

James Madison University

JMU Scholarly Commons

Senior Honors Projects, 2010-2019

Honors College

Spring 2015

Addressing the Black Box phenomenon of genome sequencing and assembly

Brandon Carter

James Madison University

Follow this and additional works at: <https://commons.lib.jmu.edu/honors201019>



Part of the [Bioinformatics Commons](#), [Computational Biology Commons](#), [Genomics Commons](#), [Other Genetics and Genomics Commons](#), and the [Virology Commons](#)

Recommended Citation

Carter, Brandon, "Addressing the Black Box phenomenon of genome sequencing and assembly" (2015). *Senior Honors Projects, 2010-2019*. 34.

<https://commons.lib.jmu.edu/honors201019/34>

This Thesis is brought to you for free and open access by the Honors College at JMU Scholarly Commons. It has been accepted for inclusion in Senior Honors Projects, 2010-2019 by an authorized administrator of JMU Scholarly Commons. For more information, please contact dc_admin@jmu.edu.

Addressing the Black Box Phenomenon of Genome Sequencing and Assembly

An Honors Program Project Presented to
the Faculty members of the Undergraduate
College of Integrated Science and Engineering
James Madison University

in Partial Fulfillment of the Requirements
for the Degree of Bachelor of Science

by Brandon Mark Carter

May 2015

Accepted by the faculty members of the Department of Integrated Science and Technology, James Madison University, in partial fulfillment of the requirements for the Degree of Bachelor of Science.

FACULTY MEMBERS COMMITTEE:

HONORS PROGRAM APPROVAL:

Project Advisor: Louise Temple, Ph.D.,
Professor, Integrated Science and Technology

Reader: Steven Cresawn, Ph.D.,
Professor, Biology

Reader: Stephanie Stockwell, Ph.D.,
Professor, Integrated Science and Technology

Philip Frana, Ph.D.,
Interim Director, Honors Program

ABSTRACT

Genomics, a study of all genetic material in an organism, is a new discipline having a great impact on medicine, agriculture, and environmental phenomena. Most undergraduate faculty members were not formally trained in genomics and must retool themselves in order to stay current with these evolving technologies. Advances in sequencing technology have resulted in an explosion of “big data” that can only be managed and analyzed using digital methods. Multiple complex computer programs are required to teach students the concepts using hands-on methods. These programs are challenging to use, especially since the same faculty members lacking genomics training were not trained in computer science, either. The Howard Hughes Medical Institute set out to provide large numbers of faculty members and students with the opportunity to isolate viruses that infect bacteria (bacteriophages) and perform genetic analyses of these new viruses. This SEA-PHAGES education program has the mission to promote research in the field of bacteriophage genomics while training undergraduate students in original research. Although training for the discovery lab and genomics is provided, there is an area in the genomics analysis that is missing from this training, and this represents a lost opportunity for faculty members to teach the entire process of genomic analysis to their students. The goal of this work was to ameliorate this deficit by creating a manual that addresses one of the most difficult aspects of genomics analysis, assembly of DNA sequencing reads that are produced by modern sequencing equipment. In order to determine the interest in such a manual and assess the content to be included in the manual, SEA-PHAGES faculty members were surveyed at 87 universities involved in teaching the SEA-PHAGES viral discovery course. Of those surveyed, 39 faculty members responded, and the responses were used to guide the development of the manual. The

manual contains procedures necessary to assemble DNA and analyze the assembled output in a manner accessible to both faculty members and students. It guides the user from handling the raw sequence data through fully assembled viral genomes ready for genetic analysis, thus illuminating the “black box” phenomenon. The manual was piloted in two genomics courses at James Madison University and by multiple professors at other SEA-PHAGES member universities. The responses have been used to make improvements in the manual. The findings from this pilot test indicate that the manual was an effective tool for science professors and their students. The manual will be distributed freely and made available to researchers associated with the SEA-PHAGES consortium.

TABLE OF CONTENTS

Abstract.....	2
List of Figures.....	5
Acknowledgements	6
Introduction.....	7
Methodology	14
Initial Survey to Gauge Interest	14
Learning Sequence Assembly and Assembly Analysis Software	14
Design Manual.....	15
Experiments During Manual Design.....	17
Downsampling	17
Finding The Ends of the Genome	19
Feedback from First Draft of Manual.....	20
Pilot Test and Second Survey to Gather Feedback.....	21
Finalized Manual	21
Results	22
Initial Survey to Gauge Interest	22
Experiments During Manual Design.....	22
Downsampling	22
Finding the Ends of the Genome	24
Pilot Test and Second Survey to Gather Feedback.....	29
Finalized Manual	30
Discussion	31
Conclusions.....	34
Future Work.....	35
References.....	37
Appendices.....	40

LIST OF FIGURES

Figure 1	12
Figure 2	17
Table 1	23
Figure 3	24
Figure 4	25
Figure 5	26
Figure 6	26
Figure 7	27
Figure 8	28
Figure 9	28
Figure 10	29

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Louise Temple and my honors thesis committee members Dr. Steven Cresawn and Dr. Stephanie Stockwell for their guidance and support throughout this project. I would also like to thank Dan Russell at the University of Pittsburgh for his wonderful tutorials and endless resources on phagesdb.org. Thank you to all of the professors in the SEA-PHAGES program and the students in Dr. Louise Temple's genomics course who provided feedback on the manual and responded to the surveys. Thank you to the Integrated Science and Technology department and the JMU Honors Program for allowing me the opportunity to complete this thesis.

INTRODUCTION

The interdisciplinary fields of bioinformatics and genomics have grown exponentially over the past few decades. In the medical field, this has allowed researchers to further their understanding of drug response and disease (1). In the biotechnology field, increased genomic knowledge has led to the growth of synthetic biology (2). However, despite its importance, a lack of understanding continues to exist regarding how to perform the bioinformatic steps involved in DNA sequence analysis. In this genomic process, DNA sequences of organisms are determined and analyzed using computer programs that create huge datasets to be stored in large databases (3). Over the years, a black box phenomenon has emerged among many faculty members trained in the pre-genomics era, in which the process of transforming raw sequence data into an assembled and annotated genome is not well understood (4). Due to this phenomenon, academic researchers bypass this aspect of genomics and instead contract the work to centers that specialize in genome assembly and analysis. As a result, educational opportunities are lost for students. One significant obstacle must be overcome in order to gain a better understanding of the processes of DNA assembly and assembly analysis. There is an absence of strong technical guidance in learning the functionality of these programs, due to the lack of many professors being trained in these processes before their teaching careers began.

One national program that knows this obstacle very well is the Science Education Alliance Phage Hunters Advancing Genomics and Evolutionary Science (SEA-PHAGES) program. The Howard Hughes Medical Institute (HHMI) founded this educational program in 2008 and it currently contains 87 actively participating universities and has reached over 4800 undergraduate students. The organization's mission is to further research in the field of bacteriophage genomics while

training undergraduate students in original research. Bacteriophages (viruses that infect bacteria) are the subject of the SEA-PHAGES program and their discovery allows students the interesting opportunity to isolate, name, sequence, and analyze them (5). Although the SEA-PHAGES program provides training to faculty members for the discovery lab and genomics portions of the course, the student manual is missing the genomics steps regarding DNA assembly and most participating schools use the finished genomes for annotation. Professors in this program want to provide their students an understanding of the entire process from bacteriophage discovery through genomic analysis without creating any knowledge gaps. Thus, the students are missing out on a vital step in the process of bacteriophage discovery in genomics. For those faculty members without formal training, they are left to learn these processes on their own or rely on help from colleagues. This is complicated by the constantly evolving technologies used in genomic analysis. As a result, faculty members must constantly retool themselves in order to relay current information to their students. Learning how to assemble DNA sequencing reads is also difficult due to the disconnect that exists between software engineers and users. The computer programs used to assemble DNA are confusing to use and many are run from the command line within the Linux operating system, which is unfamiliar to many researchers.

The entire process of bacteriophage discovery from isolation of a new bacteriophage to sequencing, assembling, and genome annotation is a time consuming process. Streamlining and making accessible the steps in the genomic analysis processes can improve the efficiency of achieving a final product and increase the learning potential for students. Two aspects of genomics that would benefit from a simple, step-by-step manual are the assembly process and finding the ends of the genome. Such a manual would enable professors to involve their students

in this aspect of the workflow, which is uncommon at present in most institutions. Although programs are available with instructions, understanding, mastering, and utilizing programs in a teaching environment is challenging. Having instructions in one manual would be helpful to faculty members who want to make these processes part of their teaching in genomics. Such a manual should contain methods for efficiency in time spent on the processes, in order for demonstration during a single class period. For example, reducing the size of the entire FASTQ dataset reduces the assembly time to minutes instead of hours. In addition, the outcomes of assembly are more accurate with a smaller dataset. Because they don't do the assembly themselves, many faculty members rely on the sequencing facility personnel to find the genome ends and assess the quality of the assembled genome. A manual with a simple explanation of this process using examples would facilitate professors adding this part of the process to their curriculum and would help students understand the bacteriophage life cycle and structure of bacteriophage genomes. These are examples of the sort of information that could be made available in an accessible format for teachers to use with their students, if they are interested in instituting the complete process with their students, from raw data to a finished genome.

Other issues can be addressed with a step-by-step, accessible set of instructions. For example, most professors in the SEA-PHAGES program use virtual machines to run these programs in a Linux environment. Virtual machines share memory allocation with the host environment, so they are limited in their ability to multitask. Not only is a virtual machine slower than a host Linux machine, it also slows down the host environment thus limiting the performance of the host operating system (6). The longer the genome is assembling, the longer that the computer's performance will be negatively affected. Streamlining this process will allow for multiple

genomes to be assembled on the same computer in the same day. It will also allow for the host operating system to be used without decreased performance. Another process in need of an efficiency improvement is finding the ends of the genome.

In order to fully understand what these computer programs do, it is essential to understand the scientific background of genomic analysis. Genome analysis includes three main steps: sequencing, assembly, and annotation. DNA sequencing is the process by which the sequence of a DNA molecule is obtained (7). In genomics, an organism's entire genome is sequenced. In the assembly process, a computer algorithm aligns and merges the small DNA fragments created during sequencing, known as reads, into one continuous DNA segment called a contig (8). In the final phase, annotation, biological information about the DNA sequence is determined (9).

DNA sequencing has traditionally been done by the dideoxy termination method, also known as the Sanger method. This method begins with fragmenting the DNA and then cloning and amplifying the DNA fragments *in vivo*. The DNA is then denatured and a sequencing primer is annealed to the single-stranded DNA. Dideoxynucleotide triphosphates (ddNTPs) then terminate the DNA chain during elongation. The resulting polymerized DNA fragments are then separated by gel electrophoresis. The fluorescently labeled ddNTPs allow the sequence to be determined for each fragment of DNA. In recent years, next generation (next gen) sequencing techniques such as Ion Torrent sequencing, 454 pyrosequencing, and Illumina sequencing have been utilized due to their high throughput. These sequencing methods use a sequence-by-synthesis approach where one base at a time is added to the template. Each time a base is incorporated, a signal is detected. Whereas Sanger sequencing utilizes *in vivo* cloning and amplification, next gen

techniques utilize *in vitro* adapter ligation to the ends of the fragments. These adapters are short, chemically synthesized, double-stranded DNA used to connect the ends of two DNA molecules. The difference in throughput between the two methods is immense since next gen methods can produce thousands of more reads per run than the Sanger method (7). Data obtained from these next generation sequencing instruments are often saved as SFF or FASTQ files. A SFF file is a binary file that can be tedious to work with, whereas a FASTQ file is a much more usable text file that combines the sequence with quality score for each base (10). Once the genomic data is obtained from a sequencing instrument, it can be assembled using a DNA assembly computer program. This is the stage of genomic analysis where the black box phenomenon exists. There are two types of assembly methods: *de novo* assembly and mapping assembly. They differ in that the reads are assembled in *de novo* assembly to create full-length sequences without the use of a reference sequence, as in mapping assembly. Since the bacteriophages that are analyzed are assumed to be novel, *de novo* assembly is used. In the first step of *de novo* assembly, the sequence and quality data contained in the FASTQ file are read and the reads are cleaned. This process of sequence cleaning involves removing the adapter sequences from the FASTQ file (11). After the sequence is free from contaminants, overlaps are detected between the reads and the reads are grouped to form a contig. In the final step of assembly, a consensus sequence is determined for the contig. Prior to annotation, the assembled sequence must be analyzed in order

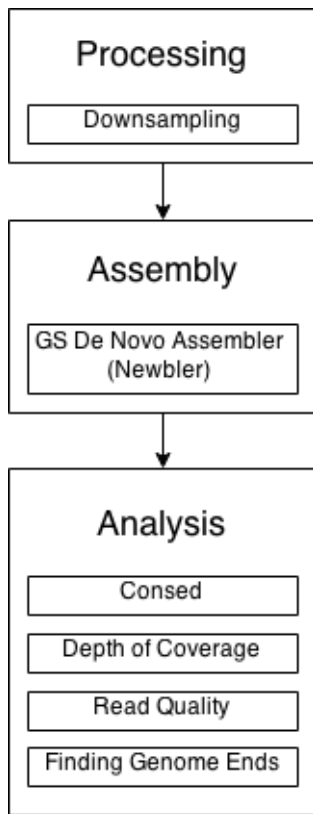


Figure 1. Workflow of DNA sequence assembly and assembly analysis.

to determine certain characteristics of the genome. One important step in this process is identifying the ends of the genome. In more advanced analysis, the quality of the assembled data is examined in a process known as finishing (8). A workflow of the procedures involved in DNA assembly and assembly analysis is illustrated in Figure 1. The final stage of genomic analysis is annotation. In this process, key features of the genome are determined using a combination of multiple computer programs. This step connects the sequence of the organism to its biological functions since this is that step at which gene mapping and protein function determination occurs (9).

Along with DNA sequencing, the steps involved in annotation tend to be understood better among biologists than those of DNA assembly. Thus, the primary scope of this project is to address the issue that faculty members in the SEA-PHAGES program are not trained in

DNA assembly. This is addressed by the creation of a manual that will take users through the black box from raw sequence data to fully assembled viral genomes. The manual will allow researchers at SEA-PHAGES member universities to increase their knowledge of scientific processes and computer programs related to DNA assembly and assembly analysis. By using this manual, more faculty members and their students will be able to assemble and analyze their own bacteriophages, thus resulting in an increase of in-house scientific research. This concept aligns

well with the mission of SEA-PHAGES program, because this increase could allow for more undergraduate research work at these universities.

In accordance with this project's goals, the manual was aimed to address both researchers who are fairly new to performing genome assembly and assembly analysis, and their undergraduate students. Students at JMU as well as faculty members at other universities pilot tested the manual in order to evaluate its effectiveness. Those who test the manual had varying degrees of prior knowledge, thus allowing for feedback from both experienced and inexperienced users. This ensures that the manual addressed the concerns of as many potential users as possible. The resulting implications of this project are not solely limited to those within the SEA-PHAGES program. While the manual will initially be implemented at JMU and other SEA-PHAGES schools, the results should have global applicability to anyone using the same computer programs for viral genomic research. By illuminating the black box phenomenon surrounding DNA assembly and assembly analysis, the barriers are reduced.

This project aimed to reduce these barriers by first sending out a survey to gauge interest in the possible creation of an assembly and assembly analysis manual from the SEA-PHAGES community. Then, the manual was then created and experiments were performed to validate methodologies regarding DNA assembly and finding the ends of a genome. This was followed by the pilot test by students in Dr. Temple's genomics course and various SEA-PHAGES professors. The final phase was to edit the manual to address the feedback from the pilot test and house the manual online for professors to download and share with their students.

METHODOLOGY

INITIAL SURVEY TO GAUGE INTEREST

A survey was designed using the online questionnaire tool SurveyMonkey in order to gauge interest from professors in SEA-PHAGES to determine critical design components of the manual (12). The survey was emailed to every faculty member in the SEA-PHAGES program. The questions posed in the survey include whether researchers in the SEA-PHAGES program were processing any genome data that preceded annotation and if they were doing so with the assistance of undergraduate students. Other questions that were asked pertained to the specific assembly and assembly analysis programs in use as well as the vehicle in which professors would prefer the manual to be displayed. The vehicle, be it a PDF or a living wiki document to be edited by members of the SEA-PHAGES community, is important in determining the number of people who would potentially utilize the manual. For those professors who were not processing genome data prior to annotation, it was important to determine what obstacles, if any, were preventing them from doing so. For those who were already processing genome data, learning their perceptions of the weakest aspects of current resources was important so that the manual can address these well. This question was critical to maximizing the number of professors who would potentially utilize the manual. The survey, approved as IRB protocol #15-0313, is displayed in its entirety in Appendix 1.

LEARNING SEQUENCE ASSEMBLY AND ASSEMBLY ANALYSIS SOFTWARE

From the results of the first survey, the most common tools for assembling bacteriophage genomes and analyzing the assembly files were identified within the population of SEA-

PHAGES respondents. For genome assembly, most of the respondents were using Roche's proprietary assembly software GS *De novo* Assembler, hereafter referred to by its more common name, Newbler (13). For assembly analysis, most were using Consed (14). Determining the basic functionalities of these programs was essential before beginning to design the manual. This included learning how to use the programs as well as understanding the flow of genome data from entering Newbler for assembly and exiting Consed after data analysis. Since both of these products are operated in a Linux operating system, a basic understanding of how to navigate the Linux command-line interface was necessary. These fundamentals were learned from Dr. Louise Temple of JMU in Spring 2014.

DESIGN MANUAL

From the results of the first survey, it was determined that the respondents would prefer a PDF document rather than a living wiki document. With this knowledge, the manual was designed to include multiple aspects that one would need to learn the basics of bacteriophage DNA assembly and assembly analysis using Newbler and Consed, respectively. The first subject addressed were the initial steps that a user would need in order to begin using these programs. This included a discussion of the workflow placing the contents of this manual in the overall context of bacteriophage discovery, downloading the SEA virtual machine that houses local downloads of the software, and an overview of the software. Next, the processes of DNA assembly and analysis are discussed in depth.

The first step in the workflow is the processing of FASTQ data prior to assembly. In the second step, using the Newbler GUI to assemble a genome is explained (15). The third step, assembly

analysis, is the most deeply discussed step in the manual because it is the most difficult and least understood process among the SEA-PHAGES community. Before beginning with the analysis procedure, opening a Newbler assembly in Consed is explained and the Consed GUI is introduced. These sections serve to ease the users into Consed so they can learn how to navigate the application, which must be launched from a command line (16, 17). The manual then addresses how to use Consed to analyze the quality of an assembly by observing areas in the genome with low consensus overlap and areas of high or low depth of coverage (18). This is the beginning of the process for finding the ends of a genome. This section on finding ends is focused on how to find two main types of genome ends: defined ends with 5' and 3' distinction, a subset of defined ends known as terminal repeats, and circularly permuted ends. After analyzing the assembly to find the ends, the manual discusses determining the cluster to which the bacteriophage belongs (19). This is important to categorize bacteriophages based on similar traits. Lastly, the manual discusses adding those reads that were removed earlier prior to the assembly in the downsampling phase back to the existing Consed project (15). The table of contents for the manual is displayed in Figure 2 and the manual in its entirety is displayed in Appendix 3.

Contents	
Workflow	4
Downloading SEA Virtual Machine	4
Software Overview	5
Steps of Processing, Assembly, and Analysis	5
Processing	5
Selecting a Subset of Reads from a FASTQ file	5
Assembly	6
GS De Novo Assembler (Newbler) [6]	6
Assembly Analysis	7
Opening Consed	7
Determining Which Contig to View	8
Consed Views [8]	9
Visual Analysis of Read Quality (Chromatogram)	11
Coverage and Weak Areas	12
Finding the Ends of the Genome	13
Determine Phage Cluster [12]	21
Adding SFF or FASTQ reads to an existing Consed project [13]	21
Newbler from the Command Line	22
Navigating the Command Line in Linux for Beginners	22
Example Phage Data	23
References	24

Figure 2. Table of Contents of the bacteriophage DNA assembly manual.

EXPERIMENTS DURING MANUAL DESIGN

Downsampling

Throughout the manual design process, there were multiple experiments performed to validate methodologies for DNA assembly that would also result in a high level of data integrity.

Assembling entire FASTQ files with all of the reads intact takes multiple hours to run. With the long period of time that it takes from isolation of a bacteriophage sample through the annotation stage, any time that can be cut off this process is valuable. Thus, determining a way to speed up the assembly process is important.

From the reviewed literature, a process was discovered known as downsampling. This process involves removing reads from a FASTQ file so that less reads are assembled in Newbler. With less reads having to assemble in Newbler, the assembly should theoretically speed up. The reads are removed by first opening a terminal and navigating to the correct directory before inputting the following command: **head -n “*the number of reads desired*” “*name of the original FASTQ file*” > “*name of the new FASTQ file*”** (20). In this command, the **head -n** portion will print the first *x* number of lines of a given file to standard output file(21). Since FASTQ is formatted as a standard text file with four lines per read, the number of reads written to the output file will be the number of lines specified in the command divided by four. For example, **head -n 400000** will only write the first 100000 reads of the original FASTQ file to the new downsampled FASTQ file.

This downsampling method was applied using the FASTQ file of the bacteriophage Karezi in order to determine if the assembly was faster than the traditional approach of an entire FASTQ file, and if the data integrity was upheld throughout the process. To do this, the full Karezi FASTQ file was assembled without downsampling using Newbler and then downsampled Karezi FASTQ files of four different sizes were assembled. The read sizes 50000, 100000, 150000, and 200000 were chosen because it is discovered from the literature review to downsample FASTQ files to as low as 50000 reads (20). The resulting assemblies were compared in order to assess the data integrity. This was done by observing the number of contigs per assembly, comparing BLASTn results of the contig from each assembly that matched the number of bases of the known Karezi bacteriophage, and comparing the sequence alignments of all four assemblies using the program Clustal Omega. The relative elapsed time to assemble the files was also

measured to determine whether downsampling was faster than the traditional method. For a more detailed explanation of how this process was done, refer to the explanation in the section *Selecting a Subset of Reads from a FASTQ File*, which is located in the manual in Appendix 3.

FINDING THE ENDS OF THE GENOME

An important aspect of genome analysis is the process of finding the ends of a genome. Without a method in place to quickly find the ends of the genome, the entire genome would have to be viewed on a base-by-base approach in Consed's Aligned Reads view. This is an arduous process considering that bacteriophage genomes tend to be tens of thousands of bases long. An approach was discovered from the reviewed literature where regions of the genome designated as having high or low depth of read coverage were isolated and analyzed for the presence of ends in the nearby vicinity (18).

Since defined ends are known to occur at dramatic decreases in read coverage and terminal repeats exhibit dramatic increases in depth of coverage, regions of the genome can possibly be isolated by this approach to expedite the process of finding genome ends. The reason that a genome with defined ends will occur at a region with a dramatic decrease in read coverage is because the bacteriophage DNA is always packaged with the same start and end positions, resulting in a buildup of reads at the start and end positions. There are two forms of defined ends as represented in the software. In one there is a small gap between the ends on the top and bottom strand, known as a 3' overhang, and in the other there is a small overlap, known as a 5' overhang (22). In both instances, the gap or overlap is only a handful of bases long. Genomes with terminal repeats appear at prolonged regions of high read coverage because the

bacteriophage DNA is packaged with consistent ends, but more than one full copy of the genome is packaged, thus resulting in a buildup of reads at the regions where overlapped genome appears. This results in a buildup of reads at the ends, but whereas the physical end with a 5' overhang may have a small overlap between the ends, there will be a much larger overlap between the ends in a bacteriophage with a terminal repeat lasting a few hundred a few thousand base pairs. Since a circularly permuted genome does not have a defined end, it should be indistinguishable by either of the methods above and as such there will be no buildup of reads (23).

This method is done by first isolating these regions of high or low depth of coverage in the Assembly view within Consed and then examining the regions in question in the Aligned Reads view to find the ends of the genome. This process was performed using three bacteriophages whose genomes had previously been determined in order to validate this method. Each of the genomes tested exhibits a different type of end so the method can be tested for each type of end. The bacteriophage genomes tested were OrionPax (defined ends), Pinkman (defined ends with terminal repeats), and Waukesha92 (circularly permuted). For a more detailed explanation of how this process was done, refer to the explanation on Finding Ends of The Genome of the manual, which is located in Appendix 3.

FEEDBACK FROM FIRST DRAFT OF MANUAL

After writing a preliminary draft of the manual, students in Dr. Louise Temple's genomics class tested it in Fall 2014 and Spring 2015. The students were instructed to provide qualitative feedback on any aspect of the manual that was either incorrect, confusing, or could be elaborated

further. They were also asked to fill out a second survey that was constructed, again using SurveyMonkey, which is described in depth in the succeeding methodology section, Pilot Test and Second Survey to Gather Feedback (12).

PILOT TEST AND SECOND SURVEY TO GATHER FEEDBACK

In order to test the effectiveness of the manual at other SEA-PHAGES member universities, the professors who responded to the original survey were emailed and asked if they would be willing to pilot test the manual with their students and provide feedback. Sent along with the email message were the following: the manual as an attachment, a link to a second survey designed to provide quantitative feedback, and an email address for the professors to provide further qualitative feedback on the manual. The survey asked questions such as whether the manual addressed the basic needs of the professor and their students, satisfaction ratings for each section, and which section(s) were in need of the most improvement. The survey, approved as IRB protocol #15-0313, is displayed in its entirety in Appendix 2.

FINALIZED MANUAL

With the feedback gathered from the students in Dr. Temple's genomics class and the pilot test, the manual was updated to improve its accuracy and effectiveness.

RESULTS

INITIAL SURVEY TO GAUGE INTEREST

The initial survey to gauge the interest of the SEA-PHAGES community was answered by thirty professors. The response rate was 24% of professors with 41% of the SEA-PHAGES universities represented. Of the professors who responded, 60% were currently processing genome data prior to annotation. The majority of professors were using Newbler and Consed to process genome data. It was reported that the most difficult aspect of this process was finding the ends of genomes. Of the 40% who were not currently processing genome data prior to annotation, 85% were interested but were not doing so for a variety of reasons. The most common obstacles were a lack of knowledge and training on how to use the programs associated with these tasks and a lack of time or desire to do so since these tasks could be done for them by another university. Of these professors, 93% would be interested in using a how-to guide that addresses the steps of sequence assembly and analysis and 80% would prefer it to be in the format of a digital user manual instead of a wiki-style webpage. It was also found that 69% of these professors were working with students to process genome data. The survey results in their entirety can be found in Appendix 4.

EXPERIMENTS DURING MANUAL DESIGN

DOWNSAMPLING

From the literature review of the processing FASTQ data step, it was determined that reducing the number of reads in the FASTQ file would significantly reduce the time required to assemble

a genome (20). Each of the downsampled Karezi files assembled a different number of contigs assembled; however, each assembly as able to produce a contig of the correct size for the previously determined Karezi genome. The Newbler assembly results are displayed in Table 1. The time comparison of the assembly runs is shown in Table 1. Whereas it takes multiple hours to assemble the full FASTQ file that has traditionally been run overnight, each of the four downsampled files were able to assemble in less than fifteen minutes.

Table 1. Newbler assembly results for downsampled Karezi FASTQ files. The full Karezi FASTQ file contains approximately 924,000 reads.

Karezi File Size	Assembly of 20084 bp Contig	Assembly Time (minutes)
50,000 reads	Yes	2:07
100,000 reads	Yes	5:20
150,000 reads	Yes	9:37
200,000 reads	Yes	14:28

The BLASTn for each of the four contigs resulted in the same information. Each contig matched the accession number X96987.2 with 70% identity and an E-value of 0. The Clustal Omega results, of which an excerpt is shown in Figure 3, indicate a 100% identity match among the DNA sequences of the four assembled contigs.

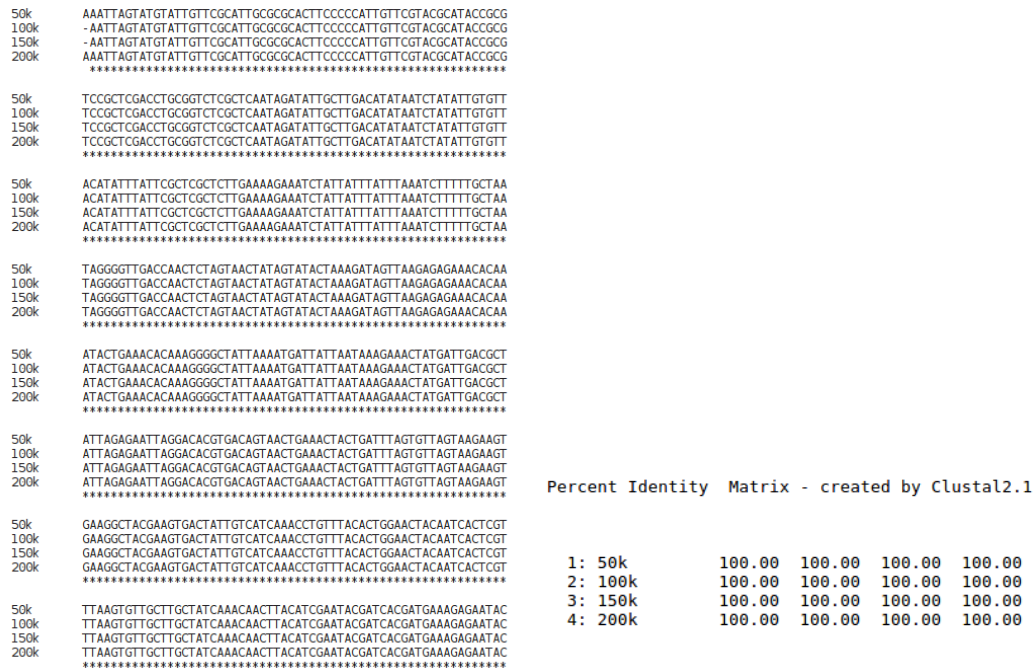


Figure 3. Clustal Omega multiple sequence alignment results and percent identity matrix for the four downsampled FASTQ files.

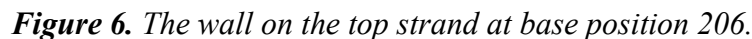
FINDING THE ENDS OF THE GENOME

From the literature review of assembly analysis, it was determined that locating regions in the genome with high or low depth of coverage could expedite the process of finding the ends of genomes (18, 22, 23). This method allows for small regions of interest to be targeted, thus expediting the process. Since OrionPax exhibits physically defined ends, areas with low coverage were examined in Consed's Assembly View. The result was multiple areas that exhibited the expected dramatic decrease in coverage. Each of these areas exhibiting low read depth (estimated to be approximately less than 150 reads in this genome) was analyzed in the Aligned Reads view. It was found that the reads on the bottom and top strands ended at base position 198 and 206, respectively, thus creating a wall on each strand. The Assembly View

results of OrionPax are displayed in Figure 4 and the Aligned Reads view of the bottom and top strands are displayed in Figures 5 and 6, respectively.



Figure 4. Assembly View of OrionPax. Low depth of coverage region with read depth circled at which walls were later found in the Aligned Reads view. Max depth of coverage parameter was set to 150 since the average read depth across the genome is above this number. The Assembly View window is zoomed in to show an excerpt of the low depth of region in question.



Since Pinkman exhibits defined ends with terminal repeats, areas with high coverage were examined in Consed's Assembly View. The result was one region around the base position 130000 that exhibited a prolonged increase in read depth. The high depth of coverage region was set to only display regions of 55 reads or above since the depth of coverage appeared to be over 90 reads in this region and the average across the rest of the genome was below 50 reads. The only region isolated by this method was between the positions 128903 and 130866, so it was assumed that the ends would be at or near these two positions. It was found that the end on the top strand was at exactly position 128903 and the end on the bottom strand was at 130902. The Assembly View results of Pinkman are displayed in Figure 7 and the Aligned Reads view of the top and bottom strands are displayed in Figures 8 and 9, respectively.

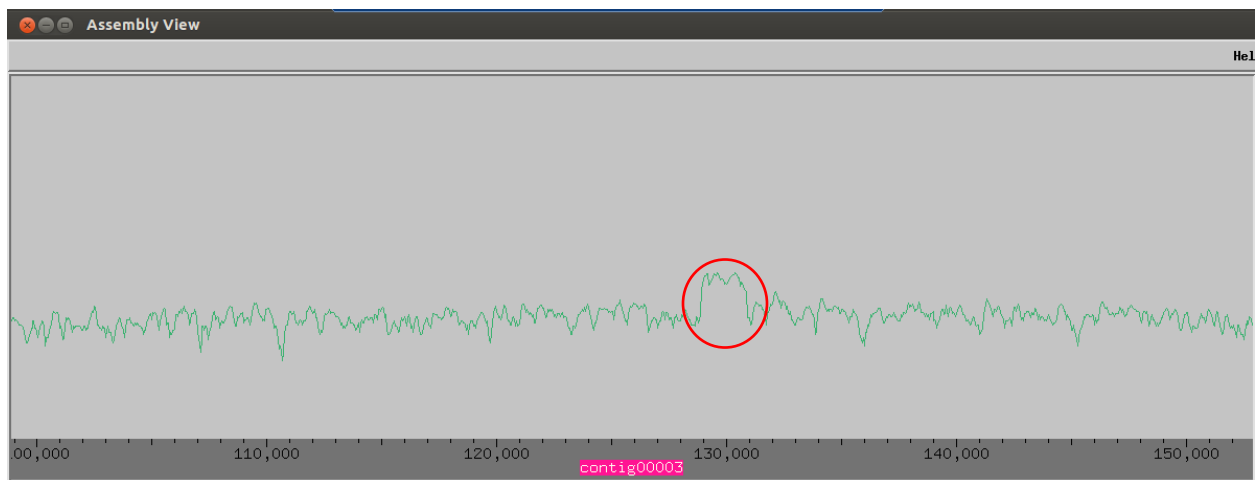


Figure 7. Pinkman Assembly View. The read depth specified for the high depth of coverage region was 55 reads. This was determined using the same method as explained in Figure 4, but for the average read depth below the high depth region as opposed to above it.

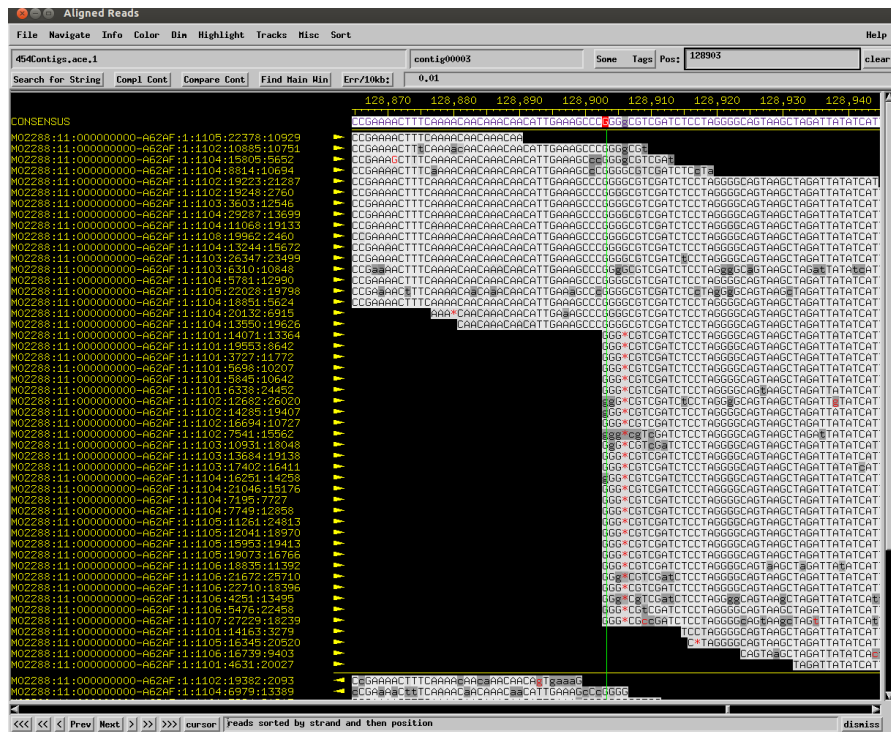


Figure 8. The wall on the top strand at base position 128903.

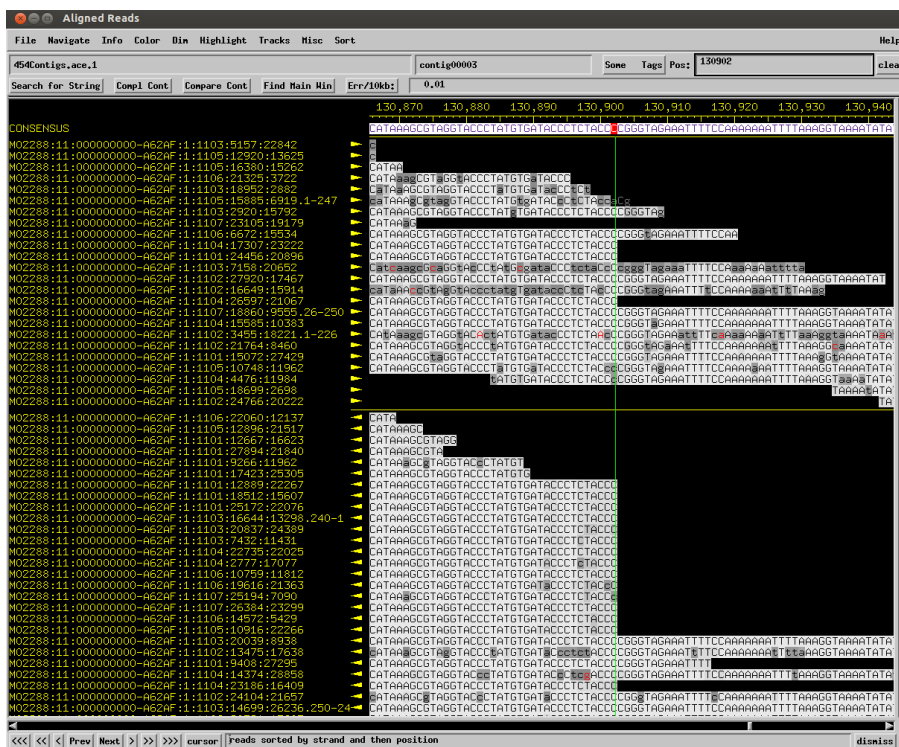


Figure 9. The wall on the bottom strand at base position 130902.

The only way to conclude that Waukesha92 exhibits circularly permuted ends is to examine it for both physically defined ends and terminal repeats. The result was no areas of high or low coverage that resulted in walls as seen in the Aligned Reads view. The Assembly View results of Waukesha92 are displayed in Figure 10.

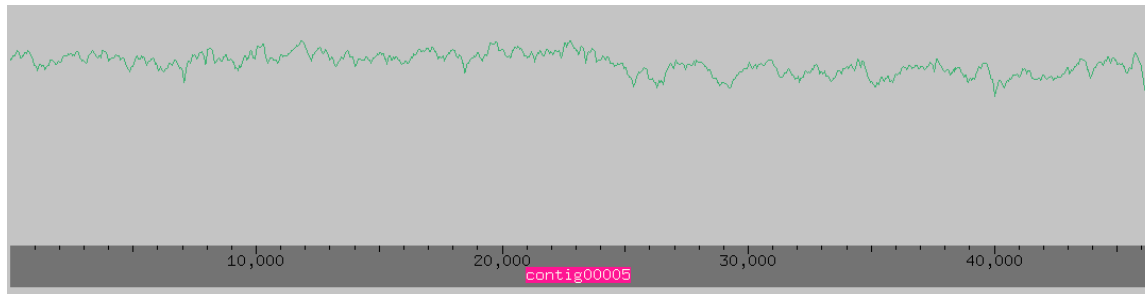


Figure 10. *Assembly View of Waukesha92 with no areas of high or low depth of coverage yielding walls in the Aligned Reads view.*

PILOT TEST AND SECOND SURVEY TO GATHER FEEDBACK

The second survey to gather feedback from the pilot test was answered by professors in the SEAPHAGES community as well as students in Dr. Louise Temple's genomics class. Of the professors and students who participated in the pilot test, the average rating for the manual addressing the basic needs of the user was 4.1/5. The most common response regarding the satisfaction with the manual's length was that the level of detail and the length are sufficient, which was the response of 73% of respondents. The processing, assembly, and assembly analysis sections of the manual received satisfaction scores averaging 4.4/5, 4.2/5, and 4.3/5, respectively. When asked which questions were in need of the most improvement, the section receiving the most votes at 36% was determining which contig to view within Consed. Other sections that

were selected by over 20% of respondents included visual analysis of read quality, coverage and weak areas, and finding the ends of the genome. Conversely, the section explaining the different views in Consed received no votes for this question. Lastly, when asked for a brief summary on specific improvements that should be made, the responses focused on providing a better explanation on how to find ends of the genome, providing more screenshots of the procedures, and providing more explanation of the dataflow specific to the files in the Linux environment. The survey results in their entirety can be found in Appendix 5.

FINALIZED MANUAL

The feedback gathered from the second survey was incorporated into the manual in order to improve its content and increase its effectiveness. The most noted concerns were focused on using the Linux terminal. Since the terminal was referred to as the generic term command line in the first version of the manual, the Linux-specific term terminal was added to the new version. Also, despite the link towards the end of the manual, there was still a desire for a more direct explanation of the basic commands for navigating the terminal. To address this, explanations of the `cd`, `ls`, and `pwd` commands were included in the section Opening Consed.

There were multiple clarity issues that needed to be addressed regarding Consed. One section that was poorly explained in the first version of the manual was how to determine which contig to view. It was initially stated to simply select the contig with the most reads assembled; however, this is not always true. This was corrected to explain the correct methodology of either predetermining the size from a restriction digest or performing a BLASTn on the exported FASTA files to compare them to other bacteriophages. There was a desire for more clarification

when discussing viewing high/low depth regions in the Assembly View window, so zooming in and out on the assembly was explained further. There was also a desire for an explanation about why each end looks the way that it does in Assembly View. To address this concern, bacteriophage DNA packaging was explained for each type of genome end. Deciding on which read depth to enter into the high/low depth of coverage window was another source of confusion for some. Clarification was added to this section to better explain how to determine the range of read depths that would be considered high or low for a given assembly and how to locate the high/low regions.

One final aspect that multiple respondents noted was the lack of an explanation of how to get from the compressed file containing the FASTQ file to the assembly. To address this, an explanation of how to unzip files and open the assembly GUI was added.

DISCUSSION

The creation of this manual will serve as a fundamental tutorial for those in the SEA-PHAGES program who choose to introduce it to their students. From the results of the first survey, it was determined that there was a clear interest in the creation of this manual (93% interest). It was also determined that most professors are currently working with students to process genome data. This concurs with the scope of this project and the SEA-PHAGES program to promote undergraduate research.

The results of the Karezi assembly experiment indicate that downsampling is an effective solution for minimizing the time required to assemble bacteriophage genomes while still maintaining data integrity. Each of the four different sized downsampled FASTQ files were able to assemble much faster than the full FASTQ file, thus indicating this as a more time efficient option. The BLASTn results of the 20084 bp contig were all identical and the Clustal Omega multiple sequence analysis indicated a percent identity matrix of 100% for each sequence. These results indicate that each successively downsampled FASTQ file was able to retain the data integrity of the original file. Depending on both the sequencing protocol used to create the FASTQ file and nature of the genome itself, 50000 reads may not be the lowest downsampled size to be able to retain the same level of quality (20). However, the results indicate that downsampling to a greater number of reads is still a significantly faster option than assembling the entire FASTQ file.

When viewed in the entire scope of phagehunting, the multiple hours that it takes to assemble the full FASTQ file may appear insignificant. However, it is important to minimize the time that it takes to complete this process in order to free up memory necessary to perform other tasks without the assembly program using up a significant portion of the computing power (6). While this issue is not as significant for those running the assembly on a host Linux machine, it is a significant hindrance for those working on a virtual machine. It is also important to expedite the assembly process in case the FASTQ data does not assemble the correct sequence. This allows the researcher to troubleshoot the issue much quicker than they would be able to otherwise. Since many professors use the pre-built virtual machine provided by the SEA-PHAGES program, downsampling to speed up the assembly process and computing performance is essential.

The results of the experiment for locating the ends of a genome indicate that using the method of high or low depth of coverage regions is an effective method for finding the ends of a genome. The Assembly View of OrionPax indicated many regions where there is a significant decrease in read depth, but no prolonged high depth of coverage region. This likely eliminates the possibility of terminal repeats. From analyzing all of these regions in the Aligned Reads view, a read buildup on both strands of the genome appeared at positions 198 and 206, indicating that there is an eight base gap. Thus, the presence of defined ends with a 3' overhang is validated for this bacteriophage genome. This suggests that the bacteriophage DNA is always packaged with the same start and end points with no short overlap between them.

When analyzing Pinkman, the Assembly View displayed a prolonged high depth of coverage region, which was analyzed for the possibility of a terminal repeat. The Aligned Reads view at the start of the high depth of coverage region indicated a buildup of reads on the top strand at position 128903 and a buildup of reads on the bottom strand at position 130902. This suggests that there is a significant overlap between the reads, much larger than that of a physical end with a 5' overhang. Thus, this genome is validated as exhibiting terminal repeats.

Waukesha92 exhibited no prolonged regions of high depth of coverage in Assembly View, thus low depth of coverage regions were analyzed. Since no buildup of reads was found on either strand, the entire genome was viewed using a base-by-base approach in the Aligned Reads view to ensure that there are no positions exhibiting a buildup of reads on either strand. None were found; thus, Waukesha92 was validated as having a circularly permuted genome.

The results of the second survey indicated that the most satisfactory aspect of the manual was the section discussing the different views within Consed. The least satisfactory sections were those pertaining to assembly analysis. Conversely, the concepts and protocols behind assembly analysis, mainly the process of analyzing high and low depth of coverage regions to find the ends of the genome, are the most complicated and thus may have been the most difficult concepts to convey in the manual. The second survey further highlighted the difficulties that many professors and students face operating within a Linux environment to use these programs. Although there was a hyperlink in the manual to a tutorial that introduces a user to navigating the Linux Terminal, it was placed towards the end of the manual (24). To address this issue, an explanation of the basic Linux Terminal commands were included in the steps so the user can understand the purpose of the commands they are entering. The intentions of this are to expose the tutorial to more users before any confusion arises later on in the manual.

CONCLUSIONS

The primary goals of this project were to illuminate the black box of DNA assembly and assembly analysis and to promote undergraduate research in this field. Based on the feedback from the second survey, it was determined that the goal of illuminating the black box has been achieved, at least in part, for those who chose to participate in the pilot test. The full ramifications of this manual will not be known for a few years as more and more faculty members within the SEA-PHAGES community adopt the manual. Only then will the effectiveness of this manual be truly known. As for the end goal of promoting undergraduate research, it is anticipated that those professors who are currently not utilizing undergraduate

researchers to perform genomic data processing will begin to do so after implementing this manual. This in turn could lead to more undergraduate students becoming interested in the fields of genomics and bioinformatics. This effect was noticed in students in Dr. Louise Temple's genomics course who showed enthusiasm towards the course while using this manual to help troubleshoot issues that arose in their work. It is anticipated that more professors will work with undergraduates after implementing this manual because it provides a straightforward, step-by-step approach that allows for a standardized methodology of DNA assembly and assembly analysis. The expectation is that this manual will serve as both an introduction to these processes for both professors and their students.

FUTURE WORK

It is not possible to fully meet the goals expressed in this thesis if work were stopped at the creation of this manual. If the end goal of this project were simply to create this manual, then it would become obsolete as soon as Newbler and Consed cease to be the primary software used by the SEA-PHAGES program. This manual should be continually updated in the future to account for the changes in software usage. Since not all universities in the SEA-PHAGES program use these two programs, a future objective could be to test out other programs that other schools are using and include them in an expanded version of the manual. This would allow for more professors to see the value in the manual. The testing of other programs should not be limited to those popular among others professors in the SEA-PHAGES program. There are many programs available that should be analyzed to determine whether they could better meet the needs of those in the SEA-PHAGES community. Any program that could potentially be more effective, faster,

or run in a variety of operating systems would be a great candidate for testing and possible inclusion in future editions of the manual.

Similarly to continually updating the manual, it was noticed in the first survey that although a majority of professors indicated that they preferred a digital user manual over a wiki-style webpage, many expressed that they would actually be open to either or even both. A future project could be to create a wiki page that could be contributed to by any professor in the SEA-PHAGES community. This process could potentially allow for professors to customize the structure of the manual in a way that caters to multiple levels of detail desired by different professors. This may be the best possible way to introduce the manual to the greatest number of professors in the program in hopes of them adopting it.

In addition to these long-term possible goals, there is one immediate goal that will be addressed by the end of this semester. The final version of the manual will be made available online for free download. Not only will the manual be accessible for those within the SEA-PHAGES consortium, but it will also be available for anyone who wishes to use it.

REFERENCES

1. Feero, W. G., and A. E. Guttmacher. 2014. Perspectives: Genomics, Personalized Medicine, and Pediatrics. *Academic Pediatrics*. 14:14-22. doi: 10.1016/j.acap.2013.06.008. <http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=edselp&AN=S187628591300199X&site=eds-live&scope=site&authtype=ip,uid>.
2. Church, G. M., and E. Regis. 2012. *Regenesis*. [electronic resource] : how synthetic biology will reinvent nature and ourselves. New York : Basic Books, c2012. <http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=cat00024a&AN=vmc.b27607033&site=eds-live&scope=site&authtype=ip,uid>; <http://www.lib.jmu.edu/resources/elog.aspx?http://proquestcombo.safaribooksonline.com/?uiCode=jmadison&xmlId=9780465033294>.
3. Mardis, E., J. McPherson, R. Martienssen, R. K. Wilson, and W. R. McCombie. 2002. What is finished, and why does it matter. *Genome Res*. 12:669-671. <http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=cmedm&AN=11997333&site=eds-live&scope=site&authtype=ip,uid>.
4. Leland Taylor, D., A. Malcolm Campbell, and L. J. Heyer. 2013. Illuminating the Black Box of Genome Sequence Assembly: A Free Online Tool to Introduce Students to Bioinformatics. *The American Biology Teacher*. 572. doi: 10.1525/abt.2013.75.8.9. <http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=edsjsr&AN=edsjsr.abt.2013.75.8.9&site=eds-live&scope=site&authtype=ip,uid>.
5. Anonymous 2015. Science Education Alliance (SEA). 2015. <https://www.hhmi.org/programs/science-education-alliance>.
6. Gupta, D., S. Lee, M. Vrable, S. Savage, A. C. Snoeren, G. Varghese, G. M. Voelker, and A. Vahdat. 2010. Difference Engine: Harnessing Memory Redundancy in Virtual Machines. *Commun ACM*. 53:85-93. doi: 10.1145/1831407.1831429. <http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=bth&AN=55028308&site=eds-live&scope=site&authtype=ip,uid>.
7. Anonymous Next Generation Sequencing: An Overview. http://www.corning.com/uploadedFiles/Lifesciences/PDFs/Axygen_PDFs/NGS%20Overview.pdf.
8. Scheibye-Alsing, K., S. Hoffmann, A. Frankel, P. Jensen, P. F. Stadler, Y. Mang, N. Tommerup, M. J. Gilchrist, A. -. Nygård, S. Cirera, C. B. Jørgensen, M. Fredholm, and J. Gorodkin. 2009. Sequence assembly. *Computational Biology and Chemistry*. 33:121-136. doi: <http://dx.doi.org/10.1016/j.compbiolchem.2008.11.003>.
9. Stein, L. 2001. Genome annotation: from sequence to biology. *Nat. Rev. Genet*. 2:493-503. <http://dx.doi.org/10.1038/35080529>.

10. Cock, P. J. A., C. J. Fields, N. Goto, M. L. Heuer, and P. M. Rice. 2010. The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res.* 38:1767-1771. doi: 10.1093/nar/gkp1137.
<http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=cmedm&AN=20015970&site=eds-live&scope=site&authtype=ip,uid>.
11. Lindgreen, S. 2012. AdapterRemoval: easy cleaning of next-generation sequencing reads. *BMC Research Notes.* 5:337-343. doi: 10.1186/1756-0500-5-337.
<http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,cookie,url,cpid,uid&custid=s8863137&db=a9h&AN=85859036&site=eds-live&scope=site&authtype=ip,uid>.
12. Anonymous 2014. SurveyMonkey: Free online survey software & questionnaire tool. 2014. <https://surveymonkey.com>.
13. Anonymous 2014. Analysis Software. 2014. <http://www.454.com/products/analysis-software/>.
14. Gordon, D. 2014. Consed: A Finishing Package. 2014.
<http://bozeman.mbt.washington.edu/consed/consed.html>.
15. Russell, D. Assembling Phage Genomes with Newbler. 2014.
<http://phagesdb.org/workflow/videos/NEWBLER01/>.
16. Russell, D. Consed & Finishing #3. 2014.
<http://phagesdb.org/workflow/videos/consed03/>.
17. Russell, D. Consed and Finishing #4. 2014.
<http://phagesdb.org/workflow/videos/consed04/>.
18. Russell, D. Consed & Finishing #6. 2014.
<http://phagesdb.org/workflow/videos/consed06/>.
19. Russell, D. Consed & Finishing #7. 2014.
<http://phagesdb.org/workflow/videos/consed07/>.
20. Russell, D. Selecting a Subset of Reads From an SFF or FASTQ File. 2014.
<http://phagesdb.org/workflow/videos/NEWBLER02/>.
21. MacKenzie, D., and J. Meyering. 2010. head(1) - Linux man page. 2014.
<http://linux.die.net/man/1/head>.
22. Russell, D. 2014. 3' versus 5' overhangs in sequencing data. 2014.
<http://phagesdb.org/blog/posts/25/>.
23. Anonymous Finishing Phage Genomes. 2014:September 5, 2014.
phagesdb.org/media/docs/Finishing_Genomes.pptx.

24. Cobbaut, P. 2014. Linux Fundamentals: Chapter 6. working with directories. 2014.
<http://linux-training.be/files/books/html/fun/ch08.html#idp5335088>.

APPENDICES

Appendix 1. Questions from the first survey sent out to gauge user interest in the manual.

1. Are you currently performing any processing of genome data that precede annotation steps?

2. If you answered "Yes" to question 1, what program(s) are you currently using for DNA sequence assembly? Are you satisfied with these program(s) or are you considering other program(s)? If you answered "No" to question 1, please proceed to question 5.

3. If you answered "Yes" to question 1, what program(s) are you currently using to analyze the assemblies for sequence quality, genome end definition, and other "finishing processes"? Are you satisfied with these program(s) or are you considering other program(s)?

4. If you answered "Yes" to question 1, what are the biggest obstacles you face in the sequence assembly and finishing stages that are not being adequately addressed in current how-to guides, user manuals, and tutorials? Please proceed to question 7.

5. If you answered "No" to question 1, are you interested in performing pre-annotation processing of genome data?

6. If you answered "No" to question 1, what obstacles (if any) exist to you performing pre-annotation processing of genome data?

7. If a how-to guide were to be made to cover the sequence assembly and finishing processes, would you utilize it?

8. If you answered "Yes" to question 7, in what medium would you most prefer it to be displayed?

Other (please specify)

9. Do you work with students to assemble, finish, and/or annotate non-mycobacteriophage genomic DNA?

If so, please list the phages or organism(s) sequenced.

10. Optional

What is your affiliated institution?

How many years have you been participating in the SEA-PHAGES program?

Appendix 2. Questions from the second survey to gather feedback on the manual from the pilot test.

***1. Overall, what rating would you give this manual for addressing the basic needs of you and your students? Choose 1 for poorly addressed our needs and 5 for perfectly addressed our needs.**

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

***2. Are you satisfied with the level of detail provided by this manual, with respect to its length?**

- ☐ Yes, the level of detail and length are sufficient
- ☐ Yes, however the manual is too long and/or too detailed
- ☐ Yes, however the manual is too short and/or not detailed enough
- ☐ No, the manual is not detailed enough to meet our needs
- ☐ No, not enough basic aspects of these processes are covered to meet our needs

If not enough aspects of these processes are covered, then what is missing?

***3. Rate your satisfaction with the following section: Processing. Choose 1 for completely unsatisfied and 5 for completely satisfied**

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

***4. Rate your satisfaction with the following section: Assembly. Choose 1 for completely unsatisfied and 5 for completely satisfied**

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

***5. Rate your satisfaction with the following section: Assembly Analysis. Choose 1 for completely unsatisfied and 5 for completely satisfied**

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

***6. In the section Assembly Analysis, which topics need the most improvement in their explanations? Select all that apply.**

- ☐ Opening Consed
- ☐ Determining Which Contig to View
- ☐ Consed Views
- ☐ Visual Analysis of Read Quality (Chromatogram)
- ☐ Coverage and Weak Areas
- ☐ Finding the Ends of the Genome
- ☐ Determine Phage Cluster
- ☐ Adding SFF or FASTQ Reads to an Existing Consed Project

7. Based on your response to the previous question, please provide a brief summary as to what specifically needs to be improved in these sections.

8. Optional information:

What is your affiliated institution?

How many years have you been participating in the SEA-PHAGES program?

Appendix 3. The user manual that was constructed as the basis for this report.

Addressing the Black Box Phenomenon of Genome Sequencing and Assembly:

Bacteriophage DNA Assembly Manual

By
Brandon M. Carter

under the faculty members guidance of
Dr. Louise Temple, PhD.

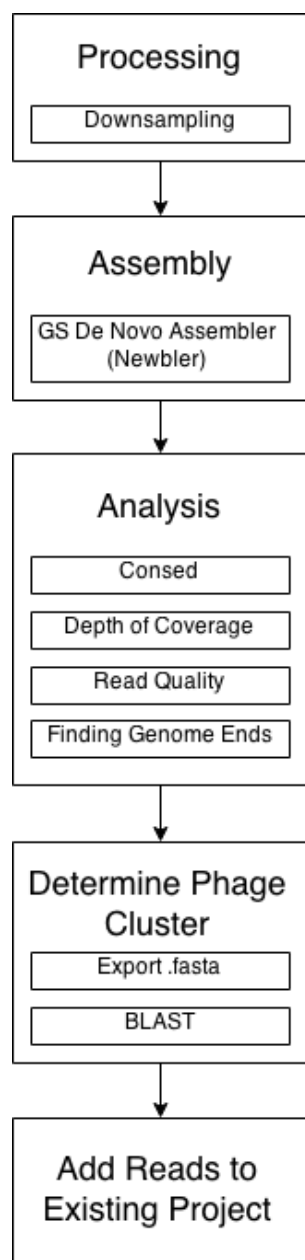
College of Integrated Science and Engineering
James Madison University

As the field of bioinformatics continues to progress there is increasing interest in the areas of DNA sequencing, assembly, and analysis. A lack of formal training in how to perform DNA assembly and assembly analysis has led to a black box phenomenon where the procedures connecting raw sequencing data and an assembled genome are not well understood.

The creation of this manual aims to ameliorate these issues by taking users from the raw data from a sequencing instrument to a finished assembly. The primary goal of this manual is to further the knowledge and understanding of DNA assembly software in hopes of increasing the breadth of undergraduate research in universities across the nation.

Contents

Workflow	46
Downloading SEA Virtual Machine	47
Software Overview	47
Steps of Processing, Assembly, and Analysis	47
Processing	47
Selecting a Subset of Reads from a FASTQ file	47
Assembly	48
GS De novo Assembler (Newbler) ^[6]	48
Assembly Analysis	50
Opening Consed	50
Determining Which Contig to View	51
Consed Views ^[8]	52
Visual Analysis of Read Quality (Chromatogram)	54
Coverage and Weak Areas	55
Finding the Ends of the Genome	57
Determine Phage Cluster ^[12]	64
Adding SFF or FASTQ reads to an existing Consed project ^[13]	65
Newbler from the Command Line	66
Navigating the Command Line in Linux for Beginners	66
Example Phage Data	66
References	67



Phagehunting consists of both wet and dry lab work that begins with isolating a bacteriophage from the environment and ends with annotating the bacteriophage genome and comparing it to other known bacteriophages. After isolating a bacteriophage from the environment, the phage sample must be purified and then amplified to obtain a larger quantity of the phage. The bacteriophage DNA is then extracted for use in DNA sequencing. Next the phage is characterized via a restriction digest to compare the bacteriophage in question to other known bacteriophages/clusters. The previously extracted DNA is then sequenced in order to obtain the DNA sequence of the bacteriophage. Once sequenced, the DNA is assembled into a contiguous sequence created by assembling overlapping clones representing regions of the genome, or a contig.^[1] This contig is then analyzed using computer programs that can assist one in “finishing” the genome by determining many aspects of the genome, such as where the ends of the genome are and what type of ends it exhibits. After “finishing”, the final step of phagehunting is to annotate the genome using computer programs to determine characteristics of the genome such as the location of the genes and their functions.^[2] The scope of this manual is limited to the assembly of the DNA sequence and the subsequent analysis of the contig prior to annotation.

Following DNA sequencing, the data must be processed by decreasing the number of reads that will be included in the assembly. This is done in order to speed up the assembly process because there are far more reads in the sequence file than are necessary for a sufficiently accurate assembly. Next, the DNA is assembled and then analyzed to examine the depth of coverage, read quality, determine the genome ends, etc. After analyzing the assembly data, the phage cluster can be determined by performing a BLAST query with the .fasta file. Lastly, the reads that were removed from the sequence file earlier can be recombined with those reads that were used for the original assembly in order to help resolve coverage and quality issues in the consensus sequence.

Figure 1 – Workflow of the steps that will be covered in this manual.

Downloading SEA Virtual Machine

The first step before assembling genomes and analyzing these assemblies is to download the programs necessary to perform these tasks. Follow this link to download and install the [SEA Virtual Machine](#), which contains all of the programs you will need for genome assembly and analysis. ^[3]

Software Overview

In this manual, the functionalities of two computer programs related to DNA processing, assembly, and analysis will be explained. The first of which is GS *De novo* Assembler, more commonly known as Newbler, a proprietary software package released by Roche that performs *de novo* sequence assembly. Newbler accepts .sff files (listed as GS reads in the program), .fastq files, and .fasta files. Newbler works by first identifying pairwise overlaps between reads and then constructing multiple alignments of overlapping reads. It then breaks the multiple alignments where consistent differences are found between the different sets of reads, which in turn produces the contigs that represent the assembled reads. It then generates isotigs by resolving branching structures between contigs. Lastly, it generates consensus base calling for the contigs using quality and flow signal information at each base in the multiple alignments. At the end of the assembly process, Newbler will output the contig consensus sequences, quality scores, alignment, and metric files for viewing with another program. ^[4]

The output of a Newbler assembly is then input into Consed for examination. Consed is a computer program that is used for viewing, editing, and finishing DNA sequence assemblies. Consed is a very powerful tool that assists in examining characteristics of the assembled genome. These include identify low coverage areas in the genome, identify the ends of the genome, examine multiple contigs, determine if there is low consensus quality, identify discrepant positions, and selecting primers just to name a few.

Steps of Processing, Assembly, and Analysis

Processing

Selecting a Subset of Reads from a FASTQ file

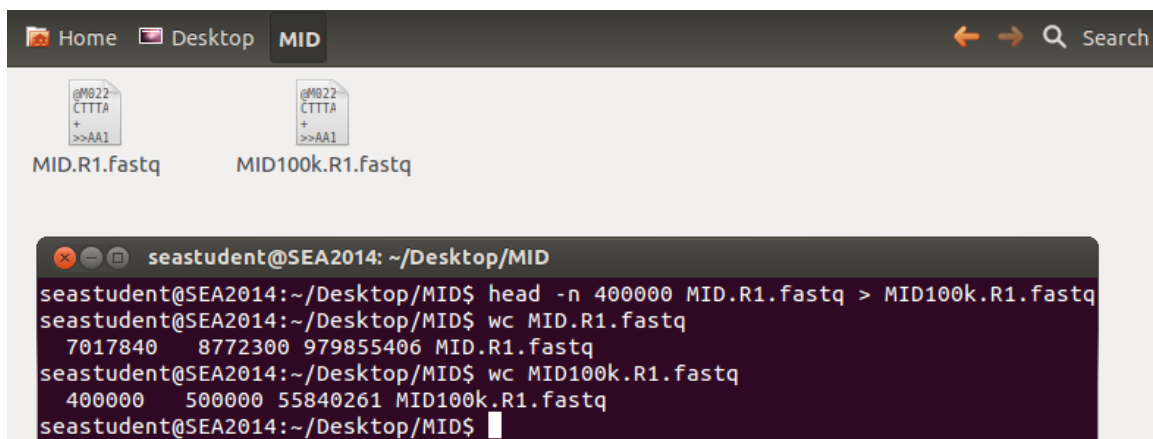
When assembling phage genomes, sequencing technologies often generate too many reads to allow for a high quality assembly for a single genome. Assembly programs tend to have an innate flaw where they break up the genome into multiple contigs when a high number of reads are present. One of the first steps that you should take prior to assembling your genome is to take a subset of sequencing reads from your .fastq file. This will allow you to condense the amount of data to generate one contig (hopefully). If you would like to convert your .sff file into .fastq file format, please see [Converting a SFF File to a FASTQ File](#). If you would like to select a subset of reads from a .sff file, see Dan Russell's tutorial on [selecting a subset of reads from a .sff or .fastq file](#). It is important to note that sometimes assembling a file that was converted from a .sff file to a .fastq file can lead to worse results due to the additional signal information contained in the original .sff file.

*Note – In order to unzip a compressed file, right click on the file and click “extract here”.

*Note – all of these steps will be performed on the command line (Terminal) in the virtual machine within the same file directory that the original .fastq file is located. It is a good rule of thumb to start downsampling to 50,000 reads for data from an Ion Torrent output or 20,000 to 30,000 reads for data from a 454 output. [5]

FASTQ Downsampling [5]

1. Generate a list of read names of a given file.
 - a. Downsampling of .fastq files uses a similar command as the first part of downsampling .sff files, except the number of reads is multiplied by four. This is because a .sff file is written in binary, whereas a .fastq file is formatted for four lines of information per read. For example, in order to downsize to 100,000 reads for a .fastq file the command would be as follows:
“head -n 400000 MID.R1.fastq > MID100k.R1.fastq”



The screenshot shows a file manager window with two files: MID.R1.fastq and MID100k.R1.fastq. Below the files is a terminal window with the following commands and output:

```
seastudent@SEA2014: ~/Desktop/MID
seastudent@SEA2014:~/Desktop/MID$ head -n 400000 MID.R1.fastq > MID100k.R1.fastq
seastudent@SEA2014:~/Desktop/MID$ wc MID.R1.fastq
7017840  8772300 979855406 MID.R1.fastq
seastudent@SEA2014:~/Desktop/MID$ wc MID100k.R1.fastq
400000    500000 55840261 MID100k.R1.fastq
seastudent@SEA2014:~/Desktop/MID$
```

Figure 2 – FASTQ Downsampling. The command “wc” (word count) is used to illustrate how the file was cut down to 400,000 lines, which is 100,000 reads in a .fastq file.

Assembly

GS De novo Assembler (Newbler) [6]

Newbler can be run from either the command line or from a GUI entitled GS *De novo* Assembler. The following instructions are for running Newbler from the GUI. If you would like to view Newbler commands to run the assembly from the command prompt refer to the glossary of [Commands for Newbler](#).

1. Launch the Newbler GUI by opening the program titled GS *De novo* Assembler.
2. Click on “New Assembly Project”. Name the new assembly project and choose a location to save it to. Save it as a genomic project.

3. Under the “Project” tab, continue under “GS Reads” tab if inputting .sff reads or continue under “FASTA and FASTQ Reads” tab if inputting .fastq or .fasta reads.
4. Press the “+” symbol on the left to input the .sff, .fastq, or .fasta file.
5. At the “Set GS Read Data Attributes” window, choose “auto detect”.
 - a. If barcodes have been separated already, do not select the multiplex filtering option on the bottom of the window.
6. Under the “parameters” tab, the default option will work well for phage genomes. The only additional parameter to select is under the “Output” tab, change Ace Format setting to “Complete Consed folder” to generate a complete Consed folder for analysis.
7. Select the “Start” button on the right hand side of the screen. In Newbler, multiple assemblies may be run at once.
8. Once the assembly is complete, go to the “Alignment Results” tab to view the list of contig(s) that the reads were assembled into.
9. If your file assembled into one contig, then simply save your file and the assembly process is complete. If your file assembled into multiple contigs, then continue to the next step.
10. If your file assembled into multiple contigs, go to the “Result files” tab and view the 454NewblerMetrics.txt file to see information such as the number of reads, number of bases in those reads, percentage assembled, error, coverage, etc. to see where something may have gone wrong with the assembly.

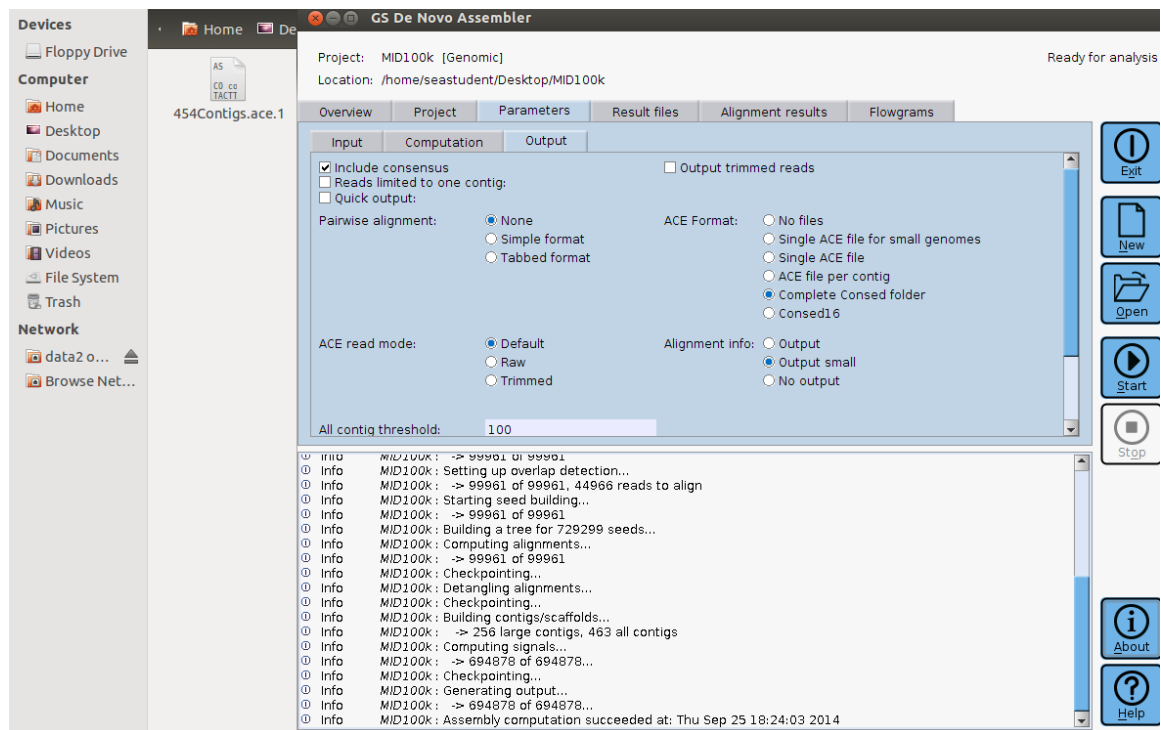


Figure 3 – Newbler output with .ace file within the edit_dir directory within the complete Consed folder created from the assembly.

Assembly Analysis

Opening Consed

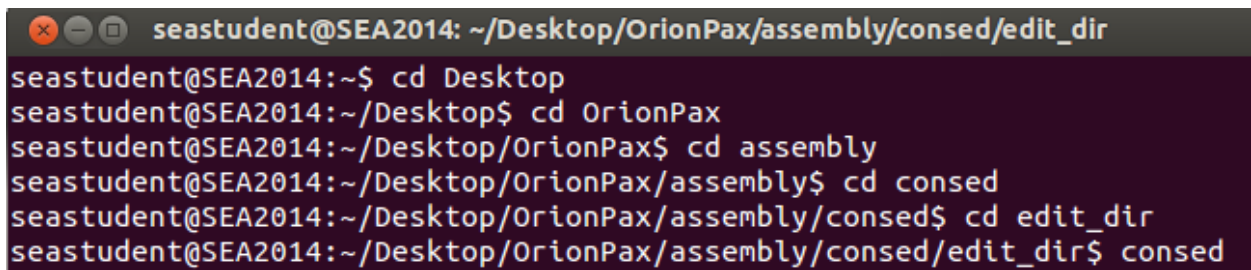
Analyzing your genome is an important step prior to annotation. This process will allow you to do identify low coverage areas in the genome, identify the ends of the genome, examine multiple contigs, determine if there is low consensus quality, identify discrepant positions, etc. The first step in analyzing your genome is to open the Consed program and begin analyzing the assembled genome. Depending on the type of sequencing performed, the consed output will have different folders. For Sanger sequencing, there will be a chromate_dir folder with chromatogram files, an edit_dir folder with the assembly files, and a phd_dir folder with phred files. A 454 assembly and an Illumina assembly will also contain these three folders. In addition to these folders, 454 and Illumina Assemblies will contain either a sff_dir folder or a solexa_dir folder, respectively, which contain raw sequence output. Consed projects are stored within the edit_dir folder of whatever directory the Consed output was saved to during the Newbler Assembly. [7]

1. Open Consed from the command line.
 - a. Locate the directory (folder) where the complete Consed folder was saved to during the Newbler assembly.
 - b. Run the Command “cd (folder that contains the output of the Newbler assembly)”. The cd command stands for change directory. This command is

used to navigate in and out of different folders. The command “cd” navigates forward one folder and the command “cd ..” will navigate back one folder.

- c. Run the command “cd assembly” to open the assembly folder.
- d. Run the command “cd consed” to open the consed folder.
- e. Run the command “cd edit_dir” to open the edit_dir folder.
- f. Run the command “consed” to open the Consed program.
- g. Once the consed program has loaded, double click on the .ace file you wish to view.

*Note - If at any point you are navigating the Terminal and are unsure what folder to open next, you can run the command “ls” by itself. This will list all of the folders and files within the current folder. If you are unsure what folder you are currently in, you can run the “pwd” command to show the present working directory.

A terminal window with a dark background and light-colored text. The title bar shows window control buttons and the text 'seastudent@SEA2014: ~/Desktop/OrionPax/assembly/consed/edit_dir'. The terminal content shows a series of 'cd' commands being executed to navigate through the directory structure: Desktop, OrionPax, assembly, consed, and finally edit_dir, where the 'consed' command is entered at the end.

```
seastudent@SEA2014: ~/Desktop/OrionPax/assembly/consed/edit_dir
seastudent@SEA2014:~$ cd Desktop
seastudent@SEA2014:~/Desktop$ cd OrionPax
seastudent@SEA2014:~/Desktop/OrionPax$ cd assembly
seastudent@SEA2014:~/Desktop/OrionPax/assembly$ cd consed
seastudent@SEA2014:~/Desktop/OrionPax/assembly/consed$ cd edit_dir
seastudent@SEA2014:~/Desktop/OrionPax/assembly/consed/edit_dir$ consed
```

Figure 4 – Opening Consed from the Command Line. In this scenario, the complete consed folder was saved to the Desktop during the assembly in Newbler.

Determining Which Contig to View

Once the .ace file has been opened in the Consed Main Window, you may notice that multiple contigs have been assembled. This is a common occurrence due to the way in which Newbler assembles genomes. In order to determine the correct contig of your assembled genome, look at the top few contigs with the highest number of reads that were used in the assembly. Note that these may not be the first few contigs listed. In order to sort the contigs by the number of reads click on the button on the upper right of the contig list “Reorder By # of Reads”. If you already know how long your bacteriophage genome should be, click on the corresponding Contig. If you do not, you will have to export .fasta files from each of the contigs with the highest number of reads assembled. Then you will have to BLAST each .fasta file to identify which contig represents your genome. Directions for how to export a .fasta file can be found in the section [Determine Phage Cluster](#).

Double click on the contig to open it up in the Aligned Reads window.

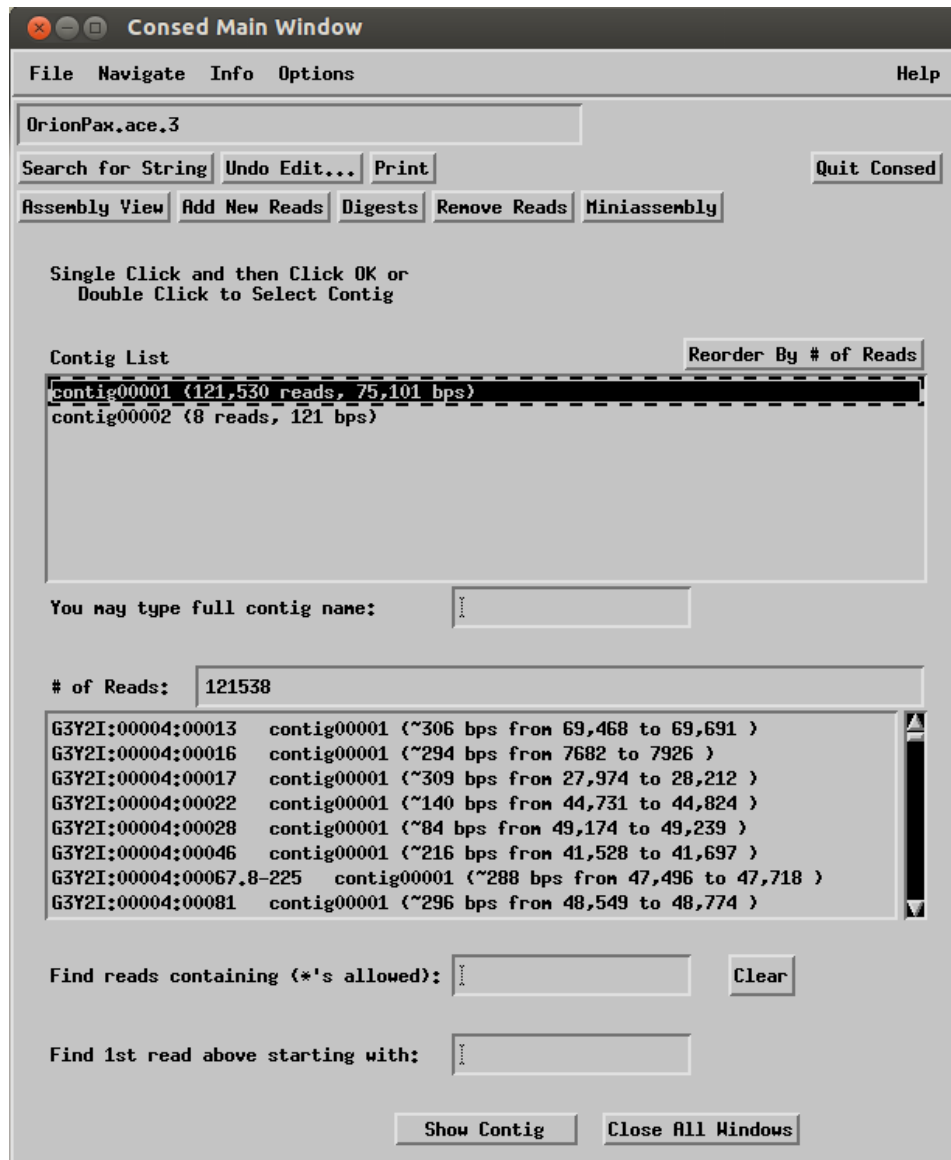


Figure 5 – Consed Main window.

Consed Views ^[8]

Once Consed is open, we need to get a feel for the different main views that it offers. These include the Main window where the contigs are listed for the given .ace file, Assembly View that illustrates a graphical output of the contig, the Aligned Reads window which shows the sequence of the consensus sequence and individual reads, and the Trace Window that provides a visual display of the read quality.

Assembly View illustrates a big picture overview of the project including a graphical output of the contig(s), the number of bases in the contig(s), clone coverage (light green line), and sequence coverage (dark green line) on the genome. To open the Assembly View, from the Main Window, single click on the contig you want to view and single click on the “Assembly View” button. If contigs are being excluded from the Assembly View click on the “What to Show” button at the bottom of the

screen, then click on the “In/Exclude contigs” parameter, and finally change the default setting of “exclude contig if depth coverage greater than” to a much higher number (around 500 should work). To show sequence matches to indicate areas of similarity click “what to show, then click “sequence matches”, and click “run cross match”.

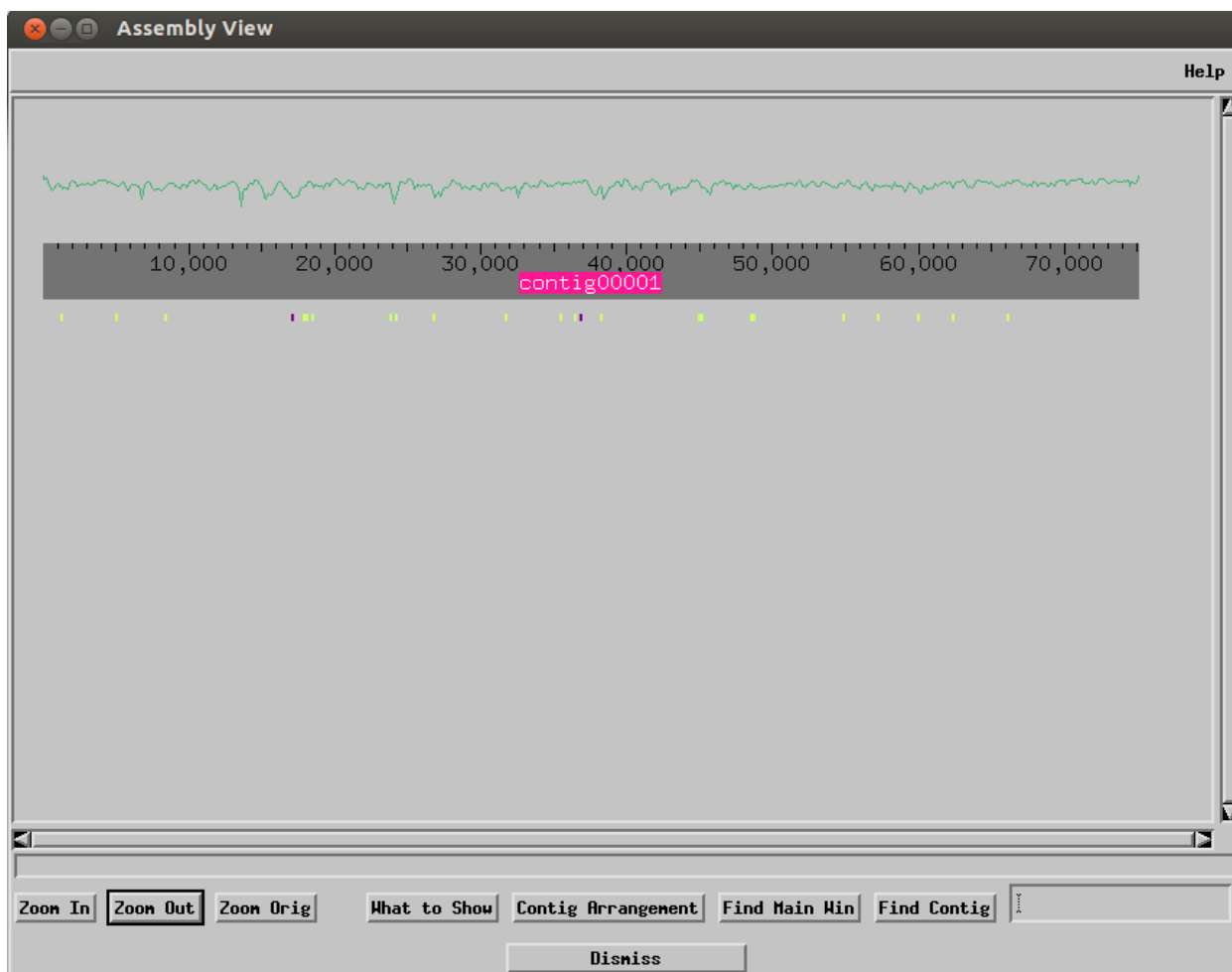


Figure 6 – Consed Assembly View window. If there are many contigs in your assembly, you may need to specify the specific contig you want to view. In order to find the desired contig search the contig number in the bottom right hand corner in the “Find Contig” field.

The Aligned Reads window is a zoomed in version of the assembly that illustrates the quality information of the reads that generated the consensus sequence. It is opened by double clicking on a contig. You can also go straight to a particular location in the aligned reads window from the Assembly View by right clicking on the base location in the grey area below the read and clicking “Goto Aligned Reads Window”. The top row is the consensus sequence and the rows are all of the reads from the project. The top reads above the yellow line were read from left to right whereas the bottom reads below the yellow line were read right to left. The quality of the reads are indicated by the color of the background of the letter and the capitalization of the letter. A white background and capital letter indicate a higher quality sequence. In addition to lower case letters, darker shades of gray indicate a lower sequence quality. A black background indicates an unaligned region where there is extremely poor quality. In order to see the quality score of a particular base in a read or in the consensus sequence, click on the base and the quality score is indicated on the bottom of the

screen. Within each read there may be asterisks. These asterisks act as spacers where an extra base may have been called in a different sequence. If the color of the base is red, then it disagrees with the consensus sequence. This is not generally an issue unless there are a long string of red bases in a region on multiple reads, which may indicate poor quality.

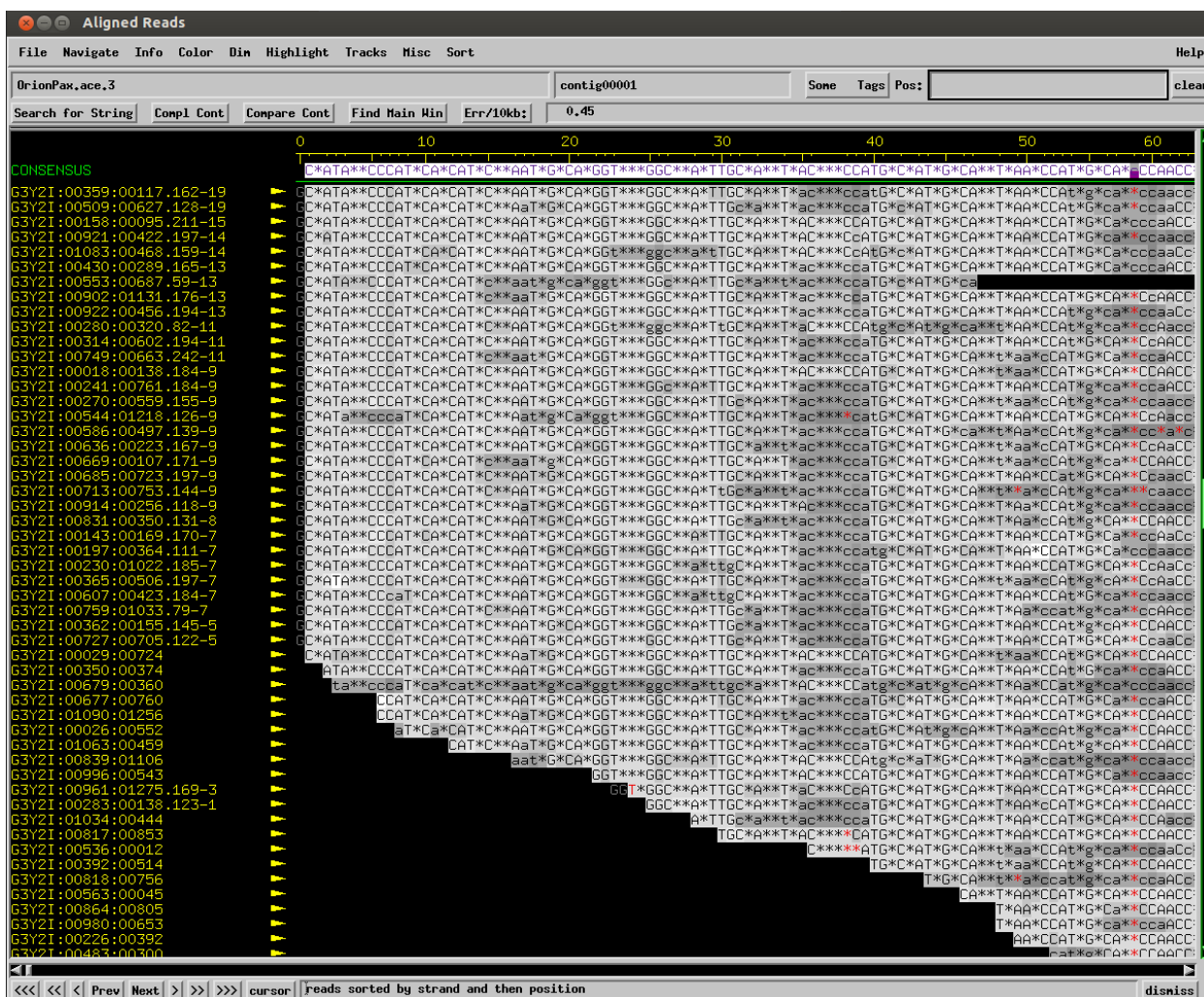


Figure 7 – Consed Aligned Reads Window.

Visual Analysis of Read Quality (Chromatogram)

To visually analyze the quality of a read, you can open the Trace Window to view the chromatogram by middle clicking on the read (clicking on the scroll wheel of the mouse). If you receive an error message from this action, you may need to create a `sff_dir` folder within the `consed` folder and move the `.sff` file there. It is important to note that the 0 point of the consensus sequence is arbitrary since Newbler randomly decides where to begin/end the circular genome. The shorter the colored bar, the lower the quality of that base as compared to the consensus sequence.

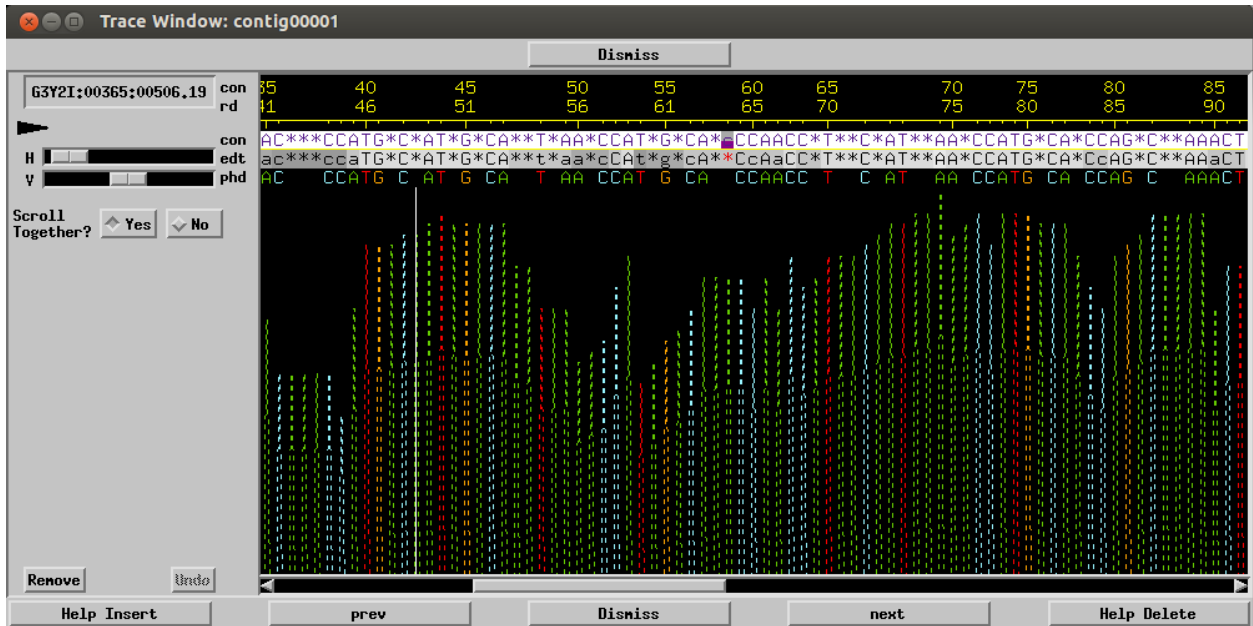


Figure 8 – Consed Trace window. Notice the red highlighted guanine is white and capitalized indicating high quality. The Trace Window verifies this since the orange bar that corresponds to it is one of the tallest bars on the screen.

Coverage and Weak Areas

In order to analyze coverage and find weak areas in the contig, open Assembly View. To view a low coverage area in the Aligned Reads window from the Assembly View window, right click on the dark grey line right under the area you want to view on the desired contig and click “GoTo Aligned Reads Window”. Under the aligned reads view click “Navigate” at the top of the window and then click “Low consensus quality (≤ 25 or 98)” to show areas where the base quality is below the threshold, i.e the sequence has low quality. To alter the threshold value from ≤ 25 or 98 go to the Consed Main Window and click “options” and then “General Preferences” where you can alter the threshold numbers to suit your needs. Increasing this value will make more areas appear to be low quality, whereas lowering this number will reduce the number of areas that appear to be low quality.

Low consensus quality (<=25 or 98)				
Contig Name	Read Name	Consensus Positions		
contig00001 (consensus)		58	base quality	below threshold
contig00001 (consensus)		148	base quality	below threshold
contig00001 (consensus)		191-192	base quality	below threshold
contig00001 (consensus)		271	base quality	below threshold
contig00001 (consensus)		541	base quality	below threshold
contig00001 (consensus)		578	base quality	below threshold
contig00001 (consensus)		1347	base quality	below threshold
contig00001 (consensus)		1734	base quality	below threshold
contig00001 (consensus)		3046	base quality	below threshold
contig00001 (consensus)		3593	base quality	below threshold
contig00001 (consensus)		5097	base quality	below threshold
contig00001 (consensus)		6370	base quality	below threshold
contig00001 (consensus)		6409	base quality	below threshold
contig00001 (consensus)		6822	base quality	below threshold

Figure 9 – Low consensus quality positions.

Another method of analyzing coverage in Consed is to analyze the high and low depth of coverage regions in the contig. From the Consed Main window click “Navigate” and then click “Search for High (or Low) Depth of Coverage Regions”. If the box is checked at the top then you will search for high depth of coverage regions and if it is unchecked then you will search for low depth of coverage regions. When searching for depth of coverage, you can adjust the “ignore read bases below this quality” parameter to “0” if you wish to not ignore any bases regardless of quality. [9]

Navigate by High (or Low) Depth of Coverage	
show high depth (not low depth)	<input type="checkbox"/>
ignore read bases below this quality	<input type="text" value="10"/>
min (for high depth regions) or max (for low depth regions) depth of coverage	<input type="text" value="300"/>
<input type="button" value="Search"/>	<input type="button" value="Dismiss"/>

Figure 10 – Parameters for Search for High (or Low) Depth of Coverage Regions. Notice that the box was not checked in the parameters so this will yield output for low depth of coverage regions.

Low Depth of Coverage Regions		
regions with at most 300 reads of at least quality 10 Regions combined if this close: 50		
contig00001 (consensus)	185-213	read depth 99-296
contig00001 (consensus)	464-987	read depth 169-300
contig00001 (consensus)	1048-1056	read depth 296-300
contig00001 (consensus)	1223-1905	read depth 164-300
contig00001 (consensus)	2354-2543	read depth 247-300
contig00001 (consensus)	2590-2722	read depth 170-300
contig00001 (consensus)	3341	read depth 300-300
contig00001 (consensus)	3378-3395	read depth 293-300

Figure 11 – Output from Search for High (or Low) Depth of Coverage Regions.

In Consed it is possible to determine whether the gene has a circularly permuted genome, defined ends, or a subset of defined ends known as terminal repeats. A genome with defined ends will exhibit a build-up of clones, known as a “wall”, on both strands where many reads start at the same location in the genome. With defined ends, there is likely to be a dramatic decrease in the depth of coverage, as seen in Assembly View. The reason that a genome with defined ends will occur at a region with a dramatic decrease in read coverage is because the bacteriophage DNA is always packaged with the same start and end positions, resulting in a buildup of reads at the start and end positions. It is important to note that the ends of the genome are likely neither at the 0 position nor at the beginning or end of the consensus sequence because the start locations are ambiguously portrayed in Consed. Some genomes with defined ends may also have a short 3’ or 5’ base pair overhang, aka sticky ends. Whether or not a phage exhibits one of these overhangs can be determined in the Aligned Reads window. A 3’ overhang will appear as a buildup of reads at each end of the genome that stops just before the overhang, resulting in a gap between the walls in most reads; however, the overhang may be present in some reads due to ligation of the ends or concatamers in the DNA sample. A 5’ overhang will appear as a buildup of reads at each end of the genome that includes the complete overhang sequence, so the overhang is present in most of the reads. ^[10]

A subset of defined ends is known as terminal repeats. In this scenario, you should see a “wall” of reads on both strands and a dramatic increase in depth of coverage; however, the “wall” will not be at the end of the genome and only some of the reads will exhibit a buildup. Genomes with terminal repeats will exhibit a prolonged region of high read coverage because the genome is always packaged at the same start and end position and there is identical information on each end of the genome. The high depth of coverage region appears at this region with identical information.

In a circularly permuted genome, no “wall” will be visible and there may not be a dramatic change in depth of coverage. This is because the DNA is variably cut with a “head full” of DNA. The head of a bacteriophage can hold more than one full copy of the genome, and a genome that is circularly permuted will package the DNA so that it fills the head. This way, there is always slightly more than one full copy of the genome package. Since the DNA is cut variably each time, the ends of this type of genome cannot be determined in Consed. ^[11]

For more information regarding finding genome ends and the utilization of primer walks to fill in gaps, please see the following reference on [finishing genomes](#).

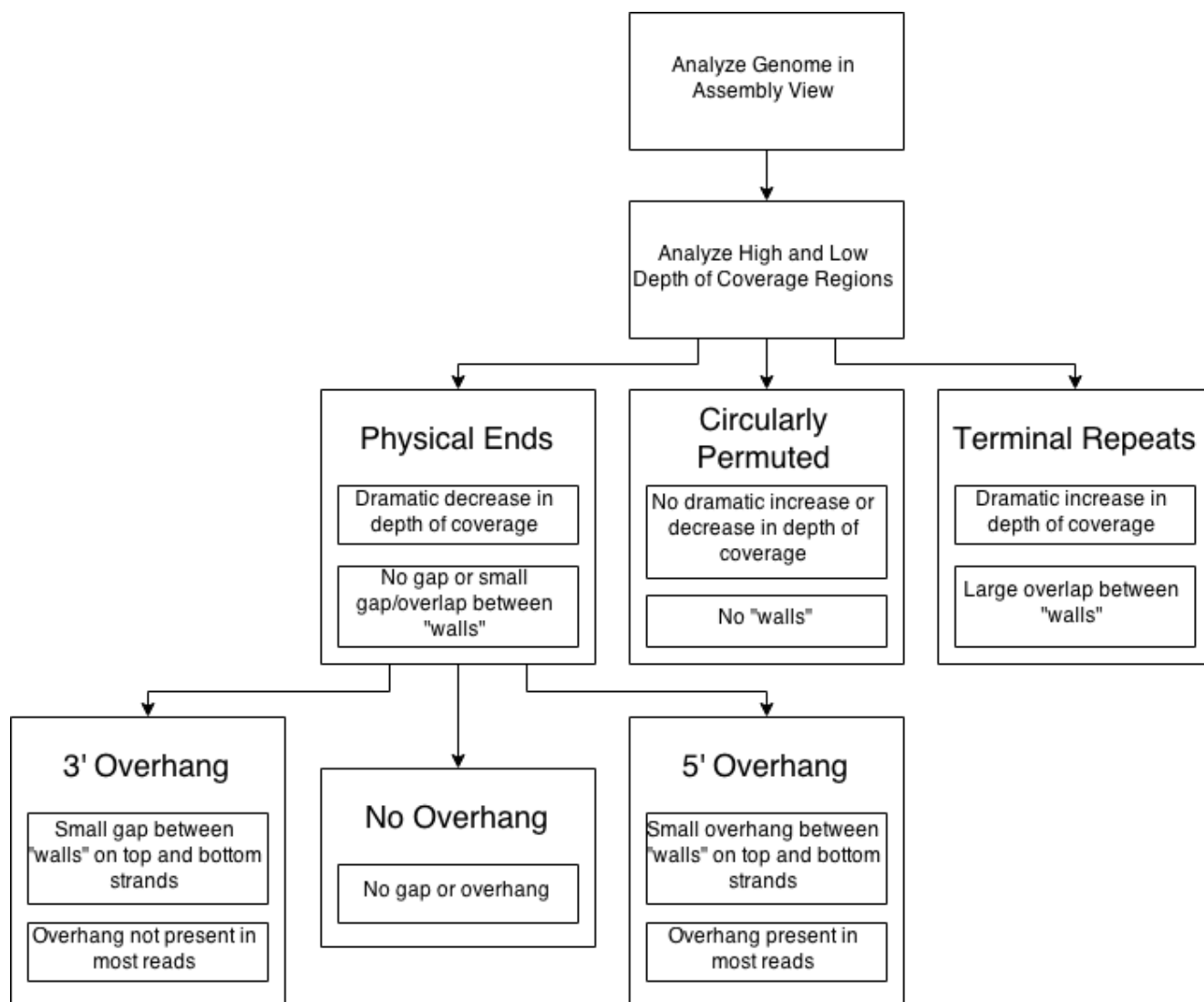


Figure 12 – Workflow for determining the ends of a bacteriophage genome.

To find the ends of the genome:

1. From the Consed Main window, open the desired read in assembly view.
 - a. A genome with physically defined ends will exhibit a dramatic decrease in depth of coverage at the ends of the genome. If you suspect your genome has defined ends then continue to step 2. A genome with terminal repeats will exhibit a dramatic increase in depth of coverage. If you suspect that your genome has terminal repeats then continue to step 3. A circularly permuted genome will not have a region of dramatically higher or lower coverage. If both high and low coverage regions have been analyzed and no ends have been identified and you suspect that your genome may be circularly permuted, then continue to step 4.

2. Physically Defined Ends:

- a. From the Consed Main Window, search for [low depth of coverage regions](#) and set the “max depth of coverage” parameter to the read depth on both ends of the low depth of coverage region (can determine this from the bottom of the Assembly View window where it says “read depth. The mouse pointer must be in the grey area for “read depth” to appear at a given base). Make a note of approximately which base the low depth of coverage region begins and ends and what the read depth is at these locations.

*Note - You may have to zoom in on larger genomes to be able to see these regions more clearly.



Figure 13 – Low depth of coverage region with read depth circled. Max depth of coverage parameter was set to 150 since the read depth just above is over 150 and appears to increase to well over 150 as soon as the low depth of coverage region begins. The Assembly View window is zoomed in to show the low depth of region in question more clearly.

- b. In the low depth of coverage results, select the region that matches closest to the region that you previously approximated in the Aligned Reads view. Click “Go” to navigate to the lower read position of the high depth of coverage region in Assembly View. Before being able to see a wall you must first sort the reads by strand by clicking the “Sort” tab, “Sort options and Help”, and then click on both “Strand/Left End” and “by method specified above”. Then, scroll down until you see a “wall of reads” on either the forward or reverse strand. This is one end of the genome. If you don’t see a wall exactly at the region where the high depth of coverage indicated, you may need to scroll left or right within the Aligned Reads window until you see the “wall”.

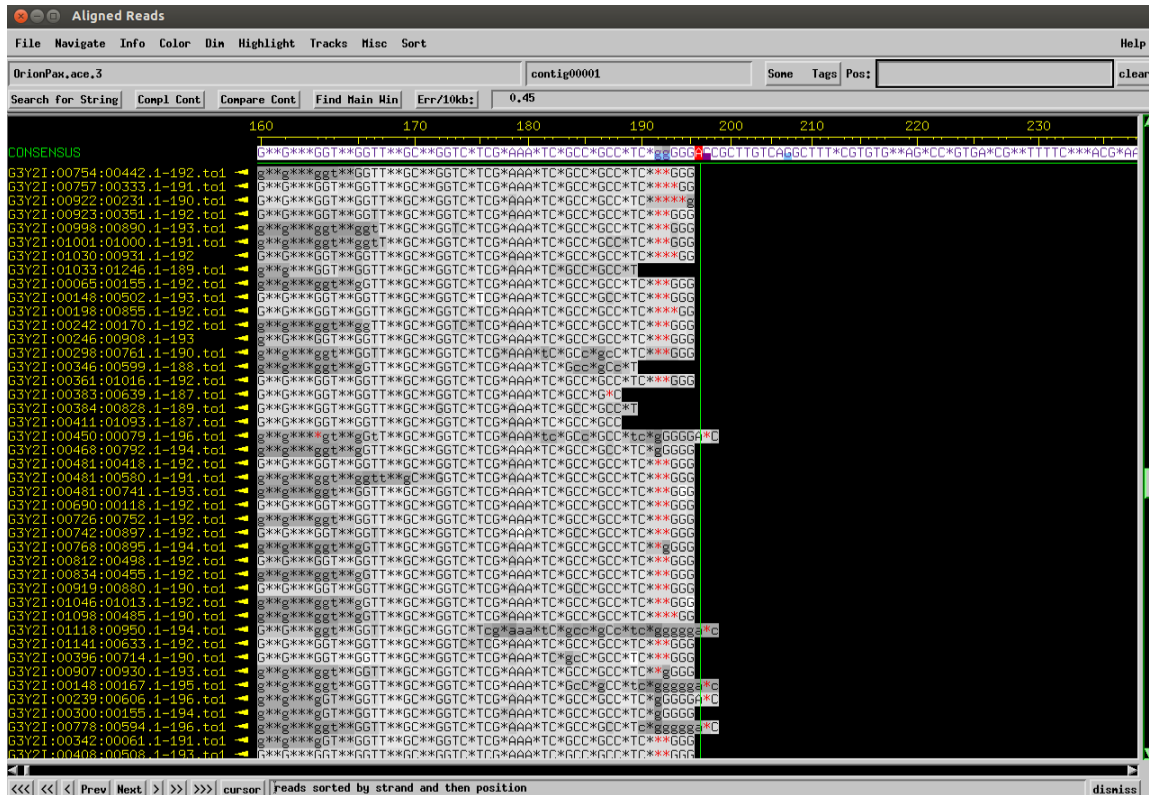


Figure 14 – The “wall” on the bottom strand.

- c. In the top right of the Aligned Reads window, type in the upper limit of the high depth of coverage region within the “Pos” field to navigate to that base position in the read. Scroll until you see a “wall” on the opposite strand as the “wall” found at the lower limit of the high depth of coverage region. This is the other end of the genome.



- *Note - You may have to zoom in on larger genomes to be able to see these regions more clearly.

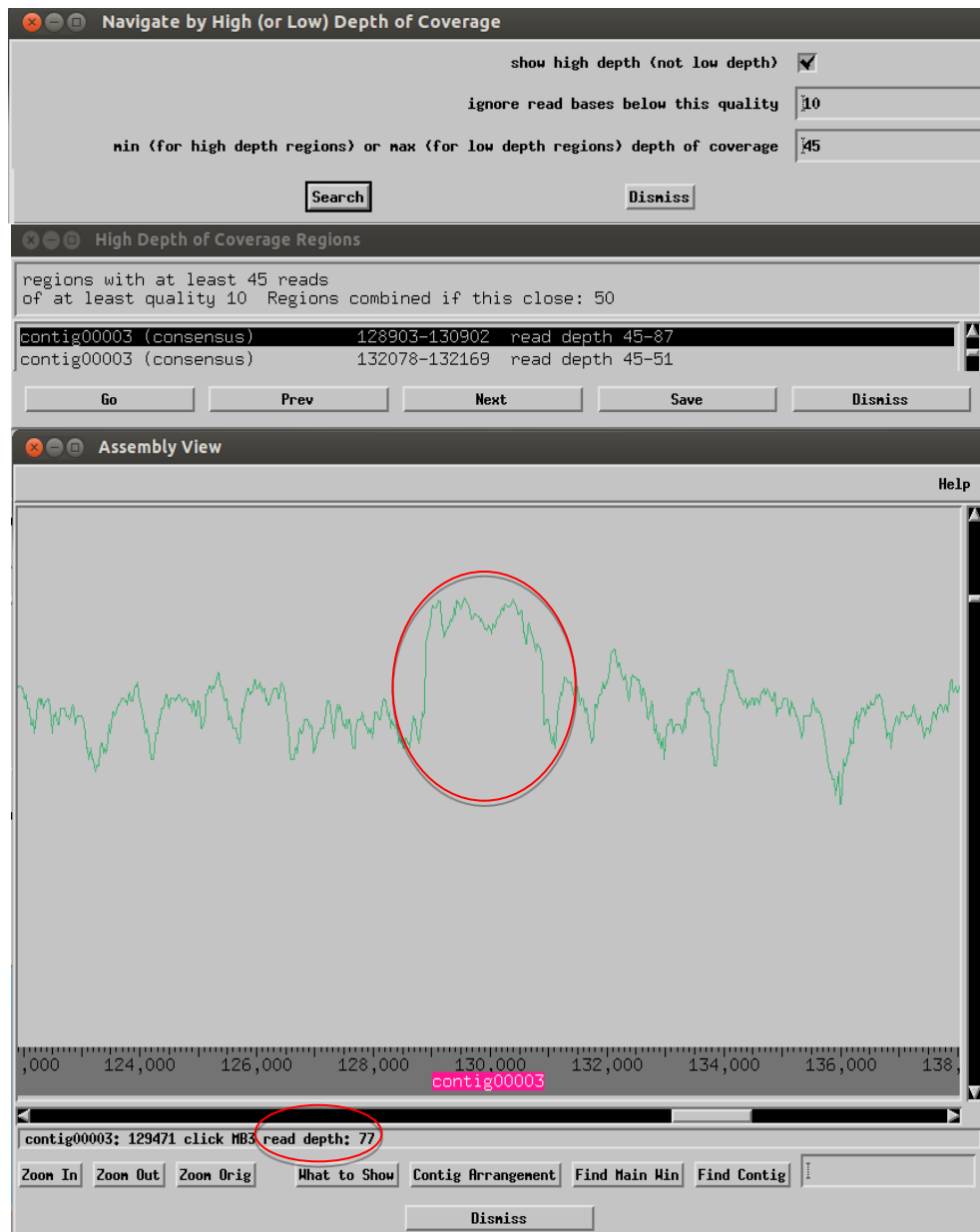


Figure 16 – High depth of coverage region with read depth circled. Min depth of coverage parameter was set to 45 since the read depth just below is under 45 and appears to increase to well over 45 as soon as the high depth of coverage region begins. The mouse pointer was moved over the grey area at base 129471 where the read depth was 77, which is much higher than the read depths on both sides of this region.

- b. In the high depth of coverage results, select the region that matches closest to the region that you previously approximated in the Aligned Reads view. Click “Go” to navigate to the lower read position of the high depth of coverage region in Assembly View. Before being able to see a wall you must first sort the reads by strand by clicking the “Sort” tab, “Sort options and Help”, and then click on both “Strand/Left End” and “by method specified above”. Then, scroll down until you see a “wall of reads” on either the forward or reverse strand. This is one end of the genome. If you don’t see a

wall exactly at the region where the high depth of coverage indicated, you may need to scroll left or right within the Aligned Reads window until you see the “wall”.

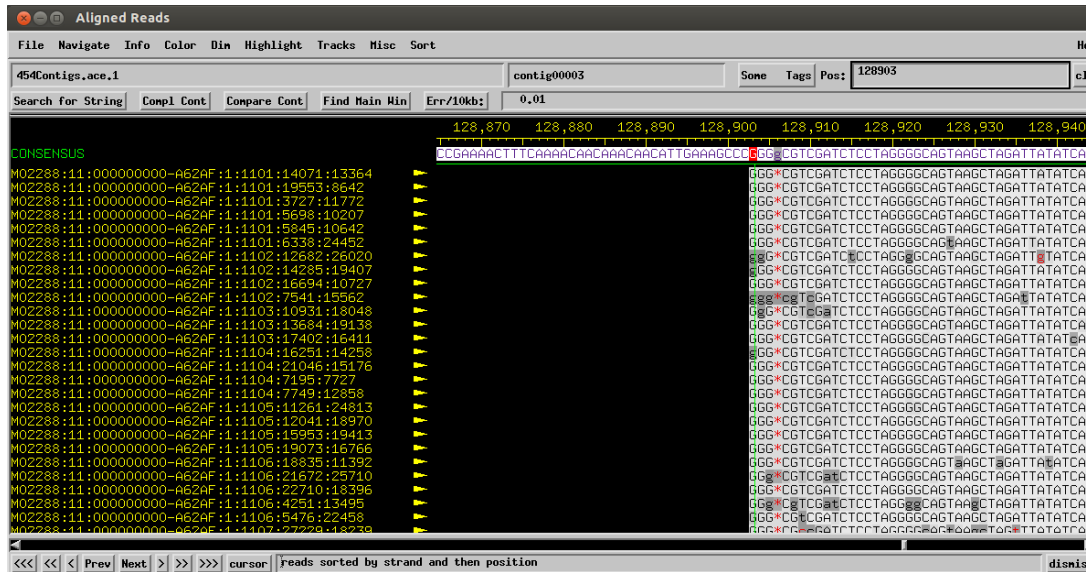


Figure 17 – The “wall” on the top strand.

- c. In the top right of the Aligned Reads window, type in the upper limit of the high depth of coverage region within the “Pos” field to navigate to that base position in the read. Scroll until you see a “wall” on the opposite strand as the “wall” found at the lower limit of the high depth of coverage region. This is the other end of the genome.

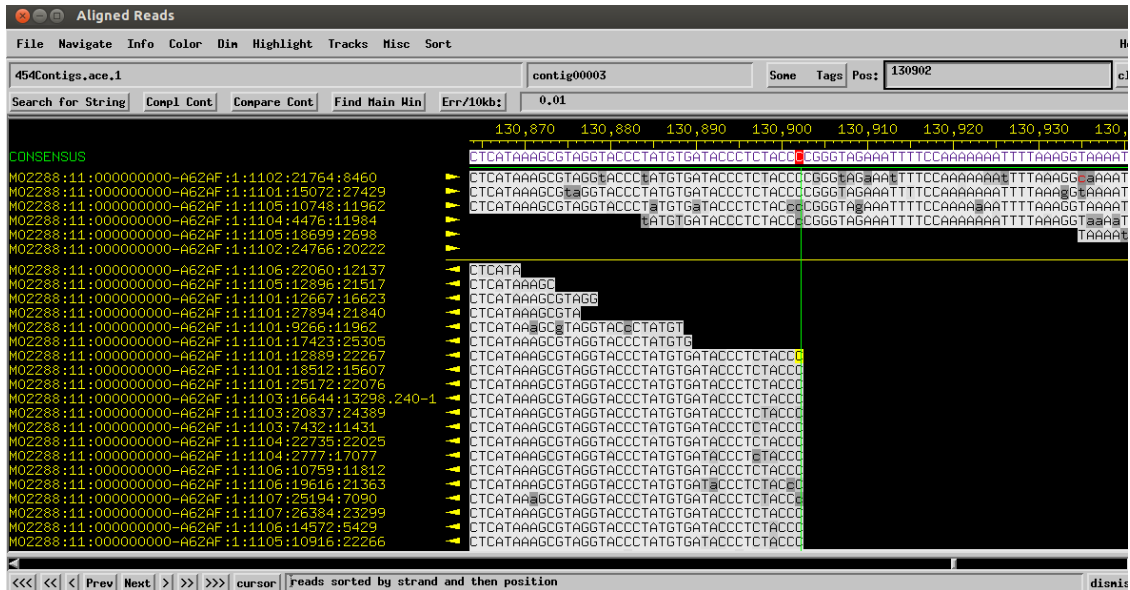


Figure 18 – The “wall” on the bottom strand.

4. Circularly Permuted:

- a. If all of the low and high depth of coverage regions have been analyzed and no defined ends or terminal repeats have been identified, then your genome may be circularly permuted. A circularly permuted genome will not exhibit any distinct “walls”. It is important to note that there might not be able to distinguish between types of genome ends by simply looking at the assembly view alone, as evidenced by the image below. A genome can only be determined as circularly permuted once the other two ends types have been definitively ruled out.

*Note - You may have to zoom in on larger genomes to be able to see these regions more clearly.

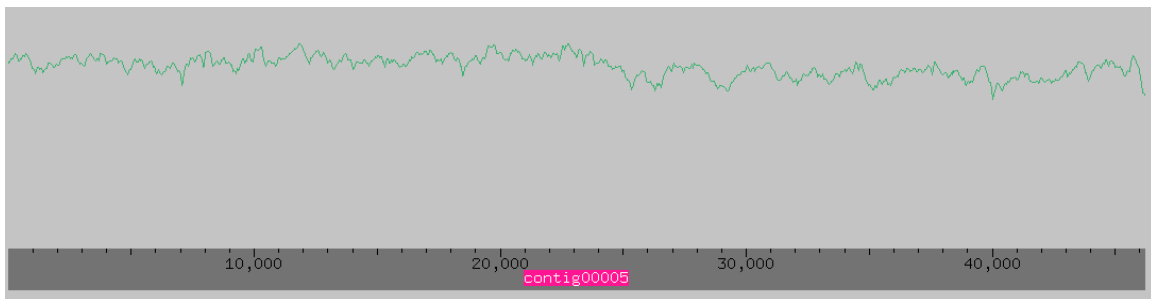


Figure 19 – A circularly permuted genome as seen in Assembly View.

Determine Phage Cluster ^[12]

In order to determine the bacteriophage's cluster, three following three actions must occur: export the sequence data from Consed, BLAST the sequence as a .fasta file, and then use phagesdb.org to find similar phages.

1. Export sequence data to .fasta file
 - a. Save the contig as a .fasta file by double clicking on the contig to get to the aligned reads window. From there click “File” and then “export consensus sequence”.
 - i. If you want to save all of the contigs to a .fasta file go “File” and click on “write all contigs to fasta file”. If you blastn this entire file the BLAST program will blast them all individually. This is only recommended for .ace files with less than twenty contigs.
2. Perform a nucleotide BLAST on the fasta sequence on the NCBI BLAST website. It is recommended that you perform a blastn of highly similar sequences (megablast) and a blastn of somewhat similar sequences.

3. Based on the top BLAST results, you can find similar phages on phagedb.org to hypothesize information of the phage such as its cluster and whether or not it has defined ends. If your blast results indicate homology with bacterial genomes, then there was likely contamination in the DNA prep. If the blastn results are poor then you can BLAST on phagesdb.org to compare your sequence with phages that have not yet been published and added to GenBank yet.

Adding SFF or FASTQ reads to an existing Consed project ^[13]

Once you have already generated an assembly and viewed it in consed, you may want to recombine the remainder reads back with the downsampled reads ([see Selecting a Subset of Reads from a SFF or FASTQ file](#)). You may want to perform this step to help resolve coverage and quality issues in the consensus sequence. This step only needs to be done if you downsampled your files prior to assembly. Note: The following steps are for a .sff file. For a .fastq file, substitute sff_dir with solexa_dir.

1. Create a .fasta file from the .ace file (the .ACE file is located within the edit_dir folder).
 - a. Navigate in the command line to the edit_dir folder.
 - b. Run the command **"ace2.fasta.perl 454Contigs.ace.1"**. This command will convert the .ACE file into .FASTA file format.
2. Create Reads.fof file.
 - a. Open a text editor and type the following
 - i. `../sff_dir/(name of .sff remainder file)`
 - ii. Save the file as Reads.fof and save it in the edit_dir folder
3. Move the file to the correct location.
 - a. Within the consed folder, remove the sff_dir shortcut folder.
 - b. Create a new sff_dir folder
 - c. Move the SFF files into the newly created sff_dir folder.
 - i. Move the downsampled .sff file and the remainder .sff file into the new folder.
4. Within the edit_dir file directory, run either of the following commands
 - a. For an sff file, run the command **"add454Reads.perl 454Contigs.ace.1 Reads.fof 454 Contigs.ace.1.contigs"**

- b. For a .fastq file, run the command “**addSolexaReads.perl 454Contigs.ace.1 Reads.fof 454Contigs.ace.1.contigs**”
5. Launch the Consed program and open the file 454Contigs.ace.2, which will be the output of the previous step.

Newbler from the Command Line

If you would like to [run Newbler from the command line](#), this website from the University of Arizona provides a list of the commands necessary to do so. ^[14]

Navigating the Command Line in Linux for Beginners

If you are unfamiliar with using the command line in Linux, this website by Paul Cobbaut explains how to [navigate file directories from the command line in Linux](#). ^[15]

Example Phage Data

The following assembly folders are used throughout this manual and are available for those who wish to follow along with the same data that was used to generate many of the images in this manual.

1. Opening an assembly in Consed, using different Consed views, and coverage/weak areas – OrionPax
2. Finding genome ends
 - a. Defined ends – OrionPax
 - b. Terminal repeats – Pinkman
 - c. Circularly permuted – Waukesha92

***Please address all feedback and suggestions to assemblymanualsuggestions@gmail.com**

References

- [1]. *Contig*. (2015). Retrieved January 12, 2015, from <http://ghr.nlm.nih.gov/glossary=contig>
- [2]. *Phagehunting procedures & protocols*. Retrieved January 12, 2015, from <http://phagesdb.org/workflow/>
- [3]. Khaja, R., & Russell, D. (2013). *Consed and phamerator virtual machine installation*. Retrieved September 25, 2014, from <http://www.hhmi.org/seawiki/display/WIKINAV/Consed+and+Phamerator+Virtual+Machine+Installation>
- [4]. Bridgett, S. (2010). *Understanding and assembling 454 transcriptome sequences*. Retrieved September 26, 2014, from http://www.nesc.ac.uk/.../GenePool_Transcriptome_Workshop_Nov2010.ppt
- [5]. Russell, D. *Selecting a subset of reads from an SFF or FASTQ file*. Retrieved April 27, 2014, from <http://phagesdb.org/workflow/videos/NEWBLER02/>
- [6]. Russell, D. *Assembling phage genomes with newbler*. Retrieved April 27, 2014, from <http://phagesdb.org/workflow/videos/NEWBLER01/>
- [7]. Russell, D. *Consed & finishing #3*. Retrieved April 24, 2014, from <http://phagesdb.org/workflow/videos/consed03/>
- [8]. Russell, D. *Consed and finishing #4*. Retrieved April 27, 2014, from <http://phagesdb.org/workflow/videos/consed04/>
- [9]. Russell, D. *Consed & finishing #6*. Retrieved April 27, 2014, from <http://phagesdb.org/workflow/videos/consed06/>
- [10]. Russell, D. (2014). *3' versus 5' overhangs in sequencing data*. Retrieved November 4, 2014, from <http://phagesdb.org/blog/posts/25/>
- [11]. *Finishing phage genomes*. Retrieved September 21, 2014, from http://phagesdb.org/media/docs/Finishing_Genomes.pptx
- [12]. Russell, D. *Consed & finishing #7*. Retrieved April 27, 2014, from <http://phagesdb.org/workflow/videos/consed07/>
- [13]. Russell, D. *Adding .sff or .fastq reads to an existing consed project*. Retrieved April 27, 2014, from <http://phagesdb.org/workflow/videos/ADDINGREADS/>
- [14]. Fryslie, B. (2012). *Newbler*. Retrieved September 5, 2014, from http://swes.cals.arizona.edu/maier_lab/kartchner/documentation/index.php/home/docs/newbler
- [15]. Cobbaut, P. (2014). *Linux fundamentals: Chapter 6. working with directories*. Retrieved October 24, 2014, from <http://linux-training.be/files/books/html/fun/ch08.html#idp5335088>

Appendix 4. Responses from the first survey sent out to gauge user interest in the manual.

Question	Answer(s)
1) Are you currently performing any processing of genome data that precede annotation steps?	60% Yes
2) If you answered "Yes" to question 1, what program(s) are you currently using for DNA sequence assembly? Are you satisfied with these program(s) or are you considering other program(s)? If you answered "No" to question 1, please proceed to question 5.	<ul style="list-style-type: none"> - CLC Bio Genomic Workbench. - Velvet, metavelvet, mira. Fairly satisfied. - I generally get help with this when I do it. - Newbler and Consed. Satisfied with these programs. - Newbler. It's fine. -We received fastA files and have not had to assemble ourselves but need to double check the quality. - Newbler Version 2.6 or higher. It works great for 454 data, which is what we have. - Laser Gene DNA Star. We use this for other sequencing, but have not used it yet for phage. This is probably not the best software for phage. - Newbler available through the Virtual Box. I'm not completely satisfied. I think it is difficult to explain to students. - CLC Suite - they work well but are not open source and therefore do not do well in a course context. We have also tried MIRA - too difficult to use! - Consed CLCBio Trinity Not completely satisfied with any, at least for undergrads. - Newbler Satisfied but welcome to suggestions - We are just beginning this process, so we will be using Consed. I can manage using this software, but I anticipate that it will be very difficult for students who have little to no bioinformatics experience. - We use Pitt's sequencing services but are looking into getting phage genomes sequenced at NCSU.

	<ul style="list-style-type: none"> - newbler, phrap, AHA, Celera assembler. - Some Newbler and some CLC Genomics Workbench. CLC is expensive commercial software that we might have accessible to students in the future, but not at this time. Newbler is still free and requires some file size reduction and manipulation for paired end reads from Illumina. I don't think we have the best approach yet for students to perform assemblies. - Newbler, consed, BLAST, and some command line commands that I can't remember right now for manipulating fastq files. I'm satisfied. - I could use all the help I can get, I am using Newbler right now, but this summer have Miseq data with paired ends and not sure it will work as well - CLC-Bio Genomics Workbench, with Microbial Finishing Module. Satisfied - very easy for students to use but it is commercial and expensive. Have also used Velvet.
<p>3) If you answered "Yes" to question 1, what program(s) are you currently using to analyze the assemblies for sequence quality, genome end definition, and other "finishing processes"? Are you satisfied with these program(s) or are you considering other program(s)?</p>	<ul style="list-style-type: none"> - CLC Bio Genomic Workbench, Consed We are always open to other programs that can create a more efficient or effective pipeline. - I'm using my own scripts. As lso, amos. I'm happy with them. - In the past, when we were doing finishing and assembly, we used Velvet, a colleague is currently using Genious and loves it. - same as questions 2 - CONSED. It's OK. - trying to use Consed but it is daunting - Consed It works moderately. It is not user friendly. Some functions are missing, i.e. can't see chromatogram data so need to rely on quality scores. We are considering using Pause to analyze genome ends, but haven't tried it just yet. - Laser Gene DNA Star. We use this for other sequencing, but have not used it yet for phage. This is probably not the best software for phage. - Newbler with Consed I'm not completely satisfied. I think it is difficult to explain consed to students.

	<ul style="list-style-type: none"> - Again CLC suite - no they are not open source. - Consed, CLCBio, thinking of trying Mix - Consed, AceUtil Satisfied but welcome to suggestions - We have not yet evaluated the programs - THE NCSU sequencing center will put together contigs for us after sequencing but the rest is up to us to determine if the sequence is satisfactory. - consed, SMRT view - CLC genomics. Not totally satisfied. - consed and something from the Hatfull lab that I can't remember the name...it works with consed. I'm largely satisfied, but am still learning how to analyze for quality and genome end. - Newbler and consed - CLC-Bio Genomics Workbench, with Microbial Finishing Module. Satisfied - very easy for students to use but it is commercial and expensive.
<p>4) If you answered "Yes" to question 1, what are the biggest obstacles you face in the sequence assembly and finishing stages that are not being adequately addressed in current how-to guides, user manuals, and tutorials? Please proceed to question 7.</p>	<ul style="list-style-type: none"> - With the addition of barcoding, we are finding that discrete ends are not obvious because some ends ligate in the process and assembled sequences do not begin at ends. I brute forced this by using alignments to closely related genomes to help me align and rotate sequence. This does not work with new sequences like the Arthrobacter phages that we also sequenced. - Locating genomic anomalies such as inversions. Also, repeats and regions of low complexity. - End definition is mysterious - Figuring out the ends of genomes... - None currently identified. - We have not found a 'how to guide' for consed and have not looked for another program yet.

	<ul style="list-style-type: none"> - There are no good instructions for Consed. The Newbler blog is alright. Obstacles: piecing together contigs when the genome doesn't auto-assemble into one full length genome. - Lack of experience. - The steps are in the tutorials, but as far as I know, there is not a user's guide and it is not easy to link to the tutorials (if you don't know where they are). - The easy programs are not open sources and the open source ones are not easy to use. - Access for Undergrads, processing power - Determining End-repeats Determining Circular vs linear genome Determining the number of contigs to use and how to best generate them - Lack of step-by-step guides, screenshots, and commonly accepted guidelines and standards for finishing - Time and knowledge of sequence assembly and finishing stages. - no obstacles at this point - Software needs to be accessible online and independent of Mac or PC client. We can't be installing software on everyone's individual computer as it is time-intensive and problematic. I like our campus Virtual Computing Lab where we can install Windows or Unix-based (Ubuntu, Redhat, etc.) software for all to access at any time. - the quality analysis step - \not getting a single contig, trying to extract a subset of sequences from the fastq file, (I got it to work last year with sff) - Identifying ends
5) If you answered "No" to question 1, are you interested in performing pre-annotation processing of genome data?	84.62% Yes
6) If you answered "No" to question 1, what obstacles (if any) exist to	- Limited access to wet bench resources, limited (actually "no") experience with certain processes except for ancient hand

you performing pre-annotation processing of genome data?	<p>sequencing protocols</p> <ul style="list-style-type: none"> - Access to raw data. The sequence coming to us is already finished, but we have chosen this route simply due to lack of time and expertise needed to ensure we have a quality sequence to annotate. - Lack of knowledge and training - None, we have plenty of help here. - I'm really not sure what I'm doing. - No sequencer on site. To answer question 7 below, If we had a sequencer then we would use the guide - Lack knowledge to do so. - Currently we have no genomes that need to be finished. - It is currently being done for us by Pitt for the one sequence we submit, so we have not needed to But we would be interested in getting more phages sequenced and are willing to do the finishing ourselves. - We don't have access to the raw data or at least fastq files. Our sequencing to date has been done at Pitt and we only get the assembled / finished sequence. Other than that we have no limitations. - Too much to do already. We cannot perform sequencing in-house, so as long as Pitt can sequence and do the assembly, we would prefer that. - Guidance. - 1. I don't know how to do it 2. I've never had to learn before, because the processing has been done through the SEA-PHAGES program.
7) If a how-to guide were to be made to cover the sequence assembly and finishing processes, would you utilize it?	93.10%
8) If you answered "Yes" to question 7, in what medium would you most	<p>20% Wiki style webpage</p> <p>80% Digital user manual</p>

prefer it to be displayed?	
9) Do you work with students to assemble, finish, and/or annotate non-mycobacteriophage genomic DNA?	68.97% Yes

Appendix 5. Responses from the second survey sent out to gather feedback on the manual from the pilot test.

Question	Answer(s)
1) Overall, what rating would you give this manual for addressing the basic needs of you and your students? Choose 1 for poorly addressed our needs and 5 for perfectly addressed our needs.	1 – 0% 2 – 0% 3 – 9.09% 4 – 72.74% 5 – 18.18%
2) Are you satisfied with the level of detail provided by this manual, with respect to its length?	Yes, the level of detail and length are sufficient – 72.73% Yes, however the manual is too long and/or too detailed – 0.00% Yes, however the manual is too short and/or not detailed enough – 18.18%
3) Rate your satisfaction with the following section: Processing. Choose 1 for completely unsatisfied and 5 for completely satisfied.	1 – 0% 2 – 0% 3 – 9.09% 4 – 45.45% 5 – 45.45%
4) Rate your satisfaction with the following section: Assembly. Choose 1 for completely unsatisfied and 5 for completely satisfied.	1 – 0% 2 – 0% 3 – 18.18% 4 – 45.45% 5 – 36.36%
5) Rate your satisfaction with the following section: Assembly Analysis. Choose 1 for completely unsatisfied and 5 for completely satisfied	1 – 0% 2 – 0% 3 – 9.09% 4 – 54.55% 5 – 36.36%

<p>6) In the section Assembly Analysis, which topics need the most improvement in their explanations? Select all that apply.</p>	<p>Opening Consed – 9.09%</p> <p>Determine Which Contig to View – 36.36%</p> <p>Consed Views – 0%</p> <p>Visual Analysis of Read Quality (Chromatogram) – 27.27%</p> <p>Coverage and Weak Areas – 27.27%</p> <p>Finding the Ends of the Genome – 27.27%</p> <p>Determine Phage Cluster – 18.18%</p> <p>Adding sff or fastq Reads to an Existing Consed Project – 18.18%</p>
<p>7) Based on your response to the previous question, please provide a brief summary as to what specifically needs to be improved in these sections.</p>	<ul style="list-style-type: none"> - More details concerning the "why" of viewing read quality. - It would be really helpful to include the names of some of the tabs we need to click on. Some buttons were hard to locate on the screen, so more pictures might be helpful. - How to determine which contig to view. - Although we used this manual to support our learning in the class, most of the information in these areas was touched on in class. I simply found the explanation of how to find the ends of the genome confusing. Specifically, I am confused as to how this is determined and what data is significant in this process. We have not yet come to "Determining Phage Cluster" or "Adding SFF or FASTQ Reads to an Existing Consed Project"; however, I found what I read in these sections to be slightly ambiguous. I tend to be detail-oriented and desire completely thorough explanation of all facets involved in processes - of course, this is an ideal and not entirely realistic. - Before opening consed, it may be helpful to put in the instructions and shortcuts on how to get to certain folders in terminal (I.e. ls, cd "folder", etc.). For some reason, the instructions in this section were a bit confusing. I had little difficulty finding the coverage areas. - some direction for students about file structure, saving files and determining which contig to view. Better incorporating the screen shots in Finding Ends would help. - Coverage and Weak Areas - it might be helpful to include an example of a way that truly low areas of coverage are teased out by altering the Low Consensus Quality numbers.

<p>General Feedback that was both sent to the email account set up for the pilot test comments and from students in Dr. Louise Temple's genomics course.</p>	<ul style="list-style-type: none"> - Mention that you can only download files through virtual box, that the outside internet does not work and you cannot upload files into virtual box. - Tell the reader where to go to enter commands (terminal) - Under to fine the end of the genome it says "If you suspect that your genome has terminal repeats them continue to step." Which step is this referring to it is not specified. Should be step 3? - Should also note that when you place your cursor over the grey ruler in assembly view you can see the depth coverage so that it is more easily determined because it is difficult to tell whether there is a drastic increase in depth or height without knowing the actual depths. - Also note that you can zoom in and out and that you have to move the ruler in order to see the same information - Explain what to do when receiving a zip file. - Explain basic uses of terminal or how to access the file on the desktop through terminal. - When opening consed, it would helpful to include the "ls" function in the directions to help visualize what folders are being opened and where you are actually looking. - (Chapter "Finding Ends of a Genome") In assembly view, explain that you can zoom in because it's difficult to see. - (Chapter "Finding Ends of a Genome") Explain how to open the "high or low depth coverage" window. - "Terminal Repeats" section is unclear. What should one type as the range in the " high/low depth coverage window"? - How to Unzip and extract files. - Doesn't say how to get to Newbler through virtual machine. - Make more illustrative than explanatory For eg: Go to File > genome > auto-annotate - Assembly section page 6: What do we do after we examined matrix
--	---

	<p>file of multiple contigs?</p> <p>- Part 1 of finding ends of the genome page number 14 Step 1 A. The step to go to for terminal repeat is missing.</p>
--	---