CS 599: The Meta-Complexity Frontier, Fall 2023
Problem Set #2

Due: 5:00PM, Friday, November 3, 2023.

Homework Policies

- Submit your completed assignment by email to `marco[at]ntime[dot]org` . Please include the string "CS599PS2" somewhere in your subject line.
- Solutions must be typeset, e.g., using LaTeX or Microsoft Word.
- You are encouraged to collaborate on the homework problems with each other in small groups (2 - 3 people). Collaboration may include brainstorming or exploring possible solutions together on a whiteboard, but should not include one person telling the others how to solve a problem. You **must write up the solutions independently** (in your own words) and **acknowledge your collaborators** at the beginning of the first page.
- You may read papers and other outside sources to help you solve these problems. If you do so, you **must cite and acknowledge any sources and write the solutions in your own words.**
- You may freely use without proof any results proved in class, in lecture notes posted on the class webpage, or in the main body of the texts assigned as reading. Note that this excludes results that appear in the texts as problems and exercises. You may, of course, use such results but you have to prove them first.
- To help your instructor calibrate the length and difficulty of future assignments, please include with each problem an estimate of how long it took you to solve it.
- Please start early! The problems are presented roughly in the order of the course content they correspond to, so you may get started on the first few problems as soon as the assignment is released. Late assignments will receive credit only with prior permission of the instructor.

Part of this and subsequent assignments will be to familiarize yourself with definitions of complexity classes and concepts that did not come up in class. These items will always be defined in our "local" complexity zoo, linked from the `Resources` section of our course webpage.

1 Formal Complexity Measures: Limits & Limit Cases

1. Suppose that f can be represented by an s-CNF and by a t-DNF. Show that Khrapchenko's measure cannot yield a lower bound larger than st for any such f.
2. A *submodular* formal complexity measure μ satisfies the additional axiom that, for all functions f, g,

$$\mu(f \vee g) + \mu(f \wedge g) \leq \mu(f) + \mu(g).$$

Submodular measures can prove *superpolynomial* lower bounds against *monotone* formulas — DeMorgan formulas without NOT gates. Solve Problem 23.4 from [AB09] to show that these techniques cannot generalize to standard DeMorgan formulas. That is, prove that for every n-bit function f_n, every submodular formal complexity measure satisfies $\mu(f_n) = O(n)$.

3. Recall Khrapchenko's measure. We denote by $H(A,B)$ the set of neighbors $(a,b) \in A \times B$ where $a, b \in \{0,1\}^n$ differ in only one bit position

$$K_{AB} = \frac{|H(A,B)|^2}{|A| \cdot |B|} \tag{1}$$

$$\mu_K(f) = \max_{A,B}\{K_{AB} : A \subseteq f^{-1}(1),\ B \subseteq f^{-1}(0)\} \tag{2}$$

Suppose we simplify Khrapchenko's measure to remove maximization over A, B in (2) and instead define:

$$\mu'_K(f) = K_{AB} \text{ where } A = f^{-1}(1) \text{ and } B = f^{-1}(0)$$

Is μ'_K still a formal complexity measure? If so, prove it. If not, identify precisely how the proof that μ_K is a formal complexity measure fails for μ'_K. Can this simpler measure still be used to prove leafsize lower bounds for the parity function?

2 Natural Proofs: Warmup

Recall that a language defined on truth-tables — that is, $\mathcal{P} = \{P_n\}_{n\in\mathbb{N}}$ where $\mathcal{P}_n \subseteq \{0,1\}^{2^n}$ — is called a *Natural Property* against complexity class Λ with density δ_n if it is

Useful: For any sequence of functions f_n if $f_n \in \Lambda$ then $f_n \notin \mathcal{P}$ for almost every n.

Constructive: The decision problem $f_n \stackrel{?}{\in} \mathcal{P}_n$ is in $\mathsf{TIME}[2^{O(n)}]$ — polynomial in the size of the input.

***and* Large:** A δ_n-fraction of n-bit functions have the property, so

$$\Pr_{f\sim\mathcal{F}_n}[f \in \mathcal{P}_n] \geq \delta_n$$

The original definition of Natural Properties sets $\delta_n \geq 2^{-O(n)}$ [RR97].

1. A function $f \in \mathcal{F}_n$ is called *non-degenerate* if it depends on all n input variables. Formally,

$$\forall i \in [n]\ \exists \alpha \in \{0,1\}^n\ f(a_1, a_2, \ldots, a_{i-1}, 1, a_{i+1}, \ldots, a_n) \neq f(a_1, a_2, \ldots, a_{i-1}, 0, a_{i+1}, \ldots, a_n).$$

Show that non-degeneracy is a natural property useful against circuits of size $O(n^\epsilon)$ for any $\epsilon < 1$.

2. Prove "largeness amplification" for natural properties. That is, suppose $\mathcal{P}$ is a natural property useful against the complexity class $\mathsf{SIZE}[s(n)]$ with largeness $\delta_n \geq 2^{-cn}$ for some constant $c \geq 0$. Show that there is another natural property $\mathcal{P}'$ useful against $\mathsf{SIZE}[s(n/(c+1))]$ with largeness $\delta'_n \geq 1/5$.

3. What properties of general circuit-size did you use in the largeness amplification argument above? Would the same trick work to improve the density of a natural property against formula leafsize? Why or why not?

3 Natural Proofs: Extract a Natural Property

First, read Section 3.4 of [RR97], which explains that shrinkage-method lower bounds on formula size are natural. The section concludes by asserting that it is "easy" to prove largeness and constructivity for C_{2n}. Give an explicit and self-contained proof that C_{2n} is a natural property, including a formal definition of C_{2n}.

4 Compression: Algorithms $\implies$ Lower Bounds

We have spent a great deal of time in this class extracting algorithms from complexity lower bounds. What about the opposite direction — do efficient algorithms for certain problems imply complexity lower bounds? Indeed they do, and such connections have been key to breakthrough results in complexity theory! For this problem you will show that efficient algorithms for "non-trivial data compression" imply circuit lower bounds.

First we define witness complexity. For a language $\mathcal{L} \in \mathsf{NEXP}$ and an n-bit string $x \in \mathcal{L}$, view each $2^{\mathsf{poly}(n)}$ certificate that $x \in \mathcal{L}$ (from the verifier definition of NEXP) as the truth table of a Boolean function on $\mathsf{poly}(n)$-bit inputs. Now, the *witness complexity* of x with respect to the NEXP verifier $V_{\mathcal{L}}$ deciding $\mathcal{L}$ is the size of the smallest circuit that computes the truth table of a witness for $x \in \mathcal{L}$. This picks out the y's of minimal circuit complexity such that $V_{\mathcal{L}}(x, y)$ accepts.

The *Easy Witness Lemma* states that, if $\mathsf{NEXP} \subseteq \mathsf{P/poly}$, then for every language $\mathcal{L} \in \mathsf{NEXP}$ there exists a verifier $V_{\mathcal{L}}$ and constant $c_{\mathcal{L}}$ such that every sufficiently long n-bit input $x \in \mathcal{L}$ has witness complexity at most $n^{c_{\mathcal{L}}}$ with respect to $V_{\mathcal{L}}$. Your solution to this problem should use the easy witness lemma.

Finally, the $s(n)$-*compression* problem is: given the truth table of a Boolean function f computed by a $s(n)$-size circuit, print in polynomial time a circuit of size strictly less than $2^n/n$ that computes f. Observe how this task is *barely non-trivial*, especially if $s(n)$ is a feasible bound — Lupanov's bound gives a circuit of size about $2^n/n$ for every Boolean function. Even so, prove that even a "weak" solution would imply breakthrough circuit lower bounds:

Theorem 1. *Suppose for every c there is a deterministic polynomial-time algorithm that compresses the truth table of input n-bit Boolean functions $f \in \mathsf{SIZE}[n^c]$ to a circuit of size less than $2^n/n$. Then $\mathsf{NEXP} \not\subset \mathsf{P/poly}$.*

References

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009. ISBN: 978-0-521-42426-4. URL: http://www.cambridge.org/catalogue/catalogue.asp?isbn=9780521424264.

[RR97] Alexander A. Razborov and Steven Rudich. "Natural Proofs". In: *J. Comput. Syst. Sci.* 55.1 (1997), pp. 24–35. DOI: 10.1006/jcss.1997.1494. URL: https://doi.org/10.1006/jcss.1997.1494.