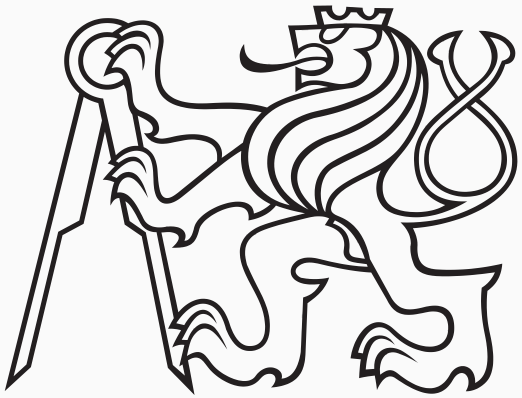
ESTIMATING SEQUENCE SIMILARITY FROM CONTIG SETS

Petr Ryšavý, Filip Železný

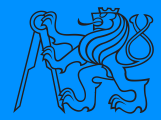
Saturday 28th October, 2017

IDA, Dept. of Computer Science, FEE, CTU



INTRODUCTION

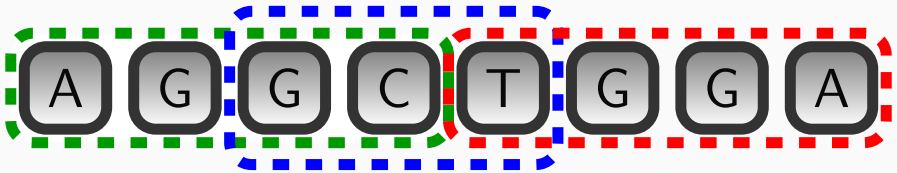
What we study.

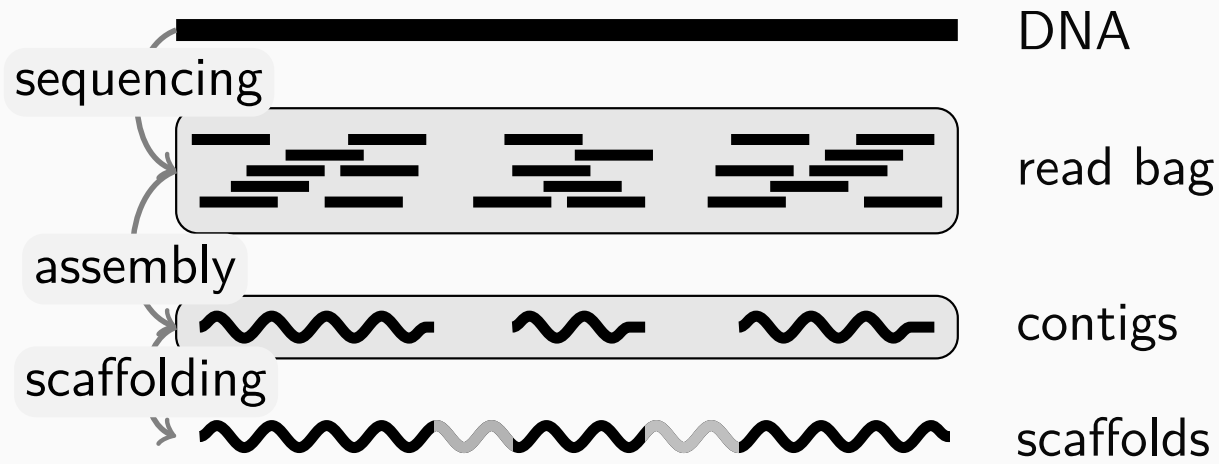


- DNA sequences encode all information needed for cell growth, replication, and function.
- Knowledge and understanding of DNA may help in medicine, biology and other fields
- Sequencing is a process of reading DNA
- Hierarchical clustering may indicate evolution of organisms

Read sets

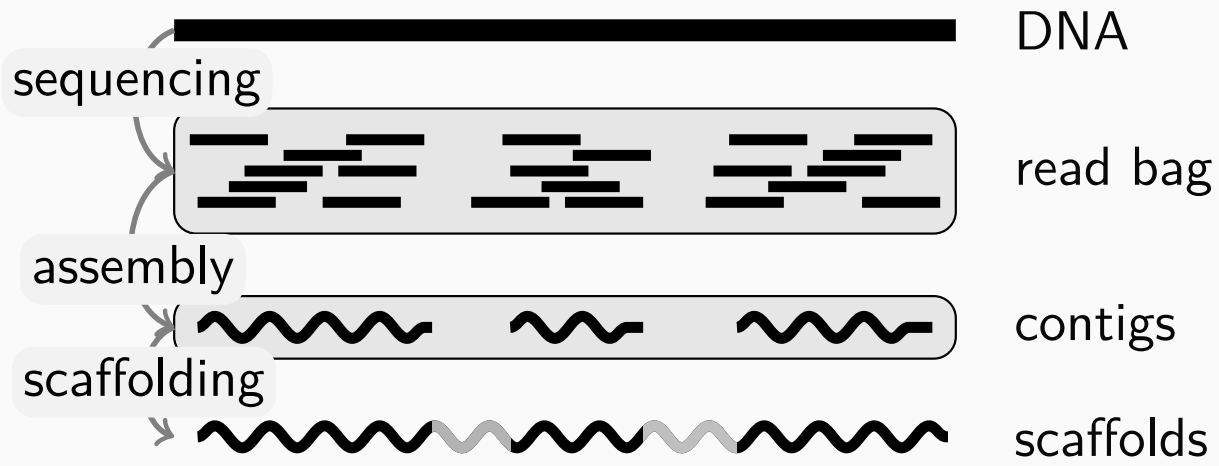
- Product of sequencing is not a long sequence, but short substrings called reads
- Reads have length of 10s to 100s of symbols
- Sequence AGGCTGGA is represented by set $\{\mathrm{AGGC}, \mathrm{TGGA}, \mathrm{GCT}\}$.





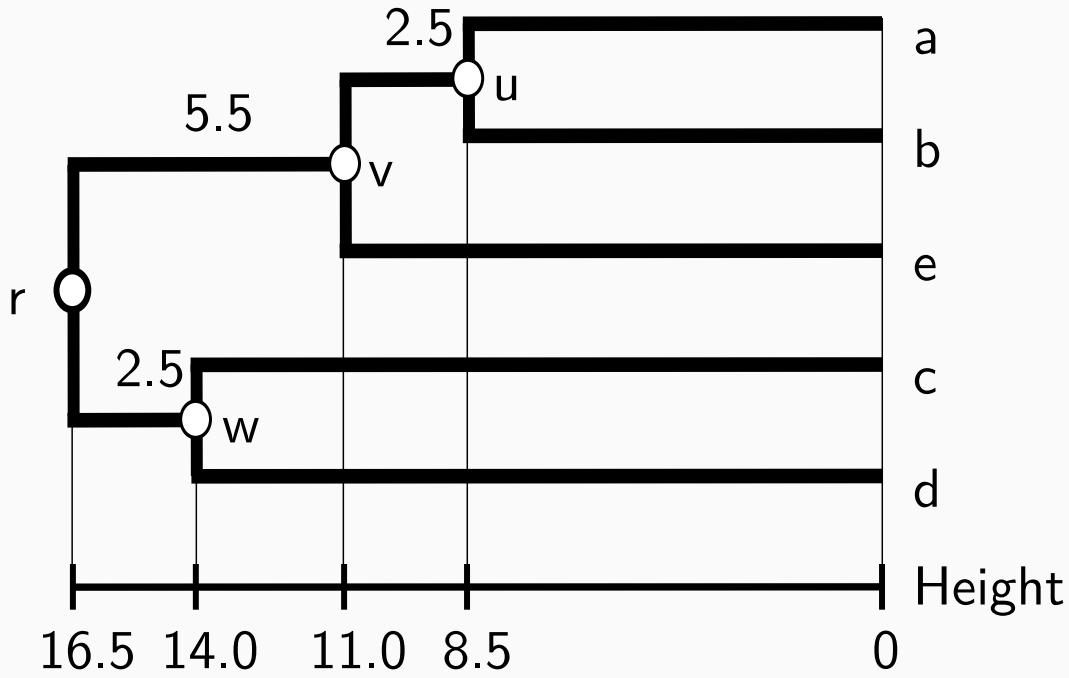
Input data

- Assembly does not produce a single putative sequence, but several contigs
- Process of scaffolding and gap filling requires some additional wet-lab work
- Contigs are approximate substrings with unknown location and orientation
- Input: contig sets of n organisms



Output - hierarchical clustering

- Output is a dendrogram of the species

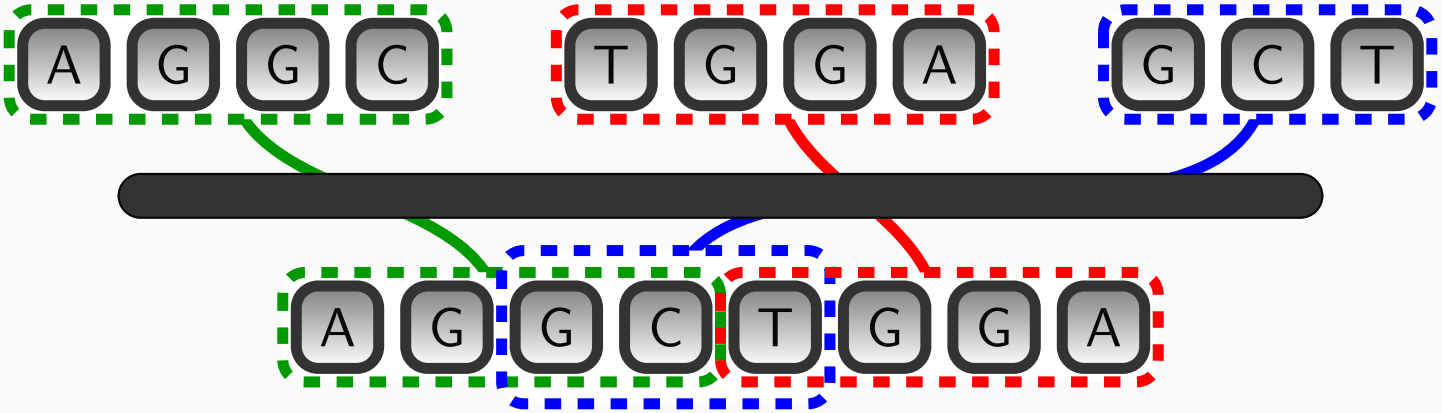


[By Manudouz (Own work) [CC BY-SA 4.0], via Wikimedia Commons]

RELATED WORK

Classical approach - first assemble.

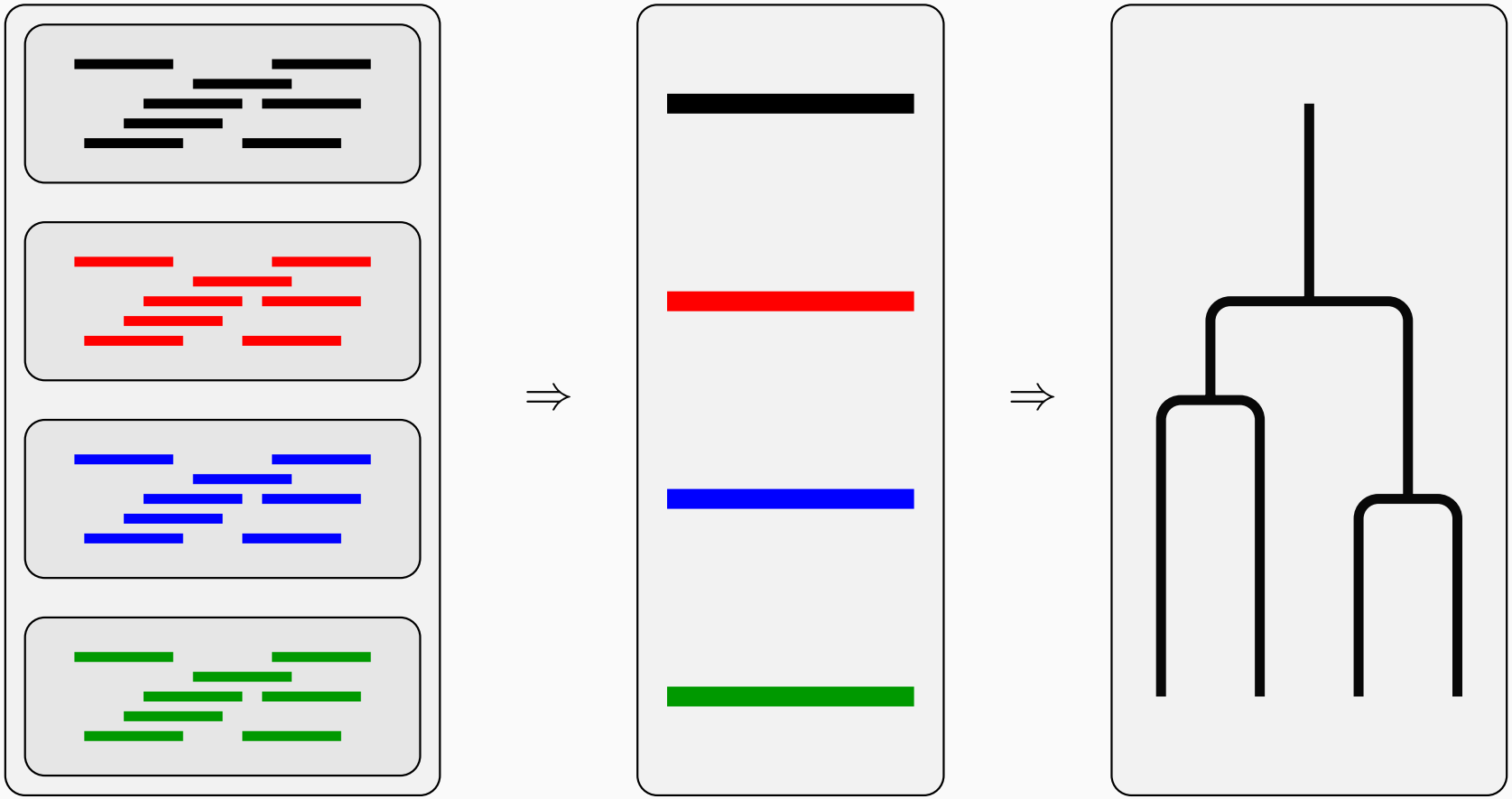
- The classical approach is to reconstruct the original sequence first.



- Genome assembly
- NP-hard problem

Classical approach - then cluster.

- Hierarchical clustering algorithm is used to build a dendrogram
- Dendrogram is based on edit distance



Alignment-free approaches

- Originally designed do avoid alignment step for genome comparison
- Genome broken into k-mers
- Some approaches work with read data

Comin and Schimd *BMC Bioinformatics* 2014, **15**(Suppl 9):S1
http://www.biomedcentral.com/1471-2105/15/S9/S1

BMC Bioinformatics

PROCEEDINGS Open Access

Assembly-free genome comparison based on next-generation sequencing reads and variable length patterns

Matteo Comin*, Michele Schimd

From RECOMB-Seq: Fourth Annual RECOMB Satellite Workshop on Massively Parallel Sequencing
Pittsburgh, PA, USA. 31 March - 05 April 2014

BRIEFINGS IN BIOINFORMATICS. VOL 15. NO 3. 343–353
Advance Access published on 23 September 2013
doi:10.1093/bib/bbt067

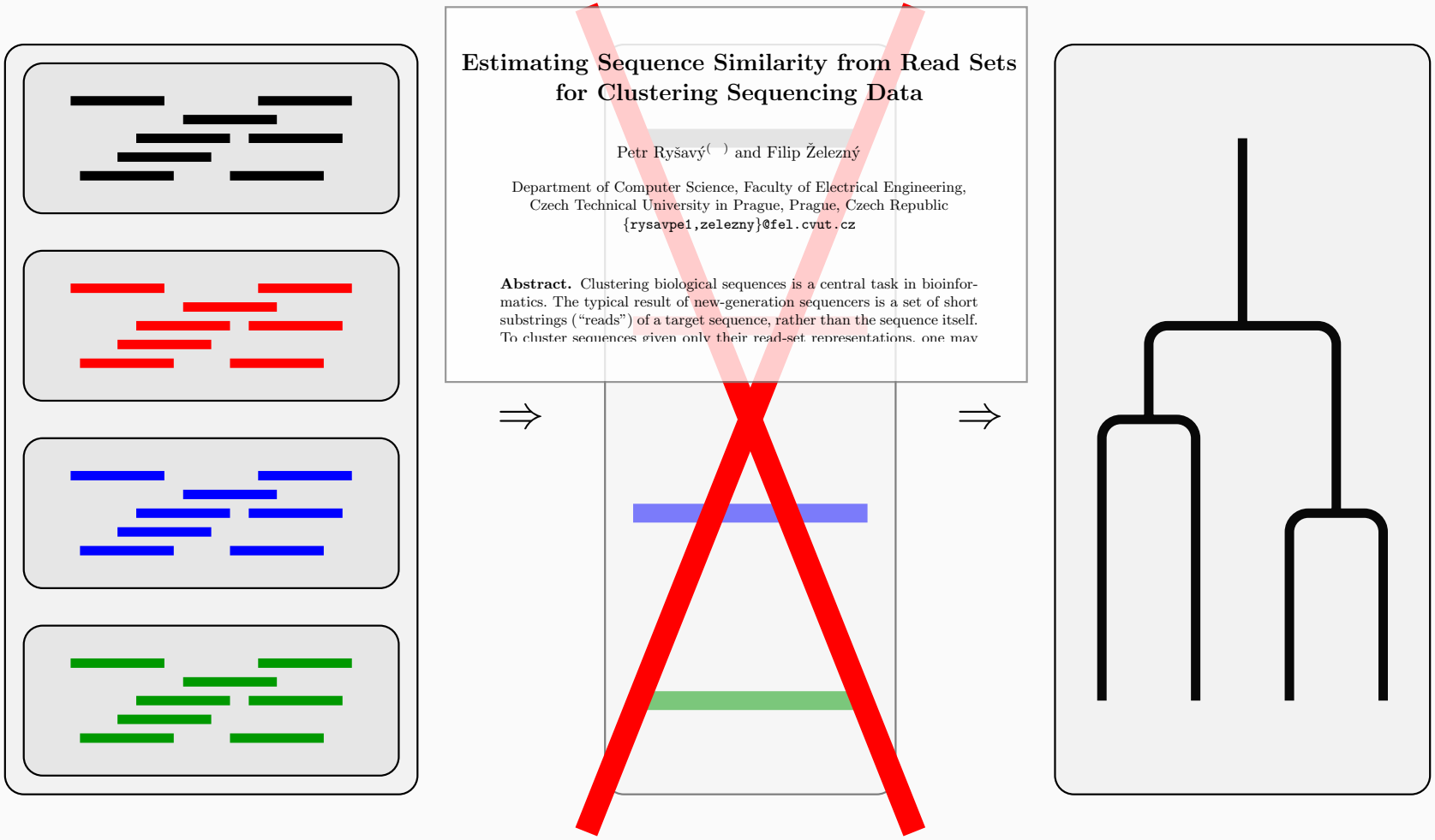
New developments of alignment-free sequence comparison: measures, statistics and next-generation sequencing

Kai Song, Jie Ren, Gesine Reinert, Minghua Deng, Michael S. Waterman and Fengzhu Sun

Submitted: 28th May 2013; Received (in revised form): 25th July 2013

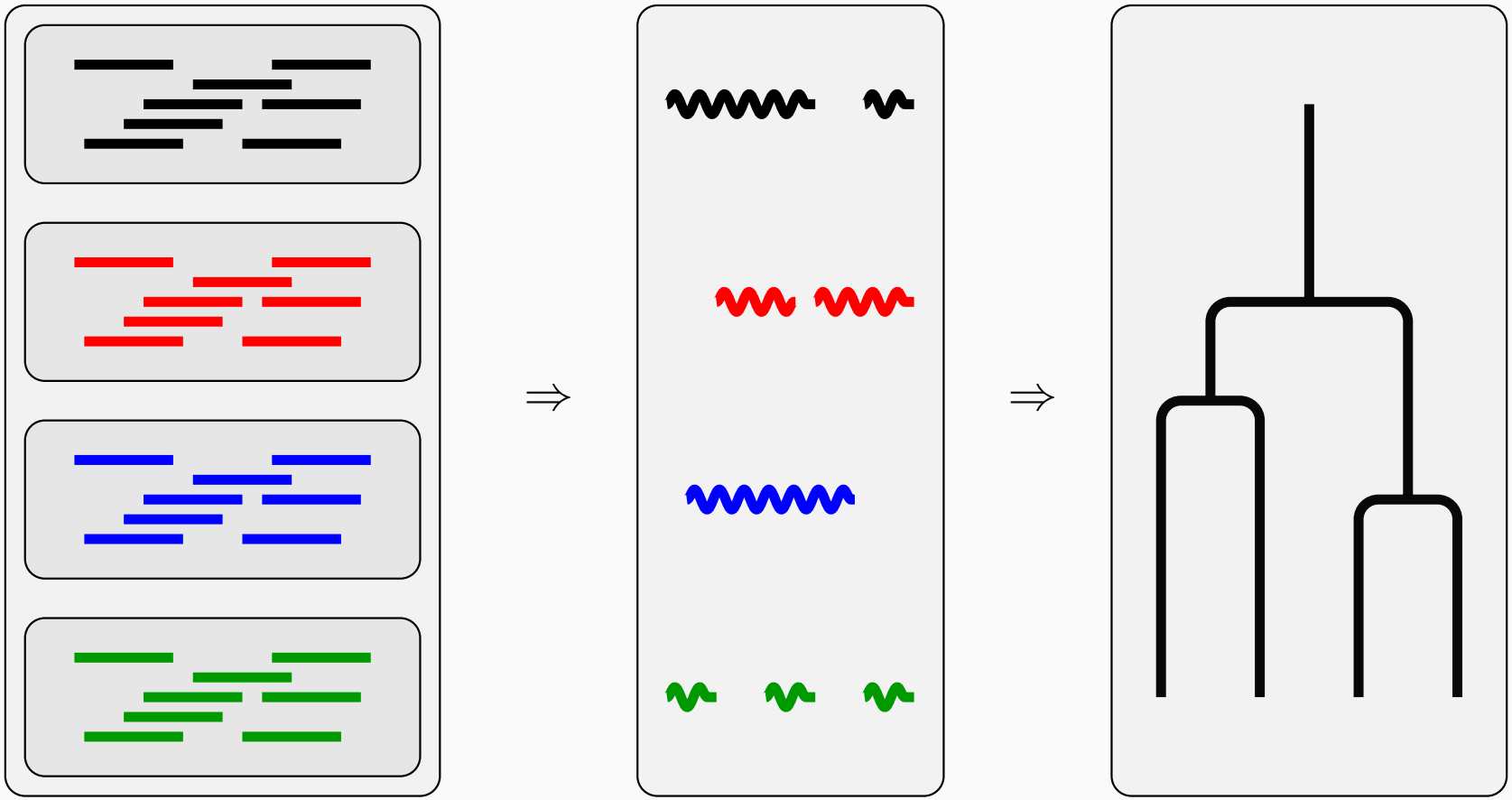
Our approach - skip assembly.

- Paper aims to avoid the assembly step
- Goal is to build dendrogram directly from the read sets



This paper

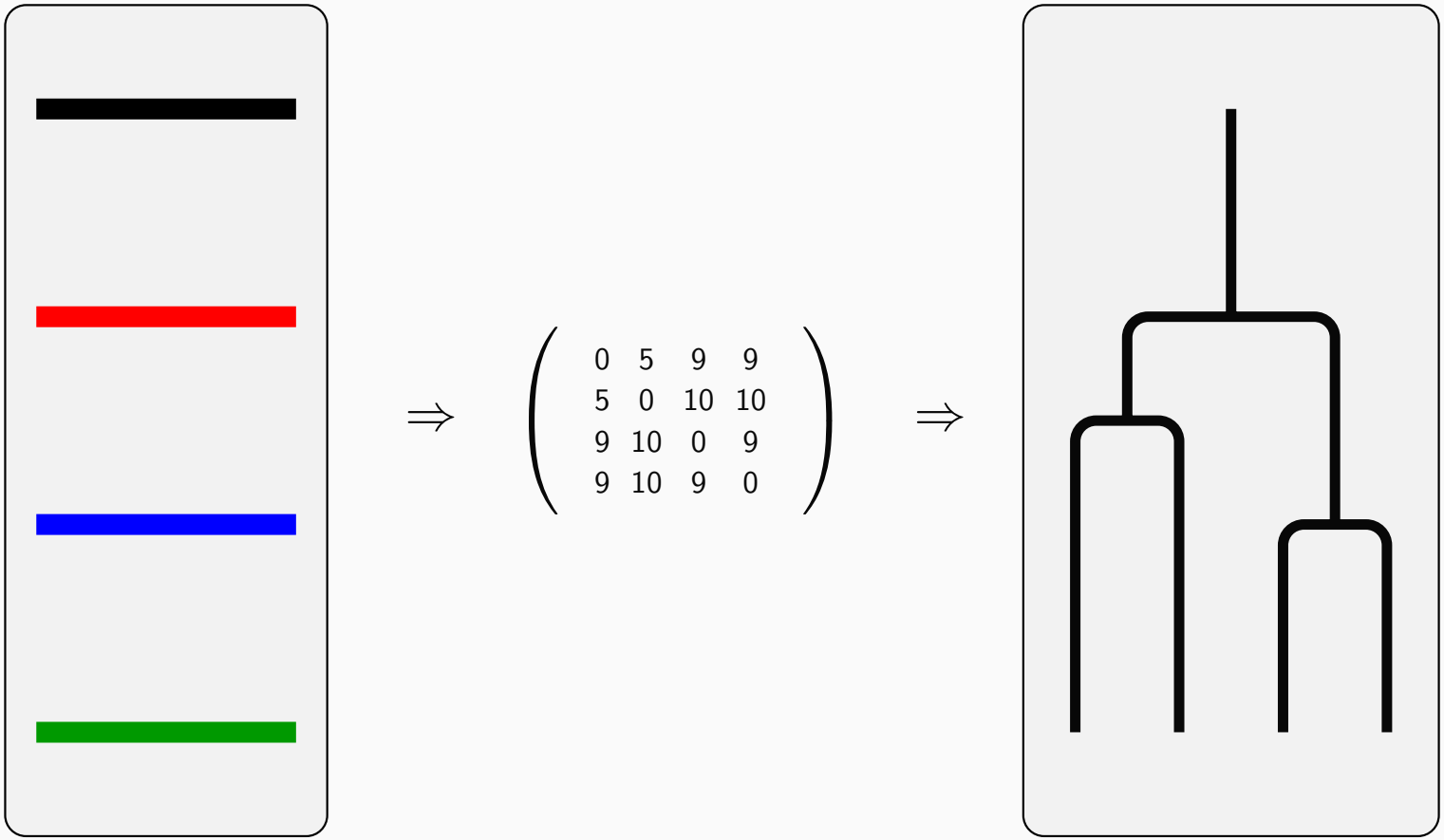
- Do not skip the assembly, do only the easy parts.



DISTANCE FUNCTION DESIGN

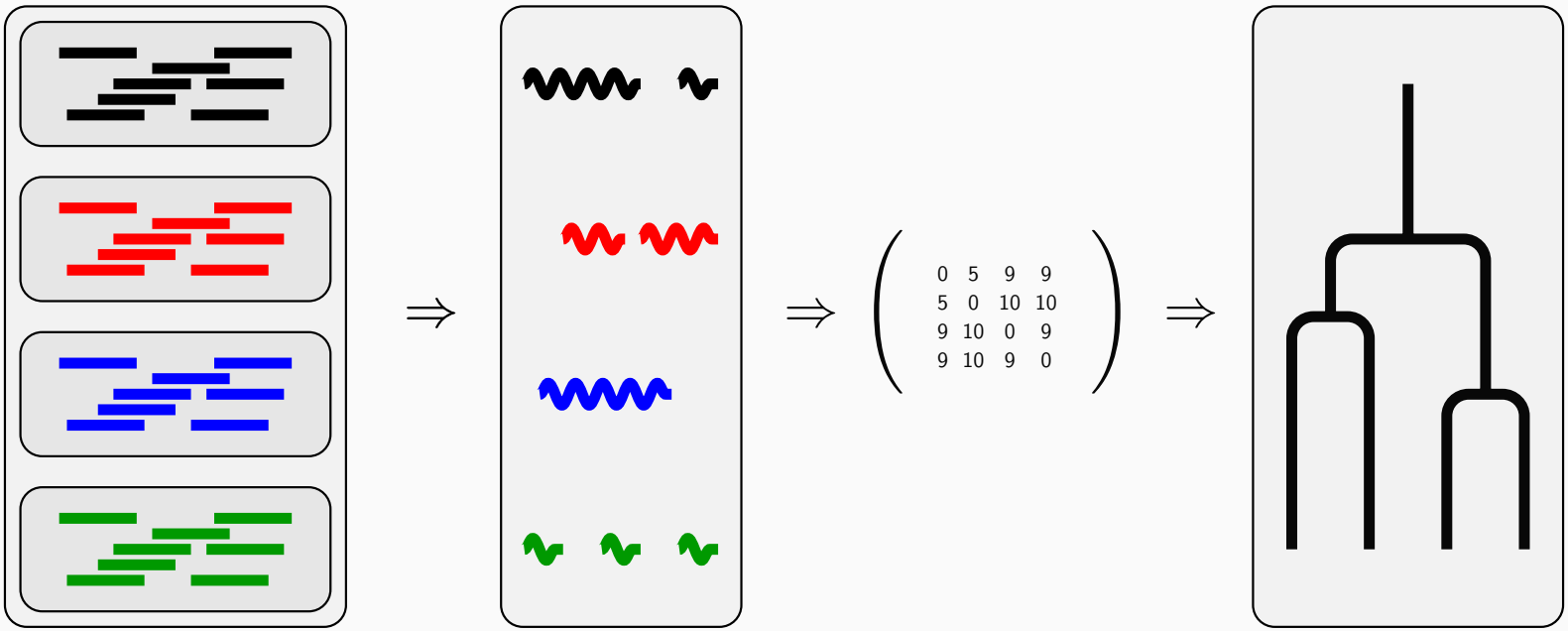
Clustering algorithms

- The only input of hierarchical clustering algorithms is a distance matrix
- This includes UPGMA and neighbor-joining



Key observation

- To build dendrogram, we need to approximate the distance matrix
- Measure that approximates edit distance needed



Problem reformulation

- Approximate edit distance between two sequences from their contig set representations.

Assumptions:

- Contigs are approximate non-overlapping substrings of the original sequence.
- All sequencing is done with the same coverage α.
- Reference genome is unknown.

Three step procedure

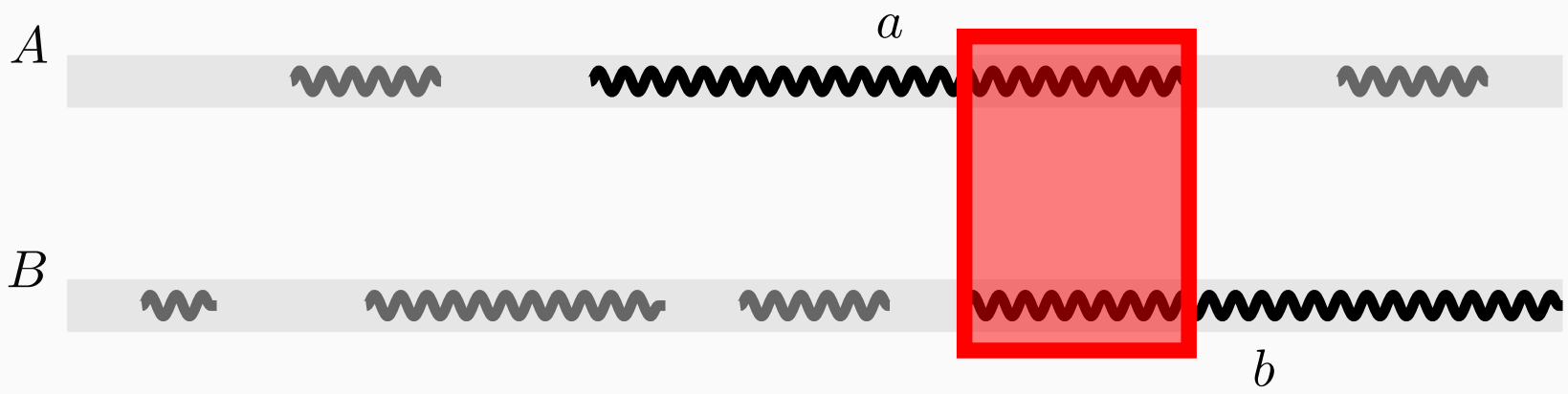
1. Calculate expected overlaps of contig pairs.
2. Select appropriate overlaps for each contig.
3. Average the distances over overlaps.

1) Estimating overlaps for contig pairs

- Consider two contigs a and b and assume they overlap in the optimal alignment
- Select overlap that minimizes the post-normalized edit distance

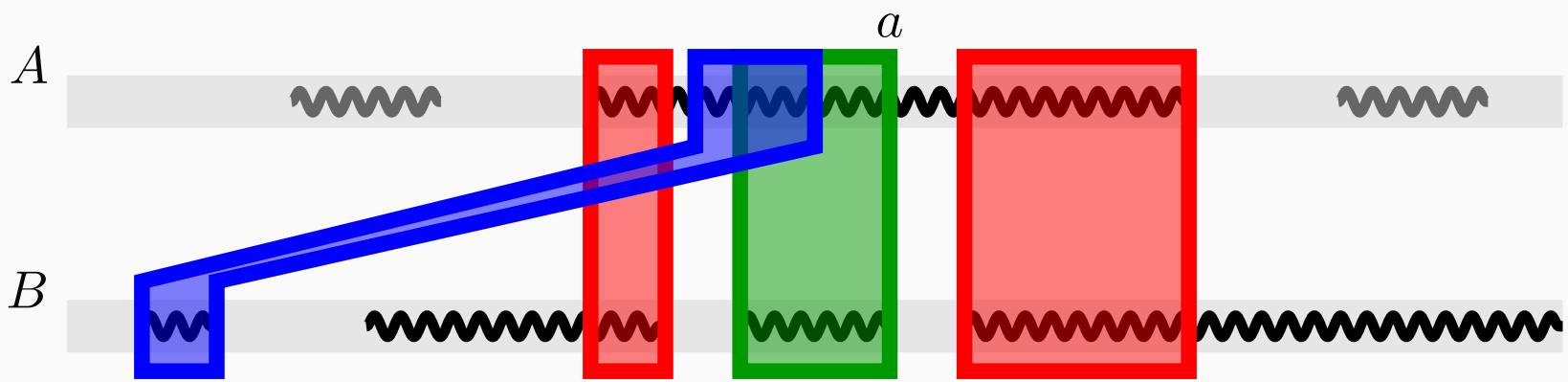
$$\overline{\mathsf{dist}}(a,b) = \frac{\mathsf{dist}(a,b)}{\max\{|a|,|b|\}}. \tag{1}$$

- Heuristic approach based on modification of Smith-Waterman algorithm



2) Estimating overlaps for contig sets

- For one contig we have overlaps with the other contig set
- Select non-overlapping regions that maximize the total value (post-normalized edit distance)
- Reduction to *weighted interval schedulling problem*

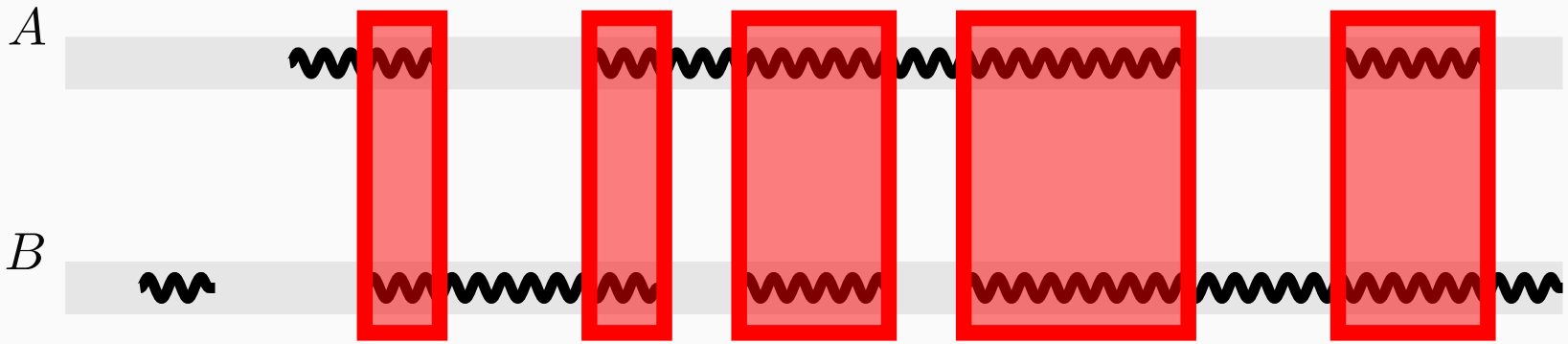


3) Combining the Results

- Sum distances of overlap pairs

$$d(C_A, C_B) = \sum_{(c,d) \in \text{overlap}(C_A, C_B)} \text{dist}(c, d).$$

- The sum does not capture contig size w.r.t. genome size

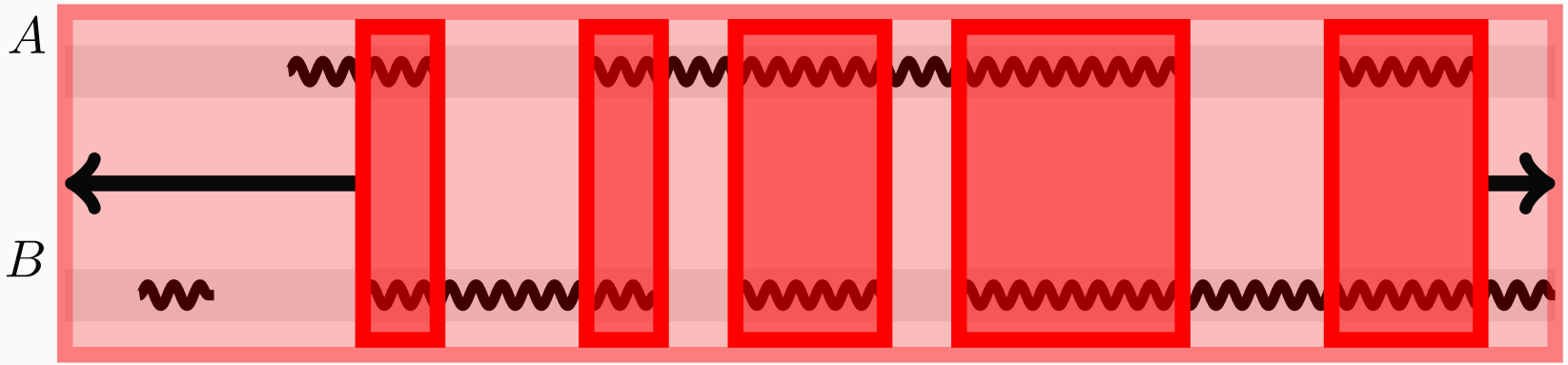


3) Combining the Results

- Normalize
- Divide by maximum possible distance of all overlaps ...
- ... and multiply by genome maximum distance

$$d(C_A, C_B) = \frac{\sum_{(c,d)\in \text{overlap}(C_A,C_B)} \text{dist}(c,d)}{\sum_{(c,d)\in \text{overlap}(C_A,C_B)} \max\{|c|,|d|\}} \cdot \frac{l \max\{|R_A|,|R_B|\}}{\alpha}.$$

- The resulting measure is not symmetric ...



3) Combining the Results

- ... average both directions

$$\text{Dist}(C_A, C_B) = \frac{d(C_A, C_B) + d(C_B, C_A)}{2}$$

EXPERIMENTAL RESULTS

Experimental setup

- *Influenza* datasets of viruses' DNA ($n = 13$)
  - Sampled with high range of coverage and read length
- Dataset of 81 *hepatitis* sequences, $(\alpha, l) \in \{10, 30, 50\} \times \{30, 70, 100\}$. ART used for Illumina sequencing simulation.
- Original DNA sequences used as a reference
- Two clustering algorithms (Neighbor-joining and UPGMA)
- Contigs produced by five common de novo assemblers (ABySS, edena, SSAKE, SPADes, velvet) and an idealized assembly algorithm
- Comparison with a straightforward approach that uses the longest contig

Measured characteristics:

- time (assembly time, distance matrix time, clustering time)
- Pearson's correlation coefficient measuring similarity of the distance matrix to the reference one
- Fowlkes-Mallows index measuring similarity of the clusterings

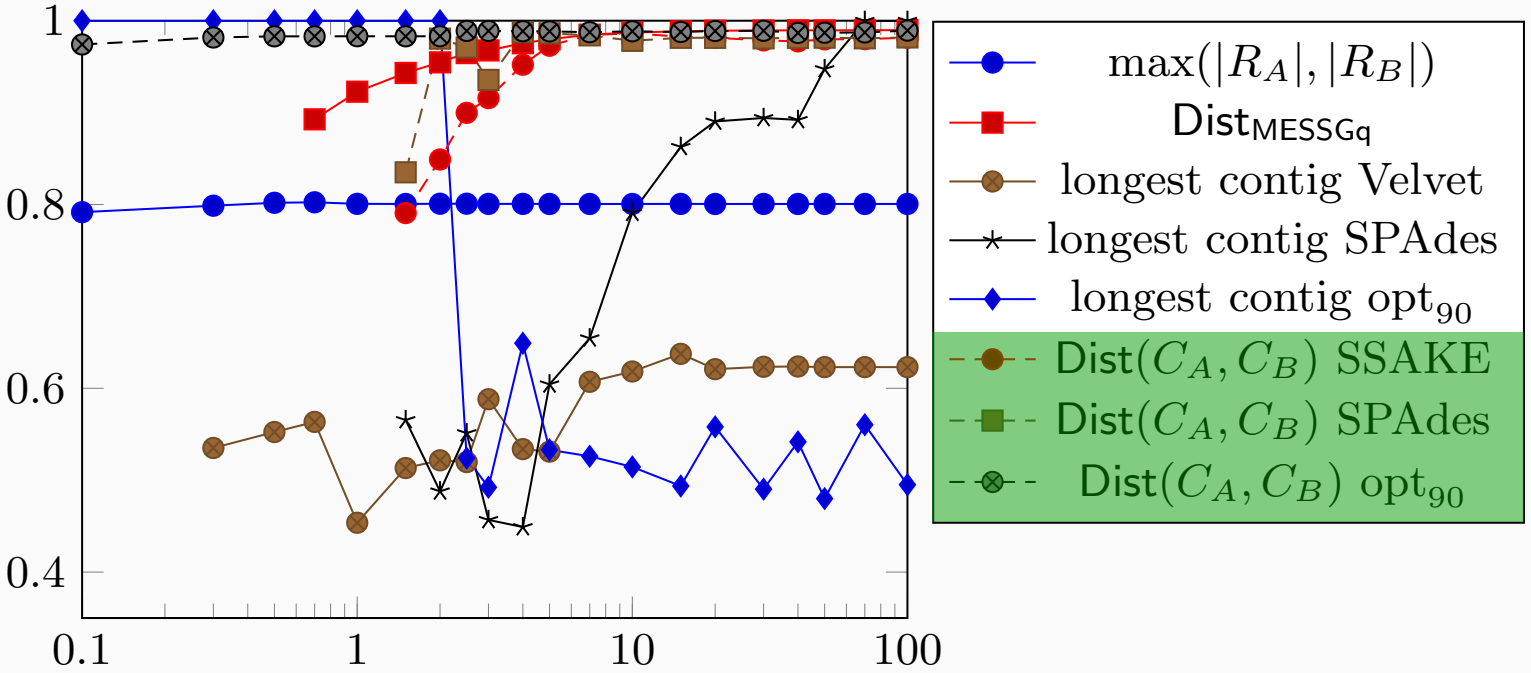
Results

- The proposed method produces results of good quality.

Dataset	method	finished	assem. ms	distances ms	corr.	B_4	B_8
Influenza	reference	112/112	0	2,518	1	1	1
	$\max(\|R_A\|, \|R_B\|)$	112/112	0	184	.801	.66	.32
	$\text{Dist}_{\text{MESSGq}}$	112/112	0	43,553	.966	1	.97
	longest contig Velvet	110/112	392	101	.569	.46	.23
	longest contig SPAdes	43/112	12,461	2,127	.751	.71	.56
	longest contig opt_{90}	112/112	0	1,208	.666	.63	.43
	$\text{Dist}(C_A, C_B)$ SSAKE	67/112	2,115	17,483	.949	.98	.87
	$\text{Dist}(C_A, C_B)$ SPAdes	43/112	12,461	20,968	.975	.99	.95
	$\text{Dist}(C_A, C_B)$ opt_{90}	112/112	0	22,239	.987	1	.98
Hepatitis	reference	9/9	0	2,145,104	1	1	1
	$\max(\|R_A\|, \|R_B\|)$	9/9	0	7,738	.181	.72	.83
	$\text{Dist}_{\text{MESSGq}}$	9/9	0	701,726	.897	1	.98
	longest contig Velvet	9/9	22,860	3,447	.234	.93	.54
	longest contig SPAdes	9/9	103,683	1,872,233	.591	.95	.84
	$\text{Dist}(C_A, C_B)$ SSAKE	9/9	96,446	29,465,436	.916	1	.9
	$\text{Dist}(C_A, C_B)$ Velvet	9/9	22,860	28,186,784	.966	1	.98

Results

- For high coverage data, the results are better than for low coverage.



Results

- The method is better than baseline and estimates based on longest contig only.

Dataset	method	finished	assem. ms	distances ms	corr.	B_4	B_8
	reference	112/112	0	2,518	1	1	1
	$\max(\lvert R_A\rvert, \lvert R_B\rvert)$	112/112	0	184	.801	.66	.32
	$\text{Dist}_{\text{MESSGq}}$	112/112	0	43,553	.966	1	.97
	longest contig Velvet	110/112	392	101	.569	.46	.23
Influenza	longest contig SPAdes	43/112	12,461	2,127	.751	.71	.56
	longest contig opt_{90}	112/112	0	1,208	.666	.63	.43
	$\text{Dist}(C_A, C_B)$ SSAKE	67/112	2,115	17,483	.949	.98	.87
	$\text{Dist}(C_A, C_B)$ SPAdes	43/112	12,461	20,968	.975	.99	.95
	$\text{Dist}(C_A, C_B)$ opt_{90}	112/112	0	22,239	.987	1	.98
	reference	9/9	0	2,145,104	1	1	1
	$\max(\lvert R_A\rvert, \lvert R_B\rvert)$	9/9	0	7,738	.181	.72	.83
	$\text{Dist}_{\text{MESSGq}}$	9/9	0	701,726	.897	1	.98
Hepatitis	longest contig Velvet	9/9	22,860	3,447	.234	.93	.54
	longest contig SPAdes	9/9	103,683	1,872,233	.591	.95	.84
	$\text{Dist}(C_A, C_B)$ SSAKE	9/9	96,446	29,465,436	.916	1	.9
	$\text{Dist}(C_A, C_B)$ Velvet	9/9	22,860	28,186,784	.966	1	.98

Results

- Runtime is comparable to reference up to a constant factor (3 tables vs. 1)

Dataset	method	finished	assem. ms	distances ms	corr.	B_4	B_8
	reference	112/112	0	2,518	1	1	1
	$\max(\|R_A\|, \|R_B\|)$	112/112	0	184	.801	.66	.32
	$\text{Dist}_{\text{MESSGq}}$	112/112	0	43,553	.966	1	.97
	longest contig Velvet	110/112	392	101	.569	.46	.23
Influenza	longest contig SPAdes	43/112	12,461	2,127	.751	.71	.56
	longest contig opt_{90}	112/112	0	1,208	.666	.63	.43
	$\text{Dist}(C_A, C_B)$ SSAKE	67/112	2,115	17,483	.949	.98	.87
	$\text{Dist}(C_A, C_B)$ SPAdes	43/112	12,461	20,968	.975	.99	.95
	$\text{Dist}(C_A, C_B)$ opt_{90}	112/112	0	22,239	.987	1	.98
	reference	9/9	0	2,145,104	1	1	1
	$\max(\|R_A\|, \|R_B\|)$	9/9	0	7,738	.181	.72	.83
	$\text{Dist}_{\text{MESSGq}}$	9/9	0	701,726	.897	1	.98
Hepatitis	longest contig Velvet	9/9	22,860	3,447	.234	.93	.54
	longest contig SPAdes	9/9	103,683	1,872,233	.591	.95	.84
	$\text{Dist}(C_A, C_B)$ SSAKE	9/9	96,446	29,465,436	.916	1	.9
	$\text{Dist}(C_A, C_B)$ Velvet	9/9	22,860	28,186,784	.966	1	.98

CONCLUSION

Conclusion

- Our method may result in less wet-lab work needed for (dis)similarity machine learning
- From contigs, we estimate sequence similarity of original sequences
- Good quality regarding Pearson's correlation coefficient between distance matrices
- `https://github.com/petrrysavy/ida2017`

Future work

- Improve runtime to find overlap faster
- Combine results with the previous work to get advantages of both
- Do a more thorough experimental evaluation (in progress)

Thank you for your attention.
Time for questions!

Acknowledgment. This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS17/189/OHK3/3T/13 and CTU UPE chapter.