The Malta Human Genome Project

Progress Report

Sara Ann Abdilla (188396M)

Supervisor(s): Dr Jean-Paul Ebejer

Faculty of ICT

University of Malta

December 2016

Submitted in partial fulfillment of the requirements for the degree of B.Sc. Computing Science (Hons.)

Contents

List of Figures

Abstract: Human genomes are being globally assembled in order to aid in medical diagnosis, forensic research, genealogy and bioinformatics among other areas. While many countries have already had a reference genome constructed for their population, Malta is still in the process of accomplishing this - a National Maltese Human Reference Genome. In this dissertation, we propose to build computational algorithms for the visualisation and analysis of DNA by developing a compression tool and a sequence aligner as deliverables, with the option (depending on time constraints) of also producing a genome browser which would involve data visualisation capabilities.

1 Introduction

Over the years, there has been an increase in sequencing of human genomes in order to identify genetic diseases. This sequencing is stored in a reference genome which is considered as a representative sample of a population's set of genes or DNA, where each individual's DNA corresponds to an arrangement of approximately 3 billion bases. Each base can be one of four possible nucleotides: adenine, cytosine, guanine and thymine - oftenly referred to by the letters A, C, G and T respectively. DNA (deoxyribonucleic acid) could be described as a model of biological life which encompasses genetic information/genes [1, 2].

Genome assembly/sequencing technologies are rapidly advancing and their costs are decreasing. DNA variations and mutations may correlate to diseases so finding differences between a reference genome and the genome of a patient afflicted with a disease has important implications for medical diagnosis and treatment [3]. After all, genes are hereditary from one generation to the next.

The University of Malta is developing a National Maltese Human Reference Genome for this reason whereby whole genome sequencing on certain Maltese DNA samples (provided by the Malta BioBank) will be performed. The American human genome sequencing facility Complete Genomics, which was founded in 2006, is a partner in this project.

The problem of aligning genomes comes into play here. In order to accurately and efficiently match alignments between a human reference genome and the respective DNA sequencing reads, different algorithms must be applied. A read is a fragment of an individual's genome as depicted hereunder:

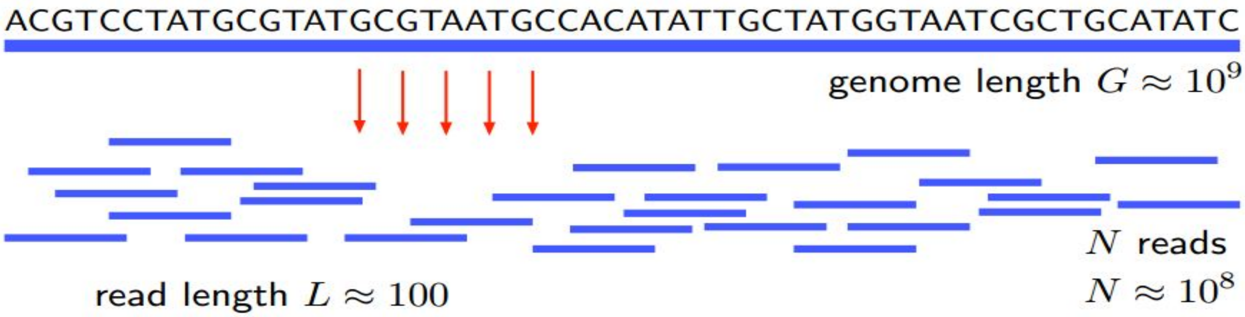

ACGTCCTATGCGTATGCGTAATGCCACATATTGCTATGGTAATCGCTGCATATC
genome length $G \approx 10^9$
N reads
$N \approx 10^8$
read length $L \approx 100$

Figure 1: DNA sequencing reads

The most common aligners use Burrows-Wheeler Transform [4, 5] but there are multiple other algorithms; all of these being explained further on.

Thus, computation wise, a tool which deals with DNA analysis and visualisation needs to be constructed in order to aid in this endeavour. These developments are only possible after the alignment of a set of sequencing reads against a reference genome is completed.

1.1 Motivation

A National Maltese Human Reference Genome is required in order to identify any DNA mutations which are predominant in the Maltese population. Comparing the provided Maltese DNA samples with any other human reference genome, for example with a Japanese reference genome, would not provide any accurate results as Maltese and Japanese people have a very diverse gene pool. This is due to the fact that both populations are descendant from diverse civilisations, so both of their evolutionary histories are different. This same fact can be said for any two distinct nationalities, thus providing motivation for a Maltese reference genome.

1.2 Why is this a non-trivial problem?

Globally, large genome projects are being sequenced rapidly. Examples of such projects include the 100,000 Genomes Project in the UK and the International Cancer Genome Project [4, 6, 7]. Malta should aim to be a part of this endeavour so that further studies can be conducted involving a larger variety of genes (i.e. the Maltese genes).

DNA is analogous to a storage device of genetic information. After all, while the reading of an individual's DNA shows the likeliness of that person developing a disease, the reading of a nations DNA shows why that population is more likely to develop a disease [3]. Such analyses can therefore prove to not only medically aid global research but also nation-wide studies.

Hence the problem is non-trivial as Malta has a diverse evolutionary history and gene pool. By developing the tools required for the Malta Human Genome Project to be a success, Maltese medical research will surely take a step forward in the right direction.

2 Background Research and Literature Review

The development of a genome assembly technology consists of multiple stages; the main ones being genome compression, sequence alignment, alignment visualisation and genome browser construction (successively). The following points detail some research which has already been conducted in these areas.

2.1 Genome Compression Tools

Genomes, particularly human ones, consist of a large amount of data - approximately 3,000 Mb (megabase pairs). For this reason, in order to efficiently analyse them, said genomes are compressed using various methods.

For example, Chen et al devised the lossless *GenCompress* algorithm which implements certain established compression algorithms such as Lempel-Ziv [8] (a variable-to-fixed-length code which parses the input sequence into non-overlapping DNA fragments of differing lengths while also constructing a dictionary of the fragments observed). By analysing approximate matches based on the evaluated edit distances, it was found that such a method not only achieves the best compression ratio but also finds common sections in DNA sequences. Another example would be the lossless reference-based compression algorithm devised by Fritz et al [6]. It implements established components such as Golomb codes (optimal prefix codes, i.e. no codeword is a prefix of any other codeword in the relative system) and De Bruijn graphs (directed graphs symbolising overlaps between sequences) and was found to be quite efficient for read alignments similar to the reference genome. Other known algorithm are *Biocompress-2* and *Cfact* [8] along with *DNACompress* and *DNAZip* [6].

2.2 Read Alignment Tools

Next-Generation Sequencing (NGS) technologies are evolving rapidly and to keep up with this evolution, multiple read aligners are being produced; the most known being BWA, Bowtie, Soapv2, MAQ, BOAT, SHRiMP2 for the NGS platforms Illumina, Roche454 and ABI SOLiD [4, 7, 5]. Reads are nucleotide sequences which are gathered from DNA by sequencers as depicted previously in Figure 1. These sequencers are given such importance due to the fact that genomic analysis would prove to be impossible without them. After all, reference genomes are produced using them so without this basic building block, no initial analysis can even commence.

The majority of read aligners implement Burrows-Wheeler Transform (BWT - permutes the bases of a sequence into another sequence) but there are also those which use FM-indexing (finds the number of occurrences of a read within a genome along with each occurrence's position), Needleman-Wunsch (finds similar regions between nucleotide sequences by comparing DNA fragments of differing lengths), and Smith-Waterman (a variation of Needleman-Wunsch) algorithms [4, 7, 5]. All these mentioned aligners are quite efficient relative to the task they are given; for example, while some aligners may be more efficient with short reads, others may be more efficient with longer reads [9, 10]. Other known algorithms which assist in approximate read mapping are the Hamming distance as well as the Levenshtein distance, both of which return a measure of the similarity between two sequences. These are not as commonly used due to their relative shortcomings. Both have an inefficient time complexity and the Hamming distance algorithm can even prove to be inaccurate as it only considers substitutions, unlike the Levenshtein distance which not only considers substitutions but also insertions and deletions.

2.3 Genome Browser Construction

Genome browsers are graphical interfaces which are used in conjuction with genomic databases. They display the information found such that individuals are able to not only browse the stored genomes but also to visualise them, leading to easier data seaching and analysis. Most browsers are web-based applications which allow certain customisations according to the user's requirements, but stand-alone browsers exist as well [11, 12]. An example of a web-based genome browser is *GBrowse* which was constructed by Stein et. al [12].

Genome browser construction has an intricate development process. Fortunately, many genome browser frameworks have already been constructed such as the most popular *GBrowse* (which was mentioned previously) as well as *Ensembl*, *JBrowse* and *LookSeq* among others. It should also be noted that there are two types of web-based genome browsers, multiple-species and species-specific browsers [11]. As this project deals with a human genome, a species-specific browser will be implemented. Following the Generic Model Organism Database (GMOD) project, there are multiple open-source tools for this type of browser; *GBrowse*, again, being one of the most used frameworks [11].

2.4 Gaps in Current Research

Firstly, regarding the genome compression tools discussed above, research about specific compression algorithms has been conducted however compression using primitive techniques such as integer conversion have not been researched. Secondly, regarding the read alignment tools also discussed above, most aligners do not use Hamming or Levenshtein distance due to their shortcomings so this will be analysed. Finally, regarding genome browser construction (which also deals with alignment visualisation), not a lot has been done in relation to comparing genome browsers against each other in terms of their features and applicability.

3 Aims and Objectives

The aim of this project is to develop algorithms and build tools for the analysis of sequenced genomes from the Malta Human Genome Project. The research areas studied are bioinformatics, big data, data storage, data visualisation and data analysis among others. The main objectives of the system are as follows:

1. The compression of a reference genome such as that done by Chen et al. [8] and Fritz et al. [6];
2. The alignment of a reference genome against a number of sequencing reads such as that done by Lee et al. [5] and Li et al. [4];

Depending on project and time constraints, the following could also be completed:

1. The construction of a genome browser using novel and established components (such as that done by Stein et al. [12]) in order to reference and visualise DNA for comparitive genomics and to analyse DNA mutations.

4 Methods Development

This section details the components which have been implemented along with a description of future plans.

4.1 Genome Compression

The genome compression method implemented makes use of certain in-built Python modules in order to compress the human genome, having a size of approximately 3GB, into a binary file by means of integers (each integer being of length 4 bytes). Each nucleotide base is assigned two bits to represent it as there are only 4 possible bases. Chen et al. used the same deduction when developing *GenCompress* [8].

Two possible implementations were considered. One compressed the genome into an integer and the other compressed it into a bit array. The former acheived a compression rate of 70% while the latter achieved a 75% rate. However, when analysing the accuracy of the compression, the former proved to be more accurate as the latter innately appends 0s to the integers produced in the bit array. The implementation chosen was thus the one which compresses the genome into an integer (the former one) as it is more accurate and it achieves a relatively similar compression rate (from 3GB to 1GB).

4.2 Sequence Alignment

Four possible alignment methods were then taken into consideration; the number of human sequencing reads being 28,094,847 with each read having a length of 60 bases; i.e. 30 base-pairs/bp. The Hamming distance method evaluates the number of mismatches between two sequences with regards to substitutions, the Levenshtein/Edit distance method evaluates the number of mismatches between two sequences with regards to substitutions, insertions and deletions, k-mer indexing evaluates all the possible subsequences of a sequence of length k into an index and the most frequently used FM-indexing evaluates all the possible subsequences of a sequence using BWT into an index. BWT transforms a base sequence into multiple runs having similar bases.

These aligners were first implemented and tested on string inputs; i.e. on an uncompressed genome of marginally less size than the human genome to confirm their applicability. Some of these functions were then converted in order to instead support integer inputs due to the previously mentioned genome compression. This was done by using bitwise operations and by finding any patterns in the integer sequence; i.e. any repetitive integers or pairs of integers or triples and so on in the compressed reference genome.

4.3 Alignment Visualisation

The data visualisation tool's initial construction is based on *Tkinter*, Python's standard Graphical User Interface (GUI) package. The result is a depiction of the reads aligned with the genome, with lines representing the position and length of each alignment/match. As of now, its interactivity is in the form of clicking a line at a certain position to output the offset of the match at said point of clicking.

4.4 Future Plans

As for future developments, firstly, the read alignment methods need to all be converted to support integer inputs and then compared in order to deduce the most efficient one in terms of the match rate with the corresponding time taken. A possible improvement on the BWT implementation could be done by incorporating the MapReduce parallel programming model, as described by Menon et al. [16], in order to accelerate said transformation. This model was originally produced by Google to query trillions of web pages, but it has started to be included in multiple other areas which involve large datasets due to its parallelisation feature; i.e. simultaneous process execution. Secondly, the data visualisation tool needs to be further refined; i.e. become more interactive in terms of genome analysis. This could be accomplished by outputting the DNA mutation found at the point of clicking instead of just the offset, or perhaps even both. Optionally, depending on the project's time constraints, the genome browser tool could also be constructed using the established technologies (described in section 3) for efficient comparitive genomics; the visualisation tool being incorporated in this. Each method developed will be analysed and compared to deduce the most feasible implementation.

5 Evaluation Stategy

The main evaluation techniques proposed are as follows:

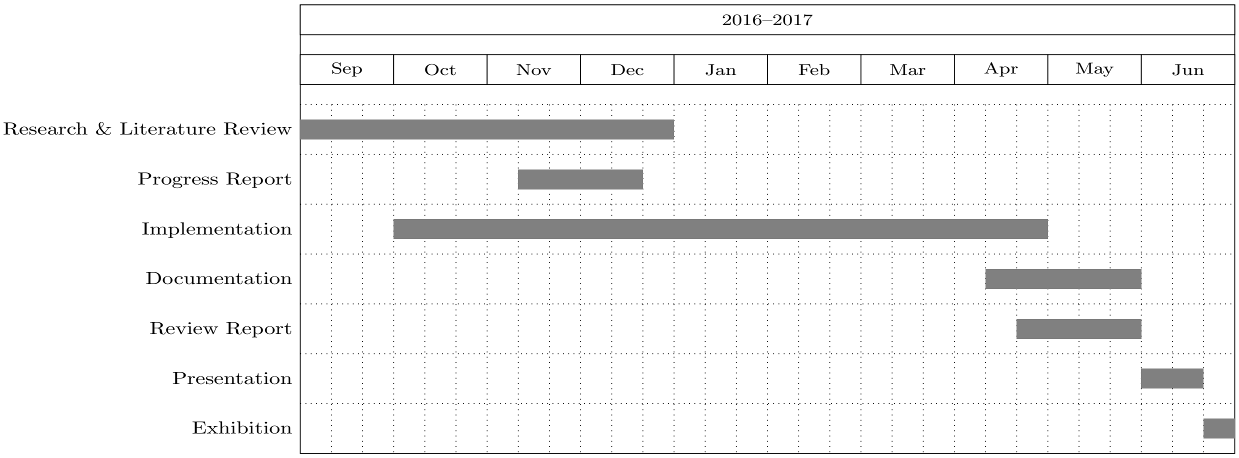
1. Comparitive review of the developed read aligner with the other developed aligners and with existing methods by comparing match rates with the corresponding time taken;
2. Review of the developed human genome visualisation tool by testing that its interface is easily understood by most users by, for example, checking that the data is output in a clear manner;
3. Possible comparitive review of the constructed genome browser by referencing DNA and analysing DNA mutations followed by comparison of results with existing methods.

6 Deliverables

1. The Final Year Report which will include all the relevant background information required to understand said project, a detailed explanation of the system and the evaluation results;
2. The implementation (code) of the designed and developed system;
3. The documentation which will explain to the users how the system should be employed; i.e. a user manual.

6.1 Project Timeline

The milestone schedule is represented in the following Gantt chart:



References

[1] L. Hunter, *Artificial Intelligence and Molecular Biology.* AAAI Press, 445 Burgess Drive, Menlo Park, California 94025, USA: MIT Press, 1993.

[2] A. M. Lesk, *Introduction to Genomics.* Great Claredon Street, Oxford, OX2 6DP, UK: Oxford University Press, 2 ed., 2012.

[3] "The hidden history of the maltese genome," *Think Magazine*, vol. 16, pp. 19–25, Apr. 2016.

[4] H. Li and R. Durbin, "Fast and accurate short read alignment with burrowswheeler transform," tech. rep., Wellcome Trust Sanger Institute, Wellcome Trust Genome Campus, Cambridge, CB10 1SA, UK, May 2009.

[5] D. Lee, F. Hormozdiari, H. Xin, F. Hach, O. Mutlu, and C. Alkan, "Fast and accurate mapping of complete genomics reads," tech. rep., Oct. 2014.

[6] M. H.-Y. Fritz, R. Leinonen, G. Cochrane, and E. Birney, "Efficient storage of high throughput dna sequencing data using reference-based compression," tech. rep., United Kingdom, Jan. 2011.

[7] L. Huang, V. Popic, and S. Batzoglou, "Short read alignment with populations of genomes," tech. rep., 2013.

[8] X. Chen, S. Kwong, and M. Li, "A compression algorithm for dna sequences and its applications in genome comparison," tech. rep., New York, NY, USA, Apr. 2000.

[9] M. Ruffalo, T. LaFramboise, and M. Koyutrk, "Comparative analysis of algorithms for next-generation sequencing read alignment," tech. rep., July 2011.

[10] J. Shang, F. Zhu, W. Vongsangnak, Y. Tang, W. Zhang, , and B. Shen, "Evaluation and comparison of multiple aligners for next-generation sequencing data analysis," tech. rep., Mar. 2014.

[11] JunWang, L. Kong, G. Gao, and J. Luo, "A brief introduction to web-based genome browsers," tech. rep., July 2012.

[12] L. D. Stein, C. Mungall, S. Shu, M. Caudy, M. Mangone, A. Day, E. Nickerson, J. E. Stajich, T. W. Harris, A. Arva, and S. Lewis, "The generic genome browser: A building block for a model organism system database," tech. rep., Dec. 2002.

[13] B. Berger, J. Peng, and M. Singh, "Computational solutions for omics data," tech. rep., USA, Mar. 2014.

[14] A. M. Lesk, *Introduction to Bioinformatics.* Great Claredon Street, Oxford, OX2 6DP, UK: Oxford University Press, 4 ed., 2014.

[15] D. W. Mount, *Bioinformatics: Sequence and Genome Analysis.* Cold Spring Harbor Laboratory Press, Cold Spring Harbor, New York, USA: Cold Spring Harbor Laboratory Press, 2001.

[16] R. K. Menon, G. P. Bhat, and M. C. Schatz, “Rapid parallel genome indexing with mapreduce,” tech. rep., June 2011.