COS 522: Complexity Theory : Boaz Barak
Handout 2: **NP**, Ladner's Theorem, Baker-Gill-Solovay.

Reading: Chapters 2,3

Review time hierarchy theorem, non-uniform hierarchy theorem.

The class NP. Definition, some examples.

NP-completeness. Karp reductions, trivial **NP**-complete language, the Cook-Levin Theorem. **NP**-completeness of independent set.

Search vs. decision reduction. If $\mathbf{P} = \mathbf{NP}$ then every **NP**-search problem can be solved in polynomial time.

Non-deterministic TMs Alternative definition of **NP**. Non-deterministic time hierarchy theorem.

Ladner's Theorem. If $\mathbf{P} \neq \mathbf{NP}$ then there are languages that are neither in **P** nor **NP**-complete.

Baker-Gill-Solovay. The **P** vs. **NP** question cannot be resolved using solely relativizing techniques.

coNP

Homework Assignments

§1 (20 points) Let HALT be the Halting problem. That is, HALT is the set containing the encoding of all pairs $\langle M, x \rangle$ such that the TM M halts on input x. Show that HALT is **NP**-hard.

§2 (30 points) Let QUADEQ be the language of all satisfiable sets of *quadratic equations* over 0/1 variables (a quadratic equations over $u_1, \ldots, u_n$ has the form $\sum_{i,j \in [n]} a_{i,j} u_i u_j = b$) where addition is modulo 2 (i.e., in the field $\mathsf{GF}(2)$). Show that QUADEQ is **NP**-complete. Will this still hold if we allow only *linear equations* (of degree 1)?

Also, give a direct reduction from the problem of checking satisfiability of *degree four* equations over $\mathsf{GF}(2)$ to QUADEQ.

§3 (30 points, Berman's Theorem 1978) A language is called *unary* if every string in it is of the form 1^i (the string of i ones) for some $i > 0$. Show that if there exists an **NP**-complete unary language then $\mathbf{P} = \mathbf{NP}$. See footnote for hint[1]

§4 (30 points) Let $\Sigma_2\mathsf{SAT}$ denote the following decision problem: given a quantified formula ψ of the form

$$\psi = \exists_{x \in \{0,1\}^n} \forall_{y \in \{0,1\}^m} \text{ such that } \varphi(x, y) = 1 \ ,$$

where φ is a CNF formula, decide whether ψ is true. (That is, whether there exists an x such that for every y, $\varphi(x, y)$ is true.) Prove that if $\mathbf{P} = \mathbf{NP}$ then $\Sigma_2\mathsf{SAT}$ is in **P**.

[1] **HINT:** Use the algorithm we used in the search to decision reduction to solve SAT.

Note: The class $\mathbf{\Sigma_2}$ is defined to be the set of all languages that reduce to $\Sigma_2\mathsf{SAT}$. Similarly, $\mathbf{\Sigma_3}$ is the set of all languages that reduce to $\Sigma_3\mathsf{SAT}$, which is the problem of determining truth of formulae of the form $\exists_x \forall_y \exists_z \varphi(x, y, z) = 1$, and for every i we can define the class $\mathbf{\Sigma_i}$. The *polynomial hierarchy* is the class **PH** consisting of the union of $\mathbf{\Sigma_i}$ for every i. A famous conjecture is that for every i, $\mathbf{\Sigma_i}$ is a proper subclass of $\mathbf{\Sigma_{i+1}}$. This is known as the conjecture that the polynomial hierarchy *does not collapse*. You can read more about the polynomial hierarchy in Chapter 5 of the book.

§5 (No points) Based on your personal taste, read one of the following surveys on the class **NP** (all are available from the course's web page):

- *"A personal view of average case complexity"* by Russell Impagliazzo (1995). Considers several possible scenarios for the potential difficulty of **NP**. For example, does **NP** have problems that are hard to solve not just in the worst case but also in the average case? Are there hard problems in **NP** that are " sufficiently structured" to allow applications for encryptions etc.? One way to phrase complexity theory's mission is to find out which of Russell's worlds is the one we live in.
- *"***P***,* **NP** *and Mathematics - a computational complexity perspective"* by Avi Wigderson (2006). Surveys the relations between the **P** vs. **NP** question and other areas of mathematical research.
- *"Is* **P** *vs.* **NP** *formally independent?"* (2003) by Scott Aaronson. Surveys what is known about the following troubling possibility: that it is impossible to resolve the **P** vs. **NP** question using the currently accepted axioms of mathematics.
- *"***NP***-complete problems and physical reality"* by Scott Aaronson (2005). Surveys various attempts that have been made by mainstream and less than mainstream scientists to solve **NP**-complete problems via various "non-traditional" computing devices.
- *"The History and Status of the* **P** *vs.* **NP** *question"* by Michael Sipser (1992). Surveys the history of this question (up to 1992) and the attempts to resolve it.

NOTE: Next week we will start talking about probabilistic algorithms and random walks. Now is a good time to refresh your memory on both probability theory and elementary linear algebra (inner product, eigenvectors, eigenvalues, etc..) The appendix of the book contains some info on these, and also the discrete math lecture notes of Lehman-Leighton, Trevisan, Papadimitriou-Vazirani are good places to look at.