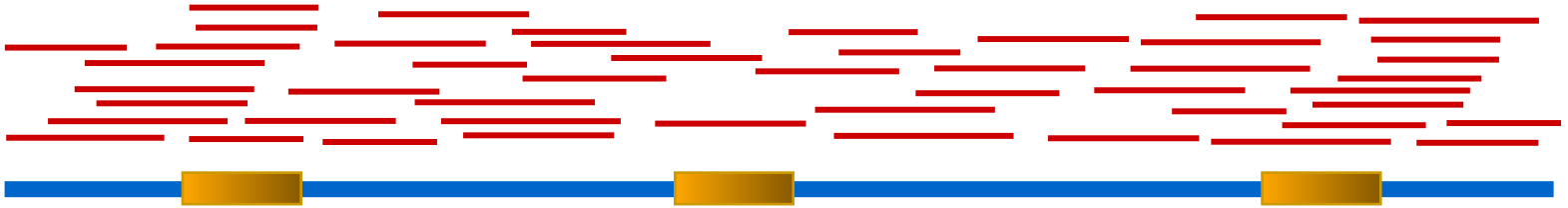
DNA Sequencing

What can we do about repeats?

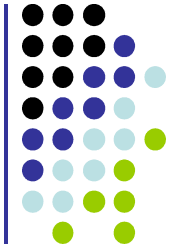
Two main approaches:

- Cluster the reads



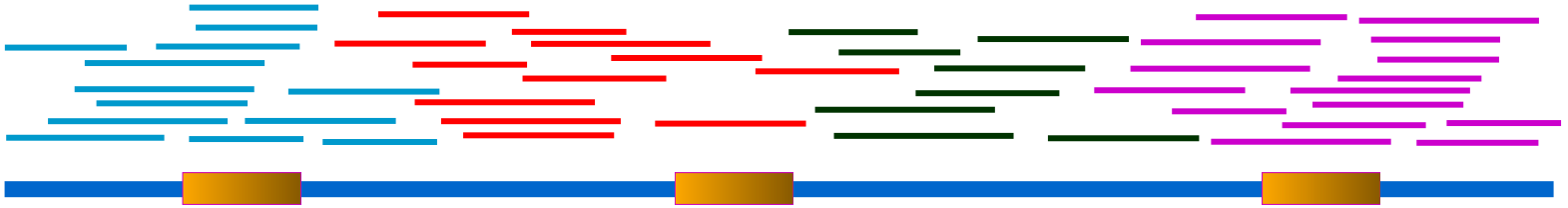
- Link the reads

What can we do about repeats?



Two main approaches:

- Cluster the reads

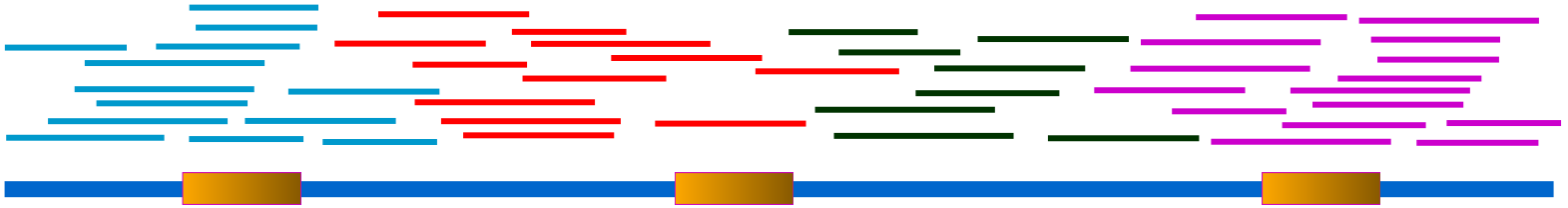


- Link the reads

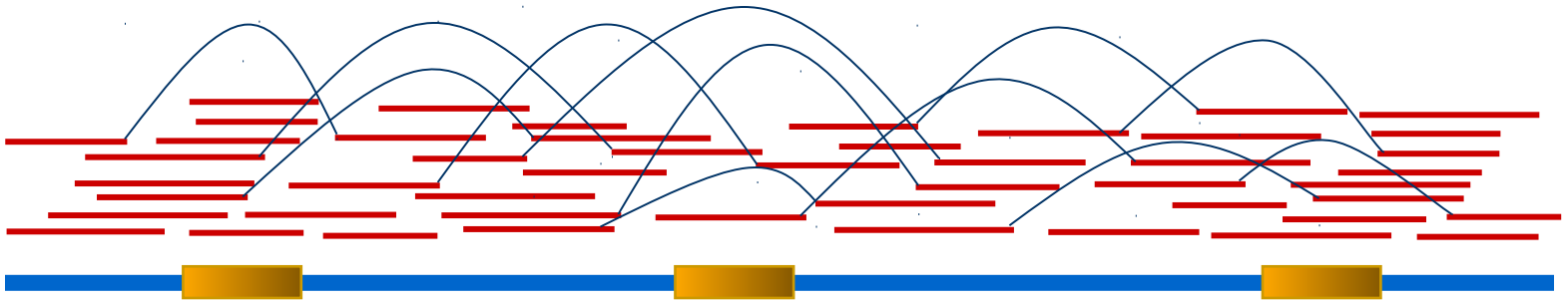
What can we do about repeats?

Two main approaches:

- Cluster the reads

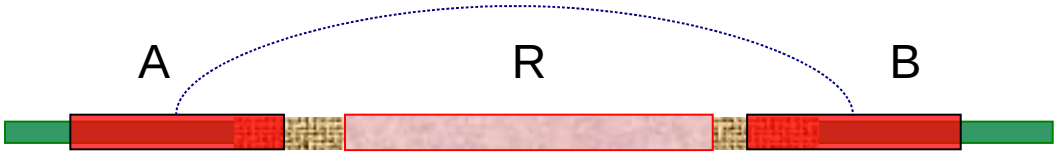


- Link the reads



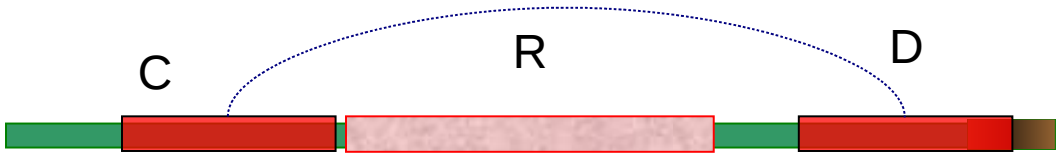
Sequencing and Fragment Assembly

$3x10^9$ nucleotides



ARB, CRD

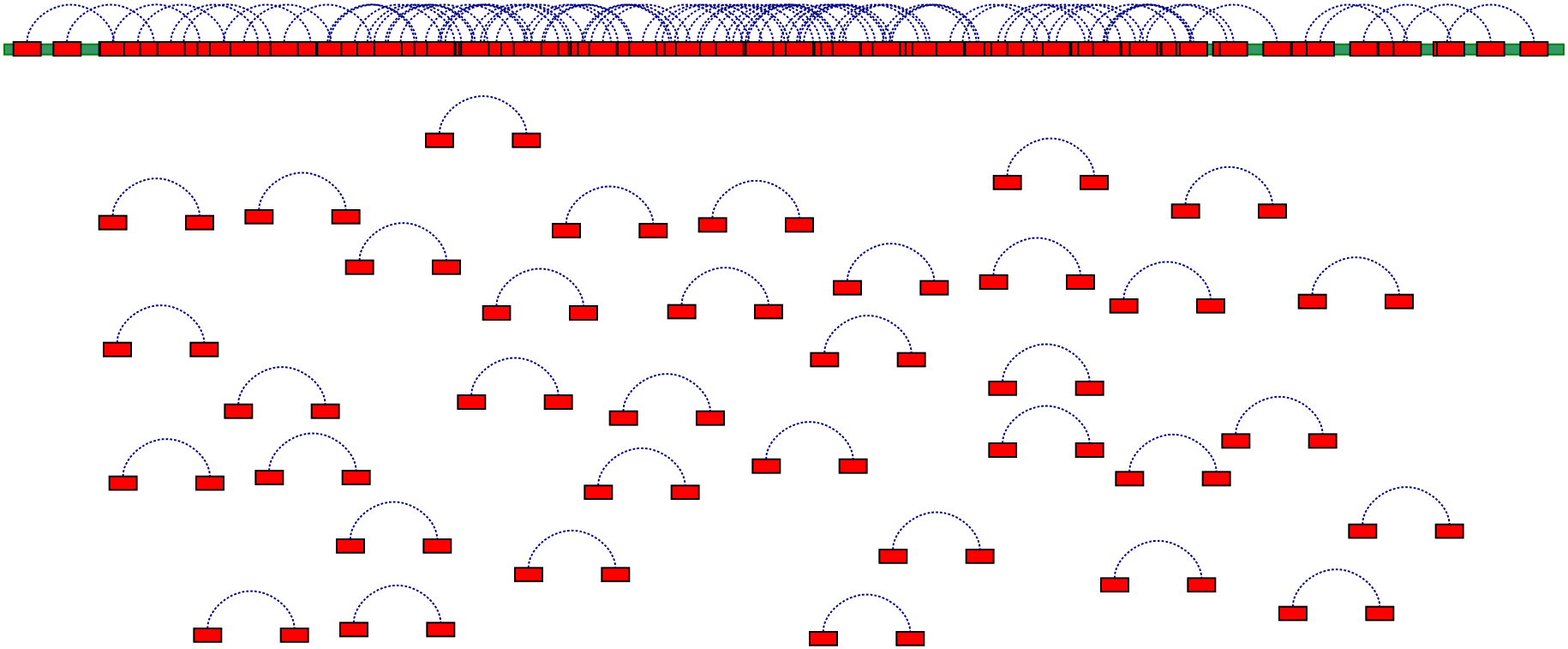
or



ARD, CRB ?

Sequencing and Fragment Assembly

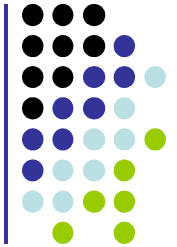
$3x10^9$ nucleotides



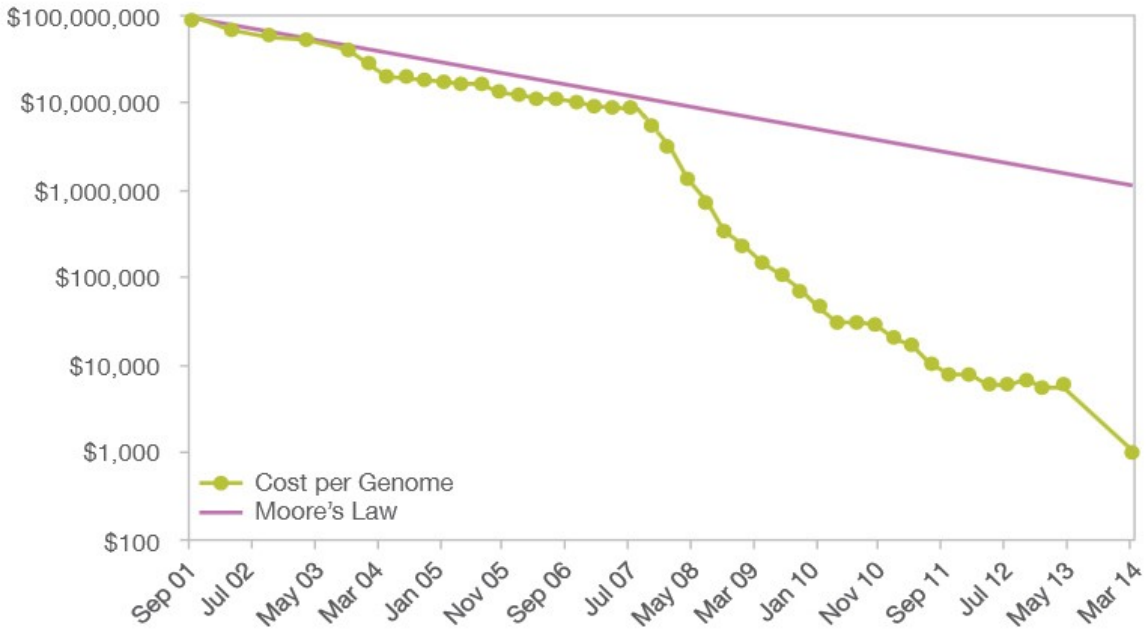
Long Reads

The Holy Grail

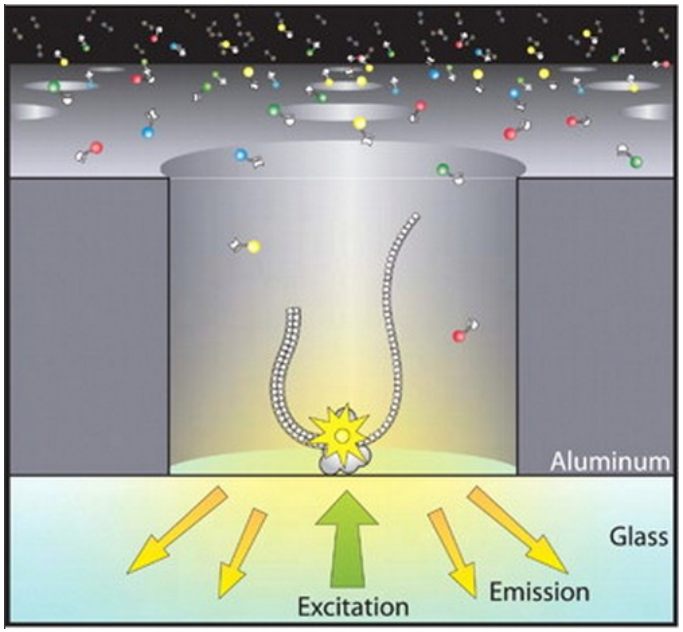
Short Read Sequencing Specs



- http://systems.illumina.com/systems/sequencing.ilmn

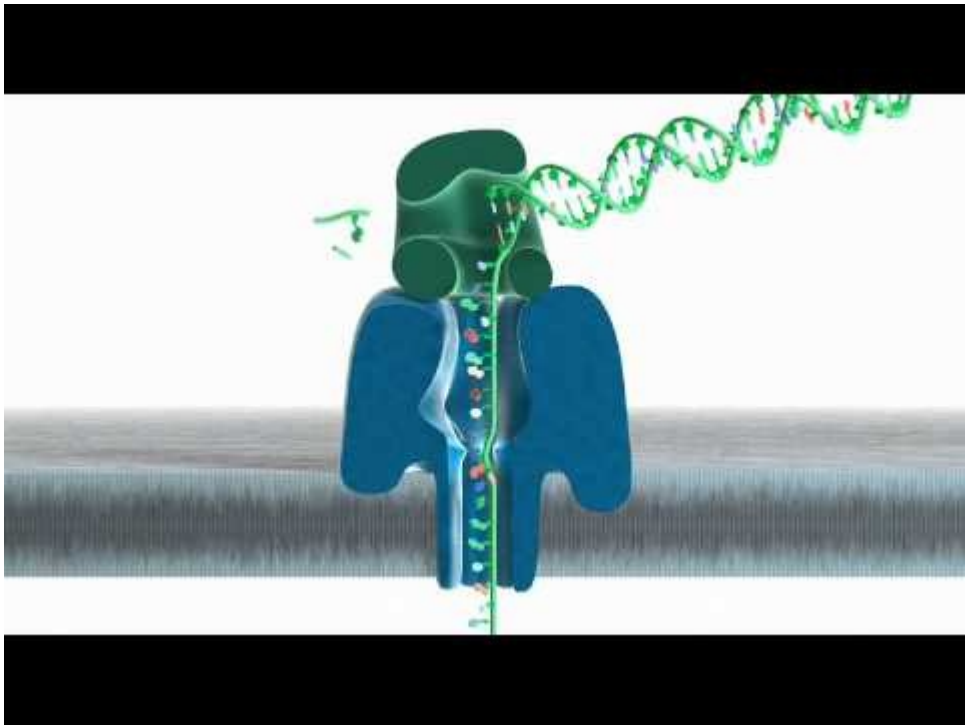


Long Reads - PacBio

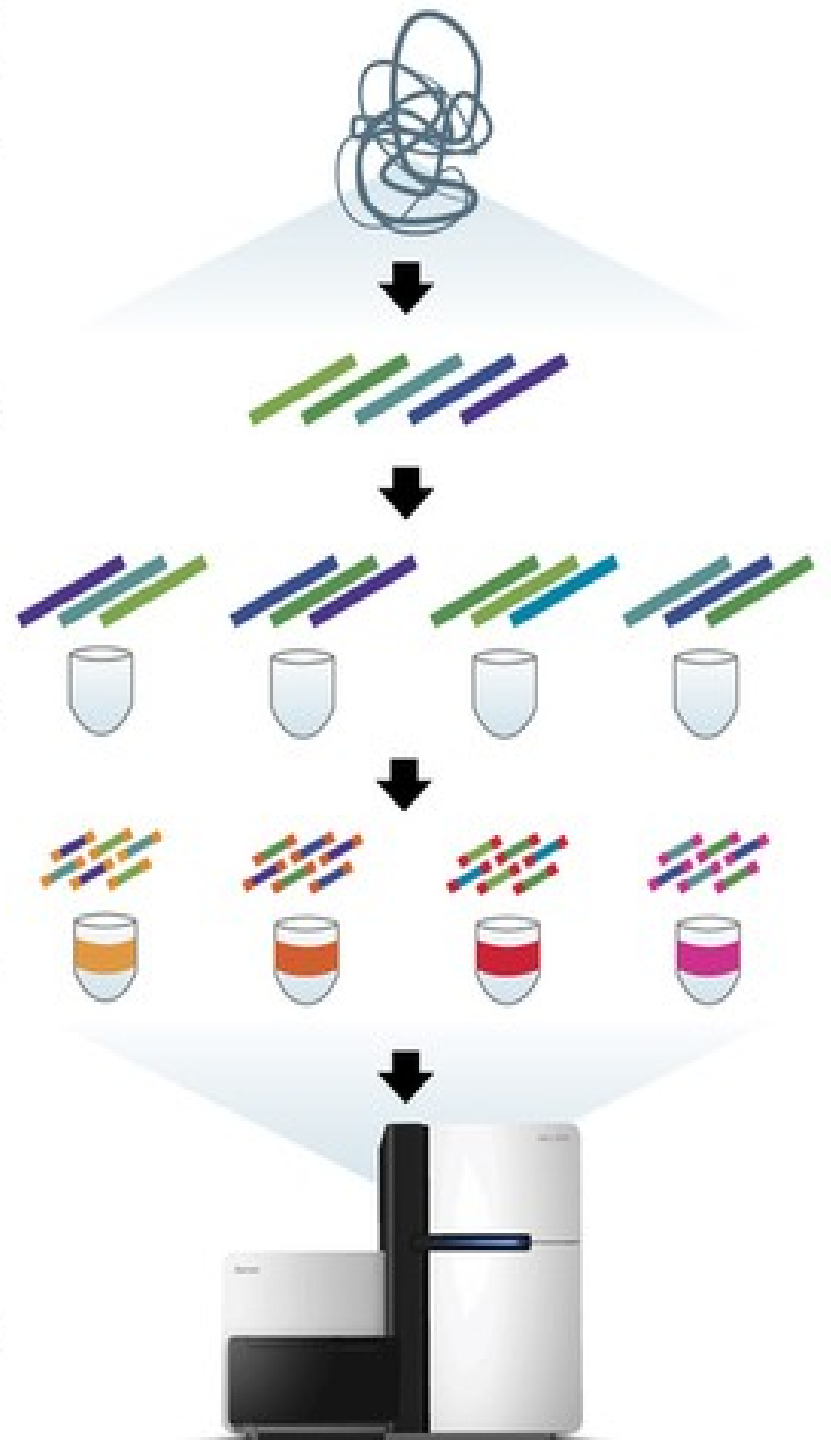


Chemistry	RS II: P4-C2	RS II: P5-C3	RS II: P6-C4
Optimized For	higher quality	longer reads	longer reads
Run time	180 min	180 min	240 min
Total output	~275 Mb	~375 Mb	~500 Mb - 1 Gb
Output/day	~2.2 Gb	~3 Gb	~2 Gb
Mean read length	~5.5 kb	~8.5 kb	~15 kb
Single pass accuracy	~86%	~83%	~86%
Consensus (50X) accuracy	>99.999%	>99.98%	>99.999%
# of reads	~50k	~50k	~50k
Instrument price	~$700k	~$700k	~$700k
Run price	~$400	~$400	~$400

Long Reads – Oxford Nanopore



Read length: 50,000+?
Cost ?



Moleculo Overview

1. Sample DNA is sheared into fragments of about 10 kbp

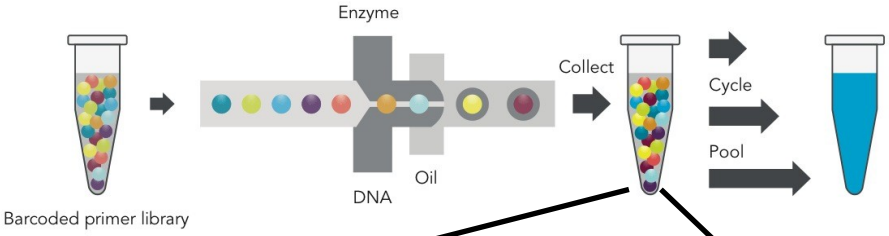
2. Fragments are diluted and placed into 384 wells

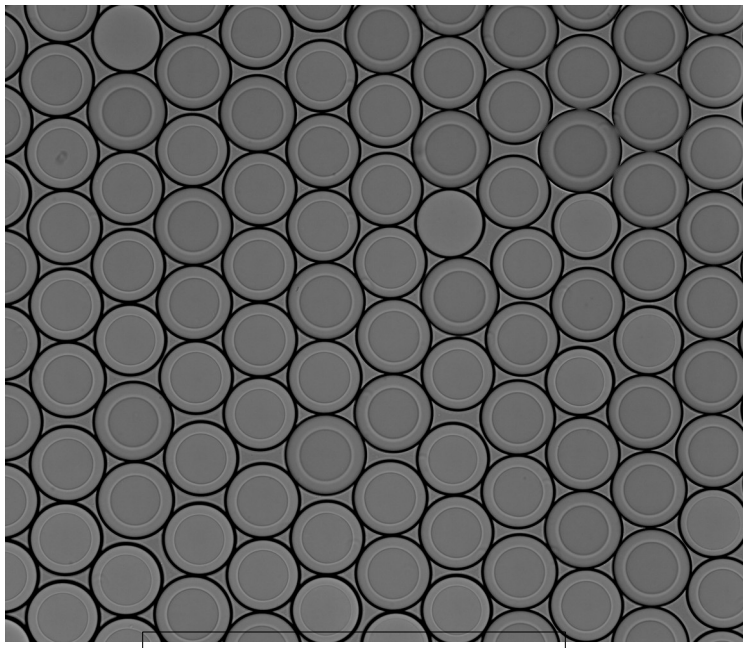
3. Fragments are amplified through long-range PCR, cut into short fragments and barcoded

4. Short fragments are pooled together and sequenced

10x System

Massively Parallel Partitioning

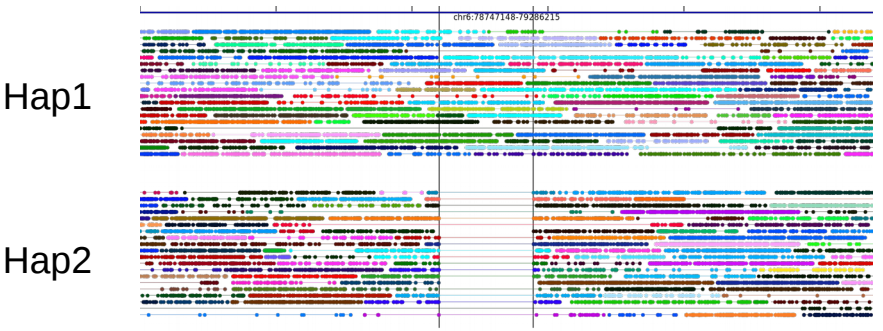




X 700,000+

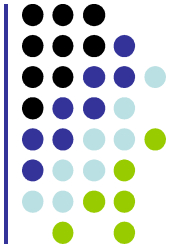
10X Instrument & Reagents

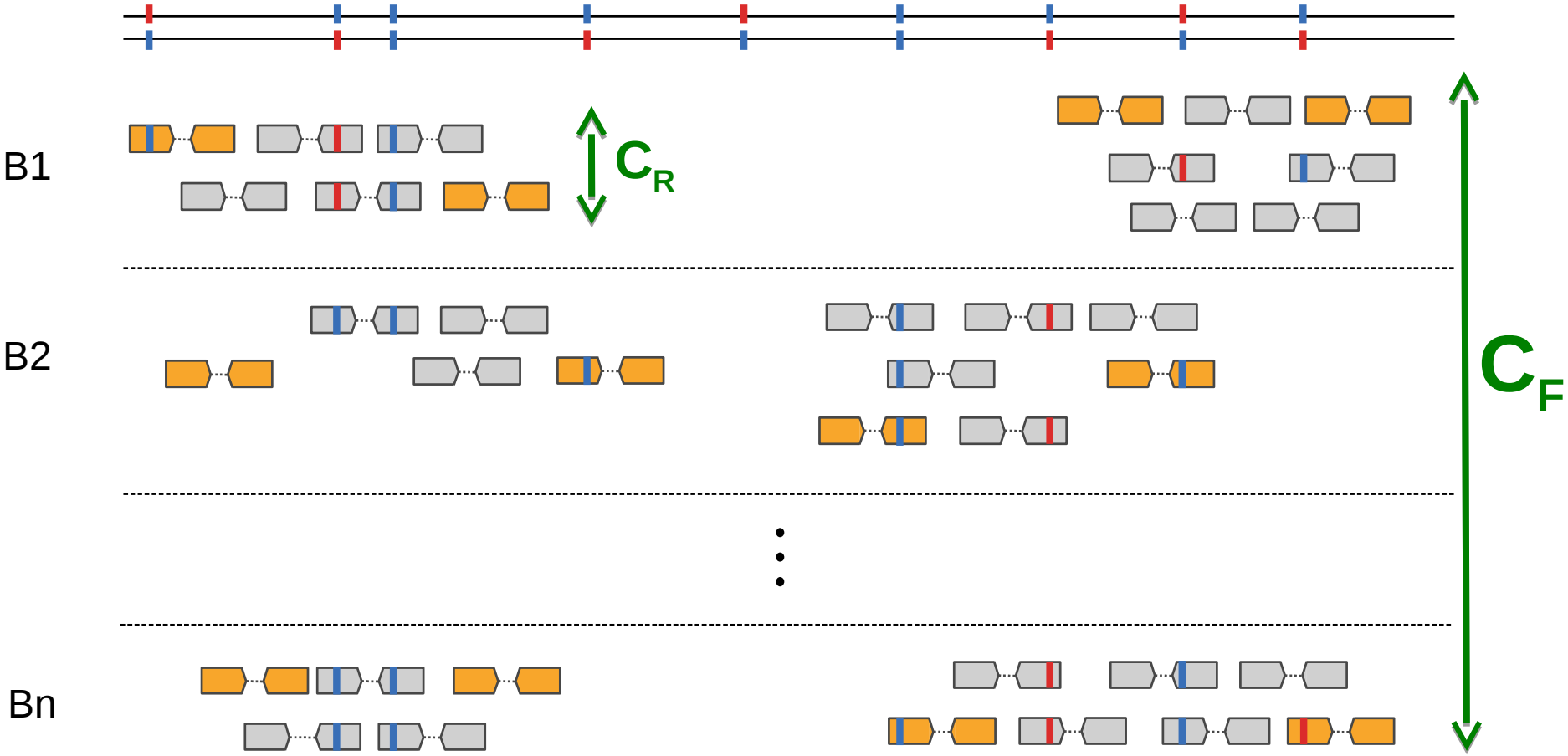
Read Clouds ("linked reads")



Phased 60Kb deletion

Read Clouds





Coverage = $C_R C_F$

Fragment Assembly

(in whole-genome shotgun sequencing)

Fragment Assembly

Steps to Assemble a Genome

Some Terminology

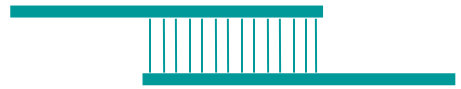
read a 500-900 long word that comes out of sequencer

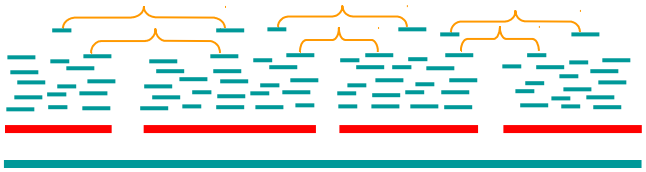
mate pair a pair of reads from two ends of the same insert fragment

contig a contiguous sequence formed by several overlapping reads with no gaps

supercontig (scaffold) an ordered and oriented set of contigs, usually by mate pairs

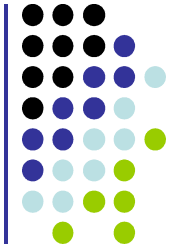
consensus sequene sequence derived from the multiple alignment of reads in a contig





..ACGATTACAATAGGTT..

1. Find Overlapping Reads

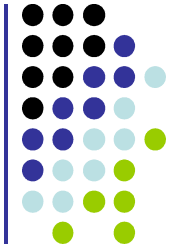


```
aaactgcagtacggatct
aaactgcag
 aactgcagt
…
         gtacggatct
          tacggatct
gggcccaaactgcagtac
gggcccaaa
 ggcccaaac
…
         actgcagta
          ctgcagtac
gtacggatctactacaca
gtacggatc
 tacggatct
…
         ctactacac
          tactacaca
```

```
(read, pos., word, orient.)
aaactgcag
aactgcagt
actgcagta
…
gtacggatc
tacggatct
gggcccaaa
ggcccaaac
gcccaaact
…
actgcagta
ctgcagtac
gtacggatc
tacggatct
acggatcta
…
ctactacac
tactacaca
```

```
(word, read, orient., pos.)
aaactgcag
aactgcagt
acggatcta
actgcagta
actgcagta
cccaaactg
cggatctac
ctactacac
ctgcagtac
ctgcagtac
gcccaaact
ggcccaaac
gggcccaaa
gtacggatc
gtacggatc
tacggatct
tacggatct
tactacaca
```

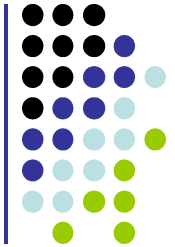
1. Find Overlapping Reads



- Find pairs of reads sharing a k-mer, k ~ 24
- Extend to full alignment – throw away if not >98% similar

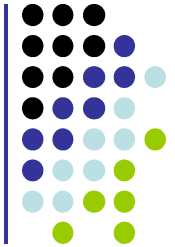
- Caveat: repeats
  - A k-mer that occurs N times, causes $O(N^2)$ read/read comparisons
  - ALU k-mers could cause up to $1{,}000{,}000^2$ comparisons
- Solution:
  - Discard all k-mers that occur “too often”
    - Set cutoff to balance sensitivity/speed tradeoff, according to genome at hand and computing resources available

1. Find Overlapping Reads



Create local multiple alignments from the overlapping reads

1. Find Overlapping Reads



- Correct errors using multiple alignment

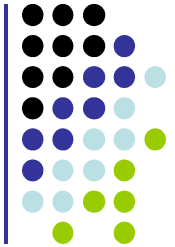
insert A

replace T with C

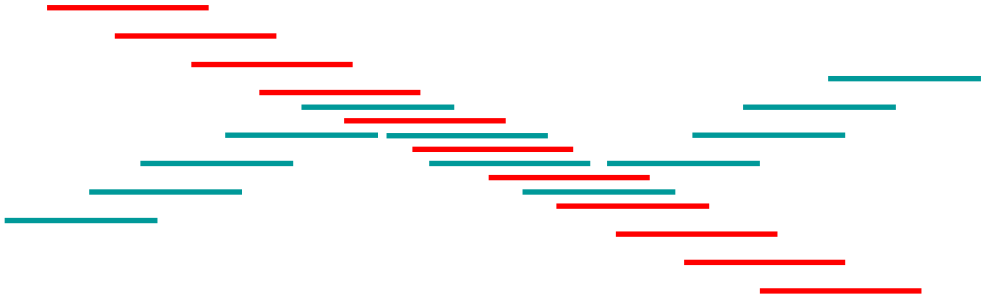
correlated errors—
probably caused by repeats
⇒ disentangle overlaps

In practice, error correction removes up to 98% of the errors

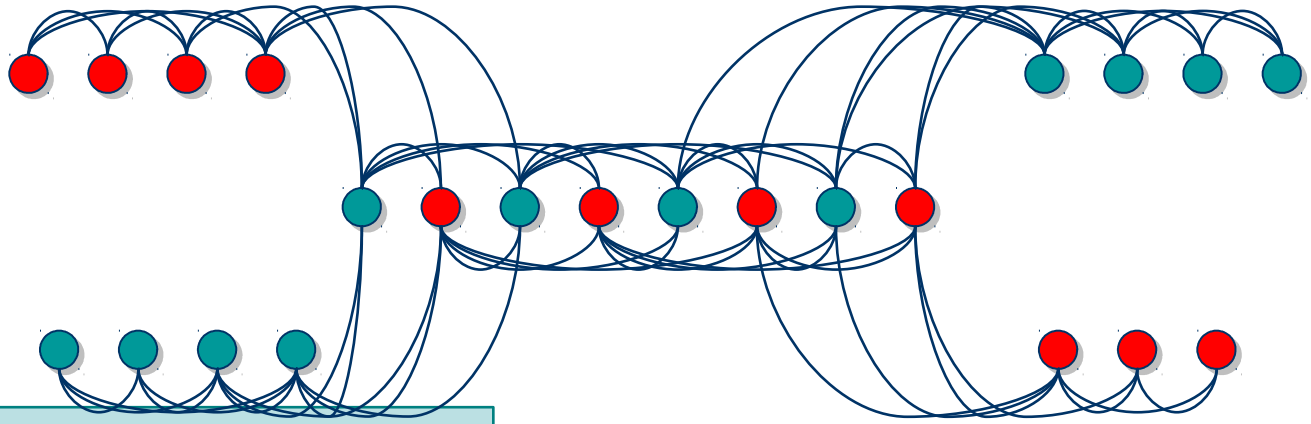
2. Merge Reads into Contigs



- Overlap graph:
  - Nodes: reads r_1…..r_n
  - Edges: overlaps (r_i, r_j, shift, orientation, score)

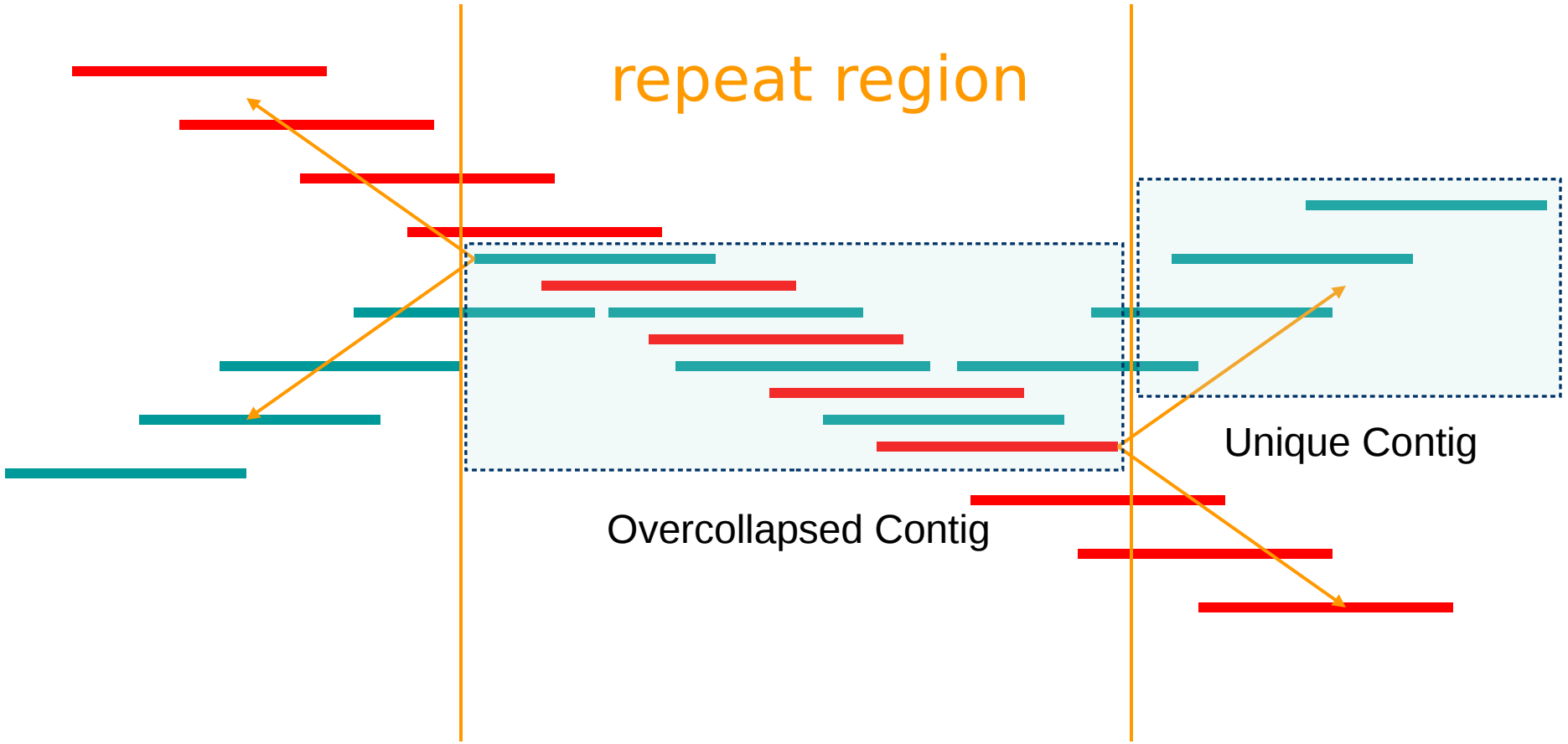


Reads that come from two regions of the genome (blue and red) that contain the same repeat



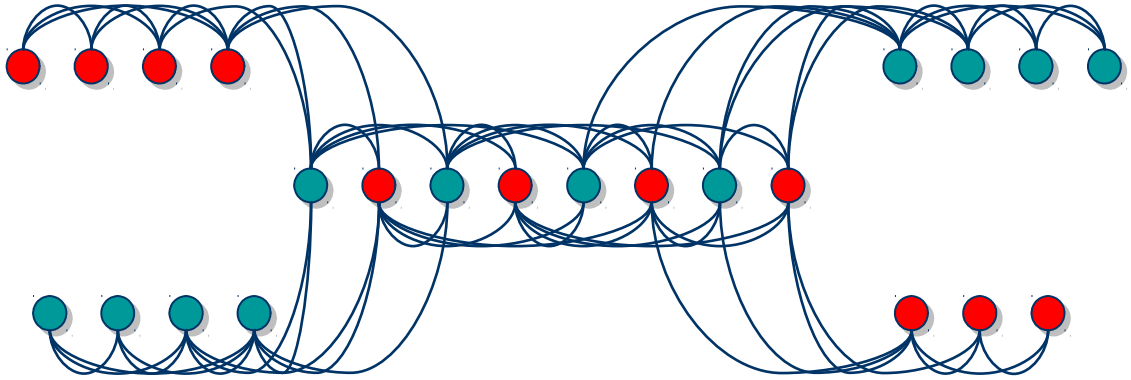
Note:
of course, we don't know the "color" of these nodes

2. Merge Reads into Contigs

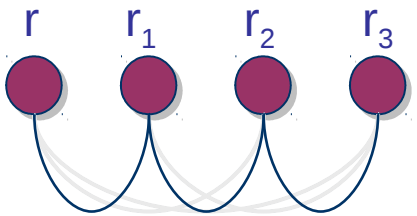


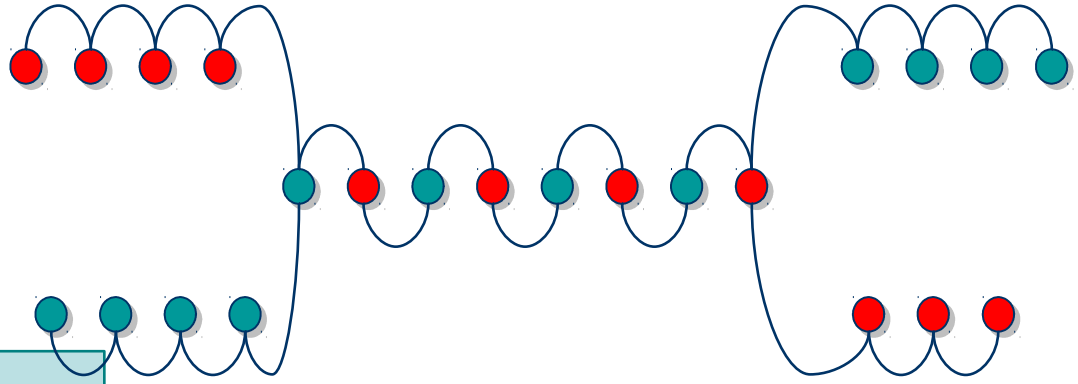
We want to merge reads up to potential repeat boundaries

2. Merge Reads into Contigs

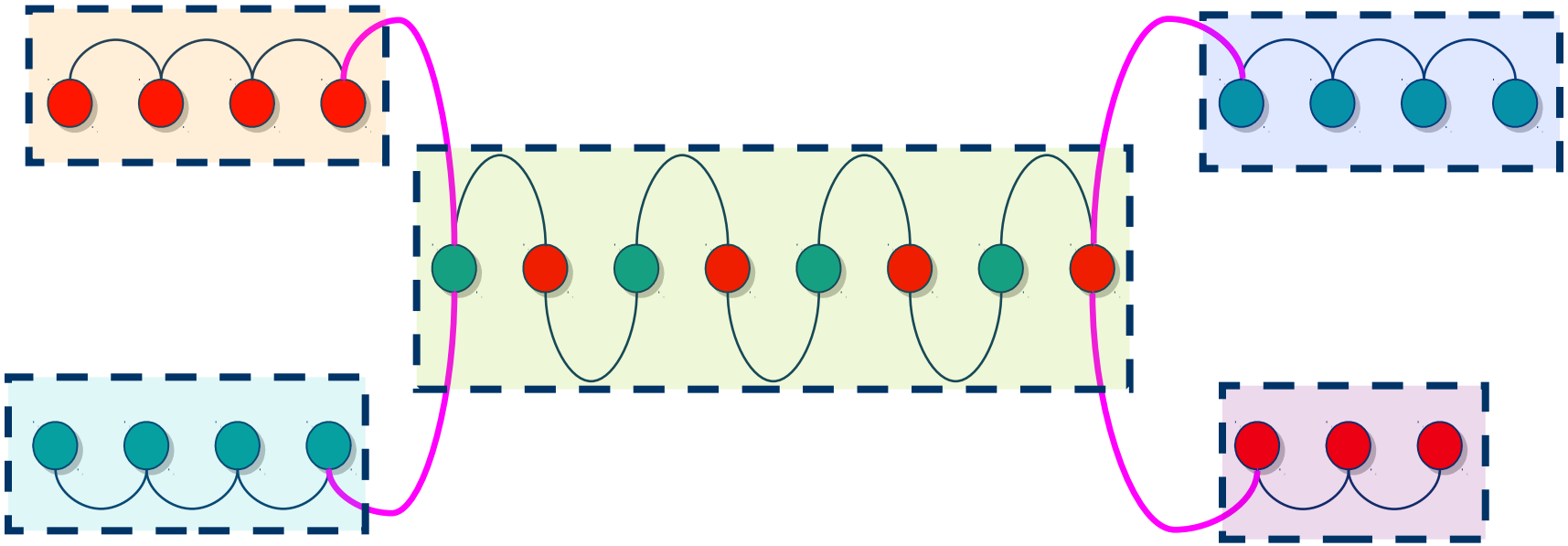


- Remove transitively inferable overlaps
  - If read r overlaps to the right reads r_1, r_2, and r_1 overlaps r_2, then (r, r_2) can be inferred by (r, r_1) and (r_1, r_2)

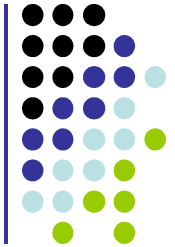




2. Merge Reads into Contigs



Repeats, errors, and contig lengths



- Repeats shorter than read length are easily resolved
  - Read that spans across a repeat disambiguates order of flanking regions
- Repeats with more base pair diffs than sequencing error rate are OK
  - We throw overlaps between two reads in different copies of the repeat
- To make the genome **appear** less repetitive, try to:
  - Increase read length
  - Decrease sequencing error rate

Role of error correction:

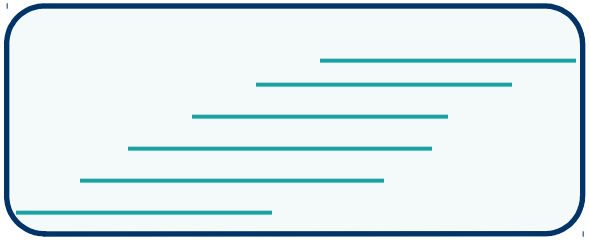
Discards up to 98% of single-letter sequencing errors

decreases error rate

⇒ decreases effective repeat content

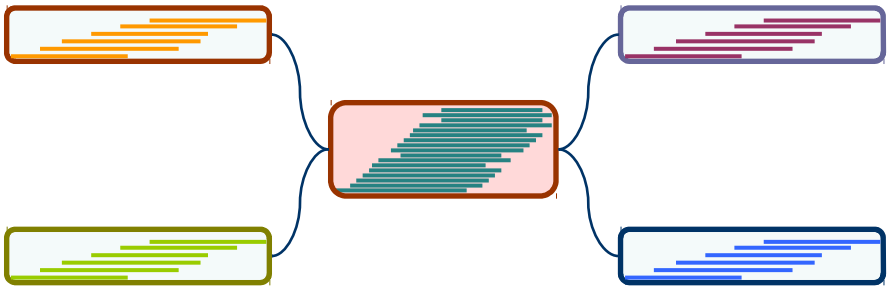
⇒ increases contig length

3. Link Contigs into Supercontigs



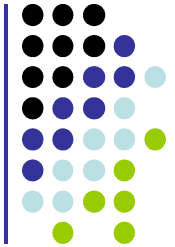
Normal density

Too dense
⇒ Overcollapsed



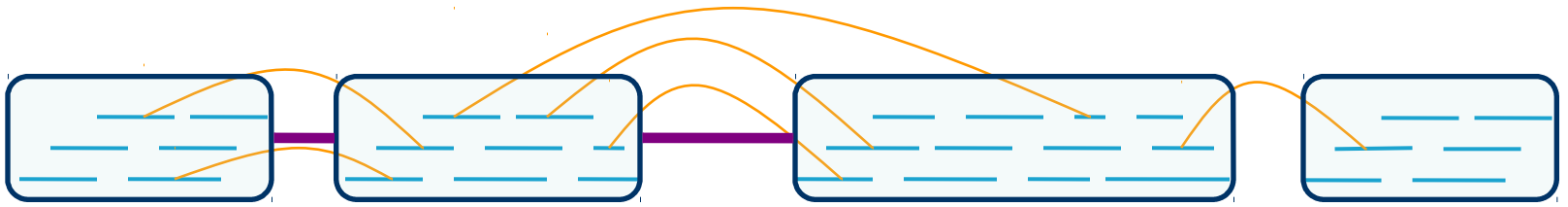
Inconsistent links
⇒ Overcollapsed?

3. Link Contigs into Supercontigs



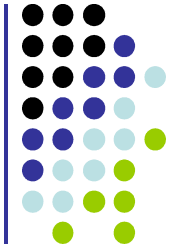
Find all links between unique contigs

Connect contigs incrementally, if ≥ 2 forward-reverse links



supercontig
(aka *scaffold*)

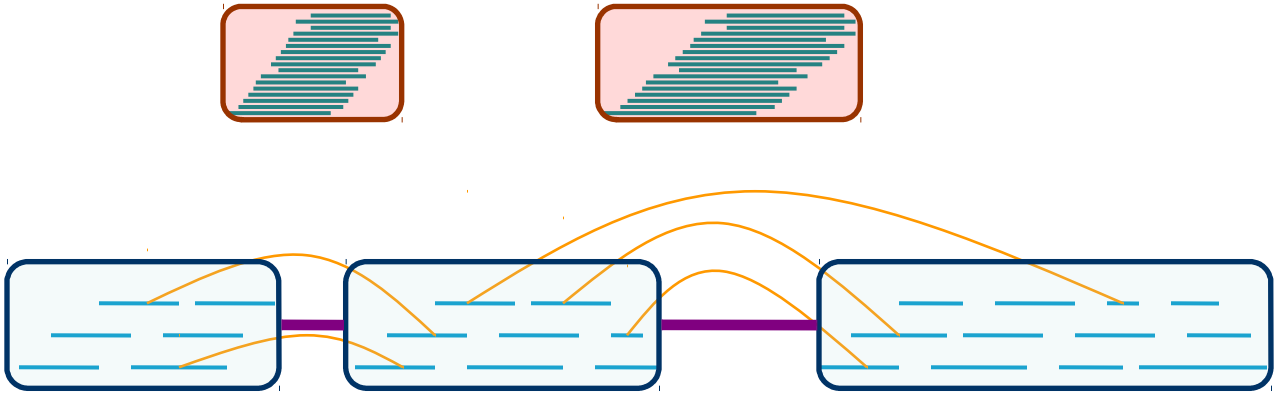
3. Link Contigs into Supercontigs



Fill gaps in supercontigs with paths of repeat contigs

Complex algorithmic step

- Exponential number of paths
- Forward-reverse links

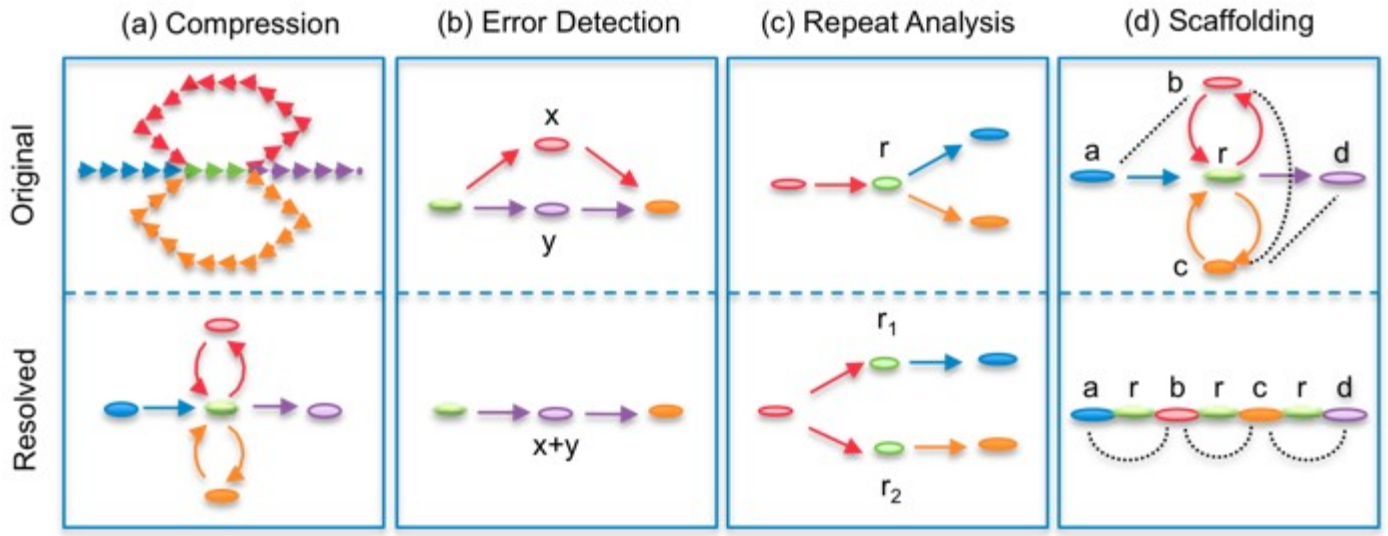


De Brujin Graph formulation

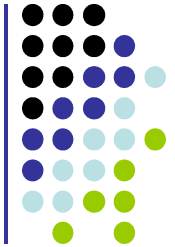
- Given sequence $x_1 \ldots x_N$, k-mer length k,

  Graph of 4^k vertices,

  Edges between words with (k-1)-long overlap



4. Derive Consensus Sequence



Derive multiple alignment from pairwise read alignments

Derive each consensus base by weighted voting

(**Alternative:** take maximum-quality letter)

Panda Genome

Table 1 | Summary of the panda genome sequencing and assembly

Step	Paired-end insert size (bp)*	Sequence coverage (×)†	Physical coverage (×)†	N50 (bp) ‡	N90 (bp) ‡	Total length (bp)
Initial contig	110–230; 380–570	38.5	96	1,483	224	2,021,639,596
Scaffold 1				32,648	7,780	2,213,848,409
Scaffold 2	Add 1,700–2,800	8.4	151	229,150	45,240	2,250,442,210
Scaffold 3	Add 3,700–7,500	6.5	450	581,933	127,336	2,297,100,301
Scaffold 4	Add 9,200–12,300	2.6	373	1,281,781	312,670	2,299,498,912
Final contig	All	56.0	1,070	39,886	9,848	2,245,302,481

Add denotes accumulative; for example, scaffold 2 uses data of 110–230, 380–570 and 1,700–2,800.

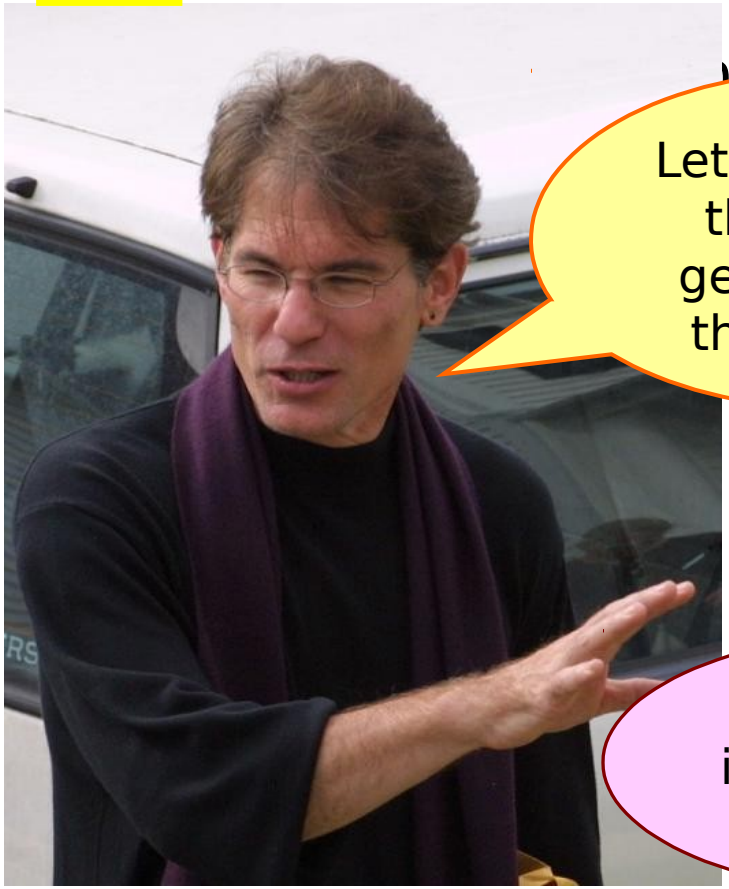
* Approximate average insert size of Illumina Genome Analyser sequencing libraries. The sizes were estimated by mapping the reads onto the assembled genome sequences.

† High-quality read sequences that were used in assembly. Coverage was estimated assuming a genome size of 2.4 Gb. Sequence coverage refers to the total length of generated reads, and physical coverage refers to the total length of sequenced clones of the libraries.

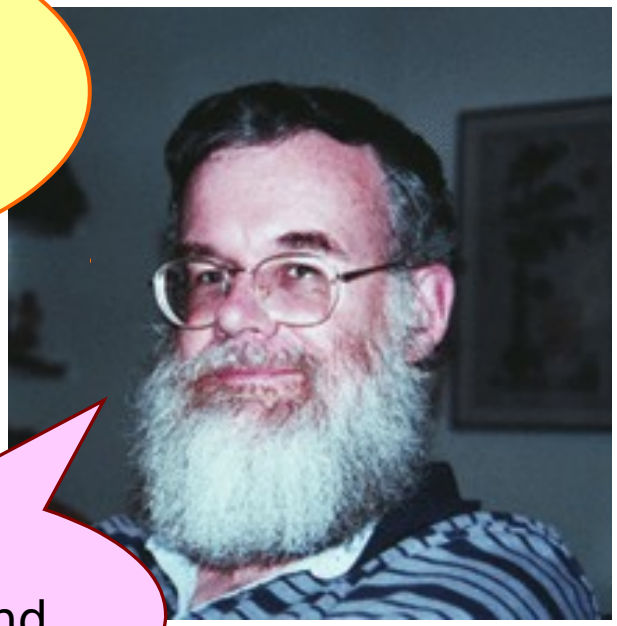
‡ N50 size of contigs or scaffolds was calculated by ordering all sequences then adding the lengths from longest to shortest until the summed length exceeded 50% of the total length of all sequences. N90 is similarly defined.

History of WGA

1997



Let's sequence the human genome with the shotgun strategy



That is impossible, and a bad idea anyway

Gene Myers

Phil Green