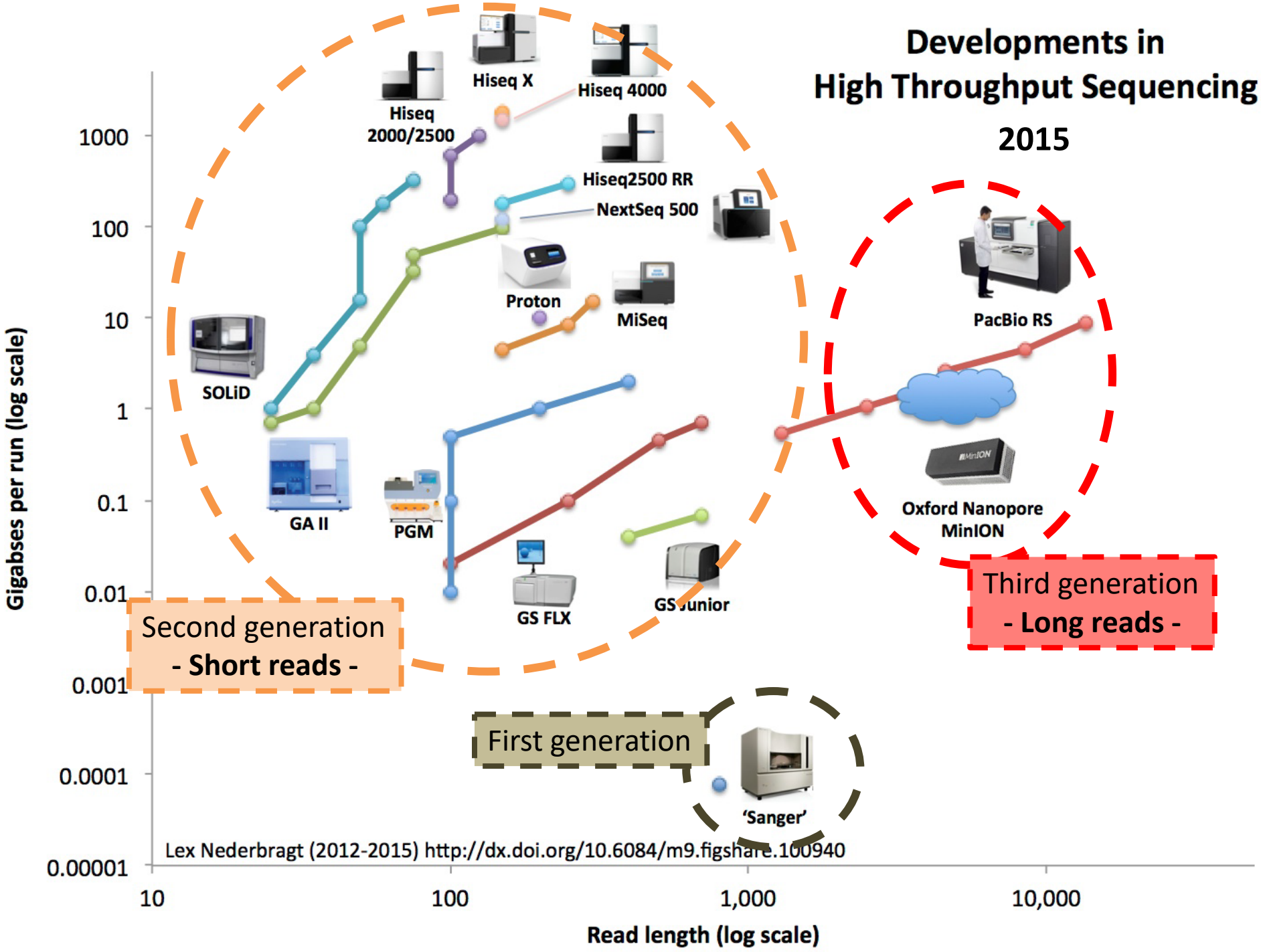
Assembly strategies for genomics and metagenomics

Linda van der Graaf – van Bloois

Faculty of Veterinary Medicine

Department of Biomolecular Health Sciences


Developments in
High Throughput Sequencing
2015
Hiseq X
Hiseq 4000
Hiseq
2000/2500
Hiseq2500 RR
NextSeq 500
Proton
MiSeq
SOLiD
GA II
PGM
GS FLX
GS Junior
PacBio RS
Oxford Nanopore
MinION
'Sanger'
Second generation
- Short reads -
Third generation
- Long reads -
First generation
Lex Nederbragt (2012-2015) http://dx.doi.org/10.6084/m9.figshare.100940
Gigabses per run (log scale)
1000
100
10
1
0.1
0.01
0.001
0.0001
0.00001
10
100
1,000
10,000
Read length (log scale)

Volume 6 Number 7 1979 Nucleic Acids Research

A strategy of DNA sequencing employing computer programs

R.Staden

MRC Laboratory of Molecular Biology, Hills Road, Cambridge CB2 2QH, UK

Received 23 March 1979

ABSTRACT

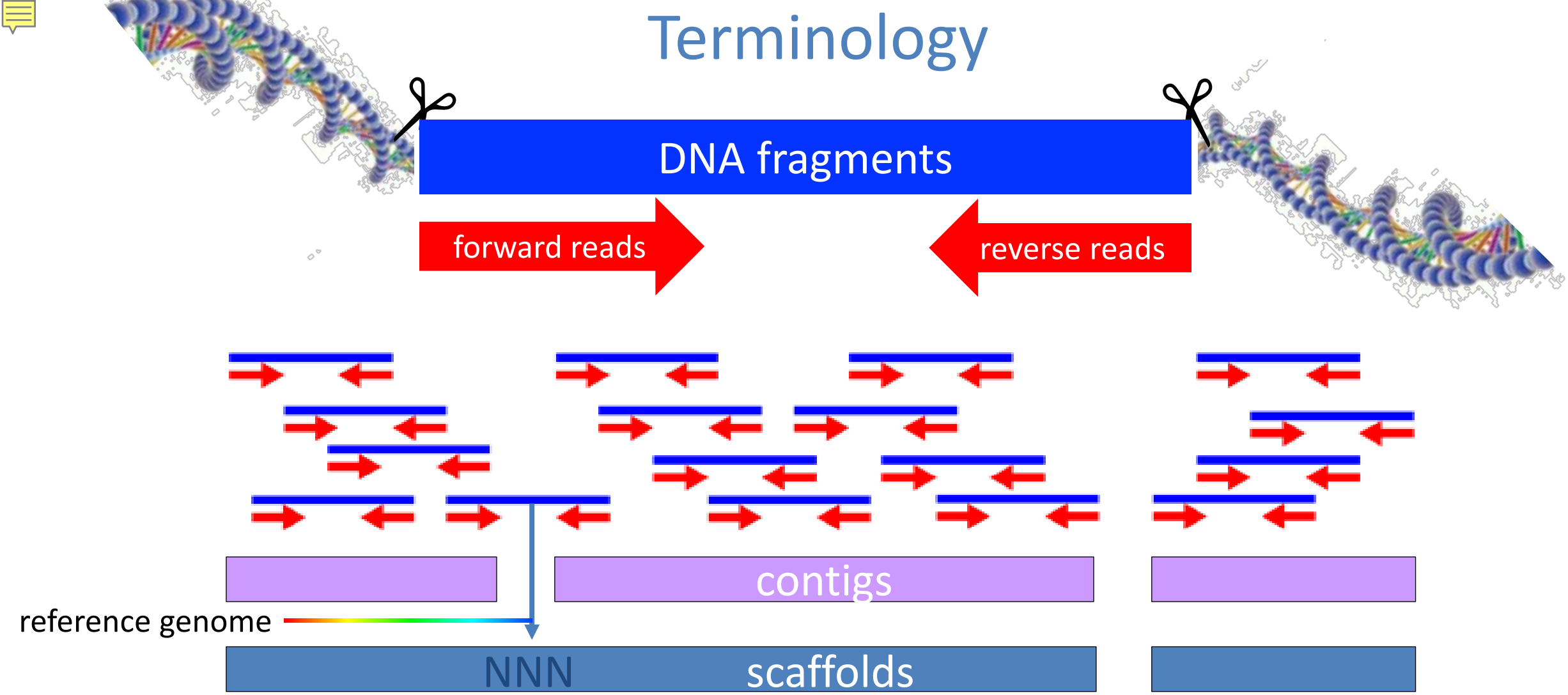
With modern fast sequencing techniques[1,2] and suitable computer programs it is now possible to sequence whole genomes without the need of restriction maps. This paper describes computer programs that can be used to order both sequence gel readings and clones. A method of coding for uncertainties in gel readings is described. These programs are available on request.

INTRODUCTION

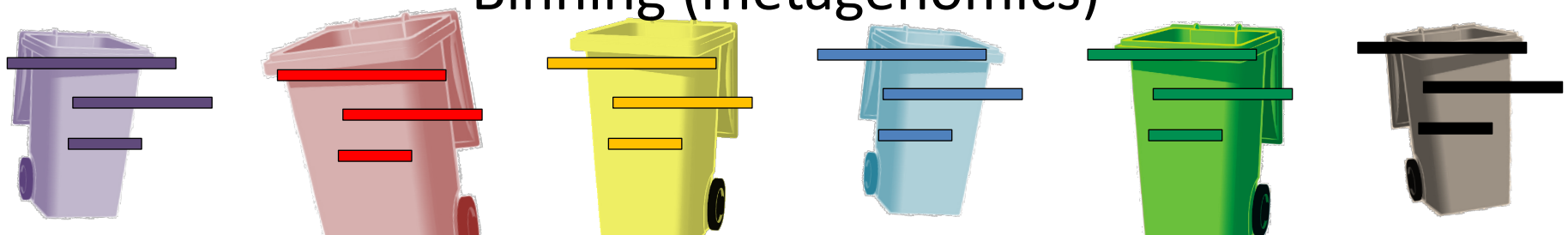
It became clear during the sequencing of bacteriophage ØX174 DNA[3] that it was necessary to use computers to handle and analyse the data. Later, while the very similar DNA sequence of bacteriophage G4 [4] was being determined, the computer was used to compare and align the G4 sequence with that of ØX174. Further advances in DNA sequencing methods[5,6] and cloning techniques have been made and work is in progress in a number of laboratories on sequences many times the length of ØX174.

The continuing rapid fall in the cost of computer components is making

Terminology

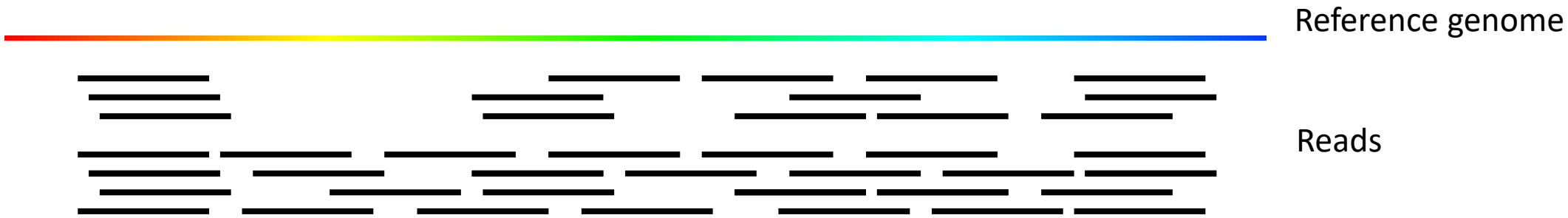


Binning (metagenomics)

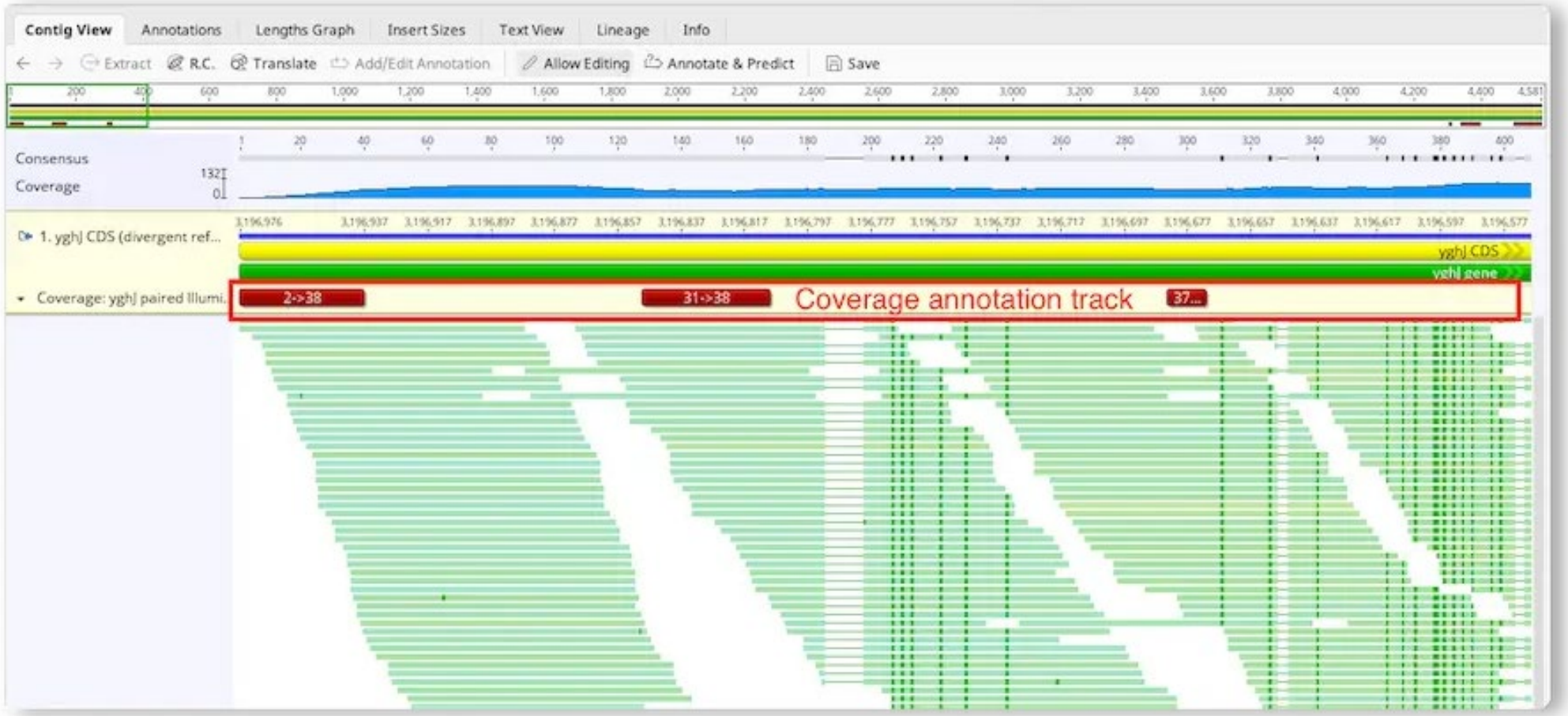


Assembly strategies

1. Reference-guided assembly
   - Mapping reads to a reference



   - Infer consensus


Contig View
Annotations
Lengths Graph
Insert Sizes
Text View
Lineage
Info
Extract
R.C.
Translate
Add/Edit Annotation
Allow Editing
Annotate & Predict
Save
Consensus
Coverage
1. yghJ CDS (divergent ref...
yghJ CDS
yghJ gene
Coverage: yghJ paired Illumi...
2->38
31->38
Coverage annotation track
37...

Assembly strategies

2. *De novo* assembly
   – Construct genome sequence from overlap between reads

Reference-guided assembly

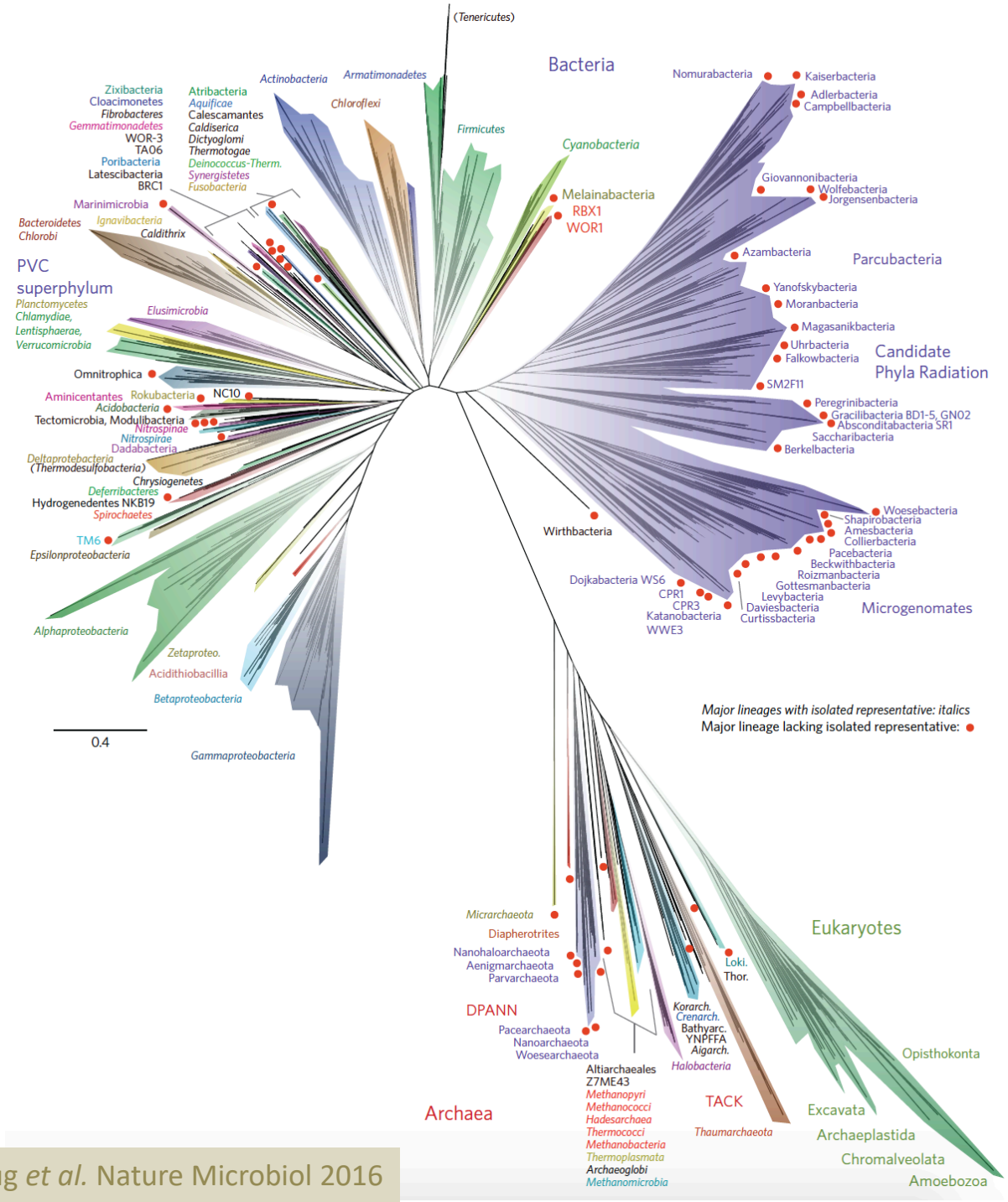
- We need a closely related reference genome:
  - Same species
- Reference genome leads to biases in the assembly
  - Genome structure
    - Insertions, deletions, rearrangements
- If we only have a distantly related reference:

→ we need to be more permissive when aligning reads

→ this will lead to more possible errors

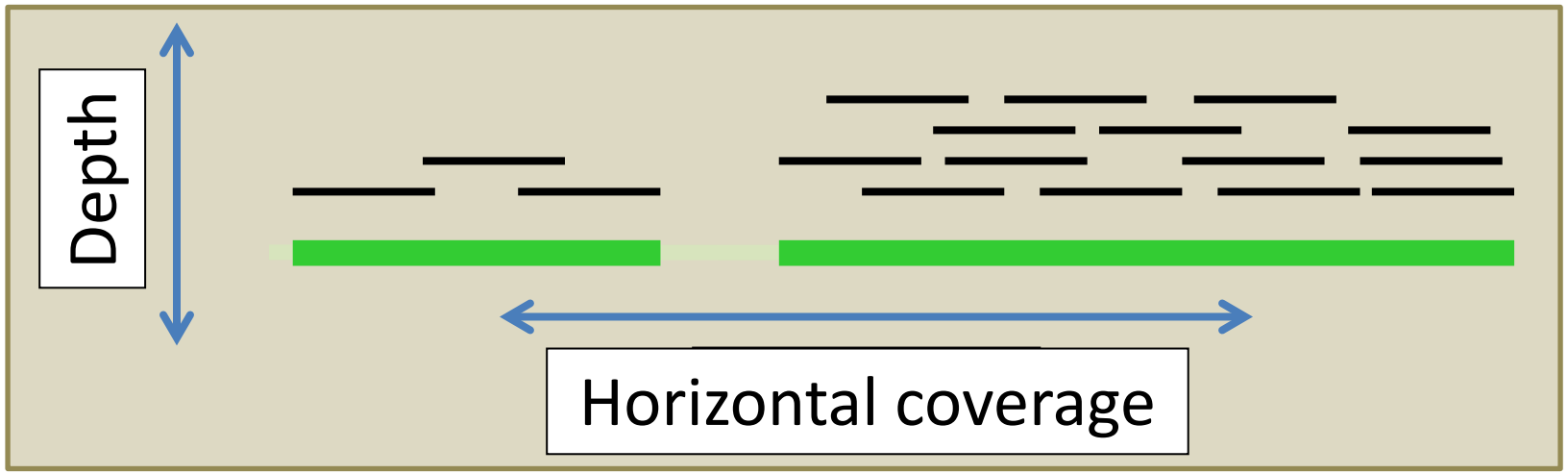
Tree of life

- *Italic: well-characterized clades*
- Red dot: clades with no representatives



Hug *et al.* Nature Microbiol 2016

Coverage *(ambiguous term!)*



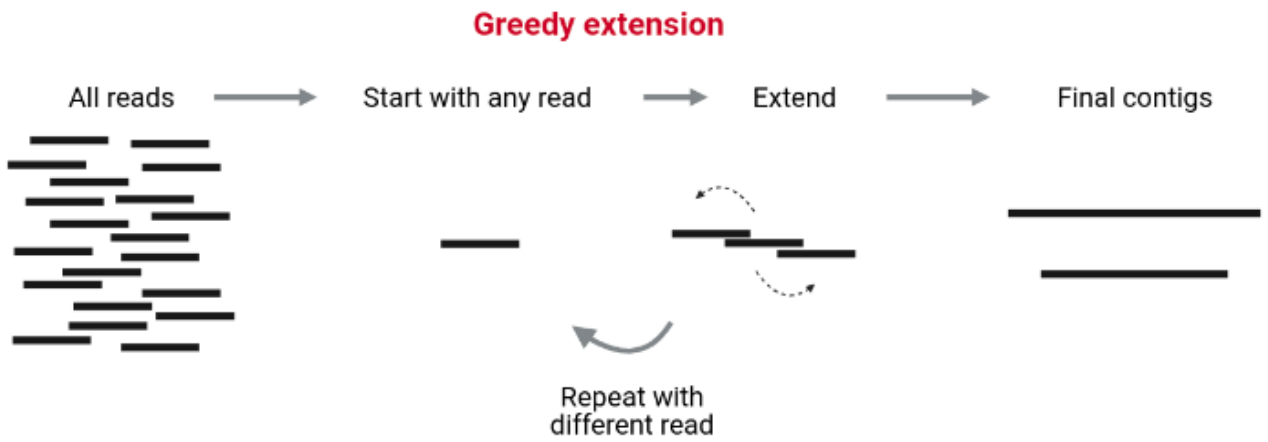
- Depth (vertical coverage):
  - Average number of reads that cover each nucleotide in the assembly
  - Corresponds to the abundance of a sequence in the sample
- Horizontal coverage:
  - Percentage of the target genome that has been (re)covered after the assembly

De novo sequence assembly

- Requires sufficient sequencing (coverage x depth)
- Breaks on repeats and low-coverage regions
- Algorithms
  - Greedy extension
  - Overlap-layout-consensus
  - De Bruijn graph

Greedy extension

1. Sequences (reads) (reads/contigs)
2. Pairwise all-vs-all similarities
3. Find best matching pair
4. Collapse/assemble

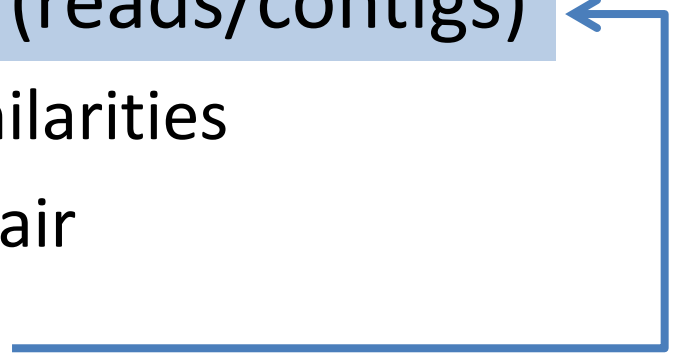


Source: https://carpentries-lab.github.io/metagenomics-analysis/04-assembly/

Greedy extension

1. Sequences (reads)
2. Pairwise all-vs-all similarities
3. Find best matching pair
4. Collapse/assemble

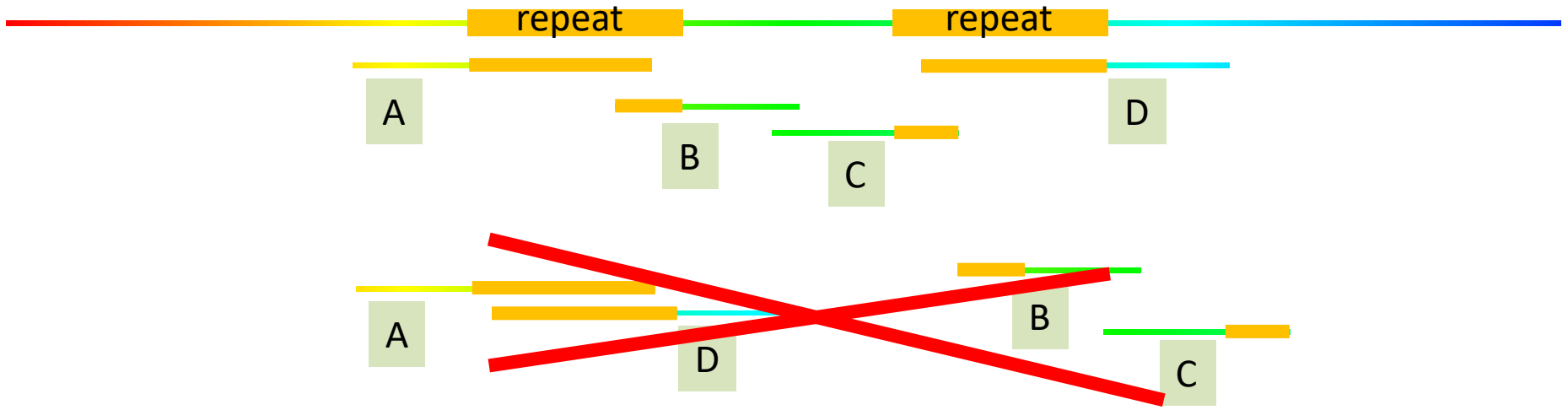
(reads/contigs)



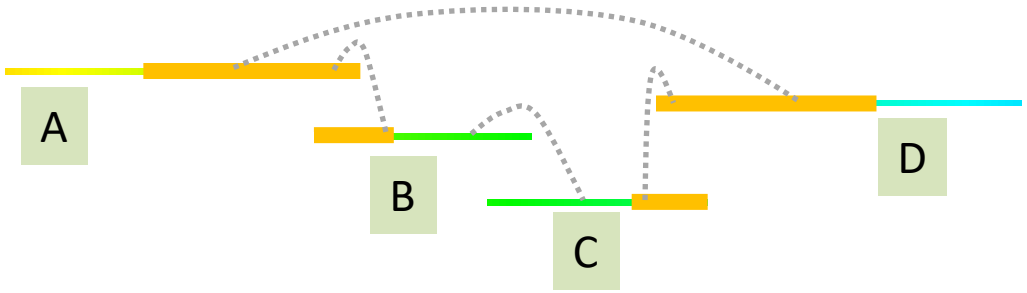
- Works well for a few long reads (1st generation)

- Does not work for many short reads (2nd gen)
  - All-vs-all comparisons are computationally "expensive" to calculate
  - Short sequences may have multiple best matches

Repetitive sequences

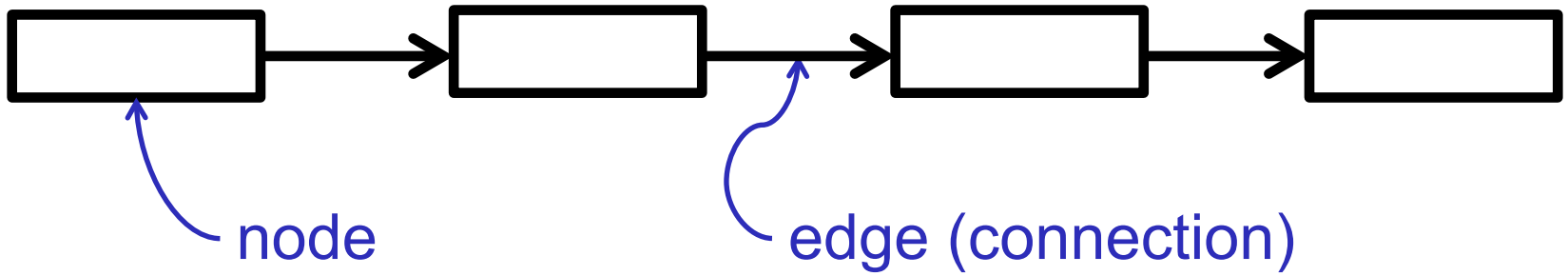


- Reads A-D are from a region with two long repeats
  - Long means longer than the read length
- Greedy approach would first join A-D with the largest overlap and then place B-C in a separate contig → this is wrong
- Resolving this requires a global view of all the possibilities before joining two reads: a graph



Assembly with a graph-approach

- A graph contains nodes and edges

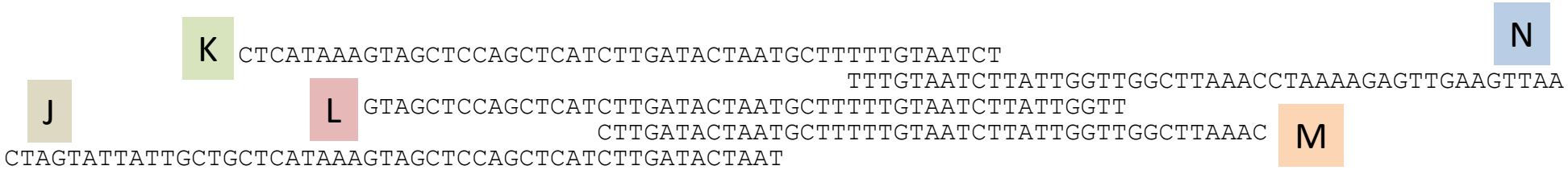


- Two types of graphs are often used in sequence assembly
  - Overlap-layout-consensus
  - De Bruijn Graph

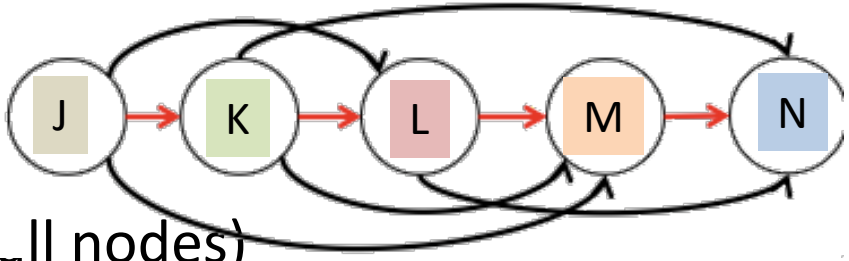
Nicolaas Govert (Dick) de Bruijn
1918-2012

Overlap layout consensus

1. Identify all overlaps between reads
   - Use cutoffs: minimum overlap and percent identity



2. Make graph of overlap connections
   - Nodes: reads
   - Edges: overlaps



3. Find path that contains all data (= all nodes)
   - Hamiltonian path
   - No efficient algorithm available
4. Determine consensus at each position

CTAGTATTATTGCTGCTCATAAAGTAGCTCCAGCTCATCTTGATACTAATGCTTTTTGTAATCTTATTGGTTGGCTTAAACCTAAAAGAGTTGAAGTTAA

De Bruijn graph

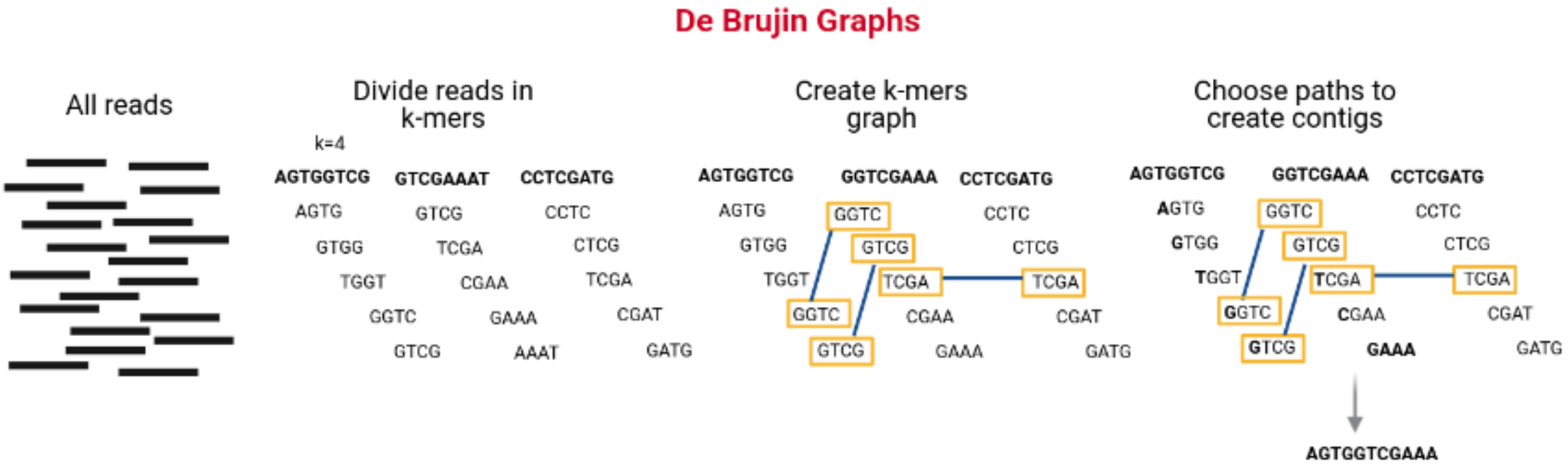
1. Break up all sequencing reads into shorter words of length *k* (*k*-mers)
   - Number of *k*-mers ≈ number of nucleotides in genome

k = 29

De Bruijn graph

2. Make graph of sequential *k*-mers in reads
   - Nodes: *k*-mers
   - Edges: linked in the data (reads)

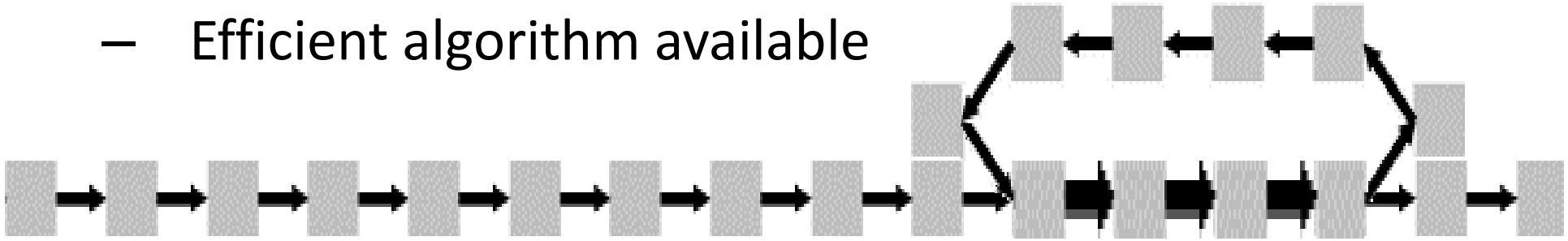
De Bruijn graph



Source: https://carpentries-lab.github.io/metagenomics-analysis/04-assembly/

De Bruijn graph

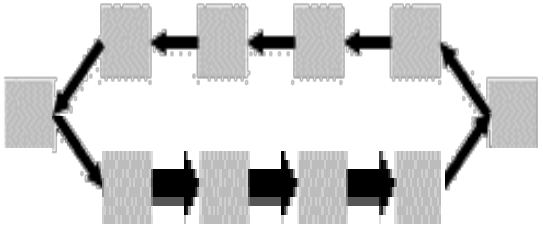
3. Find path that contains all data (= all edges)
   - Eulerian path
   - Efficient algorithm available



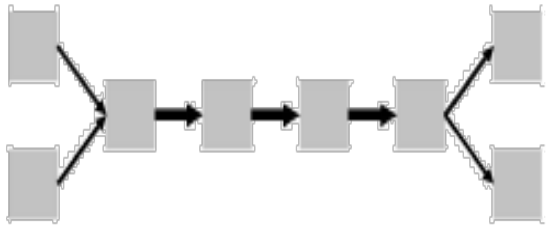
- In an optimal sequencing run of a repeat-less genome, there is one path connecting all nodes
- In practice (especially in metagenomes) there are many possible structures in the graph
- Edge width represents the number of linking reads (depth)

Possible structures in De Bruijn graphs

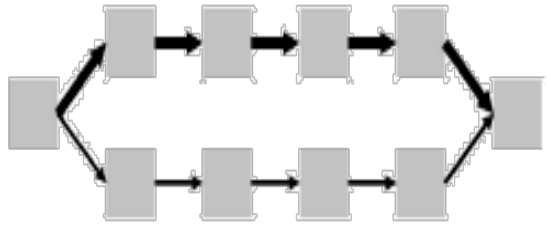
- Cycle: path converges on itself
  - Repeated region on the same contig



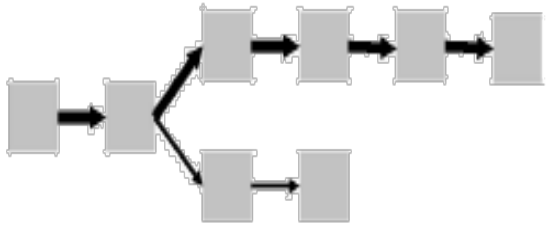
- Frayed rope: converge then diverge
  - Repeated region on different contigs



- Bubble: paths diverge then converge
  - Sequencing error in the middle of a read
  - Polymorphisms



- Spur: short dead-ends
  - Sequencing error at the end of a read
  - Zero coverage shortly after end of repeat



Short-read assembly tool: SPAdes

JOURNAL OF COMPUTATIONAL BIOLOGY
Mary Ann Liebert, Inc. publishers
Journals | Search | Alerts

J Comput Biol. 2012 May; 19(5): 455–477.
doi: 10.1089/cmb.2012.0021
PMCID: PMC3342519
PMID: 22506599

SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing

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. Pevzner[1,4]

▸ Author information ▸ Copyright and License information PMC Disclaimer

Abstract

Go to: ▸

The lion's share of bacteria in various environments cannot be cloned in the laboratory and thus cannot be sequenced using existing technologies. A major goal of single-cell genomics is to complement gene-centric metagenomic data with whole-genome assemblies of uncultivated organisms. Assembly of single-cell data is challenging because of highly non-uniform read coverage as well as elevated levels of sequencing errors and chimeric reads. We describe SPAdes, a new assembler for both single-cell and standard (multicell) assembly, and demonstrate that it improves on the recently released E+V−SC assembler (specialized for single-cell data) and on popular assemblers Velvet and SoapDeNovo (for multicell data). SPAdes generates single-cell assemblies, providing information about genomes of uncultivatable bacteria that vastly exceeds what may be obtained via traditional metagenomics studies. SPAdes is available online (http://bioinf.spbau.ru/spades). It is distributed as open source software.

Key words: assembly, de Bruijn graph, single cell, sequencing, bacteria

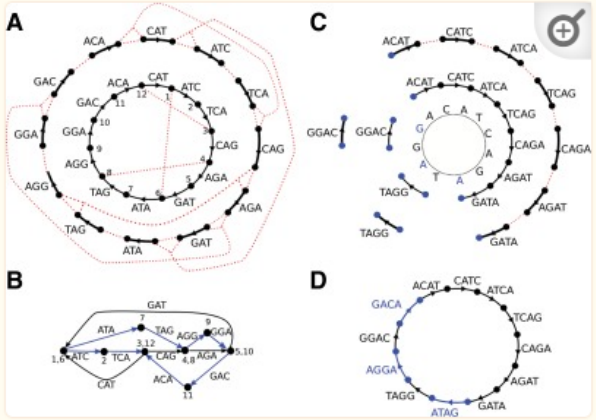


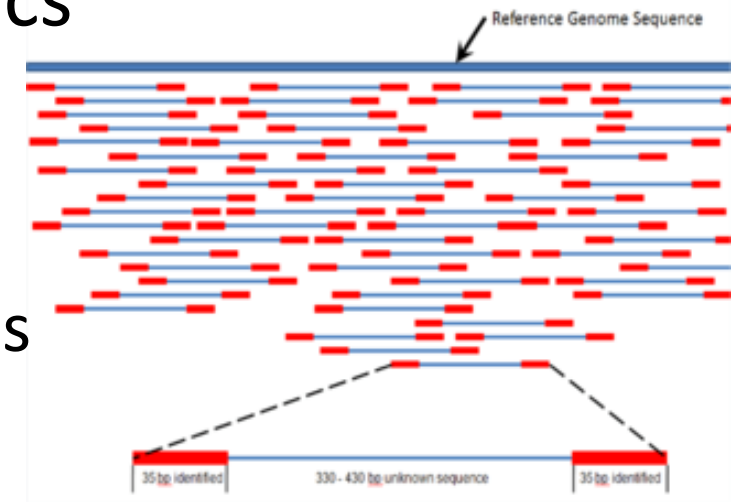
FIG. 2.

Standard and multisized de Bruijn graph. A circular GENOME CATCAGATAGGA is covered by a set READS consisting of nine 4-mers, {ACAT, CATC, ATCA, TCAG, CAGA, AGAT, GATA, TAGG, GGAC}. Three out of 12 possible 4-mers from GENOME are missing from READS (namely {ATAG,AGGA,GACA}), but all 3-mers from GENOME are present in READS. (**A**) The outside circle shows a separate black edge for each 3-mer from READS. Dotted red lines indicate vertices that will be glued. The inner circle shows the result of applying some of the glues. (**B**) The graph DB(READS, 3) resulting from all the glues is tangled. The three h-paths of length 2 in this graph (shown in blue) correspond to h-reads ATAG, AGGA, and GACA. Thus $\text{READS}_{3,4}$ contains all 4-mers from GENOME. (**C**) The outside circle shows a separate edge for each of the nine 4-mer reads. The next inner circle shows the graph DB(READS, 4), and the innermost circle represents the GENOME. The graph DB(READS, 4) is fragmented into 3 connected components. (**D**) The multisized de Bruijn graph DB(READS, 3, 4).

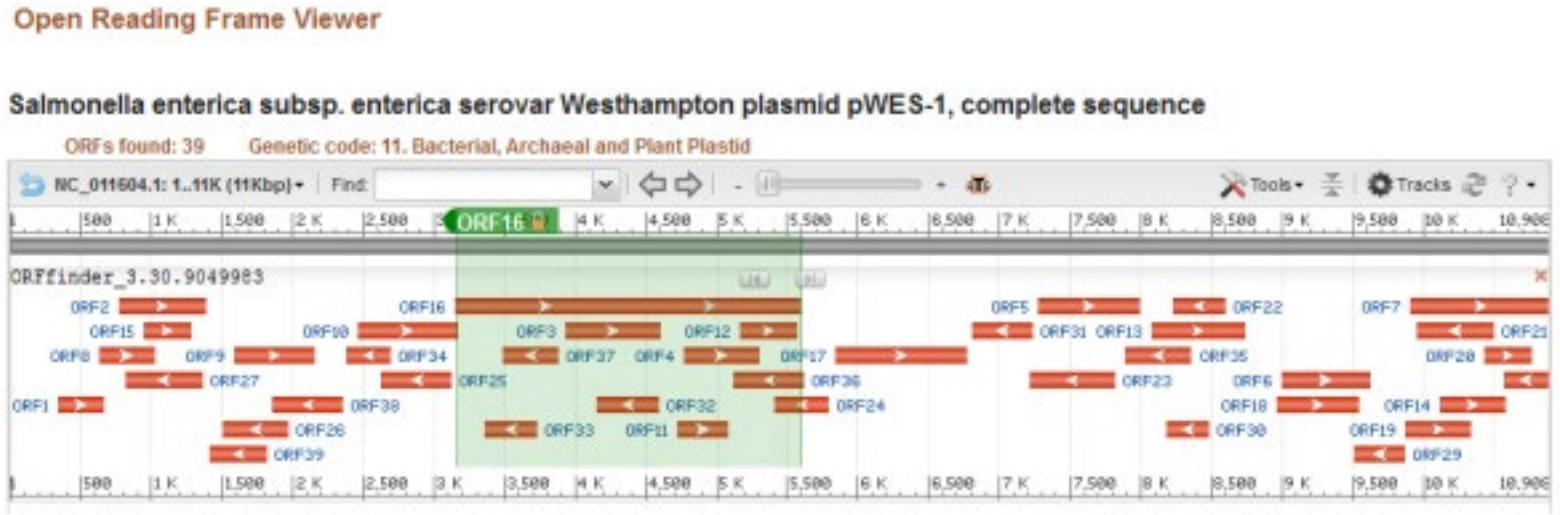
In the de Bruijn graph DB(READS, k), an h-path passing through n vertices

How to assess assembly quality?

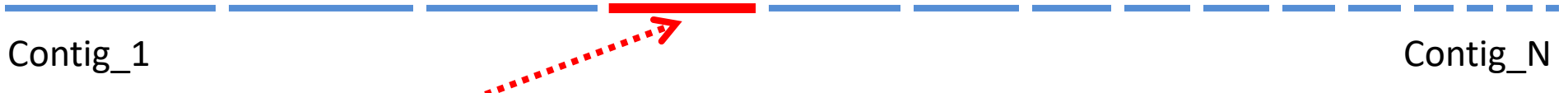
- General assembly performance metrics
  - Length of longest contig
  - N50, N80
  - Percentage of matched paired-end reads



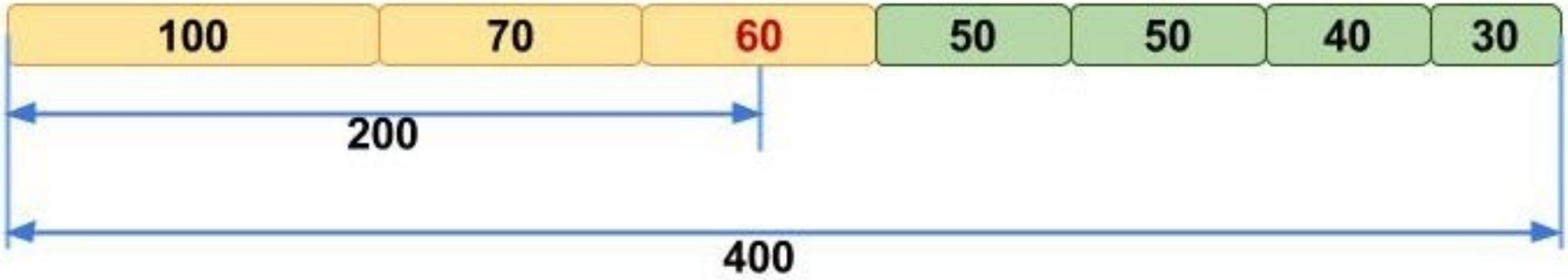
- Length distribution of open reading frames (ORFs)



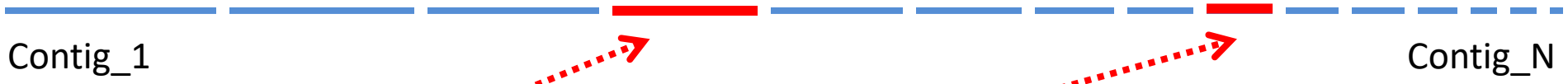
Assembly length statistics: N50 and N80



- Sort all contigs from long to short
- N50 = length of contig at 50% of cumulative length



Assembly length statistics: N50 and N80



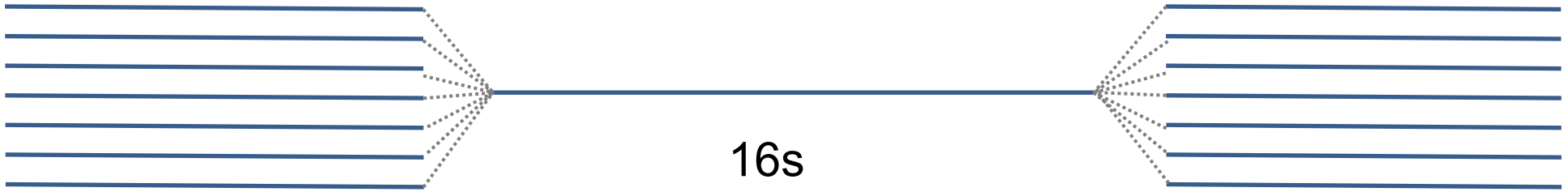
- Sort all contigs from long to short
- N50 = length of contig at 50% of cumulative length
- N80 = length of contig at 80% of cumulative length
- This measure is less meaningful for metagenomes from highly diverse communities

Quast

Genome assembly evaluation tool

QUAST stands for QUality ASsessment Tool. It evaluates genome/metagenome assemblies by computing various metrics. The current QUAST toolkit includes the general QUAST tool for genome assemblies, MetaQUAST, the extension for metagenomic datasets, QUAST-LG, the extension for large genomes (e.g., mammalians), and Icarus, the interactive visualizer for these tools.

Repeats have multiple sinks/sources



Genomic Rearrangements at *rrn* Operons in Salmonella

R. Allen Helm,* Alison G. Lee,* Harry D. Christman* and Stanley Maloy*,†,1

**Department of Microbiology, University of Illinois, Urbana, Illinois 61801 and †Center for Microbial Sciences, San Diego State University, San Diego, California 92182-4614*

Manuscript received September 19, 2002
Accepted for publication January 10, 2003

ABSTRACT

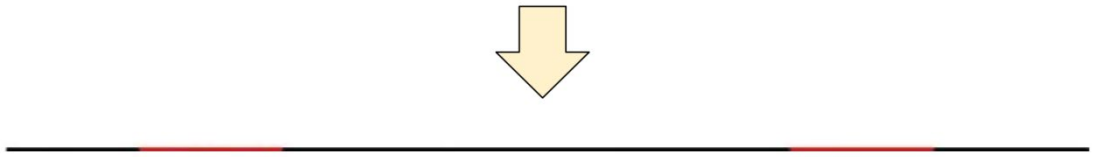
Most Salmonella serovars are general pathogens that infect a variety of hosts. These "generalist" serovars cause disease in many animals from reptiles to mammals. In contrast, a few serovars cause disease only in a specific host. Host-specific serovars can cause a systemic, often fatal disease in one species yet remain avirulent in other species. Host-specific Salmonella frequently have large genomic rearrangements due to recombination at the ribosomal RNA (*rrn*) operons while the generalists consistently have a conserved chromosomal arrangement. To determine whether this is the result of an intrinsic difference in recombination frequency or a consequence of lifestyle difference between generalist and host-specific Salmonella we determined the frequency of rearrangements *in vitro*. Using *lacZ* genes as portable regions of homology

Helm *et al.*, *Genetics* 2003

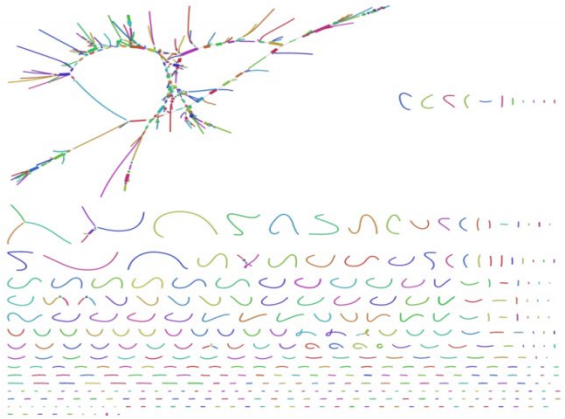
Long-read sequencing

short reads: contig breaks

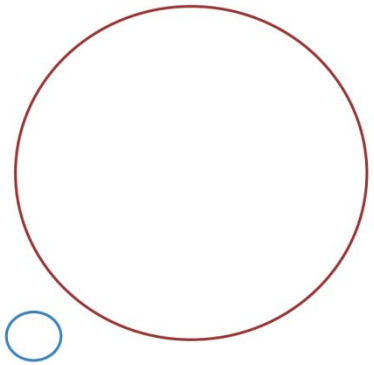
long reads span repeats



resulting in complete assemblies



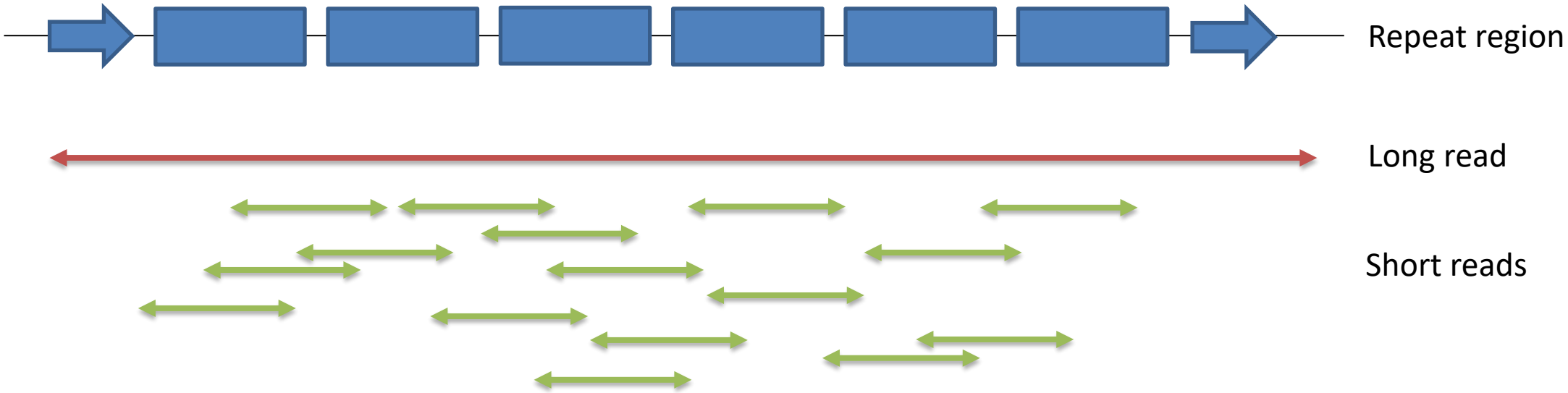
short read assembly



long read assembly

Recognizing what is plasmid and what is chromosomal DNA

Hybrid assembly



- Long-read assembly is used as scaffold

- Short-reads are used to polish the long-read assembly
  (SNPs, homopolymer tracts etc)

Hybrid assembly tool: UniCycler

PLOS | COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads

Ryan R. Wick*, Louise M. Judd, Claire L. Gorrie, Kathryn E. Holt

Department of Biochemistry and Molecular Biology, Bio21 Molecular Science and Biotechnology Institute, The University of Melbourne, Victoria, Australia

* rrwick@gmail.com

Check for updates

OPEN ACCESS

Citation: Wick RR, Judd LM, Gorrie CL, Holt KE (2017) Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. PLoS Comput Biol 13(6): e1005595. https://doi.org/10.1371/journal.pcbi.1005595

Editor: Adam M. Phillippy, National Human Genome Research Institute, UNITED STATES

Received: January 13, 2017

Accepted: May 22, 2017

Published: June 8, 2017

Data Availability Statement: All reference genomes used for simulation data are available from the NCBI assembly database (accession numbers in Table 1). *E. coli* sequence files are publicly available (links in Table 2). *Klebsiella* sequence files are available from the NCBI Sequence Read Archive database (accession numbers ERX1087708, ERX1087759, SRX2874872 and SRX2874871).

Abstract

The Illumina DNA sequencing platform generates accurate but short reads, which can be used to produce accurate but fragmented genome assemblies. Pacific Biosciences and Oxford Nanopore Technologies DNA sequencing platforms generate long reads that can produce complete genome assemblies, but the sequencing is more expensive and error-prone. There is significant interest in combining data from these complementary sequencing technologies to generate more accurate "hybrid" assemblies. However, few tools exist that truly leverage the benefits of both types of data, namely the accuracy of short reads and the structural resolving power of long reads. Here we present Unicycler, a new tool for assembling bacterial genomes from a combination of short and long reads, which produces assemblies that are accurate, complete and cost-effective. Unicycler builds an initial assembly graph from short reads using the *de novo* assembler SPAdes and then simplifies the graph using information from short and long reads. Unicycler uses a novel semi-global aligner to align long reads to the assembly graph. Tests on both synthetic and real reads show Unicycler can assemble larger contigs with fewer misassemblies than other hybrid assemblers, even when long-read depth and accuracy are low. Unicycler is open source (GPLv3) and available at github.com/rrwick/Unicycler.

This is a *PLOS Computational Biology* Software paper.

Introduction

Bacterial genomics is currently dominated by Illumina sequencing platforms. Illumina reads are accurate, have a low cost per base and have enabled widespread use of whole genome sequencing. However, much Illumina sequencing uses short fragments (500 bp or less) that

Use of Unicycler:

- De novo assembly with only long-reads (miniasm assembly)
  - Other assemblers: **Flye**, Canu
- De novo assembly with both long- and short-reads
- Most powerful as hybrid assembler:
  - Use long-read assembly as scaffold, and correct sequence with Illumina short-reads
    - Racon polishing
- Polishing always necessary

Metagenomics

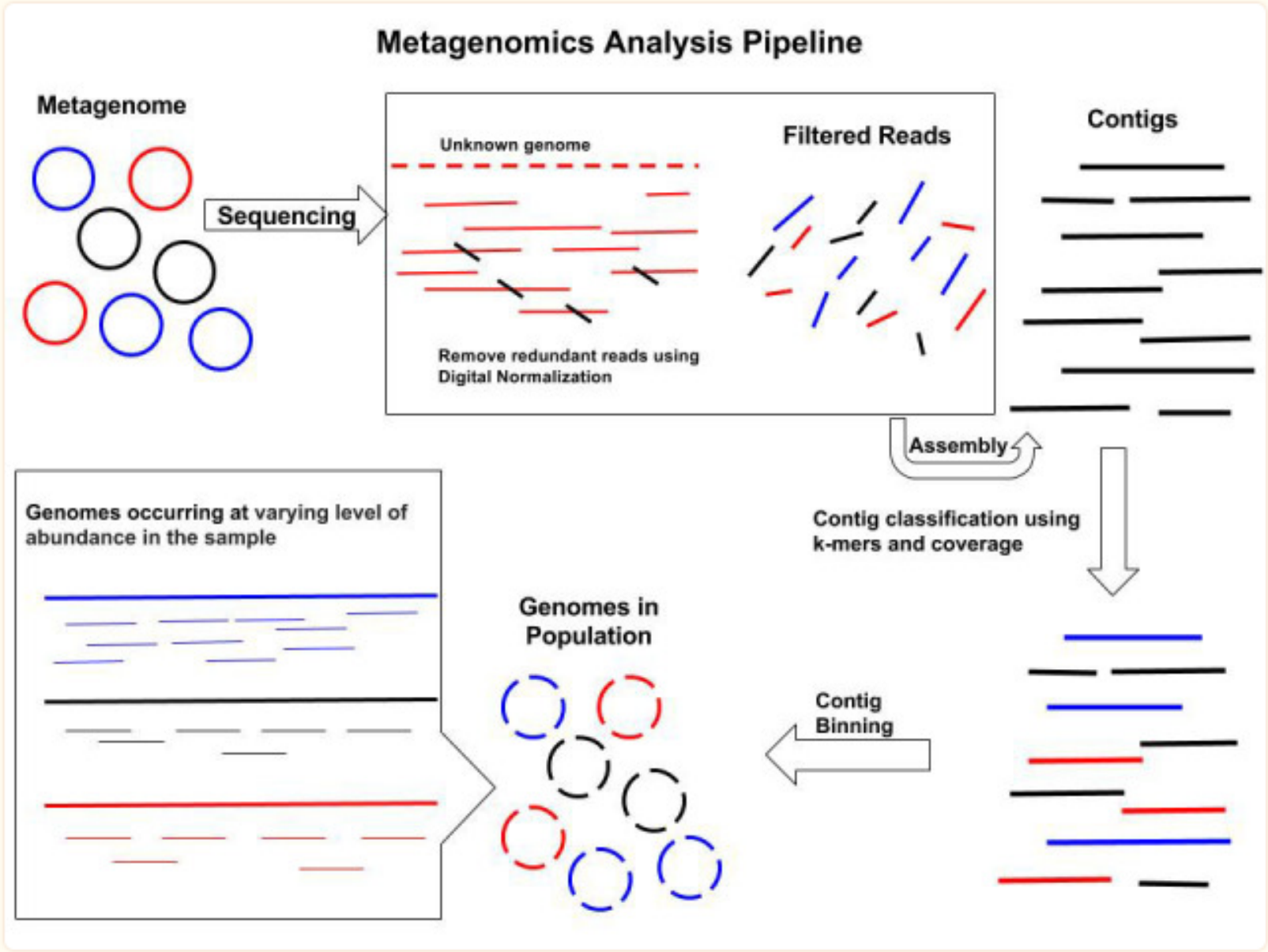
Metagenomics is the study of genetic material recovered directly from environmental or clinical samples by a method called sequencing

The study examines the genomic composition of an entire organism, including each of the microbes that exist within it

Metagenomics data from many biomes

MGnify

MG-RAST
metagenomics analysis server

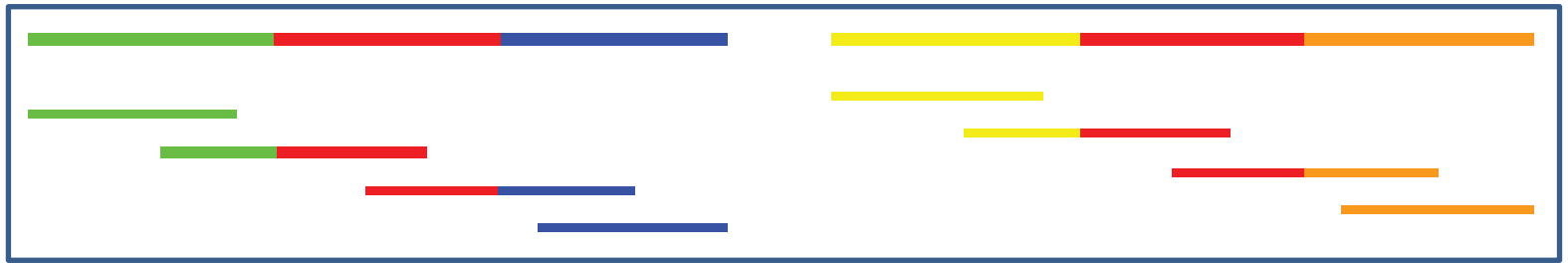


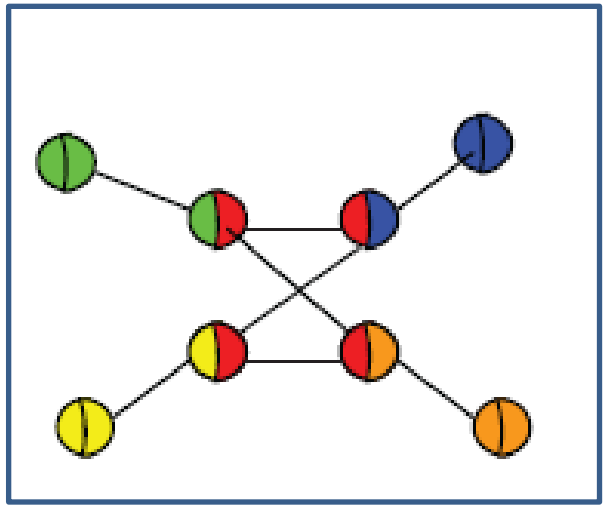
Ghurye J S, Cepeda-Espinoza V, Pop M. Metagenomic Assembly: Overview, Challenges and Applications. Yale J Biol Med. 2016 Sep 30;89(3):353-362.

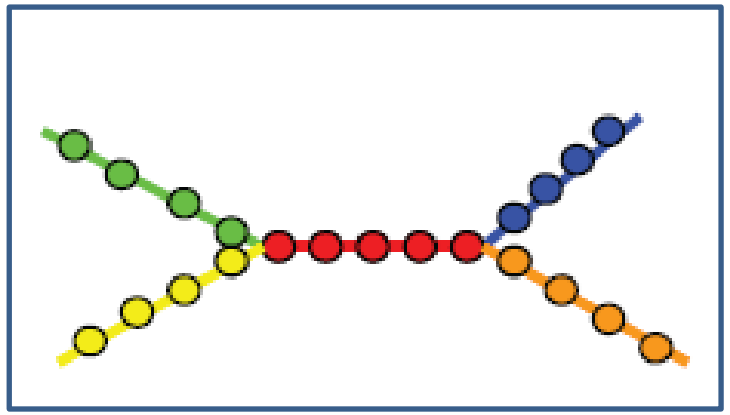
Genome versus metagenome sequencing

- Depending on coverage
  - Expect single sequence
  - Contiguous sequence
  - Even read depth
- Depending on diversity
  - Expect many sequences
  - Fragmented sequences
  - Varying read depth
- Identify sequencing errors by low coverage
  - Clonal sequence
- Sequencing errors or natural diversity?
  - Natural microdiversity
- Repeats consist of duplicated genes and conserved domains
- Repeats also include closely related strains, horizontal transfer, etc.

Assembly problem: Chimera







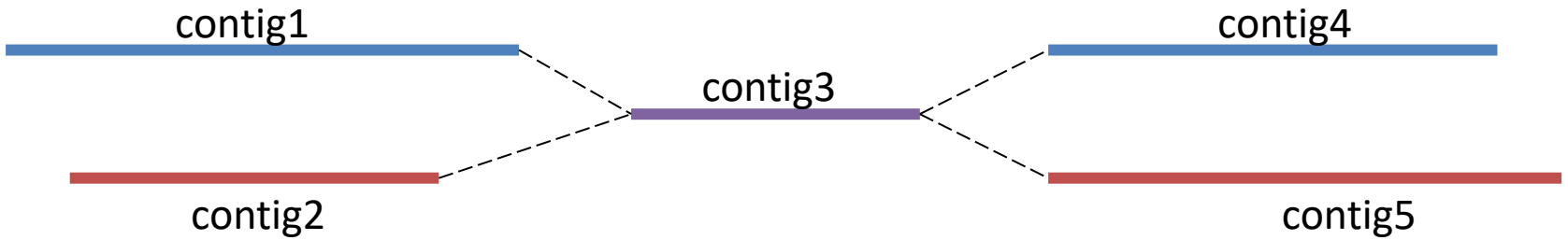
In overlap-layout-consensus

De Bruijn graph

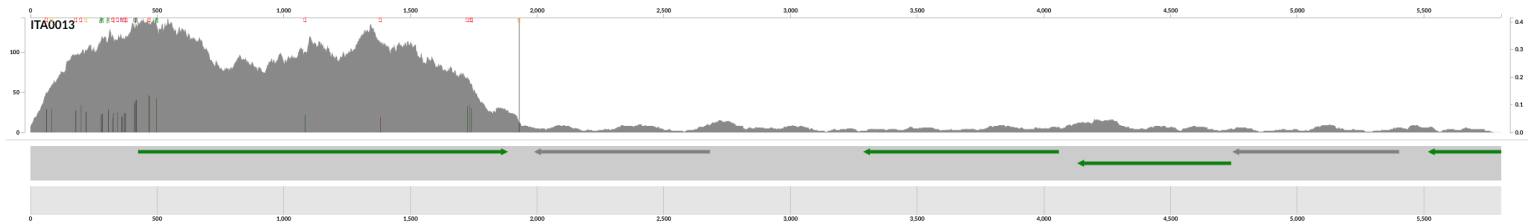
Li BFG 2012

How big is the chimerization problem?

- Assembly algorithms include “chimera protection”
  - Break contigs at ambiguities



- Chimerization is more frequent between closely related strains
  - Similar sequences
- Investigate the effect of chimerization:
  - Use different assembly parameters and assess your final conclusion
    - High stringency → few chimeras
    - Low stringency → many chimeras
  - Depth profiles



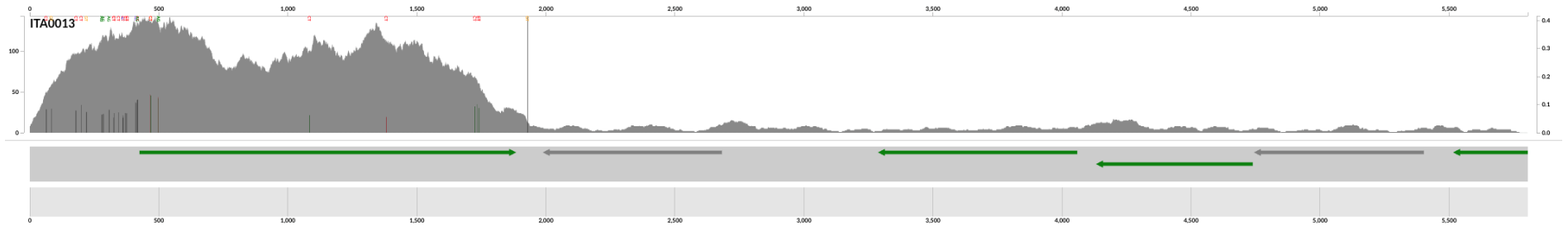
http://merenlab.org/2019/11/25/visualizing-coverages/

How to assess assembly quality for metagenomics?

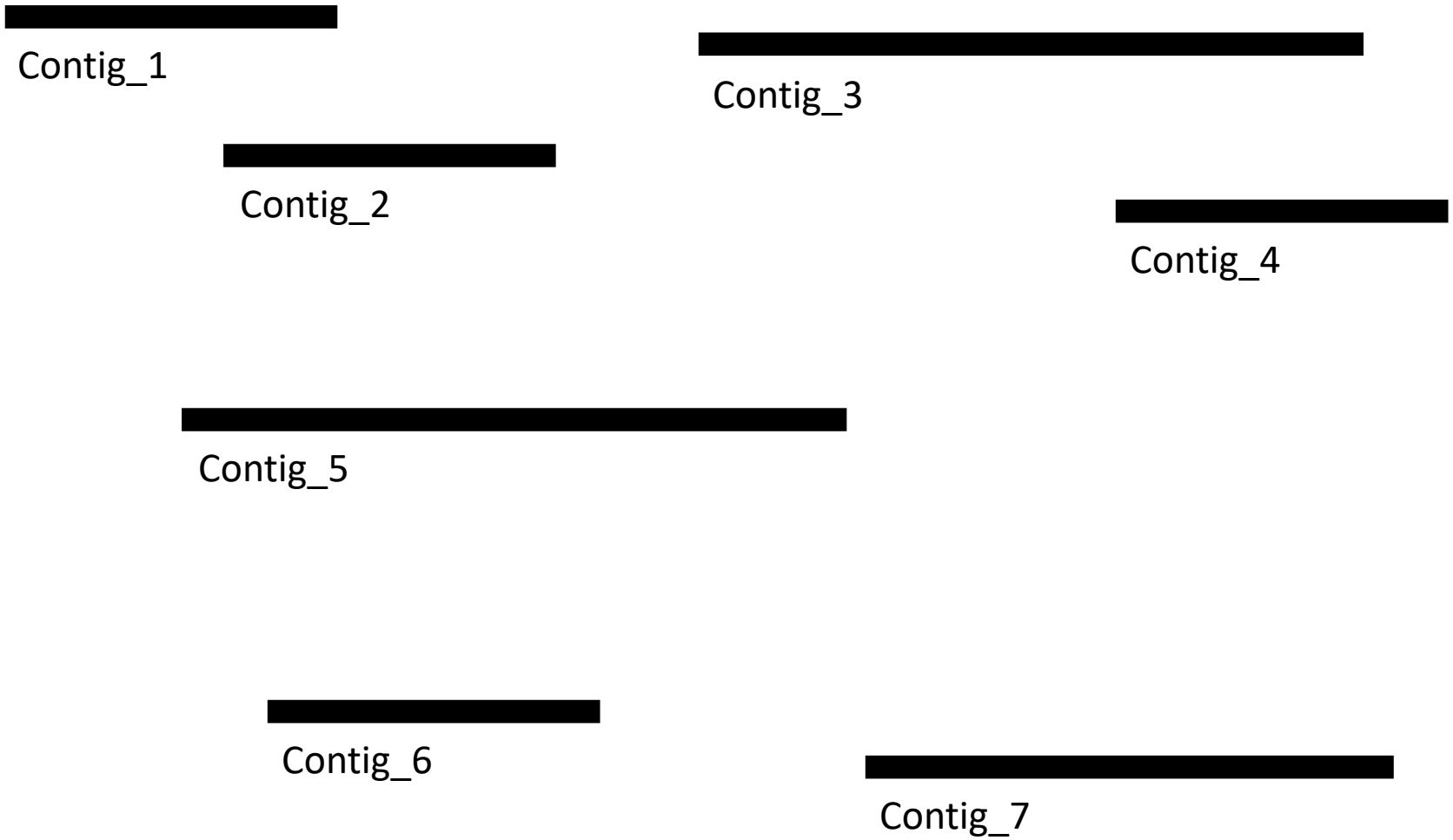
- General assembly performance metrics
  - Longest contig length, N50
  - Matched paired-end reads
  - Length distribution of open reading frames (ORFs)

How to assess assembly quality for metagenomics?

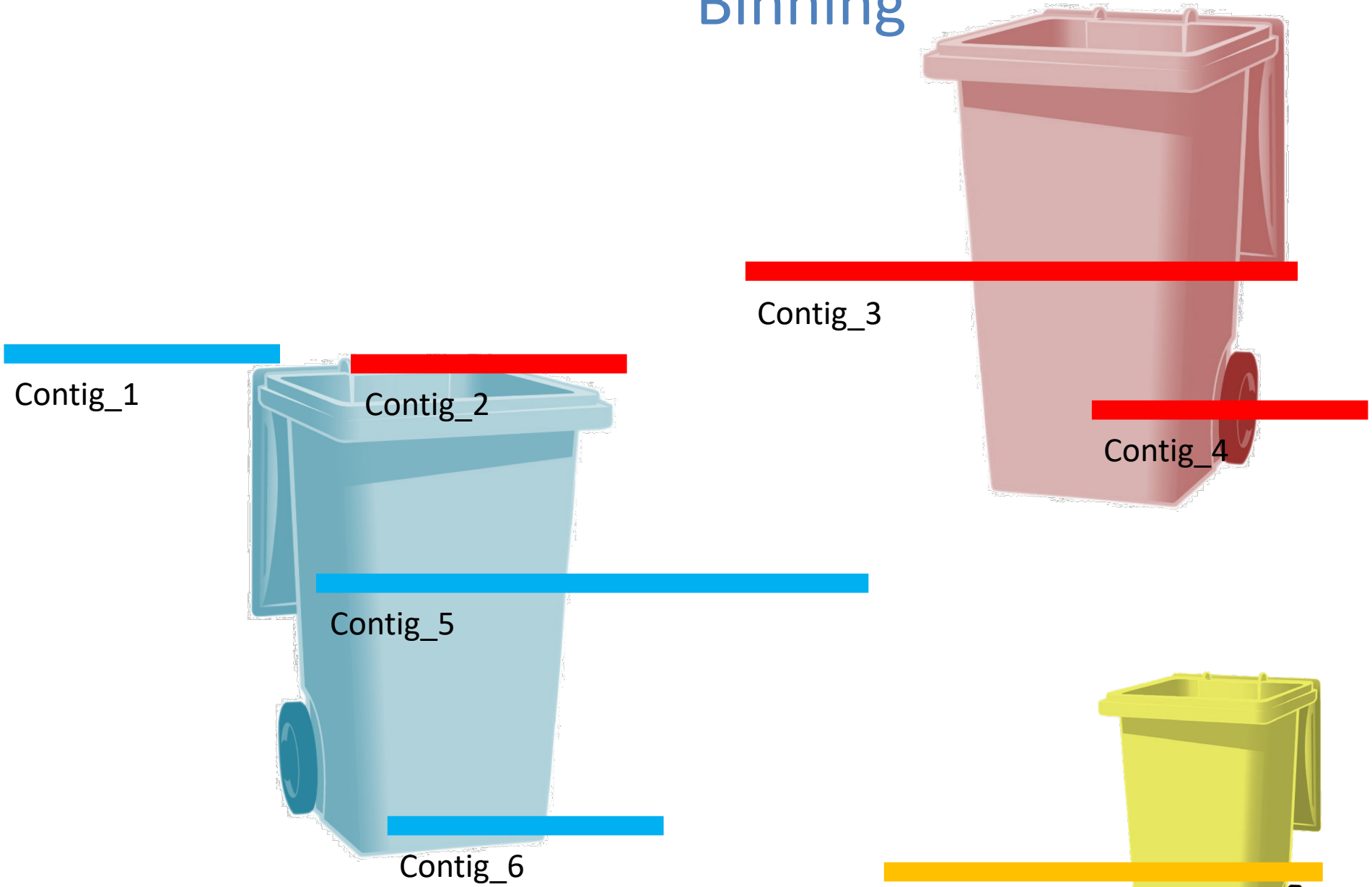
- General assembly performance metrics
  - Longest contig length, N50
  - Matched paired-end reads
  - Length distribution of open reading frames (ORFs)
- Assembly performance metrics for metagenomes:
  - Percentage of data included in the assembly
    - Align all the reads back to the assembled contigs
    - The best assembly is the one that aligns the most reads
    - This means that the assembly “explains” original data well
  - Evenness in depth along contig



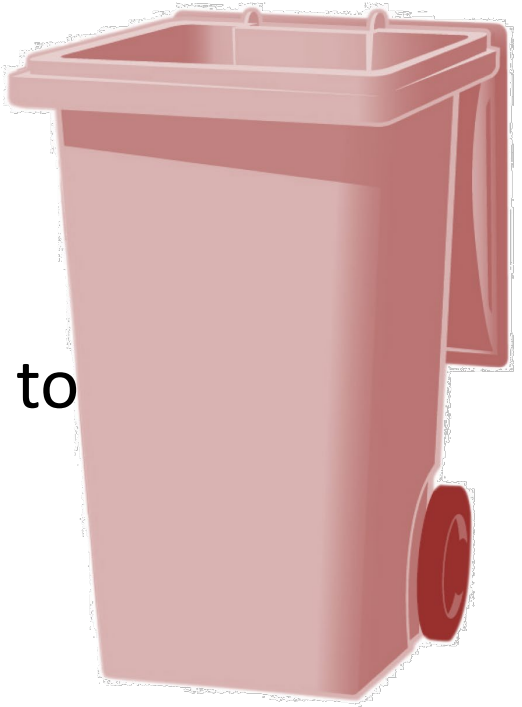
Metagenomics: Binning of contigs



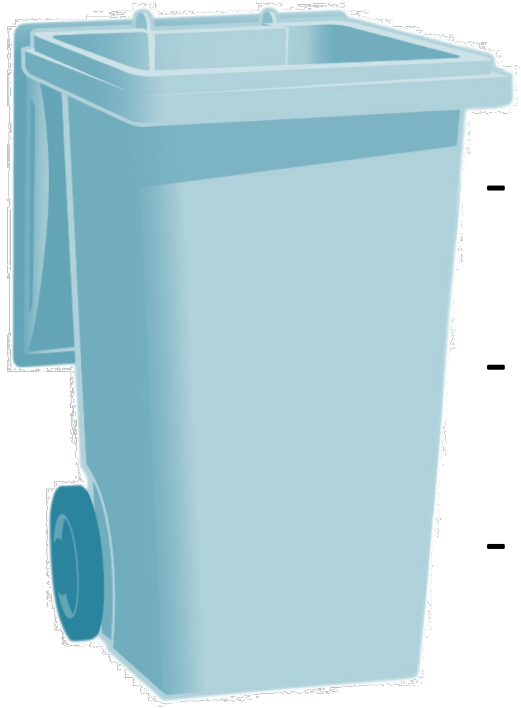
Binning



Binning



- Scaffolding
  - using forward and reverse read-pairs to make scaffolds from contigs



- GC content

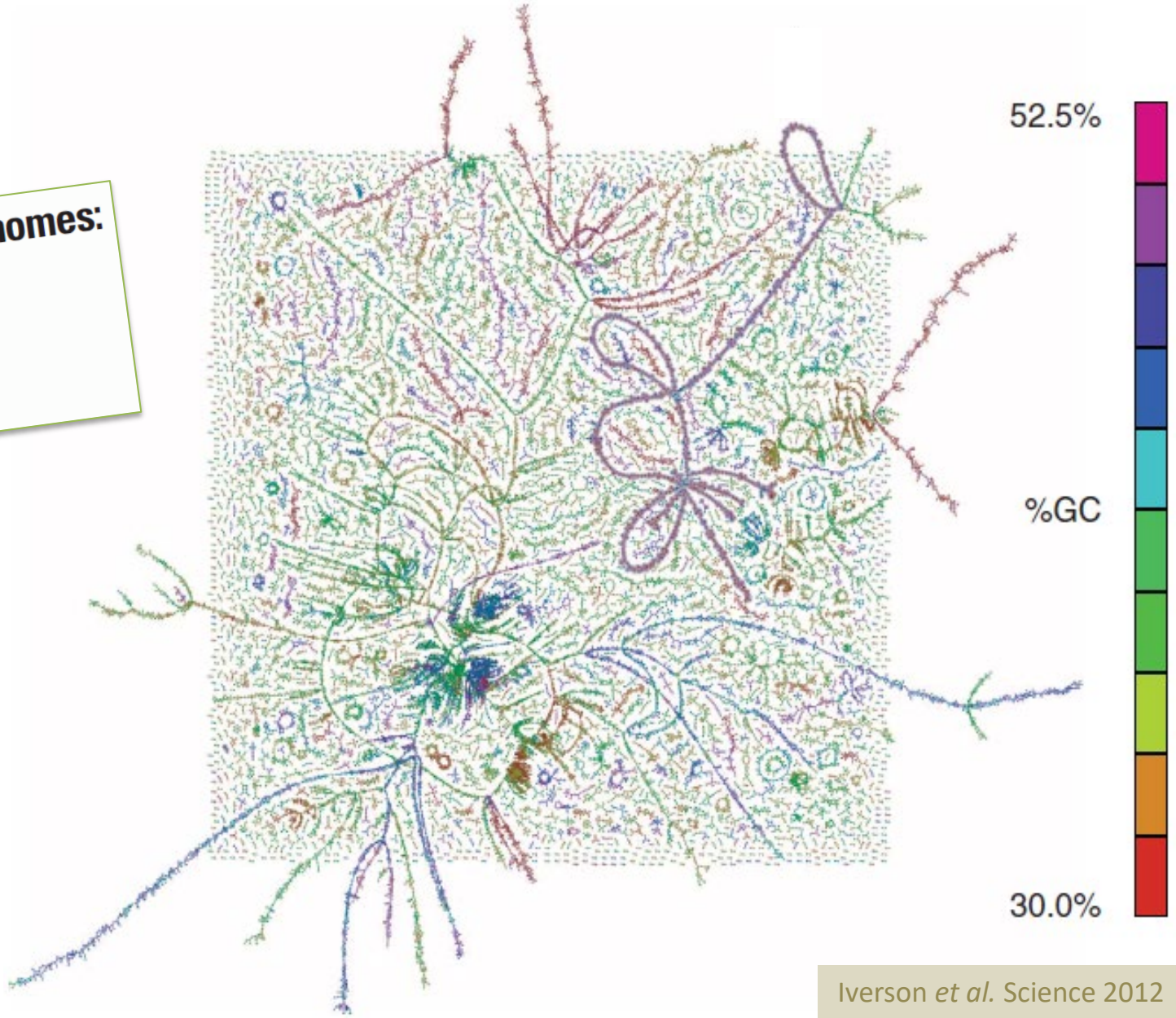
- K-mer profiles

- Depth and abundance

Linked contigs have similar GC content

Untangling Genomes from Metagenomes: Revealing an Uncultured Class of Marine Euryarchaeota

Vaughn Iverson, Robert M. Morris, Christian D. Frazar, Chris T. Berthiaume, Rhonda L. Morales, E. Virginia Armbrust*

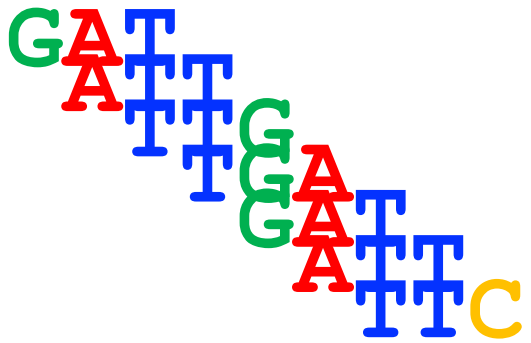


Iverson *et al.* Science 2012

k-mer profile

GATTGATTC

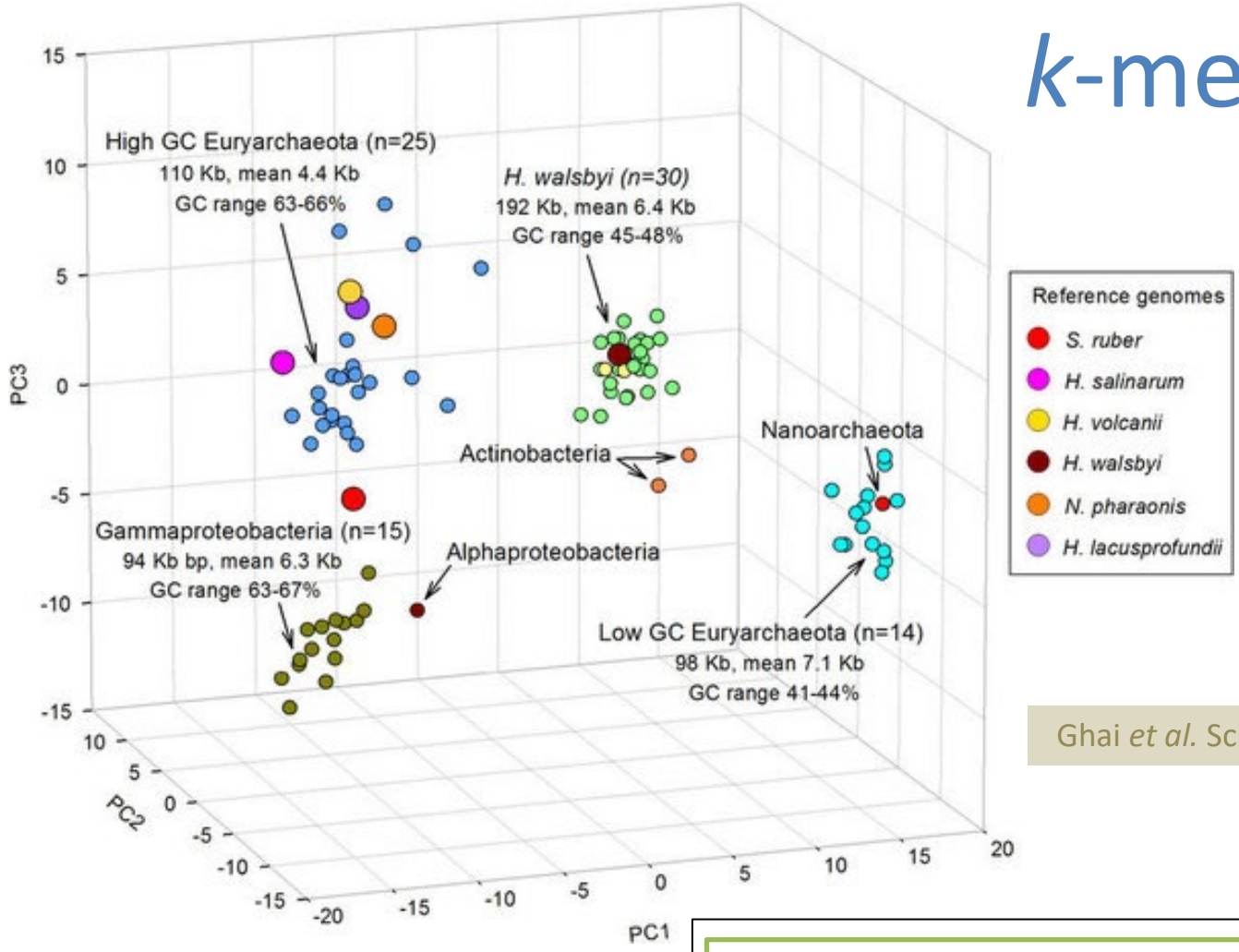
- Sequences can be divided into shorter subsequences or *k*-mers
  - *k*-mers consist of *k* nucleotides or amino acids
  - For example 3-mers:



- A k-mer profile lists the abundances of all *k*-mers in a sequence
  - This is characteristic of a genome
  - Tetramers (*k*=4) are often used for binning genomes from metagenomes

ATT	2
GAT	2
TGA	1
TTC	1
TTG	1

k-mer binning



Ghai *et al.* Scientific Reports 2011

Accurate phylogenetic classification of variable-length DNA fragments

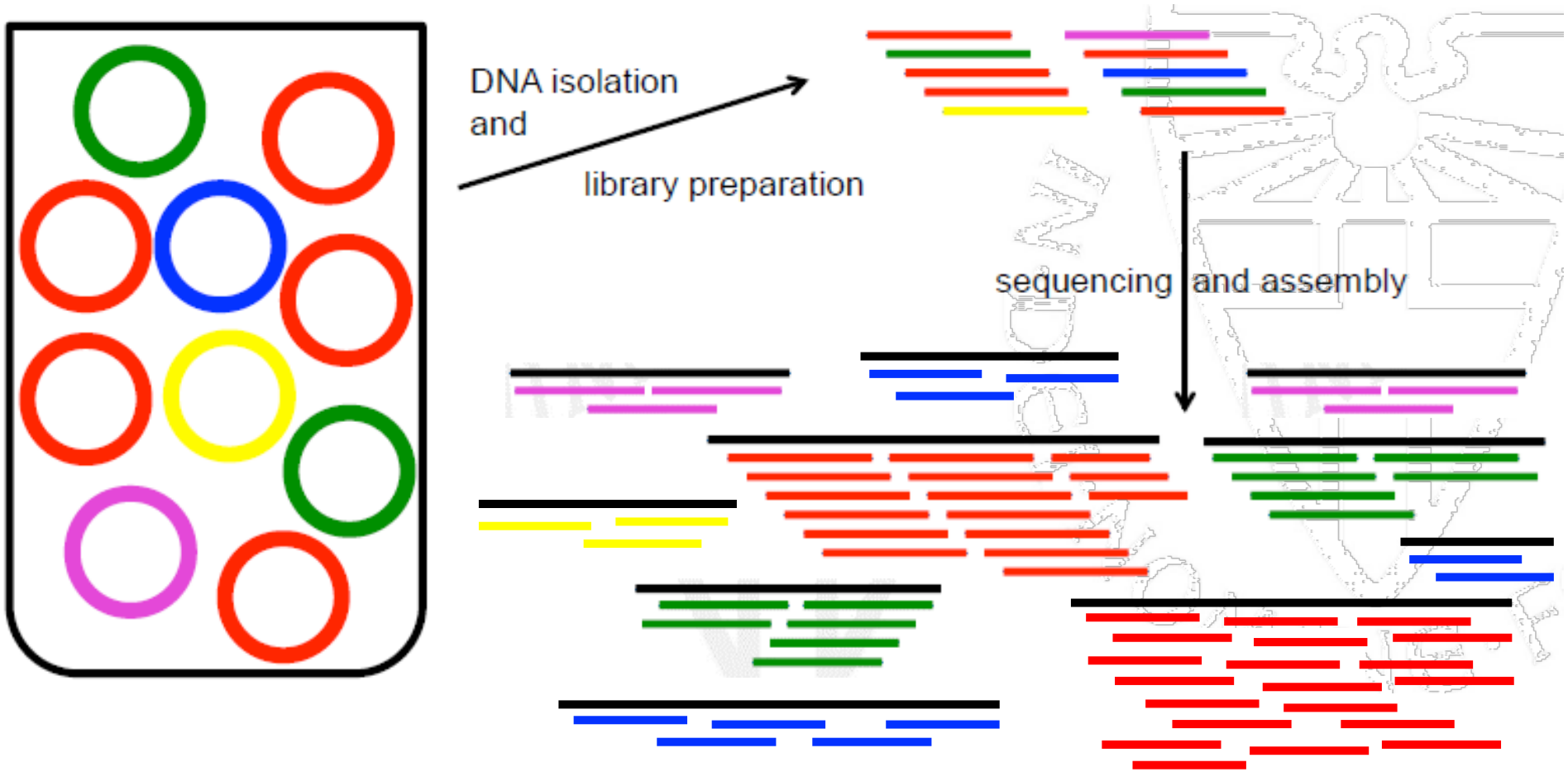
Alice Carolyn McHardy[1], Héctor García Martín[2], Aristotelis Tsirigos[1], Philip Hugenholtz[2] & Isidore Rigoutsos[1]

PhyloPythia uses a multiclass support vector machine (SVM) classifier with the oligonucleotide composition of variable-length genome fragments as the input space.

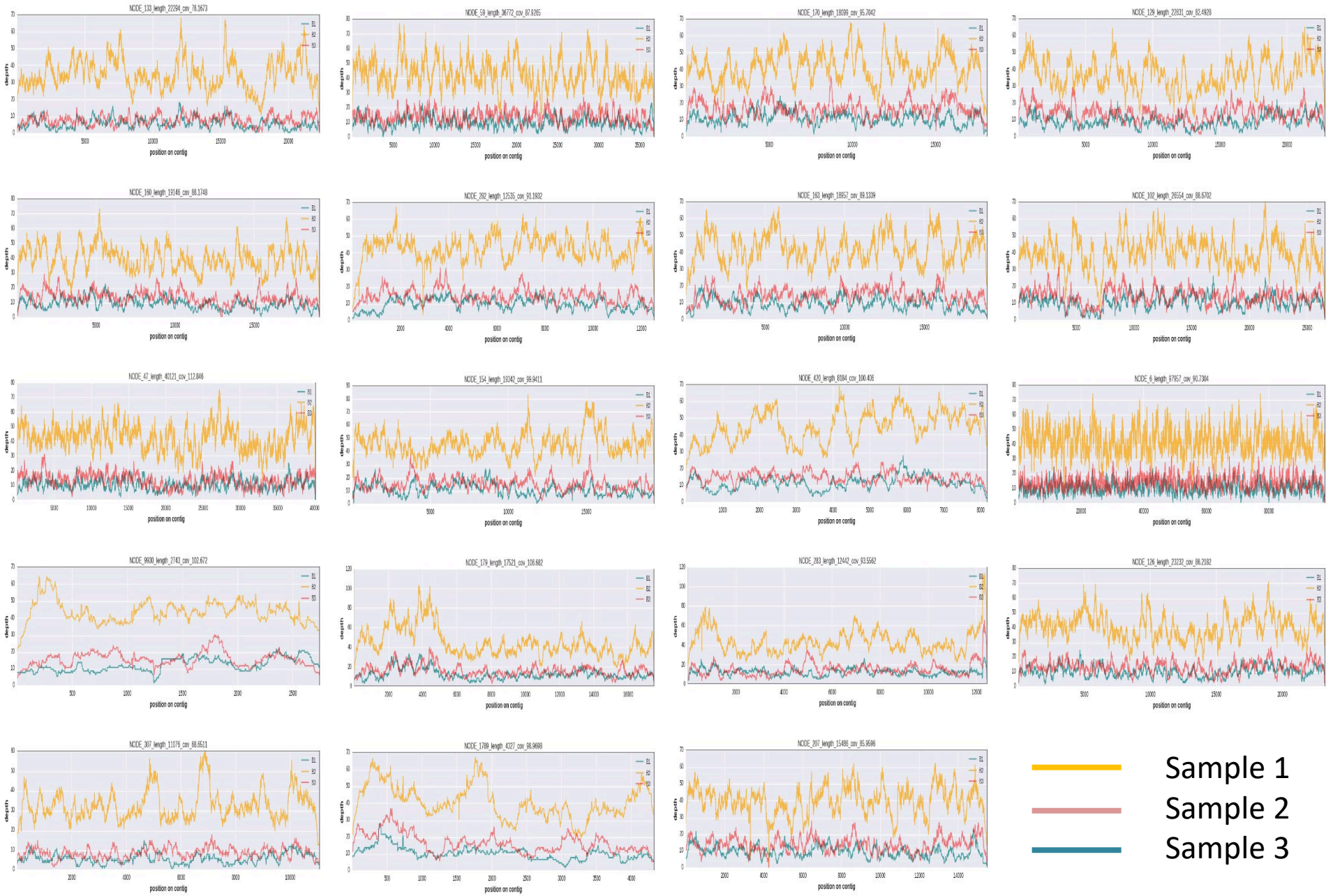
McHardy *et al.* Nature Methods 2007

Depth and abundance

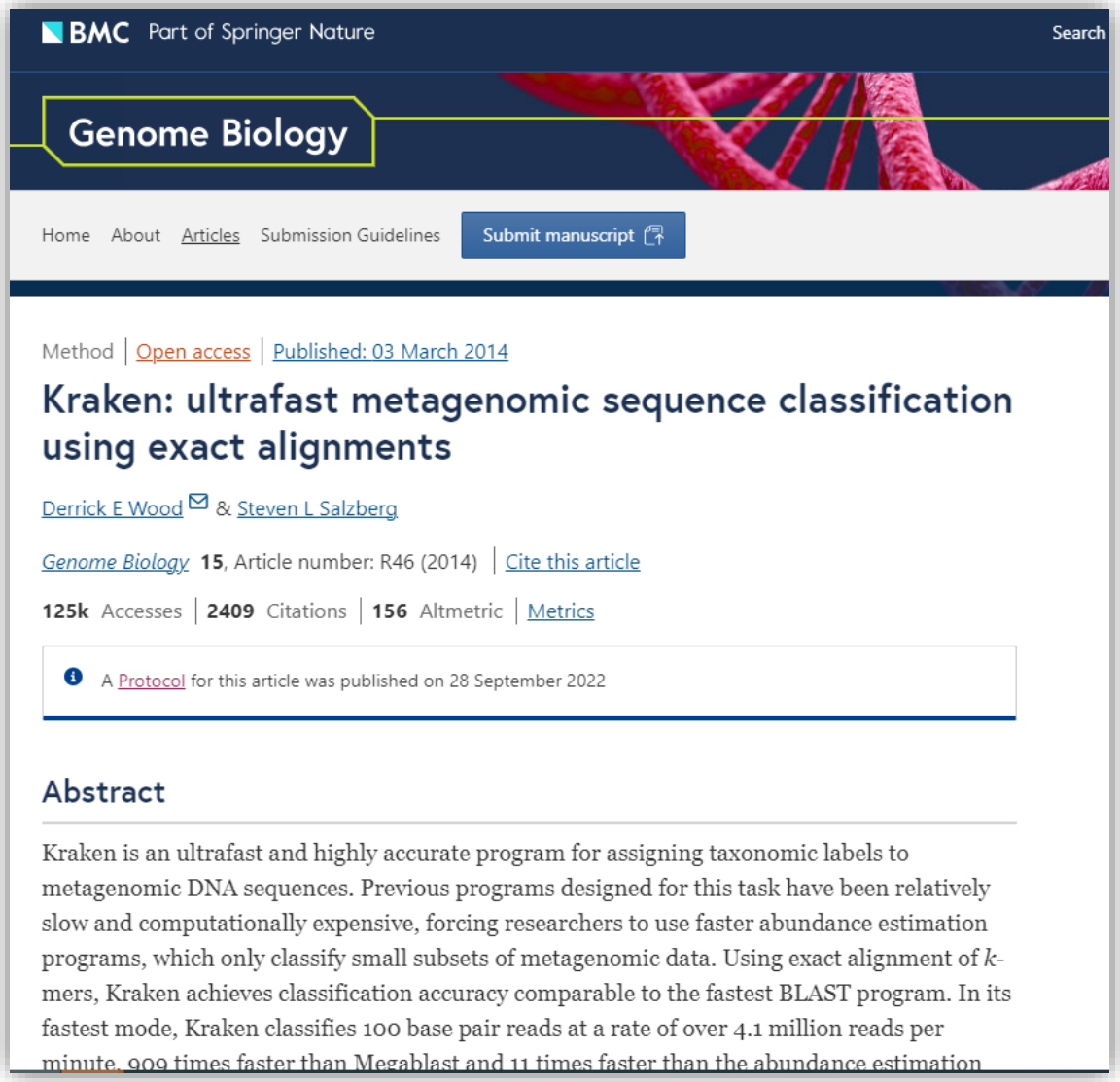
- The “depth of coverage” of a contig quantitatively reflects its abundance in a sample
  - But there is some noise, depending on the sample prep



19 contigs binned by similar depth profiles



Kraken: taxonomic sequence classification system



BMC Part of Springer Nature Search

Genome Biology

Home About Articles Submission Guidelines Submit manuscript

Method | Open access | Published: 03 March 2014

Kraken: ultrafast metagenomic sequence classification using exact alignments

Derrick E Wood & Steven L Salzberg

Genome Biology **15**, Article number: R46 (2014) | Cite this article

125k Accesses | **2409** Citations | **156** Altmetric | Metrics

A Protocol for this article was published on 28 September 2022

Abstract

Kraken is an ultrafast and highly accurate program for assigning taxonomic labels to metagenomic DNA sequences. Previous programs designed for this task have been relatively slow and computationally expensive, forcing researchers to use faster abundance estimation programs, which only classify small subsets of metagenomic data. Using exact alignment of *k*-mers, Kraken achieves classification accuracy comparable to the fastest BLAST program. In its fastest mode, Kraken classifies 100 base pair reads at a rate of over 4.1 million reads per minute, 909 times faster than Megablast and 11 times faster than the abundance estimation

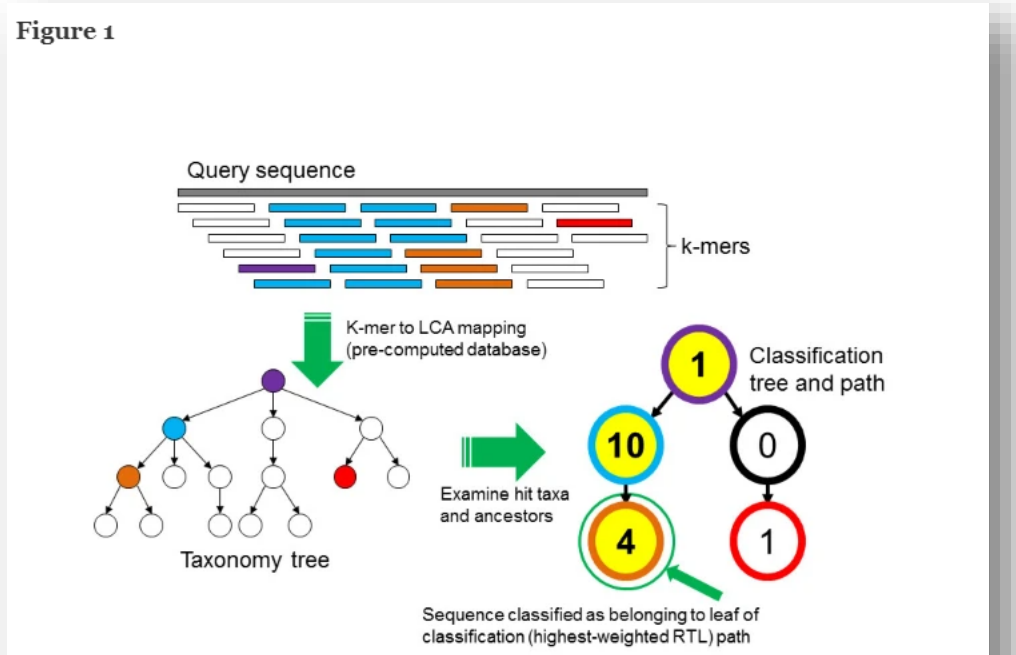
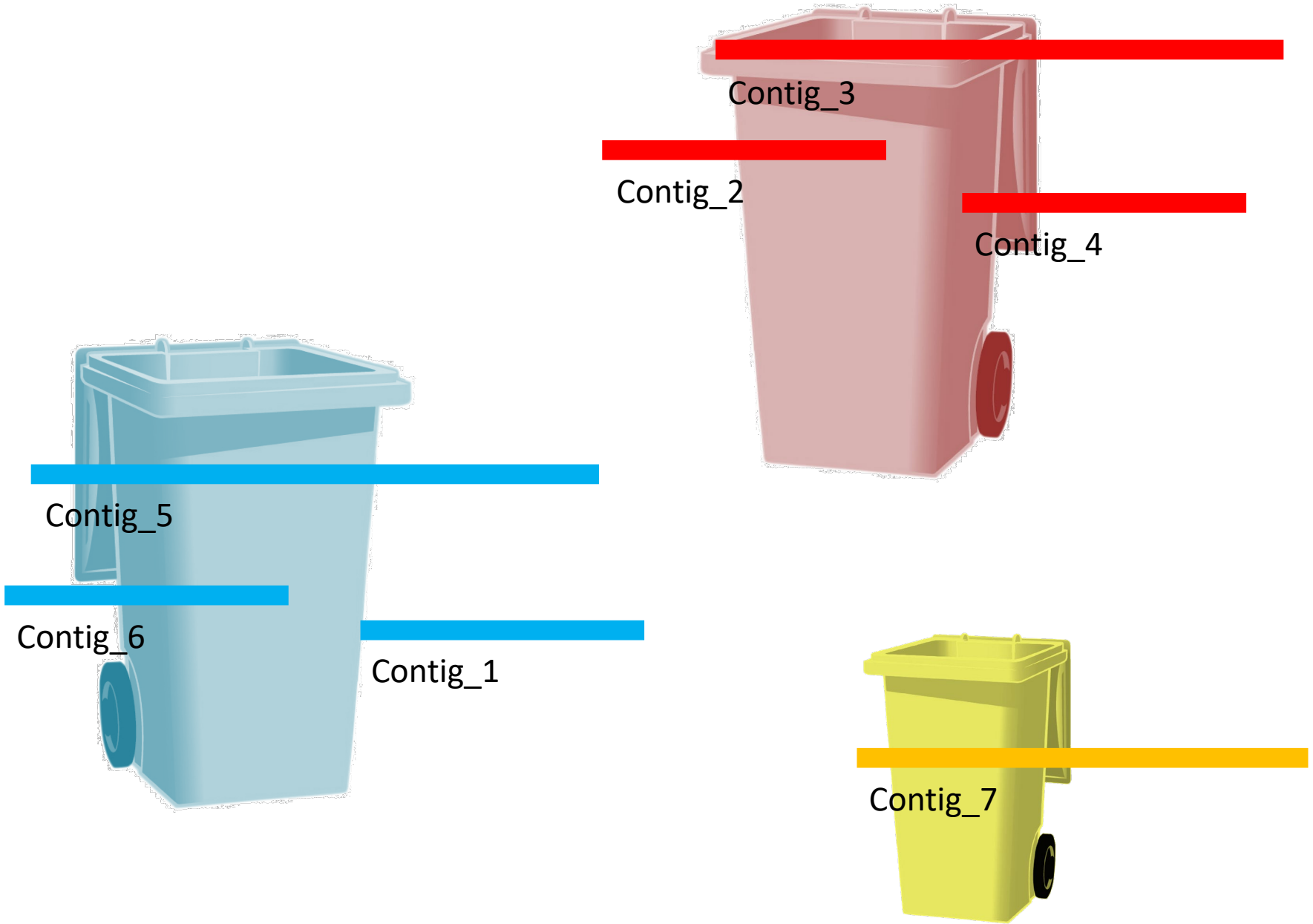


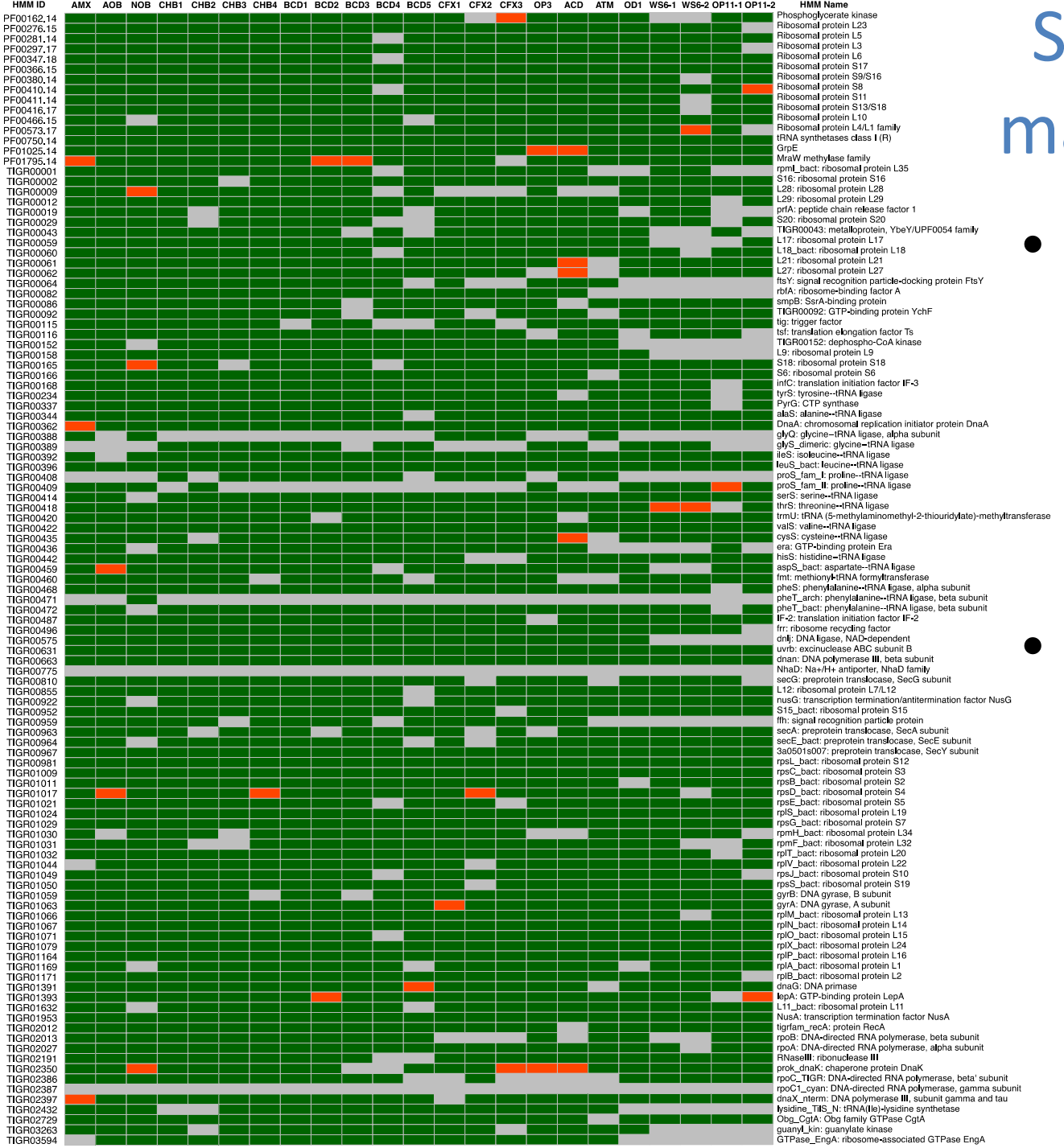
Figure 1

The Kraken sequence classification algorithm. To classify a sequence, each *k*-mer in the sequence is mapped to the lowest common ancestor (LCA) of the genomes that contain that *k*-mer in a database. The taxa associated with the sequence's *k*-mers, as well as the taxa's ancestors, form a pruned subtree of the general taxonomy tree, which is used for classification. In the classification tree, each node has a weight equal to the number of *k*-mers in the sequence associated with the node's taxon. Each root-to-leaf (RTL) path in the classification tree is scored by adding all weights in the path, and the maximal RTL path in the classification tree is the classification path (nodes highlighted in yellow). The leaf of this classification path (the orange, leftmost leaf in the classification tree) is the classification used for the query sequence.

How do we know if the genome is complete?



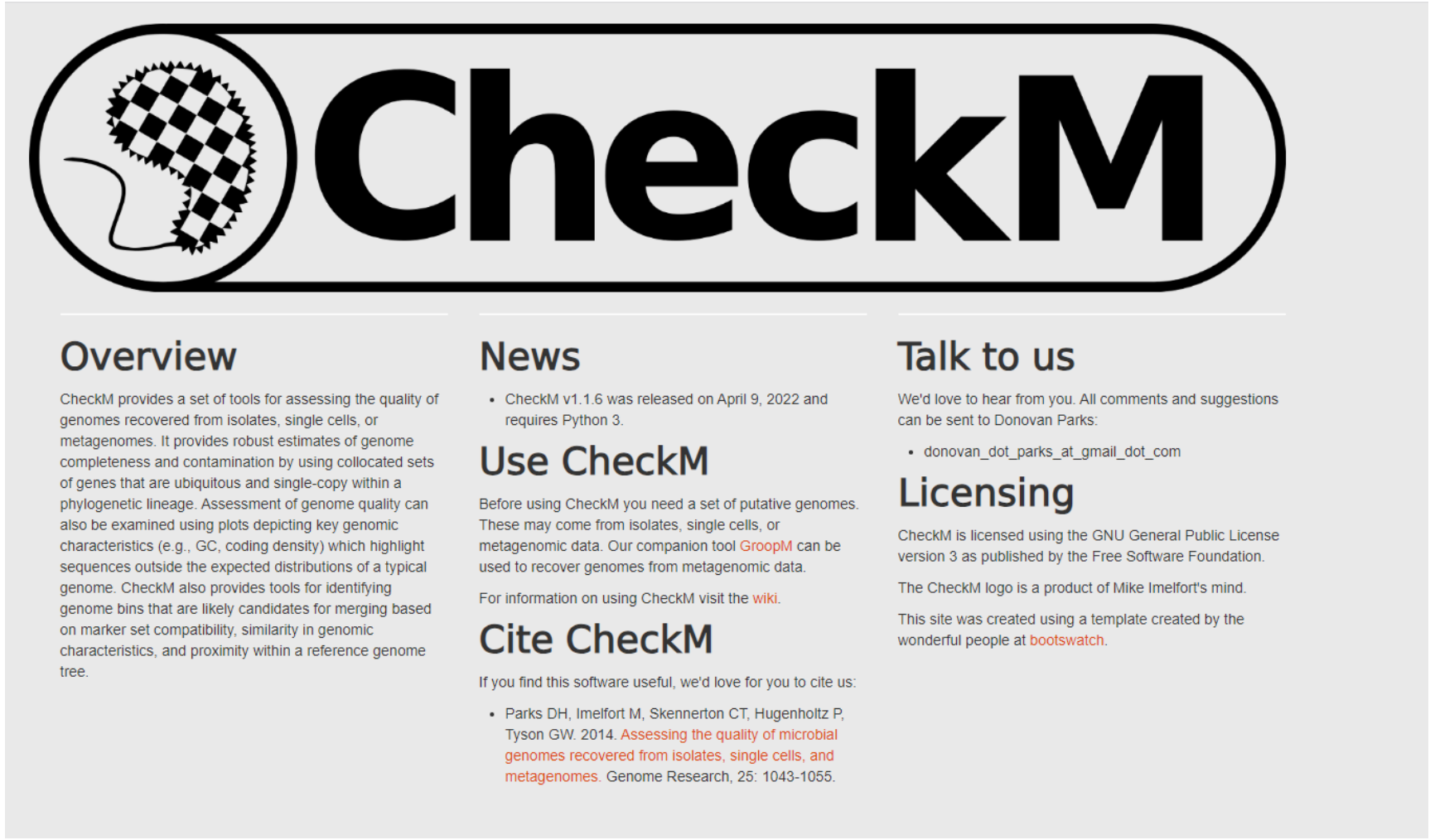
Single-copy marker genes



- Completeness
  - Marker genes are expected to be present in all bacteria
- Redundancy
  - Single-copy genes are expected to be present only once

Speth *et al.* Nature Comm. 2016

Checkm; tool for assessing the quality of genomes



CheckM

Overview

CheckM provides a set of tools for assessing the quality of genomes recovered from isolates, single cells, or metagenomes. It provides robust estimates of genome completeness and contamination by using collocated sets of genes that are ubiquitous and single-copy within a phylogenetic lineage. Assessment of genome quality can also be examined using plots depicting key genomic characteristics (e.g., GC, coding density) which highlight sequences outside the expected distributions of a typical genome. CheckM also provides tools for identifying genome bins that are likely candidates for merging based on marker set compatibility, similarity in genomic characteristics, and proximity within a reference genome tree.

News

- CheckM v1.1.6 was released on April 9, 2022 and requires Python 3.

Use CheckM

Before using CheckM you need a set of putative genomes. These may come from isolates, single cells, or metagenomic data. Our companion tool GroopM can be used to recover genomes from metagenomic data.

For information on using CheckM visit the wiki.

Cite CheckM

If you find this software useful, we'd love for you to cite us:

- Parks DH, Imelfort M, Skennerton CT, Hugenholtz P, Tyson GW. 2014. Assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. Genome Research, 25: 1043-1055.

Talk to us

We'd love to hear from you. All comments and suggestions can be sent to Donovan Parks:

- donovan_dot_parks_at_gmail_dot_com

Licensing

CheckM is licensed using the GNU General Public License version 3 as published by the Free Software Foundation.

The CheckM logo is a product of Mike Imelfort's mind.

This site was created using a template created by the wonderful people at bootswatch.

Summary

- Assembly of reads into contigs
  - Reference-based
  - De novo assembly
- Coverage and depth
- Assembly algorithms
- Assembly quality (N50, ORF length)
- Problems with repeats and chimeras
- Metagenomics assembly, binning
- Tools: SPAdes, UniCycler, Kraken, Checkm