UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

INTRODUCTION TO *DE NOVO* ASSEMBLY OF GENOMES WITH SHORT AND/OR LONG READS

June 2nd 2025

Laurent Falquet
MA/MER @ UniFr
Group Leader @ SIB

Swiss Institute of Bioinformatics

What is de novo genome assembly?

DNA sequencing

De novo genome assembly

What is Next Generation Sequencing?

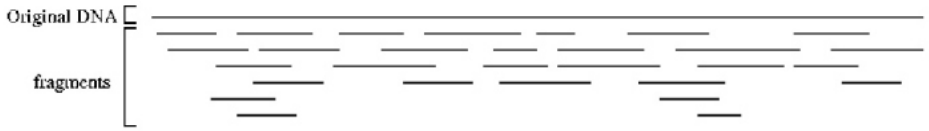


Figure 1. Original genomic DNA is broken into a collection of overlapping fragments

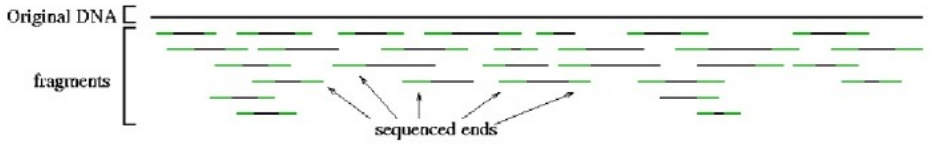


Figure 2. The ends of each fragment (drawn in green) are sequenced

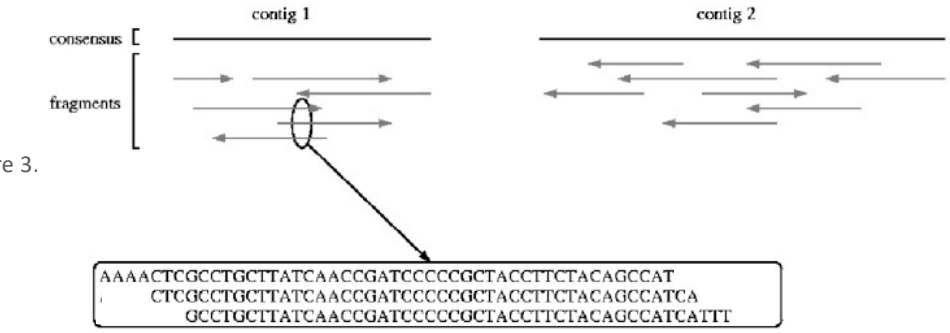


Figure 3.

This is also called "Whole Genome Shotgun Sequencing"

Whole Genome Sequencing and Genome Assembly:
Must be like a Simple Jigsaw Puzzle?

Yes, but you must deal with

- **Millions of pieces**
- **Lots of malformed pieces**
- **Often missing pieces**
- **Pieces mixed from another puzzle**
- **Lots of identical blue sky pieces…**

If *de novo* you…

Genome assembly, deep blue…

…don't even know the final picture…

Reminder of the sequencing methods

1986 - 2001 | 2005 | 2006 | 2010 | 2011 | 2014 | 2018

ABI Sanger | Roche 454 | Illumina | Life SOLiD | Life Ion Torrent | Pacific Biosciences | Oxford Nanopore

First Generation

Second Generation

Third Generation

Reminder of the sequencing methods

Limitations of the NGS techniques

All methods

- **sequencing errors (substitutions or insertions/deletions)**
- **missing data (sampling/coverage bias)**

Roche454, IonTorrent

- **long (>12) mononucleotide repeats**
- **short reads (200bp) for Ion Torrent**

Illumina, MGI

- **short reads (<300bp)**

Pacific Bioscience, Oxford Nanopore

- **high error rate (~15% indels) for long reads (raw data)**
- **Nanopore improves its chemistry regularly (<2% error rate)**
- **Good news: PacBio HiFi reads have much better quality.**

Limitations of the sequences

Repeats

transposases, IS-elements, retroviruses, duplications, etc.

Polymorphisms

SNPs, CNV, multiploid, sample mixture, etc.

Sequence bias

%GC

Repeats are a major issue for all assemblers

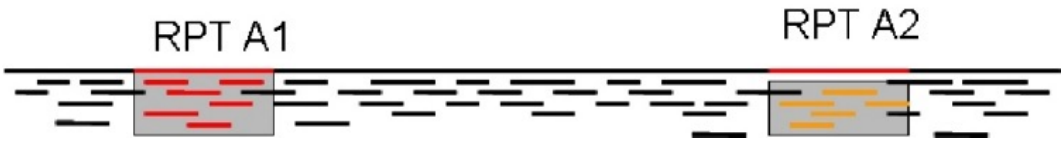


Figure top. Two copies of a repeat along a genome. The reads colored in red and those colored in yellow appear identical to the assembly program.

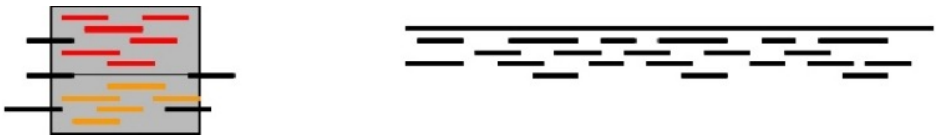
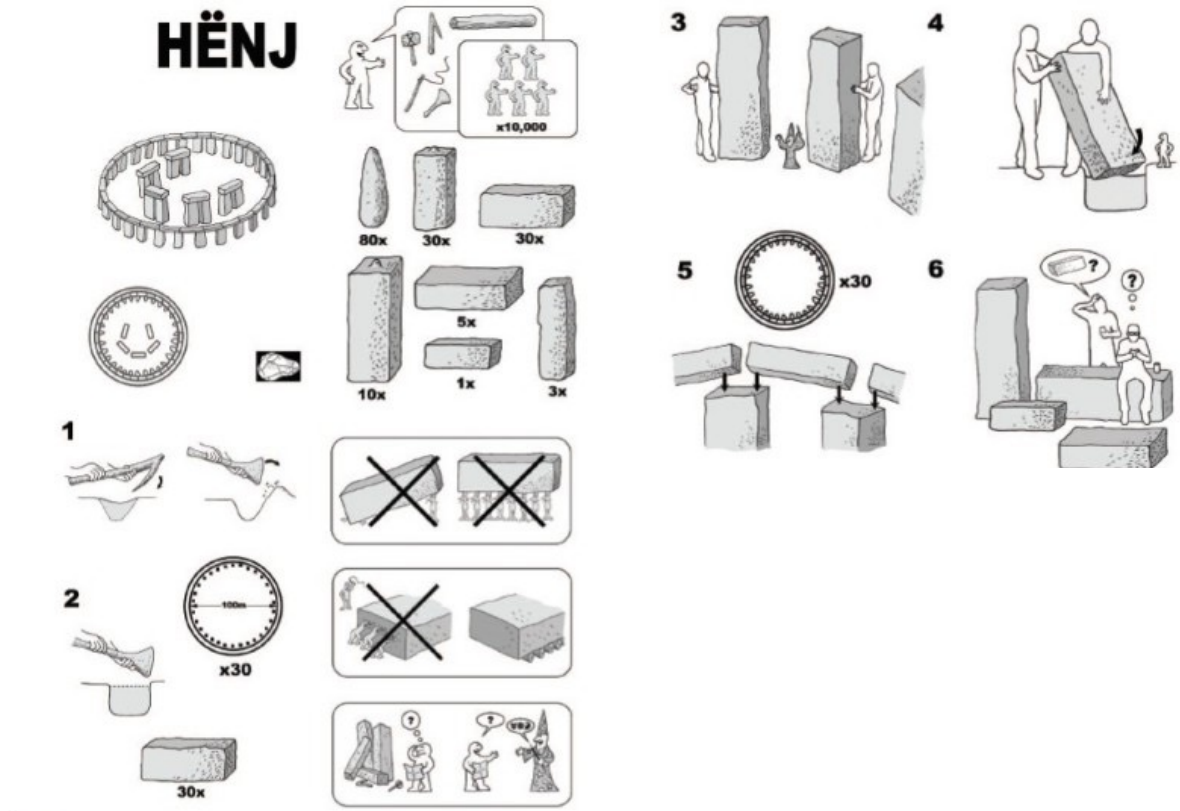


Figure bottom. Genome mis-assembled due to a repeat. The assembly program incorrectly combined the reads from the two copies of the repeat leading to the creation of two separate contigs.

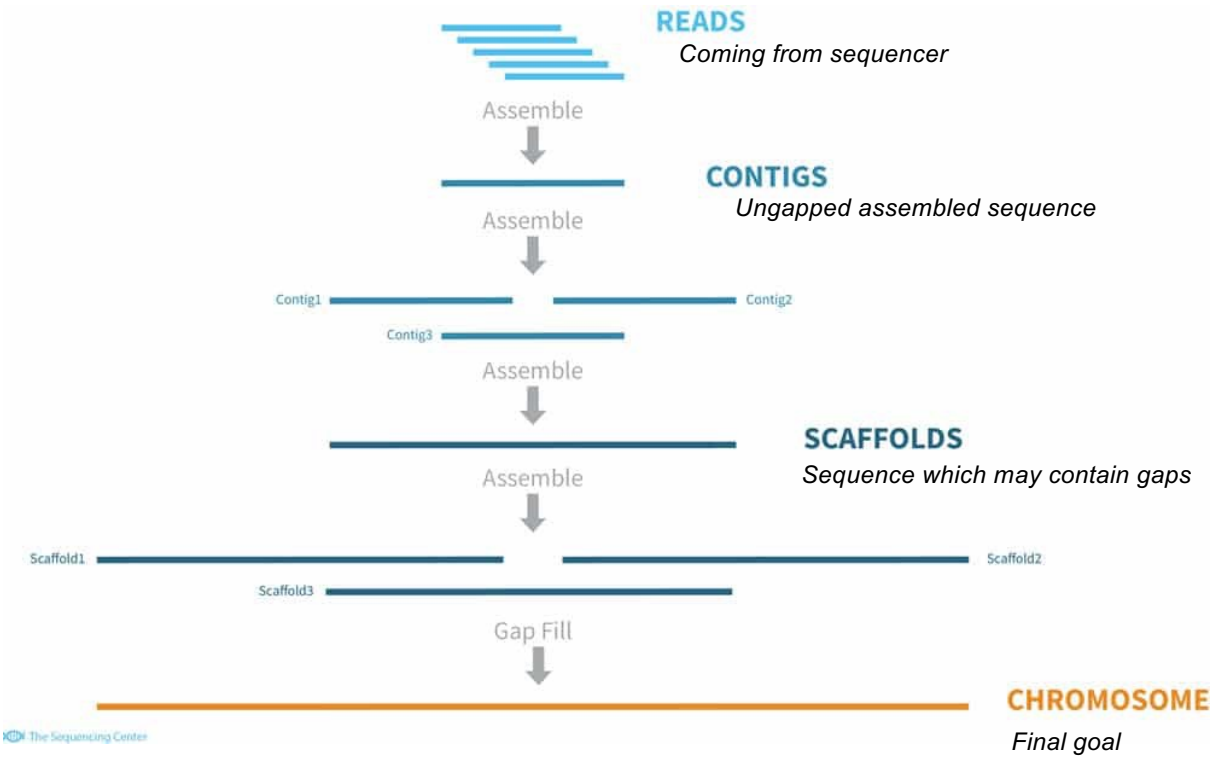
HËNJ GENOME IKEA ASSEMBLY INSTRUCTIONS



U

58

Overview of a sequencing project



Overview of a sequencing project

Select DNA and sequencing method

Quality Control

if necessary Trimming/Filtering (done, see practicals)

De novo assembly with cleaned data

1st step: Assembly

consists mainly in building contigs from short reads

2nd step: Scaffolding (optional)

where contigs are ordered and oriented

3rd step: Finishing or Closing (optional)

where gaps are closed

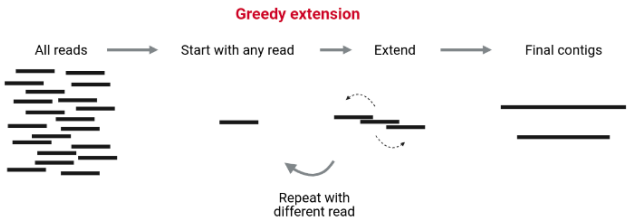
Genome annotation (draft or final)

Submission to databases & publication (not discussed here)

Genome assembly software: 3 main algorithmic classes

1) Greedy and other (mixture)

Not recommended

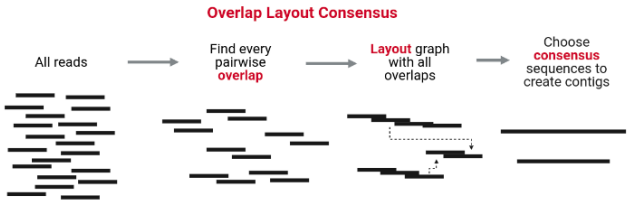


2) Overlap Layout Consensus (OLC)

or string graph assemblers

Canu, miniasm, hifiasm, Flye…

ideal for long reads

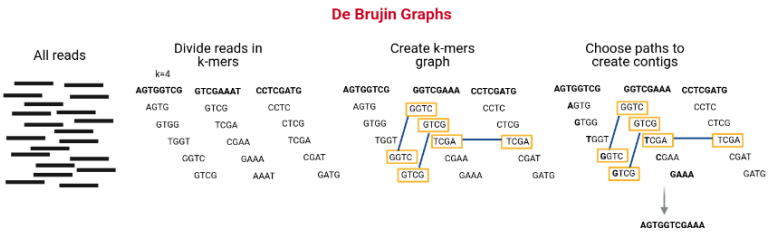


3) deBruijn graph assemblers

ABySS, SOAP-denovo,

(meta)SPAdes, MEGAHIT ...

ideal for short reads

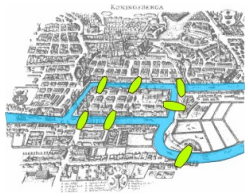


https://zenodo.org/records/7713386

Leonhard Euler
1707 - 1783

Swiss mathematician
Euler's identity, the most famous formula!

$$e^{i\pi} + 1 = 0$$



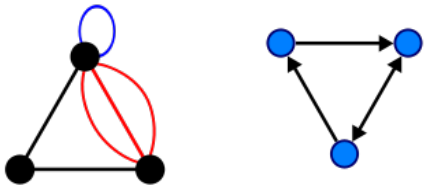
Graph theory

In 1736, Euler solved the problem known as the Seven Bridges of Königsberg. The city of Königsberg, Prussia was set on the Pregel River, and included two large islands which were connected to each other and the mainland by seven bridges.

The problem is to decide whether it is possible to follow a path that crosses each bridge exactly once and returns to the starting point. It is not: there is no Eulerian circuit.

This solution is considered to be the first theorem of graph theory, specifically of planar graph theory.

Graph theory



A graph refers to a collection of **vertices** (or '**nodes**') and a collection of **edges** (or '**vectors**') that connect pairs of vertices.

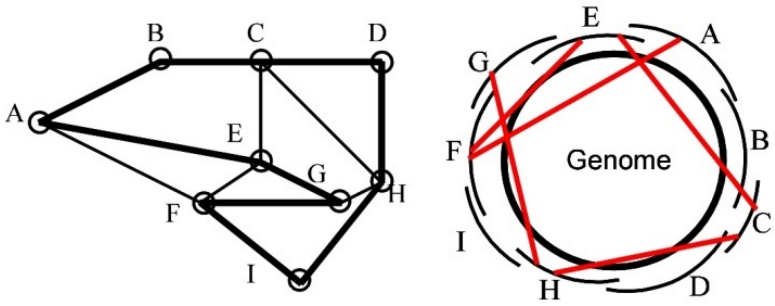
A graph may be undirected, meaning that there is no distinction between the two vertices associated with each edge, or its edges may be directed from one vertex to another (digraph).

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

Overlap-layout-consensus

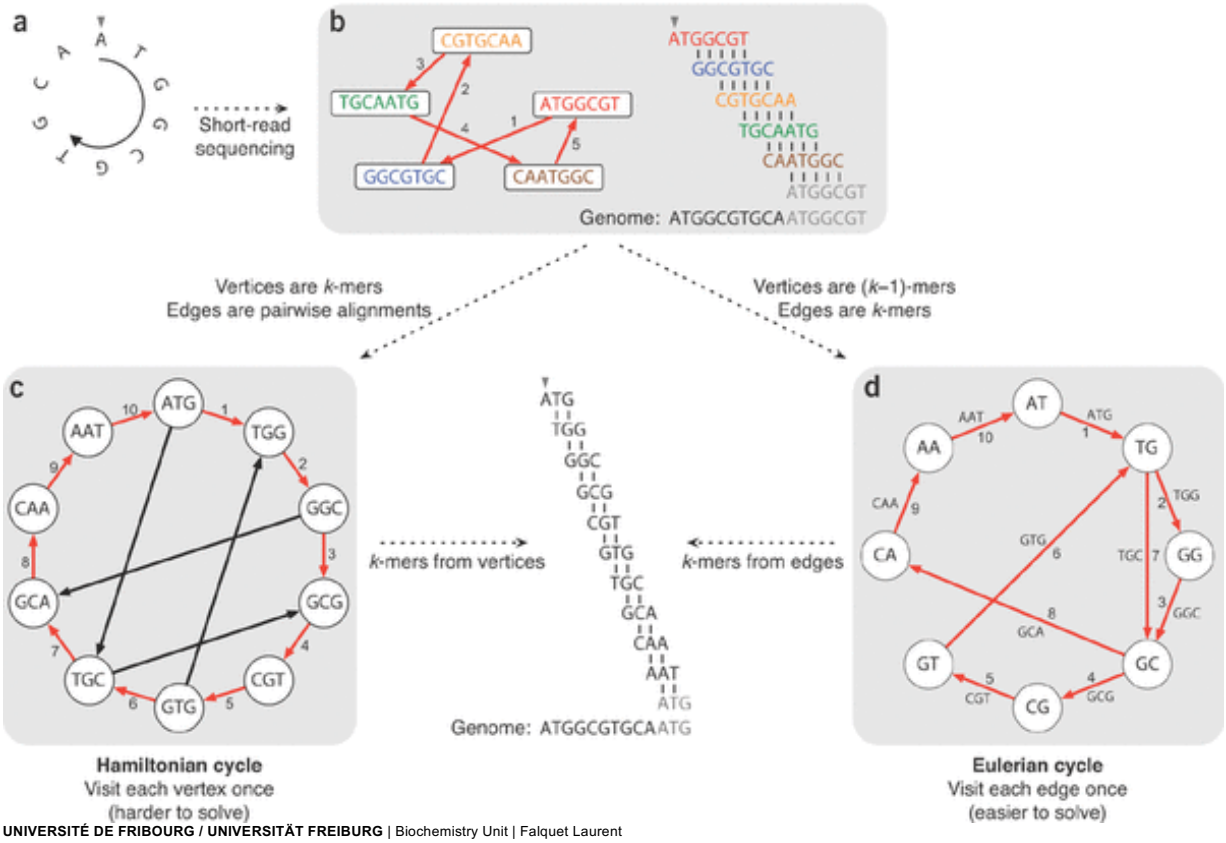
Overlap-layout-consensus - The relationships between the reads provided to an assembler can be represented as a graph, where the nodes represent each of the reads and an edge connects two nodes if the corresponding reads overlap. The assembly problem thus becomes the problem of identifying a path through the graph that contains all the nodes - **a Hamiltonian path (Figure below).** This formulation allows researchers to use techniques developed in the field of graph theory in order to solve the assembly problem.

An assembler following this paradigm starts with an **overlap stage** during which all overlaps between the reads are computed and the graph structure is computed. In a **layout stage**, the graph is simplified by removing redundant information. Graph algorithms are then used to determine a layout (relative placement) of the reads along the genome. In a final **consensus stage**, the assembler builds an alignment of all the reads covering the genome and infers, as a consensus of the aligned reads, the original sequence of the genome being assembled.

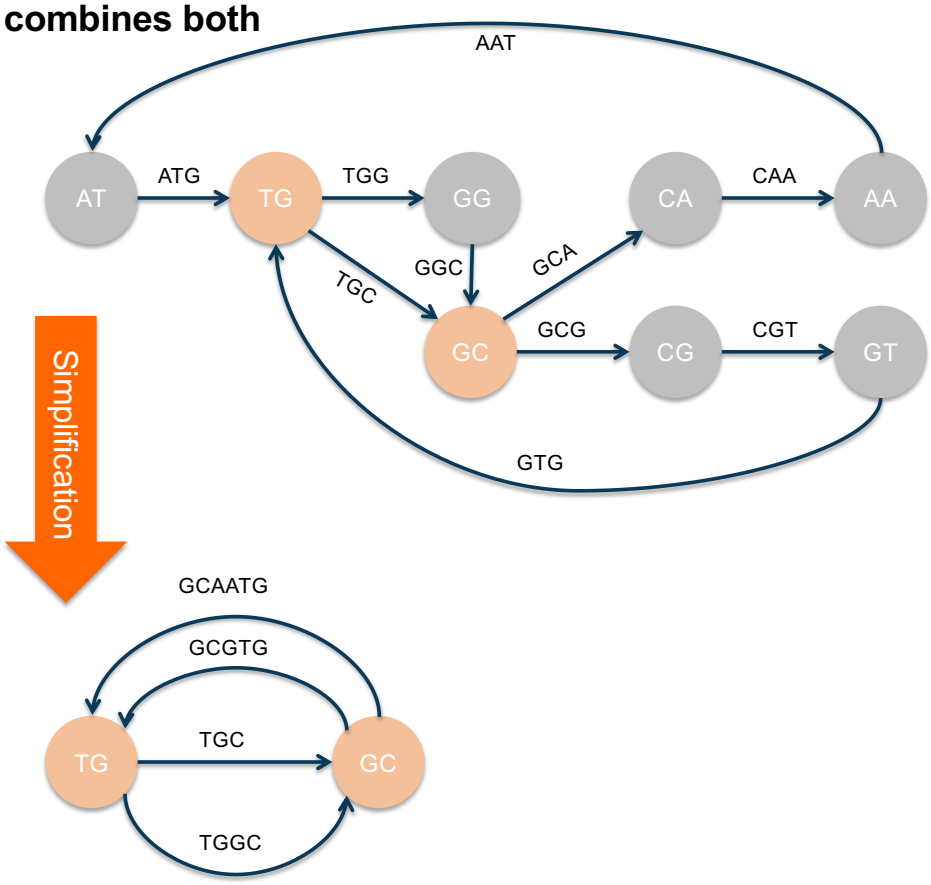


Overlap graph for a bacterial genome. The thick edges in the picture on the left (a Hamiltonian cycle) correspond to the correct layout of the reads along the genome (figure on the right). The remaining edges represent false overlaps induced by repeats (exemplified by the red lines)

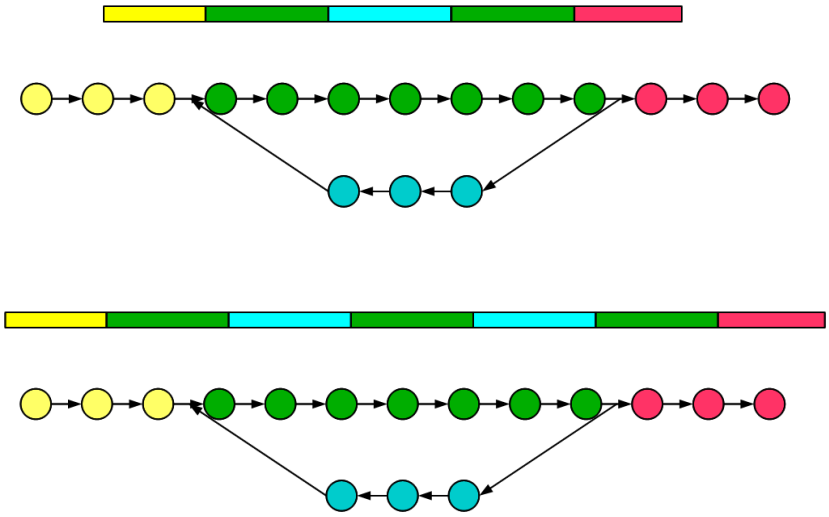
Eulerian *vs* Hamiltonian path/cycle ?



De Bruijn graph combines both



What happens in case of repeats?



No difference in the graph!

EXERCISE

In this exercise, for simplicity, ignore reverse complements.
Reads:

```
TACAGT
  CAGTC
   AGTCAG
     TCAGA
```

1. Construct the de Bruijn graph for $k = 3$.

   Reminder: edges are k-mers and nodes correspond to k-1 overlaps

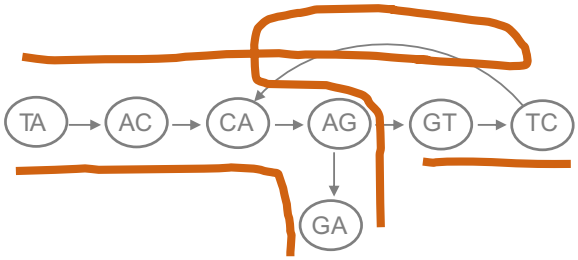
2. How many contigs can be created?
3. At which value of k is there a single contig?
4. (optional) Find a mathematical relationship between k_a, the smallest k value with which a genome can be assembled into a single contig (using a de Bruijn graph), and ℓ_r, the length of the longest exactly repeated region in that genome.

EXERCISE (SOLUTION)

In this exercise, for simplicity, ignore reverse complements.
Reads:

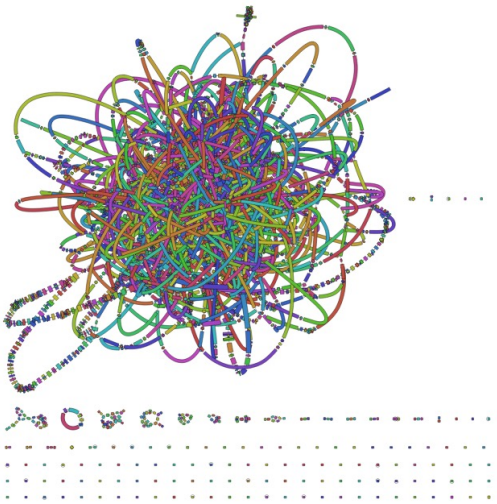
```
TACAGT
  CAGTC
   AGTCAG
     TCAGA
```

1. Construct the de Bruijn graph for $k = 3$.
   The 3-mers (edges) are: TAC, ACA, CAG, AGT, GTC, TCA, AGA

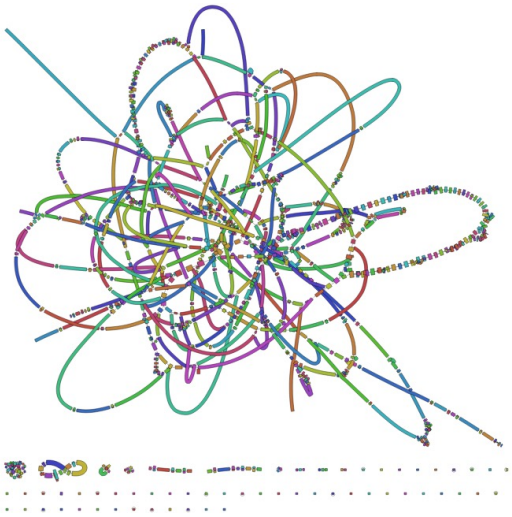


2. How many contigs can be created? 3
3. At which value of k is there a single contig? 4
4. Find a mathematical relationship between k_a, the smallest k value with which a genome can be assembled into a single contig (using a de Bruijn graph), and ℓ_r, the length of the longest exactly repeated region in that genome. $k_a = \ell_r + 1$

Comparison of assemblies with different kmers viewed with Bandage



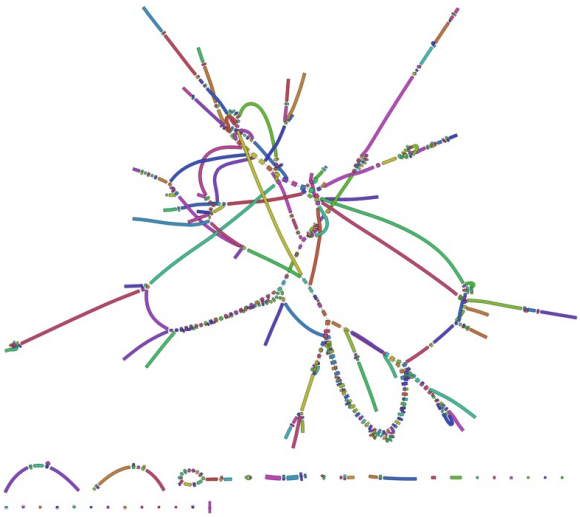
51-mer assembly
This k-mer size is too small, resulting in a complex and tangled graph with 4618 nodes and 6070 edges.



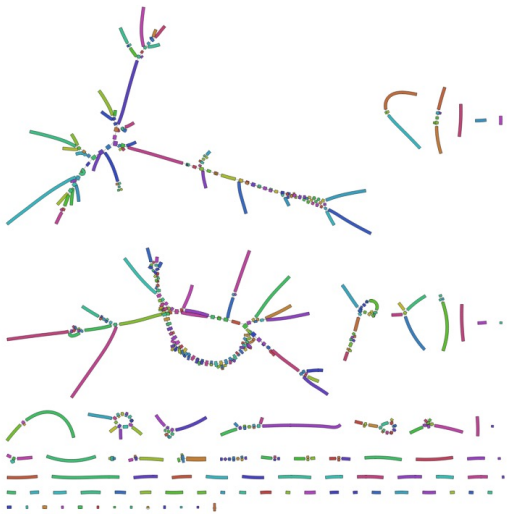
61-mer assembly
This graph is better than the 51-mer graph – it is much less complex (1357 nodes and 1768 edges) but still has very few dead ends.

Salmonella genome example see https://github.com/rrwick/Bandage/wiki/Effect-of-kmer-size

Comparison of assemblies with different kmers viewed with Bandage

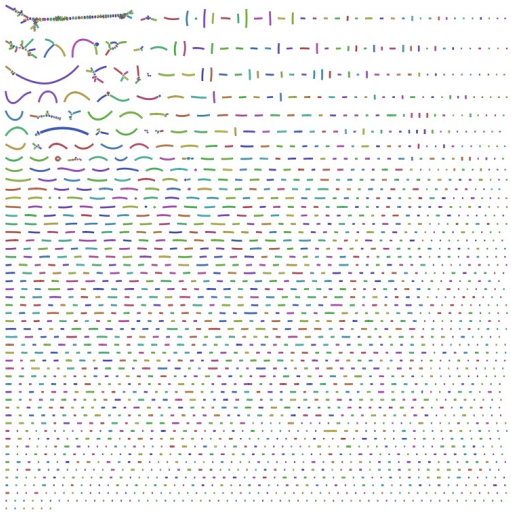


71-mer assembly
While the complexity of the graph has improved (it has 611 nodes and 765 edges), it now shows many more dead ends.



81-mer assembly
As compared to the 71-mer graph, the complexity has slightly improved (it has 490 nodes and 512 edges), but it has broken into many disconnected parts.

Comparison of assemblies with different kmers viewed with Bandage

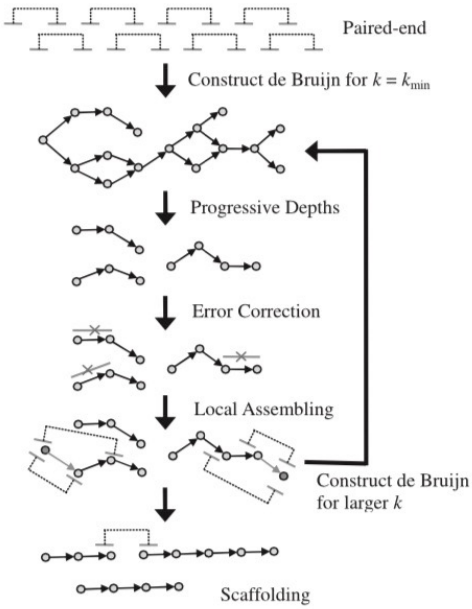


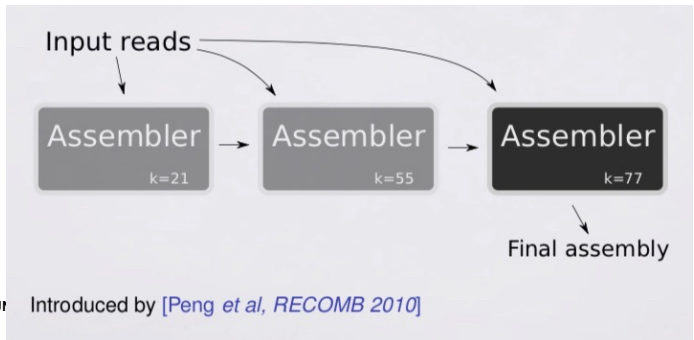
91-mer assembly
This graph has 2386 nodes and 304 edges and mostly consists of disconnected nodes.
This k-mer size is definitely too large.

Assemblers become faster and more accurate

By adapting the graph to coverage
By applying error correction
By performing local assembly

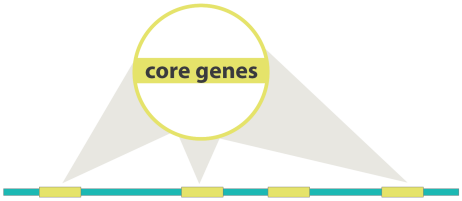
Originally proposed by IDBA (Peng et al, 2010)
Minia, MEGAHIT and (meta)SPAdes take advantage of combining multiple kmers iteratively.





Introduced by [Peng *et al, RECOMB 2010*]

Assembly quality measurements: how to?

Contiguity	Completeness	Correctness
		
Statistics: N50, QUAST, Merqury, …	BUSCO, TIDK, BlobToolKit …	Compare with similar genomes DOT, …

Assembly quality measurements

Number of contigs

Ideally 1 for a bacterial genome…, but the lower the better

Contig sizes

The larger the better (up to the size of the genome), usually given in maximum, minimum and average lengths.

Correctness

Difficult to assess for a new genome

N50 **The most used quality value for *de novo* assembly**

The N50 is the size of the smallest of all the large contigs covering 50% of the genome

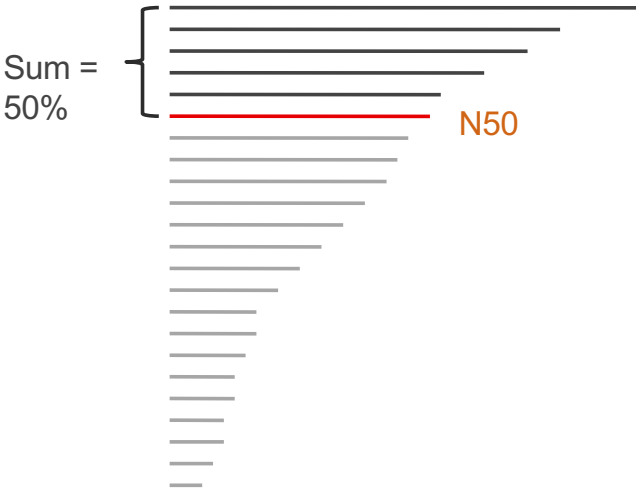
Kmer counting analysis

Allows for error and ploidy detection

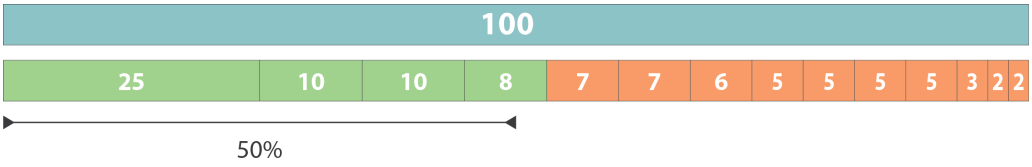
N50 what's that?

1) Sort the contigs by size
2) Sum them starting with the largest until you reach 50% of the estimated genome size

Last contig added = N50



Example: N50 = 8 and L50=4



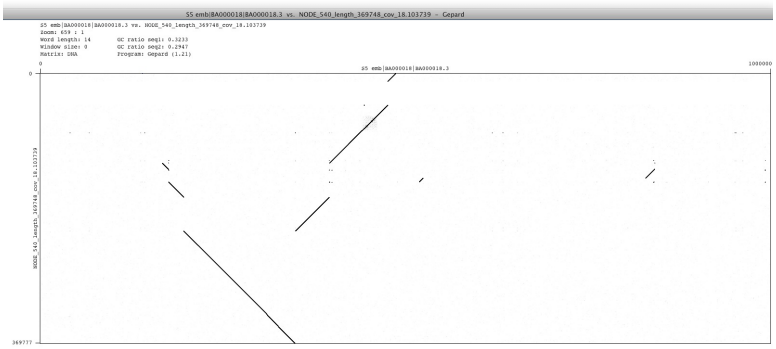
How to choose the Kmer parameter for deBruijn graphs?

K =	21	23	25	27	29	31
Velvet N50	12182	43427	67361	66898	66306	**107440**
Abyss N50	15928	25693	29334	30241	**31596**	29797
SOAPdenovo N50	98956	**99286**	82319	82910	84517	52098
SPAdes N50 (combined kmers)	44589	44589	44589	44589	44589	44589

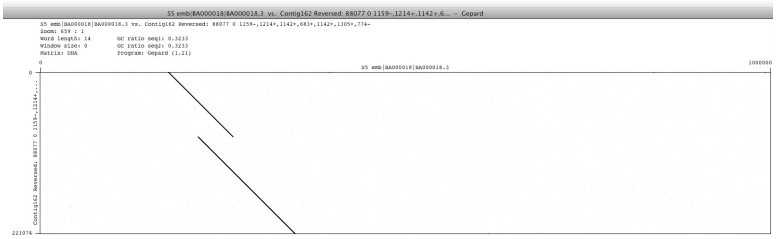
Best scores compared

K =	Velvet31	ABySS29	SOAPdenovo23	SPAdes
Nr contigs	252	1339	213	**182**
Consensus size bp	2936521	3078819	**2825424**	2851989
N50	**107440**	31596	99286	44589
Max	**369778**	132996	252985	173042

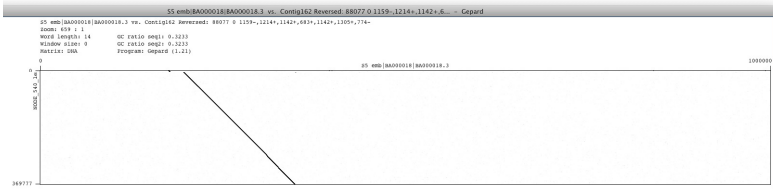
However longest contig is not always the best…



Velvet:
369’778bp



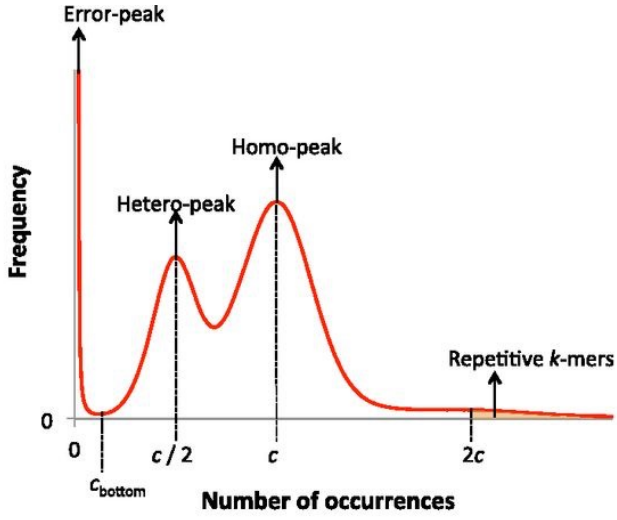
ABySS:
88’077bp and
132’996bp



SPAdes:
173’042bp

Kmer counting

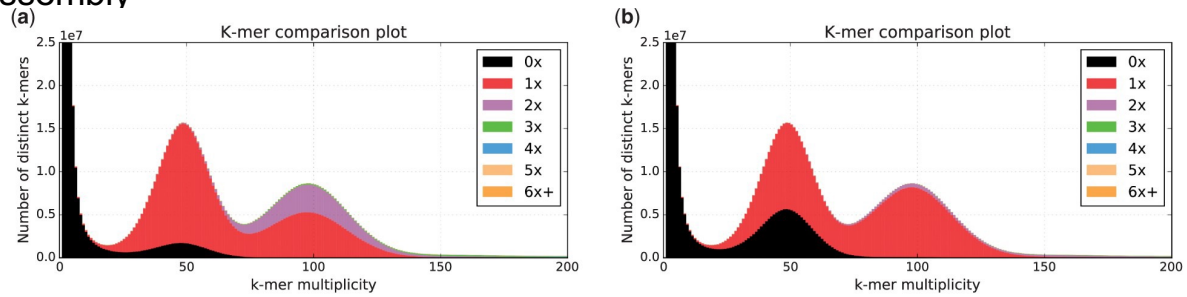
Originally proposed via Jellyfish to evaluate the error rate in reads. Reused by KAT (and Merqury) for counting kmers in both reads and genome assemblies can reveal discrepancies and help calculate the ploidy.



Guillaume Marcais and Carl Kingsford, A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. Bioinformatics (2011) 27(6): 764-770 (first published online January 7, 2011) doi:10.1093/bioinformatics/btr011
Rei Kajitani, Kouta Toshimoto, Hideki Noguchi, et al. Efficient de novo assembly of highly heterozygous genomes from whole-genome shotgun short reads. Genome Res. published online April 22, 2014 doi:10.1101/gr.170720.113

KAT or MERQURY (kmer analysis toolkit)

Visualise the copy number spectra of WGS data compared against an assembly



1st assembly contains most (but not all) the heterozygous content, and introduces more duplications on homozygous content (purple peak)

2nd assembly is more collapsed, including mostly a single copy of the homozygous content and heterozygous content removed (black peak)

Daniel Mapleson, Gonzalo Garcia Accinelli, George Kettleborough, Jonathan Wright, Bernardo J Clavijo, KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies, Bioinformatics, Volume 33, Issue 4, 15 February 2017, Pages 574–576, https://doi.org/10.1093/bioinformatics/btw663
Rhie, A., Walenz, B.P., Koren, S. et al. Merqury: reference-free quality, completeness, and phasing assessment for genome assemblies. Genome Biol 21, 245 (2020). https://doi.org/10.1186/s13059-020-02134-9

Overview of a sequencing project

Select DNA and sequencing method

Quality Control

if necessary Trimming/Filtering

De novo assembly with cleaned data

1st step: Assembly

consists mainly in building contigs from short reads

2nd step: Scaffolding (optional)

where contigs are ordered and oriented

3rd step: Finishing or Closing (optional)

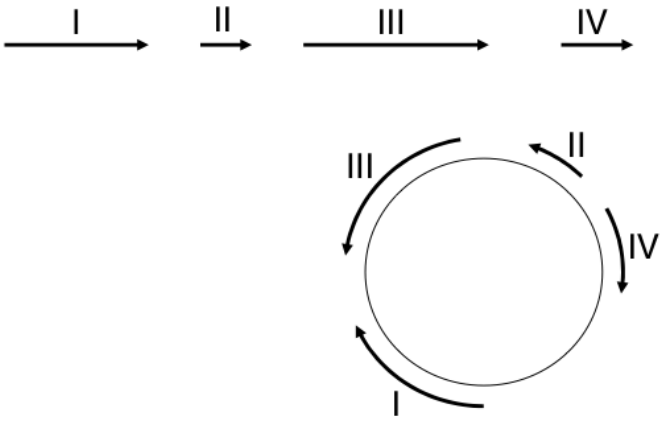
where gaps are closed

Genome annotation (draft or final)

Submission to databases & publication (not discussed here)

Scaffolding

Step by which non-overlapping contigs are ordered and oriented along a chromosome



Linking information source

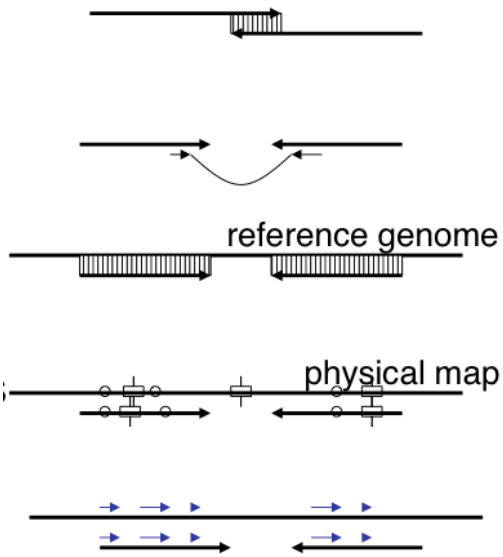
Overlaps

Paired-Ends & Mate Pairs

Reference based links

Physical markers

Gene/operon synteny



Other methods…

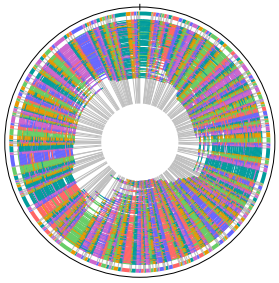
Existing methods for scaffolding and closing

More Next Generation Sequencing

- Illumina protocol (linked reads or synthetic long reads)
- **PacBio or Nanopore long reads**
- **HiC scaffolding**

Other methods (without sequencing)

- **Optical maps**



Long reads sequencing

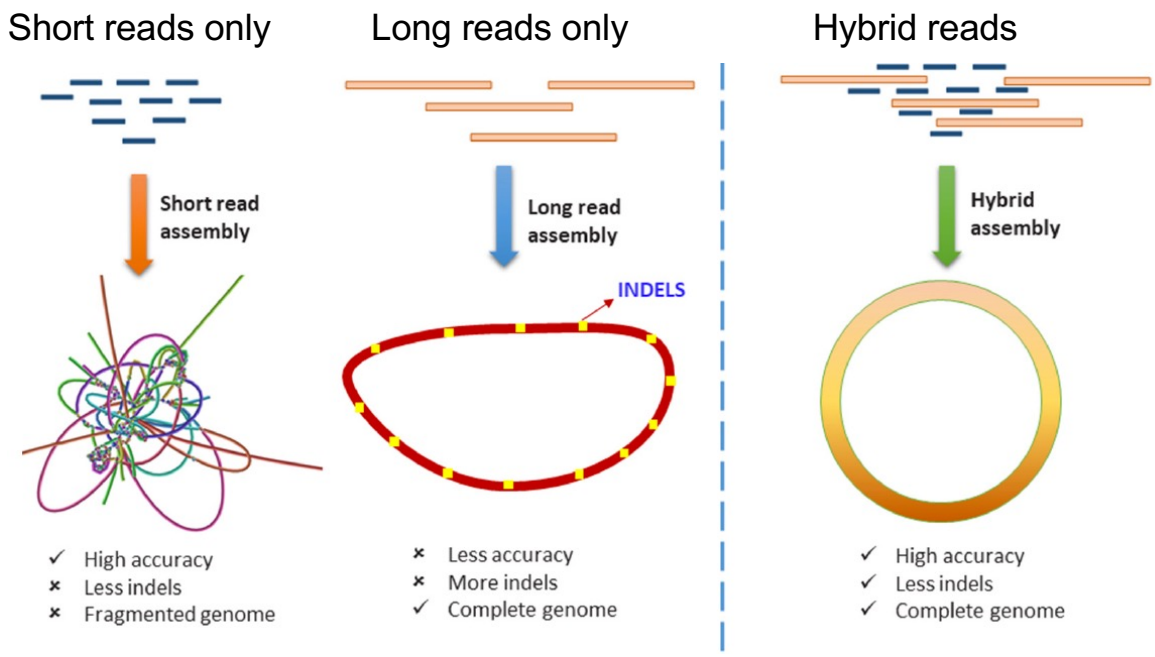
PacBio

~ 20Kbp to 200Kbp

Oxford Nanopore

Up to 1Mbp

Possible strategies for assembling long and/or short reads



Vasudevan et al, https://doi.org/10.1016/j.ygeno.2019.04.006

Assembly with long reads only

Flye

Kolmogorov, M., Yuan, J., Lin, Y. et al. Assembly of long, error-prone reads using repeat graphs. Nat Biotechnol 37, 540–546 (2019). https://doi.org/10.1038/s41587-019-0072-8

Canu

Koren S., Walenz B. P., Berlin K., Miller J. R., Bergman N. H., Phillippy A. M. (2017). Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. Genome Res. 27, 722–736. 10.1101/gr.215087.116,

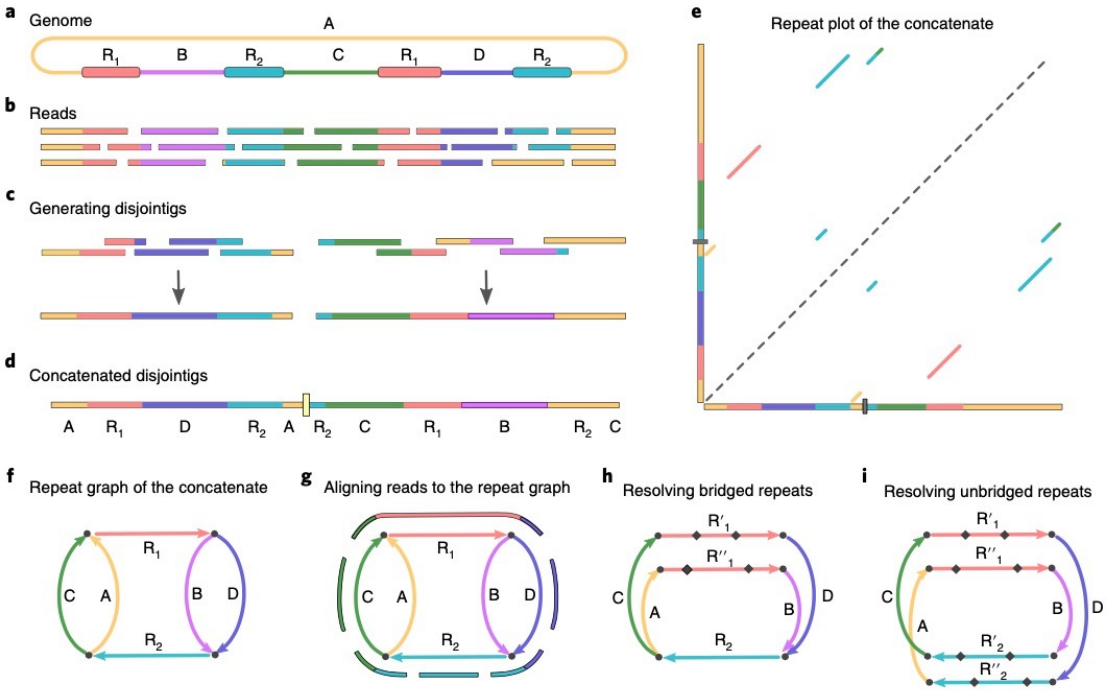
Miniasm

Li, H. Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. Bioinformatics 32, 2103–2110 (2016).

FALCON/HGAP

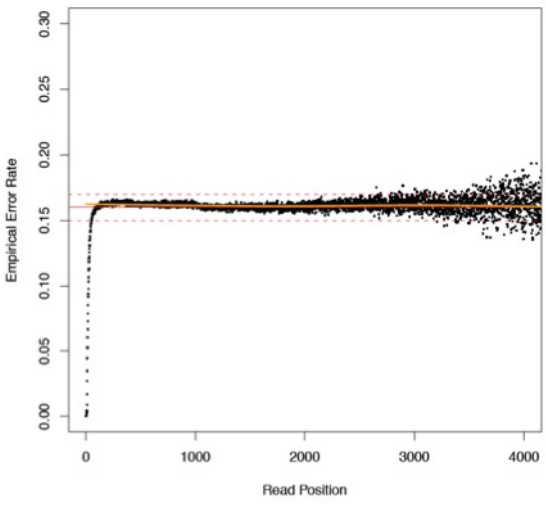
Chin et al. (2016). Phased diploid genome assembly with single-molecule real-time sequencing. Nature Methods. 13(12), 1050.

Flye algorithm to improve assembly



Kolmogorov, M., Yuan, J., Lin, Y. et al. Assembly of long, error-prone reads using repeat graphs. Nat Biotechnol 37, 540–546 (2019). https://doi.org/10.1038/s41587-019-0072-8

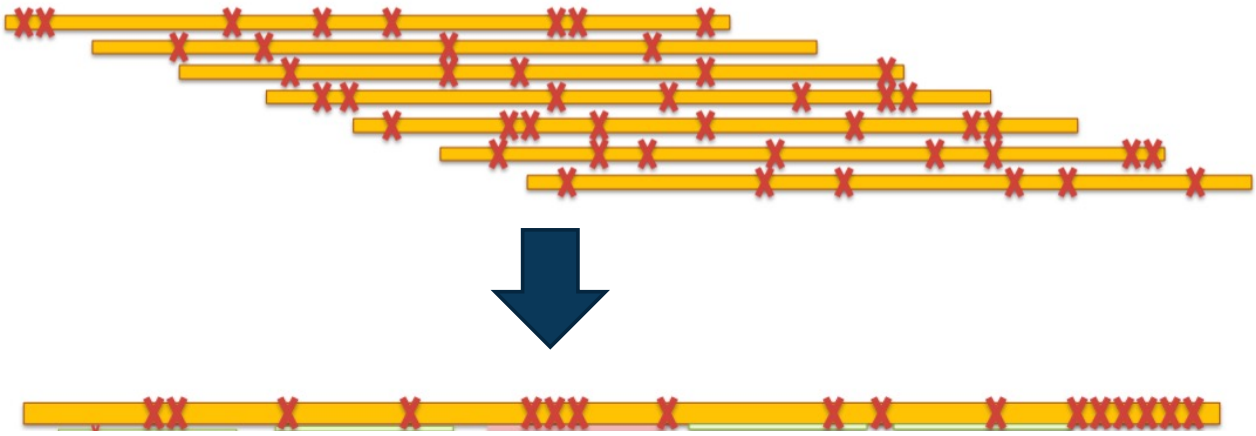
Problem: raw long reads error rate is high!!



Match	83.7%
Mismatch	1.4%
Insertions	11.5%
Deletions	3.4%

Raw long reads only

No read correction: very fast, error-prone

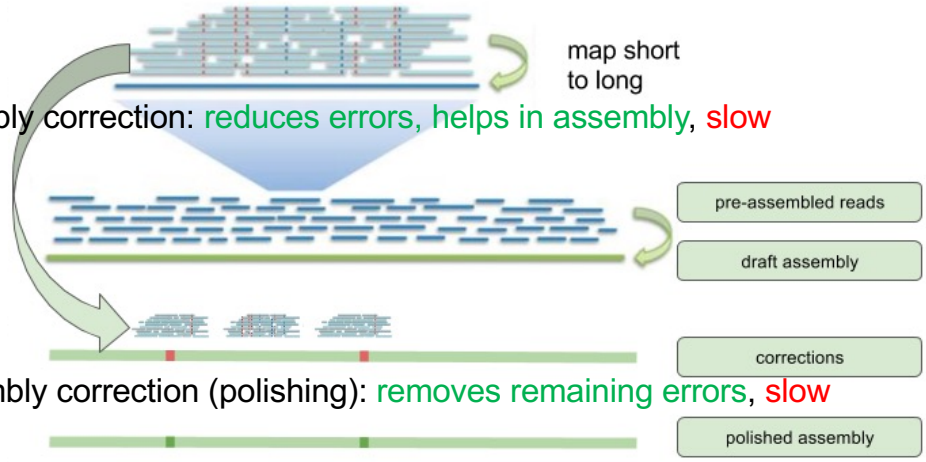


Applied by: **miniasm, smartdenovo, HINGE, wtdbg2 ...**

Not recommended!

http://schatzlab.cshl.edu/presentations/2012-01-17.PAG.SMRTassembly.pdf

Long reads only + polishing (2 options pre and/or post assembly)



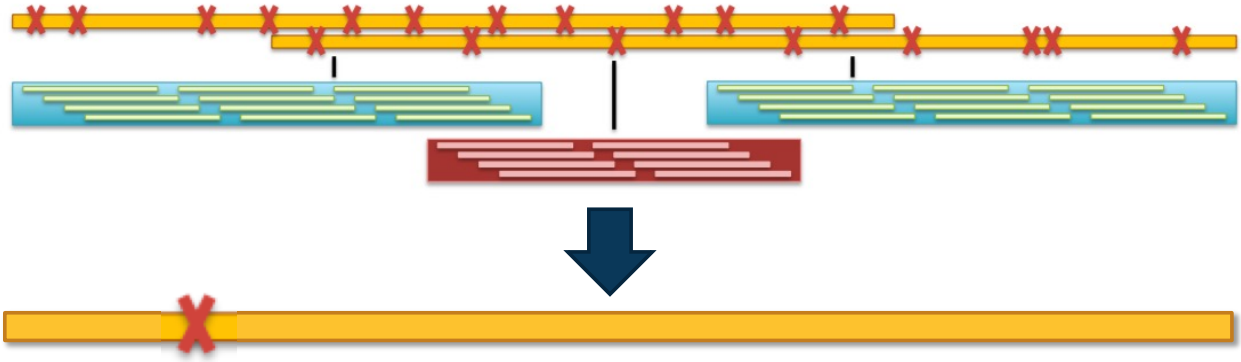
Pre-assembly correction: reduces errors, helps in assembly, slow

Post-assembly correction (polishing): removes remaining errors, slow

Applied by: **Canu, HGAP3(Quiver), HGAP4(Arrow), FALCON, MECAT, Racon, Nanopolish,…**

Hybrid assembly (Illumina short reads first)

Usually performed to scaffold the short reads contigs using a low coverage of long reads

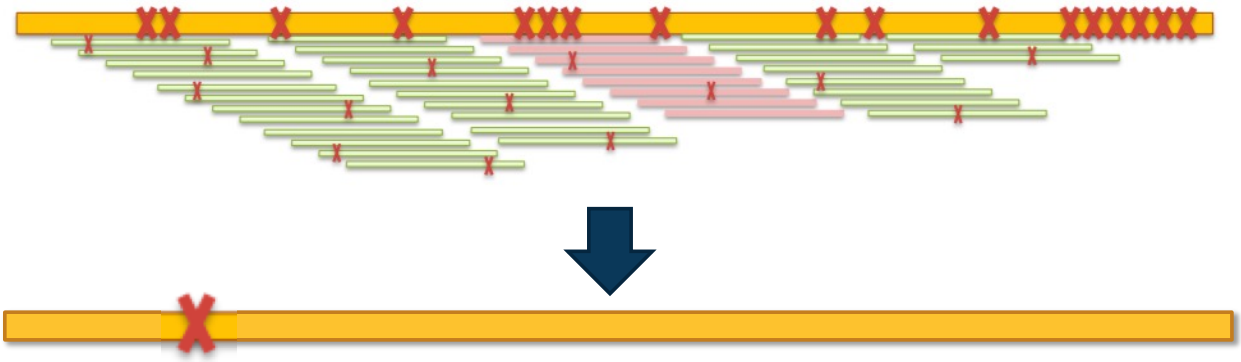


Applied by: **HybridSPAdes, UniCycler, SSPACE-LongReads, LINKS**

http://schatzlab.cshl.edu/presentations/2012-01-17.PAG.SMRTassembly.pdf

Hybrid assembly (long reads first)

Usually performed to correct the long reads contigs using a high coverage of Illumina short reads (polishing)



Applied by: **PILON, Nanopolish, Racon, LSC, PacBioToCA, LoRDEC, proovread, CoLoRMap, HECIL**

http://schatzlab.cshl.edu/presentations/2012-01-17.PAG.SMRTassembly.pdf

UNIVERSITÉ DE FRIBOURG / UNIVERSITÄT FREIBURG | Biochemistry Unit | Falquet Laurent

Error correction improves gene prediction

PacBio and Oxford Nanopore reads mainly contain **INDELs errors**, whereas Illumina reads mainly contains **substitution errors**.

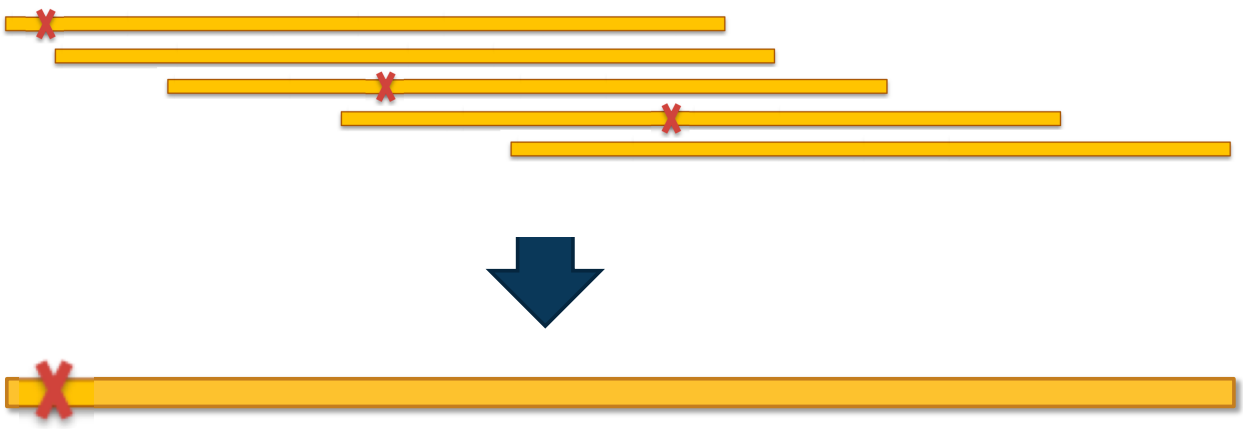
As a result a simple 1 bp indel can have dramatic consequences on the predicted genes (-> frameshifts).

It is thus critical to polish the assemblies to remove as much as possible of these indels.

Watson, M., Warr, A. Errors in long-read assemblies can critically affect protein prediction. Nat Biotechnol 37, 124–126 (2019). https://doi.org/10.1038/s41587-018-0004-z

PacBio HiFi reads avoids error correction!

No read correction: very fast, little errors



Applied by: **hifiasm, hicanu, Flye, FALCON**

http://schatzlab.cshl.edu/presentations/2012-01-17.PAG.SMRTassembly.pdf

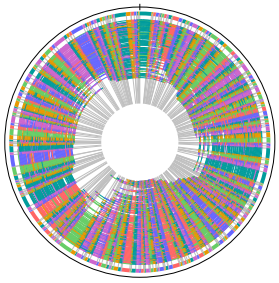
Existing methods for scaffolding and closing

More Next Generation Sequencing

- Illumina protocol (linked reads or synthetic long reads)
- PacBio or Nanopore long reads
- **HiC scaffolding**

Other methods (without sequencing)

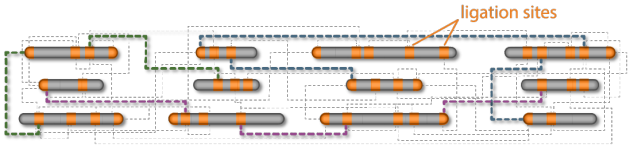
- **Optical maps**



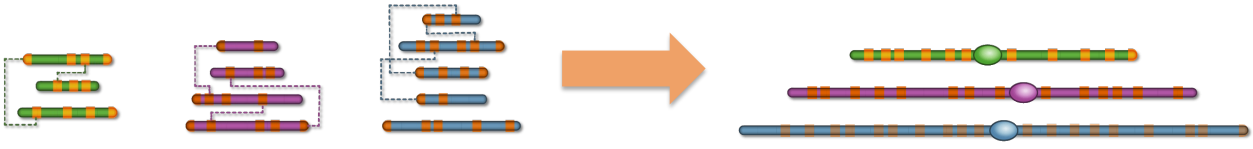
HiC Scaffolding: a very complementary technique using short reads

Takes advantage of proximity crosslinking

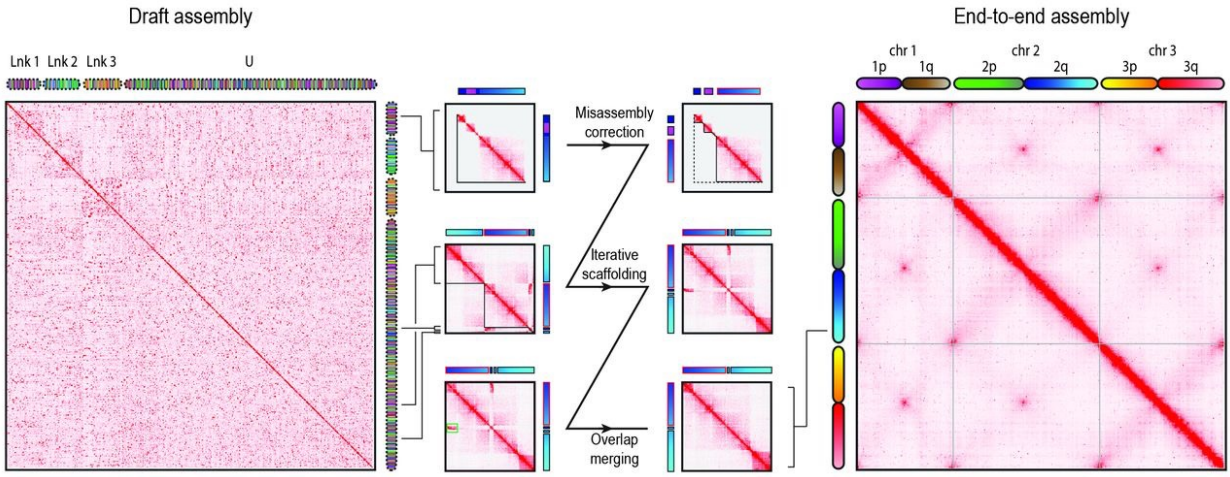
Then sequence junctions with Illumina, the reads are mapped onto contigs



Connections are identified and ordered into chromosome-like scaffolds



Example with *Aedes aegypti*



Software: instaGRAAL, SALSA2, YAHS, Juicer
Visualisation: JuiceBox

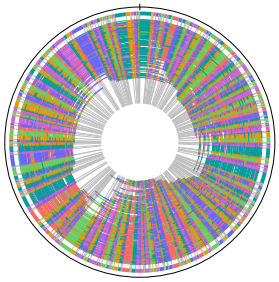
Existing methods for scaffolding and closing

More Next Generation Sequencing

- **Illumina protocol (linked reads or synthetic long reads)**
- **PacBio or MinION long reads**
- **HiC scaffolding**

Other methods (without sequencing)

- **Optical maps**

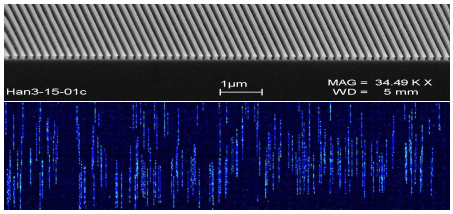


Bionano Optical Mapping (no sequencing!)

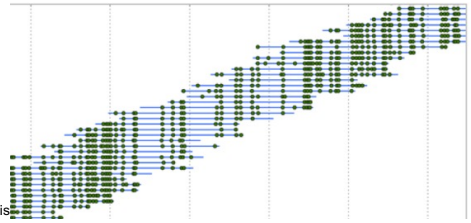
1) Extraction of ultra HMW DNA

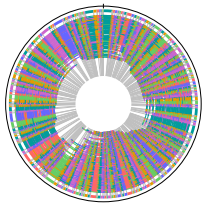
2) Label DNA at specific sequence motifs

3) Linearise DNA in NanoChannel array and image

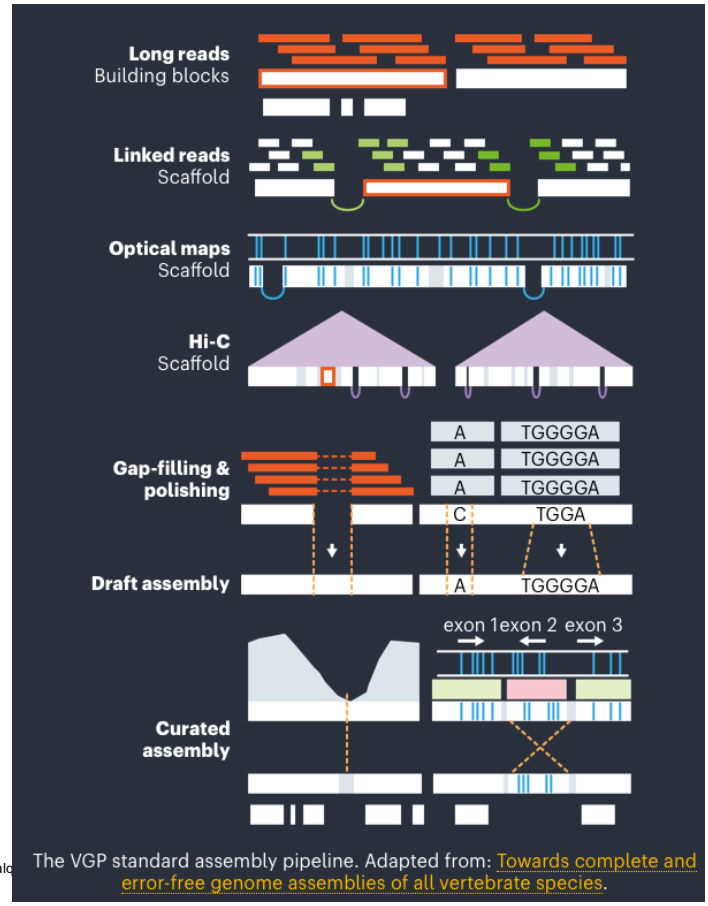


4) Combine contigs based on optical image





Recommended pipeline of the Vertebrate Genome Project



The VGP standard assembly pipeline. Adapted from: Towards complete and error-free genome assemblies of all vertebrate species.

https://vertebrategenomesproject.org

VERTEBRATE GENOMES PROJECT

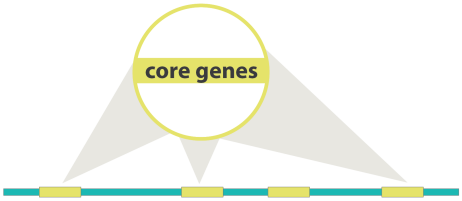
A PROJECT OF THE G10K CONSORTIUM

Assembly quality measurements: how to?

Contiguity

Statistics:
N50, QUAST,
Merqury, …

Completeness



BUSCO, TIDK,
BlobToolKit …

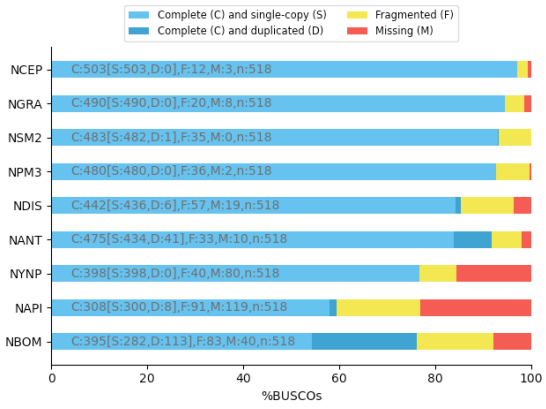
Correctness

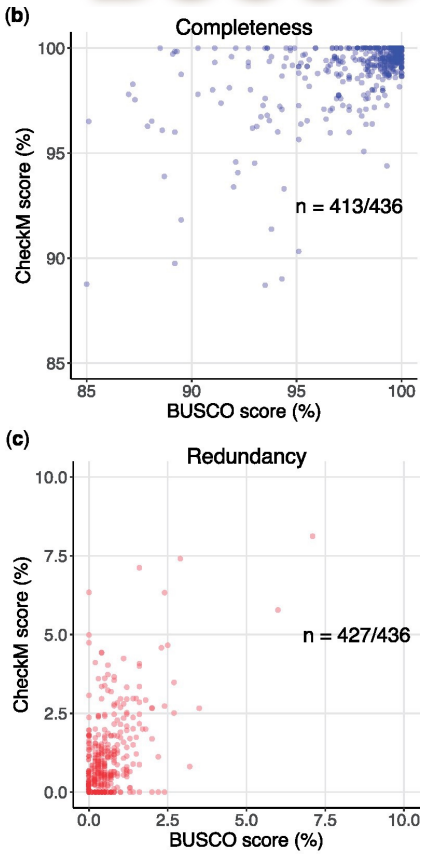
Compare with similar
genomes
DOT, …

BUSCO searches for unique genes

BUSCO

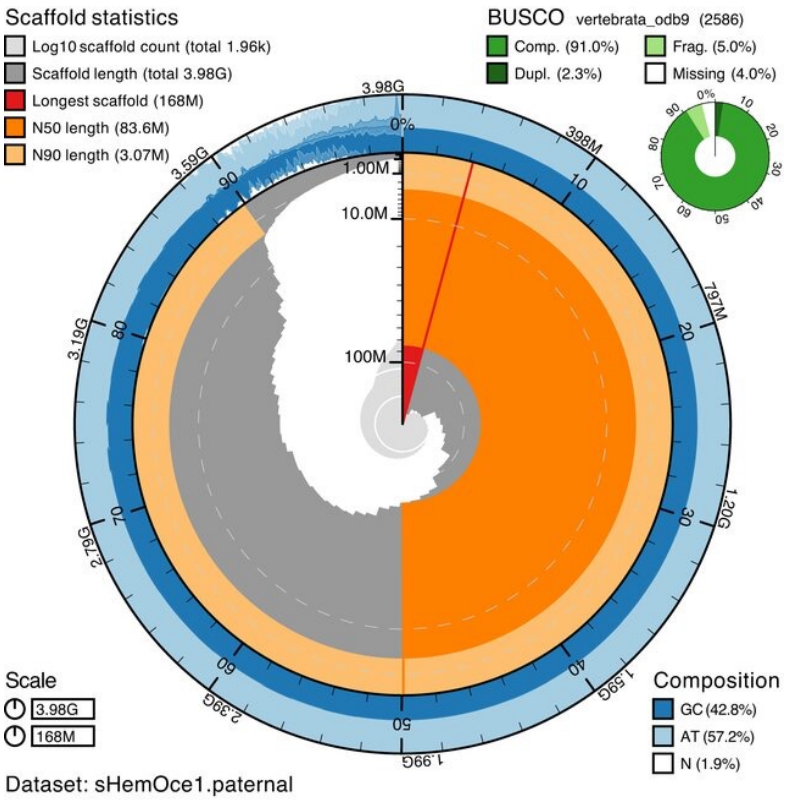
BUSCO attempts to provide a quantitative assessment of the completeness in terms of expected gene content of a genome assembly, transcriptome, or annotated gene set. The results are simplified into categories of Complete and single-copy, Complete and duplicated, Fragmented, or Missing BUSCOs.





Mosè Manni et al. Molecular Biology and Evolution, Volume 38, Issue 10, October 2021, Pages 4647–4654, https://doi.org/10.1093/molbev/msab199

BlobToolKit viewer: example snailplot

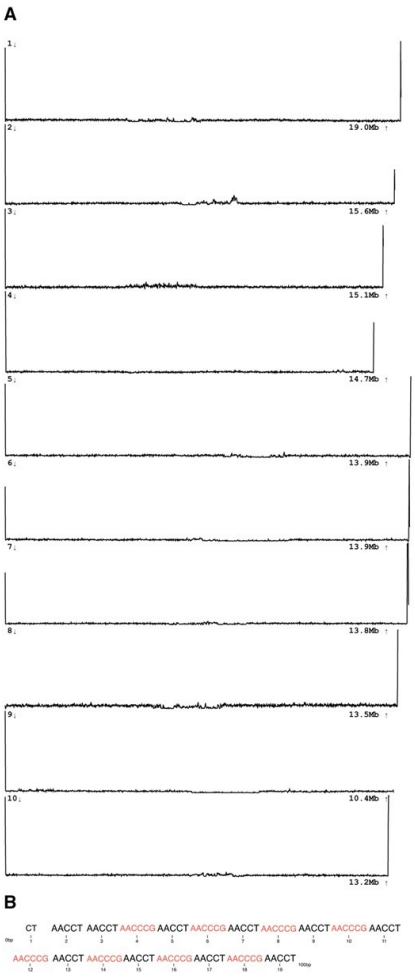


UNIVERSITÉ DE FRIBOURG / UNIVERSITÄT FREIBURG | Biochemistry Unit | Falquet Laurent

DOI: 10.1038/s41467-023-42238-x

Telomer identification toolkit (TIDK)

Use standard or user provided telomeric sequences to identify the telomeric regions (= the ends of the chromosomes)

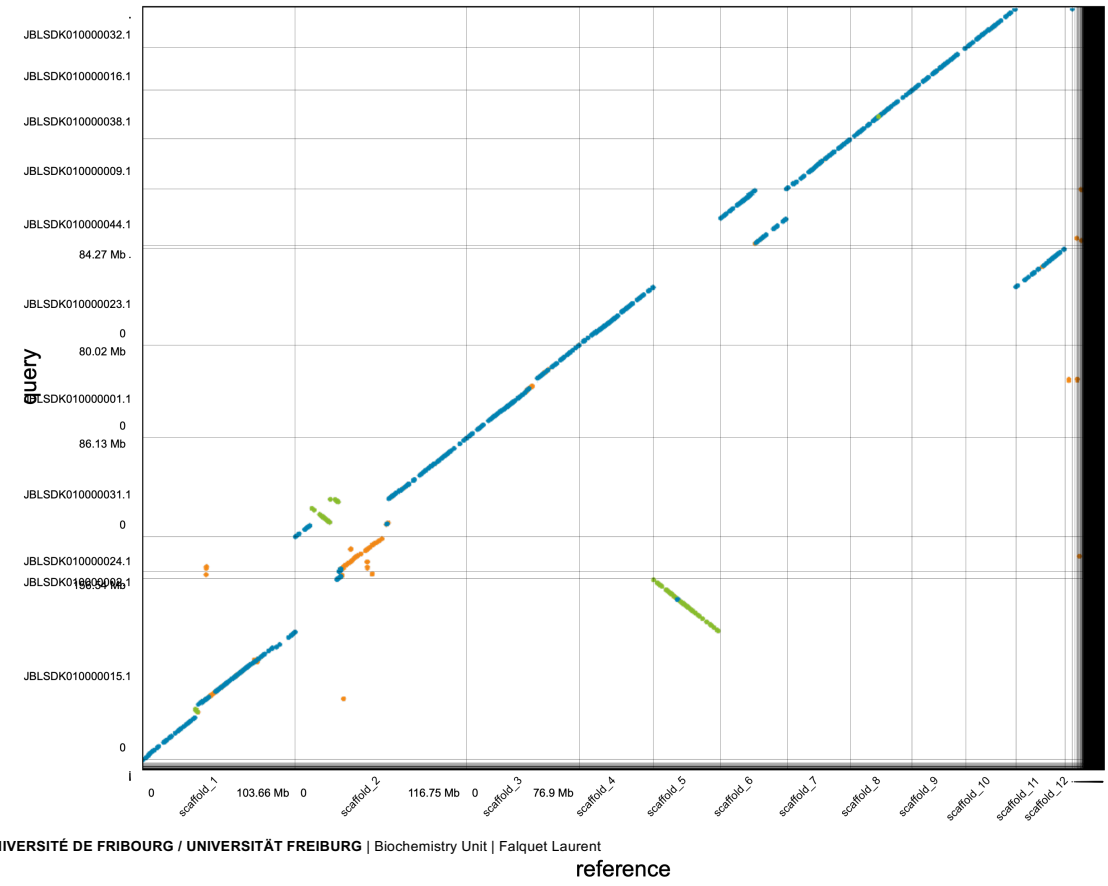


Brown et al, 2025

https://doi.org/10.1093/bioinformatics/btaf049

UNIVERSITÉ DE FRIBOURG / UNIVERSITÄT FREIBURG | Biochemistry Unit | Falquet Laurent

DOT or dotplot



Genome assembly quality assessment tools

QUAST (FASTA statistics)

https://quast.sourceforge.net

BUSCO (gene and protein completeness and contamination)

https://busco.ezlab.org

BlobToolKit

https://blobtoolkit.genomehubs.org

MERQURY (kmer based analysis)

https://github.com/marbl/merqury

Genomescope (kmer based analysis)

http://genomescope.org

DOT (graphical pairwise comparison)

https://dot.sandbox.bio

TIDK (check for telomers)

https://github.com/tolkit/telomeric-identifier

Summary

Current methods for short reads and long reads genome assembly are based on deBruijn graphs and OLC (overlap-layout-consensus), respectively.

HiFi long reads assemblies are the most successful
With hybrid assemblies combining HiC data for scaffolding

Issues and limitations are mainly due to repeat sequences

QC of the reads and validation of assemblies are essential steps

Thank you for your attention. Questions?