GROUP HOMEWORK 10, CPSC 421/501, FALL 2023

JOEL FRIEDMAN

Please note:

(1) You must justify all answers; no credit is given for a correct answer without justification.
(2) Proofs should be written out formally.
(3) You do not have to use LaTeX for homework, but **homework that is too difficult to read will not be graded.**
(4) You may work together on homework in groups of up to four, **but you must submit a single homework as a group submission under Gradescope.**

For the problems below, $\wedge, \vee, \neg$ denote the logical operations AND,OR,NOT respectively.

For the problems below, recall that to show that L is NP-complete one needs to show (i) $L \in$ NP, and (ii) any language in NP can be reduced to L (with a polynomial time reduction); (ii) is equivalent to reducing some known NP-complete language to L (with a polynomial time reduction).

(0) Who are your group members? Please print if writing by hand.

(1) Using class notes on November 20, we know that the formula

$$(x_1 \wedge x_2 \wedge x_3) \Rightarrow R$$

is equivalent to

$$\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee R.$$

Let $f = f(x_1, x_2, x_3, u_1, u_2, u_3, v_1, v_2, v_3, w_1, w_2, w_3)$ be given by

$$f = \neg x_1 \vee \neg x_2 \vee \neg x_3 \vee R,$$

where $R = R(u_1, u_2, u_3, v_1, v_2, v_3, w_1, w_2, w_3)$ be the formula

$$(u_1 \wedge u_2 \wedge u_3) \vee (v_1 \wedge v_2 \wedge v_3) \vee (w_1 \wedge w_2 \wedge w_3)$$

Research supported in part by an NSERC grant.

which can be written equivalently as

$$\left(\bigwedge_{i=1}^{3} u_i\right) \vee \left(\bigwedge_{j=1}^{3} v_j\right) \vee \left(\bigwedge_{k=1}^{3} w_k\right)$$

(a) Show that R above is equivalent to a 3CNF formula; what is the size of the formula, where the size is the number of literals (i.e., variables or their negations) appearing? (For example, the size of the formula $(y_1 \vee y_1) \vee \neg y_2 \vee y_1$ is four.) **Briefly justify your answer.**

(b) Show that

$$(x_1 \wedge x_2 \wedge x_3) \Rightarrow R$$

can therefore be written as a 6CNF, i.e., an AND ($\wedge$) of terms (or "clauses"), each term (clause) being an OR ($\vee$) of 6 variables. Show that the resulting formula is size $27 \times 6 = 162$. **Briefly justify your answer.**

(c) Recall the trick that $a_1 \vee a_2 \vee \ldots \vee a_6$ is true iff

$$(a_1 \vee a_2 \vee z_1) \wedge (\neg z_1 \vee a_3 \vee z_2) \wedge (\neg z_2 \vee a_4 \vee z_3) \wedge (\neg z_3 \vee a_5 \vee a_6)$$

is satisfiable (where z_1, z_2, z_3 are new, "auxilliary" variables). Given the 6CNF formula in part (b), write a 3CNF formula (which may introduce some new, auxilliary variables) that is satisfiable iff f is satisfiable; use a different set of 3 new, auxilliary variables for each term/clause for which you are using this trick. Show that the resulting formula size is $27 \times 12 = 324$. **Briefly justify your answer.**

(d) Is it possible to shorten the 3CNF in part (c)? Are some of the clauses redundant? Explain.

(2) Say that we have a non-deterministic Turning machine, $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{\mathrm{acc}}, q_{\mathrm{rej}})$ that decides a language, L, in time Cn^k. Recall that for an input of size n to M, we described the configurations of this Turing machine from step 1 to step Cn^k using the variables $x_{ij\gamma}$ (true iff $\gamma \in \Gamma$ is written in cell$[i,j]$, i.e., the tape cell in position j from the left, at step i); y_{ij} (true iff the tape head is over cell$[i,j]$, i.e., is in position j at step i), and z_{iq} (true iff at step i we are in state q).

(a) Say that for some $q \in Q$ and $\gamma \in \Gamma$ we have

$$\delta(q,\gamma) = \bigl\{(q_4, a, R), (q_7, b, L), (q_8, c, R)\bigr\}.$$

In the proof of the Cook-Levin theorem in class, this value of δ yields the condition (assuming $j \geq 2$):

$$(x_{ij\gamma} \wedge y_{ij} \wedge z_{iq})$$
$$\Rightarrow \Bigl((x_{i+1,j,a} \wedge y_{i+1,j+1} \wedge z_{i+1,q_4}) \vee (x_{i+1,j,b} \wedge y_{i+1,j-1} \wedge z_{i+1,q_7}) \vee (x_{i+1,j,c} \wedge y_{i+1,j+1} \wedge z_{i+1,q_8})\Bigr)$$

If we use Problem 1(c) to express the truth of this condition as the satisfiability of a 3CNF (which will introduce some new, auxilliary variables), what is the resulting formula size?

(b) Say that $j = 1$ in the above, and we adapt the convention that if the tape head is over cell 1 (i.e., the leftmost cell), then the instruction to move L means that the tape head remains in cell 1. What changes in the above formulae? Does this change the 3CNF size?

(c) What would be the resulting formula size if $\delta(q, \gamma)$ had five possible transitions instead of three? What would be the resulting size of the 3CNF formula derived in the style of Problem 1(c)? (Don't worry about writing a shorter formula using redundant clauses, like we did in Problem 1(d).)

(3) Let EVEN be the language of strings over $\{0, 1, \ldots, 9\}$ that represent positive, even integers. Of course, EVEN is a regular language, and hence decidable in time $n + O(1)$. Hence certainly EVEN $\in$ NP.

Consider the following reduction of 3SAT to EVEN (compare Definitions 7.28 and 7.29 of [Sip]): let $\Sigma_{\mathrm{Bool}} = \{0, \ldots, 9, x, \vee, \wedge, \neg, (,)\}$, our usual alphabet for describing Boolean formulas. Given a string $w \in \Sigma^*_{\mathrm{Bool}}$, do the following:

(a) check if $w = \langle f \rangle$, where f is a Boolean formula in $x_1, \ldots, x_n$ for some n that is in 3CNF form; if not, "return" the word 3, i.e., write the word 3 on the machine tape[1] and halt;

(b) otherwise, i.e., if $w = \langle f \rangle$ for some 3CNF formula f, for each possible assignment of $a_1, \ldots, a_n$ to the values T, F, see if $f(a_1, \ldots, a_n)$ evaluates to T. If this happens for one such assignment, return the word 4, otherwise return the word 5.

This algorithm gives a map

$$g \colon \Sigma^*_{\mathrm{Bool}} \to \{0, 1, \ldots, 9\}^*$$

such that

$$w \in \mathrm{3SAT} \iff g(w) \in \mathrm{EVEN}.$$

Does this mean that EVEN is NP-complete? Explain. (A brief explanation will suffice, despite the rather long statement of this problem.)

(4) Let A, B, C be languages such that $A \le_{\mathrm{P}} B$ using a reduction that takes time $5n^4$, and $B \le_{\mathrm{P}} C$ using a reduction that takes time $3n^8$ (where, as usual, n denotes the length of the input). This implies that $A \le_{\mathrm{P}} C$, by combining the two reductions; give an upper bound on the running time of this reduction. Explain.

(5) Recall that a 3CNF formula on variables $x_1, \ldots, x_n$ is a Boolean formula of the form

$$(\ell_{11} \vee \ell_{12} \vee \ell_{13}) \wedge (\ell_{21} \vee \ell_{22} \vee \ell_{23}) \wedge \ldots \wedge (\ell_{m1} \vee \ell_{m2} \vee \ell_{m3}),$$

or, in more concise notation,

$$\bigwedge_{i=1}^{m} (\ell_{i1} \vee \ell_{i2} \vee \ell_{i3}),$$

[1] i.e., write a 3 on the first tape cell and make sure all other tape cells have a blank symbol.

where each ℓ_{ij} is a *literal*, i.e., one of $x_1, \neg x_1, \ldots, x_n, \neg x_n$; in this case term $c_i = \ell_{i1} \vee \ell_{i2} \vee \ell_{i3}$ is referred to as a *clause* of the 3CNF formula, and m is the *number of clauses* of the formula.

(a) Show that any clause $\ell_{i1} \vee \ell_{i2} \vee \ell_{i3}$ is either (i) true for all values of $x_1, \ldots, x_n$, or (ii) false on **at least** $2^{n-3} = 2^n/8$ of all possible the 2^n values of $(x_1, \ldots, x_n) \in \{T, F\}^n$. [More precisely, it is false on exactly 2^{n-3} values if the clause has no repeated variables, and otherwise the same with 2^{n-3} replaced with either 2^{n-2} or 2^{n-1}.[2]]

(b) Show that the Boolean function $\{T, F\}^n \to \{T, F\}$ represented by any 3CNF is either (i) true for all values of $x_1, \ldots, x_n$, or (ii) false on **at least** $2^{n-3} = 2^n/8$ of all possible the 2^n values of $(x_1, \ldots, x_n) \in \{T, F\}^n$.

(c) Show that the Boolean function $f \colon \{T, F\}^4 \to \{T, F\}$ given by

$$f(x_1, x_2, x_3, x_4) = x_1 \vee x_2 \vee x_3 \vee x_4$$

cannot be written as a 3CNF.[3]

Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, CANADA.
E-mail address: jf@cs.ubc.ca
URL: http://www.cs.ubc.ca/~jf

[2] We thank an anonymous post on Piazza for this correction.

[3] By contrast, in the proof of the Cook-Levin theorem we make use of the fact that $f(x_1, x_2, x_3, x_4) = x_1 \vee x_2 \vee x_3 \vee x_4$ is true iff $g(x_1, x_2, x_3, x_4, z)$ given by $(x_1 \vee x_2 \vee z) \wedge (\neg z \vee x_3 \vee x_4)$ is satisfiable. Hence f is true iff g is satisfiable, but g is not a 3CNF form of f.