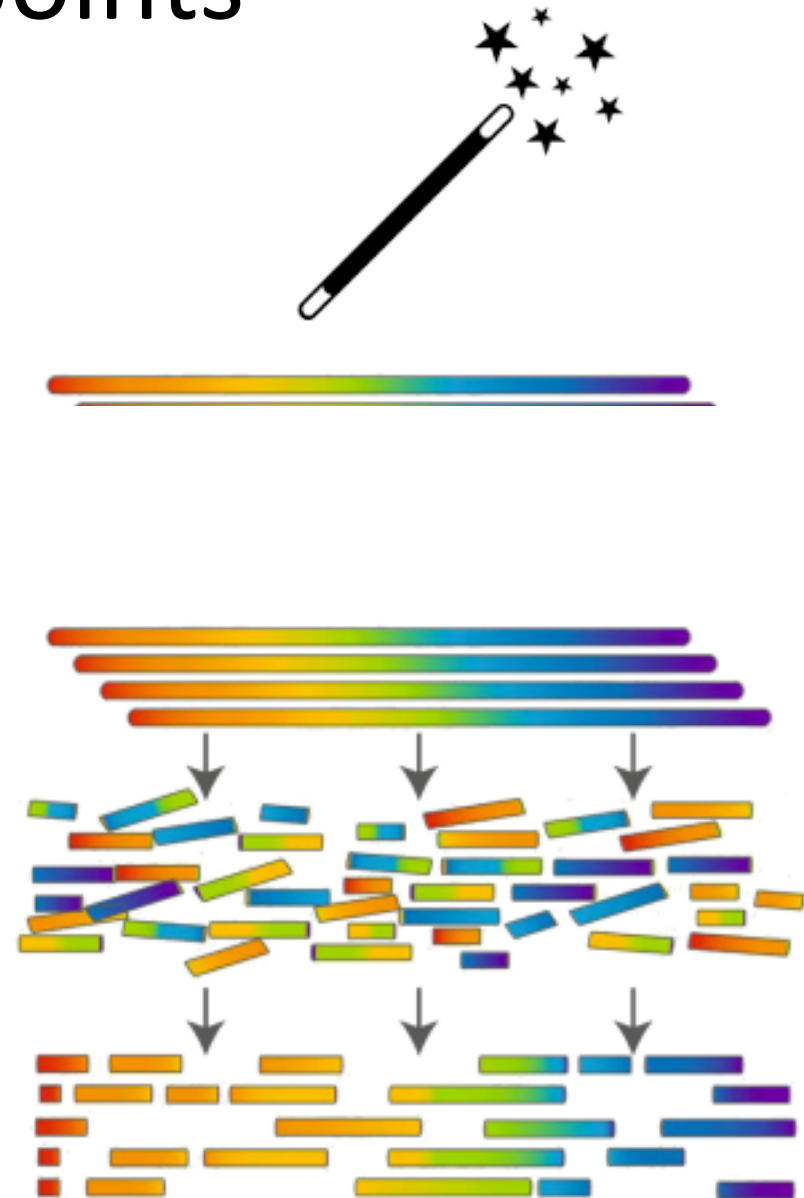


De novo Genome Assembly

- Example of basic short read assembly
- Kmers and de Bruijn graphs
- Assessing Genome Quality
 - Contiguity - N50
 - Completeness
 - Correctness
- Scaffolding

Previous points

- Ideal: read an entire chromosome from beginning to end in one long, perfect run
- Reality: Genome assemblies are messy, leading to varying levels of completeness
 - Complete, Chromosome, Scaffold, Contigs
 - Why?

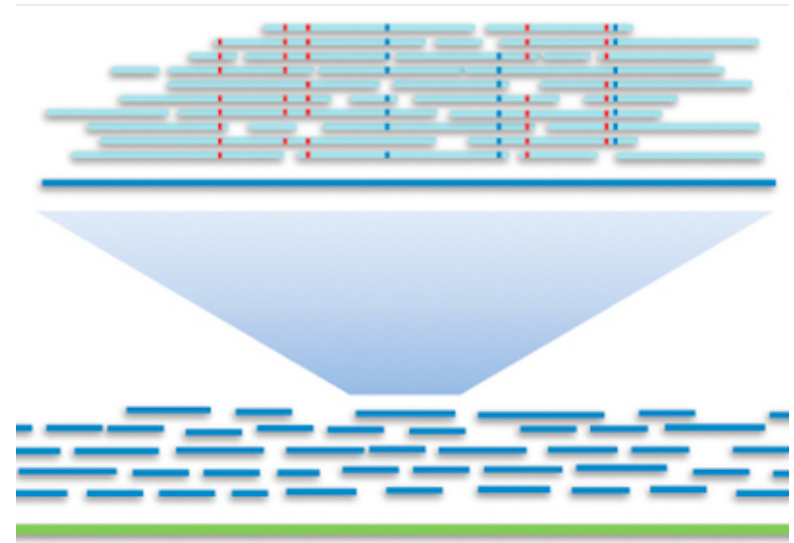


NGS Strategies

- Step 1- Create contigs with short reads
- Step 2 – Order and orient contigs into super structure (scaffolding)

Strategies for scaffolding:

- Mate pairs at a range of distances: 5kb, 10kb, 20kb, 40kb
- Long reads such as PacBio
- Proximity ligation sequencing (Hi-C)



PacBio

Chin et al., Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. 2013 Nature methods

Initial Contig construction

- Reconstructing the original full DNA molecules from (short) read fragments
- Jigsaw puzzle
- How do the pieces fit together? (overlap)
- Missing pieces (sequencing bias)
- Dirty pieces (sequencing error, real biological variation)



An example

A small “genome”:

Friends,

Romans,

countrymen

lend me your ears;

Reads:

ds, Romans, count
ns, countrymen, le
Friends, Rom
send me your ears;
cryman, lend me


Overlaps:

Friends, Rom
ds, Romans, count
ns, countrymen, le
crymen, lend me
send me your ears

Consensus:

Friends, Romans, countrymen, lend me your ears;

Bigger and more complex genomes are more challenging

- Spectrum of difficulty:
 - Biological
 - Size
 - Repetitiveness
 - Polyploidy
 - Heterozygosity
 - Technical:
 - Sequencing Error
 - Sampling Bias
- 
- Challenges:
 - Initial contig build is computationally intensive
 - Many assembly algorithms require 100s of Gb of RAM to Tbs of RAM

PHASE : INTERPRETATION
TWO

SEDMAN *Illustration*



Difficulty 1: Volume of Information

Small genome of 25Mb

- Would like to sample each base 100 times
- 2.5 billion bases in 25 million reads (100 bases per read)



Bigger genome

- Norway spruce is 20Gb
- Produced 1.9 trillion bases



100X coverage is reasonable starting point for assembly.

Difficulty 2: Repeats

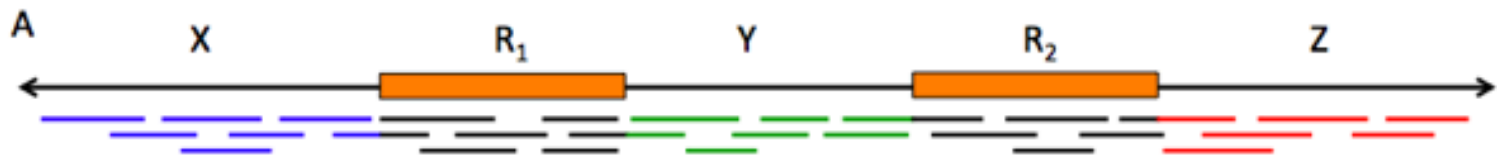
- Short repeats are problematic
- Lots of them, often longer than our (short) read length

?? ATA ??
ATATATATATAT ATATATATATA TATATATATA
ATATATATA ATATATATATATAT TATATAT
TATATATATATATATATATA TATATATATA

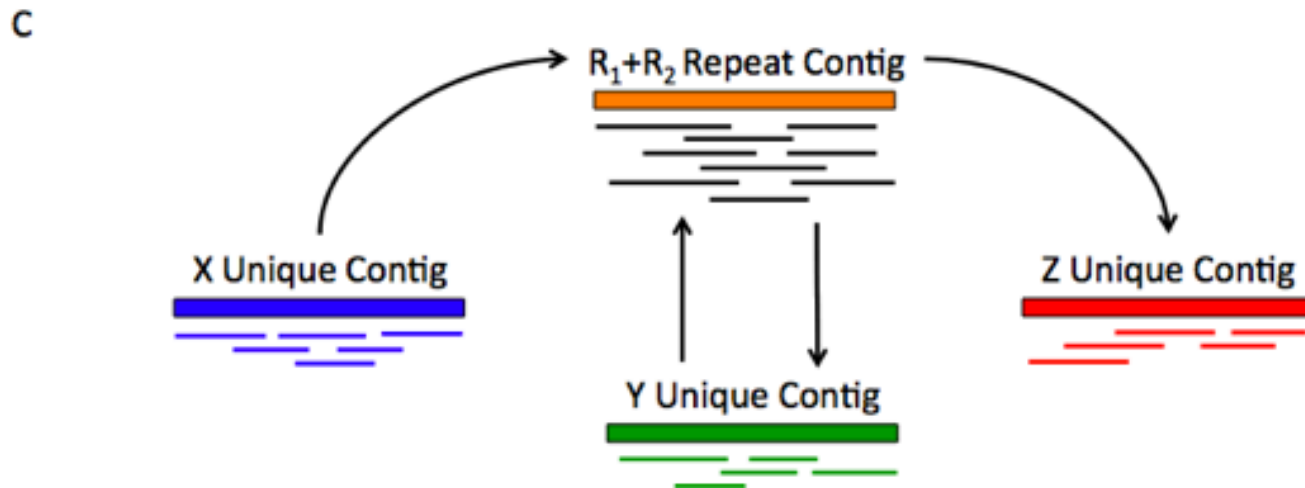
Difficulty 2: Repeats (continued)

Long repeats are problematic too.

Reality:

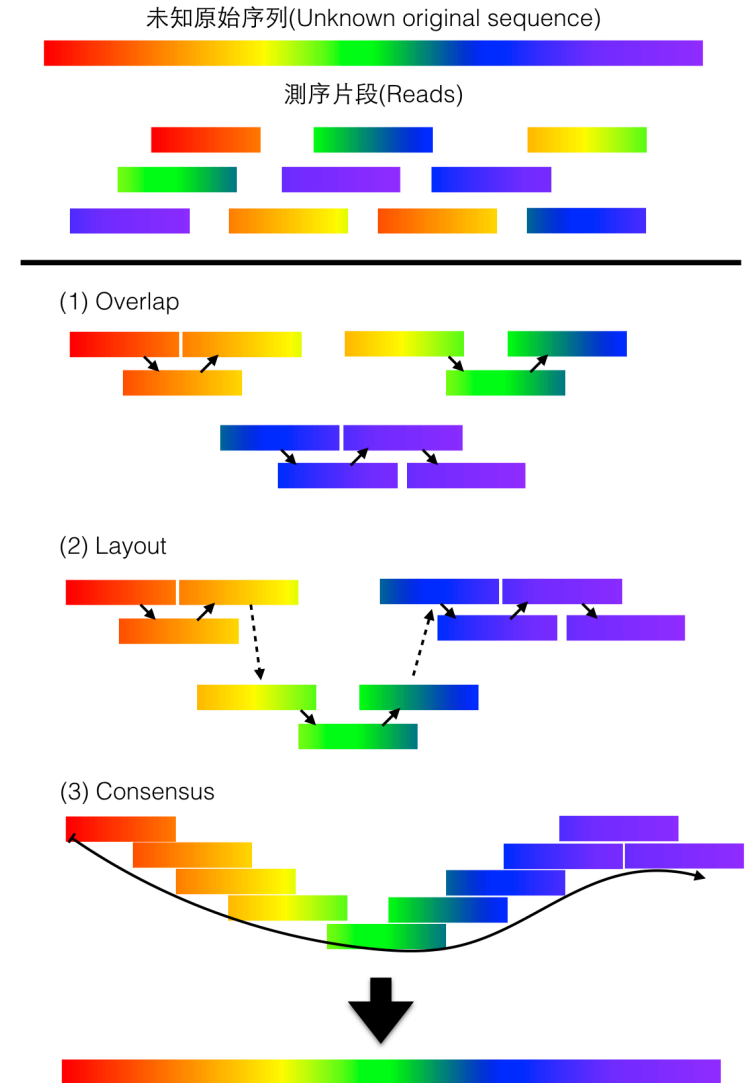


Assembly results:



Approaches

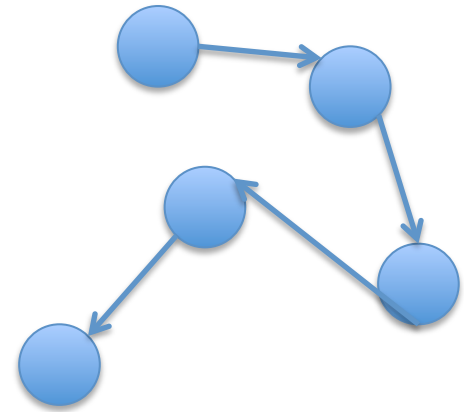
- Overlap, layout, consensus (OLC)
 - Relies on an overlap graph
 - Long reads (Sanger, PacBio)
- de Bruijn graph
 - Uses a k-mer graph
 - Short reads (Illumina)
 - Much more efficient algorithm, less accurate



The previous example used OLC.

Graph

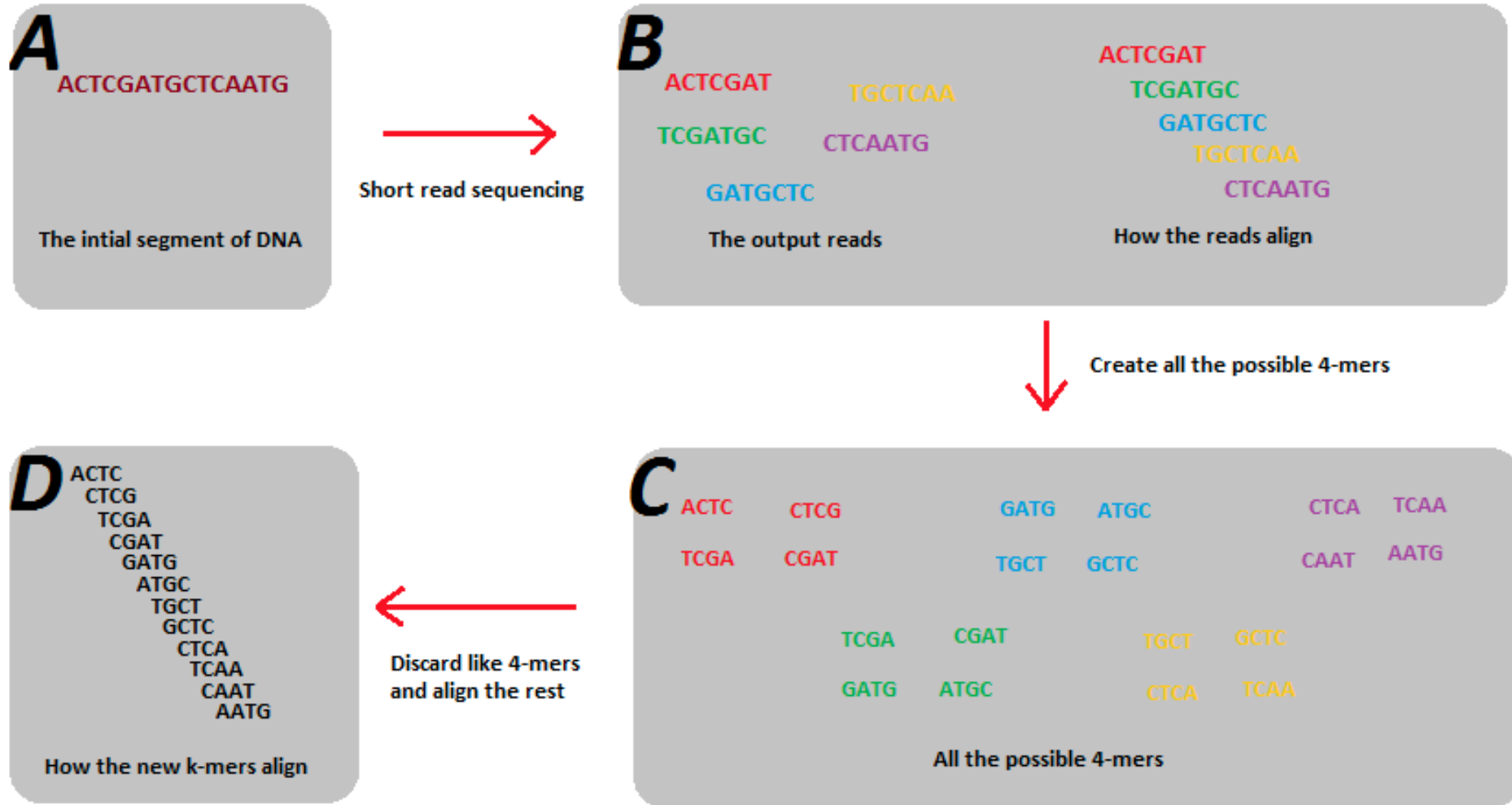
- Abstraction of data to nodes and edges
- Directed graph
 - Edges may only be traversed in one direction
 - Collections of edges form paths



de Bruijn Graph

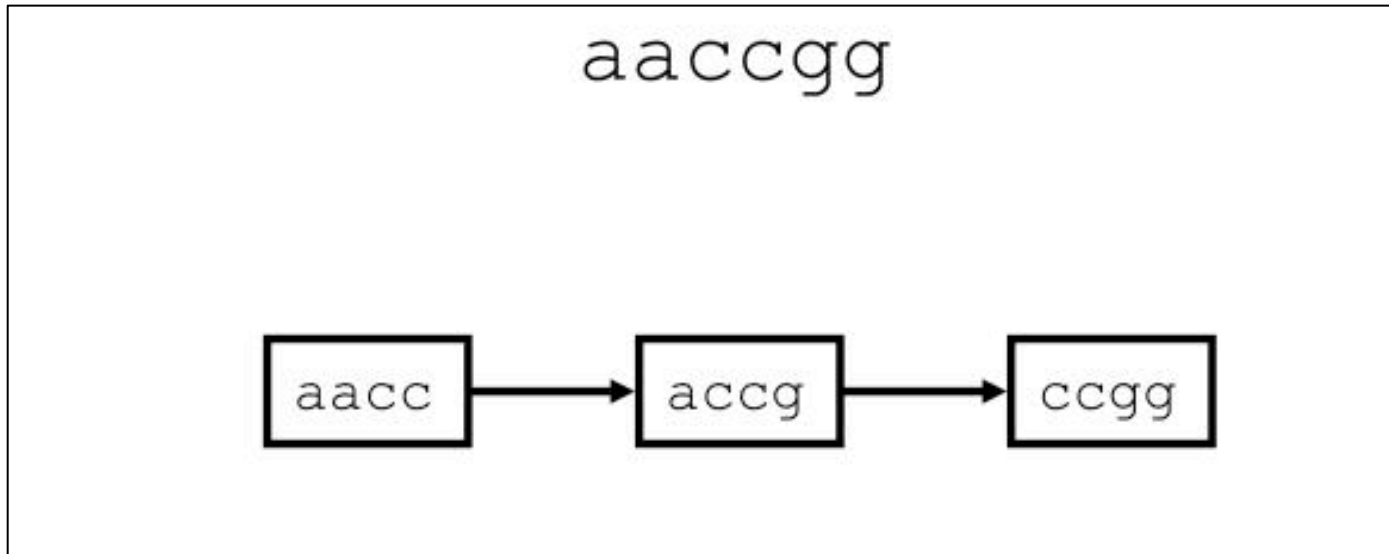
- Developed as a mathematic construct
- K-mer graph developed as a type of de Bruijn graph useful for assembly
- Nodes are subsequences of a longer sequence
- Edges are fixed length overlaps
- By not using the whole reads, much less memory is needed. Storing on unique k-mers in memory, not every read.

K-mers



A single read represented as a k-mer graph

$K = 4$



- More than one read can be represented by a de Bruijn graph
- Reads with perfect overlaps have the same path
- These overlaps are detected without calculating the alignments between every read pair
- Major computational savings!!!

Start with two reads

(a)

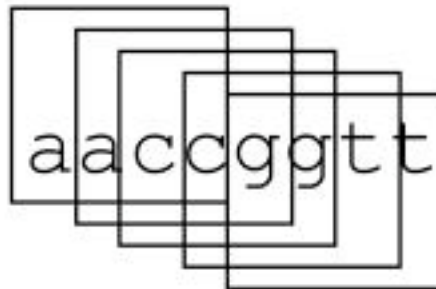
aaccgg
ccggtt

Build the graph – the graph holds the information about their overlap

(b)

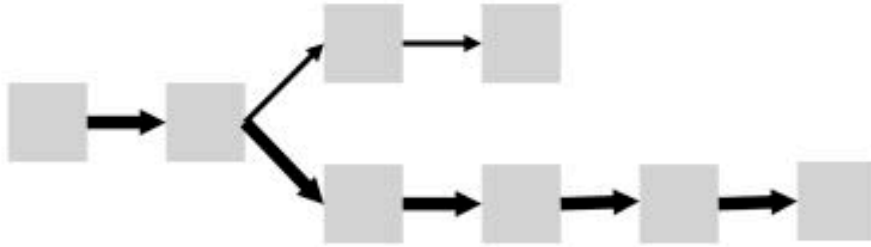


(c)

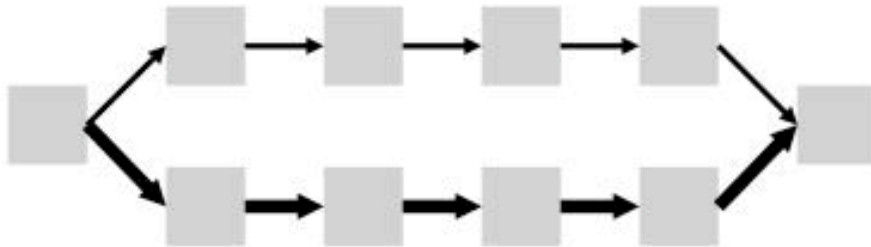


Merge the nodes to yield a sequence contig

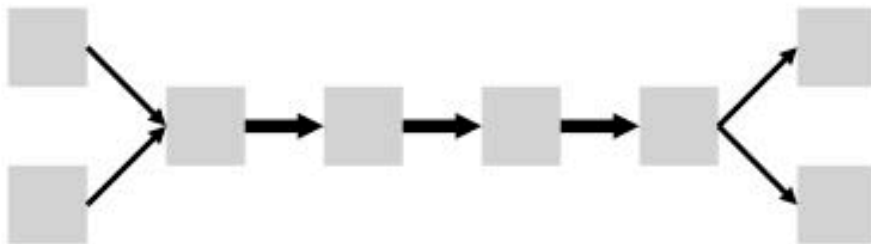
Complexity in Graphs



Error at read end causes a “spur”



Real polymorphism or error
in the middle of a read
causes a “bubble”



Repeats yield a “frayed rope”

Example

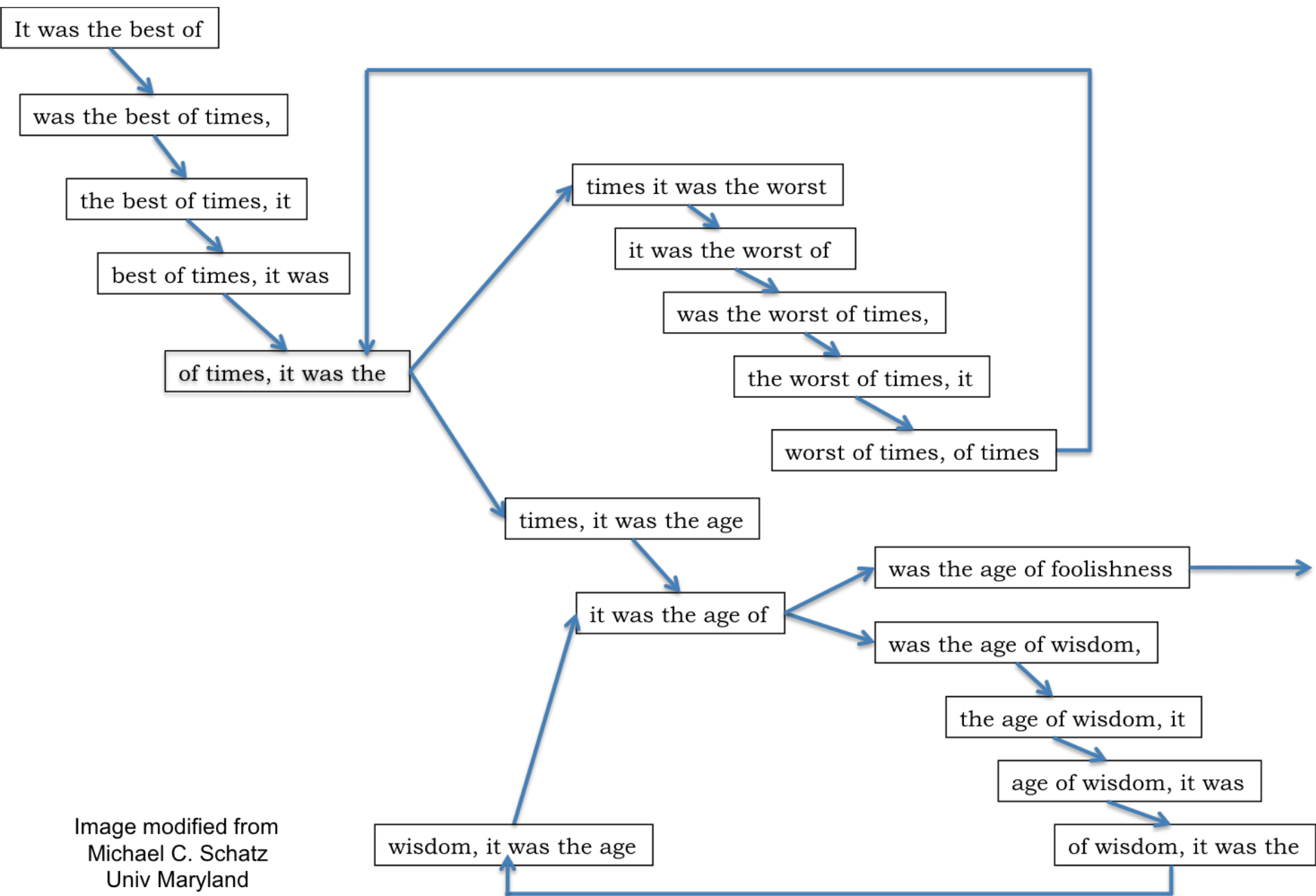
It was the best of times, it was the worst of times

It was the age of wisdom, it was the age of foolishness

Example

It was the best of times, it was the worst of times

It was the age of wisdom, it was the age of foolishness



Kmer size is important

Ecoli K12

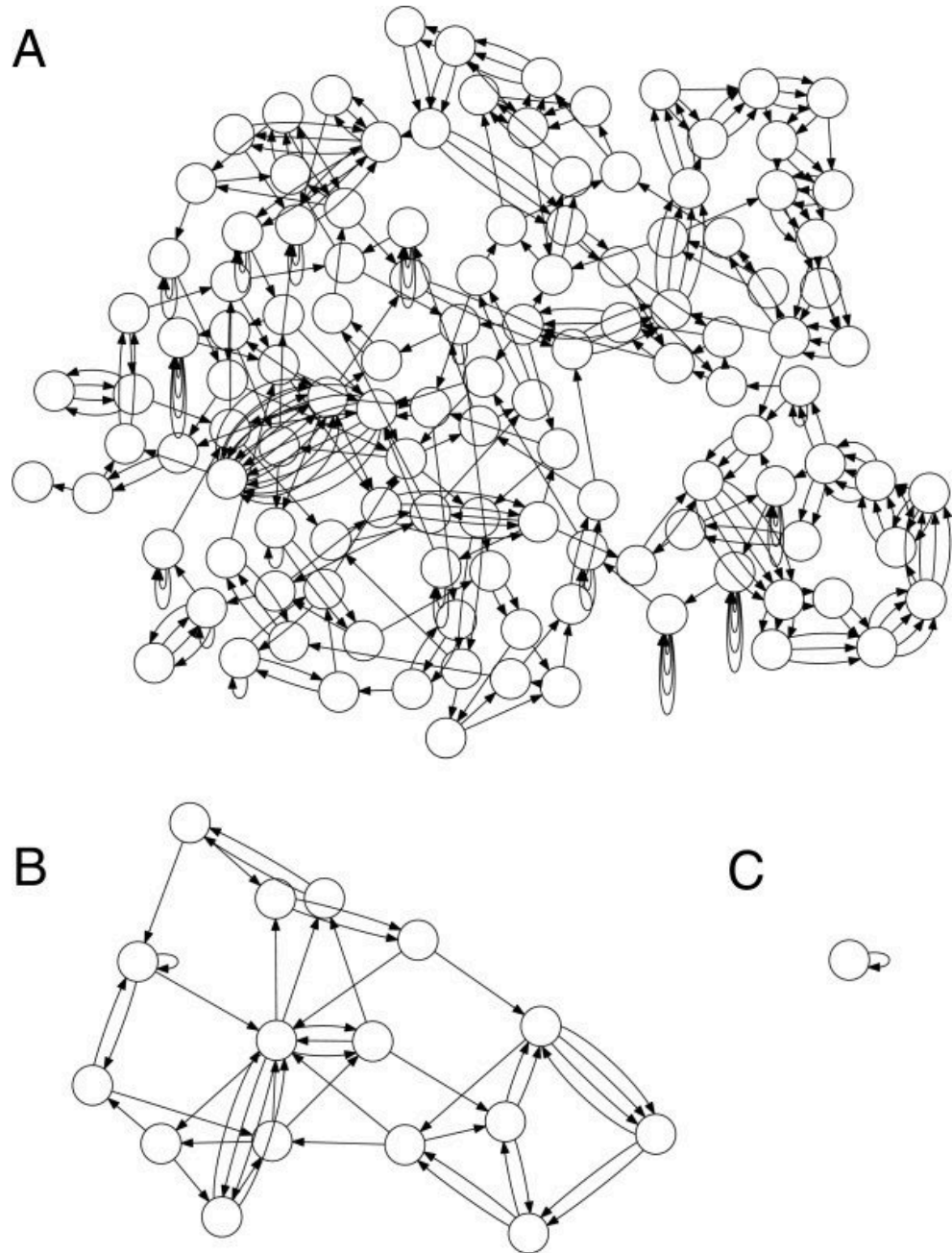
A. kmer of 50

B. kmer of 1000

C. kmer of 5000

Reducing Assembly Complexity of
Microbial Genomes with Single-
Molecule Sequencing

Koren et al 2013



A vertical stack of two blue square buttons. The top button contains a black plus sign (+) and the bottom button contains a black minus sign (-).



Selecting k-mer value

- Smaller k-mers require less memory
 - But graphs are more complex and yield smaller average contig lengths
 - Repeats longer than the k-mer length cannot be resolved.
- Larger k-mers require much more memory
 - Can yield longer contigs but also more small contigs that are actual overlapping pieces
- Usually you will want to try a set of k-mer values and pick the best
 - Start with $\frac{1}{2}$ to $\frac{2}{3}$ the read length if you have sufficient RAM resources

Try these videos for more clarity

- Ben Langmead
- [https://
www.youtube.com/
watch?v=TNYZZKrjCSk](https://www.youtube.com/watch?v=TNYZZKrjCSk)
- [https://
www.youtube.com/
watch?v=FCDJlx-W7C8](https://www.youtube.com/watch?v=FCDJlx-W7C8)

SPAdes

ABYSS

- Primarily for bacteria assemblies
- Single cell sequencing as well as (normal) multi-cell sequencing
- Proper utilization of paired end data in a de Bruijn graph

ALLPATHs-LG

- Common for large eukaryotic genomes
- Requires both a paired end and a mate pair library
- Many clever improvements on memory needs and traversing the de Bruijn graph

Assessing Genome Quality

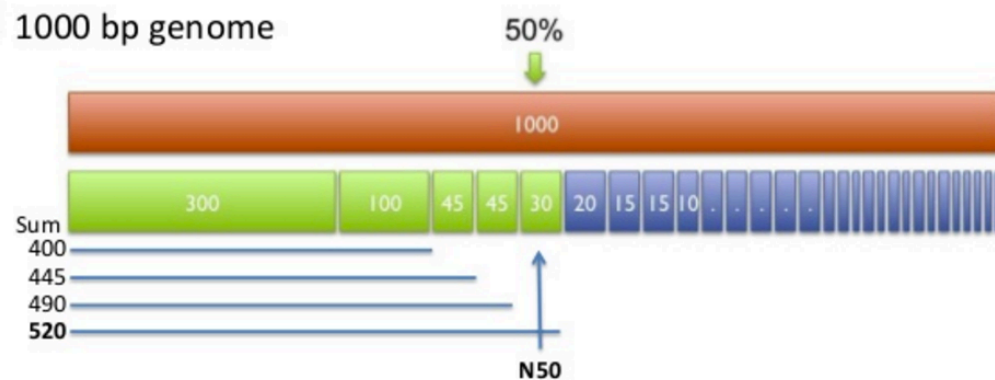
1. Contiguity
2. Completeness
3. Correctness

Assessing Genome Quality

1. Contiguity

- Would like fewer contigs in longer pieces
- Assess # of contigs/scaffolds, average length, N50

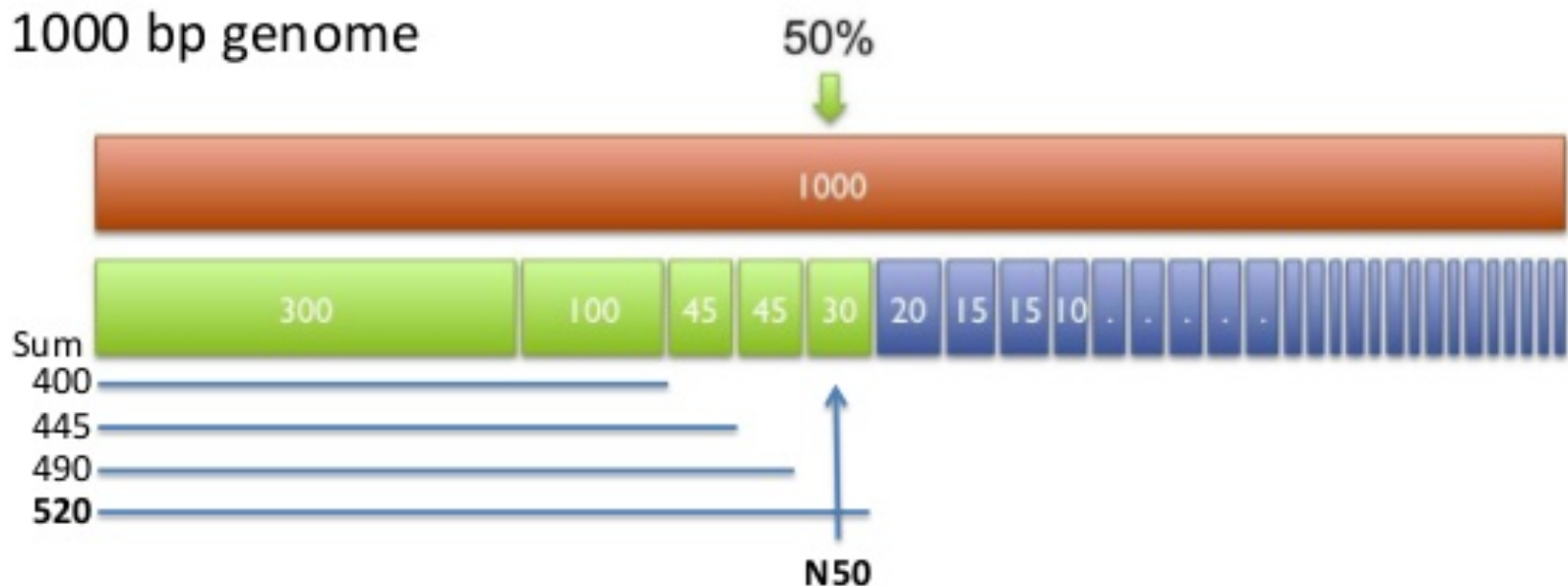
50% of the genome is in contigs as large as the N50 value



N50

N50 statistic

- the length for which the collection of all contigs of that length or longer contains at least 50%
- similar to a mean or median of contig lengths
- used widely in genome assembly, especially in reference to contig lengths within a draft assembly.



N50 Example

- Assembly A contains six contigs of lengths:
 - 80 kbp, 70 kbp, 50 kbp, 40 kbp, 30 kbp, and 20 kbp
 - Sum size of assembly A is 290 kbp
 - N50 contig length is 70 kbp
 - “Half of the assembly is contained in contigs of 70kbp or greater”
 - If you randomly selected a location, 50% of the time it would be in a contig of 70kbp or greater
- Assembly B contains eight contigs of lengths:
 - 80 kbp, 70 kbp, 50 kbp, 40 kbp, 30 kbp, 20 kbp, 10kbp, 5kbp
 - Sum size of assembly B is 305 kbp
 - N50 contig length is 50 kbp

Assessing Genome Quality



2. Completeness

- How much of the total genome was assembled?
 - Between 1 and 0
 - 80% complete vs 99% complete
 - This is based on an understanding of actual genome size
- Many times repeats are difficult, but we're mostly concerned with genes
 - How many of the genes were captured in the assembly?
 - Can assess in two ways:
 - % of RNASeq reads that map to genome
 - Core conserved single copy orthologs – BUSCO
 - Make the assumption that the proportion of conserved single copy orthologs can be extrapolated to the total proportion of assembled genes

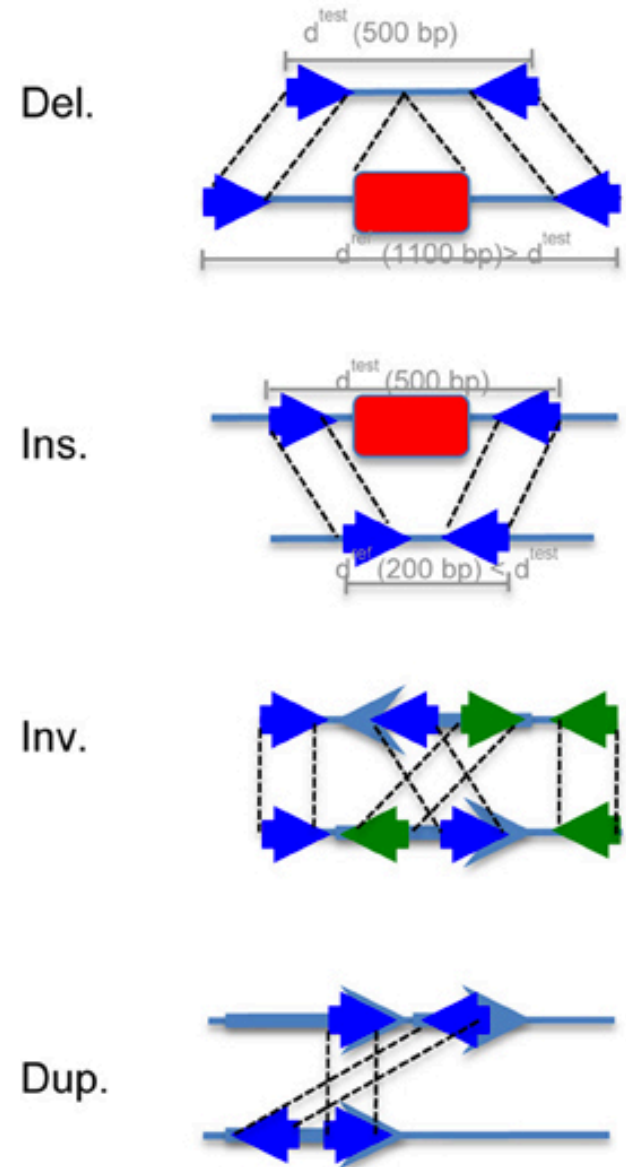
Assessing Genome Quality

3. Correctness

- Lack of errors, such as misjoins, collapsed repeats, miscalled bases, insertion/deletions
- Assess by aligning all reads back to the assembly and look for inconsistencies

(Analogous to sequencing a different individual and looking for structural variants)

Read-pair analysis



Assessing Genome Quality

- QUAST
- Can be used with or without a reference genome
- Compare assembly attempts to each other
- Reports
 - Basic statistics about contigs and scaffolds
 - Comparison to a reference genome, including misassemblies and structural variation
 - Percent of reference genome covered by the assembly

QUAST

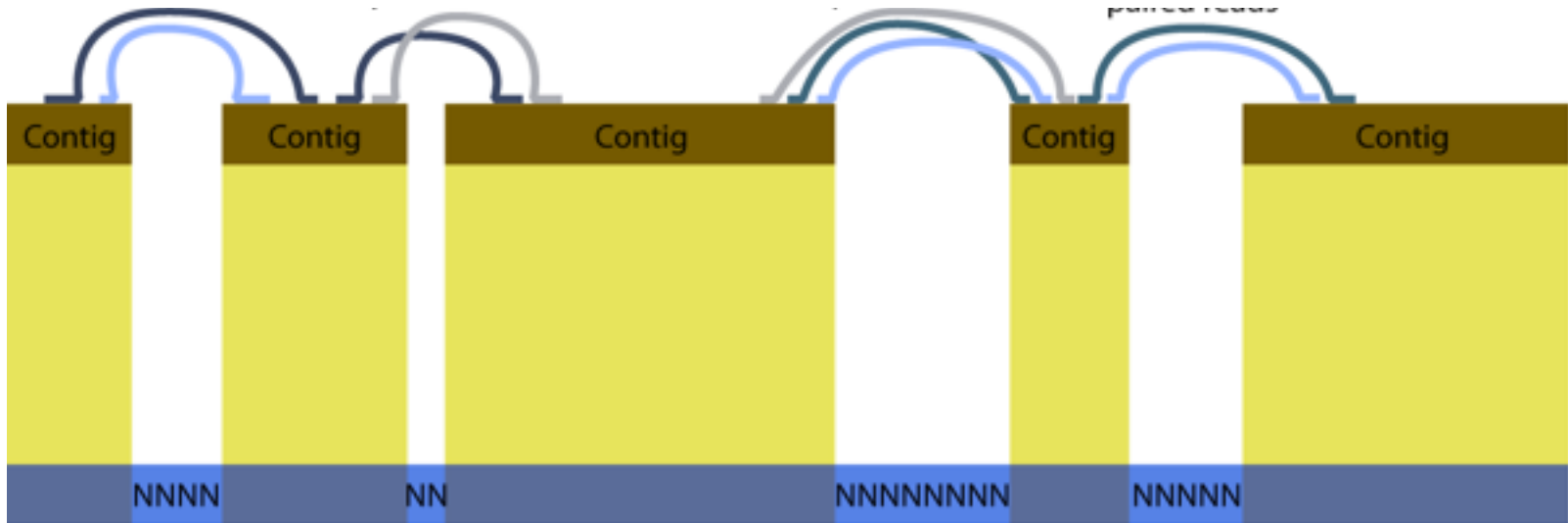
Quality Assessment Tool for Genome Assemblies by Center for Algorithmic Biotechnology

Scaffolding

1. Paired ends and mate pairs*
2. Proximity Ligation (Hi-C)**
3. PacBio or other long read sequencing
4. Optical Mapping
5. Linkage Maps

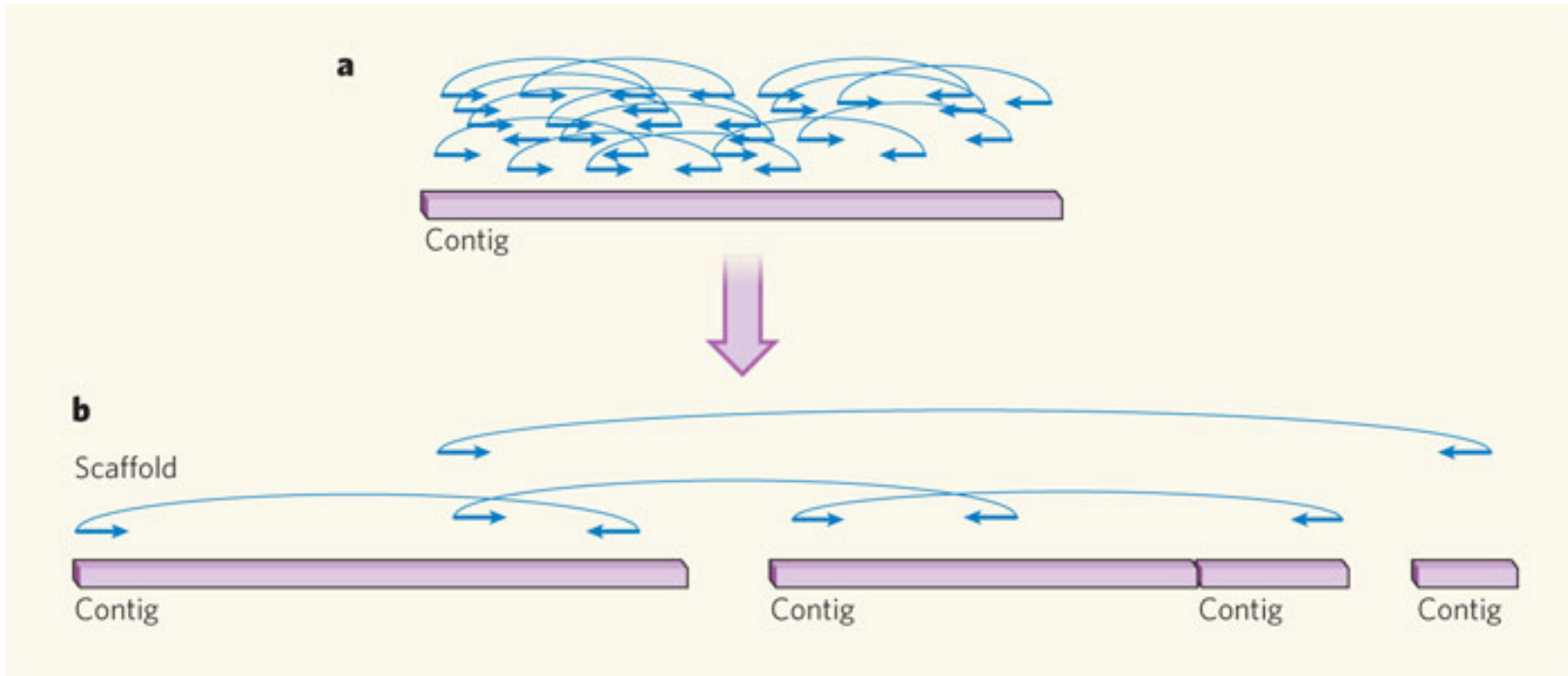
Or some combination of those...

What is a scaffold?



1. Order and orient individual contigs into a larger super structure
2. Fill the gaps with N's
3. Make a new fasta file with the long scaffold sequences

Scaffolding with mate pairs



Use long distance information to place contigs.