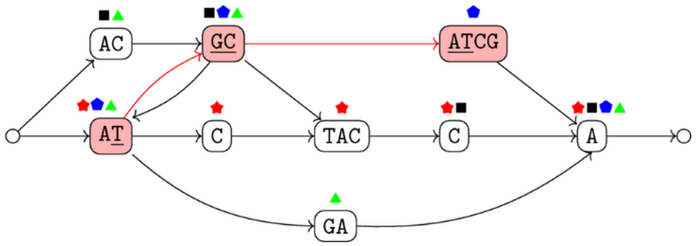
Topology and computational hardness of contig variation graphs

Author: Matej Kliment (mkliment@student.tudelft.nl)

Supervisor: Jasmijn A. Baaijens

1. Background

- Viral genomes (e.g., HIV, HCV) exist as mixtures of closely related variants (haplotypes), not as a single sequence, making reconstruction from sequencing data challenging
- Haplotype reconstruction can be modeled using contig variation graphs, where contigs are connected by read-supported adjacencies and edge flows represent estimated abundances [1]
- In this framework, each haplotype corresponds to a source-sink path, and reconstruction reduces to the Minimum Flow Decomposition (MFD) problem
- MFD is NP-hard and exhibits extreme runtime variability in practice, even for graphs of similar size [2]
- Graph structure, rather than size alone, strongly influences computational difficulty, graphs with many valid paths are typically harder to solve [3]
- This work investigates which structural properties of contig variation graphs predict MFD runtime, using correlation and classification analyses on real and synthetic datasets



Example of contig variation graph segment

2. Research Question

How can we characterize the differences between graphs that are easy versus hard to solve in the context of Minimum Flow Decomposition for viral genome analysis?

How can we predict for which graphs will the MFD solver finish and for which it will time-out?

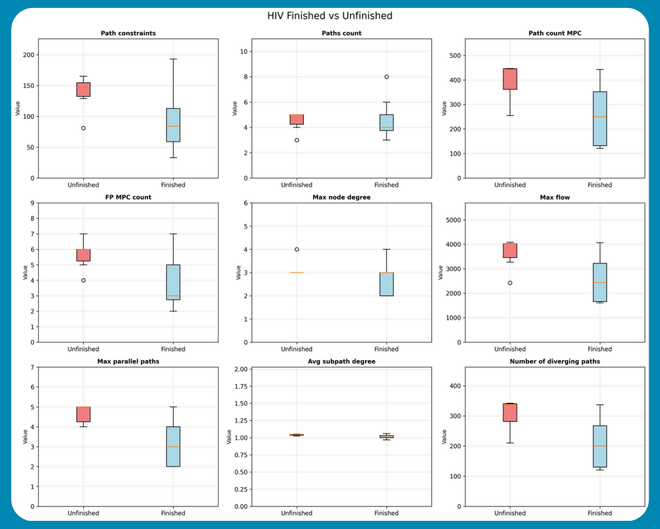
Which structural features correlate the most with the MFD solution runtime?

3. Methodology

- **Graph data**: Three datasets of contig variation graphs from HIV, HCV, and idealized ("perfect") viral assemblies
- **Feature extraction**: Computed 10 descriptive structural graph features (e.g. branching, flow values, minimal path cover)
- **Solver runs**: Each graph solved using an ILP-based Minimum Flow Decomposition solver with a fixed time limit of 8 hours
- **Correlation analysis**: Measure how strongly each feature relates to solver runtime (Pearson & Spearman)
- **Classification study**: Compare features of graphs that finish vs. graphs that timed out
- **Synthetic experiments**: Modify one feature at a time (maximum flow) in a simple graph instance to test causal effects on runtime

4. Results

Classification: Time-outs occur primarily in graphs with higher haplotype counts; once haplotype number is fixed, most structural features fail to clearly separate easy from hard instances.



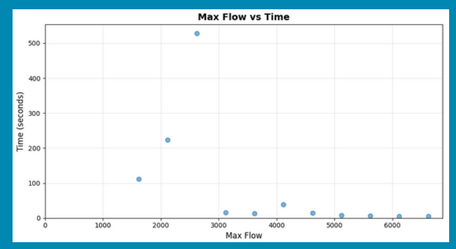
Classification of finished vs time-out HIV graphs: figure shows 9 features on HIV dataset, having no clear classifying threshold for differing the time-out data from the finished.

Runtime correlates most strongly with path-based features such as path count and minimum path cover as well as maximum flow, confirming that haplotype complexity is a major driver of MFD difficulty.

Attribute	HIV	HCV	PG
Number of vertices	0.6127	0.3082	0.9164
Number of edges	0.6108	0.3007	0.9168
Num of subpath constraints	0.6200	0.5044	0.9323
Paths count	0.8687	0.0909	0.9046
Non-overlapping MPC size	0.5855	0.0909	0.9184
MPC size	0.7509	0.6478	0.8963
Max node degree	0.6058	0.3548	0.4287
Maximum flow	0.7635	0.6084	0.9059
Max parallel paths	0.6849	0.6478	0.5195
Avg subpath degree	0.4585	0.0560	0.5454
Long-short path difference	0.5703	0.3012	-0.1522
Number of diverging paths	0.5746	0.1961	0.9176

Synthetic graph generation to isolate maximum flow from other features: Synthetic tests show that increasing maximum flow alone does not consistently increase runtime, indicating it is not an independent cause of computational hardness

Runtime compared with maximum flow on a known simple graph instance, systematicaly increased by a constant (c=500) for 10 samples



5. Conclusions

The runtime of Minimum Flow Decomposition is mainly driven by path-related complexity, especially the number of haplotypes and closely related measures such as minimum path cover MPC. Many intuitive graph features, including local node degree and maximum flow, do not independently explain computational difficulty once this path complexity is taken into account. These results show that solver performance can often be anticipated from graph structure alone, providing useful guidance for preprocessing and optimization in viral haplotype reconstruction.

[1] J. A. Baaijens, L. Stougie, and A. Schonhuth. Strain-aware assembly of genomes from mixed samples using flow variation graphs. in Research in Computational Molecular Biology, 2020

[2]. Vatinlen, F. Chauvet, P. Chretienne, and P. Mahey. Simple bounds and greedy algorithms for decomposing a flow into a minimal set of paths. (2008), The European Journal of Operational Research.

[3] A. I. Tomescu F. H. C. Dias. Accurate flow decomposition via robust integer linear programming. Institute of Electrical and Electronics Engineers (IEEE), 2024.