Biochem 3BP3

DNA Sequencing & Genome Assembly

Week of Oct 25, 2021

First genome - Bacteriophage MS2 in 1976 (3.5 Kbp)

First bacterial genome - *Haemophilus influenzae* in 1995 (1.8 Mbp)

First eukaryote genome - *Saccharomyces cerevisiae* in 1996 (12.1 Mbp)

First animal genome - *Caenorhabditis elegans* in 1998 (100 Mbp)

First plant genome - *Arabidopsis thaliana* in 2000 (119 Mbp)

Human genome in 2001 (3.2 billion bp)

Genome Sequences as References

- A wide variety of research currently uses DNA sequencing as a diagnostic tool or assay, e.g.
  - RNA-Seq to measure changes in gene expression
  - ChIP-Seq to understand regulatory binding sites
  - MolEpi to track movement of pathogens
  - Exome sequencing to determine genetic underpinnings of disease

- These techniques generate tens of millions of DNA sequencing ‘reads’ and these are analyzed by mapping reads to reference genome sequences using “short-read alignment” methods
  - e.g. Burrows-Wheeler Aligner (BWA)

- Thus, the genome sequencing revolution has open up a wide-range of new methods for research

- But what if you do not have a reference genome sequence?

Genome Assembly

- Genome Assembly is the combined laboratory and computational methods used to determine the genome sequence of an organism

- Genome Assembly can be completed *de novo* for a previously unsequenced organism or may be performed to determine genomic differences among related organisms (guided assembly)

Genome Assembly is highly dependent upon DNA sequencing technologies

- Linear (directed) sequencing: clone mapping & sub-sequencing
  - is prohibitively expensive
- Genomes are sequenced in tiny fragments using a ‘shotgun’ approach
  - Sanger DNA sequencing (low volume) 500-1200 bp
  - Illumina DNA sequencing (high volume) 250 bp
- PacBio: long reads but high error
- Nanopore: long reads but high error

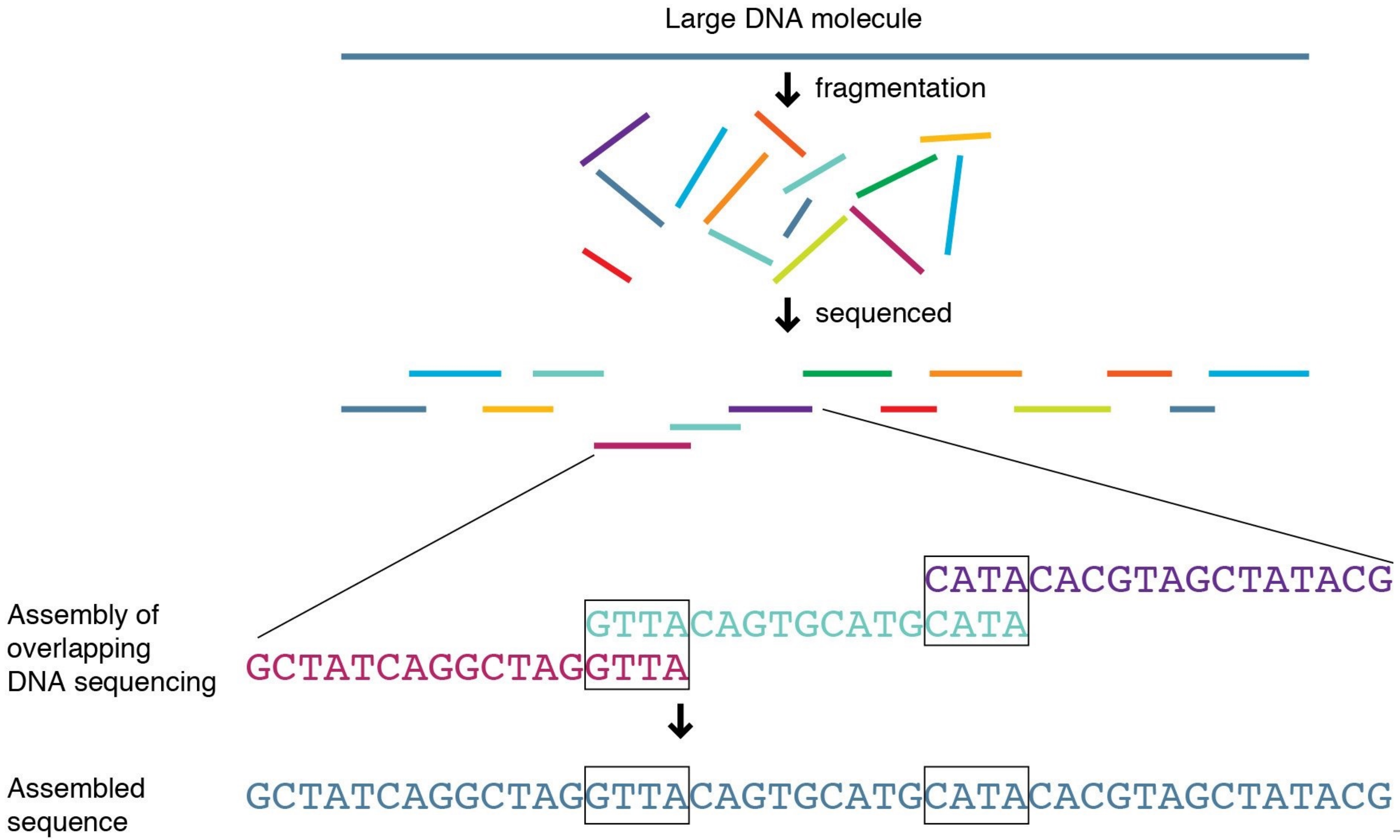
Key concepts

- Read & K-mer
- Phred score
- Mate-Pairs
- Coverage
- Contig
- Scaffold
- Closure & Gaps

Why is Genome Assembly Difficult?

- Lots of data
  - Human genome is 3 billion base pairs
  - *Arabidopsis thaliana* 135 Mbp
  - *Salmonella enterica* 4.8 Mbp
- Genomes are sequenced in tiny fragments using a ‘shotgun’ approach
  - Some regions of genome clone poorly or sequence poorly. How do you ensure that all parts are represented? (lab)
  - How do you put it all together like a massive jigsaw puzzle (computer)
- DNA sequencing has an error rate
  - A single pass is insufficient, we use multiple passes to look for **consensus**
- Genomes have repeated sequences
  - Two sequencing reads for different parts of the genome can have identical or near-identical sequence

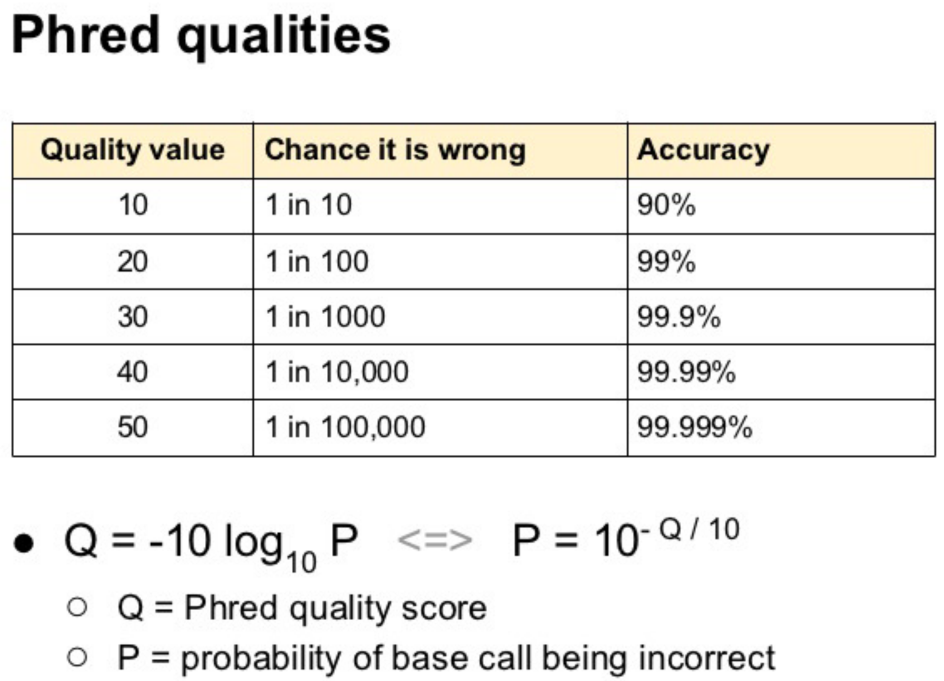
Shotgun Sequences – greedy assembly



http://knowgenetics.org

Greedy Assembly by Overlap

- The first step in assembly is joining reads of overlapping sequence
- This step must account for sequencing error rates
- Sequencing software uses the PHRED score to measure error



Phred qualities

Quality value	Chance it is wrong	Accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%

- $Q = -10 \log_{10} P \iff P = 10^{-Q/10}$
  - Q = Phred quality score
  - P = probability of base call being incorrect

Greedy Assembly by Overlap

- The first step in assembly is joining reads of overlapping sequence
- This step must account for sequencing error rates
- Sequencing software uses the PHRED score to measure error
- Shotgun sequencing samples each region of the genome multiples times to generate a PHRED-correct consensus sequence

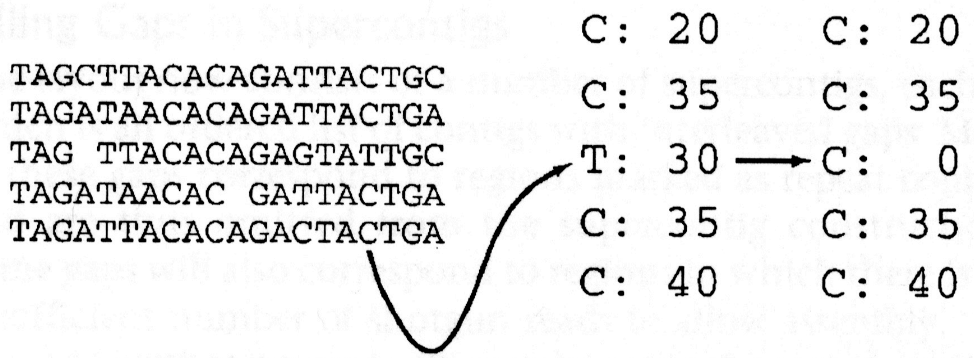


Figure 1 Correcting errors in reads. A portion of a multiple alignment between five reads is shown. In the highlighted column of the alignment, a base T of quality 30 is aligned only to bases C, some of which are of quality greater than 30. The base T is changed to a base C of quality 0.

The depth of sampling of a genome is called **'fold coverage'**

this example illustrates 5-fold coverage

Repeats

- Greedy algorithms merge reads into consensus ‘contig’ sequences
- What if reads are similar but from different parts of the genome?
- Greedy algorithms are insufficient to handle this problem

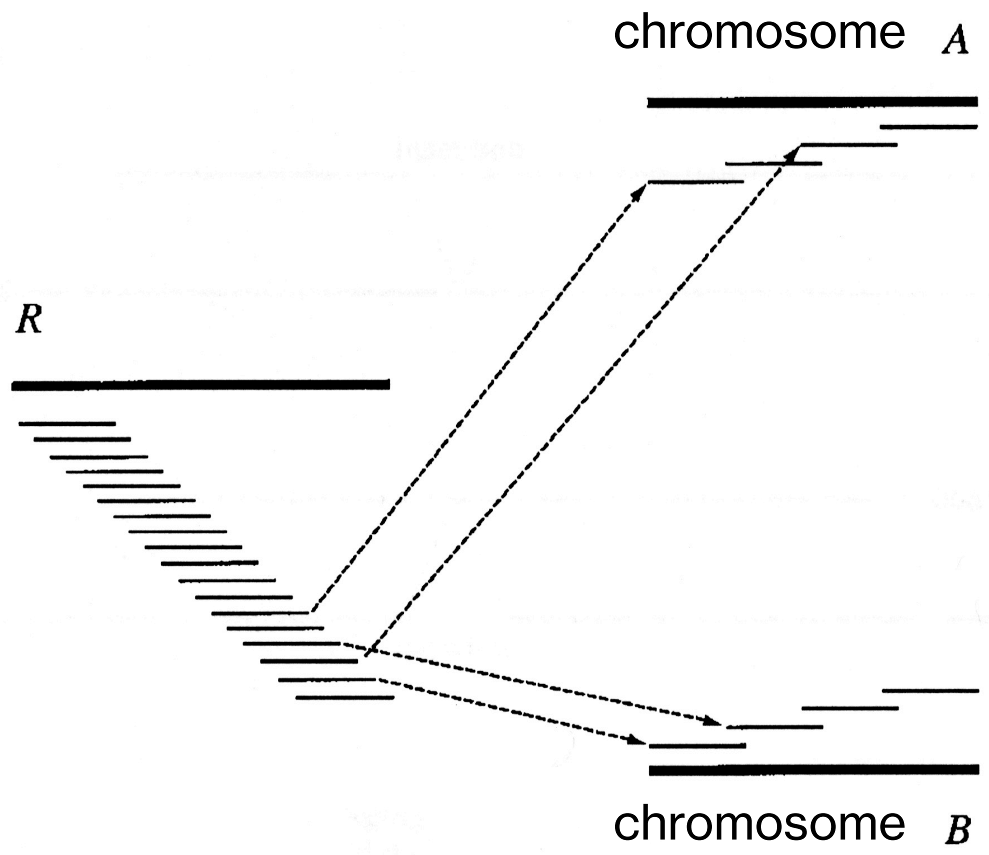


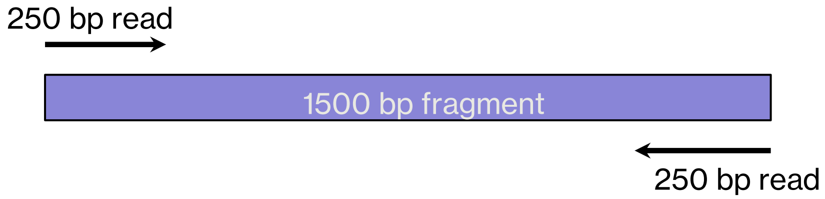
Figure 4 Detection of repeat contigs. Contig *R* is linked to contigs *A* and *B* to the right. The distances estimated between *R* and *A* and *R* and *B* are such that *A* and *B* cannot be positioned without substantial overlap between them. If there is no corresponding detected overlap between *A* and *B* (if their reads do not overlap), then *R* is probably a repeat linking to two unique regions to the right.

Bi-directional sequence (aka Mate Pairs)

- A large part of assembly algorithms are focused on detecting and correcting for repeats
- If we sequence each DNA fragment from both ends, we have paired reads with a specific geometry
    - Must point towards each other
    - Gap between them must reflect size of DNA fragments sequenced
- This geometry places constraints on the assembly algorithm and can tease apart repeats
- As long as one read is from a non-repeat region, it will constrain its mate-pair to assemble nearby in the genome

Bi-directional sequence (aka Mate Pairs)

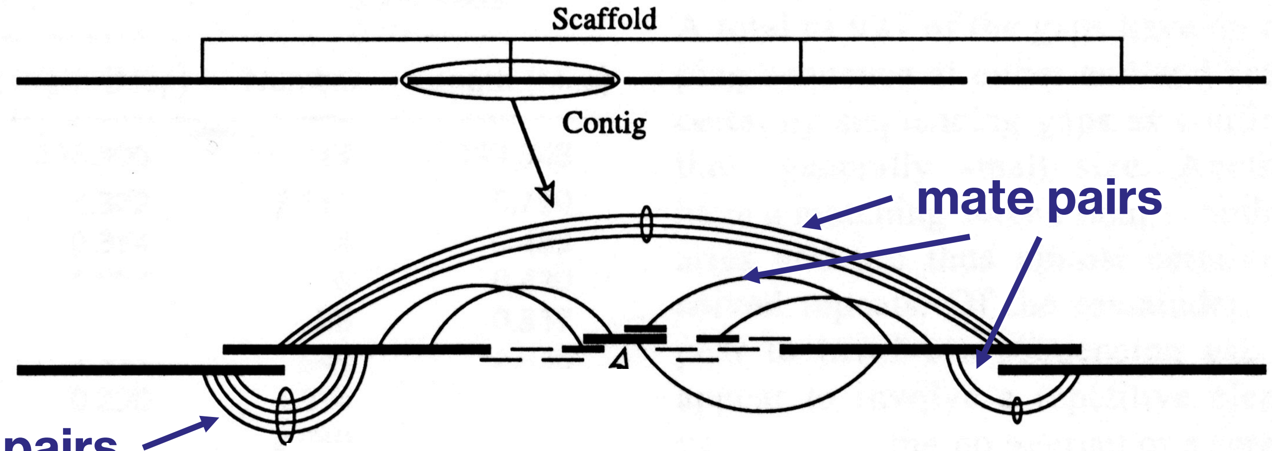
- A large part of assembly algorithms are focused on detecting and correcting for repeats
- If we sequence each DNA fragment from both ends, we have paired reads with a specific geometry



  - Must point towards each other
  - Gap between them must reflect size of DNA fragments sequenced
- This geometry places constraints on the assembly algorithm and can tease apart repeats
- As long as one read is from a non-repeat region, it will constrain its mate-pair to assemble nearby in the genome

Bi-directional sequence (aka Mate Pairs)

- Constraining greedy algorithms by using mate-pairs leads to contig sequences joined into scaffolds and separated by gaps
- This combination is very effective for accurate genome assembly, except when both reads are in repeat regions:
    - Long regions of repeats
    - Highly repetitive genomes
    - High nucleotide bias



mate pairs

mate pairs

Fig. 4. Anatomy of a scaffold. A scaffold is a collection of ordered contigs with approximately known distances between them. Our contigs are built from U-unitigs that form a scaffold via bundles and then have a series of rocks, stones, and pebbles filled into the gaps between them (where possible).

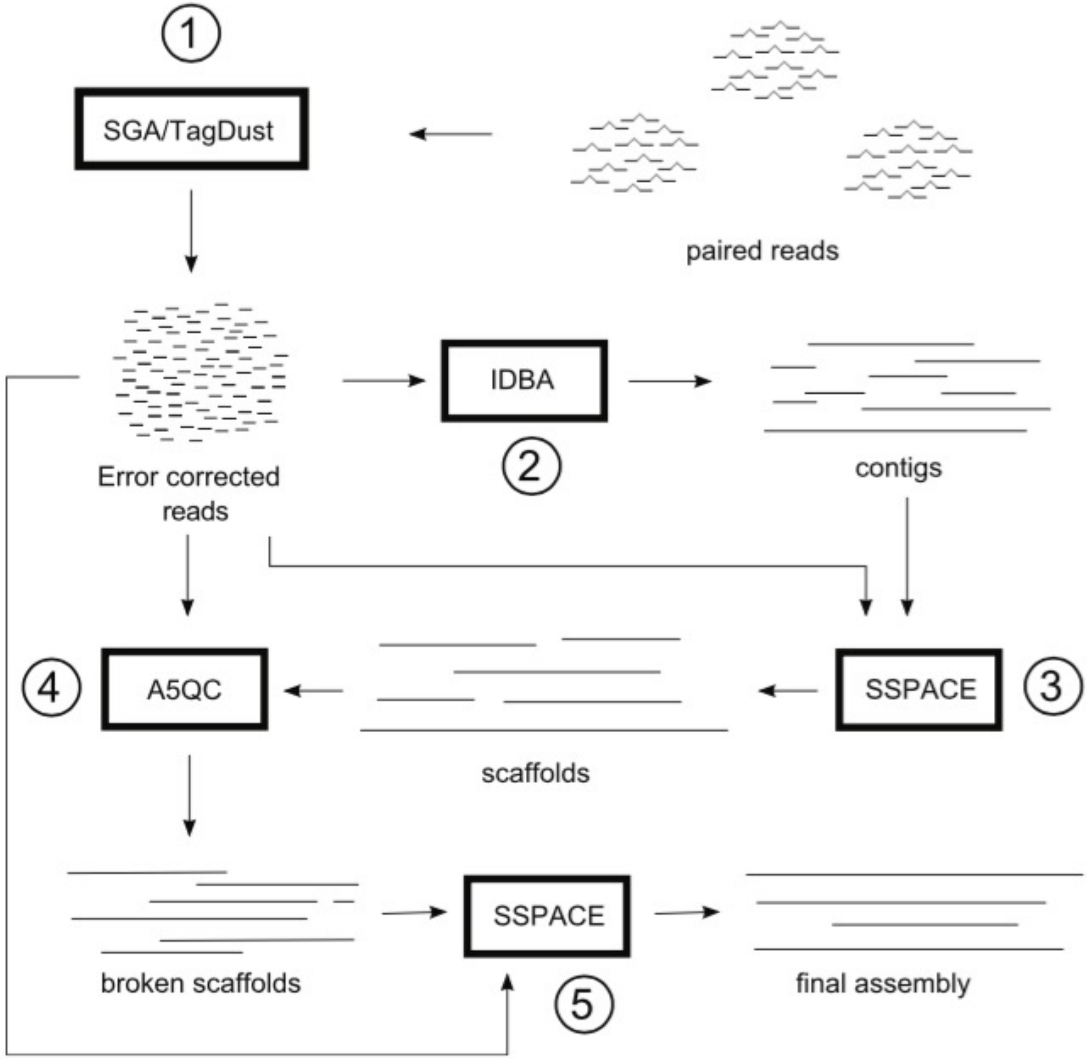
Pre-NGS used Sanger sequencing

- Long reads (e.g., 1200 bp) but very low volume
- 20-fold was considered good coverage
- Each sampled DNA fragment had to be cloned into a plasmid or phage vector
  - Costly and labour intensive!
  - Plasmid library construction will not clone all sections of the genome (e.g., telomeres, centromeres)
  - Plasmid library construction suffers from sampling bias – unenven sampling of mate-pairs across the genome
- Gel or capillary migration of DNA problematic for some regions of the genome, struggled with GC bias, and often confounded by DNA secondary structure
- PHRED was a critical advance
- Assembly software emphasized mate-pairs and used overlap-layout-consensus (OLC) ‘graph theory’ methods; long reads spanned repeat regions

NGS Genome Assembly

- Currently dominated by Illumina short read technology
- 250 bp mate pairs but hundreds-fold coverage
- **No cloning** – DNA fragments ligated directly to sequencing adapters
  - No specialist cloning skills
  - Commercial genome sequencing kits
  - Considerably less sampling bias
  - Huge cost savings
- Different sources of DNA sequencing error
  - PHRED scores adapted for NGS
  - High fold-coverage = high quality contig consensus sequences
- Shorter reads less likely to span repeat reagions
  - Repeats more problematic than long-read Sanger sequencing
- Assembly software emphasizes k-mers, de Bruijn graphs, Eulerian paths
- Modern pipelines include secondary ‘finishing’ or error-correction algorithms; optical mapping experiments can provide validation data

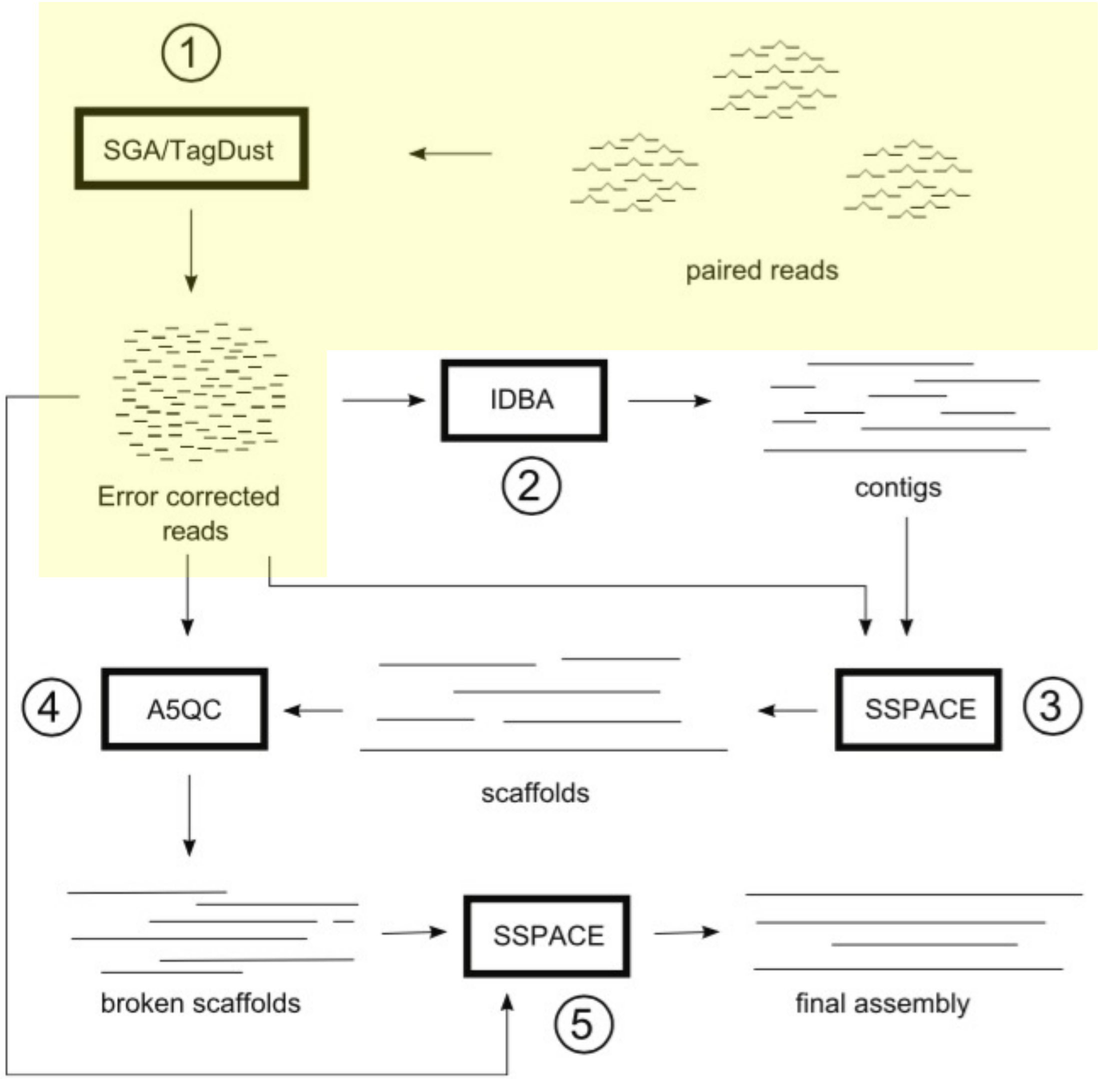
The A5 microbial genome assembly pipeline



Tritt *et al*. 2012. *PLoS One*. 7:e42304.

The A5 microbial genome assembly pipeline

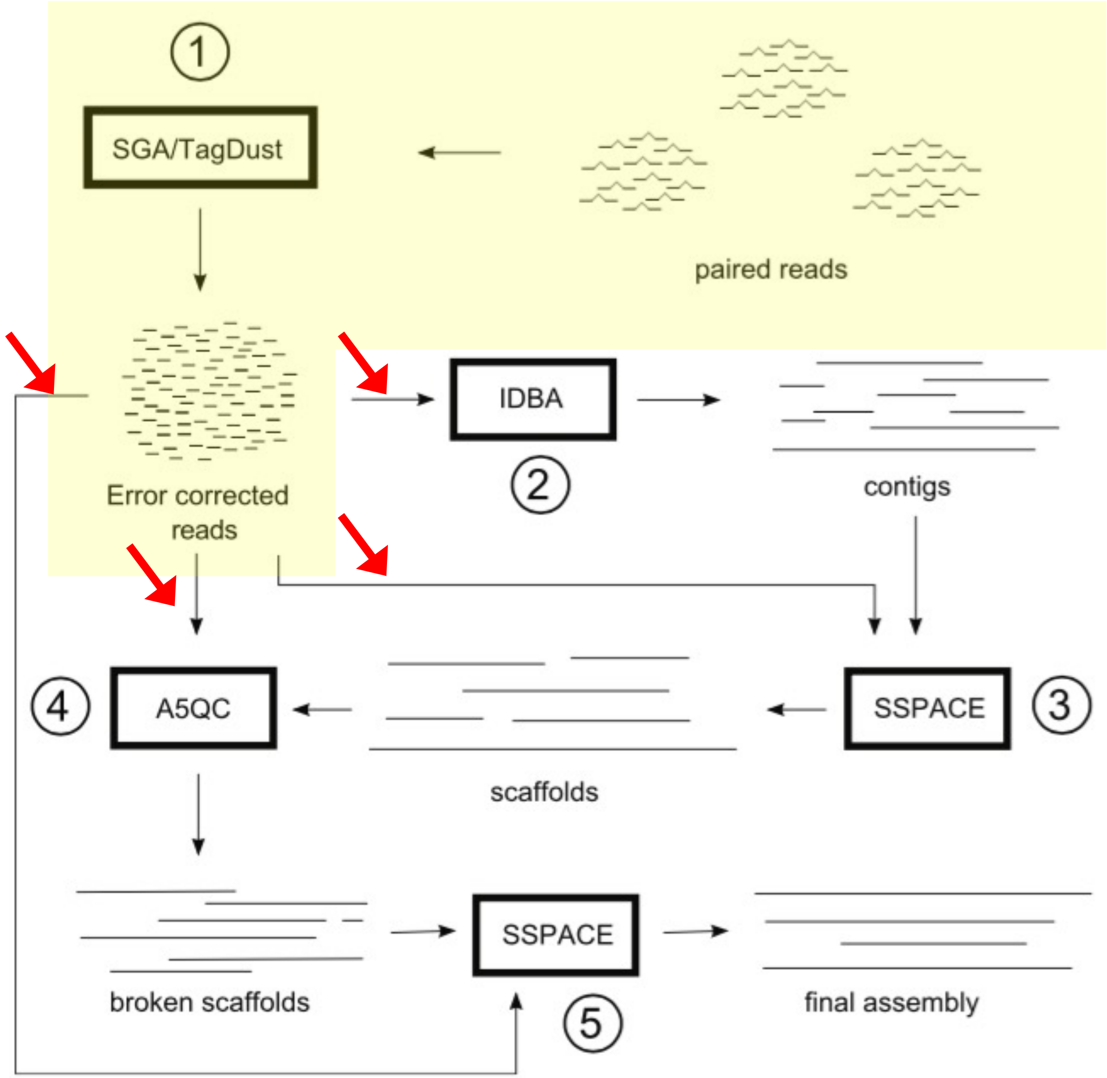
PHRED trimming, error correction – only use high quality data in genomic pipelines!



Tritt *et al*. 2012. *PLoS One*. 7:e42304.

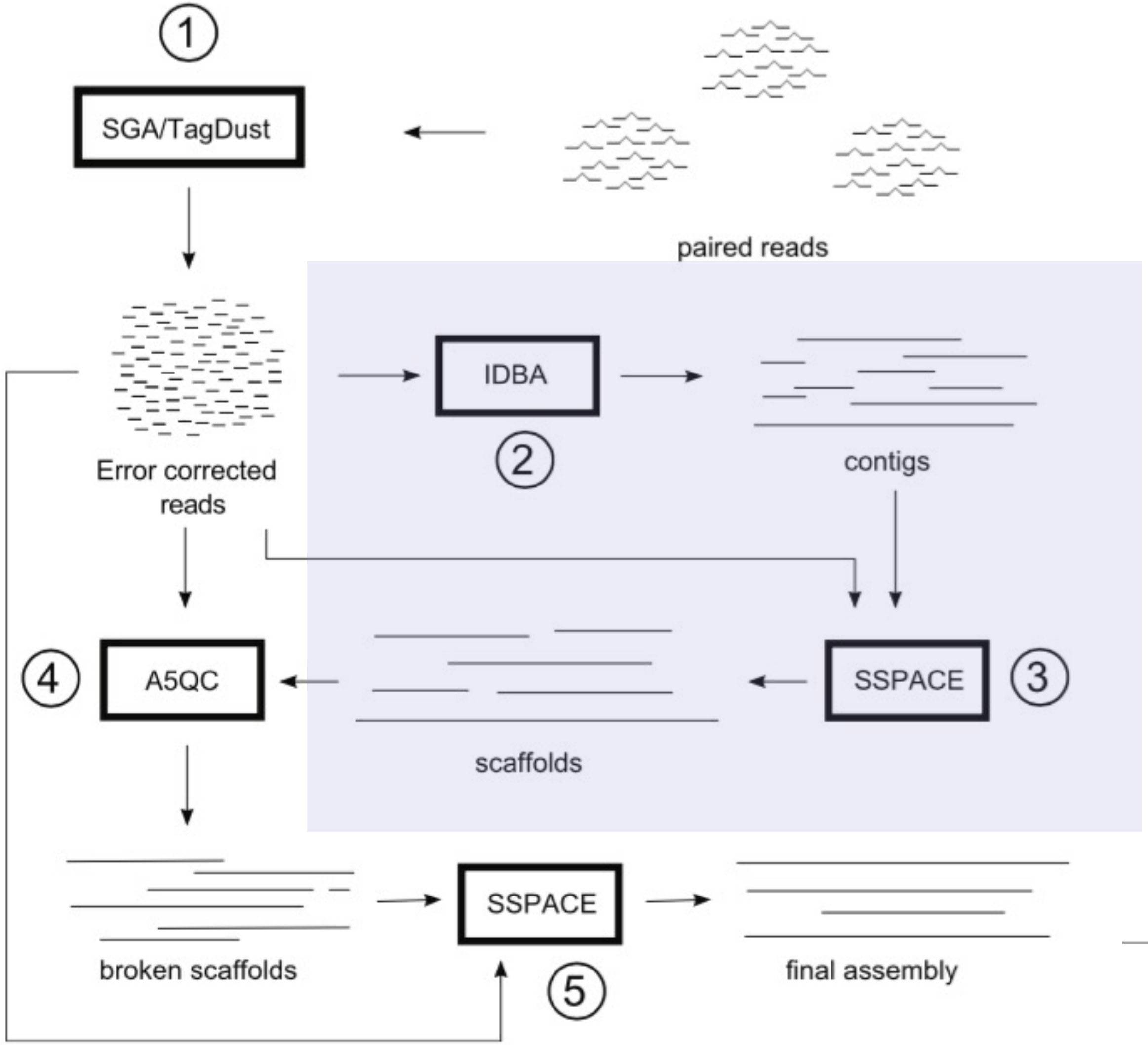
The A5 microbial genome assembly pipeline

How the error corrected reads are handled has to do with whether the assembler thinks they are representing **repeats**



Tritt *et al*. 2012. *PLoS One*. 7:e42304.

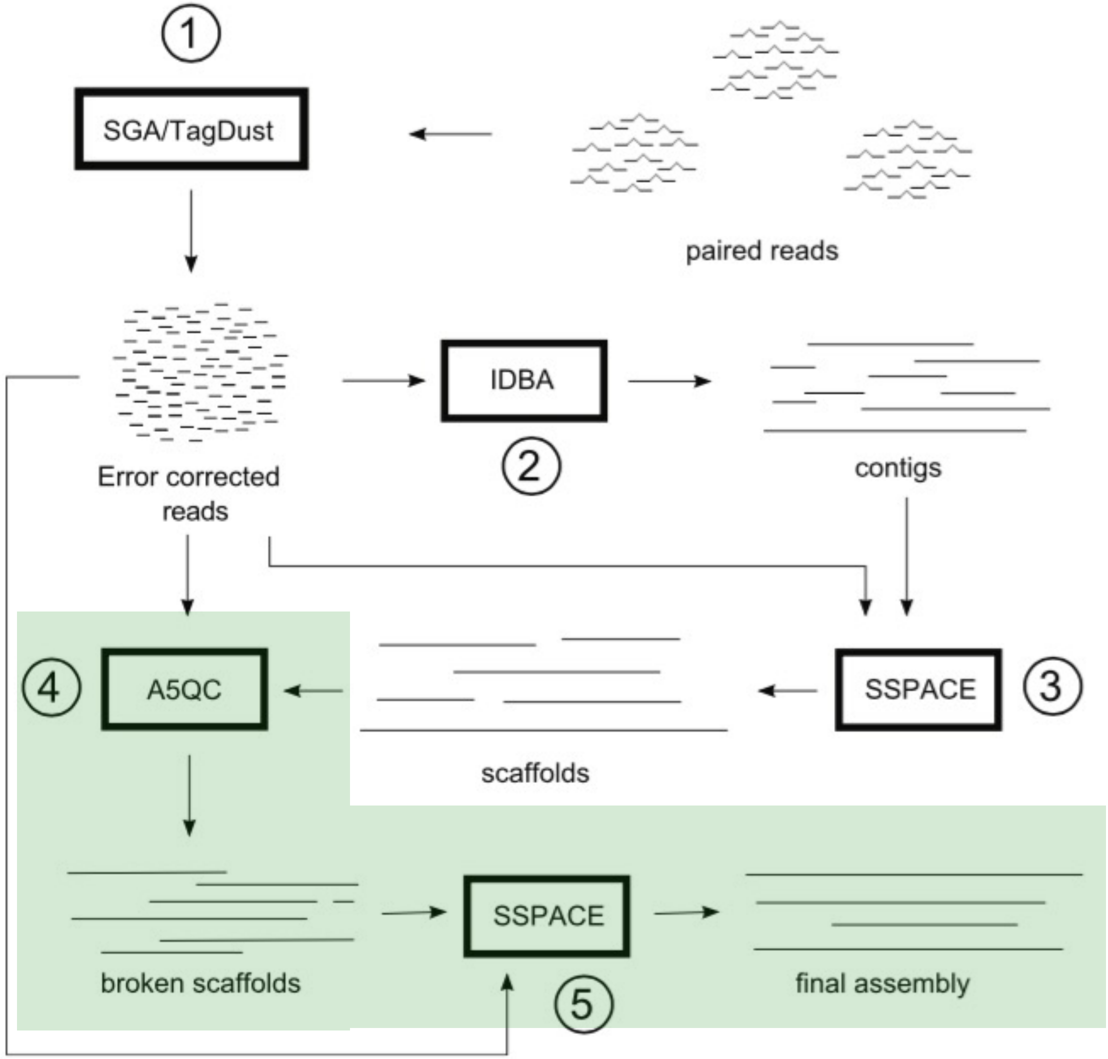
The A5 microbial genome assembly pipeline



No repeats suspected:
k-mers, de Bruijn graphs, Eulerian paths, mate-pairs

Tritt *et al*. 2012. *PLoS One*. 7:e42304.

The A5 microbial genome assembly pipeline

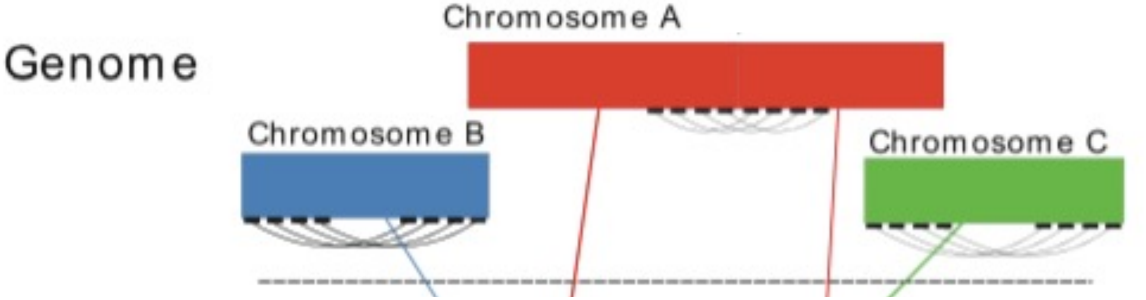


Repeats suspected:
library size estimation;
bad mate-pair geometry

Tritt *et al*. 2012. *PLoS One*. 7:e42304.

A5QC / SSPACE ‘Finishing’

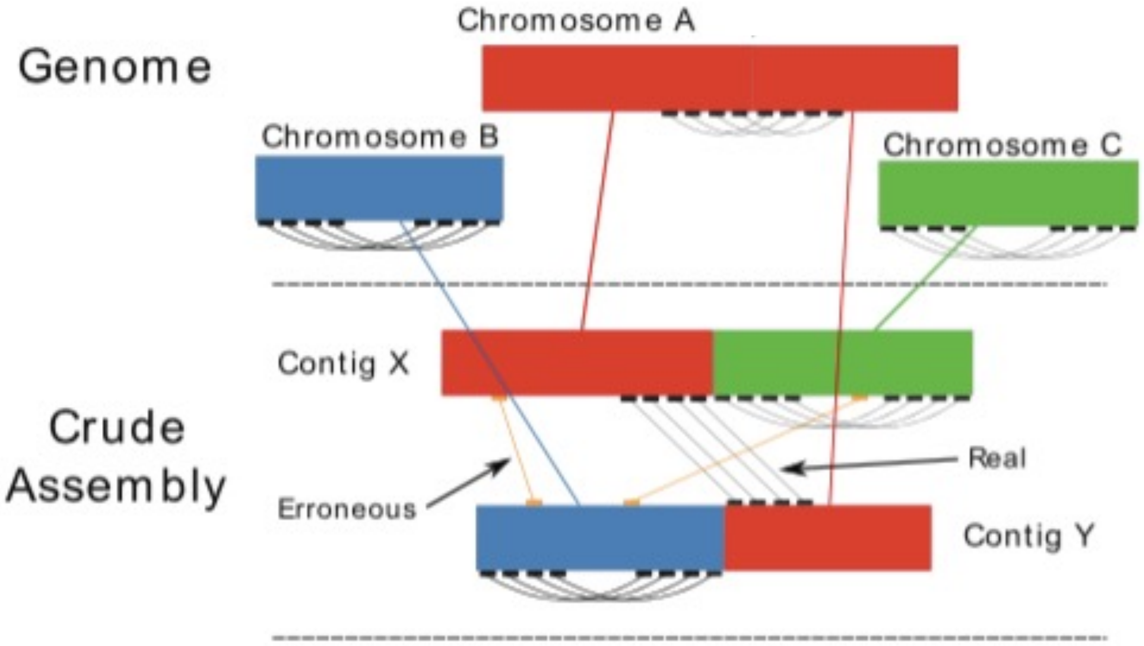
library size estimation; bad mate-pair geometry



Tritt *et al*. 2012. *PLoS One*. 7:e42304.

A5QC / SSPACE 'Finishing'

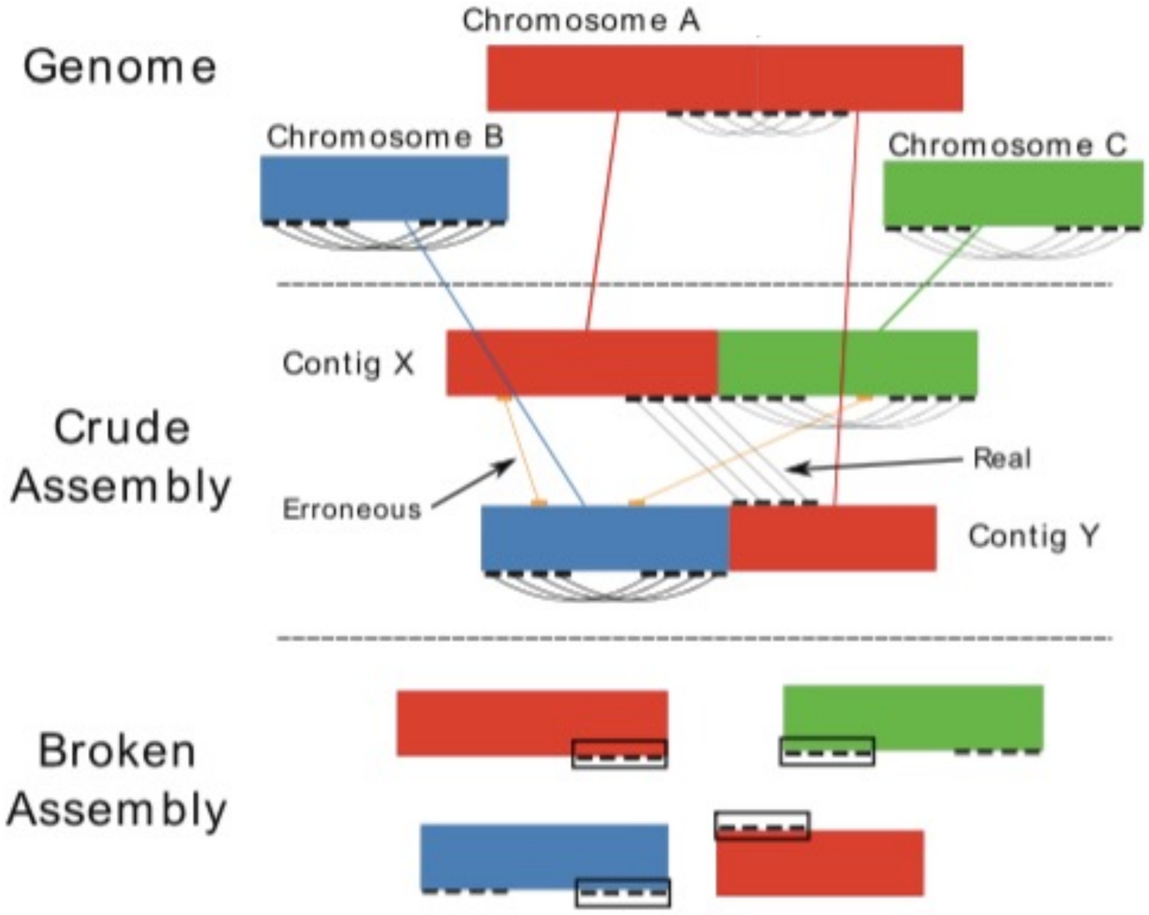
library size estimation; bad mate-pair geometry



Tritt *et al*. 2012. *PLoS One*. 7:e42304.

A5QC / SSPACE 'Finishing'

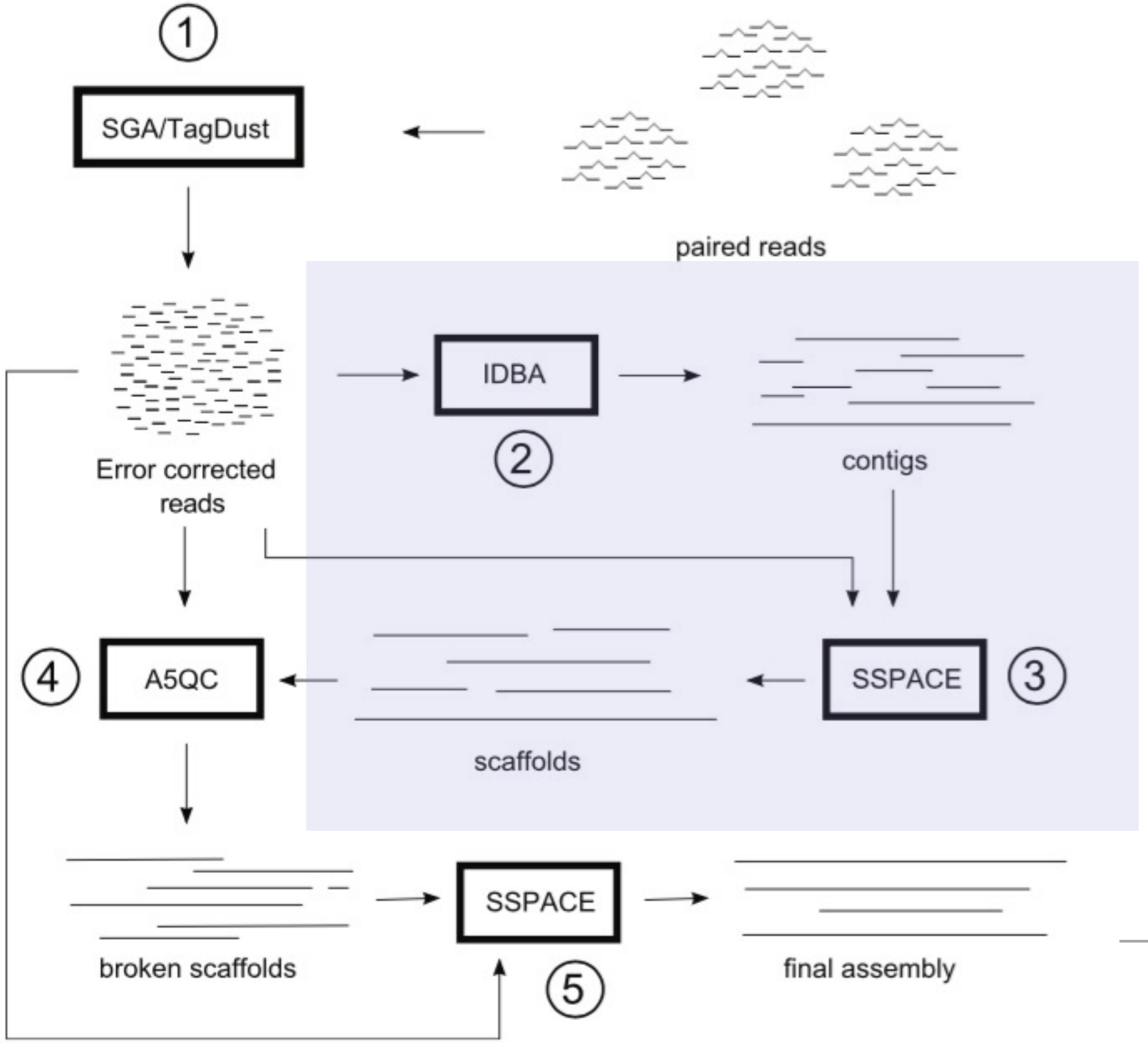
library size estimation; bad mate-pair geometry



Contigs vs. chromosomes!

Tritt *et al*. 2012. *PLoS One*. 7:e42304.

The A5 microbial genome assembly pipeline

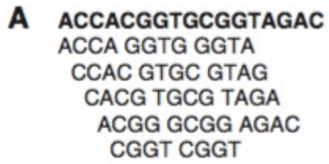


k-mers, de Bruijn graphs, Eulerian paths, mate-pairs

Tritt *et al*. 2012. *PLoS One*. 7:e42304.

kmers, de Bruijn graphs, Eulerian paths

- There are a wide variety of k-mer assemblers (e.g., Velvet, A5, SPADES, etc.)
- Some are generalist others are specialist (e.g., A5 is microbial)
- The SPAdes assembler is considered the best microbial genome assembler – uses multiple k-mer sizes
- You have seen k-mers before – BLAST word sizes
- A k-mer frequency spectrum is generated for a specific k-mer size

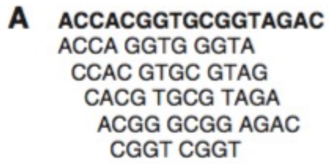


```
ACCACGGTGCGGTAGAC
ACCA GGTG GGTA
 CCAC GTGC GTAG
  CACG TGCG TAGA
   ACGG GCGG AGAC
    CGGT CGGT
```

Figure 3: (**A**) *k*-mer spectrum of a DNA string (bold) for $k=4$; (**B**) Section of the corresponding deBruijn graph. The edges are labeled with the corresponding *k*-mer and (**C**) Overlap between two reads (bold) that can be inferred from the corresponding paths through the deBruijn graph.

kmers, de Bruijn graphs, Eulerian paths

- Creation and use of k-mer indices is the most efficient method for handling the large volume of NGS data – less memory!



```
A  ACCACGGTGCGGTAGAC
   ACCA GGTG GGTA
    CCAC GTGC GTAG
     CACG TGCG TAGA
      ACGG GCGG AGAC
       CGGT CGGT
```

Figure 3: (**A**) *k*-mer spectrum of a DNA string (bold) for $k=4$; (**B**) Section of the corresponding deBruijn graph. The edges are labeled with the corresponding *k*-mer and (**C**) Overlap between two reads (bold) that can be inferred from the corresponding paths through the deBruijn graph.

kmers, de Bruijn graphs, Eulerian paths

- Creation and use of k-mer indices is the most efficient method for handling the large volume of NGS data – less memory!
- A deBruijn graph is created from the k-mers of all reads
    - all k-mers must overlap adjoining k-mers in the genome by *k*-1

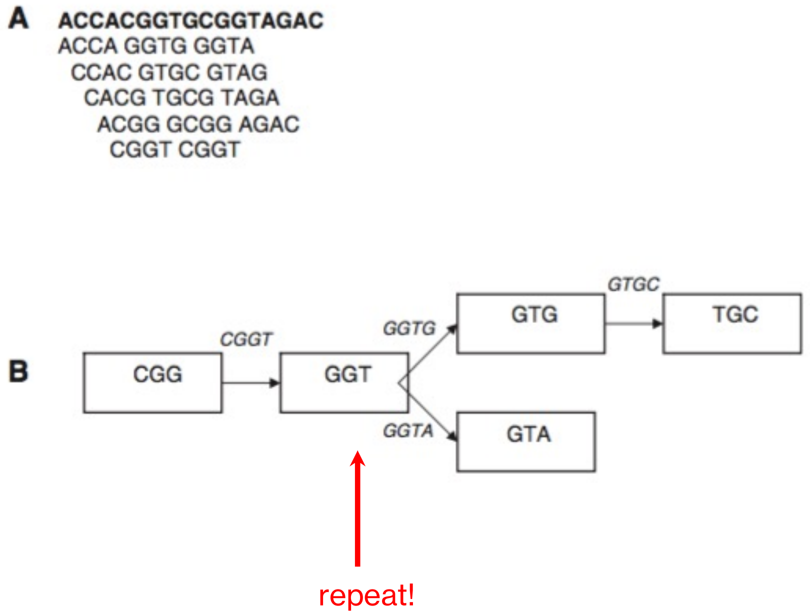


Figure 3: (**A**) *k*-mer spectrum of a DNA string (bold) for *k* = 4; (**B**) Section of the corresponding deBruijn graph. The edges are labeled with the corresponding *k*-mer and (**C**) Overlap between two reads (bold) that can be inferred from the corresponding paths through the deBruijn graph.

kmers, de Bruijn graphs, Eulerian paths

- Creation and use of k-mer indices is the most efficient method for handling the large volume of NGS data – less memory!
- A deBruijn graph is created from the k-mers of all reads
  - all k-mers must overlap adjoining k-mers in the genome by *k*-1
- A deBruijn graph can be used to efficiently find read overlap for forming contigs & resolving repeated sequences
  - graph theory – the optimal Eulerian path visits every edge exactly once
  - repeat regions produce over-abundant k-mers and have identifiable deBruijn graph properties

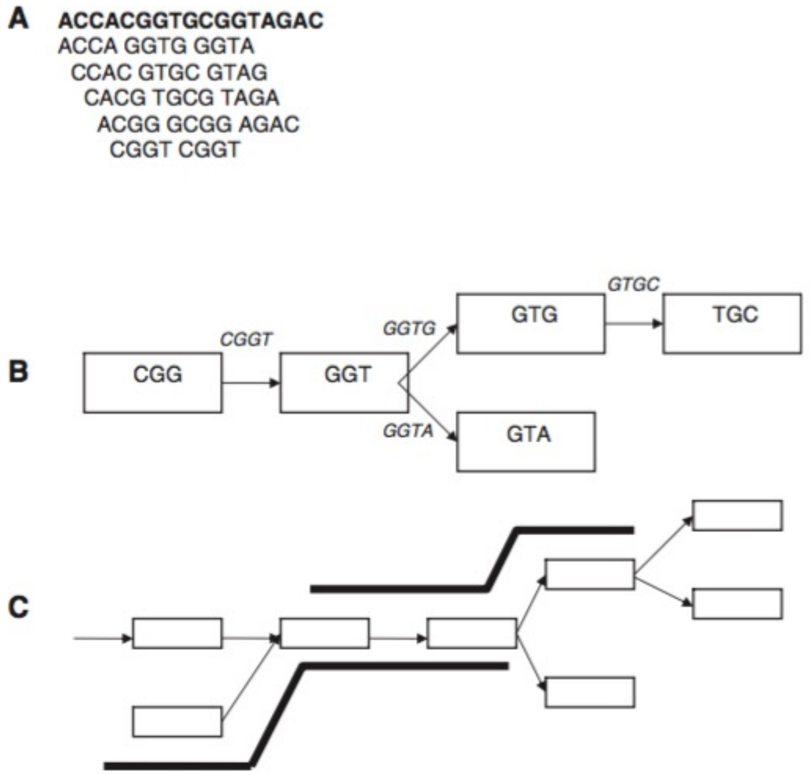


Figure 3: (**A**) *k*-mer spectrum of a DNA string (bold) for *k* = 4; (**B**) Section of the corresponding deBruijn graph. The edges are labeled with the corresponding *k*-mer and (**C**) Overlap between two reads (bold) that can be inferred from the corresponding paths through the deBruijn graph.

kmers, de Bruijn graphs, Eulerian paths

- Sequencing error creates ‘novel’ k-mers and complicates the deBruijn graph – **genome assembly is thus always preceded by an error trimming & correction step**

- The product of k-mer assembly is a robust set of contigs, but what k-mer size to use?
  - Shorter – less memory, more complex de Bruijn graph, difficulty with small tandem repeats
  - Longer – more memory, simpler de Bruijn graph, overcome small repeats, upper maximum due to sequencing gaps, error filtering critical

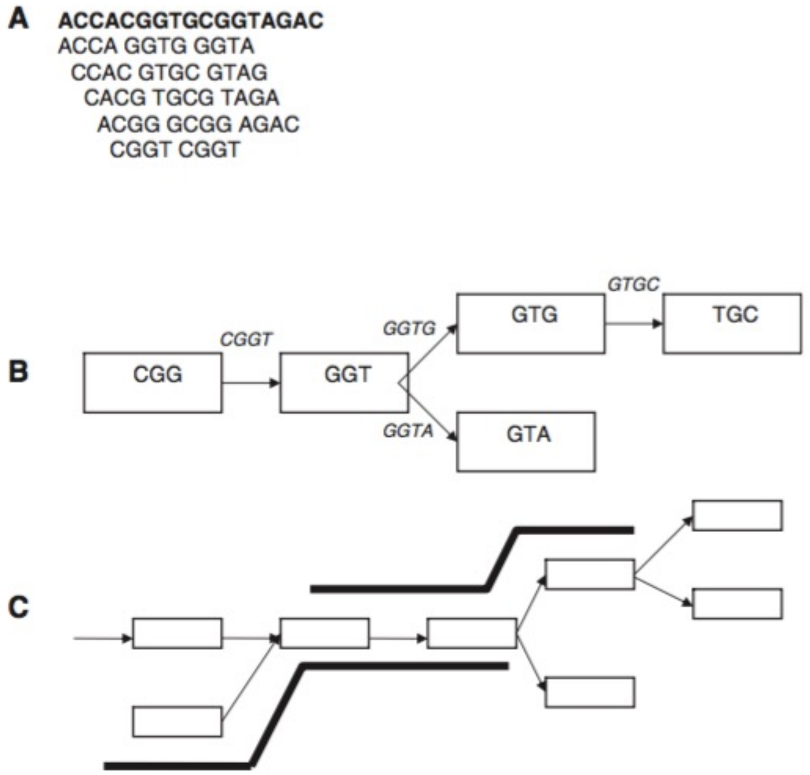
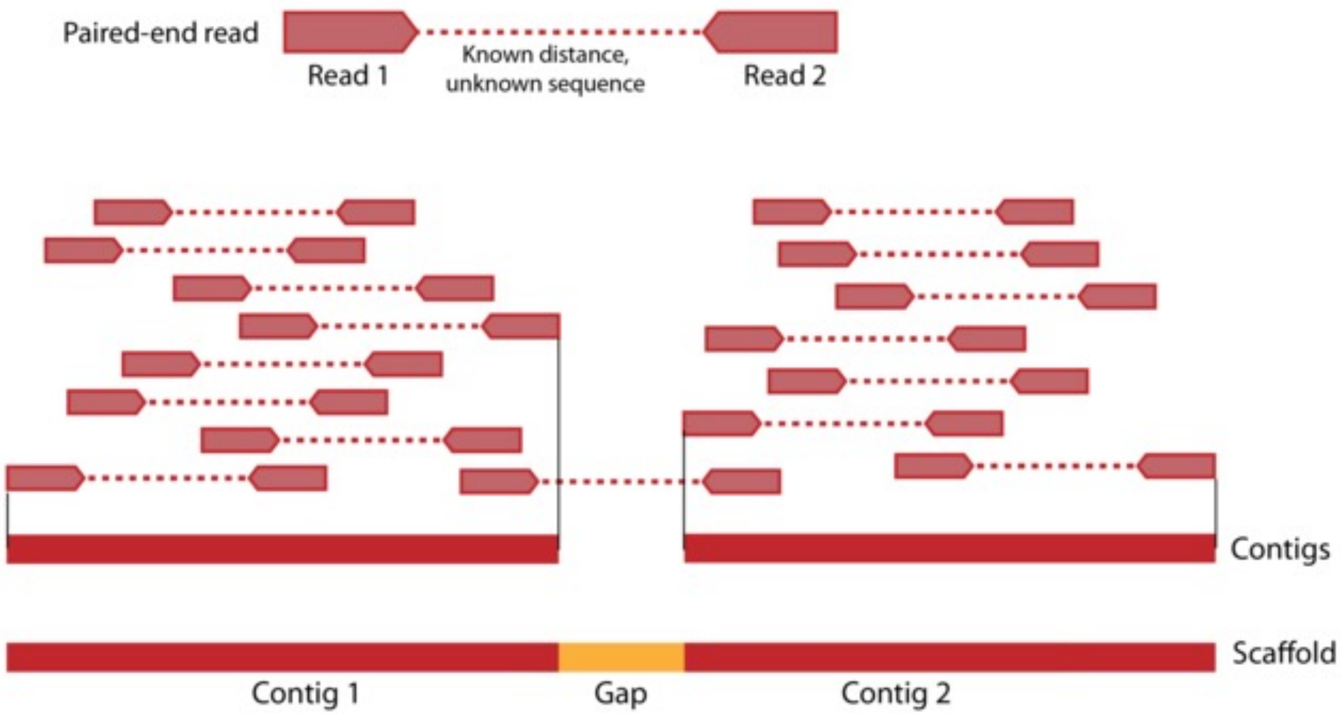


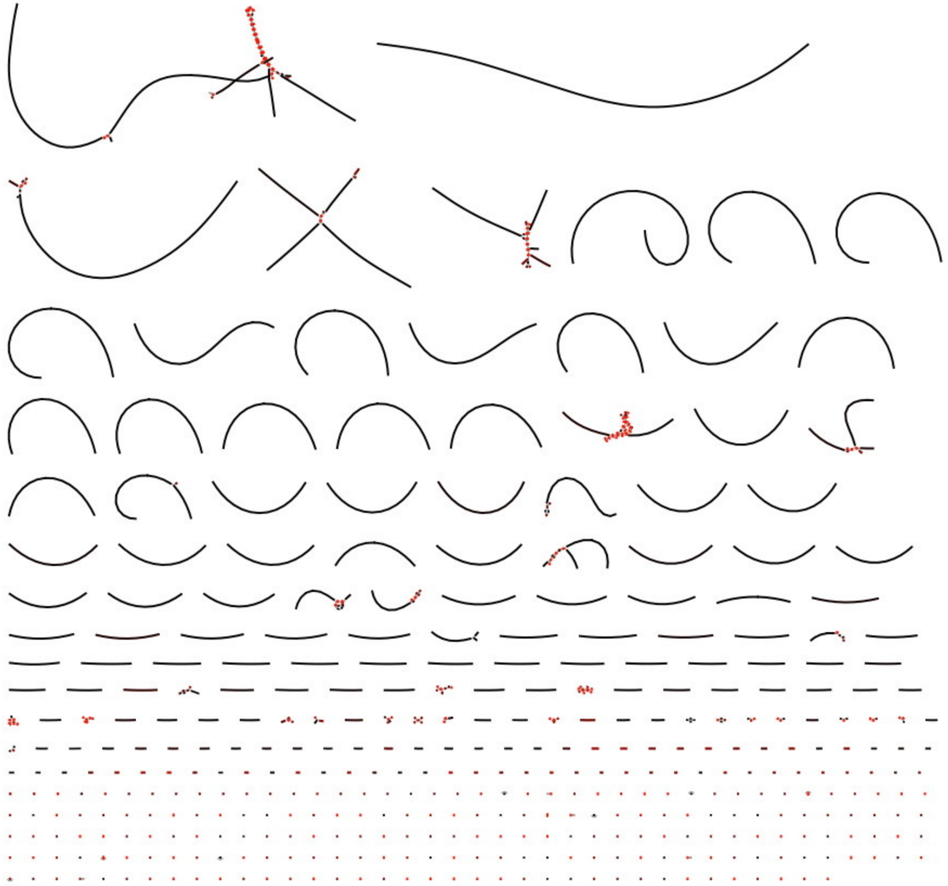
Figure 3: (**A**) *k*-mer spectrum of a DNA string (bold) for $k=4$; (**B**) Section of the corresponding deBruijn graph. The edges are labeled with the corresponding *k*-mer and (**C**) Overlap between two reads (bold) that can be inferred from the corresponding paths through the deBruijn graph.

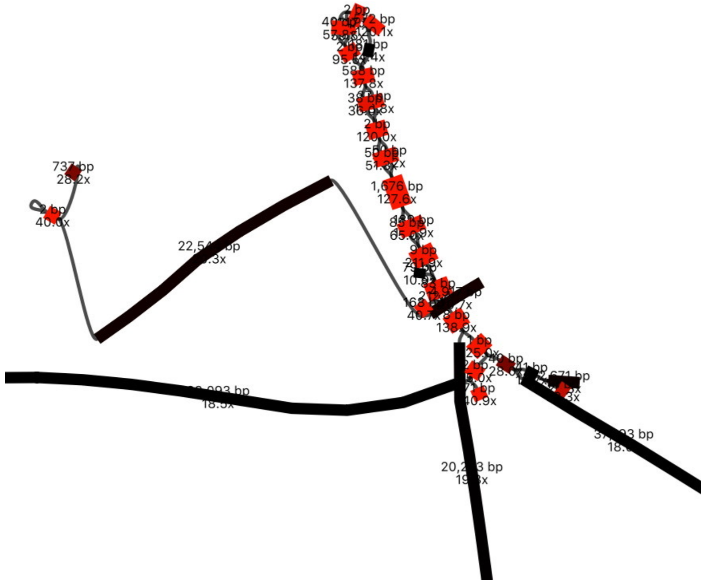
kmers, de Bruijn graphs, Eulerian paths

- K-mer assembly produces ‘**contigs**’ – stretches of consensus sequences
- More read coverage leads to denser de Bruijn graphs, fewer & bigger contigs, less assembly gaps
- Mate-pair information can then be used to determine **scaffolds** – the order of contigs along the genome
- Scaffolding algorithms are greedy too – de Bruijn graphs (Velvet) or secondary algorithms (A5) seek to limit scaffolding errors



A complete assembly...





BANDAGE visualization

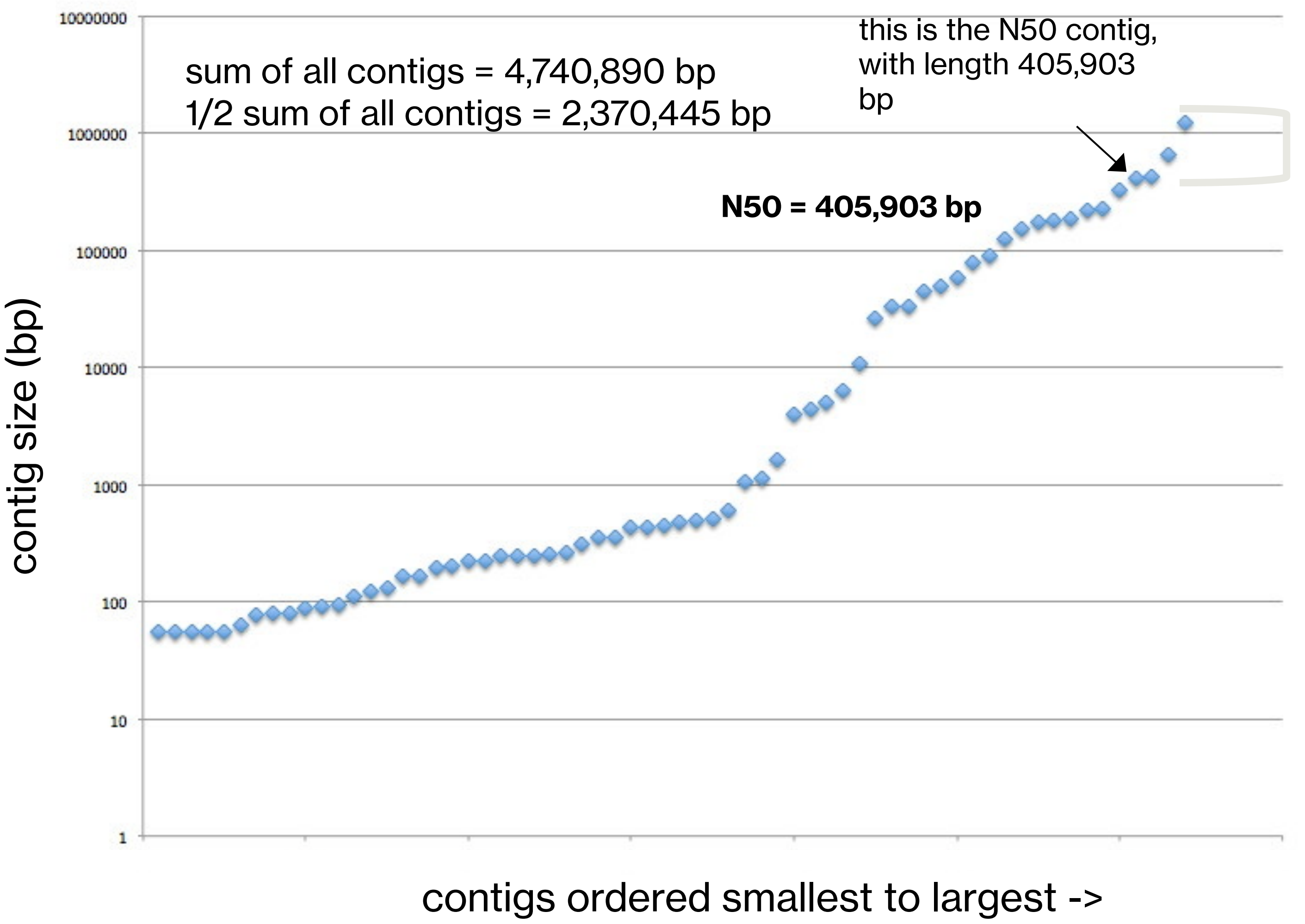
Assembly Statistics – A5 example

- McMaster *E. coli* C008 strain with Illumina HiSeq 2 x 250 bp sequencing
- Raw 3,735,008 sequencing reads totaling 933,752,000 bp
- ~200 fold coverage of the *E. coli* genome (*E. coli* K-12 is 4,639,221 bp)
- 3,666,906 error-corrected reads totaling 797,138,877 bp
- 98.18% of reads passed error-correction
- 85.03% of nucleotides passed error-correction
- 118 contigs in 118 scaffolds
- 5,247,627 bp sum contig length
- bases ≥ Q40 = 5,243,991 (99.9% of assembly; Q40 = 1 in 10,000 error rate)
- Longest scaffold = 534,047 bp
- Contig N50 = 166,170 bp (at least 50% of the assembly is contained in contigs of this size or larger)
- GC content = 50.6%
- Observed read coverage = 178.65 fold
- Median = 154 fold; 10th percentile = 97 fold

For a nice review of A5 statistics, see http://tinyurl.com/zwng5cb

N50

N50 is the size of the contig which, along with all larger contigs, contains half of sequence of an assembly.



sum of all contigs = 4,740,890 bp
1/2 sum of all contigs = 2,370,445 bp

this is the N50 contig, with length 405,903 bp

N50 = 405,903 bp

top contig = 1,223,670 bp

top 2 contigs sum = 1,871,154 bp

top 3 contigs sum = 2,299,813 bp

top 4 contigs contain 1/2 of the assembly, sum = 2,705,716 bp

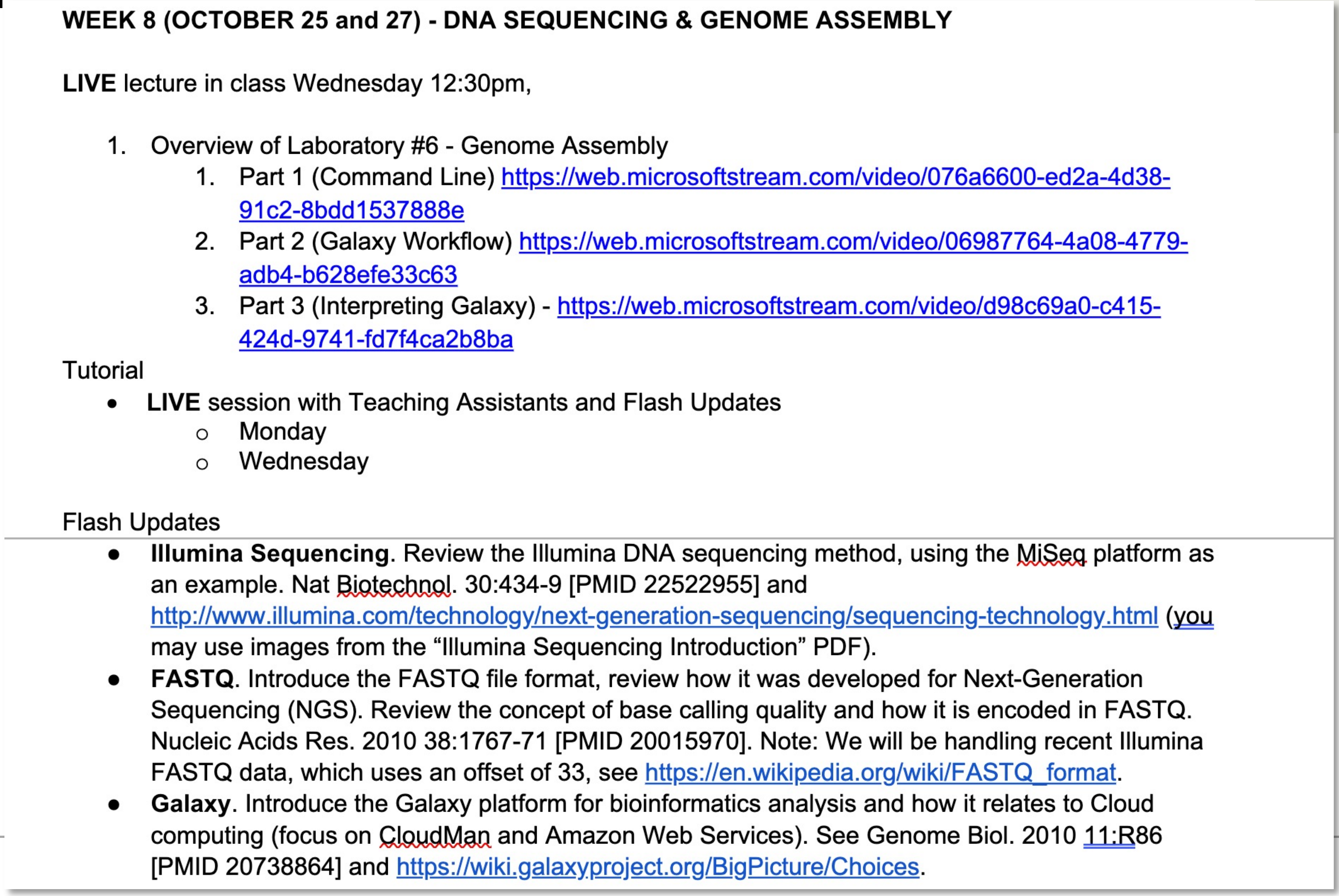
Finishing, Validation, Confidence

- ‘Finishing’ traditionally refers to the costly and labourious steps required to ‘close’ all the sequencing gaps to provide high quality and 100% complete chromosome sequences (aka “closure”)
- Finishing is workable for bacterial genomes but very hard for eukyarotic genomes
    - centromeres and teleomeres are hard to clone or sequence
    - yet important biology is encoded in these regions (e.g. *Trypanosoma*)
- Most current genome projects do not attempt closure due to:
    - the cost and time involved – closure does not get funded
    - shotgun routinely obtains >90% closure and thus the majority of the biology
    - gaps will be closed by research teams if it is relevant to their science
- ‘Finishing’ now most often refers to the scaffolding quality control steps in genome assembly – making the most out of the shotgun data – and is seen as distinct from ‘closure’

Finishing, Validation, Confidence

- Validation of genome assembly is difficult as our knowledge about the genome we are sequencing is often limited
- If a closely related genome sequence is known, a comparison can be made to determine possible errors – or are they real differences?
- Genome annotation can identify gaps in the assembly (e.g. missing genes known from PCR or biochemistry)
- With limited prior knowledge, bioinformaticians rely on the assembly statistics such as Q40, N50, etc.
- But can we trust the assembly software?
    - Peer-reviewed publication of algorithms and open source release of software
    - Head-to-head comparison of assemblers on the same data to identify consensus
    - Simulated data and Assemblathon / GAGE competitions

This week...



WEEK 8 (OCTOBER 25 and 27) - DNA SEQUENCING & GENOME ASSEMBLY

LIVE lecture in class Wednesday 12:30pm,

1. Overview of Laboratory #6 - Genome Assembly
    1. Part 1 (Command Line) https://web.microsoftstream.com/video/076a6600-ed2a-4d38-91c2-8bdd1537888e
    2. Part 2 (Galaxy Workflow) https://web.microsoftstream.com/video/06987764-4a08-4779-adb4-b628efe33c63
    3. Part 3 (Interpreting Galaxy) - https://web.microsoftstream.com/video/d98c69a0-c415-424d-9741-fd7f4ca2b8ba

Tutorial

- **LIVE** session with Teaching Assistants and Flash Updates
    - Monday
    - Wednesday

Flash Updates

- **Illumina Sequencing**. Review the Illumina DNA sequencing method, using the MiSeq platform as an example. Nat Biotechnol. 30:434-9 [PMID 22522955] and http://www.illumina.com/technology/next-generation-sequencing/sequencing-technology.html (you may use images from the “Illumina Sequencing Introduction” PDF).
- **FASTQ**. Introduce the FASTQ file format, review how it was developed for Next-Generation Sequencing (NGS). Review the concept of base calling quality and how it is encoded in FASTQ. Nucleic Acids Res. 2010 38:1767-71 [PMID 20015970]. Note: We will be handling recent Illumina FASTQ data, which uses an offset of 33, see https://en.wikipedia.org/wiki/FASTQ_format.
- **Galaxy**. Introduce the Galaxy platform for bioinformatics analysis and how it relates to Cloud computing (focus on CloudMan and Amazon Web Services). See Genome Biol. 2010 11:R86 [PMID 20738864] and https://wiki.galaxyproject.org/BigPicture/Choices.