# Genome Assembly for

# COVID-19 (SARS-CoV-2) virus and *Carsonella ruddii bacterium*

Author: Kairuo Yan

Group Partner: Yanni Guo, Xinduo Fan

Date: October 28, 2021

## Introduction

In this lab, we have two parts. In the first part, we are going to assemble the COVID-19 genome using the online software Galaxy. In the second part, we are going to write the program to assemble the reads of a bacterial genome. The generalized steps for genome assembly could be concluded in following steps: first is multiplying a dna molecule by cloning, then partitioning clones into random fragments and sequence each fragment. Then assembling the sequence into contigs. Finally using a map to guide the final assembly. In this lab, for the first part of the lab, we would use the SARS-CoV-2 dataset from Galaxy Service and run the SPAdes software to find the best contigs. For the second part of report, we already have sequenced dna fragments dataset of *Carsonella ruddii bacterium(C.ruddi)*, thus we are designing specific algorithms for the genome assemblies for both parts of the lab since we are not able to sequence along the complete length of a chromosome(Kalyanaraman, 2011). Specifically, we built de Bruijn then allowed us to find a set of paths and build a list of contigs. We assess our contigs with N50 statistics to get the best set of contigs in the whole sequence data, and eventually mapping them into the final assembly.

Genome assembly is important because current technology does not allow us to read an entire genome and locate the reads. Assembling a genome allows scientists to obtain information of medical value for future care. Also genomic sequencing can provide information on genetic variants that can lead to disease or lead to the increasing risk of disease development (Compeau and Pevzne, 2015).

## Methods

For the first part of our lab, we did research on SARS-Cov2 and then used the SPAdes assembler tool on Galaxy to assemble its genome. We obtained the basic information of SARS-CoV-2 from the Sequence Read Archive in NCBI ("Run Browser : Browse : Sequence Read Archive : NCBI/NLM/NIH" ), and these information include: the number of nucleotide bases, the GC content of the reads, and the Phred scores of reads. Then we assemble the SARS-CoV-2 genome in the SPAdes using the service of Galaxy and get the set of contigs(Phillip, 2021). Finally, we produced an assembly graph as a version of the de Bruijn graph in which maximal non-branching paths have been compressed to a single edge.

For the second part of the lab, we designed six main algorithms to assemble *C.ruddi's* genome. The first function is called *readpairs*, and the parameters are name and k. This function reads the FASTA file and returns a list of reads and a list of k-mers from the reads.

The second function is called *create_deBruijin*, and there's one parameter: kmers, which is a list of reads with the length of k that we gained from the previous function. In this function, we run through all kmers, and build the nodes for de Bruijn graph by appending the suffix to the prefix of the kmers. We would also keep track of the indegree and outdegree values for each node. As a result, the function returns three lists: graph, in_degrees and out_degrees. In the de Bruijn graph dictionary, the keys are the nodes of prefixes, and the values are the nodes of suffixes that are connected to the prefixes with edges.

The third function is called *GetMaximalNonbranchingPaths* with inputs: graph, inDegrees and outDegrees. The function is defined to be a path whose internal nodes have in-degree and out-degree both equal to one except for the starting and ending nodes not being 1. The function glues the nodes based on the de Bruijn graph we built from the previous function

into non-branching paths. Each maximal non-branching path in the de Bruijn graph corresponds to a contig in the genome.

The fourth function is called *constructContigs* with one parameter: paths, since every maximal non-branching path from the previous function corresponds to a contig, the function iterates over the paths and creates contigs and then adds the last base of the nodes to the start node. As a result, it returns a list of contigs.

The fifth function is called *calculateN50*, and it has a parameter l, which represents a list of reads. The function calculates the N50 value, which represents the contig with the shortest length that needs to be included for covering 50% of the genome.

The sixth function is called orderingContigs with parameters contigs and reads, which are the contigs list we gained from previous function and a string that are the reads of the FASTA file. The function examined each contig and its pair-read. The function searched for the paired-read number of each contig that was in a different contig except the paired-reads number is in the same contig as itself. The output was a list of pairs in a new FASTA file.

Finally, we designed a function called main(), and called previous functions. Since a small number of long contigs is preferred, we would test different k values in the range from 20 to 41 for the highest N50 value. Eventually, the function returns the best N50, best K value, number of contigs, and the length of each contig.

**Results**

In part one of our lab, we used the SPAdes assembler tool on Galaxy to analyze the genome assembly for the SARS-Cov2. Based on the NCBI Sequence Read Archive, we found for SARS-Cov2, the number of bases in the genome is 186.5Mbp, the GC content is 38.2%, and the coverage is 98.6% (Table 1). Then we found the read-pair at spot #15213 for SARS-Cov2,

and we found that the first read is longer than the second read. Thus, the first read has more nucleotides bases than the second read (Table 2).

| | |
|---|---|
| Number of bases | 186.5Mbp |
| GC content | 38.2% |
| Coverage | 98.6% |

**Table 1: SARS-Cov2's number of bases, GC content, and read coverage**

| | |
|---|---|
| First read at spot #15213 | CAACAAGGCCAAACTGTCACTAAGAA ATCTGCTGCTGAGGCTTCTAAGAAGCC TCGGCAAAACGTACTGCCACTAAAGCA TACAATGTAACACAAGCTTTCGGCAGA CGTGGTCCAGAACAAACCCAAGGAAA TTTTGGGGACCAGGA |
| Second read at spot spot #15213 | TCAATATGCTTATTCAGCAAAATGACT TGAT |

**Table 2. The read-pair at spot #15213 for SARS-Cov2**

We then observe the Phred quality score for the SARS-Cov2, which indicates the probability of the base correctly. Thus, the Phred scores would help determine a reliable path and contig in the de Bruijn graph. And we could estimate the accuracy of the Illumina genome assembly model (Zhang, 2017).

After reading the Illumina website, our group thinks there are four steps that Illumina considers to take for the quality score of a given base: first is sample preparation, second is cluster generation, third is sequencing, and the last step is data analysis. From the NCBI Sequence Archive, the histogram shows the Phred quality scores from 2 to 37 of the entire dataset. The values in each Phred quality score bin are the length of the reads. Our group believes the quality scores are not ideal because the histogram has a left-skew instead of a bell-shaped curve, which implies that the Phred quality scores are not evenly distributed. Additionally, in the NCBI Sequence Archive, the quality scores for SARS-Cov2 are shown as

14, 28, 32, and 37 (Table 3), and the majority of the quality scores are 37. The nucleotides that have quality scores of 14, 28, or 32 are not the ideal scores we are looking for.

| | |
|---|---|
| Quality scores | 14, 28, 32, and 37 |

**Table 3. Phred quality scores of reading "8595/1" of the SARS-CoV2 genome.**

According to genome assembler analysis from the SPAdes Software, there are 2 contigs produced. One is 147 long, and another one is 29,600 long (Table 4). To increase the chance of overlap, the target genome is typically sequenced in a redundant fashion, which is referred to as *genome coverage*, and higher coverage typically tends to provide information for a more accurate assembly(Baker, 2012). Finally, according to the assembly graph generated from SPAdes (Figure 1), I think the contig 2 is the yellow part on the graph, and the two contigs are the green and yellow strands.

| | |
|---|---|
| Number of contigs in the assembly | 2 contigs |
| Length of contig1 | 147 |
| Length of contig 2 | 29,600 |

**Table 4. Contigs for the SARS-CoV2 genome from genome assembler analysis**

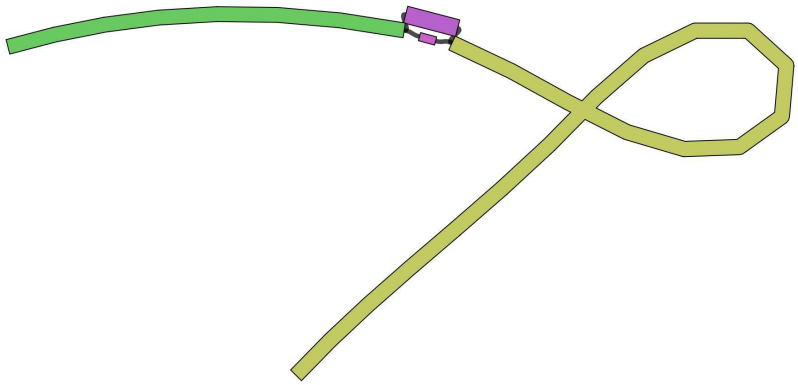

**Figure 1. Assembly Graph for SARS-CoV2 genome**

For part 2 of the report, to generate the best contigs, we would like to have a small number of long contigs. We gained the maximum N50 value when k equals 35. Therefore, when the length of reads (k-mer) is 35, we found the highest N50 value 25,419, the set of 16 contigs, and the set of each with contig lengths(Table 5).

| | |
|---|---|
| Highest N50 value | 25419 |
| K-value | 35 |
| Number of contigs | 16 |
| Length of each contig | [43, 46, 232, 607, 1147, 1175, 1686, 2898, 5621, 10674, 11784, 11863, 14378, 25419, 26425, 48930] |

**Table 5:** The table showed when K=35, the N50 value of contig assembly, the number of best contigs and the length of each contig.

**Discussion**

In conclusion of our result, for the first part of the lab, we learned the basic genetic information and Phred score of SARS-Cov2 dataset. We also found that there are two contigs for SARS-Cov2, one (29,600 bp) is much longer than the other (147 bp). Eventually, software SPAdes made an assembly graph, which visualized the length and position of contigs.

In the second part of the lab for *C.ruddi*, we learned that when the length of read is 35, there are 16 best contigs and each length varies from the shortest of 43 to the longest of 48930. The N50 value represents the contig with length of 25,419 that needs to be included for covering 50% of the genome, so that we could get a better understanding of the distribution of contig lengths. Based on our best set of contigs, we finished the final assembly of the genome, which is the FASTA file we wrote out. However, the genome assembly results are not fixed, and it is varied if we change parameters k, which is the value of read length. This is because the length of

reads determines the De Bruijn graph, which would affect the contigs, contig lengths and N50. Thus, we could have different genome assembly results of the same genome within multiple dimensions of evaluations. According to the study, using sequence reads of various read-lengths, coverages, accuracies, and with and without mate-pairs. Setting these different criteria could help Computational biologists to determine an innovative sequence assembly paradigm and obtain favorable methods for the development of "next generation" assemblers. Biotechnologists would also more likely formulate more meaningful design requirements for sequencing technology platforms(Narzisi, 2011). In one recent study, scientists tried to evaluate the choices on de novo assembly of SARS-CoV-2 genome and they performed 6648 de novo assemblies of 416 SARS-CoV-2 samples using eight different assemblers with different *k*-mer lengths. As a result, they found at least 09% (259/2873) of the variances among assemblies(Islam, 2021).

Overall, the genome assembly for both SARS-Cov2 and *C.ruddi* that we executed and designed followed the standard principles that we learned. However, testing and setting different criterias for read-lengths, coverages, accuracies and other variables would help us to gain a more comprehensive understanding of genomes.

**Bibliography**

Baker, M. *De novo* genome assembly: what every biologist should know. *Nat Methods* 9, 333–337 (2012). https://doi.org/10.1038/nmeth.1935

Narzisi G, Mishra B (2011) Comparing De Novo Genome Assembly: The Long and Short of It. PLoS ONE 6(4): e19175. https://doi.org/10.1371/journal.pone.0019175

Islam R, Raju RS, Tasnim N, et al. Choice of assemblers has a critical impact on de novo assembly of SARS-CoV-2 genome and characterizing variants. *Brief Bioinform*. 2021;22(5):bbab102. doi:10.1093/bib/bbab102

Phillip, Compeau. “SARS-COV-2 Software Assignment: Genome Assembly and Annotation: Phillip Compeau.” *Phillip Compeau, Carnegie Mellon University*, 6 Aug. 2021,compeau.cbd.cmu.edu/online-education/sars-cov-2-software-assignments/covid-19-genome-assembly-assignment/.

Kalyanaraman A. (2011) Genome Assembly. In: Padua D. (eds) Encyclopedia of Parallel Computing. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-09766-4_402

“Run Browser : Browse : Sequence Read Archive : NCBI/NLM/NIH.” *National Center for Biotechnology Information*, U.S. National Library of Medicine, trace.ncbi.nlm.nih.gov/Traces/sra/?run=SRR11528307.

Zhang, S., Wang, B., Wan, L. *et al.* Estimating Phred scores of Illumina base calls by logistic regression and sparse modeling. *BMC Bioinformatics* 18, 335 (2017). https://doi.org/10.1186/s12859-017-1743-4