where the non-uniform coverage regions or suspicious regions are marked.

4.3.5 Limitations of VirChecker and Usage Recommendations

The algorithm of VirChecker has some limitations also. One limitation of VirChecker is during the coverage checking step, VirChecker does not check the GC content of the suspicious regions. But in Illumina sequencing very high or very low GC content ($>70\%$ or $<30\%$) can result in reduced mapping coverage and higher error rates. As a result, a low coverage region with high or low GC content can be actually part of the scaffold, but our tool can wrongly mark it as suspicious region and discard that region. Another limitation of VirChecker tool is it can incorrectly mark a linear phage as circular. Some linear phages may have repeat sequences at the ends and because of this repeat sequences, assemblers can start the assembly of the phage again from the beginning and during the circularity checking step of VirChecker, it will mark this phage as a circular genome, which may not be true. So manual examination of the result of each step of VirChecker should be done by the user to ensure the accuracy of the result.

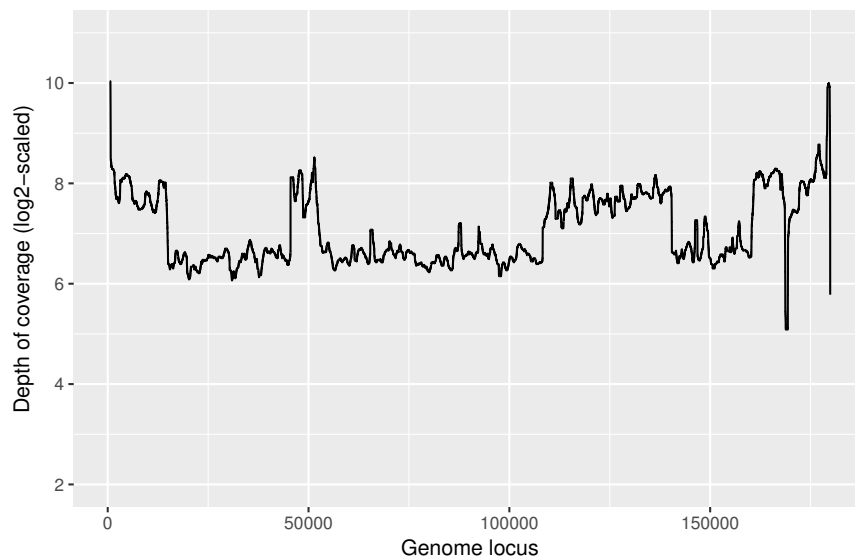


Figure 4.18: The \log_2 -scaled depth of coverage of the improved version of scaffold *S8* (length 179,890 bp) after applying VirChecker.

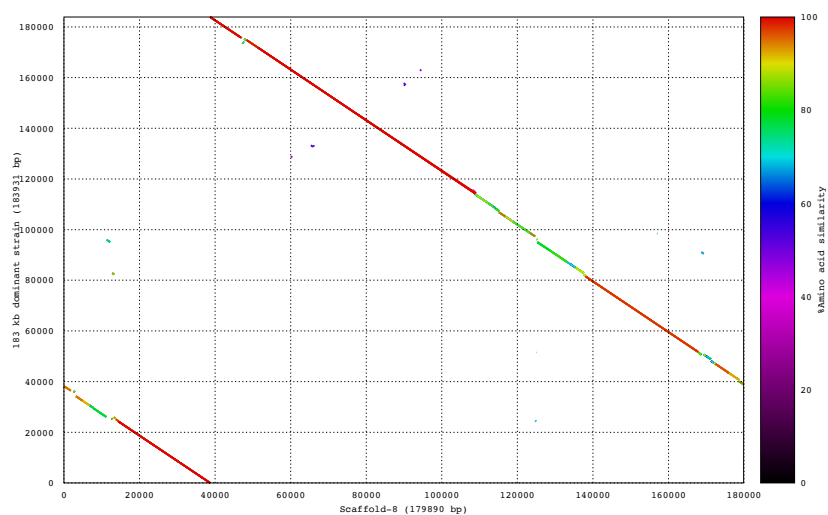


Figure 4.19: Percentage of similarity of 183 kb dominant strain of Prochlorococcus phage with the improved scaffold *S8* obtained from VirChecker.

Chapter 5

Conclusion and Future Prospects

Viruses are the most abundant micro-organism on earth. Viruses have profound impact on human health and regulating many environmental processes. Analyzing metagenomic data is a great way to analyze many viruses at the same time without cultivating them in the lab environment. To analyze metagenomic data for viruses, the first question we need to answer is what viruses are there and at what quantity? To answer this question, in chapter 2, we developed a computational pipeline called FastViromeExplorer. Our tool uses a reference database and a novel pseudoalignment based approach to quickly identify and quantify viruses present in the data set. Application of FastViromeExplorer to human gut samples and ocean microbiome sample shows that our tool can identify and quantify viruses much more quickly than the traditional approaches especially when the reference database is large. One drawback of our tool is when the reference database is large, our tool requires large computational power. A future improvement of this tool can be improve the algorithm of the tool to make it work with small computational power. Another extension of this work can be applying this tool to a large cohort of samples. We applied this tool to seven data sets including four samples collected from healthy donors and three samples collected from patients. The results showed a significant change in the abundance of the same phages from healthy samples to patient samples. Applying our tool to a large cohort including many healthy samples and many patient samples and analyzing the results can reveal important insights on the changes of microbiota due to a disease.

After identifying what viruses are present in the data set, the next analysis of the metagenomic data can be assemble the viral reads present in the data and recover the full genomes of the viruses present in the data. In chapter 3, we developed a computational pipeline, FVE-novel, which is a hybrid of reference based and de novo assembly approach to assemble viral reads and recover complete viral genomes. By applying FVE-novel to a ocean metagenome sample, we successfully recovered 268 viral scaffolds. Examination of the longest ten scaffolds out of these 268 scaffolds shows that FVE-novel successfully recovered two novel virus genomes and two strains of known phages. Examination of these ten scaffolds also reveals

that due to the complex nature of metagenomic data set, result of FVE-novel can contain chimeric sequences and not complete viral sequences. In order to mitigate the errors of the assembly due to chimeric sequences and to complete the assembly, in chapter 4, we proposed a new computational pipeline, VirChecker. VirChecker corrects the assembly errors like chimeric sequences, extends the viral assembly as much as possible, and annotate the final improved and extended assembly. Application of VirChecker tool to the six scaffolds obtained from chapter 3 shows that VirChecker can successfully mitigate the coverage error and extend the assembly to generate an extended and error-free viral genome. One limitation of FVE-novel and VirChecker is that both tools are computationally expensive. A computer cluster with large numbers of cores is needed to run these tools within a reasonable amount of time. Further study can be done to make these tools more computationally efficient. Another extension of these works can be applying these tools to a large cohort of data, i.e., time series data. As our tool can recover draft genomes of viruses, applying these tools to a time series data can reveal if any virus presents in the samples got changed over time and if so what changes or genomic variants took place over the course of time.

Bibliography

- [1] Soyeon Ahn, Ziqi Ke, and Haris Vikalo. Viral quasispecies reconstruction via tensor factorization with successive read removal. *Bioinformatics*, 34(13):i23–i31, 2018.
- [2] Stephen F. Altschul, Thomas L. Madden, Alejandro A. Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.
- [3] Frank O. Aylward, Dominique Boeuf, Daniel R. Mende, Elisha M. Wood-Charlson, Alice Vislova, John M. Eppley, Anna E. Romano, and Edward F. DeLong. Diel cycling and long-term persistence of viruses in the ocean’s euphotic zone. *Proceedings of the National Academy of Sciences*, page 201714821, 2017.
- [4] Anton Bankevich, Sergey Nurk, Dmitry Antipov, Alexey A. Gurevich, Mikhail Dvorkin, Alexander S. Kulikov, Valery M. Lesin, Sergey I. Nikolenko, Son Pham, Andrey D. Prjibelski, et al. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *Journal of Computational Biology*, 19(5):455–477, 2012.
- [5] Marten Boetzer, Christiaan V. Henkel, Hans J. Jansen, Derek Butler, and Walter Pirovano. Scaffolding pre-assembled contigs using SSPACE. *Bioinformatics*, 27(4):578–579, 2010.
- [6] Marten Boetzer and Walter Pirovano. Toward almost closed genomes with GapFiller. *Genome Biology*, 13(6):R56, 2012.
- [7] Sébastien Boisvert, Frédéric Raymond, Élénie Godzaridis, François Laviolette, and Jacques Corbeil. Ray Meta: scalable de novo metagenome assembly and profiling. *Genome Biology*, 13(12):R122, 2012.
- [8] Nicolas L. Bray, Harold Pimentel, Páll Melsted, and Lior Pachter. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology*, 34(5):525–527, 2016.
- [9] Gubio S. Campos, Antonio C. Bandeira, and Silvia I. Sardi. Zika Virus Outbreak, Bahia, Brazil. *Emerging Infectious Diseases*, 21(10):1885, 2015.

- [10] Nicholas B. Carrigy, Rachel Y. Chang, Sharon SY Leung, Melissa Harrison, Zaritza Petrova, Welkin H. Pope, Graham F. Hatfull, Warwick J. Britton, Hak-Kim Chan, Dominic Sauvageau, et al. Anti-tuberculosis bacteriophage d29 delivery with a vibrating mesh nebulizer, jet nebulizer, and soft mist inhaler. *Pharmaceutical Research*, 34(10):2084–2096, 2017.
- [11] Miles W. Carroll, David A. Matthews, Julian A. Hiscox, Michael J. Elmore, Georgios Pollakis, Andrew Rambaut, Roger Hewson, Isabel García-Dorival, Joseph Akoi Bore, Raymond Koundouno, Sad Abdellati, Babak Afrough, et al. Temporal and spatial analysis of the 2014-2015 Ebola virus outbreak in West Africa. *Nature*, 524(7563):97–101, 2015.
- [12] Tim Carver, Simon R. Harris, Matthew Berriman, Julian Parkhill, and Jacqueline A. McQuillan. Artemis: an integrated platform for visualization and analysis of high-throughput sequence-based experimental data. *Bioinformatics*, 28(4):464–469, 2011.
- [13] Ana Georgina Cobián Güemes, Merry Youle, Vito Adrian Cantú, Ben Felts, James Nulton, and Forest Rohwer. Viruses as Winners in the Game of Life. *Annual Review of Virology*, 3:197–214, 2016.
- [14] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.
- [15] Bas E. Dutilh, Noriko Cassman, Katelyn McNair, Savannah E. Sanchez, Genivaldo G.Z. Silva, Lance Boling, Jeremy J. Barr, Daan R. Speth, Victor Seguritan, Ramy K. Aziz, et al. A highly abundant bacteriophage discovered in the unknown sequences of human faecal metagenomes. *Nature Communications*, 5:ncomms5498, 2014.
- [16] David Eppstein, Maarten Löffler, and Darren Strash. Listing all maximal cliques in large sparse real-world graphs. *Journal of Experimental Algorithmics (JEA)*, 18:3–1, 2013.
- [17] Laura Fancello, Didier Raoult, and Christelle Desnues. Computational tools for viral metagenomics and their application in clinical research. *Virology*, 434(2):162–174, 2012.
- [18] Gregory K. Farrant, Mark Hoebeke, Frédéric Partensky, Gwendoline Andres, Erwan Corre, and Laurence Garczarek. WiseScaffolder: an algorithm for the semi-automatic scaffolding of Next Generation Sequencing data. *BMC Bioinformatics*, 16(1):281, 2015.
- [19] Limin Fu, Beifang Niu, Zhengwei Zhu, Sitao Wu, and Weizhong Li. CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, 28(23):3150–3152, 2012.

- [20] Rodrigo García-López, Jorge Francisco Vázquez-Castellanos, and Andrés Moya. Fragmentation and coverage variation in viral metagenome assemblies, and their effect in diversity calculations. *Frontiers in Bioengineering and Biotechnology*, 3:141, 2015.
- [21] Stephen K. Gire, Augustine Goba, Kristian G. Andersen, Rachel S. G. Sealfon, Daniel J. Park, Lansana Kanneh, Simbirie Jalloh, Mambu Momoh, Mohamed Fullah, Gytis Dudas, Shirlee Wohl, Lina M. Moses, Nathan L. Yozwiak, Sarah Winnicki, Christian B. Matranga, Christine M. Malboeuf, James Qu, Adrienne D. Gladden, Stephen F. Schaffner, Xiao Yang, et al. Genomic surveillance elucidates Ebola virus origin and transmission during the 2014 outbreak. *Science*, 345(6202):1369–1372, 2014.
- [22] Bart L. Haagmans, Said H.S. Al Dhahiry, Chantal BEM Reusken, V. Stalin Raj, Monica Galiano, Richard Myers, Gert-Jan Godeke, Marcel Jonges, Elmoubasher Farag, Ayman Diab, et al. Middle east respiratory syndrome coronavirus in dromedary camels: an outbreak investigation. *The Lancet Infectious Diseases*, 14(2):140–145, 2014.
- [23] Jo Handelsman, Michelle R. Rondon, Sean F. Brady, Jon Clardy, and Robert M. Goodman. Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products. *Chemistry & Biology*, 5(10):R245–R249, 1998.
- [24] Geoffrey D. Hannigan, Melissa B. Duhaime, Mack T. Ruffin, Charlie C. Koumpouras, and Patrick D. Schloss. Viral and bacterial communities of colorectal cancer. *bioRxiv*, page 152868, 2017.
- [25] Jaime Huerta-Cepas, Kristoffer Forslund, Luis Pedro Coelho, Damian Szklarczyk, Lars Juhl Jensen, Christian von Mering, and Peer Bork. Fast genome-wide functional annotation through orthology assignment by eggNOG-mapper. *Molecular Biology and Evolution*, 34(8):2115–2122, 2017.
- [26] Doug Hyatt, Gwo-Liang Chen, Philip F. LoCascio, Miriam L. Land, Frank W. Larimer, and Loren J. Hauser. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics*, 11(1):119, 2010.
- [27] Mark A. Kay, Catherine S. Manno, Margaret V. Ragni, Peter J. Larson, Linda B. Couto, Alan McClelland, Bertil Glader, Amy J. Chew, Shing J. Tai, Roland W. Herzog, et al. Evidence for gene transfer and expression of factor ix in haemophilia b patients treated with an aav vector. *Nature Genetics*, 24(3):257, 2000.
- [28] Matthew Kearse, Richard Moir, Amy Wilson, Steven Stones-Havas, Matthew Cheung, Shane Sturrock, Simon Buxton, Alex Cooper, Sidney Markowitz, Chris Duran, Tobias Thierer, Bruce Ashton, Peter Meintjes, and Alexei Drummond. Geneious Basic: an integrated and extendable desktop software platform for the organization and analysis of sequence data. *Bioinformatics*, 28(12):1647–1649, 2012.

- [29] Stefan Kurtz, Adam Phillippy, Arthur L. Delcher, Michael Smoot, Martin Shumway, Corina Antonescu, and Steven L. Salzberg. Versatile and open software for comparing large genomes. *Genome Biology*, 5(2):R12, 2004.
- [30] Patrick W. Laffy, Elisha M. Wood-Charlson, Dmitriy Turaev, Karen D. Weynberg, Emmanuelle S. Botté, Madeleine JH van Oppen, Nicole S. Webster, and Thomas Rattei. HoloVir: a workflow for investigating the diversity and function of viruses in invertebrate holobionts. *Frontiers in Microbiology*, 7, 2016.
- [31] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, 2012.
- [32] Sonny T.M. Lee, Stacy A. Kahn, Tom O. Delmont, Alon Shaiber, Özcan C. Esen, Nathaniel A. Hubert, Hilary G. Morrison, Dionysios A. Antonopoulos, David T. Rubin, and A. Murat Eren. Tracking microbial colonization in fecal microbiota transplantation experiments via genome-resolved metagenomics. *Microbiome*, 5(1):50, 2017.
- [33] Bruce R. Levin and James J. Bull. Population and evolutionary dynamics of phage therapy. *Nature Reviews Microbiology*, 2(2):166, 2004.
- [34] Dinghua Li, Chi-Man Liu, Ruibang Luo, Kunihiro Sadakane, and Tak-Wah Lam. MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics*, 31(10):1674–1676, 2015.
- [35] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, and Richard Durbin. The sequence alignment/map format and SAMtools. *Bioinformatics*, 25(16):2078–2079, 2009.
- [36] Yang Li, Hao Wang, Kai Nie, Chen Zhang, Yi Zhang, Ji Wang, Peihua Niu, and Xuejun Ma. VIP: an integrated pipeline for metagenomics of virus identification and discovery. *Scientific Reports*, 6, 2016.
- [37] Folker Meyer, Daniel Paarmann, Mark D’Souza, Robert Olson, Elizabeth M. Glass, Michael Kubal, Tobias Paczian, A. Rodriguez, Rick Stevens, Andreas Wilke, J. Wilkening, and R. A. Edwards. The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics*, 9(1):386, 2008.
- [38] Susan Mills, Fergus Shanahan, Catherine Stanton, Colin Hill, Aidan Coffey, and R. Paul Ross. Movers and shakers: influence of bacteriophages in shaping the mammalian gut microbiota. *Gut Microbes*, 4(1):4–16, 2013.
- [39] Mohammadali Khan Mirzaei and Corinne F. Maurice. Menage a trois in the human gut: interactions between host, bacteria and phages. *Nature Reviews Microbiology*, 15(7):397–408, 2017.

- [40] Eric Morello, Emilie Sausseureau, Damien Maura, Michel Huerre, Lhousseine Touqui, and Laurent Debarbieux. Pulmonary bacteriophage therapy on *Pseudomonas aeruginosa* cystic fibrosis strains: first steps towards treatment and prevention. *PLoS One*, 6(2):e16963, 2011.
- [41] Toshiaki Namiki, Tsuyoshi Hachiya, Hideaki Tanaka, and Yasubumi Sakakibara. MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads. *Nucleic Acids Research*, 40(20):e155–e155, 2012.
- [42] H. Bjørn Nielsen, Mathieu Almeida, Agnieszka Sierakowska Juncker, Simon Rasmussen, Junhua Li, Shinichi Sunagawa, Damian R. Plichta, Laurent Gautier, Anders G. Pedersen, Emmanuelle Le Chatelier, et al. Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes. *Nature Biotechnology*, 32(8):822, 2014.
- [43] Sergey Nurk, Dmitry Meleshko, Anton Korobeynikov, and Pavel A. Pevzner. metaSPAdes: a new versatile metagenomic assembler. *Genome Research*, 27(5):824–834, 2017.
- [44] Nuala A. O’Leary, Mathew W. Wright, J. Rodney Brister, Stacy Ciufu, Diana Haddad, Rich McVeigh, Bhanu Rajput, Barbara Robbertse, Brian Smith-White, Danso Ako-Adjei, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, 44(D1):D733–D745, 2015.
- [45] Brian D. Ondov, Todd J. Treangen, Páll Melsted, Adam B. Mallonee, Nicholas H. Bergman, Sergey Koren, and Adam M. Phillippy. Mash: fast genome and metagenome distance estimation using MinHash. *Genome Biology*, 17(1):132, 2016.
- [46] David Paez-Espino, I. Chen, A. Min, Krishna Palaniappan, Anna Ratner, Ken Chu, Ernest Szeto, Manoj Pillay, Jinghua Huang, Victor M. Markowitz, Torben Nielsen, Marcel Huntemann, T. B. K. Reddy, Georgios A. Pavlopoulos, Matthew B. Sullivan, Barbara J. Campbell, Feng Chen, Katherine McMahon, Steve J. Hallam, Vincent Deneff, Ricardo Cavicchioli, Sean M. Caffrey, Wolfgang R. Streit, John Webster, Kim M. Handley, Ghasem H. Salekdeh, Nicolas Tsesmetzis, Joao C. Setubal, Phillip B. Pope, Wen-Tso Liu, Adam R. Rivers, Natalia N. Ivanova, and Nikos C. Kyrpides. IMG/VR: a database of cultured and uncultured DNA Viruses and retroviruses. *Nucleic Acids Research*, 45(D1):D457–D465, 2017.
- [47] David Paez-Espino, Emiley A. Eloë-Fadrosh, Georgios A. Pavlopoulos, Alex D. Thomas, Marcel Huntemann, Natalia Mikhailova, Edward Rubin, Natalia N. Ivanova, and Nikos C. Kyrpides. Uncovering Earth’s virome. *Nature*, 536(7617):425–430, 2016.
- [48] Rob Patro, Geet Duggal, Michael I. Love, Rafael A. Irizarry, and Carl Kingsford. Salmon provides fast and bias-aware quantification of transcript expression. *Nature Methods*, 14(4):417, 2017.

- [49] Yu Peng, Henry C.M. Leung, Siu-Ming Yiu, and Francis Y.L. Chin. IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics*, 28(11):1420–1428, 2012.
- [50] Nadège Philippe, Matthieu Legendre, Gabriel Doutre, Yohann Couté, Olivier Poirot, Magali Lescot, Defne Arslan, Virginie Seltzer, Lionel Bertaux, Christophe Bruley, Jrome Garin, Jean-Michel Claverie¹, and Chantal Abergel. Pandoraviruses: amoeba viruses with genomes up to 2.5 mb reaching that of parasitic eukaryotes. *Science*, 341(6143):281–286, 2013.
- [51] Aaron R. Quinlan and Ira M. Hall. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842, 2010.
- [52] Simone Rampelli, Matteo Soverini, Silvia Turrone, Sara Quercia, Elena Biagi, Patrizia Brigidi, and Marco Candela. ViromeScan: a new tool for metagenomic viral community profiling. *BMC Genomics*, 17(1):165, 2016.
- [53] Jie Ren, Nathan A. Ahlgren, Yang Young Lu, Jed A. Fuhrman, and Fengzhu Sun. VirFinder: a novel k-mer based tool for identifying viral sequences from assembled metagenomic data. *Microbiome*, 5(1):69, 2017.
- [54] Forest Rohwer and Rebecca Vega Thurber. Viruses manipulate the marine environment. *Nature*, 459(7244):207–212, 2009.
- [55] Simon Roux, Jennifer R. Brum, Bas E. Dutilh, Shinichi Sunagawa, M.B. Duhaime, A. Loy, B.T. Poulos, N. Solonenko, E. Lara, J. Poulain, Stéphane Pesant, Stefanie Kandels-Lewis, Cline Dimier, Marc Picheral, Sarah Searson, Corinne Cruaud, Adriana Alberti, Carlos M. Duarte, Josep M. Gasol, Dolors Vaqu, Tara Oceans Coordinators, Peer Bork, Silvia G. Acinas, Patrick Wincker, and Matthew B. Sullivan. Ecogenomics and potential biogeochemical impacts of globally abundant ocean viruses. *Nature*, 537(7622):689–693, 2016.
- [56] Simon Roux, Joanne B. Emerson, Emiley A. Eloë-Fadrosh, and Matthew B. Sullivan. Benchmarking viromics: an in silico evaluation of metagenome-enabled estimates of viral community composition and diversity. *PeerJ*, 5:e3817, 2017.
- [57] Simon Roux, François Enault, Bonnie L. Hurwitz, and Matthew B. Sullivan. VirSorter: mining viral signal from microbial genomic data. *PeerJ*, 3:e985, 2015.
- [58] Simon Roux, Michaël Faubladièr, Antoine Mahul, Nils Paulhe, Aurélien Bernard, Didier Debroas, and François Enault. Metavir: a web server dedicated to virome analysis. *Bioinformatics*, 27(21):3074–3075, 2011.
- [59] Simon Roux, Jeremy Tournayre, Antoine Mahul, Didier Debroas, and François Enault. Metavir 2: new tools for viral metagenome comparison and assembled virome analysis. *BMC Bioinformatics*, 15(1):76, 2014.

- [60] Silvia I. Sardi, Sneha Somasekar, Samia N. Naccache, Antonio C. Bandeira, Laura B. Tauro, Gubio S. Campos, and Charles Y. Chiu. Coinfections of zika and chikungunya viruses in bahia, brazil, identified by metagenomic next-generation sequencing. *Journal of Clinical Microbiology*, 54(9):2348–2353, 2016.
- [61] Lorian Schaeffer, Harold Pimentel, Nicolas Bray, Páll Melsted, and Lior Pachter. Pseudalignment for metagenomic read assignment. *Bioinformatics*, page btx106, 2017.
- [62] Nicola Segata, Levi Waldron, Annalisa Ballarini, Vagheesh Narasimhan, Olivier Jousson, and Curtis Huttenhower. Metagenomic microbial community profiling using unique clade-specific marker genes. *Nature Methods*, 9(8):811–814, 2012.
- [63] Arian FA Smit, Robert Hubley, and P. Green. RepeatMasker, 1996.
- [64] Saskia L. Smits, Rogier Bodewes, Aritz Ruiz-González, Wolfgang Baumgärtner, Marion P. Koopmans, Albert DME Osterhaus, and Anita C. Schürch. Recovering full-length viral genomes from metagenomes. *Frontiers in Microbiology*, 6:1069, 2015.
- [65] Sergei Solonenko, J. Ignacio-Espinoza, Adriana Alberti, Corinne Cruaud, Steven Hallam, Kostas Konstantinidis, Gene Tyson, Patrick Wincker, and Matthew B. Sullivan. Sequencing platform and library preparation choices impact viral metagenomes. *BMC Genomics*, 2013.
- [66] Charlotte Sonesson, Katarina L. Matthes, Malgorzata Nowicka, Charity W. Law, and Mark D. Robinson. Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage. *Genome Biology*, 17(1):12, 2016.
- [67] Saima Sultana Tithi, Frank O. Aylward, Roderick V. Jensen, and Liqing Zhang. Fastviromeexplorer: a pipeline for virus and phage identification and abundance profiling in metagenomics data. *PeerJ*, 6:e4227, 2018.
- [68] Duy Tin Truong, Eric A. Franzosa, Timothy L. Tickle, Matthias Scholz, George Weingart, Edoardo Pasoli, Adrian Tett, Curtis Huttenhower, and Nicola Segata. MetaPhlan2 for enhanced metagenomic taxonomic profiling. *Nature Methods*, 12(10):902–903, 2015.
- [69] Andries Johannes Van der Walt, Marc Warwick Van Goethem, Jean Baptiste Raymond, Thulani Peter Makhalanyane, Oleg Reva, and Don Arthur Cowan. Assembling metagenomes, one community at a time. *BMC Genomics*, 18(1):1, 2017.
- [70] Jorge F. Vázquez-Castellanos, Rodrigo García-López, Vicente Pérez-Brocal, Miguel Pignatelli, and Andrés Moya. Comparison of different assembly and annotation tools on analysis of simulated viral metagenomic communities in the gut. *BMC Genomics*, 15(1):37, 2014.

- [71] Bruce J. Walker, Thomas Abeel, Terrance Shea, Margaret Priest, Amr Abouelliel, Sharadha Sakthikumar, Christina A. Cuomo, Qiandong Zeng, Jennifer Wortman, Sarah K. Young, et al. Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. *PloS One*, 9(11):e112963, 2014.
- [72] Ryan R. Wick, Louise M. Judd, Claire L. Gorrie, and Kathryn E. Holt. Unicycler: resolving bacterial genome assemblies from short and long sequencing reads. *PLoS Computational Biology*, 13(6):e1005595, 2017.
- [73] K. Eric Wommack, Jaysheel Bhavsar, Shawn W. Polson, Jing Chen, Michael Dumas, Sharath Srinivasiah, Megan Furman, Sanchita Jamindar, and Daniel J. Nasko. VIROME: a standard operating procedure for analysis of viral metagenome sequences. *Standards in Genomic Sciences*, 6(3):421, 2012.
- [74] Ido Yosef, Miriam Manor, Ruth Kiro, and Udi Qimron. Temperate and lytic bacteriophages programmed to sensitize and kill antibiotic-resistant bacteria. *Proceedings of the National Academy of Sciences*, 112(23):7267–7272, 2015.
- [75] Merry Youle, Matthew Haynes, and Forest Rohwer. Scratching the surface of biology's dark matter. In *Viruses: essential agents of life*, pages 61–81. Springer, 2012.

Appendix A

Supplementary Material of Chapter 2



Figure A.1: Visualization of the reads mapped to the repeat region of BeAn 58058 virus.

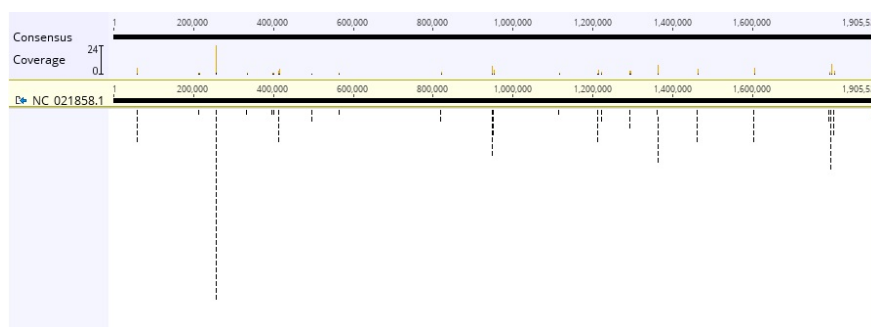


Figure A.2: Visualization of the reads mapped to the several repeat regions of Pandoravirus dulcis.

Table A.2: <https://peerj.com/articles/4227/>

Appendix B

Supplementary Material of Chapter 3

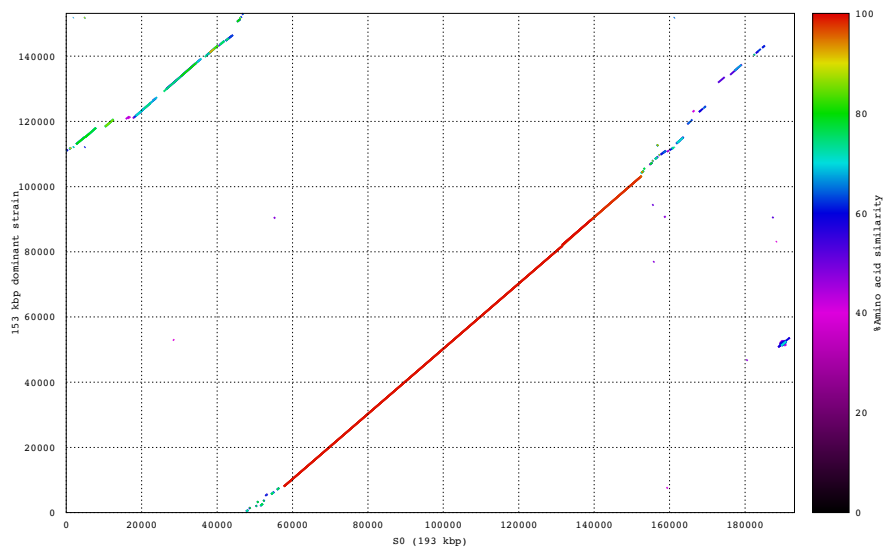


Figure B.1: Comparison of S_0 with 153 kbp scaffold representing the dominant strain of the novel virus recovered from S_0 .

Station ID	SRA ID	Paired-end reads	Time points (hr)
Station 6	SRX2912968	18,893,711	0
Station 14	SRX2912972	13,642,145	16
Station 18	SRX2912964	21,075,160	32
Station 22	SRX2912992	29,065,983	48
Station 28	SRX2912996	19,841,435	64
Station 32	SRX2912975	25,336,967	80
Station 37	SRX2912979	20,041,567	96
Station 52	SRX2912983	12,325,784	112
Station 56	SRX2912998	29,996,390	128
Station 61	SRX2913002	26,934,921	144
Station 67	SRX2912985	15,295,998	160
Station 70	SRX2912986	18,471,506	172

Table B.1: Description of the 12 viral-metagenomic samples collected from the study Aylward at el. Along with the time points when these samples were collected, as the sample at station 6 was collected first, we considered the time for collecting this sample as 0 hour.

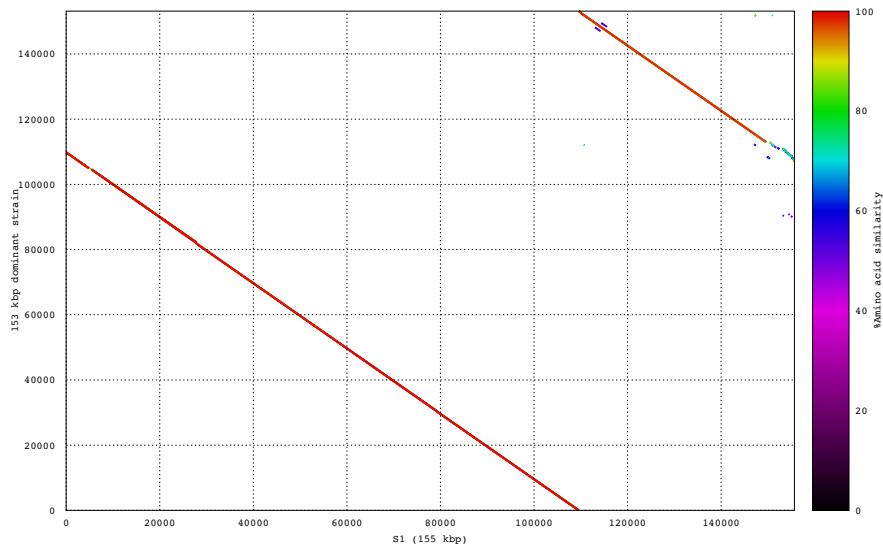


Figure B.2: Comparison of S_1 with 153 kbp scaffold representing the dominant strain of the novel virus recovered from S_0 .

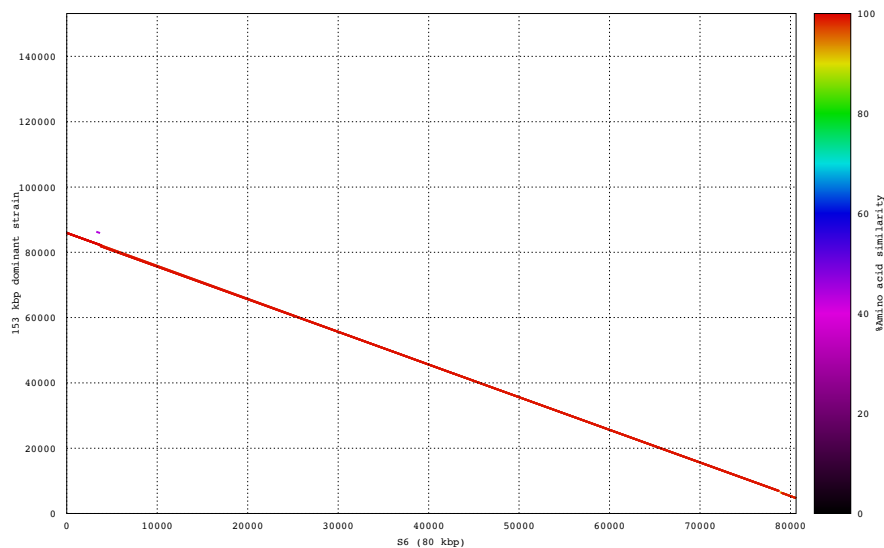


Figure B.3: Comparison of S_6 with 153 kbp scaffold representing the dominant strain of the novel virus recovered from S_0 .

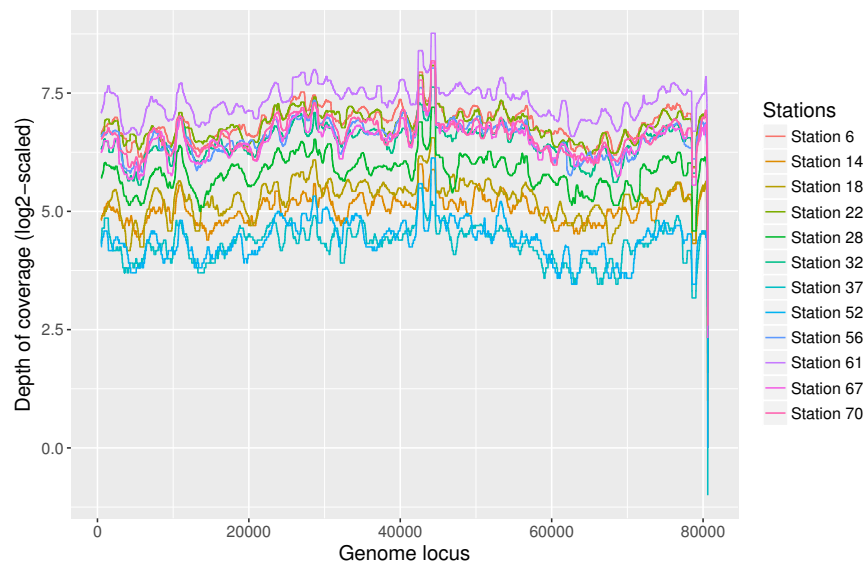


Figure B.4: The \log_2 -scaled depth of coverage of S_6 (80 kbp) across 12 ocean samples (showing median coverage of per 1000 bp window with step size 1).

Sample	Start position (bp)	End position (bp)	# of strains
1	132,719	152,719	3
2	27,126	47,126	3
3	79,335	99,335	3
4	49,195	69,195	3
5	114,254	134,254	2
6	88,301	108,301	3
7	96,101	116,101	3
8	71,396	91,396	2
9	108,715	128,715	2
10	69,477	89,477	2

Table B.2: Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 153 kbp dominant strain of the novel virus recovered from S_0 . Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.

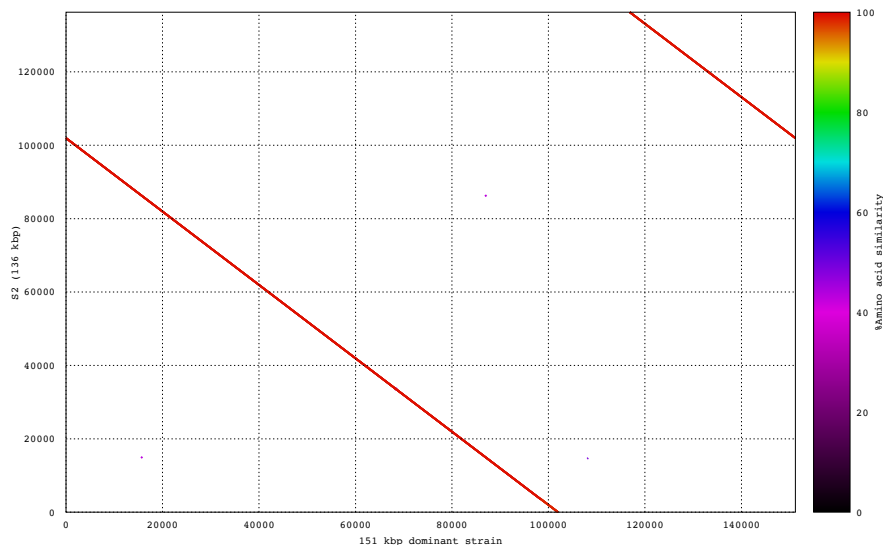


Figure B.5: Comparison of S_2 with 151 kbp scaffold representing the extended and complete version of S_2 .

Sample	Start position (bp)	End position (bp)	# of strains
1	64,623	84,623	3
2	76,337	96,337	4
3	112,158	132,158	2
4	1,012	21,012	2
5	5,581	25,581	2
6	59,837	79,837	2
7	90,933	110,933	3
8	131,132	151,132	2
9	39,143	59,143	2
10	125,497	145,497	3

Table B.3: Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 151 kbp scaffold representing the extended and complete version of $S2$. Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.

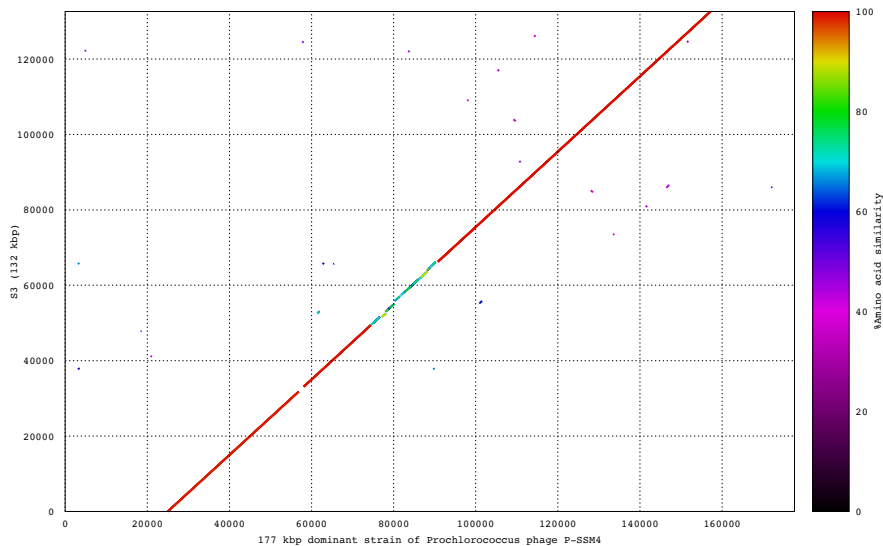


Figure B.6: Comparison of $S3$ with 177 kbp dominant strain of Prochlorococcus phage P-SSM4 (recovered from pieces of $S3$).

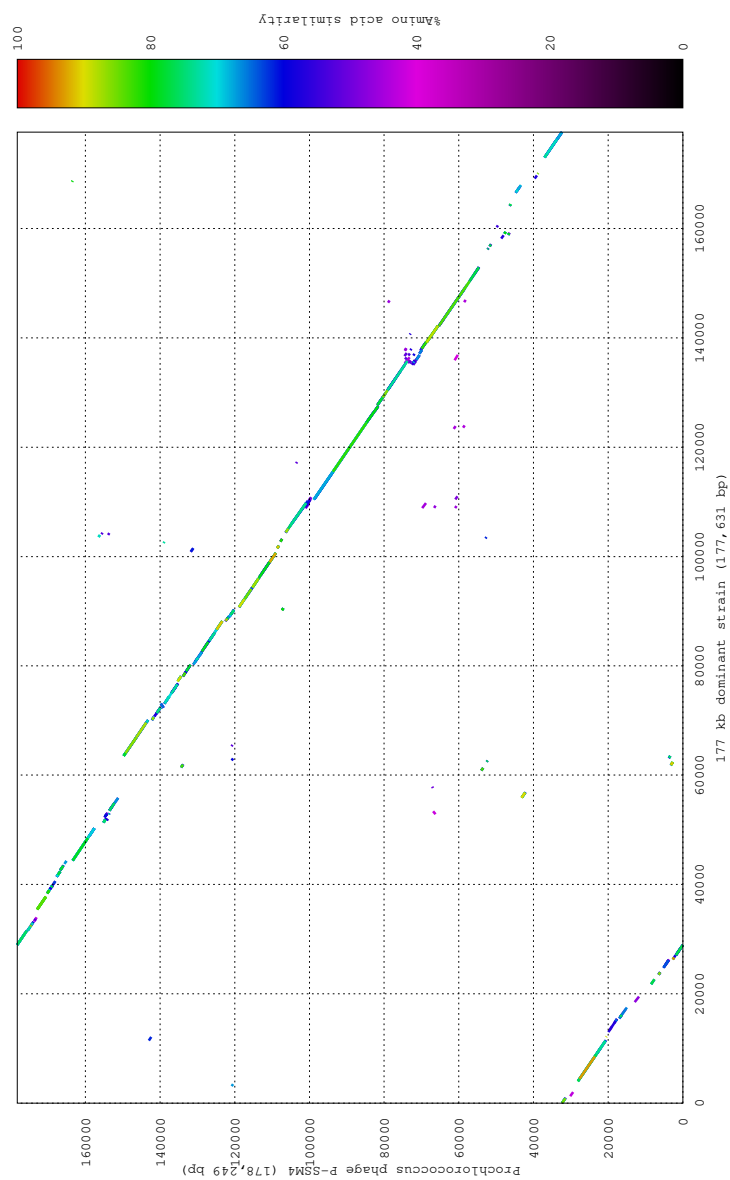


Figure B.7: Comparison of 177 kbp dominant strain with Prochlorococcus phage P-SSM4.

Sample	Start position (bp)	End position (bp)	# of strains
1	107,315	127,315	5
2	23,857	43,857	3
3	60,539	80,539	5
4	97,235	117,235	7
5	10,893	30,893	4
6	120,655	140,655	4
7	17,466	37,466	4
8	150,376	170,376	5
9	109,998	129,998	5
10	57,176	77,176	5

Table B.4: Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 177 kbp dominant strain of Prochlorococcus phage P-SSM4 (recovered from pieces of *S3*). Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.

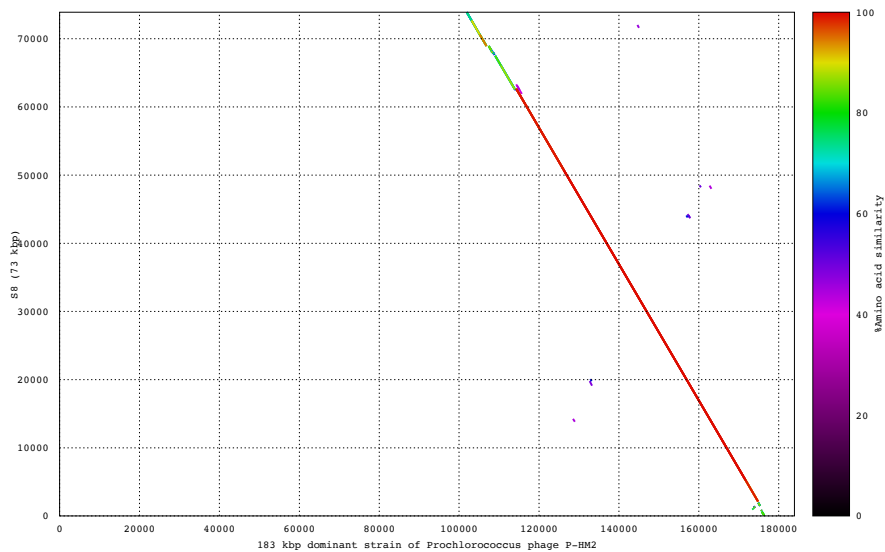


Figure B.8: Comparison of *S8* with 183 kbp dominant strain of Prochlorococcus phage P-HM2 (recovered from pieces of *S8*).

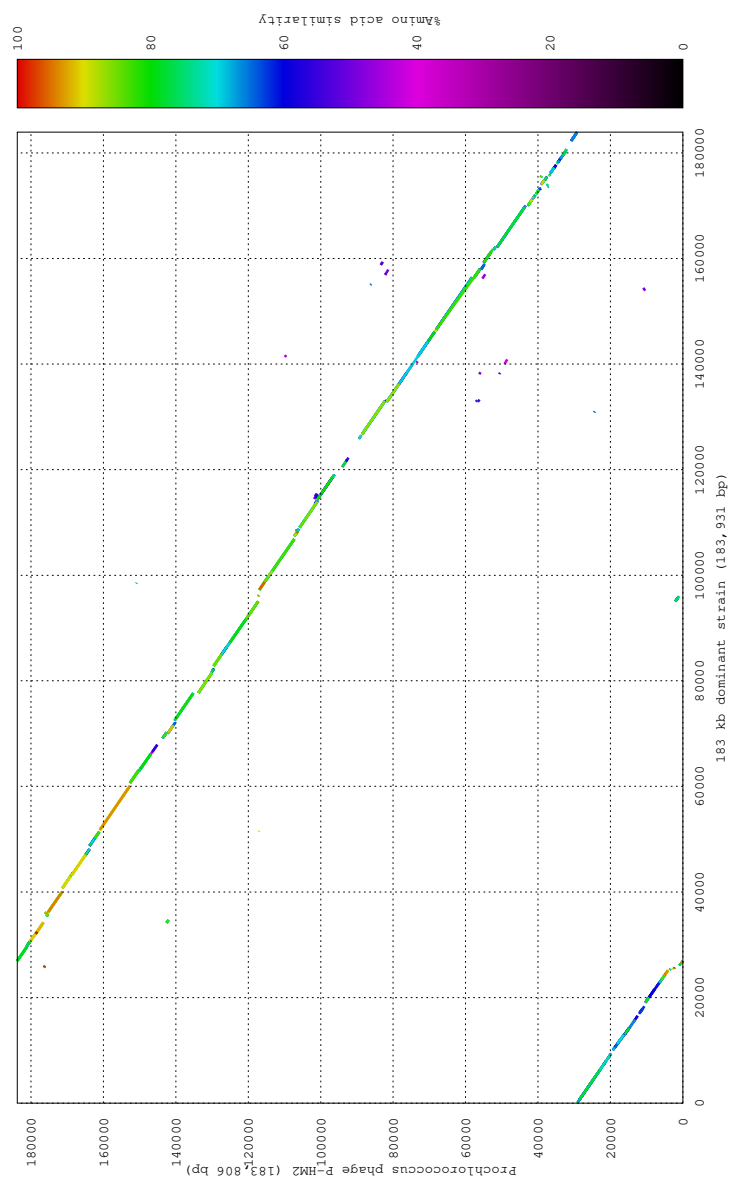


Figure B.9: Comparison of 183 kbp dominant strain with Prochlorococcus phage P-HM2.

Sample	Start position (bp)	End position (bp)	# of strains
1	161,965	181,965	3
2	8,470	28,470	3
3	37,142	57,142	3
4	78,461	98,461	2
5	123,677	143,677	2
6	30,857	50,857	4
7	68,632	88,632	2
8	99,087	119,087	2
9	117,063	137,063	3
10	2,356	22,356	2

Table B.5: Result of applying viral haplotype reconstruction tool, TenSQR on ten randomly selected pieces of 183 kbp dominant strain Prochlorococcus phage P-HM2 (recovered from pieces of *S8*). Here, each piece is 20 kbp long and for each piece, TenSQR reported multiple strains, it implies that multiple strains of this virus is present in the sample.

Appendix C

Supplementary Material of Chapter 4



Figure C.1: High coverage region of 151 kb dominant strain with length about 2 kb.

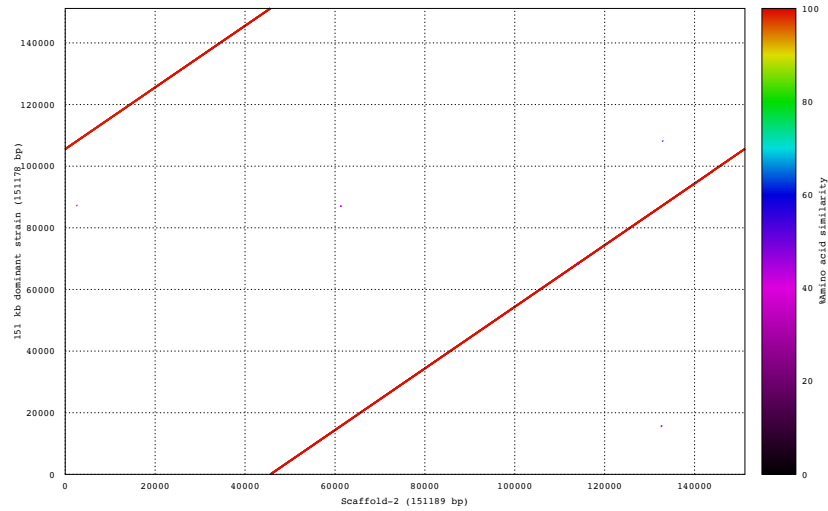


Figure C.2: Percentage of similarity of 151 kb dominant strain with the improved scaffold S2 obtained from VirChecker tool without applying Pilon tool.

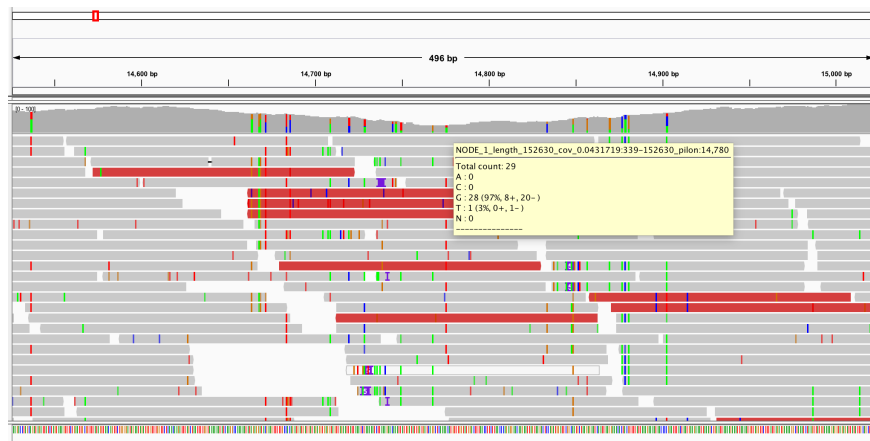


Figure C.3: Visualization of the low coverage region of S1.



Figure C.4: Visualization of the low coverage region of manually curated 177 kb dominant strain.



Figure C.5: Visualization of the low coverage region of *S3*.