22.1 Genome assembly

This entire course has been focused on studying algorithms for DNA sequence data. But how do we obtain such sequence data in the first place? It is technologically impossible to obtain the sequence of the human genome one nucleotide at a time, from start to finish. Instead, sequencing machines break the genome up into many small fragments, called reads, and output the sequence of these shorter segments.

A popular analogy for the genome assembly problem is as follows. Suppose we have 100 copies of the Sunday Times. We then feed this stack of newspapers through a paper shredder. A significant portion of the resulting paper shreds are then tossed out completely. Furthermore, the printing factory informs us that we have received a defective batch of copies, and that some percentage of the words are misspelled. Our goal is to reconstruct the complete Sunday Times.

Let’s see how this description corresponds to the process of DNA sequencing.

- **100 copies of the Sunday Times.** The genome is amplified before it is broken up into reads. If this were not done, full genome assembly would be impossible.

- **Feed this stack through a paper shredder.** Only a few hundred nucleotides can be efficiently sequenced with low probability of error. Thus, we need to break the 3 billion base pair genome into many small reads.

- **A significant portion is thrown out.** Not all of the resulting reads are sequenced. We need to deal with the problem of missing information.

- **Some of the words are misspelled.** Sequencing technologies are not perfect. Some nucleotides in a read are incorrectly identified.

Hopefully you are convinced that this is a difficult problem. A major breakthrough was to reframe genome assembly in the language of graph theory.

22.1.1 Eulerian cycles and paths

**Definition 22.1** An Eulerian path is a path through a graph in which each edge is traversed exactly once. An Eulerian cycle is an Eulerian path that starts and ends at the same node.
Definition 22.2 Consider a directed graph $G = (V, E)$.

- The **indegree** of a vertex $v \in V$ is the number of edges that terminate at $v$ and is denoted by $d_{in}(v)$.
- The **outdegree** of a vertex $v \in V$ is the number of edges that originate at $v$ and is denoted by $d_{out}(v)$.
- A vertex $v \in V$ is **balanced** if $d_{in}(v) = d_{out}(v)$. The graph $G$ is balanced if every vertex in $V$ is balanced.
- The graph $G$ is **strongly connected** if it is possible to reach any vertex from every other vertex.

Theorem 22.3 (Euler’s Theorem) Every directed graph that is balanced and strongly connected is also Eulerian, i.e. contains an Eulerian cycle.