DD2446 Complexity Theory: Problem Set 3

Due: Monday October 21, 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 3: ⟨your name⟩`. Name the PDF file `PS3_⟨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 or 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 around 50 points should be enough for grade E, 80 points for grade D, 110 points for grade C, 140 points for grade B, and 170 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) Show that if one-way functions exist, then $\mathsf{P} \neq \mathsf{NP}$.

2 (20 p) Show that $\mathsf{ZPP} = \mathsf{RP} \cap \mathsf{coRP}$.

3 (20 p) In our lecture on property testing, we studied the 2^n-dimensional vector space of functions $f : \{\pm 1\}^n \to \mathbb{R}$ with inner product $\langle f, g \rangle = \frac{1}{2^n} \sum_{x \in \{\pm 1\}^n} f(x)g(x)$. In class, we claimed without too much of a proof that the set of functions $\{\chi_\alpha\}_{\alpha \subseteq [n]}$ defined by $\chi_\alpha(x) = \prod_{i \in \alpha} x_i$ form an orthonormal basis for this vector space, namely the *Fourier basis* that we then used to analyze the linearity test.

Fill in the details to establish this claim! That is, show that

$$\langle \chi_\alpha, \chi_\beta \rangle = \begin{cases} 1 & \text{if } \alpha = \beta, \\ 0 & \text{otherwise.} \end{cases}$$

Hint: Consider the symmetric difference $\gamma = \alpha \triangle \beta = (\alpha \cup \beta) \setminus (\alpha \cap \beta)$ and prove that it holds that $\sum_{x \in \{\pm 1\}^n} \chi_\gamma(x) = 0$ if $\gamma \neq \emptyset$.

4 (20 p) In our lecture on proof complexity, we defined a *resolution refutation* $\pi : F \vdash \bot$ of an unsatisfiable CNF formula F to be a sequence of clauses $\pi = (C_1, C_2, \ldots, C_L)$ such that each C_i is either a clause in F (an *axiom*) or is derived from two clauses $C_j, C_k \in \pi$, $j < k < i$, by the *resolution rule*

$$\frac{C \lor x \qquad D \lor \overline{x}}{C \lor D} ,$$

and such that the final clause C_L is the empty clause containing no literals, denoted $\bot$. The *length* of the refutation π is L.

When proving results about resolution, it is often convenient to also allow a clause $C_i \in \pi$ to be derived from some C_j, $j < i$, by the *relaxation rule*

$$\frac{C}{C \lor D} ,$$

where one deduces the strictly weaker clause $C \lor D$ from C. Show that adding this rule does not really change the proof system. Formally, prove that if $\pi : F \vdash \bot$ is a resolution refutation of an unsatisfiable CNF formula F using also the relaxation rule, then there is a standard resolution refutation $\pi' : F \vdash \bot$ in at most the same length without any applications of relaxation.
Hint: Use induction over the sequence of clauses $\pi = (C_1, C_2, \ldots, C_L)$ in the relaxed resolution refutation.

5 (30 p) For a language $L \subseteq \{0,1\}^*$, let $L_k = \{x \in L; |x| \leq k\}$ denote all strings in L of length at most k. We say that L is *downward self-reducible* if there is a polynomial-time algorithm A that given x and oracle access to $L_{|x|-1}$ decides correctly whether $x \in L$ or not. Prove that if L is downward self-reducible, then $L \in \mathsf{PSPACE}$.

6 (40 p) In our lecture on proof complexity, we defined the CNF encoding of the (negation of the) pigeonhole principle PHP^m_n for any number of pigeons m and pigeonholes n, but then focused on $m = n+1$ when proving the $\exp(\Omega(n))$ lower bound on resolution refutation length for PHP^{n+1}_n.

What would happen with this lower bound proof if we considered more than $n + 1$ pigeons, say $m = n + 2$, $m = 2n$, $m = n^2$, or even $m = 2^n$ pigeons? Would the proof still work, and would we still get a lower bound on the form $\exp(\Omega(n))$? Describe how to adapt the proof to work for larger m; determine for how large m you can make it work; and/or explain when or why the approach we used in class fails.
Hint: In order to solve this problem, it is not necessary to give a full answer to the question of how the hardness of the formula PHP^m_n depends on m—it is fully sufficient to analyze the concrete lower bound approach that we employed in class and try to understand how far this technique can (or cannot) be pushed. You do not need to prove all claims you make beyond reasonable doubt—in particular, it is not necessary to prove any claims that we left unproven in class—but it should be possible to see how to plausibly fill in any gaps in your arguments.

7 (50 p) Let multiprover interactive protocols be defined as the interactive protocols in Section 8.1 in Arora-Barak, except that there are several provers and that the verifier's message in each round depends on previous messages from all provers (and on the verifier's private randomness). The messages sent by each prover only depends on the communication with the verifier, however, just as before. Let $\mathsf{MIP}[N]$ denote the set of languages that can be decided by N-multiprover interactive protocols in a polynomial number of rounds (in analogy with $\mathsf{IP} = \mathsf{MIP}[1]$ in Definition 8.6 in Arora-Barak).

Prove that, as claimed in class, only two provers are needed to realize the full power of multiprover interactive protocols. That is, prove that $\mathsf{MIP}[2] = \mathsf{MIP}[\mathrm{poly}]$, where $\mathsf{MIP}[\mathrm{poly}]$-protocols have a number of provers scaling polynomially with the size of the input.

8 (40 p) When proving a lower bound on resolution refutation length, we studied a prosecutor-defendant game and proved a lower bound on the size of a prosecutor strategy for PHP_n^{n+1} in this game. It is not hard to see that the same game can be played on any unsatisfiable CNF formula F (which the defendant claims to be satisfiable), where the prosecutor asks about assignments to variables $x \in \mathit{Vars}(F)$, or forgets such assignments, and the "explicit contradictions" the prosecutor is trying to force are partial assignments falsifying some axiom clause $C \in F$. The same reasoning we used in class shows that any resolution refutation of F in length L yields a strategy for the prosecutor of size $\mathrm{O}(L)$ (i.e., with $\mathrm{O}(L)$ rules in the instruction book).

In this problem we are interested in the other direction. Suppose that the prosecutor has a strategy for some formula F that requires consideration only of L cases in order to secure the conviction of the defendant. Can such a strategy be converted to a refutation of F in length $\mathrm{O}(L)$? Describe how to convert a prosecutor strategy to a resolution refutation in essentially the same size, or explain why it seems hard to do the transformation in this other direction.

9 (60 p) The goal of this exercise is to give a complete proof that $\mathsf{PSPACE} \subseteq \mathsf{IP}$, strengthening the result $\mathsf{coNP} \subseteq \mathsf{IP}$ that was proven in class.

Given a quantified Boolean formula (QBF) $\psi = \forall x_1 \exists x_2 \forall x_3 \cdots \exists x_n \, \phi(x_1, \ldots, x_n)$, we can use arithmetization as in our proof of $\mathsf{coNP} \subseteq \mathsf{IP}$ to construct a polynomial P_ϕ such that ψ is true if and only if $\prod_{b_1 \in \{0,1\}} \sum_{b_2 \in \{0,1\}} \prod_{b_3 \in \{0,1\}} \cdots \sum_{b_n \in \{0,1\}} P_\phi(b_1, \ldots, b_n) \neq 0$. However, the SumCheck protocol we used to decide the $\#\mathrm{SAT}_D$ problem for CNF formulas no longer works, since each multiplication corresponding to a $\forall$-quantifier can double the degree of the polynomial.

9a (20 p) Suppose that ψ is a QBF formula (not necessarily in *prenex normal form* as described in Definition 4.10 and discussed further below in Arora-Barak) satisfying the following property: if $x_1, \ldots, x_n$ are the variables of ψ sorted in order of first appearance, then for every variable x_i there is at most a single universal quantifier involving x_j for any $j > i$ appearing before the last occurrence of x_i in ψ. Show that in this case, when we run the SumCheck protocol with the modification that we check $s(0) \cdot s(1) = K$ for product operations (i.e., $\forall$-quantifiers), the prover only needs to send polynomials of degree $\mathrm{O}(n)$ since the degree blow-up is at most an additive term 2.

9b (20 p) Assuming that any QBF formula ψ can be rewritten to satisfy the property in Problem 9a, use this to show that $\mathrm{TQBF} \in \mathsf{IP}$ and hence $\mathsf{PSPACE} \subseteq \mathsf{IP}$.

9c (20 p) Show that any QBF formula ψ of size m can be transformed into a logically equivalent formula ψ' of size $\mathrm{O}(m^2)$ that satisfies the property in Problem 9a.

Hint: Introduce a new variable y_i for any occurrence of x_i we need to get rid of and encode that x_i and y_i take the same truth value.