KTH Computer Science and Communication

DD2446 Complexity Theory: Problem Set 2

Due: October 4, 2013, at 23:59. Submit your solutions as a PDF file by e-mail to `jakobn at kth dot se` with the subject line `Problem set 2: ⟨your name⟩`. Name the PDF file `PS2_⟨YourName⟩.pdf` (with your name coded in ASCII without national characters), and also state your name and e-mail address at the top of the first page. Solutions should be written in LaTeX or some other math-aware typesetting system. Please try to be precise and to the point in your solutions and refrain from vague statements. *Write so that a fellow student of yours can read, understand, and verify your solutions.* In addition to what is stated below, the general rules stated on the course webpage always apply.

Collaboration: Discussions of ideas in groups of two people are allowed—and indeed, encouraged—but you should write down your own solution individually and understand all aspects of it fully. You should also acknowledge any collaboration. State at the beginning of the problem set if you have been collaborating with someone and if so with whom.

Reference material: Some of the problems are "classic" and hence it might be easy to find solutions on the Internet, in textbooks or in research papers. It is not allowed to use such material in any way unless explicitly stated otherwise. Anything said during the lectures on in the lecture notes, or which can be found in chapters of Arora-Barak covered in the course, should be fair game, though, unless you are specifically asked to show something that we claimed without proof in class. It is hard to pin down 100% formal rules on what all this means—when in doubt, ask the lecturer.

About the problems: Some of the problems are meant to be quite challenging and you are not necessarily expected to solve all of them. A total score of 60 points is the threshold for grade E, 90 points for grade D, 120 points for grade C, 150 points for grade B, and 180 points for grade A on this problem set. Any corrections or clarifications will be given at `piazza.com/kth.se/fall2013/dd2446/` and any revised versions will be posted on the course webpage `www.csc.kth.se/utbildning/kth/kurser/DD2446/kplx13/`.

1 (10 p) We say that a language $L \subseteq \{0,1\}^*$ is *sparse* if there is a polynomial p such that it holds for every $n \in \mathbb{N}^+$ that $\left|L \cap \{0,1\}^n\right| \leq p(n)$. Show that if L is sparse, then $L \in \mathsf{P/poly}$.

2 (20 p) Let us say that a function $f : \{0,1\}^* \to \{0,1\}^*$ is *write-once logspace computable* if f can be computed by a Turing machine M that uses $\mathrm{O}(\log n)$ space on its work tapes and whose output tape is *write-once*. By a write-once tape we mean a tape where at every time step M can either keep its head at the same position on the tape or write a symbol to it and move one location to the right, but M can never read from the tape or move left. The used cells on the write-once tape are not counted towards the space bound on M.

Prove that f is write-once logspace computable if and only if it is *implicitly logspace computable* as defined in class.

3 (30 p) Show that the language $\Sigma_i\textsc{Sat}$ is Σ_i^p-complete (under the polynomial-time reductions studied in class).

Hint: Use the NP-completeness of $\textsc{CnfSat}$.

4 (30 p) We proved in class that the language $\textsc{Path} = \{\langle G, s, t\rangle | \exists \text{ path from } s \text{ to } t \text{ in } G\}$ is NL-complete. We also proved that NL = coNL, and, in particular, that it holds for the complement language $\overline{\textsc{Path}} = \{\langle G', s', t'\rangle | \neg\exists \text{ path from } s' \text{ to } t' \text{ in } G'\}$ that $\overline{\textsc{Path}} \in \mathsf{NL}$.

But this means that there must exist an implicitly logspace computable function that takes a directed graph G' and two vertices $s', t' \in V(G')$ and outputs a directed graph G and two vertices $s, t \in V(G)$ such that there is *some path* from s to t in G if and only if there is *no path* from s' to t' in G'. Describe such a function and how to compute it.

You do not need to decribe every nut and bolt in the construction of G from G', but your description should contain enough details so that you could code it up in principle in your favourite high-level programming language (using well-defined subroutines that we also know can be coded up in principle).

5 (30 p) When we proved in class that $\textsc{Parity} \notin \mathsf{AC}^0$, we started with a bounded-depth polynomial-size circuit C that supposedly computed the parity of its input bits, and then preprocessed it to get an equivalent circuit C' with the following properties:

1. All gates in C' have fan-out 1 (i.e., it is what is known as a *formula*, with a DAG structure that is a tree).
2. All NOT ($\neg$) gates are at the input level of C' (i.e., they only apply to variables).
3. The AND ($\wedge$) and OR ($\vee$) gates alternate, so that at each level of C' all gates are either AND or OR.
4. The bottom level has AND gates of some small, bounded fan-in (we picked fan-in 1 in class but noted that any small enough fan-in was fine).

Show how this preprocessing can be done without increasing the circuit depth by more than a constant and the size more than polynomially (so that C' is also a bounded-depth polynomial-size circuit computing the parity of its input bits). If C is a circuit of size S and depth d, what size and depth do you get for C'?

6 (70 p) The purpose of this problem is to investigate some of the conditions in Håstad's switching lemma, in particular, the requirement of bounded fan-in (i.e., that the restrictions operate on k-CNF and k-DNF formulas).

6a (25 p) Let $f : \{0,1\}^n \to \{0,1\}$ be some Boolean function. Prove that if all (minimal) maxterms of f have size at most s, then f can be represented as an s-CNF formula.

Does the other direction hold as well? That is, is it true that if f can be represented as an s-CNF formula then all (minimal) maxterms of f have size at most s?

6b (25 p) Prove that any CNF formula that computes parity of n bits must have size exponential in n. For full credit, prove an exact, tight bound. (And for concreteness, define the *size* of a CNF formula as the number of literals in it, counted with repetitions).

6c (20 p) Argue that in view of Problem 6b, we actually do not need the added requirement of bounded fan-in in the final step of the proof of PARITY $\notin \mathsf{AC}^0$, i.e., after $(d-2)$ rounds of restrictions have been applied on C' so that the circuit has collapsed to a CNF formula. In our proof in class, we crucially used in this step that the formula obtained was a k'-CNF formula for some constant k'. (Let us note in passing that there is a lower bound on DNF formulas analogous to that in Problem 6b in case the circuit collapses to a DNF formula, but there is no need to prove this or even consider the DNF case to get a full score.)

This raises the question whether we could in fact drop the restriction on fan-in in the bottom layer completely at all $(d-2)$ stages of the proof if we just did a little bit of extra work. Explain how to modify the proof of PARITY $\notin \mathsf{AC}^0$ to work also if there is no bound on the bottom-level fan-in of C' (if this can be done), or point out where in the proof we run into trouble (if it cannot be done).

7 (40 p) Let R_t denote the set of all restrictions of subsets of exactly t out of n variables, where n is supposed to be large and $t \geq n/2$. When proving Håstad's switching lemma in class, we argued that the set $B \subseteq R_t$ of *bad* restrictions for which the conclusion of the lemma does not hold is very small compared to all of R_t, and hence it is very unlikely that a randomly chosen restriction will be bad (which is exactly what the lemma claims).

More formally, we constructed (although by this time we were going pretty fast) a one-to-one mapping from B to $R_{t+s} \times \{0,1\}^\ell$ for some $\ell = \mathrm{O}(s \log k)$ (where s and k are the parameters in the switching lemma), and claimed this showed that the probability to get a bad restriction is

$$\frac{|B|}{|R_t|} \leq \frac{\left|R_{t+s} \times \{0,1\}^\ell\right|}{|R_t|} = n^{-\Omega(s)} \ .$$

The purpose of this problem is to fill in the details in these calculations and show that one gets a failure probability for the restriction as small as the one claimed in Håstad's switching lemma *exactly as stated in the textbook Arora-Barak.*

That is, just trusting that the one-to-one map $m : B \to R_{t+s} \times \{0,1\}^\ell$ constructed in class was correct, show that the qoutient $\left|R_{t+s} \times \{0,1\}^\ell\right| / |R_t|$ is small enough to give the probability bound in the switching lemma as stated in the textbook.
Hint: Show that for $t > n/2$ it holds that

$$\binom{n}{t+s} \leq \binom{n}{t} \left(\frac{e(n-t)}{n}\right)^s$$

by first proving

$$\binom{n}{t+s} = \binom{n}{t}\binom{n-t}{s} \Big/ \binom{t+s}{t}$$

(and try to find a nice combinatorial proof for this latter equality). You can use the well-known inequalities

$$\left(\frac{n}{k}\right)^k \leq \binom{n}{k} \leq \left(\frac{en}{k}\right)^k$$

without proof.

8 (50 p) Show that $\mathsf{P} \neq \mathsf{SPACE}(n^k)$ for any fixed $k \in \mathbb{N}^+$.
Hint: Use padding. Also, just to avoid confusion, note that $\mathsf{P} \subseteq \bigcup_{k \in \mathbb{N}^+} \mathsf{SPACE}(n^k) = \mathsf{PSPACE}$, but the point here is that we are fixing k.