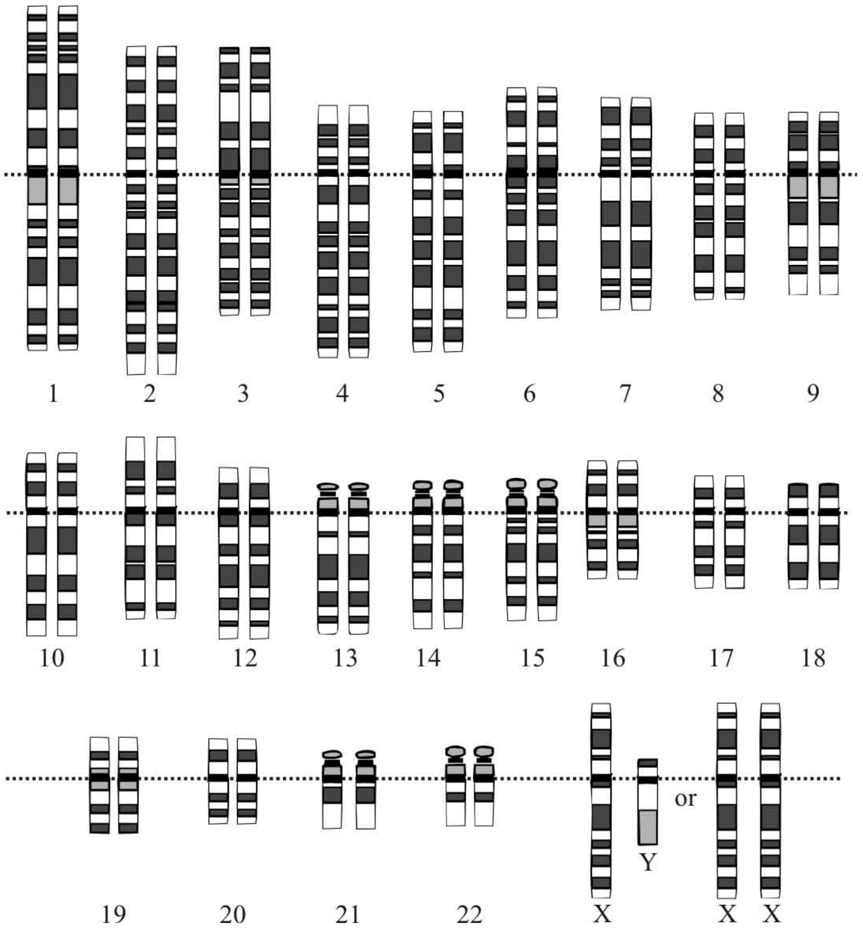
Sequence Assembly and Alignment

Jim Noonan
Department of Genetics
james.noonan@yale.edu
www.yale.edu/noonanlab

The assembly problem

>>10^9 sequencing reads

36 bp - 1 kb



3 Gb

Outline

Basic concepts in genome sequencing and assembly

- Hierarchical vs. whole-genome shotgun methods

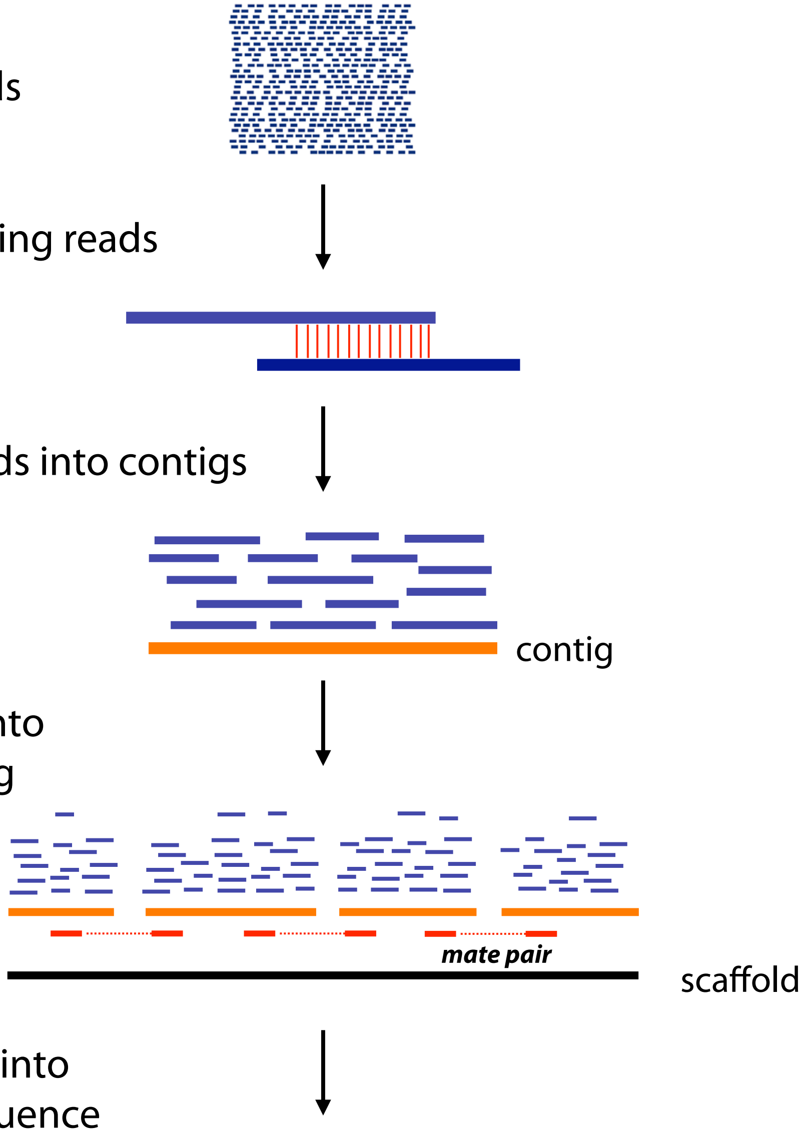
Sources of error in assemblies

- Repeats
- Polymorphism
- Sequencing errors

Alignment and assembly of next-generation sequencing data

- Tiling reads onto reference vs. *de novo* assemblies
- some methods

Sequence assembly: the basic approach



Generate reads

Find overlapping reads

Assemble reads into contigs

Join contigs into scaffolds using mate pairs

Join scaffolds into "finished" sequence

AGTTGTATTATTAGAAACTGAGGGCTAAAAACTGTGCACATACACAGACACACATATTATTTTAATATAGATTTTCAATAATTGGTCTAGGATAAGGATAATATACAG

Terminology and concepts

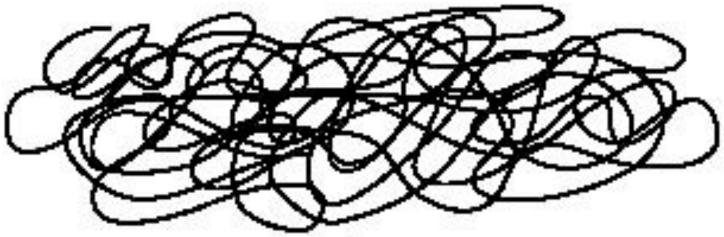
genomic clone:
A vector containing an insert of genomic DNA

BAC: 150-200 kb
Fosmid: 40 kb
Plasmid: 3-5 kb

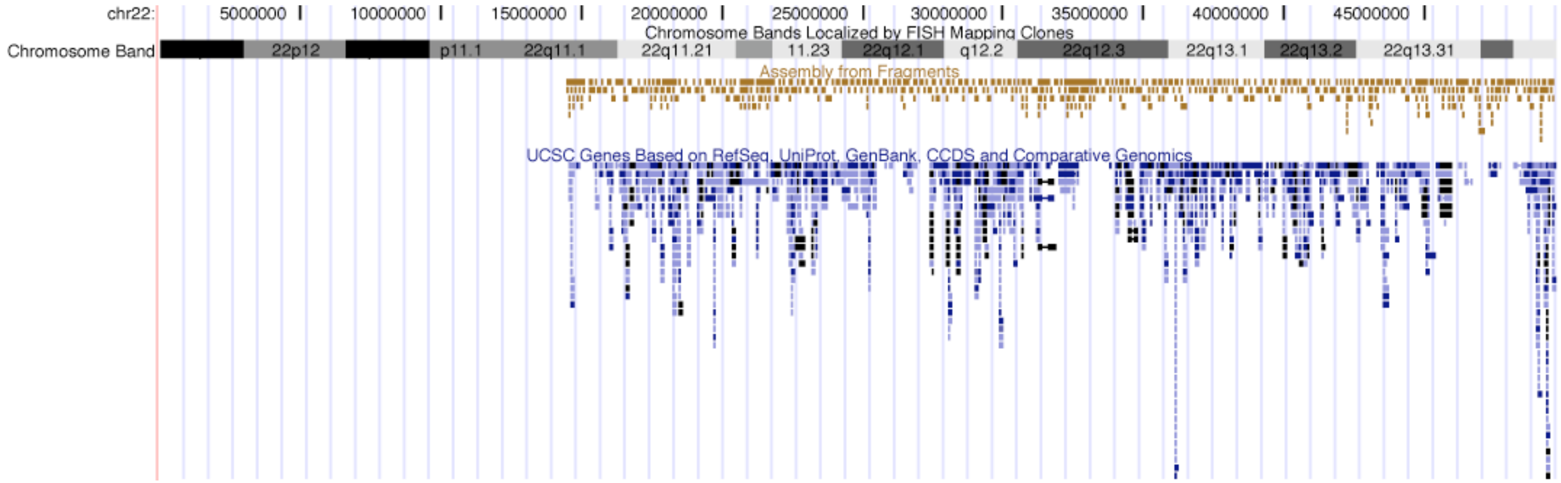
mate pair:
reads from two ends of a clone (plasmid, BAC or fosmid) containing an insert physically mapped to the genome; used to order and orient contigs and scaffolds

coverage:
average number of reads covering a particular position in the assembly

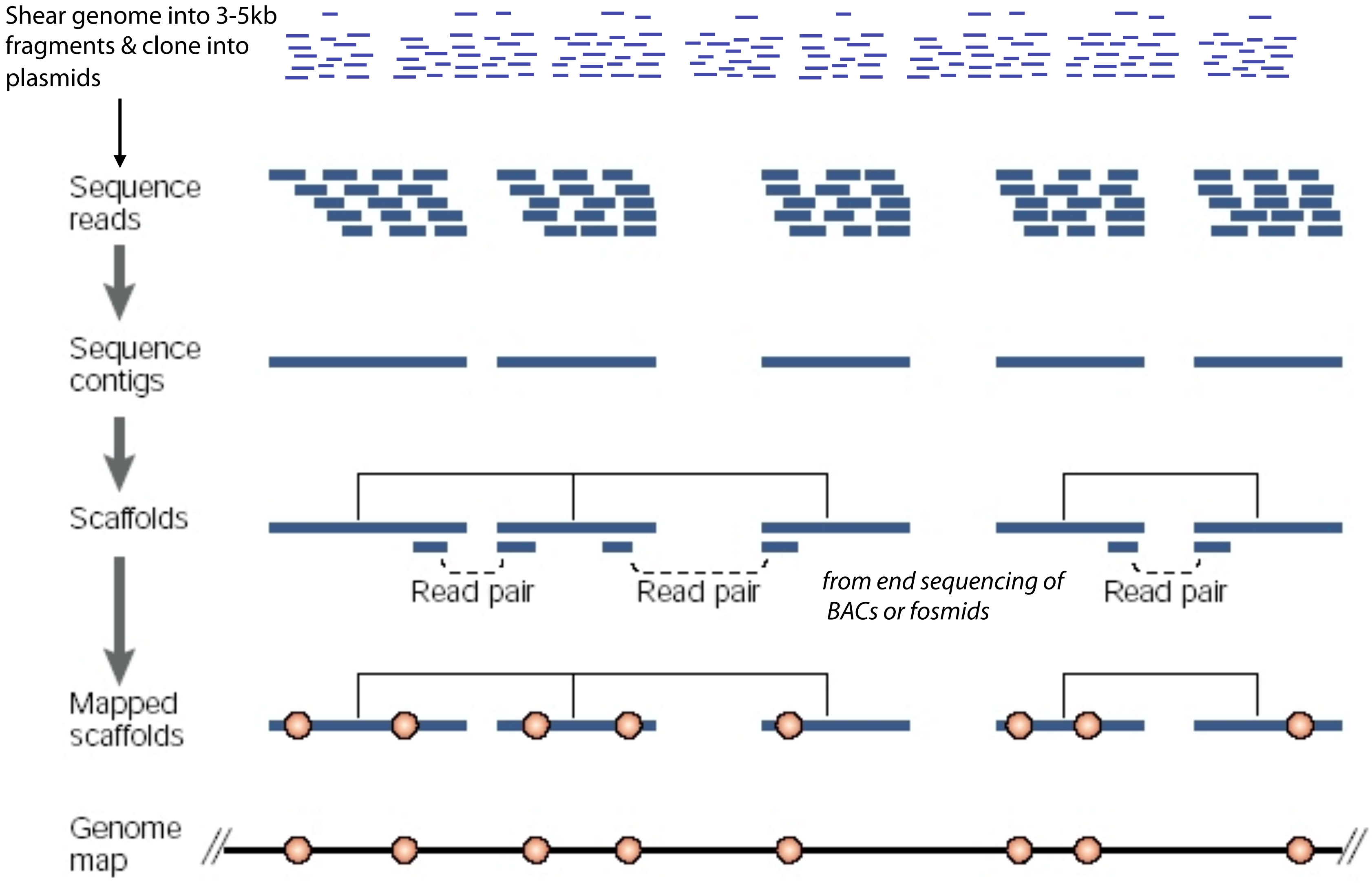
Hierarchical shotgun sequencing



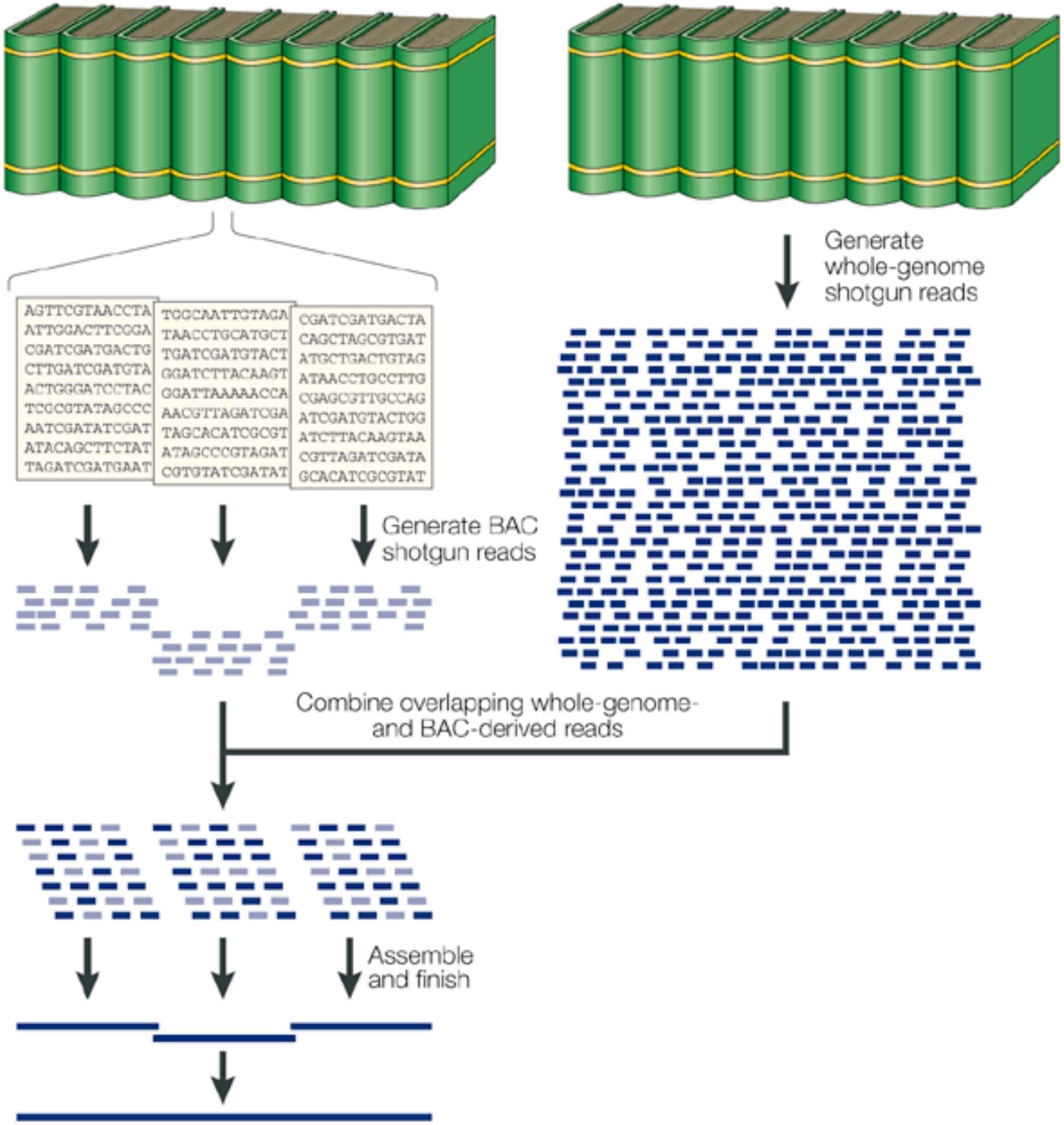
Assembling the human genome



Whole genome shotgun sequencing


Shear genome into 3-5kb fragments & clone into plasmids
Sequence reads
Sequence contigs
Scaffolds
Read pair
Read pair
from end sequencing of BACs or fosmids
Read pair
Mapped scaffolds
Genome map

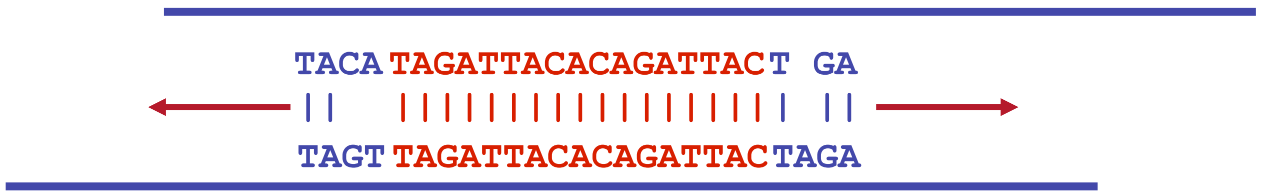
Combined hierarchical - whole genome shotgun



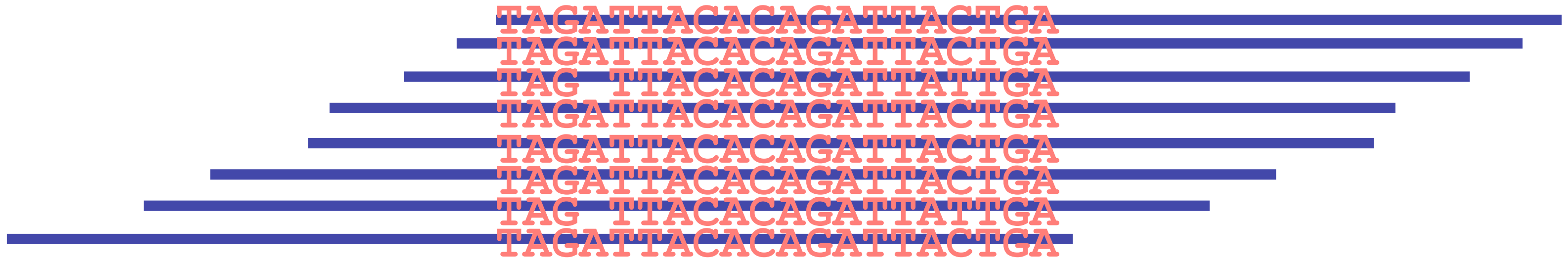
Assembly from individual reads

Identify pairs of reads sharing a common sequence (*k*-mer; *k* > 20)

Extend to full alignment - discard if alignment < 98% identical



Create multiple alignment from overlapping reads



build contigs, scaffolds, etc.

Issues:
- repeats
- sequence errors
- polymorphism

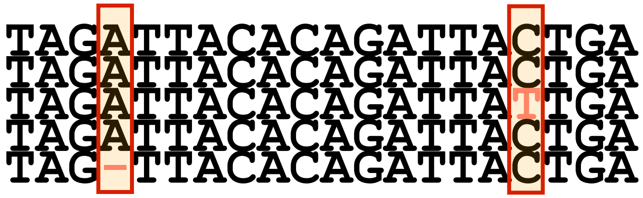
Assembly from individual reads: issues

Repeats

- a *k*-mer represented 1,000,000 times results in $1{,}000{,}000^2$ comparisons
- remove "overrepresented" *k*-mers
- increase read length = increase *k*
- problematic for short read methods

Sequencing errors

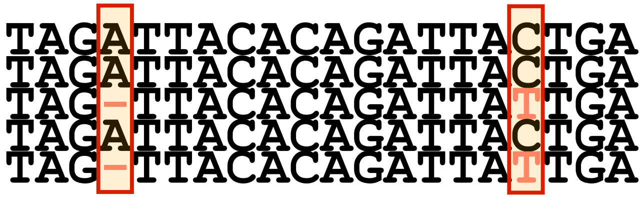
- increase coverage



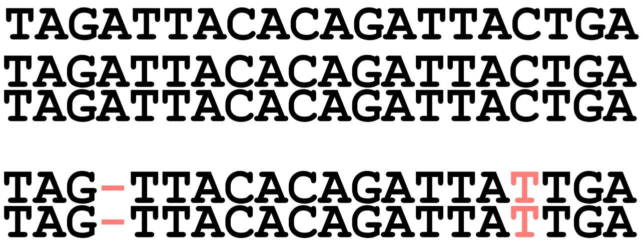
```
TAGATTACACAGATTACTGA
TAGATTACACAGATTACTGA
TAGATTACACAGATTATTGA
TAGATTACACAGATTACTGA
TAG-TTACACAGATTACTGA
```

Polymorphism

- produce consistent high-quality mismatches in one contig or multiple virtually identical contigs



```
TAGATTACACAGATTACTGA
TAGATTACACAGATTACTGA
TAG-TTACACAGATTATTGA
TAGATTACACAGATTACTGA
TAG-TTACACAGATTATTGA
```



```
TAGATTACACAGATTACTGA

TAGATTACACAGATTACTGA
TAGATTACACAGATTACTGA

TAG-TTACACAGATTATTGA
TAG-TTACACAGATTATTGA
```

repeats can also cause this

- increase coverage
- sequence multiple people

Assembly quality

Human draft

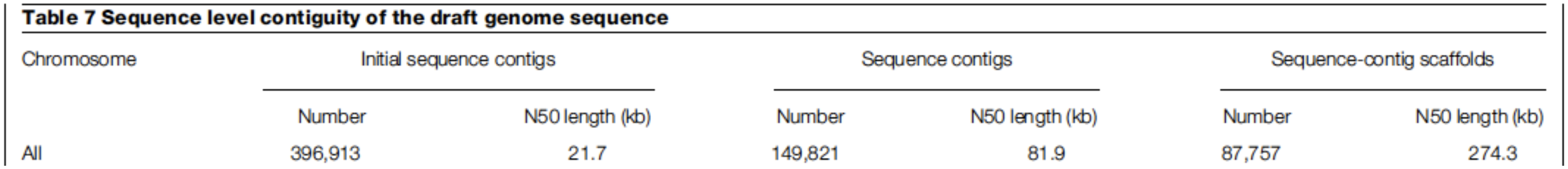


Table 7 Sequence level contiguity of the draft genome sequence

Chromosome	Initial sequence contigs		Sequence contigs		Sequence-contig scaffolds	
	Number	N50 length (kb)	Number	N50 length (kb)	Number	N50 length (kb)
All	396,913	21.7	149,821	81.9	87,757	274.3

~7.5x coverage

Mouse draft

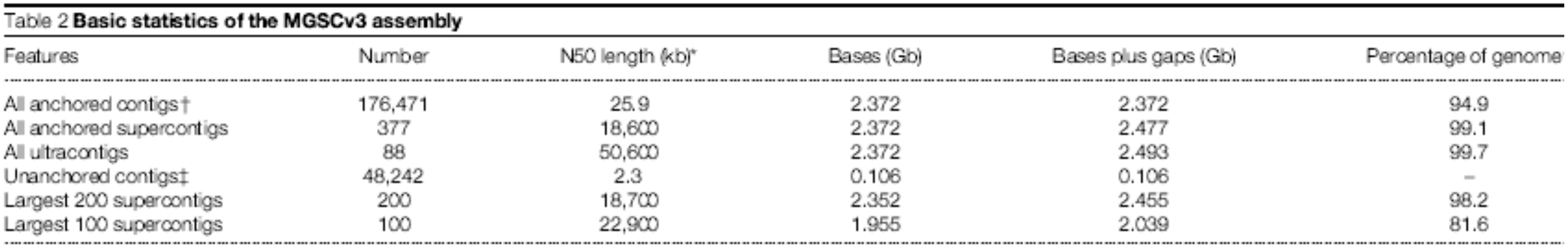


Table 2 **Basic statistics of the MGSCv3 assembly**

Features	Number	N50 length (kb)*	Bases (Gb)	Bases plus gaps (Gb)	Percentage of genome
All anchored contigs†	176,471	25.9	2.372	2.372	94.9
All anchored supercontigs	377	18,600	2.372	2.477	99.1
All ultracontigs	88	50,600	2.372	2.493	99.7
Unanchored contigs‡	48,242	2.3	0.106	0.106	–
Largest 200 supercontigs	200	18,700	2.352	2.455	98.2
Largest 100 supercontigs	100	22,900	1.955	2.039	81.6

~7.7x coverage

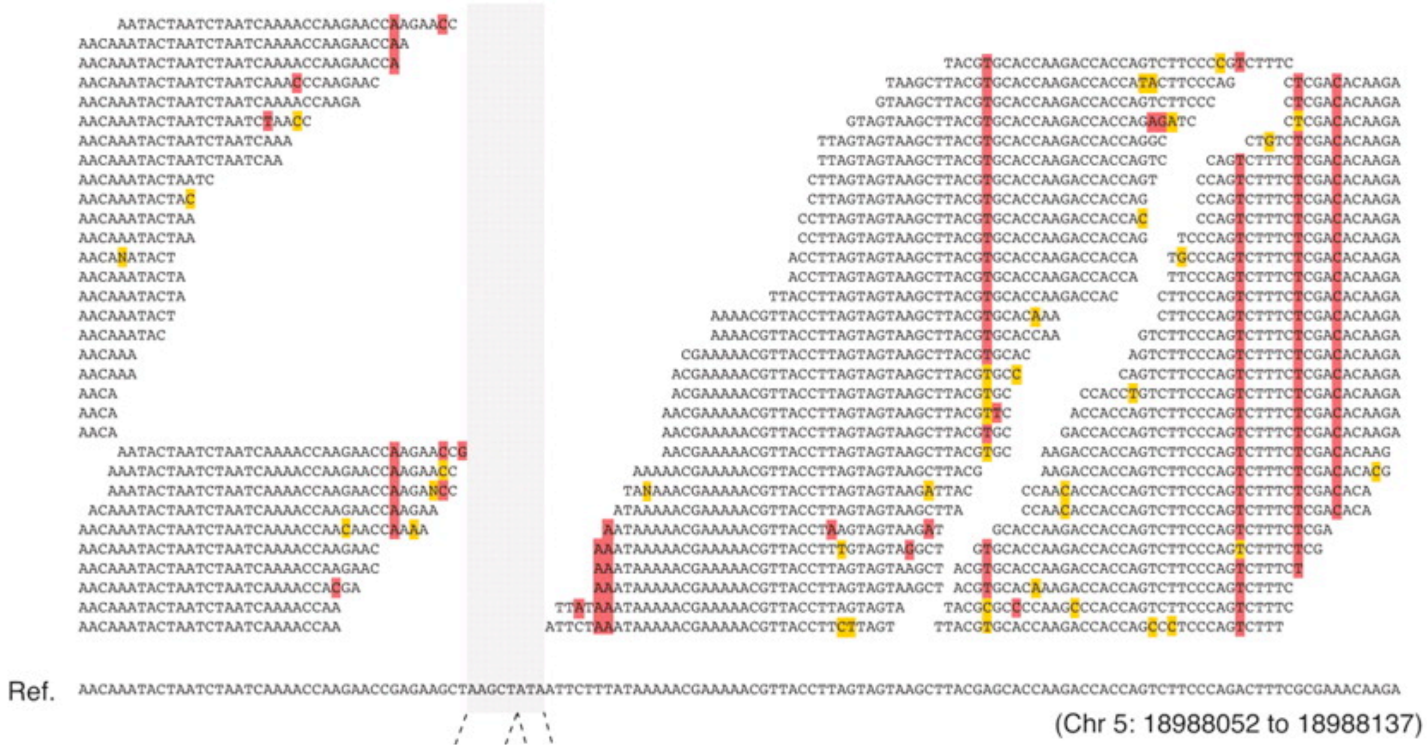
N50 length:
contig length containing a typical nucleotide, i.e. the maximum length *L* such that 50% of all bases lie in contigs at least *L* bases long.

Assemblers

- Phrap
- Celera
- Arachne

designed for Sanger sequencing
(read length, errors, quality scores)

Alignment and assembly with short reads


Ref.
AACAAATACTAATCTAATCAAAACCAAGAACCGAGAAGCTAAGCTATAATTCTTTATAAAAACGAAAAACGTTACCTTAGTAGTAAGCTTACGAGCACCAAGACCACCAGTCTTCCCAGACTTTCGCGAAACAAGA
(Chr 5: 18988052 to 18988137)

Two tasks:

Map to reference genome

- many tools

De novo assembly

- much harder
- reference-guided assembly (MOSAIK)
- “true” *de novo* assembly (Velvet)

Analysis depends on application

Mapping to reference genome

- useful for interrogating the “known” genome
- RNA sequencing
- ChIP sequencing
- SNP detection (targeted and whole-genome)
- methyl-seq
- CNV detection (sometimes)

De novo assembly

- no genome sequence

- unbiased ascertainment of variation in known genome by whole-genome reseq

Mapping short reads to a reference

Eland

aligner for Illumina data

alignment policies:

•allows up to 2 mismatches/alignment

•non-unique alignments are discarded

Maq

•quality aware - takes seq quality into account

•allows non-unique alignments

Index methods

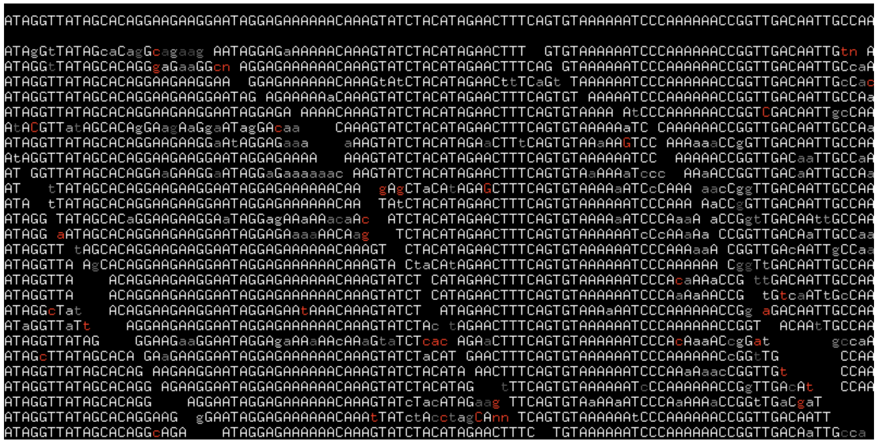
•reference genome is loaded into active memory as *k*-mers

•very fast alignments

•SOAP

•Bowtie

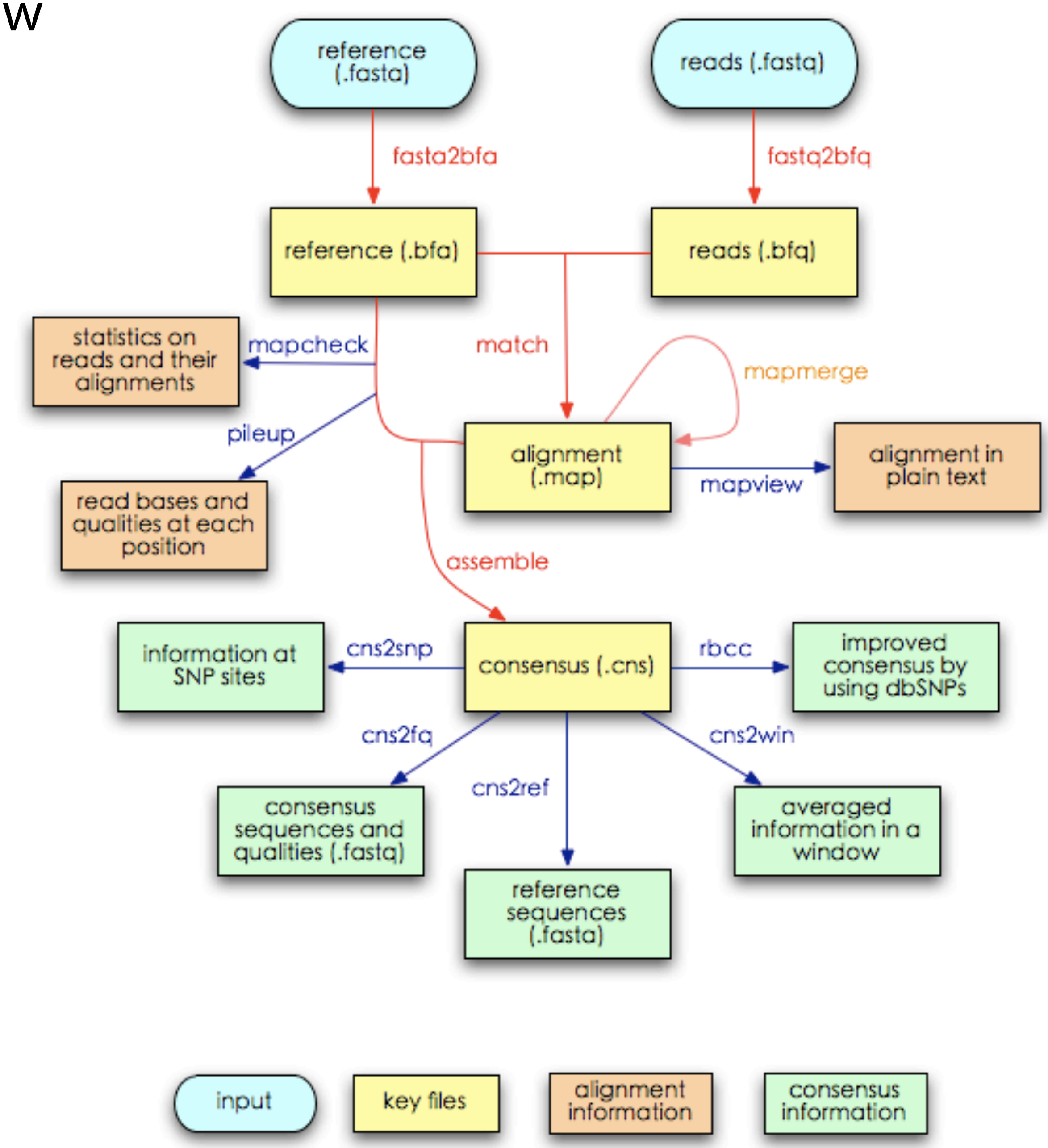
SNP detection, paired-end mapping, RNA-seq, ChIP-seq, etc.



```
ATAGGTTATAGCACAGGAAGAAGGAATAGGAGAAAAAACAAAGTATCTACATAGAACTTTCAGTGTAAAAAATCCCAAAAAACCGGTTGACAATTGCCAA

ATAgGtTATAGcaCagGcagaag AATAGGAGaAAAAACAAAGTATCTACATAGAACTTT  GTGTAAAAAATCCCAAAAAACCGGTTGACAATTGtn A
ATAGGtTATAGCACAGGgaGaaGGcn AGGAGAAAAAACAAAGTATCTACATAGAACTTTCAG GTAAAAAATCCCAAAAAACCGGTTGACAATTGCcaA
ATAGGTTATAGCACAGGAAGAAGGAA  GGAGAAAAAACAAAGtAtCTACATAGAACttTCaGt TAAAAAATCCCAAAAAACCGGTTGACAATTGcCac
ATAGGTTATAGCACAGGAAGAAGGAATAG AGAAAAAaCAAAGTATCTACATAGAACTTTCAGTGT AAAAATCCCAAAAAACCGGTTGACAATTGCCAa
ATAGGTTATAGCACAGGAAGAAGGAATAGGAGA AAAACAAAGTATCTACATAGAACTTTCAGTGTAAAA AtCCCAAAAAACCGGTCGACAATTgcCAA
AtACGTTatAGCACAgGAagAaGgaATagGAcaa    CAAAGTATCTACATAGAACTTTCAGTGTAAAAAaTC CAAAAAACCGGTTGACAATTGCCAa
ATAGGTTATAGCACAGGAAGAAGGaAtAGGAGaaa    aAAGTATCTACATAGAaCTTtCAGTGTAAaAAGTCC AAAaaaCCgGTTGACAATTGCCAA
AtAGGTTATAGCACAGGAAGAAGGAATAGGAGAAAA   AAAGTATCTACATAGAACTTTCAGTGTAAAAAATCC  AAAAACCGGTTGACaaTTGCCaA
AT GGTTATAGCACAGGAaGAAGGaATAGGaGaaaaaac AAGTATCTACATAGAACTTTCAGTGTAaAAAaTccc  AAaACCGGTTGACaATTGCCAa
AT    tTATAGCACAGGAAGAAGGAATAGGAGAAAAAACAA  gAgCTaCAtAGAGCTTTCAGTGTAAAAaATCcCAAA aacCggTTGACAATTGCCAA
ATA  tTATAGCACAGGAAGAAGGAATAGGAGAAAAAACAA  TAtCTACATAGAACTTTCAGTGTAAAAAATCCCAAA AaCCgGTTGACAATTGCCAA
ATAGG TATAGCACaGGAAGAAGGAaTAGGagAAaAAacaAc  ATCTACATAGAACTTTCAGTGTAAAAaATCCCAaaA aCCGgtTGACAAttGCCAA
ATAGG aATAGCACAGGAAGAAGGAATAGGAGAaaaAACAag   TCTACATAGAACTTTCAGTGTAAAAAATcCcAAaAa CCGGTTGACAaTTGCCaa
ATAGGTT tAGCACAGGAAGAAGGAATAGGAGAAAAAACAAAGT  CTACATAGAACTTTCAGTGTAAAAAATCCCAAAaaA CGGTTGAcAATTgCCaa
ATAGGTTA AgCACAGGAAGAAGGAATAGGAGAAAAAACAAAGTA CtaCAtAGAACTTTCAGTGTAAAAAATCCCAAAAAA CggTtGACAATTGCCAA
ATAGGTTA    ACAGGAAGAAGGAATAGGAGAAAAAACAAAGTATCT CATAGAACTTTCAGTGTAAAAAATCCCAcaAAaCCG ttGACAATTGCCAA
ATAGGTTA    ACAGGAAGAAGGAATAGGAGAAAAAACAAAGTATCT CATAGAACTTTCAGTGTAAAAAATCCCAaAaAACCG  tGtcaATtGcCAA
ATAGGcTat   ACAGGAAGAAGGAATAGGAGAAtAAACAAAGTATCT  ATAGAACTTTCAGTGTAAAaAATCCCAAAAAAACCGg aGACAATTGCCAA
ATaGGTTaTt    AGGAAGAAGGAATAGGAGAAAAAACAAAGTATCTAc tAGAACTTTCAGTGTAAAAAATCCCAAAAAAACCGGT  ACAAtTGCCAA
ATAGGTTATAG    GGAAGaaGGAATAGGAgaAAaAAcAAaGtaTCTcac AGAaCTTTCAGTGTAAAAAATCCCAcAaaACcgGat       gccaA
ATAGcTTATAGCACA GAaGAAGGAATAGGAGAAAAAACAAAGTATCTaCAT GAACTTTCAGTGTAAAAAATCCCAAAAAACcGGtTG       CCAA
ATAGGTTATAGCACAG AAGAAGGAATAGGAGAAAAAACAAAGTATCTACATA AACTTTCAGTGTAAAAAATCCCAAaAaacCGGTTGt       CCAA
ATAGGTTATAGCACAGG AGAAGGAATAGGAGAAAAAACAAAGTATCTACATAG   tTTCAGTGTAAAAAATcCCAAAAAACCGgTTGAcAt   CCAA
ATAGGTTATAGCACAGG    AGGAATAGGAGAAAAAACAAAGTATcTacATAGaag TTCAGTGTAaAAaATCCCAaAAAaCCGGtTGaCgaT
ATAGGTTATAGCACAGGAAG  gGAATAGGAGAAAAAACAAAtTATctAcctagCAnn TCAGTGTAAAAAAtCCCAAAAAACCGGTTGACAATT
ATAGGTTATAGCACAGGcAGA    ATAGGAGAAAAAACAAAGTATCTACATAGAACTTTC  TGTAAAAAATCCCAAAAAACCGGTTGACAaTTGcca
```

Maq dataflow



De novo assembly

- Sequencing a new genome
- Resequencing an existing genome
- Accomodate repeats, polymorphism, sequence errors

"Reference guided" assembly

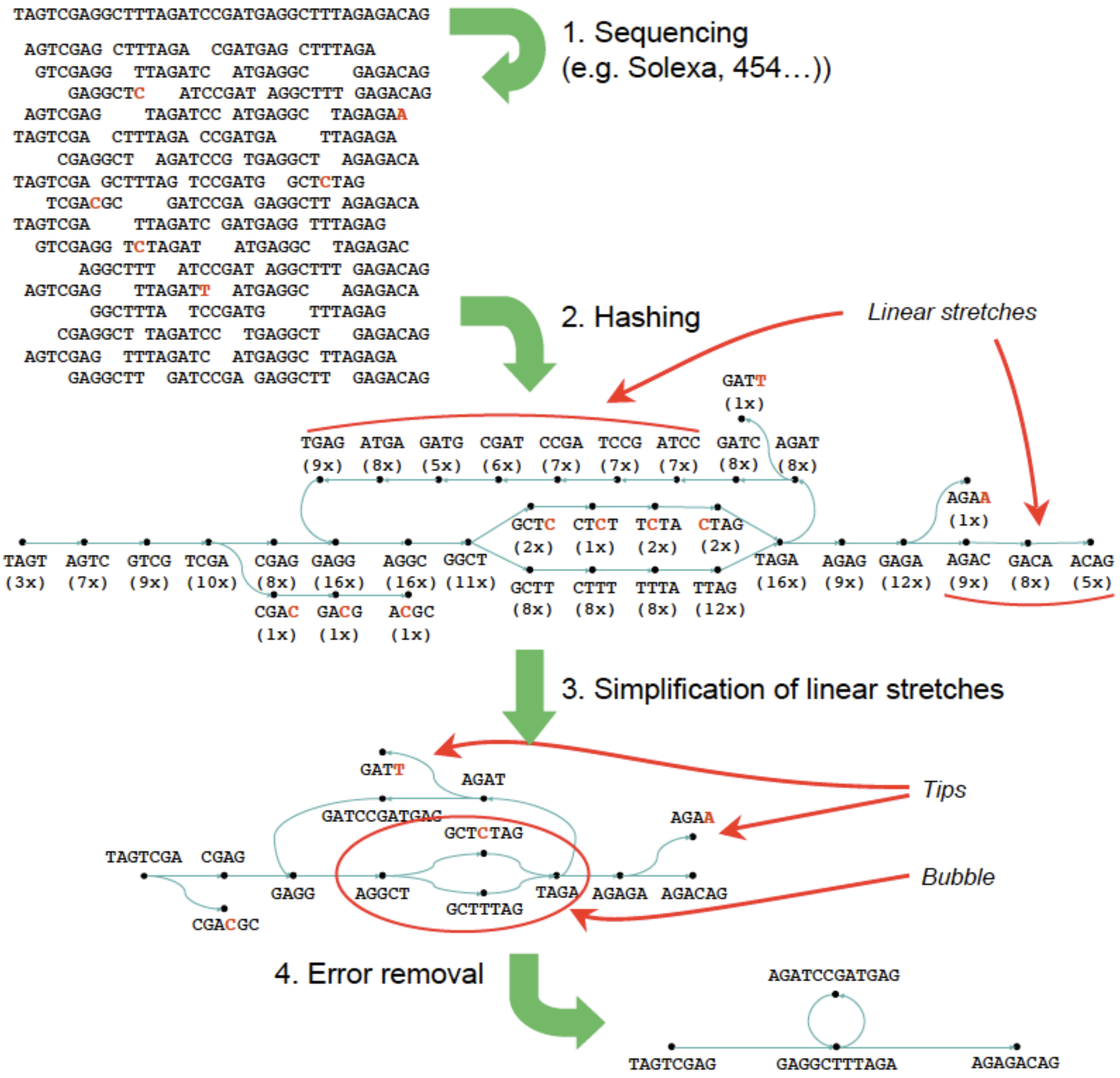
- use pairwise alignments to reference genome to guide assembly
- allows gapped alignments

mosaik
assembler

"True" *de novo* assembly

- Velvet: graph-based analysis observed *k*-mers, rather than pairwise alignment of reads

Velvet assembly process



```
TAGTCGAGGCTTTAGATCCGATGAGGCTTTAGAGACAG

 AGTCGAG CTTTAGA  CGATGAG CTTTAGA
  GTCGAGG  TTAGATC  ATGAGGC    GAGACAG
     GAGGCTC    ATCCGAT AGGCTTT GAGACAG
 AGTCGAG     TAGATCC ATGAGGC  TAGAGAA
TAGTCGA  CTTTAGA CCGATGA     TTAGAGA
    CGAGGCT  AGATCCG TGAGGCT  AGAGACA
TAGTCGA GCTTTAG TCCGATG  GCTCTAG
  TCGACGC     GATCCGA GAGGCTT AGAGACA
TAGTCGA     TTAGATC GATGAGG TTTAGAG
  GTCGAGG TCTAGAT    ATGAGGC  TAGAGAC
      AGGCTTT  ATCCGAT AGGCTTT GAGACAG
 AGTCGAG     TTAGATT  ATGAGGC   AGAGACA
       GGCTTTA  TCCGATG    TTTAGAG
    CGAGGCT TAGATCC  TGAGGCT   GAGACAG
 AGTCGAG  TTTAGATC  ATGAGGC TTAGAGA
     GAGGCTT  GATCCGA GAGGCTT  GAGACAG
```

1. Sequencing (e.g. Solexa, 454…))

2. Hashing

Linear stretches

3. Simplification of linear stretches

Tips

Bubble

4. Error removal