Lecture 7

Remark 2. For any $f\colon \{0,1\}^n \to \{0,1\}$ that depends on all n bits, $D(f) \geq \log(n+1)$.

There exists a family of $f_k\colon \{0,1\}^{k+2^k} \to \{0,1\}$ depending on all $n \coloneqq k+2^k$ bits such that $D(f) = k+1 \approx \log(n)$. Why? Consider the address function $f\colon \{0,1\}^{k+2^k} \to \{0,1\}$ defined by $f(a_1a_2\dots a_kx_0x_1\dots x_{2^k}) = x_{\overline{a_1\dots a_k}}$, where $\overline{a_1\dots a_k}$ is the integer represented by $a_1\dots a_k$ in binary. (Note that this does not contradict the lower bound since $\log_2(k+2^k+1) \leq D(f) \leq k+1 = \log_2(2^{k+1})$.)

The $\mathrm{OR}_n\colon \{0,1\}^n \to \{0,1\}$ function is defined by

$$\mathrm{OR}_n(x) = x_1 \vee x_2 \vee \cdots \vee x_n. \tag{26}$$

Proposition 1. *$D(\mathrm{OR}_n) = n$.*

Proof. $D(\mathrm{OR}_n) \leq n$ is obvious (what's the DDT?).

For $D(\mathrm{OR}_n) \geq n$. Suppose for contradiction that there is a DDT T with $\mathrm{depth}(T) < n$ that computes OR_n. Consider the root-to-leaf path defined by following edges labelled by 0. We may assume wlog (without loss of generality) that the leaf vertex on this path is labelled by 0, else $T(0^n) = 1 \neq \mathrm{OR}_n(0^n)$, contradiction. Suppose the vertices on this path are labelled by $i_1,\dots,i_d$, where $d < n$. Let $j \in [n] - \{i_1,\dots,i_d\}$ (exists since $d<n$). Let $x \in \{0,1\}^n$ be the all-zeros bitstring except for a 1 at position j. Then $T(x) = 0 \neq \mathrm{OR}_n(x)$ contradiction. □

Definition 6 (Randomized decision tree (or query algorithm))**.** A randomized decision tree is a probability distribution $\mathcal{T}$ over deterministic decision trees.

Definition 7 (Randomized query computation)**.** Given $x \in D$ and a randomized decision tree $\mathcal{T}$, we write $\mathcal{T}(x)$ for the random variable on Γ defined by:

$$\forall i \in \Gamma,\ \Pr[\mathcal{T}(x) = i] \coloneqq \Pr[T(x) = i \mid T \leftarrow \mathcal{T}]. \tag{27}$$

Let $\epsilon \in (0,1/2)$. We say that a randomized decision tree $\mathcal{T}$ computes f with (two-sided) error ϵ if

$$\forall x \in D,\ \Pr[\mathcal{T}(x) = f(x)] \geq 1-\epsilon. \tag{28}$$

Note that

$$\Pr[\mathcal{T}(x) = f(x)] = \Pr[T(x) = f(x) \mid T \leftarrow \mathcal{T}] = \sum_T \Pr[T \mid T \leftarrow \mathcal{T}] \cdot \mathbb{1}[T(x) = f(x)]. \tag{29}$$

Definition 8 (Randomized query complexity)**.** Given a randomized decision tree (RDT) T, its depth is defined by

$$\mathrm{depth}(\mathcal{T}) \coloneqq \max\{\mathrm{depth}(T) \mid \Pr[T \mid T \leftarrow \mathcal{T}] > 0\}. \tag{30}$$

Then for $\epsilon \in (0,1/2)$,

$$R_\epsilon(f) \coloneqq \min\{\mathrm{depth}(\mathcal{T}) \mid \mathcal{T}\ \text{RDT},\ \mathcal{T}\ \text{computes}\ f\ \text{with bounded-error}\ \epsilon\}. \tag{31}$$

Also standard to write

$$R(f) \coloneqq R_{1/3}(f). \tag{32}$$

Definition 9 (Quantum query algorithm)**.** A quantum query algorithm of depth $d \in \mathbb{N}$ is defined by the following data:

1. $w \in \mathbb{N}$. (Called the dimension of the workspace of the algorithm.)
2. $d+1$ unitary matrices $U_0, U_1, \dots, U_d \in \mathbb{C}^n \otimes \mathbb{C}^m \otimes \mathbb{C}^w = \mathbb{C}^{nmw}$.
3. A Γ-outcome projective measurement $\mathcal{M} \coloneqq \{\Pi_s \mid s \in \Gamma\}$ on $\mathbb{C}^{nmw}$.

Definition 10 (Quantum oracle)**.** For $x \in \{0,\dots,m-1\}^n$, the quantum oracle of x is the unitary matrix $O_x \in \mathbb{C}^{nm \times nm}$ defined by

$$O_x \left|i\right\rangle \left|j\right\rangle = \left|i\right\rangle \left|(j + x_i) \mod m\right\rangle, \tag{33}$$

for all $i \in [n]$ and $j \in [m]$. (And linearly extended. For $z \in \mathbb{Z}$, $z \mod m$ is the unique integer in the range $\{1,\dots,m\}$ with the same remainder as z when divided by m.)[5]

In the special case of $m = 2$, the definition is equivalent to[6]

$$O_x \left|i\right\rangle \left|b\right\rangle = \left|i\right\rangle \left|b \oplus x_i\right\rangle, \tag{34}$$

for all $i \in [n]$ and $b \in \{0,1\}$, where $\oplus$ denotes XOR and $\left|b\right\rangle$ represents a 1-qubit quantum state.

[5] Note that this definition is slightly different than in my lecture notes because in this course, I decided to use $\left|1\right\rangle, \dots, \left|m\right\rangle$ to denote the standard basis vectors in $\mathbb{C}^m$, whereas in the lecture notes, I used $\left|0\right\rangle, \dots, \left|m-1\right\rangle$.

[6] Recall that for $m = 2$, we also use $\left|0\right\rangle$ to denote $(1,0)^\top$ and $\left|1\right\rangle$ to denote $(0,1)\top$.

Definition 11 (Quantum query computation)**.** Given $x \in D$ and a quantum query algorithm $\mathcal{A}$, we write $\mathcal{A}(x)$ for the random variable on Γ defined by:

$$\forall i \in \Gamma,\ \Pr[\mathcal{A}(x) = i] := \|\Pi_i \cdot U_d(O_x \otimes \mathbb{1}_w) \dots U_1(O_x \otimes \mathbb{1}_w) U_0 \left|1\right\rangle \|^2, \tag{35}$$

where $\mathbb{1}_w \in \mathbb{C}^{w \times w}$ is the identity matrix and we recall $\left|1\right\rangle \in \mathbb{C}^{nmw}$ is the first computational basis vector. (Note there are d occurrences of O_x on the RHS.)

For $\epsilon \in (0, 1/2)$, we say that a quantum query algorithm $\mathcal{A}$ computes f with (two-sided) error ϵ if

$$\forall x \in D,\ \Pr[\mathcal{A}(x) = f(x)] \geq 1 - \epsilon, \tag{36}$$

where the probability is over the random variable $\mathcal{A}(x)$.

Definition 12 (Quantum query complexity)**.** For $\epsilon \in (0, 1/2)$, $Q_\epsilon(f)$ is defined to be the minimum depth of any quantum query algorithm that computes f with (two-sided) error ϵ. Also standard to write $Q(f) = Q_{1/3}(f)$.