De novo whole genome assembly

Qi Sun

Bioinformatics Facility
Cornell University
Sequencing platforms

• **Short reads:**
  - Illumina (150 bp, up to 300 bp)

• **Long reads (>10kb):**
  - PacBio SMRT;
  - Oxford Nanopore
Contiging/Scaffolding Platforms

- **Illumina**: Mate pair
- **10X Chromium**: Linked Reads
- **BioNano**: Optical Mapping
- **Dovetail; Phase Genomics**: Hi-C
- **NRGene**: proprietary DeNovoMAGIC software
- Long-read platforms (PacBio or Nanopore)
DNA fragment length vs Sequencing read length

DNA fragment

Sequencing Read: ACGGGAGGGGACCCG...
Short-read Sequencing Platform: Illumina

**Paired-end**
- Fragment: <500bp fragment
- Read: 100-250 bp

**Mate pair reads**
- Fragment: 5kb, 8kb or 15kb
- Read: 50 bp
Illumina Reads Assembly Strategy

Contiging

Paired-end reads

Contigs

Scaffolding

Mate-pair reads

Scaffolds
Long-read Sequencing Platform: PacBio SMRT

Fragment: >>10kb
Read: 10-20 kb

From PacBio web site
PacBio: High error rate (>10%)

Error correction methods
- Self error correction (100x depth)
- By Illumina reads

Oxford Nanopore

Error rate is equivalent or better than PacBio but much cheaper

MinION
From contiging to Scaffolding

Contigs

Using Mate-pair

Using Long reads

Using Physical map: optical map & Hi-C
Scaffolding strategies. Physical maps

Optical Mapping:
Generating high-quality genome maps by labeling specific 7-mer nickase recognition sites in a genome with a single-color fluorophore

Hi-C:
Sequence cross-linked and ligated DNA fragments

BioNano

Dovetail & Phased Genomics
10X Genomics

• < $5000 for assembled and phased genome;

• Large scaffold, capture the gene region very well
Genome assembly strategies

• The Goal
  ➢ Gene space vs whole genome;
  ➢ Contig vs Scaffold vs pseudo-chromosome;
  ➢ Resolving paralogs?
  ➢ Resolving haploid?

• The price
Genome assembly strategies

Bacterial genomes (<10Mb): Illumina paired-end reads

Eucaryotic genomes (<1Gb):
- PacBio
- Nanopore
- 10x

Ultimate high quality chromosomal level assembly:
- PacBio/Nanopore + BioNano + Dovetail/PhaseGenomics
- Genetic linkage map
Two categories of contiging strategies

Long reads  
overlap–layout–consensus

Short reads  
de-bruijn-graph

Canu, Falcon, Celera et al.  
Velvet, Soap-denovo, Abyss, Trinity et al

de-bruijn-graph for contiging short reads

Kmer Paths in paralogous regions

Paralogous regions

GGATGGAAGTCG..............CGATGGAAGGAT

7mer

GGATTGGA

CGATTGGA

GATGGAA

ATGGAAG

TGGAAGT

TGGAAGG

9mer

GGATGGAAGT

CGATGGAAGG

GATGGAAGT

GATGGAAGG
Sequencing errors and

Tips, bubbles and crosslinks

Deal with sequencing errors and repetitive regions

1. Sequencing errors
   • Remove low depth kmers in a bubble;
   • Too long kmers would cause coverage problem;

2. Repetitive region
   • Longer kmers

Tiny repeat:
Separate the path

Break boundary between low and high copy regions
Examine a de-bruijn graph assembly software: Soap denovo
Read depth vs Kmer depth

Read depth: number of reads at a genome position

Kmer depth: number of occurrence of an identical kmer

Read depth: 4

Kmer occurrence: 3
Impact of kmer-length (2)  
Read depth vs Kmer depth

Longer kmers could results in too little kmer depth
Estimate genome size based on kmer distribution

Kmer depth -> Read depth -> Genome size
Estimate genome size based on kmer distribution

Step 1: convert read depth to kmer depth

\[ N = \frac{M \times L}{(L-K+1)} \]

M: kmer depth = 112  
L: read length = 101 bp  
K: Kmer size = 21 bp  
N: read depth = 140

Step 2: genome size is total sequenced basepairs divided by read depth

Genome size = \( \frac{T}{N} \)

T: total base pairs = 0.505 gb  
N: read depth = 140  
Genome size: 3.6 mb
Estimate genome size with ErrorCorrectReads.pl from ALLPATHS-LG

```bash
ErrorCorrectReads.pl \
PAIRED_READS_A_IN=R1.fastq.gz \
PAIRED_READS_B_IN=R2.fastq.gz \
KEEP_KMER_SPECTRA=1 \
PHRED_ENCODING=33 \
PLOIDY=1 \
READS_OUT=corrected_out \
>& report.log &
```
Heterozygous genomes

Experimentally:

• Create inbred lines or haploid cell culture.
• Assembly of clonal fragments, merging allelic regions.

Assembly

• SNP: merge bubbles
• Highly polymorphic regions

Kmer coverage distribution

Highly polymorphic regions
Planning check list of whole genome assembly

- Sequencing platform;
- High quality DNA-extraction;
- Sequencing Read length and read depth;
- Contiging software;
- Scaffolding strategy software;
- Gene annotation strategy (RNA-seq recommended)
## Genome Assembly Software for Different Technology Platforms

<table>
<thead>
<tr>
<th>PacBio/Nanopore</th>
<th>Illumina</th>
</tr>
</thead>
<tbody>
<tr>
<td>Canu</td>
<td>Soap Denovo</td>
</tr>
<tr>
<td>Falcon</td>
<td>MaSuRCA</td>
</tr>
<tr>
<td><strong>10x</strong></td>
<td>Discovar</td>
</tr>
<tr>
<td>SuperNova</td>
<td>Platinus</td>
</tr>
</tbody>
</table>
Workflow of *de novo* assembly

- Clean sequencing data as required; (trim adapter and low quality sequences)
- Contiging, scaffolding, gap filling, polishing
- Evaluation of assembly;
Scaffolding and Polishing

<table>
<thead>
<tr>
<th>Commercial Service</th>
<th>Polishing Software</th>
<th>Hybrid data</th>
</tr>
</thead>
<tbody>
<tr>
<td>BioNano</td>
<td>Pilon</td>
<td>Using long reads for gap filling and scaffolding:</td>
</tr>
<tr>
<td>Dovetail</td>
<td>Racon</td>
<td>e.g. SPAdes;</td>
</tr>
<tr>
<td>Phase Genomics</td>
<td></td>
<td>PBJelly2</td>
</tr>
</tbody>
</table>

* Tend to cause errors
Running de-bruijn-graph assembly software

Test different kmer sizes

SOAP denovo2 on BioHPC Lab

/programs/SOAPdenovo2/SOAPdenovo-63mer [options]
/programs/SOAPdenovo2/SOAPdenovo-127mer [options]

Two binary codes with max kmer size 63 or 127

SOAPdenovo-127mer all -s config.txt -K 101 -R -o assembly

SOAPdenovo-127mer all -s config.txt -K 127 -R -o assembly
Multi-Kmer Approach in Soap Denovo2

- **Start building graph with small kmer;**
- **Iteratively rebuild kmer by mapping larger kmers to previous graph**

//Start

\[ k \leftarrow k_{\text{min}} \quad (k_{\text{min}} \text{ is set at graph construction ‘pregraph’ step}); \]

Construct initial *de Bruijn* graph with \( k_{\text{min}} \);

Remove low depth \( k \)-mers and cut tips;

Merge bubbles of the *de Bruijn* graph;

Repeat {
\[ k \leftarrow k + 1; \]

Get contig graph \( H_k \) from previous loop or construct from *de Bruijn* graph;

Map reads to \( H_k \) and remove the reads already represented in the graph;

Construct \( H_{k+1} \) graph base on \( H_k \) graph and the remaining reads with \( k \);

Remove low depth edges and weak edges in \( H_k \);

} Stop if \( k \geq k_{\text{max}} \) (\( k_{\text{max}} = k \) set in contig step(-m));

Cut tips and merge bubbles;

Output all contigs;

//End
# maximal read length
max_rd_len=101

[LIB]
# average insert size
avg_ins=300
# if sequence needs to be reversed
reverse_seq=0
# in which part(s) the reads are used
asm_flags=3
# in which order the reads are used while scaffolding
rank=1
# cutoff of pair number for a reliable connection (at least 3 for short insert size)
pair_num_cutoff=3
# minimum aligned length to contigs for a reliable read location (at least 32 for short insert size)
map_len=32
# a pair of fastq file, read 1 file should always be followed by read 2 file
q1=r1.fastq
q2=r2.fastq

/programs/SOAPdenovo2/SOAPdenovo-127mer all -s config.txt -K 127 -R -o assembly
Evaluating genome assembly

- Completeness
  - Estimated genome size vs assembly size;
  - Gene space completeness: BUSCO

- Contig/scaffold length
  - N50

- Contig/scaffold quality
  - Chimeric contigs/scaffolds;
  - Collapsed paralogs;
BUSCO
Evaluate the completeness in Genespace

BUSCO gene sets:
single-copy orthologs in at least 90% of the species in each lineage.

Arthropods  Vertebrates  Fungi  Bacteria
Metazoans  Eukaryotes  Plants
BUSCO assessment workflow

Sima~o et al. Bioinformatics, 2015, 1–3
# BUSCO Output

**C:** complete  **D:** duplicated  **F:** fragmented  **M:** missing  
(Report % of genes in each category)

<table>
<thead>
<tr>
<th>Species</th>
<th>Size</th>
<th>BUSCO notation assessment results</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>D. mela</em></td>
<td>139 Mbp</td>
<td>C:98% [D:6.4%], F:0.6%, M:0.3%, n:2675</td>
</tr>
<tr>
<td></td>
<td>13918 genes</td>
<td>C:99% [D:3.7%], F:0.2%, M:0.0%, n:2675</td>
</tr>
<tr>
<td><em>C. eleg</em></td>
<td>100 Mbp</td>
<td>C:85% [D:6.9%], F:2.8%, M:11%, n:843</td>
</tr>
<tr>
<td></td>
<td>20447 genes</td>
<td>C:90% [D:11%], F:1.7%, M:7.5%, n:843</td>
</tr>
<tr>
<td><em>H. sapi</em></td>
<td>3381 Mbp</td>
<td>C:89% [D:1.5%], F:6.0%, M:4.5%, n:3023</td>
</tr>
<tr>
<td></td>
<td>20364 genes</td>
<td>C:99% [D:1.7%], F:0.0%, M:0.0%, n:3023</td>
</tr>
<tr>
<td><em>L. giga</em></td>
<td>359 Mbp</td>
<td>C:89% [D:2.3%], F:4.3%, M:5.8%, n:843</td>
</tr>
<tr>
<td></td>
<td>23349 genes</td>
<td>C:90% [D:13%], F:7.8%, M:2.1%, n:843</td>
</tr>
<tr>
<td><em>A. nidu</em></td>
<td>30 Mbp</td>
<td>C:98% [D:1.8%], F:0.9%, M:0.2%, n:1438</td>
</tr>
<tr>
<td></td>
<td>10534 genes</td>
<td>C:95% [D:7.3%], F:3.8%, M:0.9%, n:1438</td>
</tr>
</tbody>
</table>
Run BUSCO on BioHPC Lab

https://biohpc.cornell.edu/lab/userguide.aspx?a=software&i=255#c

cp -r /programs/augustus-3.2.1 ./

# set PATH for required software
export AUGUSTUS_CONFIG_PATH=/workdir/XXX/augustus.2.5.5/config
export PATH=/programs/hmmer/binaries:/programs/emboss/bin:$PATH
export PATH=/workdir/XXX/augustus-3.2.1/bin:$PATH

python3 /programs/BUSCO_v1.2/BUSCO_v1.2.py -o SAMPLE -in assembly.fa -l lineage_db -m genome
Evaluation of Genome assembly 1
Metrics for contig length

**N50 and L50** *

**N50**  50% (base pairs) of the assemblies are contigs above this size.

**L50**  Number of contigs greater than the N50 length.

**NG50 and LG50**

N50 is calculated based on assembly size. NG50 is calculated based on estimated genome size.
Evaluate scaffold quality based on alignment to a closely related genome

Comparing synteny of ten biggest contigs from 10X and PACBio assembly of Cinerea
Evaluate scaffolding quality based on genetic mapping

Use mapped GBS sequence tags to evaluate each contig

Fei Lu, Buckler lab
http://www.nature.com/ncomms/2015/150416/ncomms7914/full/ncomms7914.html
From assembled genome to annotated genome

**Procaryotic genomes**

Gene annotation servers (web based)
1. RAST
2. NCBI

**Eucaryotic genomes**

Gene prediction pipeline: Maker

Function annotation pipeline: Blast2GO