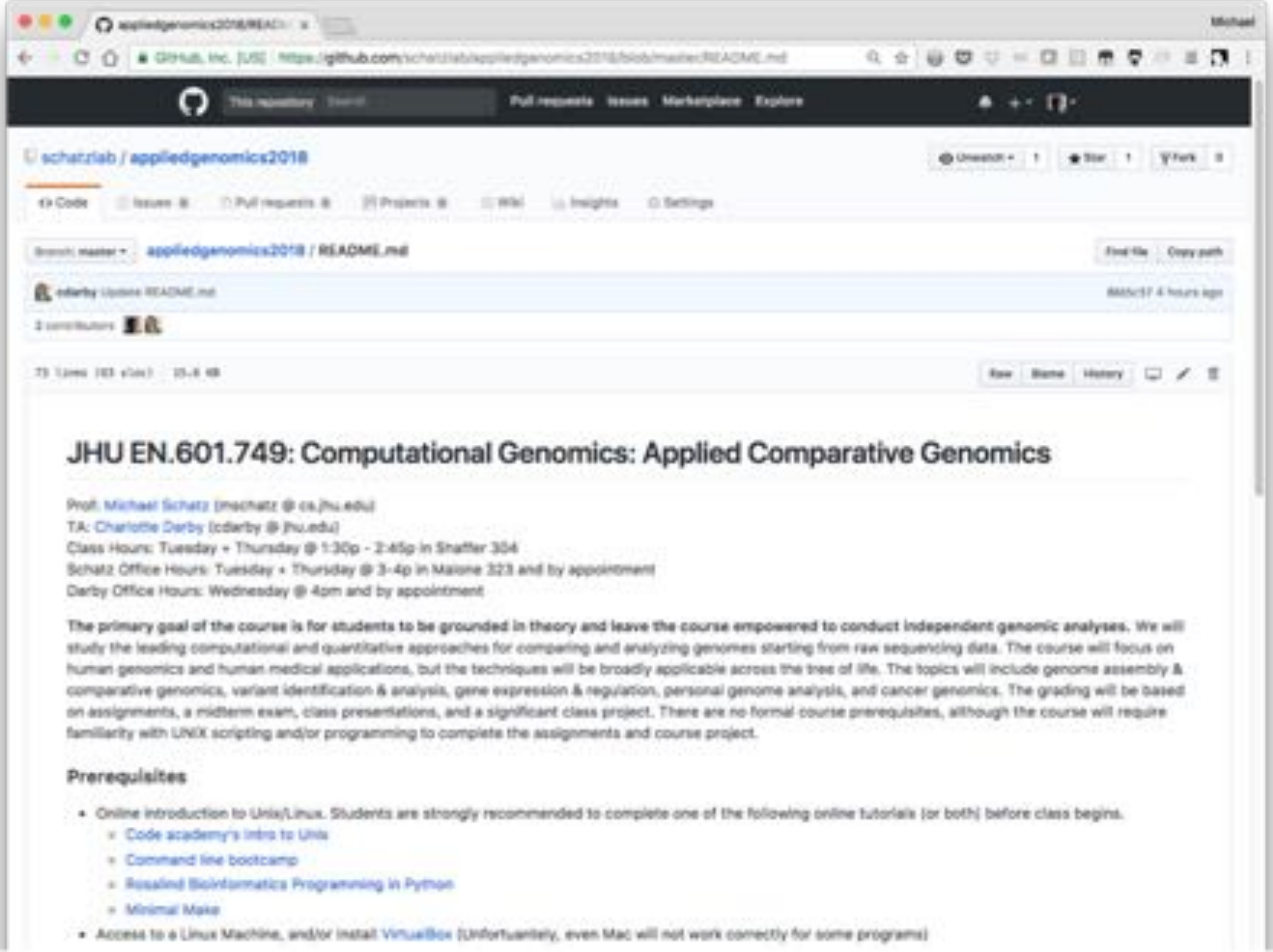
Genome Assembly

Michael Schatz

Feb 6, 2018

Lecture 3: Applied Comparative Genomics

Course Webpage



JHU EN.601.749: Computational Genomics: Applied Comparative Genomics

Prof. Michael Schatz (mschatz @ cs.jhu.edu)
TA: Charlotte Darby (cdarby @ jhu.edu)
Class Hours: Tuesday + Thursday @ 1:30p - 2:45p in Shaffer 304
Schatz Office Hours: Tuesday + Thursday @ 3-4p in Malone 323 and by appointment
Darby Office Hours: Wednesday @ 4pm and by appointment

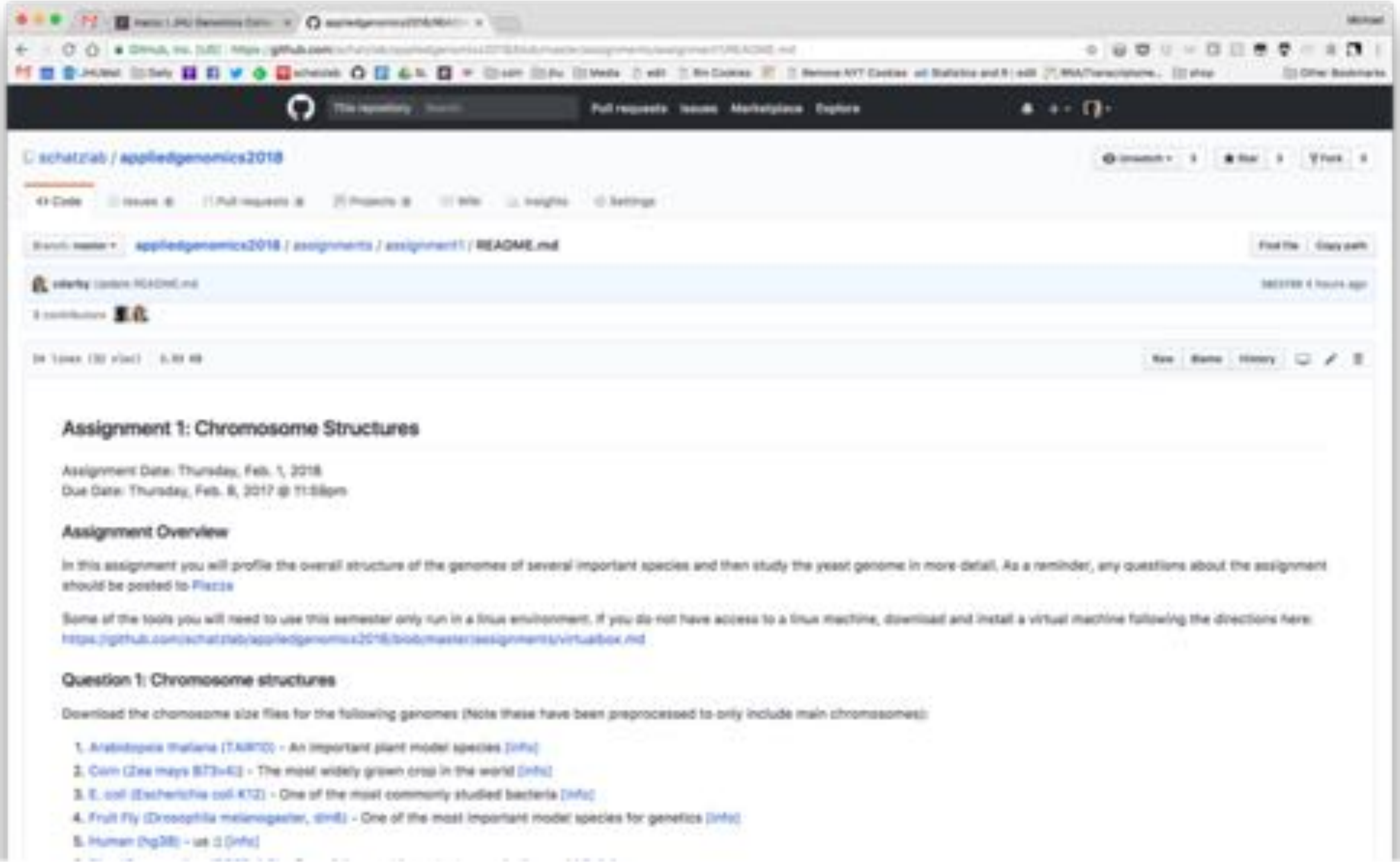
The primary goal of the course is for students to be grounded in theory and leave the course empowered to conduct independent genomic analyses. We will study the leading computational and quantitative approaches for comparing and analyzing genomes starting from raw sequencing data. The course will focus on human genomics and human medical applications, but the techniques will be broadly applicable across the tree of life. The topics will include genome assembly & comparative genomics, variant identification & analysis, gene expression & regulation, personal genome analysis, and cancer genomics. The grading will be based on assignments, a midterm exam, class presentations, and a significant class project. There are no formal course prerequisites, although the course will require familiarity with UNIX scripting and/or programming to complete the assignments and course project.

Prerequisites

- Online introduction to Unix/Linux. Students are strongly recommended to complete one of the following online tutorials (or both) before class begins.
  - Code academy's Intro to Unix
  - Command line bootcamp
  - Rosalind Bioinformatics Programming in Python
  - Minimal Make
- Access to a Linux Machine, and/or install VirtualBox (Unfortuantely, even Mac will not work correctly for some programs)

https://github.com/schatzlab/appliedgenomics2018

Assignment 1: Chromosome Structures
Due Feb 8 @ 11:59pm


schatzlab / appliedgenomics2018
Code
Issues
Pull requests
Projects
Wiki
Insights
Settings
appliedgenomics2018 / assignments / assignment1 / README.md
Raw
Blame
History

Assignment 1: Chromosome Structures

Assignment Date: Thursday, Feb. 1, 2018
Due Date: Thursday, Feb. 8, 2017 @ 11:59pm

Assignment Overview

In this assignment you will profile the overall structure of the genomes of several important species and then study the yeast genome in more detail. As a reminder, any questions about the assignment should be posted to Piazza

Some of the tools you will need to use this semester only run in a linux environment. If you do not have access to a linux machine, download and install a virtual machine following the directions here: https://github.com/schatzlab/appliedgenomics2018/blob/master/assignments/virtualbox.md

Question 1: Chromosome structures

Download the chomosome size files for the following genomes (Note these have been preprocessed to only include main chromosomes):

1. Arabidopsis thaliana (TAIR10) - An important plant model species [info]
2. Corn (Zea mays B73v4) - The most widely grown crop in the world [info]
3. E. coli (Escherichia coli K12) - One of the most commonly studied bacteria [info]
4. Fruit Fly (Drosophila melanogaster, dm6) - One of the most important model species for genetics [info]
5. Human (hg38) - us :) [info]

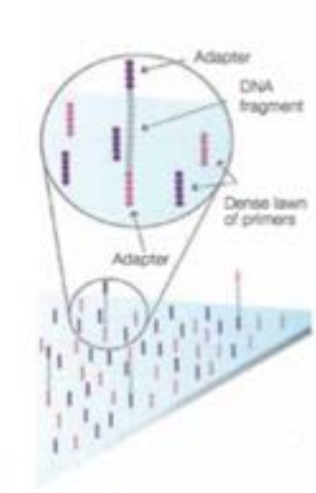
https://github.com/schatzlab/appliedgenomics2018

Part 1: Recap

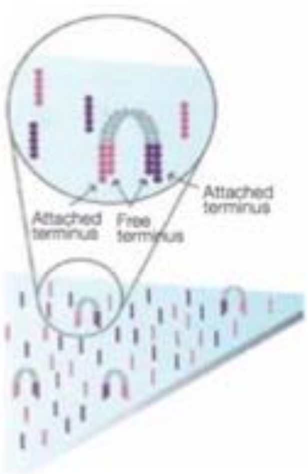
Second Generation Sequencing

Illumina HiSeq 2000
Sequencing by Synthesis

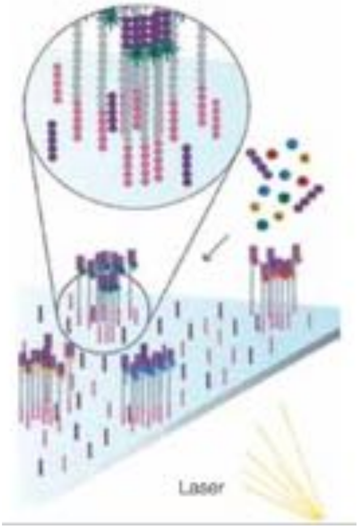
>60Gbp / day



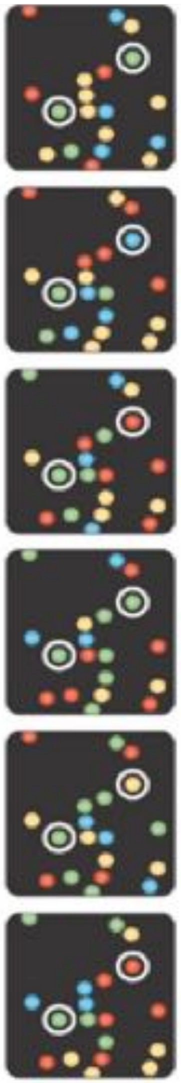
1. Attach



2. Amplify



3. Image

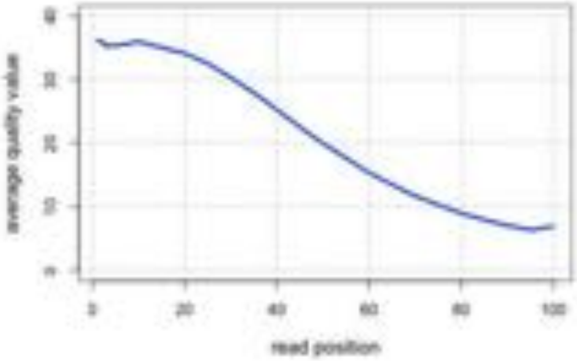


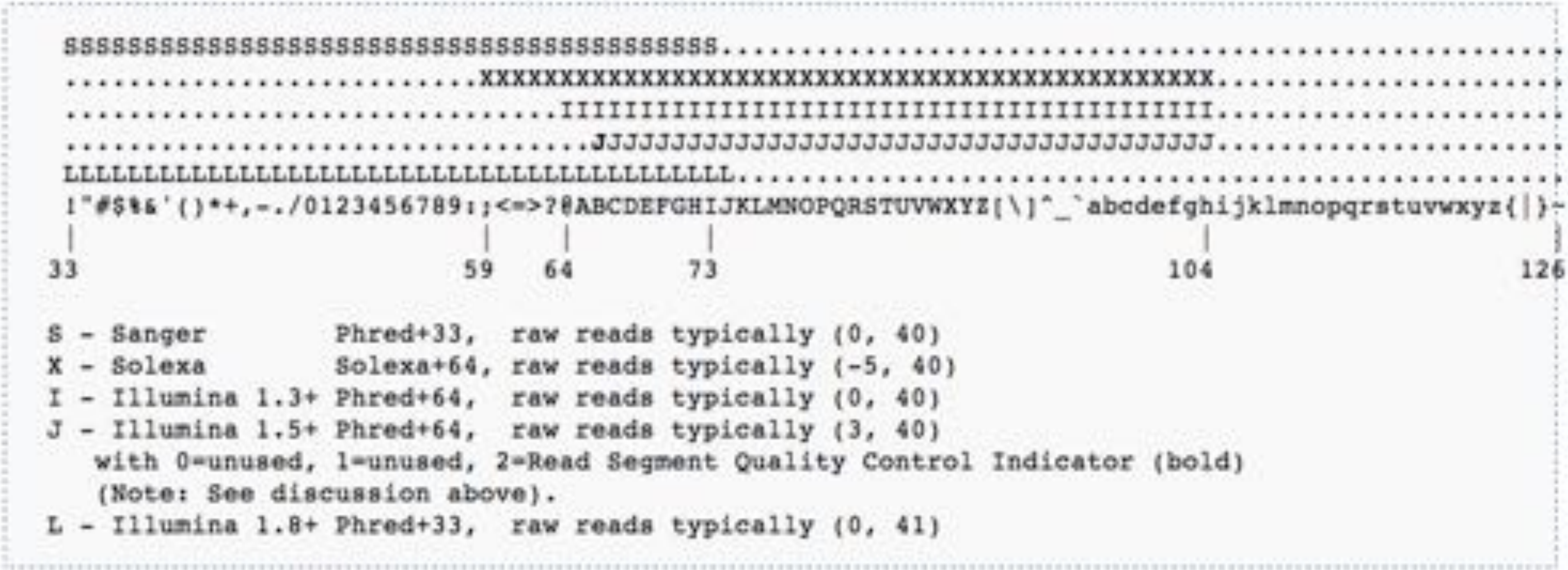
Metzker (2010) Nature Reviews Genetics 11:31-46
https://www.youtube.com/watch?v=fCd6B5HRaZ8

Illumina Quality

QV	P_{error}
40	1/10000
30	1/1000
20	1/100
10	1/10

$$Q_{\text{sanger}} = -10\ \log_{10} p$$



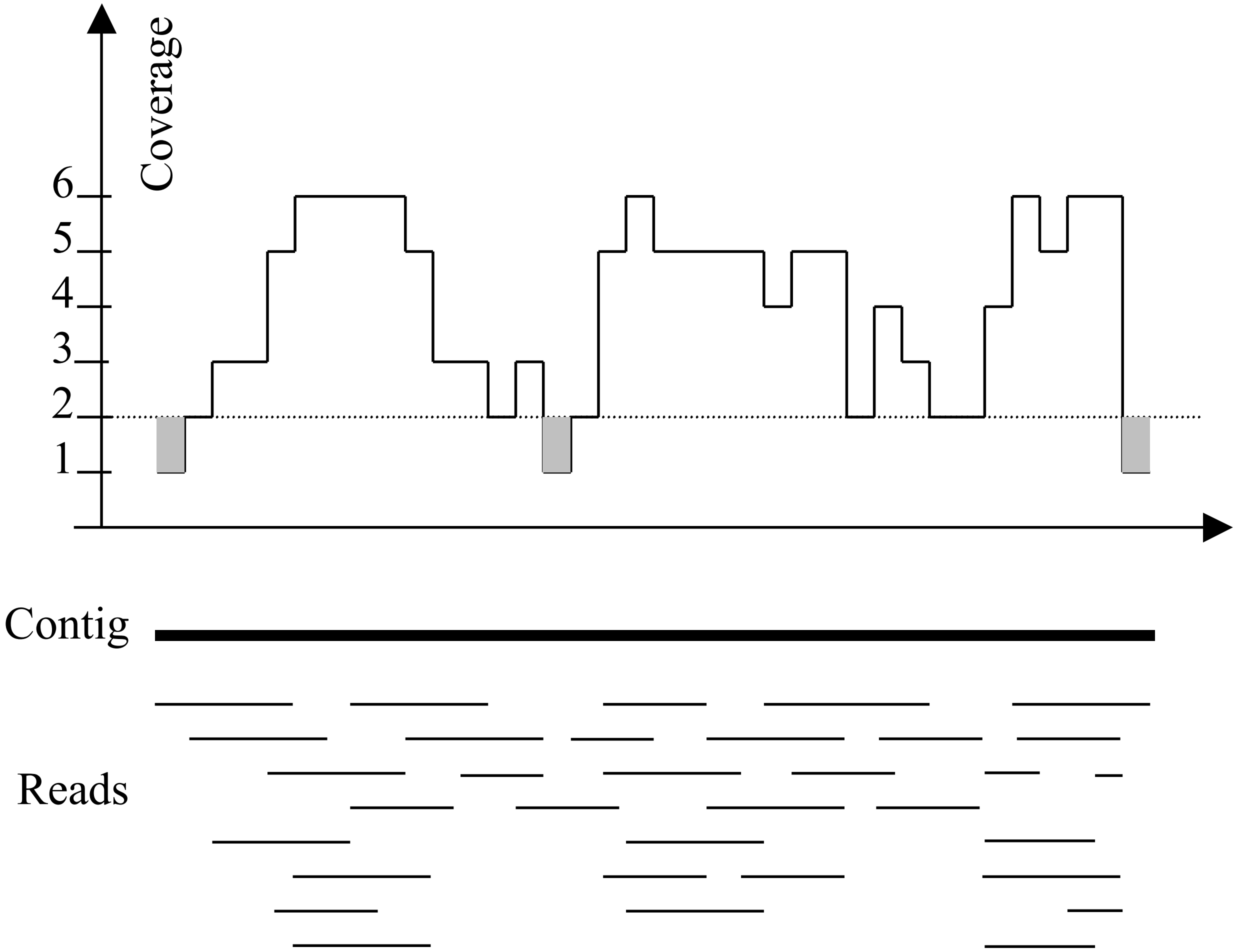


```
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
...............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
.................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL....................................................
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
|                         |    |        |                              |                     |
33                       59   64       73                            104                   126

S - Sanger        Phred+33,  raw reads typically (0, 40)
X - Solexa        Solexa+64, raw reads typically (-5, 40)
I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
   with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
   (Note: See discussion above).
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

http://en.wikipedia.org/wiki/FASTQ_format

Typical sequencing coverage



Imagine raindrops on a sidewalk

We want to cover the entire sidewalk but each drop costs $1

If the genome is 10 Mbp, should we sequence 100k 100bp reads?

Illumina Sequencing Summary

Advantages:

- Best throughput, accuracy and read length for any 2nd gen. sequencer
- Fast & robust library preparation

Disadvantages:

- Inherent limits to read length (practically, 150bp)
- Some runs are error prone
- Requires amplification, sequences a population of molecules

Illumina HiSeq
~3 billion paired 100bp reads
~600Gb, $10K, 8 days
(or “rapid run” ~90Gb in 1-2 days)

Illumina X Ten
~6 billion paired 150bp reads
1.8Tb, <3 days, ~1000 / genome($$)
(or “rapid run” ~90Gb in 1-2 days)

Illumina NextSeq
One human genome in **<30 hours**

Ira Hall

Part 2: De novo genome assembly

Outline

1. ***Assembly theory***
   - Assembly by analogy

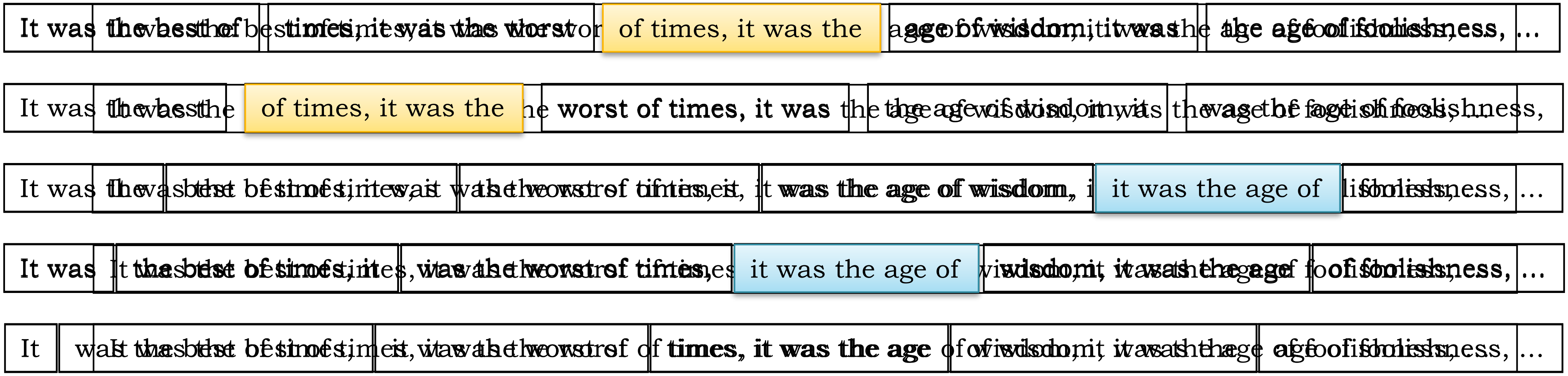
2. ***Practical Issues***
   - Coverage, read length, errors, and repeats

3. ***Next-next-gen Assembly***
   - Canu: recommended for PacBio/ONT project

4. ***Whole Genome Alignment***
   - MUMmer recommended

Shredded Book Reconstruction

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



- 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,

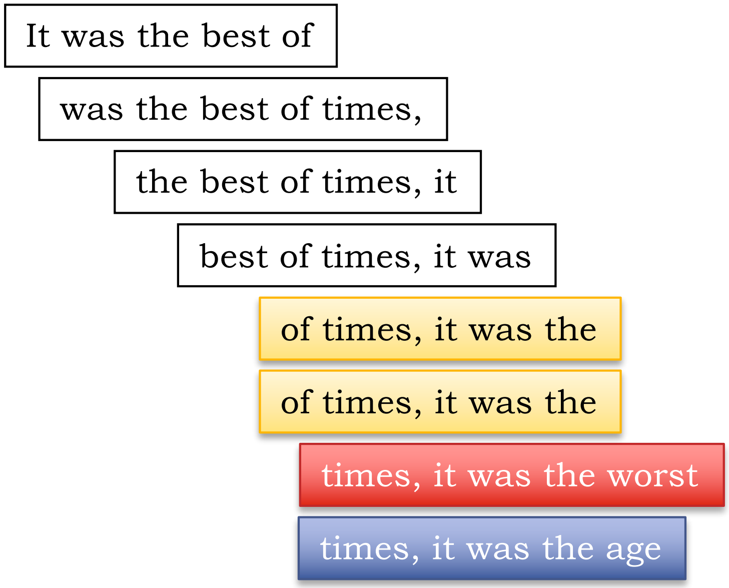
was the age of foolishness,

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]

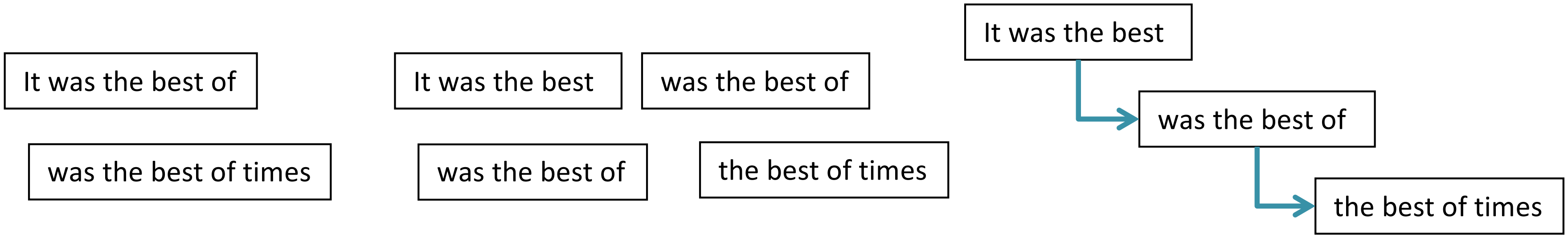
Model the assembly problem as a graph problem

How long will it take to compute the overlaps?

de Bruijn Graph Construction

- $G_k = (V,E)$
  - V = Length-*k* sub-fragments
  - E = Directed edges between consecutive sub-fragments
    - Sub-fragments overlap by *k*-1 words

Fragments |f|=5 Sub-fragment *k*=4 Directed edges (overlap by *k*-1)



It was the best of

was the best of times

It was the best

was the best of

was the best of

the best of times

It was the best

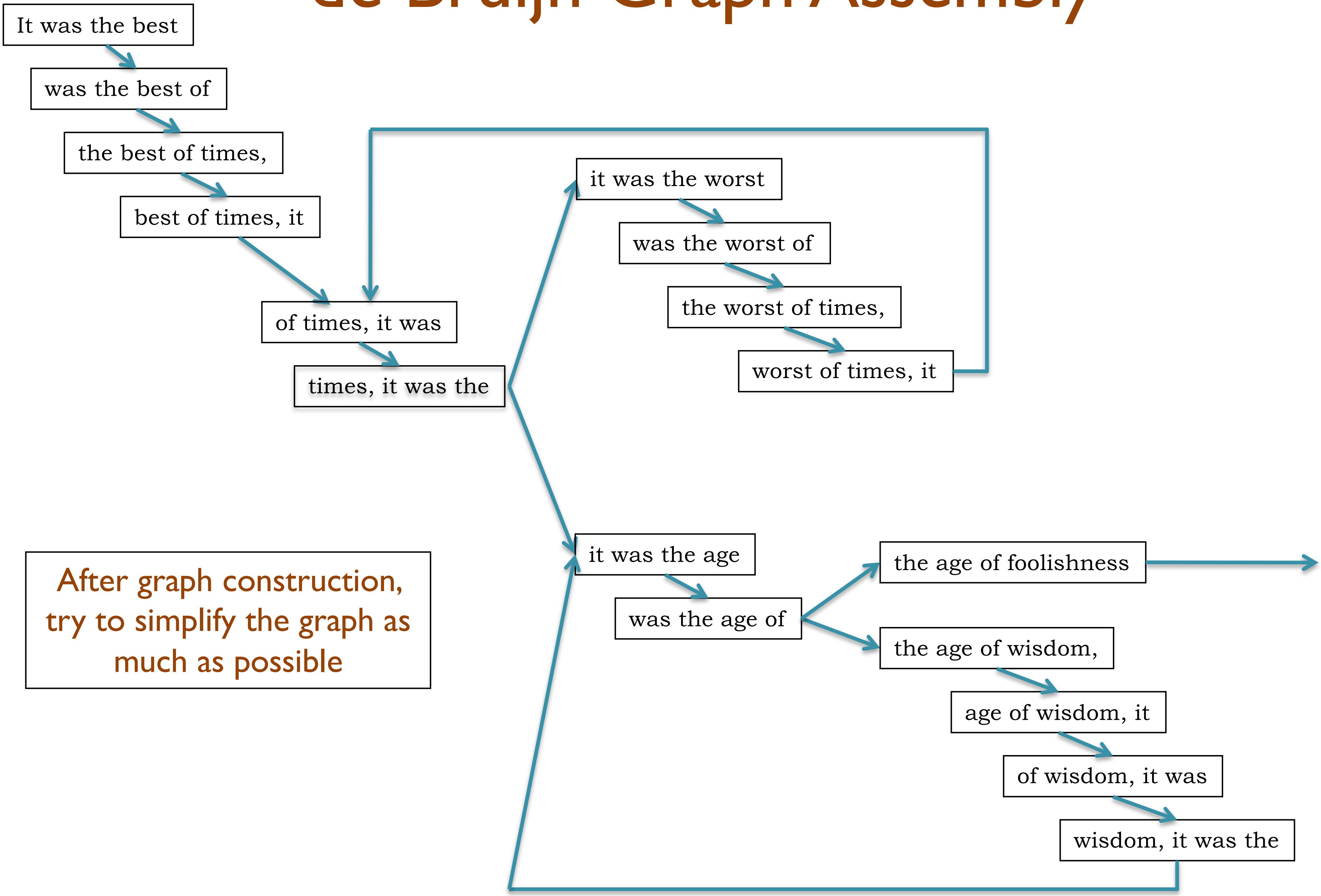
was the best of

the best of times

– Overlaps between fragments are implicitly computed

de Bruijn, 1946
Idury et al., 1995
Pevzner et al., 2001

de Bruijn Graph Assembly



It was the best

was the best of

the best of times,

best of times, it

of times, it was

times, it was the

it was the worst

was the worst of

the worst of times,

worst of times, it

it was the age

was the age of

the age of foolishness

the age of wisdom,

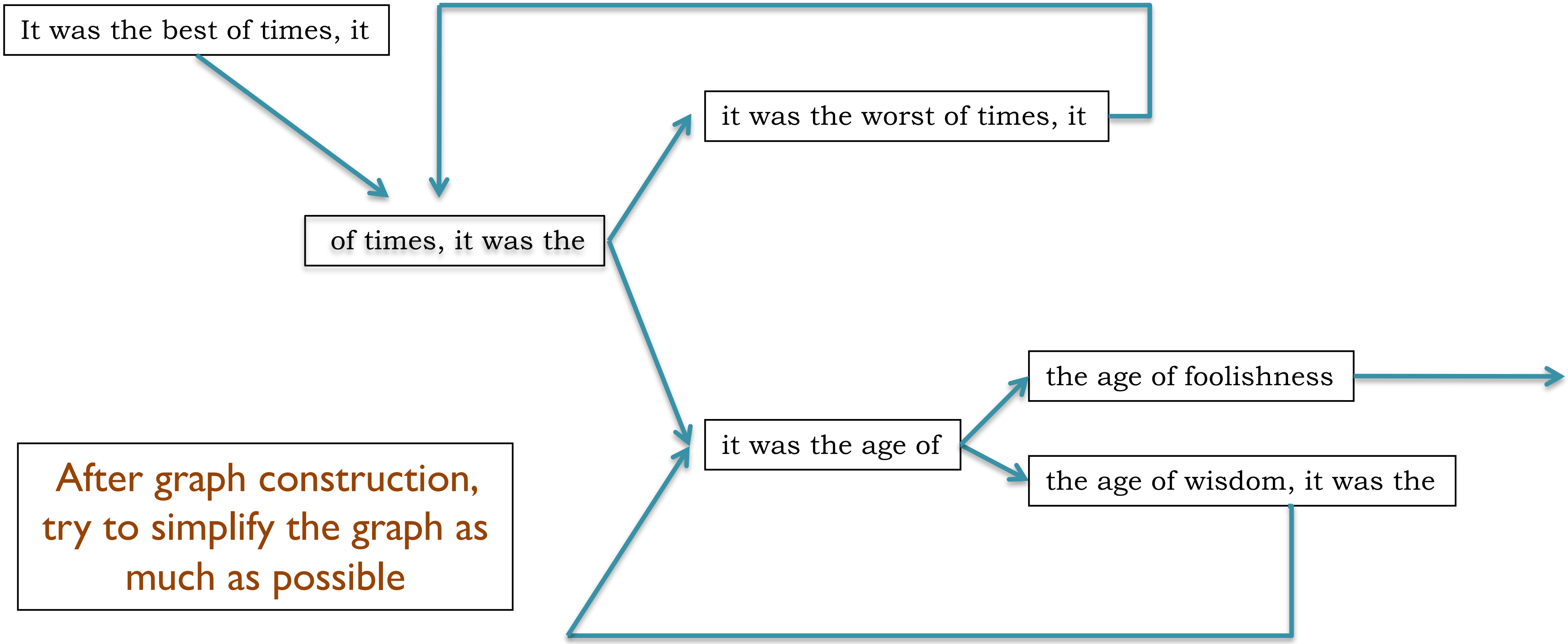
age of wisdom, it

of wisdom, it was

wisdom, it was the

After graph construction, try to simplify the graph as much as possible

de Bruijn Graph Assembly



It was the best of times, it

of times, it was the

it was the worst of times, it

it was the age of

the age of foolishness

the age of wisdom, it was the

After graph construction, try to simplify the graph as much as possible

The full tale

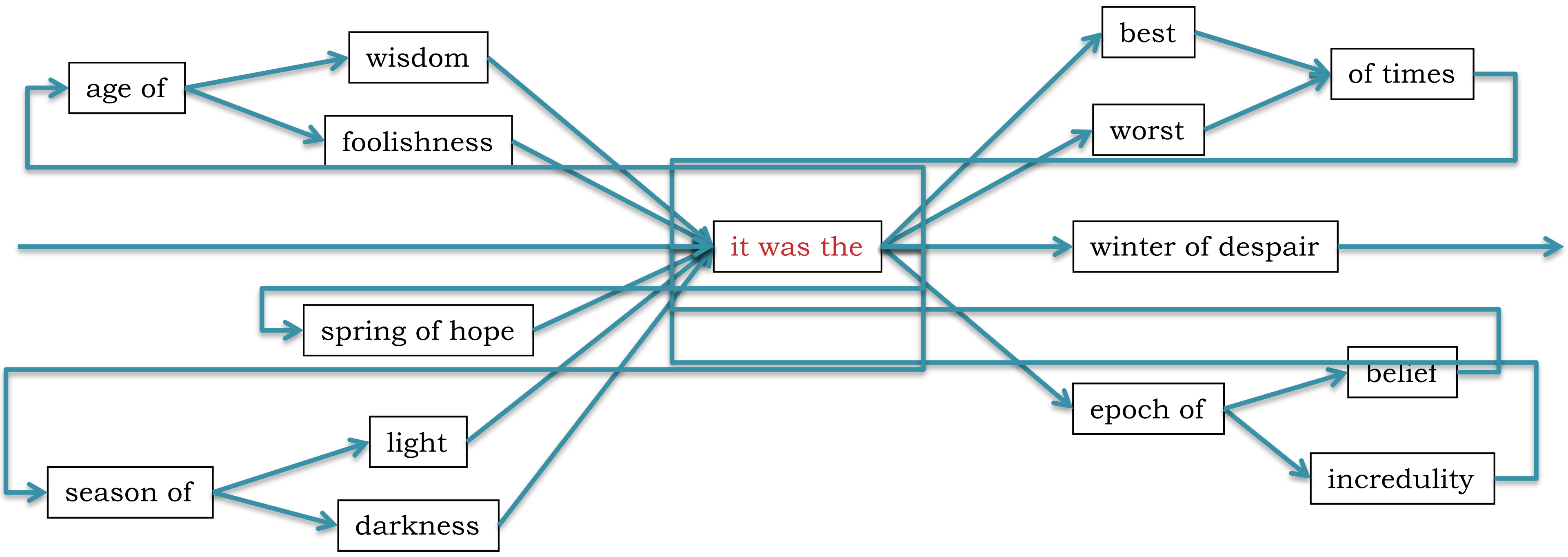
… it was the best of times it was the worst of times …

… it was the age of wisdom it was the age of foolishness …

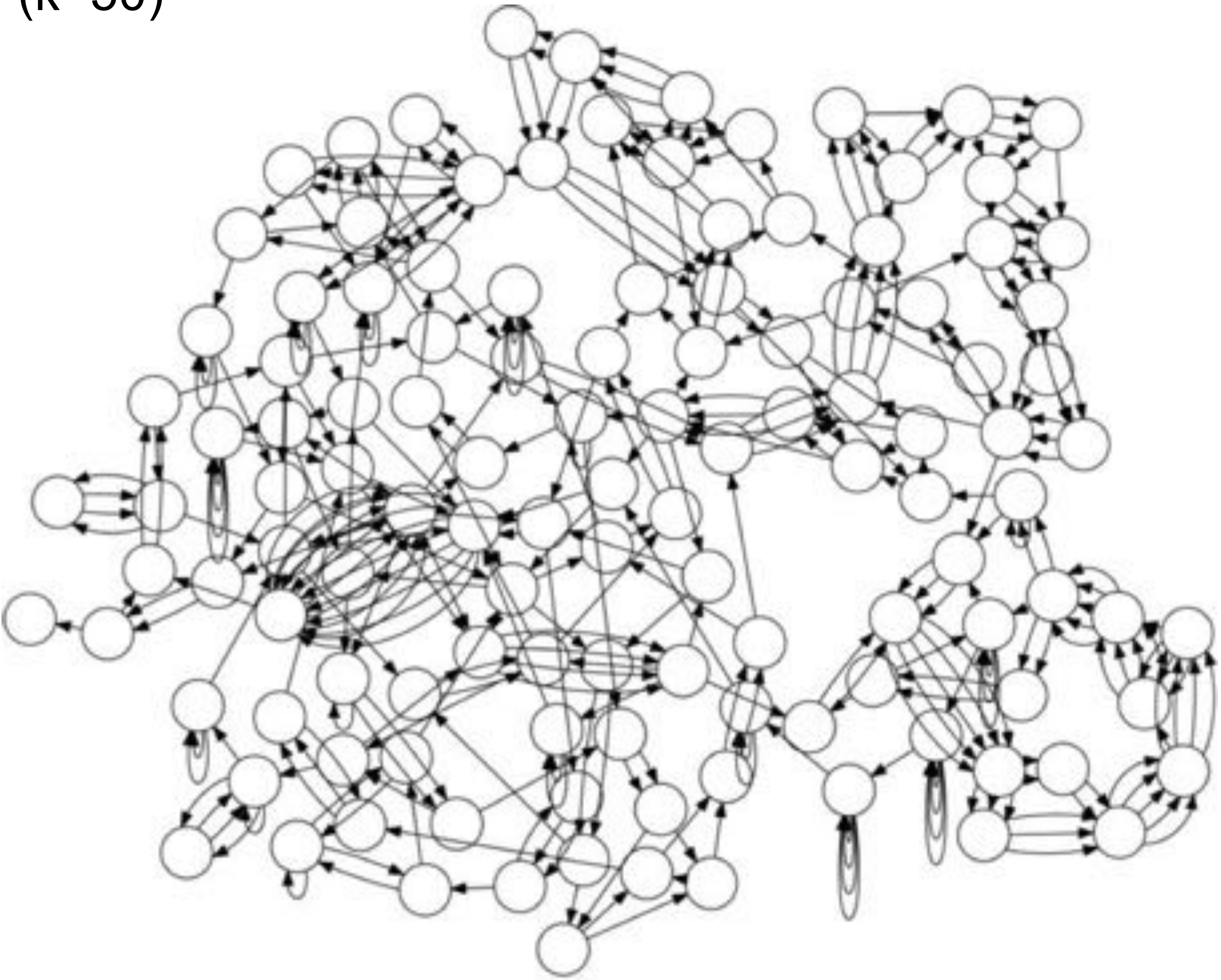
… it was the epoch of belief it was the epoch of incredulity …

… it was the season of light it was the season of darkness …

… it was the spring of hope it was the winder of despair …

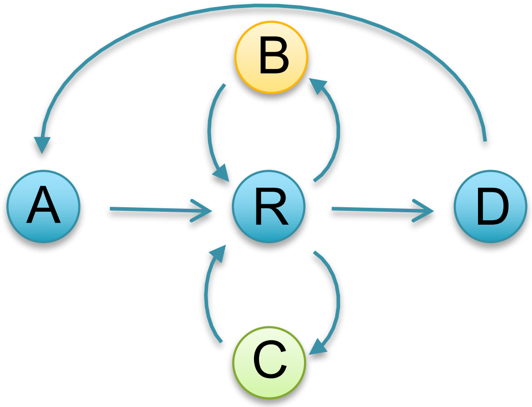


E. coli (k=50)



Reducing assembly complexity of microbial genomes with single-molecule sequencing
Koren et al (2013) Genome Biology. **14**:R101 https://doi.org/10.1186/gb-2013-14-9-r101

Counting Eulerian Cycles



ARBRCRD
or
ARCRBRD

Generally an exponential number of compatible sequences

– Value computed by application of the BEST theorem (Hutchinson, 1975)

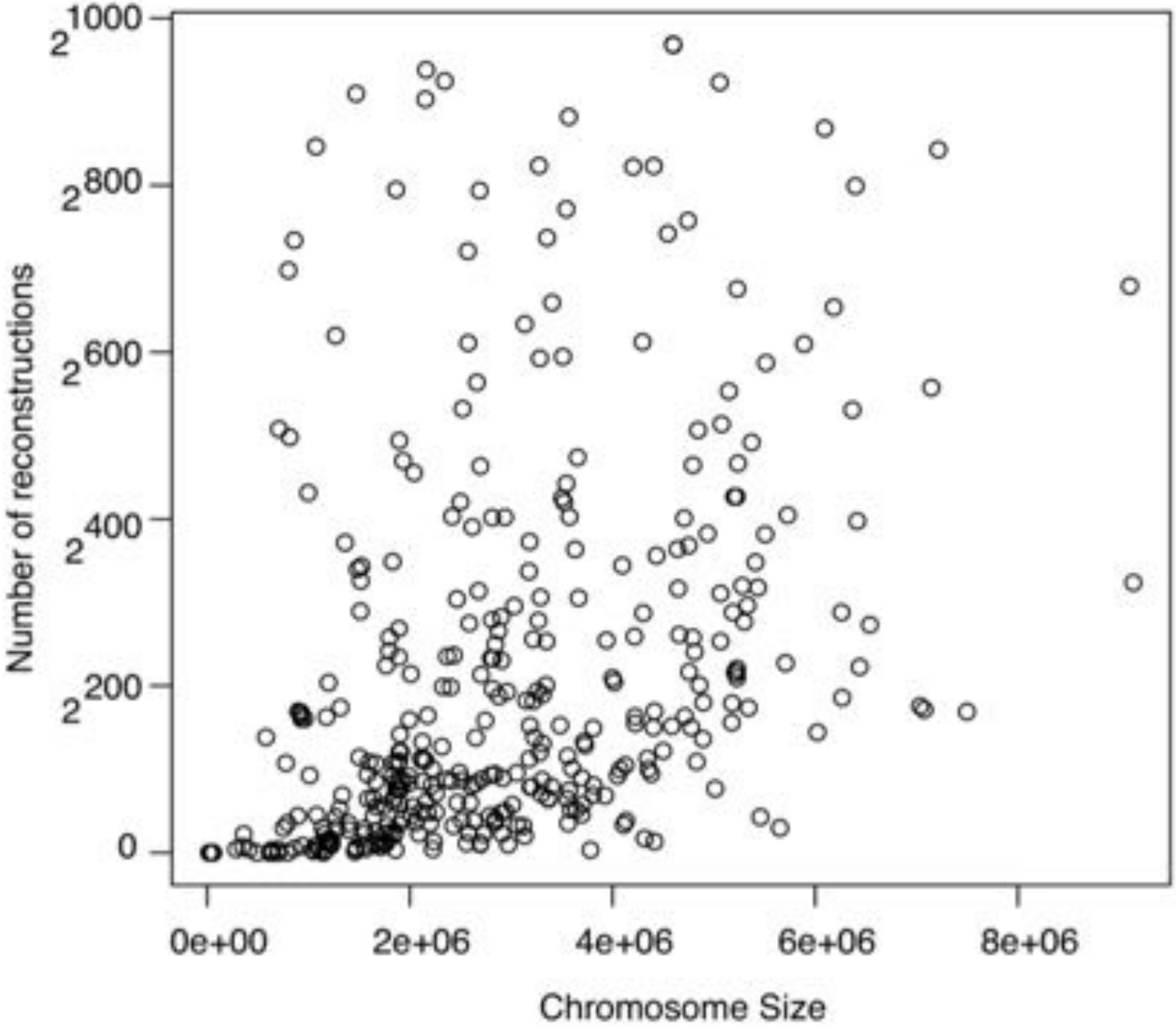
$$\mathcal{W}(G,t) = (\det L)\left\{\prod_{u\in V}(r_u - 1)!\right\}\left\{\prod_{(u,v)\in E} a_{uv}!\right\}^{-1}$$

L = $n \times n$ matrix with r_u-a_{uu} along the diagonal and -a_{uv} in entry uv

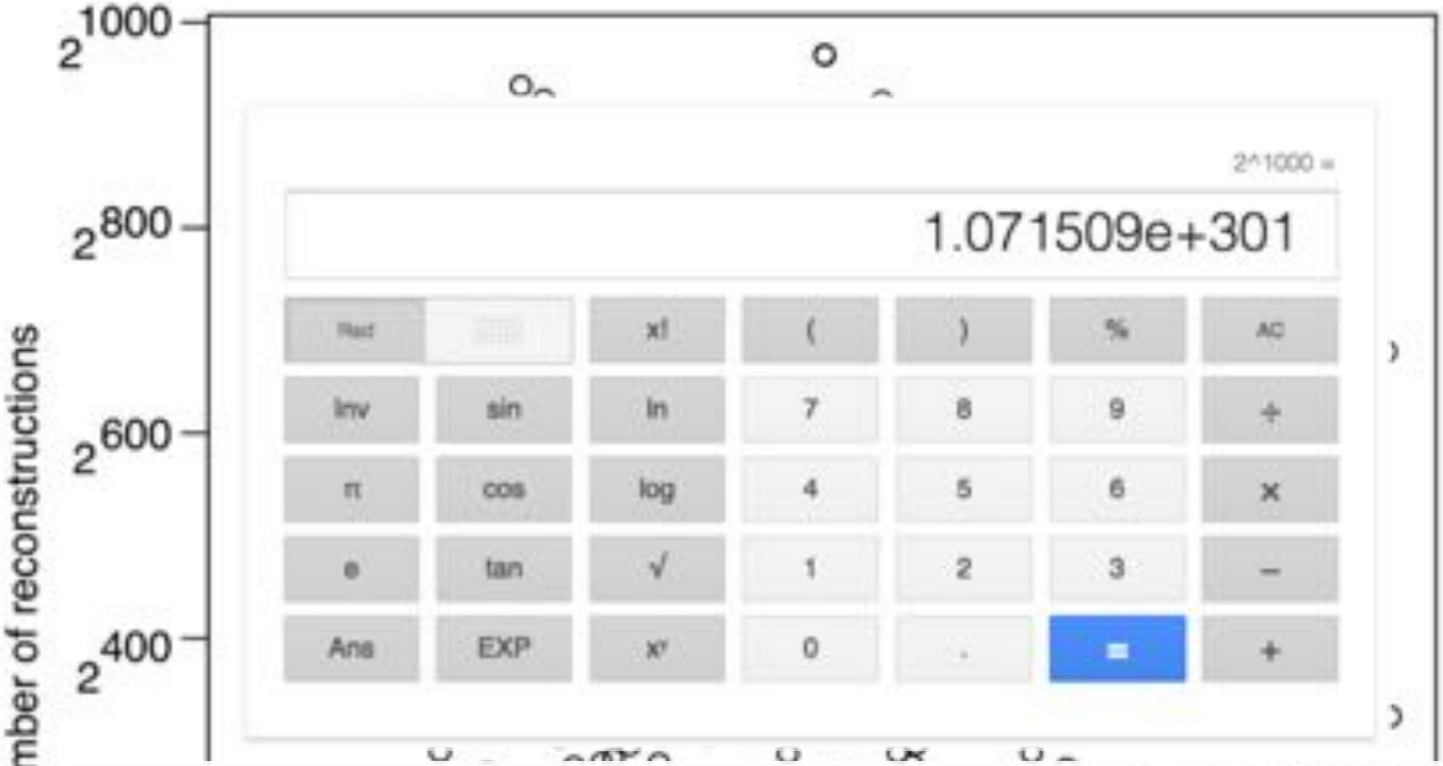
$r_u = d^+(u)+1$ if $u=t$, or $d^+(u)$ otherwise

a_{uv} = multiplicity of edge from u to v

Assembly Complexity of Prokaryotic Genomes using Short Reads.
Kingsford C, Schatz MC, Pop M (2010) *BMC Bioinformatics*.

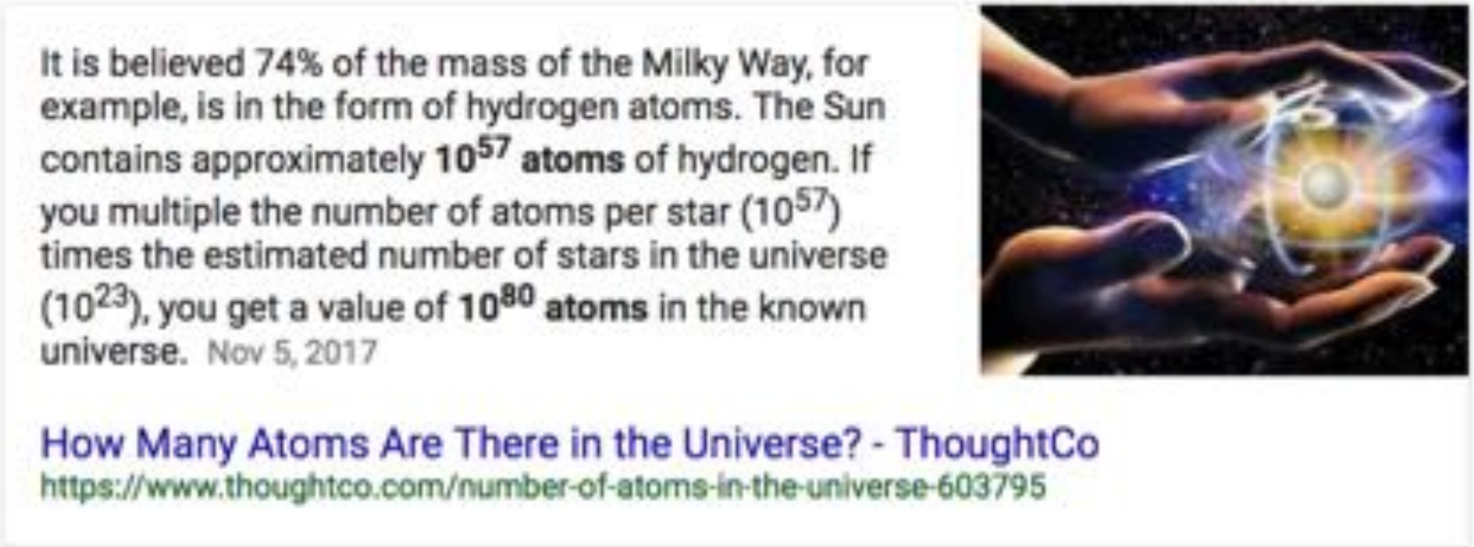


Assembly Complexity of Prokaryotic Genomes using Short Reads.
Kingsford C, Schatz MC, Pop M (2010) *BMC Bioinformatics*.



2^1000 =

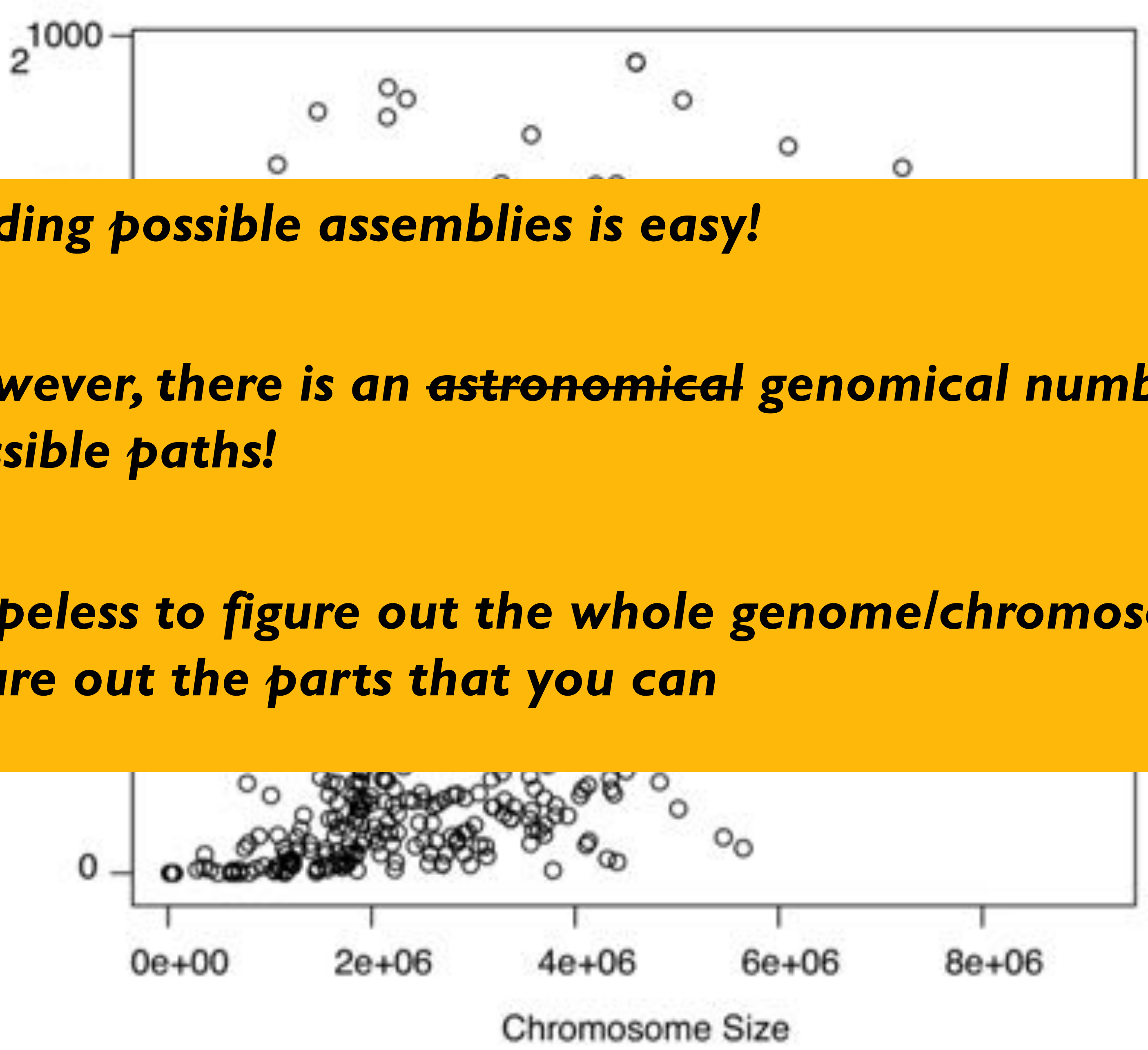
1.071509e+301



It is believed 74% of the mass of the Milky Way, for example, is in the form of hydrogen atoms. The Sun contains approximately **10^{57} atoms** of hydrogen. If you multiple the number of atoms per star (10^{57}) times the estimated number of stars in the universe (10^{23}), you get a value of **10^{80} atoms** in the known universe. Nov 5, 2017

How Many Atoms Are There in the Universe? - ThoughtCo
https://www.thoughtco.com/number-of-atoms-in-the-universe-603795

Assembly Complexity of Prokaryotic Genomes using Short Reads.
Kingsford C, Schatz MC, Pop M (2010) *BMC Bioinformatics*.



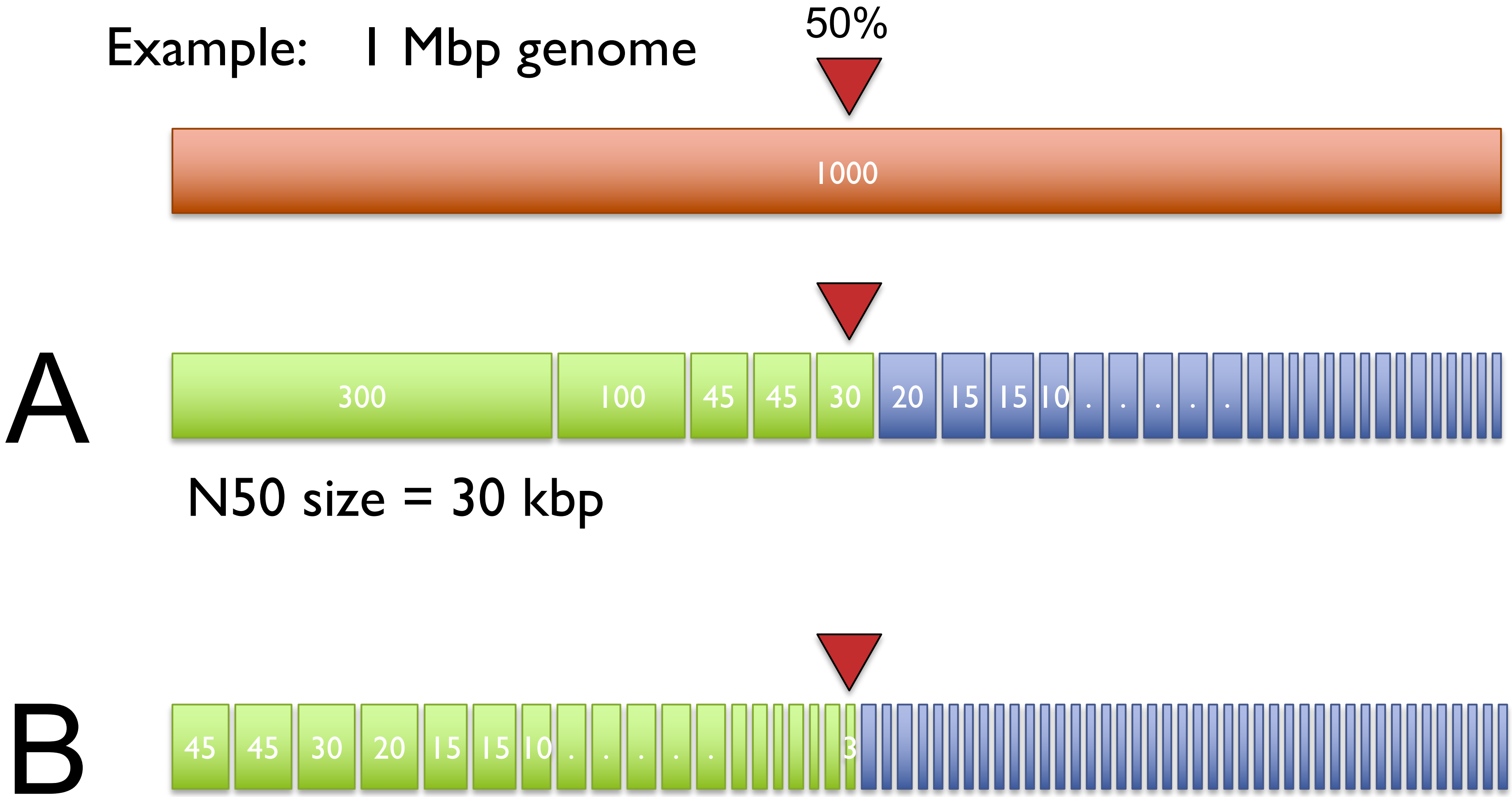
- ***Finding possible assemblies is easy!***
- ***However, there is an ~~astronomical~~ genomical number of possible paths!***
- ***Hopeless to figure out the whole genome/chromosome, figure out the parts that you can***

Assembly Complexity of Prokaryotic Genomes using Short Reads.
Kingsford C, Schatz MC, Pop M (2010) *BMC Bioinformatics*.

Contig N50

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

Example: 1 Mbp genome



N50 size = 30 kbp

N50 size = 3 kbp

Contig N50

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

50%

Better N50s improves the analysis in every dimension

- Better resolution of genes and flanking regulatory regions
- Better resolution of transposons and other complex sequences
- Better resolution of chromosome organization
- Better sequence for all downstream analysis

Just be careful of N50 inflation!

- *A very very very bad assembler in 1 line of bash:*
- *cat *.reads.fa > genome.fa*

N50 size = 3 kbp

Pop Quiz 1

Assemble these reads using a de Bruijn graph approach (k=3):

```
ATTA
GATT
TACA
TTAC
```

Pop Quiz I

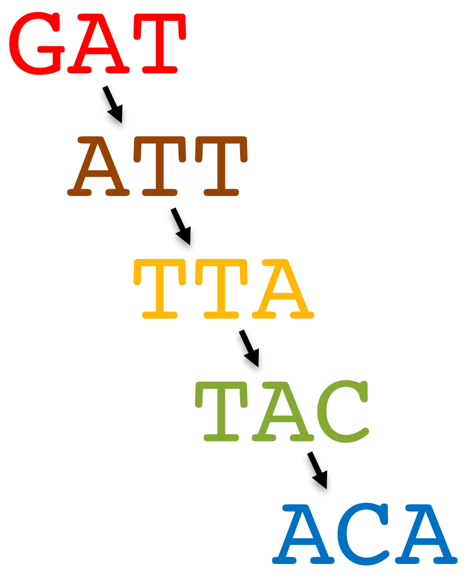
Assemble these reads using a de Bruijn graph approach (k=3):

```
ATTA: ATT -> TTA
GATT: GAT -> ATT
TACA: TAC -> ACA
TTAC: TTA -> TAC
```

Pop Quiz I

Assemble these reads using a de Bruijn graph approach (k=3):

```
ATTA: ATT -> TTA
GATT: GAT -> ATT
TACA: TAC -> ACA
TTAC: TTA -> TAC
```



GAT

ATT

TTA

TAC

ACA

GATTACA

Pop Quiz 2

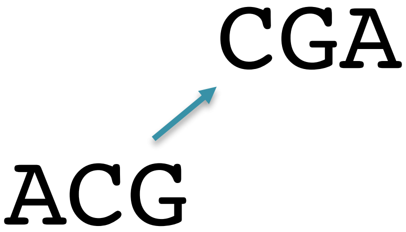
Assemble these reads using a de Bruijn graph approach (k=3):

```
ACGA
ACGT
ATAC
CGAC
CGTA
GACG
GTAT
TACG
```

Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~
ACGT
ATAC
CGAC
CGTA
GACG
GTAT
TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

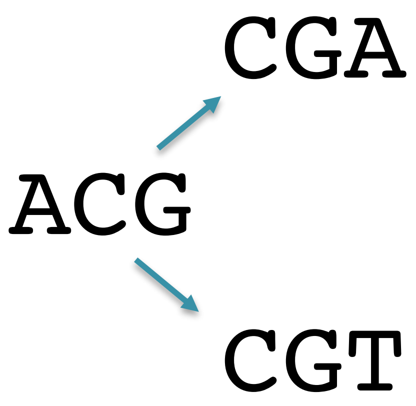
CGAC

CGTA

GACG

GTAT

TACG



CGA

ACG

CGT

Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

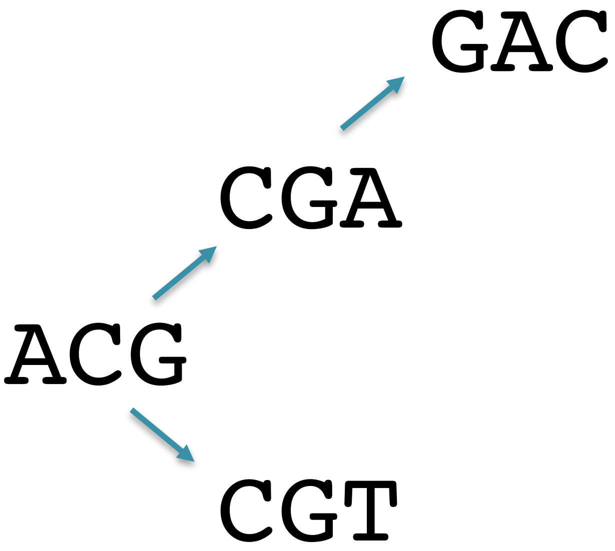
~~CGAC~~

CGTA

GACG

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

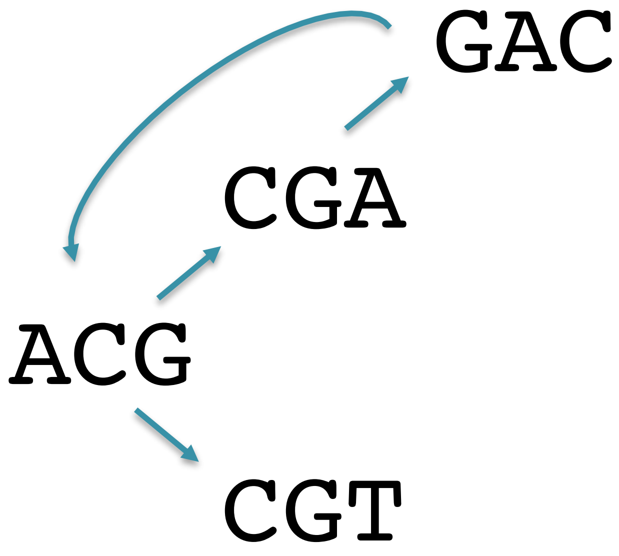
~~CGAC~~

CGTA

~~GACG~~

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

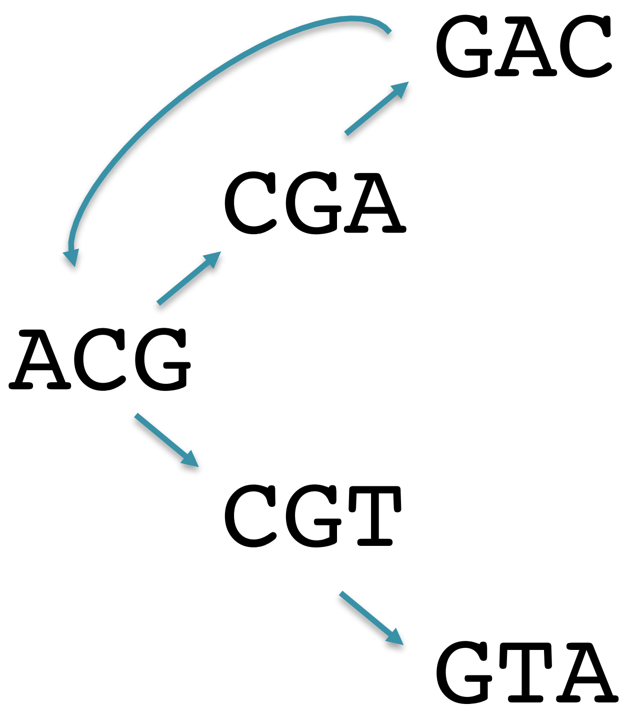
~~CGAC~~

~~CGTA~~

~~GACG~~

GTAT

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

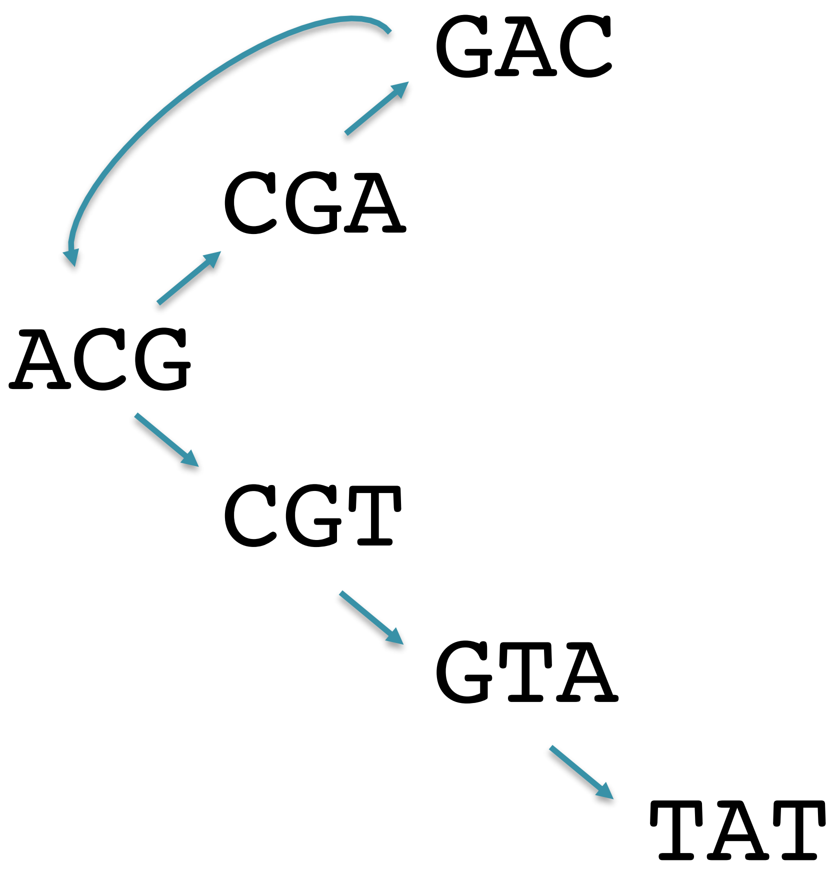
~~CGAC~~

~~CGTA~~

~~GACG~~

~~GTAT~~

TACG



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

ATAC

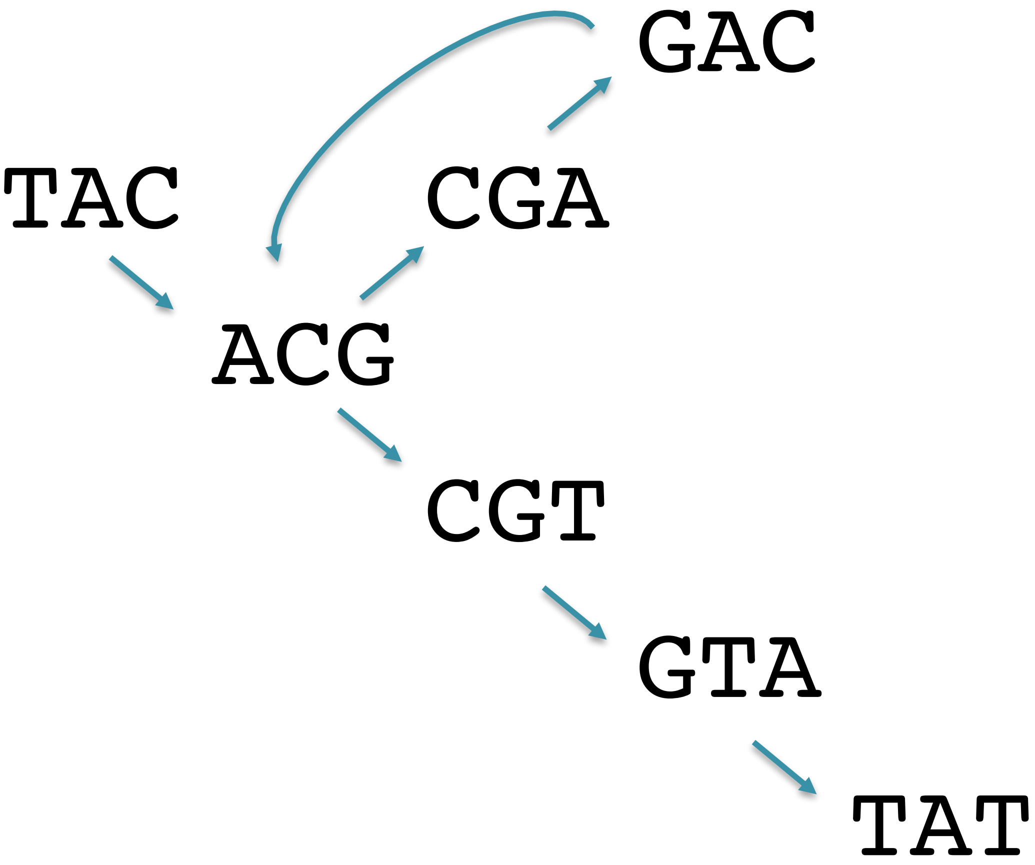
~~CGAC~~

~~CGTA~~

~~GACG~~

~~GTAT~~

~~TACG~~



GAC

TAC

CGA

ACG

CGT

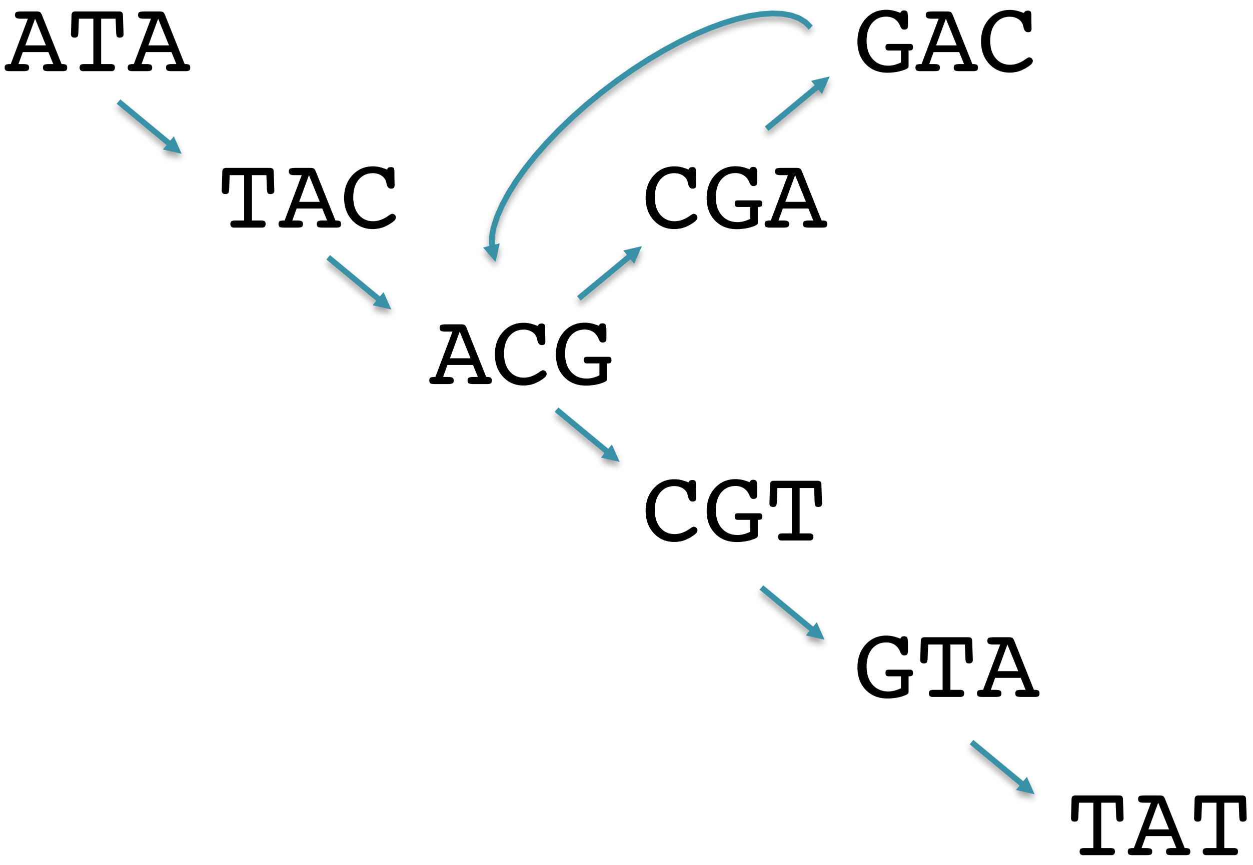
GTA

TAT

Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~
~~ACGT~~
~~ATAC~~
~~CGAC~~
~~CGTA~~
~~GACG~~
~~GTAT~~
~~TACG~~



Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

~~ATAC~~

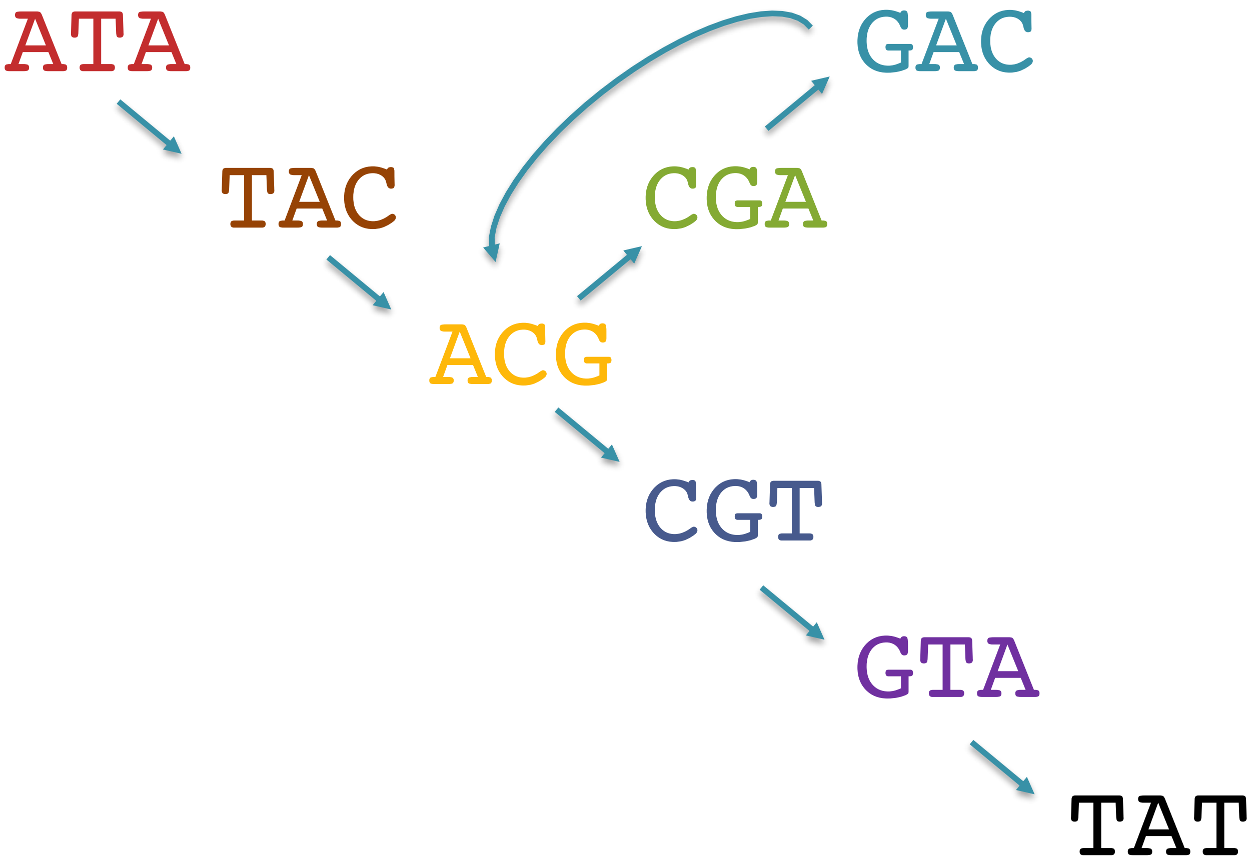
~~CGAC~~

~~CGTA~~

~~GACG~~

~~GTAT~~

~~TACG~~



ATACGACGTAT

Pop Quiz 2

Assemble these reads using a de Bruijn graph approach (k=3):

~~ACGA~~

~~ACGT~~

~~ATAC~~

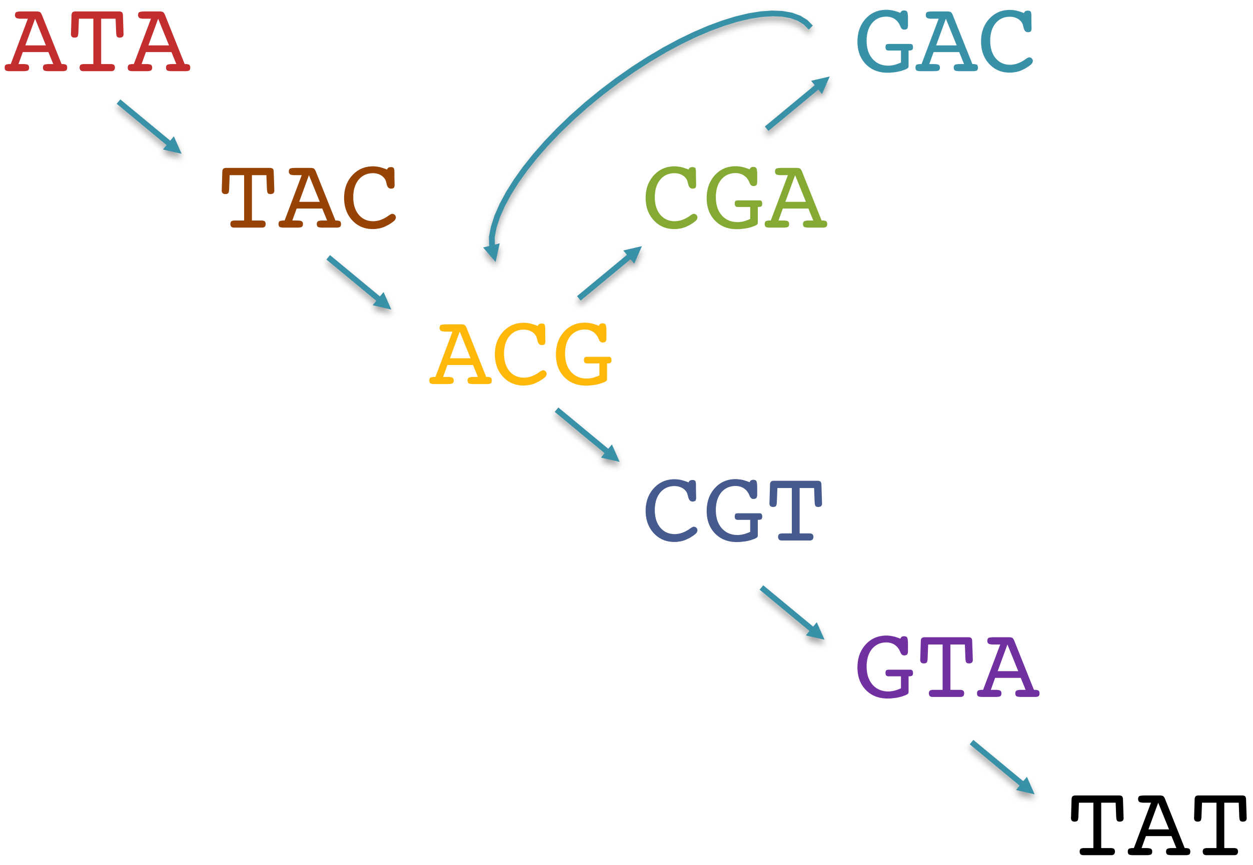
~~CGAC~~

~~CGTA~~

~~GACG~~

~~GTAT~~

~~TACG~~



Whats another possible genome?

ATACGACGTAT

Outline

1. ***Assembly theory***
   - Assembly by analogy

2. ***Practical Issues***
   - Coverage, read length, errors, and repeats

3. ***Next-next-gen Assembly***
   - Canu: recommended for PacBio/ONT project

4. Whole Genome Alignment
   - MUMmer recommended

Assembly Applications

- Novel genomes

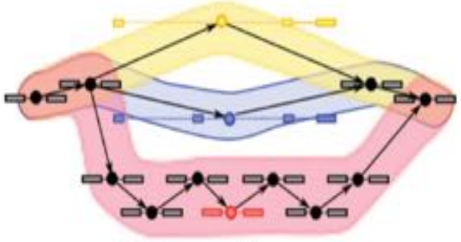
GENOME 10K

i5K

- Metagenomes

HMP

- Sequencing assays
  - Structural variations
  - Transcript assembly
  - …



Why are genomes hard to assemble?

1. ***Biological***:
   - (Very) High ploidy, heterozygosity, repeat content

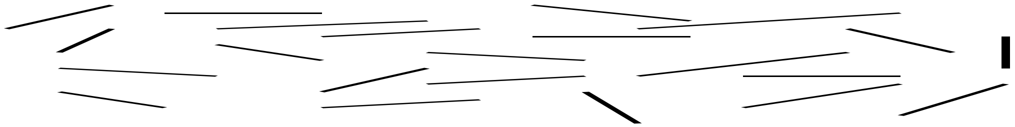
2. ***Sequencing***:
   - (Very) large genomes, imperfect sequencing

3. ***Computational***:
   - (Very) Large genomes, complex structure

4. ***Accuracy***:
   - (Very) Hard to assess correctness

Assembling a Genome

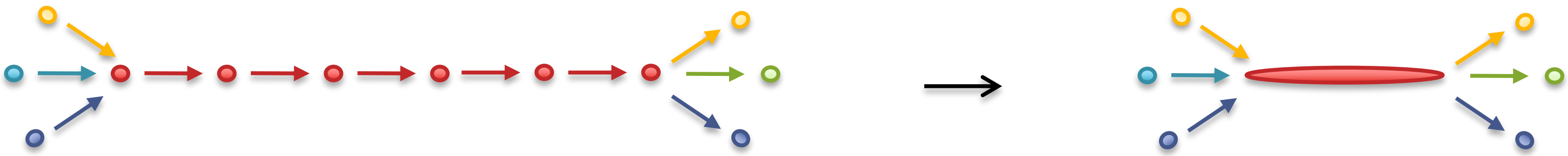
1. Shear & Sequence DNA



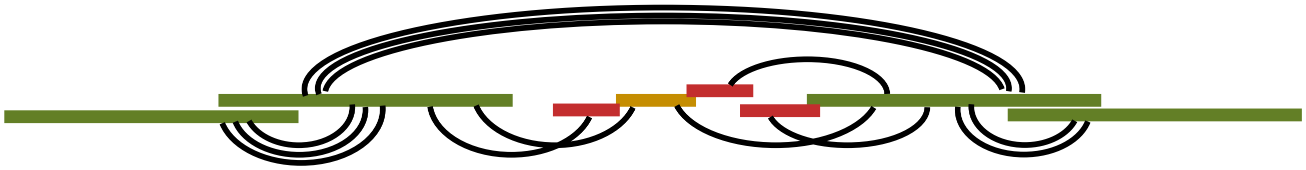
2. Construct assembly graph from reads (de Bruijn / overlap graph)

```
...AGCCTAGGGATGCGCGACACGT
          GGATGCGCGACACGTCGCATATCCGGTTTGGTCAACCTCGGACGGAC
                                         CAACCTCGGACGGACCTCAGCGAA...
```

3. Simplify assembly graph

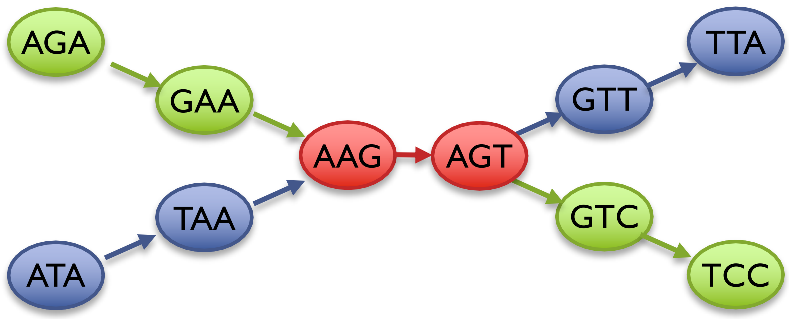


4. Detangle graph with long reads, mates, and other links



Two Paradigms for Assembly

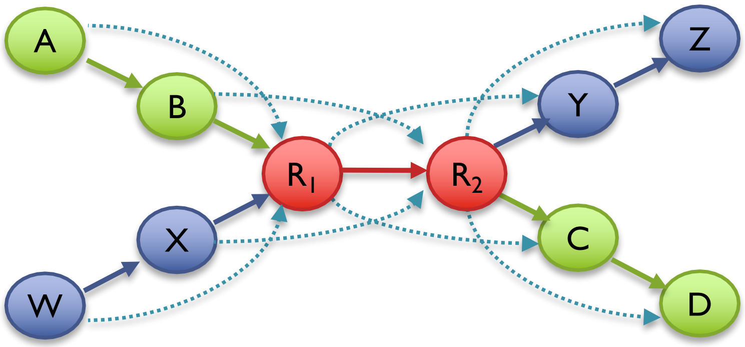
de Bruijn Graph



Short read assemblers

- Repeats depends on word length
- Read coherency, placements lost
- Robust to high coverage

Overlap Graph



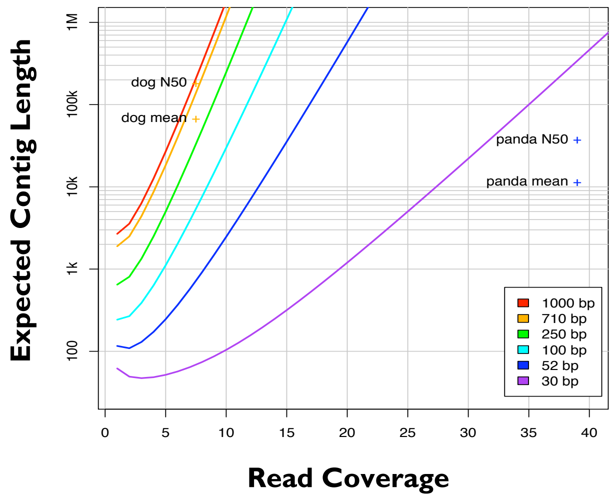
Long read assemblers

- Repeats depends on read length
- Read coherency, placements kept
- Tangled by high coverage

Assembly of Large Genomes using Second Generation Sequencing
Schatz MC, Delcher AL, Salzberg SL (2010) *Genome Research*. 20:1165-1173.

Ingredients for a good assembly

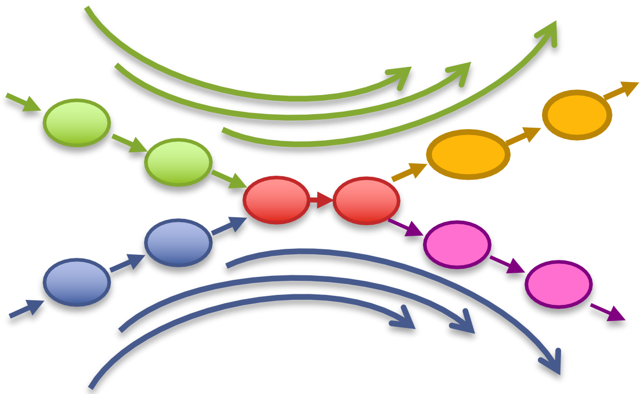
Coverage



High coverage is required

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

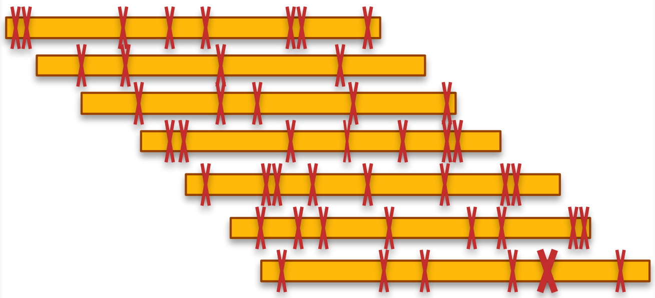
Read Length



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

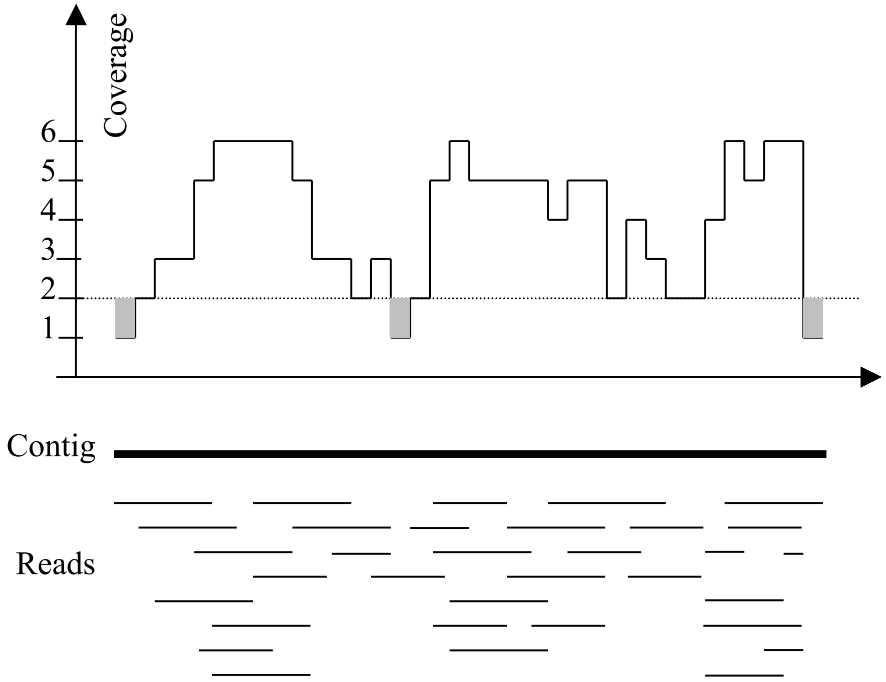


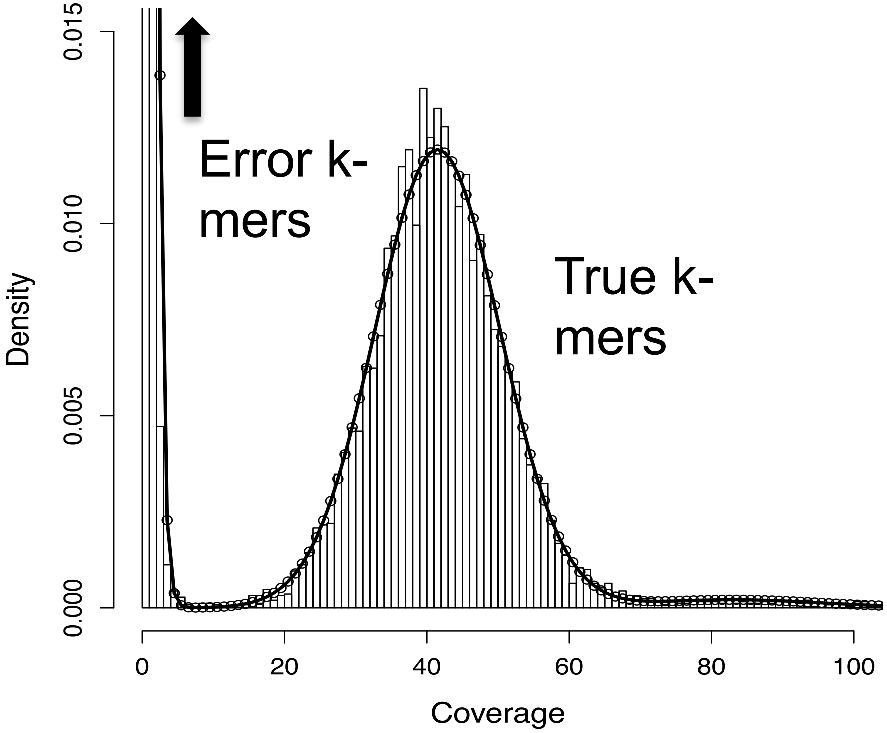
Errors obscure overlaps

- 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, WR (2012) *Genome Biology*. 12:243

Kmer-based Coverage Analysis

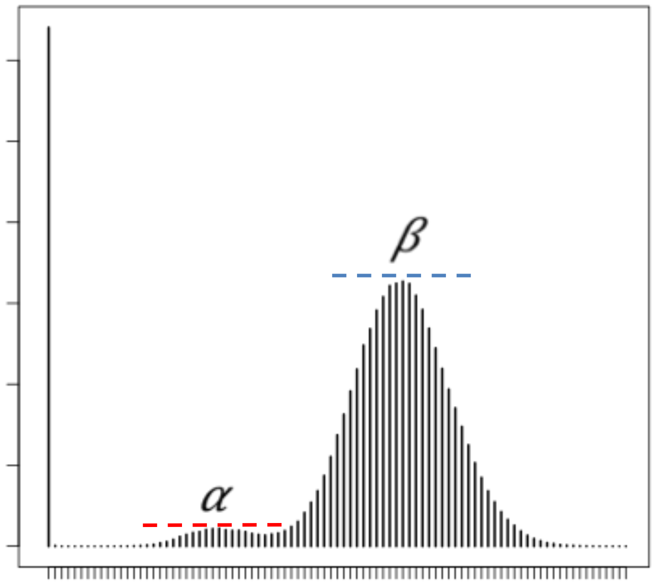




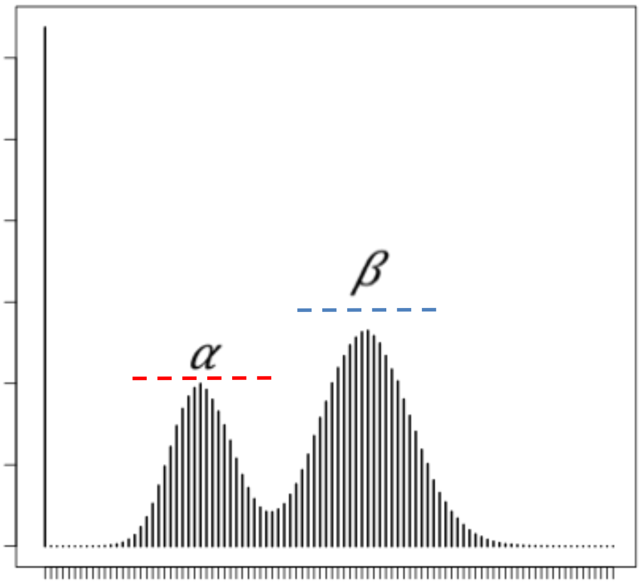
Even though the reads are not assembled or aligned (or reference available),
Kmer counting is an effective technique to estimate coverage & other genome properties

Quake: quality-aware detection and correction of sequencing reads.
Kelley, DR, Schatz, MC, Salzberg SL (2010) *Genome Biology.* 11:R116

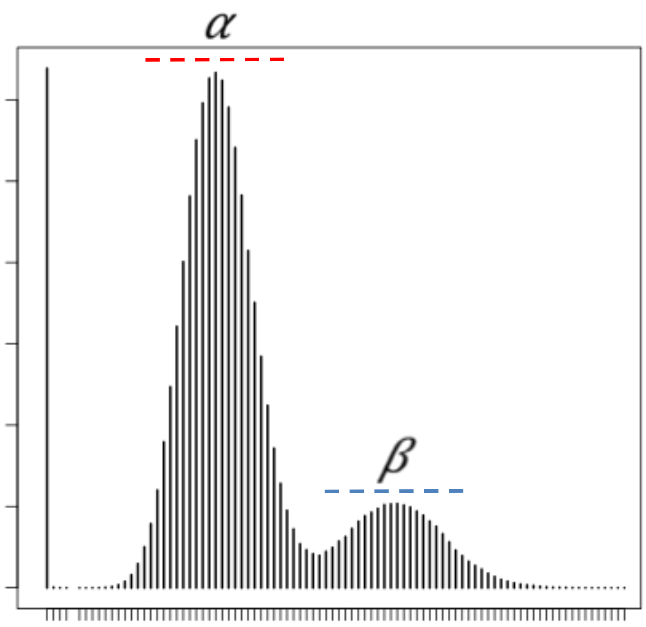
Heterozygous Kmer Profiles



0.1% heterozygosity



1% heterozygosity



5% heterozygosity

- ***Heterozygosity creates a characteristic "double-peak" in the Kmer profile***
  - Second peak at twice k-mer coverage as the first: heterozygous kmers average 50x coverage, homozygous kmers average 100x coverage

- ***Relative heights of the peaks is directly proportional to the heterozygosity rate***
  - The peaks are balanced at around 1.25% because each heterozygous SNP creates 2*k heterozygous kmers (typically k = 21)

GenomeScope Model

$$f(x) = G\Big\{\alpha NB(x, \lambda, \lambda/\rho) + \beta NB(x, 2\lambda, 2\lambda/\rho) + \gamma NB(x, 3\lambda, 3\lambda/\rho) + \delta NB(x, 4\lambda, 4\lambda/\rho)\Big\}$$

Analyze k-mer profiles using a mixture model of 4 negative binominal components

- Components centered at 1,2,3,4 * λ

- Four components capture heterozygous and homozygous unique (α,β) and 2 copy repeats (γ,δ). Higher order repeats do not contribute a significant number of kmers

- Negative binomial instead of Poisson to account for over dispersion observed in real data (especially PCR duplicates); variance modeled by ρ

$$\alpha = 2(1-d)(1-(1-r)^k) + 2d(1-(1-r)^k)^2 + 2d((1-r)^k)(1-(1-r)^k)$$

$$\beta = (1-d)((1-r)^k) + d(1-(1-r)^k)^2$$

$$\gamma = 2d((1-r)^k)(1-(1-r)^k)$$

$$\delta = d(1-r)^{2k}$$

k is the *k-mer* length

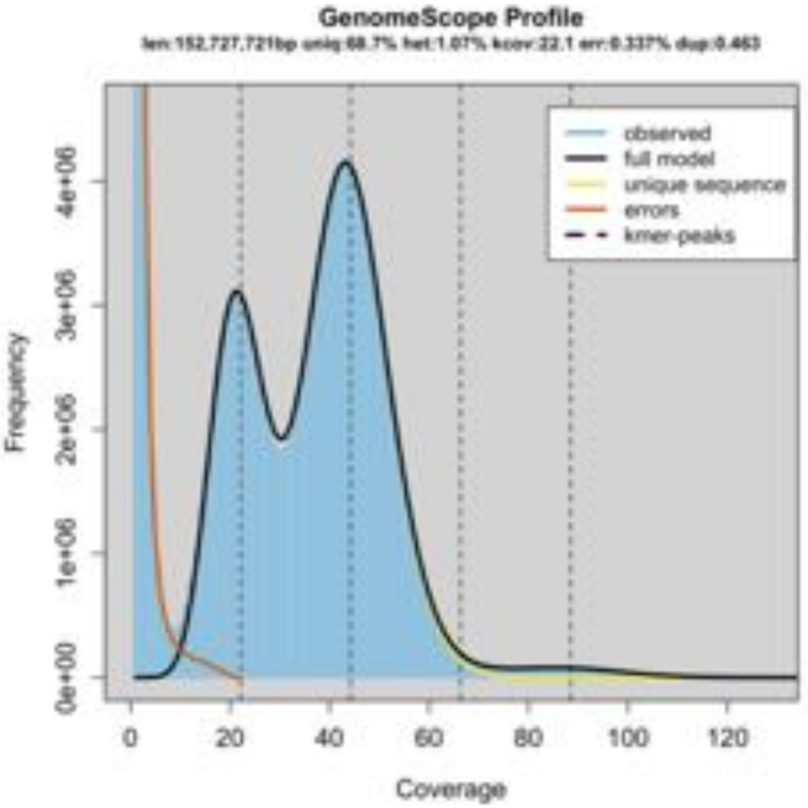
r is the rate of heterozygosity

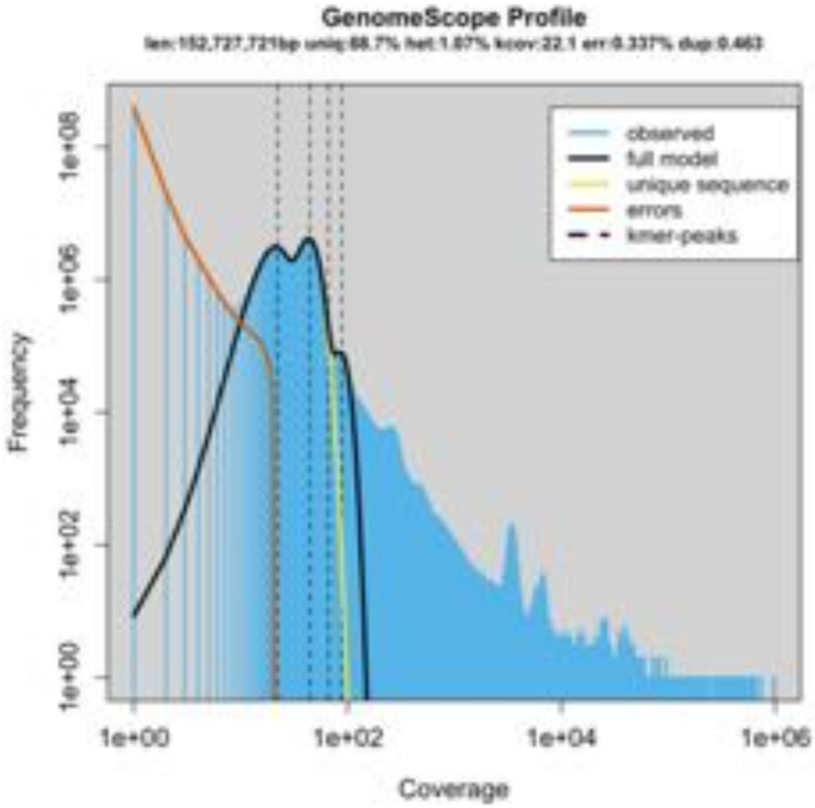
d represents the percentage of the genome that is two-copy repeat

Fit model with nls, infer rate of heterozygosity, genome size, unique/repetitive content, sequencing error rate, rate of PCR duplicates

GenomeScope: Fast genome analysis from short reads

http://genomescope.org





- Theoretical model agrees well with published results:
  - Rate of heterozygosity is higher than reported by other approaches but likely correct.
  - Genome size of plants inflated by organelle sequences (exclude very high freq. kmers)

Vurture, GW*, Sedlazeck FJ*, et al. (2017) *Bioinformatics*

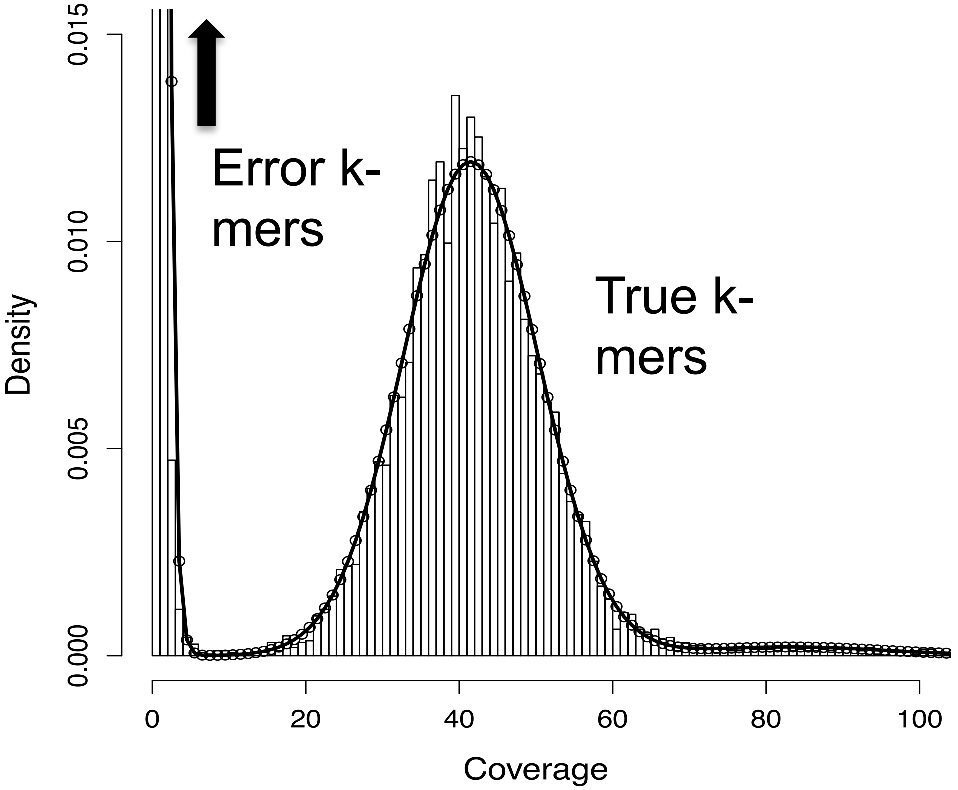
Error Correction with Quake

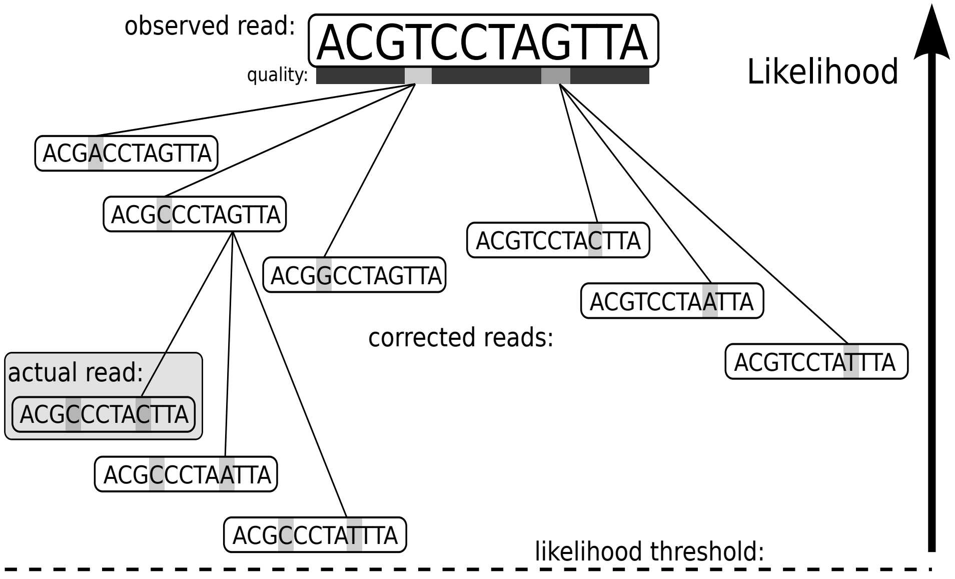
1. Count all "Q-mers" in reads

- Fit coverage distribution to mixture model of errors and regular coverage
- Automatically determines threshold for trusted k-mers

2. Correction Algorithm

- Considers editing erroneous kmers into trusted kmers in decreasing likelihood
- Includes quality values, nucleotide/nucleotide substitution rate



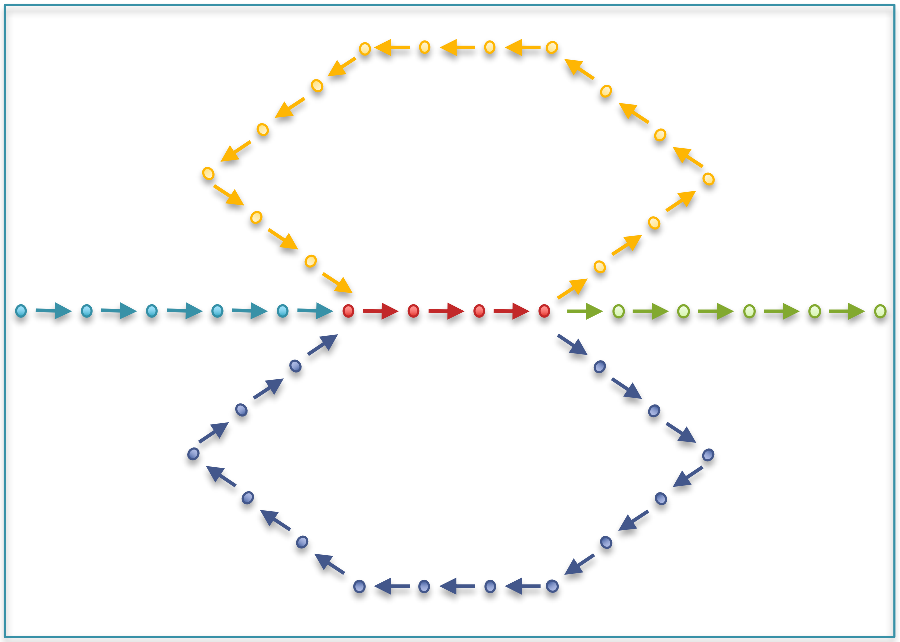


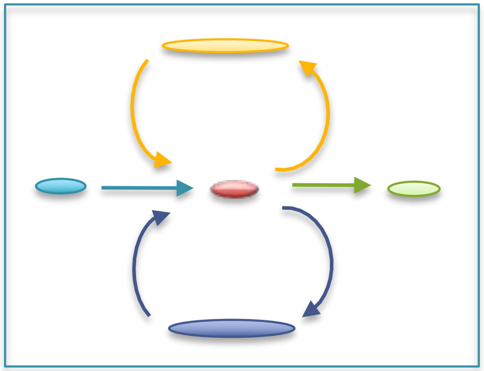
Quake: quality-aware detection and correction of sequencing reads.
Kelley, DR, Schatz, MC, Salzberg SL (2010) *Genome Biology.* 11:R116

Errors

Unitigging / Unipathing

- After simplification and correction, compress graph down to its non-branching initial contigs
  - Aka "unitigs", "unipaths"



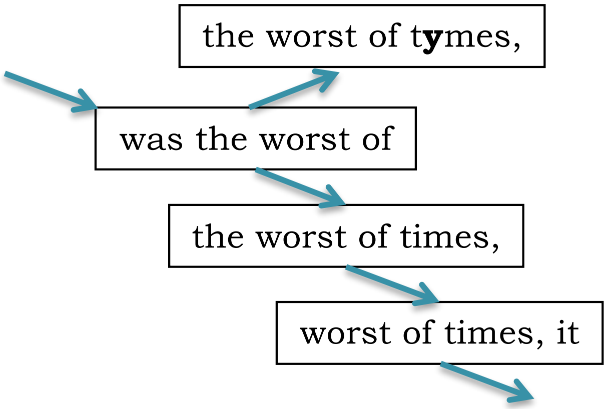
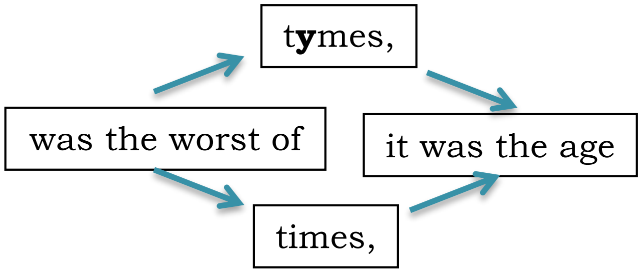


Why do contigs end?

(1) End of chromosome! ☺, (2) lack of coverage, (3) errors, (4) heterozygosity and (5) repeats

Errors in the graph

(Chaisson, 2009)

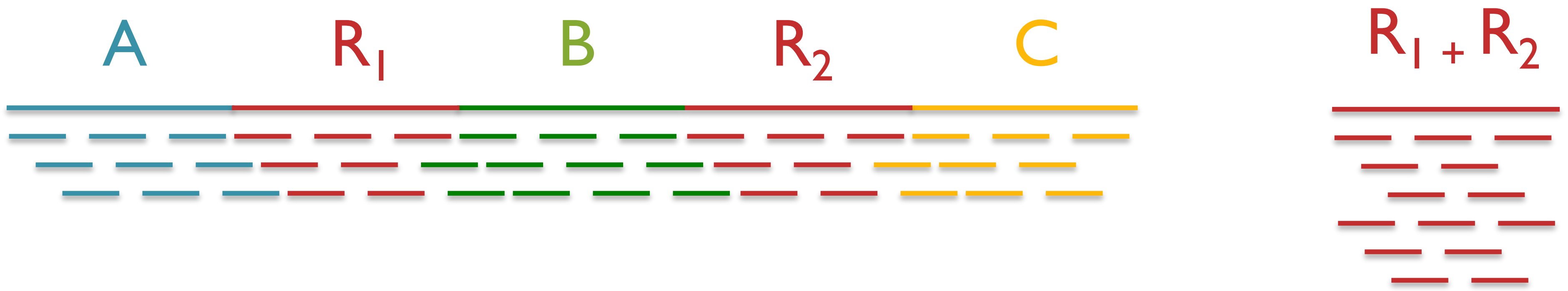
Clip Tips	Pop Bubbles
was the worst of times, was the worst of t**y**mes, the worst of times, it	was the worst of times, was the worst of t**y**mes, times, it was the age t**y**mes, it was the age
 the worst of t**y**mes, was the worst of the worst of times, worst of times, it	 t**y**mes, was the worst of it was the age times,

Repetitive regions

Repeat Type	Definition / Example	Prevalence
Low-complexity DNA / Microsatellites	$(b_1b_2\ldots b_k)^N$ where $1 \leq k \leq 6$ CACACACACACACACACACA	2%
SINEs (Short Interspersed Nuclear Elements)	*Alu* sequence (~280 bp) Mariner elements (~80 bp)	13%
LINEs (Long Interspersed Nuclear Elements)	~500 – 5,000 bp	21%
LTR (long terminal repeat) retrotransposons	Ty1-copia, Ty3-gypsy, Pao-BEL (~100 – 5,000 bp)	8%
Other DNA transposons		3%
Gene families & segmental duplications		4%

- Over 50% of mammalian genomes are repetitive
  - Large plant genomes tend to be even worse
  - Wheat: 16 Gbp; Pine: 24 Gbp

Repeats and Coverage Statistics



- If *n* reads are a uniform random sample of the genome of length *G*, we expect *k*=*n Δ* /*G* reads to start in a region of length *Δ* .
  - If we see many more reads than k (if the arrival rate is > A) , it is likely to be a collapsed repeat

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

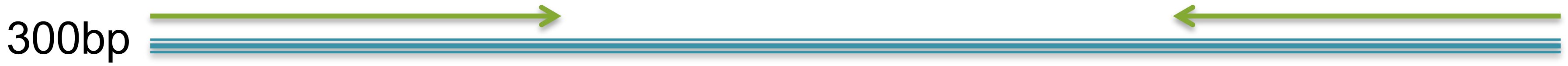
$$A(\Delta,k) = \ln\left(\frac{\Pr(1-copy)}{\Pr(2-copy)}\right) = \ln\left(\frac{\frac{(\Delta n / G)^k}{k!}e^{\frac{-\Delta n}{G}}}{\frac{(2\Delta n / G)^k}{k!}e^{\frac{-2\Delta n}{G}}}\right) = \frac{n\Delta}{G} - k\ln 2$$

The fragment assembly string graph
Myers, EW (2005) Bioinformatics. 21(suppl 2): ii79-85.

Paired-end and Mate-pairs

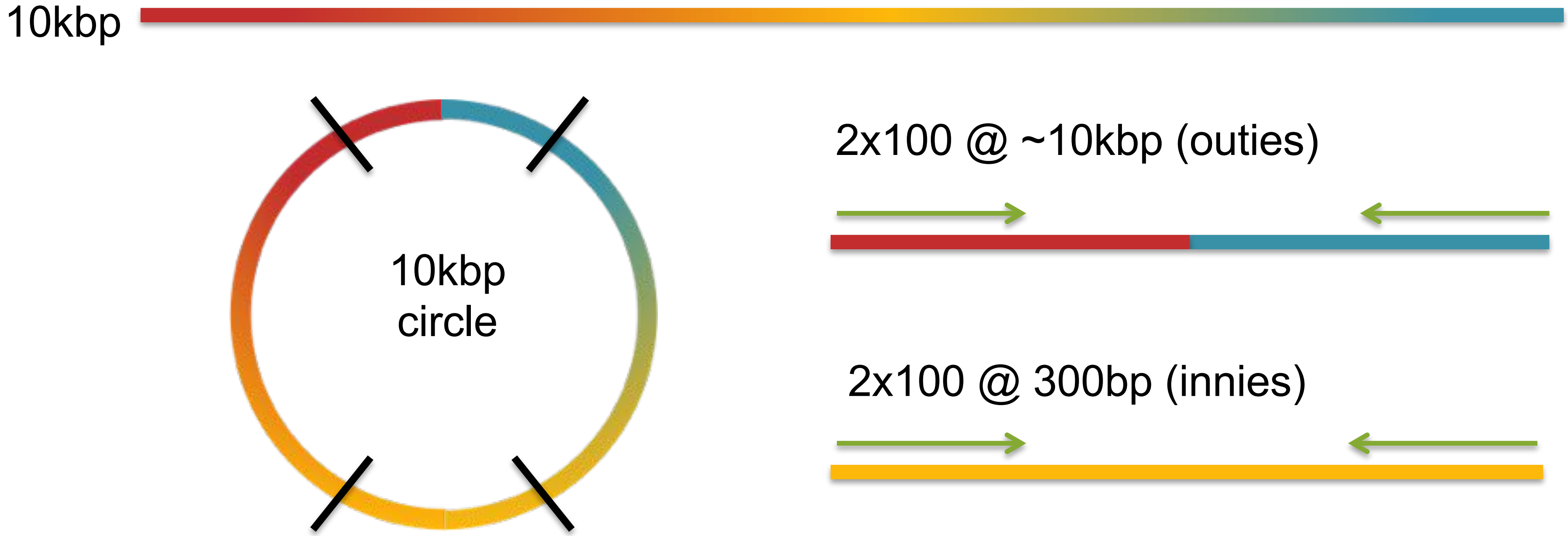
Paired-end sequencing

- Read one end of the molecule, flip, and read the other end
- Generate pair of reads separated by up to 500bp with inward orientation



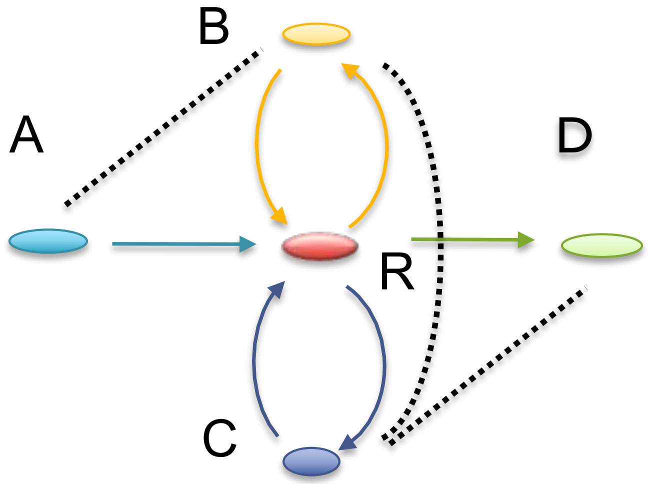
Mate-pair sequencing

- Circularize long molecules (1-10kbp), shear into fragments, & sequence
- Mate failures create short paired-end reads

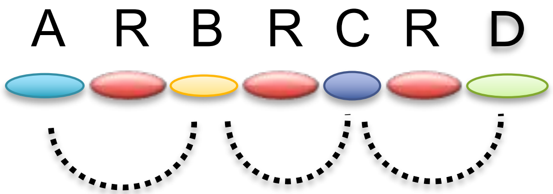


Scaffolding

- Initial contigs (*aka* unipaths, unitigs) terminate at
  - *Coverage gaps*: especially extreme GC
  - *Conflicts*: errors, repeat boundaries



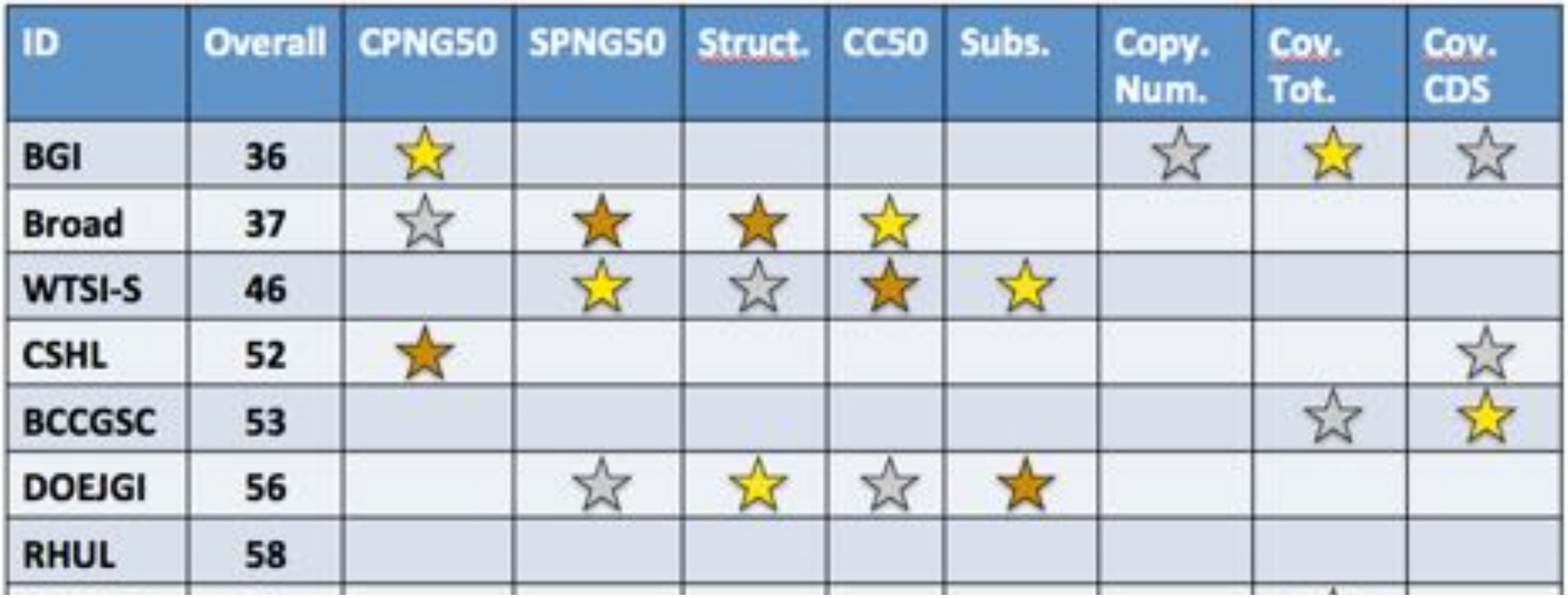
- Use mate-pairs to resolve correct order through assembly graph
  - Place sequence to satisfy the mate constraints
  - Mates through repeat nodes are tangled



- Final scaffold may have internal gaps called sequencing gaps
  - We know the order, orientation, and spacing, but just not the bases. Fill with Ns instead

Why do scaffolds end?

Assemblathon Results



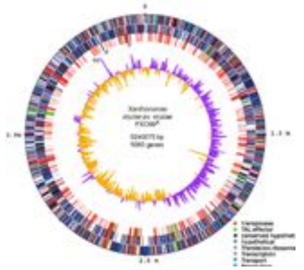
ID	Overall	CPNG50	SPNG50	Struct.	CC50	Subs.	Copy. Num.	Cov. Tot.	Cov. CDS
BGI	36	⭐					☆	⭐	☆
Broad	37	☆	★	★	⭐				
WTSI-S	46		⭐	☆	★	⭐			
CSHL	52	★							☆
BCCGSC	53							☆	⭐
DOEJGI	56		☆	⭐	☆	★			
RHUL	58								

- 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 or Spades

Assemblathon 1: A competitive assessment of de novo short read assembly methods

Earl et al. (2011) Genome Research. 21: 2224-2241

Assembly Summary



Assembly quality depends on

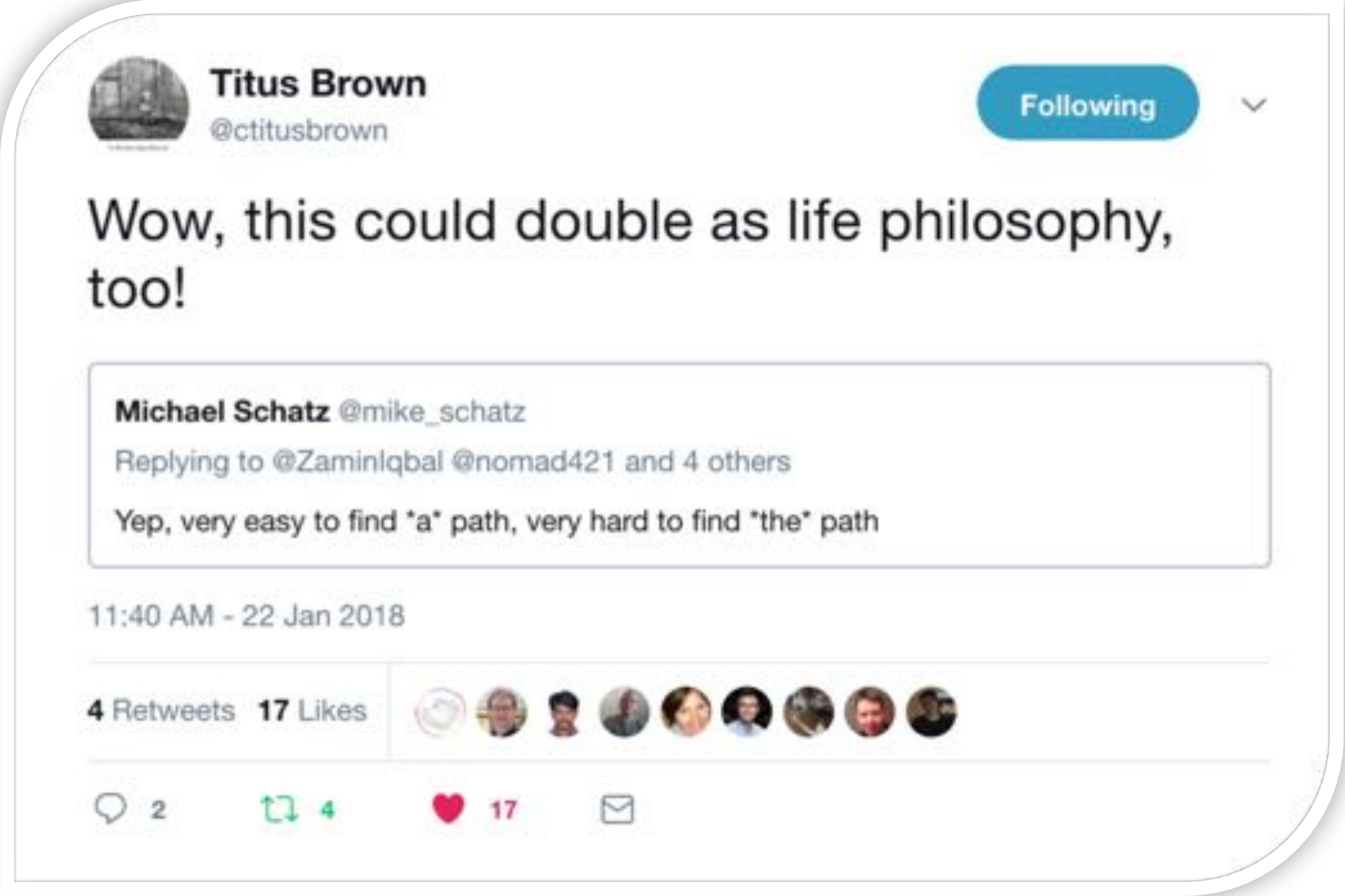
1. ***Coverage***: low coverage is mathematically hopeless
2. ***Repeat composition***: high repeat content is challenging
3. ***Read length***: longer reads help resolve repeats
4. ***Error rate***: errors reduce coverage, obscure true overlaps

- 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

Next Steps

1. Reflect on the magic and power of DNA ☺
2. Check out the course webpage
3. Register on Piazza
4. Work on Assignment 1
    1. Set up Linux, set up Virtual Machine
    2. Set up Dropbox for yourself!
    3. Get comfortable on the command line



Titus Brown
@ctitusbrown

Wow, this could double as life philosophy, too!

Michael Schatz @mike_schatz

Replying to @ZaminIqbal @nomad421 and 4 others

Yep, very easy to find *a* path, very hard to find *the* path

11:40 AM - 22 Jan 2018

4 Retweets 17 Likes

Welcome to Applied Comparative Genomics

https://github.com/schatzlab/appliedgenomics2018

Questions?