



De Bruijn Graph Construction in Genome Assembly



Project Summary

- Objectives: To construct De Bruijn Graph in Genome assembly
- Scope:



Research Design : Methods and Algorithms

- The answer to the stated problem now was to find a path through the graph that traverses each edge exactly once, or in other words **Eulerian trail**.
- Reads are broken into smaller fragments of a specified size k .
- k -mers are identified and a de Bruijn graph with $(k-1)$ -mers as nodes and k -mers as edges drawn as described in the text. A Eulerian path is traced through this network resulting in the reconstruction of the original genome sequence.



Research Design : Methods and Algorithms

Take all $(k-1)$ -mers from the set of k -mers

Construct a multi-graph with nodes being $k-1$ -mers; draw an edge between two $k-1$ mers only if the two $k-1$ mers are taken from the same read.

Graph constructed this way is guaranteed to have a Eulerian trail, follow the trail and connect the nodes to form our original sequence.

this algorithm can be used to assemble k -mer reads

Build kmers or reads from the input string

After generating the reads, debruijnize the reads to form $(k-1)$ mers

Make node – edge map using the obtained $k-1$ mers

Construct an Eulerian trail using the $k-1$ mers as nodes by drawing an edge between them only if they are taken from the same reads

Graph constructed will have a Eulerian trail and thus print the Debruijn graph.

Follow or trace the Eulerian path and connect the nodes to get the original sequence

Results

Discussion

Conclusion



Reference

- E. Drezen, G. Rizk, R. Chikhi et al., “Gatb: genome assembly & analysis tool box,” *Bioinformatics*, vol. 30, no. 20, pp. 2959–2961
- R. Li, H. Zhu, J. Ruan et al., “De novo assembly of human genomes with massively parallel short read sequencing,” *Genome Research*, vol. 20, no. 2, pp. 265–272, 2010.
- T. C. Conway and A. J. Bromage, “Succinct data structures for assembling large genomes,” *Bioinformatics*, vol. 27, no. 4, pp. 479–486, 2011