Genome Assembly

Arif Harmanci

February 3rd, 2011

Outline

- What is genome assembly?
  - Contig generation
  - Scaffolding
- Mapping and Comparative Assembly
- De-novo Assembly
  - Greedy approaches
  - Overlap-Layout-Consensus (OLC) based approaches
  - Overlap graphs vs de-Bruijn graphs
    - Hamiltonian vs Eulerian path problems
  - Read error correction (Many reads, which have so many errors create lots of paths that do not contribute to anything. Therefore, the read error correction is important even if it introduces errors)
- Paired end data (Scaffolding, path resolution, graph simplification)
- Complications from sequencing
- Assembly validation
- Conclusions

References

**** Pevzner et. al. "An Eulerian path approach to DNA fragment assembly". *Proceedings of National Academy of Sciences*, 2010.

*** Paskiewicz et. al. "De novo assembly of short sequence reads". *Briefings in Bioinformatics*, 2010.

*** Pop et al. "Genome assembly reborn: recent computational challenges". *Briefings in Bioinformatics*, 2009.

** Nagarajan et al. "Sequencing and genome assembly using next-generation technologies". *Methods in molecular biology*, 2010.

** Schatz et. al. "Assembly of large genomes using second-generation sequencing". *Genome Research*, 2010.

* Miller et al., "Assembly algorithms for next-generation sequencing data". *Genomics*, 2010.

What is genome assembly?

- Assembling a genome sequence from reads: Sanger, NGS
- Input: Many reads (typically billions)
- Output: Ordered/unordered set of contigs
- Genome assembly is possible because the total set of reads overlap and *cover* the genome in a redundant fashion
- Like solving a puzzle
  - More pieces → Harder problem
  - Tradeoff between size of the pieces vs. the number of pieces
    - Sanger, Roche 454: Millions of approximately kilobase long reads
    - Illumina, SOLiD: Billions of $35 - 45$ base long reads
    - Helicos and Pacific Biosciences sequencers: Not applied to large scale, yet.
  - Complications:
    - Repeats complicate the assembly process
    - The sequencing errors and biases

Steps of genome assembly

It is typically not possible to regenerate the whole genome in the assembly

- Contig assembly: Assemble the reads into *contigs*
  - The algorithms mostly concentrate on getting contigs together, first.
    - Contig-only assemblies are useful for certain applications, e.g., transcriptome sequencing, ChIP-Seq experiments
  - Contigs are usually flanked by regions that cannot be resolved:
    - Repeats
    - Low read depth regions
- Scaffolding: Stitching of the contigs into *scaffolds*
  - Mate-pair information is highly valuable
  - Optical restriction maps can be utilized for post-processing the assembled contigs [Samad et al., 1995].

Mapping and Comparative Assembly (Genome Re-sequencing)

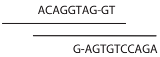
- A genome for a related organism is available
- The reads are mapped to the available genome
- The mapping is used a guide in assembly of the reads
- Methods for read mapping:
  - Index the genome or the reads (Suffix/prefix trees, hash tables)
  - Burrows-Wheeler Transform

Read Mapping

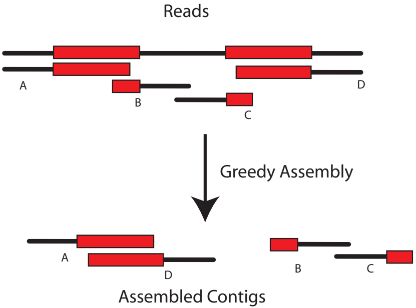
- Indexing: Restructure the data such that searching takes (approximately) constant time
    - Sacrifices memory for fast searching
    - Usually do not allow inserts/deletes but certain number of mismatches in the mapping
        - MAQ, SOAP, SHRiMP
- Burrows-Wheeler Transform: Enables indexing/searching with much less memory requirement
    - BWA, Bowtie and TopHat(for mapping RNA-seq reads to splice junctions, distributed with BowTie)

De-novo Assembly: Greedy Approach

- Start by the best overlapping pairs of reads and extend those.



Example of an imperfect overlap



- Wrong assemblies must be corrected in the later stages.
- Sanger sequencing: phrap, TIGR Assembler,
- NGS: SSAKE, VCAKE, SHARCGS.
  - Contigs are initiated with the best overlapping reads and extended with next best overlapping reads

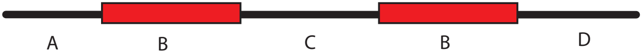
Overlap-Layout-Consensus (OLC) Approach

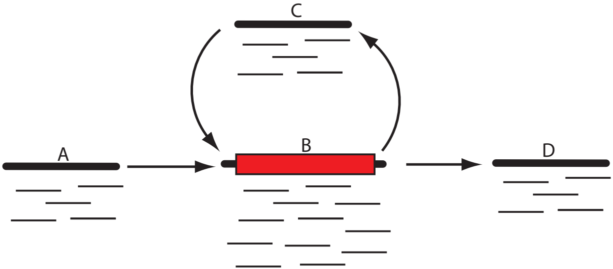
Breaks assembly into three distinct steps:

- **Overlap:**
  - Do pairwise comparison of all the reads (Very compute intensive) to identify overlaps Generate the *overlap graph*
  - Each read is a node, edges connect nodes for which an overlap is detected
- **Layout:**
  - The overlap graph is analyzed for the paths that define contigs
  - Celera Assembler uses the terminology *unitigs*: The uniquely assemblable contig
  - The unitig construction is intentionally conservative
- **Consensus:**
  - A multiple sequence alignment of the reads determines the final layout of the assembly

Arachne , Newbler , Celera Assembler , Minimus , CAP3 , CABOG , Edena

Overlap-Layout-Consensus (OLC) Approach

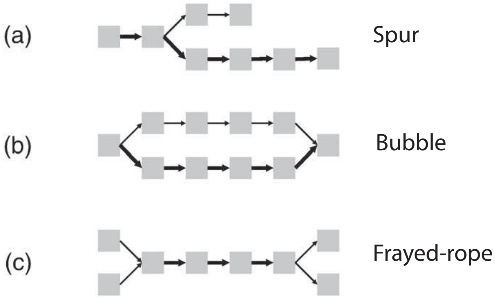




Overlap Graph

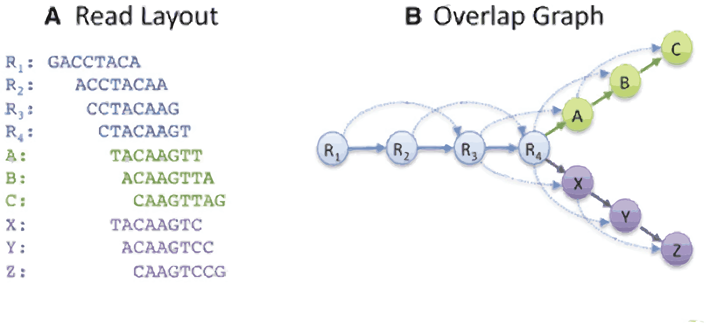
- The repeat introduces a *fork* in the overlap graph.

Graphical Motifs to Resolve



- Once the overlap graph is built, the problem boils down to resolving the *tangles* introduced by sequencing errors and repeats
- Tangles may not be so bad:
  - The tangles may become useful for inferring isoforms while assembling transcriptomic data.

Overlap Graphs



A Read Layout

R_1:	GACCTACA
R_2:	ACCTACAA
R_3:	CCTACAAG
R_4:	CTACAAGT
A:	TACAAGTT
B:	ACAAGTTA
C:	CAAGTTAG
X:	TACAAGTC
Y:	ACAAGTCC
Z:	CAAGTCCG

B Overlap Graph

- Sequencing error caused a fork in the overlap graph.
- Rather complex representation for the reads.
- The solution requires finding a path that passes through all the nodes in the graph
  - Hamiltonian Path problem: NP-complete
  - Other alternative is the de-Bruijn graph approach

de-Bruijn Graph/Eulerian Path Approaches

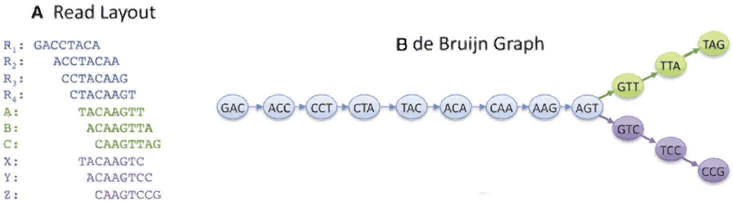
- For each read, generate k-mers starting at each position
  - Referred to as k-mer spectrum
- For AGCTGCCGA, 4-mers are:

```
AGCTGCCGA
AGCT
 GCTG
  CTGC
   TGCC
    GCCG
     CCGA
```

- Each k-mer is utilized separately in building the de-Bruijn graph.
- Seems counterintuitive since the connectivity information is lost, but it is incorporated later.
- The constant k-mer size enables indexing and fast-searching of the k-mers against other k-mers.

de-Bruijn Graph/Eulerian Path Approaches

- Each node is a (k-1)-mer.
- Each k-mer is represented by an **edge** in the de-Bruijn graph such that the source and destination nodes overlap with 1 nucleotide offset.



A Read Layout

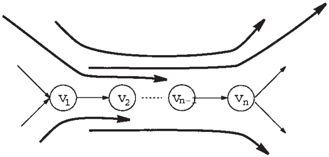
```
R1: GACCTACA
R2:  ACCTACAA
R3:   CCTACAAG
R4:    CTACAAGT
A:      TACAAGTT
B:       ACAAGTTA
C:        CAAGTTAG
X:      TACAAGTC
Y:       ACAAGTCC
Z:        CAAGTCCG
```

B de Bruijn Graph

- Much simpler representation
- Assembly requires finding a path that passes through all the edges
    - Eulerian path problem: Algorithms with linear time complexity (in the number of edges) exist

de-Bruijn Graph/Eulerian Path Approaches

- Graph simplification: Follow the *read-paths*



Uppermost path resolves the tangle

- Spur pruning
- Read threading: Extend the read paths that are consistent (with the smallest number of mutations) with consecutive reads
- Equivalent Transformations : *Detachment*, *Cut*

de-Bruijn Graph/Eulerian Path Approaches

- Eulerian approach was proposed for sequencing by hybridization (SBH) technology
  - Generate the k-mer spectrum of the genome (at every nucleotide position) via chips/microarrays
- The technology was not successful due to experimental difficulties but the computational methodology is perfectly suitable for short read sequencing
- Velvet, ABySS, ALLPATHS, SOAPdenovo

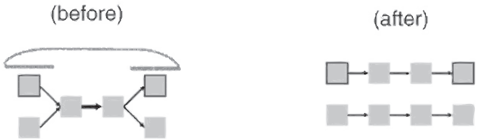
de-Bruijn Graph/Eulerian Path Approaches: Error correction/Graph simplification

- Read errors complicate the graph:
  - Short reads, high coverage, and high error rates makes the graph too *tangly*
- Prunes/Corrects the reads whose k-mers have low frequencies in the whole k-mer spectrum
- Error correction
  - Spectral Alignment: For k-mers that have low frequency, k-mers are modified with the lowest number of substitutions so as to increase their frequency
    - The correction can mask the polymorphism information
    - It can change a correct k-mer which are part of read that are in low-coverage region, i.e., introduce an error
    - The authors claim that these are justifiable for the sake of completing the assembly

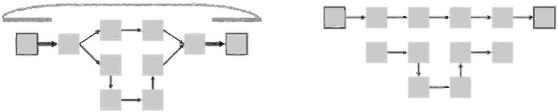
Paired End Data

- A paired end data is a very long read whose intermediate nucleotides are missing with a known length
- Useful for:
  - Graph simplification
  - Scaffolding: Utilize the paired ends that span between two contigs

Paired End Data



The paired end reads are consistent with upper path



The insert size for the paired end reads are consistent with upper path

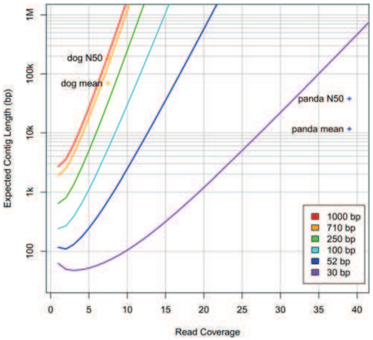
Complications from Sequencing

- The sequenced DNA is double helical:
  - The reverse strand should be accounted for
- Bias in sequencing:
  - GC bias (in Illumina reads)
  - The genome is not uniformly sampled while sequencing: Amplification bias
    - Cannot apply statistical tests to utilize read depths for inferring repeats
- Palindromes (AGGCGCCU) introduce self referring loops in the graphs
  - Velvet requires k to be odd.

Assembly validation and metrics of quality

- Contiguity: Distribution of lengths of assembled contigs
  - N_{50}: The length for which the total length of contigs that are longer than this length is 50% of the total length of all contigs
  - High $N_{50} \rightarrow$ More contiguous assembly, higher quality.
    - ABySS assembled 3.5 billion Illumina reads (from an African individual) resulting in 2.76 million contigs with N_{50} of 1.5 kbps in 4 days. Covers 68% of the human reference genome.
    - SOAP-denovo assembled same set of reads on a 32 core 512 Gbyte RAM computer, yielding N_{50} contig length of 4.6 kbps in 40 hours. Covers 85% of the reference genome.
- Accuracy:
  - Accuracy evaluation is based on comparing the predicted assemblies with the assemblies based on mapping the reads to reference genomes.

Assembly validation and metrics of quality: Computational models



- The average contig lengths estimated from computational model (Lander and Waterman 1988) and computed from assemblies of panda (52 bp reads) and of dog genome (710 bp reads).
- Changing read length also changes the expected percentage of repetition in the genome and invalidates the assumptions of computational model

Conclusions and Take Home Messages

- Genome assembly is very compute intensive
- Longer reads and paired end reads are vital to high quality genome assembly
  - Hybrid assembly: Utilize strength of different sequencing platforms
    - "Sanger-454" assembly [Goldberg et al., 2006]: Assemble with 454, shred then re-assemble with Celera Assembler.
- With new generation sequencing, the methods for short read assembly may become obsolete
  - May still be useful for other applications like transcriptome assembly

S. Goldberg, J. Johnson, D. Busam, T. Feldblyum, S. Ferriera, R. Friedman, A. Halpern, H. Khouri, S.A. Kravitz, F.M. Lauro, et al. A Sanger/pyrosequencing hybrid approach for the generation of high-quality draft assemblies of marine microbial genomes. *Proceedings of the National Academy of Sciences*, 103 (30):11240, 2006.

A. Samad, EF Huff, W. Cai, and D.C. Schwartz. Optical mapping: a novel, single-molecule approach to genomic analysis. *Genome research*, 5(1):1, 1995. ISSN 1088-9051.