2. Bioinformatics of Next Generation sequencing: Sequence assembling, bacterial genome annotation

Applied Biosystems
ABI 3730XL
1 Mb / day

Roche / 454
Genome Sequencer
FLX
) Mb / run

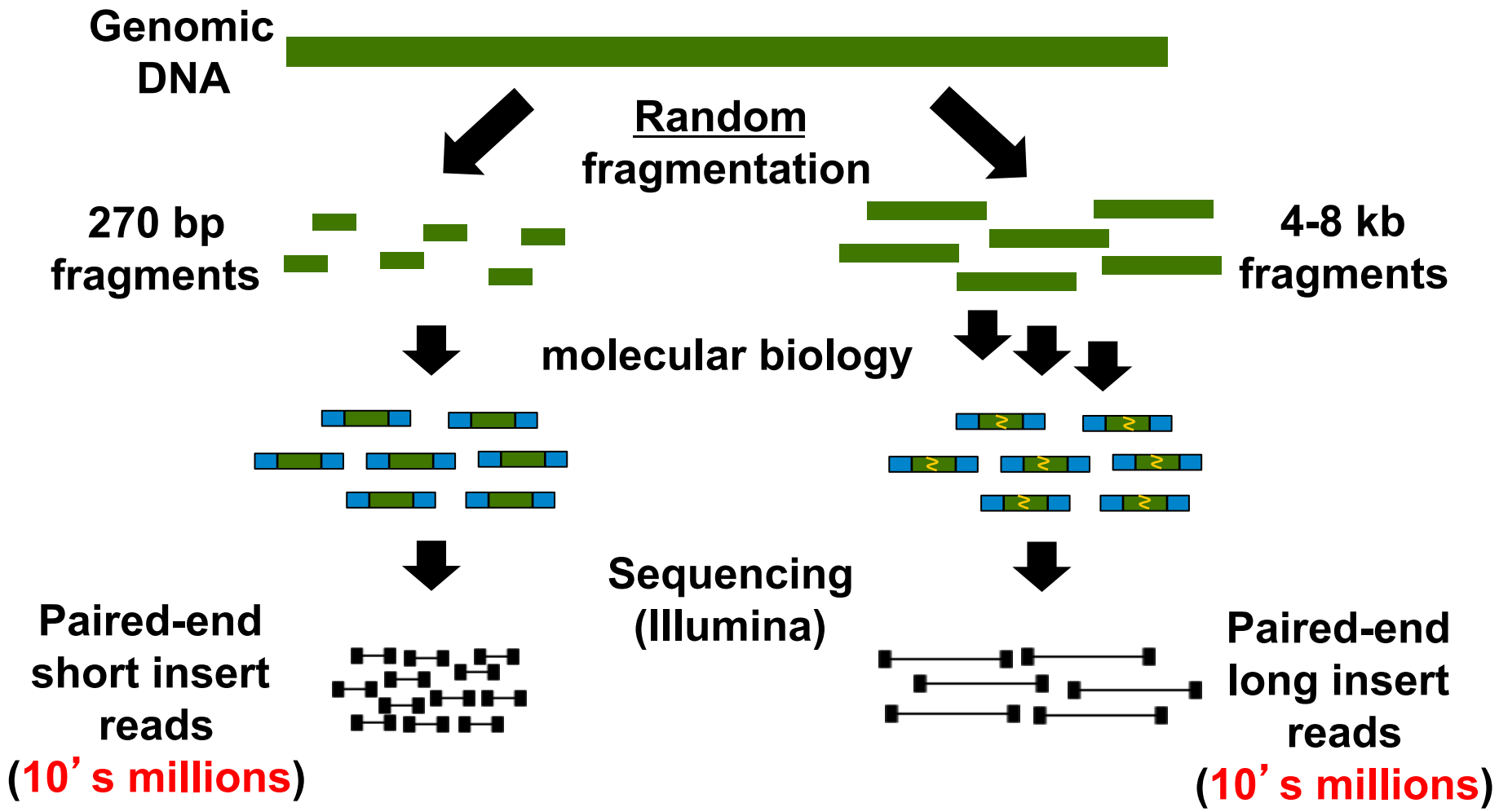
Illumina / Solexa
Genetic Analyzer
2000 Mb / run

Applied Biosystems
SOLiD
3000 Mb / run

Victor Solovyev

The lecture uses personal as well as publicly available WEB and publications materials

Short read genome sequencing



Illumina HiSeq2000

- 8 days per run
- 1 billion reads/run
- Read length of 100bp (x2)
- Generates ~ 200 Gb per run

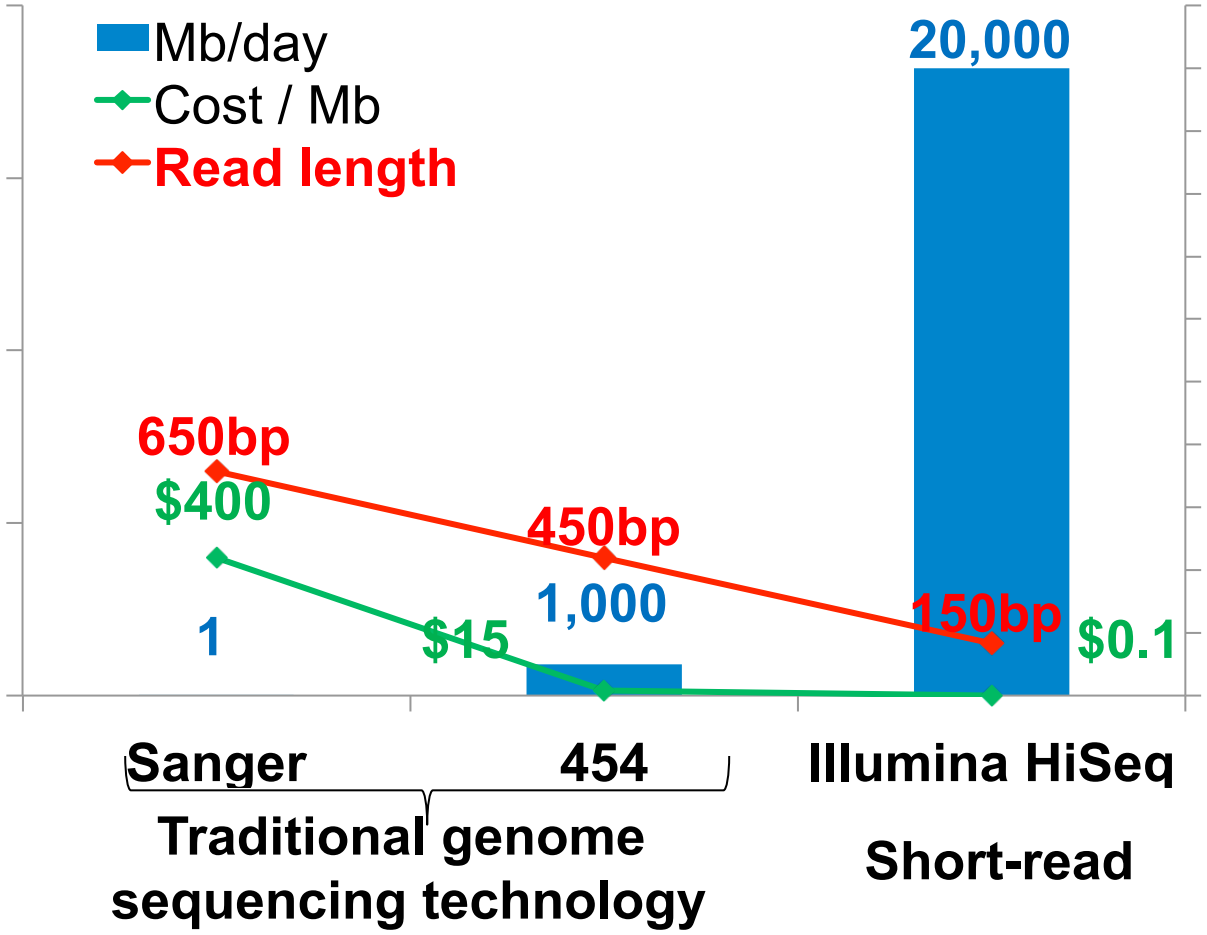
Read Length	Run Time	Output
1 × 35 bp	~1.5 days	26–35 Gb
2 × 50 bp	~4 days	75–100 Gb
2 × 100 bp	~8 days	150–200 Gb

*Sequencing output generated with a PhiX library and cluster densities between 260,000–347,000 clusters/mm^2 that pass filtering on a HiSeq 2000.

Throughput

Up to 25 Gb per day for a 2 × 100 bp run.

Why sequence genomes using short reads?



Some Applications of NGS Whole genome Sequencing

- 1000 Human Genomes Project

  An international effort to map variability in the genome

  The 1000 Genomes Project Consortium, *Nature (Oct 2010) 467: 1061–1073*

- Prostate Cancer Genomics

  M.F. Berger et al., *Nature (Feb 2011) 470: 214-220*

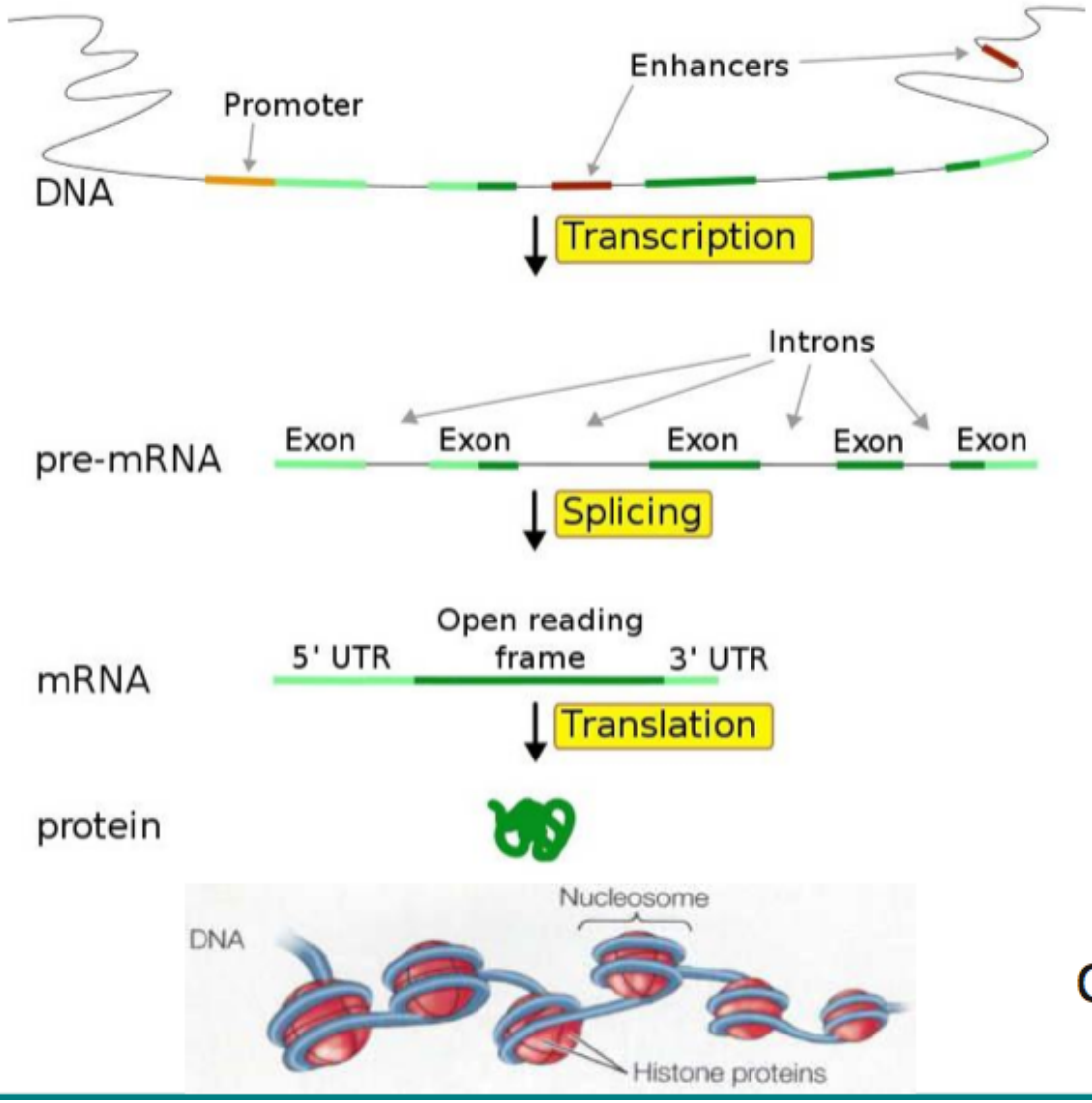
- Genome 10K Project

  – A continuation of Human (2001), Mouse (2002), Rat (2004), Chicken (2004), Dog (2005), Chimpanzee (2005), Macaque (2007), Cat (2007), Horse (2007), Elephant (2009), Turkey (2011), etc. genomes.

  – An international effort to sequence, *de novo* assemble, and annotate 10,000 vertebrate genomes; 300+ species are started in 2011.

  Genome 10K Community of Scientists, *J Heredity (Sep 2009) 100 (6): 659-674*

Next Generation Sequencing

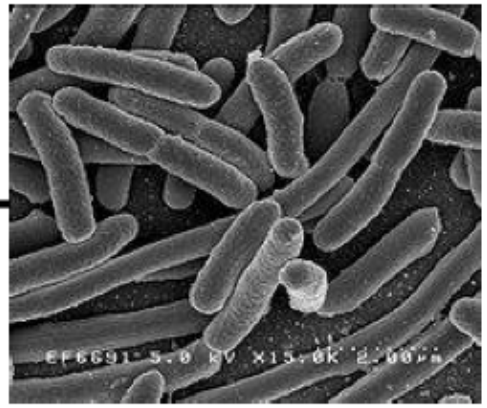


Whole Genome sequencing

RNA-Seq
Whole Transcriptome sequencing

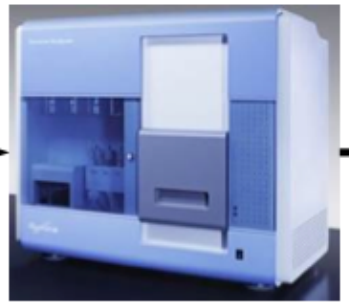
ChIP-Seq
Chromatin Immunoprecipitation with DNA sequencing

Next Generation Sequencing

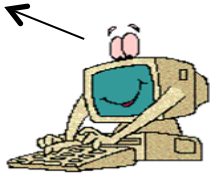


.....

Fragments



@...../1
ATGC....
+....
UVWZ...
@...../2



ACGTGACCGGTACTGGTAACGTACA
CCTACGTGACCGGTACTGGTAACGT
ACGCCTACGTGACCGGTACTGGTAA
CGTATACACGTGACCGGTACTGGTA
ACGTACACCTACGTGACCGGTACTG
GTAACGTACGCCTACGTGACCGGTA
CTGGTAACGTATACCTCT...

Sequenced genome

Assembly

ACGTGGTAA CGTATACAC TAGGCCATA
GTAATGGCG CACCCTTAG
TGGCGTATA CATA...

ACGTGGTAATGGCGTATACACCCTTAGGCCATA

De Novo Assembly paradigms

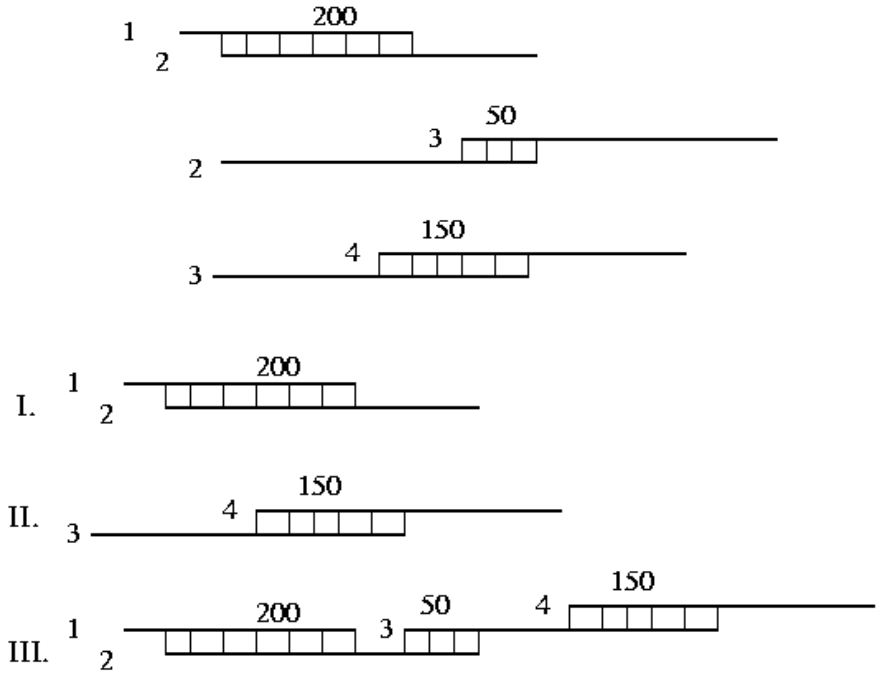
- overlap-layout-consensus methods
  - greedy (TIGR Assembler, Phrap, CAP3...)
  - overlap graph-based (Celera Assembler, Arachne)

- k-mer graph (especially useful for assembly from short reads)

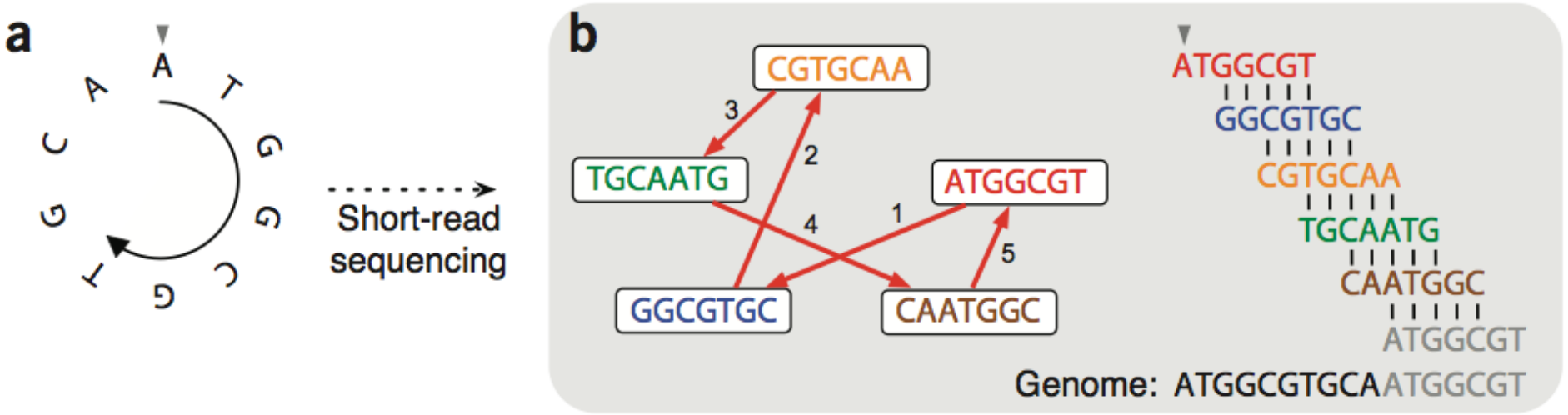
TIGR Assembler/phrap

Greedy

- Build a rough map of fragment overlaps
- Pick the largest scoring overlap
- Merge the two fragments
- Repeat until no more merges can be done



Assembling using overlap graph



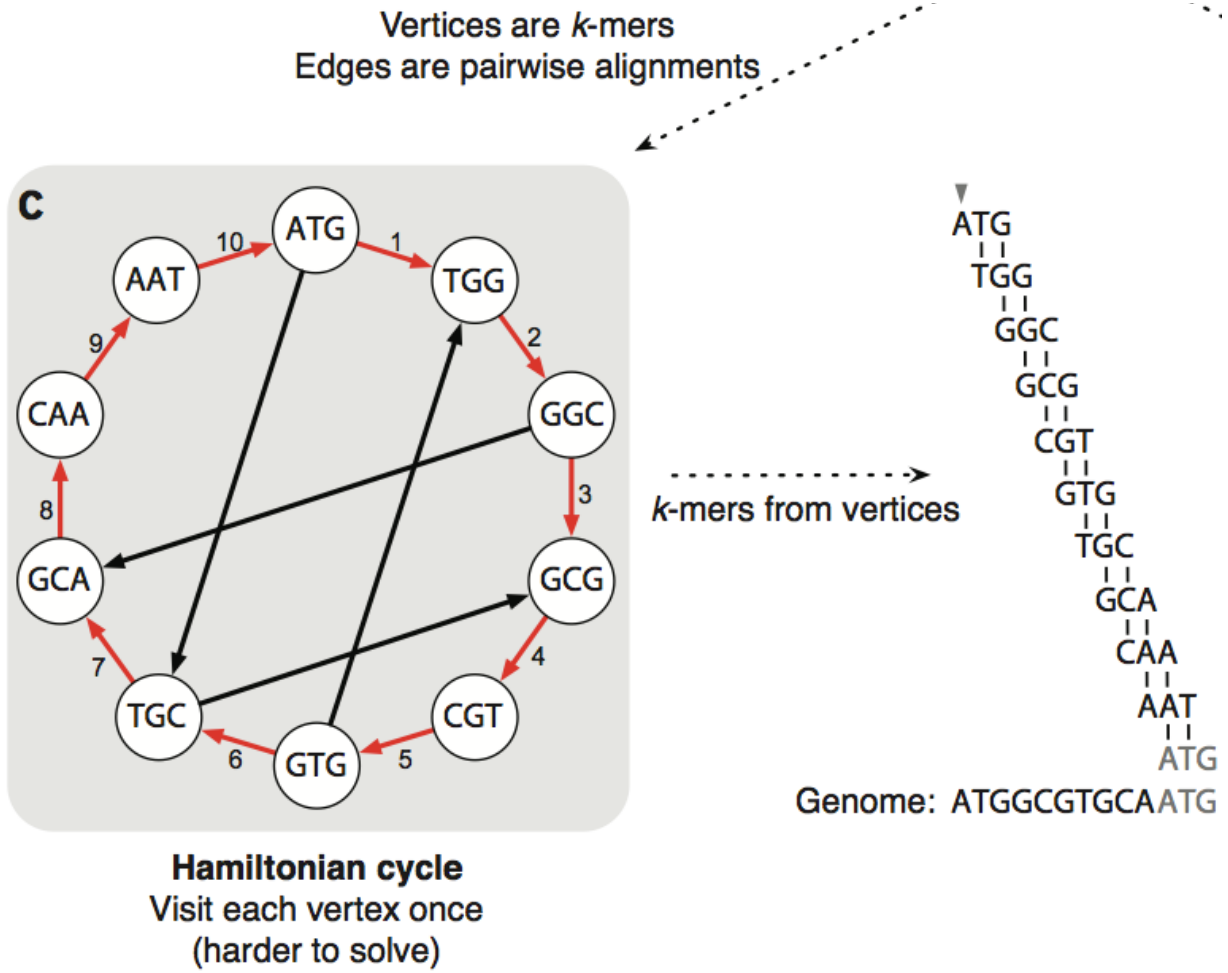
Objective: Find a Hamiltonian Path (for linear genomes) or a Hamiltonian Circuit (for circular genomes)

A billion (10^9) reads necessitate a quintillion (10^{18}) alignments.

(b) In traditional Sanger sequencing algorithms, reads were represented as nodes in a graph, and edges represented alignments between reads. Walking along a Hamiltonian cycle by following the edges in numerical order allows one to reconstruct the circular genome by combining alignments between successive reads. At the end of the cycle, the sequence wraps around to the start of the genome.

Hamiltonian Path Approach

- Hamiltonian Path is NP-hard (but good heuristics exist), and can have multiple solutions
- Dependency on detecting overlap (errors in reads, overlap length)
- Running time (all-pairs overlap calculation)
- Repeats
- Tends to produce fragmented assemblies (contigs)

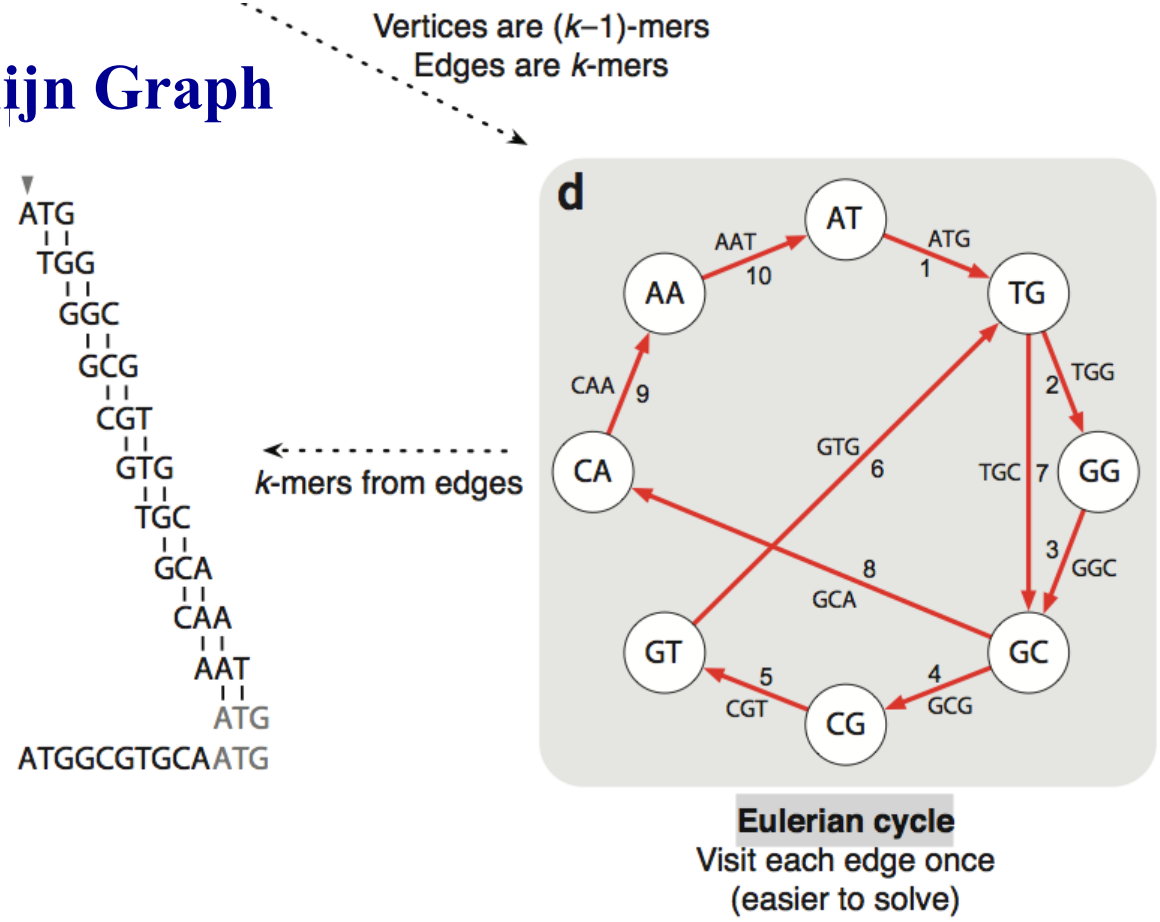


c) An alternative assembly technique first splits reads into all possible *k*-mers: with $k = 3$, ATGGCGT comprises ATG, TGG, GGC, GCG and CGT. Following a Hamiltonian cycle (indicated by red edges) allows one to reconstruct the genome by forming an alignment in which each successive *k*-mer (from successive nodes) is shifted by one position. This procedure recovers the genome but does not scale well to large graphs.

Assembling with de Bruijn Graph

Time ~number of edges

Size ~200GB for human genome



d) modern short-read assembly algorithms construct **a de Bruijn graph** by representing all *k*-mer prefixes and suffixes as nodes and then drawing edges that represent *k*-mers having a particular prefix and suffix.

For example, the *k*-mer edge ATG has prefix AT and suffix TG. Finding an Eulerian cycle allows one to reconstruct the genome by forming an alignment in which each successive *k*-mer (from successive edges) is shifted by one position.

De Bruijn graphs were first brought to bioinformatics in 1989 as a method to assemble *k*-mers generated by sequencing by hybridization; this method is very similar to the key algorithmic step of today's short-read assemblers.

Pevzner, P.A. *J. Biomol. Struct. Dyn.* **7**, 63–73 (1989).

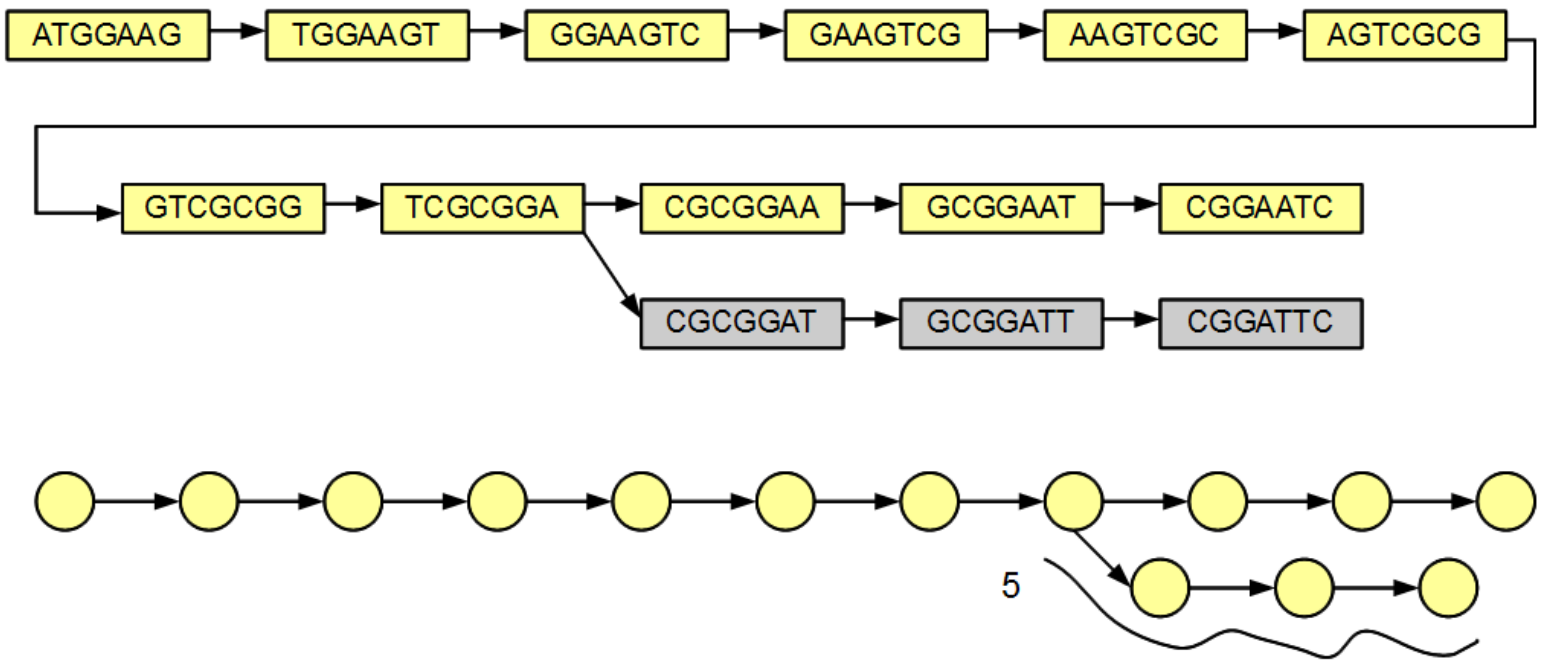
Pevzner, P.A., Tang, H., and Waterman, M.S. 2001. An Eulerian path approach to DNA fragment assembly. *Proc. Natl. Acad. Sci.* **98:** 9748–9753.

- ***Does not require all-pairs overlap calculation!***
- But: loss of information about reads can lead to "chimeric" contigs, and incorrect assemblies
- Also produces fragmented assemblies (even shorter contigs)

Sequencing Errors Generate Lightly Travelled Divergent Paths in *de Bruijn* graphs

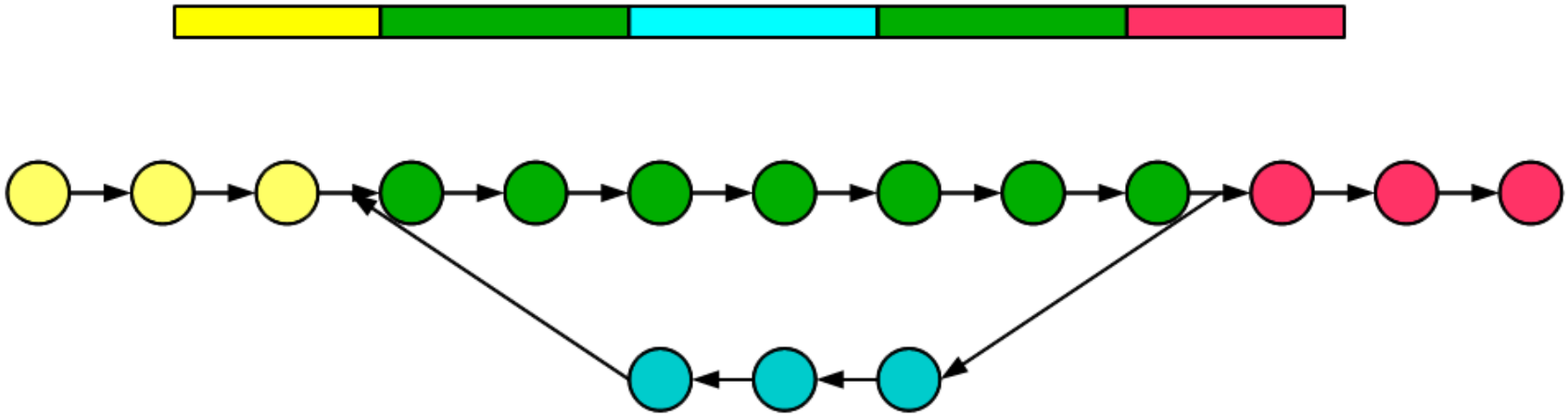
sequence **ATGGAAGTCGCGGAATC**

Short read 5* - TCGCGGATTC

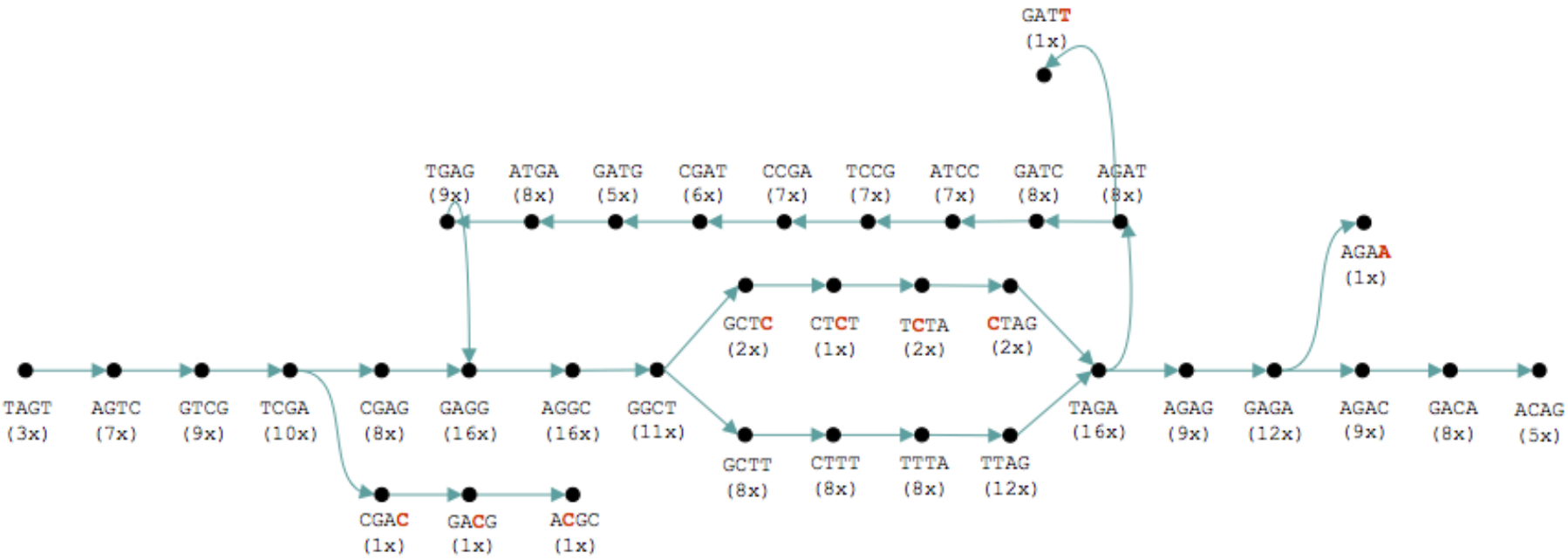


Sequence assembly algorithms can prune such lightly travelled paths but reconstruct the genome from heavily traversed paths.

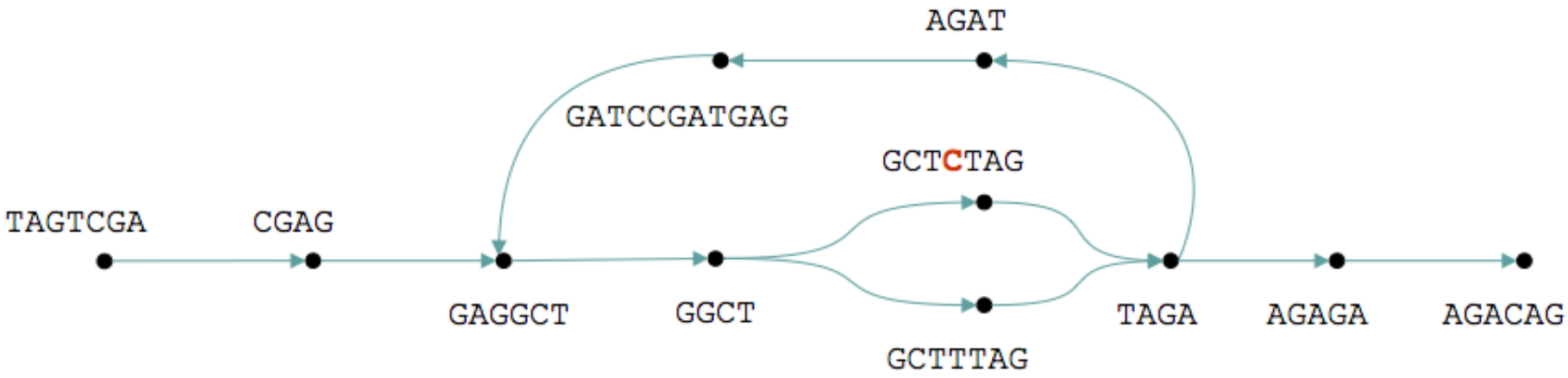
Repeat Content in Targets Add Graph Cycles

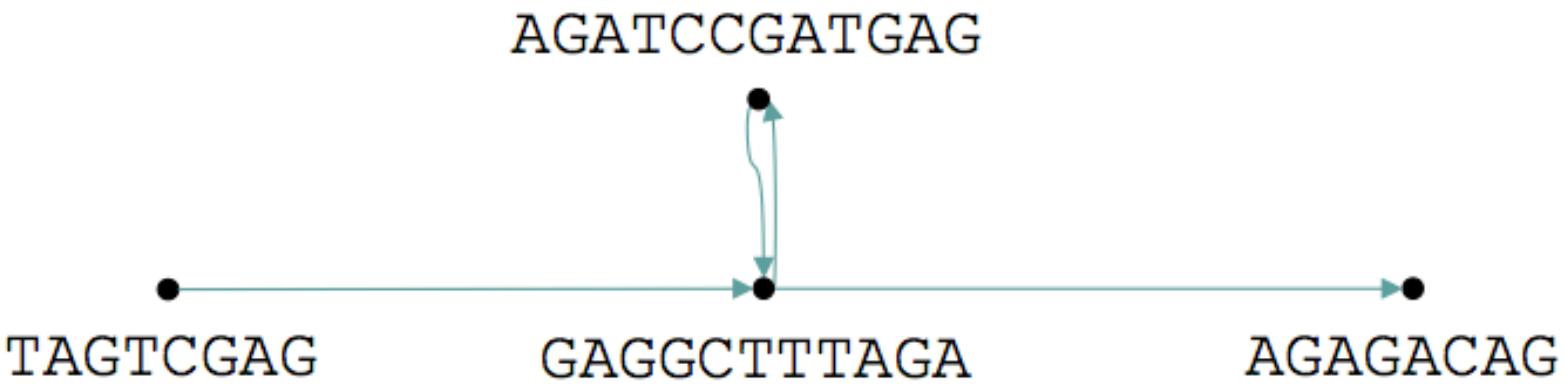


De Bruijn graph example



De Bruijn graph after simplification

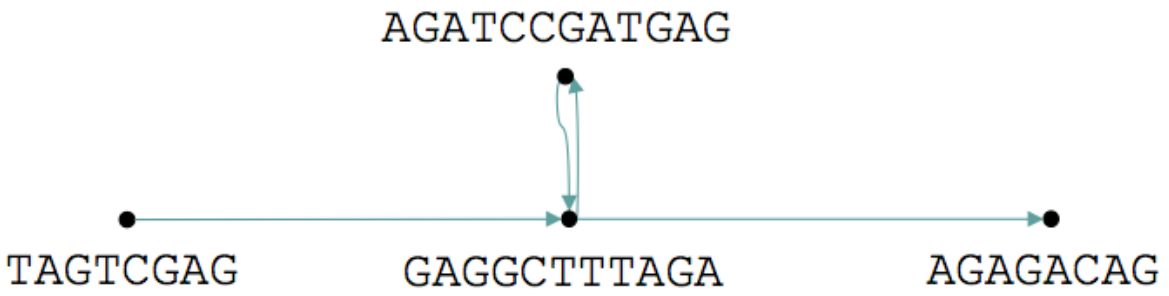




Generating the sequence:

TAGTCGAGGCTTTAGATCCGATGAGGCTTTAGAGACAG

Final simplification…



One possible walk through the graph ...

TAGTCGAG
GAGGCTTTAGA
AGATCCGATGAG
GAGGCTTTAGA
AGAGACAG

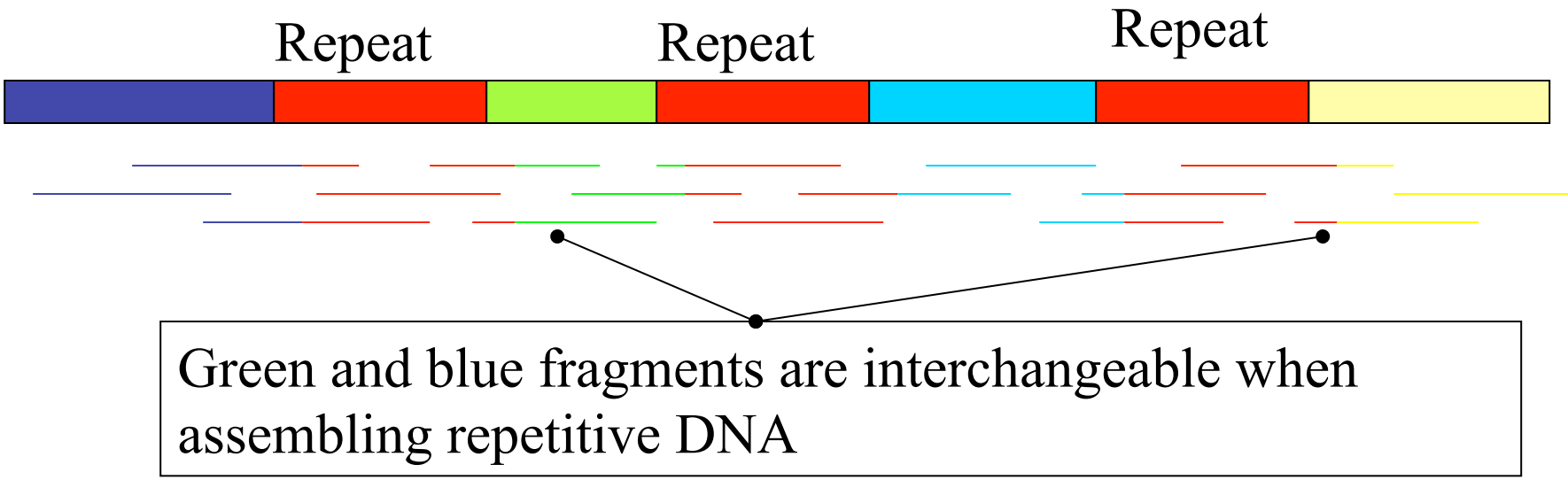
No matter what

- Because of
  - Errors in reads
  - Repeats
  - Insufficient coverage

  the overlap graphs and de Bruijn graphs generally don't have Hamiltonian paths/circuits or Eulerian paths/ circuits
- This means the first step doesn't completely assemble the genome

Challenges in Fragment Assembly

- Repeats: A **major** problem for fragment assembly
- > 50% of human genome are repeats:
  - over 1 million *Alu* repeats (about 300 bp)
  - about 200,000 LINE repeats (1000 bp and longer)

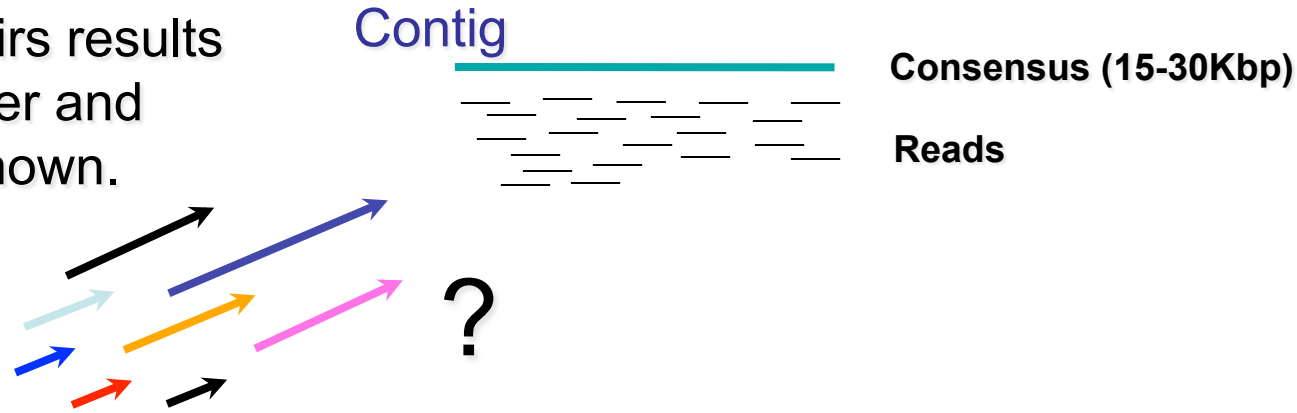


Reads, Contigs, and Scaffolds

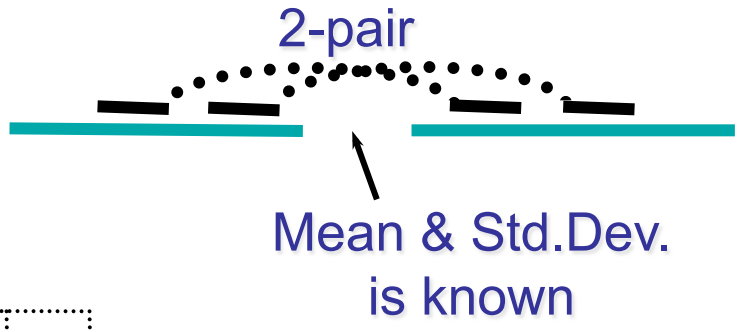
- Reads are what you start with (35bp-800bp)
- Fragmented assemblies produce contigs that can be kilobases in length
- Putting contigs together into scaffolds is the next step

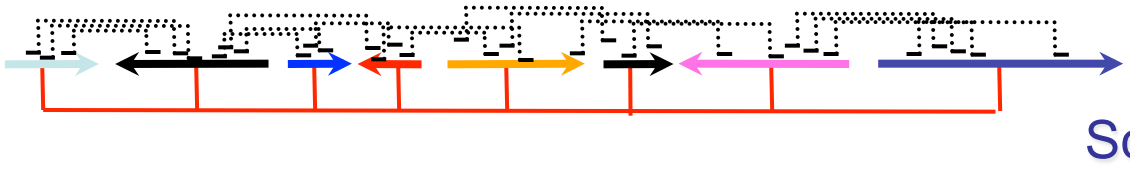
Mate Pairs Give Order & Orientation

Assembly without pairs results in contigs whose order and orientation are not known.



Pairs, especially groups of corroborating ones, link the contigs into scaffolds where the size of gaps is well characterized.





Jute Genome Project

A consortium of researchers from University of Dhaka**, Bangladesh** Jute Research Institute and private software company DataSoft Systems Bangladesh Limited in close collaboration with Centre for Chemical Biology at University of Science Malaysia and University of Hawaii, USA have **successfully decoded the draft genome of jute.** The project was funded by the Ministry of Agriculture, Government of Bangladesh.

The public announcement of this major discovery and the unveiling of the high through-put technology used in this Discovery was made on 24th of June, 2010

1.2 GB genome $2 million dollars
Fgenesh has been applied to annotate the genome

Jute genome sequences ~ 1.1 GB

number of sequences = 1240856
minimal sequence length = 200.00
maximal sequence length = 1188242.00
average sequence length = 873.58

range	number of seq.	%
0	0	0.0
100	0	0.0
200	801229	64.6
500	271108	21.8
1000	120844	9.7
3000	21873	1.8
5000	15356	1.2
10000	4732	0.4
15000	2127	0.2
20000	3587	0.3
0	1240856	100.0
100	1240856	100.0
200	1240856	100.0
500	439627	35.4
1000	168519	13.6
3000	47675	3.8
5000	25802	2.1
10000	10446	0.8
15000	5714	0.5
20000	3587	0.3

Genome Assembly Workshop, Genome 10K, March 2011

The challenge: Assemble genome of species 'A'
112 MB diploid synthetic genome
2 paired-read + 2 mate pairs libraries

17 teams from

Assemblathon 2: evaluating *de novo* methods of genome assembly in three vertebrate species (2013) 21 teams

"From over 100 different metrics, we chose ten key measures by which to assess the overall quality of the assemblies"

Table 1 Assemblathon 2 participating team details

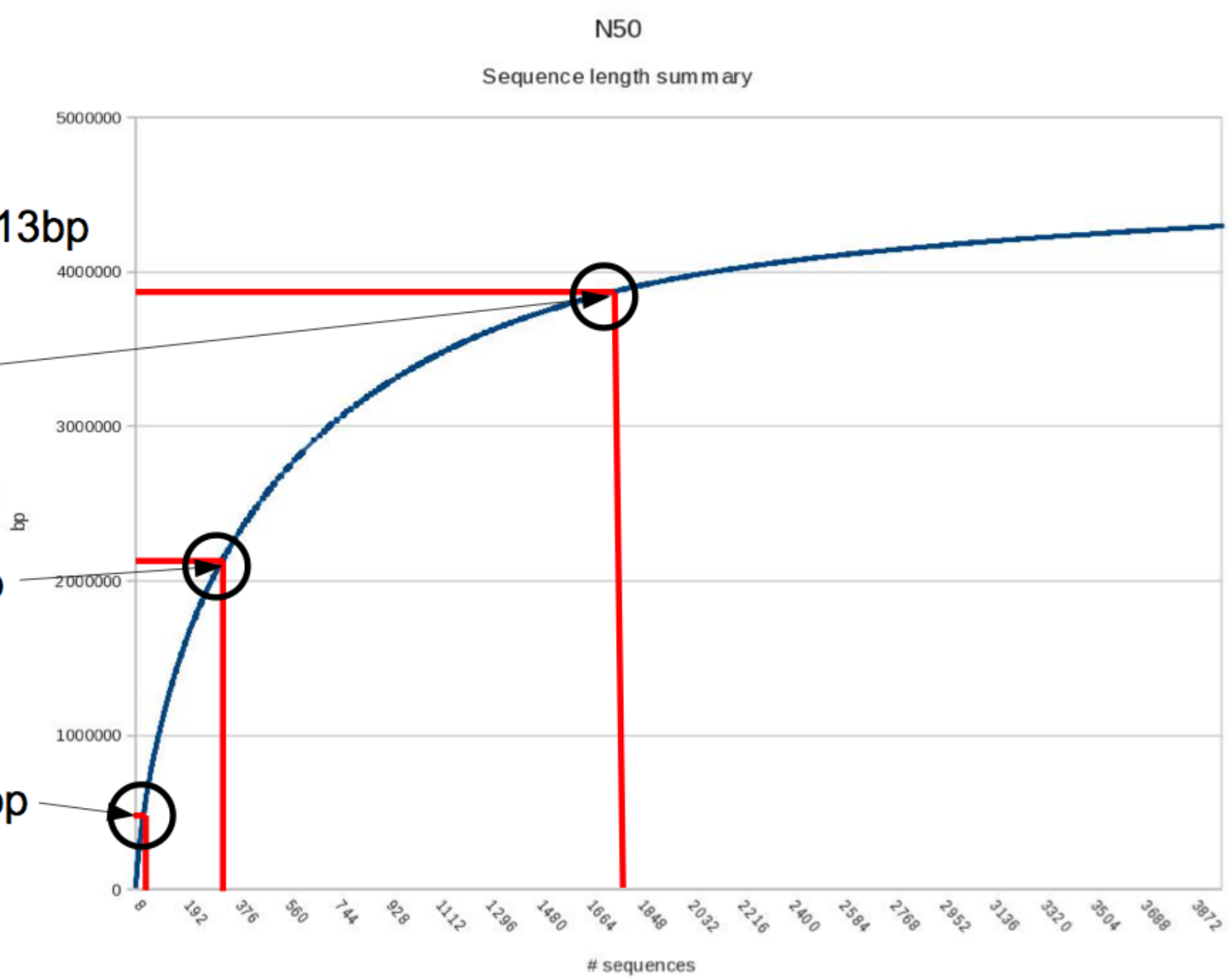
Team name	Team identifier	Number of assemblies submitted			Sequence data used for bird assembly	Institutional affiliations	Principal assembly software used
		Bird	Fish	Snake			
ABL	ABL	1	0	0	4 + I	Wayne State University	HyDA
ABySS	ABYSS	0	1	1		Genome Sciences Centre, British Columbia Cancer Agency	ABySS and Anchor
Allpaths	ALLP	1	1	0	I	Broad Institute	ALLPATHS-LG
BCM-HGSC	BCM	2	1	1	4 + I + P[1]	Baylor College of Medicine Human Genome Sequencing Center	SeqPrep, KmerFreq, Quake, BWA, Newbler, ALLPATHS-LG, Atlas-Link, Atlas-GapFill, Phrap, CrossMatch, Velvet, BLAST, and BLASR
CBCB	CBCB	1	0	0	4 + I + P	University of Maryland, National Biodefense Analysis and Countermeasures Center	Celera assembler and PacBio Corrected Reads (PBcR)
CoBiG[2]	COBIG	1	0	0	4	University of Lisbon	4Pipe4 pipeline, Seqclean, Mira, Bambus2
CRACS	CRACS	0	0	1		Institute for Systems and Computer Engineering of Porto TEC, European Bioinformatics Institute	ABySS, SSPACE, Bowtie, and FASTX
CSHL	CSHL	0	3	0		Cold Spring Harbor Laboratory, Yale University, University of Notre Dame	Metassembler, ALLPATHS, SOAPdenovo
CTD	CTD	0	3	0		National Research University of Information Technologies, Mechanics, and Optics	Unspecified
Curtain	CURT	0	0	1		European Bioinformatics Institute	SOAPdenovo, fastx_toolkit, bwa, samtools, velvet, and curtain
GAM	GAM	0	0	1		Institute of Applied Genomics, University of Udine, KTH Royal Institute of Technology	GAM, CLC and ABySS
IOBUGA	IOB	0	2	0		University of Georgia, Institute of Aging Research	ALLPATHS-LG and SOAPdenovo

MLK Group	MLK	1	0	0	I	UC Berkeley	ABySS
Meraculous	MERAC	1	1	1	I	DOE Joint Genome Institute, UC Berkeley	meraculous
Newbler-454	NEWB	1	0	0	4	454 Life Sciences	Newbler
Phusion	PHUS	1	0	1	I	Wellcome Trust Sanger Institute	Phusion2, SOAPdenovo, SSPACE
PRICE	PRICE	0	0	1		UC San Francisco	PRICE
Ray	RAY	1	1	1	I	CHUQ Research Center, Laval University	Ray
SGA	SGA	1	1	1	I	Wellcome Trust Sanger Institute	SGA
SOAPdenovo	SOAP	3	1	1	I^2	BGI-Shenzhen, HKU-BGI	SOAPdenovo
Symbiose	SYMB	0	1	1		ENS Cachan/IRISA, INRIA, CNRS/ Symbiose	Monument, SSPACE, SuperScaffolder, and GapCloser

N50 is the length of the smallest contig when we take the fewest (largest) contigs, whose combined length represents at least 50% of the assembly

N50

- Total
  - 4,295,113bp
- N90
  - 439bp
- N50
  - 3,119bp
- N10
  - 13,519bp



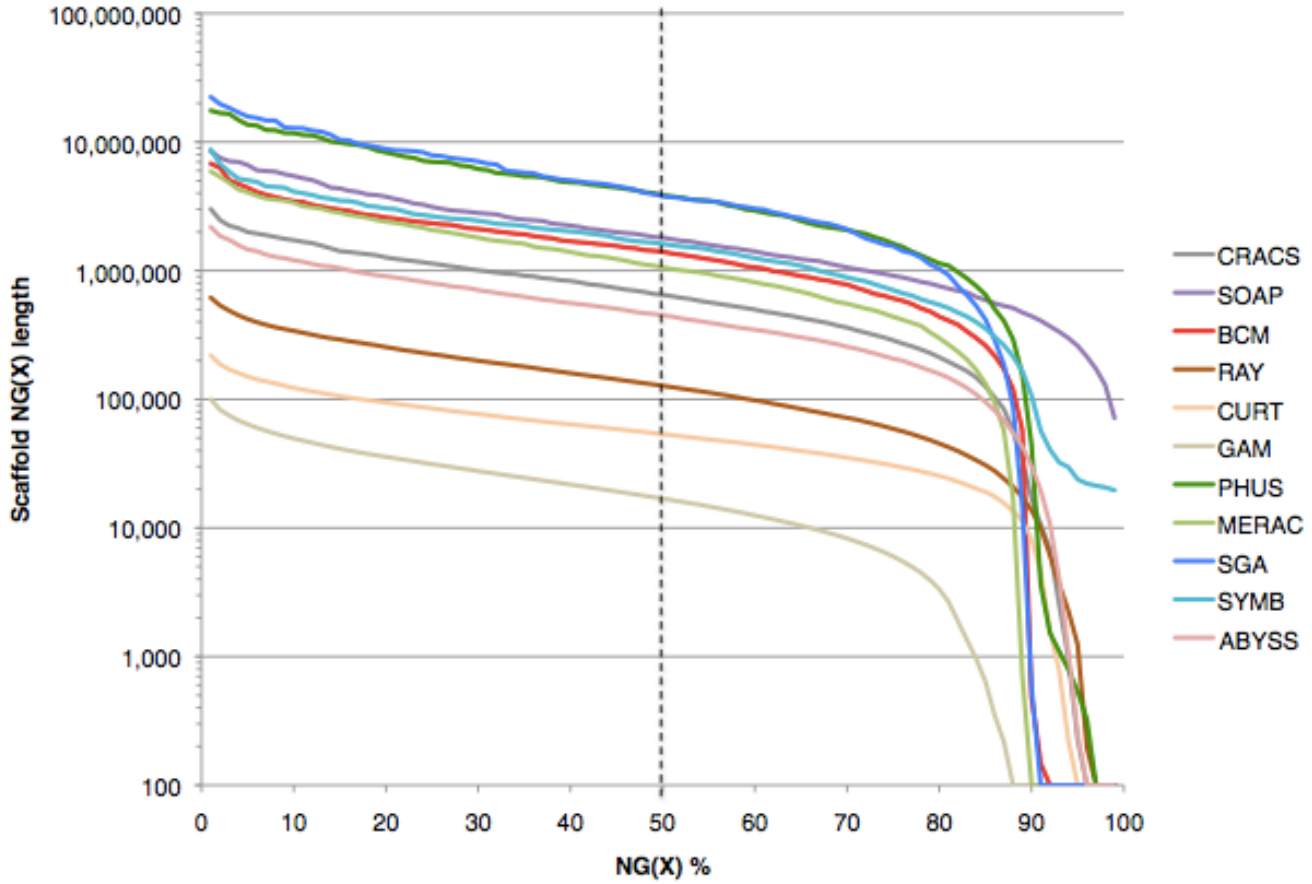
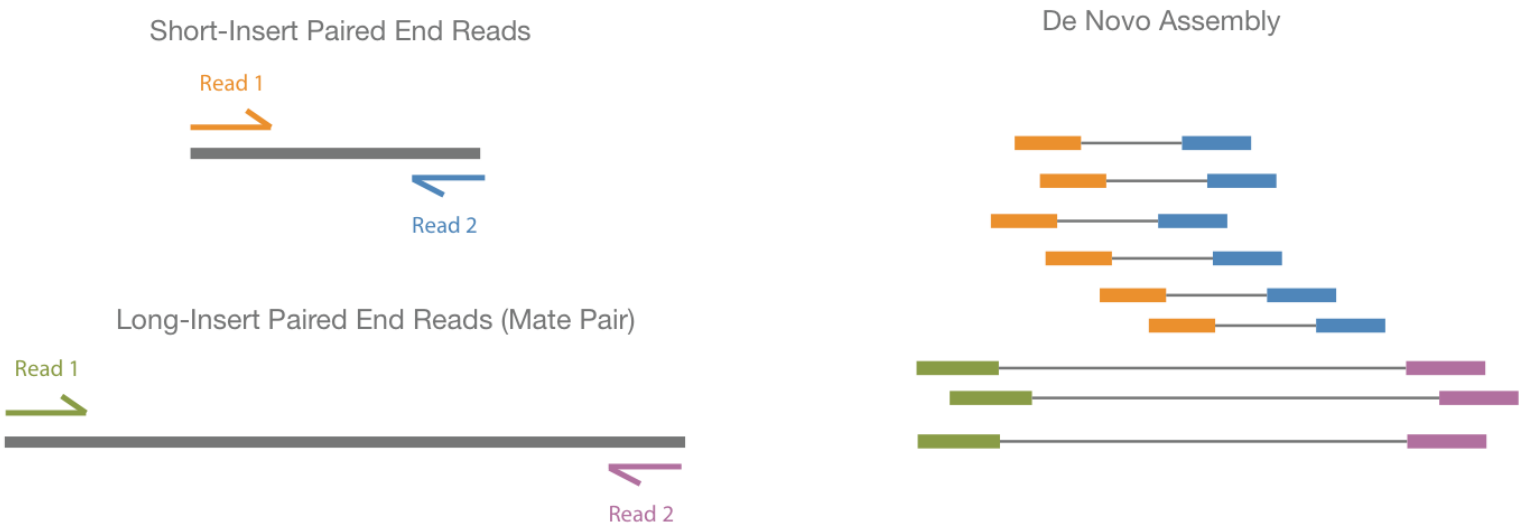
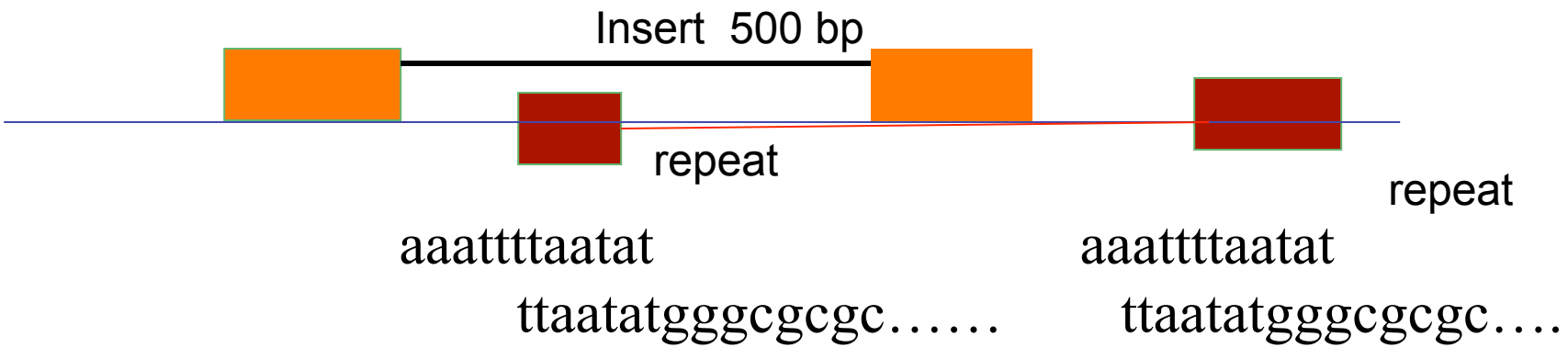


Figure 3 NG graph showing an overview of snake assembly scaffold lengths. The NG scaffold length (see text) is calculated at integer thresholds (1% to 100%) and the scaffold length (in bp) for that particular threshold is shown on the y-axis. The dotted vertical line indicates the NG50 scaffold length: if all scaffold lengths are summed from longest to the shortest, this is the length at which the sum length accounts for 50% of the estimated genome size. Y-axis is plotted on a log scale. Snake estimated genome size = ~1.0 Gbp.

Figure 5. *De Novo* Assembly with Mate Pairs



Using a combination of short and long insert sizes with paired-end sequencing results in maximal coverage of the genome for de novo assembly. Because larger inserts can pair reads across greater distances, they provide a better ability to read through highly repetitive sequences and regions where large structural rearrangements have occurred. Shorter inserts sequenced at higher depths can fill in gaps missed by larger inserts sequenced at lower depths. Thus a diverse library of short and long inserts results in better de novo assembly, leading to fewer gaps, larger contigs, and greater accuracy of the final consensus sequence.



Adapter Trimming

Universal Adapter

DNA Fragment of Interest

Indexed Adapter

6 Base Index Region

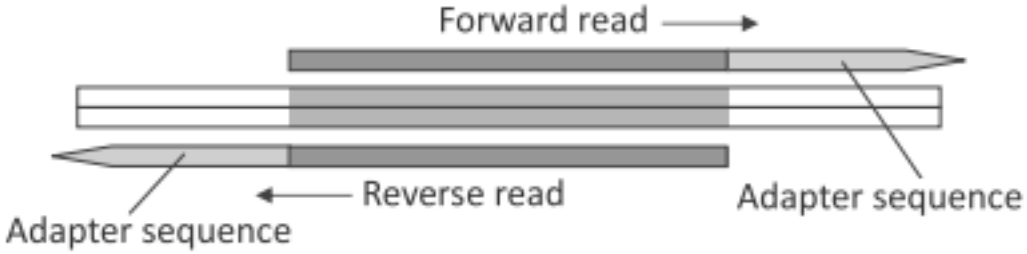


Figure 5 Layout of paired-end reads that have adapter contaminants.

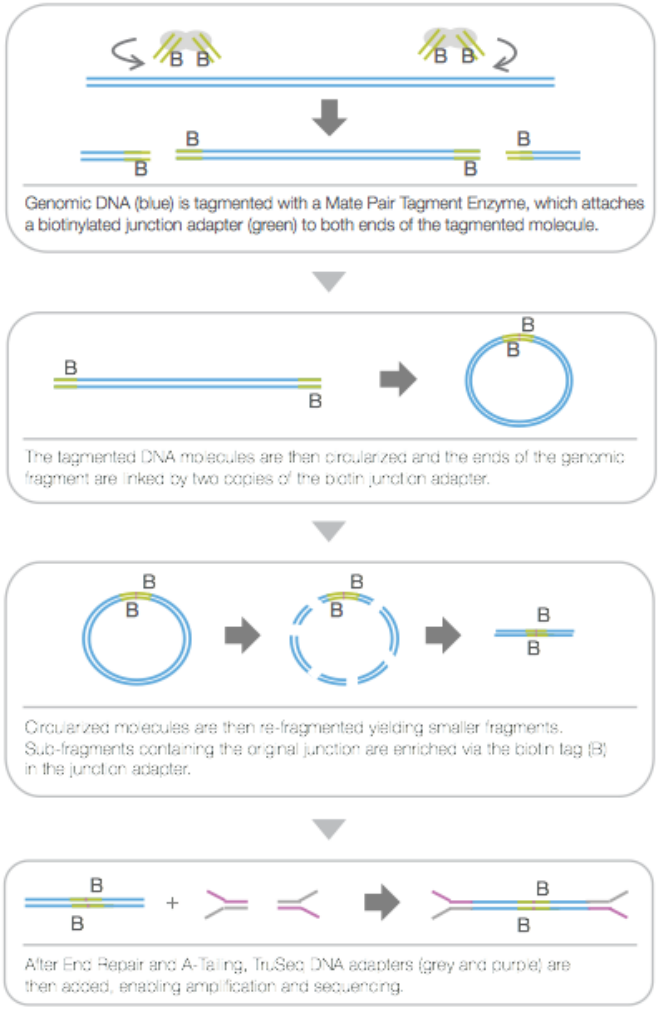
Jiang et al. BMC Bioinformatics 2014, 15:182

Table 2 Performance of adapter trimmers on 2Gbp simulated data

Method (Single End/Paired End)		Speed (Mbp/s)	Memory (Mb)	PPV (%)	Sen. (%)	Spec. (%)	mCC
FastX	SE	0.92	13.8	68.90	90.84	77.97	0.6683
SeqTrim	SE	0.03	115.7	67.07	85.27	81.24	0.6618
TagCleaner	SE	0.54	37.6	100.0	45.50	100.0	0.5898
EA-Tools	SE	12.04	17.7	59.24	99.72	61.32	0.6010
	PE	11.54	30.0	59.16	99.43	61.36	0.5983
Cutadapt	SE	4.36	34.5	94.55	96.27	96.93	0.9286
	PE	3.44	42.8	94.55	96.00	96.93	0.9266
TrimGalore	SE	3.81	19.4	59.24	99.72	61.32	0.6010
	PE	3.26	19.6	59.16	99.44	61.36	0.5984
SeqPrep	PE	0.64	22.0	99.84	99.82	99.92	0.9975
Btrim	SE	23.63	11.2	99.96	53.44	100.0	0.6503
	PE	5.79	15.3	99.89	53.30	100.0	0.6490
Scythe	SE	3.15	11.2	99.56	90.86	99.92	0.9283
Flexbar	SE	2.82	9.5	57.90	99.12	59.48	0.5814
	PE	2.70	9.7	57.77	99.09	59.29	0.5795
Trimmomatic	SE	16.73	2593.0	99.99	72.31	100.0	0.7907
	PE	16.40	2292.0	100.0	71.54	100.0	0.7850
AdapterRemoval	SE	1.67	6.3	75.09	97.74	81.89	0.7675
	PE	0.73	8.3	99.93	94.47	99.97	0.9566
AlienTrimmer	SE	1.64	2319.9	85.62	57.11	99.96	0.6769
	PE	1.61	2248.9	83.71	55.67	99.95	0.6659
Skewer	SE	8.79	13.6	94.56	96.32	96.93	0.9291
	PE	8.88	22.2	100.0	99.86	100.0	0.9989

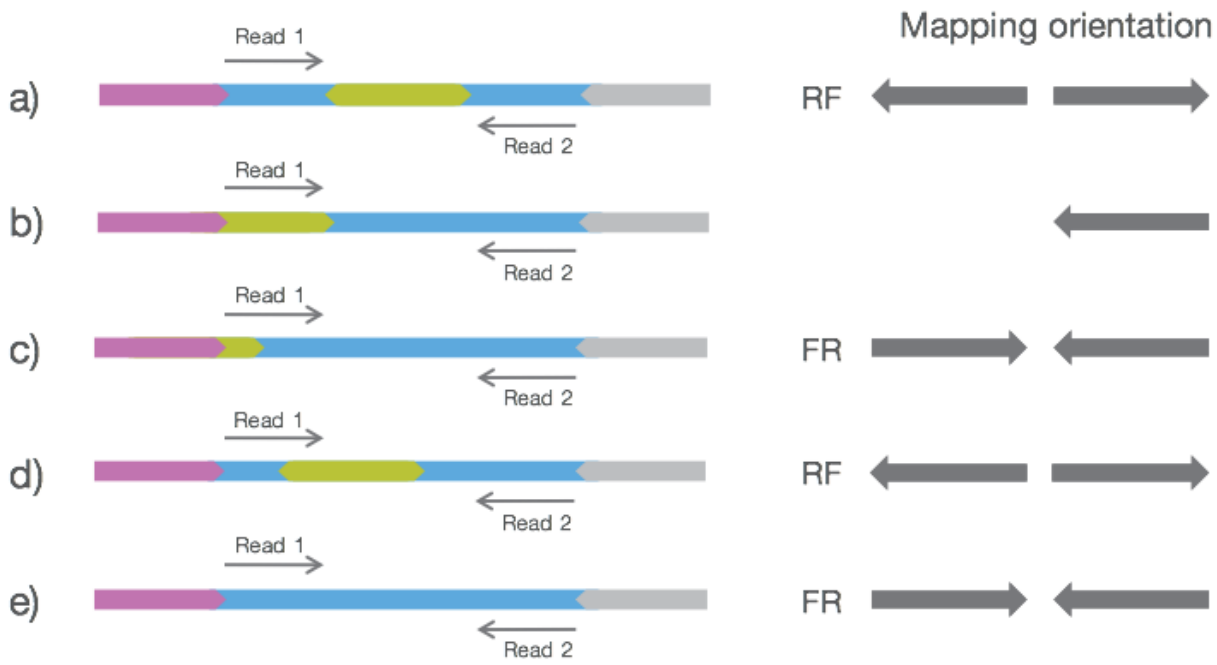
Methods that process only single-end (SE) or paired-end (PE) reads are indicated.

Figure 1: Nextera Mate Pair Workflow

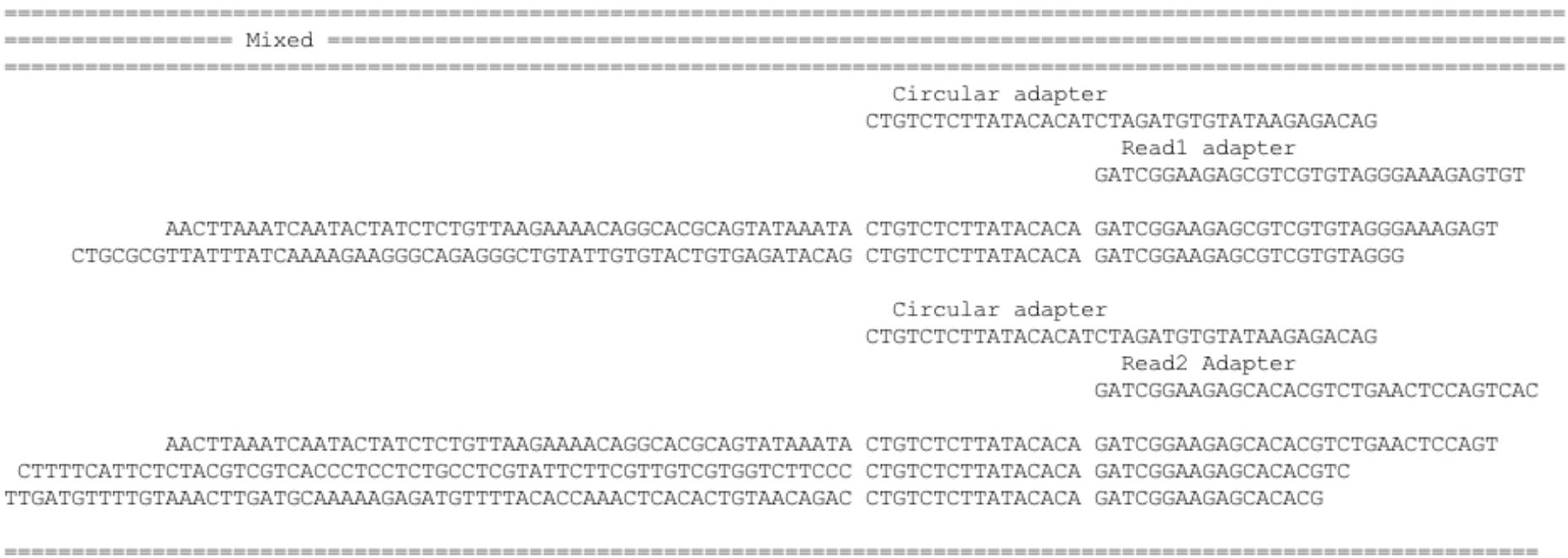


The Nextera Mate Pair Tagment Enzyme simultaneously fragments and attaches an adapter (green) to the ends of dsDNA (blue). These fragments are circularized and subsequently re-fragmented, yielding smaller fragments suitable for clustering. Subfragments containing the original circularization junction are enriched via a biotin pull-down tag (B) in the original adapter. Sequencing adapters (grey and purple) are then added to the enriched set, enabling amplification and sequencing on an Illumina flow cell.

Figure 2: Junction Adapter Positions and Mapping Orientations



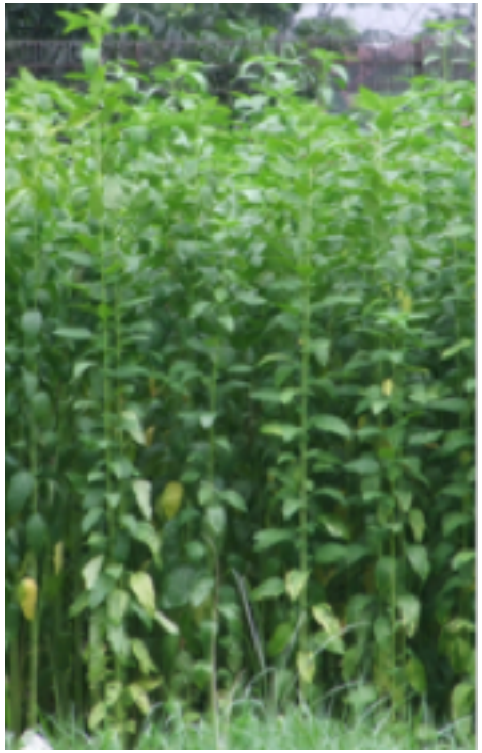
Composition of example library templates from a mate pair experiment. For each example (a–e), the position of the junction adapter sequence is shown in green and the mapping orientation (either FR or RF, 'forward-reverse' and 'reverse-forward', respectively) of the resulting read pairs is shown to the right. Sections of genomic DNA sequence are shown in blue and the TruSeq adapter sequences are shown in purple and grey.
Amplification/sequencing primer adapters are shown in grey and purple.


Mixed
Circular adapter
CTGTCTCTTATACACATCTAGATGTGTATAAGAGACAG
Read1 adapter
GATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
AACTTAAATCAATACTATCTCTGTTAAGAAAACAGGCACGCAGTATAAATA CTGTCTCTTATACACA GATCGGAAGAGCGTCGTGTAGGGAAAGAGT
CTGCGCGTTATTTATCAAAAGAAGGGCAGAGGGCTGTATTGTGTACTGTGAGATACAG CTGTCTCTTATACACA GATCGGAAGAGCGTCGTGTAGGG
Circular adapter
CTGTCTCTTATACACATCTAGATGTGTATAAGAGACAG
Read2 Adapter
GATCGGAAGAGCACACGTCTGAACTCCAGTCAC
AACTTAAATCAATACTATCTCTGTTAAGAAAACAGGCACGCAGTATAAATA CTGTCTCTTATACACA GATCGGAAGAGCACACGTCTGAACTCCAGT
CTTTTCATTCTCTACGTCGTCACCCTCCTCTGCCTCGTATTCTTCGTTGTCGTGGTCTTCCC CTGTCTCTTATACACA GATCGGAAGAGCACACGTC
TTGATGTTTTGTAAACTTGATGCAAAAAGAGATGTTTTACACCAAACTCACACTGTAACAGAC CTGTCTCTTATACACA GATCGGAAGAGCACACG

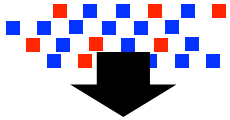
Jute genome sequences ~ 1.1 GB

number of sequences = 1240856
minimal sequence length = 200.00
maximal sequence length = 1188242.00
average sequence length = 873.58

range	number of seq.	%
0	0	0.0
100	0	0.0
200	801229	64.6
500	271108	21.8
1000	120844	9.7
3000	21873	1.8
5000	15356	1.2
10000	4732	0.4
15000	2127	0.2
20000	3587	0.3
0	1240856	100.0
100	1240856	100.0
200	1240856	100.0
500	439627	35.4
1000	168519	13.6
3000	47675	3.8
5000	25802	2.1
10000	10446	0.8
15000	5714	0.5
20000	3587	0.3



Oligozip Assembly outline



PE and MP Reads

Removing poly-N, low quality reads, or read parts

Adapter trimming

Error correction

AGCCTAGACCTACAGGATGCGCGACACTT
AGCCTAGACCTACAGGATGCGAGACACGT
AGCCTAGACCTACAGGATGCGCGACACGT

Find overlaps . . .AGCCTAGACCTACAGGATGCGCGACACGT
GGATGCGCGACACGTCGCATATCCGGT. . .

Build Contigs

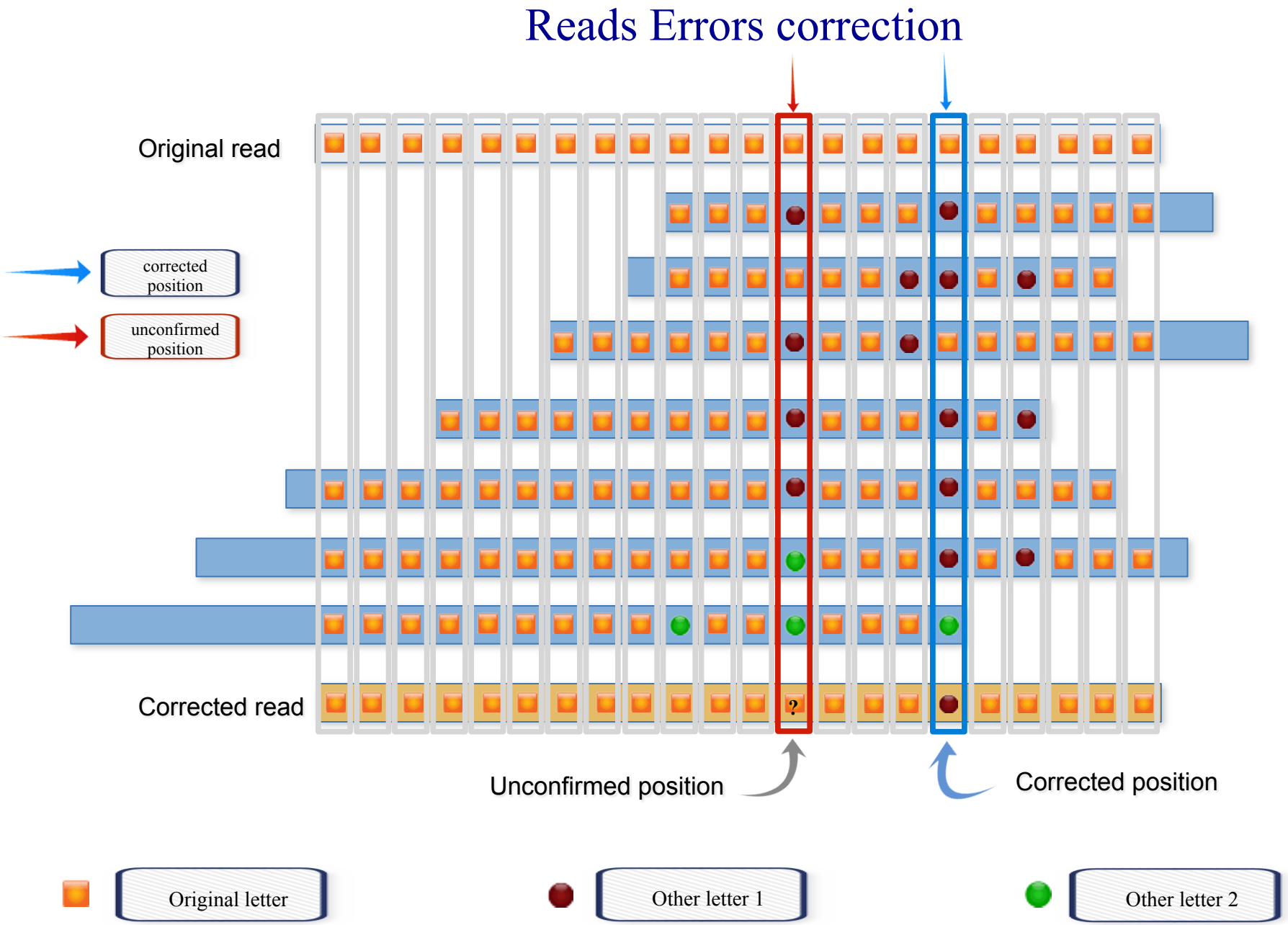
Assemble overlaps into contigs

Construct contig connections graph

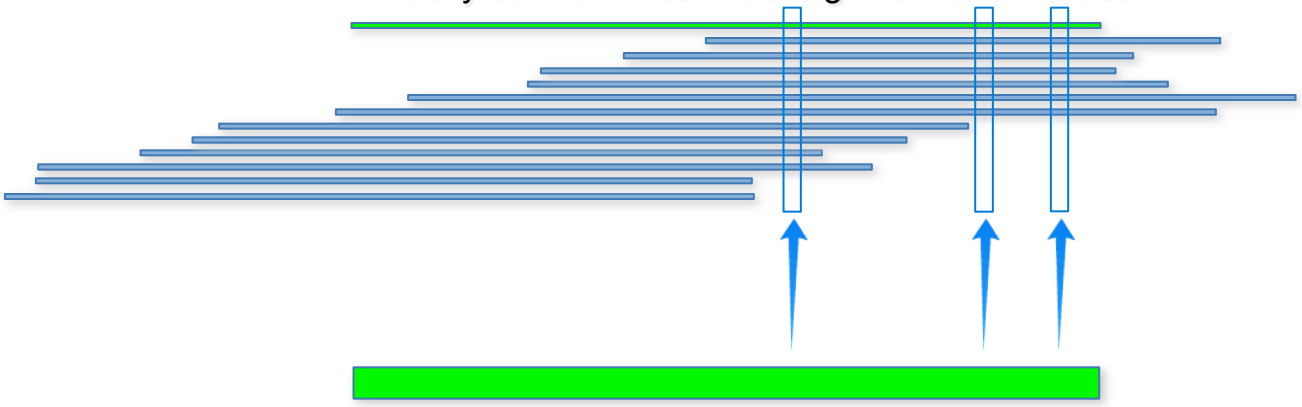
Build Scaffolds

Assemble contigs into scaffolds

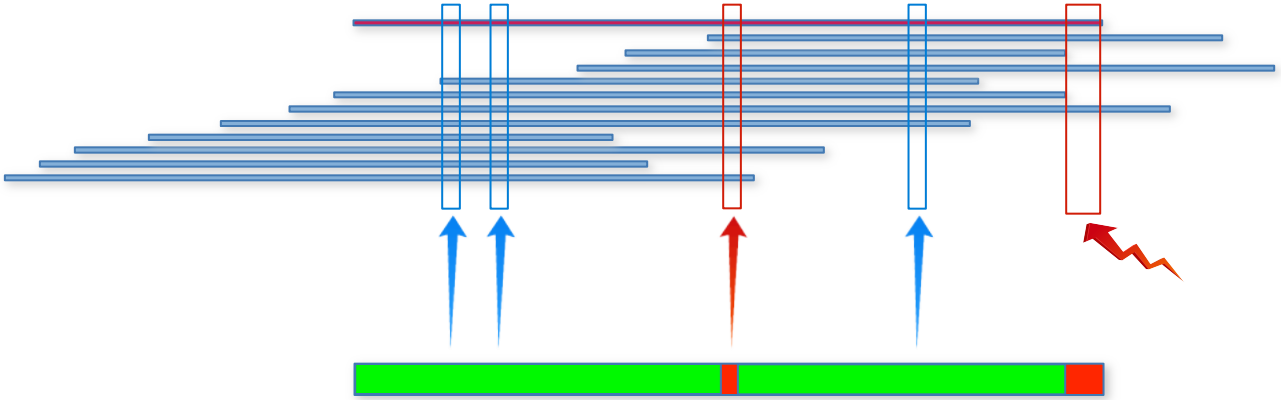
Assembly algorithms


Reads Errors correction
Original read
corrected position
unconfirmed position
Corrected read
?
Unconfirmed position
Corrected position
Original letter
Other letter 1
Other letter 2

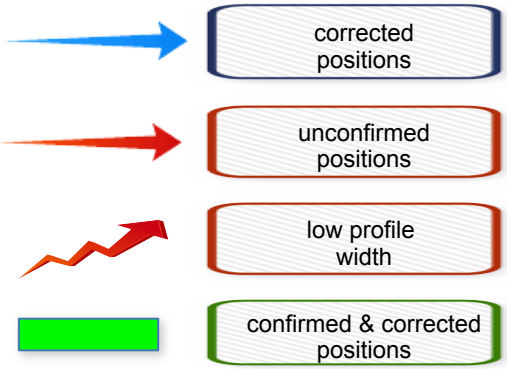
Fully corrected read. Moving into «Clean» base



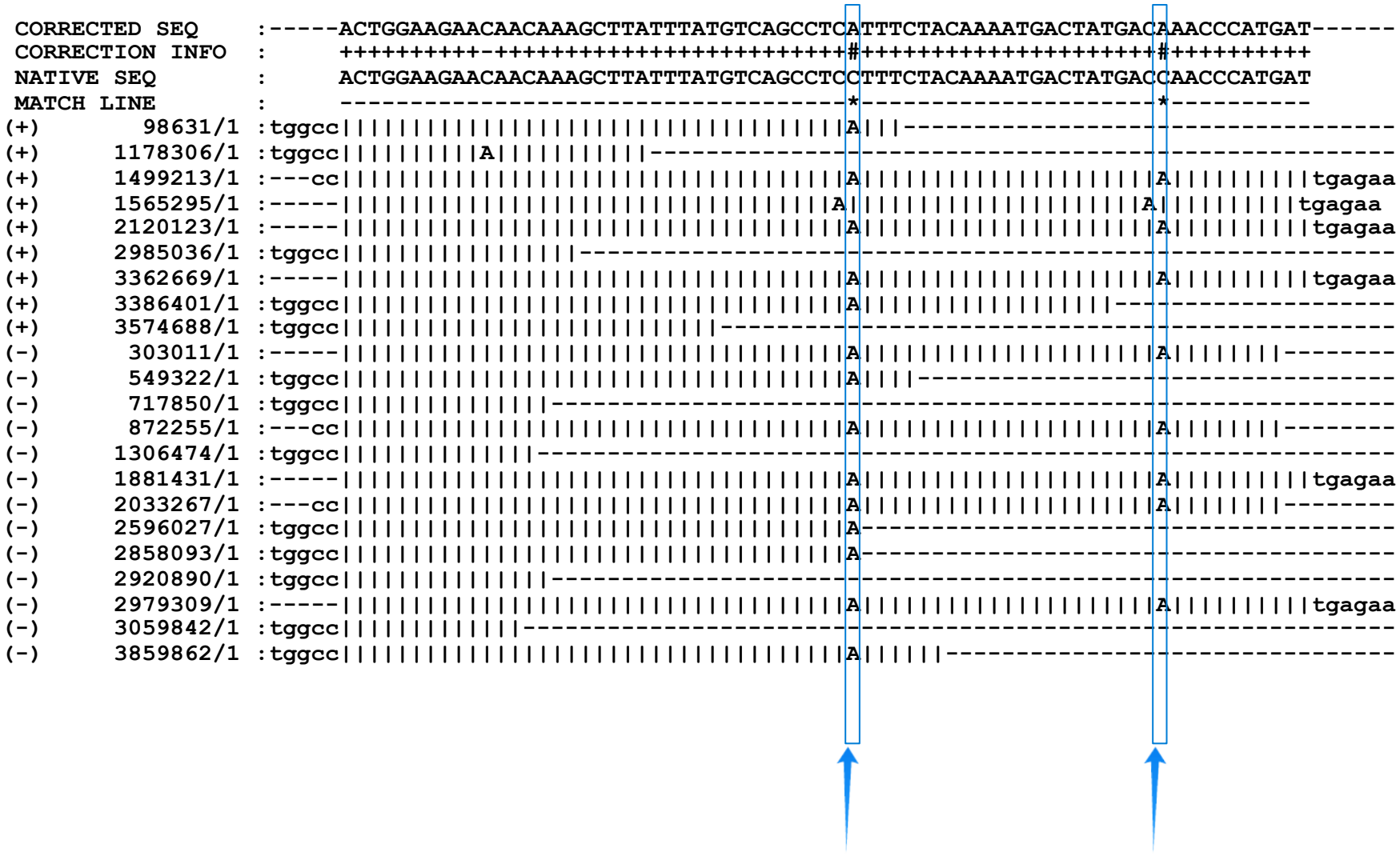
Partially corrected read. Moving into «Dirty» base



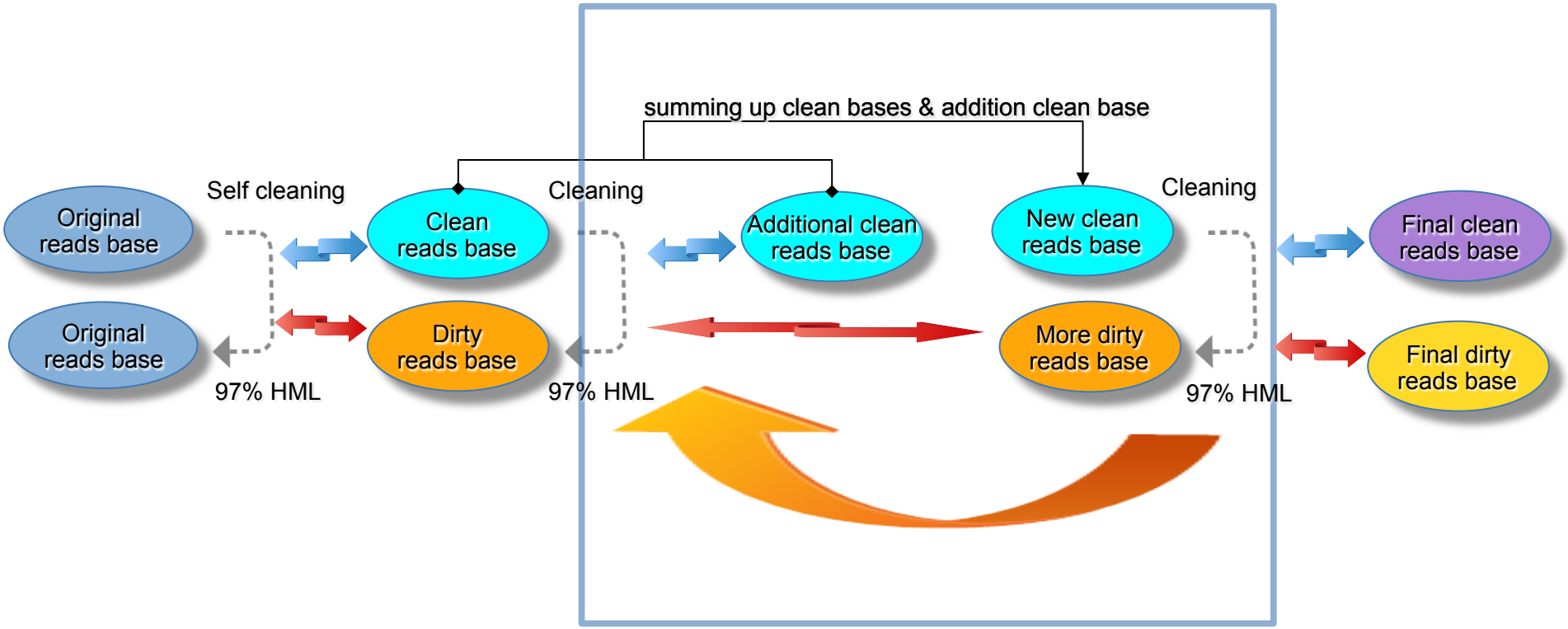
Not-corrected read. Moving into «Dirty» base



Fully corrected read. Moving into «Clean» base



Iterative procedure for cleaning NGS reads



Loop. Iterate until «Additional clean reads base» is not too small or empty

CleanReads program accuracy on E.coli data: 1% errors

```
CLEANED READS
Nucleotides   185448000/ 185587000   99.925103%
left errors       :          99    0.005706%    0.000053%
non-corrected err :          93    0.005360%    0.000050%
wrong-corrected   :           6    0.000346%    0.000003%
over-corrected    :           0    0.000000%    0.000000%
miss-corrected    :           6    0.000346%    0.000003%
initial mutation  :     1735154  100.000000%    0.935655%
seq   :    4639675
cln   :    4639552   99.997414%   99.997349%
ori   :    4639672  100.000000%   99.999935%

DIRTY READS
Nucleotides      139000/ 185587000    0.074897%
left errors       :         359   17.950000%    0.258273%
non-corrected err :         327   16.350000%    0.235252%
wrong-corrected   :          32    1.600000%    0.023022%
over-corrected    :           0    0.000000%    0.000000%
miss-corrected    :          32    1.600000%    0.023022%
initial mutation  :        2000  100.000000%    1.438849%

************************************************************
```

BIOINFORMATICS

Vol. 00 no. 00 2014
Pages 1–7

Karect: Accurate Correction of Substitution, Insertion and Deletion Errors for Next-generation Sequencing Data

Amin Allam*, Panos Kalnis and Victor Solovyev

Computer, Electrical and Mathematical Sciences & Engineering, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

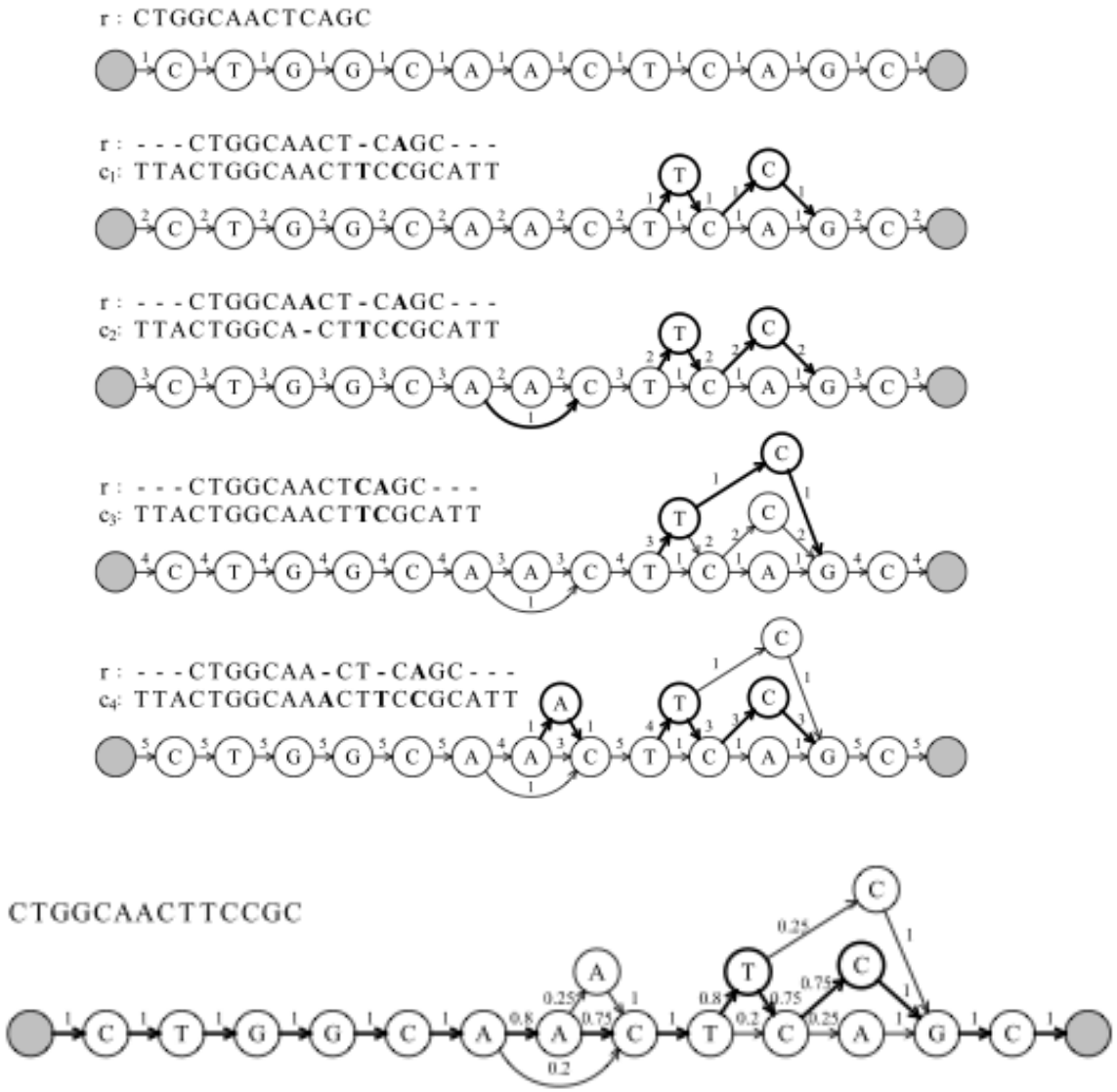
Motivation: Next-generation sequencing generates large amounts of data affected by errors in the form of substitutions, insertions, or deletions of bases. Error correction based on the high-coverage information, typically improves *de novo* assembly. Most existing tools can correct substitution errors only; few support insertions and deletions, but accuracy is low.

Results: We present *Karect*, a novel error correction technique based on multiple alignment. Our approach supports substitution, insertion and deletion errors. It can handle non-uniform coverage as well as moderately covered areas of the sequenced genome. Experiments with data from Illumina, 454 FLX and Ion Torrent sequencing machines demonstrate that Karect is more accurate than previous methods, both in terms of correcting individual-bases errors

Table 1. NGS error types. Read lengths are as follows: Small ≈ 36-200 bps, moderate ≈ 200-700 bps, very long ≈ 1000-10000 bps.

Brand	Read length	Throughput	Dominant error type
Illumina	small	very high	substitutions
SOLiD	small	high	substitutions
454 FLX	moderate	moderate	insertions, deletions
Ion Torrent	moderate	high	insertions, deletions
Pacific Biosciences	very long	very high	insertions, long gaps

assemblers, such as SOAPdenovo (Li *et al.*, 2010), ALLPATHS-LG (Gnerre *et al.*, 2011), SGA (Simpson and Durbin, 2012)


r : CTGGCAACTCAGC
r : - - - CTGGCAACT - CAGC - - -
c1: TTACTGGCAACTTCCGCATT
r : - - - CTGGCAACT - CAGC - - -
c2: TTACTGGCA - CTTCCGCATT
r : - - - CTGGCAACTCAGC - - -
c3: TTACTGGCAACTTCGCATT
r : - - - CTGGCAA - CT - CAGC - - -
c4: TTACTGGCAAACTTCCGCATT
CTGGCAACTTCCGC

Assembling E.coli using only CLEANED READS by Oligozip

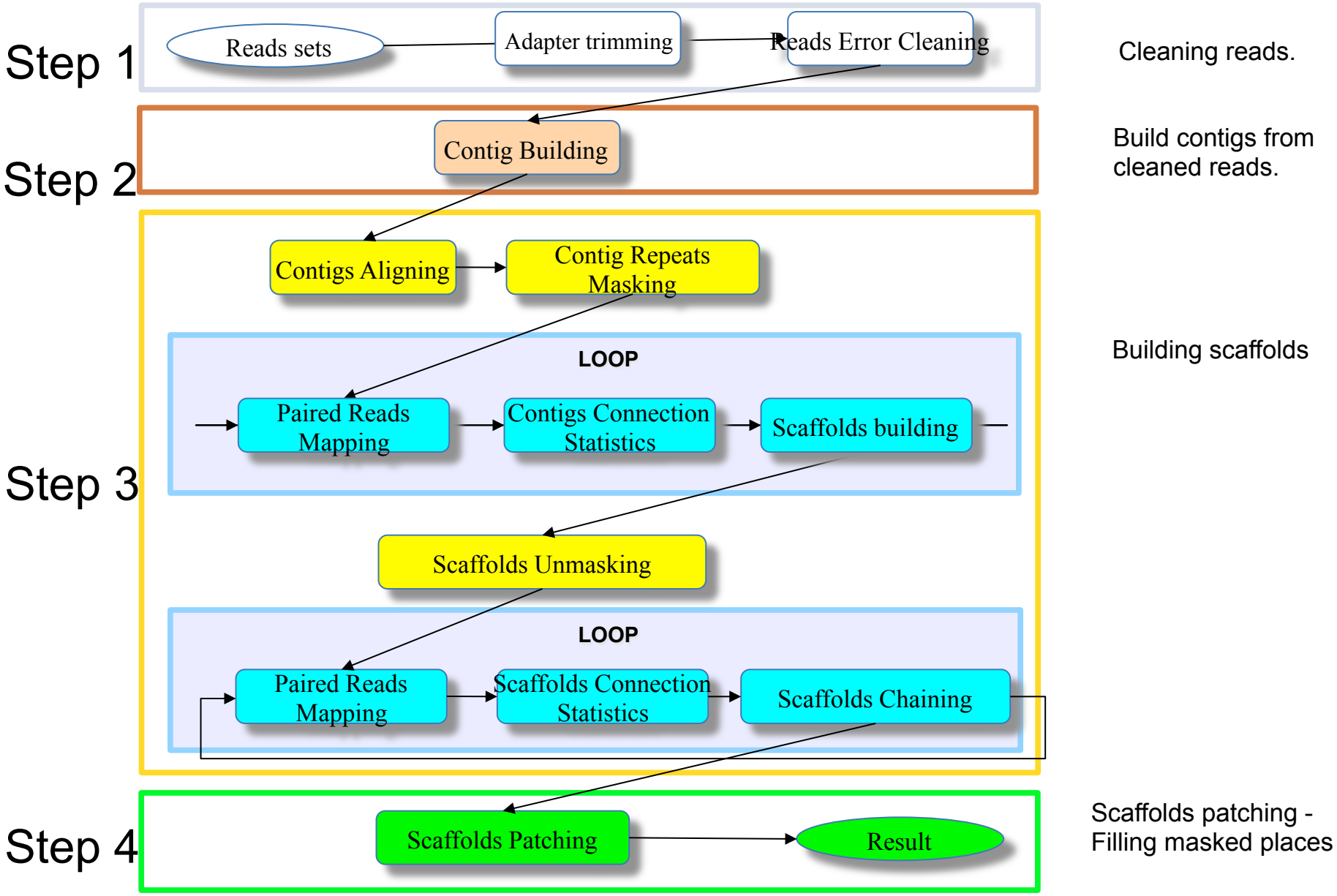
1	513877	0.1107
4	1265087	0.2726
10	2378497	0.5126
17	3284011	0.7078
30	4185763	0.9021
37	4417366	0.9520
48	4552357	0.9811
54	4571792	0.9854

Assembling Karect corrected READS (all set)

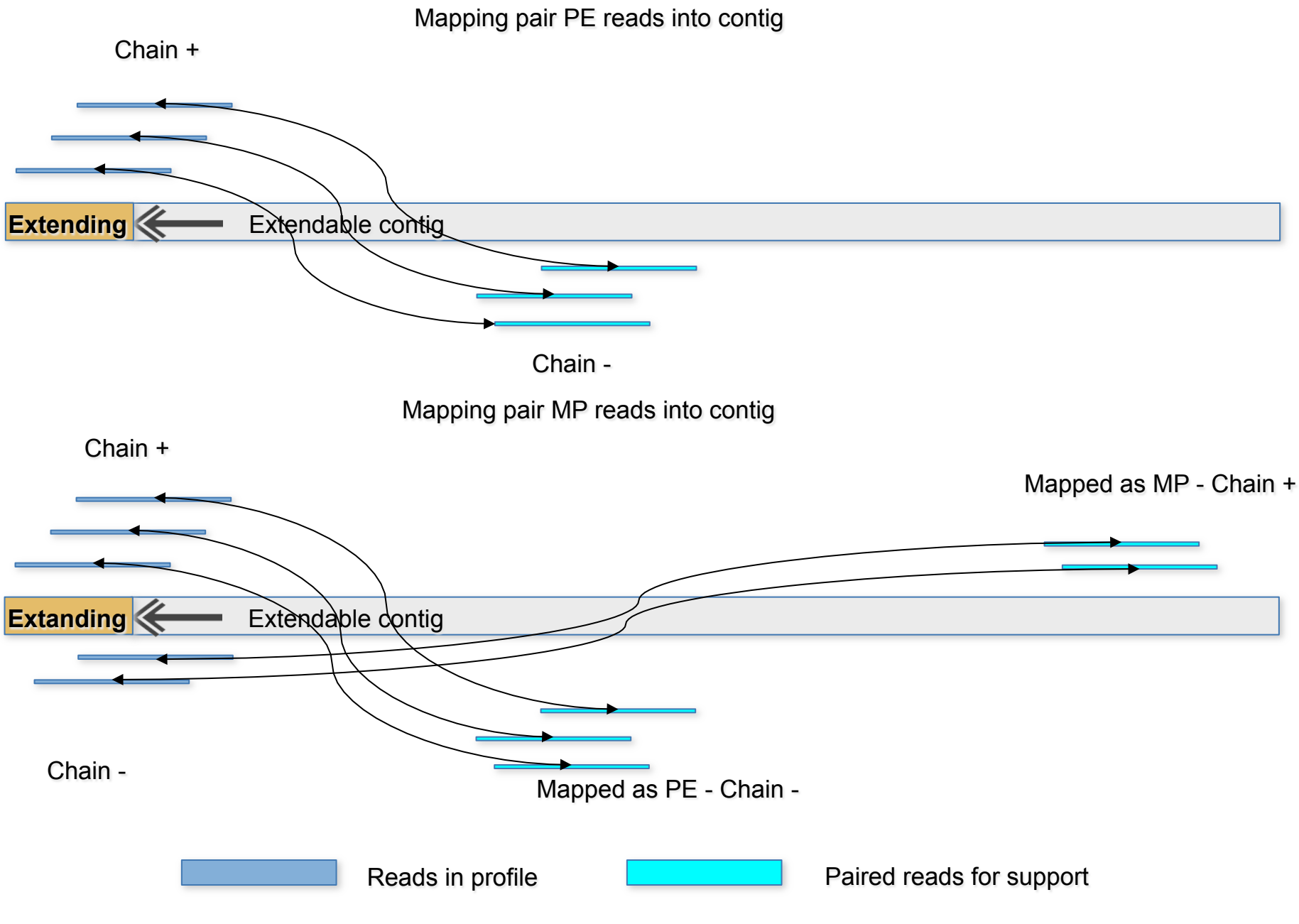
1	393827	0.0848
4	1213458	0.2615
10	2397824	0.5168
17	3290989	0.7093
30	4187960	0.9026
37	4420252	0.9527
47	4548090	0.9802
54	4572496	0.9855

```
Assembling Chr 22 on PE CLEANED READS (without dirty reads)
    1 |      401933   0.011519

   49 |     8777626   0.251547

  131 |    17470752   0.500673

  284 |    26189746   0.750540

  473 |    31407197   0.900060

  624 |    33153525   0.950106

  854 |    33913502   0.971885

Assembling on Karect CLEANED READS
    1 |      129453   0.003710

  168 |     8755653   0.250918

  500 |    17484657   0.501071

 1159 |    26176317   0.750155

 2033 |    31408372   0.900094

 2656 |    33151562   0.950050

 2818 |    33384913   0.956737

============================================================
```

Main steps of Oligozip de novo reads assembling

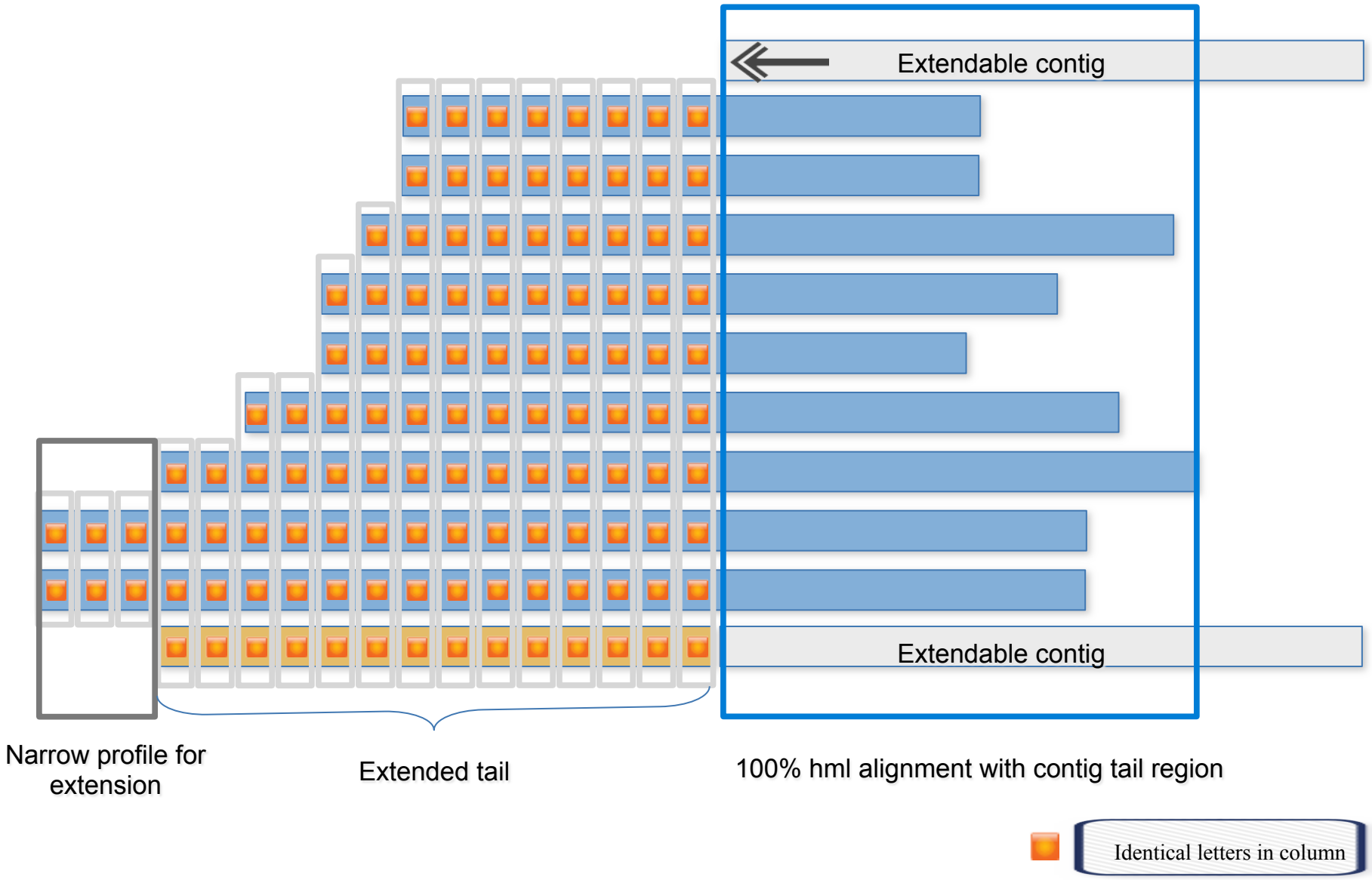


Support by paired reads

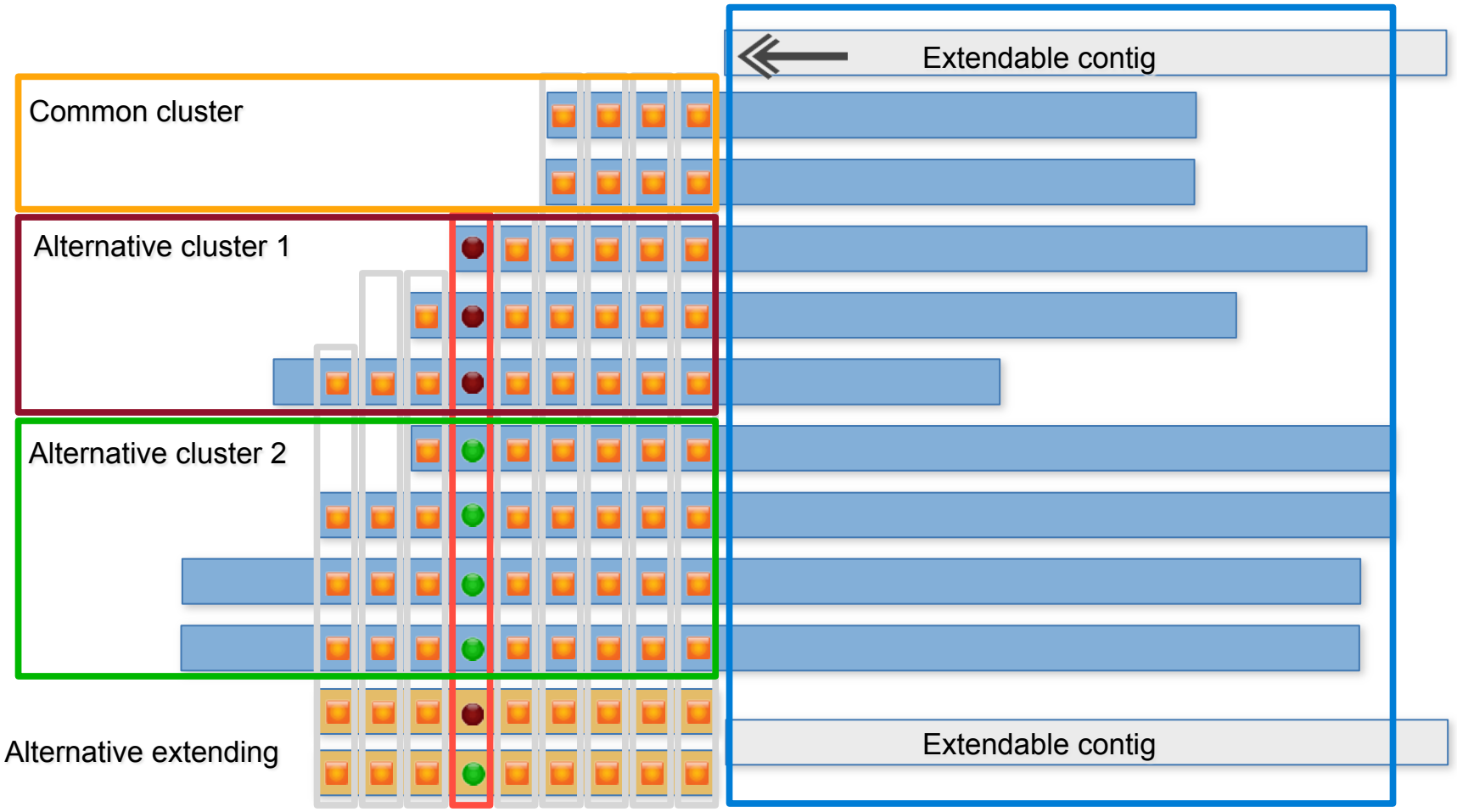


Contig extending

contig tail with 100% similar reads



Contig alternative extending.
Pair alignment contig tail with reads base with 100% homology



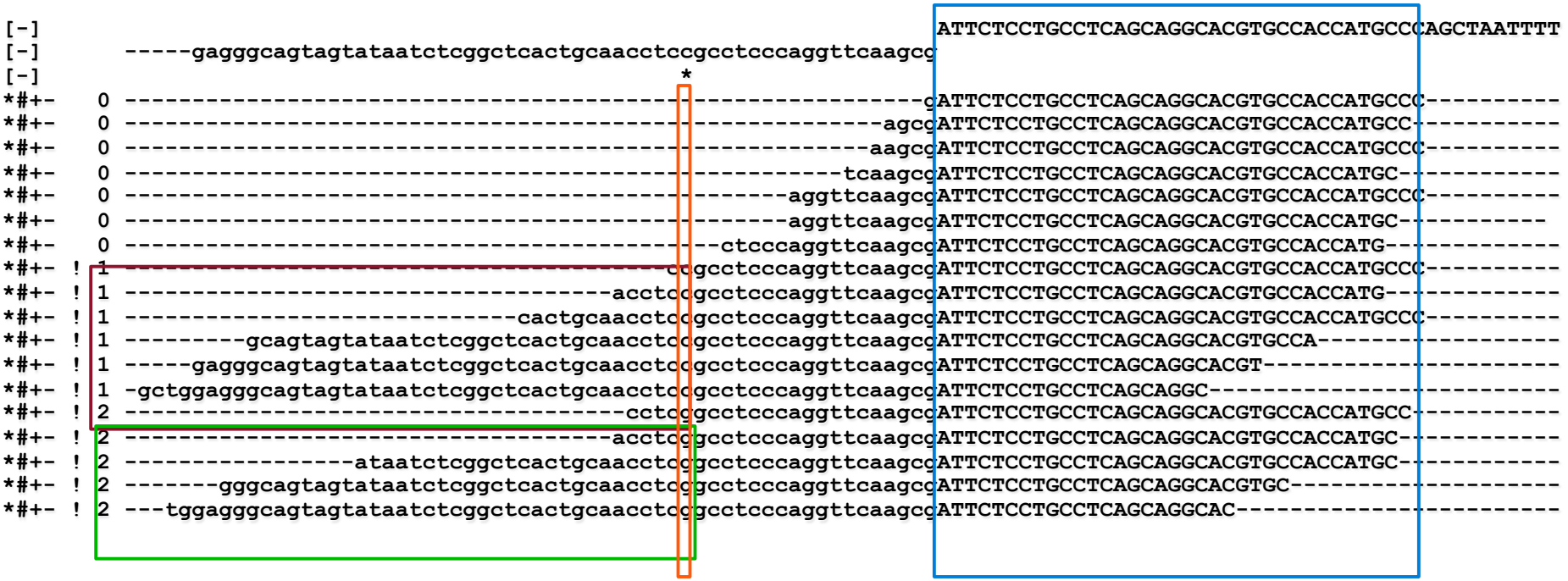
100% hml alignment with contig tail region

Identical letter in column

Alternative letter 1

Alternative letter 2

Alternative expanding. Both clusters valid, more tan one way for extension. Stop extension in this direction.



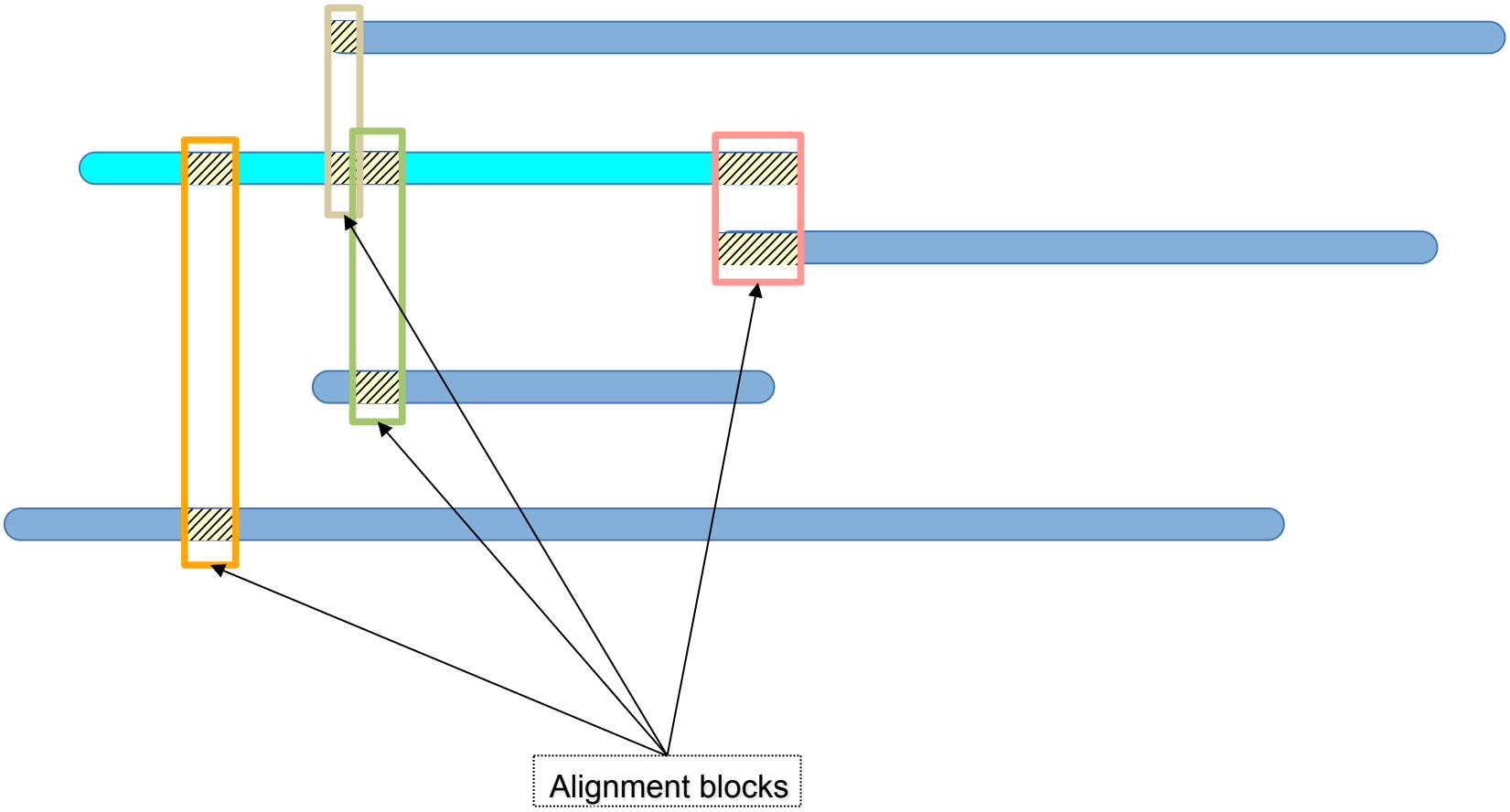
100% hml alignment with contig tail region

Different letters. Split into clusters zone.

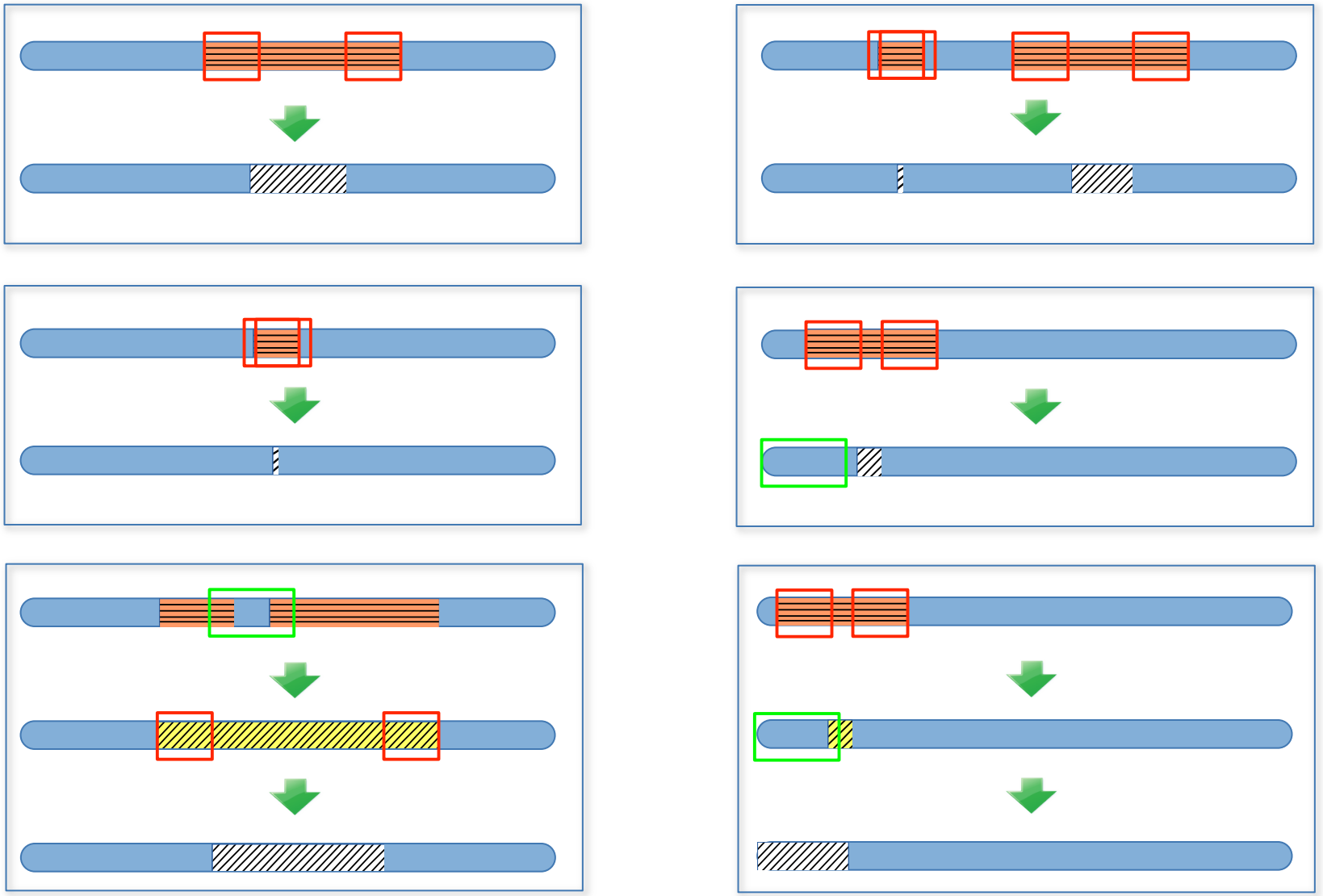
Good cluster 1

Good cluster 2

Pairwise alignment between the assembled contigs

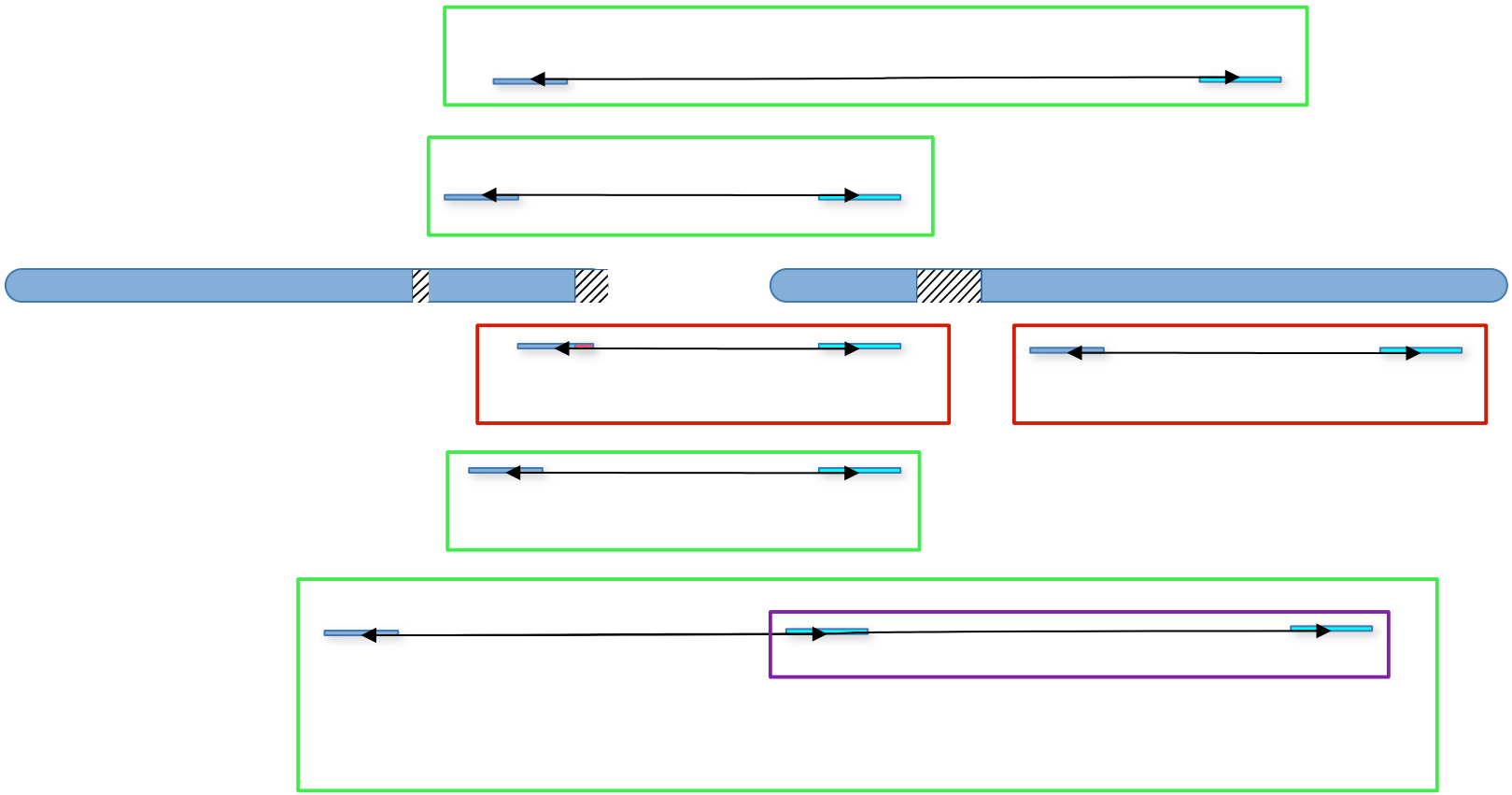


Masking contigs parts with >= 99% homology



Base repeat region
Temporary masked region
Masked region
Cutting part.
~ Half min read length.
Expanding part.
~ Min read length.

Paired reads mapping on pair contigs.



Masked region

Paired reads

Not mapped read part - masked region

Valid pair mapped

Not valid pair mapped

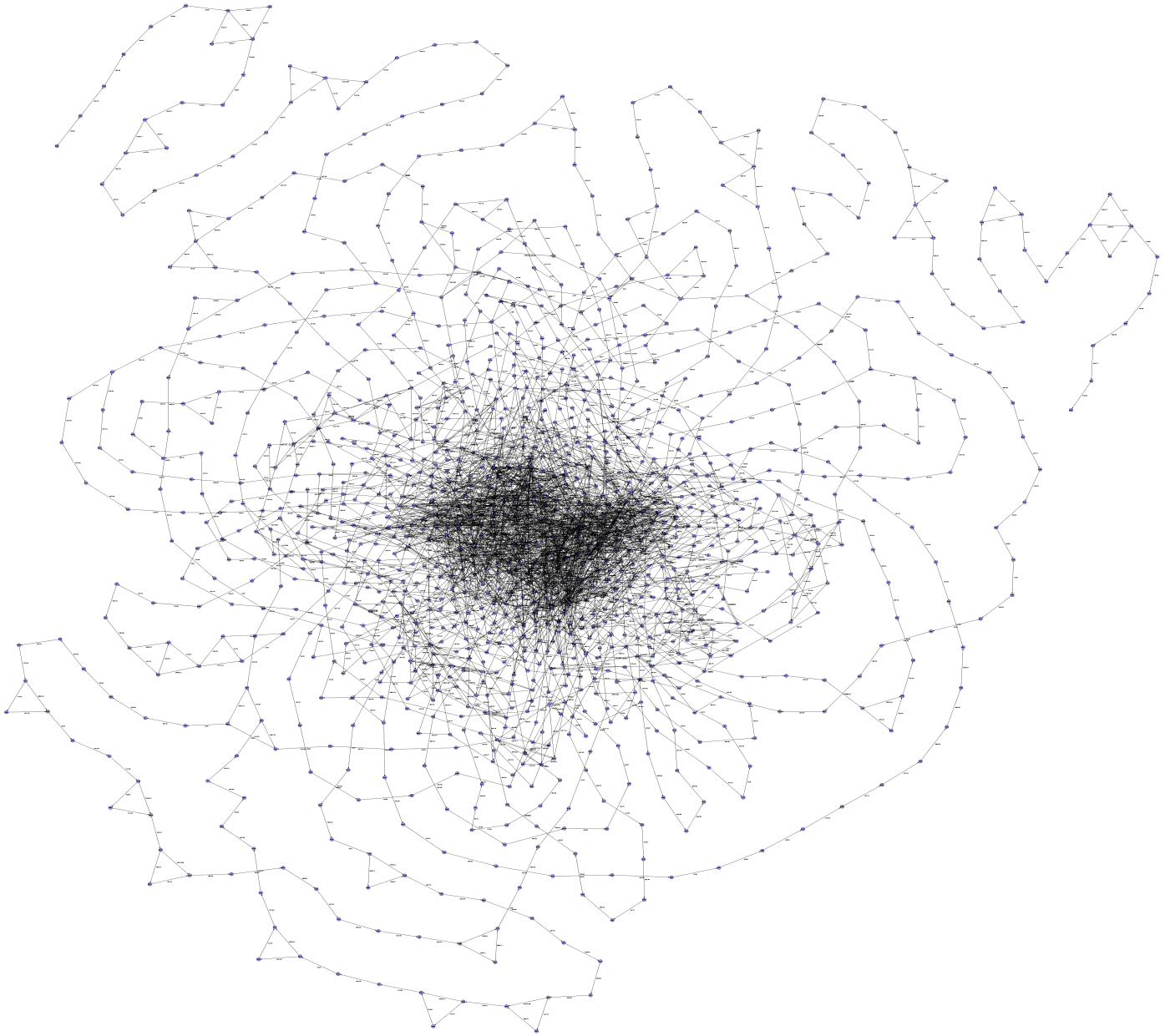
Alternative valid pair mapping.
Only for MP reads

Contig pairs connection statistics

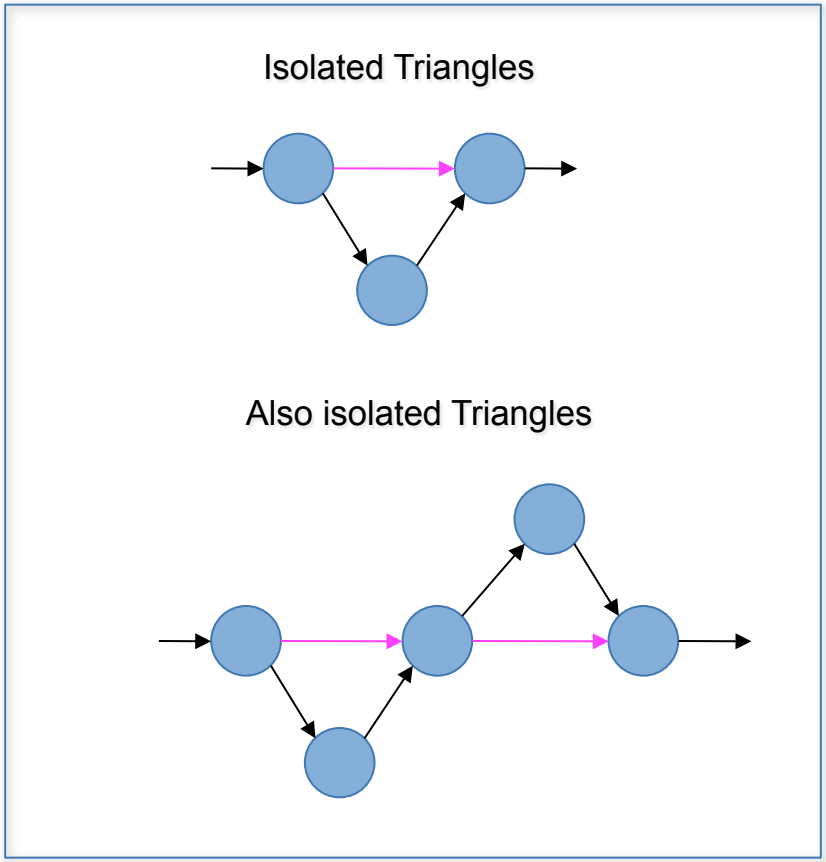
```
CONTIG:66 132769
N:19
        0     +0.000     +0.000     +0.000
        0     +0.000     +0.000     +0.000
        1   -703.000   +500.000   +500.000
        0     +0.000     +0.000     +0.000
N:28
        0     +0.000     +0.000     +0.000
       60   -315.500    +58.332    +50.000
        0     +0.000     +0.000     +0.000
        0     +0.000     +0.000     +0.000
CONTIG:37 141721
N:17
        0     +0.000     +0.000     +0.000
        0     +0.000     +0.000     +0.000
        8  +1172.667   +559.241   +500.000
        0     +0.000     +0.000     +0.000
N:19
      167   -480.329   +517.342   +500.000
        0     +0.000     +0.000     +0.000
        3  -1413.000   +451.114   +500.000
        0     +0.000     +0.000     +0.000
N:28
        0     +0.000     +0.000     +0.000
        0     +0.000     +0.000     +0.000
        0     +0.000     +0.000     +0.000
        3   +967.000   +602.719   +500.000
CONTIG:62 86790
N:48
        0     +0.000     +0.000     +0.000
        0     +0.000     +0.000     +0.000
       82  +1358.293   +487.531   +500.000
        0     +0.000     +0.000     +0.000
```

Contigs connections Human chromosome 21.

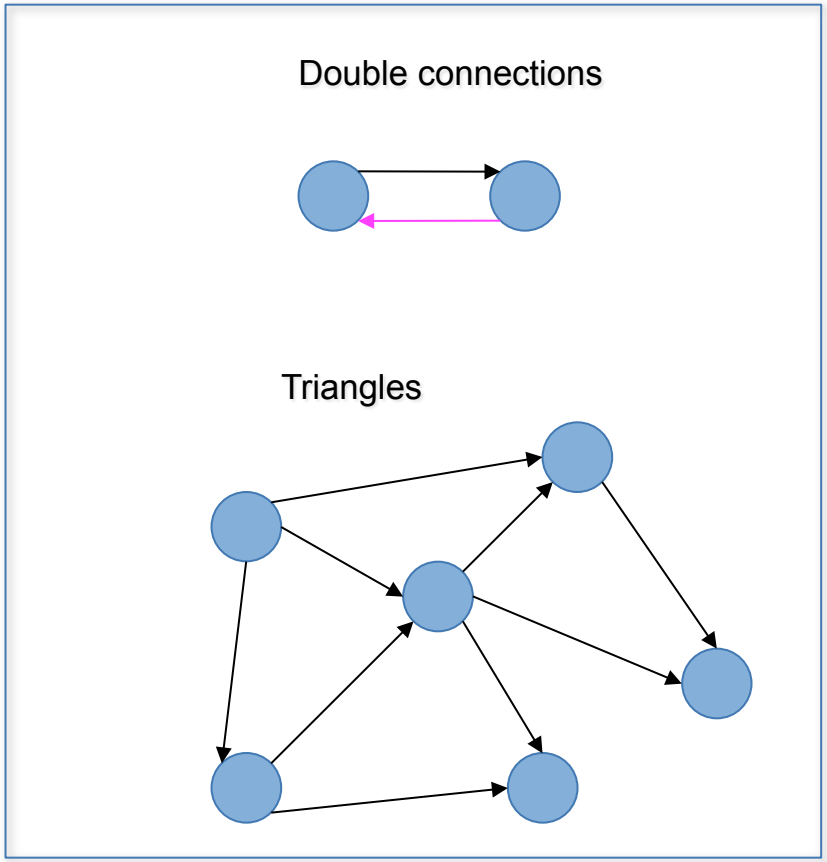
Iteration 1 (connections graph)



Simplifying contigs chains by removing non-informative connections.



Main priority for solution



Minor priority for solution

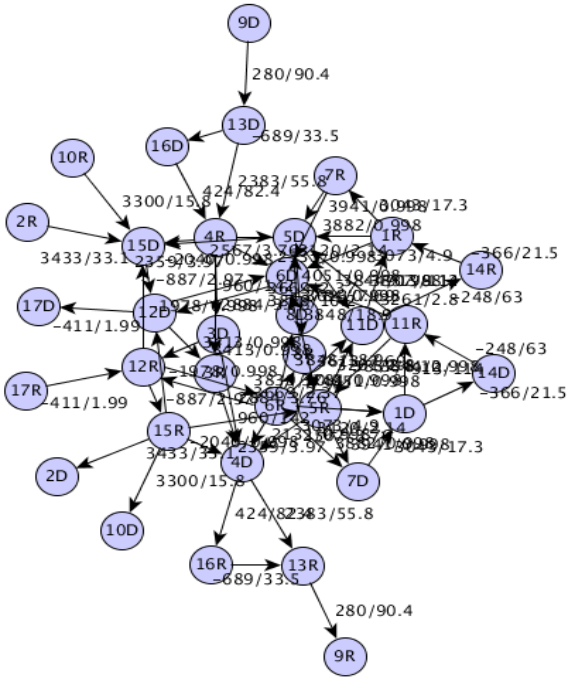
Contig

Contigs connection

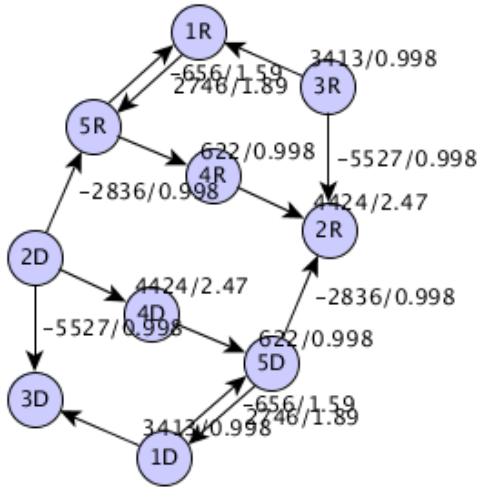
Candidate connection for removing

Contigs connections Human chromosome 21.

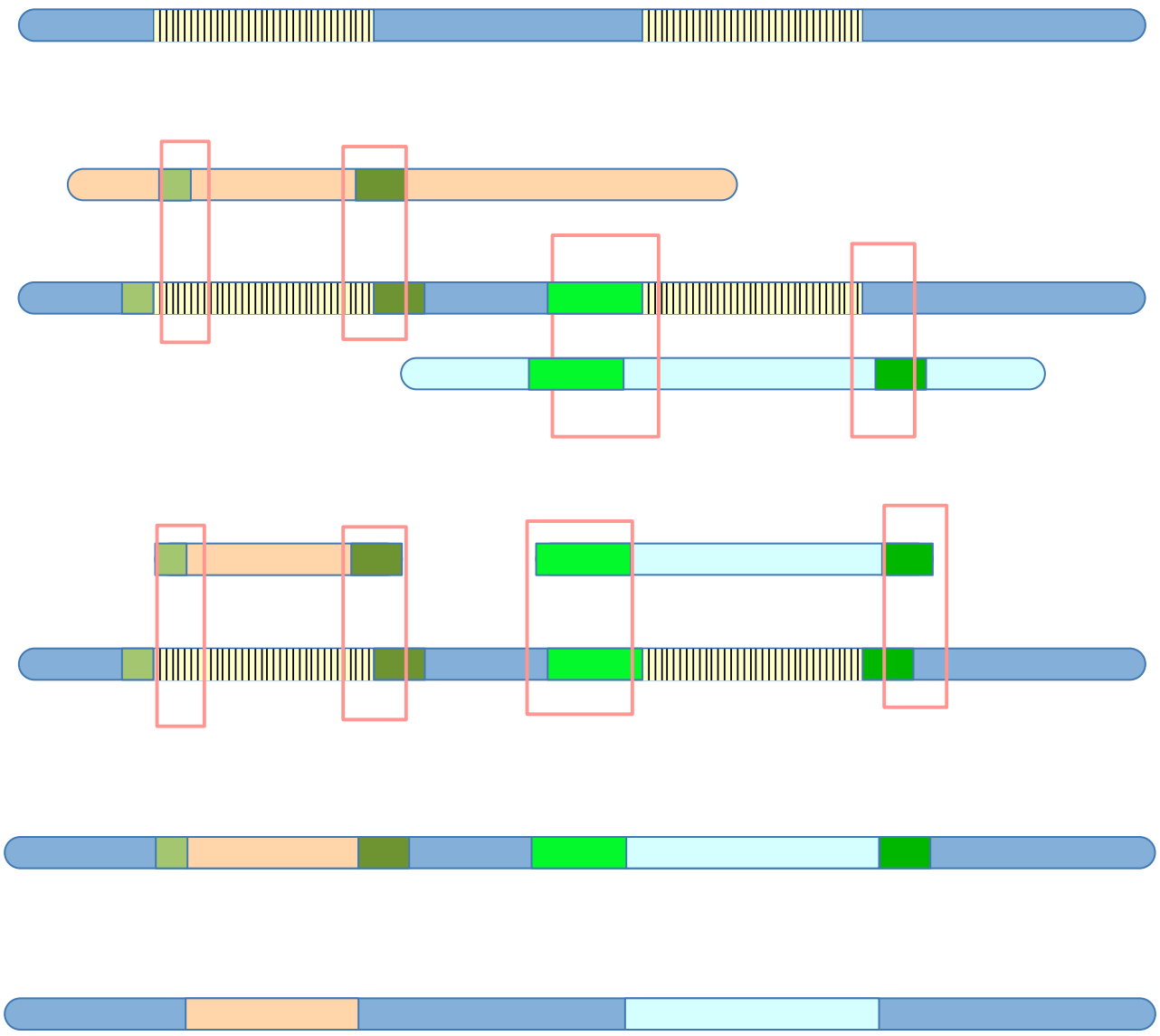
Iteration 2 (connections graph)



Iteration 3 (connections graph)



Contigs patching



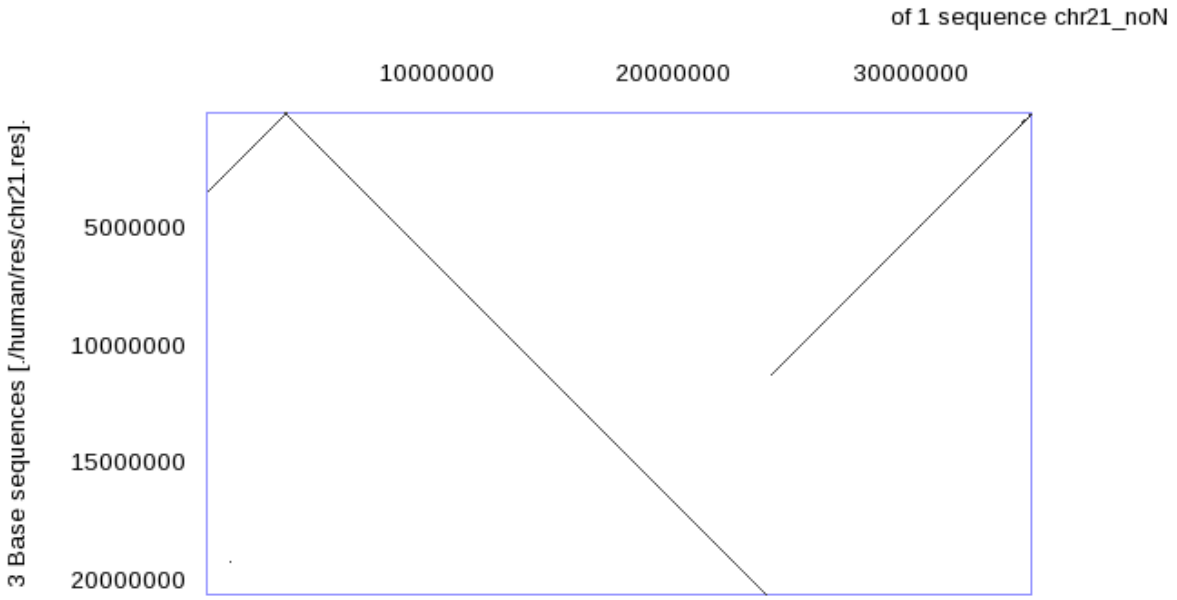
Masked region

Alignment regions
near the masked regions

Patches by alignment contigs.
Patched length must be approximately
equal the masked region length

Assembling Human chromosome 21 produce 3 largest contigs covering 99%.
Original length – 35106642 (without poly-N).

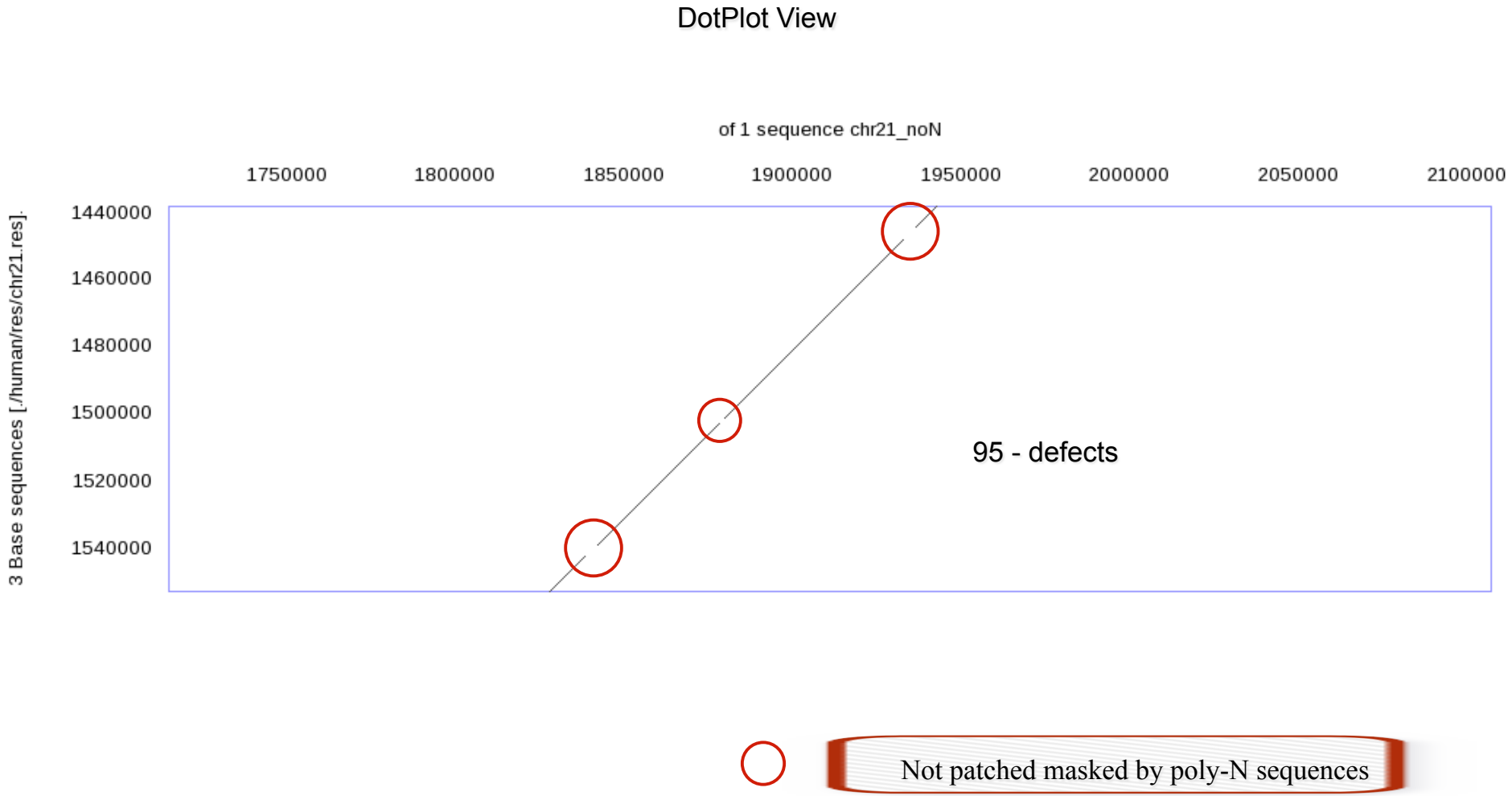
Contig	Length	Overall Length	Coverage	Overall Coverage	Defects	Sum Defects
1	20546395	20546395	0.583788	0.583788	95	95
2	11119456	31665851	0.315790	0.899578	70	165
3	3373878	35039729	0.095639	0.995217	37	202



GenomeMatch assembled contigs alignment to sequence of Human chromosome 21.

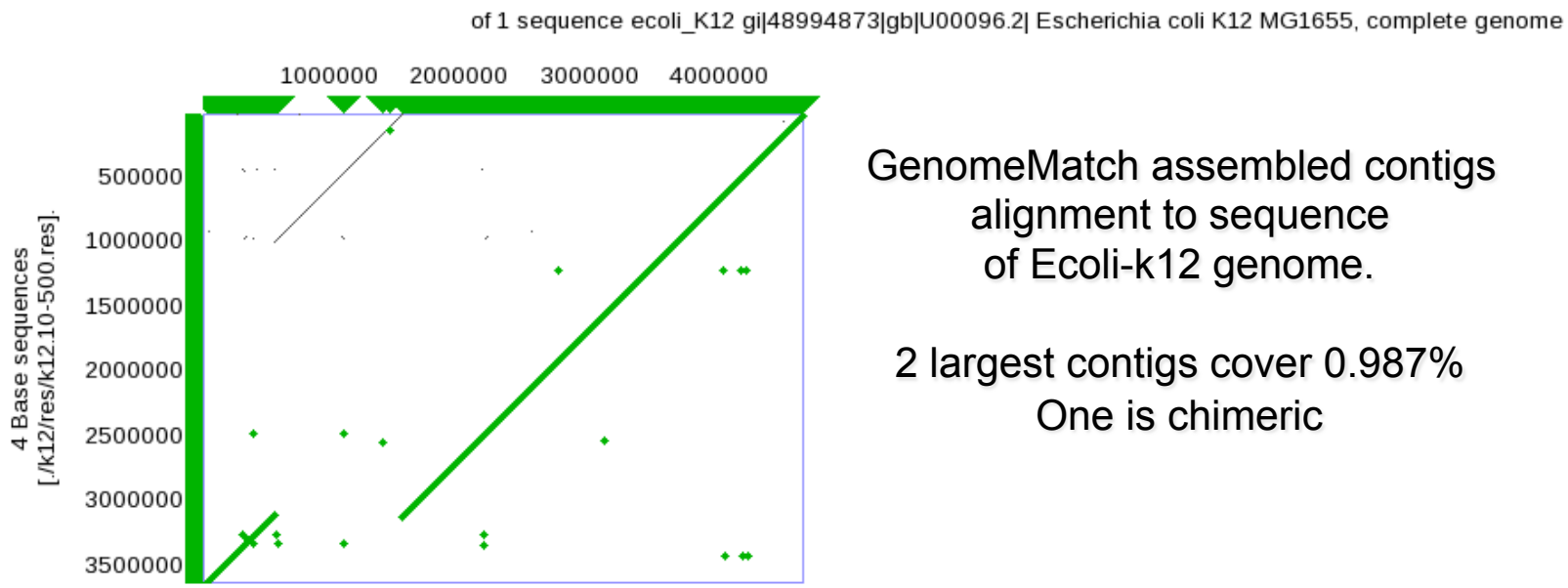
Total execution time ~ 6.5 hours

Gsbl alignment contigs to original Human chromosome 21.
Contig 1 – defects.



Assembling Ecoli-k12 2 largest contigs.
Original length – 4639675

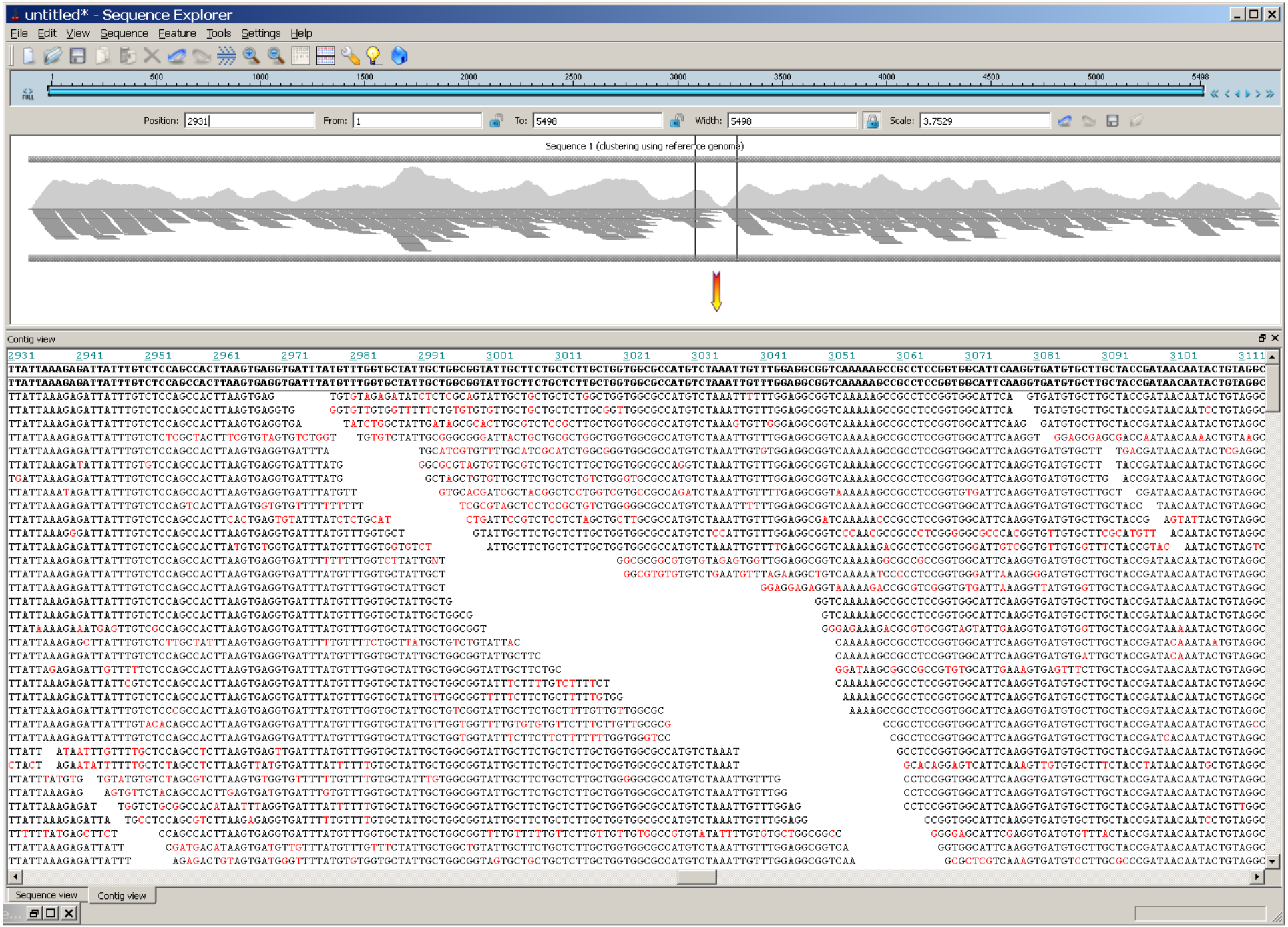
Contig	Length	Overall Length	Coverage	Overall Coverage	Defects	Total Defects
1	3632383	3632383	0.772727	0.772727	23	23
2	1000365	4632748	0.214343	0.987070	5	28



GenomeMatch assembled contigs
alignment to sequence
of Ecoli-k12 genome.

2 largest contigs cover 0.987%
One is chimeric

Total execution time ~ 15 minutes

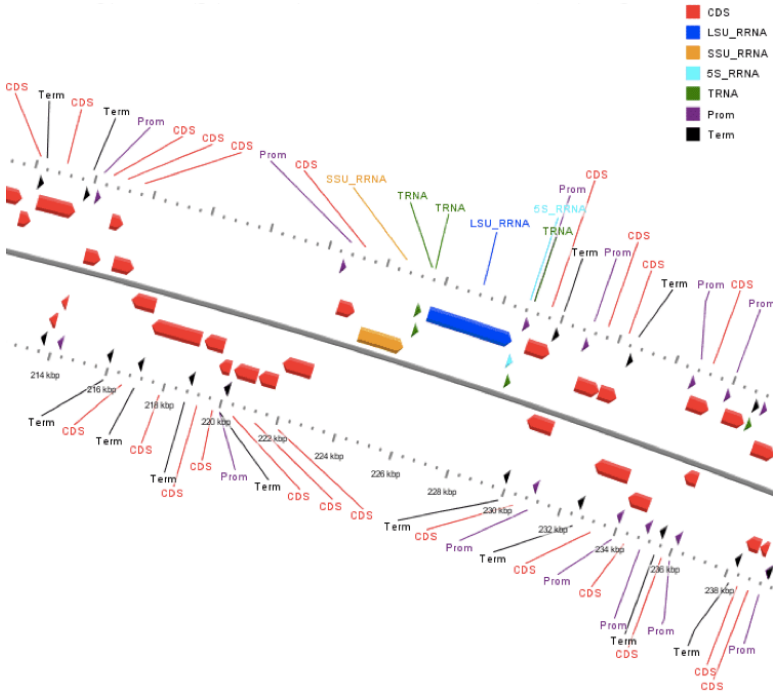

untitled* - Sequence Explorer
File Edit View Sequence Feature Tools Settings Help
Position: 2931
From: 1
To: 5498
Width: 5498
Scale: 3.7529
Sequence 1 (clustering using reference genome)
Contig view
Sequence view
Contig view

KAUST example: Pathogenic bacteria isolated from a patient in Mecca by Arnab Pain group *Stenotrophomonas maltophilia* is an emerging global opportunistic pathogen

Only PE reads 300 bp

Overall number of sequences = 34
Overall length of sequences = 4386843

Assembled by Oligozip
minimal sequence length = 6058.00
maximal sequence length = 461551.00
average sequence length = 129024.79



Annotation by Fgenesh pipeline

'Antibiotic resistance' gene: the aminoglycoside/hydroxyurea kinase found in Mecca isolate

Velvet:

Total number of contigs: 1067
n50: 11757
length of longest contig: 69162
Total bases in contigs: 4368900
Number of contigs > 1k: 530 Av: 8059
Total bases in contigs > 1k: 4271434

Assembled by Oligozip

Overall number of sequences > 1K= 34
Overall length of sequences = 4386843
maximal sequence length = 461551.00
average sequence length = 129024.79

Results comparable with the best Bacterial assembler Spade (that made 38 contigs > 1K Total L =4371351

JOURNAL OF COMPUTATIONAL BIOLOGY Volume 19, Number 5, 2012
; ANTON BANKEVICH,1,2 SERGEY NURK,1,2 DMITRY ANTIPOV,1 ALEXEY A. GUREVICH,1 MIKHAIL DVORKIN,1 ALEXANDER S. KULIKOV,1,3 VALERY M. LESIN,1
SERGEY I. NIKOLENKO,1,3 SON PHAM,4 ANDREY D. PRJIBELSKI,1 ALEXEY V. PYSHKIN,1 ALEXANDER V. SIROTKIN,1 NIKOLAY VYAHHI,1 GLENN TESLER,5
MAX A. ALEKSEYEV,1,6 and PAVEL A. PEVZNER1,4

Bacterial reads ~ $5*10^6 * 40 \sim 2* 10^8$ ~ 200 MB
~ 10^6 reads

Eukaryotic reads ~ $3*10^9 * 40 \sim 10^{11}$ ~ 100 GB
~ 10^9 reads

Linear: 10^3 Quadratic x 10^6

0.1 hour ~> 4 days 4000 days

TASK: Implementation algorithms on cluster computer

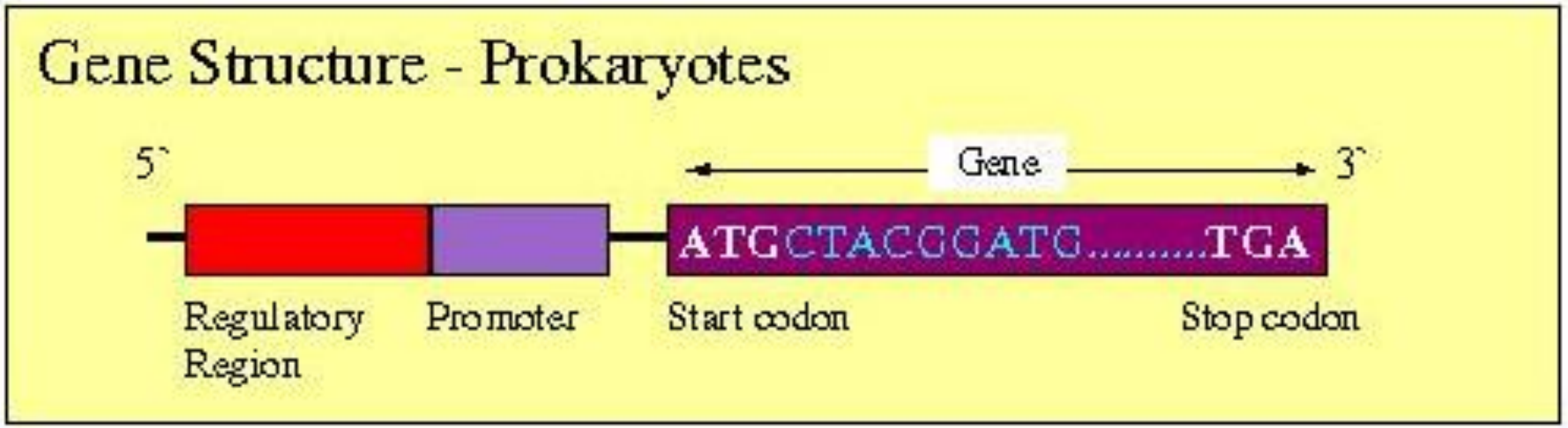
Find genes in DNA sequence

GAATTCTAATCTCCCTCTCAACCCTACAGTCACCCATTTGGTA
AGTAGTGTCAGGAATTAGTCATTTAAATAGTCTGCAAGCCAC
GTAGAAGTGGGAGGACTGCTTGAGCTCAAGAGTTTGATATTA
AAAAAAAAATTAGCCAGGCATGTGATGTACACCTGTAGTCC
TCAGGAGGTCAAGGCTGCAGTGAGACATGATCTTGCCACTG
AAACAGAATACAAAAACAAACAAACAAAAAACTGCTCCGC
TTTGTACACATTATCTCATTGCTGTTCGTAATTGTTAGATTAA
CTCAAGATGATAACTTTTATTTTCTGGACTTGTAATAGCTTTC
AACAATATAAAGTTATTGTGAGTTTTTGCAAACACATGCAAA
TGTCAATTTATGGGAAAACAAGTATGTACTTTTTCTACTAAG
ACATTTTCCGAAATTACTTGAGTATTATACAAAGACAAGCAC
GTGGAGACAAATGCAGGTTTATAATAGATGGGATGGCATCTA
GGACCCCAGTACACAAGAGGGGACGCAGGGTATATGTAGAC
TGACCTGAGTTTATAGACAATGAGCCCTTTTCTCTCTCCCAC
GGCTGACTCACTCCAAGGCCCAGCAATGGGCAGGGCTCTGT
AAGGGGTGGACTCCAGAGACTCTCCCTCCCATTCCCGAGCA
TAAAAGAAATAACAGGAGACTGCCCAGCCCTGGCTGTGACA
CCTTCTTTCAGTTAGAGGAAAAGGGGCTCACTGCACATACA

Escherichia coli K-12

Overview of Important Features the Sequence

- 4,639,221 bp of circular DNA
- Protein-coding genes account for 87.8% of the genome
- 0.8% encodes stable RNAs and tRNAs
- 0.7% consists of noncoding repeats
- 11% available for regulatory and other functions



A widely used approach: Markov models

•Markov chain models (1st order, higher order and inhomogeneous models; parameter estimation; classification)

• Hidden Markov models (forward, backward and Baum-Welch algorithms; model topologies; applications to gene finding and protein family modeling

Markov Chain Models

- a Markov chain model is defined by:
  - a set of states
    - some states *emit* symbols
    - other states (e.g. the *begin* state) are *silent*
  - a set of transitions with associated probabilities
    - the transitions emanating from a given state define a distribution over the possible next states

Markov Chain Models

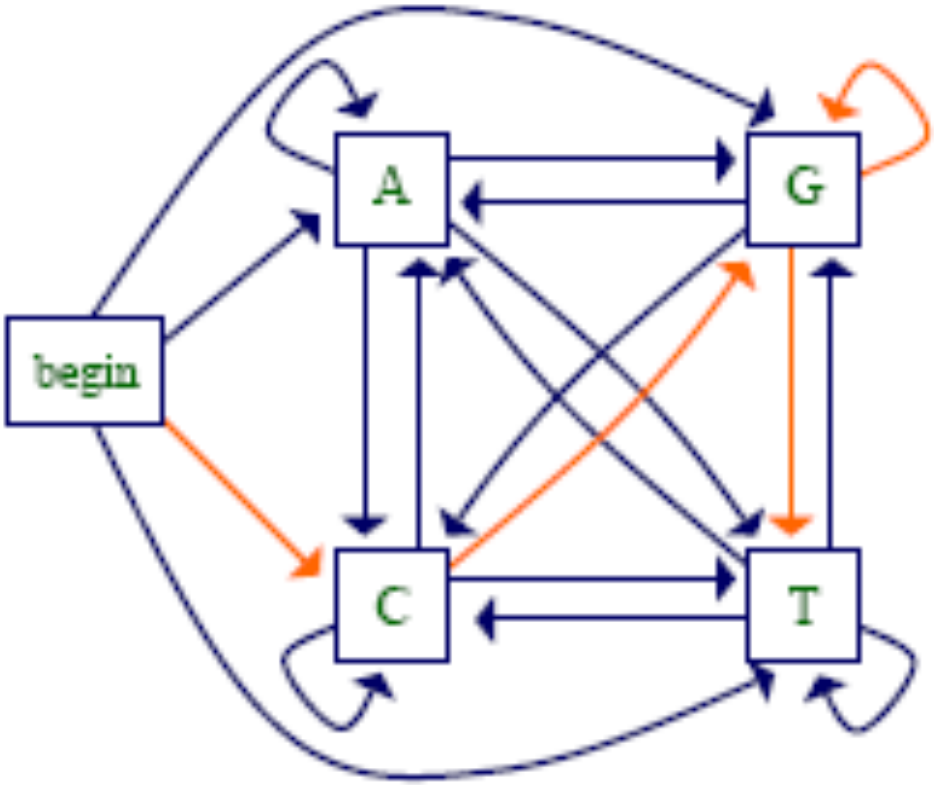
- given some sequence x of length L, we can ask how probable the sequence is given our model
- for any probabilistic model of sequences, we can write this probability as

$$\begin{aligned}\Pr(x) &= \Pr(x_L, x_{L-1}, \ldots, x_1) \\ &= \Pr(x_L | x_{L-1}, \ldots, x_1) \Pr(x_{L-1} | x_{L-2}, \ldots, x_1) \ldots \Pr(x_1)\end{aligned}$$

- key property of a (1st order) Markov chain: the probability of each X_i depends only on X_{i-1}

$$\begin{aligned}\Pr(x) &= \Pr(x_L | x_{L-1}) \Pr(x_{L-1} | x_{L-2}) \ldots \Pr(x_2 | x_1) \Pr(x_1) \\ &= \Pr(x_1) \prod_{i=2}^{L} \Pr(x_i | x_{i-1})\end{aligned}$$

Markov Chain Models



$$\Pr(cggt) = \Pr(c)\Pr(g|c)\Pr(g|g)\Pr(t|g)$$

Higher Order Markov Chains

- the Markov property specifies that the probability of a state depends only on the probability of the previous state
- but we can build more "memory" into our states by using a higher order Markov model
- in an *n*th order Markov model

$$\Pr(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_1) = \Pr(x_i \mid x_{i-1}, \ldots, x_{i-n})$$

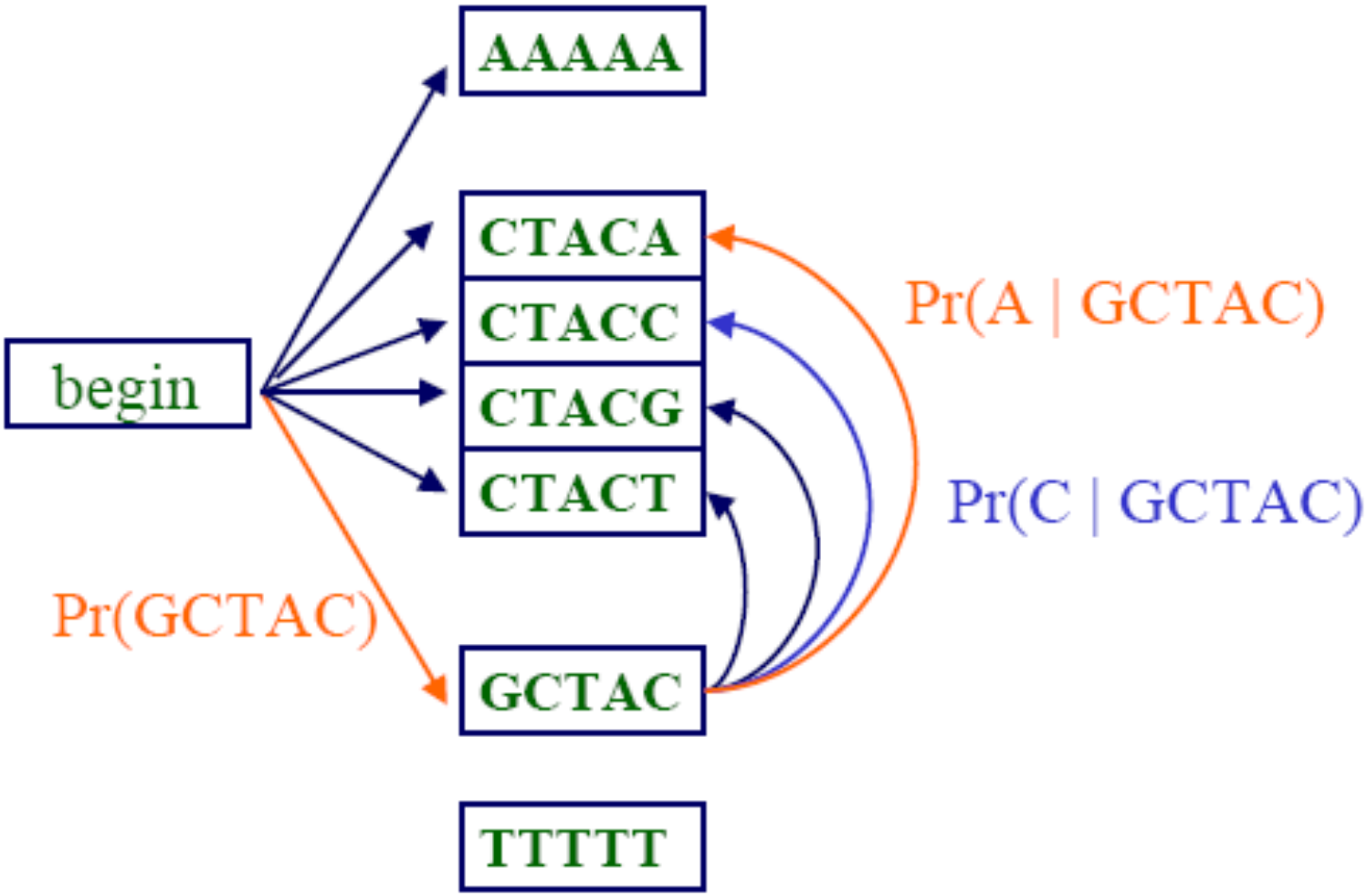
Higher Order Markov Chains

- An *n*th order Markov chain over some alphabet is equivalent to a first order Markov chain over the alphabet of *n*-tuples

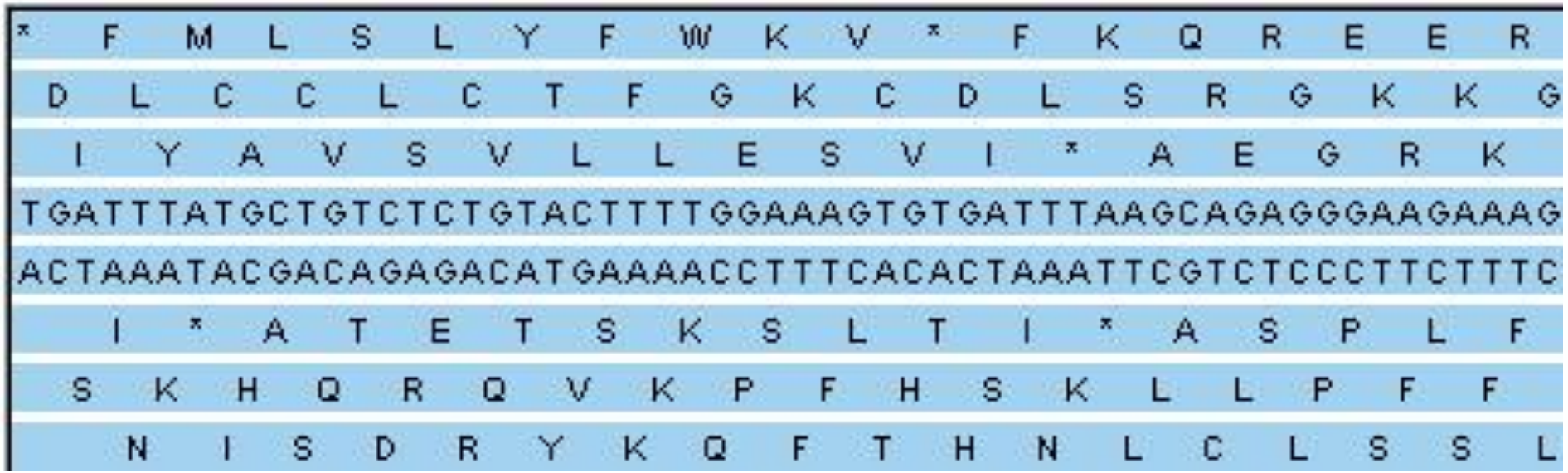
- Example: a 2nd order Markov model for DNA can be treated as a 1st order Markov model over alphabet:

  AA, AC, AG, AT, CA, CC, CG, CT, GA, GC, GG, GT, TA, TC, TG, and TT (i.e. all possible dinucleotides)

A Fifth Order Markov Chain



Translation to 6 ORF




```
DNA: CTTGCGCTTTCTCACACCAGCAAACATGGCGCTTCCAGGCTCCACAATGAA
 +3:   C  A  F  S  H  Q  Q  T  W  R  F  Q  A  P  Q  *  T
 +2:  L  R  F  L  T  P  A  N  M  A  L  P  G  S  T  M  N
 +1: L  A  L  S  H  T  S  K  H  G  A  S  R  L  H  N  E

DNA: CTCCAGCGCGTTGAGCTGGTCCAGCAGCAATTCCAGGTCAGAGGCCTGGCC
 -1: L  Q  R  V  E  L  V  Q  Q  Q  F  Q  V  R  G  L  A
 -2:  S  S  A  L  S  W  S  S  S  N  S  R  S  E  A  W  P
 -3:   P  A  R  *  A  G  P  A  A  I  P  G  Q  R  P  G  P
```

Codon Composition

Nucleotide variation at codon position:

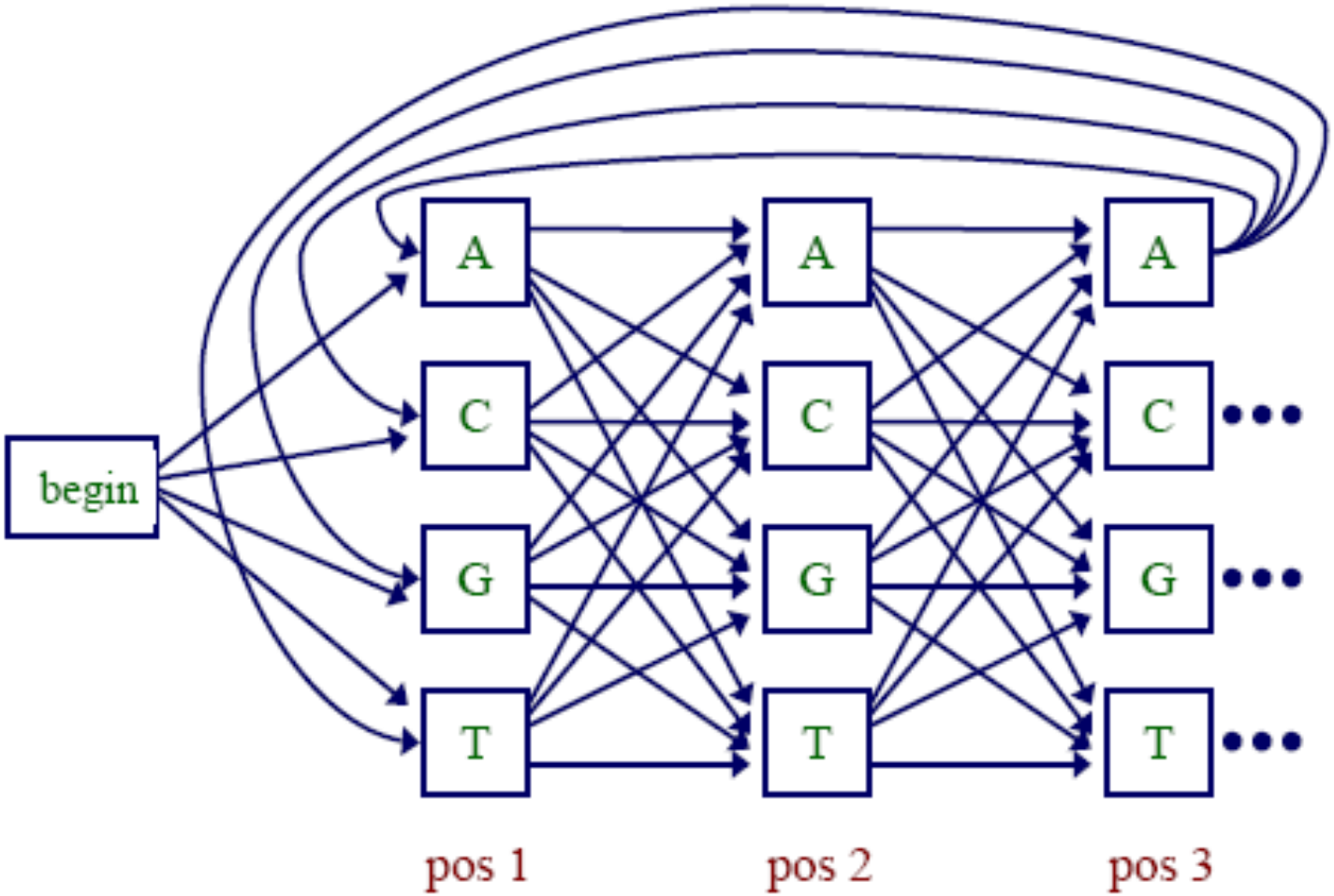
Campylobacter jejuni

	Codon Position		
	1	2	3
a	36%	36%	36%
c	13%	17%	9%
g	30%	14%	10%
t	21%	33%	44%

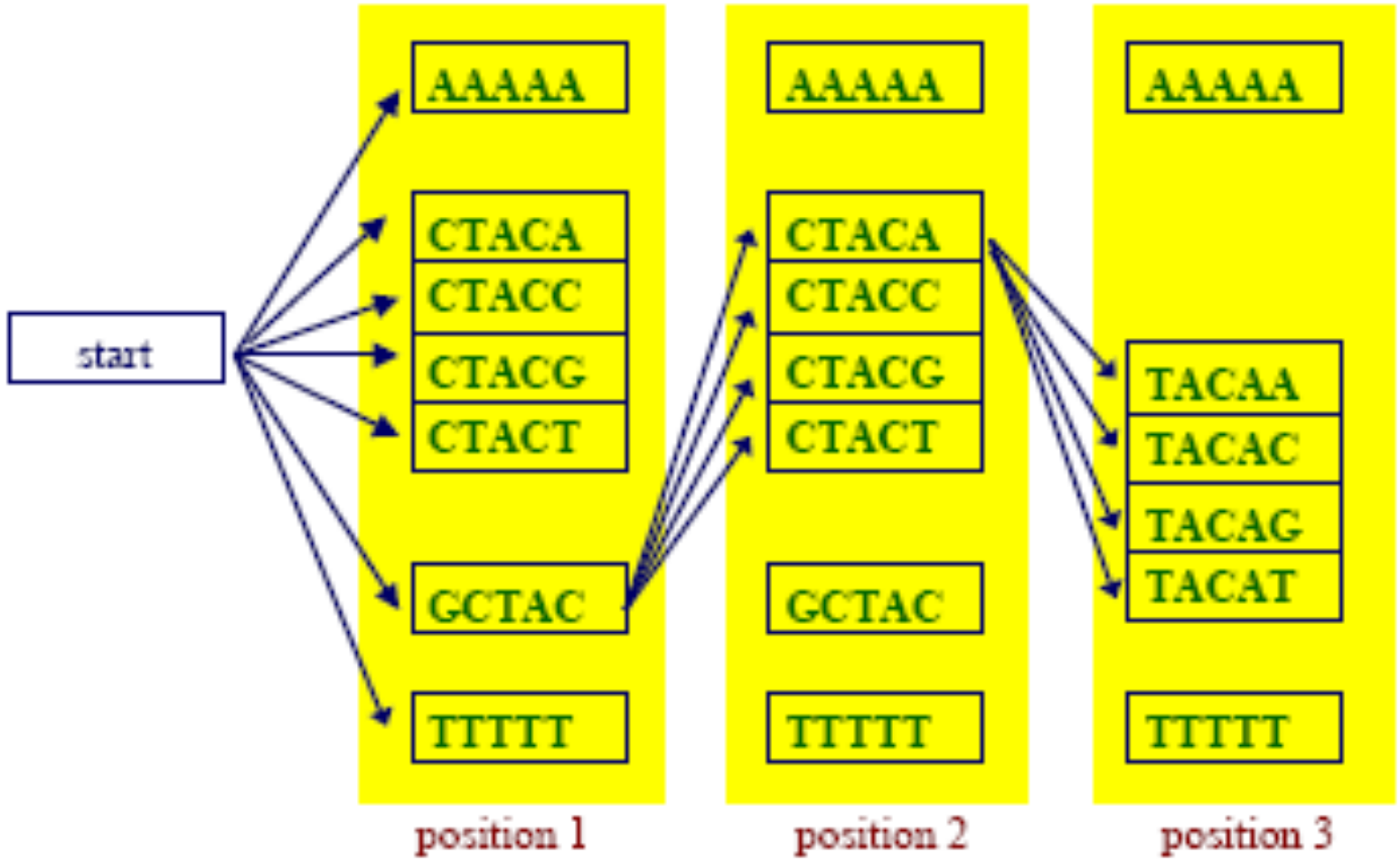
Mycobacterium smegmatis

	Codon Position		
	1	2	3
a	19%	23%	6%
c	27%	28%	48%
g	42%	20%	39%
t	12%	28%	7%

Inhomogenous Markov Chains



A Fifth Order Inhomogenous Markov Chain



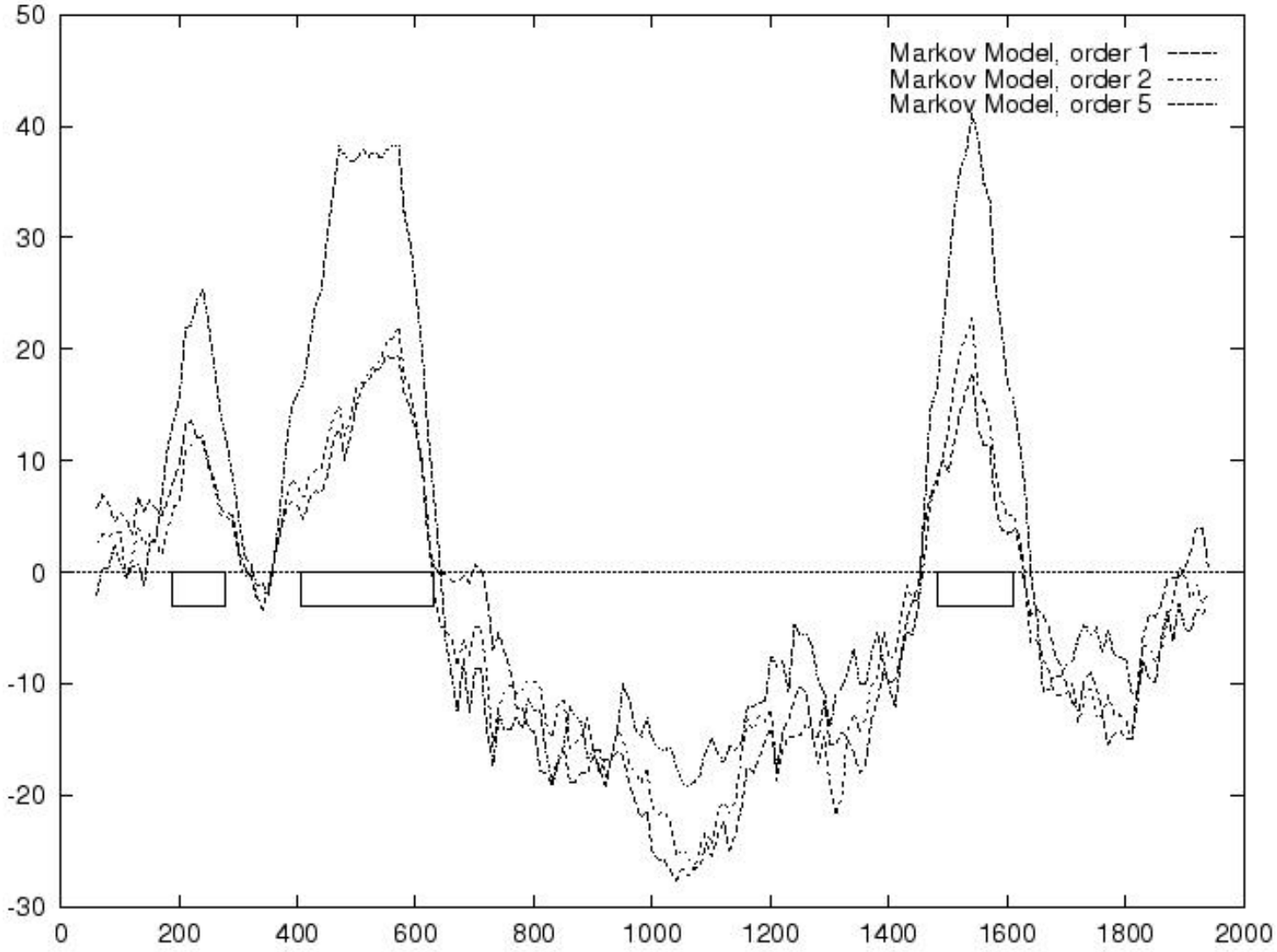
Selecting the Order of a Markov Chain Model

- Higher order models remember more "history"
- Additional history can have predictive value
- Example:

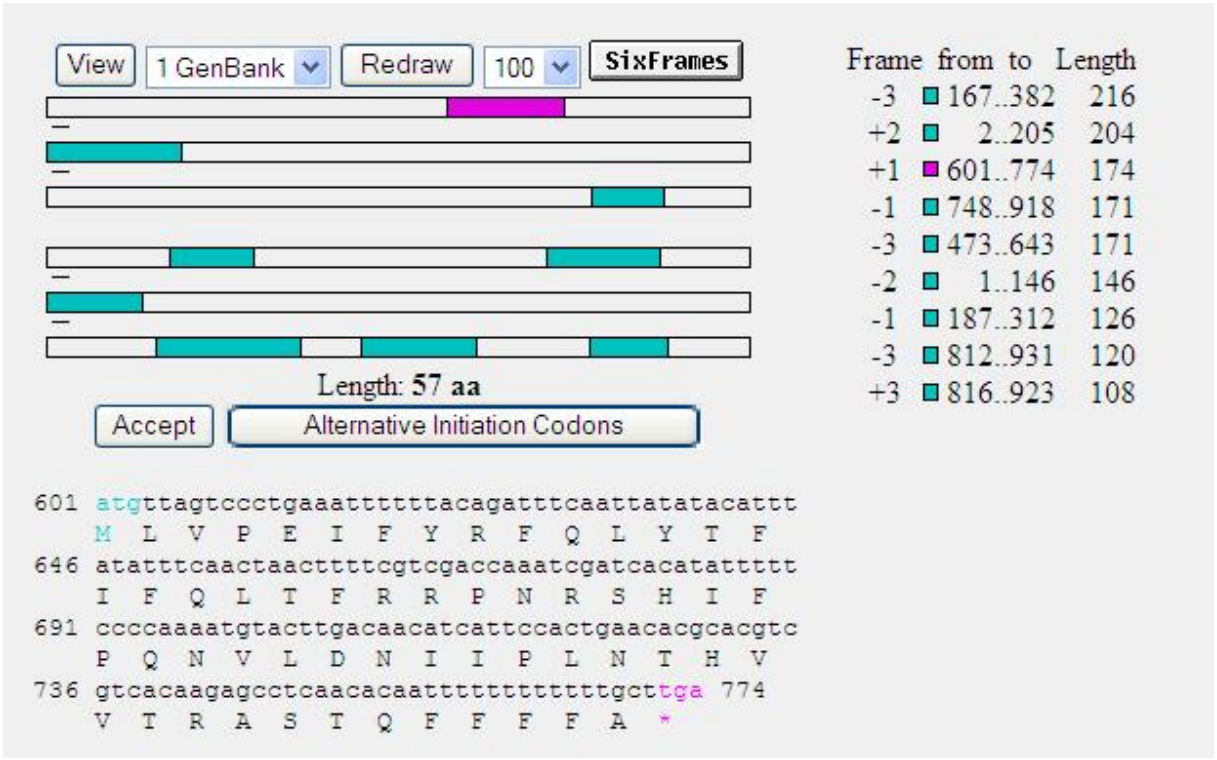
  – predict the next word in this sentence fragment "… finish __" (up, it, first, last, …?)

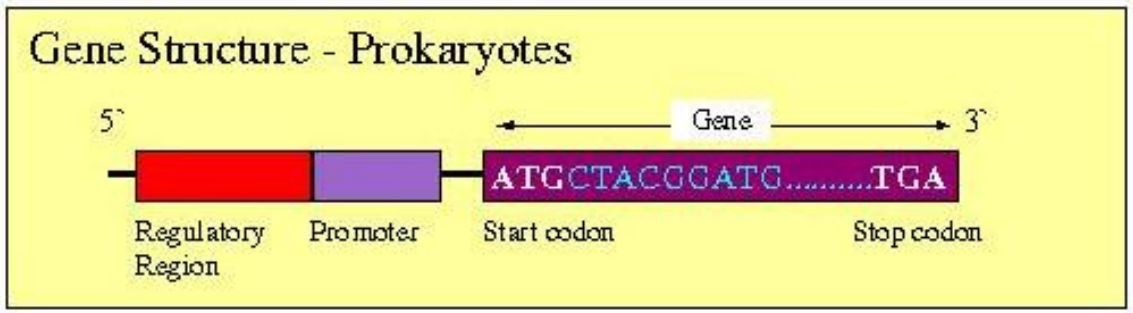
  – now predict it given more history
- "Fast guys finish __"

Sliding window Plot (length 120 nt)

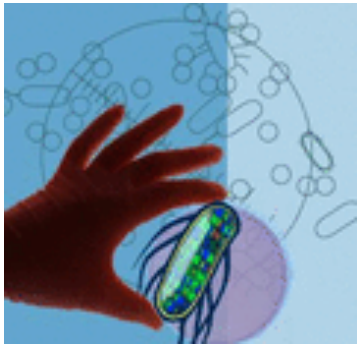


Genes as long ORFs with signals





Sequencing bacterial communities



Many microorganisms are uncultivated.

For survival and reproductive success, species of bacteria often rely on close relationships with other species.

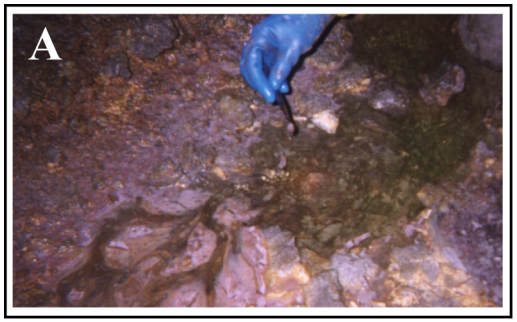
A collection of bacteria occupying the same physical habitat is called a 'community' toxic and non-toxic bacterial serotypes

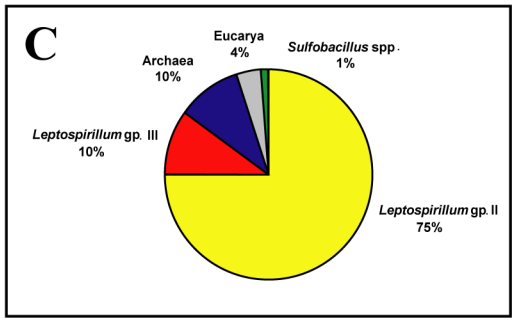
Biofilms have been implicated in numerous chronic infections including cystic fibrosis, otitis media and prostatitis.

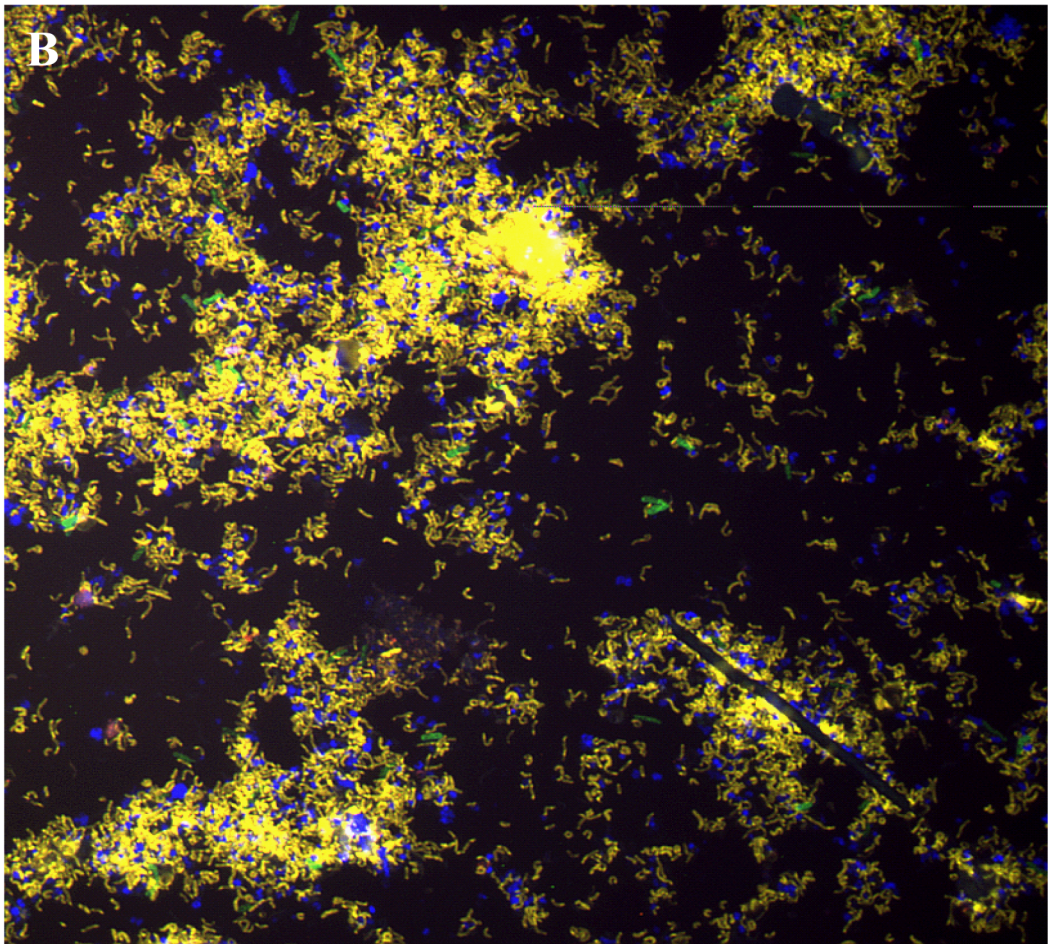
The sequencing techniques of a genomic DNA sample directly from the environment produce
shorter average sequence fragment length,
higher frequency of sequencing errors,
and the phylogenetic heterogeneity of the organisms in the sample

It presents additional challenges in computational gene finding

Photograph of the biofilm in the Richmond mine. B) Probes targeting bacteria (EUBmix; fluoresceinisothiocyanate (green)) and archaea (ARC915; Cy5 (blue)) were used in combination with aprobe targeting the Leptospirillum genus (LF655; Cy3 (red)). Overlap of red and green(yellow) indicates Leptospirillum cells and shows the dominance of Leptospirillum. C) Relative microbial abundances.



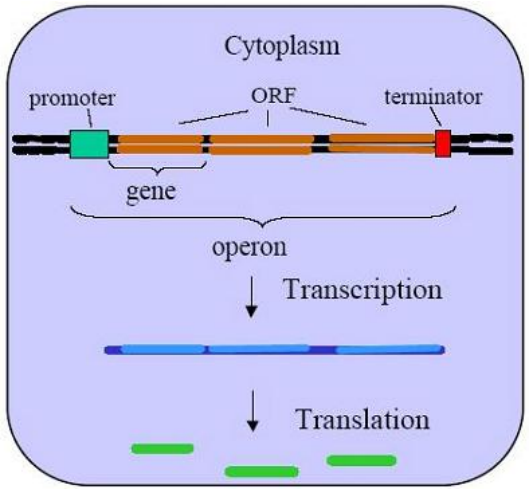




Nature (2004) 428 (6978) , p. 37–43

Community structure and metabolism through reconstruction of microbial genomes from the environment

Gene W. Tyson[1], Jarrod Chapman[3,4], Philip Hugenholtz[1], Eric E. Allen[1], Rachna J. Ram[1], Paul M. Richardson[4], Victor V. Solovyev[4], Edward M. Rubin[4], Daniel S. Rokhsar[3,4] & Jillian F. Banfield[1,2]



Fgenesb annotator (*Solovyev & Salamov*)

a complex pipeline for automatic annotation of bacterial genomes and metagenomic sequences has been developed to annotate millions bacterial sequences from acid mine drainage biofilm community.

Analysis of the predicted genes revealed the pathways for carbon and nitrogen fixation and energy generation

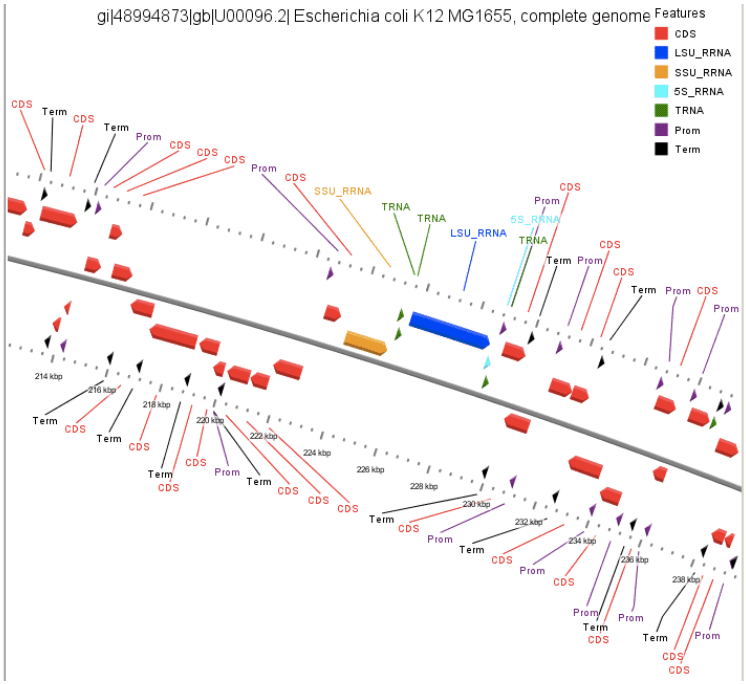
Fgenesb Bacterial Gene/Operon Prediction and Annotation Pipeline

Pipeline gene prediction algorithm is based on Markov chain models of coding regions and translation and termination sites.

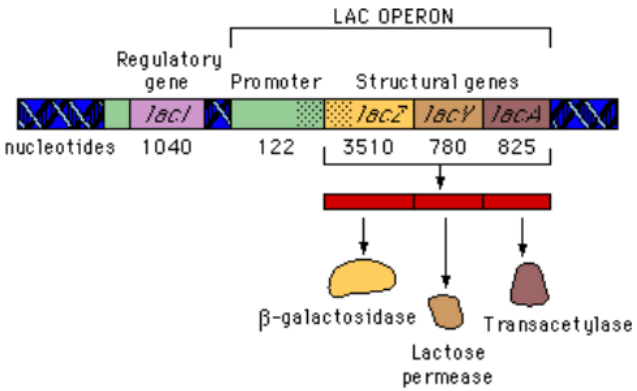
The parameters of gene prediction are self-learning, so the only input necessary is a set of sequences.

Pipeline identifies :

- rRNA and tRNA genes
- Protein coding genes
- Promoter and Terminator signals
- Operon structure



Annotate function of predicted proteins
Using COG, KEGG and NR databases



rRNA and tRNA annotation

1. Finds all potential ribosomal RRNA genes using BLAST against bacterial and/or archaeal RRNA databases and masks detected RRNA genes.

2. Predicts and masks tRNA genes using tRNAscan-SE program.

Genes and Operon identification

3. Initial predictions of long, slightly overlapping ORF are used as a starting point for calculating parameters of predictions. Iterates until stabilizes.

4. Automatically generates gene identification parameters as 5th-order in-frame Markov chains for coding regions, 2nd-order Markov models for region around start codon and upstream RBS site, Stop codon and probability distributions of ORF lengths. Uses these parameters for protein coding genes prediction

5. Predicts operons based only on distances between predicted genes.

Annotate genes comparing with user selected databases of known proteins

6. Runs blastp for predicted proteins against COG and KEGG databases and annotate genes/proteins by COGs and KEGG descriptions

7. Run blastp against NR for proteins having no COGs or KEGG hits and annotate genes/proteins by NR descriptions.

Promoters and Terminators prediction and improvement of operons assignment

8. Improve operon prediction using information on conservation of neighbor gene pairs in known genomes.

9. Predict potential promoters in the corresponding 5'-upstream region of predicted genes using dicriminant function with characteristics of sequence features of promoters (such as conserved motifs, binding sites and etc)

10. Predict pho-independent terminators as specific hairpins.

11. Refines operon predictions using predicted promoters and terminators

Fgenesb_annotator output:

```
1     1 Op  1  21/0.000   +    CDS        407 -      1747   1311  ## COG0593 ATPase involved in DNA
                           +    Term      1786 -      1823    3.2
                           +    Prom      1847 -      1906   10.5
2     1 Op  2   3/0.019   +    CDS       1926 -      3065   1237  ## COG0592 DNA polymerase
                           +    Term      3074 -      3122    9.1
                           +    Prom      3105 -      3164    4.0
3     2 Op  1   4/0.002   +    CDS       3193 -      3405    278  ## COG2501 Uncharacterized ACR
4     2 Op  2   4/0.002   +    CDS       3418 -      4545    899  ## COG1195 Recombinational DNA
5     2 Op  3  16/0.000   +    CDS       4578 -      6506   2148  ## COG0187 DNA gyrase B subunit
                           +    Term      6516 -      6551    4.7
```

>contig00033_scaffold00003_82492_86064 GENE 1 3 - 926 811 307 aa, chain + ## HITS:2 COG:BS_yumD KEGG:BCB4264_A5578 NR:ns
COG: BS_yumD COG0516 # Protein_GI_number: 16080266 # Func_class: F Nucleotide transport and metabolism # Function: IMP dehydrogenase/GMP reductase # Organism: Bacillus subtilis # 1 305 21 325 326 538 84.0 1e-153
KEGG: BCB4264_A5578 # Name: guaC # Def: guanosine 5'-monophosphate oxidoreductase (EC:1.7.1.7) # Organism: B.cereus_B426 Pathway: Purine metabolism [PATH:bcb00230] # 1 307 22 328 328 546 87.0 1e-154
SRTECDTTVEFGGRTFKLPVVPANMQTIIDERISIQLAEKNYFYIMHRFQPEKRLAFVRD
MKSRGLYASISVGVKEEEYTFVQQLAEENLVPEYITIDIAHGHSNAVIKMIQHIKQLLPG
SFVIAGNVGTPEAVRELENAGADATKVGIGPGKVCITKIKTGFGTGGWQLAALRWCAKAA
SKPIIADGGIRTHGDIAKSVRFGASMVMIGSLFAGHEESPGETVEVNGKLYKEYFGSASE
FQKGEKKNVEGKKMHVEYKGALEDTLIEMEQDLQSSISYAGGNKLSAIKNVDYVIVKNSI
FNGDKVY

Togenbank: A set of scripts to convert FgenesB output to GenBank and Sequin formats, for visualization in popular viewers like Artemis and for submitting annotated sequences to Genbank.

Comparative accuracy estimated on comprehensive tests:

Mavromatis et al. VOL.4 NO.6 | JUNE 2007 | **NATURE METHODS**:

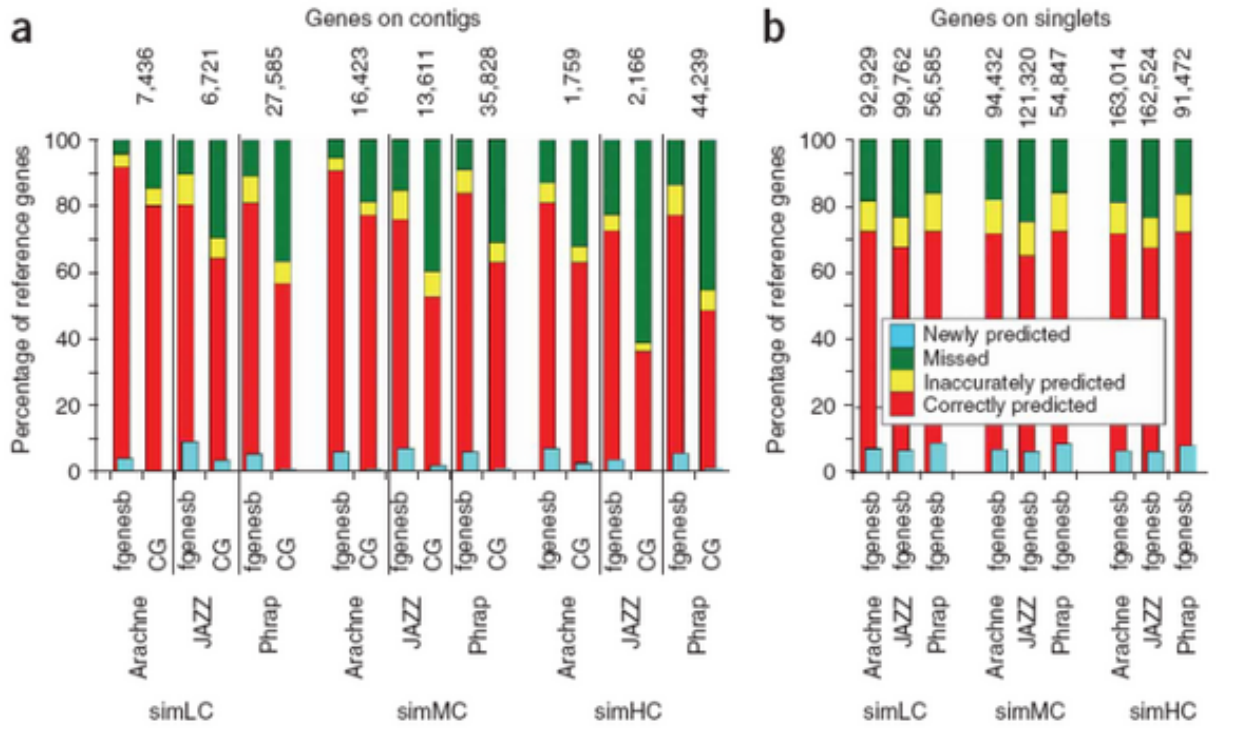


Figure 2 | Gene prediction in data sets. (**a**) Predicted genes on assembled sequences. (**b**) Predicted genes on unassembled reads. The combination of assembler/gene prediction method is shown on the *x* axis. The total number of original genes included in these sequences are shown on the top of the columns.

http://www.nature.com/naturemethods

Fgenesb correctly **identified 10-30% more reference genes on the contigs than the Critica-Glimer pipeline** in every data set

Test of modern gene predictors on difficult artificial shotgun sequences (700 bp fragments from a set of 216 bacterial genomes).

The sequences of real genes cover only part of each sequence (its 5'- or 3'-fragment)

	(Sn+Sp)/2
FgenesB	95.55
GeneMark	94.05
Matagene	91.65

Accuracy can be increased further by using protein similarity

a)Predicting weak/short coding regions or correcting frame shifts or other sequencing errors

b)Improving accuracy of start AUG identification

a) and b) can be optionally accounted by Fgenesb pipeline.

Fgenesb pipeline applications:

~ 340 published bacterial genome/metagenomic sequencing projects

Examples:

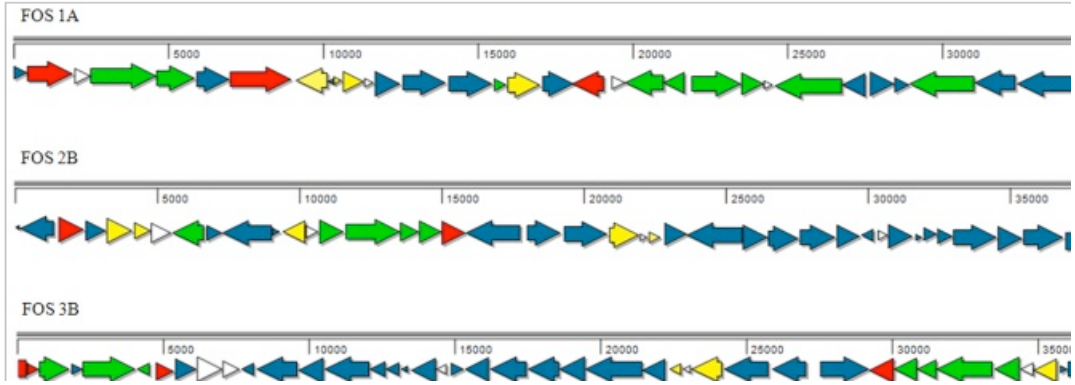
•New Hydrocarbon Degradation Pathways in the **Microbial Metagenome from Brazilian Petroleum Reservoirs**. **PLoS One**. 2014 Feb 26;9(2):e90087

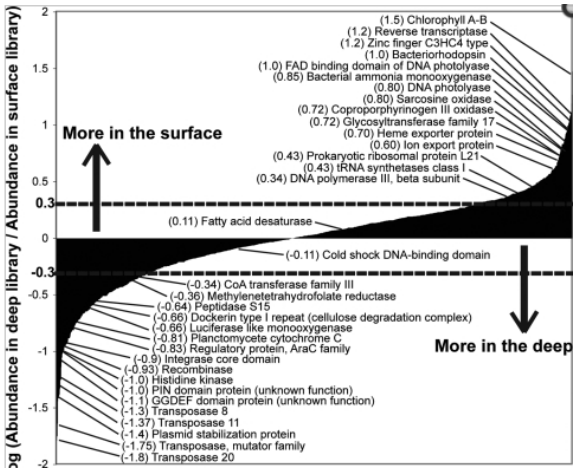
•Proteorhodopsin lateral gene transfer between **marine planktonic Bacteria and Archaea**. **Nature**, 2006, **439**, 847-850

•Comparative metagenomic analysis of **a microbial community residing at a depth of 4,000 meters** at station ALOHA in the North Pacific subtropical gyre. **Appl Environ Microbiol**. 2009 Aug;75(16):5345-55

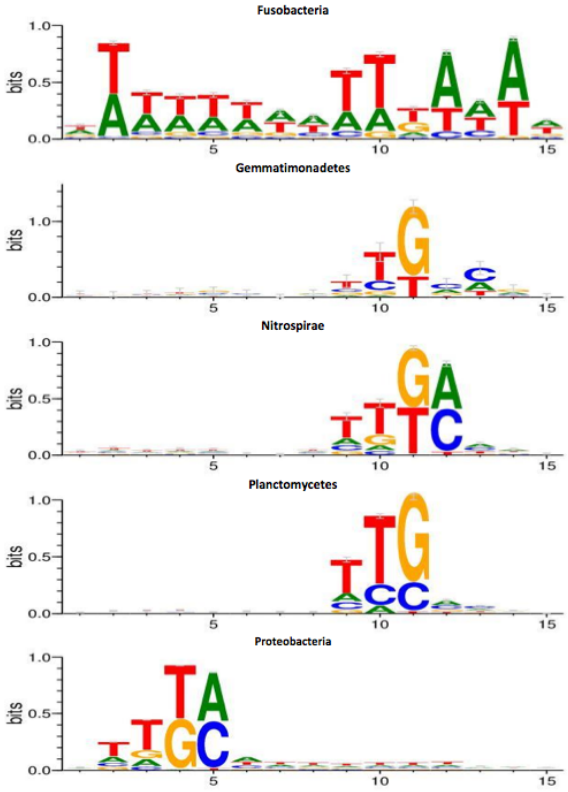
Bprom (promoter prediction) and FindTerm (terminator prediction) modules of Fgenesb used in

~ 800 papers on study of bacterial gene regulation

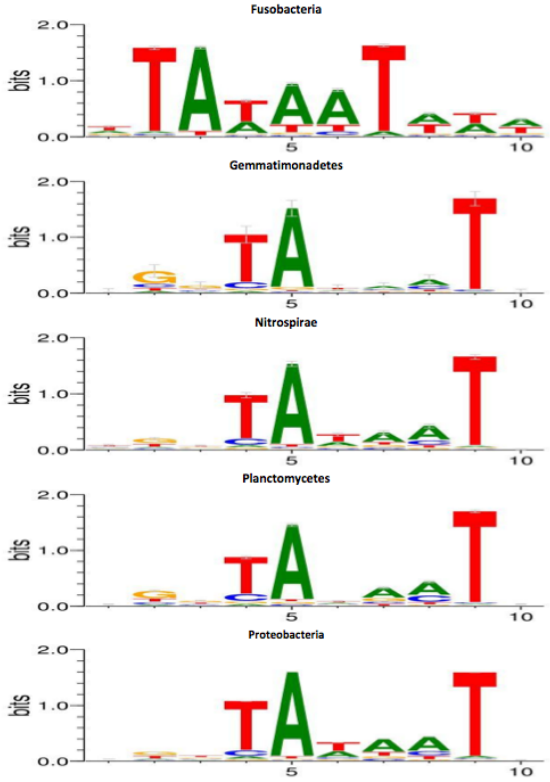




Current project to study structure of gene regulatory signals of distant bacterial groups (having sequenced ~ 10K bacterial genomes)



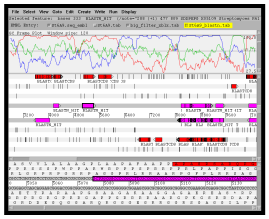
Promoter -35 box consensus



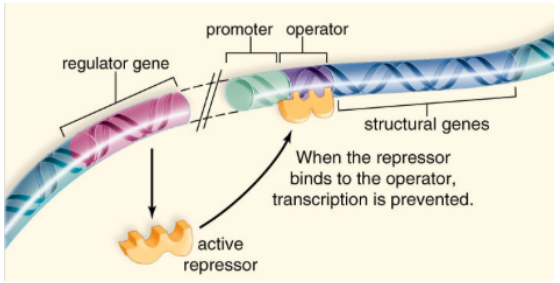
Promoter -10 box consensus

Knowledge of regulatory site variations is necessary for creating organism specific synthetic gene constructs

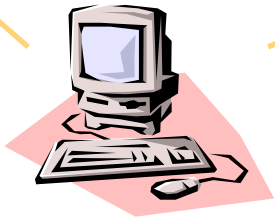
Current projects to study bacterial communities



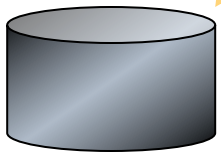
Assembling and Annotation of Bacterial community sequences

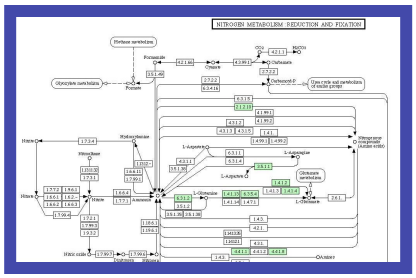


Analysis gene regulation in distant phylogenetic groups



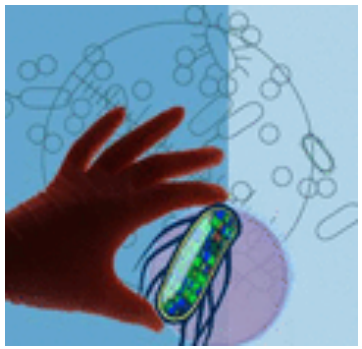
Building annotated Bacterial genomes database including complete genomes (~ 2K), draft genomes (~7K) and metagenomic sequences





Discovery new Metabolic Pathways

Why Sequence Microbes?



- By studying their DNA, scientists hope to find ways to use microbes to develop new pharmaceutical and agricultural products, energy sources, industrial processes, and solutions to a variety of environmental problems.

NIH Human Microbiome Project (2008) explores how complex communities of microbes interact with the human body to influence health and disease.

The oral microbiome consists of more than 600 different taxa of bacteria, viruses, fungi and protozoa

New ferments Biofuel New drugs