# *Problem set 3 — Space in proof complexity*

*Massimo Lauria —* lauria.massimo@gmail.com
*Office 1107, Ookayama West 8th Building*

*(This document was updated on June 21, 2017)*

**Due:** Saturday, December 5th, 2015 at 23:59. Submit your solutions as a PDF file by e-mail to lauria.massimo@gmail.com with the subject line

`Problem set 3: ⟨your full name⟩`

Name the PDF file `PS3⟨YourFullName⟩.pdf` (with your name coded in ASCII without national characters and no spaces), and also state your name (in both national and latin character) 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. If you are not confident with English please limit yourself to simple, short and clear sentences. Nevertheless the solutions needs to be explained in reasonable precision. *Write so that a fellow student of yours can read, understand, and verify your solutions.*

**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. **Please don't do that.**

☺ You can use and refer to anything said during the lectures or written in the lecture notes.

☹ You cannot use textbooks/internet/papers to find the answer to the problems in the set.

☺ You can refer to research papers/textbooks/internet for those proofs that we saw in class, either because they are missing from the lecture notes or because you feel the lecture notes are not clear enough.

☹ The previous permission does not apply to missing pieces of those proofs that the lecturer explicitly asked you to prove as an exercise.

It is hard to pin down 100% formal rules on what all this means—when in doubt, ask the lecturer. Obviously these rules are designed with honest students in mind. I am confident that there is no need to develop rules against malicious students.

**Assessment of the final grade:** Some of the problems are meant to be quite challenging and you are not necessarily expected to solve all of them. As a general guideline, **a total score of 80 should be sufficient to pass**. Partial score may be given for partial solutions and for partially (but mostly) correct solutions. Please refrain from providing answers if you are not confident of their correctness. The tentative plan it to have three problem sets, published:

- at the 3rd lecture;
- at the 6th lecture;
- at the 9th lecture.

Passing three problem sets is sufficient to pass the course. If a student fails to pass one problem set by a small amount of points, he/she could still pass the course if he/she has a good score (well above pass) at the other two problem sets. How good depends on how far for the threshold the student was in the failed problem set.

The total points and the passing thresholds of the three problem sets may be different. Beware that the each passing threshold may be lowered (but never increased!) during the grading.

## *CDCL Solvers*

**Problem 1** (15 points)**.** *Prove that the clause learned in the conflict analysis of a CDCL solver, as described in Lecture 7, can be derived efficient in resolution from the clauses in the clause database. Do not assume any specific learning scheme.*

## *Resolution space*

Recall the definition of embedding and the tree complexity measure $\mathcal{C}$ from lecture 8.

**Problem 2** (10 points)**.** *Show that $\mathcal{C}(T) \geq h$ if and only if the complete binary tree of height $h$ can be embedded into $T$.*

**Problem 3** (15 points)**.** *Consider a tree-like resolution refutation such that its underlying structure is represented by tree $T$. Show that when $\mathcal{C}(T) \leq h$ if and only if the proof can be represented (in the blackboard memory model) in space $h + 2$.*

**Problem 4** (15 points)**.** *Show that for every unsatisfiable CNF with $n$ variables there is a resolution refutation of length $O(2^n)$ and space $n + 2$.*

**Problem 5** (20 points)**.** *Show that if a $k$-CNF formula has a tree like refutation of length $S$, then it is possible to find a resolution refutation in time $n^{O(\log S+k)}$.*

**Problem 6** (40 points)**.** *Show that if a $k$-CNF formula has a tree like refutation of length $S$, then it is possible to find a tree-like resolution refutation in time $n^{O(\log S+k)}$.*

**Problem 7** (20 points)**.** *Show that the Tseitin formula over an $w \times \ell$ grid as in Lecture 8 has a resolution refutation of length $O(2^{O(w)}\ell)$ and clause space $2^{O(w)}$.*
(Hint: think of summing all linear equations left to right, keeping at each point in time a linear equation with $O(w)$ variables in memory.)

**Problem 8** (20 points)**.** *Show that the Tseitin formula over an $w \times \ell$ grid as in Lecture 8 has a tree-like resolution refutation of length $\ell^{O(w)}$ and clause space $O(w \log \ell)$.*
(Hint: build a decision tree of height $O(w \log \ell)$ using a divide and conquer approach. Please provide a proof that this construction gives the desired upper bound.)

**Problem 9** (15 points)**.** *Assume that there is a refutation of $\phi$ in either* Pc*,* Pcr *or cutting planes where all memory configurations have at most $s$ variables. Show that there is resolution refutation of $\phi$ in variable space $s$.*

## *Pebbling tautologies*

**Problem 10** (20 points)**.** *Consider a black-white pebbling of $G$, and enumerate its pebbling configurations $P_1, P_2, \ldots P_\ell$. Show that $G$ has another* *black-white pebbling $P'_\ell, P'_{\ell-1}, \ldots, P'_2, P'_1$, where $P'_i$ is a copy of configuration $P_i$ but with the black and the white pebbles flipped.*

**Problem 11** (10 points)**.** *Consider a black pebbling of cost $s$. There is always a way to transform it in such a way that the time and space do not increase, but after every black pebble placement on sources the space of the configuration is at most $s - 1$.*

**Problem 12** (10 points)**.** *Use the worst case upper bound for black pebbling to show that any formula that has a refutation of length $L$ can be refutated in space $O(L/\log L)$.*

**Problem 13** (15 points)**.** *Prove that any graph with a pebbling of constant space can be pebbled in polynomial time and constant space simultaneously.*

**Problem 14** (10 points)**.** *Show that for every $G$ there is a refutation of* $\mathrm{Peb}(G)$ *of simultaneously*

- *length $O(n)$ and width $O(1)$;*

*and one of simultaneously*

- *length $O(n)$ and clause space $O(1)$.*

*Can we get a refutation of width and clause space $O(1)$ simulteneously, regardless the graph?*

**Problem 15** (15 points)**.** *Show that for every $G$ there is a refutation of* $\mathrm{Peb}^{\oplus_2}(G)$ *of simultaneously length $O(n)$ and width $O(1)$.*

**Problem 16** (15 points)**.** *Show how to obtain, from a black pebbling strategy of graph $G$ in time $t$ and space $s$, a refutation of* $\mathrm{Peb}(G)$ *of length $O(t)$, variable space $O(s)$ and width $O(1)$*

**Problem 17** (20 points)**.** *Show how to obtain, from a black pebbling strategy of graph $G$ in time $t$ and space $s$, a refutation of* $\mathrm{Peb}^{\oplus_2}(G)$ *of length $O(t)$, clause space $O(s)$ and width $O(1)$.*

**Problem 18** (35 points)**.** *Consider, without loss of generality, a resolution refutation*

$$M_0, M_1, \ldots M_t$$

*for* $\mathrm{Peb}(G)$*, where we assume that*

- *there is no weakening;*
- *all clauses in the proof are* essential*, where (1) the empty clause in $M_t$ is essential, (2) each clause used to derive an essential clause is essential.*

*Show that we can extract from any such proof of length $t$ and variable space $s$ a black-white pebbling strategy of length $O(t)$ and variable space at most $s$.*

(Hint: turn each memory configuration into a pebbling configuration, so that negative literals correspond to white pebbles and positive literals corresponds to black pebbles.)

(Hint: argue that the sink node is mentioned at some points and that this implies that the conversion produces a pebbling.)

## *References*