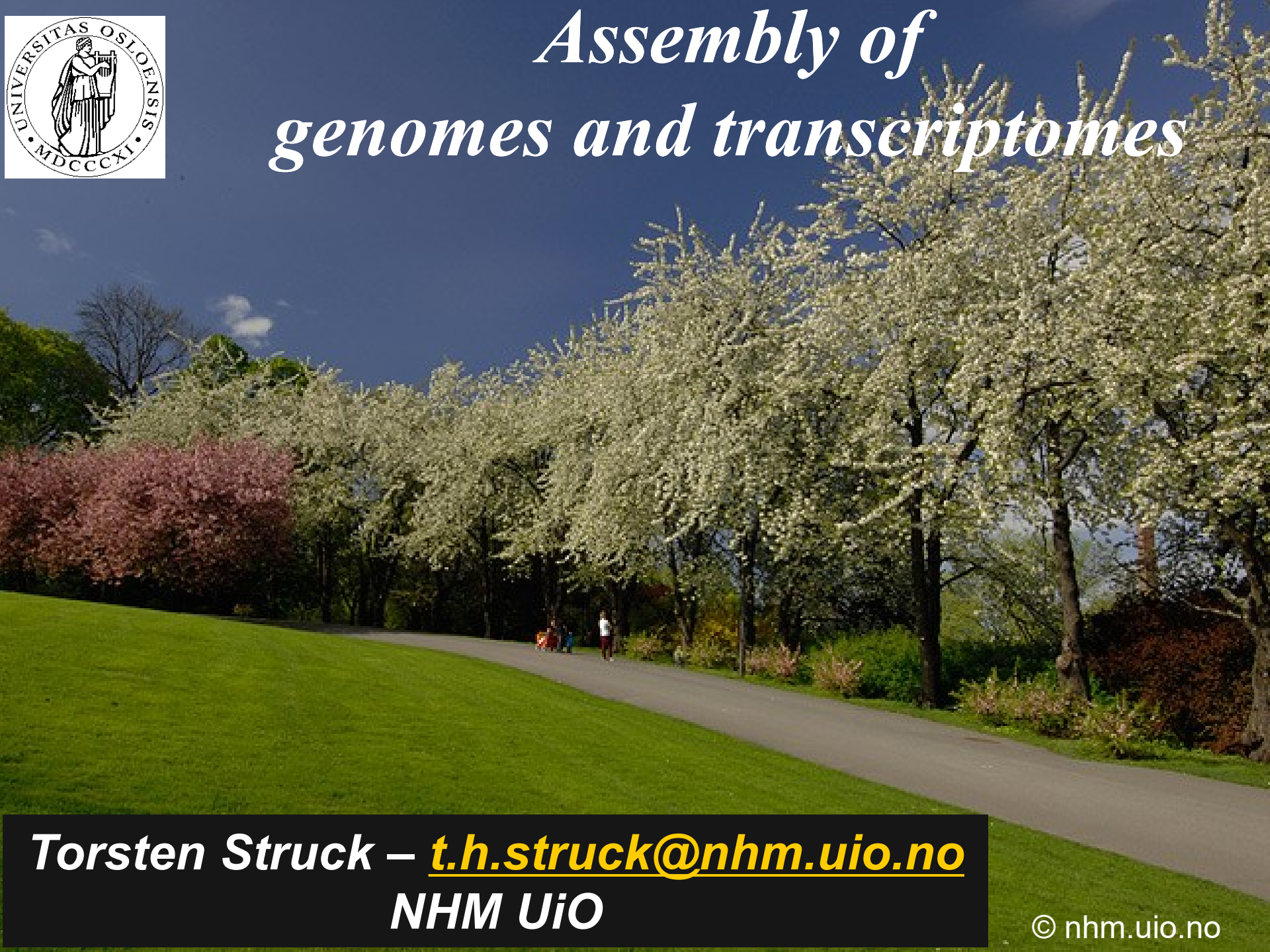
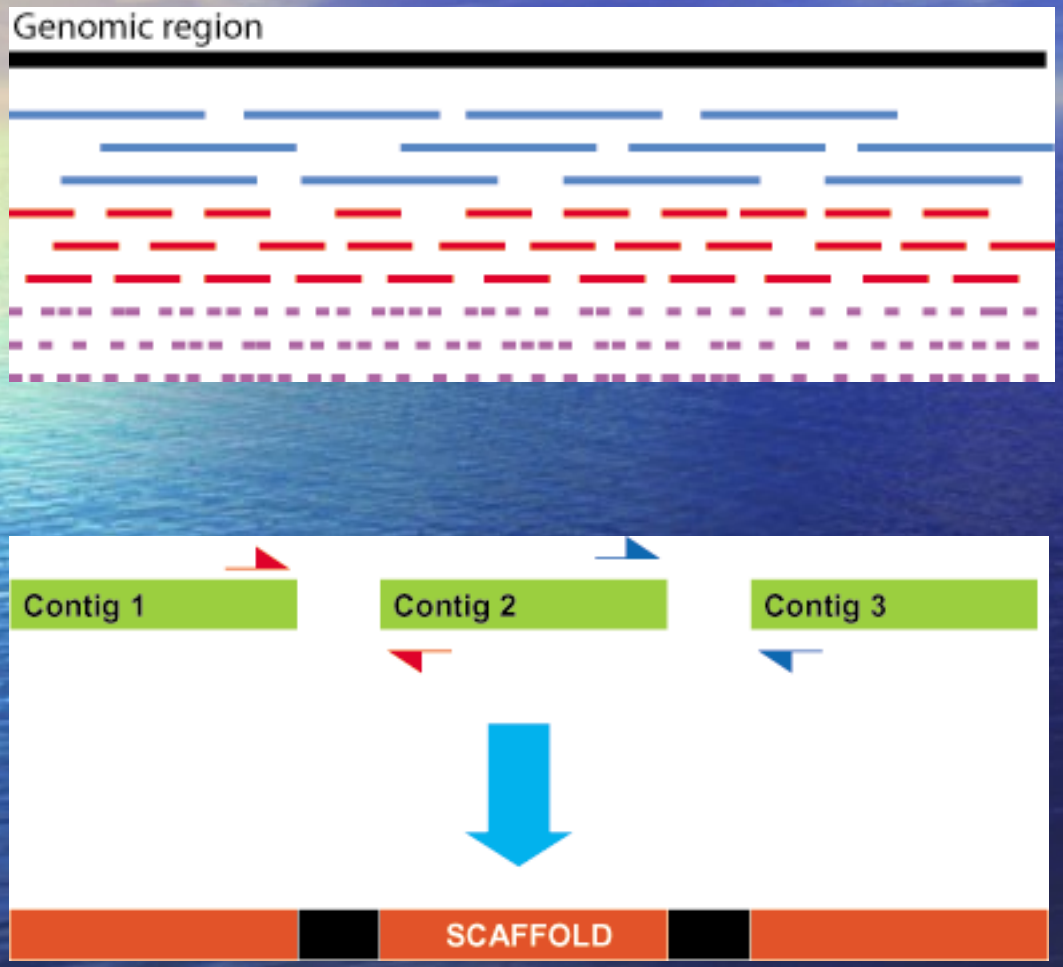


Assembly of genomes and transcriptomes

Torsten Struck – t.h.struck@nhm.uio.no
NHM UiO

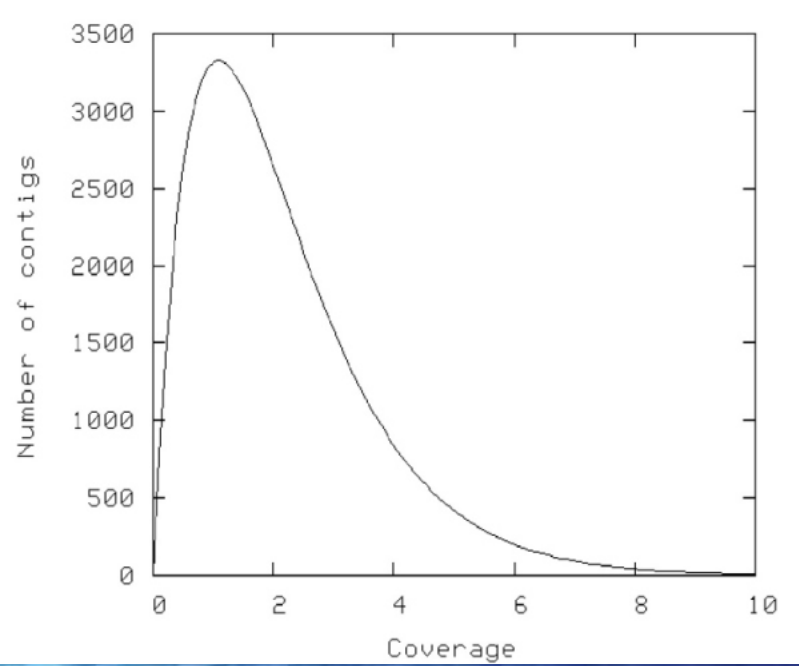
The problem

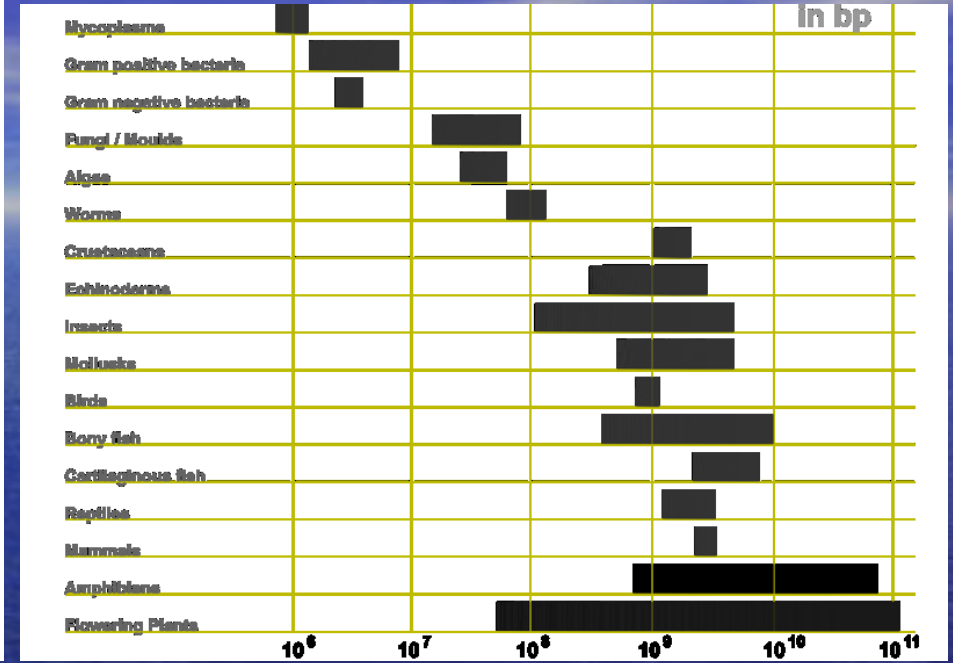

Genomic region
Contig 1
Contig 2
Contig 3
SCAFFOLD
Sequencing strategy
Assembly goal

The problematic issues

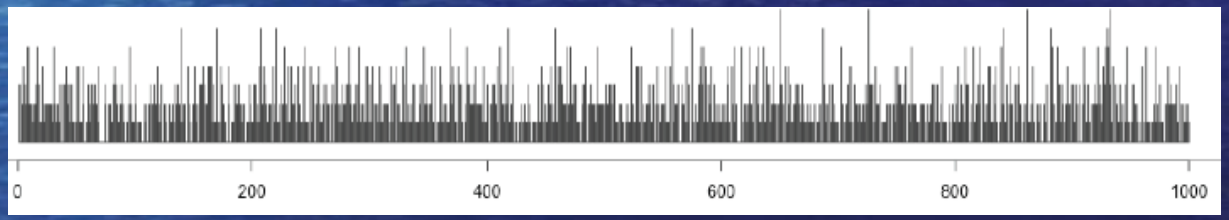
sequencing errors
ambiguities
uneven coverage
complexity of the genome: large in size
non-random sequence composition
repetitive elements
often longer than existing read lengths

Coverage

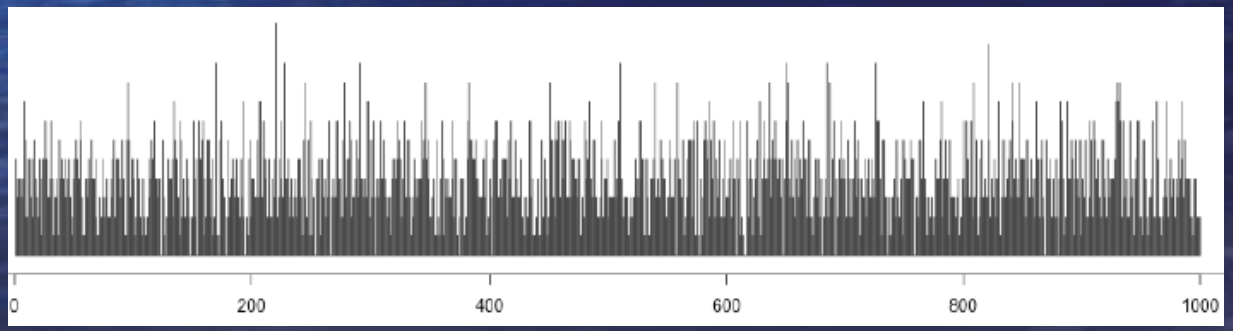




2x



8x

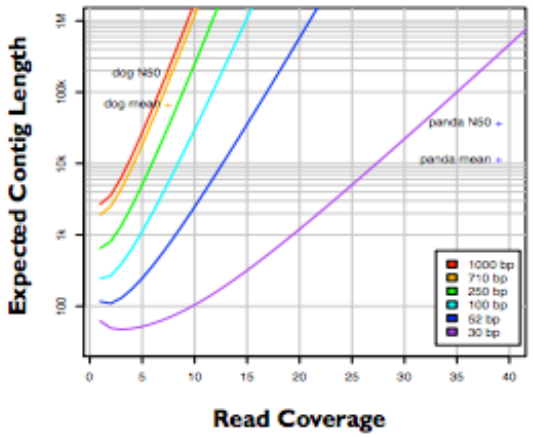


Schatz, 2014

Coverage

Ingredients for a good assembly

Coverage



High coverage is required

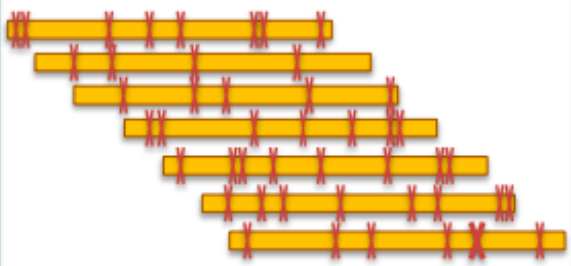
- Oversample the genome to ensure every base is sequenced with long overlaps between reads
- Biased coverage will also fragment assembly

Read Length

Reads & mates must be longer than the repeats

- Short reads will have ***false overlaps*** forming hairball assembly graphs
- With long enough reads, assemble entire chromosomes into contigs

Quality



Errors obscure overlaps

- Reads are assembled by finding kmers shared in pair of reads
- High error rate requires very short seeds, increasing complexity and forming assembly hairballs

Schatz et al. (2012) Genome Biology

Principles

intuitively obvious assumptions

if two sequence reads share a common overlapping substring of letters, then it is because they are likely to have originated from the same chromosomal regions in the genome

the estimated distance between two reads provides additional information

once such overlap structures among the sequence reads are determined, the assembler places the reads in a layout and combines the reads together to create a consensus sequence

greedy and graph-based

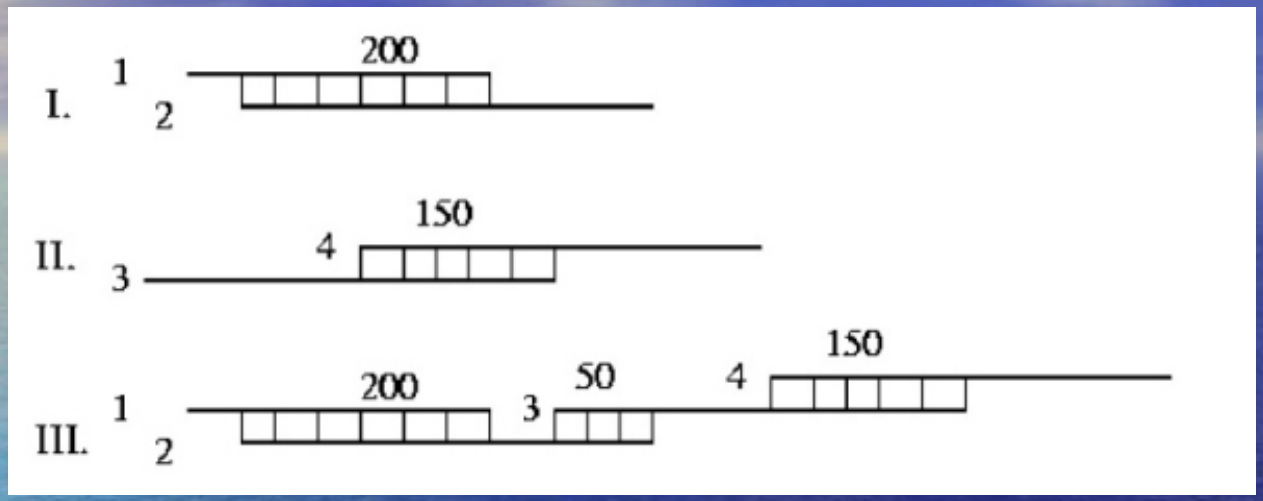
The greedy method

The greedy algorithm joins a sequence read with another read that has the best overlap score, then to the next read with the next best score until no more reads can be joined

The graph-based methods

These methods generate a graph using reads/substrings of reads and overlaps. The nodes of the graph are the reads/substrings, and the edges represent overlaps of them. In this way, the assembly process becomes synonymous with finding a pathway through the graph that visits every node exactly or at least once.

Greedy method

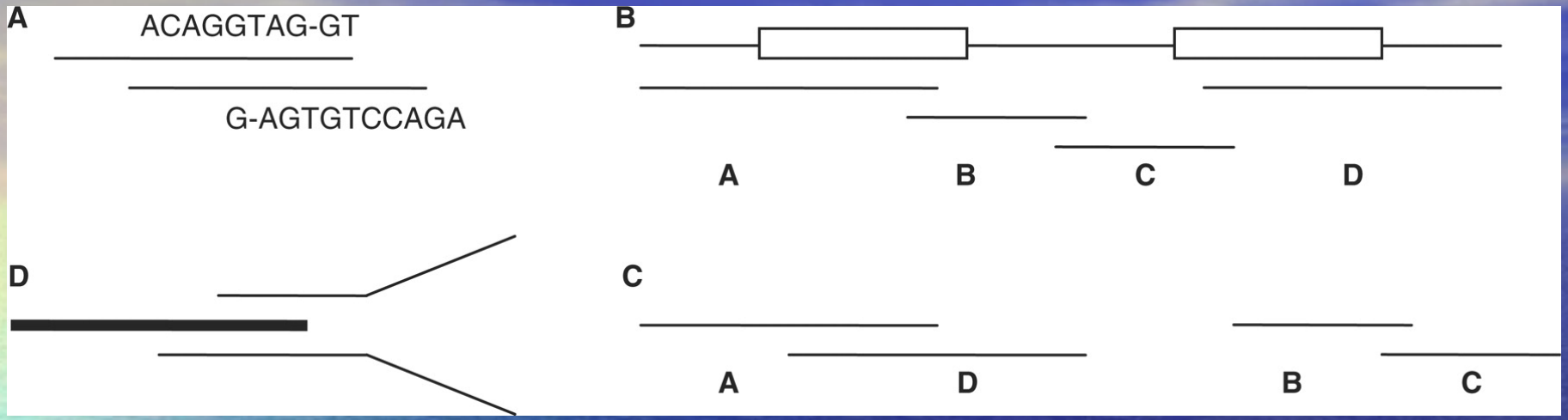


Greedy algorithms represent the simplest, most intuitive, solution to the assembly problem.

Individual reads are joined together into contigs in an iterative fashion, starting with the reads that overlap best, and ending once no more reads or contigs can be joined.

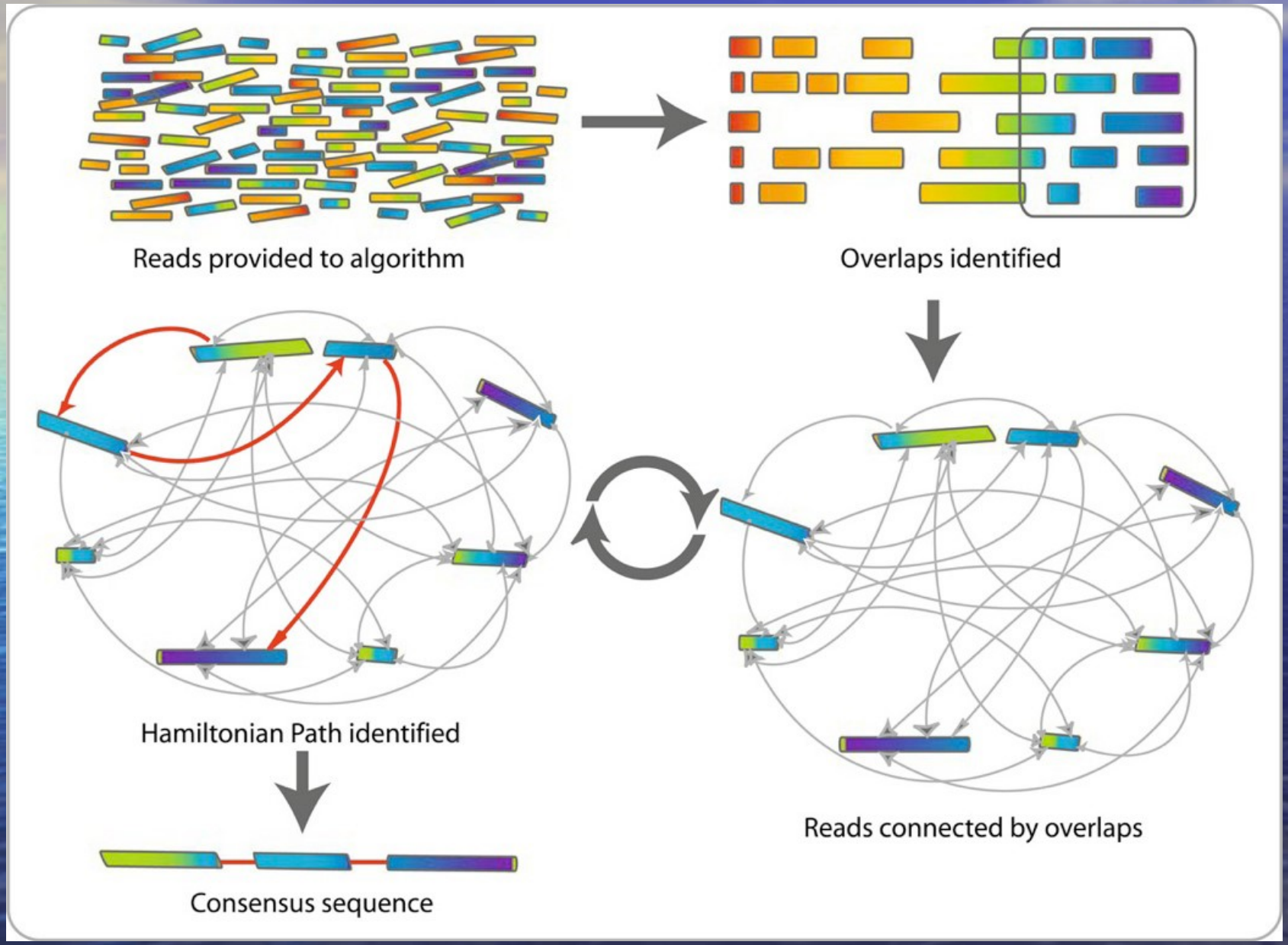
http://www.cbcb.umd.edu/research/assembly_primer.shtml

Greedy method



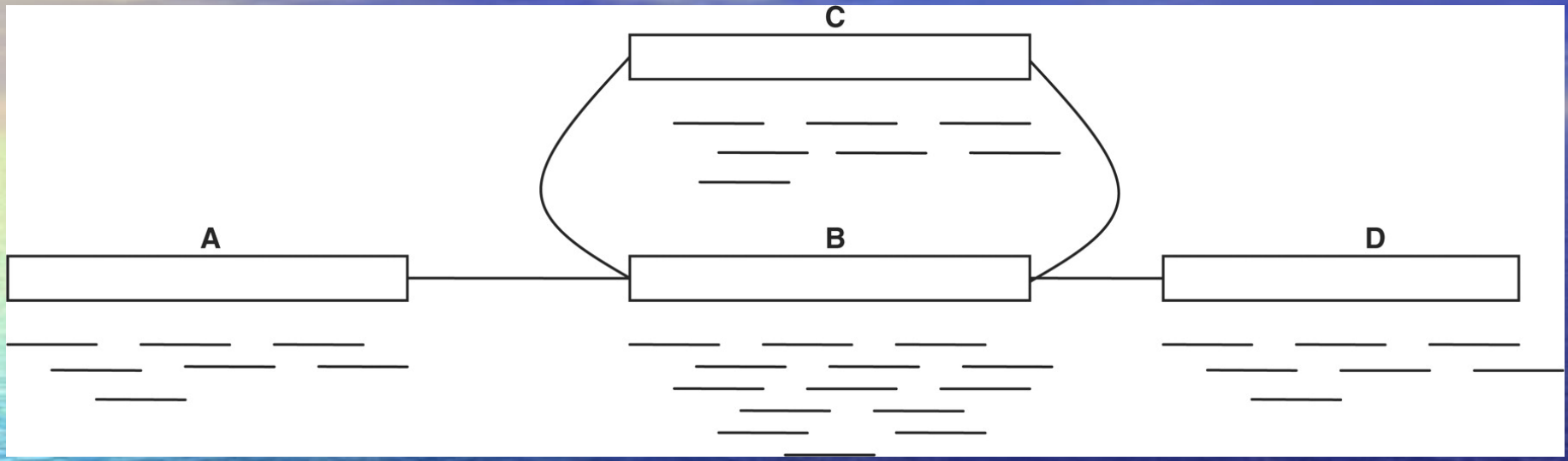
(A) Overlap between two reads - note that agreement within overlapping region need not be perfect;
(B) Correct assembly of a genome with two repeats (boxes) using four reads A–D;
(C) Assembly produced by the greedy approach. Reads A and D are assembled first, incorrectly, because they overlap best and
(D) Disagreement between two reads (thin lines) that could extend a contig (thick line), indicating a potential repeat boundary. Contig extension must be terminated in order to avoid misassemblies.

The OLC method



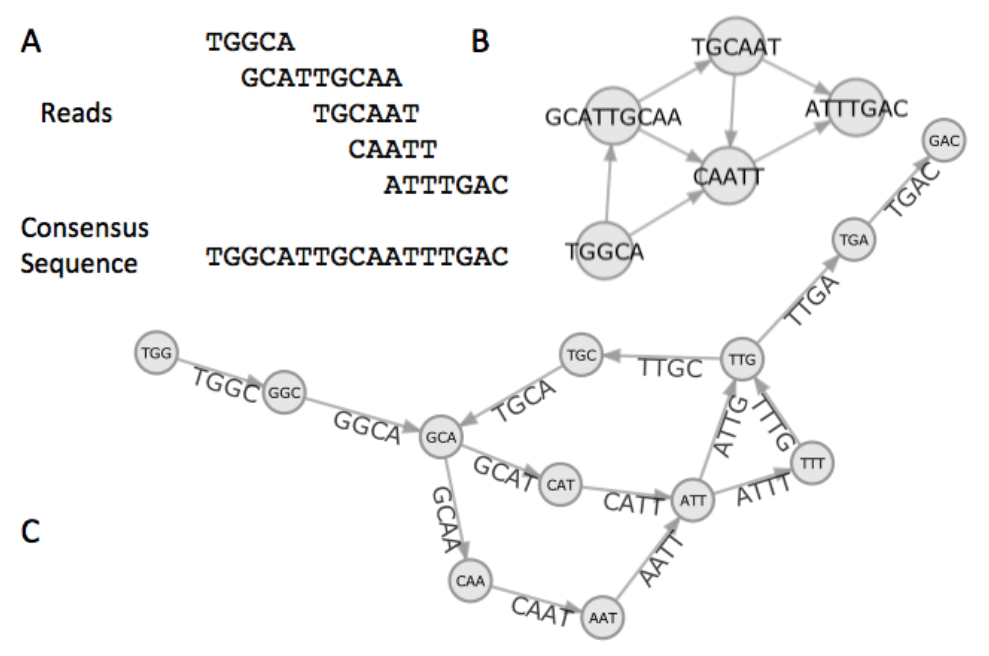
Commins et al. (2009) Biological Procedures Online.

The OLC method



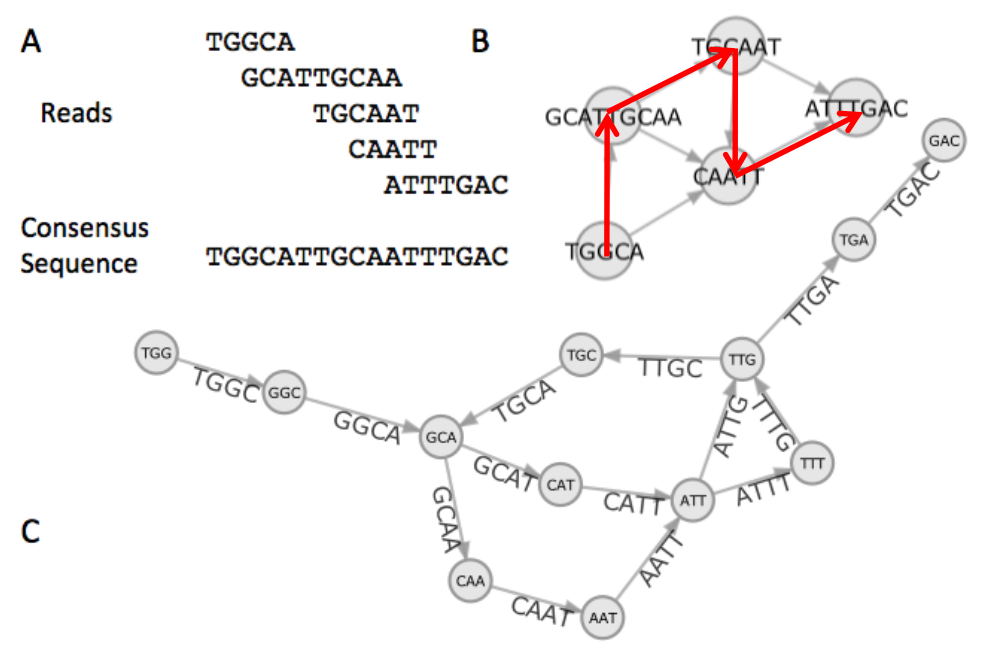
Overlap graph of a genome containing a two-copy repeat (B). Note the increased depth of coverage within the repeat. The correct reconstruction of this genome spells the sequence ABCBD, while conservative assembly approaches would lead to a fragmented reconstruction.

The de Bruijn method



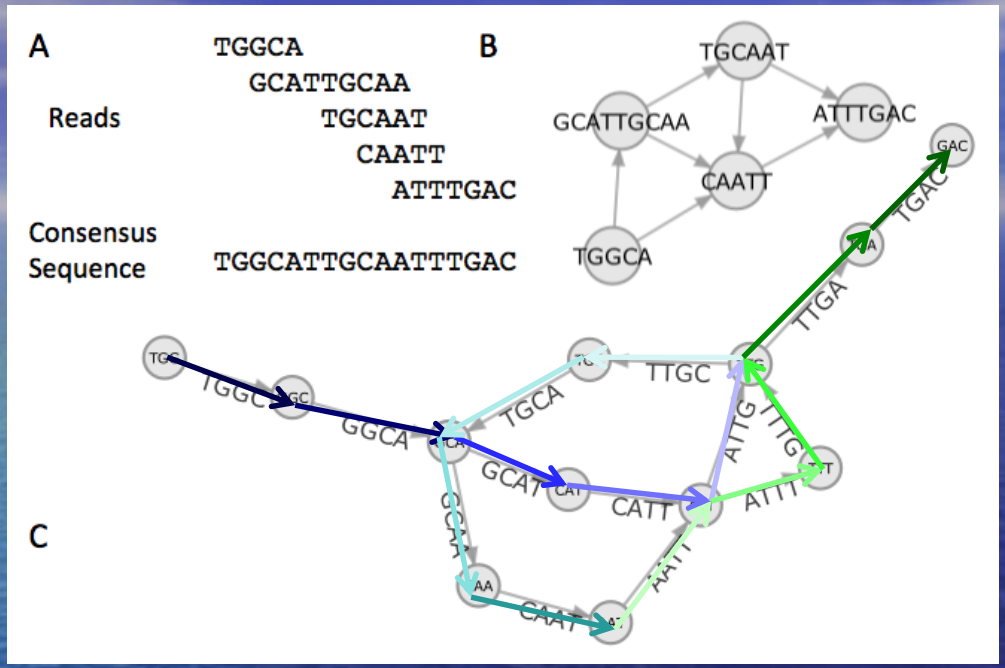
Reads and two possible assembly graphs.
A: Hypothetical reads aligned to the consensus sequence.
B: The OLC assembly graph created from these reads. Edges represent **overlaps** of 2 or more nt. The assembly process is to visit every **node**.

The de Bruijn method



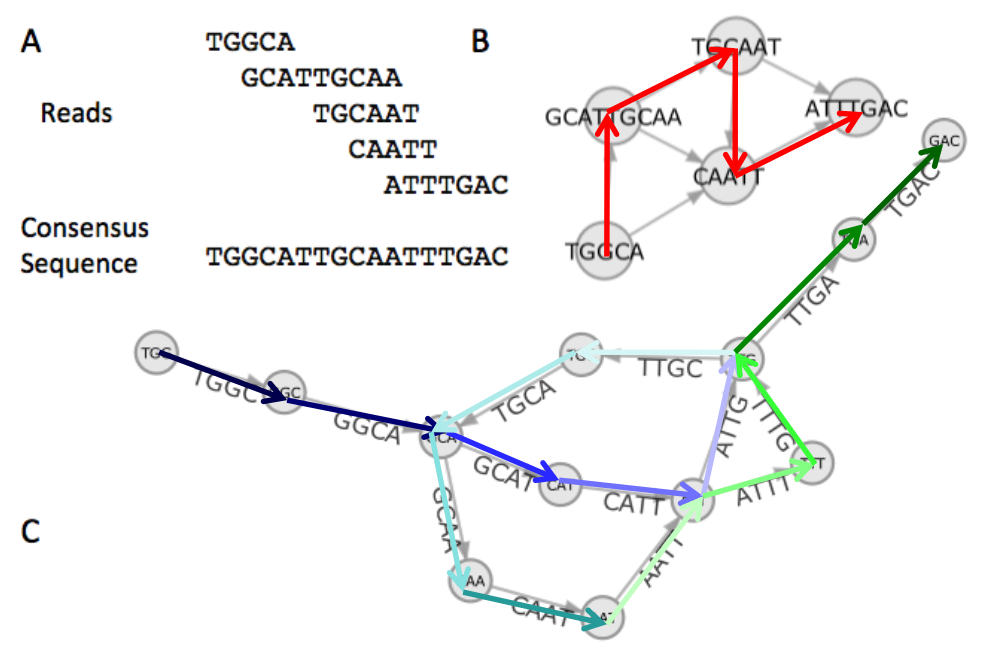
Reads and two possible assembly graphs.
A: Hypothetical reads aligned to the consensus sequence.
B: The OLC assembly graph created from these reads. Edges represent **overlaps** of 2 or more nt. The assembly process is to visit every **node**.

The de Bruijn method



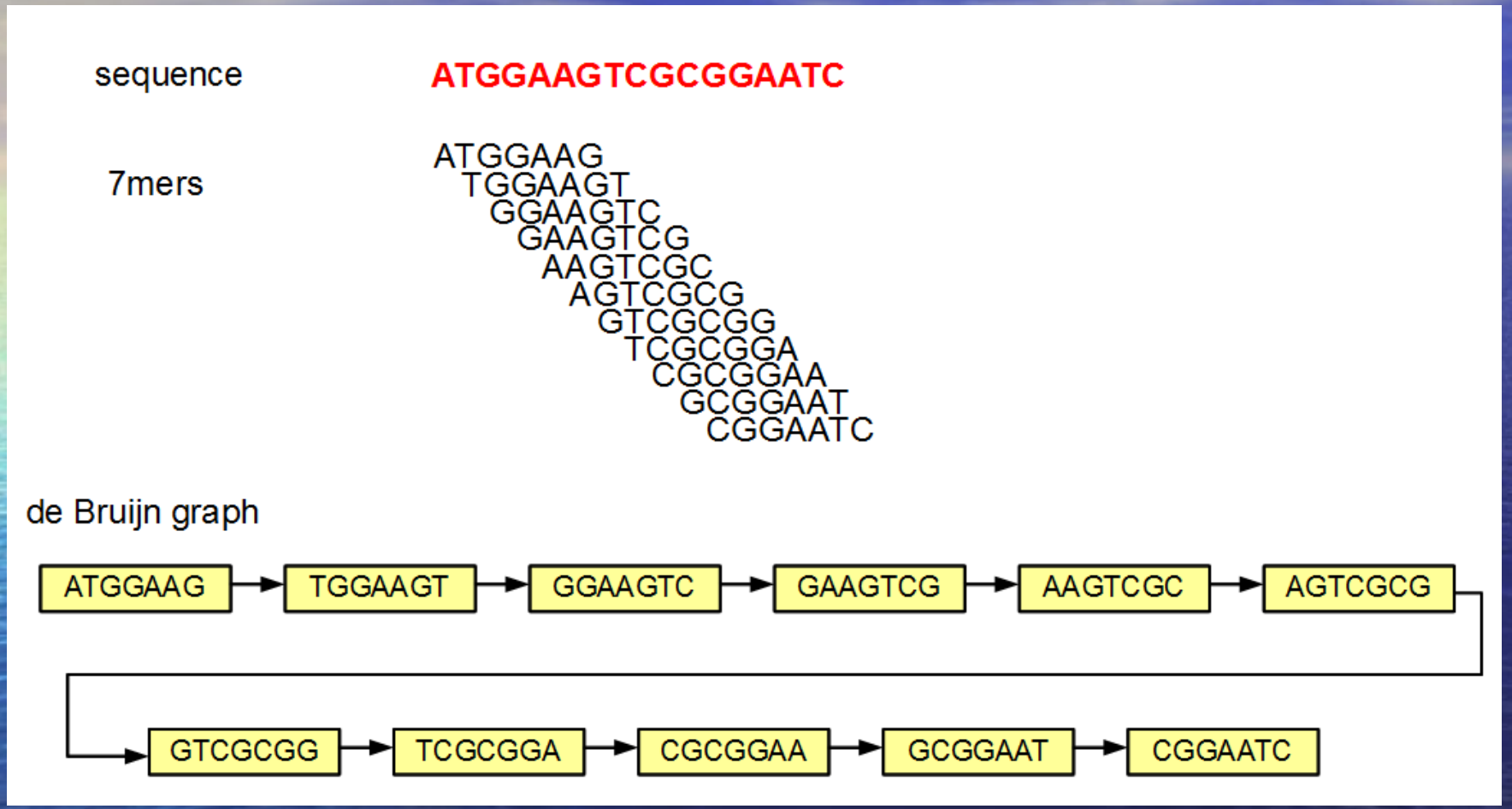
Reads and two possible assembly graphs.
A: Hypothetical reads aligned to the consensus sequence.
B: The OLC assembly graph created from these reads. Edges represent **overlaps** of 2 or more nt. The assembly process is to visit every **node**.
C: The de Bruijn assembly graph created from **substrings** of these reads. Edges represent 4 nt segments with 2 nt overlap between the nodes. The assembly process is to visit every **edge**.

The de Bruijn method



Reads and two possible assembly graphs.
A: Hypothetical reads aligned to the consensus sequence.
B: The OLC assembly graph created from these reads. Edges represent **overlaps** of 2 or more nt. The assembly process is to visit every **node**.
C: The de Bruijn assembly graph created from **substrings** of these reads. Edges represent 4 nt segments with 2 nt overlap between the nodes. The assembly process is to visit every **edge**.

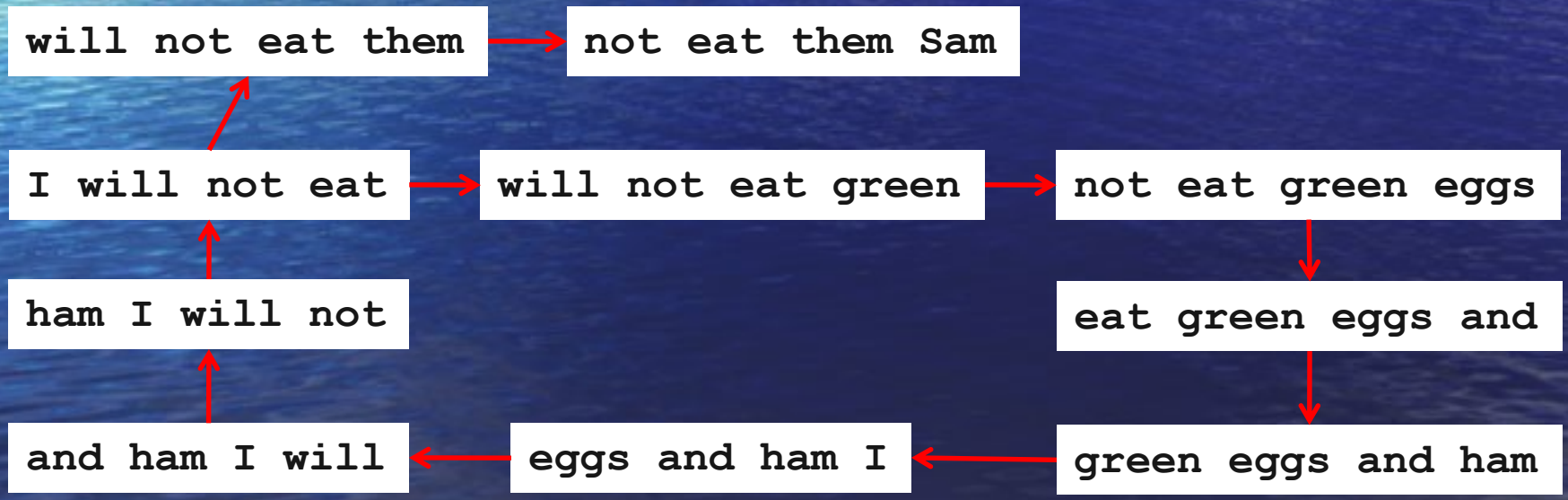
The de Bruijn method



Splitting sequence reads into substrings of a certain size; these are called kmers.

Replicates

```
I will not eat green eggs
  will not eat green eggs and
       not eat green eggs and ham
           eat green eggs and ham I
               green eggs and ham I will
                     eggs and ham I will not
                          and ham I will not eat
                              ham I will not eat them
                                  I will not eat them Sam
```



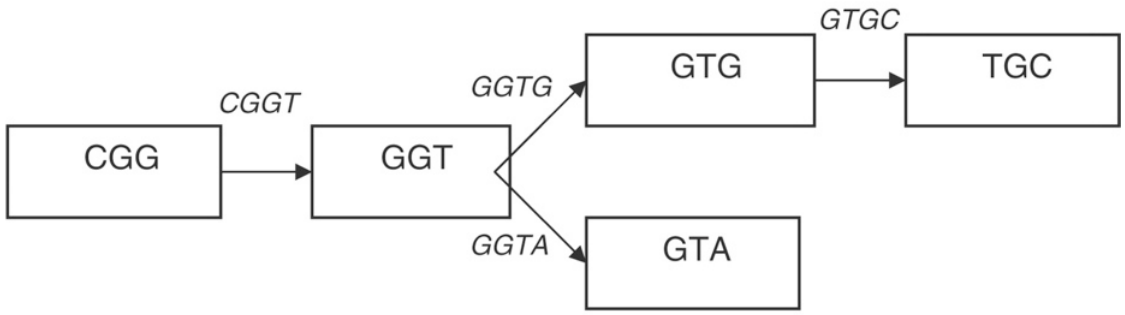
Replicates

A

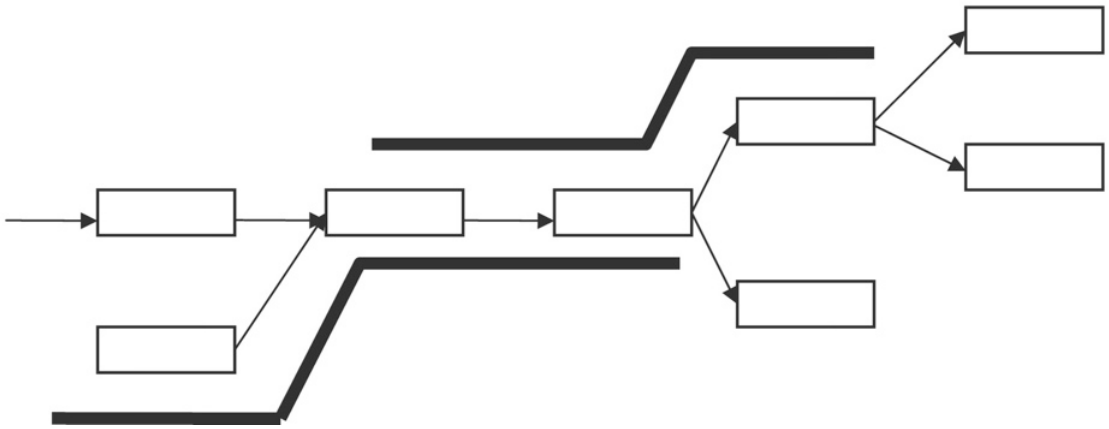
ACCACGGTGCGGTAGAC
ACCA GGTG GGTA
CCAC GTGC GTAG
CACG TGCG TAGA
ACGG GCGG AGAC
CGGT CGGT

(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
(C) Overlap between two reads (bold) that can be inferred from the corresponding paths through the deBruijn graph.

B

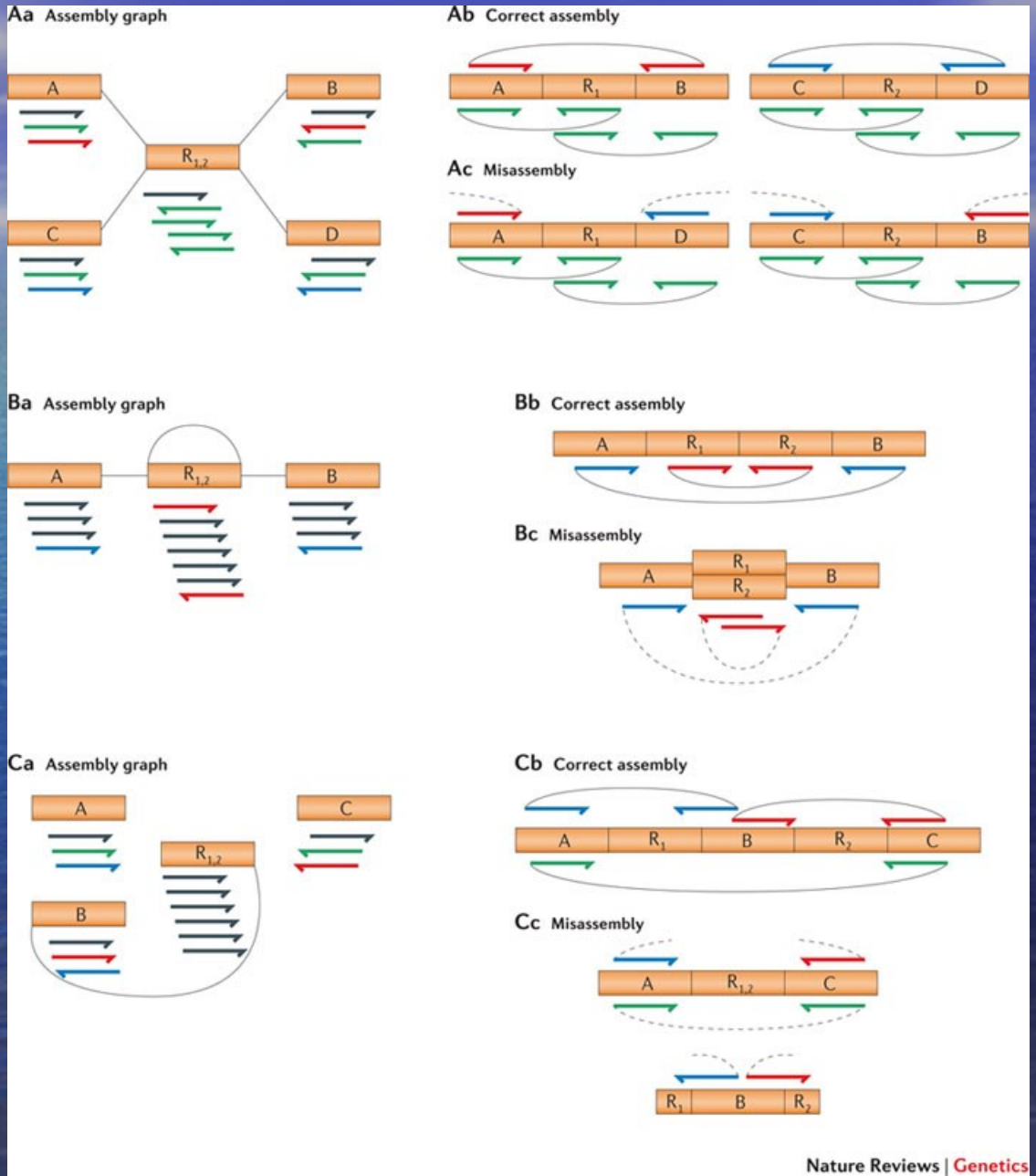


C

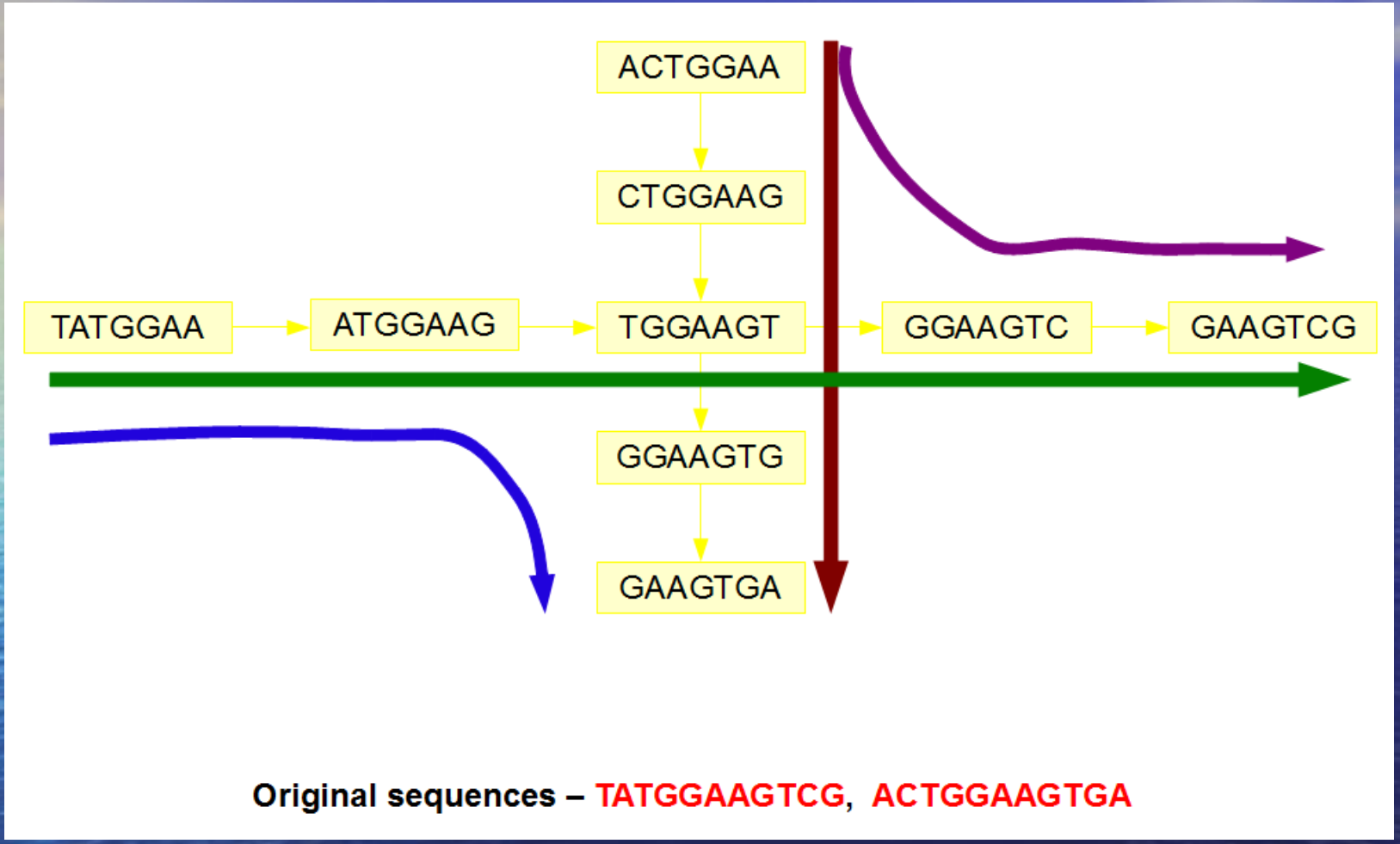


In the case where reads are error-free and cover the whole genome and when there are no repeated sequences with more than k letters, the solution is a single walk in the de Bruijn graph that goes through each arc exactly once. Such a walk is called Eulerian.

Problems


Aa Assembly graph
A
B
C
D
R1,2
Ab Correct assembly
A
R1
B
C
R2
D
Ac Misassembly
A
R1
D
C
R2
B
Ba Assembly graph
A
R1,2
B
Bb Correct assembly
A
R1
R2
B
Bc Misassembly
A
R1
R2
B
Ca Assembly graph
A
B
R1,2
C
Cb Correct assembly
A
R1
B
R2
C
Cc Misassembly
A
R1,2
C
R1
B
R2
Nature Reviews | Genetics

Problem kmer size



Construction of de Bruijn graph leads to loss of information as shown in the following figure. The original reads cannot be derived from the graph structure, because two spurious paths are generated. Increasing k-mer size is one way to resolve those spurious paths, but increasing kmer size leads to insufficient coverage in other genomic regions.

Difference Genome - Transcriptome

Genome
Single contig per locus
Uniform distribution of reads across genome

Transcriptome
Multiple contigs (e.g, isoforms) per locus
Dynamic range of expression values lead to unevenness

Never ever use a genome assembler for transcriptomes (or vise versa)

What to consider

What biological question am I asking?

What are the best data for that?

Can achieve this best by genomic or transcriptomic data?

Can I eventually get these data?

Can I afford it?