# *De novo* whole genome assembly

**Qi Sun**

**Bioinformatics Facility**
**Cornell University**

# The Concept of Reference Genome

>personA_chr1-paternal

GATGGGATTGGGGTTTTCCCCTCCCATGTGCTCAAGACTGGCGCTAAAAGTTTTGAGCTTCTCAAAAGTC
TAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGGACACTTTGCGTTCGGGCTGGGAGCGTG
CTTTCCACGACGGTGACACGCTTCCCTGGATTGGCAGCCAGACTGCCTTCCGGGTCACTGCCATGGAGGA
GCCGCAGTCAGATCCTAGCGTCGAGCCCCCTCTGAGTCAGGAAACATTTTCAGACCTATGGAAACTACTT
CCTGAAAACAACGTTCTGTCCCCCTTGCCGTCCCAAGCAATGGATGATTTGATGCTGTCCCCGGACGATA
TTGAACAATGGTTCACTGAAGACCCAGGTCCAGATGAAGCTCCCAGAATGCCAGAGGCTGCTCCCCCCGT
GGCCCCTGCACCAGCAGCTCCTACACCGGCGGCCCCTGCACCAGCCCCCTCCTGGCCCCTGTCATCTTCT

>personA_chr1-maternal

GATGGGATTGGGGTTTTCCCCTCCCATGTGCTCAAGACTGGCGCTAAAAGTTTTGAGCTTCTCAAAAGTC
TAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGGACACTTTGCGTTCGGGCTGGGAGCGTG
CTTTCCACGACGGTGACACGCTTCCCTGGATTGGCAGCCAGACTGCCTTCCGGGTCACTGCCATGGAGGA
GCCGCAGTCAGATCCTAGCGTCGAGCCCCCTCTGAGTCAGGAAACATTTTCAGACCTATGGAAACTACTT
CCTGAAAACAACGTTCTGTCCCCCTTGCCGTCCCAAGCAATGGATGATTTGATGCTGTCCCCGGACGATA
TTGAACAATGGTTCACTGAAGACCCAGGTCCAGATGAAGCTCCCAGAATGCCAGAGGCTGCTCCCCCCGT
GGCCCCTGCACCAGCAGCTCCTACACCGGCGGCCCCTGCACCAGCCCCCTCCTGGCCCCTGTCATCTTCT

>personB_chr1-paternal

GATGGGATTGGGGTTTTCCCCTCCCATGTGCTCAAGACTGGCGCTAAAAGTTTTGAGCTTCTCAAAAGTC
TAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGGACACTTTGCGTTCGGGCTGGGAGCGTG
CTTTCCACGACGGTGACACGCTTCCCTGGATTGGCAGCCAGACTGCCTTCCGGGTCACTGCCATGGAGGA
GCCGCAGTCAGATCCTAGCGTCGAGCCCCCTCTGAGTCAGGAAACATTTTCAGACCTATGGAAACTACTT
CCTGAAAACAACGTTCTGTCCCCCTTGCCGTCCCAAGCAATGGATGATTTGATGCTGTCCCCGGACGATA
TTGAACAATGGTTCACTGAAGACCCAGGTCCAGATGAAGCTCCCAGAATGCCAGAGGCTGCTCCCCCCGT
GGCCCCTGCACCAGCAGCTCCTACACCGGCGGCCCCTGCACCAGCCCCCTCCTGGCCCCTGTCATCTTCT

>personB_chr1-maternal

GATGGGATTGGGGTTTTCCCCTCCCATGTGCTCAAGACTGGCGCTAAAAGTTTTGAGCTTCTCAAAAGTC
TAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGGACACTTTGCGTTCGGGCTGGGAGCGTG
CTTTCCACGACGGTGACACGCTTCCCTGGATTGGCAGCCAGACTGCCTTCCGGGTCACTGCCATGGAGGA
GCCGCAGTCAGATCCTAGCGTCGAGCCCCCTCTGAGTCAGGAAACATTTTCAGACCTATGGAAACTACTT
CCTGAAAACAACGTTCTGTCCCCCTTGCCGTCCCAAGCAATGGATGATTTGATGCTGTCCCCGGACGATA
TTGAACAATGGTTCACTGAAGACCCAGGTCCAGATGAAGCTCCCAGAATGCCAGAGGCTGCTCCCCCCGT
GGCCCCTGCACCAGCAGCTCCTACACCGGCGGCCCCTGCACCAGCCCCCTCCTGGCCCCTGTCATCTTCT

Reference

GATGGGATTGGGGTTTGAGCTTCTCAAAAGTC

personA ……T/A……..

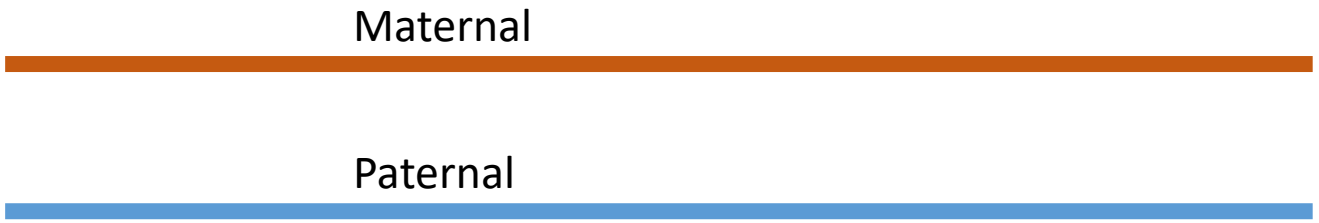
personB ……T/T……..

personC ……T/A……..

personD ……T/A……..

# Reference genome is a mosaic of paternal and maternal genomes from one individual

**Diploid genome**




**Reference**

# What individual to use as the reference?

Use an individual that is imbred.

Cornell professor Dr. Doug Antczak with Twilight, DNA donor for the horse reference genome.

**The human reference is a composite genome from multiple anonymous individuals**

# Sequencing platforms

- **Short reads** (150bp)
  - Illumina

0.1% Error

- **Long reads** (>10kb)
  - PacBio
  - Oxford Nanopore

10% Error

# Steps in genome assembly

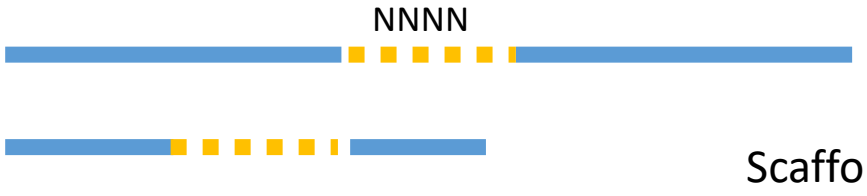
## Contiging

Assemble reads into longer pieces called contigs


Raw sequencing reads

## Polishing & Scaffolding

Error correction and connecting neighboring pieces



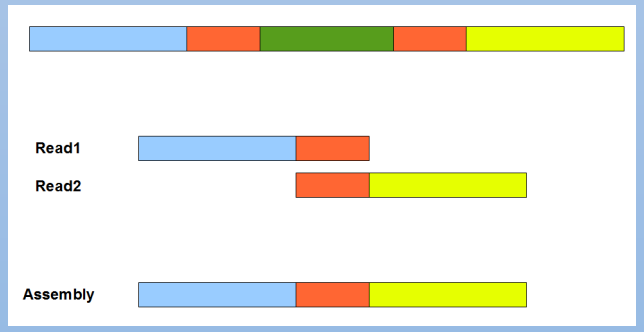

## Pseudo-molecules

Chromosomal level finish

# Two assembly strategies

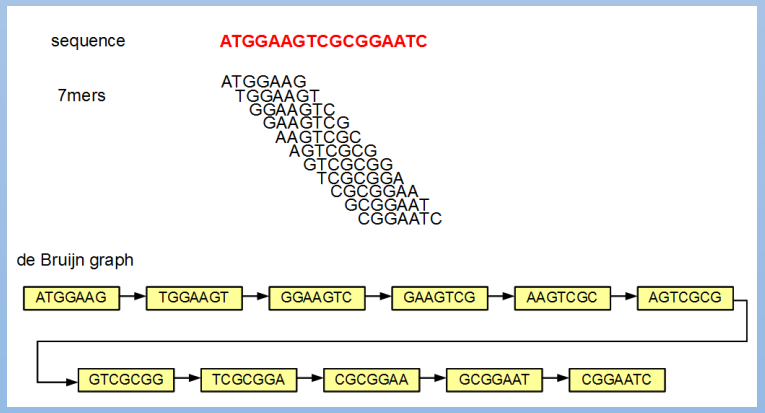
## Long reads

## overlap–layout–consensus




## Short reads

## de-bruijn-graph




source: http://www.homolog.us/Tutorials/index.php

Canu, Falson, Flye, et al.

Spades, Abyss, et al

ATGGAAGTCGCGGAATC

ATGGAAG
TGGAAGT
GGAAGTC
GAAGTCG
AAGTCGC
AGTCGCG
GTCGCGG
TCGCGGA
CGCGGAA
GCGGAAT
CGGAATC

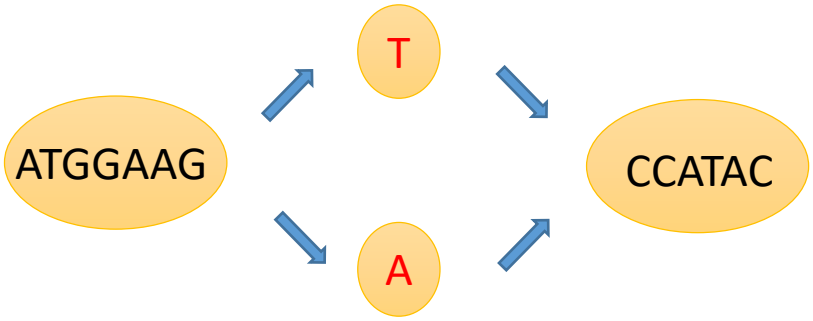
# Branching in de-bruijn-graph

Kmer 1 ATGGAAG

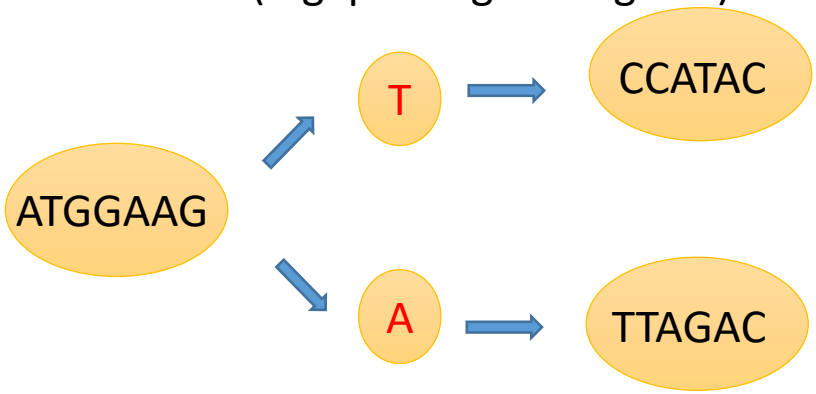
Kmer 2 TGGAAGT

Kmer 3 TGGAAGA

Bubble (e.g. sequencing errors)




Crosslinks (e.g. paralogous regions)



# Short kmers are more likely to branch

**Genome:** **GGATGGAAGTCG**…… **CGATGGAAGGAT**

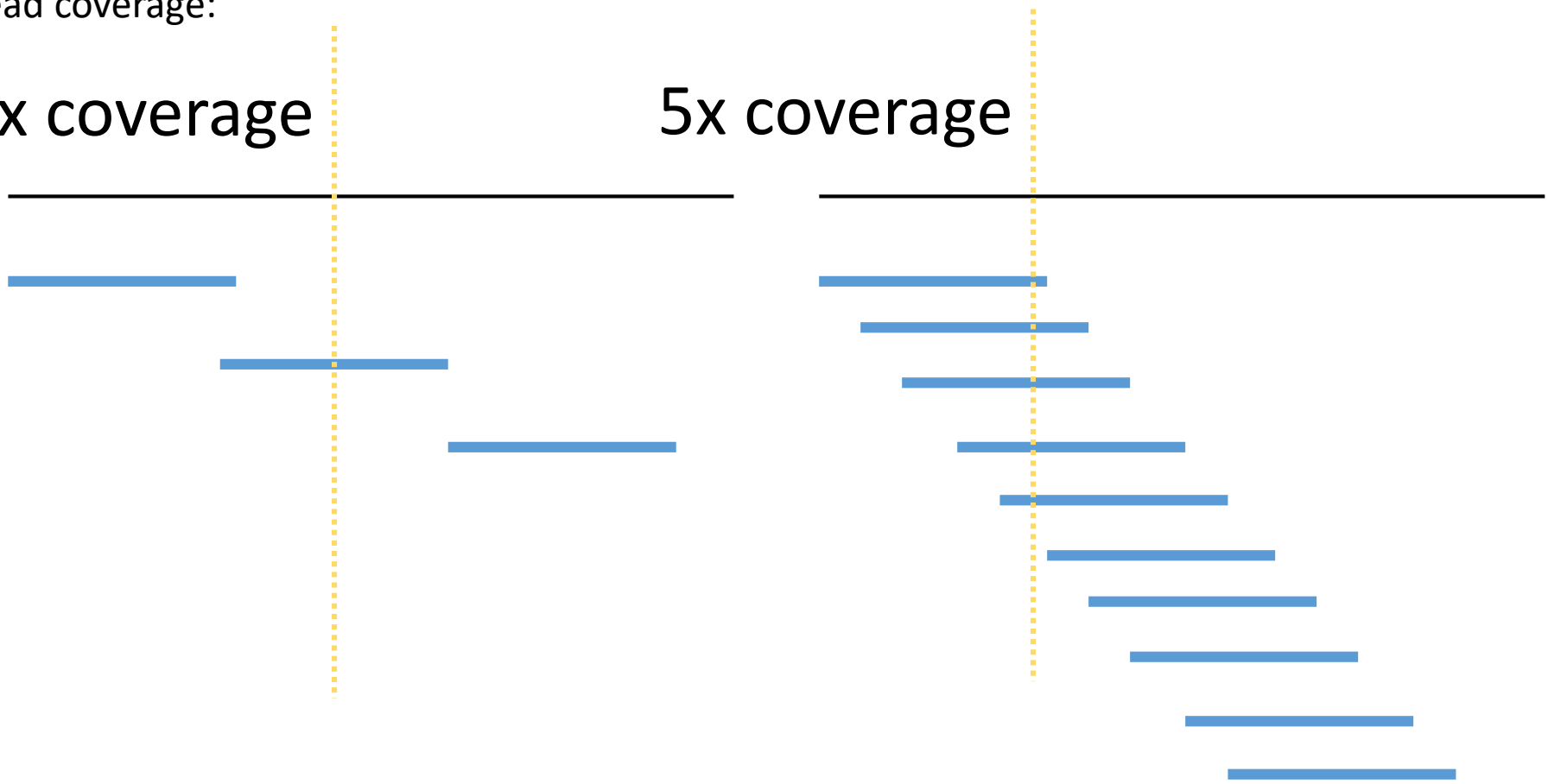
(black regions are identical sequence)

Short kmer


Longer kmer

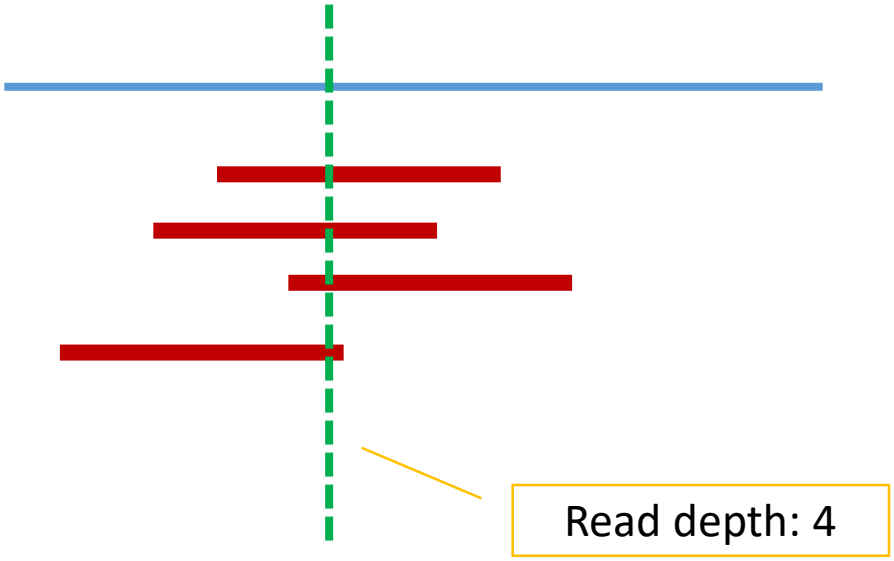
# Longer **kmer** = Lower **kmer** coverage

Read coverage:

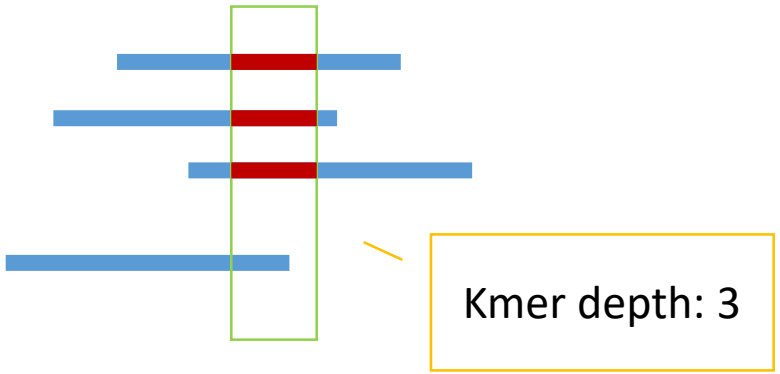


# Read depth vs Kmer depth

**Read depth: number of reads at a genome position**




**Kmer depth: number of occurrence of each kmer**



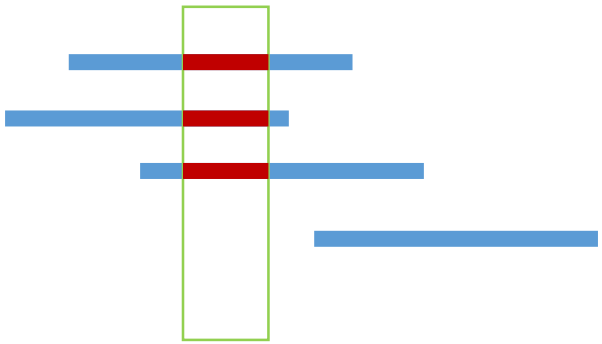

For de-bruijn-graph, it is kmer depth that matters.

**Longer the kmer,**

**Lower the kmer depth**

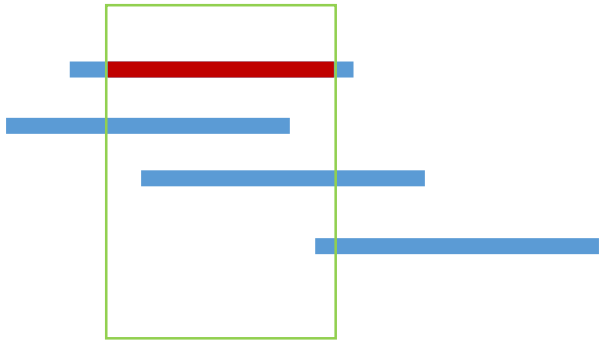
**Read depth: 3**

30mer



Kmer depth =3

80mer

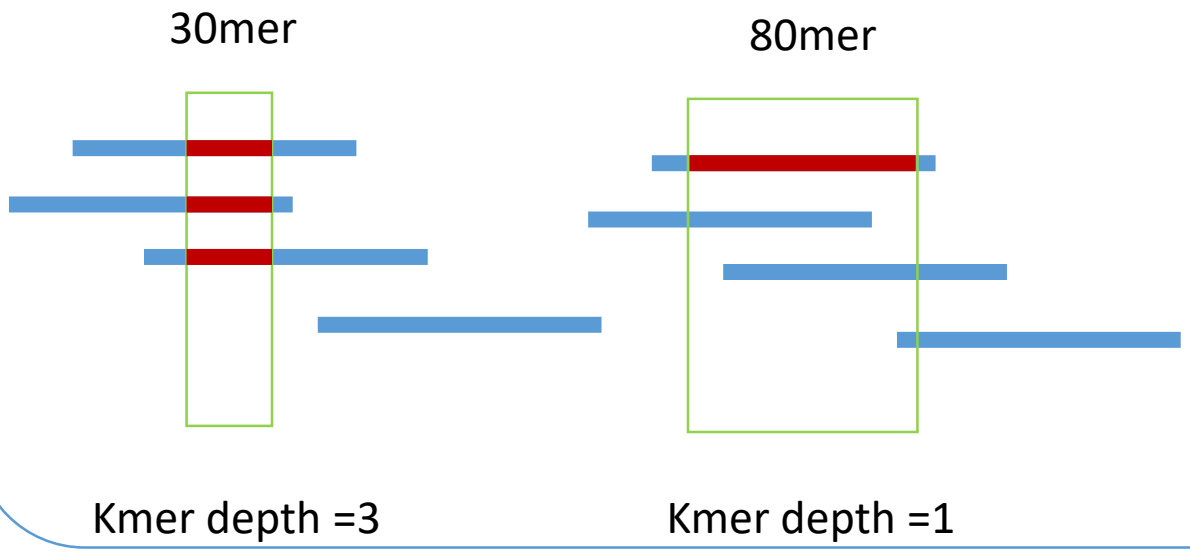


Kmer depth =1

## Longer kmer = less branching


# Optimize kmer length

## Longer kmer = lower kmer depth



# de-bruijn-graph for contiging short reads

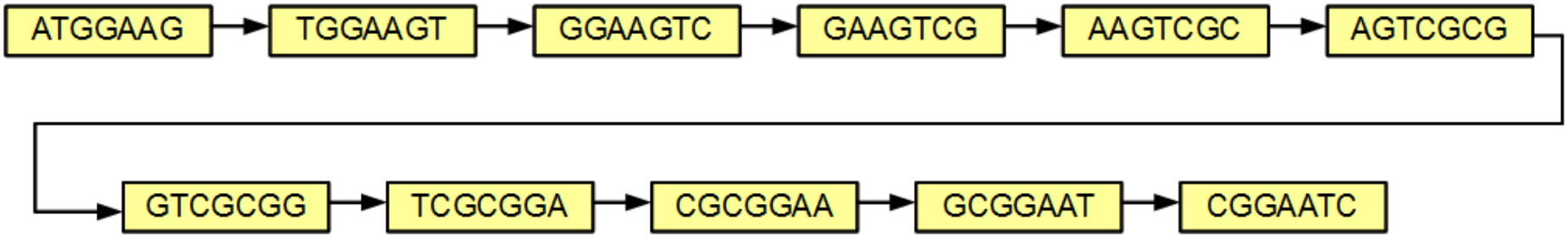
sequence **ATGGAAGTCGCGGAATC**

7mers

ATGGAAG
TGGAAGT
GGAAGTC
GAAGTCG
AAGTCGC
AGTCGCG
GTCGCGG
TCGCGGA
CGCGGAA
GCGGAAT
CGGAATC

Kmers

de Bruijn graph




source: http://www.homolog.us/Tutorials/index.php

## N50 with different kmer (kb)

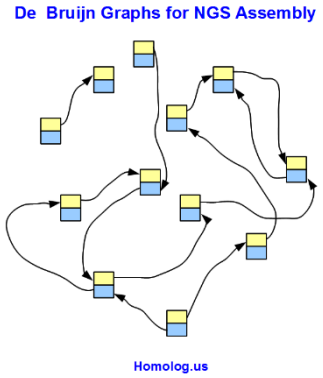
| Kmer size | 75 | 95 | 105 | 115 |
|---|---|---|---|---|
| contig N50 | 268 | 476 | 476 | 268 |
| scaffold N50 | 543 | 543 | 543 | 268 |

Read length: 199 bp Coverage: ~100 x

SPAdes: use a series of kmers.

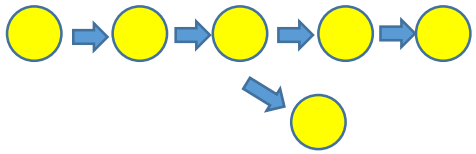
# Branching in the de-bruijn-graph and how to solve

## Tips, bubbles and crosslinks

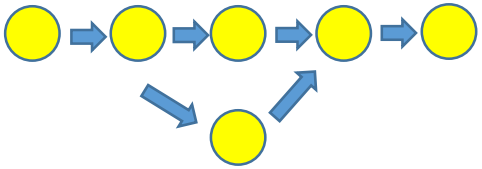



source: http://www.homolog.us/Tutorials/index.php

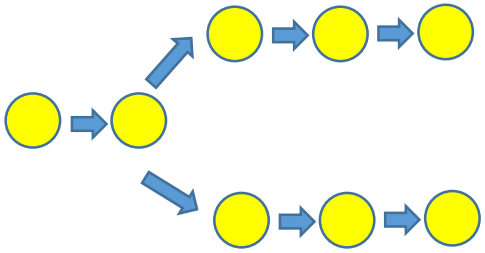
**Tips**



**Bubbles**



**Crosslinks**



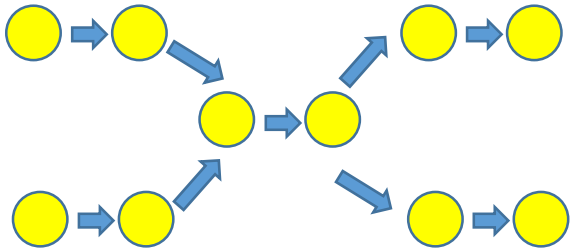
**Deal with sequencing errors and repetitive regions**

# 1. Sequencing errors

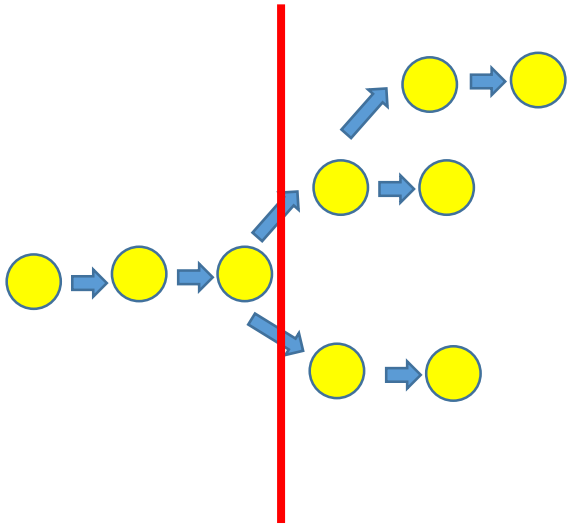
- Remove low depth kmers in a bubble;
- Too long kmers would cause coverage problem;

# 2. Repeatitive regions

- Longer kmers

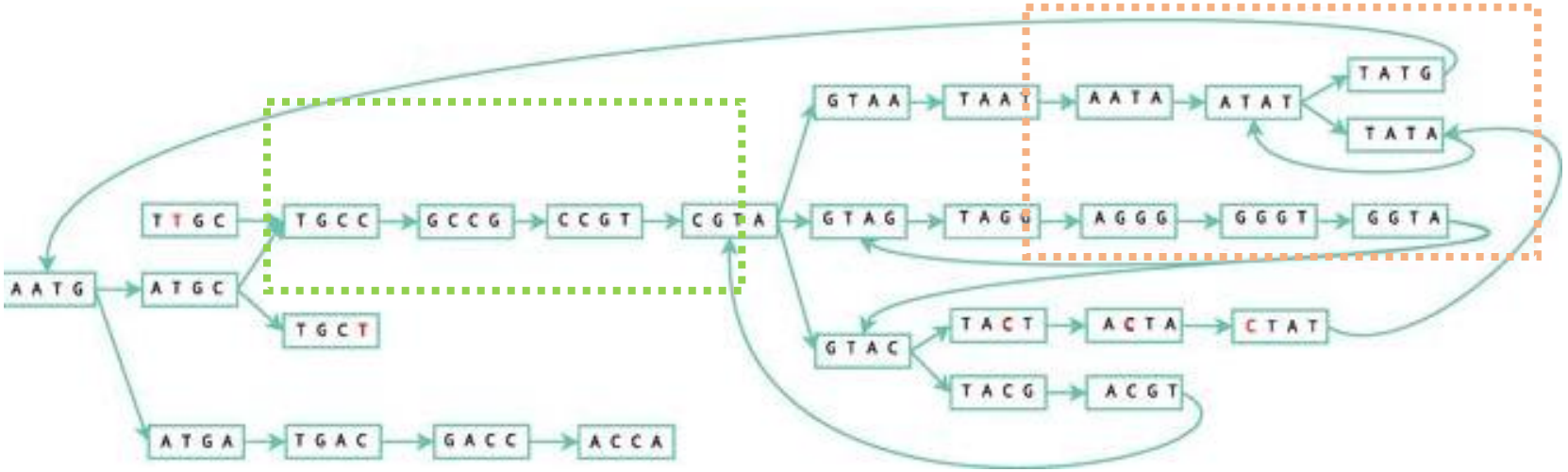


Tiny repeat:
Separate the path



Break boundary between low and high copy regions

**Genome assembly software gives us a graph, then algorithmically identify a path in the graph**

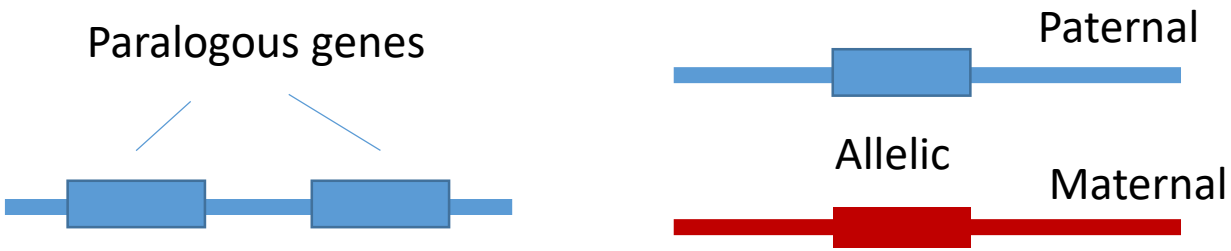



https://en.wikipedia.org/wiki/Velvet_assembler

Common errors:

- Collapsed paralogous genes;
- Chimeric contigs/scaffolds;

# Assembly software is tuned to collapse allelic genes, but not paralogous genes;



# Kmer distribution can be used to estimate genome size

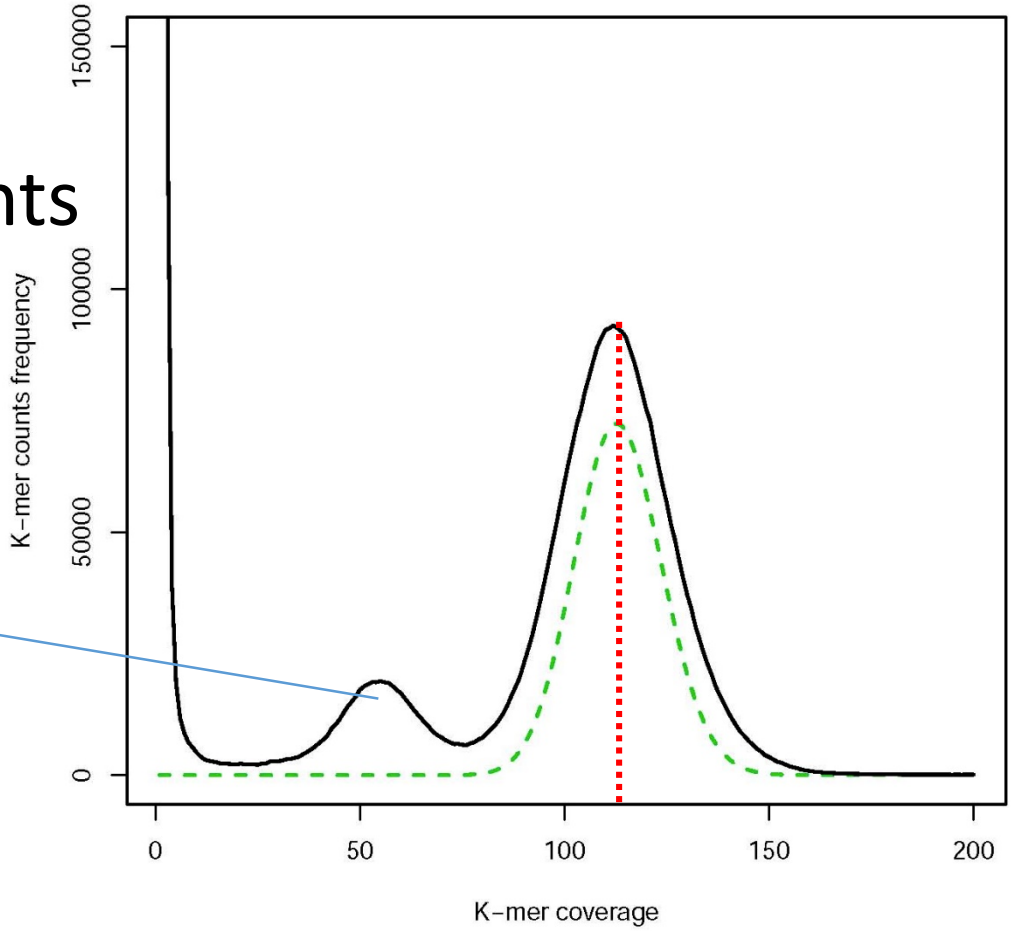
Sequencing data:  20 GB
Coverage: 10x
Genome size = 2GB

$$\text{Genome size} = \frac{\text{Total base pairs}}{\text{Coverage}}$$

Calculated by modeling kmer distruction

# Kmer coverage distribution

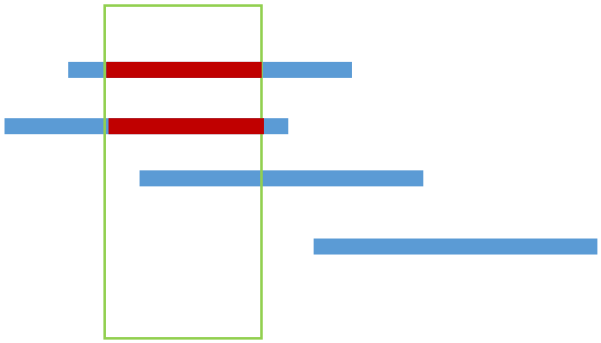
Frq of counts




Kmers from heterozygous regions

Kmers counts

# Estimate genome size based on kmer distribution

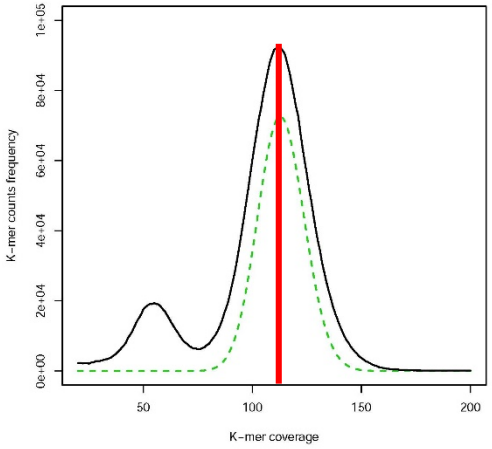


75 mer
Read depth = 3
Kmer depth = 2

Step 1: convert read depth to kmer depth

$$N = M^* L/(L-K+1)$$

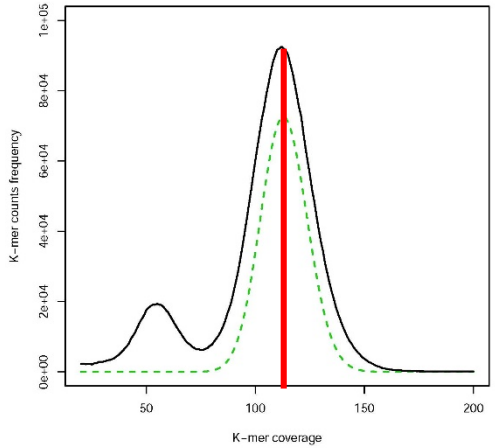
M: kmer depth = 112
L: read length = 101 bp
K: Kmer size =21 bp
N: read depth =140




Step 2: genome size is total sequenced base pairs devided by read depth

$$\text{Genome size} = T/N$$

T: total base pairs = 0.505 gb
N: read depth = 140
Genome size: 3.6 mb




**Software to estimate genome size:**

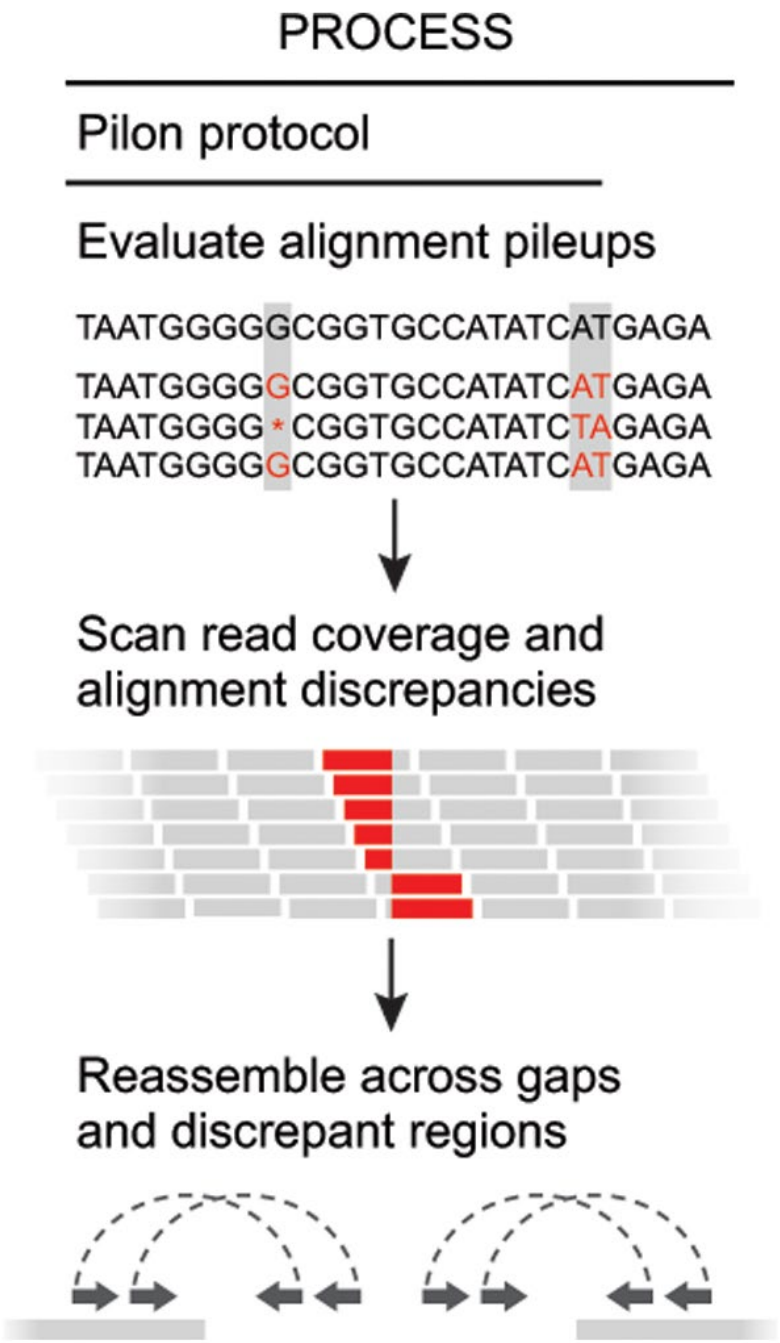
ErrorCorrectReads.pl (from ALLPATHS-LG)

```
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 &
```

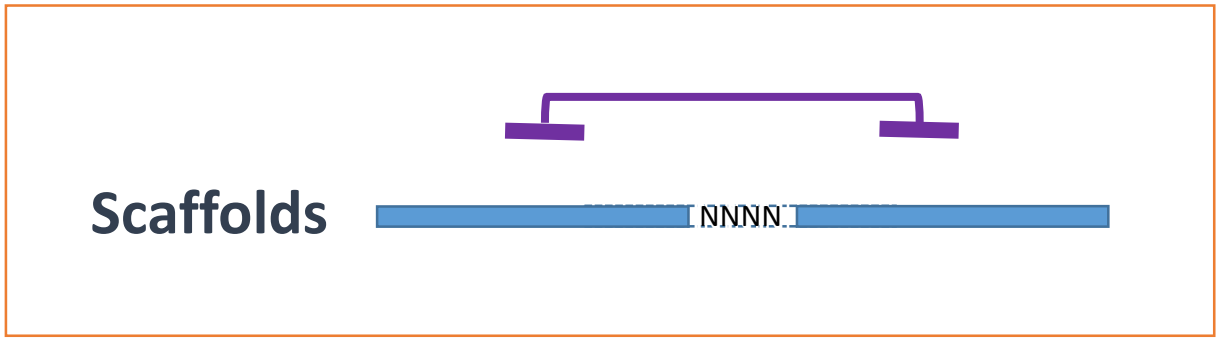
# Polishing with pilon

Align raw reads back to the assembly and identify discrepancies

SPAdes has a “--careful” option that does error correction with read alignment

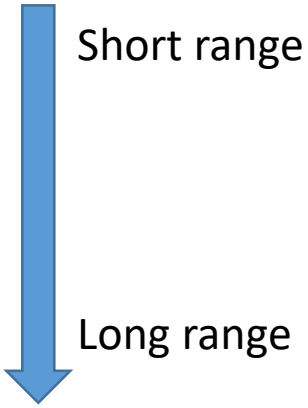


# Scaffolding contigs




Technologies

- **Long-read:** PacBio or Nanopore
- **BioNano:** Optical Mapping:
- **Hi-C:** Dovetail; Phase Genomics

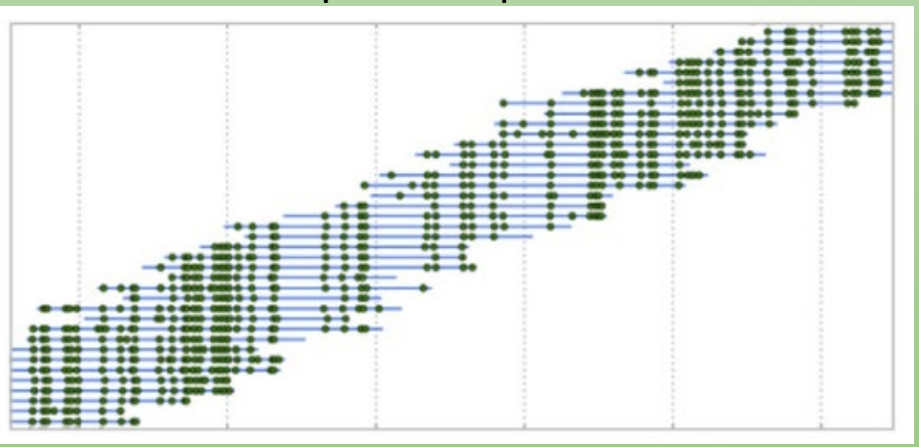


# Scaffolding strategies: Physical maps

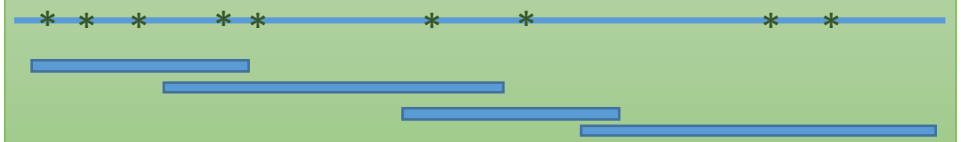
BioNano optical map

- Label 7-mer nickase recognition sites;
- Measure fragment length

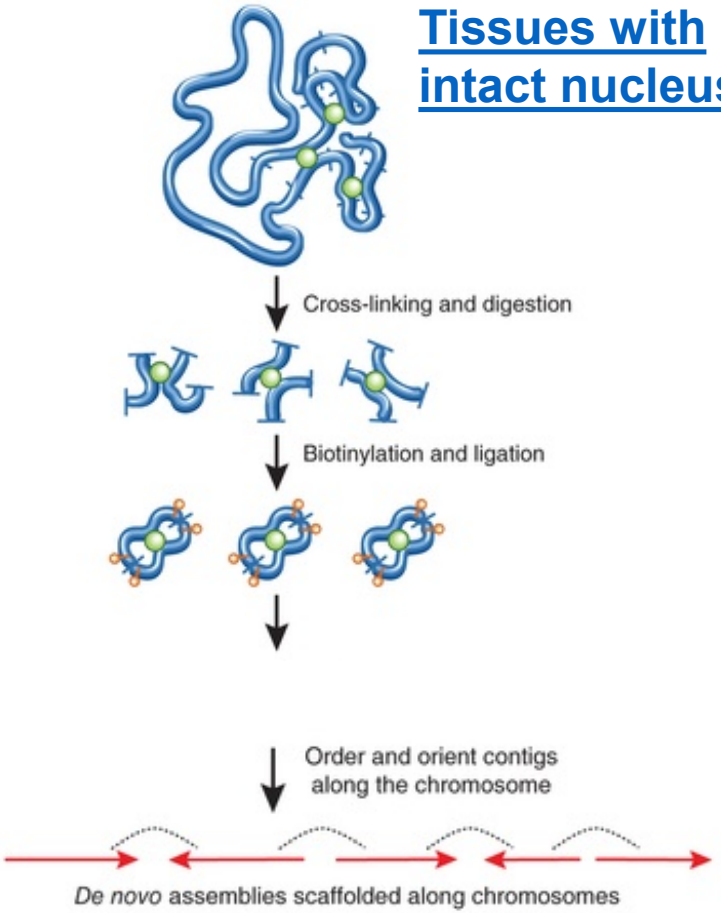
Assemble the optical map



Place contig to the optical map



Hi-C:



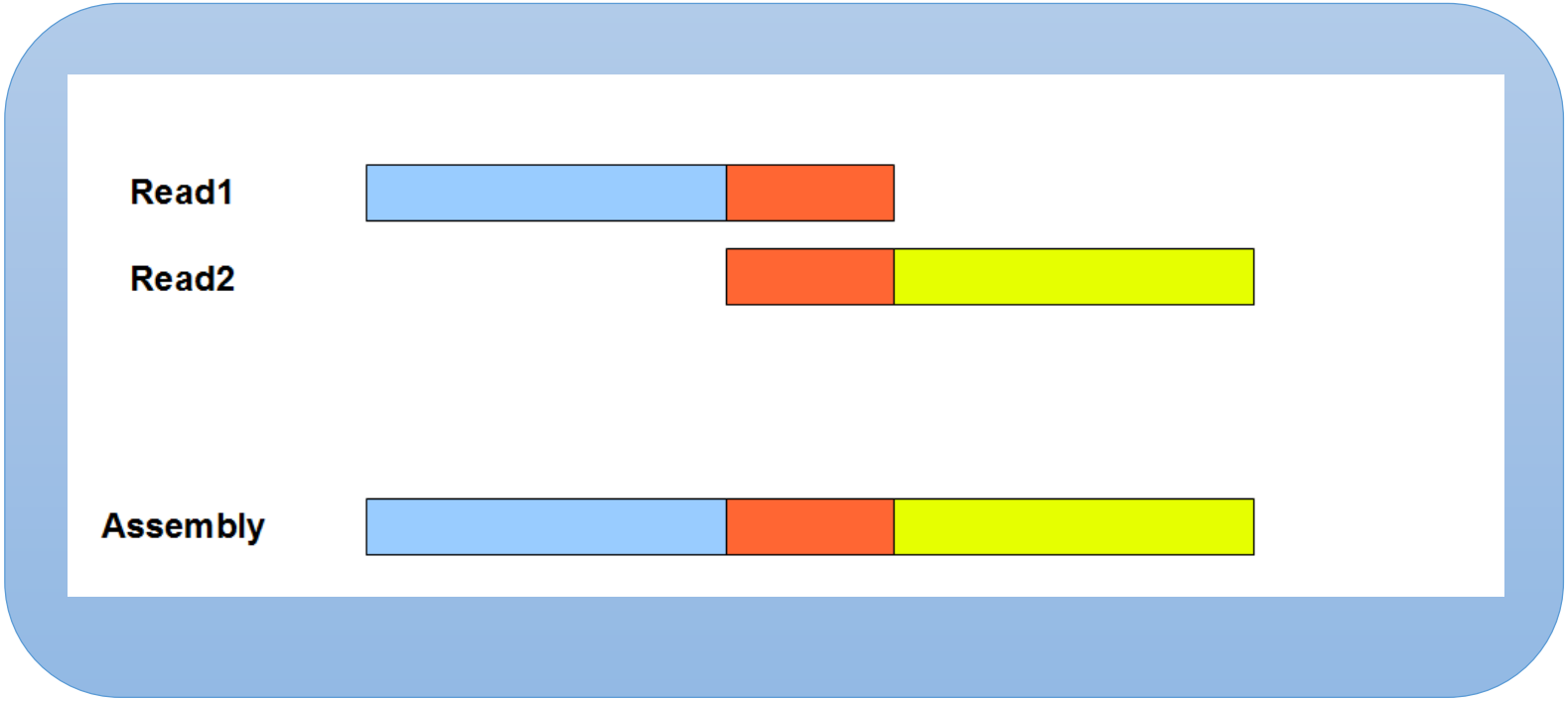

Dovetail & Phased Genomics

**Assembly pipeline:**

1. Trim adapters: Trimmomatic
2. Contiging: SPAdes
3. Polishing: (included in SPAdes with –careful option)
4. Scaffolding: PBJelly. (If you have both long and short reads, it is better to run hybrid assembly tool, e.g MaSuRCA or SPAdes)
5. Assessment: QUAST, BUSCO

# Long reads

# overlap–layout–consensus



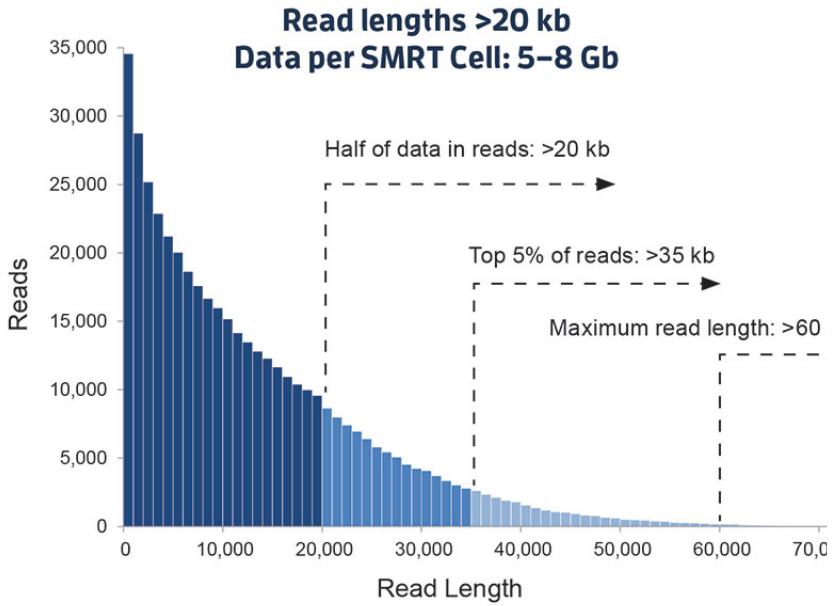

- PacBio
- Oxford Nanopore

# Long-read Sequencing Platform: PacBio SMRT

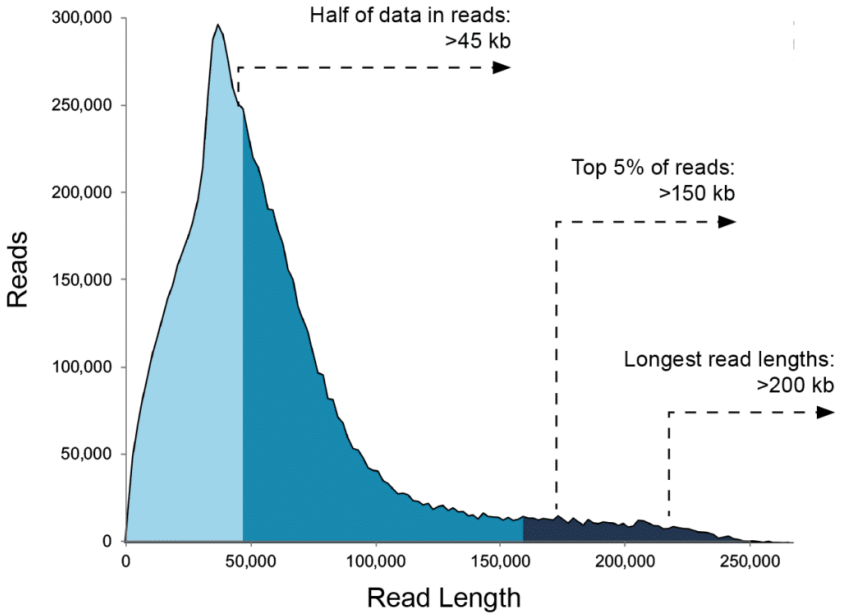
Fragment: >>10kb

Read: 10-20 kb

2017




2018




From PacBio web site

# DNA fragment length

vs

# Sequencing read length

## Assembly pipeline:

1. **Run basecaller**
2. Assembly with Canu
3. Polishing
4. Assessment
5. Scaffolding

Raw signal data => FASTQ

- Run Nanopore basecaller “guppy” on a computer with good GPU.

https://biohpc.cornell.edu/lab/userguide.aspx?a=software&i=653#c

## Assembly pipeline:

1. **Run basecaller**
2. **Assembly with Canu**
3. Polishing
4. Assessment
5. Scaffolding

1. Correct sequencing errors
   - All-versus-all alignment
   - Correct errors through overlaps

1. Trim reads
   - Trim regions of reads not supported by other reads

2. Assemble
   - Contiging corrected reads

## Assembly pipeline:

1. **Run basecaller**
2. **Assembly with Canu**
3. Polishing
4. Assessment
5. Scaffolding

**Default setting:**

**rawErrorRate (raw reads)**

PacBio: 0.300

Nanopore: 0.500

**correctedErrorRate (corrected reads)**

PacBio: 0.045

Nanopore: 0.144

(decrease with higher coverage)

# Assembly pipeline:

1. **Run basecaller**
2. **Assembly with Canu**
3. **Polishing**
4. Assessment
5. Scaffolding

**Polishing tools**

Alignment: minimap2 for long reads; bwa for short reads.

Arrow: polish with pacbio reads

Nanopolish: polish with nanopore reads

Pilon: polish with Illumina reads (optional for PacBio assembly, needed for Nanopore assembly.

# Assembly pipeline:

1. **Run basecaller**
2. **Assembly with Canu**
3. **Polishing**
4. Assessment
5. Scaffolding

**Polishing with Pilon**
**(Illumina, Nanopore or PacBio)**

--frags illumina.bam

--nanopore np.bam

--pacbio pb.bam

## Assembly pipeline:

1. **Run basecaller**
2. **Assembly with Canu**
3. **Polishing**
4. **Assessment**
5. Scaffolding

BUSCO: completeness

QUAST: length

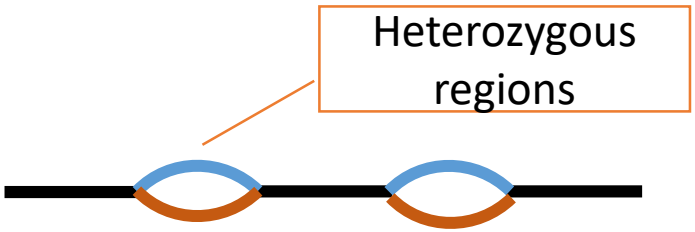
**Assembly pipeline:**

1. **Run basecaller**
2. **Assembly with Canu**
3. **Polishing with Pilon**
4. **Assessment**
5. **Scaffolding**

BioNano: optical map

Hi-C:  physical map from chromatin structure

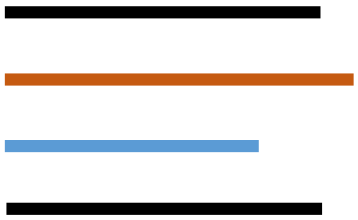
# A diploid genome


# Assembly results

**Random phased assembly**

**Haplotigs**



# Why hybrid assembly (PacBio + Illumina)

If you have lots of money,

50 -100x PacBio

If you are in the middle,

- 50 -100x Illumina
- 10x PacBio

If you have very little money,

50 -100x Illumina

# Why hybrid assembly (PacBio + Illumina)

Medium/large genomes

50 -100x PacBio

Extra large genomes

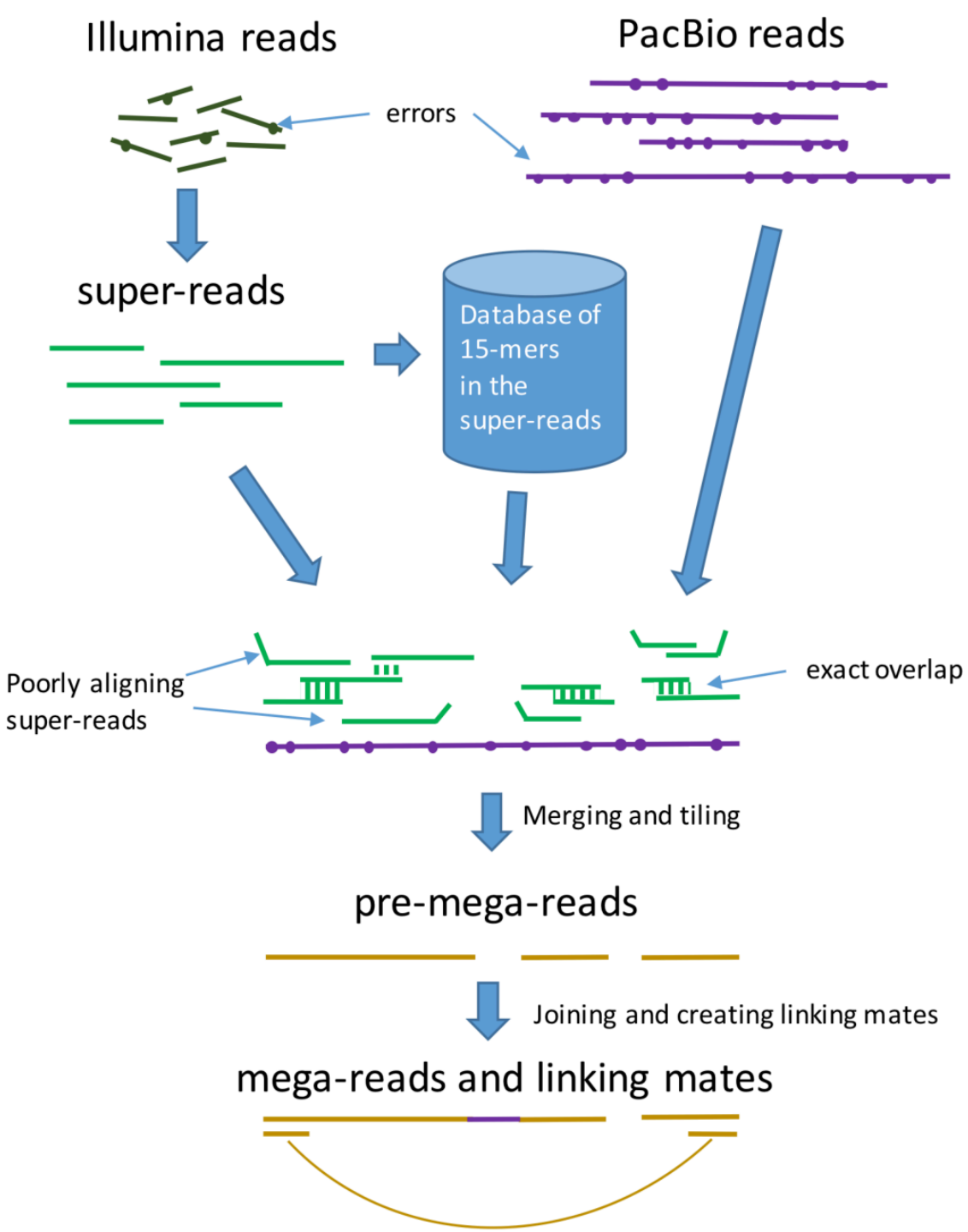
- 50 -100x Illumina
- 10x PacBio

Bacterial/fungal genomes

50 -100x Illumina

# MaCuRCA

Zimin, Aleksey V. et al. (2017) *Genome research* 27 5: 787-792 .



# Assessment of assemblies

- 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 Lineages

| | | | |
|---|---|---|---|
| Arthropods | Vertebrates | Fungi | Bacteria |
| Metazoans | Ekaryotes | Plants | |

# BUSCO gene sets:

Present in at least 90% of the species in each lineage, single copy.

# Evaluation of assembly quality

https://github.com/trinityrnaseq/trinityrnaseq/wiki/Transcriptome-Assembly-Quality-Assessment

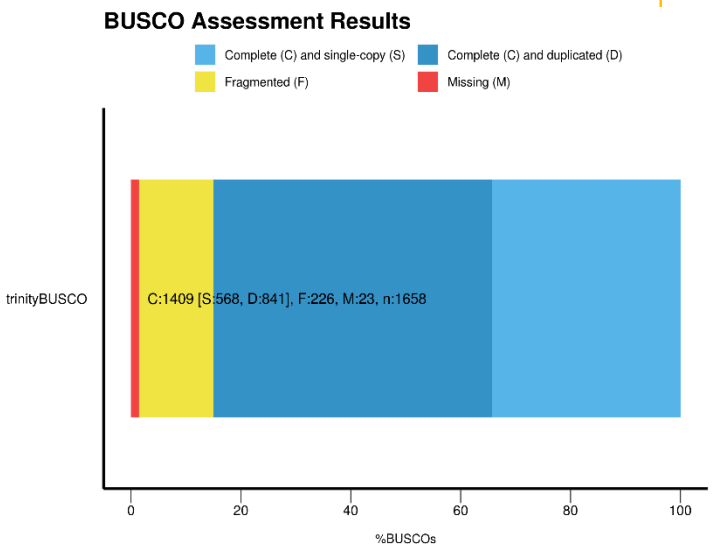
## BUSCO Score

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

C: 85% [S:34%,D:51%],
F: 14%,
M: 1%,
n: 1658

C: Complete
- S: single copy;
- D: duplicated;

F: Fragmented;
M: Missing;
n: Total groups;



# 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.

# Usage statistics of assembly software on BioHPC

| | **Usage** | **Short reads** | **Long reads** | **hybrid** |
|---|---|---|---|---|
| SPAdes | 60.31% | x | | x |
| canu | 17.93% | | x | |
| MaSuRCA | 5.47% | x | | x |
| supernova | 0.50% | x | | |
| Unicycler | 0.46% | x | x | x |
| Flye | 0.26% | | x | |
| abyss | 0.17% | x | | x |
| mccortex | 0.06% | x | | |
| velvet | 0.06% | x | | |
| SOAPdenovo2 | 0.01% | x | | |

# From assembled genome to annotated genome