CSCI 6114 Fall 2026: Problem Set 1
Circuits

Joshua A. Grochow

August 25, 2026

Readings

Read these:

- Viola Chapter 2 ($\sim$ 8 pages)
- Arora & Barak §6.1 (in the freely available draft version is fine, $\sim$ 6 pages)
- Homer & Selman §8.1, up to but excluding Theorem 8.3 ($\sim$ 9 pages).

And if you want additional resources on these topics, check out these:

- Sipser §9.3
- Du & Ko §6.2
- Hemaspaandra & Ogihara *Complexity Theory Companion* p. 276
- Wigderson §5.2.1.
- Moore & Mertens §6.5

Key definitions and terms

A *circuit family* is a sequence $C = (C_1, C_2, C_3, \dots)$ of Boolean circuits C_i where C_i takes i inputs. The language decided by a circuit family C is $L(C) = \{x : C_{|x|}(x) = 1\}$. Following Viola, we use CktP to denote the class of languages that can be decided by a circuit family of polynomial size, that is, where $|C_n| \leq \text{poly}(n)$ (most books and authors still call this class $\mathsf{P/poly}$; we'll see the origin of that name below, but I think Viola's name CktP is much more mnemonic).

Exercises

1. Definition: A circuit family C is P-uniform if there is a polynomial-time algorithm that, on input 1^n, outputs a description of the circuit C_n.

   Show that P-uniform CktP is equal to P.

2. Given a class $\mathcal{C}$ of languages and a function $f\colon \mathbb{N} \to \mathbb{N}$, we define "$\mathcal{C}$ with f-bounded advice", denoted $\mathcal{C}/f$, as the class of languages L such that there exists $L' \in \mathcal{C}$ and there exist strings $a_1, a_2, a_3, \ldots$ ("a" for "advice") with $|a_n| \leq f(n)$ such that for all x,

   $$x \in L \iff (x, a_{|x|}) \in L'.$$

   In other words, for each n, there is a single advice string a_n that helps L' decide membership in L for *all* strings x of length n.

   We define
   $$\mathsf{P/poly} := \bigcup_k \mathsf{P}/O(n^k).$$

   Prove that $\mathsf{P/poly} = \mathsf{CktP}$.

3. A language L is *(polynomially) sparse* if there is a polynomial p such that the number of strings in L of length $\leq n$ is at most $p(n)$.

   (a) Show that all sparse languages are in $\mathsf{P/poly}$.

   (b) (Before you get to this exercise, we will talk about these terms and definitions in class) Show that $\mathsf{P/poly} = \mathsf{P}^{\mathsf{SPARSE}}$, that is, $\mathsf{P/poly}$ is the class of languages L such that there is some sparse language S and L reduces to S by a polynomial-time oracle Turing machine (denoted $L \leq_T^p S$).

4. (a) Show that $\mathsf{P} \neq \mathsf{P}/O(\log n)$, by showing that the latter has uncomputable languages.

   (b) Show that advice length and circuit size are not necessarily the same. In particular, show that there are languages in $\mathsf{P}/O(\log n)$ that require linear-size circuits.

5. (a) Show that search reduces to decision for SAT: there is a function in $\mathsf{FP}^{\mathsf{NP}}$ that, given a Boolean formula φ, either outputs a satisfying assignment to φ (if one exists), or correctly reports that no satisfying assignments exist.

(b) Despite Question 4, show that $\mathsf{NP} \subseteq \mathsf{P}$ iff $\mathsf{NP} \subseteq \mathsf{P}/O(\log n)$.

(c) What can you say if $\mathsf{NP} \subseteq \mathsf{P}/\mathsf{poly}$?

6. It is natural to wonder whether uncomputable languages are the only thing standing in the way of P being equal to P/poly. Here, show that's not the case, i.e., that $\mathsf{P}/\mathsf{poly} \cap \mathsf{COMP} \neq \mathsf{P}$, i.e., that there are computable languages in P/poly that aren't in P. *Hint:* Pick a hard but computable language, far outside of P, and encode it in unary. You may assume the Time Hierarchy Theorem: if $T(n) \log T(n) < o(T'(n))$, then $\mathsf{DTIME}(T(n)) \subsetneq \mathsf{DTIME}(T'(n))$. How large must T' be to get this to work against P?