Master's Thesis

University of Rennes 1

Bioinformatics and genomics Master's Degree

(2014 - 2015)

Test and benchmarking of a new scaffolding methodology

Institute for Research in IT and Random Systems, Genscale
263 Avenue General Leclerc, 35000 Rennes, France

Author:
Alexandrina Bodrug

Supervisors:
Pr. Univ. Rennes 1 Rumen Andonov
Dr. CNRS Dominique Lavenier

22 June, 2015

Thanks

Abbreviations & github link

Contents

Introduction

Backgroud

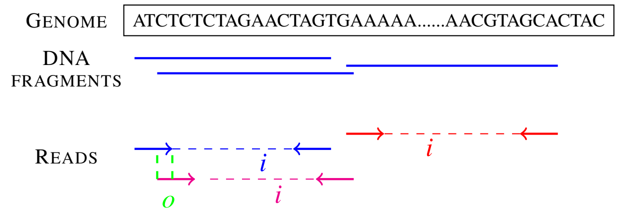
De novo assembly is the process which pieces together overlapping small fragmented DNA sequences produced by Next Generation Sequencing methods into larger sequences. The aim is to obtain complete genomes (or chromosomes) containing gaps of known lengths because the less fragmented the genome is, the easier the downstream analysis are[1]. However an incomplete assembly is still sufficient for most of the analysis performed on DNA which explains why databases mainly contain partially assembled genomes. Nonetheless the uninterrupted genome sequence is a precious information and there has been an important effort made to improve the performance of assembly algorithms and the quality of NGS data. The the detailed process of assembly is described in subsection 1.2 Assembly terminology; the two main steps are building contigs from reads (sometimes referred to as assembly) and scaffolding, the ordering and relative orientation of contigs or unitigs. The 2011 and 2013 Assemblathon projects[2,3] aimed at benchmarking existing assembly tools with high coverage diploid genomes. The studies focused mainly on the contig building step, concluding that although many tools found quality assemblies, the tool and quality criteria should be adjusted to the type of genome and the goal of the assembly project. For example a good N50, an extensively used metric which is the contig length such that using equal or longer contigs produces half the bases of the genome, is not essential in a gene detecting assembly project.
The first stand-alone scaffolder named Bambus[4], originally part of the MetAMOS[5] assembly and analysis pipeline, was published in 2004. Previously the scaffolding step was missing or presented as an option within conting builders, for instance the Velvet[6] assembler *'scaffolding yes or no'* option. In the 2014 comprehensive evaluation of scaffolding tools[7], Hunt *et al* found that no tool identified more than 90% of joins between real-data Velvet assembled contigs, meaning genomes were still fragmented into many scaffolds as joins were missing for a complete and accurate ordering and orientation. The study also used simulated data highlighting the fact that perfect data doesn't always yield perfect results. Despite its simply formulated goal - order and orient contigs - scaffolding is a challenging computational problem. It was first described and modeled in 2002 by Hudson *et al.*[8] which proposed a greedy path-merging strategy, described in subsection 1.3 A history of scaffolding strategies along with other proposed algorithms.

Assembly terminology

In this report *assembly* will refer to the whole multi-step process which starts from once filtered out-of-the-sequencer data and results, in the best case scenarios, in highly uninterrupted sequence of a genome or chromosome. As previously mentioned, the two main steps are contig/unitig building and contig/unitg scaffolding. The difference between contig and unitig is fundamental to understanding the Genscale scaffolding challenges. Another key point is the construction of joins between contigs/unitigs - also referred to as links, edges, bonds . . .

Reads, pairing and overlaps

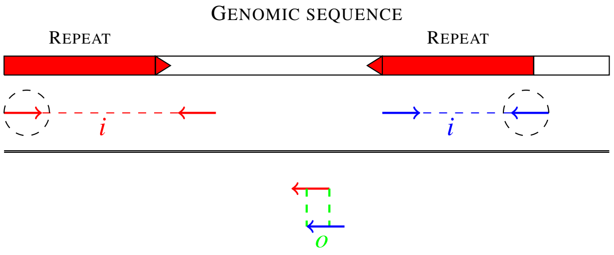
A read is a short $(< 500pb)$ copy of a DNA fragment of known length and nucleic acid order. It is produced differently depending of the sequencing technology. Paired reads are copies of the two extremities of a DNA molecule. The DNA sequence between two reads of a pair is called an insert. The size of the insert is variable. Reads with small insert sizes $(< 500bp)$ are called paired-end reads. Mate-paired reads are reads whose insert size is very big (up to tens of kilobases). The pairing information and the size of the insert are provided by the sequencer. A collection of reads with their associated insert size is called a a genomic library.



Each end of a DNA molecule is cloned to produce paired reads. Here is represented a mate-paired pair (red) with a big insert size *(i)* and two paired-end pairs (blue and magenta) which slightly overlap *(o)*.

Figure 1: Alignment of paired reads on fragmented DNA

Figure 1 represents three pairs of reads. Within the pairs, reads are facing each other: this configuration is called *Forward-Reverse* read orientation. To be sequenced the genome represented in figure 1 is first amplified by Polymerase Chain Reaction and then fragmented into numerous DNA molecules by sonication or nebulization. Each end of the molecule is then cloned. Overlapping of reads occurs when two reads sequence a portion of the same genomic region, but not only. The overlapping concept implies a common origin but unfortunately overlapping can occur if two reads sequence two different repeated genomic regions. Figure 2 shows how repeated regions create false positive overlaps. Such reads can be detected and filtered out by ignoring high-frequency overlaps (higher than the coverage at which the genome was sequenced). However this can result in false negatives and makes the task of assembling repeated regions very hard.

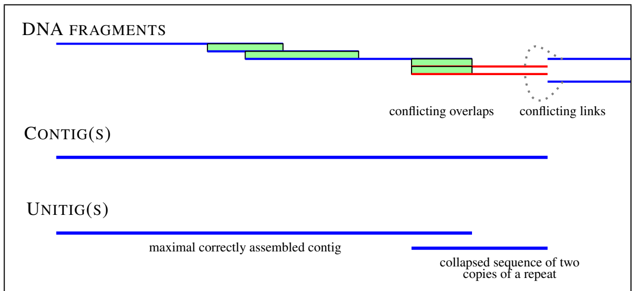


The two circled reads will have a significantly long and accurate overlap to imply a common genomic origin when in fact they come from distant regions.

Figure 2: Overlapping induced by repeated sequences

Unitigs and Contigs

Unitigs, also sometimes called chunks, are an uniquely assemblable subset of overlapping fragments. At the end of an unitig data shows multiple dubious overlaps as seen in subsubsection 1.2.1 Reads, pairing and overlaps creating joins with multiple other unitigs. Contigs are larger than unitigs, extended through repeat boundaries but are still ungapped sequences. Contigs are interesting to construct because there is a higher chance to detect genes, despite the risk of misassemblies and chimeric sequences. Taking the example shown in figure 2, a contig could cover all the green repeated area and afterwards be extended though ambiguous overlaps on both sides. Unitigs however will stop at the end of the green areas. In a sense, unitigs are either an unambiguous contig or a compression of several copies of a repeat. See example in figyre 3.


DNA FRAGMENTS
conflicting overlaps
conflicting links
CONTIG(S)
UNITIG(S)
maximal correctly assembled contig
collapsed sequence of two copies of a repeat

Unitigs end at multiple overlaps indicating a possible repeat. Contigs can be extended through conflicting overlaps. Here, the red DNA fragments are two copies of a repeat. When no more overlaps exist, contigs can be linked (gray dotted line) thanks to information provided by read pairs. This is the scaffolding task. Here alternative paths are possible due to the repeated region.

Figure 3: The difference between unitigs and contigs

Scaffolding and obtaining a consensus sequence

A scaffold is a linear ordering of contigs (or unitigs). The ordering and relative orientation of contigs is possible thanks to paired reads information. The first step of scaffolding is mapping reads on the previously constructed reads: the two most used mappers are bwa[9] and bowtie[10;11]. A pair of reads mapping on two different contigs provide a join, which holds the information of distance between the two contigs, and relative orientation (see figure 4). The concept of insert size is essential to understand the challenges of scaffolding.

Note

Figure 4: Creating joins between contigs thanks to read-pair information

A history of scaffolding strategies

The contig scaffolding problem was first introduced in 2002 by Hudson and al.[8] following the challenges which arose during the clone-bt-clone Whole Genome Project and whole genome shotgun assembly project. Most tools model the problem as a graph where vertices represent contigs and links represent bundles of pairs of reads linking two contigs. Erroneous data (fake links due to poorly assembled contigs, low quality libraries), missing data (low quality libraries, unfit insert size, low genome coverage) and inherent genome characteristics (repeated regions, heterozygosity) stand in the way of a perfect and easy scaffolding process.

Context and goal of the internship project

Material and methods

Input data for genscale scaffolders

The input data for the genscale scaffolders contains a list of unitigs and a list of links between unitigs.

Format

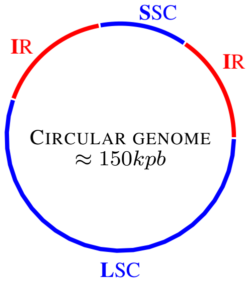
Methods

Features of the assembled genomes

Chloroplasts

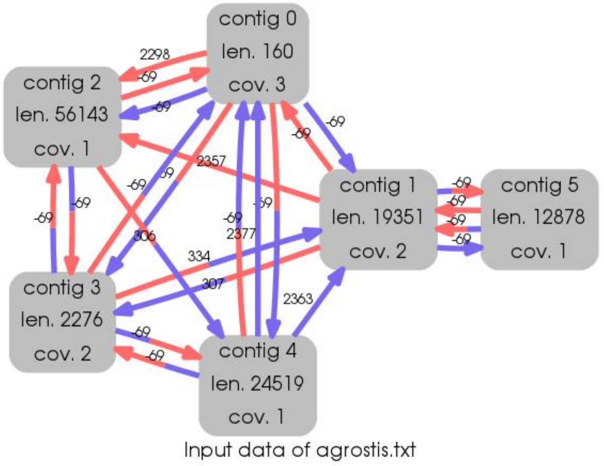
Chloroplasts are small organelles in plant photosynthetic tissues which possess their own DNA. The chloroplast genomes are small ($\approx 150kpb$), circular and have a large inverted repeated sequence of around $25kpb$.
The instances used for this study are the chloroplast genomes of the following organisms: *Eucalyptus globulus, Acorus calamus, Atropa belladonna, Agrostis stolonifera, Cucumis sativus, Lecomlella madagascariensis, Oenothera elata, Pinus koraiensis* and *Euglena gracilis*. Despite having the same general structure, the amount of repeated sequences is different from an instance to another, making them more or less easy to assemble.

Other genomes



Inverted Repeat (IR $\approx 23kpb$) ; Long Single Copy (LSC); Small Simple Copy (SSC $\approx 85kpb$)

Figure 5: Chloroplast genome structure



The image was generated with the *graph_generator.py* script. The color on the links indicate if the links points to the contigs in forward (blue) or reverse (red) orientation.

Figure 6: Input data of *Agrostis stolonifera* viewed in form of a graph

Genscale scaffolding strategy

Benchmarking

Published scaffolders chosen for benchmarking

Benchmarking strategy

Chosen scaffolding tools to benchmark against Benchmarking workflow Comparisons QUAST Comparison function Visualization MUMMER Visualization tool

Results

Data sets compared

QUAST and comparison function

Visualization

Conclusion

References

[1] Hunt, M., Newbold, C., Berriman, M. & Otto, T. D. A comprehensive evaluation of assembly scaffolding tools **15**, R42.

[2] Earl, D. *et al.* Assemblathon 1: A competitive assessment of de novo short read assembly methods **21**, 2224–2241. URL http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3227110/.

[3] Bradnam, K. R. *et al.* Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species **2**, 10. URL http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3844414/.

[4] Pop, M., Kosack, D. S. & Salzberg, S. L. Hierarchical scaffolding with bambus **14**, 149–159. URL http://genome.cshlp.org/content/14/1/149.

[5] Treangen, T. J. *et al.* MetAMOS: a modular and open source metagenomic assembly and analysis pipeline **14**, R2. URL http://genomebiology.com/2013/14/1/R2/abstract.

[6] Zerbino, D. R. & Birney, E. Velvet: Algorithms for de novo short read assembly using de bruijn graphs **18**, 821–829. URL http://genome.cshlp.org/content/18/5/821.

[7] Hunt, M., Newbold, C., Berriman, M. & Otto, T. D. A comprehensive evaluation of assembly scaffolding tools **15**, R42. URL http://genomebiology.com/2014/15/3/R42/abstract.

[8] Huson, D. H., Reinert, K. & Myers, E. W. The greedy path-merging algorithm for contig scaffolding **49**, 603–615. URL http://doi.acm.org/10.1145/585265.585267.

[9] Li, H. & Durbin, R. Fast and accurate short read alignment with burrows-wheeler transform **25**, 1754–1760.

[10] Langmead, B. & Salzberg, S. L. Fast gapped-read alignment with bowtie 2 **9**, 357–359. URL http://www.nature.com/nmeth/journal/v9/n4/full/nmeth.1923.html.

[11] Langmead, B., Trapnell, C., Pop, M. & Salzberg, S. L. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome **10**, R25. URL http://genomebiology.com/2009/10/3/R25/abstract.