Computational Complexity — tutorial 8

Boolean circuits 3, probabilistic algorithms 1

1. Prove that MAJORITY is not in AC^0. *Hint: solve* PARITY *using* MAJORITY *as a black-box.*

We consider the relations between the uniform circuit complexity classes and log-space classes. This will allow us to show that

$$\mathsf{u\text{-}AC}^0 \subsetneq \mathsf{u\text{-}NC}^1 \subseteq \mathsf{LogSpace} \subseteq \mathsf{NLogSpace} \subseteq \mathsf{u\text{-}AC}^1 \subseteq \mathsf{u\text{-}NC}^2 \subseteq \mathsf{u\text{-}AC}^2 \subseteq \cdots \subseteq \mathsf{u\text{-}NC} = \mathsf{u\text{-}AC} \subseteq \mathsf{P}.$$

2. Prove that LogSpace is closed under compositions.

3. Prove that the $\mathsf{u\text{-}NC}^1$ circuit evaluation problem is in LogSpace: given the description of the circuit of logarithmic depth and fan-in $\leqslant 2$ and the input to this circuit, decide if the circuit returns true on this input.

4. Conclude that $\mathsf{u\text{-}NC}^1 \subseteq \mathsf{LogSpace}$.

5. Prove that DIRECTED REACHABILITY is in $\mathsf{u\text{-}AC}^1$; formally, for a given $n \in \mathbb{N}$, given a sequence of $n^2 + 2n$ bits denoting the $n \times n$ adjacency matrix of a graph and the source and the destination encoded in unary, check if we can reach the destination from the source in the graph.

6. Using the exercise above, prove that $\mathsf{NLogSpace} \subseteq \mathsf{u\text{-}AC}^1$.

7. For $n \in \mathbb{N}$, let p_n be a probability that a random function $f : \{0,1\}^n \to \{0,1\}$ can be recognized by a circuit with fan-in 2 of size at most $\frac{2^n}{1000n}$ (and arbitrary depth). Prove that $\lim_{n\to\infty} p_n = 0$.

It can be also proved that every function $f : \{0,1\}^n \to \{0,1\}$ can be recognized by a circuit with fan-in 2 of size at most $\frac{1000 \cdot 2^n}{n}$, but the details are quite messy. We can solve this problem during the consultations if you want.

Probabilistic complexity classes compared with P:

Class	$\mathbb{P}$[algorithm accepts x] if		Running time
	$x \in L$	$x \notin L$	
P	1	0	polynomial
RP	$\geqslant \frac{1}{2}$	0	polynomial
co-RP	1	$\leqslant \frac{1}{2}$	polynomial
BPP	$\geqslant \frac{3}{4}$	$\leqslant \frac{1}{4}$	polynomial
PP	$\geqslant \frac{1}{2}$	$< \frac{1}{2}$	polynomial
ZPP	1	0	**expected** polynomial

Open problems: $\mathsf{RP} \stackrel{?}{=} \mathsf{P}$. The inclusion $\mathsf{P} \subseteq \mathsf{RP}$ is trivial, the other one is hard. In other words: can randomized algorithms be efficiently derandomized?

Schwartz-Zippel lemma *(simplified version)* Fix a prime p. Given a non-zero polynomial Q (over the integers mod p) with variables $x_1, x_2, \ldots, x_n$ and total degree $d \geqslant 0$, the probability that $Q(x_1, x_2, \ldots, x_n) = 0 \mod p$ for a random tuple of variables $x_1, \ldots, x_n$ is bounded from above by $\frac{d}{p}$.

8. For an undirected graph G with n vertices, we define the *Tutte matrix* as an $n \times n$ matrix defined as follows:

$$A_{i,j} = \begin{cases} x_{i,j} & \text{if } (i,j) \text{ is an edge of } G \text{ and } i < j, \\ -x_{j,i} & \text{if } (i,j) \text{ is an edge of } G \text{ and } i > j, \\ 0 & \text{otherwise.} \end{cases}$$

Here, each $x_{i,j}$ is a separate variable. For instance, the graph which is a cycle $1 - 2 - 3 - 4$ on 4 vertices has the following Tutte matrix:

$$\begin{pmatrix} 0 & x_{1,2} & 0 & x_{1,4} \\ -x_{1,2} & 0 & x_{2,3} & 0 \\ 0 & -x_{2,3} & 0 & x_{3,4} \\ -x_{1,4} & 0 & -x_{3,4} & 0 \end{pmatrix}$$

It can be proved that the determinant of this matrix is non-zero if and only if G has a perfect matching on n vertices.

Prove that verifying whether a given graph has a perfect matching is in RP.

Note: we can also prove that this problem is in P *by utilizing Edmonds' blossom algorithm, but we don't talk about this here.*

9. Consider the PERMUTATION PATH problem: you're given a directed/undirected graph G with n vertices where each vertex is colored with one of k colors (called $1, 2, \ldots, k$). Does there exist a simple path with k vertices such that the color of the first vertex is 1, the color of the second vertex is 2, ..., the color of the k-th vertex is k?

Prove that PERMUTATION PATH can be solved in polynomial time (without randomization).

10. Consider the COLORFUL PATH problem: you're given a directed/undirected graph G with n vertices where each vertex is colored with one of k colors (called $1, 2, \ldots, k$). Does there exist a simple path with k vertices hitting each color exactly once?

Prove that PERMUTATION PATH can be solved in time $2^k \cdot \text{poly}(|G|)$ (without randomization).

11. Consider the k-PATH problem: you're given a directed/undirected graph G with n vertices. Does there exist a simple path with k distinct vertices?

Prove that k-PATH can be solved in time $(2e)^k \cdot \text{poly}(|G|)$ with randomization. Conclude that k-PATH for $k = O(\log n)$ is in RP.

12. Prove the *amplification lemma* for RP: if we replace „$\geqslant \frac{1}{2}$", in the definition of RP with „$\geqslant \varepsilon$" for any constant $0 < \varepsilon < 1$, we'll get exactly the same definition of RP.

In other words: if we run the probabilistic algorithm over and over again, we'll be more and more confident about its answer.

13. Prove the *amplification lemma* for BPP: if we replace „$\geqslant \frac{3}{4}$", „$\leqslant \frac{1}{4}$" in the definition of BPP with „$\geqslant 1 - \varepsilon$", „$\leqslant \varepsilon$" for any constant $0 < \varepsilon < \frac{1}{2}$, we'll get exactly the same definition of BPP.

14. As above, but we assume that $\varepsilon = \frac{1}{2} - \frac{1}{n}$ where n is the length of the input.

15. Prove that $\mathsf{RP} \cap \text{co-}\mathsf{RP} = \mathsf{ZPP}$.

16. Prove that $\mathsf{NP} \subseteq \mathsf{PP}$.

17. Prove that BPP is closed under Kleene star.