CSE491/596, Fall 2023 Problem Set 4 Due Wed. 11/1, 11:59pm

Lectures this week continue with the ALR notes, chapter 28, section 4. The problems there rather than those in Debray's section 16 are being lectured on. Next week, however, we will give full treatment to space complexity and its relation to time complexity before handling the completeness results in section 4.5 of that chapter. There is a "hole" in both ALR chapter 27 and in Debray's notes, where ALR states Theorem 2.3 (in the short "Basic Relationships" section 2.4) without giving proofs, while Debray sketches the proof tersely at the top of page 56 (near the end of section 17) but does not make a separate theorem statement. The hole is filled by page 1 of this handout linked from the course webpage as item 6 of required reading (please ignore the 2018 calendar dates):

https://cse.buffalo.edu/~regan/cse491596/CSE596inclusions.pdf

The remaining pages 2–4 of this handout are heavy going for the DSPACE and DTIME Hierarchy Theorems, but they tie together material that is equally heavy going in Debray's notes (split between Theorems 13.7–13.8 and Theorem 18.1) and ALR chapter 27, Theorem 2.5 in section 2.6 (skip the NTIME proof). Thus the reading for next week becomes:

- The rest of Debray, sections 13–18, except we are still postponing the discussion of oracles in the first two pages of section 17, and the TQBF proof will be covered the following week when we rejoin ALR chapter 28 in section 5.
- ALR chapter 27, sections 2.4–2.7, and the above handout.

Lectures are currently aligned with the Week 10 Wednesday lecture from Fall 2021 (which the Fall 2022 lectures point to) in the 2021 version.

—————-*Assignment 4, due Wed. 11/1 "midnight stretchy" on CSE Autolab*—————-

(1) Give a polynomial-time many-one reduction from 3SAT to the following problem:

Instance: A system of equations of the form $p_1(x_1,\dots,x_n)=0$, $p_2(x_1,\dots,x_n)=0$, ..., $p_m(x_1,\dots,x_n)=0$, where each p_j is a polynomial of degree at most 3 in (some subset of) the variables $x_1,\dots,x_n$.

Question: Does the system have a solution in which all the x_i are 0 or 1?

Incidentally, if we add the equations $x_1^2-x_1=0$, $x_2^2-x_2=0$, ..., $x_n^2-x_n=0$ to the system, then we actually enforce the condition that all x_i are 0 or 1. Thus this shows that solving general systems of equations is NP-hard, even when each equation is limited to degree 3 and a handful of variables. Apply the idea of converting any given 3CNF clause into an equation that is solved only by those 0-1 assignments that satisfy the clause. (18 pts.)

(2) Let $C=(\ell_1\vee\ell_2\vee\ell_3)$ be a clause with three literals, where ℓ_1 is x or $\bar{x}$, ℓ_2 is y or $\bar{y}$, and ℓ_3 is z or $\bar{z}$. (It is often convenient to use this notation so that you can write, e.g., $\bar{\ell}_1$ without having to say whether ℓ_1 is x or $\bar{x}$.) Show how to map C to a formula ψ_C, where ψ_C is a conjunction of clauses $C_1\wedge C_2\wedge\cdots\wedge C_k$ with extra variables $w,v,u,\dots$, such that:

An assignment a to x, y, z satisfies C if and only if a can be extended to an assignment a' of all the variables that not only satisfies ψ_C, but also makes **exactly one** literal in each of its k clauses true.

Note that if a is the one assignment (out of eight) that fails to satisfy C, then you may still be able to satisfy ψ_C, but the point is that you won't be able to make *exactly* one literal in each of its clauses true. Use this fact to reduce 3SAT to each of the following two problems:

EXACTLY ONE 3SAT

Instance: A Boolean formula $\phi(x_1, \dots, x_n) = C_1 \wedge \cdots \wedge C_m$ in 3CNF.

Question: Is there an assignment to the variables that makes *exactly one literal* in each clause true?

EXACTLY TWO 3SAT

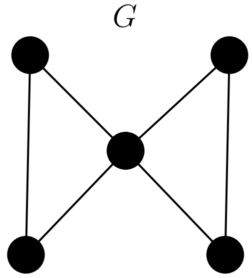
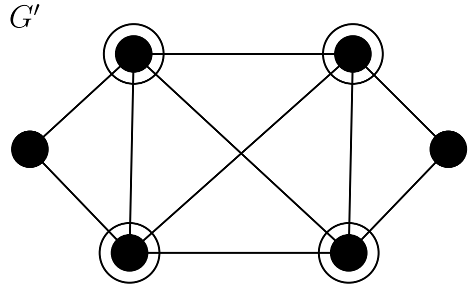
Instance: A Boolean formula $\phi(x_1, \dots, x_n) = C_1 \wedge \cdots \wedge C_m$ in 3CNF.

Question: Is there an assignment to the variables that makes *exactly two literals* in each clause true? (30 pts. total)

(3) For an undirected graph $G = (V, E)$, define its *line graph* $G' = (V', E')$ by $V' = E$ and

$$E' = \{((u, v), (v, w)) : (u, v) \in E \wedge (v, w) \in E\},$$

recalling that in an undirected graph, $(u, v) \in E \iff (v, u) \in E$. That is, G' has a vertex for every edge of G, and two edges that "touch" in G become an edge in G'. For example, the graph G' at right is the line graph of the 5-node "bowtie graph" at left, and the circles show how the four edges touching the center vertex of the bowtie become a clique in G':

G  G' 

Now consider the following decision problem:

EDGE COVER

Instance: An undirected graph G and an integer $k \geq 1$.

Question: Does there exist a set H of at most k edges such that every other edge in G touches an edge in H?

(a) Does $f(G, k) = (G', k)$ give a polynomial-time mapping reduction (i) from VERTEX COVER to EDGE COVER?, (ii) from EDGE COVER to VERTEX COVER? (iii) both? (iv) neither? Justify your answer. (12 pts.)

(b) Show that EDGE COVER is NP-complete *by mapping reduction from* 3SAT. (You are allowed to use any of the variant forms of 3SAT we have seen thus far, namely "not all equal 3SAT" and the versions in problem (2), but you need not do so. 30 pts., making 42 on the problem and 90 on the set)