

Genome Assembly & Alignment Primer

Michael Schatz

Sept 27, 2012
Beyond the Genome



@mike_schatz / #BTG2012

Outline

1. Assembly by Analogy
2. Genome Assembly
 1. Coverage, read length, repeats, and errors
 2. Genome assemblers & Assemblathon
3. Whole genome alignment



Shredded Book Reconstruction

- Dickens accidentally shreds the first printing of A Tale of Two Cities
 - Text printed on 5 long spools

It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness, ...	
It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness, ...	
It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness, ...	
It was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness, ...	
It	was	the	best	of	times,	it	was	the	worst	of	times,	it	was	the	age	of	wisdom,	it	was	the	age	of	foolishness, ...

- How can he reconstruct the text?
 - 5 copies x 138,656 words / 5 words per fragment = 138k fragments
 - The short fragments from every copy are mixed together
 - Some fragments are identical

Greedy Reconstruction

It was the best of
age of wisdom, it was
best of times, it was
it was the age of
it was the age of
it was the worst of
of times, it was the
of times, it was the
of wisdom, it was the
the age of wisdom, it
the best of times, it
the worst of times, it
times, it was the age
times, it was the worst
was the age of wisdom,
was the age of foolishness,
was the best of times,
was the worst of times,
wisdom, it was the age
worst of times, it was

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

The repeated sequence make the correct reconstruction ambiguous

- It was the best of times, it was the [worst/age]

Model the assembly problem as a graph problem

de Bruijn Graph Construction

- $D_k = (V, E)$
 - $V =$ All length- k subfragments ($k < l$)
 - $E =$ Directed edges between consecutive subfragments
 - Nodes overlap by $k-1$ words

Original Fragment

It was the best of

Directed Edge

It was the best → was the best of

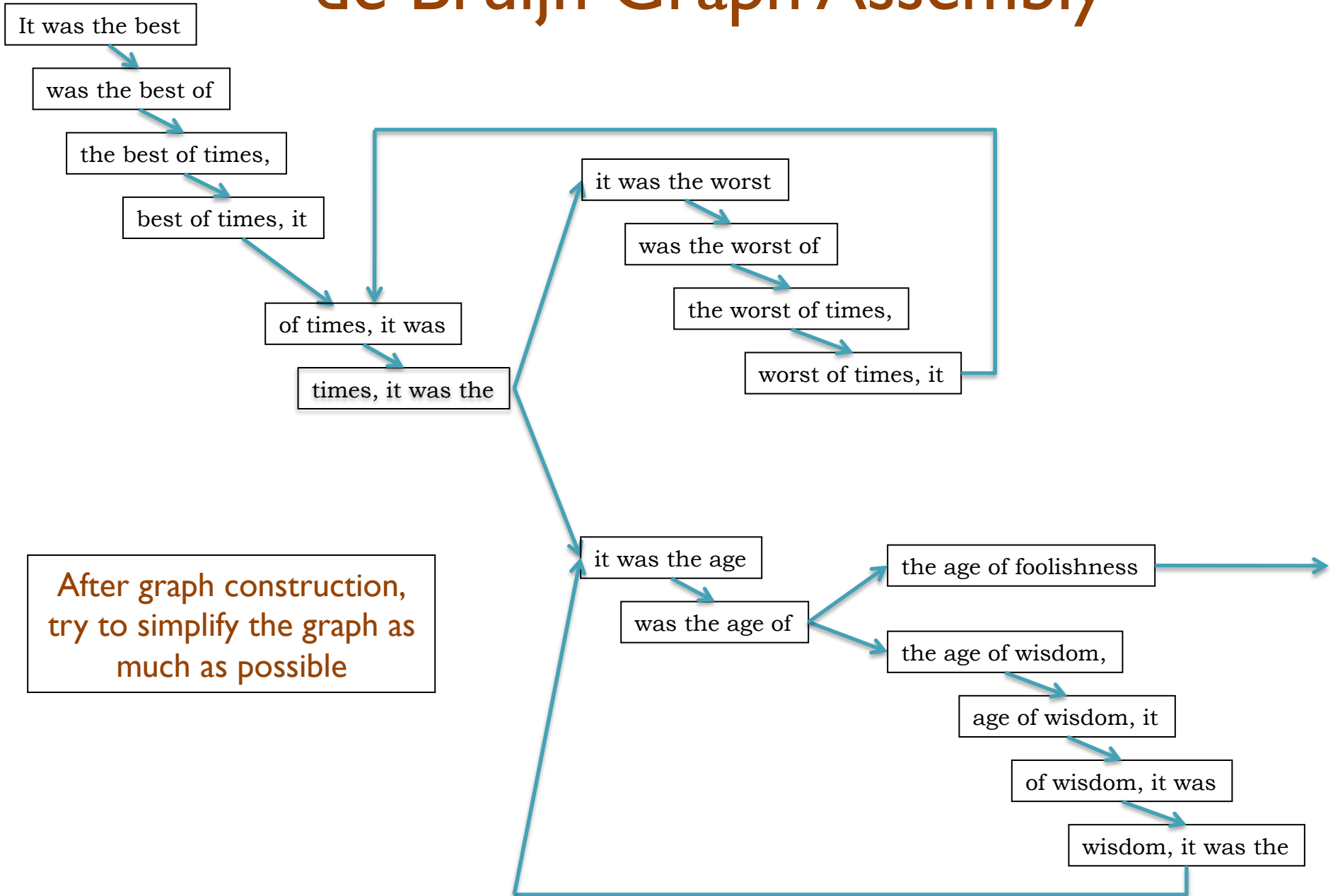
- Locally constructed graph reveals the global sequence structure
 - Overlaps between sequences implicitly computed

de Bruijn, 1946

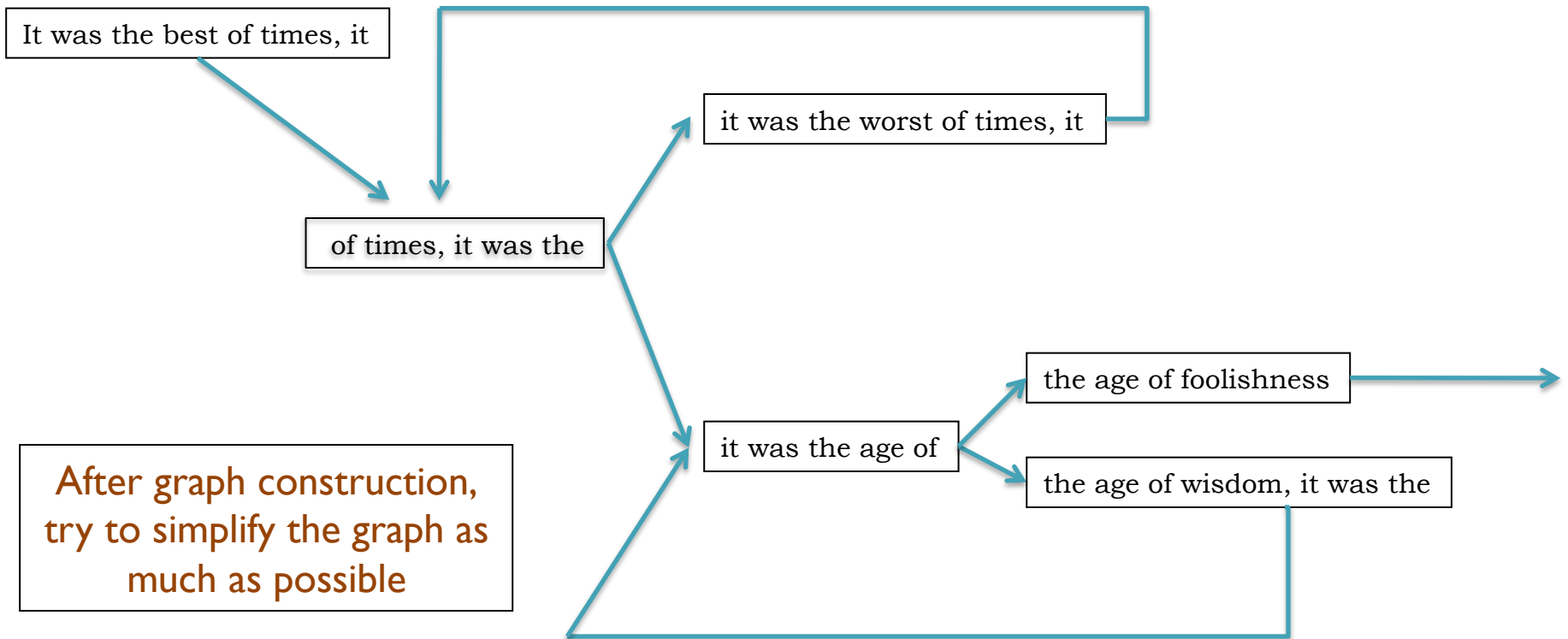
Idury and Waterman, 1995

Pevzner, Tang, Waterman, 2001

de Bruijn Graph Assembly



de Bruijn Graph Assembly



Assembly Applications

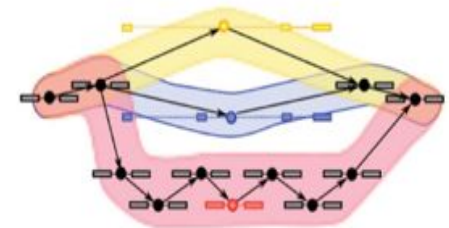
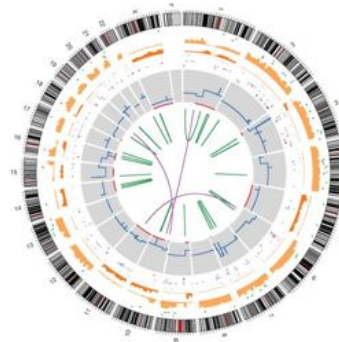
- Novel genomes



- Metagenomes



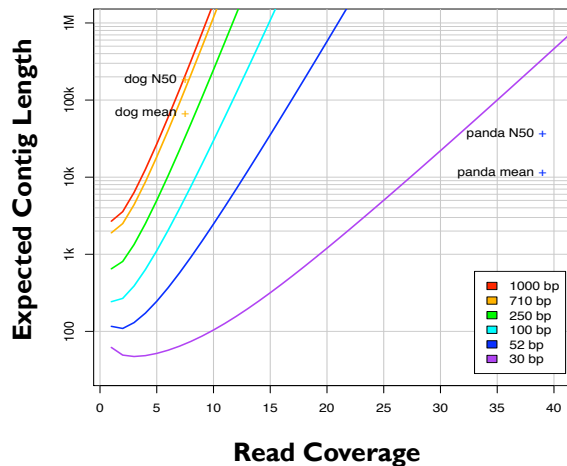
- Sequencing assays
 - Structural variations
 - Transcript assembly
 - ...



Like Dickens, we must computationally reconstruct a genome from short fragments

Ingredients for a good assembly

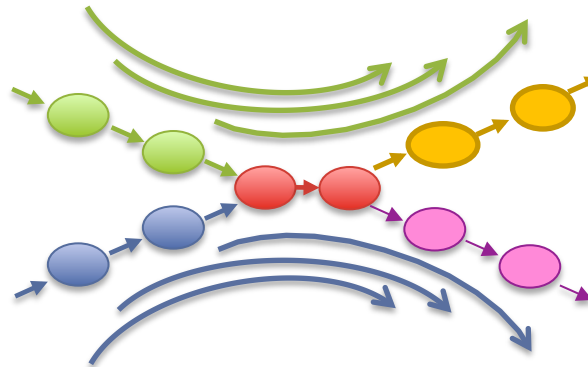
Coverage



High coverage is required

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

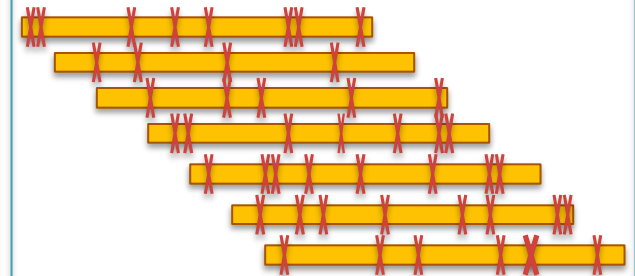
Read Length



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

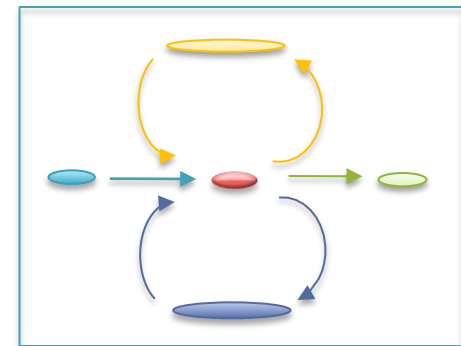
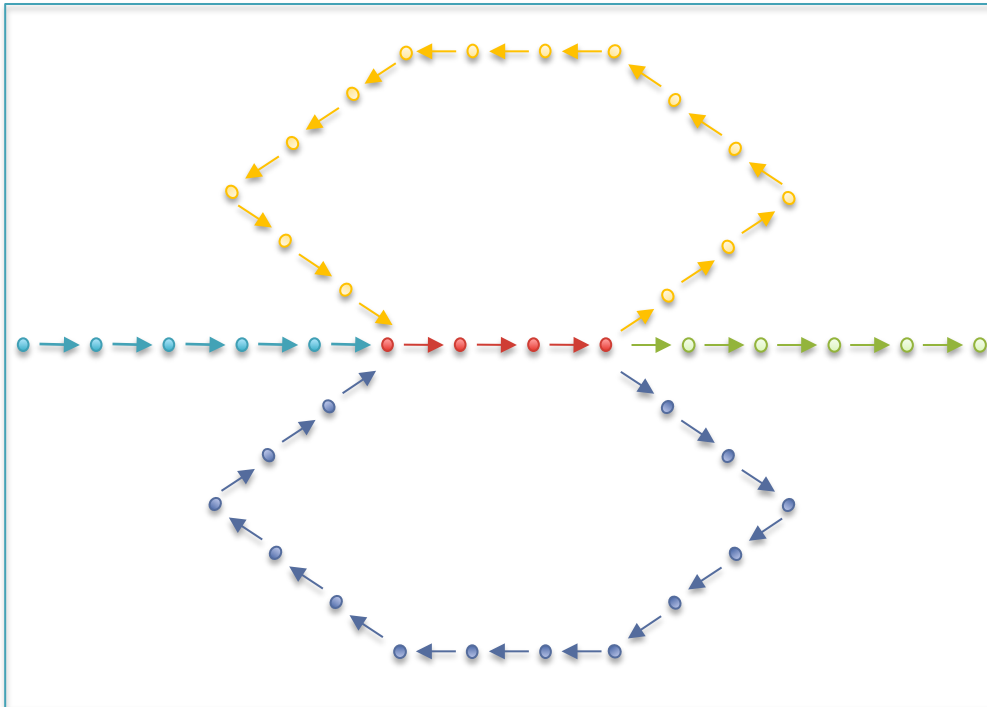
- Reads are assembled by finding kmers shared in pair of reads
- High error rate requires very short seeds, increasing complexity and forming assembly hairballs

Current challenges in *de novo* plant genome sequencing and assembly

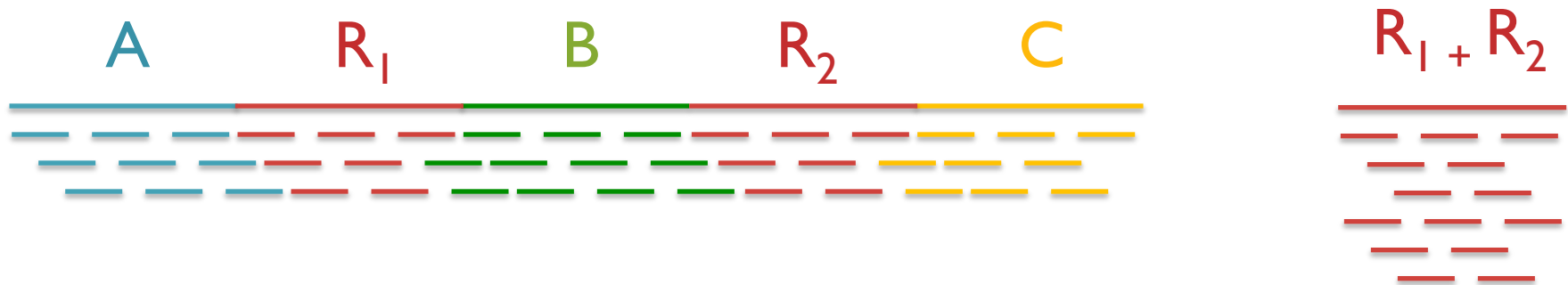
Schatz MC, Witkowski, McCombie, VWR (2012) *Genome Biology*. 12:243

Initial Contigs

- After constructing assembly graph, compress graph down to its non-branching initial contigs
 - Aka “unitigs”, “unipaths”



Repeats and Coverage Statistics



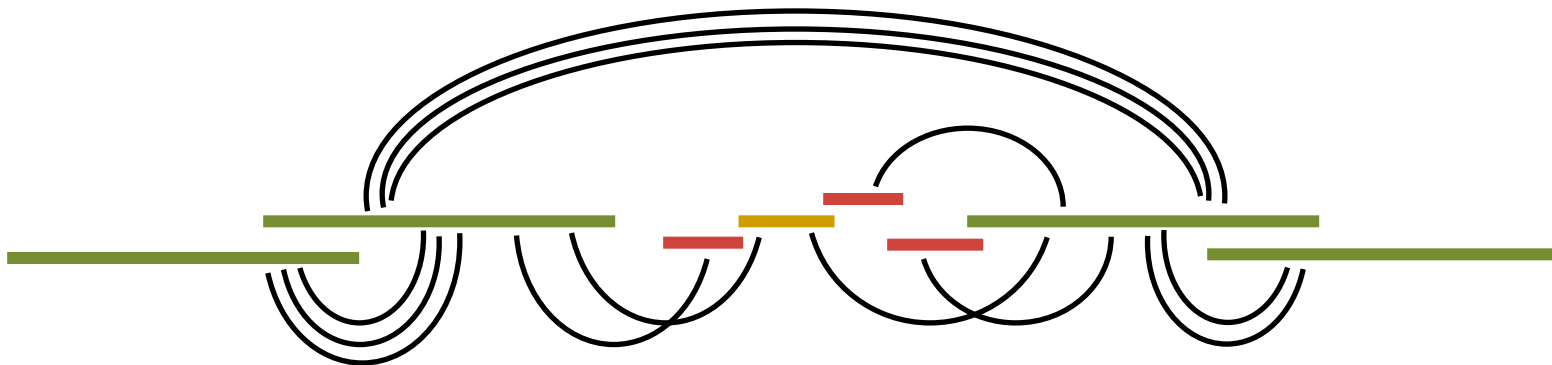
- If n reads are a uniform random sample of the genome of length G , we expect $k = n \Delta / G$ reads to start in a region of length Δ .
 - If we see many more reads than k (if the arrival rate is $> \lambda$), it is likely to be a collapsed repeat
 - Requires an accurate genome size estimate

$$\Pr(X - \text{copy}) = \binom{n}{k} \left(\frac{X\Delta}{G} \right)^k \left(\frac{G - X\Delta}{G} \right)^{n-k}$$

$$A(\Delta, k) = \ln \left(\frac{\Pr(1 - \text{copy})}{\Pr(2 - \text{copy})} \right) = \ln \left(\frac{\frac{(\Delta n / G)^k e^{-\frac{\Delta n}{G}}}{k!}}{\frac{(2\Delta n / G)^k e^{-\frac{2\Delta n}{G}}}{k!}} \right) = \frac{n\Delta}{G} - k \ln 2$$

Scaffolding

- Initial contigs (*aka* unipaths, unitigs) terminate at
 - *Coverage gaps*: especially extreme GC regions
 - *Conflicts*: sequencing errors, repeat boundaries
- Iteratively resolve longest, ‘most unique’ contigs
 - Both overlap graph and de Bruijn assemblers initially collapse repeats into single copies
 - Uniqueness measured by a statistical test on coverage



N50 size

Def: 50% of the genome is in contigs larger than N50

Example: 1 Mbp genome

50%



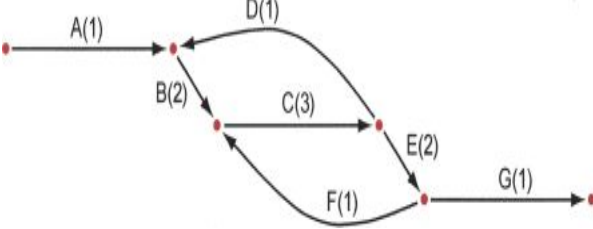
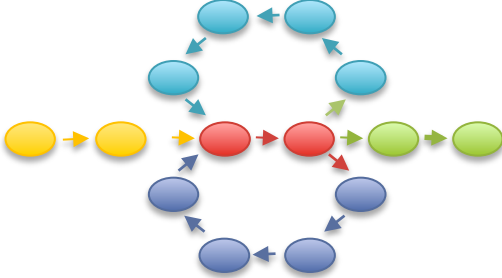
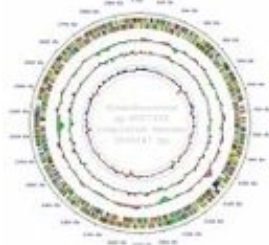
N50 size = 30 kbp

(300k + 100k + 45k + 45k + 30k = 520k \geq 500kbp)

Note:

N50 values are only meaningful to compare when base genome size is the same in all cases

Assembly Algorithms

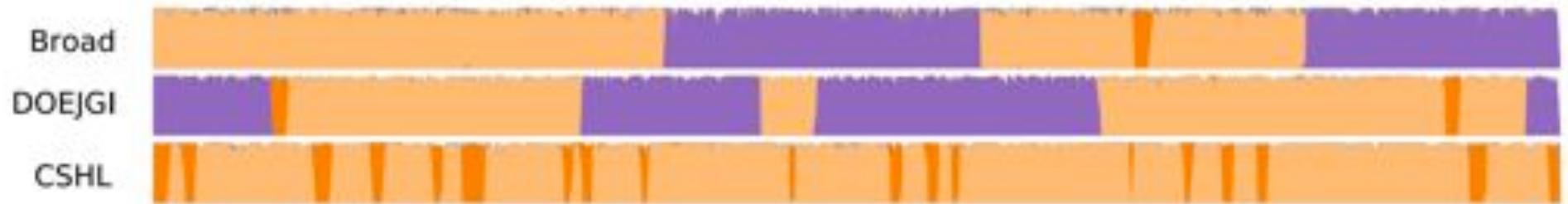
ALLPATHS-LG	SOAPdenovo	Celera Assembler
		
<p>Broad's assembler (Gnerre et al. 2011)</p>	<p>BGI's assembler (Li et al. 2010)</p>	<p>JCVI's assembler (Miller et al. 2008)</p>
<p>De bruijn graph Short + PacBio (patching)</p>	<p>De bruijn graph Short reads</p>	<p>Overlap graph Medium + Long reads</p>
<p>Easy to run if you have compatible libraries</p>	<p>Most flexible, but requires a lot of tuning</p>	<p>Supports Illumina/454/PacBio Hybrid assemblies</p>
<p>http://www.broadinstitute.org/ software/allpaths-lg/blog/</p>	<p>http://soap.genomics.org.cn/ soapdenovo.html</p>	<p>http://wgs-assembler.sf.net</p>

THE ASSEMBLATHON

- Attempt to answer the question:
“What makes a good assembly?”
- Organizers provided simulated sequence data
 - Simulated 100 base pair Illumina reads from simulated diploid organism
- 41 submissions from 17 groups
- Results demonstrate trade-offs assemblers must make

Assembly Results

Scaffolds



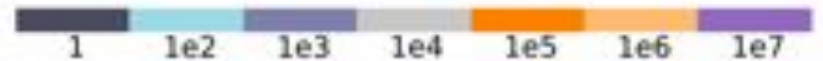
Scaffold Paths



Contig Paths



Fill Color Key
Item \geq

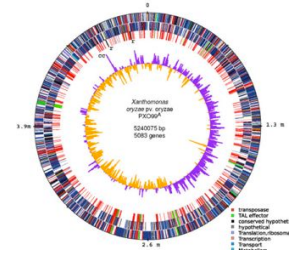


Final Rankings

ID	Overall	CPNG50	SPNG50	Struct.	CC50	Subs.	Copy. Num.	Cov. Tot.	Cov. CDS
BGI	36	★					★	★	★
Broad	37	★	★	★	★				
WTSI-S	46		★	★	★	★			
CSHL	52	★							★
BCCGSC	53							★	★
DOEJGI	56		★	★	★	★			
RHUL	58								
WTSI-P	64							★	
EBI	64						★		
CRACS	64					★			

- SOAPdenovo and ALLPATHS came out neck-and-neck followed closely behind by SGA, Celera Assembler, ABySS
- My recommendation for “typical” short read assembly is to use ALLPATHS

Assembly Summary



Assembly quality depends on

- 1. Coverage:** low coverage is mathematically hopeless
 - 2. Repeat composition:** high repeat content is challenging
 - 3. Read length:** longer reads help resolve repeats
 - 4. Error rate:** errors reduce coverage, obscure true overlaps
- Assembly is a hierarchical, starting from individual reads, build high confidence contigs/unitigs, incorporate the mates to build scaffolds
 - Extensive error correction is the key to getting the best assembly possible from a given data set
 - Watch out for collapsed repeats & other misassemblies
 - Globally/Locally reassemble data from scratch with better parameters & stitch the 2 assemblies together



Whole Genome Alignment with MUMmer

Slides Courtesy of Adam M. Phillippy

amp@umics.umd.edu

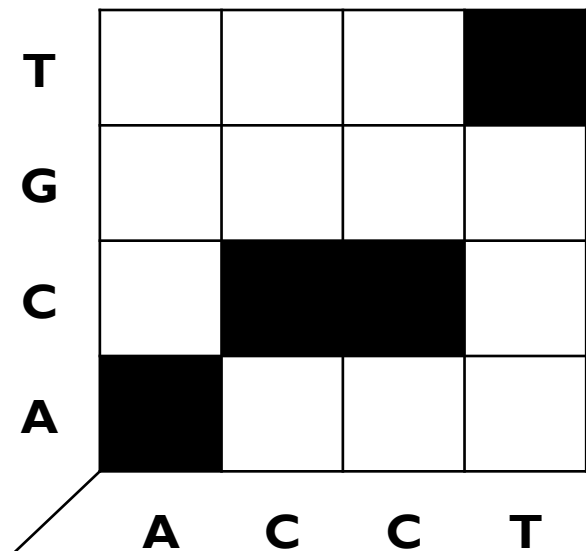
WGA visualization

- How can we visualize *whole* genome alignments?

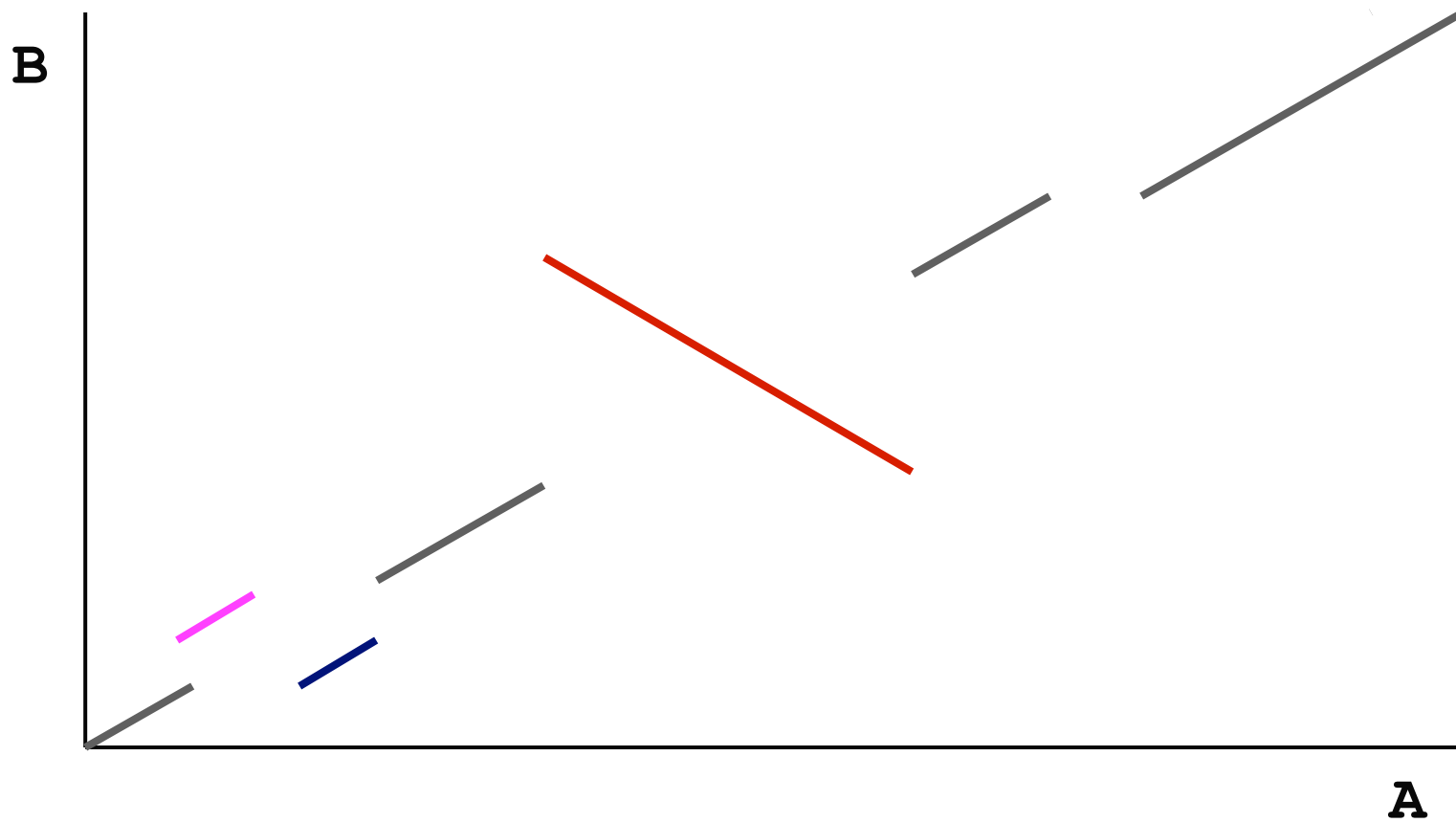
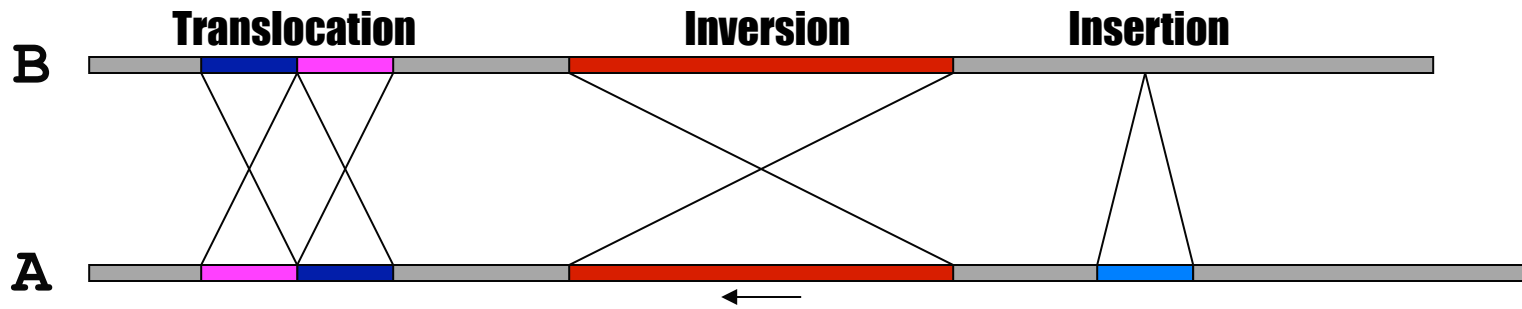
- With an alignment dot plot

- $N \times M$ matrix

- Let i = position in genome A
 - Let j = position in genome B
 - Fill cell (i,j) if A_i shows similarity to B_j



- A perfect alignment between A and B would completely fill the positive diagonal



Seed and Extend

How can quickly find large alignments?

1. Find short exact matches

- ◆ using a suffix tree

2. Cluster exact matches

- ◆ using size, gap and distance parameters

3. Extend clusters & report alignments

- ◆ using modified Smith-Waterman algorithm

WGA example with nucmer

Yersina pestis CO92 vs. *Yersina pestis* KIM

- High nucleotide similarity, 99.86%
 - Two strains of the same species
- Extensive genome shuffling and highly repetitive
 - Global alignment will not work

```
nucmer -maxmatch CO92.fasta KIM.fasta
```

```
-maxmatch    Find maximal exact matches (MEMs)
```

```
delta-filter -1 out.delta > out.filter.m
```

```
-n           Many-to-many mapping
```

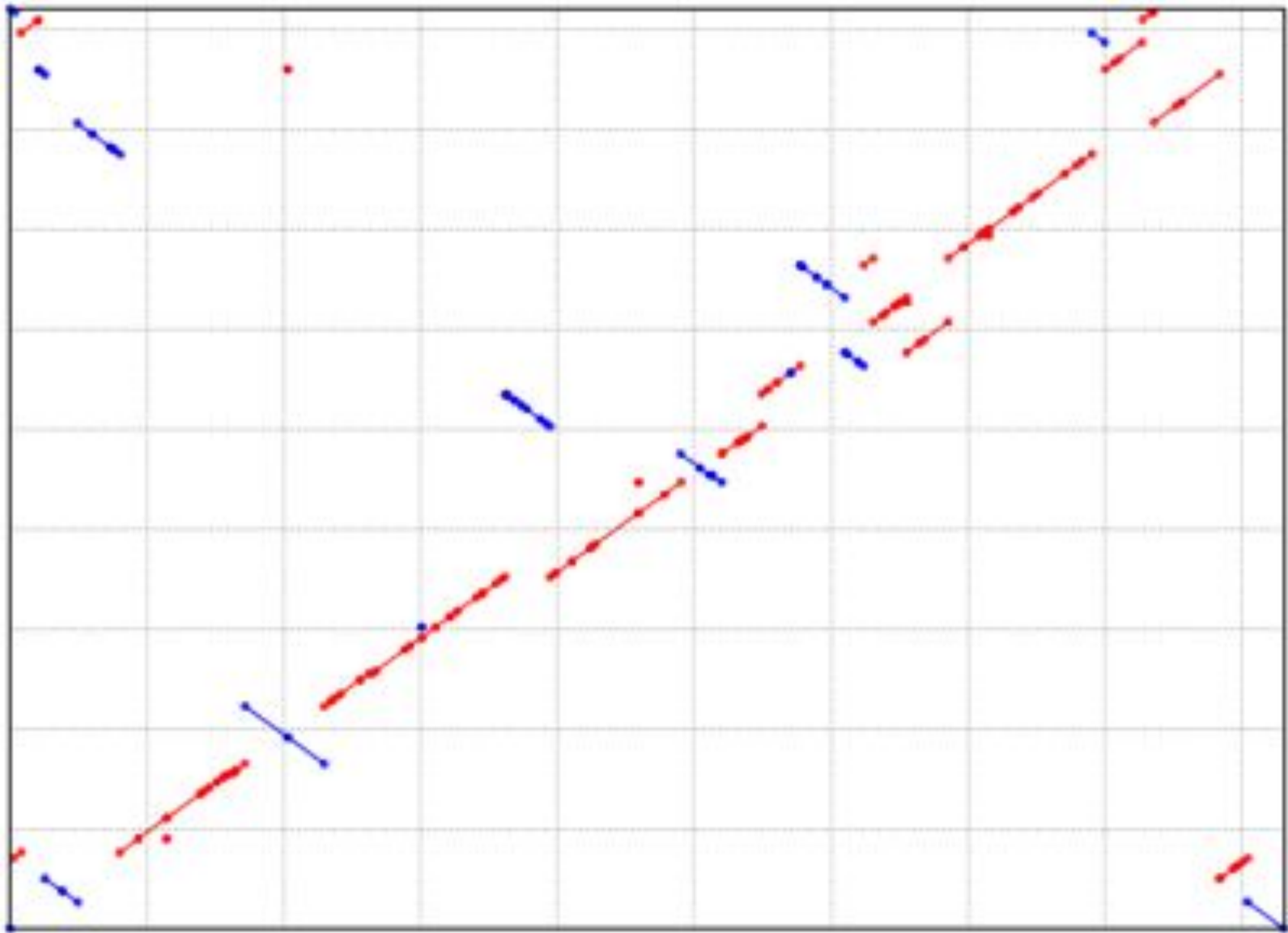
```
show-coords -r out.delta.m > out.coords
```

```
-r           Sort alignments by reference position
```

```
mummerplot --large --layout out.delta.m
```

```
--large      Large plot
```

```
--layout     Nice layout for multi-fasta files
```



<http://mummer.sourceforge.net>

Thank You

<http://schatzlab.cshl.edu>
@mike_schatz / #BTG2012

