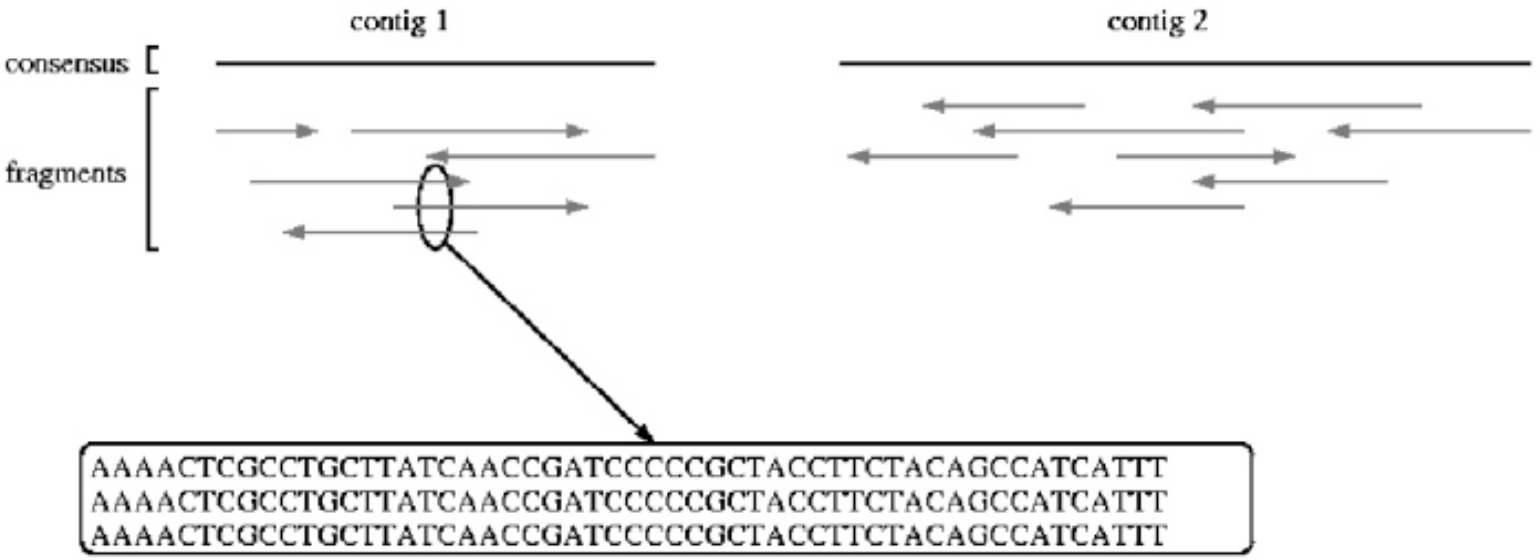
Assembly

Outline

- Random sequencing and coverage
- Assembling by unique overlaps
  - Clustering
  - Memory use
- De Bruijn k-mer graphs
- Heuristics
- Challenges with assembly
  - More data not necessarily better

Assembly

- Short-read assembly is problematic
- Relies on very deep coverage, ruthless read trimming, paired ends.



UMD assembly primer (cbcb.umd.edu)

Short read lengths are hard.

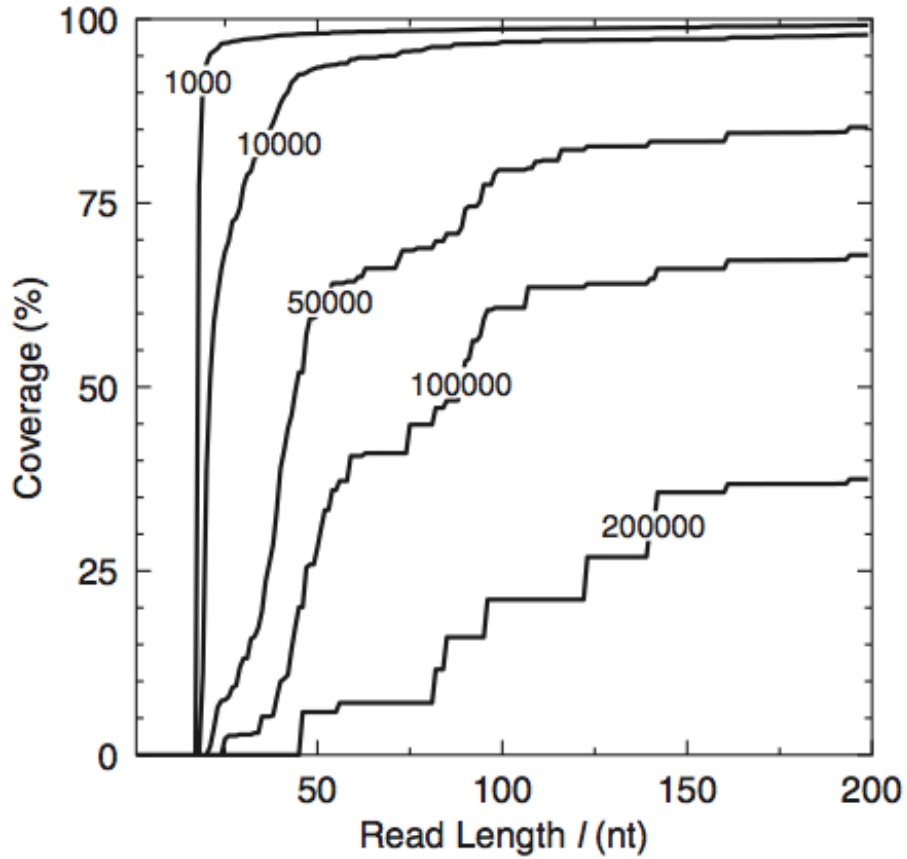


Figure 3. Percentage of the *E.coli* genome covered by contigs greater than a threshold length as a function of read length.

Whiteford et al., Nuc. Acid Res, 2005

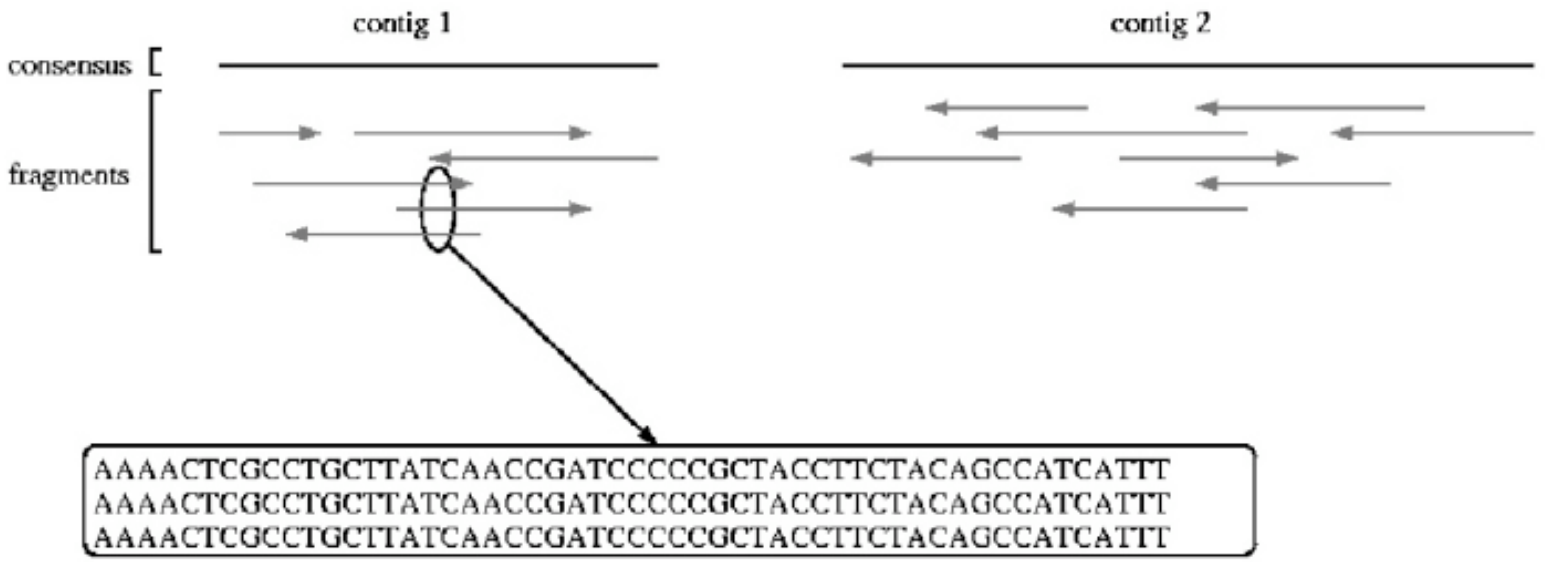
Two main challenges for *de novo* sequencing, apart from read length

- Repeats.
- Low coverage.

Both introduce breaks in the construction of contigs.

Repeats

- Overlaps don't place sequences uniquely when there are repeats present.

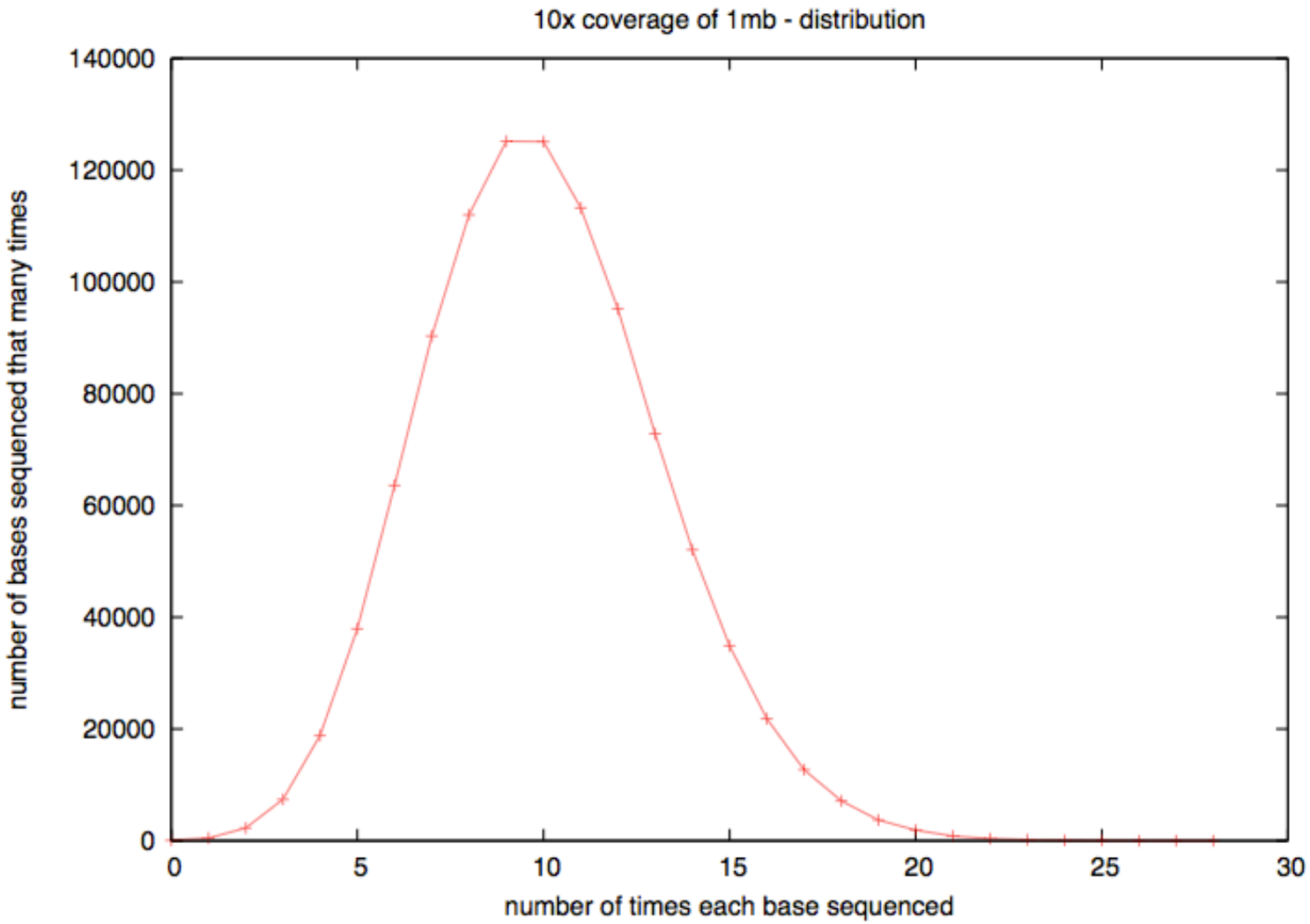


UMD assembly primer (cbcb.umd.edu)

Coverage

- "1x" doesn't mean every DNA sequence is read once.
- It means that, if sampling were *systematic*, it would be.
- Sampling isn't systematic, it's random!

Coverage varies widely for low x



Two basic assembly approaches

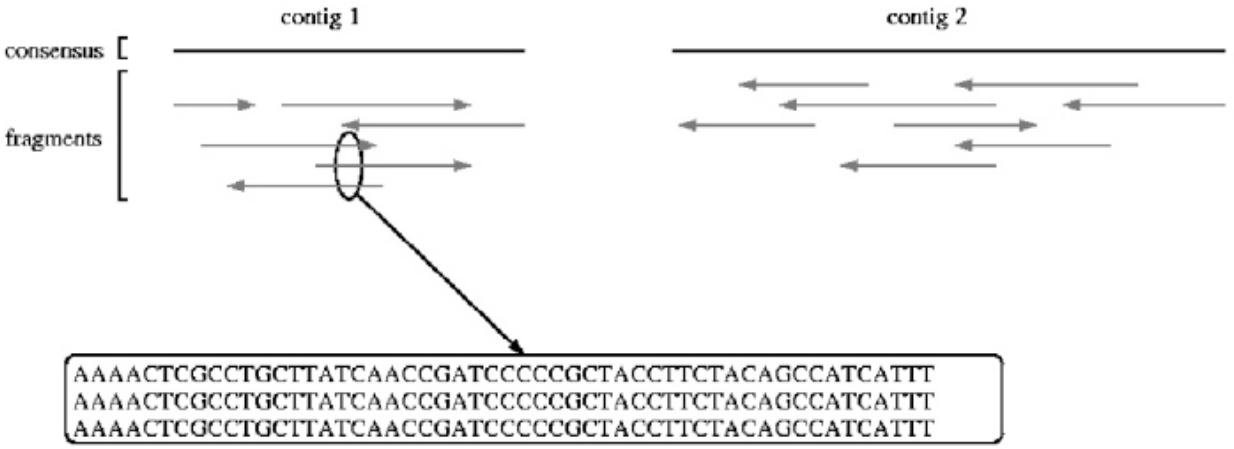
- Overlap/layout/consensus
- De Bruijn k-mer graphs

The former is used for long reads, esp all Sanger-based assemblies. The latter is used because of memory efficiency.

Overlap/layout/consensus

Essentially,

1. Calculate all overlaps
2. Cluster based on overlap.
3. Do a multiple sequence alignment



UMD assembly primer (cbcb.umd.edu)

K-mers

Essentially, break reads (of any length) down into multiple overlapping words of fixed length *k*.

```
ATGGACCAGATGACAC (k=12) =>

ATGGACCAGATG
 TGGACCAGATGA
  GGACCAGATGAC
   GACCAGATGACA
    ACCAGATGACAC
```

K-mers

Table 1A. Mean number of false placements of *K*-mers on the genome

K	*Escherichia coli*	*Saccharomyces cerevisiae*	*Arabidopsis thaliana*	*Homo sapiens*
200	0.063	0.26	0.053	0.18
160	0.068	0.31	0.064	0.49
120	0.074	0.39	0.086	1.7
80	0.082	0.49	0.15	7.2
60	0.088	0.58	0.27	18
50	0.091	0.63	0.39	32
40	0.095	0.69	0.65	78
30	0.11	0.77	1.5	330
20	0.15	1.0	5.7	2100
10	18	63.8	880	40,000

Butler et al., Genome Res, 2009

K-mers

Table 1B. Fraction of *K*-mers having a unique placement on the genome

K	*E. coli* (%)	*S. cerevisiae* (%)	*A. thaliana* (%)	*H. sapiens* (%)
200	98.5	95.9	97.4	97.6
160	98.3	95.6	97.1	97.2
120	98.2	95.2	96.6	96.6
80	98.0	94.7	95.4	95.2
60	97.8	94.4	94.4	93.1
50	97.7	94.2	93.4	91.2
40	97.6	93.9	92.2	88.3
30	97.4	93.5	90.4	83.4
20	97.0	92.9	86.5	71.8
10	0.0	0.0	0.0	0.0

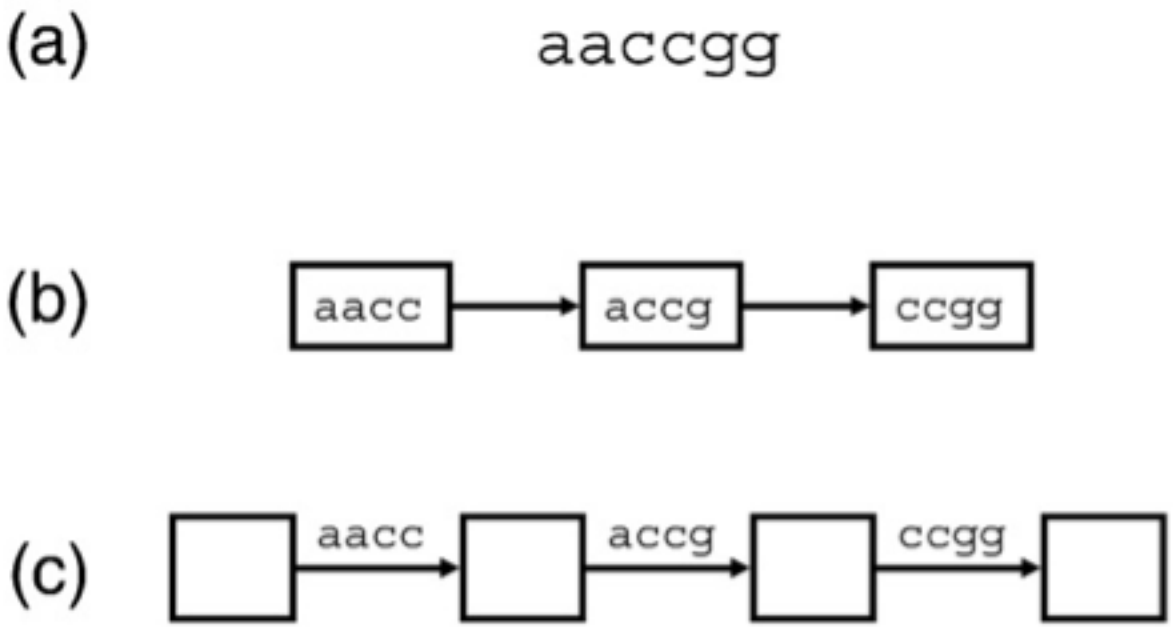
Butler et al., Genome Res, 2009

Big genomes are problematic

Species	Ploidy	Genome size (kb)	Reference N50 (kb)	Component N50 (kb)	Edge N50 (kb)	Ambiguities per megabase	Coverage (%)	Coverage by perfect edges ≥10 kb (%)
C. jejuni	1	1800	1800	1800	1800	0.0	100.0	100.0
E. coli	1	4600	4600	4600	4600	0.0	100.0	100.0
B. thailandensis	1	6700	3800	1800	890	2.7	99.8	99.5
E. gossypii	1	8700	1500	1500	890	2.6	100.0	99.9
S. cerevisiae	1	12,000	920	810	290	28.7	98.7	94.9
S. pombe	1	13,000	4500	1400	500	19.1	98.8	97.5
P. stipitis	1	15,000	1800	900	700	8.6	97.9	96.3
C. neoformans	1	19,000	1400	810	770	4.5	96.4	93.4
Y. lipolytica	1	21,000	3600	2200	290	6.2	99.1	98.6
Neurospora crassa	1	39,000	660	640	90	17.4	97.0	92.5
H. sapiens region	2	10,000	10,000	490	2	68.2	97.3	0.2

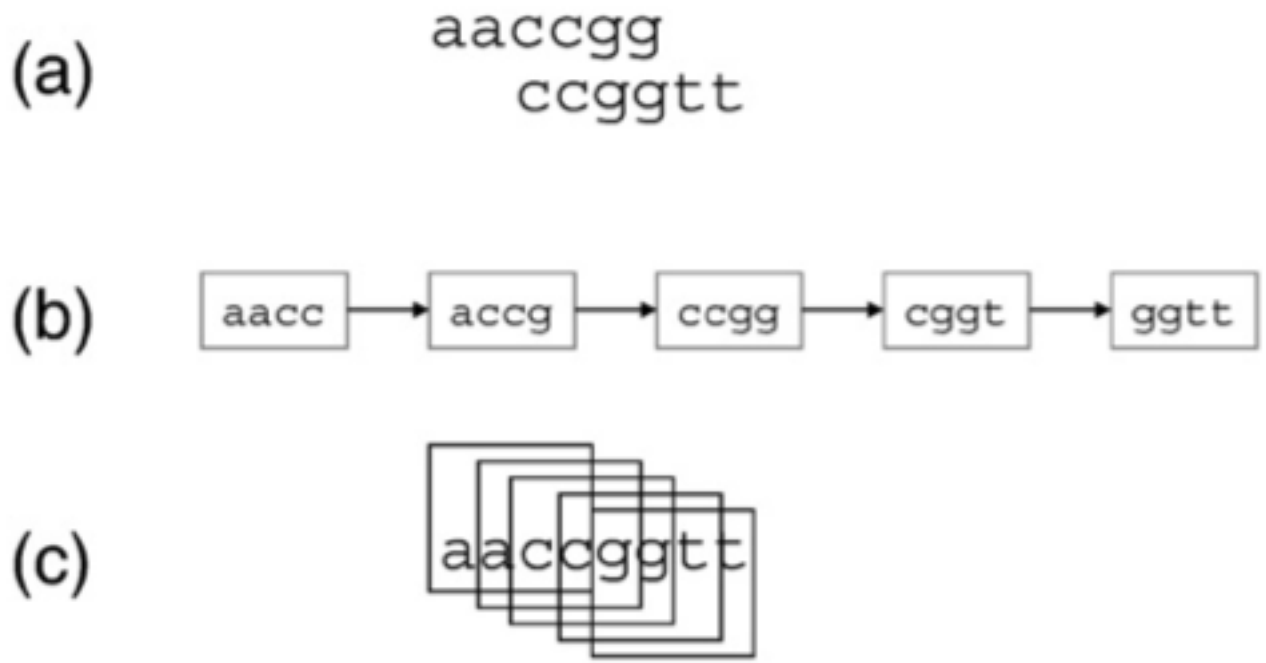
Butler et al., Genome Res, 2009

K-mer graphs



J.R. Miller et al. / Genomics (2010)

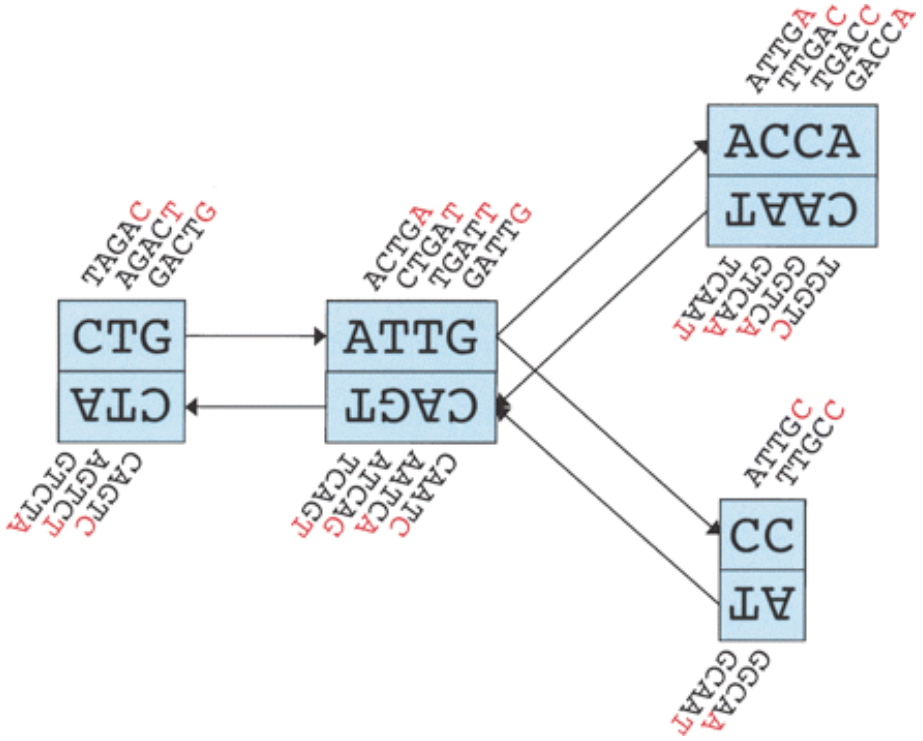
K-mer graphs - overlaps



J.R. Miller et al. / Genomics (2010)

K-mer graphs - vlvet

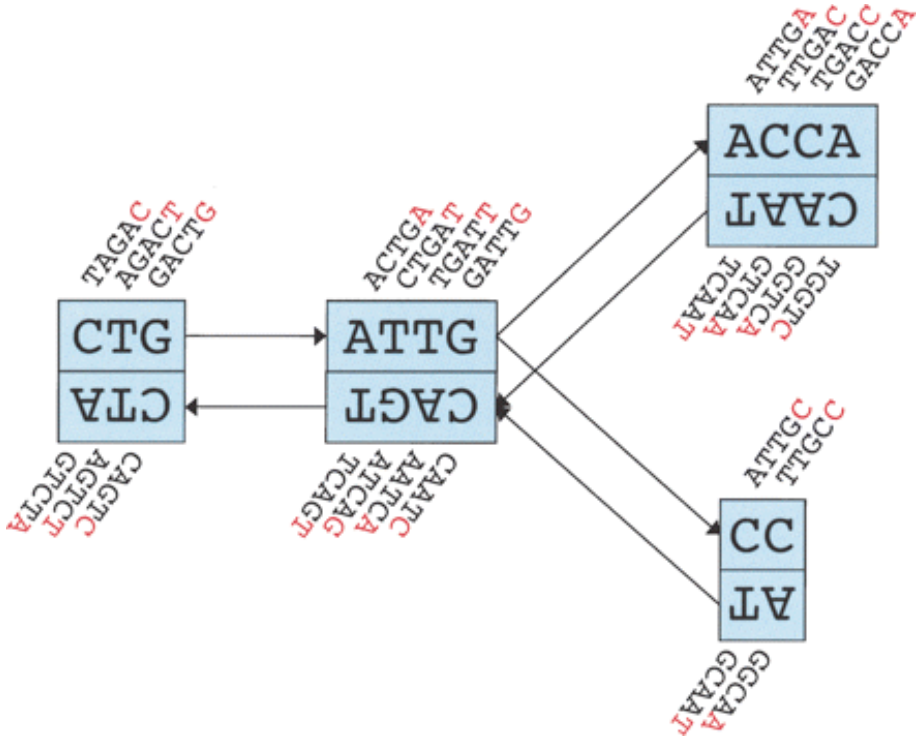
Each rectangle represents k-mer from overlapping reads (at angles).



Zerbino and Birney, Genome Res, 2008

K-mer graphs

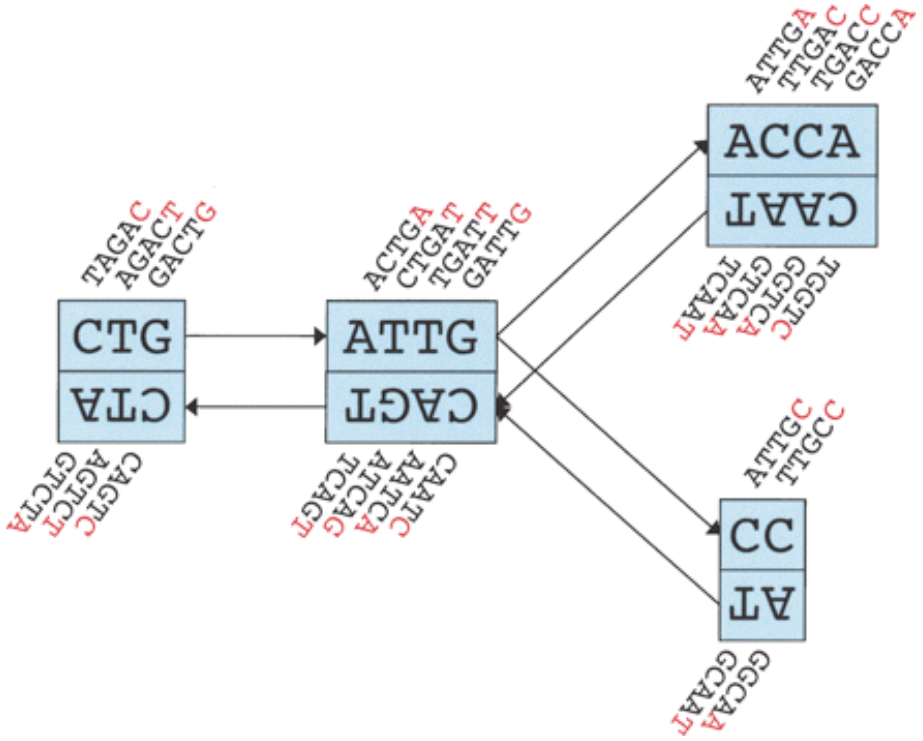
Connectivity shows *possible* contiguous sequences.



Zerbino and Birney, Genome Res, 2008

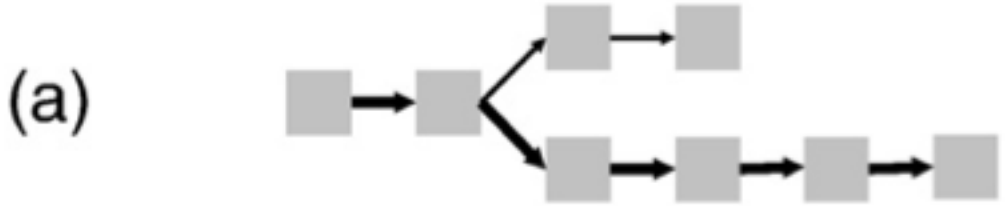
K-mer graphs

Goal is to construct a *single* path through this graph.



Zerbino and Birney, Genome Res, 2008

K-mer graph complexity - spur



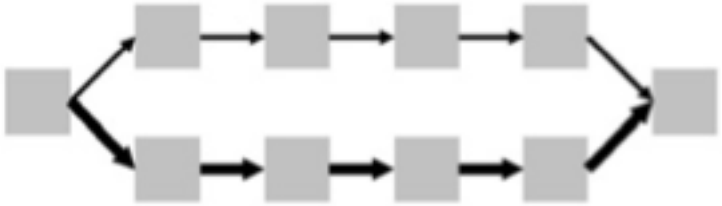
(Short) dead-end in graph.

Can be caused by error at the end of some overlapping reads

J.R. Miller et al. / Genomics (2010)

K-mer graph complexity - bubble

(b)



Multiple parallel paths that diverge and join.

Caused by sequencing error and true polymorphism / polyploidy in sample.

J.R. Miller et al. / Genomics (2010)

K-mer graph complexity – "frayed rope"

(c)

Converging, then diverging paths.

Caused by repetitive sequences.

J.R. Miller et al. / Genomics (2010)

Resolving graph complexity

- Primarily heuristic approaches.
- Detecting complex graph structures can generally not be done efficiently.
- Much of the divergence in functionality of new assemblers comes from this.
- Three examples:

Read threading

Single read spans k-mer graph => extract the single-read path.

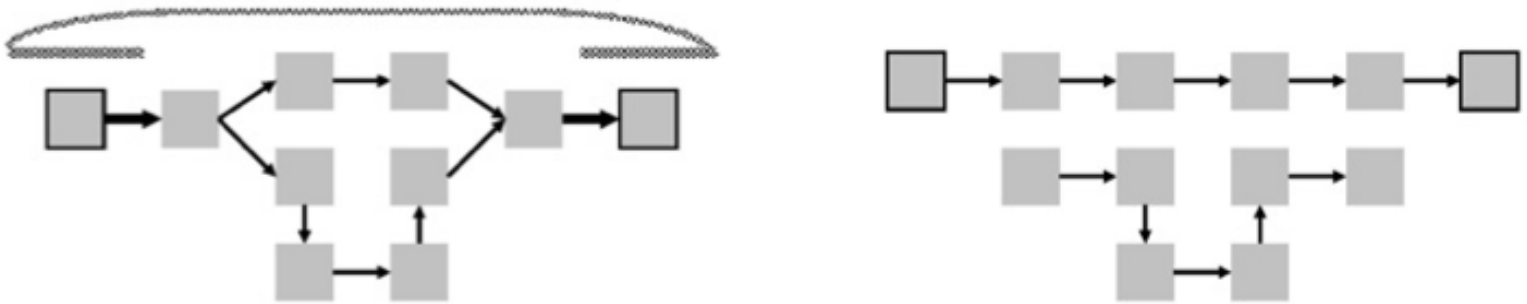
J.R. Miller et al. / Genomics (2010)

Mate threading

Resolve "frayed-rope" pattern caused by repeats, by separating paths based on mate-pair reads.

J.R. Miller et al. / Genomics (2010)

Path following

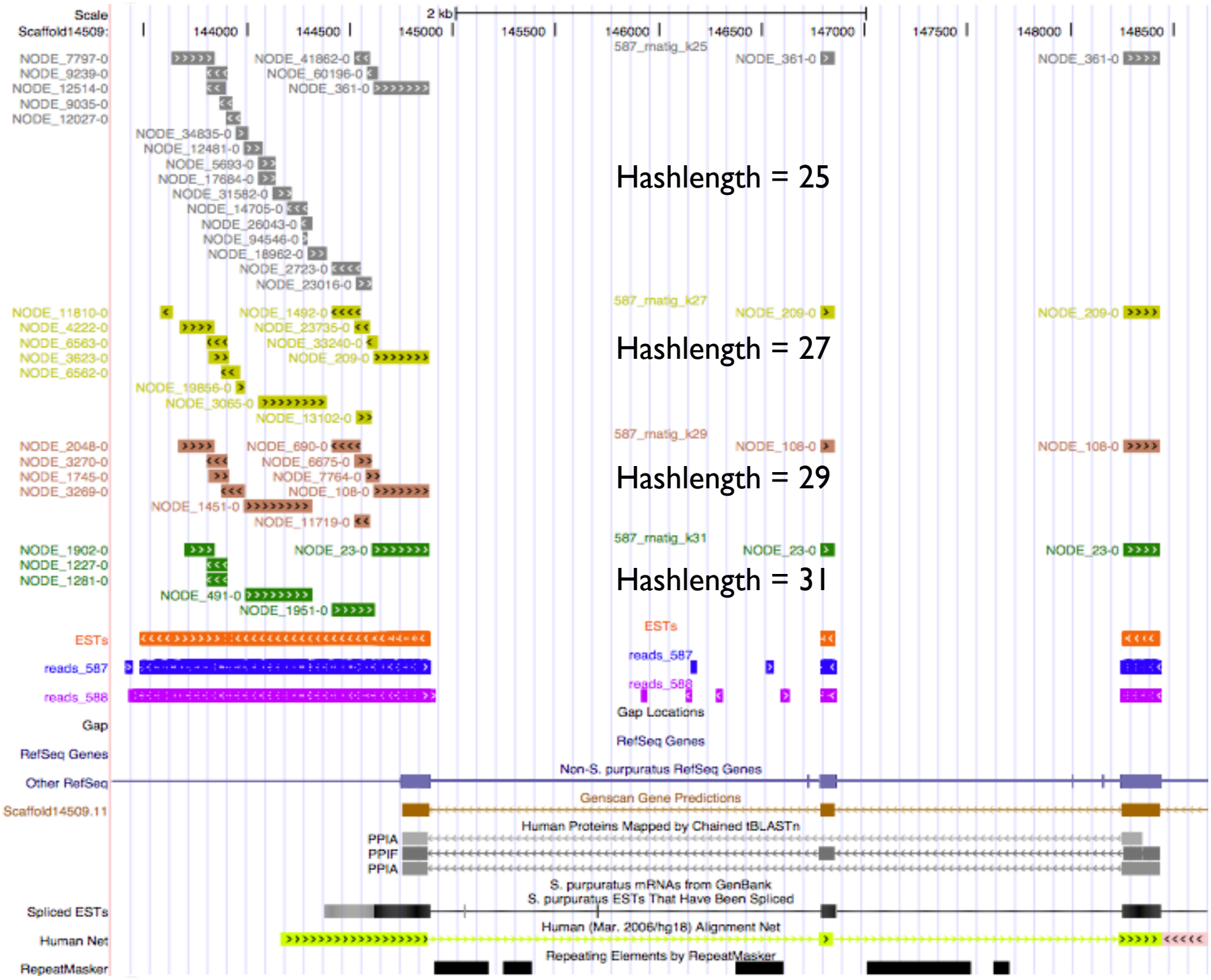


Reject inconsistent paths based on mate-pair reads and insert size.

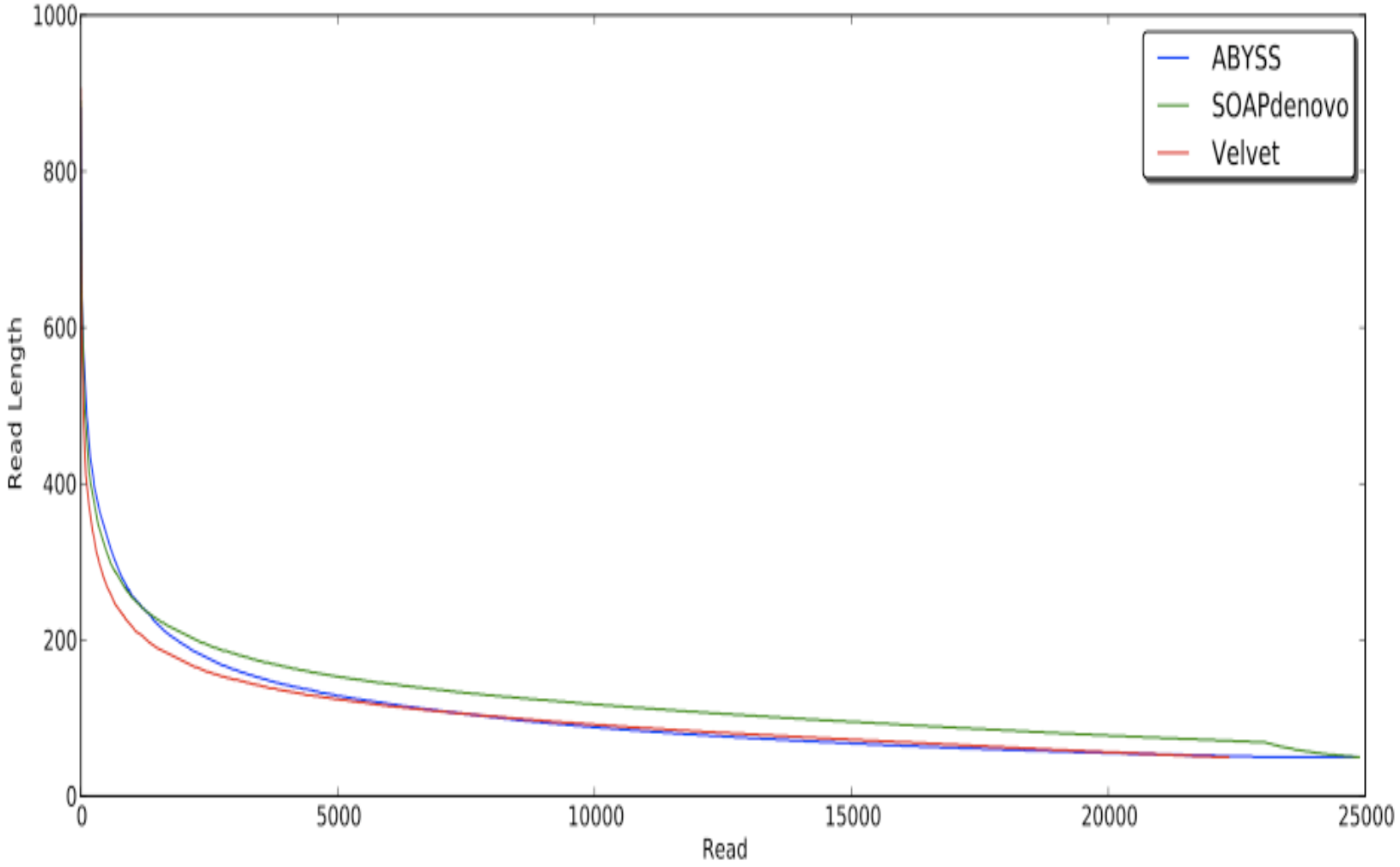
J.R. Miller et al. / Genomics (2010)

More assembly problems

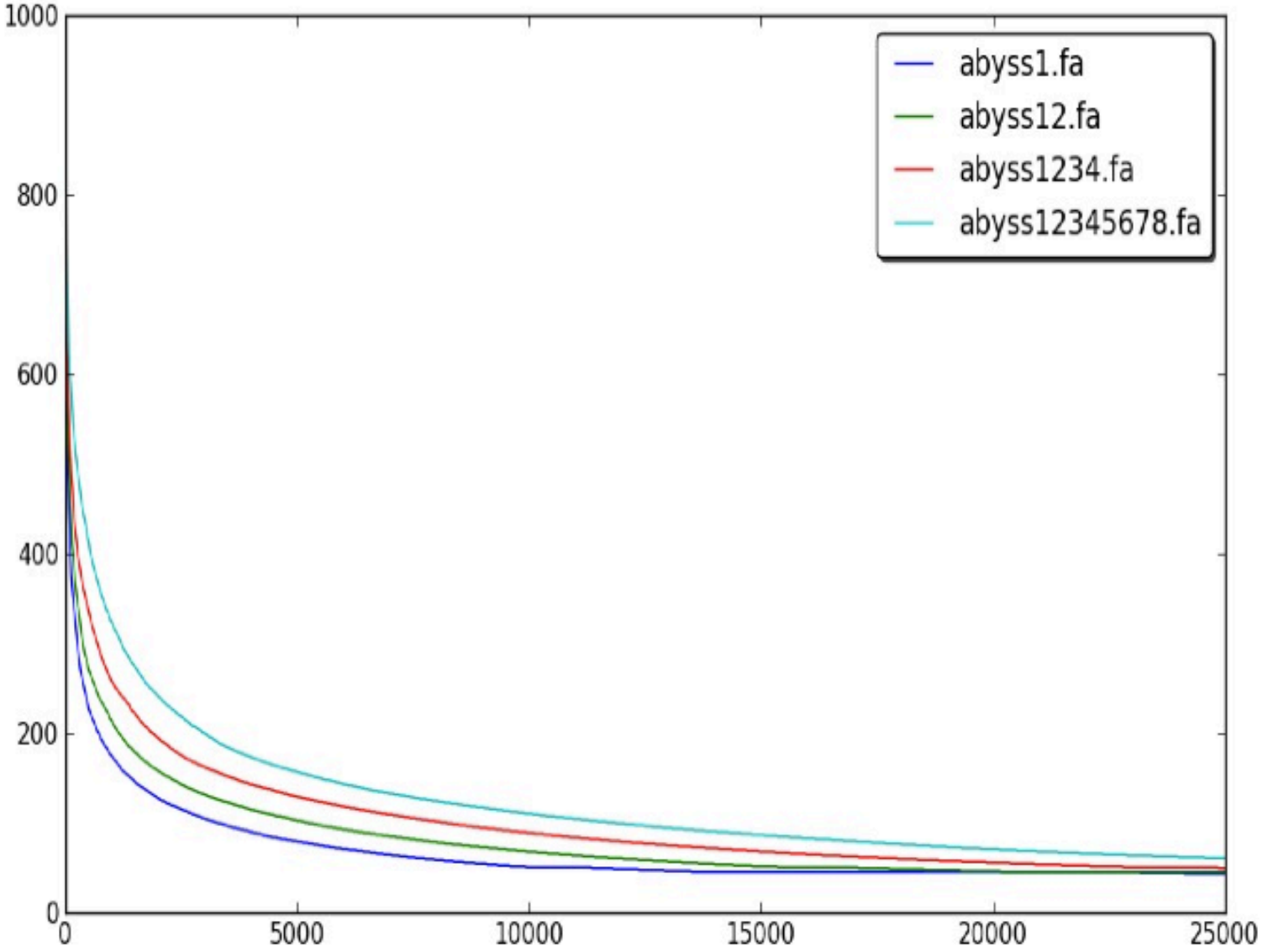
- RNAseq has variation in copy number; naïve assemblers can treat this as repetitive and eliminate it.
- Assembly requires gobs of memory (4 lanes, 60m reads => ~ 150gb RAM)
- How to evaluate?
  - Lamprey RNAseq: 25% of reads map to assembled contigs.


Scale
Scaffold14509:
2 kb
144000
144500
145000
145500
146000
146500
147000
147500
148000
148500
587_rnatig_k25
NODE_7797-0
NODE_9239-0
NODE_12514-0
NODE_9035-0
NODE_12027-0
NODE_41862-0
NODE_60196-0
NODE_361-0
NODE_361-0
NODE_361-0
NODE_34835-0
NODE_12481-0
NODE_5693-0
NODE_17684-0
NODE_31582-0
NODE_14705-0
NODE_26043-0
NODE_94546-0
NODE_18962-0
NODE_2723-0
NODE_23016-0
Hashlength = 25
587_rnatig_k27
NODE_11810-0
NODE_4222-0
NODE_6563-0
NODE_3623-0
NODE_6562-0
NODE_1492-0
NODE_23735-0
NODE_33240-0
NODE_209-0
NODE_209-0
NODE_209-0
NODE_19856-0
NODE_3065-0
NODE_13102-0
Hashlength = 27
587_rnatig_k29
NODE_2048-0
NODE_3270-0
NODE_1745-0
NODE_3269-0
NODE_690-0
NODE_6675-0
NODE_7764-0
NODE_108-0
NODE_108-0
NODE_108-0
NODE_1451-0
NODE_11719-0
Hashlength = 29
587_rnatig_k31
NODE_1902-0
NODE_1227-0
NODE_1281-0
NODE_23-0
NODE_23-0
NODE_23-0
NODE_491-0
NODE_1951-0
Hashlength = 31
ESTs
reads_587
reads_588
Gap
RefSeq Genes
Other RefSeq
Scaffold14509.11
Spliced ESTs
Human Net
RepeatMasker
ESTs
reads_587
reads_588
Gap Locations
RefSeq Genes
Non-S. purpuratus RefSeq Genes
Genscan Gene Predictions
Human Proteins Mapped by Chained tBLASTn
PPIA
PPIF
PPIA
S. purpuratus mRNAs from GenBank
S. purpuratus ESTs That Have Been Spliced
Human (Mar. 2006/hg18) Alignment Net
Repeating Elements by RepeatMasker

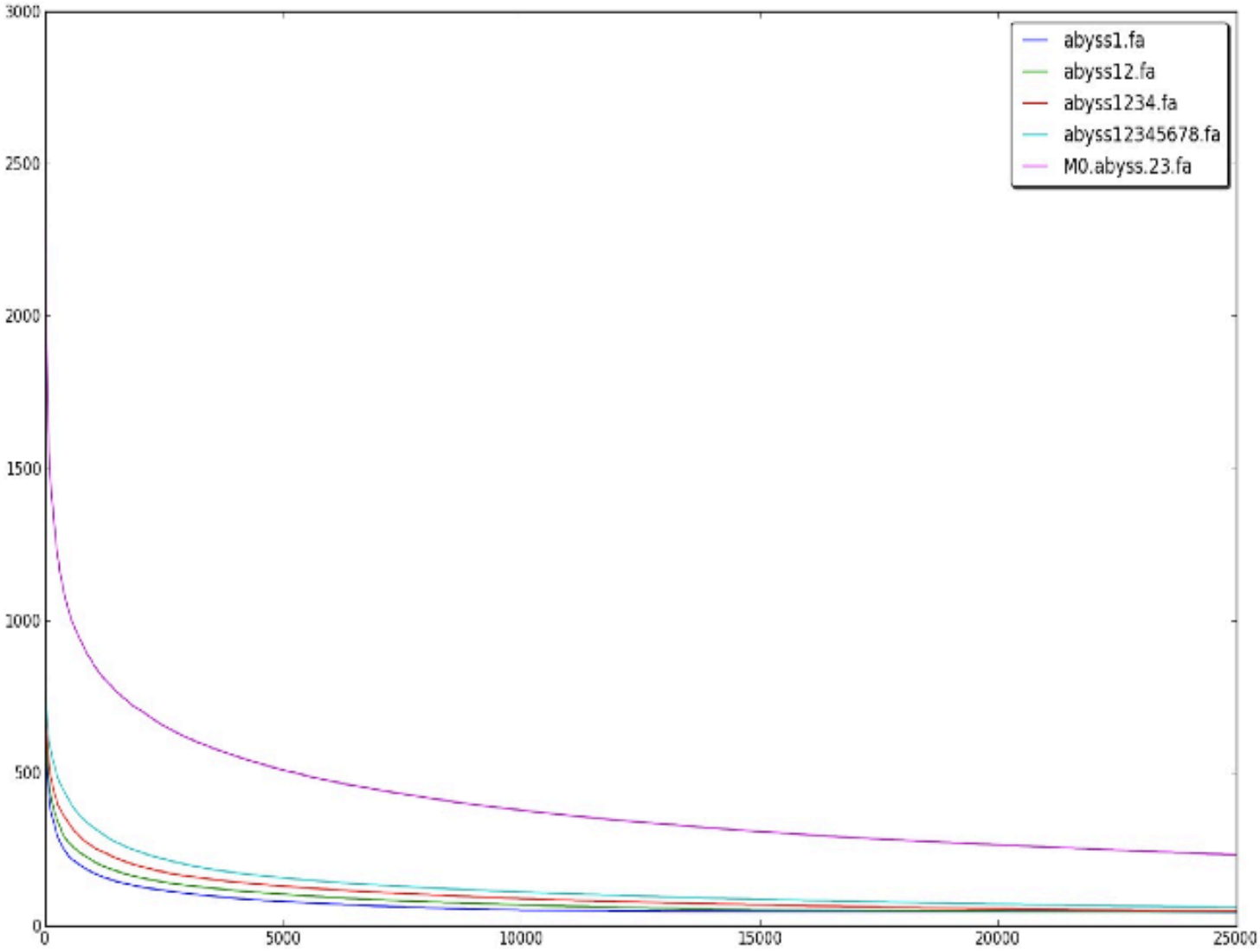
Trying out different assemblers



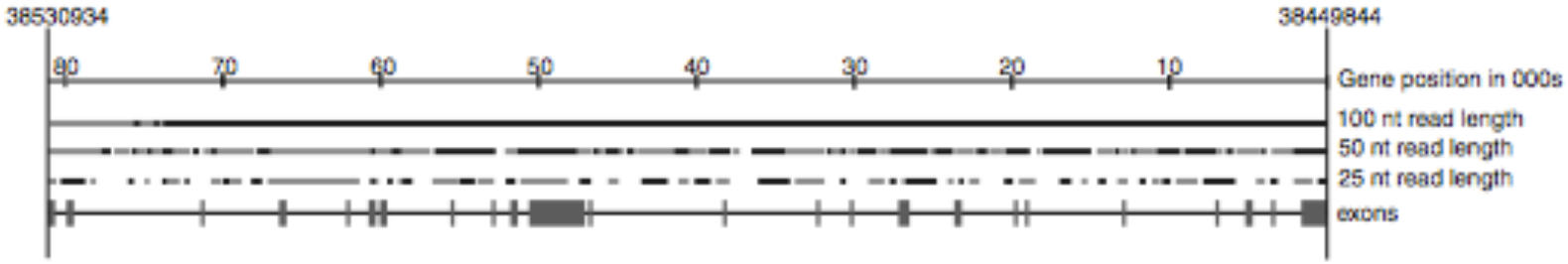
Adding more data starts to saturate



Longer reads help a lot… add 75bp

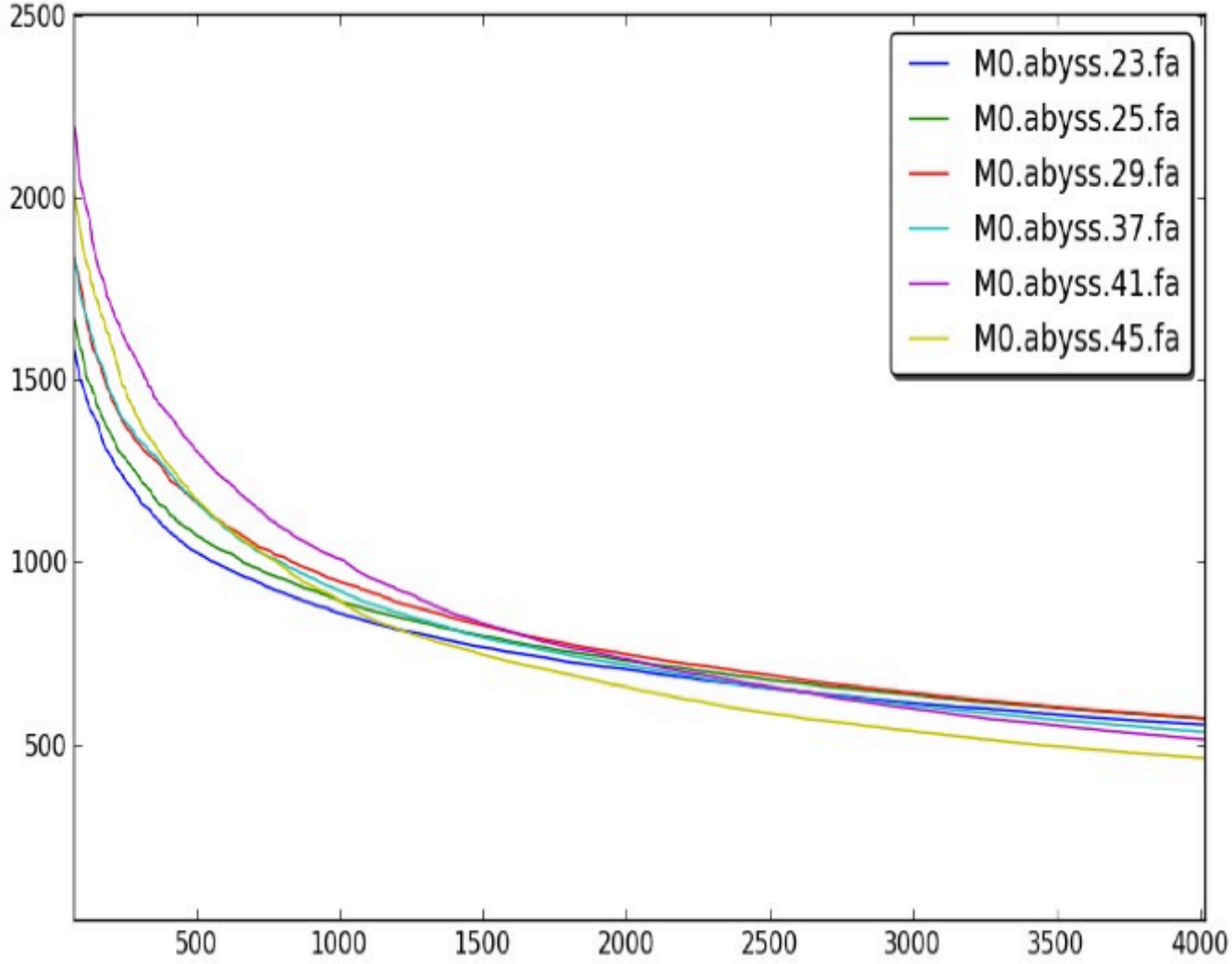


…this matches simulations

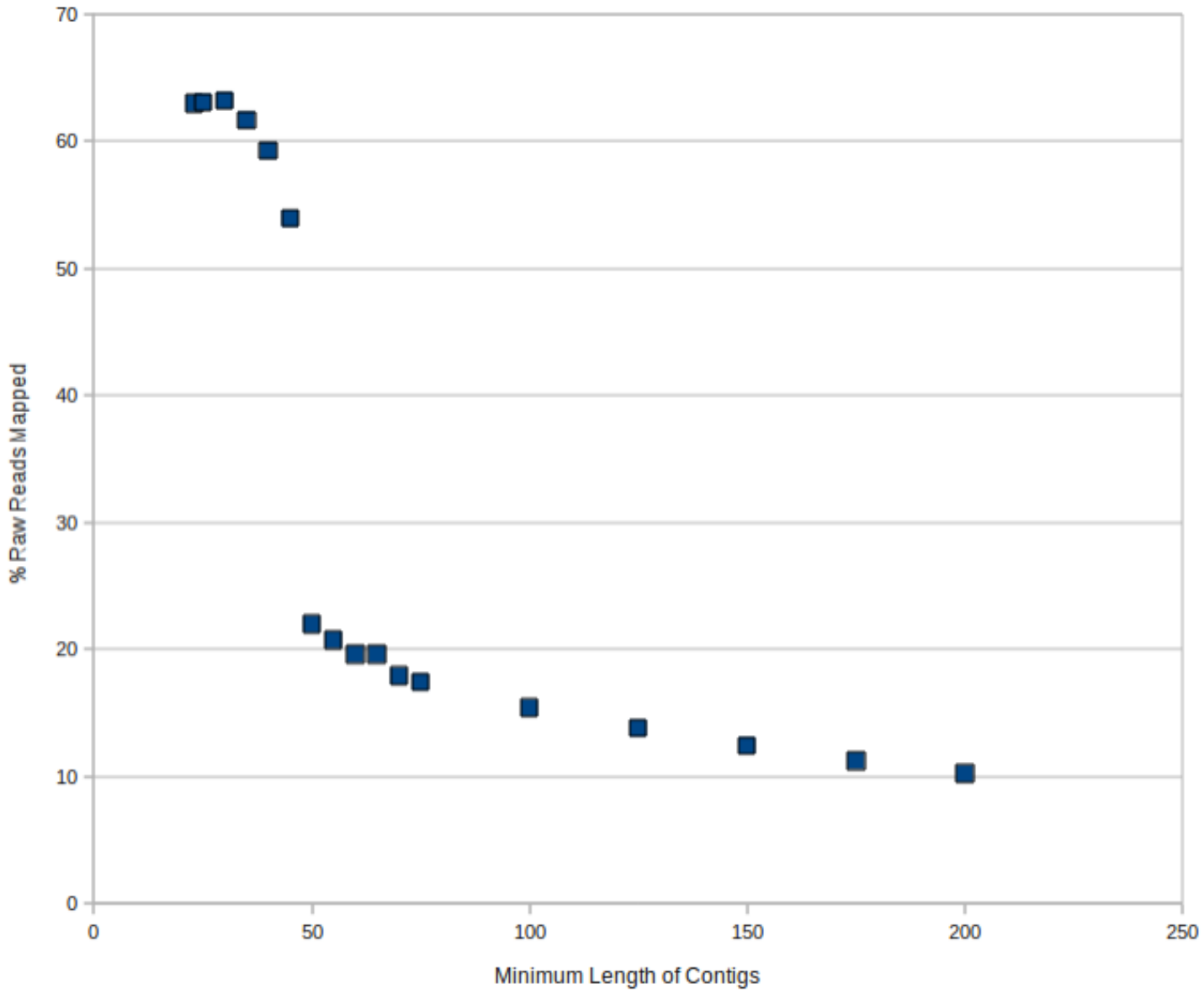


Whiteford et al., Nuc. Acid Res, 2005

Different k values matter a bit



How useful is assembly, anyway?



RNAseq issues

- Most assemblers expect ~1:1 ratio between different portions of the genome.
- Therefore they treat high-coverage k-mers as likely repeats.
- In RNAseq, genes are differentially expressed.

Practical issues

- Do you have enough memory?
- Trim vs use quality scores?
- When is your assembly as good as it gets?
- Paired-end vs longer reads?

- More data is not *necessarily* better, if it introduces more errors.

Today's schedule

- Short-read assembly (Jason; morning)
- Using UCSC genome browser to visualize things (Likit; afternoon)

Reminder: terminate your instances, please ☺

- BBQ and fire.