# Problem Set #1

CMSC 858L
Instructor: Daniel Gottesman

Due on Gradescope Sep. 24, 2026, 11:59 PM

General instructions: No use of LLMs. If you collaborate or use any outside resources, remember to cite them in your solutions.

**Problem #1. Different error cutoffs (50 pts.)**

Define $\mathrm{BPP}_\epsilon$ the same way as BPP, but with an error cutoff of $1/2+\epsilon$ instead of 2/3. That is, for $\mathrm{BPP}_\epsilon$,

1. If $x \in L$, then $\mathrm{Prob}(A(x) = 1) \geq 1/2 + \epsilon$.
2. If $x \notin L$, then $\mathrm{Prob}(A(x) = 0) \geq 1/2 + \epsilon$.

In this problem, we will consider $\epsilon$ to be a function of $|x|$, although in part a, it is a constant function.

Results in this problem should be unconditional (that is, not depending on additional complexity-theoretic assumptions, such as the assumption that P = BPP).

**Hint:** For parts of this problem, you may want to consider that a randomized algorithm can be thought of as a deterministic algorithm that reads from a pre-determined string of uniformly random bits as well as from its usual input.

You may also want to use some statistical tail bounds. If you are not very familiar with these, let me suggest Hoeffding's inequality: When $X_1, \ldots, X_n$ are independent random variables with values in the interval $[a, b]$, let $X$ be their sum and let $\mu$ be the mean value of $X$. Then $\forall\, \delta > 0$,

$$\mathrm{Prob}(X \geq (1+\delta)\mu) < \exp\left(-\frac{2\delta^2\mu^2}{n(a-b)^2}\right) \tag{1}$$

a) (10 pts.) Prove that for any constant $\epsilon$, $0 < \epsilon < 1/2$, $\mathrm{BPP}_\epsilon = \mathrm{BPP}$.

b) (10 pts.) Prove that if $\epsilon = 1/f(|x|)$, with $f(|x|) = O(\mathrm{poly}(|x|))$, then $\mathrm{BPP}_\epsilon = \mathrm{BPP}$.

c) (15 pts.) Prove that if $\epsilon = 2^{-|x|}$, then $\mathrm{BPP}_\epsilon \subseteq \mathrm{PSPACE}$.

d) (15 pts.) Find a value for $\epsilon$ (which again will be a function of $|x|$) such that you can prove that $\mathrm{BPP}_\epsilon = \mathrm{P}$.

**Problem #2. Variations of BQP(50 pts.)**

Let $\mathcal{G} = \{G_1, G_2, \ldots, G_g\}$ be a finite efficiently-computable approximately universal set of quantum gates each acting on at most 2 qubits. Given any gate $G_a \in \mathcal{G}$ acting on qubits $i$ and $j$ with $0 \leq i, j \leq n-1$, we can specify it using $r = \lceil \log g \rceil + 2\lceil \log n \rceil$ bits as $(a, i, j)$. In this problem and generally throughout the course log is base 2. If $G_a$ is a 1-qubit gate, we simply set $j = 0$. We can then specify a quantum circuit $Q$ of size $T$ using gates from $\mathcal{G}$ with $rT$ bits by listing the gates of $Q$ in order.

Let us say that a quantum Turing machine *moves deterministically* if given a classical input string $x$, then after every time step, the measurement of register $D$ determining the movement or halting of the machine can only have a single outcome $|\mathrm{left}\rangle$, $|\mathrm{right}\rangle$, or $|\mathrm{halt}\rangle$ with non-zero probability. It is allowed that that the

outcome of measuring $D$ at step $t$ may depend on the input $x$. We will say that a quantum Turing machine *has a deterministic classical output* if, given a classical input string $x$, it always halts with a classical output string $f(x)$ in a specified output region of the tape.

In this problem, you are asked to design circuits and you may do so using gates from a gate set $\mathcal{G}'$ of your choice. $\mathcal{G}'$ should include efficiently-computable gates and act on a bounded number of qubits (which could be more than 2). You do not have to use the same $\mathcal{G}'$ for all problem parts.

a) (15 pts.) Design a programmable quantum circuit family with the following properties: It uses gates from $\mathcal{G}'$ of your choice. Given an input $(x, y)$ which specifies a quantum circuit $Q$ acting on $n$ qubits (including ancillas) and using $T$ gates from $\mathcal{G}$, the programmable circuit outputs $Q|y\rangle$ and uses at most $O(\text{poly}(n, T))$ gates from $\mathcal{G}'$. Here you may assume that $y$ is a classical bit string and that $Q$ has a quantum output.

b) (10 pts.) Suppose $Q$ is a quantum circuit that uses gates from $\mathcal{G}$ and mid-circuit measurements and conditions future gates on the outcomes of the measurements. Find a circuit $Q'$ that only involves measurements at the end and has the same output distribution as $Q$ when given the same input. You may use ancillas and a gate set $\mathcal{G}'$ of your choice, but the size of $Q'$ should be $O(\text{poly}(T))$ if $Q$ uses at most $T$ gates on all sequences of measurement outcomes.

For this problem part, you can assume that gates depend directly on the outcome to a single measurement, but the result can be extended to allow gates conditioned on the outcome of efficient classical computations based on multiple measurement outcomes.

c) (15 pts.) Suppose we have a Turing machine $(K, \Sigma, \delta)$ which moves deterministically and for which $\delta$ may be implemented exactly as a quantum circuit $U$ using $u$ gates from $\mathcal{G}$. Let $v = \lceil \log K \rceil$ and $w = \lceil \log \Sigma \rceil$, so $U$ takes $v + w$ input qubits and outputs $v + w + 2$ qubits (with the last 2 qubits representing $D$ having an unused state). Further suppose that on input $x$, the Turing machine halts after at most $T(|x|)$ steps.

Design a quantum circuit family that on input $x$ gives the same output as the Turing machine $(K, \Sigma, \delta)$ on that same input. You may use ancilla qubits and a gate set $\mathcal{G}'$ of your choice. You may also use mid-circuit measurements and may condition future quantum gates on measurement outcomes (but of course you could invoke part b to make a fully unitary circuit). The quantum circuit family should use at most $O(\text{poly}(T(|x|)))$ gates.

d) (10 pts.) We defined BQP using uniform families of quantum circuits of polynomial size. Uniform means computable on a classical Turing machine using $O(\text{polylog})$ space. Suppose we define a complexity class BQP' which instead uses quantum circuits $Q_n$ of size $O(\text{poly}(n))$ which are given as outputs of a quantum Turing machine on input $1^n$. The Turing machine moves deterministically, has a deterministic classical output, and on input $1^n$ halts in time $O(\text{poly}(n))$. Show that $BQP' \subseteq BQP$. (The reverse inclusion is straightforward, so this shows that BQP' = BQP.)

You can assume BQP' is defined using quantum circuits made up of gates from $\mathcal{G}$ and that BQP uses quantum circuits made up of gates from $\mathcal{G}'$ of your choice, provided that $\mathcal{G}'$ is also approximately universal.

Recall that $1^n$ is a bit string of $n$ 1's. It is used to ensure that the Turing machines runs in a time polynomial in $n$, not in $\log n$.