Error Correction in Graphs for Viral DNA Reconstruction

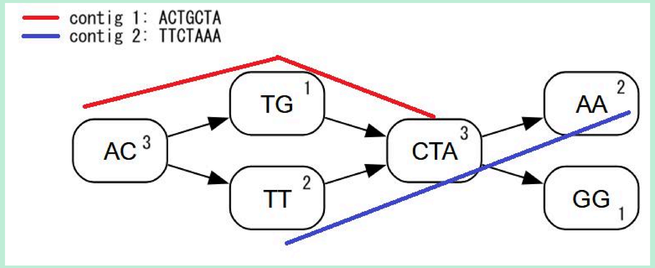
Author: Senne Drent
s.drent@student.tudelft.nl

Supervisor: Jasmijn Baaijens

TU Delft Delft University of Technology

1. Introduction

- Haplotypes are distinct copies of the genome.
- Minimum Flow Decomposition (MFD) can be used to reconstruct haplotypes from a DNA sample.
- A DNA sample is encoded into a contig variation graph.



- Errors in the contig variation graph cause longer runtime and less accurate assemblies.
- MFD is NP-hard; runtime grows exponentially for amount of haplotypes in a graph.

2. Research questions

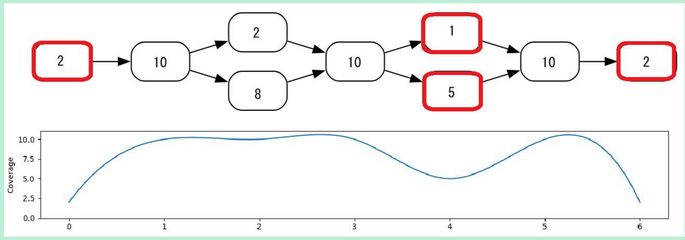
How can a contig variation graph be pre-processed to improve performance of the MFD genome aware haplotype assembly?

- How can noise in the abundance be corrected using the coverage of DNA reads at certain positions in the viral sample?
- How do different thresholds for low-abundance noise filtering impact the solution in runtime and accuracy?
- How can cycles be resolved efficiently without impacting the solution results?
- What is the effect of automatic cycle resolution on the accuracy of the MFD?

3. Methodology

Abundance correction

- Certain areas in the DNA sample may vary in coverage, which causes bias in the contig variation graph.



- The strategy takes a reference sequence from the contig variation graph.
- DNA reads are aligned to the sequence to get coverage.
- Nodes are placed on reference to get coverage for each node.
- Abundance is scaled based on coverage.
- MFD performance is compared with and without applying abundance correction.

Low-abundance node filtering

- The threshold is defined as a set percentage of the maximum abundance.
- Find balance between MFD runtime and accuracy.
- threshold configurations are tested between 0% and 3%, with 0.1% steps.

Cycle resolution

- MFD is only applicable with DAG graphs.
- The Contig variation graph can contain cycles caused by alignment errors of DNA sequences or repeats.
- Two algorithms will be tested on ability to correctly remove cycles: H-voting and CycleDFS.

4. Results

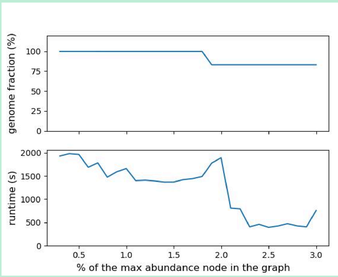
Abundance correction

- Runtime reduction of 15.8% on the Poliovirus dataset.
- ZIKV shows slower runtime.
- Overall, there is no impact on accuracy.

	Abundance correction	
	Without	With
3-strain ZIKV		
Assembly error (%)	0.0033	0.0033
Genome fraction (%)	99.029	99.029
Runtime (s)	0.87	0.94
6-strain Poliovirus		
Assembly error (%)	0.0987	0.0987
Genome fraction (%)	99.687	99.687
Runtime (s)	2102	1768

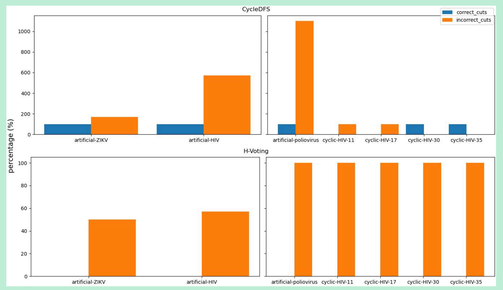
Low-abundance node filtering

- Minimal filtering causes faulty haplotypes in the assembly.
- Found a runtime reduction of 25% without impacting performance.



Cycle resolution

- H-voting removes cycles using a different objective.
- CycleDFS breaks up to ten times more incorrect edges than correct edges.



5. Conclusion

- Abundance correction is promising for improving runtime but currently fragile.
- For the discussed dataset, Low-abundance node filtering is optimal at threshold of 1.8%.
- Neither cycle resolution algorithms are applicable to the contig variation graph.

Future work

- investigate mapping errors in abundance correction.
- Verify results on more datasets.
- Evaluate other cycle resolution algorithms.