ComS 331 **Spring 2025** **Name:** Mobina Amrollahi

HW 4 Due: Feb 28th 2025

1. **NOTE: This is a somewhat open-ended assignment. Read the questions carefully. To derive your answers, use your imagination, and state your assumptions.**

   A DNA strand can be represented as a (very long) string w over the alphabet $\{A, C, G, T\}$. For example, the human DNA has length $\approx 3 \times 10^9$. Because of the double-helix nature of DNA, we really should be talking about the *base pairs* A-T and G-C, in the sense that DNA is made of base-paired sequences: for example, instead of $w = ACTGGACT$, we could instead look at its *reverse complement* $\overline{w^R} = AGTCCAGT$, obtained by reversing w and then applying to it the "complement" homomorphism $A \to T, T \to A, C \to G, G \to C$.

   To match DNA from a sample to a *reference* DNA w, or even to build *de novo* a reference DNA w, a *sequencer* can be used to generate a large number of relatively short substrings appearing in w (or in w^R, the sequencer has no way to tell in which direction the piece of DNA is oriented when it reads it) called *reads*. Sequencing technology is rapidly evolving, but let's assume for simplicity that it is possible to generate a large number (e.g., 10^9) reads of length 100 each, in a reasonable time (e.g., hours).

   In reality, the length of these reads may vary a little, sometimes we may have reads over $\{A, C, G, T, N\}$, where "N" indicates that the sequencer was not able to determine the exact value being read, and sometimes the sequencer may even misread a value; let's ignore these possibilities.

   (a) What is the number μ of possible reads of length 100 over $\{A, C, G, T\}$?

   Answer

   If we allow any symbols of the given alphabet at each one of the 100 positions, then we have $\mu = 4^{100}$ possible reads.

   (b) Assuming that the human reference DNA has length exactly equal to 3×10^9, what fraction of the μ possible reads is present in the human DNA?

   Answer

   To extract all possible 100-base reads from the human reference DNA, we will be sliding a window of size 100 over the sequence until we reach the last 100 symbols, and we cannot slide anymore:

   $$\frac{3 \times 10^9 - 100 + 1}{4^{100}} \sim \frac{3 \times 10^9}{4^{100}}$$

   (c) Describe how one could use an MDD to encode all the reads present in the human reference DNA, and then efficiently (question: how efficiently?) determine whether a sample read is present in the human reference DNA (application: a CSI technician collects some genetic material at a crime scene and wants to determine whether it may be of human origin).

   Answer

   First, to construct an MDD based on the human reference DNA, extract all possible 100-base reads from the human reference DNA by sliding a window of size 100 over the sequence and putting them in a set $\mathcal{L}$. Hence, $\mid \mathcal{L} \mid = 3 \times 10^9 - 100 + 1$. Further, Insert each read into the MDD by:

i. Start at the root

ii. For each base in the read, traverse or create a child node corresponding to that base.

iii. At level 100, mark the terminal node indicating the completion of a valid read.

Therefore, each node at level i represents one of the four possible nucleotides (A, C, G, T). A path from the root to a terminal node corresponds to a valid 100-base read found in the human genome. The following MDD has exponential space complexity [2]. However, by applying reduced-order MDDs, we can likely reduce the space complexity to polynomial [1]. For instance, a commonly used optimization technique for MDDs, called *null pointer elimination*, removes all edges that always lead to the '0' leaf, as well as nodes that have only such edges.

To analyze the time complexity of checking whether a base read exists in the DNA reference, we construct the corresponding Deterministic Finite Automaton (DFA) from the given MDD. To construct such a DFA, we follow a process that involves mapping each MDD node to a DFA state while ensuring deterministic transitions. The time complexity of checking whether a given string reaches a final state in a DFA is $\mathcal{O}(n)$, where n is the length of the input string. Hence, checking for the presence of a read in the MDD also runs in linear time [3].

References

[1] Henrik Reif Andersen, Tarik Hadzic, John N Hooker, and Peter Tiedemann. A constraint store based on multivalued decision diagrams. In *Principles and Practice of Constraint Programming–CP 2007: 13th International Conference, CP 2007, Providence, RI, USA, September 23-27, 2007. Proceedings 13*, pages 118–132. Springer, 2007.

[2] David Bergman, Andre A Cire, Willem-Jan Van Hoeve, and John Hooker. *Decision diagrams for optimization*, volume 1. Springer, 2016.

[3] Shuhei Denzumi, Ryo Yoshinaka, Hiroki Arimura, and Shin-ichi Minato. Sequence binary decision diagram: Minimization, relationship to acyclic automata, and complexities of boolean set operations. *Discrete applied mathematics*, 212:61–80, 2016.