Genome Assembly

CHRIS FIELDS

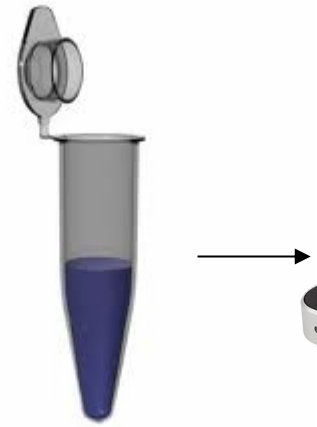
MAYO-ILLINOIS COMPUTATIONAL GENOMICS WORKSHOP
JUNE 8, 2020

Outline

- *Introduction*
- *Sequencing technology overview*
- *Steps in a genome assembly*
- *Planning an assembly project*
- *Assembly assessment*
- *Annotation*

Ideal world

I would not need to give this talk!



Human DNA

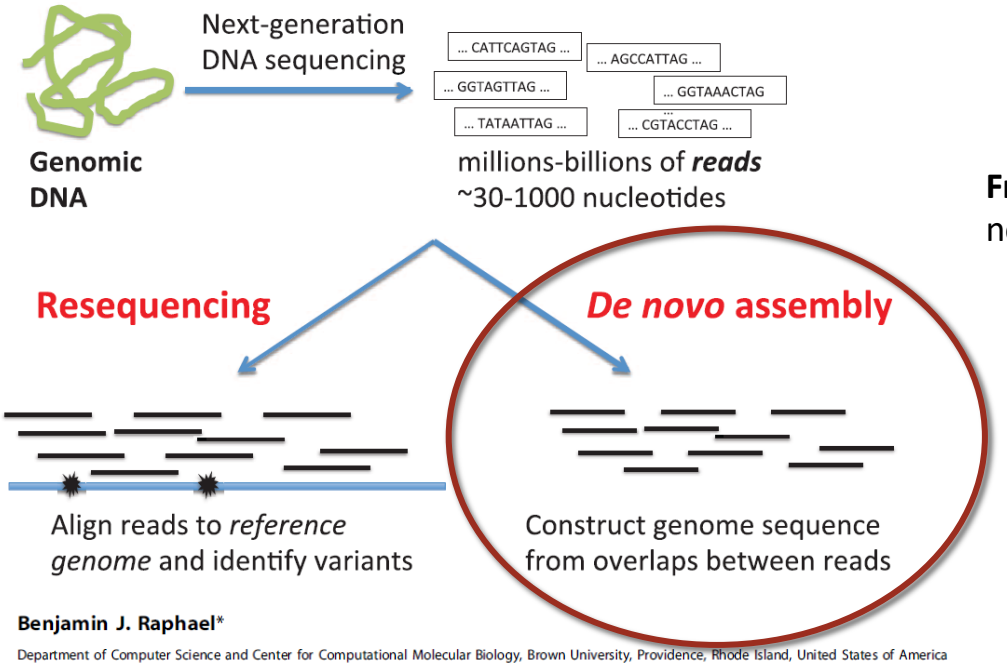
iSequencer™

```
AGTCTAGGATTCGCT
ACAGATTCAGGCTCT
GAAGCTAGATCGCTA
TGCTATGATCTAGAT
CTCGAGATTCGTATA
AGTCTAGGATTCGCT
ATAGATTCAGGCTCT
GATATAT
```

46 complete haplotype chromosome sequences

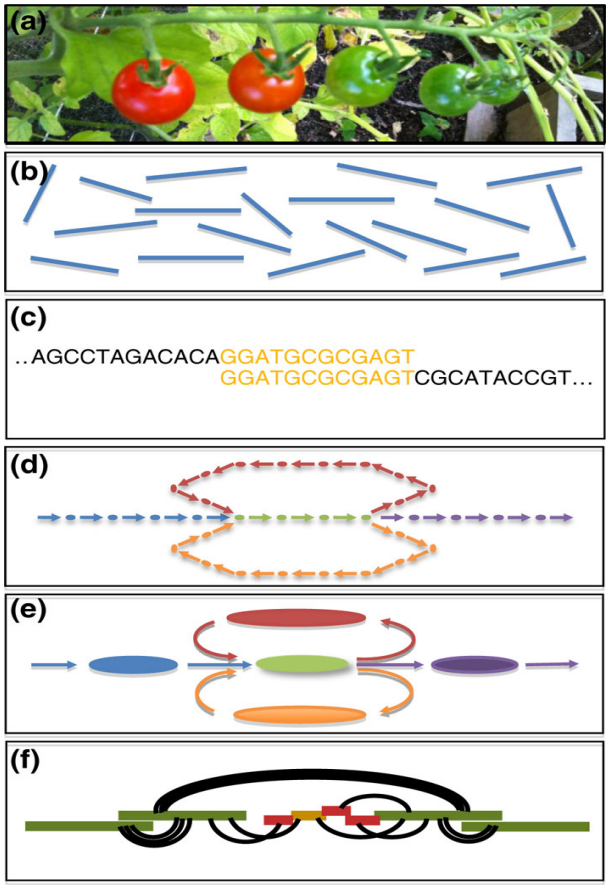
T. Seemann

What is a Genome Assembly?



From scratch:
no prior knowledge

Schematic overview of genome assembly.



(a) DNA is collected from the biological sample, fragmented, and sequenced.

(b) The output from the sequencer consists of many billions of short, unordered DNA fragments from random positions in the genome.

(c) The short fragments are compared with each other in some way to discover how they overlap.

(d) The overlap relationships are captured in a large assembly graph

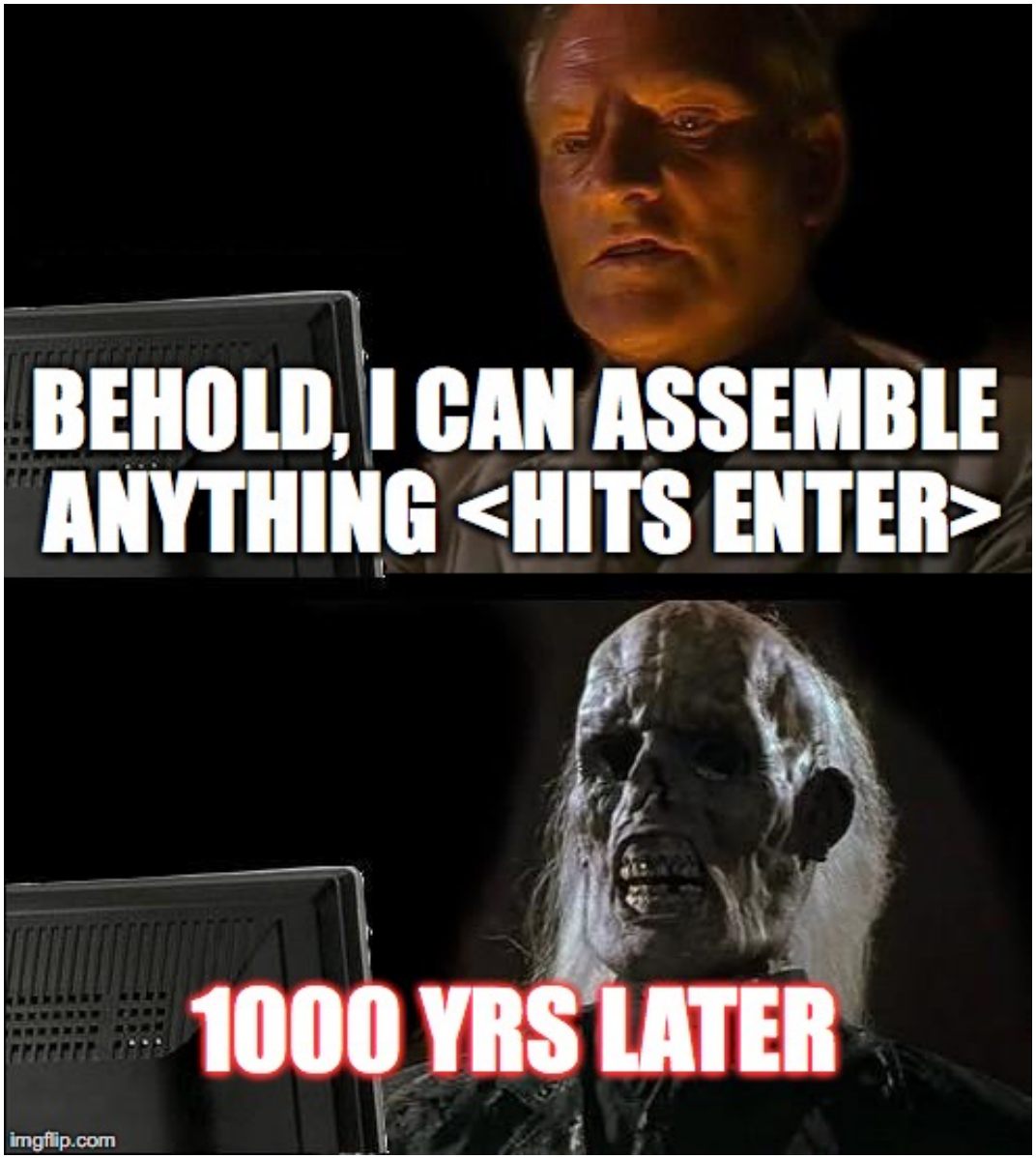
(e) The graph is refined to correct errors and simplify

(f) Finally, additional information such as mates, markers and other long-range information can be used to order and orient the initial assembly (**contigs**) into large **scaffolds**

Schatz et al. Genome Biology 2012 13:243

This Looks Easy, Let's Do a Genome Assembly!

- It must be easy to get an assembly, right?
- How do you find overlaps between sequences (when you have millions to billions of them)?
  - ***You compare them all (overlapping pieces)***
  - ***You find shorter perfectly overlapping segments using some tricks***
    - ***Faster but has a lot of assumptions!!!***
- How do you store all this information?
- How long does it take?


BEHOLD, I CAN ASSEMBLE ANYTHING <HITS ENTER>
1000 YRS LATER
imgflip.com

Results

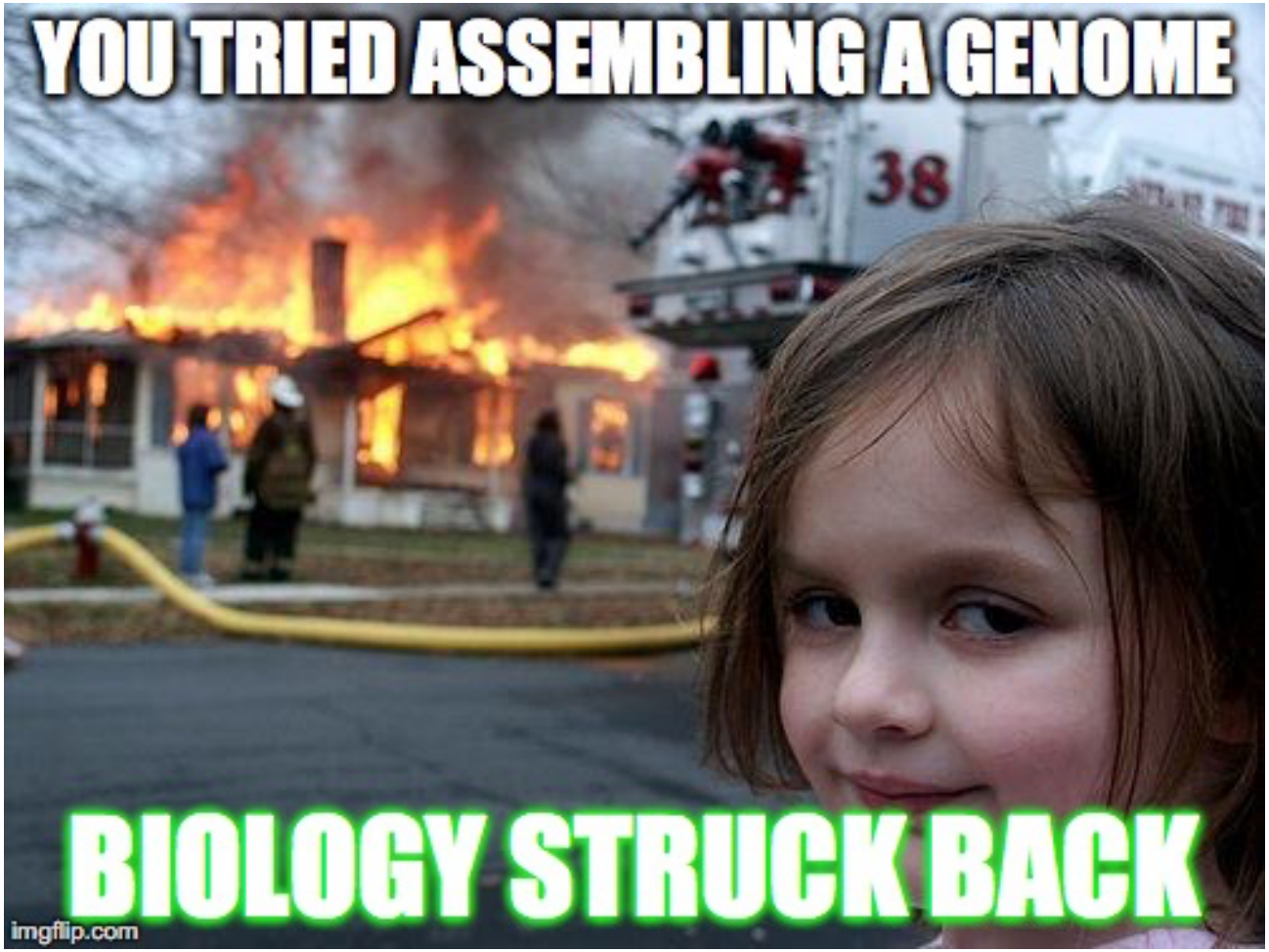
Now you power up and buy a monster multi-core high-memory server

You assembled your genome with your favorite genome assembly tool that your buddy down the hall recommends

You waited a week to a month and you now have results!

Wait, why do I have a million scaffolds? And why is my server on fire?!?

Biology


YOU TRIED ASSEMBLING A GENOME
38
BIOLOGY STRUCK BACK
imgflip.com

Current Sequencing Technologies

Illumina

Millions to billions of short but highly accurate reads

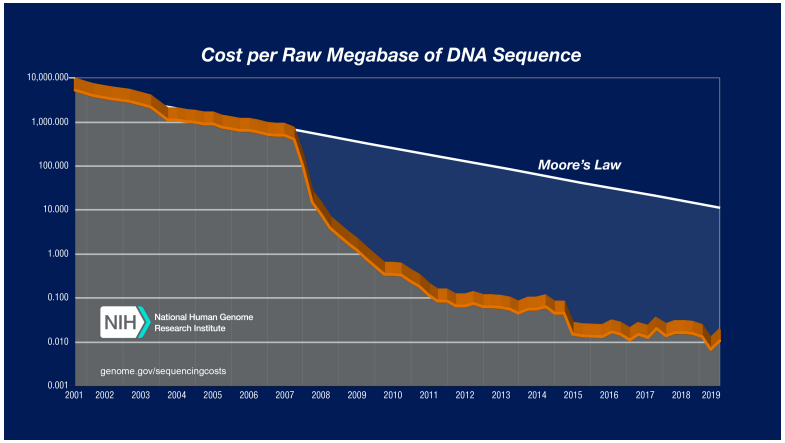
Can be paired-end (sequence ends of fragments)

Advantages

- *Highly accurate* (~99.9%)
- Relatively even coverage of the genome
- Well-vetted technology
- Most cost-effective, as low as $10 per **billion** bases
- (Geneally) robust to sample issues

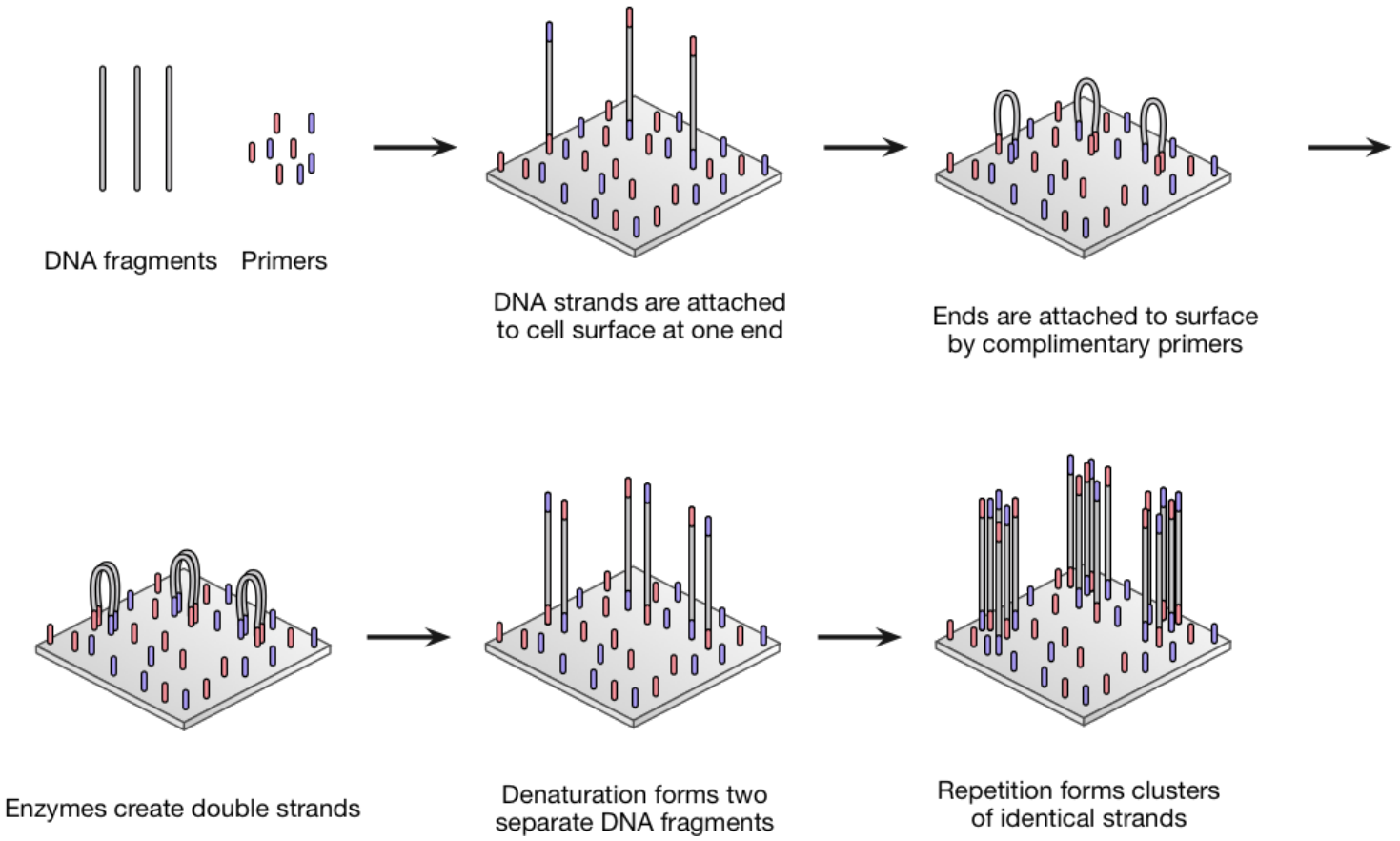
Disadvantages

- Requires high depth for many applications (**30-50x**)
- Sequence length (100-150nt reads) are problematic
- Maximum fragment length (<800bp) is an issue



https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data

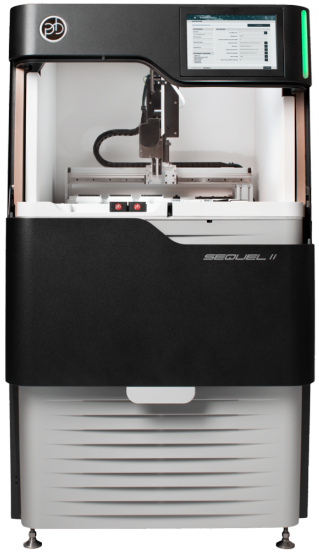
Illumina



https://www.atdbio.com/content/58/Next-generation-sequencing

‘Long reads’

Pacific Biosciences (PacBio)



Oxford Nanopore (ONT)

MinION

‘Long Reads’

Advantages

- Reads can be very long (1kb – 100kb)
- Relatively even coverage of the genome
- (Oxford) real-time sequencing
- (Oxford) portable

Disadvantages

- Expensive compared to Illumina
  - though Sequel II is pretty close!
- Error-prone, and Oxford can have *systematic errors*
  - *But see HiFi reads*
- Need high quality DNA samples

Basic Steps for Genome Assembly

Steps

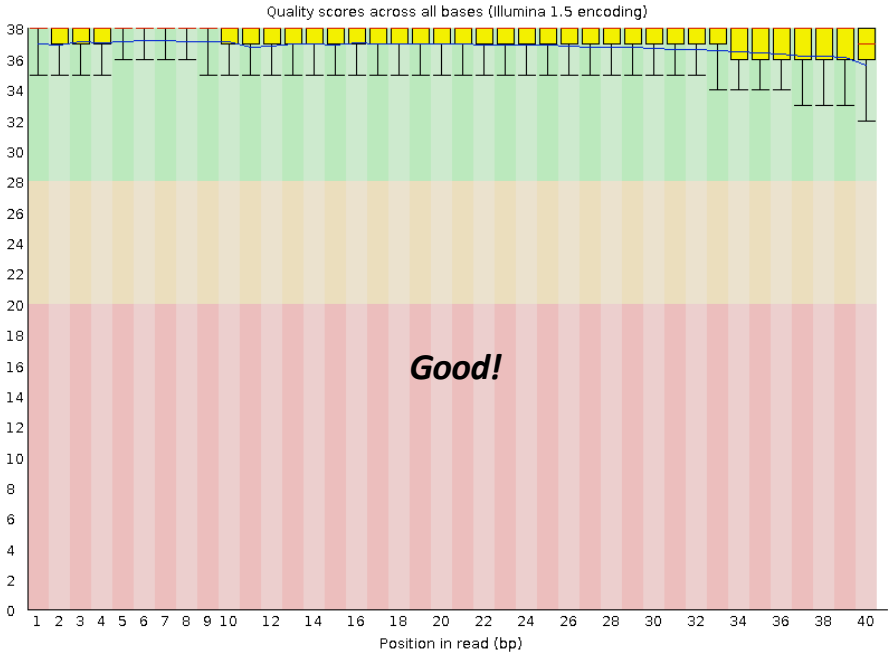
1. Basic DNA sequence cleanup and evaluation
2. Contig building
3. Scaffolding
4. Post-assembly processing and analyses

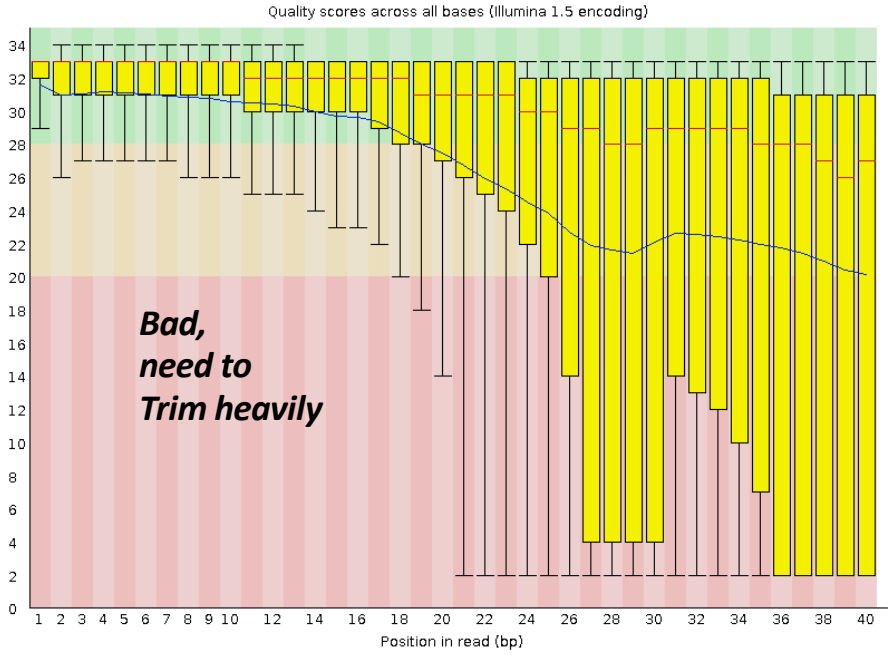
Basic cleanup and evaluation

- Is the DNA sequence high quality?
- Does it need to be trimmed?
- Evaluate libraries for read ‘coverage’
- Any additional sequence preparation steps

DNA Quality (FASTQC)

Illumina Data





https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

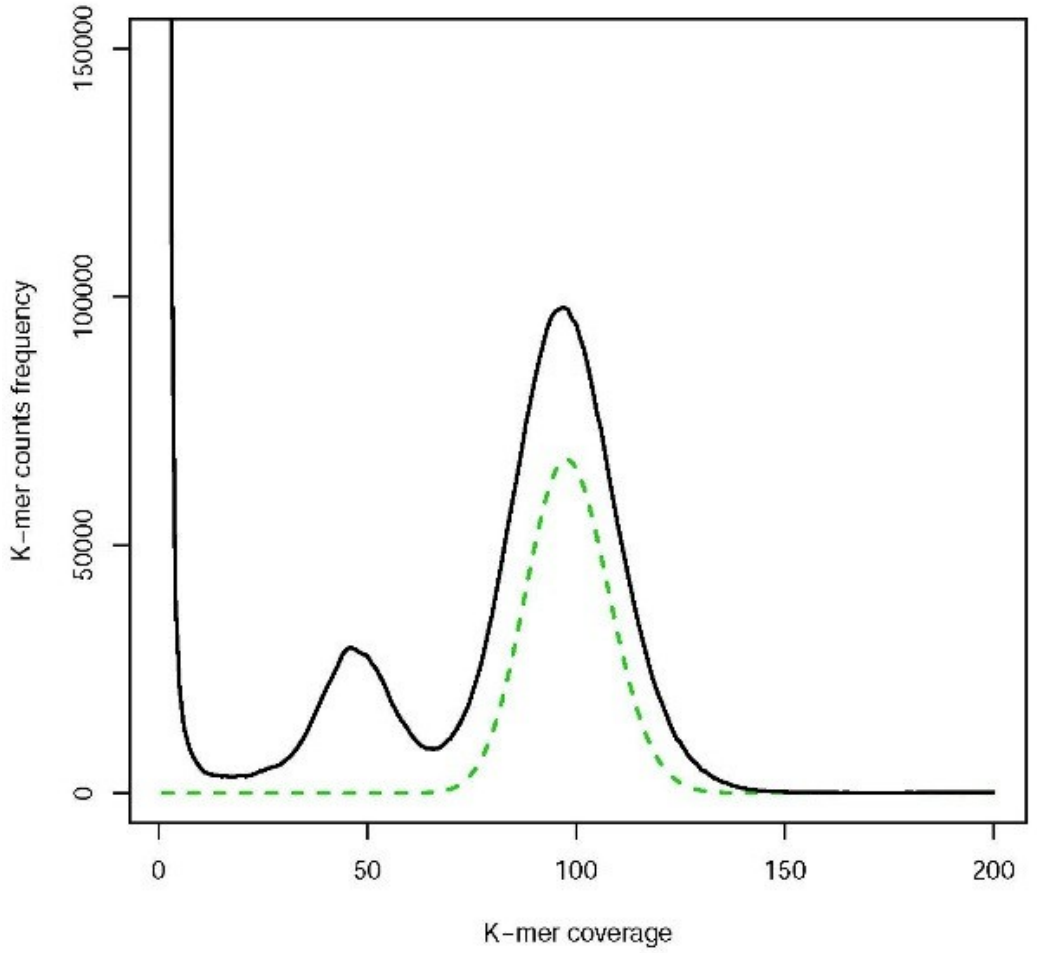
Adapters

Overrepresented sequences

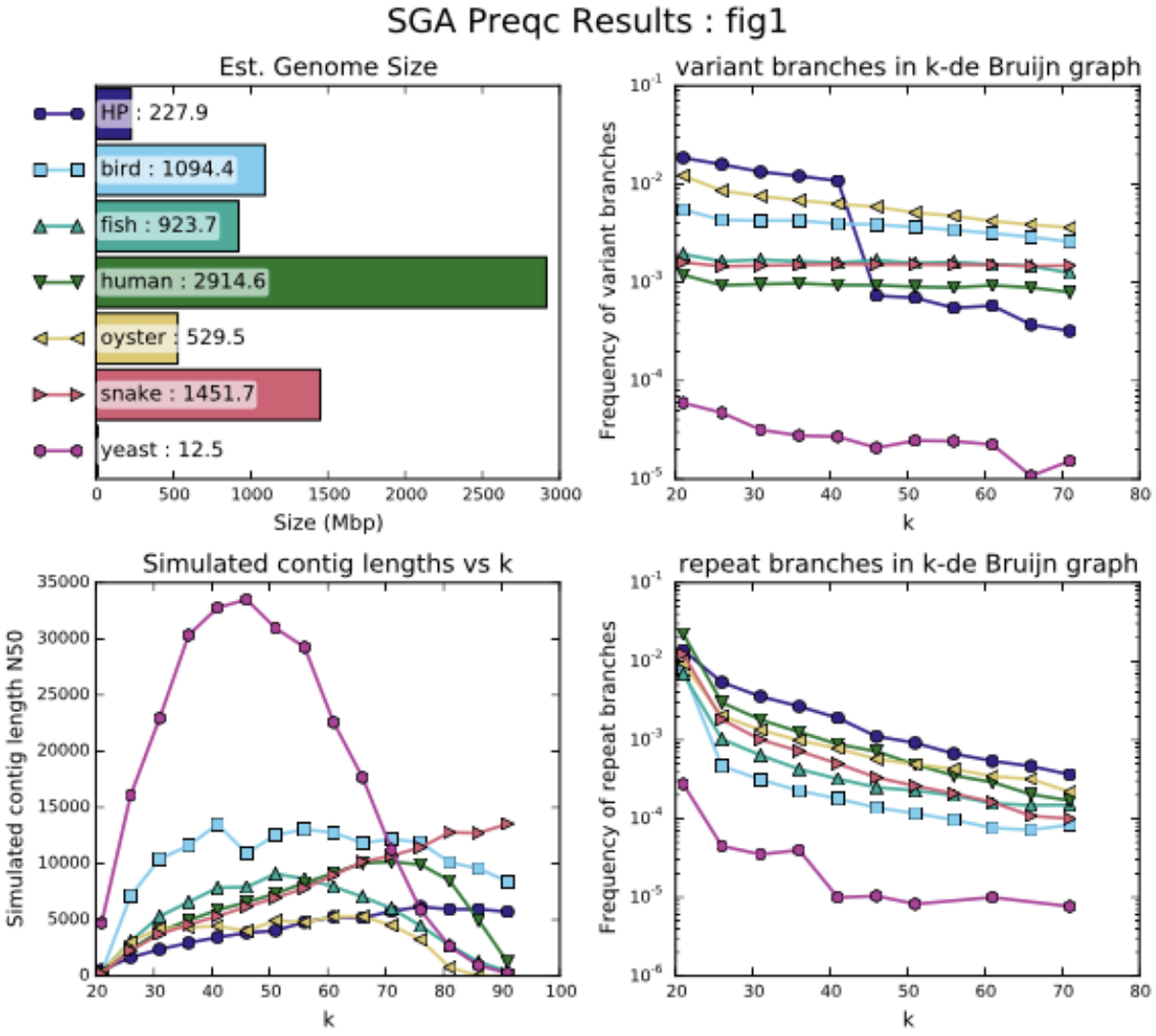
Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGATCT	8122	8.122	Illumina Paired End PCR Primer 2 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGATCGGAAG	5086	5.086	Illumina Paired End PCR Primer 2 (97% over 36bp)
AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTAC	1085	1.085	Illumina Single End PCR Primer 1 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGAAG	508	0.508	Illumina Paired End PCR Primer 2 (97% over 36bp)
AATTATACGGCGACCACCGAGATCTACACTCTTTCCCTAC	242	0.242	Illumina Single End PCR Primer 1 (97% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAAGATCGGAA	235	0.23500000000000001	Illumina Paired End Adapter 2 (96% over 31bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCGAGATCGGAAGA	228	0.22799999999999998	Illumina Paired End Adapter 2 (96% over 28bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGACG	205	0.20500000000000002	Illumina Paired End PCR Primer 2 (97% over 36bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGGATCGGAA	183	0.183	Illumina Paired End Adapter 2 (100% over 32bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGGTCGGAAG	183	0.183	Illumina Paired End Adapter 2 (100% over 32bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGAACT	164	0.164	Illumina Paired End PCR Primer 2 (97% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGTCT	129	0.129	Illumina Paired End PCR Primer 2 (97% over 40bp)
AATTATACTTCTACCACCTATATCTACACTCTTTCCCTAC	123	0.123	No Hit
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGACT	122	0.122	Illumina Paired End PCR Primer 2 (97% over 36bp)
CGGTTCAGCAGGAATGCCGAGATCGGAAGAGCGGTTCAGC	113	0.11299999999999999	Illumina Paired End PCR Primer 2 (96% over 25bp)

Coverage

Qi Sun, Minghui Wang
Cornell



PreQC

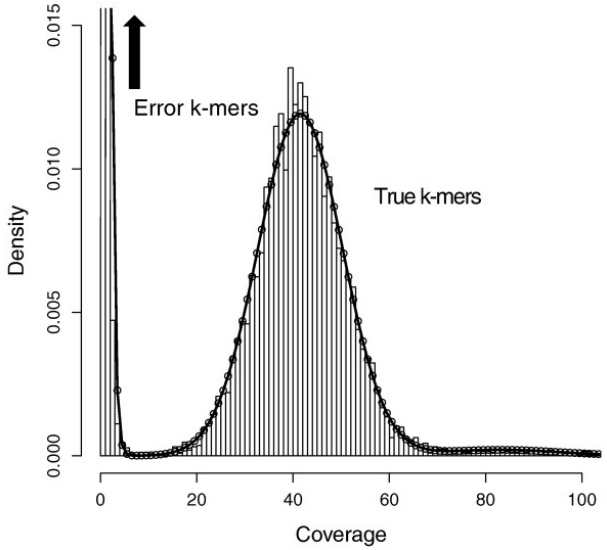


https://github.com/jts/sga/wiki/Preqc

Other pre-assembly steps

Depending on the assembler you use, you may want to:

- Run error correction on the sequence data
- Join paired-end reads
- Assess reads for contaminants



Starting the assembly

Contig building

Greedy assembly

Seed and extend

Overlap graph

de Bruijn graphs

String graphs

..etc etc

… all essentially doing similar things,

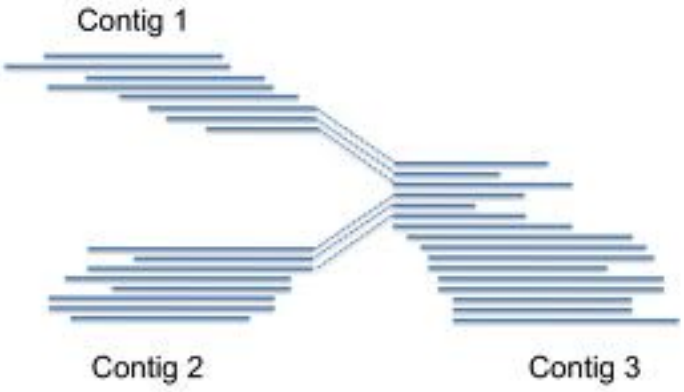
but taking different 'shortcuts' based on needs

Contigs

Contiguous, unambiguous stretches of assembled DNA sequence

Contigs ends correspond to

- Real ends (for linear DNA molecules)
- Dead ends (missing sequence)
- Decision points (forks in the road)



Assembly recipe

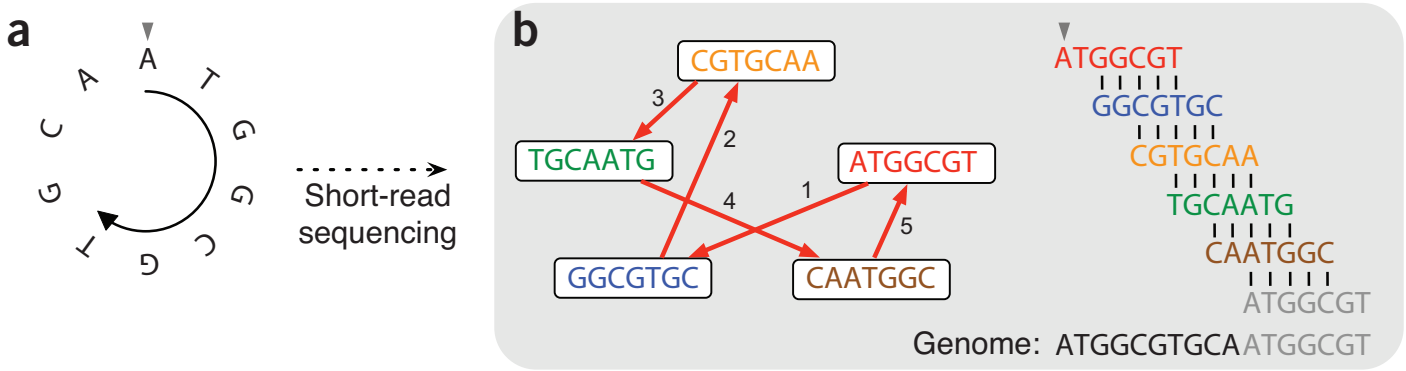
- Find all overlaps between reads
  - hmm, sounds like a lot of work…
- Build a graph
  - a picture of read connections
- Simplify the graph
  - sequencing errors will mess it up a lot
- Traverse the graph
  - trace a sensible path to produce a consensus

Graph

Review: A structure where object are related to one another somehow

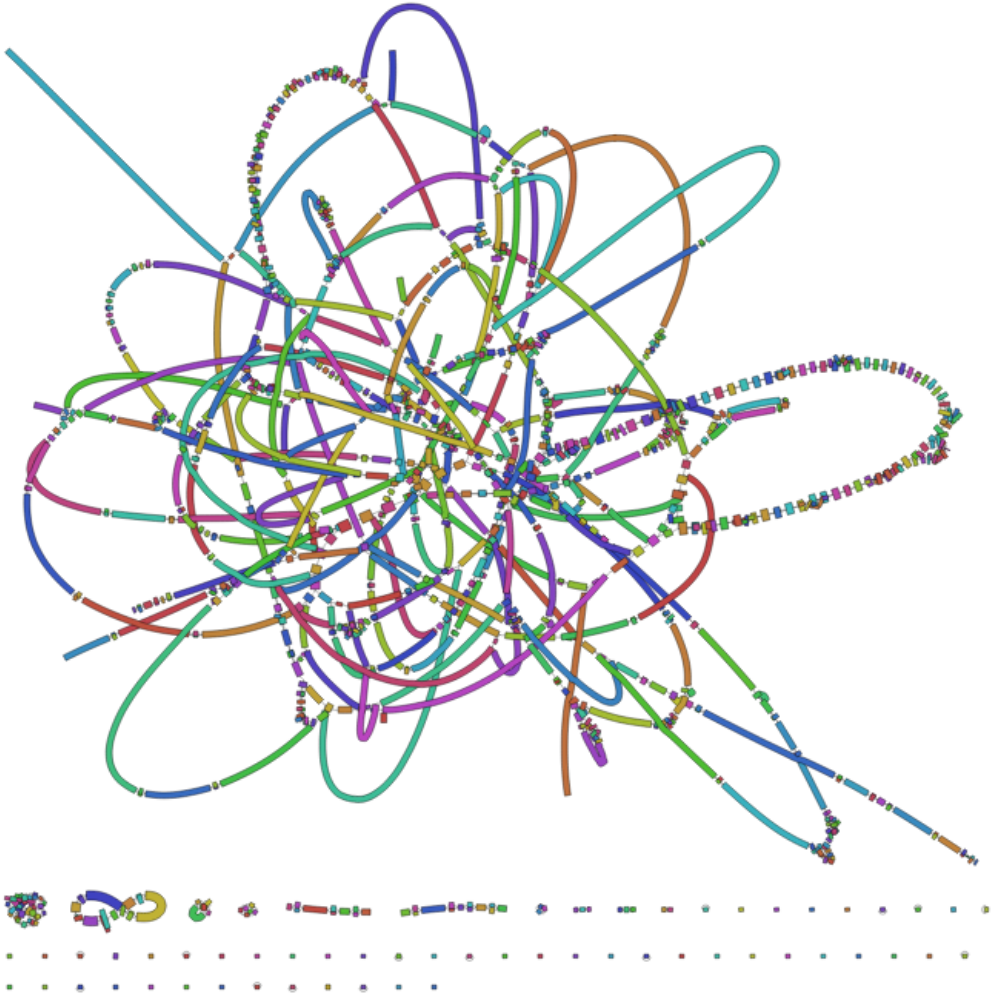
Nodes/Vertices = objects (sequence)

Edges = relationship (overlap)



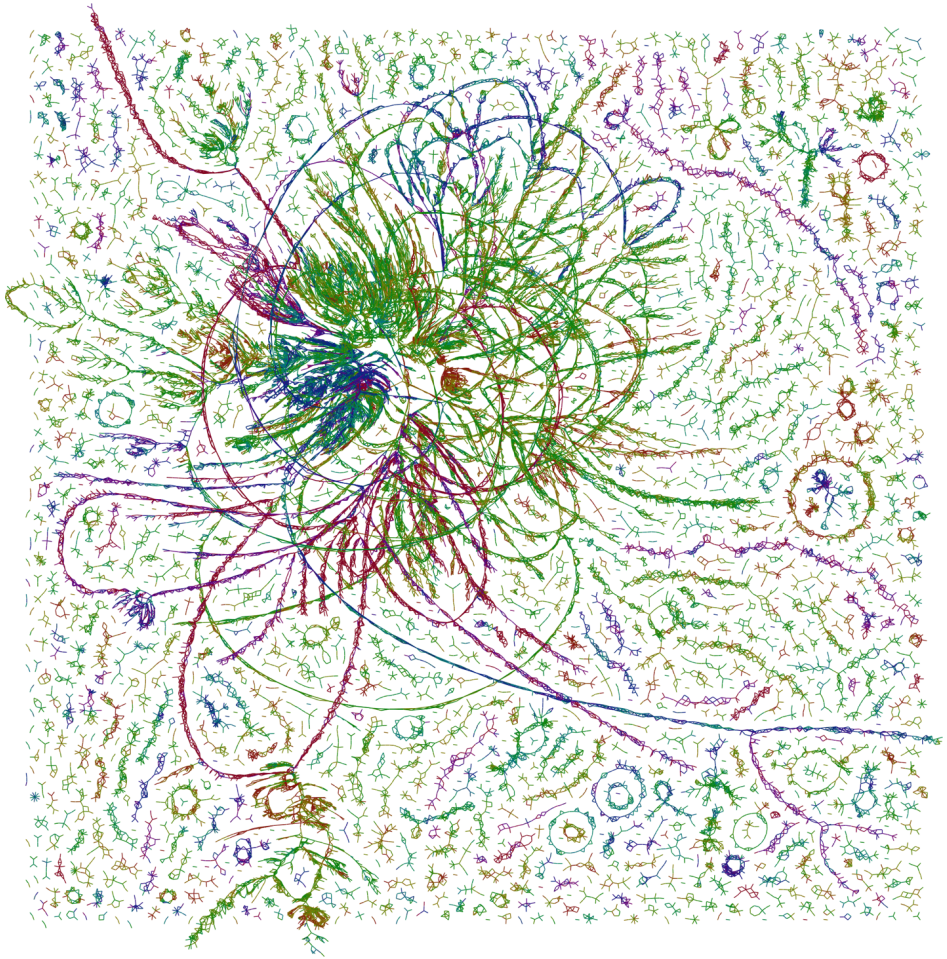
Compeau *et al,* Nature Biotech, 29(11), 2011; https://en.wikipedia.org/wiki/Graph_(discrete_mathematics)

Simple?



https://github.com/rrwick/Bandage/wiki/Effect-of-kmer-size

Erm…



http://armbrustlab.ocean.washington.edu/seastar

OLC and de Bruijn overview

In essence…

For each unconnected graph, at least one per replicon in original sample

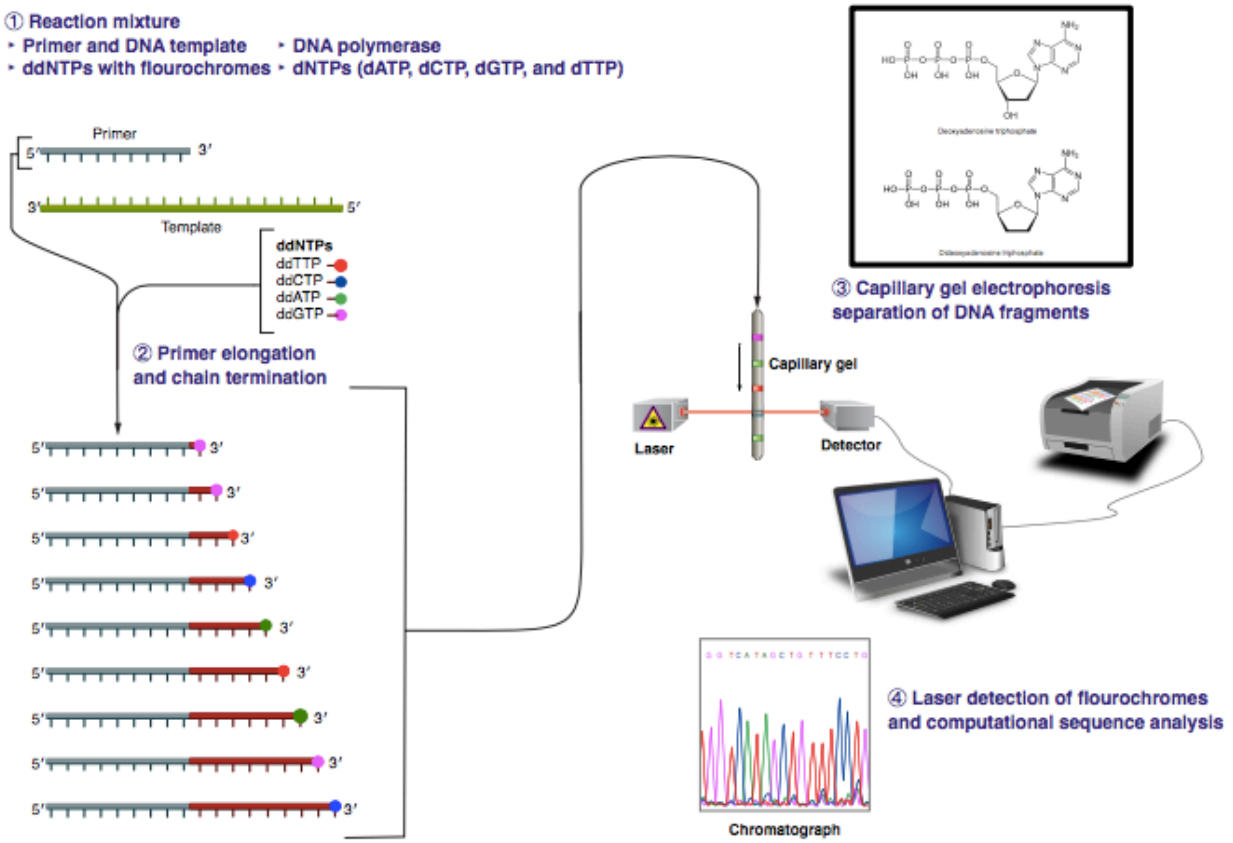
- **Find a path** which visits each node once
  - This is referred to as a **Hamiltonian path/cycle**
- **Form consensus sequences** from paths
  - use all the overlap alignments
  - each of these collapsed paths is a ***contig***

Overlap Layout Consensus Assembly

Used for longer read data

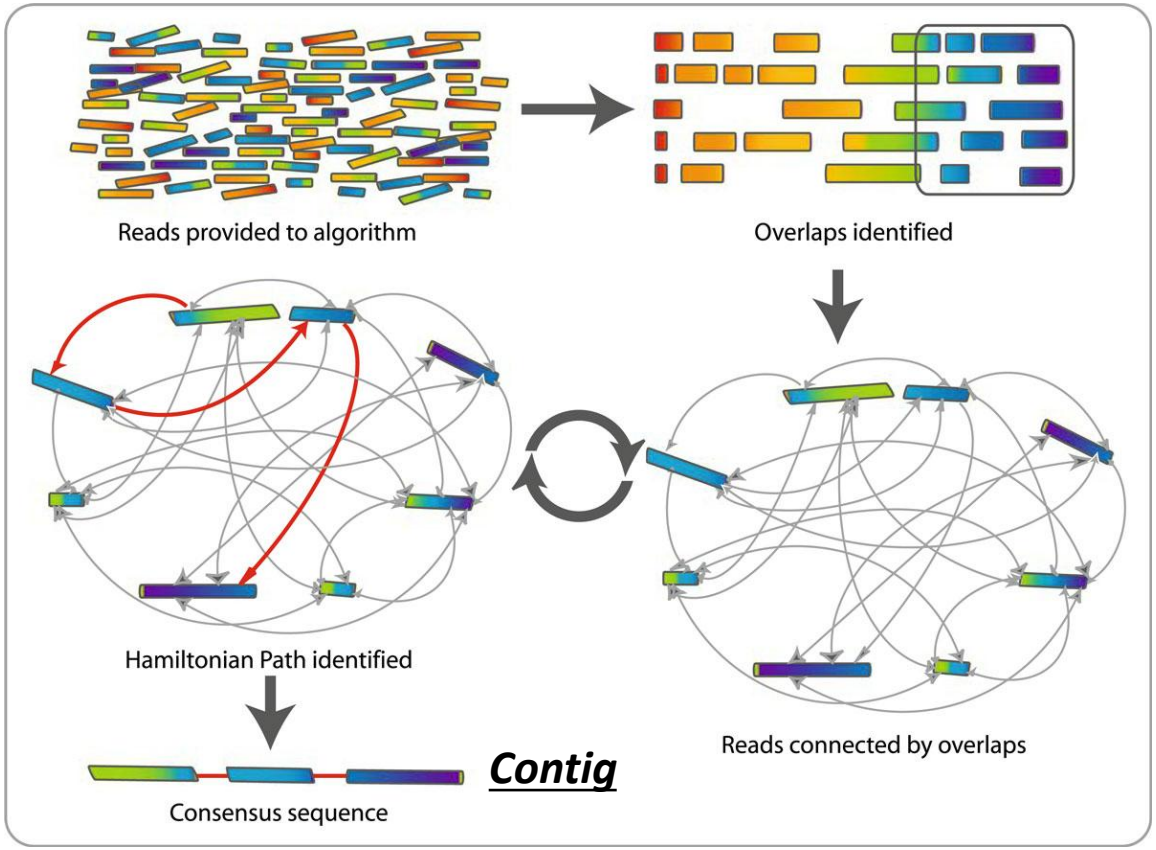
Sanger

Newer variants for PacBio and Oxford Nanopore



By Estevezj - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=23264166

For each unconnected graph, at least one per replicon in original sample



Find a path which visits each node once

Form consensus sequences from paths

OLC assembly steps

Calculate ***overlays***

- Can use BLAST-like methods, but finding common strings (**k-mers**) more efficient

Assemble ***layout*** graph, try to simplify graph and remove nodes (reads) – find Hamiltonian path

Generate ***consensus*** from the alignments between reads (overlays)

Some OLC-based assemblers

Celera Assembler with the Best Overlap Graph (**CABOG**)

- Designed for Sanger sequences, but works with other technologies like 454

Canu – is a fork of the Celera Assembler designed for high-noise single-molecule sequencing (PacBio, Oxford Nanopore)

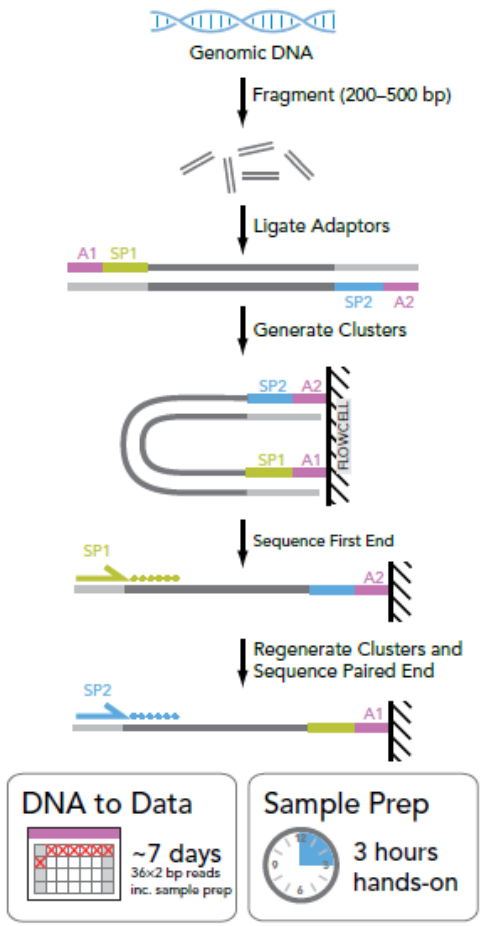
Newbler, a.k.a. GS de novo Assembler - designed for 454 sequences, but works with Sanger reads

De Bruijn graph assemblers

Developed to deal with high-throughput highly accurate short-read data

Uses shotgun data (generally paired-end fragments of 300-500nt)

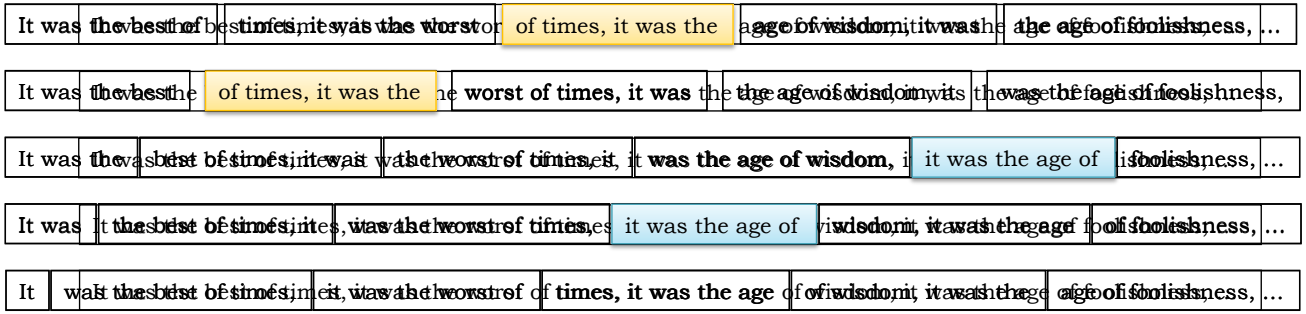
Figure 6B: Paired-End Sequencing



Adapters containing attachment sequences (A1 & A2) and sequencing primer sites (SP1 & SP2) are ligated onto DNA fragments (e.g., genomic DNA). The resulting library of single molecules is attached to a flow cell. Each end of every template is read sequentially.

Shredded Book Reconstruction

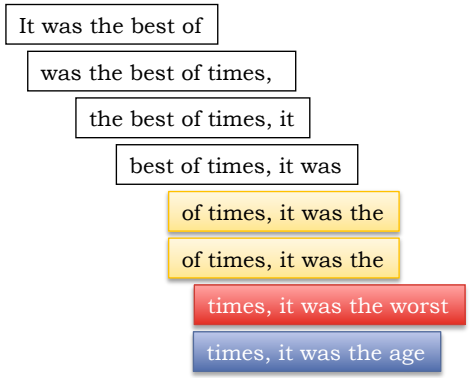
- Dickens accidentally shreds the first printing of A Tale of Two Cities
  - Text printed on 5 long spools



- How can he reconstruct the text?
  - 5 copies x 138, 656 words / 5 words per fragment = 138k fragments
  - The short fragments from every copy are mixed together
  - Some fragments are identical

Greedy Reconstruction

It was the best of
age of wisdom, it was
best of times, it was
it was the age of
it was the age of
it was the worst of
of times, it was the
of times, it was the
of wisdom, it was the
the age of wisdom, it
the best of times, it
the worst of times, it
times, it was the age
times, it was the worst
was the age of wisdom,
was the age of foolishness,
was the best of times,
was the worst of times,
wisdom, it was the age
worst of times, it was



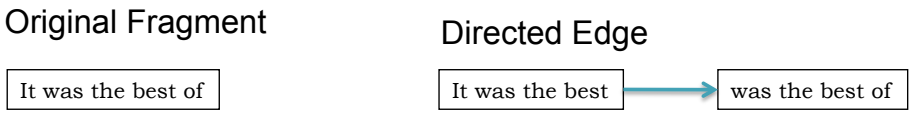
The repeated sequence make the correct reconstruction ambiguous

- It was the best of times, it was the [worst/age]

Model the assembly problem as a graph problem

de Bruijn Graph Construction

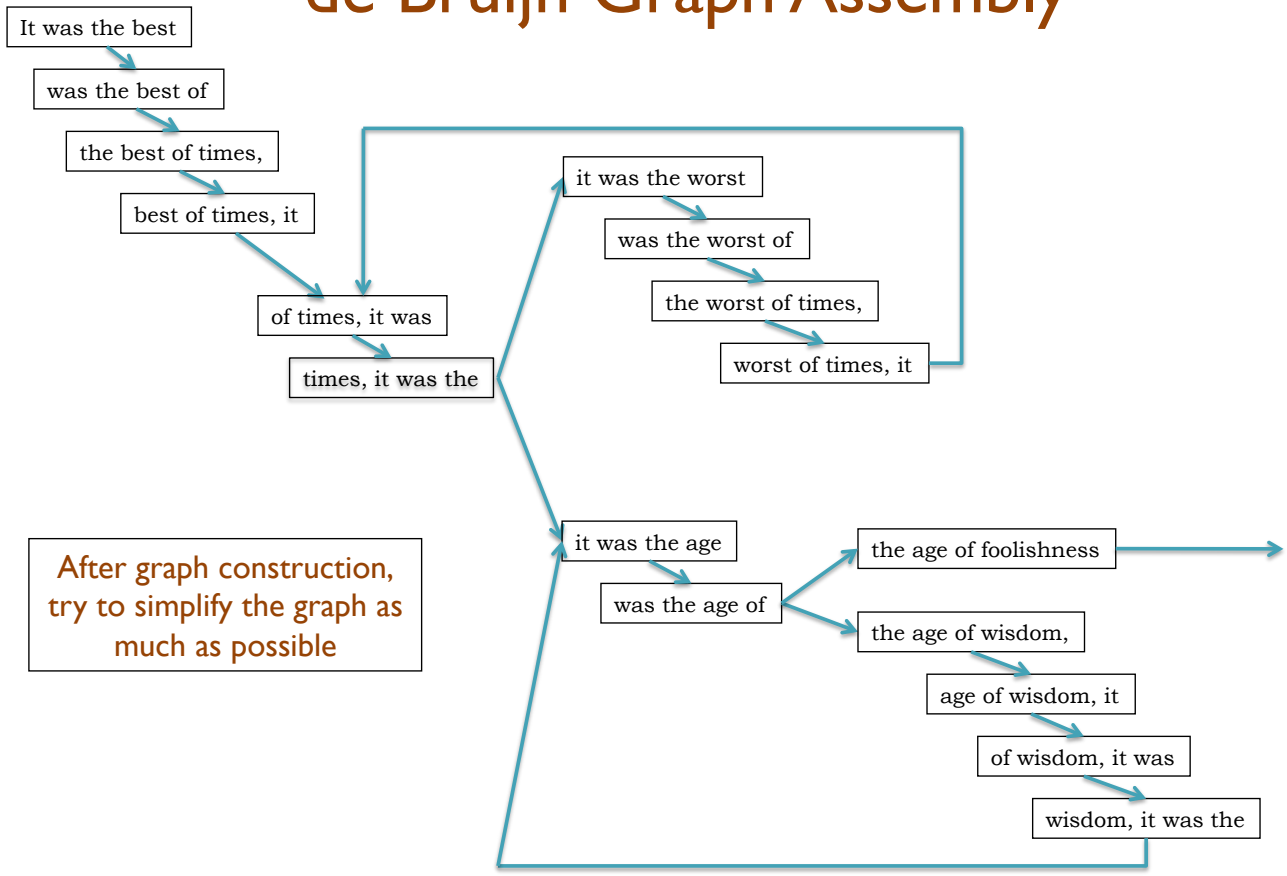
- D_k = (V,E)
  - V = All length-k subfragments (k < l)
  - E = Directed edges between consecutive subfragments
    - Nodes overlap by k-1 words



- Locally constructed graph reveals the global sequence structure
  - Overlaps between sequences implicitly computed

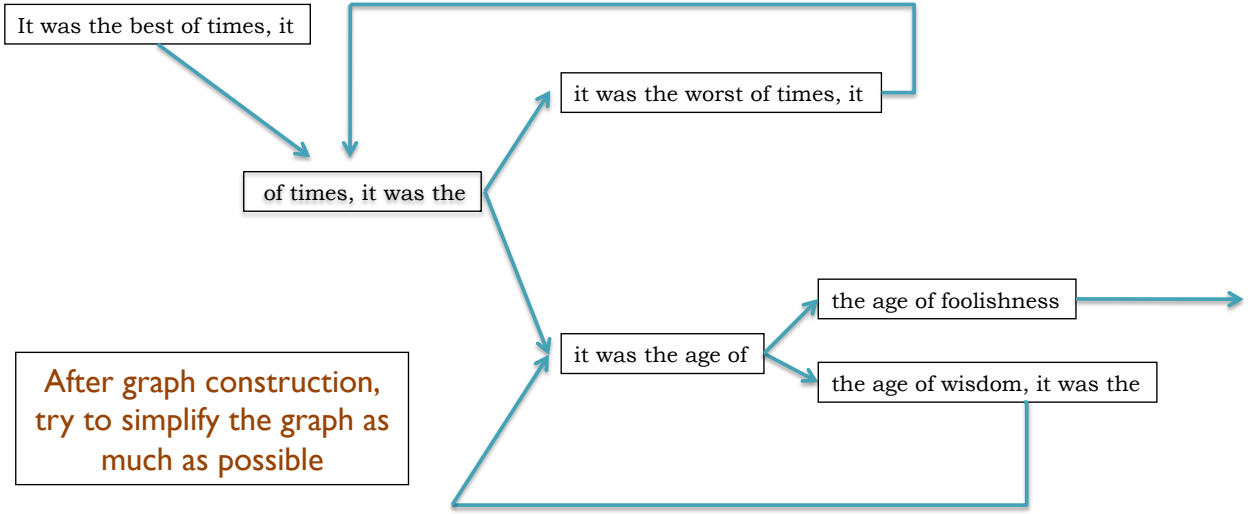
de Bruijn, 1946
Idury and Waterman, 1995
Pevzner, Tang, Waterman, 2001

de Bruijn Graph Assembly



After graph construction, try to simplify the graph as much as possible

de Bruijn Graph Assembly



The full tale

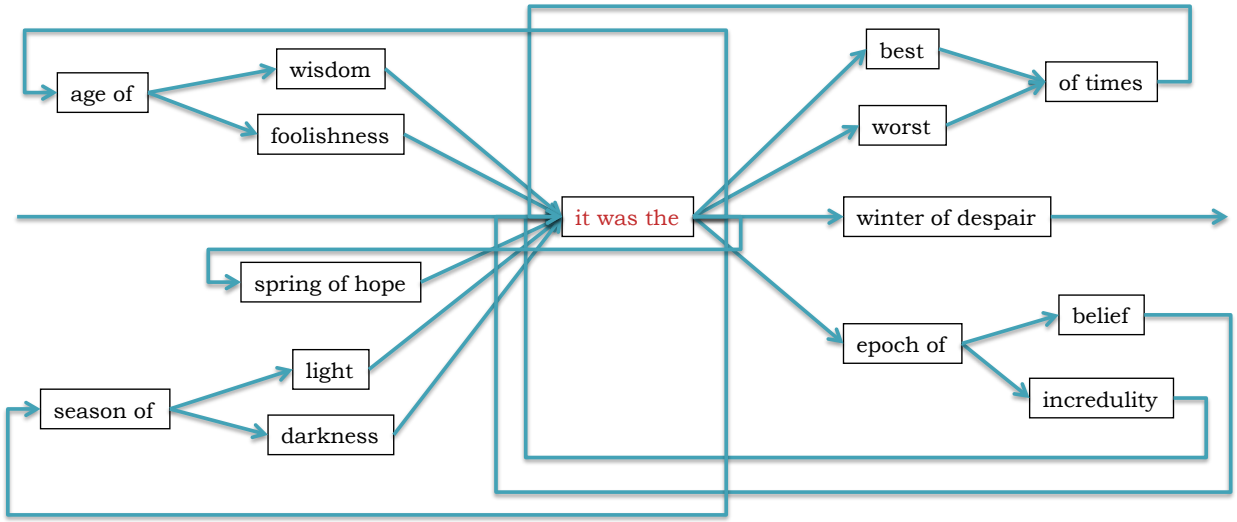
… it was the best of times it was the worst of times …

… it was the age of wisdom it was the age of foolishness …

… it was the epoch of belief it was the epoch of incredulity …

… it was the season of light it was the season of darkness …

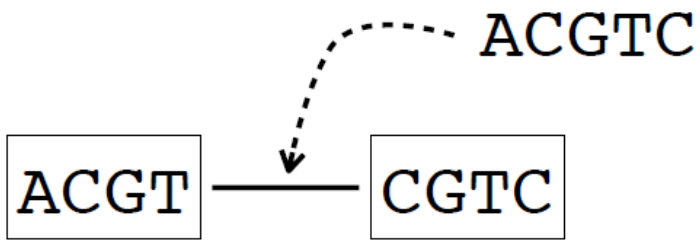
… it was the spring of hope it was the winder of despair …



De Bruijn graphs - concept

- de Bruijn graph
  - k-dimensional graph over four symbols {A, C, G, T}
  - vertex: k-mer -- a string of k nucleotides
  - edge: (k+1)-mer

k=4



Scaffolding

- Now, you have a huge pile of contigs but you want to make them larger. How?
- Add context!
- Link together contigs using ***other*** genomic information
  - Infer contigs position on the genome relative to one another

Linking Contigs via DNA Seq

Illumina sequencing (100 to 300 bp reads)

Paired-end reads

<500bp fragment

Mate pair reads

>250 bp PCRfree reads

10x Genomics
Linked reads

PacBio/ONT long-reads

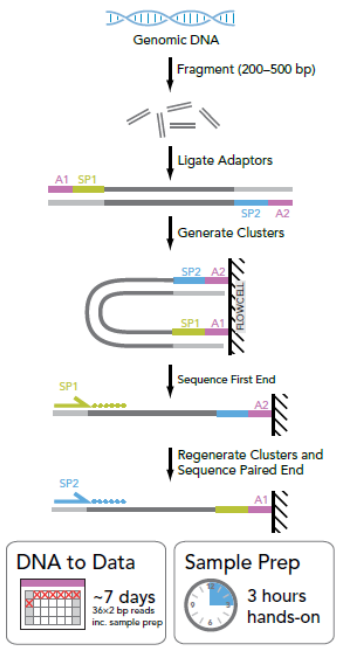
10-100 kb+

Illumina

Modified from Qi Sun, Minghui Wang, Cornell Univ.

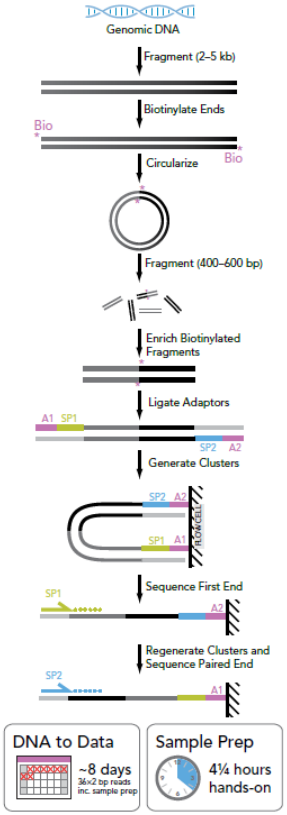
Illumina Sequencing

Figure 6B: Paired-End Sequencing

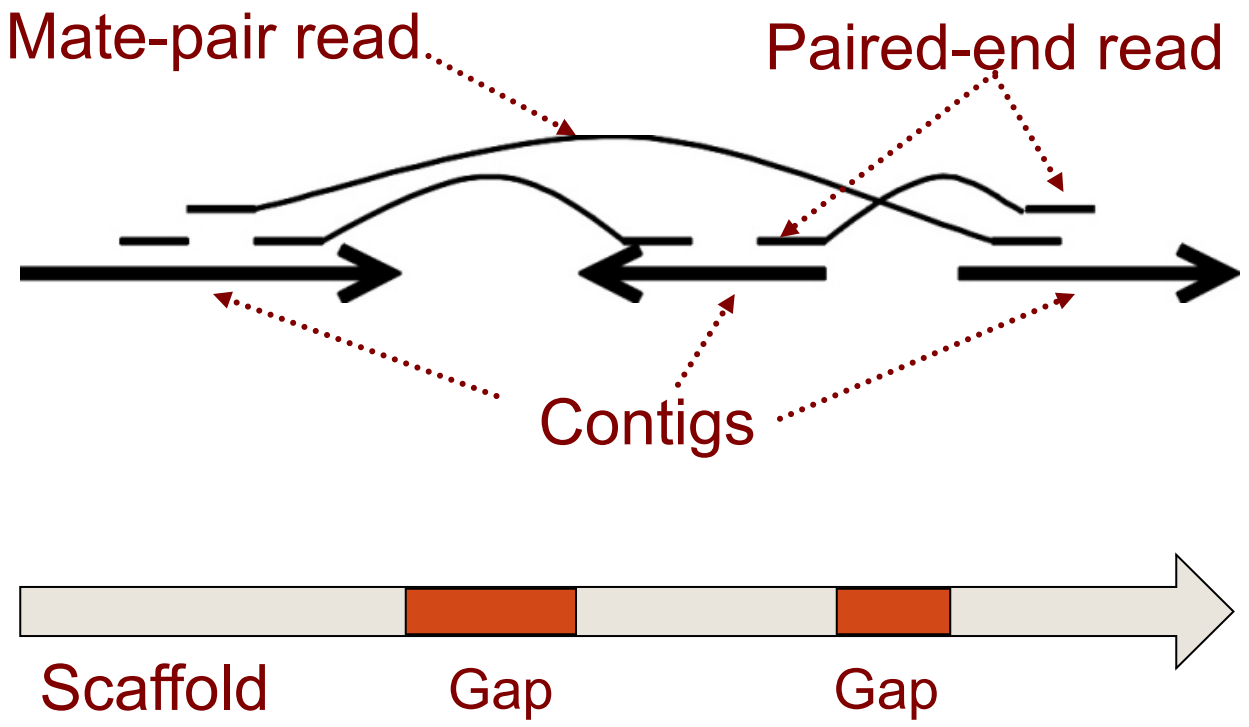


Adapters containing attachment sequences (A1 & A2) and sequencing primer sites (SP1 & SP2) are ligated onto DNA fragments (e.g., genomic DNA). The resulting library of single molecules is attached to a flow cell. Each end of every template is read sequentially.

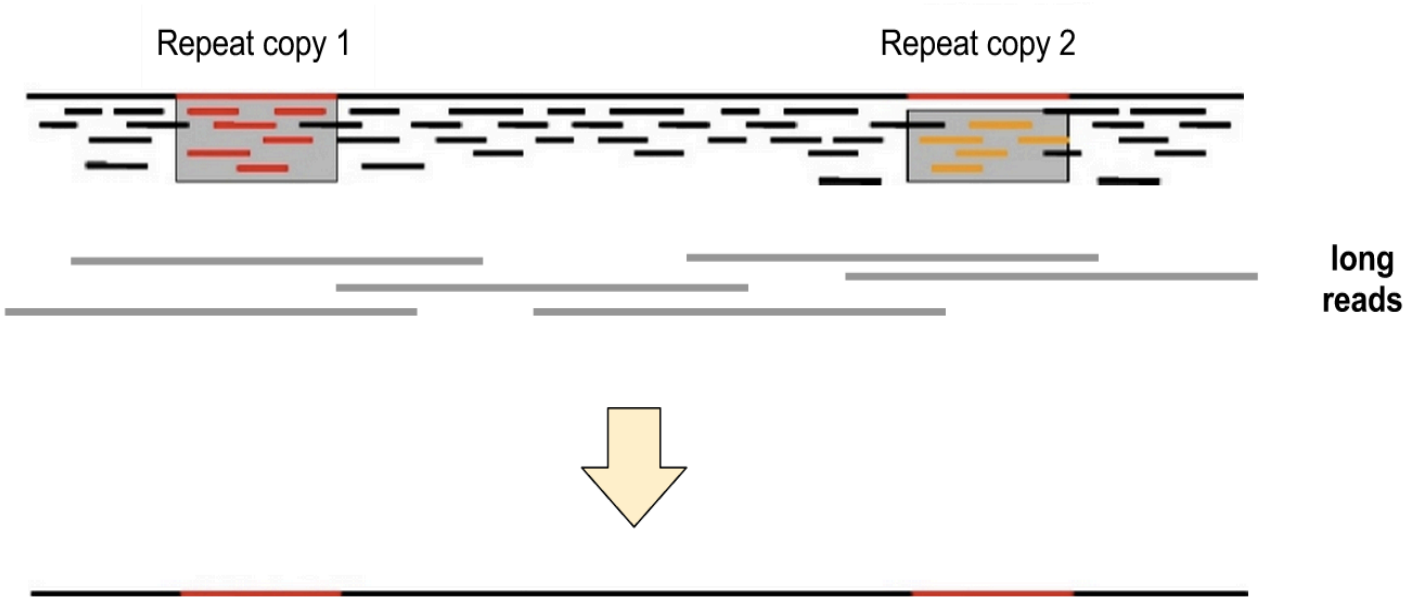
Figure 6C: Mate Pair Sequencing



Contigs to scaffolds



Long reads



Starting a new assembly project

Planning a genome sequencing project?

- How **large and/or complex** is my genome?
- **Ploidy** (number of sets of chromosomes) of the genome?
- Are the cells **multinucleated**?
- How much of the genome is **repetitive**? What is the repeat size distribution?
- Is my genome **heterozygous**?
- Is a good quality **genome of a related species** available?
- Does **other useful data** exist that could improve my assembly
- ***BUDGET***

What complicates assembly?

Repetitive sequence

- causes nodes to be shared, locality confusion

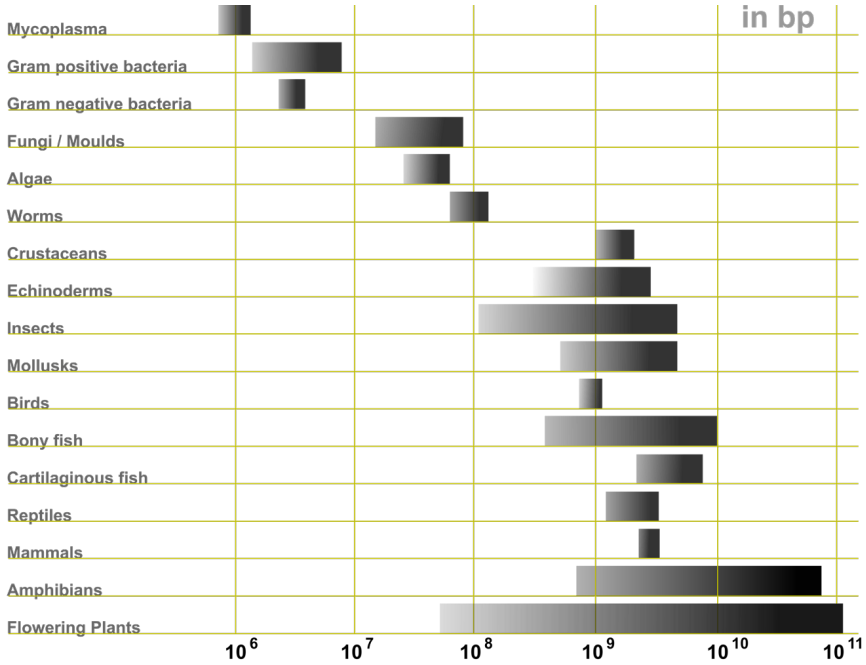
Variation and Heterozygosity

- Causes lots of branch points in a graph

Sequencing read issues (technical, but very importan

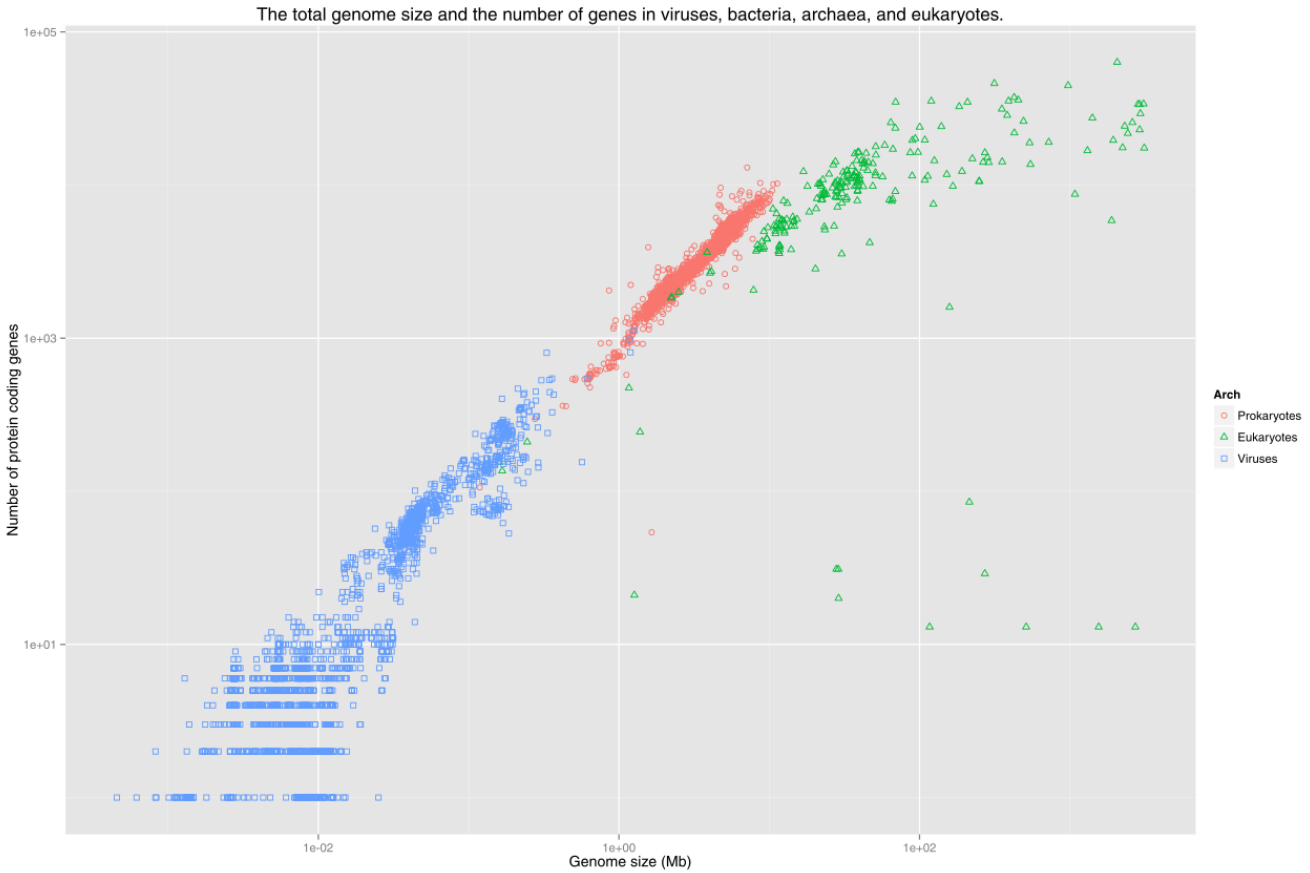
- Sequencing artifacts (adapters, vector sequence, etc)
- Sequence errors
  - Generally random, but some technologies have systematic errors
- You remembered to QC, trim, and cleam up the data, right?

Genome size/complexity



By Abizar at English Wikipedia, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=19537795

Genome size/complexity



By Estevezj - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=23704739

How large is my genome?

The size and complexity of the genome can be estimated from the ploidy of the organism and the DNA content per cell

This will affect:

- How many reads will be required to attain sufficient coverage (typically 10x to 100x, depending on read length)
- What sequencing technology to use (short vs. long reads)
- What computational resources will be needed (generally amount of memory needed and length of time resources will be used)

Ploidy

Number of sets of chromosomes in a cell (N)

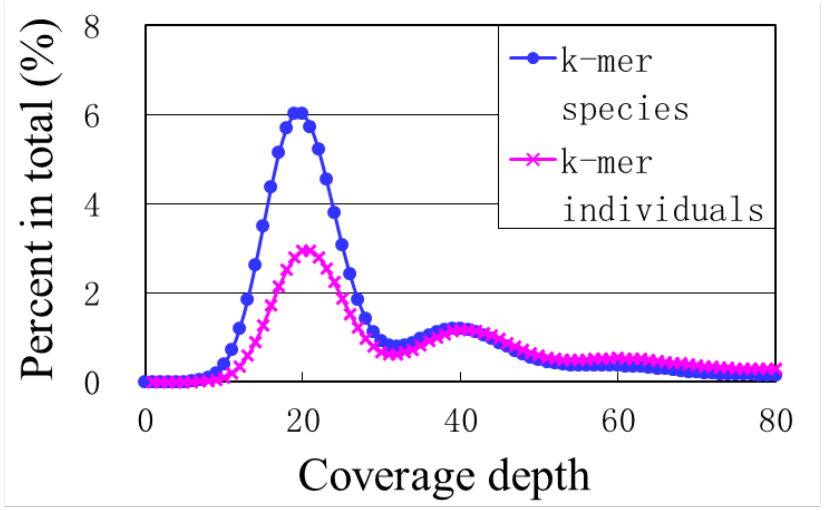
- Bacteria – 1N
- Vertebrates – 2N (human, mouse, rat)
- Amphibians – 2N to 12N
- Plants – 2N to ??? (wheat is 6N)

Repetitive sequences

Most common source of assembly errors

If sequencing technology produces reads > repeat size, impact is much smaller

Most common solution: generate reads or mate pairs with spacing > largest known repeat

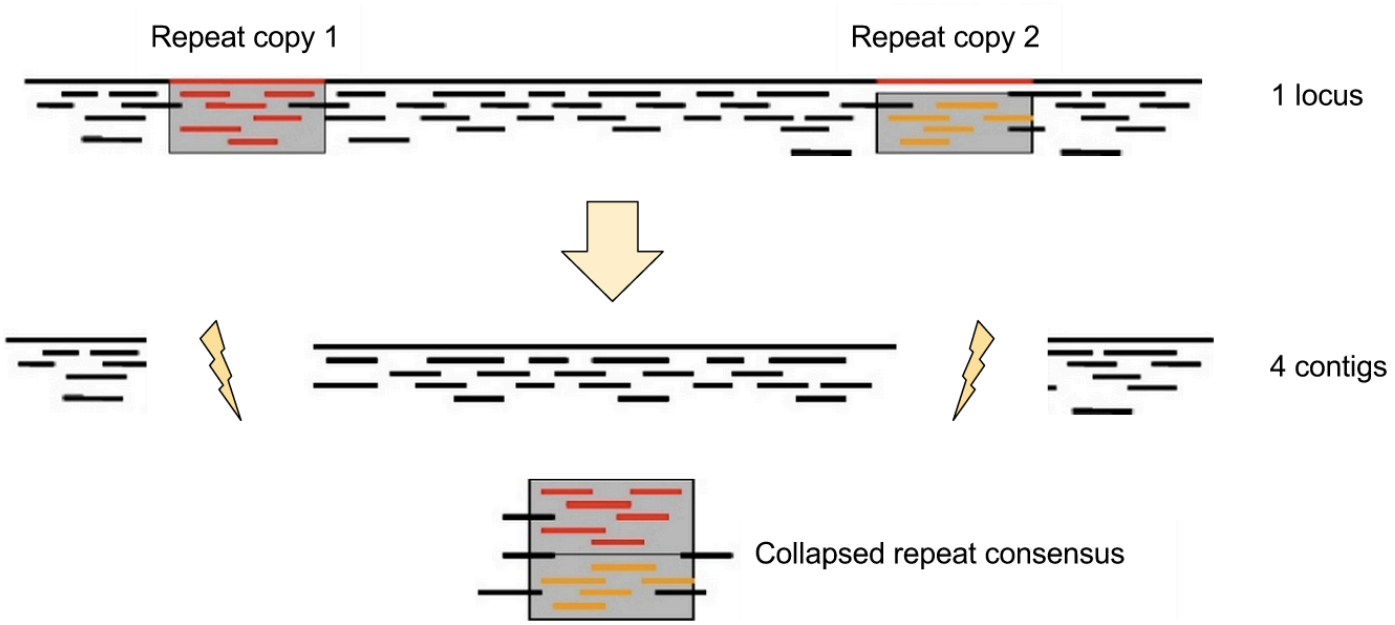


What is a repeat?

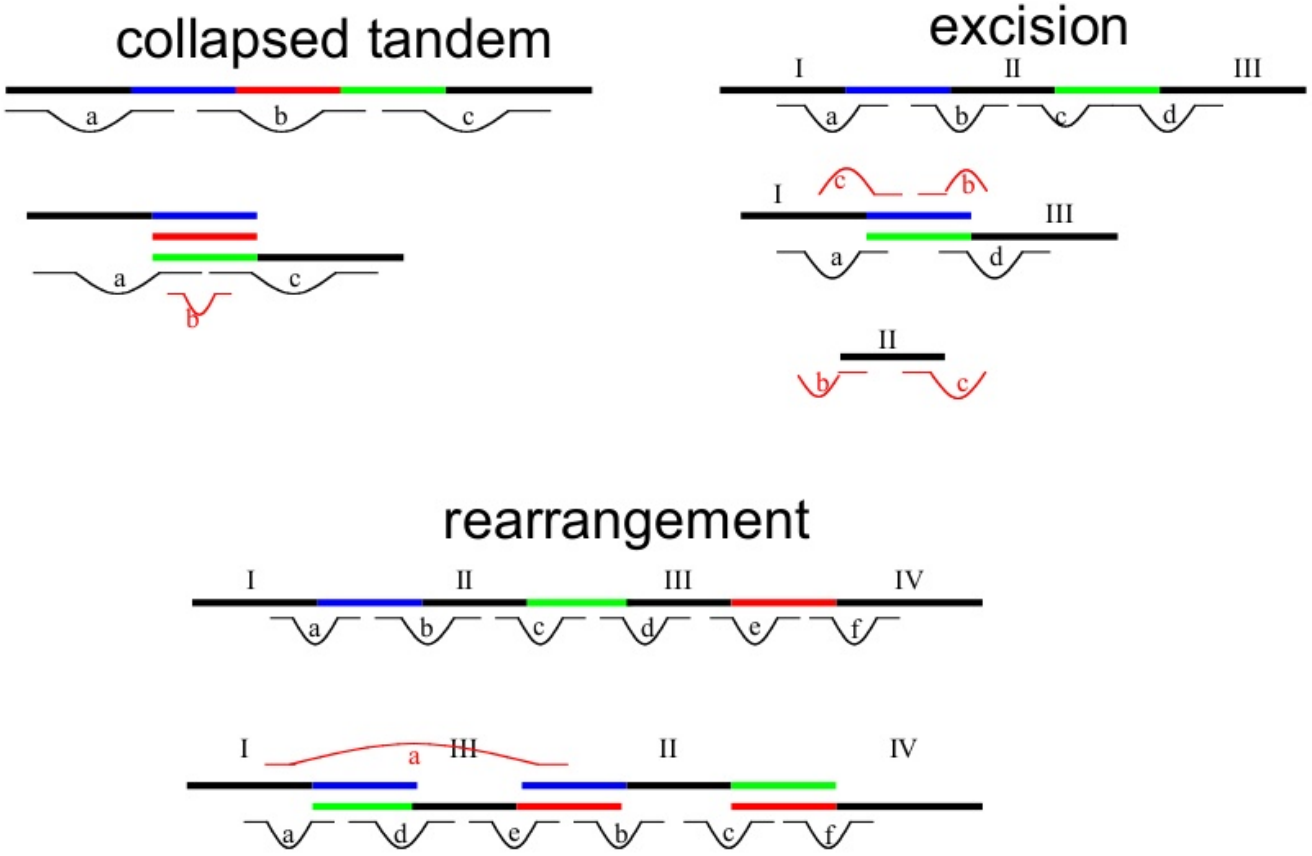
A segment of DNA
which occurs more than once
in the genome sequence

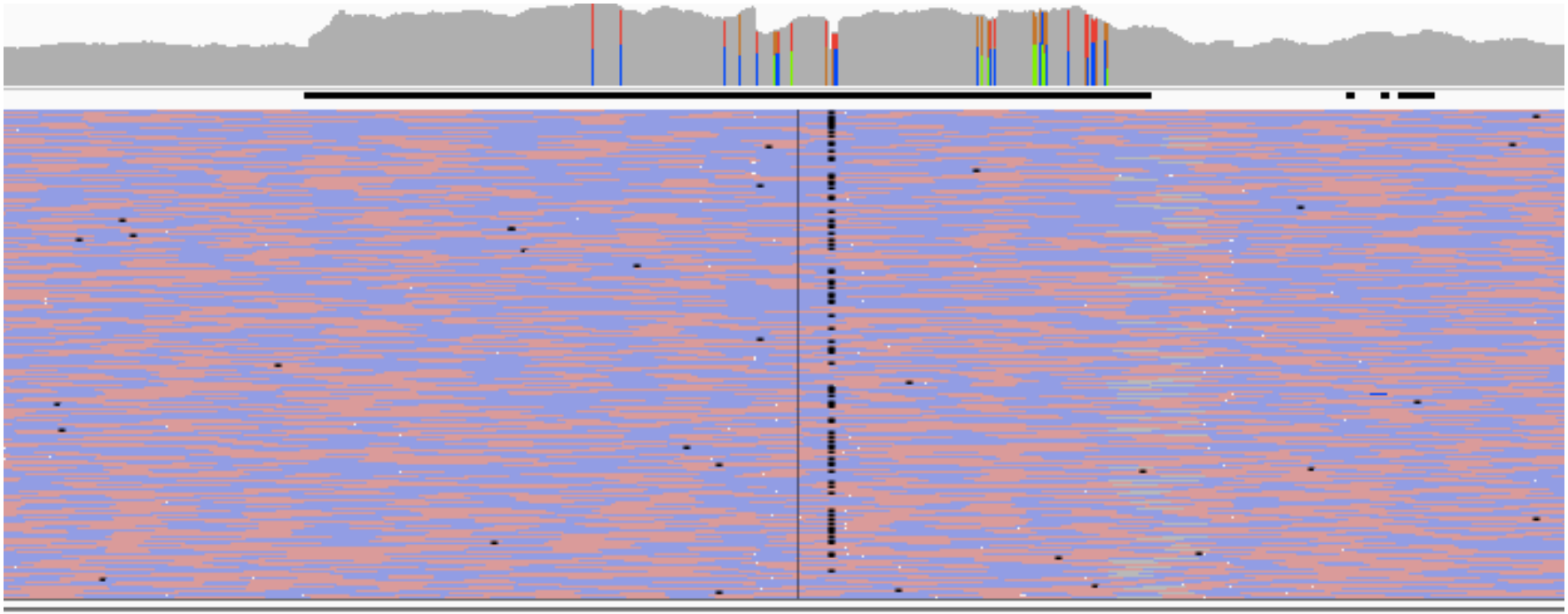
- Very common
  - Transposons (self replicating genes)
  - Satellites (repetitive adjacent patterns)
  - Gene duplications (paralogs)

Assembling repeats



Repeat mis-assembly



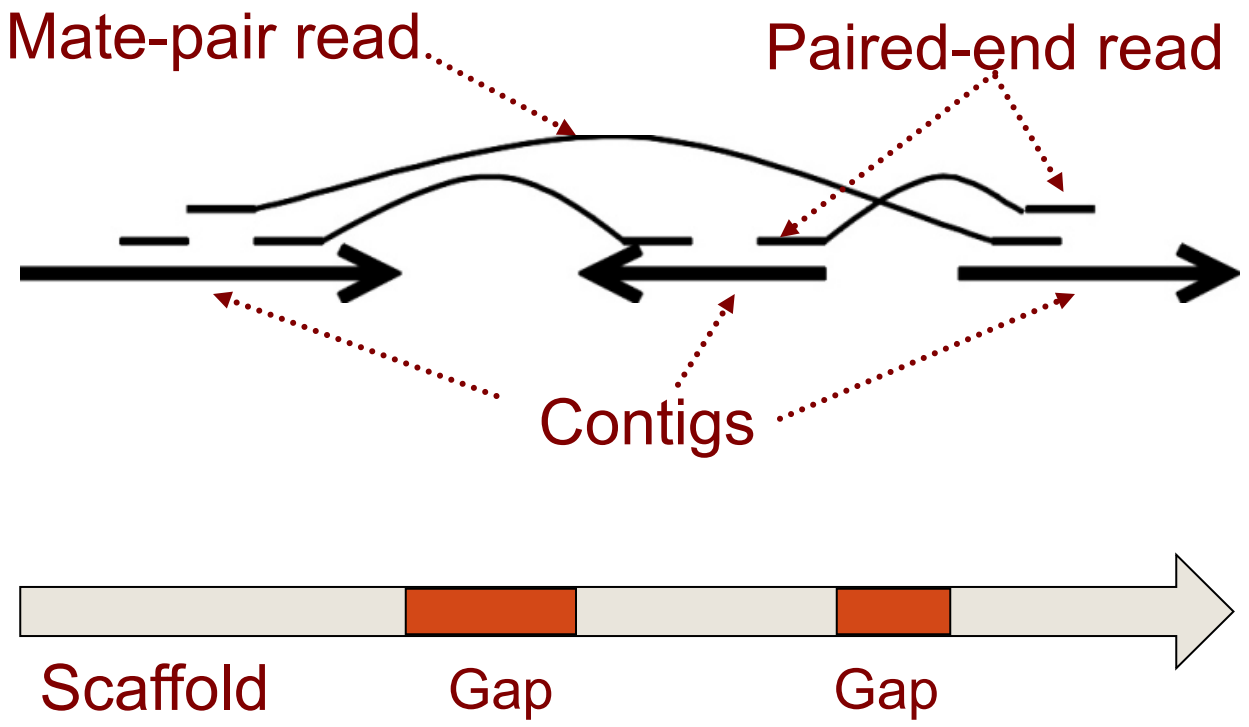


Getting around repeats

Use 'other information' to work around the limitation you have with short reads

- Use '***mate-pairs***' to span repetitive regions and create larger scaffolds
- Use long read data to span repetitive regions
- Cloned libraries – BACS, cosmids, fosmids, etc
- Use 'linked reads' via technology like 10x
- Chromosomal interactions (HiC, Dovetail)

Contigs to scaffolds



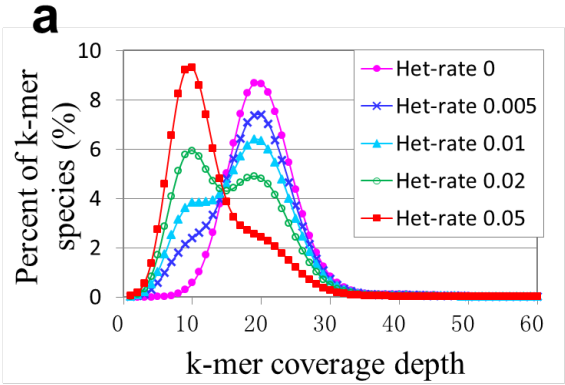
Heterozygosity

Heterozygous – Locus-specific; diploid organism has two different alleles at the same locus.

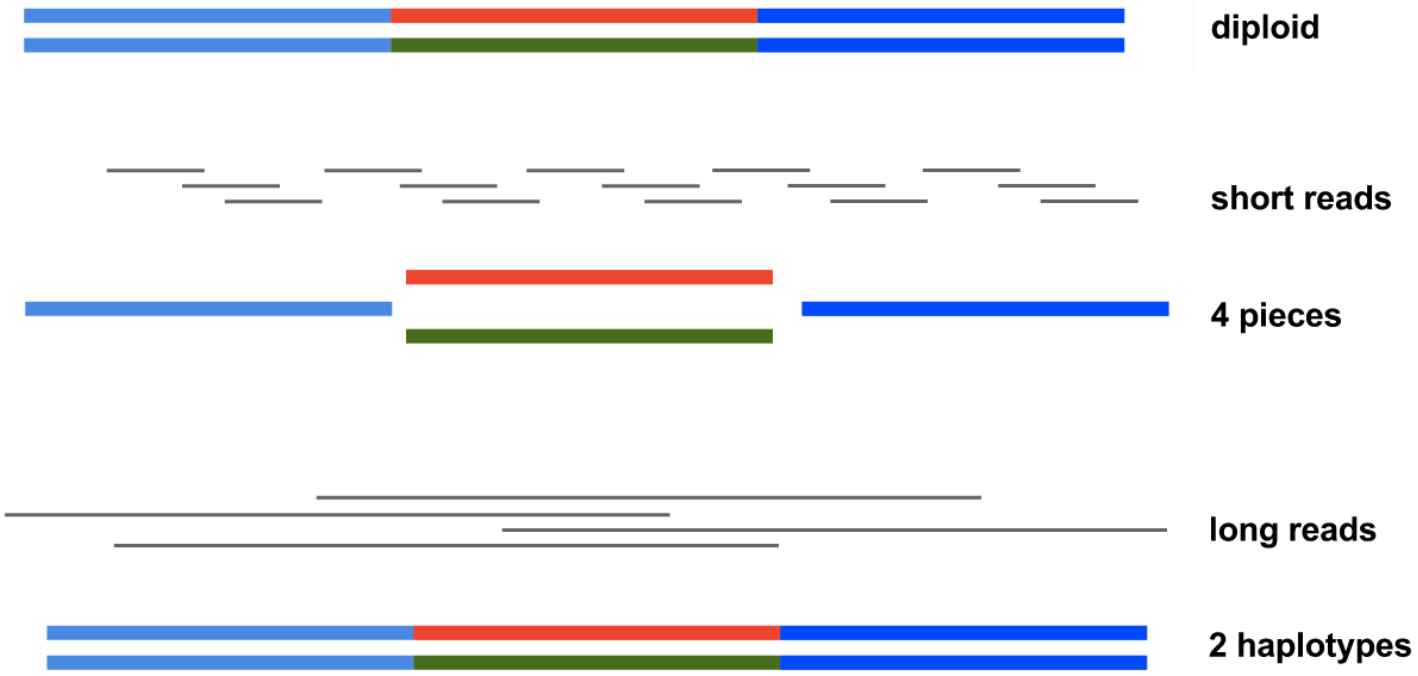
Heterozygosity is a metric used to denote the probability an individual will be heterozygous at a given allele.

Higher heterozygosity == more diverse == harder to assemble

Unfortunately, assemblies are represented (for now) as haploid. So this is a major problem!



Heterozygosity



Genome(s) from related species

Preferably of good quality, with large reliable scaffolds

Help guiding the assembly of the target species

Help verifying the completeness of the assembly

Can themselves be improved in some cases

But to be used with caution – can cause errors when genome architecture is different!

- ***Large-scale genomic rearrangement in particular is a problem***

Strategies for assembly

Sequencing and assembly strategies are interdependent!

- ***Example: Bacterial genome assembly***
  - Generate PacBio or Oxford Nanopore reads, assemble with Flye or Canu
  - Generate both PacBio and Illumina, use a hybrid assembler like Unicycler
  - Generate Illumina reads and assemble with Velvet or SPAdes

Strategies and technology differ per assembler

- **Supernova –** only utilizes 10x Chromium linked-read libraries
- **SOAPdenovo** – allows more flexibility, but optimized for Illumina shotgun libraries generated by BGI (no surprise, since they developed SOAPdenovo)
- **ALLPATHS-LG** – strict requirement for two library types
- **Canu –** Long-reads only, PacBio or Oxford
- **Falcon –** only PacBio

Typical sequencing strategies

Bacterial genome

- PacBio or Oxford Nanopore sequences at 40-50x coverage, self-correction and/or hybrid correction (using Illumina data) and assembly using Canu, Flye
- 2 x 300bp overlapping paired-end reads from Illumina MiSeq

Larger genomes

- Paired-end (2x250 nt overlapping fragments) 50-100x
- Mate-pair (1, 3 and 10 kb libraries) 150-250 nt reads – 25-50x
- 10x Genomics run, assembly using ***Supernova***
- Long-reads if $$$ is available

Additional useful data

Fosmid libraries

- End sequencing adds long-range contiguity information
- Pooled fosmids (~5000) can often be assembled more efficiently

Genetic or physical maps

- Location of alleles or physical markers in reference to one another
- May be hard to produce (years to decades, $$$$)

RNA-Seq (transcripts)

- Assembled transcripts from genes, which can be used to order scaffolds
- Require their own independent assembly

Assembly strategies and algorithms

In most cases, start with cleanup and error correction of raw reads

For long reads (>500 nt), Overlap/Layout/Consensus (OLC) algorithms work best. **Modern example: Canu, wtdbg2**

For short reads, De Bruijn graph-based assemblers are most widely used. **Modern example: MEGAHIT, SPAdes**

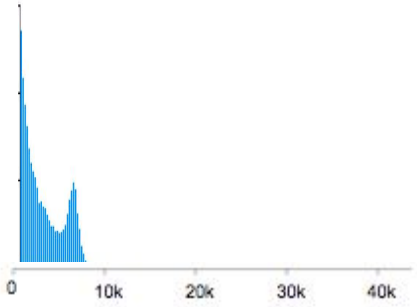
Key points:

- There is no simple solution, best to try different assemblers and strategies
- Use simple metrics to gauge quality of assembly
- The field is rapidly evolving, like the sequencing technology

Long-read sequencing

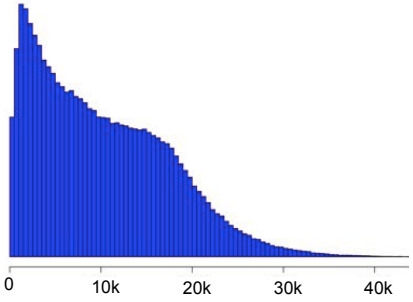
Long Read Sequencing Technology

Moleculo



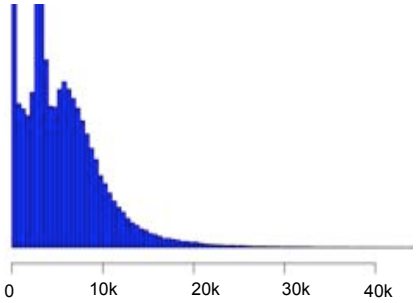
(Voskoboynik et al. 2013)

PacBio RS II



CSHL/PacBio

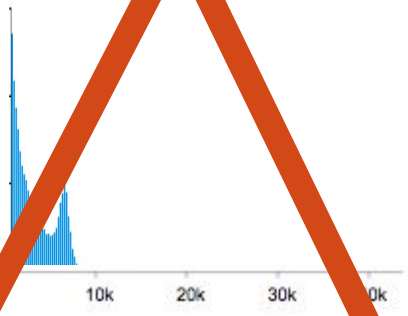
Oxford Nanopore



CSHL/ONT

Long Read Sequencing Technology

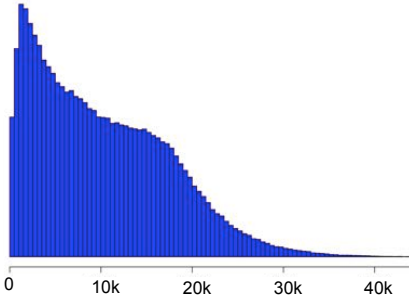
Moleculo



(Voskoboynik et al. 2013)

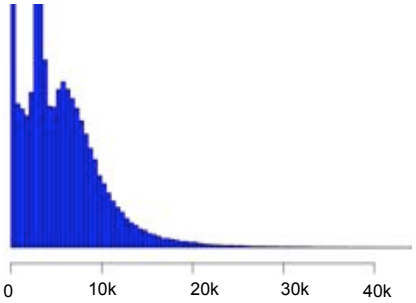
PacBio Sequel II

(~~owned by Illumina~~)



CSHL/PacBio

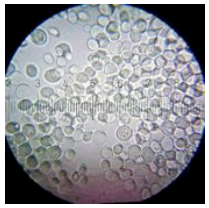
Oxford Nanopore

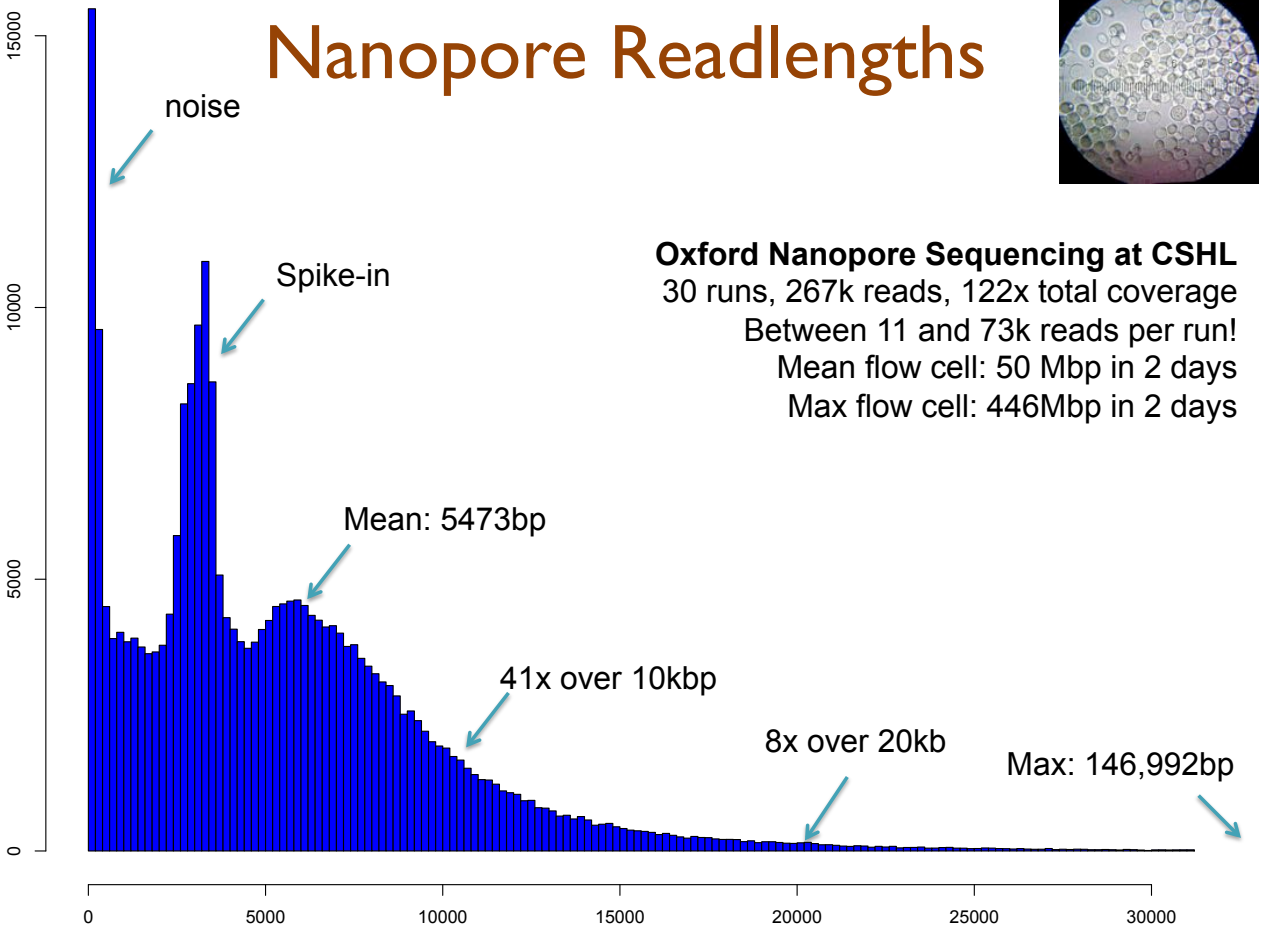


CSHL/ONT

What a difference a year makes…

Nanopore Readlengths



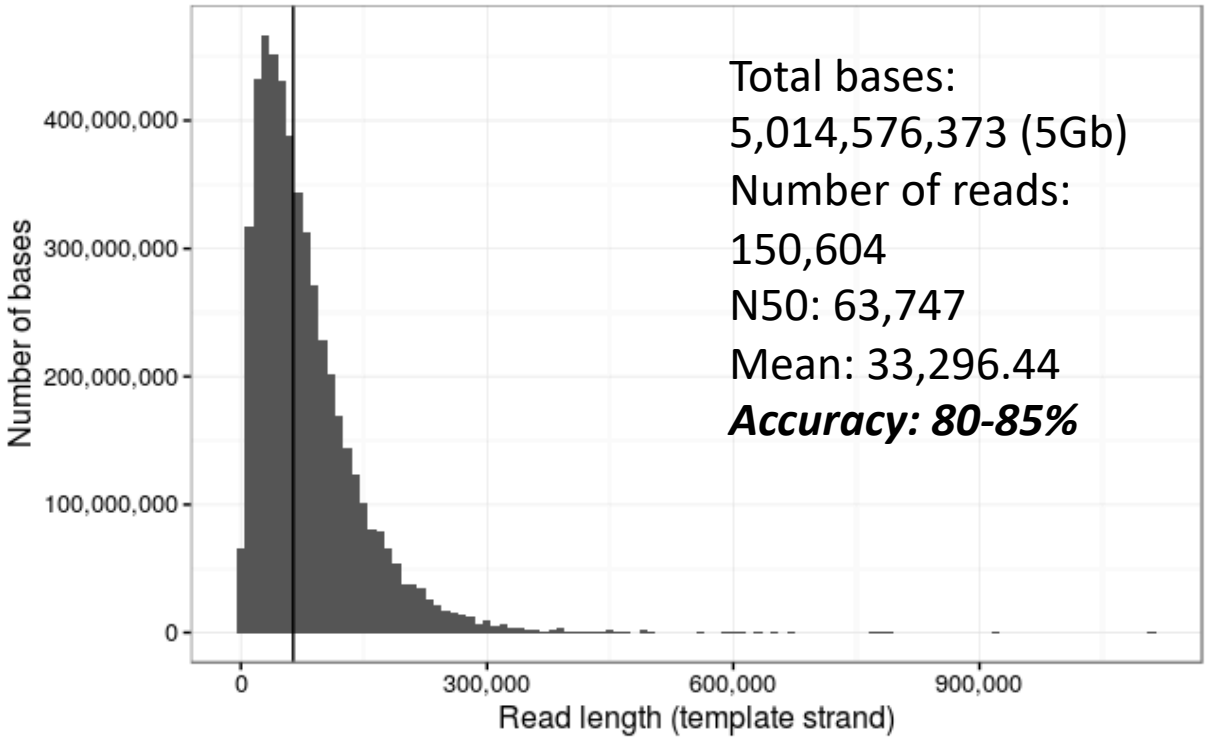


Oxford Nanopore Sequencing at CSHL
30 runs, 267k reads, 122x total coverage
Between 11 and 73k reads per run!
Mean flow cell: 50 Mbp in 2 days
Max flow cell: 446Mbp in 2 days

Oxford Nanopore

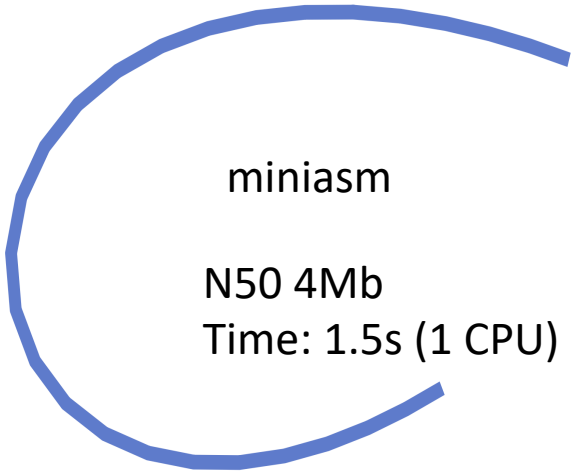
2017

Whale watching: *E. coli*



E. coli: genome assembly in 8 reads

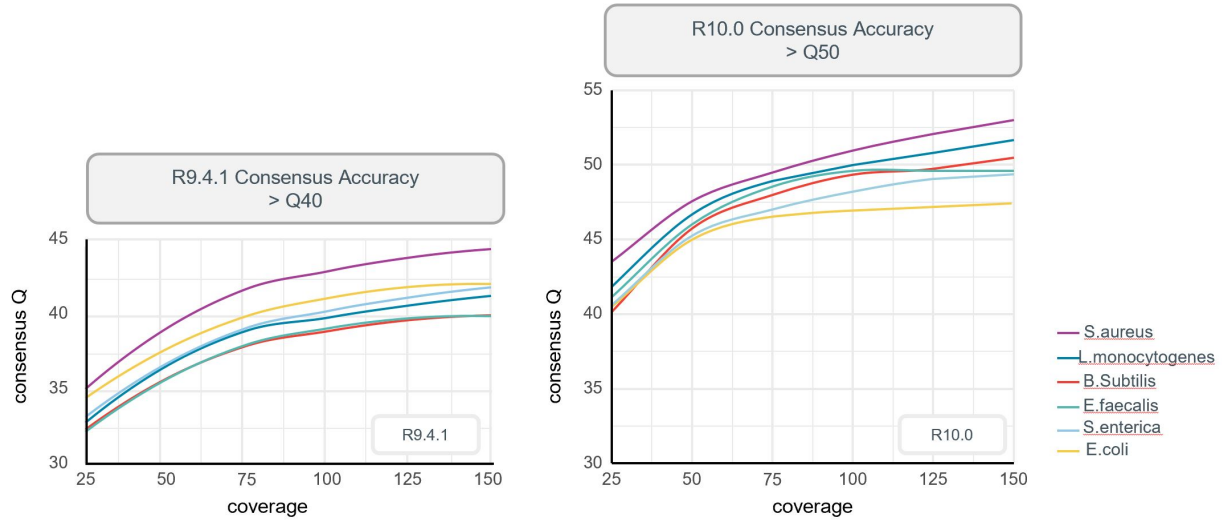
Read	Length	Ref start	Ref end	Time (m)
1	876991	4398844	634183	32.48
2	696402	470003	1166405	25.79
3	799047	1137438	1936485	29.59
4	642071	1759431	2401502	23.78
5	826662	2106227	2932889	30.61
6	883962	2699626	3583588	32.73
7	825191	3285196	4110387	30.56
8	463341	3995967	4459308	17.16



1x coverage!

Oxford Nanopore

2020



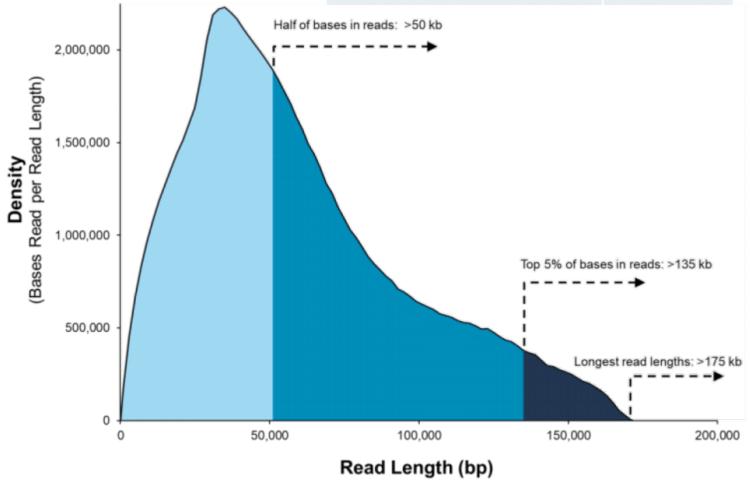
PacBio

PACBIO

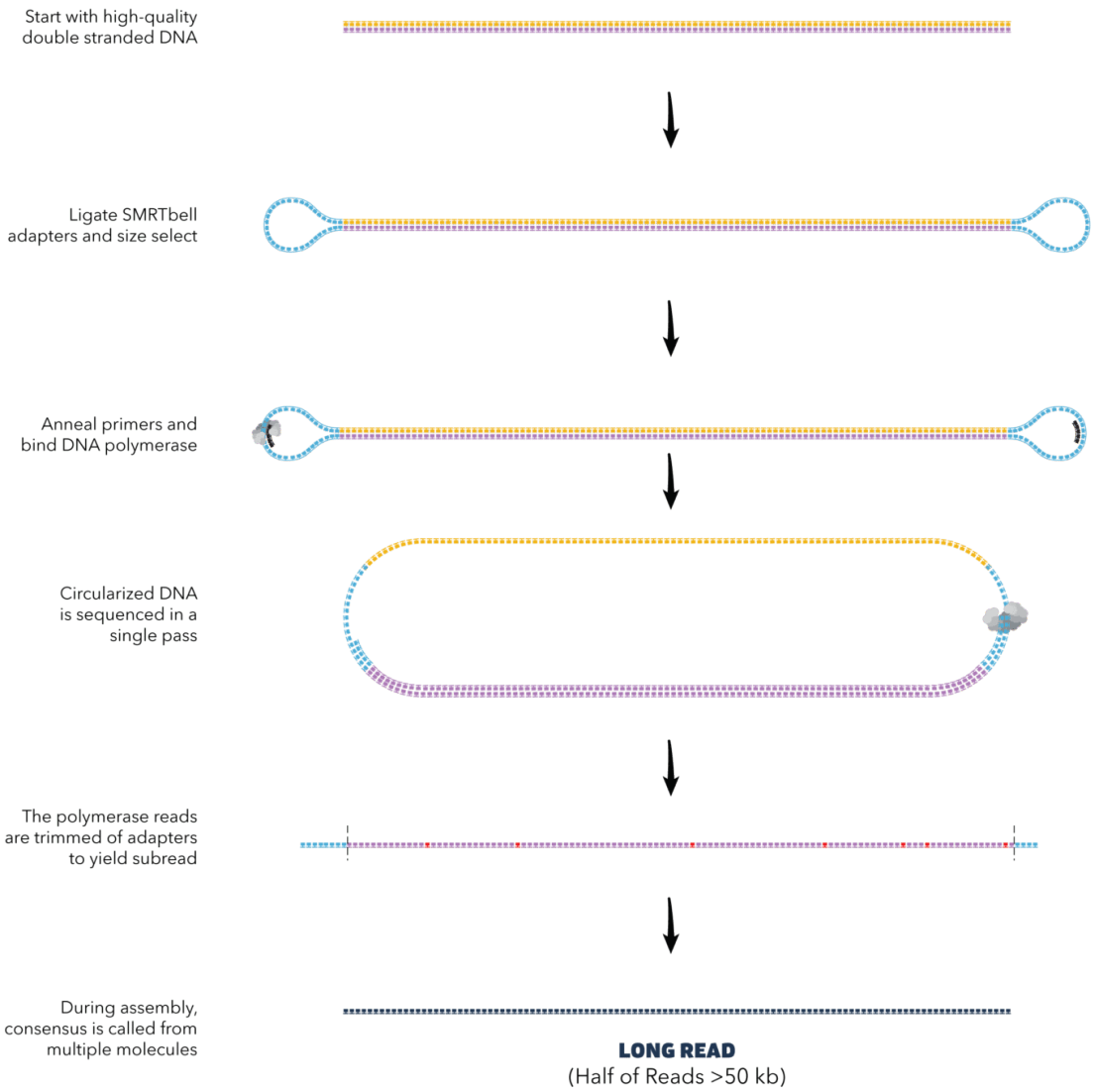
SEQUEL II SYSTEM PERFORMANCE

- ~8× increase in yield
- Reduced project time
- Lower cost
- Equivalent performance to Sequel
  - Read length
  - Accuracy
  - Minimal bias

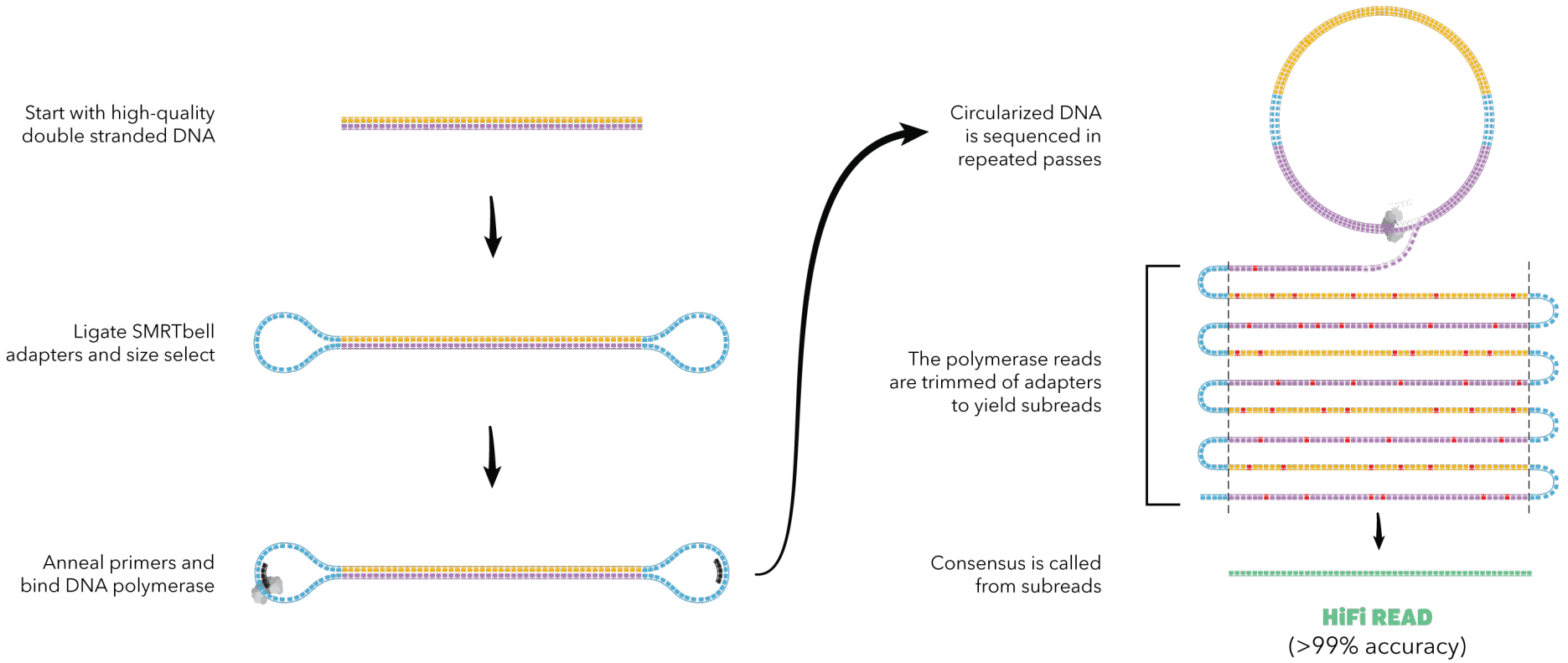
METRIC	
Number of Raw Bases	166 Gb
Total Reads	5,201,973
Half of Bases in Reads	>51,863
Longest Read Lengths	>175,000



35 kb size-selected E. coli library using the SMRTbell Template Prep Kit on a Sequel II System (1.0 Chemistry, Software v7.0, 15-hour movie)



PacBio – Raw reads (CLR)



PacBio Circular Consensus (HiFi)

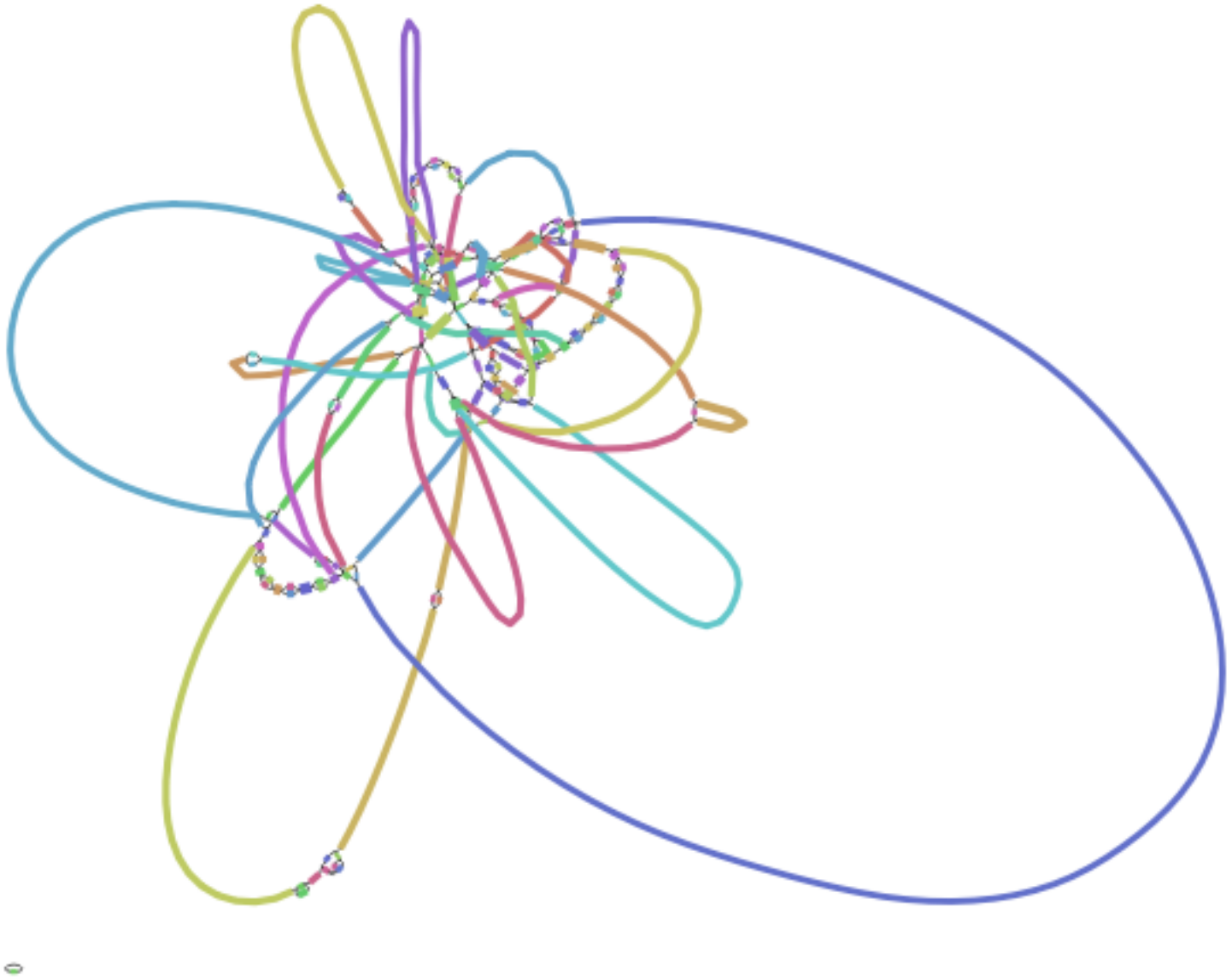
Long read assembly

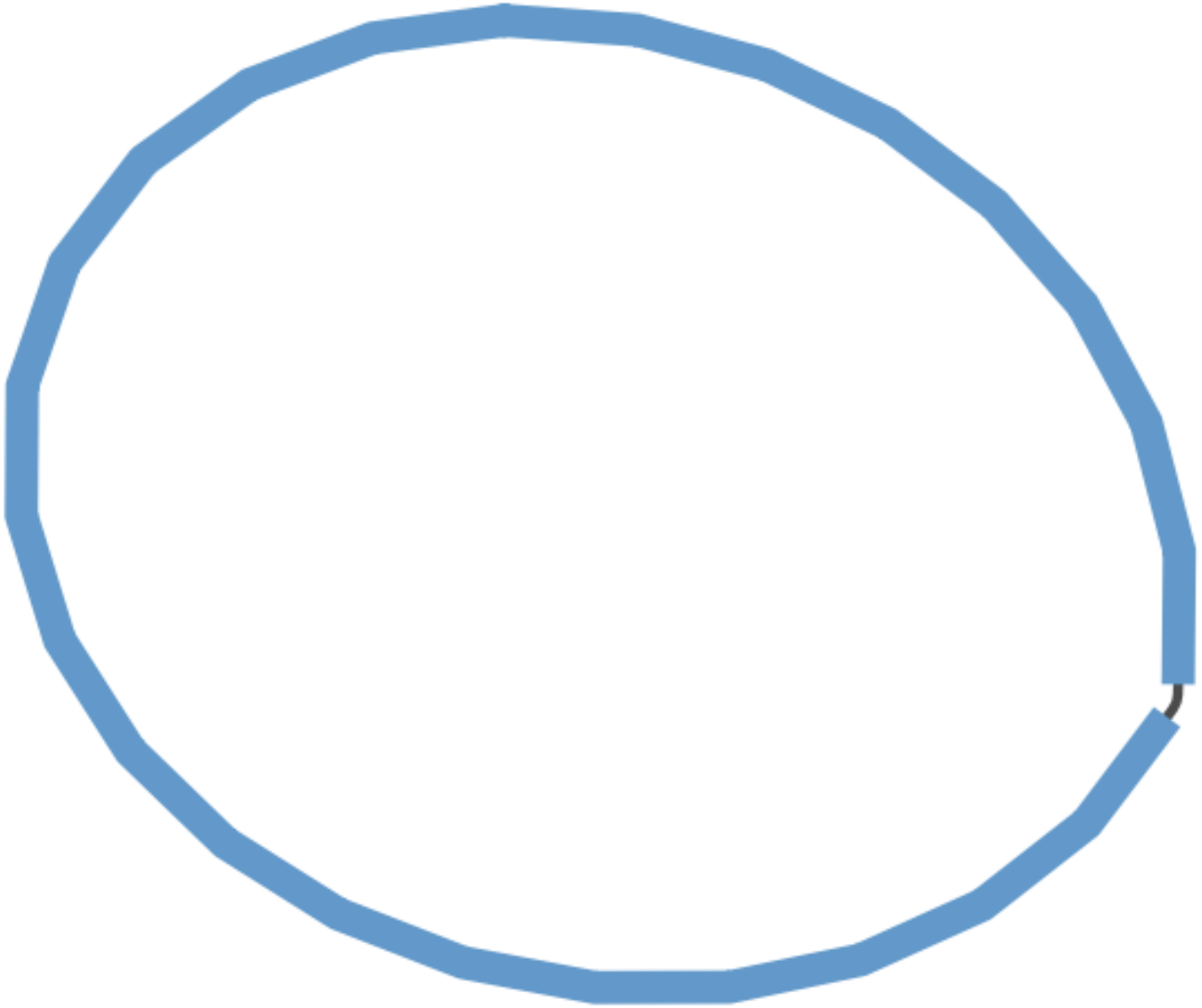
Lots of errors in reads, but (very) gradually improving

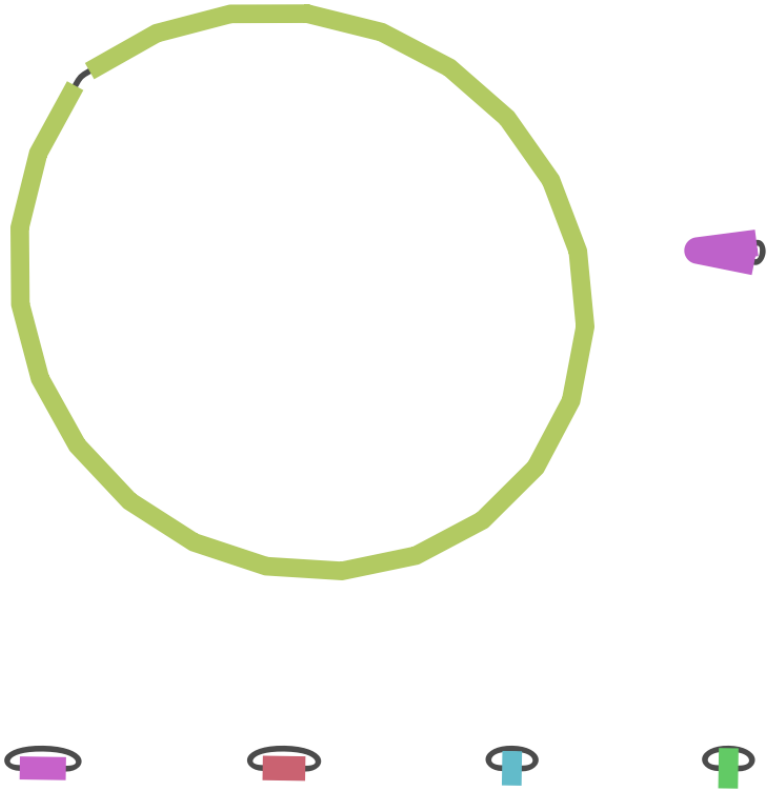
- Competition is good (for use!)
- PacBio HiFi is one way to improve this, but requires re-reading the fragment
- Upside: Assemblies with PacBio HiFi are *very fast!*
- Oxford Nanopore may be catching up (e.g. R10, improved yield in late 2020)

Can perform a long-read only assembly with very high coverage data (50-100x)

- Requires very high coverage, minimum 50x, with 80x preferred
- **Expensive** for large genomes (but getting competitive)
- ***Issues with homopolymers (Oxford)***
- …or hybrid assembly, with shorter Illumina reads used to error-correct the long-read assembly (Canu)







Assessing your assembly

How good is my assembly?

How much total sequence is in the assembly relative to estimated genome size?

How many pieces, and what is their size distribution?

Are the contigs assembled correctly?

Are the scaffolds connected in the right order / orientation?

How were the repeats handled?

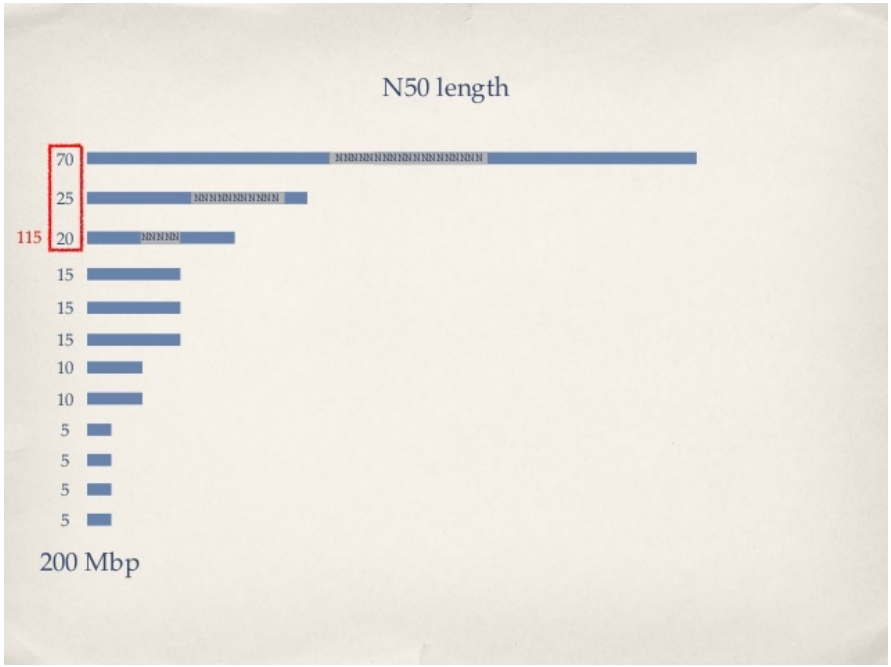
Are all the genes I expected in the assembly?

N50: the most common measure of assembly quality

N50 = length of the shortest contig in a set making up 50% of the total assembly length (***Larger is better***)

NG50 = length of the shortest contig in a set making up 50% of the **estimated genome size**

NG50 is generally better



N50 concerns

Optimizing for N50

- Encourages mis-assemblies!
- Encourages ‘gaming’ the stats

An aggressive assembler may over-join:

- 1,1,3,5,**8,12**,20 (previous)
- 1,1,3,5,**20**,20 (now)
- 1+1+3+5+20+20 = 50 (unchanged)

N50 is the “halfway sum” (still 25)

- 1+1+3+5+20= 30 (≥ 25) so **N50 is 20** (was 12)

You can also filter contigs below a certain (arbitrary) size, which lowers overall assembly size (and increases N50)

Comparative analysis

Compare against

- A close reference genome
- Results from another assembler
- Self-comparison
- Versions of the same assembly

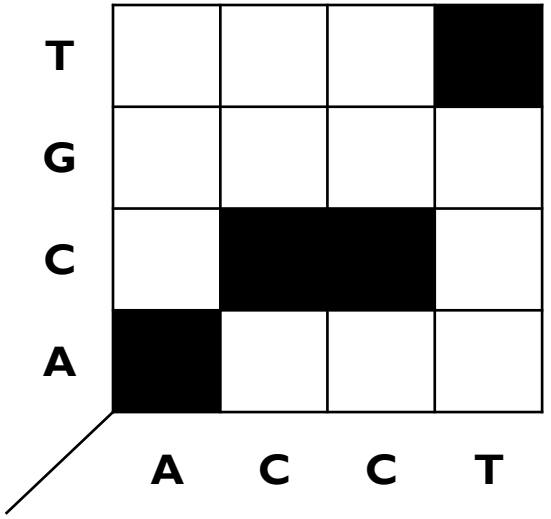
Whole genome alignment

- ***MUMmer***
- ***Lastz***

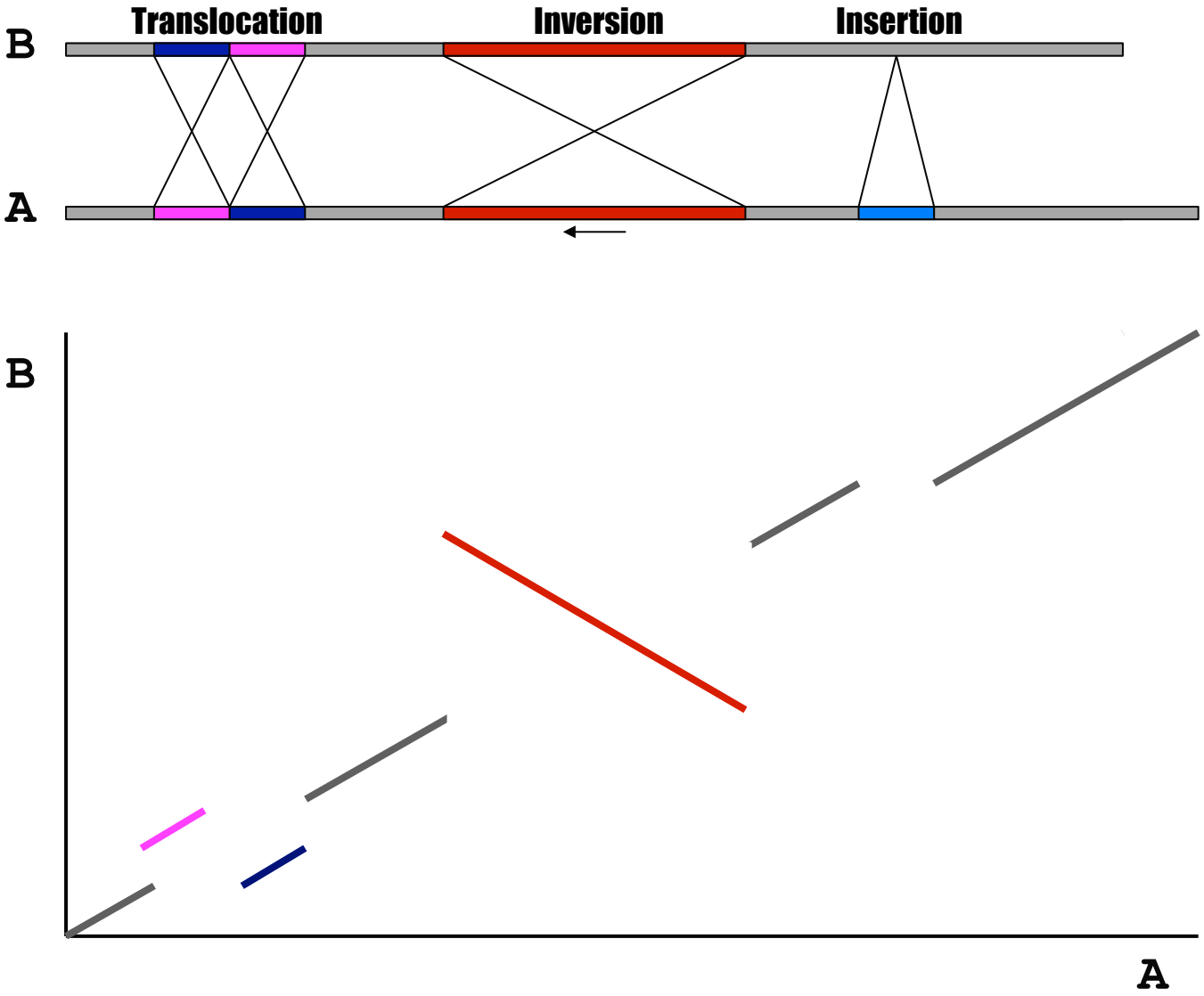
Generates an alignment and a ***dot plot***

Dot Plot

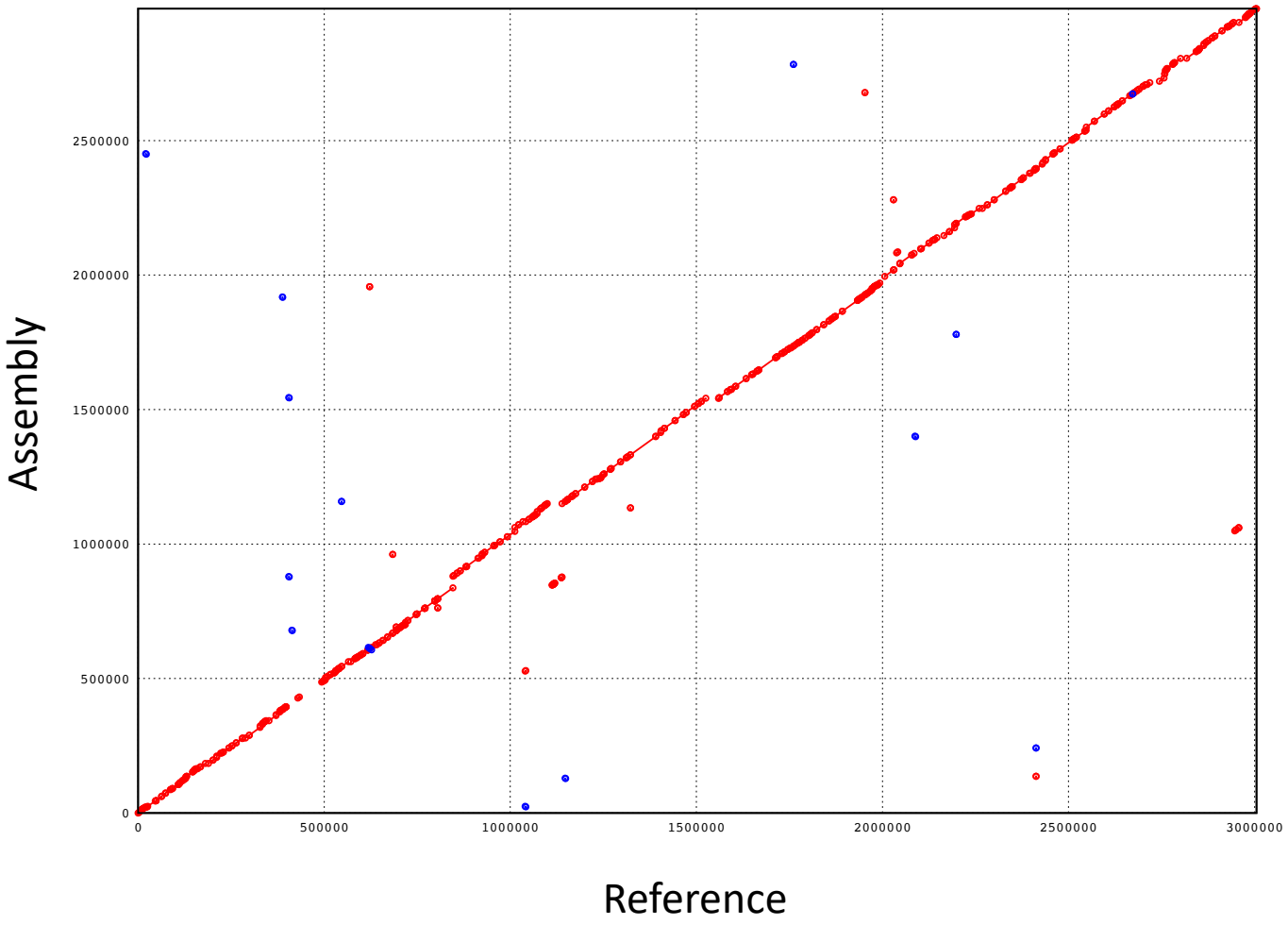
- How can we visualize *whole* genome alignments?
- With an alignment dot plot
  - *N* x *M* matrix
    - Let *i* = position in genome *A*
    - Let *j* = position in genome *B*
    - Fill cell *(i,j)* if A_i shows similarity to B_j

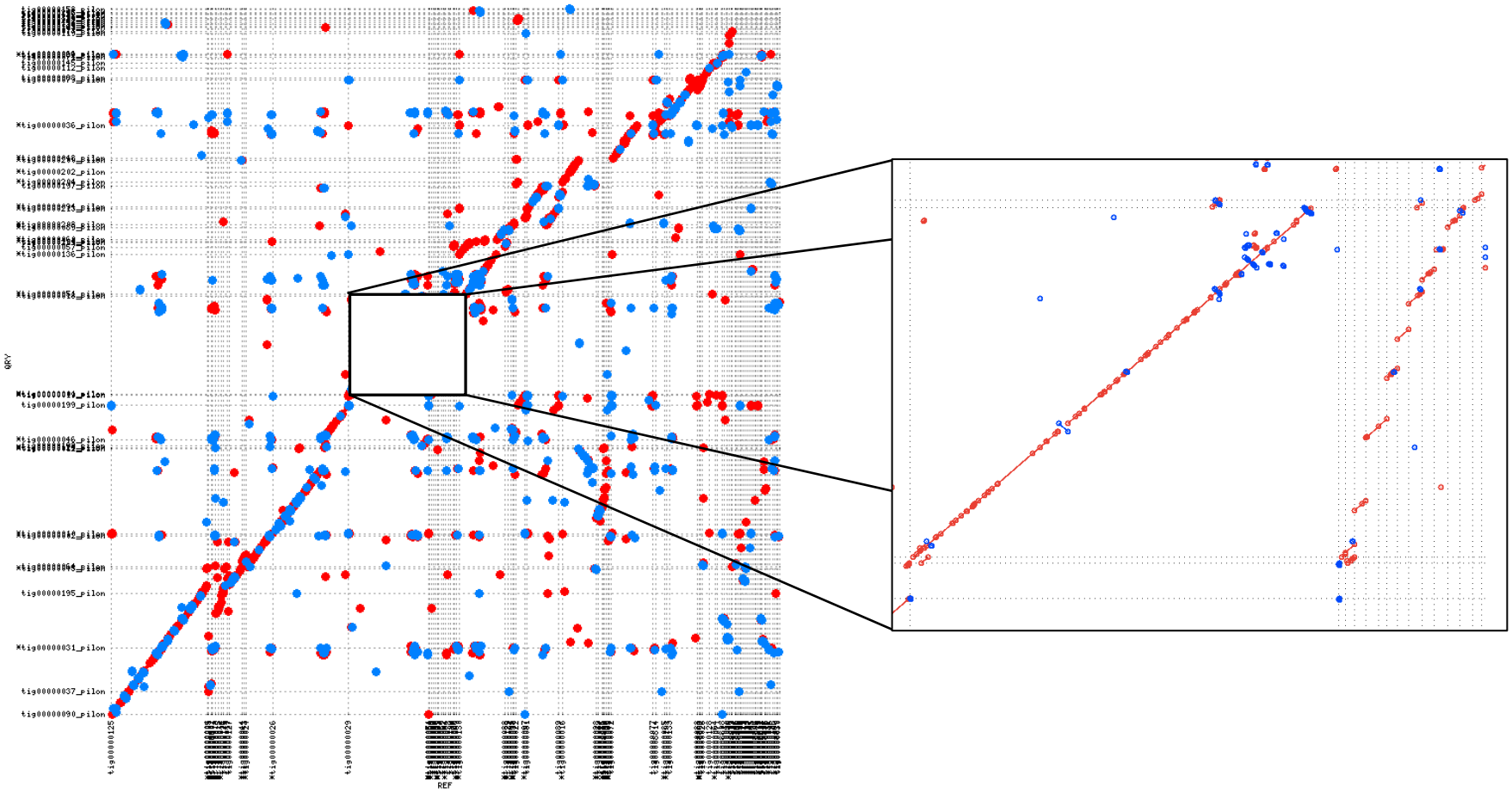


  - A perfect alignment between *A* and *B* would completely fill the positive diagonal



http://mummer.sourceforge.net/manual/AlignmentTypes.pdf


Assembly
Reference
0
500000
1000000
1500000
2000000
2500000
3000000

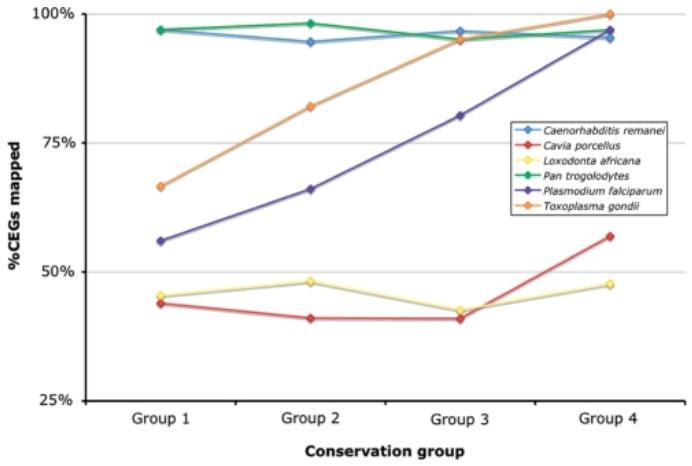

QRY
REF

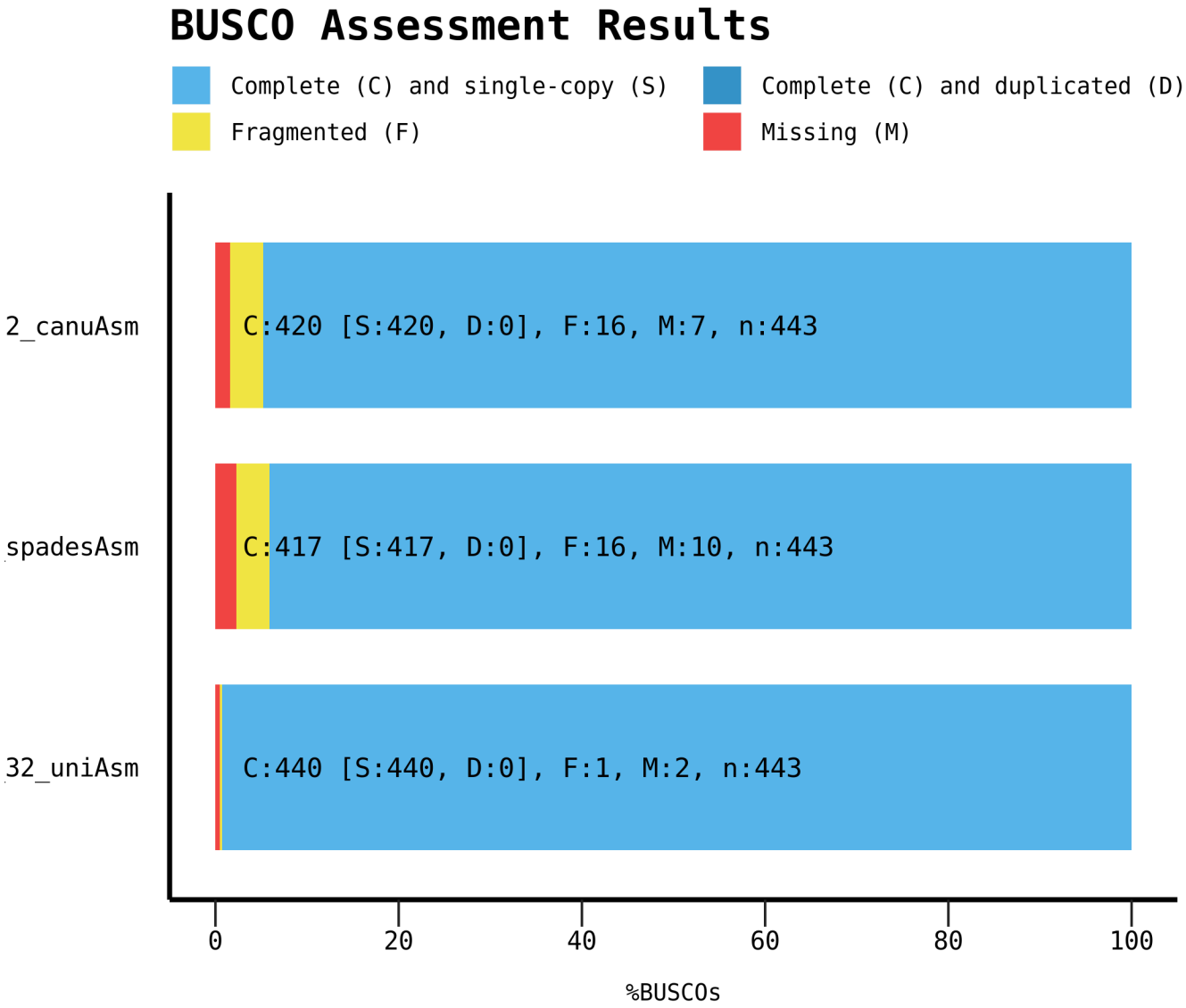
BUSCO and CEGMA: conserved gene sets

BUSCO: From Evgeny Zdobnov's group, University of Geneva

CEGMA : From Ian Korf's group, UC Davis Mapping Core Eukaryotic Genes

Coverage is indicative of quality and completeness of assembly




BUSCO Assessment Results
Complete (C) and single-copy (S)
Complete (C) and duplicated (D)
Fragmented (F)
Missing (M)
2_canuAsm
C:420 [S:420, D:0], F:16, M:7, n:443
spadesAsm
C:417 [S:417, D:0], F:16, M:10, n:443
32_uniAsm
C:440 [S:440, D:0], F:1, M:2, n:443
0
20
40
60
80
100
%BUSCOs

QUAST

QUality ASsessment Tool

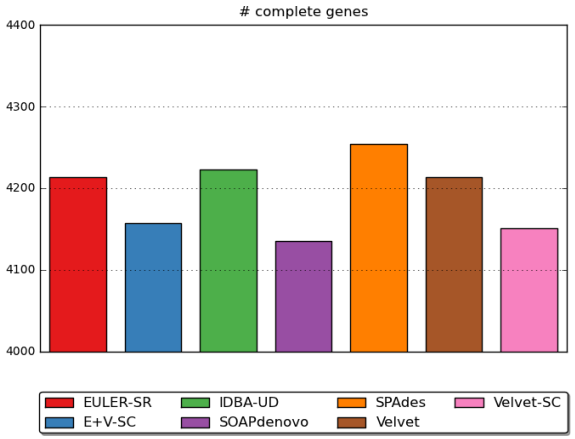
- Small (bacterial, fungal) and large (eukaryotic) genomes
- Metagenomes
- Icarus for contig alignment visualization

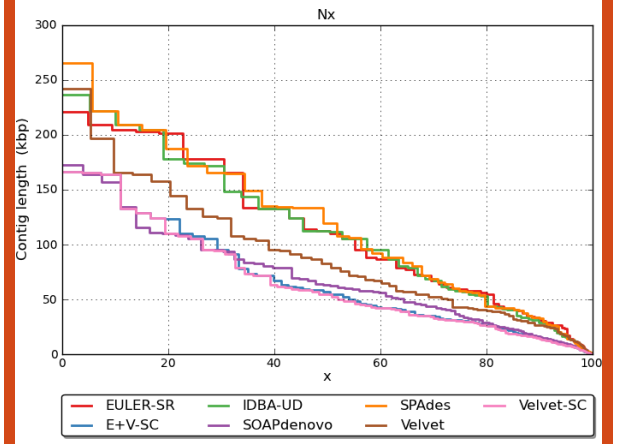
Can compare multiple assemblies against one another

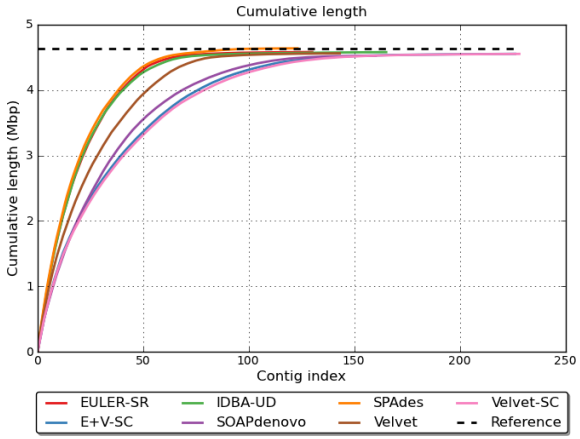
Compare against a known (or close) reference

Optional: Predict genes or include annotations (checks for odd issues like frameshifts)

Generates a summary HTML report


complete genes
4400
4300
4200
4100
4000
EULER-SR
E+V-SC
IDBA-UD
SOAPdenovo
SPAdes
Velvet
Velvet-SC


Nx
Contig length (kbp)
300
250
200
150
100
50
0
0
20
40
60
80
100
x
EULER-SR
E+V-SC
IDBA-UD
SOAPdenovo
SPAdes
Velvet
Velvet-SC


Cumulative length
Cumulative length (Mbp)
5
4
3
2
1
0
0
50
100
150
200
250
Contig index
EULER-SR
E+V-SC
IDBA-UD
SOAPdenovo
SPAdes
Velvet
Velvet-SC
Reference

QUAST

Even the best genomes are not perfect

Caenorhabditis elegans

- Genome published 1998
- 2004: last N removed
- 1998–2013: genome sequence changes
  - 558 insertions
  - 230 deletions
  - 614 substitutions

Assembler comparisons

Assessing assembly strategies

Assemblathon (UC Davis and UC Santa Cruz)

- Provide challenging datasets to assemble in open competition (synthetic for edition 1, real for edition 2)
- Assess competitor assemblies by many different metrics
- Publish extensive reports

GAGE (U. of Maryland and Johns Hopkins)

- Select datasets associated with known high-quality genomes
- Run a set of open source assemblers with parameter sweeps on these datasets
- Compare the results, publish in scholarly Journals with complete documentation of parameters

There is no such thing as a “perfect” assembler (results from GAGE competition)

Table 2. Assemblies of *S. aureus* (genome size 2,872,915)

	Contigs				Scaffolds			
Assembler	Num	N50 (kb)	Errors	N50 corr. (kb)	Num	N50 (kb)	Errors	N50 corr. (kb)
ABySS	302	29.2	14	24.8	246	34	1	28
ALLPATHS-LG	**60**	96.7	16	**66.2**	**12**	1,092	**0**	**1,092**
Bambus2	109	50.2	177	16.7	17	1,084	**0**	1,084
CABOG	Could not run: incompatible read lengths in one library							
MSR-CA	94	59.2	22	48.2	17	**2,412**	3	1,022
SGA	252	4.0	**6**	4.0	456	208	1	208
SOAPdenovo	107	**288.2**	48	62.7	99	332	8	284
Velvet	162	48.4	28	41.5	45	762	17	126

Table 3. Assemblies of *R. sphaeroides* (genome size 4,603,060)

	Contigs				Scaffolds			
Assembler	Num	N50 (kb)	Errors	N50 corr. (kb)	Num	N50 (kb)	Errors	N50 corr. (kb)
ABySS	1915	5.9	55	4.2	1701	9	3	5
ALLPATHS-LG	204	42.5	43	**34.4**	**34**	**3192**	**0**	**3192**
Bambus2	**177**	93.2	368	12.8	92	2439	2	2419
CABOG	322	20.2	34	17.9	130	66	5	55
MSR-CA	395	22.1	42	19.1	43	2,976	5	2966
SGA	3067	4.5	**8**	2.9	2096	51	**0**	51
SOAPdenovo	204	**131.7**	414	14.3	166	660	3	658
Velvet	583	15.7	35	14.5	178	353	6	270

Columns are the same as in Table 2.

The computational demands and effectiveness of assemblers are very different

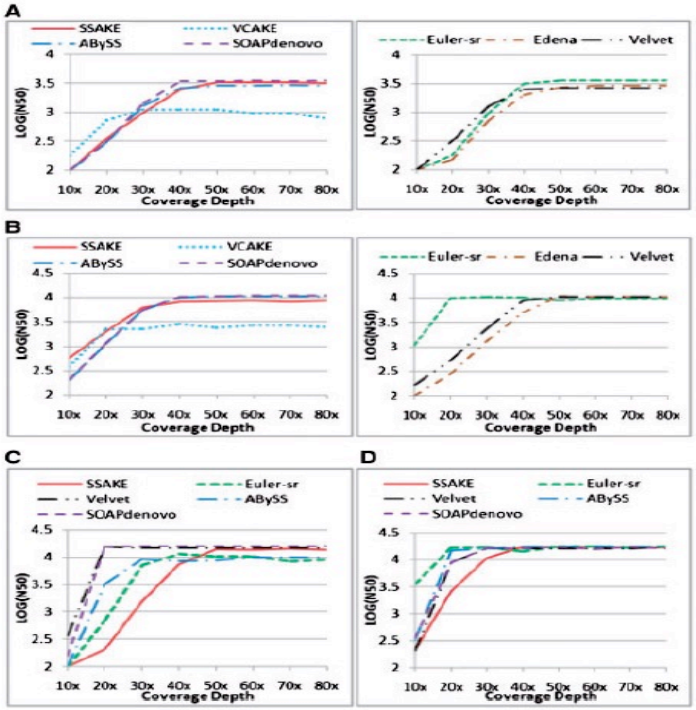


Fig. 1. Comparison of the effect of various coverage depths on N50 length in *T.bru* assembly when BCER was 0.6%. **(A)** Single-end reads assembly, read length (RL) = 35 bp; **(B)** single-end assembly, RL = 75 bp; **(C)** paired-end reads assembly, RL = 35 bp; **(D)** paired-end assembly, RL = 75 bp.

Table 6. Comparison of runtime and RAM in the computational demand test

	Bench.Seq (length: bp)	*E.coli* (4.6M)	*C.ele* (20.9M)	*H.sap*-2 (50.3M)	*H.sap*-3 (100.5M)
Runtime (s)					
SE	SSAKE	2776	–	–	–
	VCAKE	1672	16 742	–	–
	Euler-sr	1689	11 961	29 622	–
	Edena	895	8450	17 043	–
	Velvet	205	1003	2786	6098
	ABySS	265	1300	3307	6608
	SOAPdenovo	62	253	560	1029
PE	SSAKE	9163	–	–	–
	Euler-sr	1455	15 068	–	–
	Velvet	229	1351	55 581	–
	ABySS	458	3081	9199	21 683
	SOAPdenovo	78	374	889	2257
RAM (MB)					
SE	SSAKE	9933	–	–	–
	VCAKE	4099	17 408	–	–
	Euler-sr	1536	7065	13 312	–
	Edena	1741	7557	30 720	–
	Velvet	1229	4045	9830	22 528
	ABySS	1126	3993	8909	18 432
	SOAPdenovo	935	2867	8089	18 227
PE	SSAKE	16 384	–	–	–
	Euler-sr	1638	7578	–	–
	Velvet	1331	5324	30 720	–
	ABySS	950	4505	9830	18 432
	SOAPdenovo	1638	5939	10 342	19 456

Bench.Seq, benchmark sequence; SE, single-end reads assembly; PE, paired-end reads assembly. '–' denotes the RAM of computer is not enough or runtime is too long (>10 days) to get assembly results.

Some advice on running assemblies

Perform parameter sweeps

- Use many different values of key parameters, especially k-mer size for DBG assemblers, and evaluate the output (some assemblers can do this automatically)

Try different subsets of the data

- Sometimes libraries are of poor quality and degrade the quality of the assembly
- Artefacts in the data (e.g. PCR duplicates, homopolymer runs, …) can also badly affect output quality

Try more than one assembler

- There is no such thing as “the best” assembler

Genome Graphs

Genome graphs

With the release of the latest human genome reference, there is more pressure to represent more data with a genome.

Current representations are **haploid** (one copy)

Newer representations are ***genome graphs***, where variant information is retained (e.g. heterozygosity)

Tools are still catching up

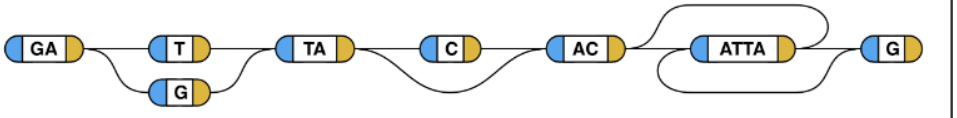
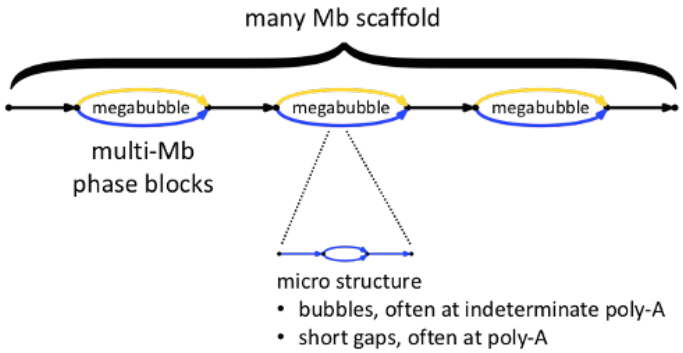
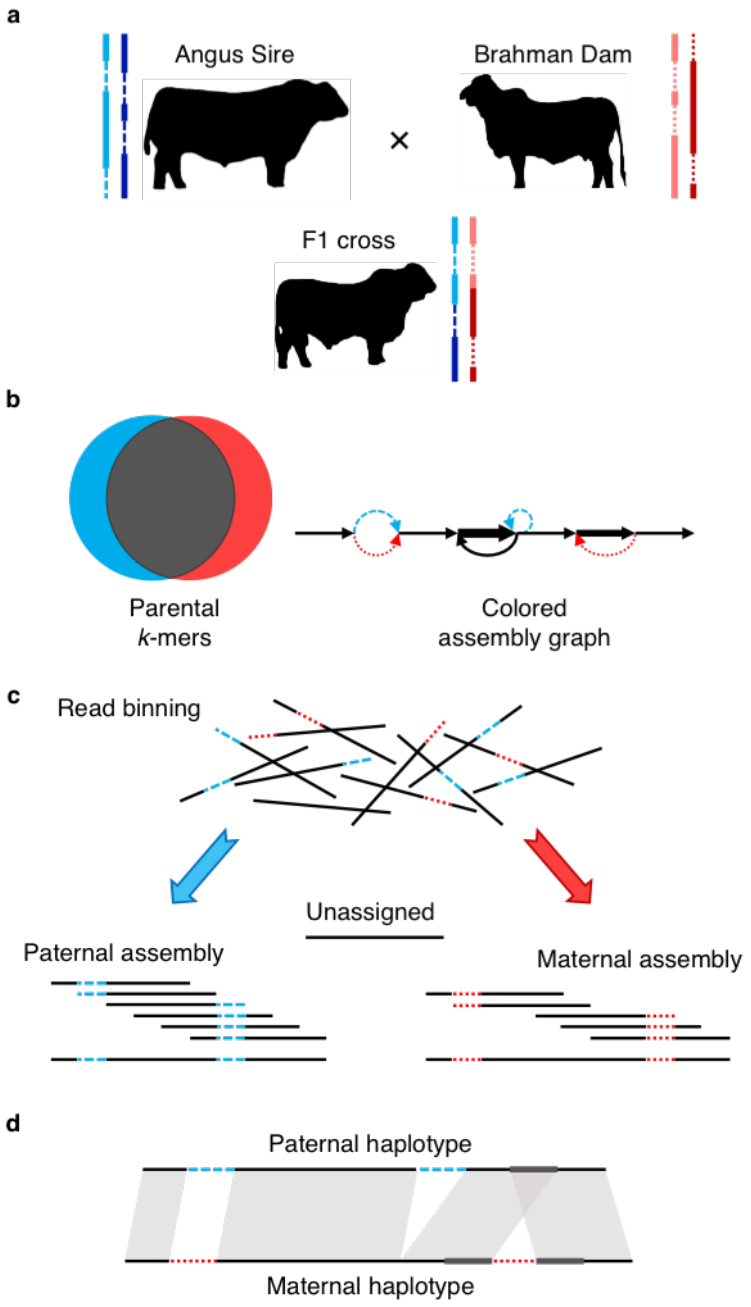


Figure 1: Example sequence graphs. *Each node holds a string of bases. An edge can connect, at each of its ends, to a base on either the left (5', blue) or the right (3', yellow) side of the node. When reading through a thread to form a DNA sequence, a valid walk must leave each node via the opposite side from that through which it was entered; a node's sequence is read as reverse-complemented if the node is entered on the 3' side. One thread that this graph spells out (reading from the left side of the leftmost sequence to the right side of the rightmost sequence, along the nodes drawn in the middle) is the sequence "GATTACACATTAG". Straying from this path, there are three variants available: a substitution of "G" for "T", a deletion of a "C", and an inversion of "ATTA". If all of these detours are taken, the sequence produced is "GAGTAACTAATG". All 8 possible threads from the leading G to the trailing G are allowed.*





Genome graphs

One interesting application of graphs: ***trio assembly***

Short- and long-read sequencing

Advantage:

- *Better assembly*
- *Phased variants*
- *Structural variants!*

Koren *et al*, Nature Biotech. 2018, Oct 22

Genome Annotation

Methods for genome annotation

Ab initio, i.e. based on sequence alone

- INFERNAL/rFAM (RNA genes), miRBase (miRNAs), RepeatMasker (repeat families), many gene prediction algorithms (e.g. AUGUSTUS, Glimmer, GeneMark, …)

Evidence-based

- Require transcriptome data for the target organism (the more the better)
- Align cDNA sequences to assembled genome and generate gene models: TopHat/Cufflinks, Scripture

Methods for biological annotation

BLAST of gene models against protein databases

- Sequence similarity to known proteins

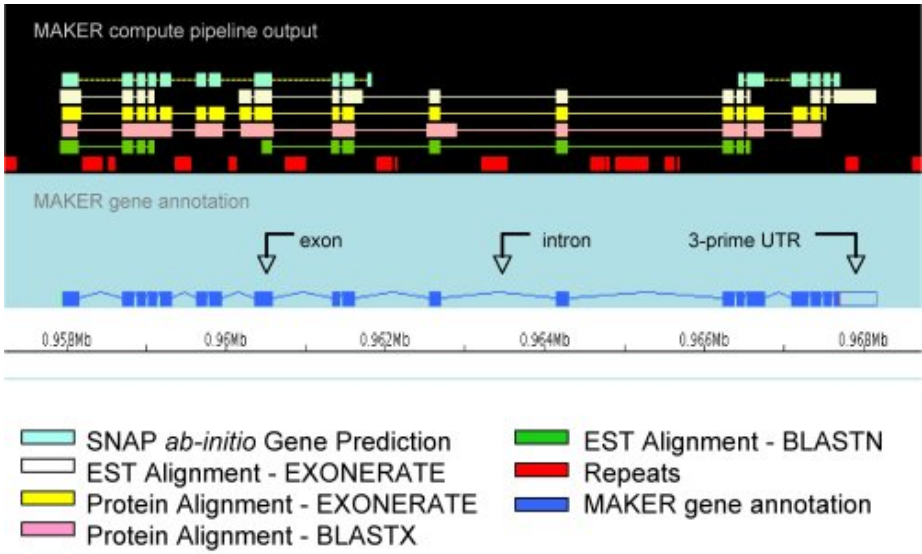
InterProScan of predicted proteins against databases of protein domains (Pfam, Prosite, HAMAP, PANTHER, …)

Mapping against Gene Ontology terms (BLAST2GO)

MAKER, integration framework for genome annotation

MAKER runs many software tools on the assembled genome and collates the outputs

See http://gmod.org/wiki/MAKER



Others?

Prokka (bacterial/archaeal/viral)

Joint Genome Institute – IMG/ER (Integrated Microbial Genome Expert Review)

NCBI – RefSeq pipeline

Acknowledgements

Materials from this slide deck include figures and slides from many publications, Web pages and presentations by:

- M. Schatz, A. Phillipy, T. Seemann, S. Salzberg, K. Bradnam, D. Zerbino, M. Pop, G. Sutton, Nick Loman, Carson Holt, Ryan Wick.
- I highly recommend Ben Langmead's teaching materials; he has a ton fabulous (and much more in-depth) notes on his lab page: http://www.langmead-lab.org/teaching-materials/
- Thank you!