Computational Complexity Theory - Assignment 7

Alex Hoppen (334784), Jan Uthoff (310336)

June 16, 2016

Exercise 25

Exercise 26

Most current results of the time/space tradeoff focus on SAT.

1. $\overline{\text{SAT}}$ cannot be solved in $n^{1+o(1)}$ time and $n^{1-\epsilon}$ space for any $\epsilon > 0$ general random-access non-deterministic Turing machines [For00].
2. A Turing machine that run in super-linear time an sublinear space can be simulated in alternating linear time [For00].
3. SATcannot be solved on general purpose random-access Turing machines in time $n^{1.618}$ and space $n^{O(1)}$.' SATcannot be solved for any constant b less than the golden ratio in time n^a and space n^b[FvM00]. The value of 1.618 is nearly equal to the value named in the book.

Exercise 27

In the following the i-th enumeration represents the i-th paragraph in the proof description.

1. $\mathsf{NP} = \Sigma_1^p$ and $\mathsf{co\text{-}NP} = \Pi_1^p$. Known from theorem 5.4: If $\Sigma_i^p = \Pi_i^p$ the polynomial hierarchy collapses to the i-th level. In this case $i = 1$ so it collapses not to the second level but first level. Furthermore $\mathsf{NP} = \mathsf{co\text{-}NP}$ does not follow by $\mathsf{NP} = \mathsf{NP}^{\mathsf{NP}}$. Because $\mathsf{NP} \subseteq \mathsf{P}^{\mathsf{NP}} \subseteq \mathsf{NP}^{\mathsf{NP}}$(trivial) and $\mathsf{co\text{-}NP} \subseteq \mathsf{P}^{\mathsf{NP}}$(trivial), $\mathsf{co\text{-}NP} \subseteq \mathsf{NP}$ follows.
2. This paragraph wants to show that $\mathsf{NP} \subseteq \mathsf{NP}^{\mathsf{NP}}$ holds. Therefore they name for every $\mathcal{L} \in \mathsf{NP}$ and non-deterministic polynomial time TM $\mathcal{M}$ that recognizes $\mathcal{L}$. In case a non-deterministic polynomial time oracle TM $\mathcal{M}'$ behaves exactly like $\mathcal{M}$ and so ignoring its oracle $\mathcal{M}'$ recognizes $\mathcal{L}$ too. So $\mathsf{NP} \subseteq \mathsf{NP}^{\mathsf{NP}}$.
3. This paragraphs wants to show that $\mathsf{NP}^{\mathsf{NP}} \subseteq \mathsf{NP}$. So assume $\mathcal{L} \in \mathsf{NP}^{\mathsf{NP}}$. Because $\mathcal{L} \in \mathsf{NP}^{\mathsf{NP}}$ there exists a non-deterministic polynomial time oracle TM $\mathcal{M}'$ that recognizes $\mathcal{L}$. Because the oracle, $\mathcal{M}'$ uses, decides problems in NP there exists a non-deterministic polynomial time TM $\mathcal{M}''$ that can decide the same problems as the oracle. Because the running time of both

TMs $\mathcal{M}'$ and $\mathcal{M}''$ are polynomial, a new non-deterministic polynomial time TM $\mathcal{M}^\star$ can be constructed that calculates the result of the oracle with the TM $\mathcal{M}''$ and uses that result and continues running like $\mathcal{M}'$, after $\mathcal{M}'$ uses its oracle. So $\mathcal{L} \in \mathsf{NP}$. In the last sentence it is mentioned that $\mathsf{NP} = \mathsf{co\text{-}NP} = \mathsf{P}^{\mathsf{NP}} = \mathsf{NP}^{\mathsf{NP}}$ holds. But $\mathsf{co\text{-}NP} = \mathsf{P}^{\mathsf{NP}}$ still must be shown. $\mathsf{co\text{-}NP} \subseteq \mathsf{P}^{\mathsf{NP}}$ is easy to show.

Proof $\mathsf{co\text{-}NP} \subseteq \mathsf{P}^{\mathsf{NP}}$ Let be $\mathcal{L}' \in \mathsf{co\text{-}NP}$. So $\overline{\mathcal{L}'} \in \mathsf{NP}$. Thus there is a non-deterministic polynomial time TM $\mathcal{M}$ that can recognize $\overline{\mathcal{L}'}$. By calculating the result of $\mathcal{M}$ and invert it, which can obviously be done deterministically, a deterministic polynomial time oracle TM can recognize $\mathcal{L}'$ as well.

$\mathsf{P}^{\mathsf{NP}} \subseteq \mathsf{co\text{-}NP}$ is non trivial and still to show. So the proof is incomplete.

Exercise 28

Definition: Padding Padding is the technique that fills a given word x and generates a new one $v =< x, 1^{u(|x|)} >$. With the new word the TM that is running on it has additional time for the calculation.

Theorem 1. *If there is a language $\mathcal{L} \in NTIME(f(n))$ than there exists a padding $u(|x|)$ such that $< x, 1^{u(|x|)} > \in \mathcal{L}_{pad}$ and $\mathcal{L}_{pad} \in DTIME(g(n))$ and*

$$x \in \mathcal{L} \Leftrightarrow < x, 1^{u(|x|} > \in \mathcal{L}_{pad}$$

for some f, g, u with $f(n) \in O(g(n))$.

References

[For00] Lance Fortnow. Time-space tradeoffs for satisfiability. *Journal of Computer and System Sciences*, 60:337–353, 2000.

[FvM00] L. Fortnow and D. van Melkebeek. Time-space tradeoffs for nondeterministic computation. In *Computational Complexity, 2000. Proceedings. 15th Annual IEEE Conference on*, pages 2–13, 2000.