2023

A Computational Analysis of Hybrid Genome Assembly Strategies

Joseph Walewski
Connecticut College, josephwalewski27@gmail.com

A Computational Analysis of Hybrid Genome Assembly Strategies.

An honors thesis presented by
Joseph Walewski

And advised by
Professor Douglass (Computer Science, Biology)

Readers: Professor Izmirli (Computer Science), Dean Eastman (Biology)

To the Department of Computer Science
In partial fulfillment of the requirements for
Honors

Connecticut College
New London, Connecticut

May 3rd, 2023

Abstract

The central dogma of molecular biology states that DNA is transcribed to RNA and then translated into proteins. Since DNA is the starting material for many of biology's macromolecules, it has been referred to as "nature's instruction book." The sum of all DNA in a cell is referred to as the genome, and genome sequencing is how we interpret the DNA.

Due to limitations on currently available technology, it is not possible to retrieve the entire genome in one contiguous set of data. Therefore, genome sequencing is a computer science problem as sequencing "reads" must be stitched together to obtain the complete genome sequence. There are two types of reads: short, accurate ones and long, inaccurate ones, and it is currently unclear how to most efficiently combine them. This is especially true of large genomes, where the cost of data acquisition is more expensive and the assembly step is harder. Therefore, we were motivated to simulate the process of genome sequencing on various organisms and then to reassemble their genomes based on varying levels of short and long read coverages.

Our results, while incomplete due to the nature of genomic data, show that approximately 25X short, accurate read coverage and 14X long, inaccurate read coverage are sufficient to assemble most large (>100 Mbp) genomes. Critically, the amount of coverage required stays relatively constant, even as genome size increases by over an order of magnitude.

This surprising find suggests that large genomes may be slightly easier to assemble than previously thought. As the cost of sequencing continues to fall the bioinformatics community should continue to heavily invest in the field of genomics, hopefully aided by our results to do the most efficient work possible.

Acknowledgements

I have been interested in biology ever since I could remember. To that end, I would like to thank my parents, who initially let me explore this passion by taking me on nature walks and allowing me to keep fish and salamanders as pets. This passion for biology inspired me to take BIO 120D ("From genes to the genome") with Professor Douglass, my computer science advisor, thesis advisor, and research mentor. His guidance has allowed me to explore my interest in newts and genomics far beyond what I thought was possible for an undergraduate.

While my love for computer science came much later, I am grateful to have taken COM 110 my second semester here at Connecticut College. While I was initially motivated to do so by the computational nature of BIO 120D, I quickly realized that I loved the incredibly logical nature of computer science in its own right. I then rushed to complete the major as quickly as I could while keeping up with the Biochemistry, Cellular and Molecular Biology courses required to complete my original major. While completing my coursework, I met Professor Izmirli and Dean Eastman, my readers. I thank both of them immensely for their time and dedication to my work. With eighteen days until my commencement, I am incredibly grateful to have attended Connecticut College and graduated with two majors that will allow me to "solve nearly any problem that comes my way."

Lastly, I would like to thank other students who have helped me along the way and suggested valuable edits, especially Vivian Taylor. She has always been by my side and she was the only other person there when I first saw the data suggesting that genome size doesn't significantly affect the level of coverage needed to properly assemble a genome.

Contents

Introduction

Biological Background

An Introduction to DNA

Your 23&me results come back: brown hair, likely lactose tolerant, cystic fibrosis carrier. Between spitting in the tube and the information displayed online lies a complicated process of DNA extraction, genome sequencing, and genome assembly. Each one of these three steps has seen dramatic technical improvements in the past 20 years, which has led to an enormous increase in both the scale and affordability of understanding the genome, the sum of all DNA inside of any one cell. It is often referred to as "nature's instruction book" for this reason. With cutting edge technology it is becoming possible to tackle the most complicated genomes in the tree of life, which may provide significant insights into human health and medicine. Salamanders, for example, are famous for their exceptional regenerative abilities, and hope remains high that one day human therapies may be derived from them.[1] However, their genomes are ten times the size of our own, so understanding them is currently very challenging. Biological data are massive, and to solve future problems in medicine comprehensive backgrounds in both biochemistry and computer science will be essential.

One of the most essential properties of biological matter is that it has heritable material. Heritable material allows life to operate on a day to day basis and can be passed down to future generations. In all living organisms(therefore excluding viruses and prions) the heritable material is stored as deoxyribonucleic acid (DNA). DNA is a macromolecule, made up of repeating units of molecular fragments called nucleotides. The nucleotide is the smallest discrete unit of the genetic code and therefore is a key starting point for understanding molecular genetics.

[1] Joven, Alberto, et al. "Model Systems for Regeneration: Salamanders."

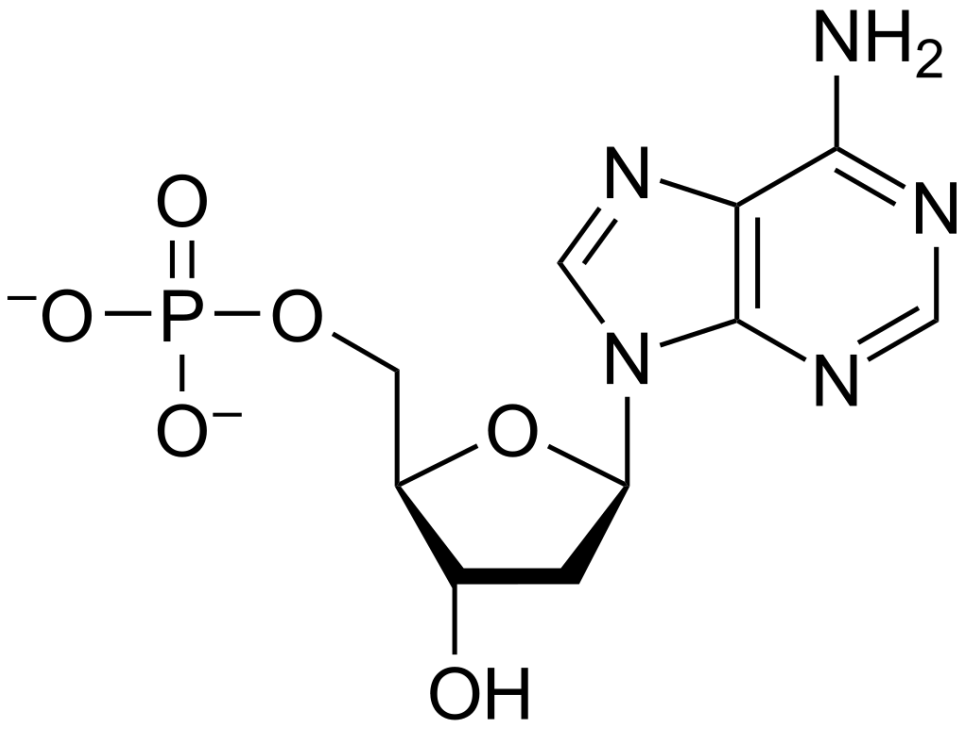

Figure 1. The chemical structure of a nucleotide. The lines indicate covalent bonds, double lines indicate double covalent bonds, H represents hydrogen, N represents nitrogen, O represents oxygen, and P represents phosphorus. Line junctions without labels indicate a carbon atom at their intersection. [2]

Each nucleotide in turn has three components, the first of these being the nitrogenous bases Adenine ("A"), Thymine ("T"), Cytosine ("C"), and Guanine ("G") which actually encode the heritable information. Adenine and Guanine are purines, nitrogenous bases with two rings, while cytosine and thymine are pyrimidines as they feature one ring. The primary implication of this relevant to genome sequencing is that Adenine and Guanine are significantly larger than Cytosine and Thymine. Since DNA must be a fixed width, purine-purine bonds are prohibited (too wide) as are pyrimidine-pyrimidine bonds (too narrow). Covalent bonding is the primary

[2] Hbf878. "File:Damp Chemical Structure.svg."

method in which atoms in molecules share electrons, and the end of a covalent bonding network is what delineates individual molecules from one another. However, there is another type of bond: the hydrogen bond.

Hydrogen bonds exist when a hydrogen on one molecule is bonded to a very electronegative atom (either nitrogen or oxygen) on the same molecule, pulling away the hydrogen's electron and rendering the proton very exposed. This leaves a strong positive charge on the outside of the molecule, which an electronegative atom (again nitrogen or oxygen, but additionally fluorine as well) on the exterior of another molecule can be attracted to. This bond is not as strong as a covalent bond, but it is key for holding DNA together and is the only type of bond holding nitrogenous base pairs together. The concept of the base pair as the fundamental unit of genetic information is so ubiquitous that "base pair" (bp) is one of the terms commonly used to measure genome size. Additionally, a key advantage of having hydrogen bonds holding the base pairs together is that, while individually weak, the sheer number of bases in DNA renders the collective base pair bonding incredibly strong. This dual behavior of being locally weak but globally strong allows DNA to open in a very controlled fashion. This is ideal for targeted gene regulation and replication, both essential to life itself.

Adenine and Thymine each have two hydrogen bonding sites, while Cytosine and Guanine each have three. This is another constraint on how the base pairs bond - while size prohibits Adenine from bonding with itself or guanine, the difference in the number of hydrogen bonds between it and Cytosine also renders it an invalid partner. Thus, the only base which Adenie can bind to is Thymine. Therefore, A and T are considered complementary, as are C and G. Therefore, the genetic code is very, very precise as each base read conveys very specific information.

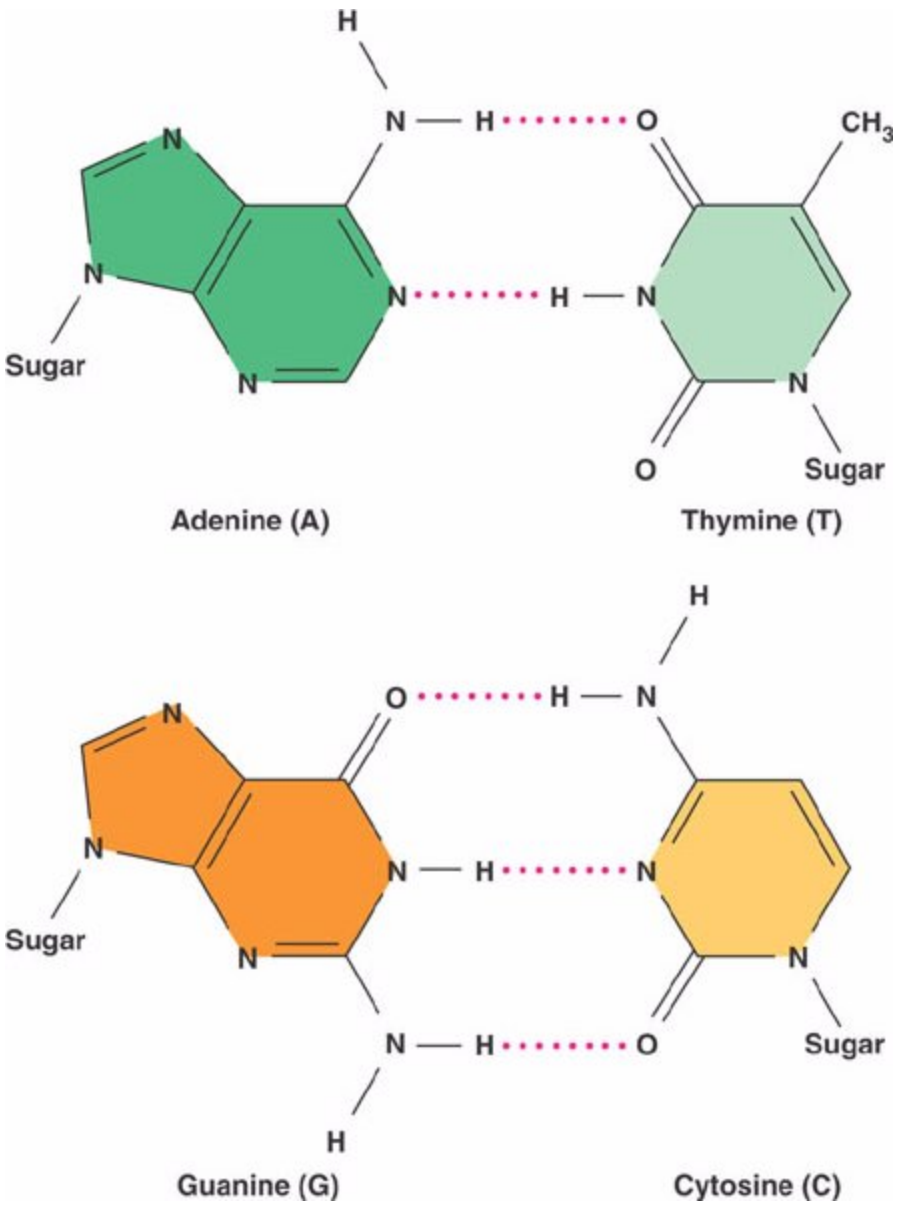

Figure 2. The nitrogenous bases present in DNA. Note that, due to the combination of being either a purine or pyrimidine, and due to either having two or three hydrogen bonding sites (indicated by dotted lines), each nitrogenous base can be paired with exactly one other.[3]

[3] "Base Pairing in DNA." *Base-Pairs.html 16_08dnabasepairing_l.Jpg*, http://bio1152.nicerweb.com/Locked/media/ch16/base-pairs.html.

The genetic code itself is read three letters at a time - such a grouping is referred to as a codon. According to the central dogma of molecular biology, these codons are then transcribed into Ribonucleic acid (RNA). RNA is a similar but less stable version of DNA that features Uracil ("U") instead of T, and then the RNA is translated into proteins. These proteins then carry out most of the operations of the cell and provide most of the structural support required for the organism to live. Proteins are made of up their own building blocks called amino acids, of which 20 are found across all life and represented in the genetic code.[4] This is why a codon is 3 nucleotides long - if a codon were only two nucleotides long, there would only be 16 representations available for amino acids (four in the first position multiplied by four in the second position). Three nucleotides, however, allows for 64 unique signals - since this is significantly more than the 20 needed, there is a great deal of redundancy. Critically, however, the genetic code is unambiguous as one codon always codes for exactly one amino acid. An additional advantage of having extra signals is that it allows for the use of stop codons that do not actually code an amino acid - this way, a protein can end with any amino acid. As the twenty amino acids vary widely in terms of reactivity, solubility, and many other chemical properties, having a variety of options allows for greater protein diversity. This is especially important near the end of the protein sequence, which is likely to be exposed and somewhat mobile after protein folding. Curiously, however, the singular start codon AUG additionally codes for methionine, one of the 20 amino acids. Yet another advantage of the redundant genetic code is that certain codons that code for the same amino acid are more ideally suited for different environments - A and T are considered "weakly bonded" due to their two hydrogen bonds between them; C and G are "strongly bonded" as they have three. Therefore, in conditions that favor bond breaking, such as high temperatures, C and G are favored to keep DNA intact when it is not being replicated or

[4] Niu, C.-H., et al. "The Code within the Codons."

read; when conditions favor bond integrity A and T are favored so that the DNA can be opened at all.[5] This has important implications for genome sequencing - since the reactions that copy and read DNA to be analyzed have been optimized to work at a specific temperature, they reflect the average GC content across species. Therefore, it is slightly easier to sequence the genomes of species with lower GC content (and therefore more As and Ts), and within a genome reads are biased towards areas that have proportionally lower GC content.[6]

The second component present in DNA is phosphate, an anion (negatively charged molecule) that along with deoxyribose forms the "backbone" of DNA. The phosphate anion has a very negative charge for its size (-3 elementary charges with a molecular weight of 94.971 atomic mass units), yielding an anion with a great deal of potential energy. This energy is used by the cell to allow the process of DNA synthesis to be favorable, especially since three phosphate anions are attached to an incoming nucleotide (but only one remains on the DNA strand when synthesis is complete).

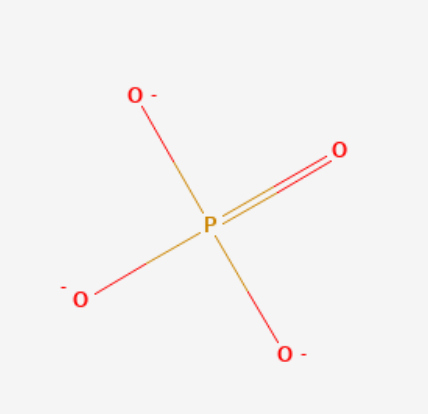

Figure 3. The phosphate anion. All oxygens with a single line to the central phosphate are singly bonded and therefore have an extra electron, resulting in a negative charge on each atom.[7]

[5] Benjamini, Yuval, and Terence P Speed. "Summarizing and Correcting the GC Content Bias in High-Throughput Sequencing."
[6] Ibid
[7] "Phosphate Ion." *National Center for Biotechnology Information.*

The final component is Deoxyribose, more precisely D-2-deoxyribose, is a five-carbon sugar ring that forms an essential part of the backbone of DNA. With ribose as a precursor, the removal of the oxygen at the second carbon allows for greater flexibility. This is required for DNA to form the classic "double helix" shape, each helix with its own sugar-phosphate backbone and nucleotides. This shape, in turn, offers greater stability than a planar molecule and additionally allows for extensive compaction at larger scales.[8] When deoxyribose is incorporated into DNA the 5' hydroxy (OH) group is replaced with the tri phosphate group mentioned earlier.

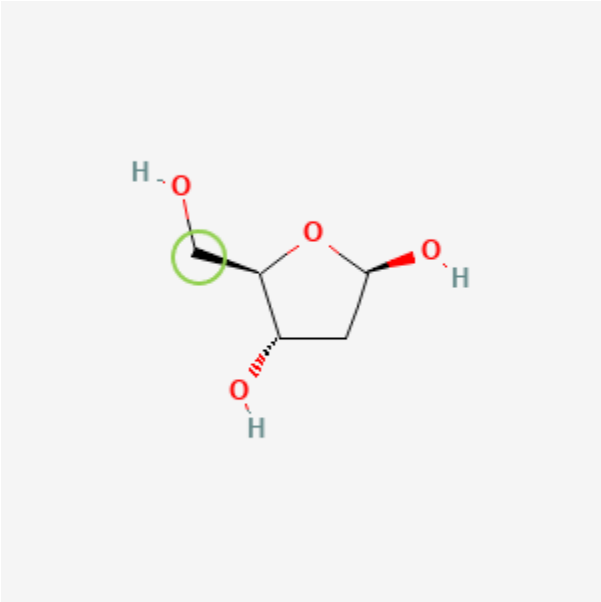

Figure 4. The structure of 2-D-deoxyribose, with the 5' carbon circled in green.[9]

However, the structure of deoxyribose's covalent bond with Phosphate yields an interesting property: DNA polymerization (extension of the molecule, required for replicating it and thus reproduction of all life) can only occur in one direction (from the 5' carbon to the 3'

[8] Gruenwedel, D W. "Nucleic Acids: Properties and Determination."
[9] "Beta-D-2-Deoxyribose." *National Center for Biotechnology Information. PubChem Compound Database*, U.S. National Library of Medicine,
The image was modified to include a green circle on the 5-carbon for clarity.

carbon in deoxyribose). Due to this, the two strands of DNA wrap around in an antiparallel fashion (the directions each strand can add in point in the opposite direction). This way, one strand can serve as the template for the other during the replication process. Consider what would happen if the strands were oriented the same way: an extending DNA molecule would have no "backup" to copy off of as both strands would be extending in the same direction simultaneously. If this was the case, DNA would not be able to preserve the information in the previously existing molecule and inheriting information from the previous generation would be impossible. The method of DNA replication that exists in nature is referred to as semiconservative replication, shown in the figure below.

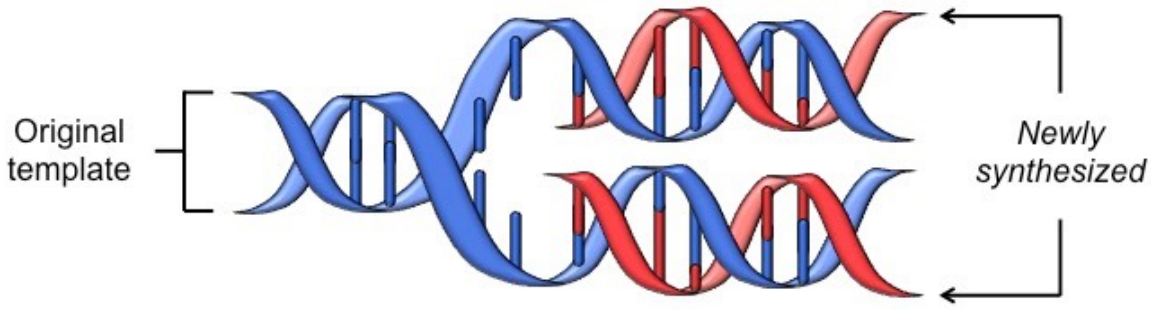

Figure 5. A diagram of semiconservative replication. Each strand of DNA consists of one strand used in the previous generation that exists as a template for the newly synthesized strand.[10]

One of the most important implications of semiconservative replication is that genes may appear on either strand of DNA. This complicates the process of inferring codons, genes, and other sequences of interest (promoters, enhancers, splice sites, etc) during genome sequencing as either the original sequence or its reverse complement (the exact opposite sequence in reverse

[10] Cornell, Brent. "Semi-Conservative DNA Replication."

order) may appear. Therefore, computer algorithms attempting to identify, align, or assemble reads must account for this scrambling of information.

Chromosomes and Gene Expression

Other problems exist for sequencing machines as well due to the multitude of layers of DNA compaction present in cells. DNA is compacted very tightly as the total length of all DNA in a human cell would be over two meters long if stretched end to end. The first layer of compaction is done by histone proteins: each one can store 147 nucleotides in a coil around it, and histones can be linked and grouped together, significantly compacting the DNA. Each DNA-wrapped histone is referred to as a nucleosome. Once the H1 protein bonds to a nucleosome it is then a chromatosome. Chromatsomes, in turn, wrap around each other to form a 30 nanometer wide chromatin fiber. Chromatin fiber is the highest level of compaction that exists at all times in the cell; even greater compaction exists during times of cell division.

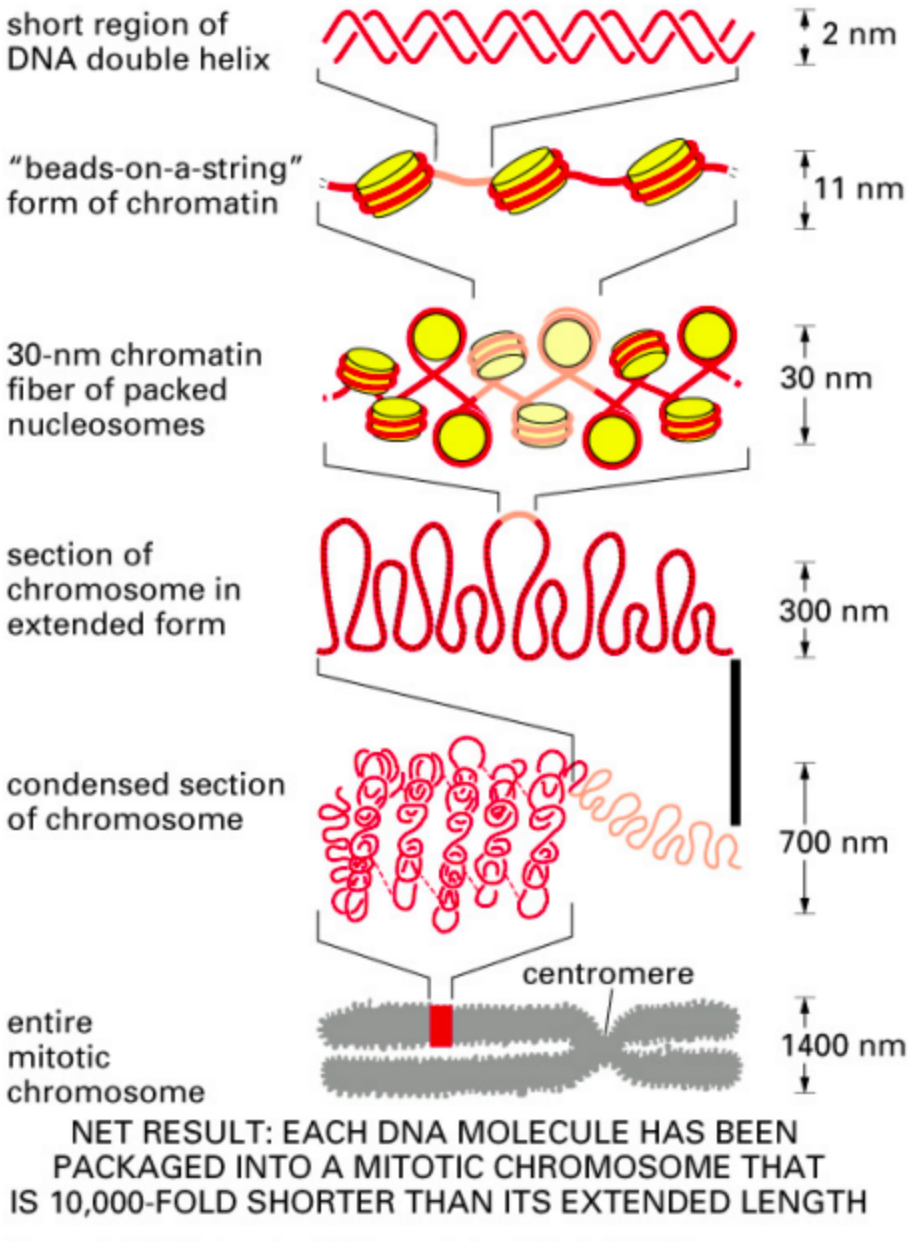

Figure 6. The increasing scale of DNA compaction, from histones to chromosomes, and a brief overview on the chromosomal structure. Note that the chromatosmes are referred to as nucleosomes, although the difference is minor.[11]

[11] Alberts, Bruce, et al. "Chapter 4, Figure 4.55." *Molecular Biology of the Cell, Fourth Edition.*

Regions that are highly coiled by histones are referred to as heterochromatin, while regions without much coiling are called euchromatin. Because the DNA in heterochromatin is very tightly wrapped around histones it is harder for sequencing machines to access, resulting in a read frequency bias towards regions of euchromatin.[12] Other areas that are difficult to access for sequencing technologies are the centromere and telomeres, regions of the chromosome that are located at the center and ends of the chromosome respectively. These regions generally contain a lot of heterochromatin and are very repetitive, suggesting that their main functions are to stabilize the DNA cell replication as compared to storing meaningful information. The term "chromosome" refers to contiguous DNA fragments within a cell. Some bacteria only feature one chromosome (which is circular), while one species of protist has a whopping 16,000 chromosomes, hinting that the ways in which DNA is stored is just as diverse as life itself.[13] Many commonly known organisms (some plants, nearly all animals, and most fungi) are considered diploid (also referred to as n=2) as they have two copies of each chromosome - that is, every gene that exists can be found on nearly identical chromosomes in each cell.

[12] Benjamini, Yuval, and Terence P Speed. "Summarizing and Correcting the GC Content Bias in High-Throughput Sequencing."
[13] Chen, Xiao. *The Architecture of a Scrambled Genome Reveals Massive Levels of ..."*

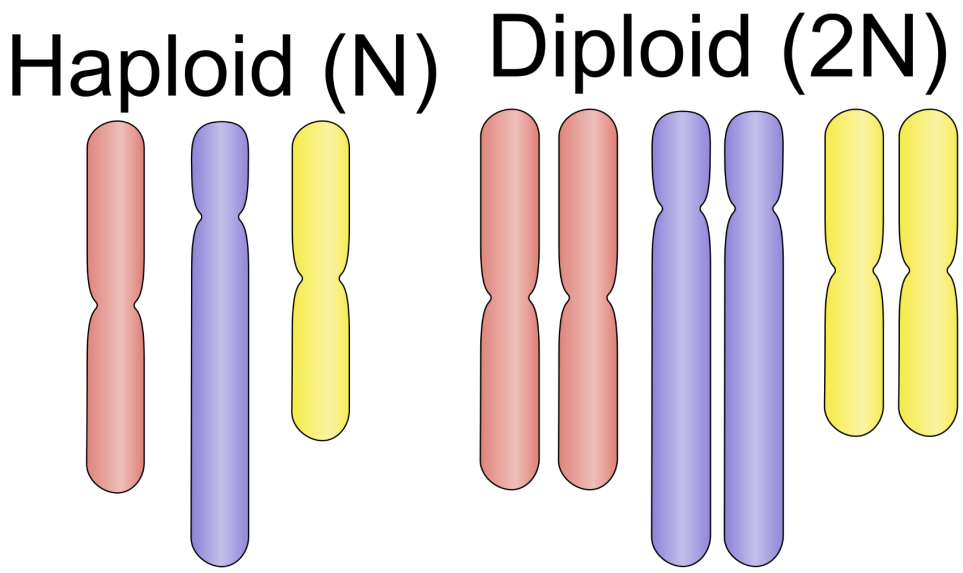

Figure 7. An example of two levels of ploidy: haploid (n) and diploid (2n).[14]

The differences in chromosomal copies are due to the fact that one chromosome is inherited from each parent and, since each individual is genetically distinct, there will be extremely slight differences along with extremely infrequent *de novo* (new) mutations that may have affected only one of the two chromosomes. Other ploidy numbers exist: n=1 is considered haploid, these organisms have only one parent. On the other hand, many plants are polyploid - that is, they have more than two copies of each chromosome, and sometimes many more. Strawberries, for instance, can have up to nine copies of each chromosome in the wild (this is referred to as nonaploidy, n=9).[15] The varying levels of ploidy offer a tradeoff from the lens of genome sequencing: the more copies of a gene there are, the more likely one is to have obtained a read from it, but it becomes harder to resolve exactly which copy of the chromosome that read belongs to. An additional possibility is that it actually belongs to both copies.

---

[14] Hamburg, E. "File:Haploid vs Diploid.svg." *Wikimedia Commons*, 10 May 2010.
The image was modified for page length such that, instead of being above or below one another, the two examples of ploidy appear side by side.

[15] T;, Hummer KE;Nathewet P;Yanagi. "Decaploidy in Fragaria Iturupensis (Rosaceae)."

While a codon may be the most basic unit through which information from DNA is relayed to RNA and proteins, the larger scale term "gene" is also very useful to know. Genes are regions of DNA that code for an RNA, which may or may not be translated to protein.[16] Previously it was thought protein was the only catalytic (biologically active) macromolecule in the central dogma of molecular biology, however many catalytic RNAs have been recently discovered. All genes share several key features, such as the promoter, the region where RNA polymerase binds and begins transcription to RNA. Since promoter sequences are remarkably similar throughout life, these are easy for genome assembly and analysis programs to recognize and flag as points of interest. Additionally, genes have stop codons, indicating the end of the region that is transcribed to RNA. Again these stop codons are easy for computers to recognize as only three exist (UGA, UAA, and UGA when represented as RNA), and they must occur in a predictable fashion along the DNA sequence. Two primary conditions exist: first, the stop codon must exist more than 300 but less than 3 million bases (3 Mbp) "downstream" of the promoter (towards the 3' direction).[17] Second, the stop codon must be available to read in the frame given by the promoter. The stop codon is available when it is in frame 1, and all nucleotides are being read in groups of three. If one or two extra nucleotides are needed to group the stop codon together, then it is in frame two or three and is considered a "closed" reading frame. A promoter followed by a stop codon in an appropriate window is referred to as an open reading frame.[18]

[16] Epp, Christopher D. "Definition of a Gene."
[17] RG;, Tennyson CN;Klamut HJ;Worton. "The Human Dystrophin Gene Requires 16 Hours to Be Transcribed and Is Cotranscriptionally Spliced."
[18] Shchelochkov, Oleg. "Open Reading Frame." *Genome.gov*

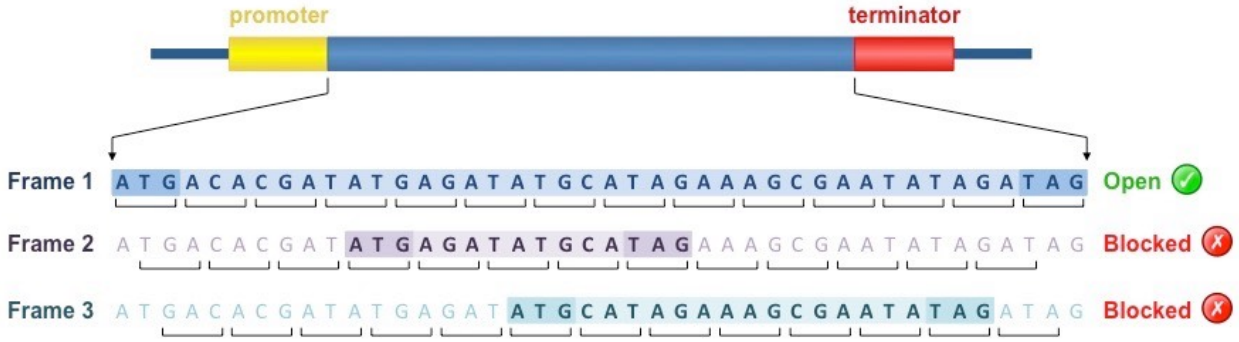

Figure 8. An open reading frame and two closed reading frames. Note that the sequence being given is the "sense" strand of DNA, so the codons are equivalent to the RNA ones previously mentioned other than T substituting for U.[19]

However, not all open reading frames are transcribed, complicating the process of true gene identification for genome assembly programs. Open reading frames that do not code for RNA are referred to as "pseudogenes" and are likely remnants of previously used genes. One subtle way to distinguish most pseudogenes from proper genes is that pseudogenes usually have somewhat degenerated promoters as they are no longer under selection pressure to be efficient at guiding transcription. However, due to the slight natural variation in working genes' promoters, this is not always a clear diagnostic of a pseudogene. Promoter variation is one of the modalities used to alter gene expression (transcription and translation). Certain promoter sequences favor transcription by favoring binding of RNA polymerase, the molecule that actually transcribes DNA into RNA, more favorably than others. Other methods of altering transcription levels include DNA methylation, where Cytosine has a methyl group attached to its 5-carbon. Methylated DNA is read less frequently than "normal" DNA and therefore this is a method of

[19] Cornell, Brent. "Gene Identification." *BioNinja*

down regulation, one of the two types of gene regulation present. The other, predictably, is up regulation, and an example of it is histone acetylation. Since DNA as a whole carries a negative charge (due to its phosphate groups), histone proteins carry a positive charge so that DNA will stay coiled around them. Acetyl groups, however, carry a negative charge, and acetylating histones neutralizes the positive charge they carry. Without the electrical attraction, DNA is more loosely coiled around the histone, and more accessible to RNA polymerase. This is why acetylation is a form of upregulation. Other forms of regulation include enhancers, sites where other proteins can bind to DNA and aid transcription, and inhibitors, other sequences where proteins can bind and inhibit RNA from transcribing the given gene. Since enhancers and inhibitors are DNA sequences, they are a part of the genome and many genome analysis tools attempt to predict their location and exactly which genes they regulate. Due to the extreme level of compaction in chromosomes, sequences that are very far apart on a linear model of DNA may physically be adjacent in the cell. Therefore, this process can be quite complex and may require an understanding of similar enhancers in other species. Inhibitors, conversely, are usually located near their target genes, simplifying their identification process for computer algorithms. Gene regulation is the mechanism through which an eye cell and a bone cell differ drastically in shape and function despite having the same genome. Additionally, it is responsible for some of the differences between individuals, although other factors influence this variation more.

An Introduction to Genomics

Since every living thing has a distinct genetic code, each and every lifeform on this planet has its own genome that encodes exactly which RNA and proteins it expresses. However, for practicality, most experiments in molecular biology only require one "reference" genome per species from which the cellular and metabolic characteristics of any member of that species can be inferred and manipulated with. This relative similarity between individuals in a species is one the genome's key advantages over the transcriptome and the proteome. The transcriptome is the total sum of all RNA expressed. The proteome, meanwhile, is the sum of all of the proteins present in an organism or species. While the transcriptome and proteome are also of interest to molecular biologists, only the genome is constant throughout an organism's life. This has multiple benefits: first, it is the only one of the three representative of the information passed down between generations. Additionally, it is also simpler to study as the transcriptome and proteome can change rapidly, and not all of the organism's possible RNA and protein is present in any one cell and at any one time. Therefore, to obtain a transcriptome or proteome, multiple rounds of sample extraction have to occur, as compared to once to obtain the genome. Lastly, DNA's larger size and double helix (as compared to RNA, which only has a single helix) render it more resilient while doing lab work.

With the genome being an attractive candidate to understand lifeforms it should come as no surprise that many techniques have been developed in the past 50 years to attempt to retrieve a reference genome for every species with the highest possible levels of completion and accuracy. To this end, the vertebrate genomes project has an ambitious goal: sequence the genome of every vertebrate.[20] Several key metrics have been developed to assess the quality of genome assembly, with most involving either completion or accuracy given their importance to

[20] Springer Nature Limited. "The Vertebrate Genomes Project."

any sequencing project. For genomes where a reference previously exists, it is possible to measure completeness as a fraction of the genome recovered in the new assembly. However, the first time a species' genome is sequenced, this is not possible. Thus, a *de novo* genome assembly is required. In this case it is not possible to measure completeness as a fraction of the genome size as the true size is not known. While sometimes it is possible to have an estimate on the full genome size either due to inference from a closely related species or from physically obtaining the mass of the DNA per cell in a sample, usually a different statistic is used: N50.[21] To obtain a genome's N50, the largest contigs (reads that have contiguously been attached together) must be added together until they sum to at least 50% of the reported genome assembly length. After this size is achieved, report the length of the most recently added contig.

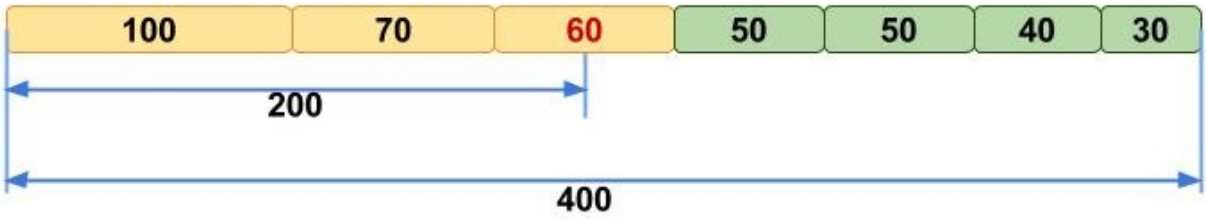

Figure 9. An abstract representation of a genome, with its contigs sorted by size. Since the genome's length is 400, the size needed to achieve 50% of it is 200. The read that takes us over 200 is of length 60, so the genome's N50 is 60.[22]

N50 is a very valuable metric even when a genome assembly has a currently existing reference - this is because it also gives information about how fragmented the newly assembled genome is. Reducing fragmentation is critical as it allows molecular biologists to determine where genes are located in relation to each other. If the genome assembly is contiguous enough,

[21] Heslop-Harrison, J S. "Crop Improvement: Plant Genomes."
[22] Videvall, Elin. "What's N50?"

it may be possible to understand which chromosome each gene lies on. If the position of genes on chromosomes are known, then it is possible to create a linkage map which displays this information. Since chromosomes are the basic unit passed down between generations, this can have very important implications for understanding heritability patterns of certain traits.[23] N50 is highly dependent on "coverage." Coverage can be calculated by taking the length of all sequencing reads obtained and dividing it by the length of the genome. For a *de novo* assembly, the reported genome size is used, even though it is understood to be incomplete.[24] High levels of coverage are recommended as, although reads are semi randomly distributed along the genome, gaps are likely to be present at low coverages. This is both due to randomly missing regions and bias away from areas with high GC content and/or high levels of heterochromatin. However, the exact levels of coverage needed are not currently known, and this was one of the aims of our study.

Additionally, coverage can affect accuracy. The more reads obtained to represent a single base in the original genome sequence, the more likely it is that any sequencing error present on a single read will be corrected by other reads. Since the twenty amino acids are very diverse in their structure and function, a single error can have profound implications for the interpretation of a genome. Errors fall into two main categories: substitution errors, where a certain number of bases are swapped out for an equal number, or "indels", where the number reported is different from the number that actually exists. Reflecting this, the term indel is an abbreviation of "insert/delete." Because indels shift reading frames, and many genome analysis programs rely on open reading frames to identify gene sequences, a single indel can have potentially steep consequences on the quality of the contig affected. However, genomes are massive. The smallest

[23] Vidal, Adrien, et al. "SESAM: Software for Automatic Construction of Order-Robust Linkage Maps - BMC Bioinformatics."
[24] Zimin, Aleksey. "Sequencing and Assembly of the 22-Gb Loblolly Pine Genome."

genomes belong to viruses and can be as small as a few thousand bases, while the largest genomes approach a terabase (1 trillion bases).[25] Due to this, indels are inevitable, and must be dealt with as much as possible.

The myriad of genome sequencing technologies that have been developed have tried to maximize completion of and minimize accuracy, although, tradeoffs always exist. The very first method of genome sequencing used, Sanger sequencing, was developed in 1977 by Frederick Sanger.[26] Sanger sequencing involved extracting DNA, breaking the double helix, and then extending new strands by adding a small portion of dideoxy nucleotides to the reaction mixture, in addition to the naturally occurring 2-D-deoxyribose. Since these dideoxy nucleotides do not have the hydroxyl group required for the addition of the next incoming nucleotide, each strand that has a dideoxy nucleotide attached terminates. Since this happens at random lengths along the DNA, and each of the four nucleotides is added in a repeated fashion, it then becomes possible to determine which nucleotide is added at each position along the read. This makes it possible to determine the read sequences, and slowly assemble contigs. While cumbersome, it was a starting point for the field of genome sequencing and had read lengths of 900bp and an error rate of .001%, metrics that would not be improved upon for quite some time.[27,28]

Sanger Sequencing, however, was prohibitively expensive for all but a few select genomes. The first genome ever sequenced was the bacteriophage (virus) fx174, with a genome of only 5,368bp.[29] After this initial effort many of the next organisms chosen were the "model" organisms, species which for one reason or another have particular scientific utility. Some commonly known model organisms are *Rattus norvegicus* (the lab rat) and *Drosophila*

[25] Karami, Ali. "Largest and Smallest Genome in the World - Researchgate."
[26] "Timeline: History of Genomics." *Your Genome*
[27] Ibid
[28] Cheng, Chu, and Pengfeng Xiao. "Evaluation of the Correctable Decoding Sequencing as a New Powerful Strategy for DNA Sequencing."
[29] "Genome Sequencing: A History."

melanogaster (the fruit fly), while lesser known ones include *Saccharomyces cerevisiae* (a species of fungus) and *Caenorhabditis elegans* (a nematode).[30] Given its extreme relevance, the first draft of the human genome was published in 2001 after a decade and $300 million were spent on the effort. Currently, however, sequencing a patient's individual genome has become a common practice given its relatively low cost. As of April 2023, it costs under $1,000 to fully sequence a human genome.[31] This, however is rare now as the human genome has been fully sequenced many times, so reference sequencing is commonplace instead as it is even more cost effective. Regardless, the cost of *de novo* sequencing has fallen low enough that stories exist of undergraduate labs and even individuals sequencing a species of personal interest. What has allowed this drastic shift in cost?

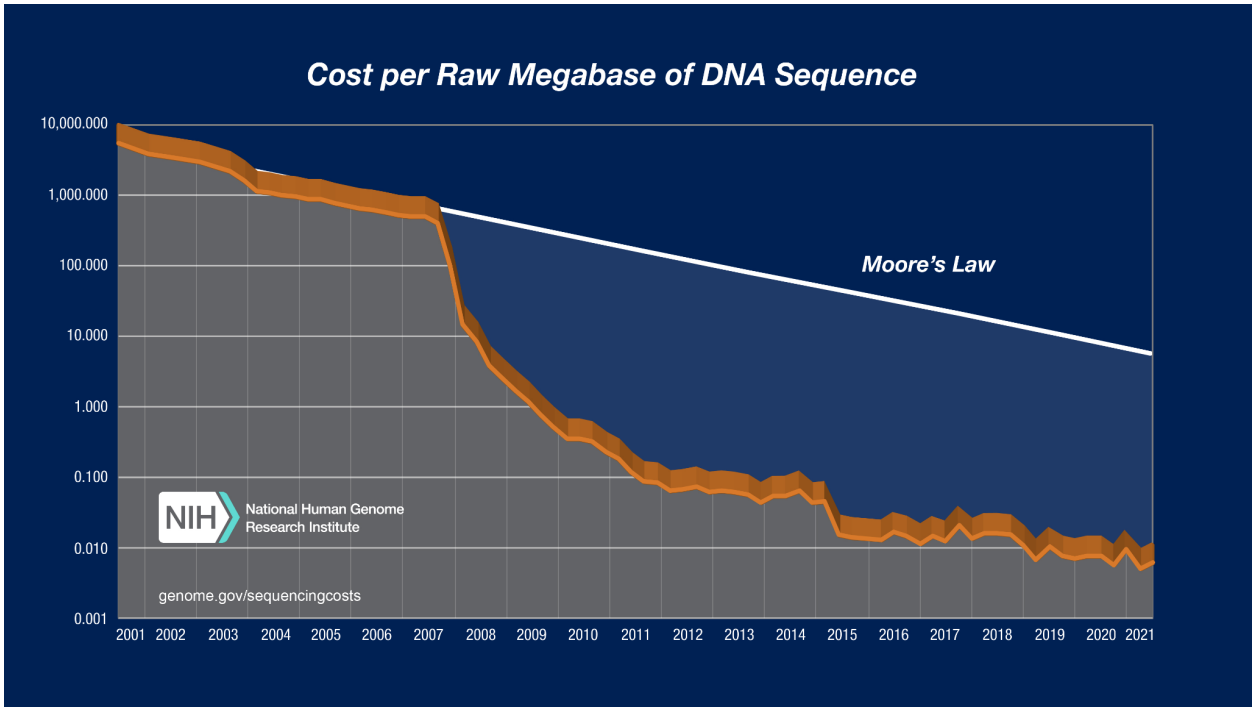

Figure 10. Cost of sequencing per megabase of genome. For reference, the human genome is 3 gigabases.[32]

[30] "Model Organism Sharing Policy." *National Institutes of Health*
[31] Wetterstrand, Kris A. "DNA Sequencing Costs: Data."
[32] Ibid

Throughout the end of the 20th century, Sanger sequencing gradually became more efficient, but the first significant leap came in 2005 when "next-generation" sequence reads were released by Illumina, a biotech company.[33] This new technology works by instead using fluorescent dideoxynucleotides in the DNA addition step and photographing between each addition, resulting in a drastic contraction of the space and resources required to generate each read. Instead of each read taking up hundreds of lanes on a polyacrylamide gel with each lane being a few centimeters long, 40 million wells each with 1000 copies of a specific read fit on a "flow cell" (sequencing chip) that could fit in the palm of one's hand. This increased sequencing output by many orders of magnitude, and resultantly cost fell five orders of magnitude.[34] Illumina reads, however, left a lot to be desired: while their error rate was low, it was higher than that of Sanger reads (~.1% vs .001%). This however, was easily overcome by the increase in coverage, but the issue of read length could not be overcome so easily. In contrast to Sanger reads almost reaching a kilobase (kbp), the first Illumina reads were only 35 nucleotides long. While this is sufficient in most gene coding portions of DNA, repetitive elements (where a short pattern, such as ATATAT… repeats) elsewhere in the genome put a serious cap on the N50 achievable as it became impossible to determine the exact position in the genome those sequence reads originated.

To this end, yet another technology arrived in the mid 2010s: continuous long reads (CLRs) by Pacific Biosciences (Pacbio). Instead of using flow cells and wells, CLRs operate by taking advantage of the molecules behind natural DNA replication (primarily DNA polymerase) to add the dideoxynucleotides. This allowed for reads over 15kbp long, however, the error rate

[33] Morozova, Olena, and Marco A Marra. "Applications of next-Generation Sequencing Technologies in Functional Genomics."
[34] Cheng, Chu, and Pengfeng Xiao. "Evaluation of the Correctable Decoding Sequencing as a New Powerful Strategy for DNA Sequencing."

suffered even more: CLRs have an error rate of up to 13%. Even worse, while Illumina reads have primarily substitution errors, Pacbio reads primarily have indels.[35] A frame shift once in every eight nucleotides is extremely challenging to operate with, and as such, the technology needed significant refinement before becoming truly viable. Pacbio then released High-Fidelity (HiFi) reads to overcome the error rate of CLRS by taking advantage of their circular nature. By having slightly shorter sequences, it becomes possible to ensure that DNA polymerase makes multiple trips around the circle. This ensures that every sequenced spot has high "coverage" (although this is slightly different from the technical use of the word). Although each subread (pass around the circle) still has up to a 13% error rate, the other subreads correct most of these. Therefore, Hifi reads have an approximately 1% error rate.[36] In October 2022, PacBio announced the Revio sequencing machine, which has made HiFi reads 15 times cheaper. While they were previously cost prohibitive for medium to large size eukaryotic genomes, HiFi reads are rapidly becoming the standard for genome assembly.

As Pacbio was making significant improvements, an Oxford University startup (Oxford Nanopore Technologies) released a new read type in 2014: nanopore reads. Nanopore reads are unique as, unlike every sequencing strategy discussed so far, they do not involve dideoxynucleotides- instead, they operate by running a strand of DNA through a protein on an artificial membrane. As the DNA passes through this membrane is it possible to record the difference in charge. Since each nucleotide's molecular formula is slightly different, they affect the charge differently, and it is possible to determine the read sequence. Nanopore reads are currently a top notch technology, with read lengths reaching into the megabases. At this scale, they can easily resolve even the longest repetitive regions, and can even be used to cross

[35] Logsdon, Glennis A, et al. "Long-Read Human Genome Sequencing and Its Applications."
[36] Ibid

significant portions of some chromosomes.[37] However, their error rates are as high as 13%, similar to Pacbio's CLRs.

Therefore, cutting edge genome assembly techniques combine both HiFi and Nanopore reads, taking advantage of the former's accuracy and the latter's length.[38] The process of combining multiple read types is referred to as hybrid assembly, and with these read types it is currently unclear how to maximize information gained per dollar spent. Therefore, we were motivated to investigate this question, especially as it pertains to large genomes. The only relevant studies have benchmarked this work in bacteria, which have orders of magnitude smaller genomes than the ones on the frontier of genome science.[39] This work becomes very computational after putting the tissue sample through the sequencing machine, highlighting the interdisciplinary nature of the field.

Computational Methods

Understanding Algorithms and Runtime.

While the average person may think of computer science either as hackers sitting alone behind a screen, or a bunch of tech billionaires, at its core computer science is all about automation: understanding a problem well enough such that its solution can be enumerated through a series of independently reproducible steps. However, at times, it can be proven that no such series of steps exist due to the very nature of the problem itself - these are referred to as the "undecidable" problems. Understanding which problems are undecidable is just as important as

[37] Warburton, Peter E, et al. "Analysis of the Largest Tandemly Repeated DNA Families in the Human Genome - BMC Genomics."
[38] `Wang, Jeremy R. "Polishing De Novo Nanopore Assemblies of Bacteria and Eukaryotes With FMLRC2 ."
[39] Wick, Ryan R., et al. "Assembling the Perfect Bacterial Genome Using Oxford Nanopore and Illumina Sequencing."

knowing how to code, as one may accidentally try to write an algorithm that is impossible to finish. Additionally, speed is a critical consideration as a solution is worthless if it takes 5,000 years to get the answer to today's problems. In a field such as bioinformatics, where cutting edge technology becomes obsolete in 5 years, and with such massive datasets, speed is an absolutely critical consideration when programming.

The core of any program is the algorithm. An algorithm is a series of discrete, repeatable steps taken to reach a solution. Algorithms typically have input, abstractly referred to as a "string". Strings are sequences of characters such as "Hello world!" and "Nice weather today, isn't it?". Sometimes it is convenient to limit the set of allowable characters in a string. When we do this, that set of characters is referred to as its alphabet. If the computer is attempting to parse English text, then the alphabet is the English alphabet, plus the punctuation symbols. The alphabet DNA operates over, from a computational perspective, are the four nitrogenous bases. Data in computers is fundamentally stored in binary logic represented by the presence or absence of a charge on a piece of electronic hardware. The two characters in the binary alphabet are "0" (false) and "1" (true). While seemingly very minimal, it is possible to prove that this alphabet can represent any other.

Therefore, most abstract models of computation describe machines that operate over the binary alphabet. Since this is the case, the input string *"w"* is a sequence of 0s and 1s that abstract models of computation operate on. These abstract models are referred to as "Turing Machines" (TM) after their inventor Alan Turing, who theorized their existence in 1930. These machines operate on any given string *w* by either accepting or rejecting it. The Church-Turing thesis showed this very simple decision can be representative of any operation of any

deterministic system. Since computers are deterministic, any computer on earth can be represented as a TM.

With the great diversity in computer programs that exist, ranging from Google search to stock market modeling programs to genome assemblers, it seems natural to attempt to classify certain problems by their difficulty. For example, is solving a game of sudoku or chess harder? Luckily, there is an objective way to answer this question with our abstract model of computation. We can convert any program input into a binary string w of length n (so w has n characters). Every text, mouseclick, key typed, image given, and more can ultimately be represented as a string of 0s and 1s of a certain length.

We can then run the algorithm we are considering on w and see how many steps the program takes to decide whether to accept or reject it. Since an algorithm has discrete steps, this will be a specific number. Since any reasonable computer program has more than one valid input w (there are multiple options for google search, likewise an assembler should work on any genome given), we can then express the number of steps taken as a function of n (the length of any particular input w). This function is referred to as the algorithm's runtime.

To express runtime, we use Big O notation in terms of n. Since we care about runtime most on the largest input strings (since they take the most amount of time to run), we care about how this function behaves as n approaches infinity. Therefore, we take the mathematical limit of our runtime function and attempt to find another function that is asymptotically bound to it. In other words, we find the function that most closely represents our function as n approaches infinity. If no perfect approximation is possible, we pick the worst case possible for our algorithm (the maximum number of steps it takes on any input) and we pick a function that grows at a faster rate than the runtime function. This way, we are ensured our algorithm will

decide on any string *w* in no more steps than the amount given by the runtime function. When creating the runtime function for our algorithm, we look at how long each step takes in terms of *n* and add the steps together. As we do this, we drop any constants and lower growing terms (for example, drop *n* if n^2 has already been seen). This is to make the runtime function simpler, as $O(n^3)$ is much more easily understood than $O(8n^3+2n^2+31n+3)$. Additionally, it allows us to classify problems of similar difficulty together into "complexity classes." The jumps in difficulty between complexity classes are much, much larger than any differences in difficulty inside of a complexity class.

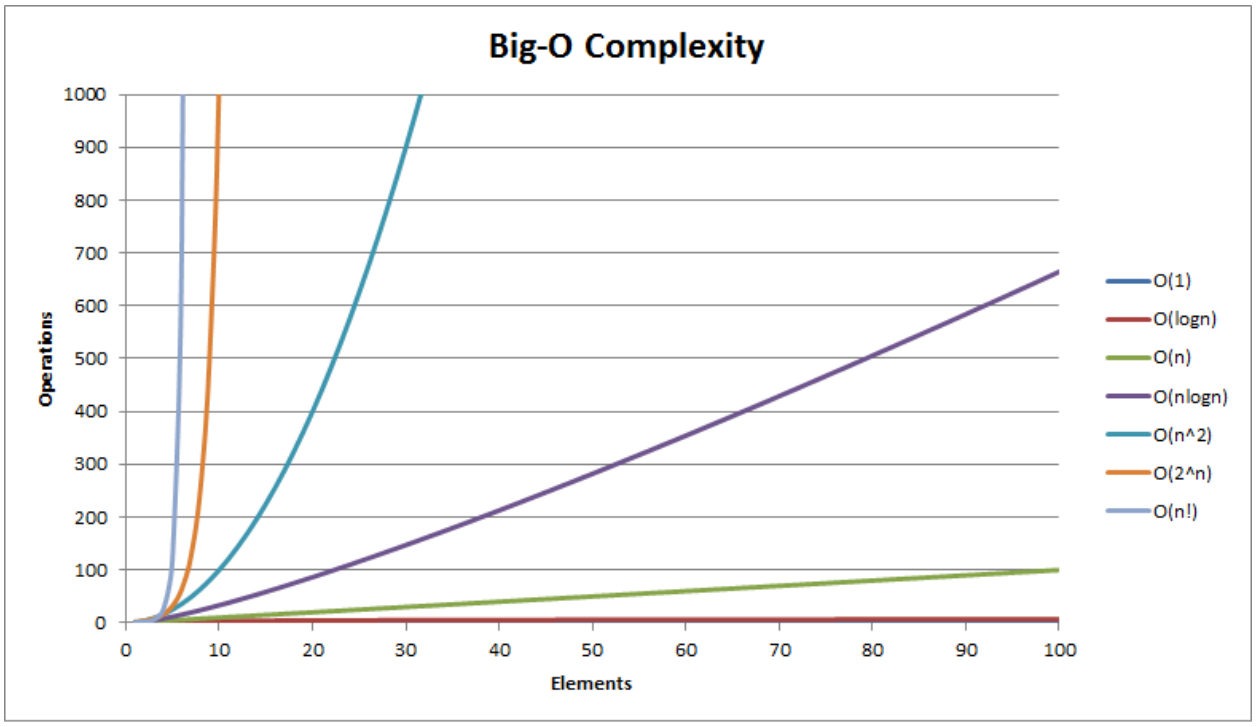

Figure 11. A graph with examples of common runtimes for computer programs, along with the prohibitive exponential runtime of $O(2^n)$ and even worse factorial runtime $O(n!)$.[40]

The most important complexity class to understand is "P." P refers to the polynomial time problems, or the problems which can be solved by a computer in a polynomial number of steps

[40] R, varun N. "Big O Cheatsheet - Data Structures and Algorithms with Thier Complexities ."

relative to the input size (or faster, such as a logarithmic or constant number of steps). Polynomial runtime can be expressed as $O(n^k)$, where k is any integer. Problems in P are typically referred to as "easy" for computers to solve, as they require relatively few steps per character in *w*. Some examples include searching through a list to find an item of interest, arithmetic, and (important for bioinformatics) sequence alignment. This can be used to compare two sequences of nucleotides, allowing for information to be obtained about genomes. The level of similarity seen can show if one has any mutations relative to another. This can hint at how closely related the two samples are, or if one is likely to have any genetic diseases.

Because the raw data for genome assembly is a collection of sequencing reads, it is a sequence matching problem as well. However, it is not as easy as aligning sequences of DNA. This is because, instead of comparing two genome (or individual gene) sequences, there are a variable amount of reads that must all be compared with each other in an attempt to piece together the newly assembled genome. The naïve approach to solve this problem can be given with the following algorithm *R*: start with a sequencing read *A* and compare every remaining read in the input file to it. If they match, line up the overlap and "concatenate" them (add them together). Otherwise, do nothing and move onto the next read. However, *R* will run an exponential number of times in terms of *n*. Every read added requires more and more decisions than the one previously added, and as a result, this approach is prohibitive for computers. In theoretical notation, $O(2^n) > O(n^k)$, and as a result, this problem is not in P. However, there are many programs that assemble genomes. How is this possible?

Dynamic Programming

The exponential nature of our algorithm derives from the fact that, after each pair of reads *(A,B)* that do not match, *R* discards the result. Since this preserves no information about the lack of a match between *A* and a non matching read *B*, any concatenations *AC* that occur will attempt again to find a partial match between *A* and *B* (as *AC* will be paired with *B*). This is inefficient as the *A* portion of *AC* will certainly not align with *B*. Regardless, *R* still checks if it will. Therefore, if we could store the negative result, an algorithm *R'* could be exponentially faster and therefore could solve problems easily for computers. This is possible via the technique of dynamic programming.

Dynamic programming requires an algorithm to store results in memory as it operates. While this has the advantage of an exponential increase in speed, the only drawback is that computers optimized for genome assembly require enormous amounts of RAM (computer memory). This is especially true for large genomes, and rapid advances in computer hardware have allowed for genome assembly to become more accessible as of late.

To actually implement the fast algorithm *R'* one must select a "data structure" to represent the reads along with information about how they match, if at all. This is frequently done with a "graph." Despite the name, these are not related to graphs in other disciplines (they feature no axes). Graphs in computer science instead have nodes and edges, each of which can be labeled individually. Nodes can only be attached to edges, and likewise edges can only connect to nodes. While there is no limit on the number of edges a node can be connected to, each edge must connect to exactly two nodes. Additionally, graphs can be "directed." A directed graph features edges with directionality where one node functions as a starting node and one node functions as a destination node.

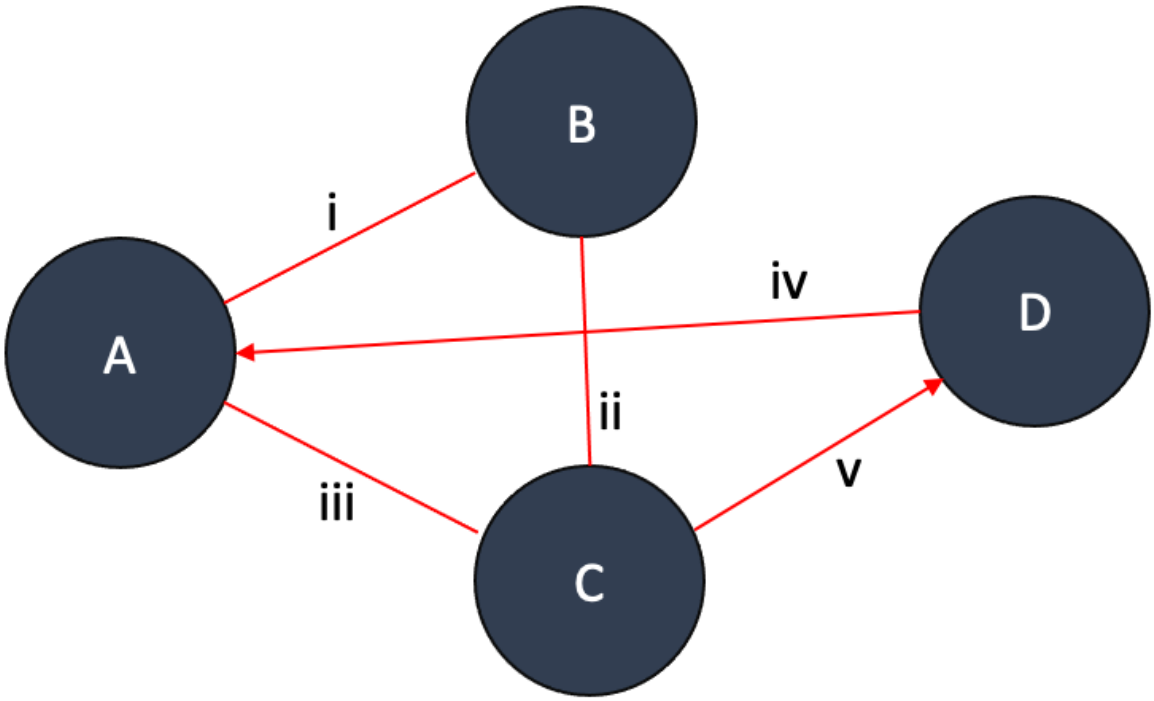

Figure 12. An example graph *G,* with nodes *A, B, C, D,* and edges *i, ii, iii, iv, v*. Note that *iv* and *v* are directed edges. Figure generated by the author.

In genome assembly, de Bruijn graphs are used, named after their creator Nicolaas de Bruijn.[41] In a de Bruijn graph, each edge is labeled with a "k-mer" (a sequence read of length *k*). The nodes, meanwhile, are labeled with k-n-mers (k minus n mers), where n is an integer. In any particular de Bruijn graph, n is constant, but it may change from graph to graph. Therefore, each node is labeled with a read "substring" (portion of the read). For example, with a sequence read "ATGATC" and $n = 2$, the nodes would be labeled "ATGA" and "GATC", and the edge in

[41] Compeau, Phillip E C. "How to Apply De Bruijn Graphs to Genome Assembly - Eaton-Lab.org."

between would be “ATGATC.” Additionally, the edge would be directed, with the first portion of the read functioning as the start node and the second portion being the destination node.

To store information on how reads overlap, we check again if reads *A* and *B* match. First, we create two nodes (again, k-n-mers) for each. One node in each stores the first k-n-mer, and a directed edge runs from this node to the second node, which stores the latter k-n-mer. Then, we check if the first node in *B* matches either the first node in *A* or the second node. If it matches, “merge” the nodes by deleting one and moving the edge that started from the deleted one onto the preserved one. After observing this, one may notice that n can be at most half the length of the k-mer, rounded down, while still overlapping with the other node. If the nodes did not overlap, then the nucleotides in the middle would never be represented in a node, and their information would be lost when trying to “match” nodes and assemble the genome.

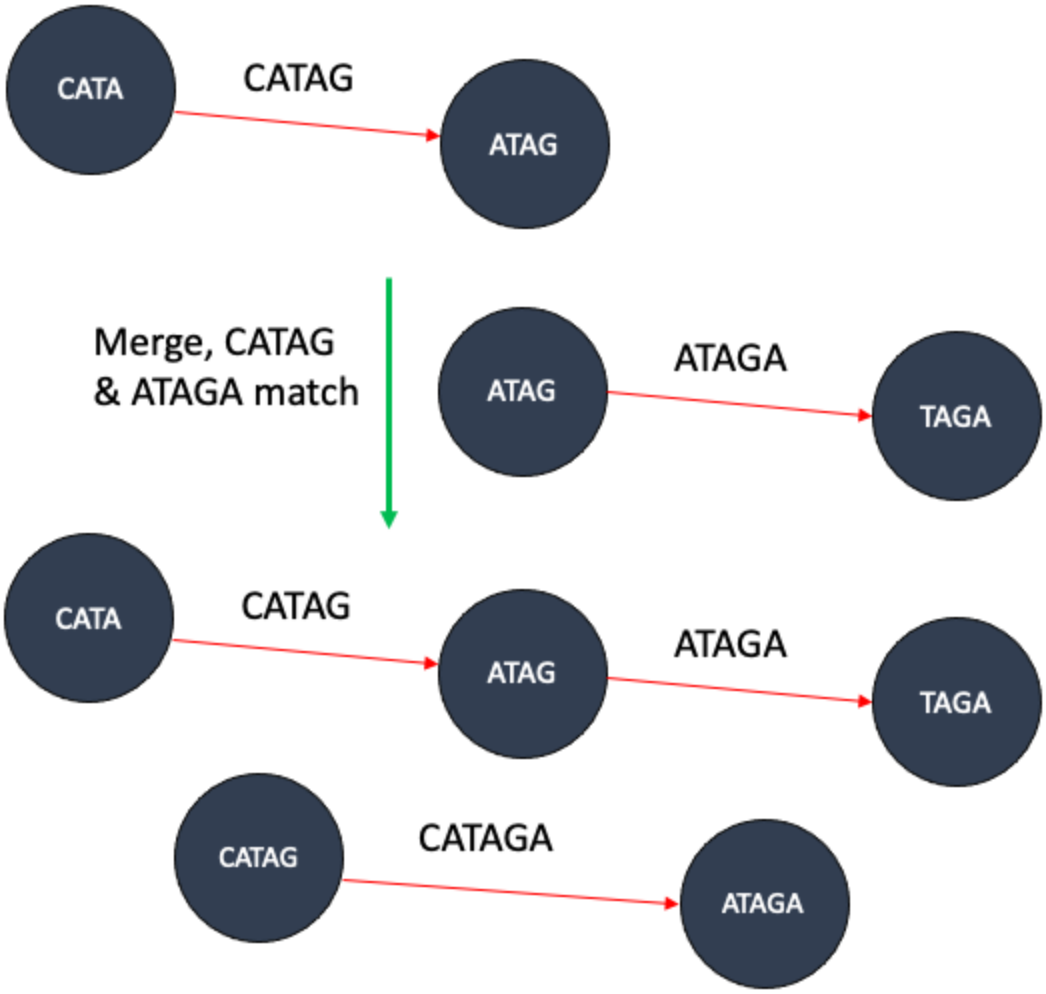

Figure 13. An example of merging nodes in a de Bruijn graph. Figure generated by the author.

If the reads do not match at all, we preserve all of the nodes. This way, a third read *C* can be compared against both *A* and *B*. This time, however, we know and have recorded that *A* and *B* do not match, so if *A* and *C* match *C* will simply be concatenated (merged) with *A* where appropriate. No further test of *AC* against *B* is needed, showing the efficiency of this approach. While this is the core of a genome assembler, other minor algorithmic components are needed.

First, due to the antiparallel strands of DNA, one must convert a read into its reverse complement and also check if that sequence is a valid match. Second, there must be some level of error tolerance as each read is extremely likely to have at least one sequencing error (and likely many). If there was zero error tolerance, "...TGCATCATGCTAGCTACGTATTCGTCATCGAT" would not match with "TGCATCATGCTAGCTACGTGTTCGTCATCGAT…" due to their 1 base pair difference. However, it is extremely likely that these reads are matching, as the chance of randomly having 31 identical bases is $1/4^{31}$, or approximately one in five quintillion. This is approximately the same odds as picking a perfect March Madness bracket, or three million times less likely than randomly picking a tree anywhere on earth and having another person guess which tree correctly on the first try.[42] Another consideration involves ploidy as the genome in question may have multiple copies of the same chromosome. If this is the case, both sequences may actually be correct. So, most genome assemblers have to account for slight differences in overlapping sequences, and the level of fault tolerance is usually up to the user. Additionally, most have options to treat substitution errors differently from indels.

These techniques can assemble contigs well, and sometimes it is possible to group contigs into "scaffolds." Scaffolds are contigs that are known to be near each other, but it may be impossible to understand exactly what lies in between them. This can happen either with "paired end" reads, where one has information that two DNA sequences are nearby each other, but with no data on the middle nucleotides, or if the de Bruijn graph is ambiguous. Ambiguity can arise when the path from source nodes to destination nodes branches. These branches may or may not rejoin, complicating interpretation of the original sequence while still indicating relative

[42] NCAA.com, Daniel Wilco |. "The Absurd Odds of a Perfect NCAA Bracket." *NCAA.com*, NCAA.com, 16 Mar. 2023

proximity on the genome. The other source of ambiguity is with "cycles": regions where a series of nodes and directed edges repeat. While one may think that it could be possible to keep track of the number of times edges that make up the cycle are seen, and then repeat the loop that many times to estimate the length of the repetitive DNA, the issue with this approach is that coverage is not universal across the genome. Thus, the cycle is likely to have more or fewer reads than the actual DNA sequence. If this is the case, then the estimated size of the repetitive element is incorrect. Therefore, the only option is to leave the cycle in the de Bruijn graph.

The previously described techniques are sufficient to assemble a genome when only one read type is being considered. However, due to the accuracy of HiFi reads and the length of nanopore reads, "hybrid" assembly using both read types is a very attractive option for achieving very complete and accurate assemblies. At its core, a hybrid assembler must merge shorter strings with a lower error rate with longer strings with a higher error rate.

Hybrid Genome Assembly

The intuitive approach to doing this is to use the HiFi reads to correct the nanopore reads. This way, one has long, accurate reads, which are then ideal for use in the de Bruijn graph. Such an approach is implemented by FMLRC2, an algorithm that uses a de Bruijn graph to map short reads onto the long reads (FMLRC2 was originally intended to work with Illumina short reads and CLR long reads). FMLRC2 is the second version of FMLRC, which was published by Jeremy Wang, et al in 2018. The acronym stands for FM-index Long Read Correction. An FM-index is a string index that allows for faster than linear, $< O(n)$, queries (lookups) of particular DNA sequences that are then mapped with two de Bruijn graphs to the long reads. Once they are mapped, the long reads can be corrected, and a simple genome assembler can be

used. Therefore, this algorithm is extremely efficient, faster by over an order of magnitude in comparison to other read correction softwares.[43]

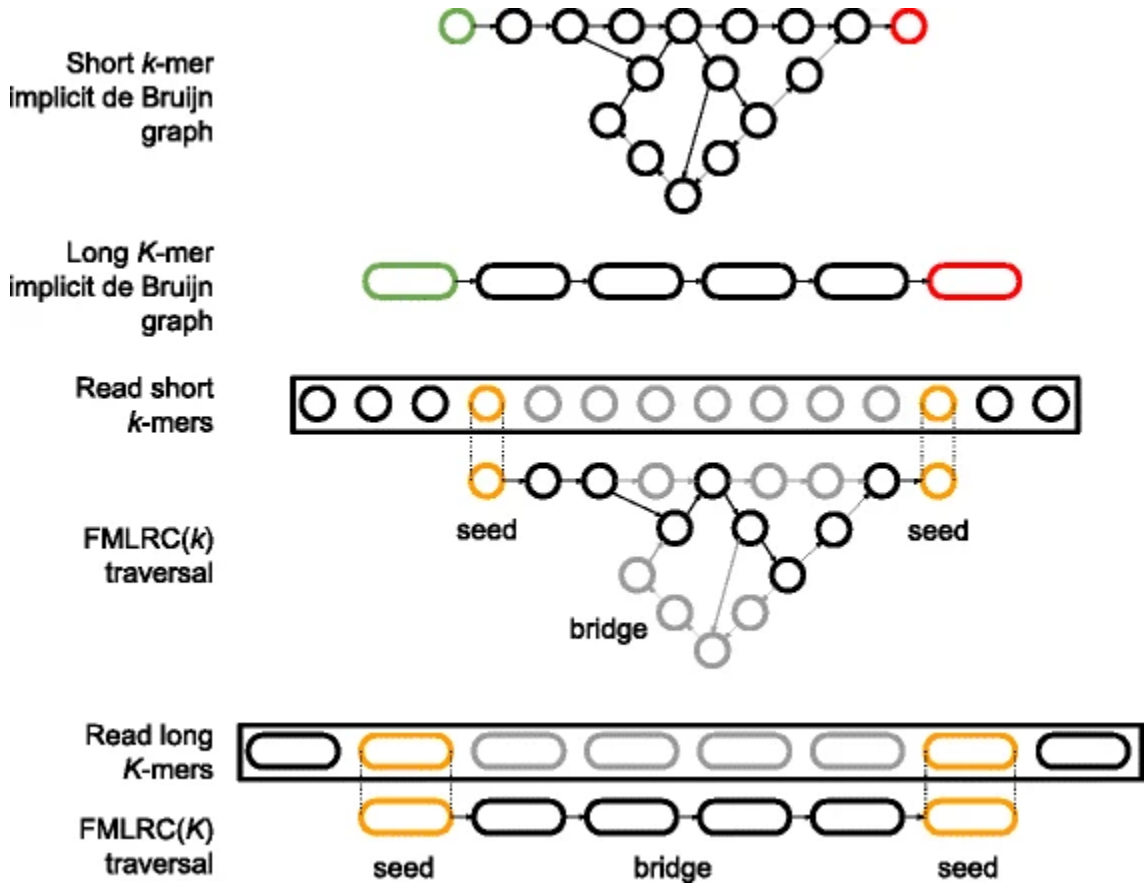

Figure 14. An outline of the FMLRC2 read correction workflow. The de Bruijn graphs are referred to as implicit as they exist within the FM-index. The traversal with a smaller k value allows for quick resolution of easy to infer sequences, such as gene coding regions, while long K-mers allow for cycle resolution.[44]

Additionally, another possible approach is to assemble the "easy" portions of the genome to the highest possible accuracy with the HiFi reads and save the nanopore reads for the merging of contigs. This strategy is how LongStich assembles genomes. Longstitch was developed by

[43] `Wang, Jeremy R. "Polishing De Novo Nanopore Assemblies of Bacteria and Eukaryotes With FMLRC2 ."
[44] Ibid

Lauren Coombe, et al in 2021, with the aim of producing high quality genomes from a mix of short and long reads. One advantage of LongStitch's method (as compared to FMLRC2) is that it may avoid overcorrection bias. If only a few short reads are available to correct similar but subtly different sequences present on long reads, it is possible for FMLRC2 to incorrectly infer the deviations in the long reads as sequencing error. LongStitch, meanwhile, would preserve these deviations as the short reads are assembled first.

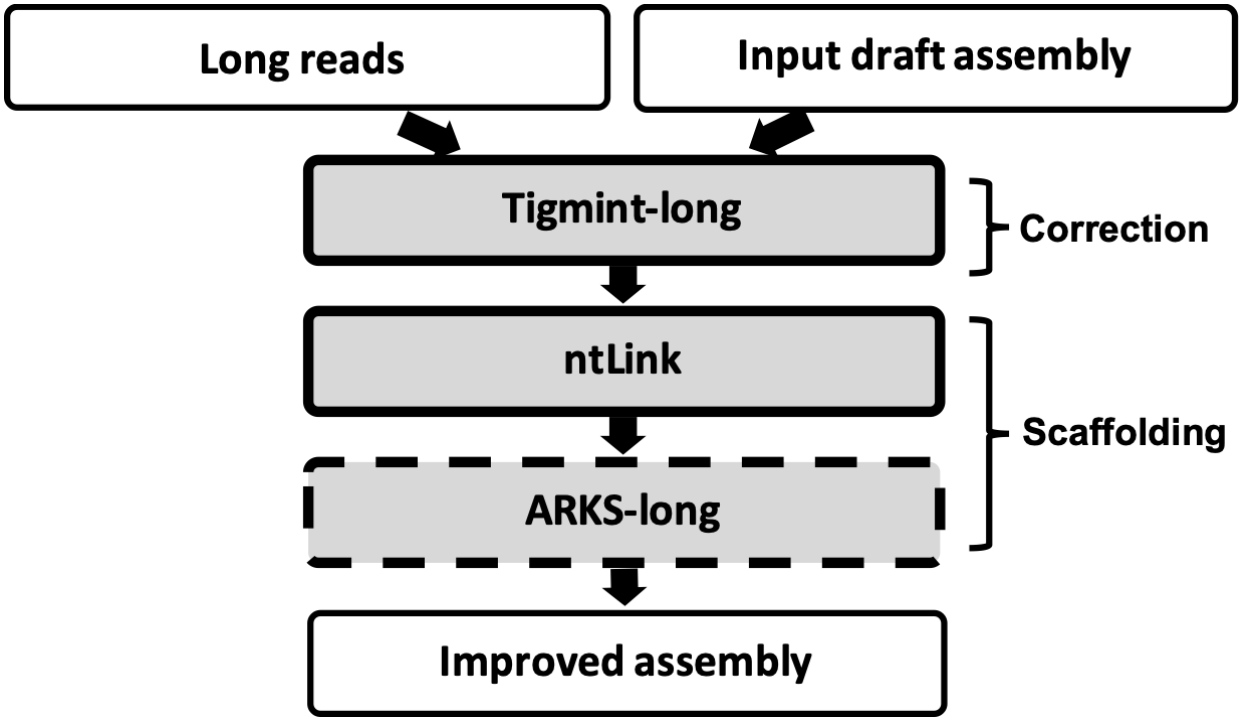

Figure 15. An overview of the LongStitch pipeline. While ARKS-long is optional, we included it in our study.[45]

[45] Coombe, Lauren. "BCGSC/Longstitch: Correct and Scaffold Assemblies Using Long Reads." *GitHub*, 2020, https://github.com/bcgsc/LongStitch.

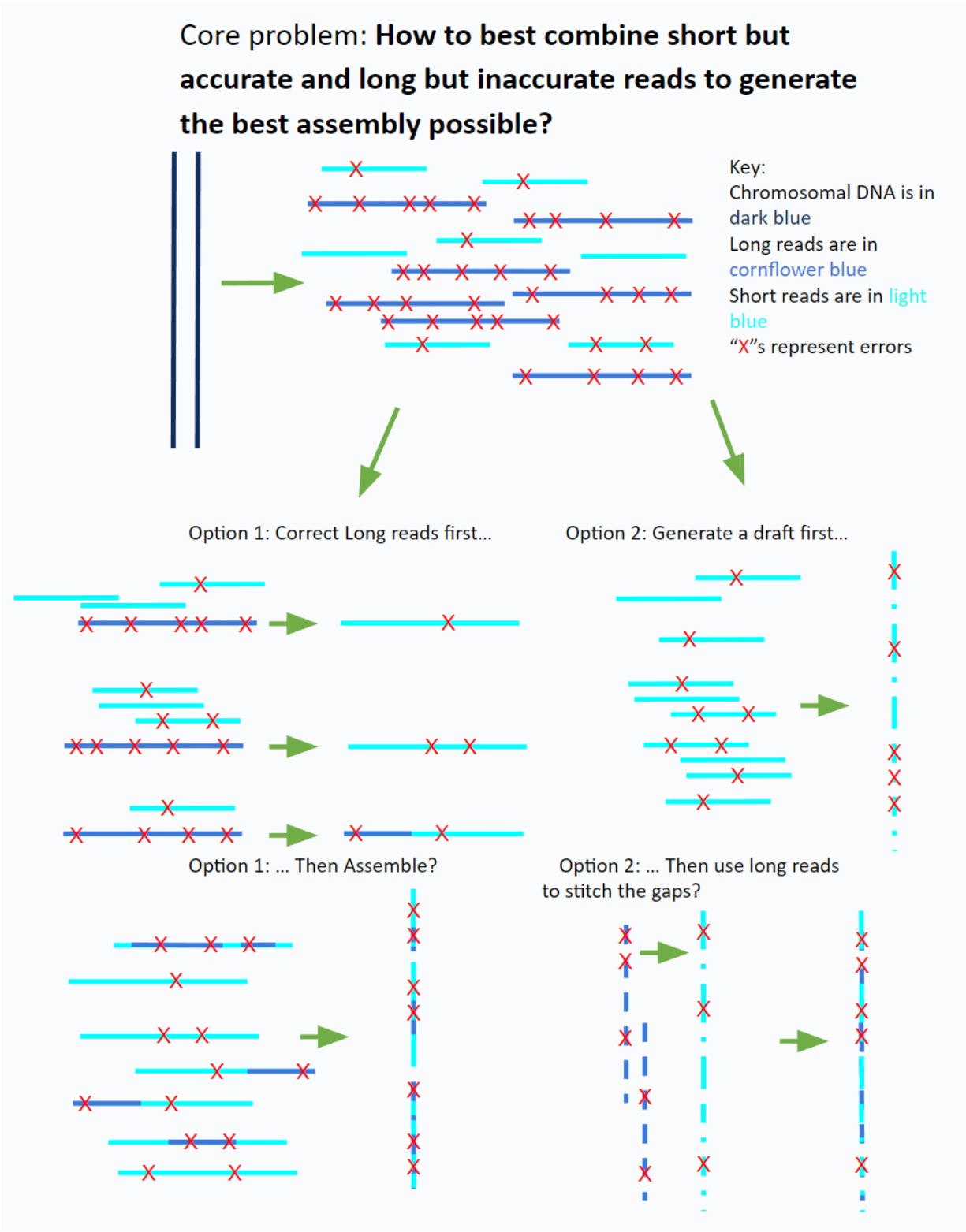

Figure 16. A visual abstract of the core strategy behind FMLRC2 (left) and LongStitch (right). Figure generated by the author.

With both FMLRC2 and LongStitch as attractive options, we wished to compare them in large genomes as there are no currently available studies that compare these two assemblers. The most similar paper published was a review article from 2010, which at the time was concerned with hybrid assembly techniques that used Sanger reads as the long reads.[46] Additionally, the problem is of greater importance in large genomes, where cost is a limiting factor in data availability.

[46] Schatz, Michael C, et al. "Assembly of Large Genomes Using Second-Generation Sequencing."

Experimental Methods

Selection of Model Organisms.

Arabidopsis thaliana, the thale-cress, was chosen as the smallest genome (135 Mbp) in our dataset. Advantages of choosing this species include its small genome, especially by plant standards, and its diploidy. Therefore, the genome has a notable lack of repetitive elements, which often inflate genome size.[47] As diploidy is less common among plants, its genome has been extensively sequenced, resulting in a high quality genome. Additionally, the genome only has five chromosomes, a comparatively small number. With fewer chromosomes there are fewer centromeres and telomeres, once again hard regions for both sequences and assemblers. All of these conditions combine to justify the inclusion of *A. thaliana* as a sufficiently "easy" eukaryotic genome to assemble.

Danio rerio, the zebrafish (also known as the zebra danio), was chosen as another model genome for our project. *D.rerio* features many advantages as a model system, including a high regeneration potential. A high regeneration potential is a key feature of our target genome, *Notophthalmus viridescens* (the eastern newt). Therefore, understanding how genomes of highly regenerative species (with potentially complicated genes and gene expression elements) is of great interest to us. Also, transgenic lines of *D.rerio* have been developed for many reasons, from exploring its regenerative potential to the pet trade. For this reason, its genome has also been sequenced many times, making it another strong choice as a model genome. Additionally, its approximately 1.4 Gbp size is approximately an order of magnitude larger than that of *Arabidopsis*.[48] With 70% of human genes having at least one zebrafish orthologue and the

[47] Saraswathy, Nachimuthu. "Genomes of Model Organisms."
[48] Howe, Kerstin, et al. "The Zebrafish Reference Genome Sequence and Its Relationship to the Human Genome."

genome having 25 chromosomes (2n=50), it also has some significant similarities to our own genome.[49,50] This renders it an excellent choice for transitioning to the larger genomes used in this study.

The next genome selected was the human reference genome. The human genome is an excellent candidate for inclusion into our genome assembler benchmarking study as it is of exceptional quality due to the unparalleled interest in its sequencing. Since the ultimate goal of most scientific research is to understand how biological phenomena affect human health, having a clear understanding of how the human genome is structured and behaves is of extreme importance and warrants significant study. Additionally, it is also fairly representative of the average mammalian genome due to its size (approximately 3.1 Gbp). This is very close to *mus musculus* (the mouse; 2.7Gbp) a frequent choice among mammalian model organisms This additionally confers the advantage of understanding how the mouse genome may be best assembled, which is also of great relevance due to the frequent genomic and genetic modification studies that take place in mice.

Pluerodeles waltl (the Iberian newt) was the next organism considered for our study. While not as widely used in studies as *Arabidopsis* or *Danio*, *P.waltl* is the only currently available genome of the family *Salamandridae*, the same family that our target genome *Notophthalmus* is in. The *Salamandridae* salamanders are of increasing interest due to their exceptional regenerative abilities. Species of this family can regenerate a wide variety of organs and body structures, including but not limited to entire limbs, the heart, and the central nervous system. The regenerative abilities of *Salamandridae* even extend to the lens of the eye, a structure that *Ambystoma mexicanum* (the axolotl, famous for its own regenerative abilities)

[49] Ibid
[50] Freeman, Jennifer L, et al. "Definition of the Zebrafish Genome Using Flow Cytometry and Cytogenetic Mapping."

cannot regrow.[51] While outside the genus *Notophthalmus*, the Iberian newt is an ideal candidate for our analysis as it is estimated that the eastern newt has eleven chromosomes, close to the twelve found in *P.waltl*. Additionally, the very large genome size (approximately 20 Gbp) is on the cutting edge of accurate genome assembly, offering a challenge to both FMLRC2 and LongStitch.

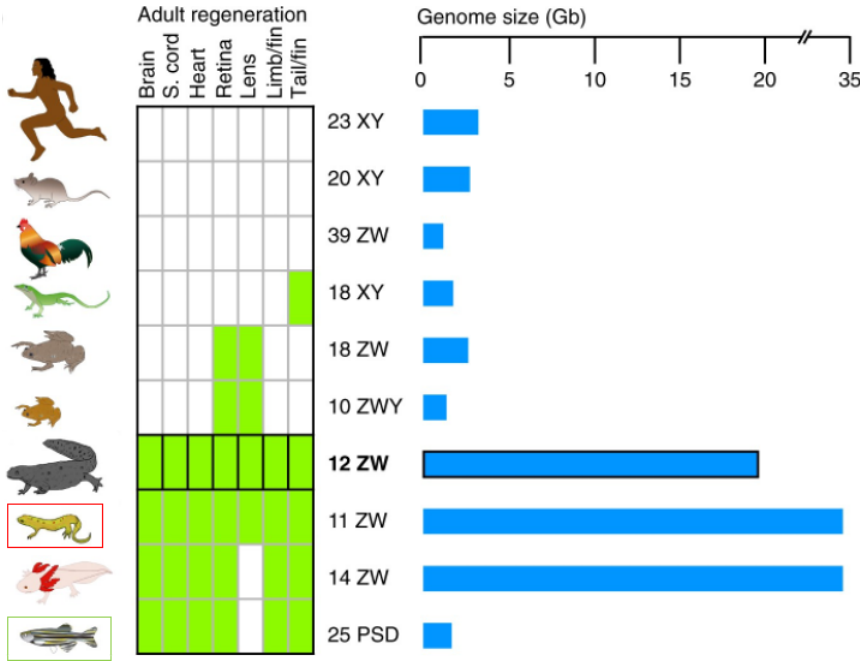

Figure 17. A diagram showing the regenerative abilities and genome sizes for humans and several model organisms. *Danio* is located on the bottom, boxed in green; *Notophthalmus* is boxed in red. The Iberian newt is located immediately above it. Figure taken from *Reading and editing the Pleuroedles waltl genome reveals novel features of tetrapod regeneration*, with cropping of an evolution chart and addition of boxes to identify organisms used in this study for clarity.[52]

[51] Elewa, Ahmed, et al. "Reading and Editing the Pleurodeles Waltl Genome Reveals Novel Features of Tetrapod Regeneration." *Nature News*, Nature Publishing Group, 22 Dec. 2017, https://www.nature.com/articles/s41467-017-01964-9/.
[52] Ibid

As 20+ Gbp genomes are currently the target of accurate *de novo* assemblies, the *P. waltl* genome currently available has some limitations. At the time of its publication (2017), HiFi reads were prohibitively expensive and the concept of hybrid genome assembly was in its infancy. Therefore, the assembly was done exclusively with Illumina reads, and after conducting an N50 analysis the genome was found to be too fragmented to include in our work. This highlights the importance of finding out the most efficient way to assemble large genomes, and that the answer is likely to involve multiple read types. Therefore, we chose *Pinus taeda* (Loblolly Pine) as our "large" and "difficult" genome, given its similar size to *P.waltl* (22 Gbp). The loblolly pine genome was originally published in 2010, but has since been updated with long reads; therefore, its N50 is sufficiently large enough to be included in our study.[53] Additionally, the *P. taeda* genome is diploid, a rarity amongst plant genomes of this size. Another similarity to *P.waltl* is that both genomes have 12 chromosomes (2n = 24), rendering it an excellent candidate amongst plants to substitute for an animal genome assembly.

The genomes for the four reference organisms were obtained as follows:

Scientific Name	Refseq/GenBank number of reference genome
A. thaliana	GCF_000001735.4
D.rerio	GCF_000002035.6
H.sapiens	GCF_000001405.40
P.taeda	GCA_000404065.3

[53] Zimin, Aleksey V, et al. "An Improved Assembly of the Loblolly Pine Mega-Genome Using Long-Read Single-Molecule Sequencing."

Software Used.

Ubuntu was installed on a Windows 10 Pro custom - built computer (hardware specifications can be found in the appendix). Miniconda3 and Bioconda were used as package managers to install all future softwares (exceptions noted explicitly).

To simulate the generation of genomic reads PBSIM3 was used on default settings. For simulated error generation, the error hidden markov model was used. This error was corrected by Pacific Bioscience's ccs to generate the simulated HiFi reads. For the oxford nanopore reads, PBSIM3 was used on default settings with the quality score hidden markov model configuration. To assemble genomes using the nanopore read correction via HiFi reads, FMLRC2 was used with all default dependencies and with the default settings other than 16 threads and a k-mer precompute cache size of 13 to correct reads. Then, minimap2 and miniasm were used with default settings other than the mapping, which was switched to ava-ont to optimize for nanopore reads. To assemble genomes by first creating a draft assembly with the HiFi reads, and then using the nanopore reads to fill in any gaps, LongStitch was used in ARKS-long mode with all default dependencies and settings other than 16 threads and nanopore longmapping.

To parse the newly generated genome assemblies three independent variables were assessed. They were: 1) reference genome size (constant per species), simulated short read coverage, and simulated long read coverage. These were used to assess seven dependent variables. In order of analysis, there were: 1) size of newly generated genome assembly, 2) N50, 3) number of insertions, 4) average insertion size, 5) number of deletions, 6) average deletion size, and 7) rate of substitution. Genome Assembly size and N50 were assessed with an in-house C++ program, and figures were generated with BANDAGE.[54] The variables associated with the accuracy of the newly assembled genomes were quantitatively assessed by aligning them to their

[54] Wick, Ryan R. "Bandage: Interactive Visualization of De Novo Genome Assemblies."

respective species' reference genome using GSAlign and then parsing its output files with an in-house C++ program. GFA and VCF files were used for FMLRC2 assemblies, while MAF files were used for LongStitch (LongStitch does not generate GFA files during its data generation, therefore, VCF files are inappropriate to use).

Values for the seven dependent variables were stored in txt files and then uploaded to Microsoft Excel for downstream statistical analysis and creation of figures.

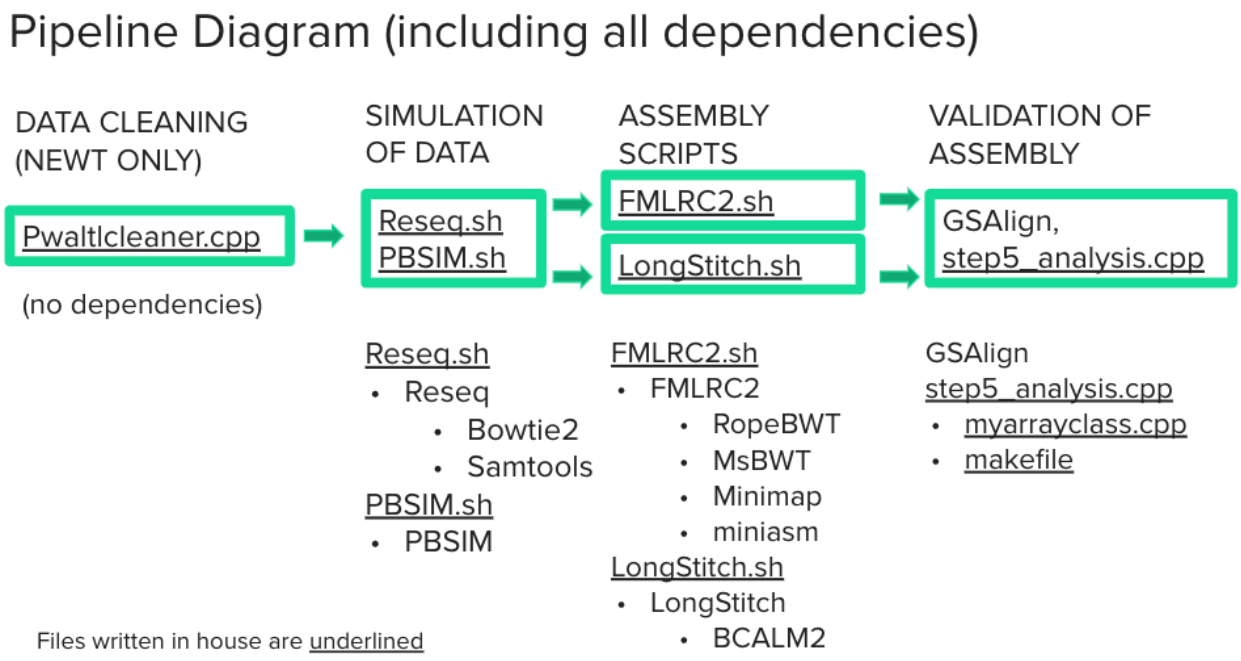

Figure 18. The overall software pipeline to generate the data discussed in this paper. As noted, programs written by the author for the completion of this study are underlined.

Results

Completeness and Contiguity.

A.thaliana

We found that, for FMLRC2, the fraction of the original genome recovered was very heavily correlated with the coverage of PacBio reads used.

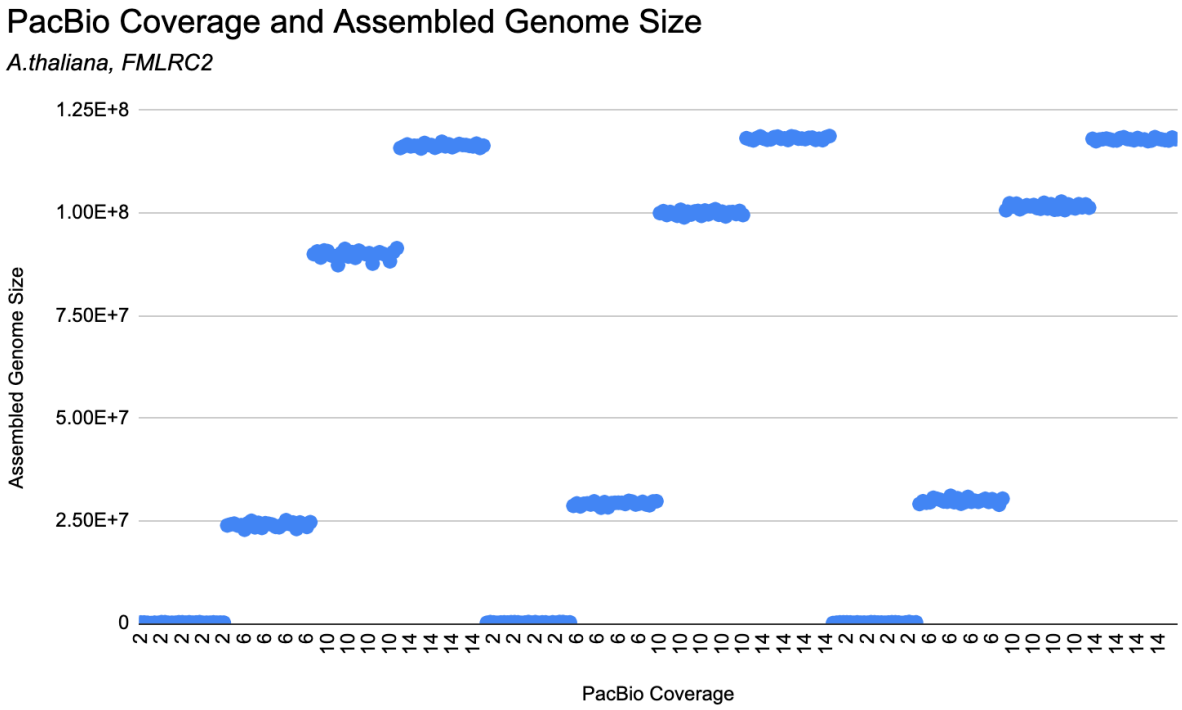

Figure 19. The size of the *A.thaliana* genome assembled across replicates, grouped by Illumina coverage and labeled by PacBio coverage. The left group of genome assemblies represents 5X Illumina coverage, the middle group represents 15X Illumina coverage, and the right group represents 25X Illumina coverage. Note that the maximum possible assembled genome size would be 121 Mbp, just beneath the top value on the Y axis.

Similar results were obtained for the N50, where the PacBio coverage was the primary determinant of how fragmented the assembled genome was. Increasing Illumina coverage was also slightly correlated with an increased N50 in *Arabidopsis* (figure 20).

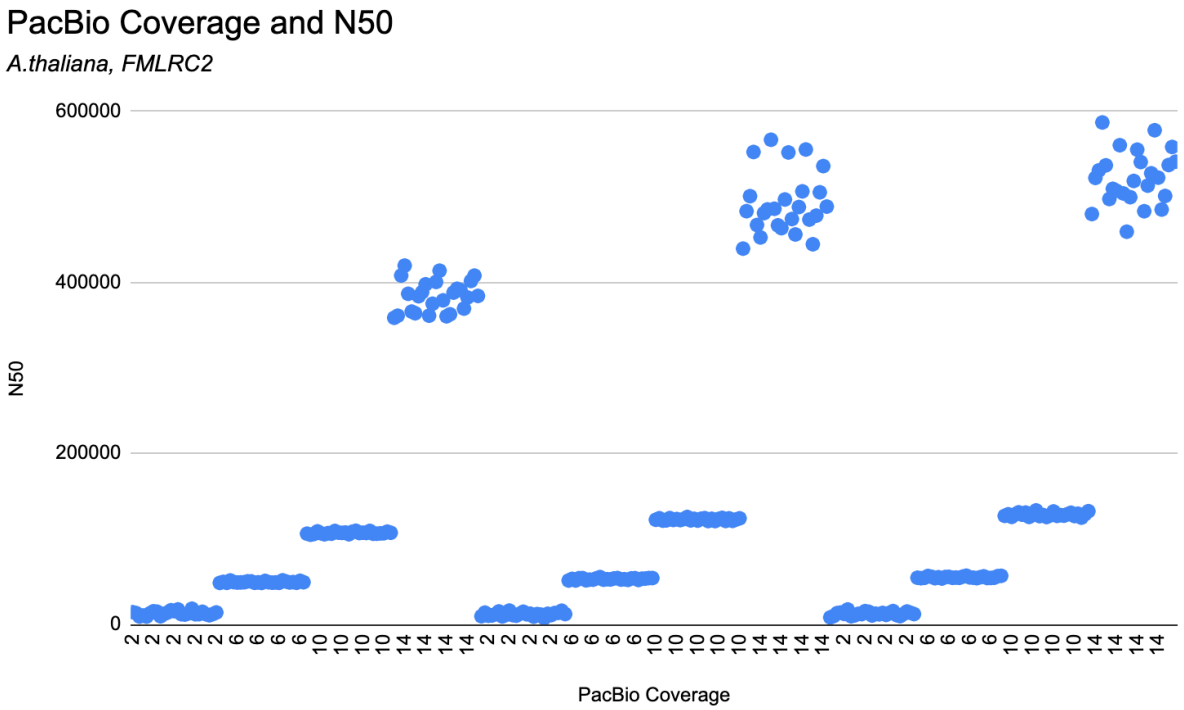

Figure 20. A scatter plot of the N50 for the *A.thaliana* genomes assembled with varying Illumina (5X on the left, 15X in the middle, and 25X on the right) and Pacbio coverages. A key difference between the genome size assembled and N50 is that, while assembly size started to reach a maximum value, the maximum N50 reached was still significantly beneath the hypothetical maximum.

The LongStitch data, meanwhile, did not have as meaningful trends. There was very little correlation between either Illumina coverage or PacBio coverage and N50 for *A.thaliana* genome

assemblies generated with it. Due to this, LongStich was not considered further, and further results are for FMLRC2 assemblies (except when noted).

D.rerio

The FMLRC2 data also behaved largely as expected in *D.rerio*:

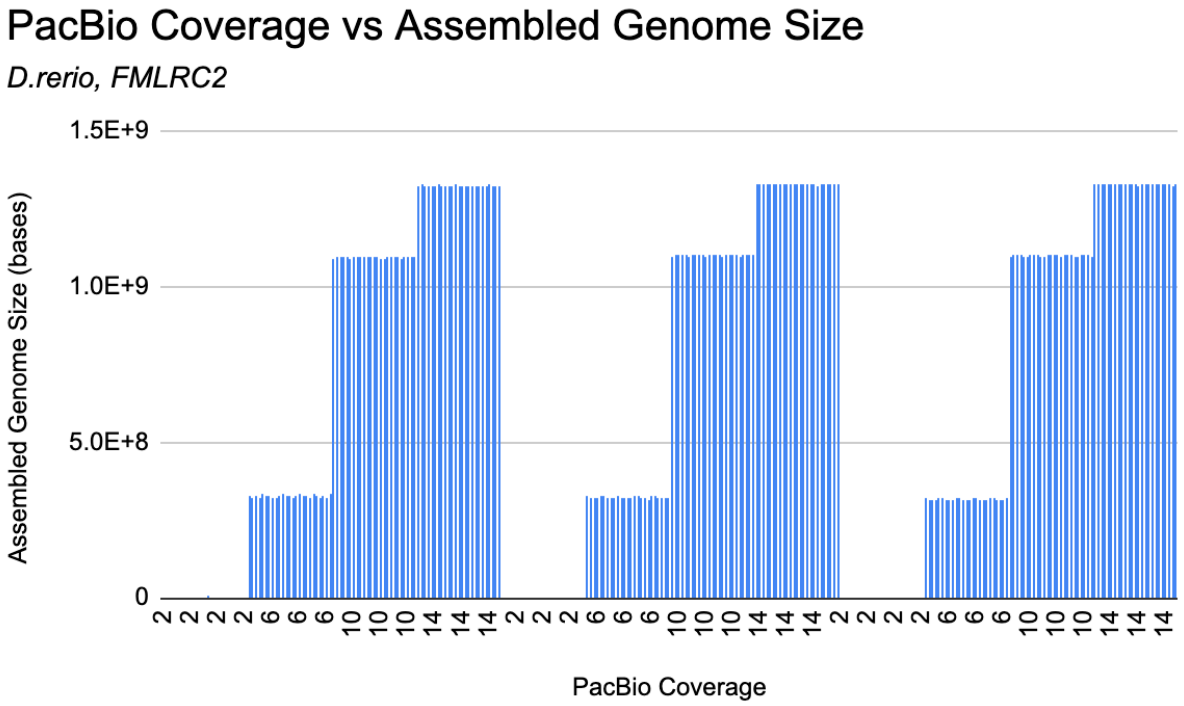

Figure 21. Pacbio coverage and assembled genome size, *D.rerio*, FMLRC2. Note, again, that assembled genome size is highly correlated with PacBio coverage and slightly correlated with Illumina coverage (5X on the left, 15X in the middle group, and 25X in the right group). Like the *A.thaliana* genomes, 14X coverage yielded nearly the entire reference genome (1.3 Gbp).

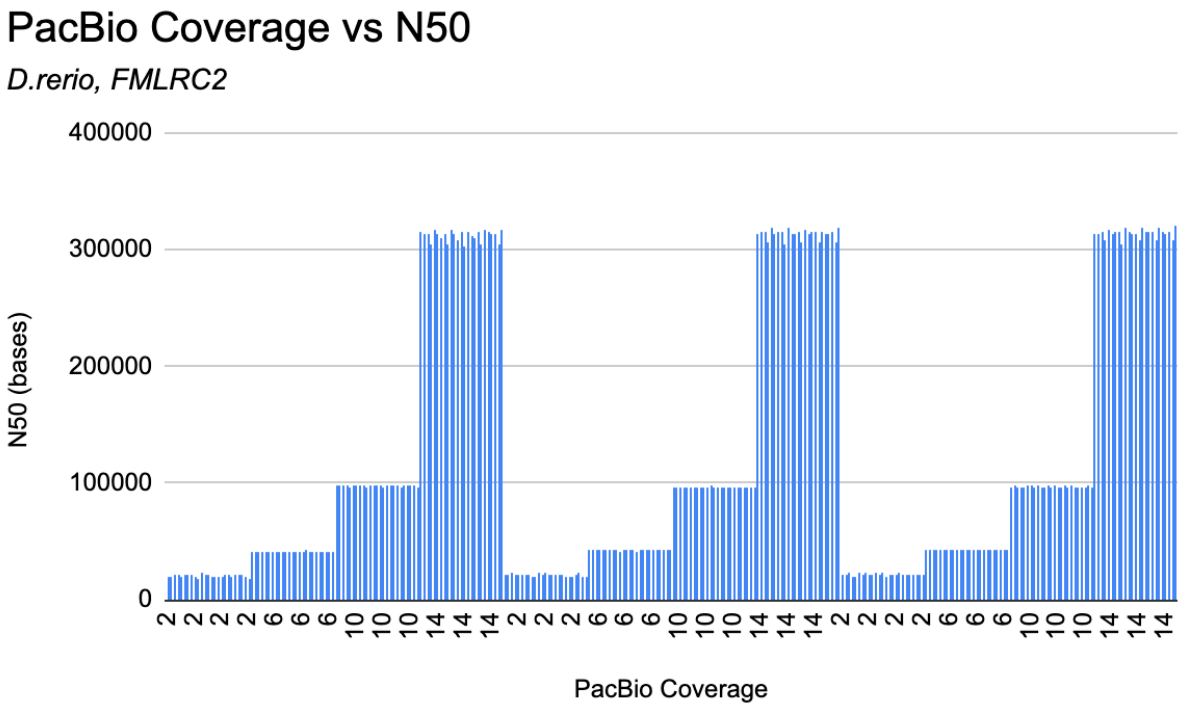

Figure 22. Pacbio coverage and N50, *D.rerio*, FMLRC2. Once again, while the fraction of the original genome recovered approached its theoretical maximum of 1, N50 could be improved upon. Also of note was that N50 was slightly lower than that of *A.thaliana*, but that difference is likely due to the more fragmented reference assembly (thereby lowering the possible maximum for our simulated genome assemblies).

Larger genomes and LongStitch

Data for *H.sapeins* was more qualitative as its generation was somewhat hampered by the available tools. FMLRC2 relies on miniasm to merge reads into contigs, and at the scale of the human genome, the program broke due to the number of reads for the higher coverages (10X and 14X PacBio reads). A fix is currently in the works, but the currently available data is not complete enough to warrant its own graphs. The data that has been generated so far mirrors both *A.thaliana* and *D.rerio*, where at 2X coverage nearly none of the genome is recovered and at 6X

coverage approximately a quarter of the genome is. Likewise, the generation of assemblies for either *P.taeda* or *P.waltl* is prohibitive given the same algorithmic flaw. Thus, no completion, contiguity, or accuracy statistics currently exist for these, although they are being pursued.

LongStitch data, meanwhile, exist for *A.thaliana*, although LongStitch had very little correlation between read coverage and assembly size, other than a jump from 5X to 15X Illumina coverage. However, at even 25X Illumina 14X PacBio coverage, the assembled genome size was significantly under its theoretical maximum (1.35 E+8). Therefore, with an assembler that could achieve that level of completeness at the given coverages, we did not pursue LongStitch further.

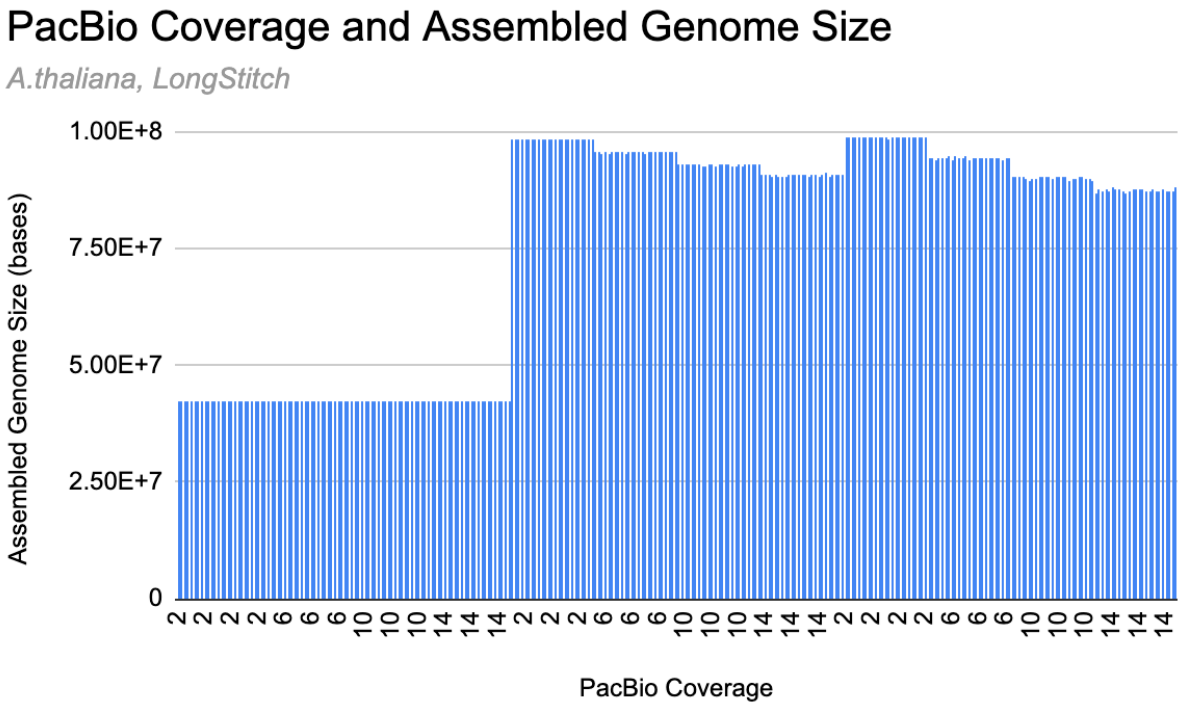

Figure 23. PacBio coverage versus assembled genome size. Note the lack of meaningful trend between PacBio read coverage and assembled genome size.

Accuracy.

A.thaliana

A. thaliana assemblies had the highest number of substitutions when PacBio coverage was at its highest (figure 24). However, this trend only exists due to the drastic increase in the number of bases available for errors to occur on, and the highest coverage PacBio assemblies began to hit the theoretical limit for *Arabidopsis*. Assemblies with more Illumina coverage had a lower rate of error when corrected for differences in Pacbio coverage (figure 24, figure 25). On this note, when controlling for PacBio coverage, as coverage increased to 25X, the number of substitutions fell nearly 3 fold while genome size increased slightly (figure 25, figure 21). Also of note was that, while the number of substitutions increased while PacBio coverage increased from 2X to 10X, many 14X PacBio assemblies had fewer substitutions than 10X PacBio assemblies. This is especially noteworthy as the assembled genome size slightly increased from 10X to 14X, hinting at parital long read self correction. Meanwhile, we also saw that Illumina reads were correcting the mistakes present in PacBio reads (figure 26, figure 27).

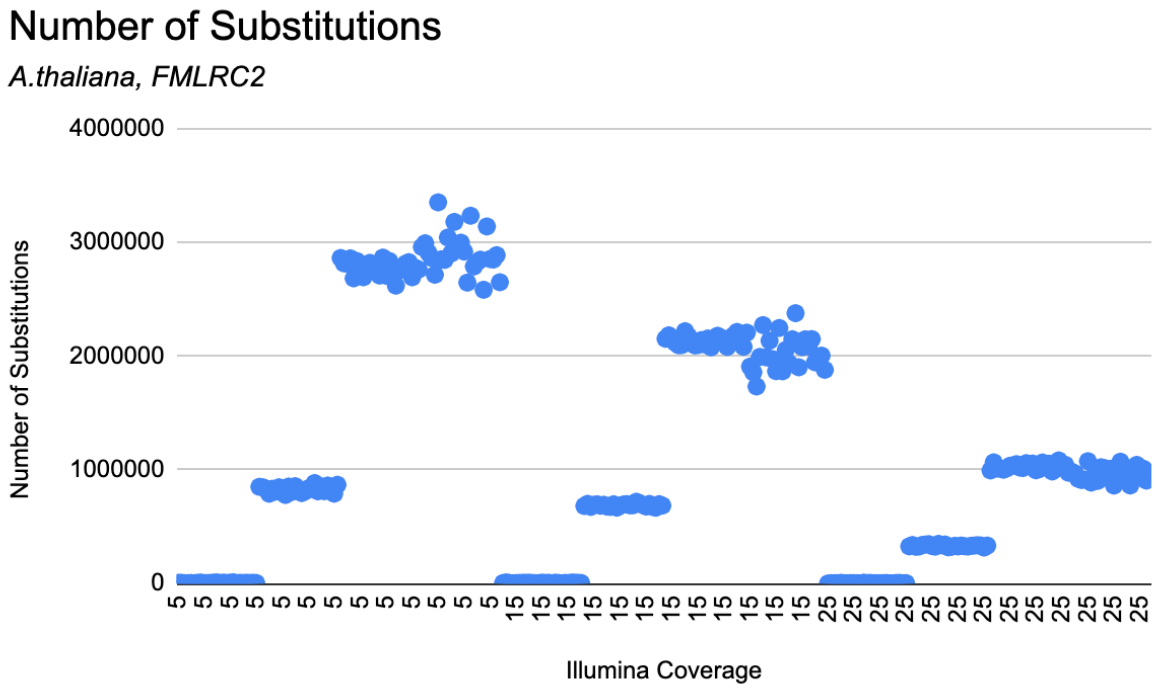

Figure 24. A scatter plot comparing the Illumina coverage used with the frequency of substitutions. Note that, within each level of Illumina coverage, PacBio coverage increases from 2X to 6X and eventually 14X. Therefore, the first set of 5X Illumina assemblies should be compared only to the first set of 15X assemblies, and likewise with the other sets to control for the differences in PacBio coverage. Additionally, note that there is nearly no difference in mean error rate between 10X and 14X PacBio assemblies, although 14X assemblies

feature greater variation in error rate (figure 25).

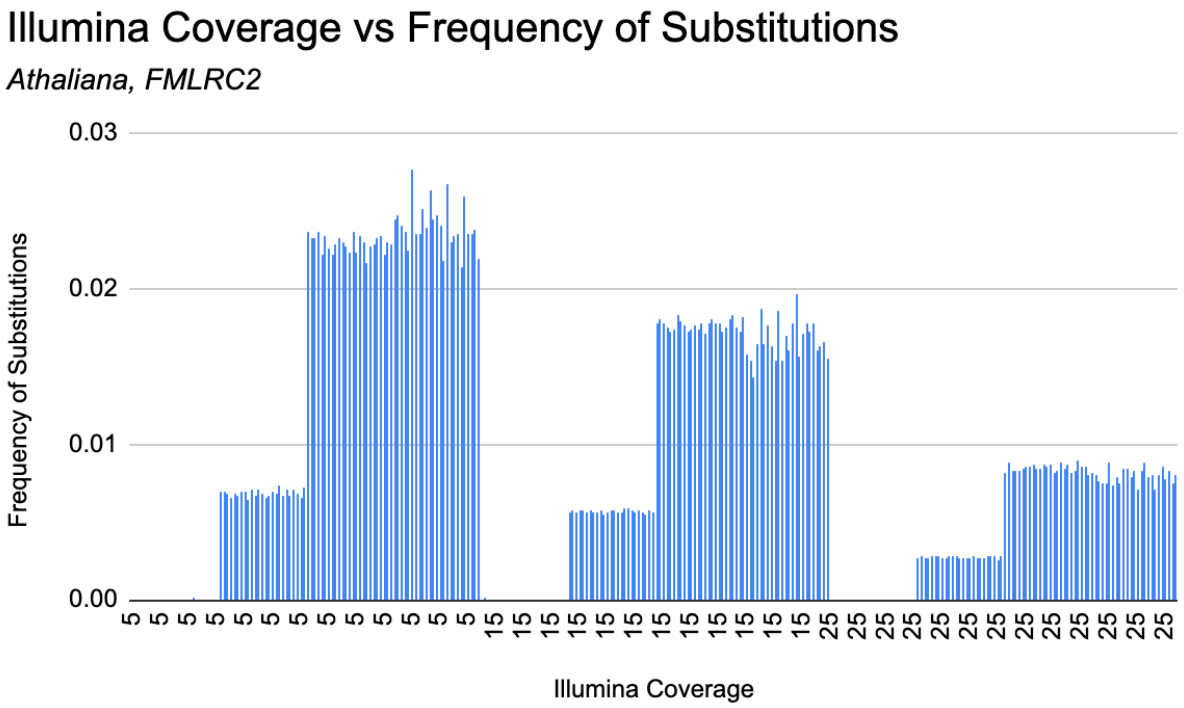

Figure 25. A histogram of the frequency of substitutions present in each *A.thaliana* genome assembled. Note that the genomes with 2X PacBio coverage recovered such a small fraction of the genome that their inclusion is insignificant.

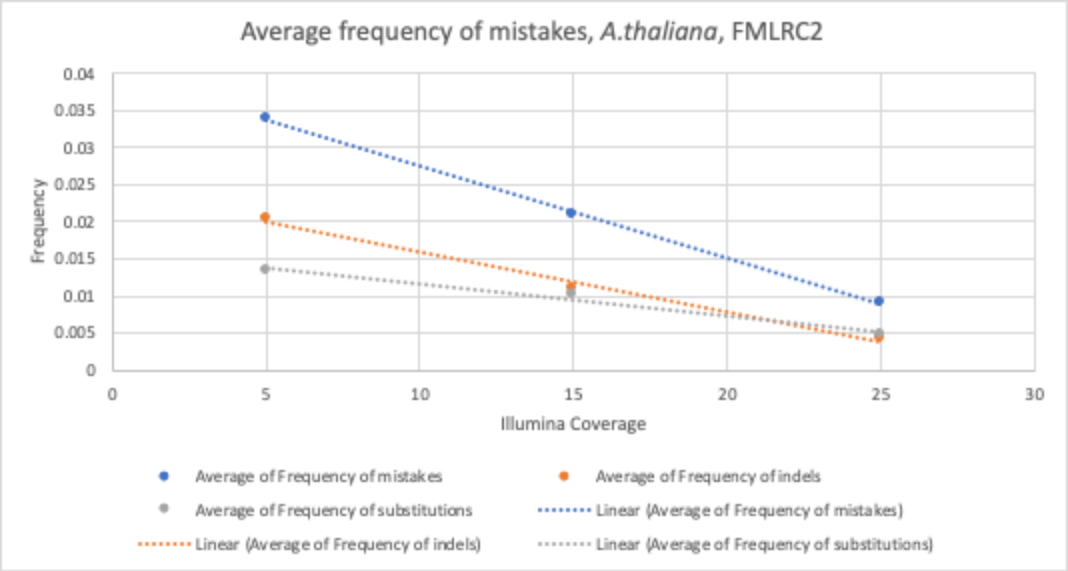

Figure 26. The average rate of error for each Illumina coverage (aggregating all PacBio coverages). Note that, for genomes with less Illumina coverage, indels were the primary source of error. For genomes with more Illumina coverage, substitutions became the more common source of error. Since Illumina reads have primarily substitution errors, and PacBio reads have indel errors, this result implies that Illumina reads are correcting the indels seen in the PacBio reads.

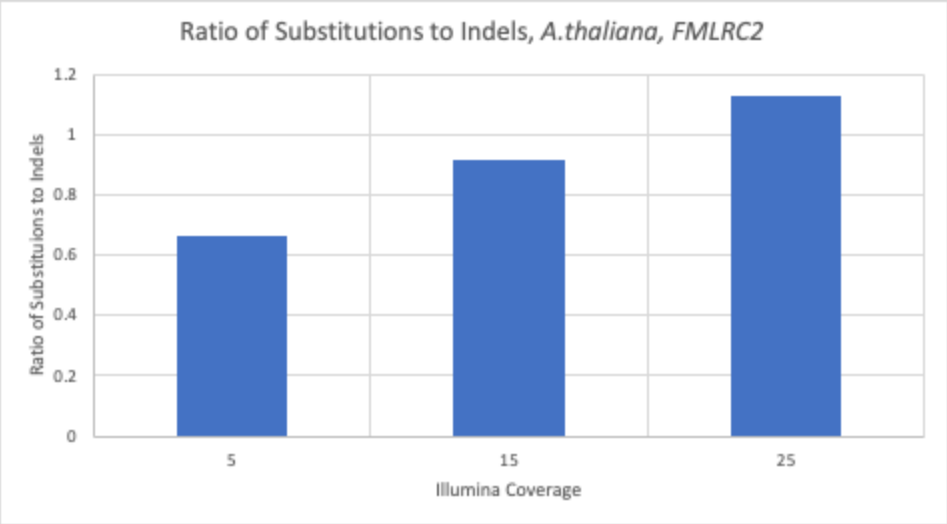

Figure 27. The Ratio of Substitutions to Indels in *A.thaliana* using FMLRC2. As Illumina coverage grew higher, substitutions were proportionally more frequent than indels.

D.rerio

The same trends existed in the *D.rerio* assemblies generated. Interestingly, the error rate fell off more steeply than *A.thaliana* between 5X and 15X Illumina coverage (figure 28). As a result, the error rate for *D.rerio* was actually lower than that for *A.thaliana*, despite an approximately order of magnitude increase in genome size (figure 29).

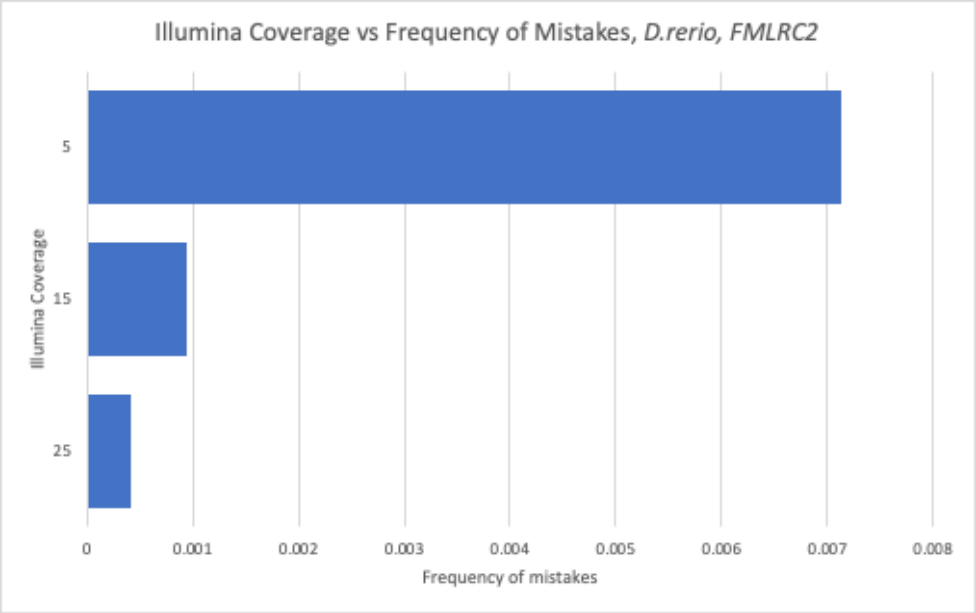

Figure 28. The level of Illumina coverage along with the average frequency of mistakes in *D.rerio*. Note that for 15 and 25X, the error rate is less than 0.1%, indicating an extreme level of accuracy and sufficient read correction.

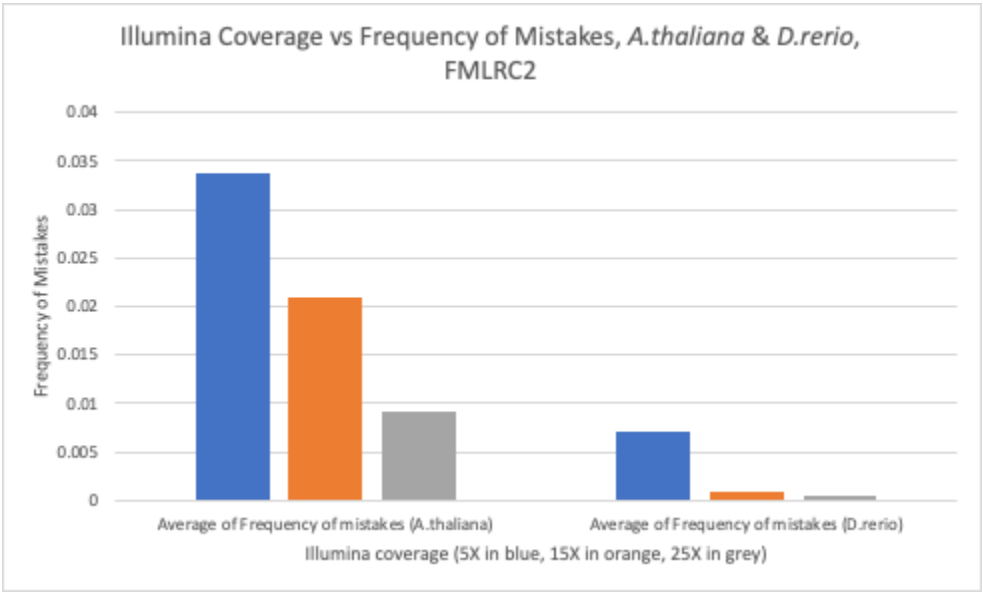

Figure 29. *Athaliana* and *D.rerio* error rate side-by-side. Despite having a significantly larger genome, *D.rerio* had significantly fewer errors at all levels of coverage.

Composite Results.

One of the most insightful graphs generated in this study was obtained by mapping the fraction of the original genome recovered to the error rate per base. By doing this, we can see both how complete and accurate the genome assemblies returned are. On the graph shown in figure 23 (FMLRC2 assemblies), the assemblies generated a "fan" shape where, as PacBio read coverage increased, both the fraction of the assembled genome recovered and the error rate per base increased. However, this increase could be somewhat mitigated in some assemblies by increasing the level of Illumina Coverage. The same graph for *A.thaliana* assemblies generated with LongStitch (figure 31), meanwhile, did not have clear trends. Most assemblies returned around 75% of the genome with little correlation to either Illumina or PacBio coverage.

Figure 30. The fraction of the original genome recovered (X axis) compared to the error rate per base (Y axis). Since a perfect genome assembly would be 100% complete and have no

errors, the bottom right corner of the graph represents the ideal genome assembly. The closest assemblies to this zone are the *Danio* assemblies with 25X Illumina read coverage and 14X PacBio coverage.

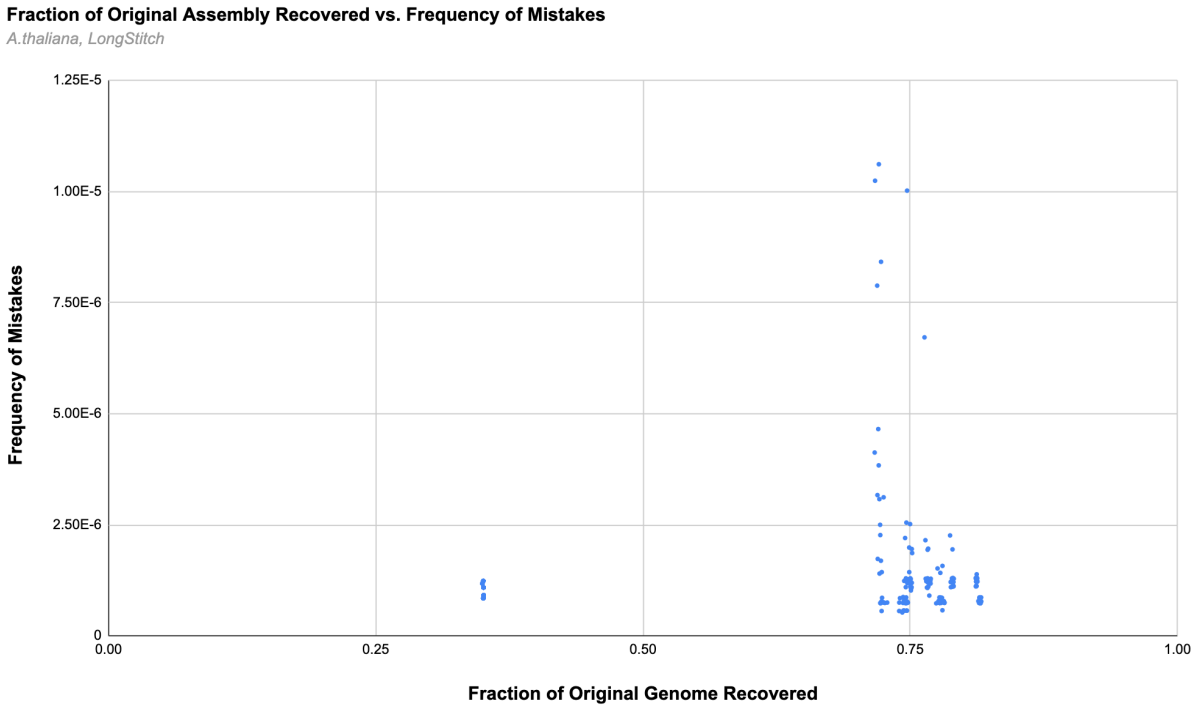

Figure 31. The fraction of the *A.thaliana* genome assembly recovered and its error rate when assembled with LongStitch. There is very little correlation between either PacBio or Illumina read coverage and the fraction of the genome recovered. There was a slight correlation between increasing PacBio coverage and increase in error rate, however, these assemblies were also slightly more complete.

The results we observed for continuity (N50) largely mirrored the trends seen for completeness. Again, assemblies with FMLRC2 were quite dependent on high levels of PacBio coverage to obtain a high N50 (figure 32, figures 22 and 20).

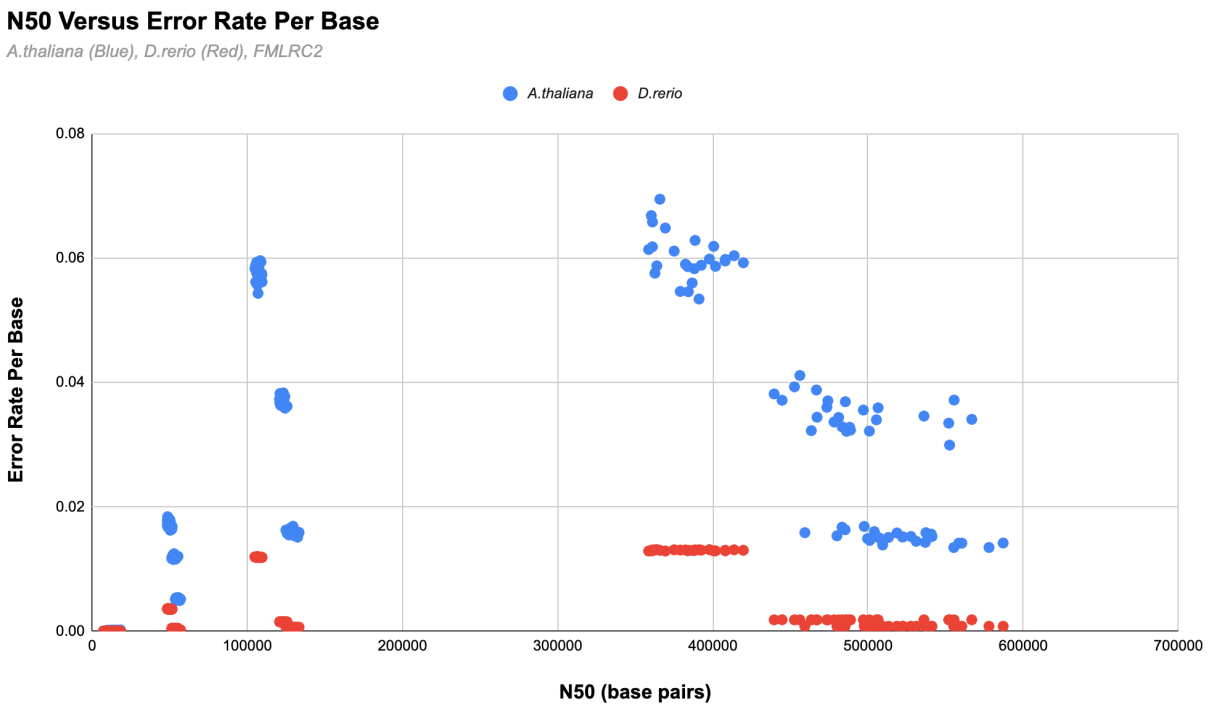

Figure 32. N50 (X axis) compared to error per base (Y axis). Ulike the fraction of the original genome recovered, the bottom right corner is not perfect (although it is the most ideal), as N50 does not reach a maximum within the tested coverages.

Discussion

We found that, as expected, long reads are critical for continuity, and short reads greatly improve accuracy. For this reason, hybrid genome assembly is the most efficient way to generate both complete and accurate large genomes, as one would have to spend significantly more to achieve the same level of accuracy by only using long reads. If only short reads are used it may not be possible to achieve the same levels of completion and especially continuity. This is due to repetitive elements that are longer than the length of the short reads, which make the beginning and end of the sequence unreachable.

Given that the fraction of the genome recovered approaches 100% at 14X PacBio coverage in *A.thaliana* and *D.rerio* (and preliminary results suggests this trend continues in human), and that error rate stays within 1% for *D.rerio* at 25X Illumina coverage, it is likely that these amounts are sufficient to assemble a genome that is both complete and accurate. N50, meanwhile, continues to see improvements. Past a certain size, however, comes a significant decision point for those interested in *de novo* assembly: is paying thousands of dollars more worth a negligible increase in both completeness and accuracy, only to gain from understanding where contigs are in relation to each other? The answer will depend on both the goals of the assembly and the price of the technology at the time of assembly. Molecular biologists primarily interested in gene function may think of understanding gene linkage as a lower priority task compared to an evolutionary biologist. This is because, while enhancers can be tens to even hundreds of thousands of base pairs away from the gene in question, regulatory elements for a given gene are unlikely to be megabases away from the given sequence. Therefore, attaining an N50 into the megabases is unlikely to resolve any extra information on gene expression. Such work, however, could easily update linkage maps available for a species or change previously

inferred inheritance patterns, resulting in a great deal of information important to an evolutionary biologist.

One of the most surprising and important implications of our work is that it seems the same levels of coverage generate both complete and accurate genome assemblies regardless of genome size. While our work is incomplete, preliminary data suggest that the fraction of a human genome that can be recovered given a PacBio read coverage amount is within one percent of that of *A.thaliana*. This is surprising as the *Arabidopsis* genome is approximately 23 times smaller than the human genome, and has considerably fewer repetitive elements. Seemingly, assembling the human genome would be a much more difficult task, even when accounting for the fact that coverage is measured relative to the sample genome size. Another implication of this work is that, although less important, the scientific community may be spending too much assembling smaller genomes. A 2013 study claims that 50X coverage is the optimal amount of short read coverage to assemble a bacterial genome, however, our hybrid approach can assemble much larger genomes with near perfect completion and accuracy with approximately 39X coverage (25 Illumina + 14 PacBio = 39 total).[55] One consideration, however, is that the standard of completion is much, much higher for bacterial genomes. Due to their small and circular nature of bacterial genomes, many groups now target obtaining the complete genome in one contiguous scaffold. That level of resolution is not currently possible for eukaryotic genomes due to the presence of centromeres, telomeres, and the sheer scale of the data. Regardless, it suggests that the amount of coverage required to obtain a high fraction of the hypothetical total genome size is relatively constant, as one of the genomes benchmarked was the *E.coli* genome. Since this genome is 280 times smaller than even the *D.rerio* genome, it is likely that this trend will

[55] Desai, Aarti, et al. "Identification of Optimum Sequencing Depth Especially for De Novo Genome Assembly of Small Genomes Using Next Generation Sequencing Data."

continue into even the largest genomes. As our work continues, we will include the *P.taeda* data and be able to test this claim. Additionally, the *de novo* sequencing and assembly of our target genome, *N.viridescens*, will be the true test of our claim.

Conclusion

Genome assembly is a very exciting field as it is rapidly evolving, interdisciplinary, and has important implications for health, medicine, ecology, zoology, agriculture, climate change, and countless other fields. We hope our work can provide insight into the most cost effective way to sequence the most challenging genomes, which should hopefully aid study on organisms which to this point has been hindered due to their genome size.

Our work is somewhat limited in scope as, only a few months ago, PacBio announced the Revio sequencing machine. This has lowered the cost of HiFi reads 15 fold and now renders them competitive at reasonable price points, even for the largest genomes. As a result, leading edge hybrid assemblies should now be done with short reads that are 100 times larger than Illumina reads, with nearly no sacrifice in accuracy. Therefore, the data presented is somewhat obsolete, and “short” reads are likely to be more important than our data account for.

Future work that could build off our research would include a similar project involving more species. This would mitigate bias from the organisms chosen. For example, a group wishing to know the most effective way to sequence polyploid genomes should replicate this work using a series of polyploid references. An important observation we made while collecting data was that ensuring k-mer size was as large as possible was critical for avoiding cycles and increasing N50 to the maximum value possible. An additional avenue for future work would be to test a series of k-mer sizes while leaving other independent variables (most importantly coverage) constant so that a trend between k-mer size and N50 could be quantitatively assessed.

Once again, I would like to thank Professor Dougalss, Professor Izmirli, Professor

Eastman, and the departments of Biology, Chemistry, and Computer Science for the opportunity to have done this work during my undergraduate years at Connecticut College.

Appendix

Computer Hardware Specifications.

The custom built computer used for this study has a Ryzen 3950X processor, featuring 16 cores running at 4.5 GHz, 128 GB DDR4 RAM, and 3 Seagate 18TB SATA hard drives which data was stored on. While these specifications were top of the line for a consumer PC at the time of its construction (2021), genome assembly is a computationally difficult enough task that some datasets were intractable due to the length of time it took to operate (weeks in the case of simulating *P. waltl* and *P. taeda* reads). Hardware upgrades and/or algorithm improvements will be needed to continue this work.

Bibliography

Alberts, Bruce, and Bruce Alberts. “Chapter 4, Figure 4.55.” *Overhead Transparencies for Molecular Biology of the Cell, Fourth Edition*, Garland Science, London, 2002.

“Base Pairing in DNA.” *Base-Pairs.html 16_08dnabasepairing_l.Jpg*, http://bio1152.nicerweb.com/Locked/media/ch16/base-pairs.html.

Benjamini, Yuval, and Terence P Speed. “Summarizing and Correcting the GC Content Bias in High-Throughput Sequencing.” *Nucleic Acids Research*, U.S. National Library of Medicine, May 2012, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3378858/.

“Beta-D-2-Deoxyribose.” *National Center for Biotechnology Information. PubChem Compound Database*, U.S. National Library of Medicine, https://pubchem.ncbi.nlm.nih.gov/compound/beta-D-2-Deoxyribose#section=Structures.

Chen, Xiao. *The Architecture of a Scrambled Genome Reveals Massive Levels of ... - Cell*. 2014, https://www.cell.com/cell/fulltext/S0092-8674(14)00984-2.

Cheng, Chu, and Pengfeng Xiao. “Evaluation of the Correctable Decoding Sequencing as a New Powerful Strategy for DNA Sequencing.” *Life Science Alliance*, U.S. National Library of Medicine, 14 Apr. 2022, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9012935/.

Compeau, Phillip E C. “How to Apply De Bruijn Graphs to Genome Assembly - Eaton-Lab.org.” *The Eaton Lab at Columbia University*, Nov. 2011, https://eaton-lab.org/slides/genomics/readings/Compeau-etal-2011.pdf.

Cornell, Brent. “Gene Identification.” *BioNinja*, https://ib.bioninja.com.au/options/untitled/b2-biotechnology-in-agricul/gene-identification.html.

Cornell, Brent. “Semi-Conservative DNA Replication.” *BioNinja*, http://ib.bioninja.com.au/standard-level/topic-2-molecular-biology/27-dna-replication-transcri/semi-conservative.html.

Desai, Aarti, et al. “Identification of Optimum Sequencing Depth Especially for De Novo Genome Assembly of Small Genomes Using Next Generation Sequencing Data.” *PloS One*, U.S. National Library of Medicine, 12 Apr. 2013, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3625192/.

Elewa, Ahmed, et al. “Reading and Editing the Pleurodeles Waltl Genome Reveals Novel Features of Tetrapod Regeneration.” *Nature News*, Nature Publishing Group, 22 Dec. 2017, https://www.nature.com/articles/s41467-017-01964-9/.

Epp, Christopher D. “Definition of a Gene.” *Nature News*, Nature Publishing Group, 1997, https://www.nature.com/articles/39166.

Freeman, Jennifer L, et al. “Definition of the Zebrafish Genome Using Flow Cytometry and Cytogenetic Mapping.” *BMC Genomics*, U.S. National Library of Medicine, 27 June 2007, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1925092/#:~:text=The%20zebrafish%20(Danio%20rerio)%20has,%3D%2050)%20%5B3%5D.

“Genome Sequencing: A History.” *Genome Sequencing: A History*, http://bioinfo.mbb.yale.edu/course/projects/final-4/.

Gruenwedel, D W. “Nucleic Acids: Properties and Determination.” *Encyclopedia of Food Sciences and Nutrition (Second Edition)*, Academic Press, 6 Dec. 2003, https://www.sciencedirect.com/science/article/pii/B012227055X008361.

Hamburg, E. “File:Haploid vs Diploid.svg.” *Wikimedia Commons*, 10 May 2010, https://commons.wikimedia.org/wiki/File:Haploid_vs_diploid.svg#/media/File:Haploid_vs_diploid.svg.

Hbf878. “File:Damp Chemical Structure.svg.” *Wikimedia Commons*, Nov. 2018, https://commons.wikimedia.org/w/index.php?curid=74023664.

Heslop-Harrison, J S. “Crop Improvement: Plant Genomes.” *Encyclopedia of Applied Plant Sciences*, Elsevier, 28 Nov. 2004, https://www.sciencedirect.com/science/article/pii/B0122270509001836.

Howe, Kerstin, et al. “The Zebrafish Reference Genome Sequence and Its Relationship to the Human Genome.” *Nature News*, Nature Publishing Group, 17 Apr. 2013, https://www.nature.com/articles/nature12111.

Joven, Alberto, et al. “Model Systems for Regeneration: Salamanders.” *Development (Cambridge, England)*, U.S. National Library of Medicine, 22 July 2019, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6679358/#:~:text=While%20all%20amphibians%20exhibit%20regenerative,parts%20throughout%20their%20entire%20life.

Karami, Ali. “Largest and Smallest Genome in the World - Researchgate.” *Research Gate*, Jan. 2013, https://www.researchgate.net/profile/Ali-Karami-5/publication/235907922_Largest_and_Smallest_Genome_in_the_World/links/00463514052ec05cbb000000/Largest-and-Smallest-Genome-in-the-World.pdf.

Logsdon, Glennis A, et al. “Long-Read Human Genome Sequencing and Its Applications.” *Nature Reviews. Genetics*, U.S. National Library of Medicine, Oct. 2020, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7877196/#:~:text=Continuous%20long%20reads%20(CLR)%20are,few%20passes%20around%20the%20template.

“Model Organism Sharing Policy.” *National Institutes of Health*, U.S. Department of Health and Human Services, https://sharing.nih.gov/other-sharing-policies/model-organism-sharing-policy.

Morozova, Olena, and Marco A Marra. “Applications of next-Generation Sequencing Technologies in Functional Genomics.” *Science Direct*, Academic Press, 24 Aug. 2008, https://www.sciencedirect.com/science/article/pii/S0888754308001651?via%3Dihub.

NCAA.com, Daniel Wilco |. “The Absurd Odds of a Perfect NCAA Bracket.” *NCAA.com*, NCAA.com, 16 Mar. 2023, https://www.ncaa.com/webview/news%3Abasketball-men%3Abracketiq%3A2023-03-16%3Aperfect-ncaa-bracket-absurd-odds-march-madness-dream.

Niu, C.-H., et al. “The Code within the Codons.” *Biosystems*, Elsevier, 30 Sept. 2003, https://www.sciencedirect.com/science/article/pii/0303264789900592?via%3Dihub.

“Phosphate Ion.” *National Center for Biotechnology Information. PubChem Compound Database*, U.S. National Library of Medicine, https://pubchem.ncbi.nlm.nih.gov/compound/1061.

R, varun N. “Big O Cheatsheet - Data Structures and Algorithms with Thier Complexities .” *HackerEarth*, https://www.hackerearth.com/practice/notes/big-o-cheatsheet-series-data-structures-and-algorithms-with-thier-complexities-1/.

RG;, Tennyson CN;Klamut HJ;Worton. “The Human Dystrophin Gene Requires 16 Hours to Be Transcribed and Is Cotranscriptionally Spliced.” *Nature Genetics*, U.S. National Library of Medicine, 1995, https://pubmed.ncbi.nlm.nih.gov/7719347/#:~:text=Abstract,least%202%2C300%20kilobases%20(kb).

Saraswathy, Nachimuthu. “Genomes of Model Organisms.” *Concepts and Techniques in Genomics and Proteomics*, Woodhead Publishing, 27 Mar. 2014, https://www.sciencedirect.com/science/article/pii/B9781907568107500031.

Schatz, Michael C, et al. “Assembly of Large Genomes Using Second-Generation Sequencing.” *Genome Research*, U.S. National Library of Medicine, Sept. 2010, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2928494/.

Shchelochkov, Oleg. “Open Reading Frame.” *Genome.gov*, May 2023, https://www.genome.gov/genetics-glossary/Open-Reading-Frame.

Springer Nature Limited. “The Vertebrate Genomes Project.” *Nature News*, Nature Publishing Group, 2021, https://www.nature.com/immersive/d42859-021-00001-6/index.html.

T;, Hummer KE;Nathewet P;Yanagi. “Decaploidy in Fragaria Iturupensis (Rosaceae).” *American Journal of Botany*, U.S. National Library of Medicine, Mar. 2009, https://pubmed.ncbi.nlm.nih.gov/21628226/#:~:text=iturupensis.,2n%20%3D%209x%20%3D%2063.

Vidal, Adrien, et al. “SESAM: Software for Automatic Construction of Order-Robust Linkage Maps - BMC Bioinformatics.” *BioMed Central*, BioMed Central, 19 Nov. 2022, https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-022-05045-7.

Videvall, Elin. “What's N50?” *The Molecular Ecologist*, 29 Mar. 2017, https://www.molecularecologist.com/2017/03/29/whats-n50/.

`Wang, Jeremy R. “Polishing De Novo Nanopore Assemblies of Bacteria and Eukaryotes With FMLRC2 .” *Academic.oup.com*, 3 Mar. 2023, https://academic.oup.com/mbe/article/40/3/msad048/7069220.

Warburton, Peter E, et al. “Analysis of the Largest Tandemly Repeated DNA Families in the Human Genome - BMC Genomics.” *BioMed Central*, BioMed Central, 7 Nov. 2008, https://bmcgenomics.biomedcentral.com/articles/10.1186/1471-2164-9-533.

Wetterstrand, Kris A. “DNA Sequencing Costs: Data.” *Genome.gov*, 1 Nov. 2021, https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data.

Wick, Ryan R. “Bandage: Interactive Visualization of De Novo Genome Assemblies.” *Academic.oup.com*, Oct. 2015, https://academic.oup.com/bioinformatics/article/31/20/3350/196114.

Wick, Ryan R., et al. “Assembling the Perfect Bacterial Genome Using Oxford Nanopore and Illumina Sequencing.” *PLOS Computational Biology*, Public Library of Science, 2 Mar. 2023, https://journals.plos.org/ploscompbiol/article?id=10.1371%2Fjournal.pcbi.1010905.

Zimin, Aleksey. “Sequencing and Assembly of the 22-Gb Loblolly Pine Genome.” *Academic.oup.com*, 2014, https://academic.oup.com/genetics/article/196/3/875/5935688.

Zimin, Aleksey V, et al. “An Improved Assembly of the Loblolly Pine Mega-Genome Using Long-Read Single-Molecule Sequencing.” *GigaScience*, U.S. National Library of Medicine, 1 Jan. 2017, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5437942/.