Weight sets for paths in minimum flow decomposition

Agnese Ēlerte
A.Elerte-1@student.tudelft.nl

Supervisor / Responsible Professor
Jasmijn Baaijens

TU Delft

1. Introduction

MFD (Minimum Flow Decomposition) - a minimum set of weighted paths so that for each node, the weights of the paths going through it sum to the weight of the node

- Used in viral genome reconstruction
- Nodes are genetic sequences; weights are the abundances of the sequence in the data
- Nodes are connected with an edge if the sequences are found next to each other in the data
- The resulting paths are longer genetic sequences, representing mutated virus sequences
- MFD is an NP-Hard problem; running it in an unoptimized Integer Linear Programming (ILP) solver is slow
- By providing the ILP with a set of weights to choose from for the paths, the runtime can be reduced

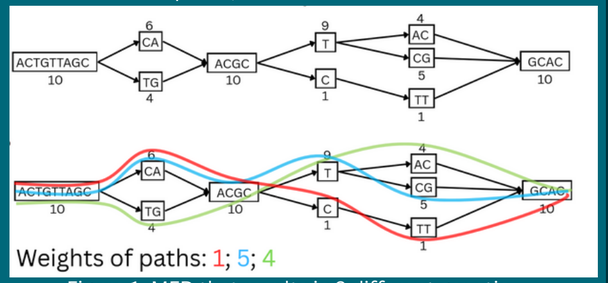

Figure 1. MFD that results in 3 different genetic sequences (paths) and their weights

2. Research question

How can we find a set of candidate weights that reduce the runtime bottlenecks of flow decomposition using integer linear programming for viral genome analysis?

- What methods for selecting weight sets for MFD ILP have already been used or tested? What methods are suggested by previous research?
- How does the selection of weight sets for MFD ILP affect the runtime and accuracy of the ILP solver?
- How does the level of noise in node weights affect the runtime and accuracy of the ILP solver using different weight selection methods?

3. Experiment

Datasets from simulated perfect graphs (no noise) and HIV/HCV viruses were run with different weight selection methods. In addition, noise was added to the perfect graphs to experiment with different controlled levels of noise in node weights.

Weight selection methods tested:

- No weights added
- Powers of 2: $2^i + 2^j$, where $j < i$ and $2^i + 2^j <=$ largest node weight
- Cluster with subtracting (CWS): Cluster all node weights, extract cluster centers, remove centers that can be expressed as a sum of two other centers, calculate differences between cluster centers to estimate smaller weights
- Cluster with MinGenSet (CWM): Cluster all nodes, extract cluster centers, and use MinGenSet to generate weights

For methods relying on clustering, two methods are used - KMeans and Agglomerative clustering.

4. Results

- KMeans is more accurate for estimating weights for graphs, but sometimes slower than Agglomerative Clustering, KMeans is chosen as the main method for research
- Runtime improves for cluster-based methods (Figure 2); weight errors increase if a weight set is added (Figure 3)
- Cluster-based methods reduce runtime, but increase misassemblies in the paths; Cluster with MinGenSet causes the solver to be more inaccurate than cluster with subtracting

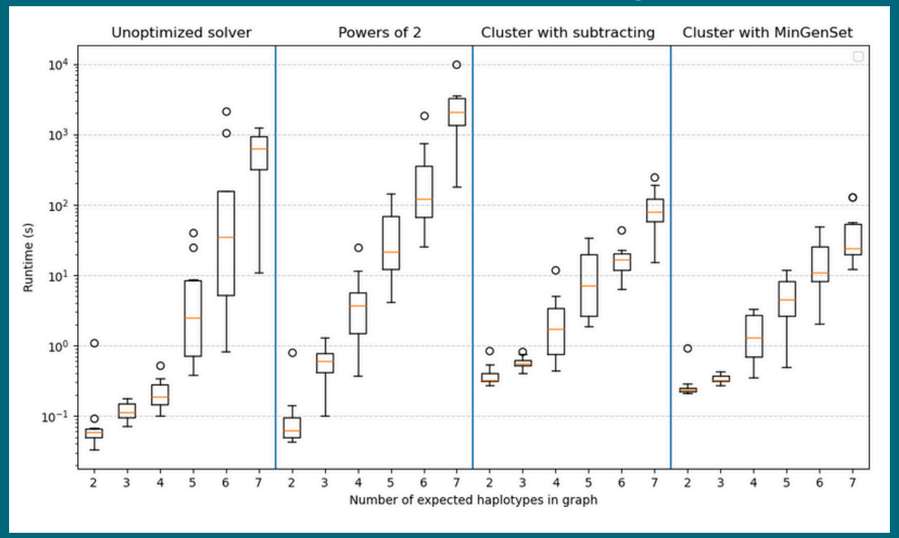

Figure 2. Runtimes for the 4 weight generation methods, grouped by expected number of haplotypes

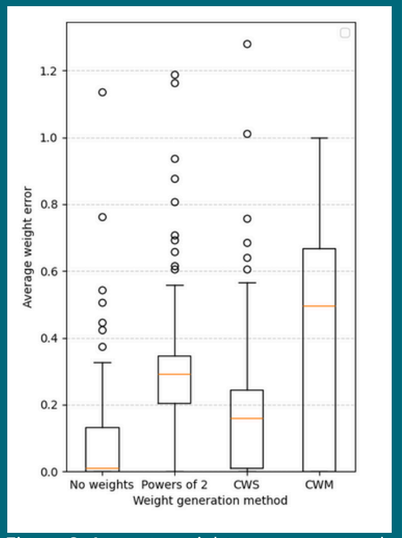

Figure 3. Average weight errors per graph for the 4 weight generation methods

5. Conclusions and discussion

- Using weight sets for the MFD solver can be used to improve the runtime
- Using powers of 2 does not provide any benefits; not including the node weights is likely the cause of this
- Cluster-based methods provide a significant improvement for runtime
- Weight sets that contain more accurately estimated weights do not guarantee a lower runtime
- There are sometimes multiple paths assigned to the same true path, possibly due to inaccurate weight sets
- There is no consistent correlation between runtime and the accuracy/length of the weight set; for perfect graphs, correlations are found, for HIV/HCV graphs, no significant correlation is found
- The method of clustering has a large influence on the weight accuracy
- Noise in graphs causes the decomposition to be more inaccurate, both for paths and their assigned weights, for all weight generation methods
- Experimental setup was not ideal; some samples were not run due to previous samples getting stuck

6. Future recommendations

- Explore heuristics that assign some weights from a set and others from the solver, or remove a number of the highest weight paths in the graph and allow the solver to find the remaining ones
- Explore further clustering methods for cluster-based methods and further refine the distance threshold for agglomerative clustering
- Implement a solution for graphs that have multiple paths with the same weight (clustering methods would only generate one weight for them)