Revision and circuits

Exercise 1  Revision

1. How fast can you simulate a TM with $k$ tapes working on an alphabet $\Sigma$, using alphabet $\{0, 1\}$ and using 2 read&write tapes (with input, then output on first one)? and with 1 tape? $\log |\Sigma| \times T \log T$ and $\log |\Sigma| \times T^2$.

2. Assuming $P \neq NP$, show that there exists an undecidable problem which is not NP-hard. Hint: enumerate all reductions of SAT that have infinite image.

- First, $P = NP \implies$ any non trivial language is NP-complete. Hence the hypothesis is more than natural.
- Second, we will need this lemma:
  As $P \neq NP$, any poly time reduction from SAT to a problem $A$ has infinite image.
- Let $\{f_n\}$ be an enumeration of all polynomial reductions. Given a language $A$,
  $$\text{SAT} \leq_p A \iff \exists i \forall \phi (\phi \in \text{SAT} \iff f_i(\phi) \in A)$$
  Let us build $A$ in order to contradict this assumption.
  - We construct a family $\{\phi_i\}$ with $|f_i(\phi_i)| > \max_{j=1..i-1} |f_j(\phi_j)|$ such that
    $$(\phi \notin \text{SAT} \iff f_i(\phi) \in A)$$
    It is possible using the lemma, and fixing membership to $A$ on $\{f_i(\phi_i)\}$ guarantees that $A$ is not NP-hard.
  - In such $A$s, we now pick an undecidable one. It exists because the family of such $A$s is uncountable, and the set of decidable languages is countable (using a TMs enumeration).

3. Recall that $C^{C_2} = \bigcup_{C \subseteq C} C^{C_2}$. Do we always have $C_1 = C_2 \implies C^{C_1} = C^{C_2}$? $C_1 = C_2 \implies C^{C_1} = C^{C_2}$? This definition does not really make sense, since an oracle is provided to a machine, not a language. Still, the first equality uses the same machines on the left and the right, hence is true whatever computational model each language of $C$ was defined with.

4. Show that if this implication holds:

$$f, g \in \text{SPACE}(s(n)) \implies f \circ g(x) \in \text{SPACE}(s(|x|) + \log(|g(x)|))$$

then EXP = PSPACE. What do you know about this equality? You know that EXP $\supset$ PSPACE, and that equality is an open question. So this proves my complexity bound is stupid.

Now, $x \mapsto x\#1^{|x|} \mapsto 1_{x \in A}$ along with the stupid bound gives a PSPACE algo for any $A \in \text{EXP}$. A correct bound is $s(|x|) + \log(|g(x)|) + s(|g(x)|)$.

Exercise 2  Shannon 1949

1. Show that any function can be computed using a boolean circuit of size less than $O(2^n)$. Write $\phi(x, y) = (x \land \phi(1, y)) \lor (\neg x \land \phi(0, y))$ and unroll the recurrence.

2. Improve this bound to $O(2^n/n)$. For $n$ variables, unroll only up to $k = \log(n)$, and use a huge circuit to compute all possible functions on $k$ variables once and forall. This is sometimes called baby steps giant steps. You saw in course that this asymptotic behavior is best possible (up to the constant).
Exercise 3  Circuits warm-up

1. Show that given $c > 0$ and $n$ large enough, not every subset of $\Sigma^n$ can be decided by a circuit of size $n^c$.
2. Recall an undecidable language in $\text{P/poly}$.

Exercise 4  Schaefer, 1999

If $S = \{S_i\}$ is a finite collection of concepts (=subsets) of a finite universe $U$, the Vapnik-Chervonenkis (VC) dimension, denoted $\text{VC}(S)$ is the size of the largest $X \subseteq S$ which has the following property:

$$\forall X' \subseteq X, \quad \exists i, \ X' = X \cap S_i.$$

We are interested in studying this dimension — which is a major indicator of difficulty in statistical learning — in cases where the concepts may have exponential sizes, with regard to their description. Typically, this is the case for monomials in $\{0, 1\}^n$ for instance, where one bit encodes one concept.

A boolean circuit $C$ succinctly represents $S$ if $S_i = \{x; \ C(i,x) = 1\}$ for all $i$. Finally, the problem VC-dimension is the set of $(C,k)$ where $C$ represents a collection $S$ s.t. $\text{VC}(S) \geq k$

1. Show that if $(C,k)$ is in VC-dimension, then $k \leq |C|$.
2. Show that this problem is in $\Sigma^p_3$.
3. How would you prove it is $\Sigma^p_3$-complete?
4. Do it. (this is hard, try and ask me when you get there)

Exercise 5  Puzzle

Ask me when you are finished!