Exercise VII, Computational Complexity 2024

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. Solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

Randomised Complexity

1 Characterisation of ZPP.

(a) Prove that $L \in \mathsf{ZPP}$ if and only if there is a polynomial-time probabilistic TM M that outputs values in $\{0, 1, ?\}$ such that for every input x, we have $M(x) \in \{L(x), ?\}$ and $\Pr[M(x) = ?] \leq 1/2$. *(Hint: Suppose we have a PTM M with expected runtime $t(n)$. Use Markov's inequality to show that M halts within $2t(n)$ steps with probability at least $1/2$.)*

(b) Show that $\mathsf{ZPP} = \mathsf{RP} \cap \mathsf{coRP}$. *(Hint: How to combine an* RP*-style algorithm and an* coRP*-style algorithm into a single* ZPP*-algorithm?)*

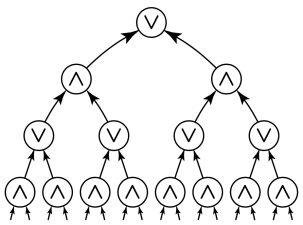
2 Prove that $\mathsf{NP} \subseteq \mathsf{BPP}$ implies $\mathsf{NP} = \mathsf{RP}$.

3 Consider the 4-bit function $f(x) = (x_1 \wedge x_2) \vee (x_3 \wedge x_4)$. Suppose we want to evaluate f using a simple query algorithm (a decision tree) that queries, in arbitrary adaptive order, individual input variables $x_i \in \{0, 1\}$ until it learns enough information about x in order to output $f(x) \in \{0, 1\}$.

(a) Show that any deterministic query algorithm for f needs to query all variables in the worst case. That is, for any deterministic query strategy, there is some input x such that the strategy makes 4 queries.

(b) Show that there is a randomised 0-error strategy (i.e., ZPP style) that, for every input x, makes 3 queries *in expectation* and always outputs $f(x)$ correctly.

(c) (*) For $n = 2^k$, define $g\colon \{0,1\}^n \to \{0,1\}$ as the function computed by a depth-k binary tree that has $\vee$- and $\wedge$-gates on alternating levels, and with n variables at the leaves:



Show that every deterministic query algorithm requires n queries to compute g but that there exists an 0-error randomised algorithm with expected query cost

$$3^{k/2} \;=\; n^{\log_2 \sqrt{3}} \;=\; n^{0.79248\ldots}.$$

4 Suppose Alice holds an n-bit string $x \in \{0,1\}^n$ and Bob holds an n-bit string $y \in \{0,1\}^n$. Alice and Bob want to decide whether $x = y$ using a *one-way communication protocol* where Alice sends a single message $m = m(x)$ to Bob, and Bob outputs one bit indicating whether or not $x = y$. Their goal is to minimise the bit-length $|m|$ of the message.

(a) Show that any deterministic protocol requires message length $|m| \geq n$.
(*Hint: Exercise V-4.*)

(b) Suppose Alice and Bob now have access to a shared random string $r \in \{0,1\}^\ell$ and that Alice's message $m = m(x,r)$ can depend on r. Show that Alice and Bob can succeed with probability $\geq 2/3$ (i.e., BPP style) by sending only $|m| \leq O(1)$ bits.
(*Hint: Suppose $\ell = n$. What can you say about the mod-2 inner products $\langle x, r\rangle \bmod 2$ and $\langle y, r\rangle \bmod 2$? Here $\langle x, r\rangle = \sum_{i=1}^{n} x_i r_i$.*)