

Mathematical model of phylogenetic compression

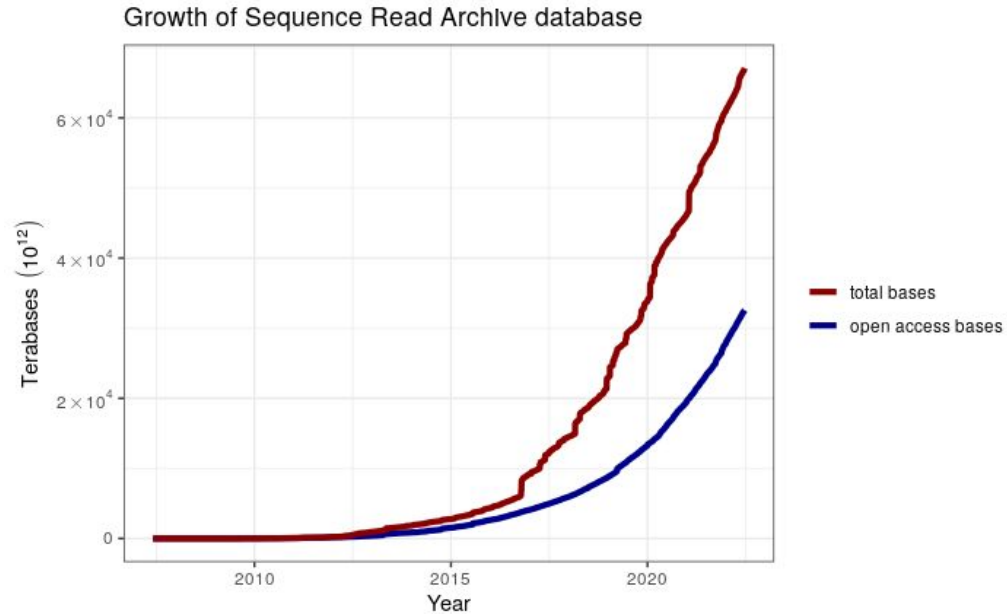
Veronika Hendrychová, Karel Břinda



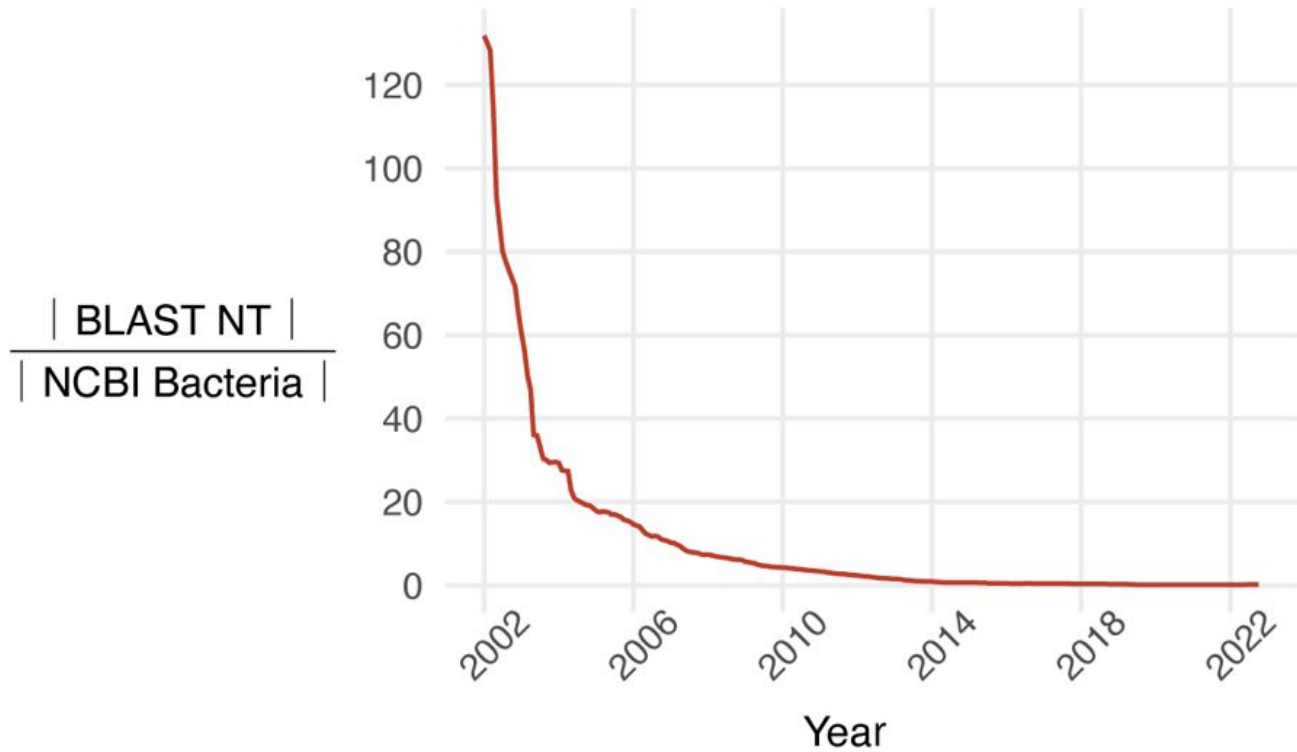
Inria

Challenge: Genomic data grow faster than computational capacities

- Development of sequencing technologies leads to exponential growth of genomic data
- BLAST and its successors don't keep up
 - small database, not scalable to modern data



Consequence: Proportion of searchable bacteria decreases exponentially



Genome compression widely studied, but existing compressors unscalable to modern bacterial collections (millions of genomes, high-divers.)

Rich toolbox of compression techniques

reviews: [Giancarlo&Scaturro, 2009], [Deorowicz&Grabowski, 2013], [Giancarlo&al., 2013], [Zhu&al., 2015], ...

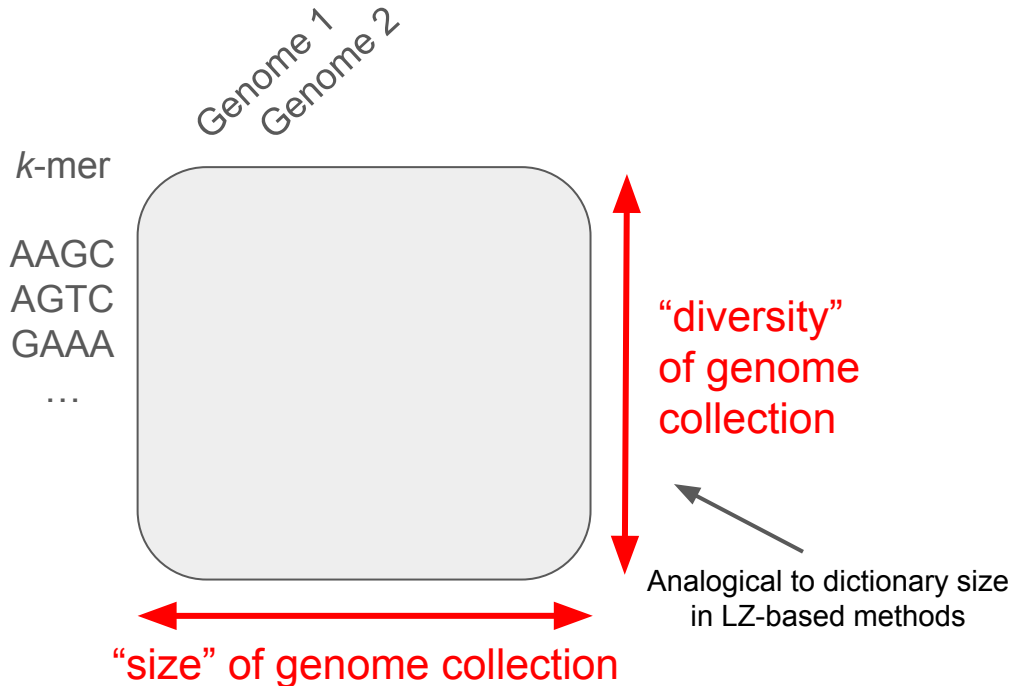
- **Dictionary compression** – using a dictionary of repetitive phrases
 - General: gzip, bzip2, lzma/xz/7z,...
 - Specialized: mbgc [*GigaScience*, 2022], AGC [*Bioinformatics*, 2023]
- **Statistical compressors**
 - E.g., GeCo3 based on neural networks [*GigaScience*, 2020]
- **K-mer-based tools**
 - E.g., Metagraph [Karasikov et al, *BioRxiv*, 2020], Themisto [Alanko et al, *Bioinformatics*, 2023], Fulgor [Fan et al, *bioRxiv*, 2023]

General issue:

Difficult to identify redundancies in a scalable manner across millions of genomes of variable diversity

Why is it difficult to detect redundancies / compress?

Example: compression of k -mer matrices



For microbial data huge
in both dimensions

Example (661k collection):
661k \times 44.3 G
= 3.6 petabytes (if stored as 8
values in 1 byte)

Recent breakthrough: phylogenetic compression

Key idea: Reversible reordering of input data according to their evolutionary history, in order to simplify compression by existing tools

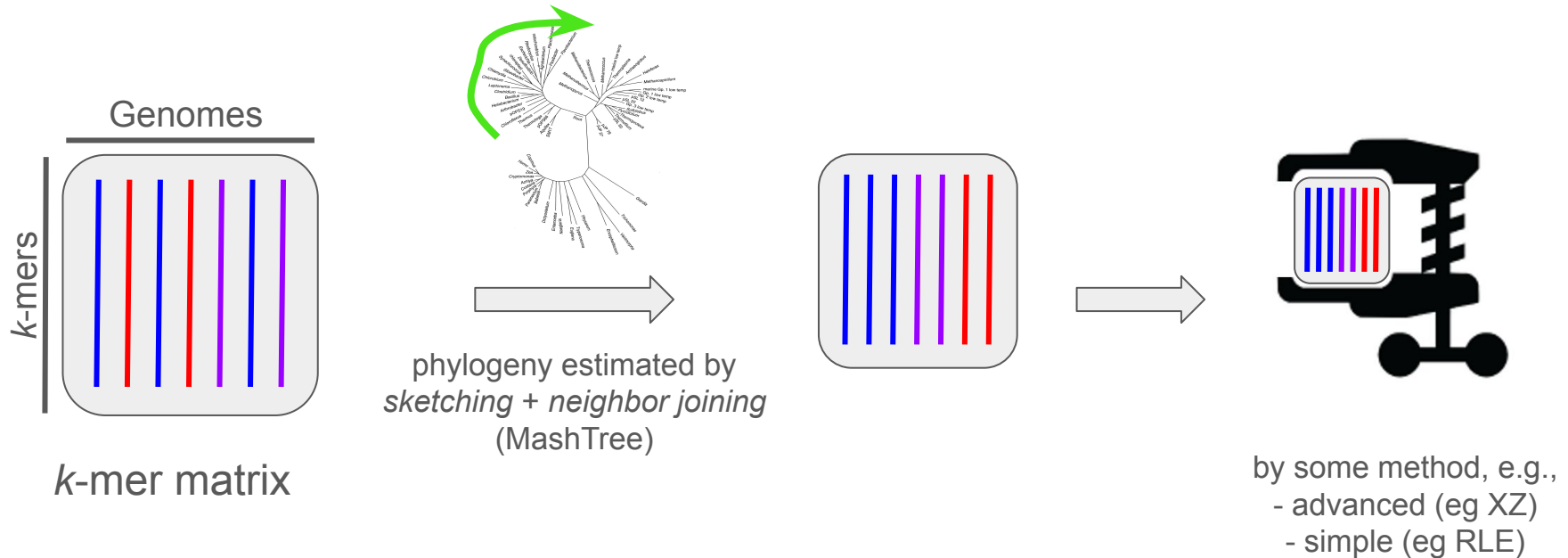
Makes data almost trivially compressible

Highly general, applicable to *assemblies*, *de Bruijn graphs*, *Bloom filters*, ...

Can be instantiated to individual protocols for different data types & use-cases

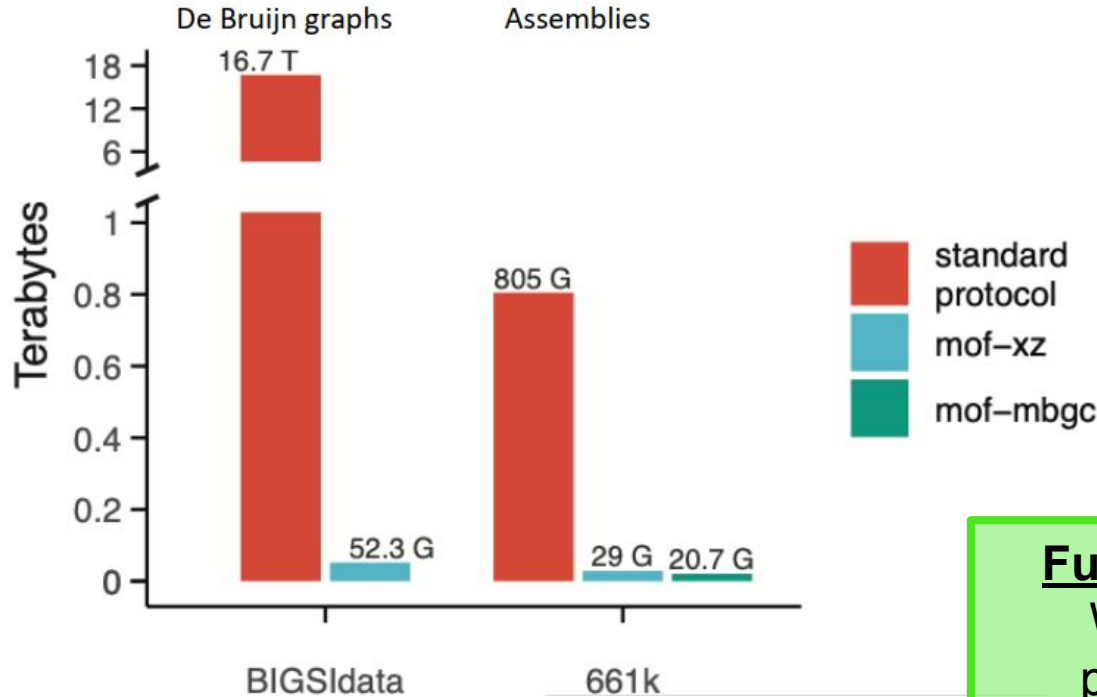


Example protocol: simplified protocol for k -mer matrices



(Note: Protocols for large collections are more complex and involve clustering by metagenomics, see [Břinda et al., 2023])

On modern collections, phylogenetic compression improves state-of-the-art by 1–2 orders of magnitude






Fundamental question:
Which mathematical principles drive these improvements?

What does phylogenetic compression
do on a *mathematical level*?

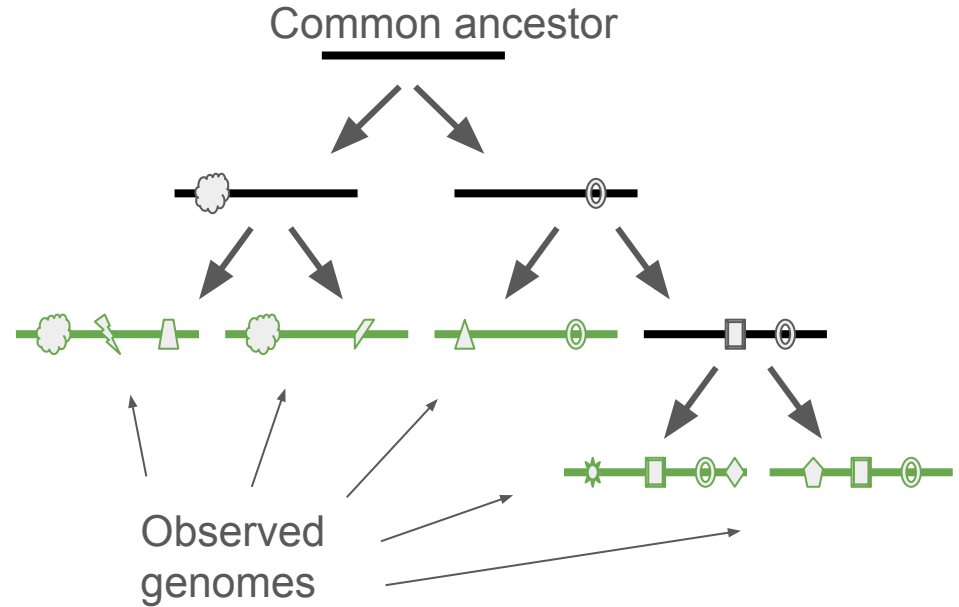
Methodology: Mathematical modeling of phylogenetic compression

In our case:

- I. modeling the structure of input data  infinite-site model
- II. fixing one genome data representation
- III. fixing one protocol of phylogenetic compression  *k*-mer matrices
- IV. studying compression as an optimization problem 
 - 1. phylogeny approximated via Mashtree
 - 2. left-to-right reordering
 - 3. run-length encoding
- V. comparing compression with and without guiding by evolutionary history

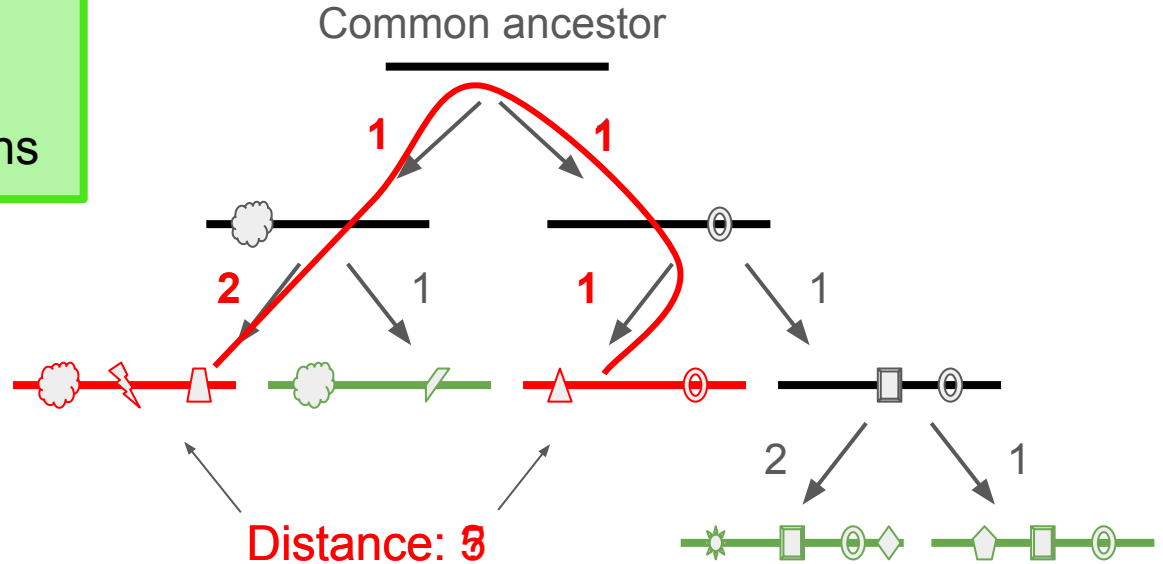
Component 1: Data modeling by infinite-site model (with substitutions)

1. Infinite number of positions
(\approx genomes sufficiently long)
 2. Each new substitution
occurs at a novel position
 3. No recombination
- Models realistically
oversampled parts of the tree
of life (e.g., data from hospital
outbreaks)

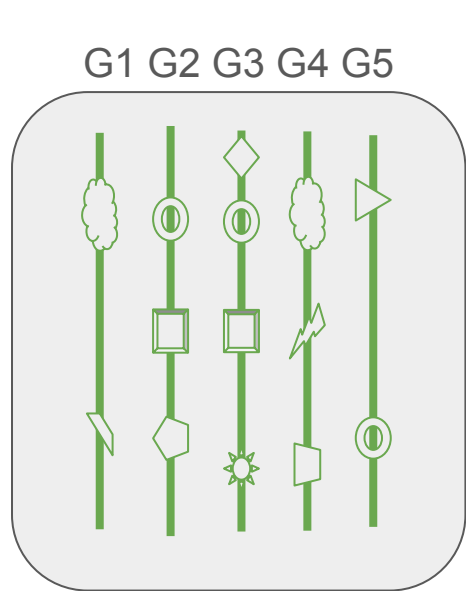


Consequence 1: genome distances perfectly explained by the tree (i.e, so-called additive distances)

Genomes' distance
=
distinct mutated positions



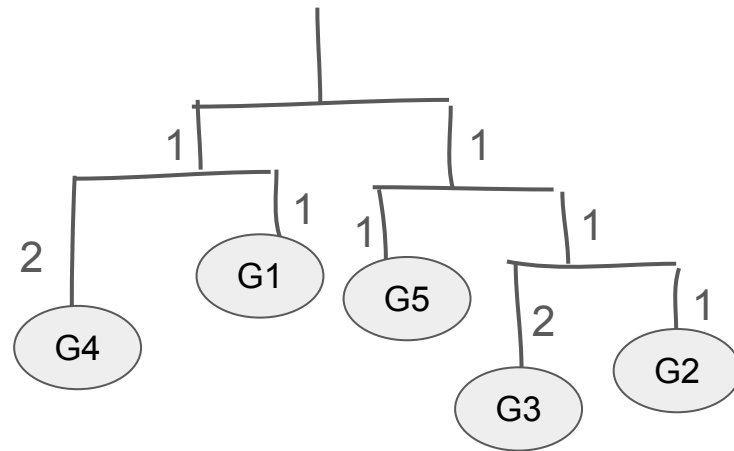
Consequence 2: tree perfectly inferable from input genomes by Neighbor Joining



Assuming additive distances between observed genomes



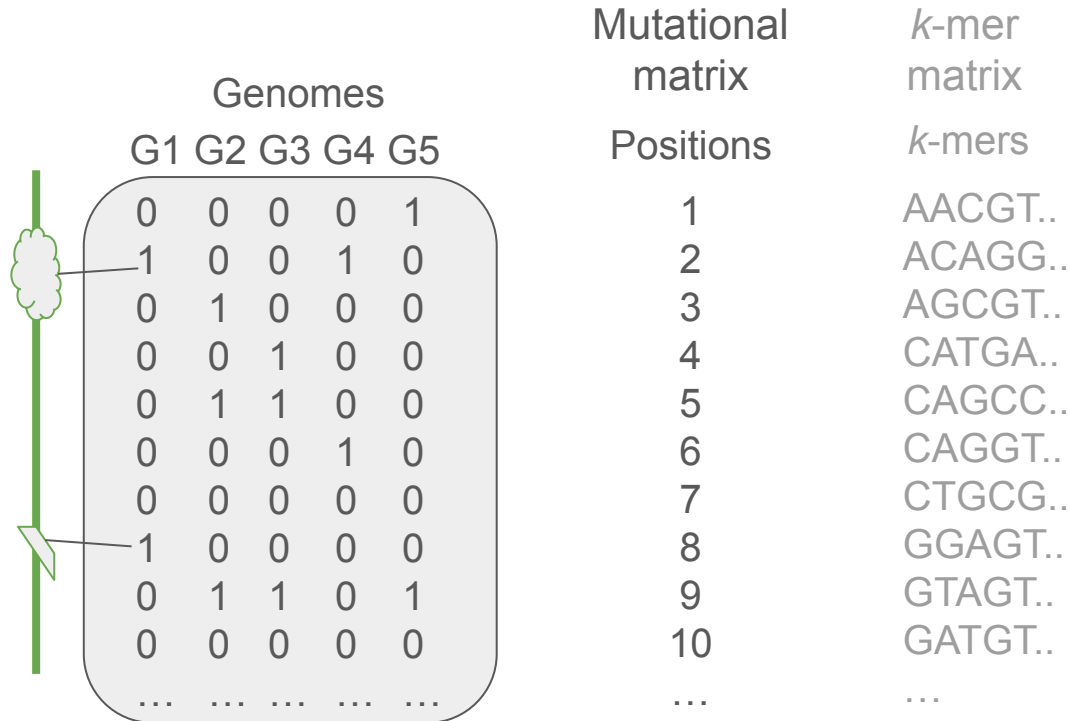
[Saitou&Nei, *Molecular Biology and Evolution*, 1987]



Guaranteed to infer one evolutionary tree perfectly describing the distances

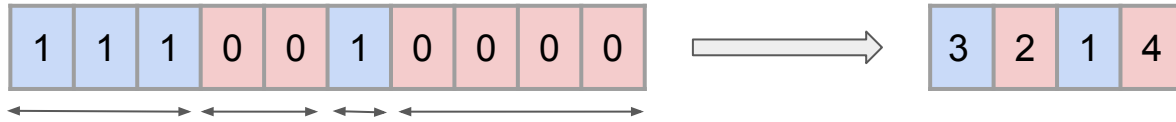
Component 2: Representing genomes via binary matrices

We work with k -mer matrices, but for simplicity for now let's assume mutational matrix

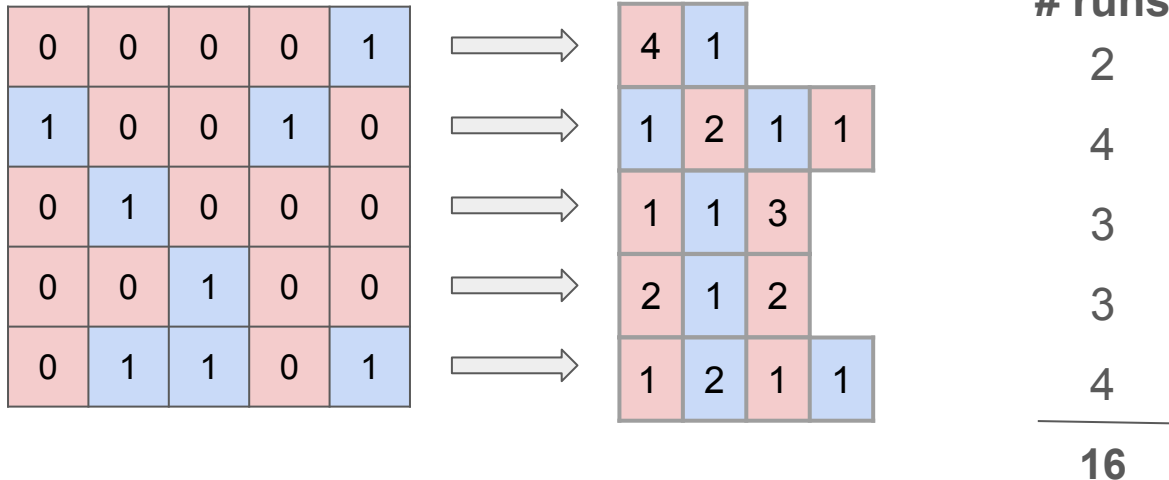


Component 3: Compression by Run-Length Encoding (RLE)

Principle: encoding lengths of runs of identical characters



Compressing matrix: RLE of individual rows



Compressed size
=
runs

Quick recap:

We have:

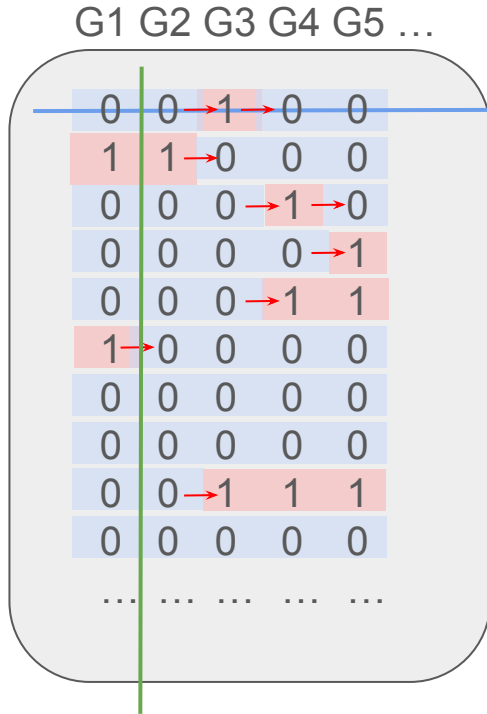
- **Input:** genome collection, modeled by the *infinite-site model*, represented in a *binary matrix*
- **Low-level compressor:** RLE
- **Phylogenetic compression protocol:** column reordering left-to-right according to the NJ tree

Want to compare:

- no phylogenetic compression (random order)
- phylogenetic compression (left-to-right order with respect to phylogeny)
- optimal compression (mathematically optimal order minimizing size of RLE)

Property of binary matrices

#Runs corresponds to Hamming distance



$$\sum_{\text{rows}} \# \text{ runs} = \# \text{ rows} + \sum_{\text{adjacent cols } i, j} \text{Ham}(i, j)$$

Minimizing # runs
 =
 Minimizing columns' cumulative Hamming distances
 =
 Solving *Traveling Salesman Problem* (TSP)

Hamming distance: 1
(= # distinct characters)

Travelling Salesman Problem (TSP)

What's the shortest possible route between cities (=genomes)?



Travelling Salesman Problem (TSP)

Generally NP-hard, but good *approximation algorithms* as well as *efficient solvers* exist (e.g, Concorde [Cook et al., 1997])



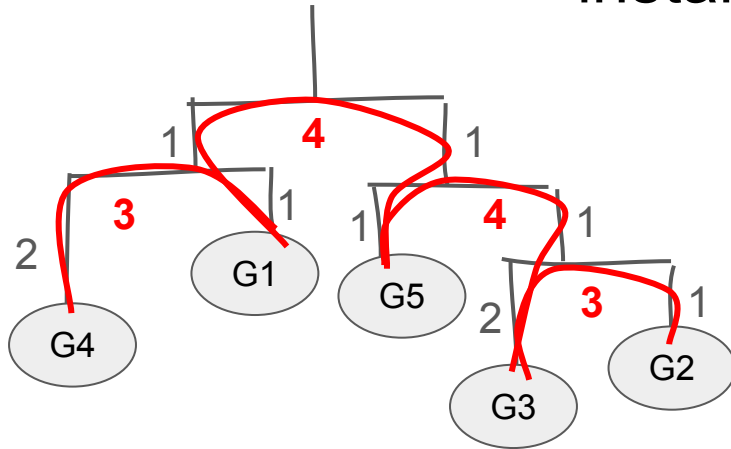
Heuristic solution



Optimal solution

(Note: NP-hardness in our specific case of matrix column reordering unclear)

Main result: phylogenetic compression solves this TSP instance optimally



Theorem: Shortest path in the TSP = left-to-right order in the NJ phylogeny

Distances perfectly explained by our unique inferred tree => what is the shortest leaves traversal?

Consequence: Phylogenetic compression provides **optimal** RLE of input genomes

Evaluation with experimental data

Our idealized model vs. Reality

- Infinite-site model for point mutations + k -mer matrices

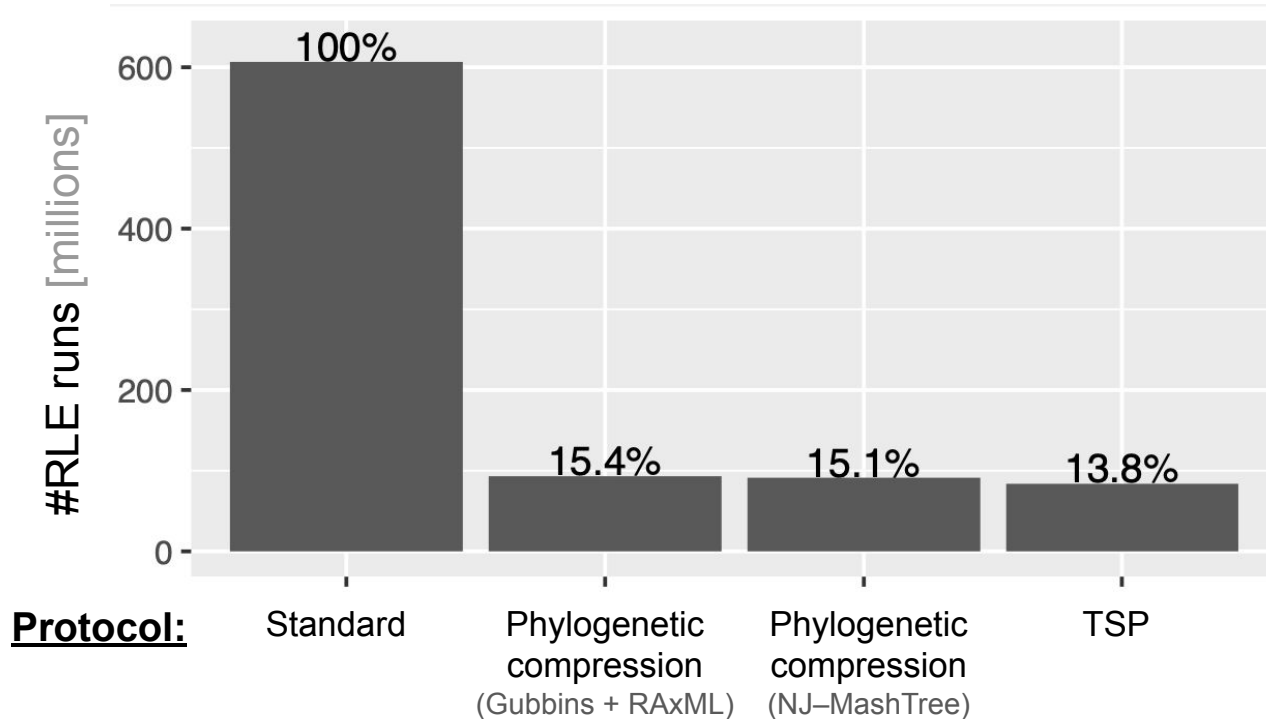
- bacterial genomes not infinite
- horizontal gene transfer in bacteria
- other mutations than point mutations
- mutations may occur in a close proximity
- etc. etc.

How well do our mathematical models explain real data?

(Is phylogenetic compression still (near-)optimal?)

With RLE as a low-level compressor:

Phylogenetic compression near-optimal for single species



Consequence:
Phylogenies
near-perfectly
locally
approximate the
geometry of the
bacterial genome
space

Conclusions

- Effectivity of the phylogenetic compression is **supported by the evolutionary processes** and profound mathematical principles
- Data resulting from evolutionary processes feature a **tree-like structure**
- Phylogenies **well approximate the geometry** of microbial genome space locally
- **Our long-term vision:** using these principles to develop efficient entropy-scaling algorithms to achieve search sublinearity

Thank you for your attention!



Karel Břinda

