A Compression Algorithm for DNA Sequences and Its Applications in Genome Comparison (Summary)

DNA sequences only contain 4 nucleotide bases so 2 bits are enough to store each one. The compressibility of DNA sequences is supported by the fact that they contain multiple approximate repeats, even though this is often shadowed by mutation, cross-over, translocation, reversal events and sequencing errors.

GenCompress is a lossless DNA compression algorithm which is based on approximate matching. It even aids in comparing two sequences using a relatedness measure. Two other lossless compression algorithms are Biocompress-2 and Cfact, both being based on exact matching.

The approximate matching algorithm implemented considers three standard edit operations: replace, insert and delete. It can be easily deduced that an infinite number of edit sequences can be derived from a single transformation – the list of edit operations being referred to as the Edit Transcription $\lambda(u, v)$. Thus, by knowing u and $\lambda(u, v)$, v can be derived/encoded using any of these four methods: two bits encoding, exact matching, approximate matching and approximate matching using edit operation sequence. The third case returns the minimal number of bits, with the first case coming in at a close second.

GenCompress generalises the dictionary based, Lempel-Ziv compression algorithms for approximate matching. It is a one-pass algorithm. In order to limit the search, a certain condition C is used as a constraint; i.e. only approximate matches which satisfy $C = (k, b)$ are searched for – experimentally (12, 3) is best. A compression gain function G is also defined to check if a specific repeat leads to encoding benefits. Thus, with both C and G, an optimal prefix can be deduced using a parsing procedure. By analysing G, it was found that no approximate match in the database to help save bits exists. However, the approximate reverse palindrome can be detected.

The algorithm's main aim is to acquire an accurate compression ratio for DNA sequences; time complexity being considered afterwards. Exhaustively searching for optimal prefixes is too time consuming so some observations are made: an optimal prefix always ends right before a mismatch and the optimal edit operation sequence is reflective in a sense. Using both of these, the optimal prefix and palindrome searches can be constructed.

Two versions of GenCompress were tested – one with replacement operations only and the other with all the edit operations – against BioCompress-2 and Cfact. It was concluded that approximate matching achieves the best compression ratio and is best for finding common parts in DNA sequences. If however there are not enough approximate repeats in the sequence, then BioCompress-2 performs better.

Furthermore, the relatedness or mutual information between two DNA sequences can be calculated using the 'distance' between their pairs; close sequences being close on the evolutionary tree. Minimum alignment scores, genome rearrangement distance and reversal distance are usually used for closely related sequences. For not closely related sequences, a symmetric distance is defined using Kolmogorov complexity and it can also be used to build evolutionary trees from un-alignable DNA sequences such as complete genomes. Note that for this simple alignment distance, GenCompress has been converted into conditional compression.