Assignment 11, Complexity Theory, SoSe 15

Markus Bläser, Holger Dell, Karteek Sreenivasaiah
http://www-cc.cs.uni-saarland.de/course/47/

Due: July 15, 2015, 11:00

Exercise 11.1 Let $\mathcal{C}$ be a complexity class. Show the following:

- If $\mathcal{C} = \text{co-}\mathcal{C}$, then $\mathsf{BP}\text{-}\mathcal{C} = \text{co-}\mathsf{BP}\text{-}\mathcal{C}$.
- If $\mathsf{BP}\text{-}\mathcal{C}$ allows probability amplification, then $\mathsf{BP}\text{-}\mathsf{BP}\text{-}\mathcal{C} = \mathsf{BP}\text{-}\mathcal{C}$.

Exercise 11.2 Show that the following problem is $\exists\oplus\mathsf{P}$ complete: Given a formula in 3-CNF in variables $y_1, \dots, y_n$ and $z_1, \dots, z_m$, is there an assigment to $y_1, \dots, y_n$ such that the resulting formula has an odd number of satisfying assignments to $z_1, \dots, z_m$?

A *straight line program* (or SLP for short), is a sequence of instructions that correspond to a sequential evaluation of an arithmetic circuit. Formally an SLP is a sequence of instructions $g_{-n}, \dots, g_{-1}, g_1, \dots, g_m$, where $g_{-n}, \dots, g_{-1} \in \mathbb{Z}$ and for $1 \le i \le m$, either $g_i \in \mathbb{Z}$, or $g_i = (*, L_i, R_i)$, with $* \in \{+, \times, -\}$, $L_i, R_i \in \{g_{-n}, \dots, g_{-1}, g_1, \dots, g_{i-1}\}$. Naturally for every SLP we can associate an integer value with every instruction g_i. The integer represented by the given SLP is the value of g_m. We define two computational problems on SLPs:

EquSLP Given an SLP P representing an integer N, decide if $N = 0$.

BitSLP Given an SLP representing an integer N, $n, i \in \mathbb{N}$, test if the ith bit in the n-bit binary representation of N is 1.

PosSLP Given an SLP computing an integer N, test if N is positive, i.e, $N > 0$.

Exercise 11.3 Show that *EquSLP* is polynomial time many-one equivalent to ACIT.

Exercise 11.4 Show that *BitSLP* is $\#\mathsf{P}$ hard under polynomial time Turing reductions.

In 2006, Allender, Bürgisser, Kjeldgaard-Pedersen and Miltersen showed that *PosSLP* lies in the counting hierarchy (CH), where CH is defined as the union of the classes PP, $\mathsf{PP}^{\mathsf{PP}}$, $\mathsf{PP}^{\mathsf{PP}^{\mathsf{PP}}}, \cdots$. However, it is believed that this bound is not tight, and an exact complexity characterization of *PosSLP* is an important open question. For example, showing NP hardness of *PosSLP* will certainly get one a Master's thesis, or even a PhD thesis! A PH upperbound will also be a breakthrough in understanding *PosSLP*.