CSCI 570 - Fall 2022 - HW 10 Solution

Due: November 16, 2022

Problem 1 (25pts)

Consider the partial satisfiability problem, denoted as 3-Sat(α). We are given a collection of k clauses, each of which contains exactly three literals, and we are asked to determine whether there is an assignment of true/false values to the literals such that at least αk clauses will be true. Note that 3-Sat(1) is exactly the 3-SAT problem from lecture.

Prove that 3-Sat(15/16) is **NP**-complete.

Hint: If x, y, and z are literals, there are eight possible clauses containing them: $(x \vee y \vee z), (!x \vee y \vee z), (x \vee !y \vee z), (x \vee y \vee !z), (!x \vee !y \vee z), (!x \vee y \vee !z), (x \vee !y \vee !z), (!x \vee !y \vee !z)$

Solution:

To prove it's in NP: given a truth value assignment, we can count how many clauses are satisfied and compare it to 15k/16.

To prove it's NP-hard:
We will show that 3-SAT $\leq_p$ 3-SAT(15/16). For each set of 8 original clauses, create 8 new clauses using 3 new variables, and construct the clauses by adding the collection of all possible clauses on the 3 new variables.

Claim: there exists a solution to the original 3-SAT problem if and only if there exists a solution to the constructed 3-SAT(15/16) problem. If the number of clauses is a multiple of 8, we are done: any assignment will satisfy 7/8 of the new clauses; in a valid solution to the 3-SAT(15/16) problem where at least 15/16 of all the clauses are satisfied, all of the original clauses must be satisfied. If the number of clauses is not a multiple of 8, a valid solution will satisfy more than 15/16 of all the clauses. If any one of the original clauses is not satisfied, less than 15/16 of the clauses can be satisfied in the new formula.
Therefore, 3-Sat (15/16) is in the intersection of NP and NP-hard which is NP-Complete.

Example: If the original collection of clauses is

$(a \vee b \vee c) \wedge (!a \vee b \vee c) \wedge (a \vee !b \vee c) \wedge (a \vee b \vee !c) \wedge (!a \vee !b \vee c) \wedge (d \vee e \vee f) \wedge (g \vee !b \vee !c) \wedge (!a \vee !h \vee !c)$

We add 8 new clauses so that the formula now contains 16 clauses in total, i.e.: $(a \vee b \vee c) \wedge (!a \vee b \vee c) \wedge (a \vee !b \vee c) \wedge (a \vee b \vee !c) \wedge (!a \vee !b \vee c) \wedge (d \vee e \vee f) \wedge (g \vee !b \vee !c) \wedge (!a \vee !h \vee !c) \wedge (x \vee y \vee z) \wedge (!x \vee y \vee z) \wedge (x \vee !y \vee z) \wedge (x \vee y \vee !z) \wedge (!x \vee !y \vee z) \wedge (!x \vee y \vee !z) \wedge (x \vee !y \vee !z) \wedge (!x \vee !y \vee !z)$

Note we add 8 new clauses for every 8 original clauses. And as described above, if there is a solution to the constructed 3-SAT (15/16) problem, at least 15 clauses must be satisfied. Since only 7 of the 8 new clauses can be satisfied, all of the original clauses must be satisfied, i.e., there is a solution to the original 3-SAT problem.

Rubric (25 pts)

- 5 pt: State the problem is in NP and mention how to verify it in polynomial time by checking the number of satisfied clauses.
- 10 pt: Choose a NP-complete problem and say that we need to prove that chosen problem $\leq_p$ 3-SAT (15/16). Note: wrong direction of reduction will get 0.
- 10 pt: Show the polynomial reduction with details and proof including: 5 pt: instance construction, 5 pt: show polynomial reduction using the construction.

Problem 2 (25 pts)

Given a graph $G = (V, E)$ and two integers k, m, the Dense Subgraph Problem is to find a subset V' of V, whose size is at most k and are connected by at least m edges. Prove that the Dense Subgraph Problem is NP-Complete.

Solution:

Algorithm: To prove it's in NP: given a subgraph of G, we check if the number of vertices in the subgraph is less than or equal to k and there are at least m edges in the graph.
To prove it's NP-hard:
We prove that the Independent set problem $\leq_P$ Dense Subgraph Problem. Given a graph $G(V, E)$ and an integer k, an independent set decision problem outputs yes, if the graph contains an independent set of size k. For an arbitrary graph $G = (V, E)$ of n vertices, we first get its complementary graph G_c.
A clique is a subset of vertices of an undirected graph G such that every two distinct vertices in the clique are adjacent, i.e., it's an induced subgraph of G that is complete. We know that a clique will always contain $k(k-1)/2$ edges if there are k vertices in G, and an independent set in G is a clique in G_c (the complement graph of G) and vice versa.
Then we set m to $k(k-1)/2$ and test with the dense subgraph problem.

Claim: There exists an independent set of size k in G (equivalently, a clique in G_c of size k), if and only if there exists a subgraph of G_c with at most k vertices and at least $m = k(k-1)/2$ edges.

$\rightarrow$) If there exists a clique in G_c of size at least k, there exists a subgraph of G_c with at most k vertices and at least $k(k-1)/2$ edges.

If there is a clique of size at least k, there is a clique of size exactly k. Moreover, by explanation above, a clique of size k would have $\mathrm{k} * (\mathrm{k}-1)/2$ edges.

$\leftarrow$) If there exists a subgraph of G_c with at most k vertices and at least $k(k-1)/2$ edges, there is a clique of size at least k.

For a subgraph to have at least $k(k-1)$ edges, it must have at least k vertices. So there are exact k vertices in the subgraph. Hence, this subgraph with k vertices forms a clique in G_c of size k.

Example: Consider a graph $G = (V, E)$ where $V = \{1, 2, 3, 4\}$ and $E = \{(1, 2), (2, 3)\}$. To determine if there's an independent set in G of size 3, we first generate G_c, where $E_c = \{(1, 3), (1, 4), (2, 4), (3, 4)\}$. Since there's a dense graph with 3 vertices and 3*2/2 = 3 edges in G_c, which is $V' = \{1,3,4\}, E' = \{(1,3),(1,4),(3,4)\}$, we can say there's an independent set in G of size 3 . On the other hand, if we set $k = 4$ and $m = 4 * 3/2 = 6$, there's no such dense graph in G_c, thus no independent set of size 4 in G.

Rubric (25 pts)

- 5 pt: state the problem is in NP and mention how to verify it in polynomial time.
- 5 pt: Choose a NP-complete problem and say that we need to prove that chosen problem $\leq_p$ Dense Subgraph problem. Other reduction can also get 3 points. Note: wrong direction will get 0 .
- 15 pt: Show the polynomial reduction with details and its proof including: 5 pt: instance construction, 10 pt: prove the claim. Each direction worths 5 pts.

Problem 3 (25 pts)

Consider a modified SAT problem, SAT' in which given a CNF formula having m clauses and n variables $x_1, x_2, \ldots, x_n$, the output is YES if there is an assignment to the variables such that exactly $m-2$ clauses are satisfied, and NO otherwise. Prove that SAT' is NP-Complete.

Solution:

To show that SAT' is NP-Complete,
First we will show that SAT' $\in$ NP: Given the assignment values as certificate, we can evaluate the SAT' instance and verify if it is satisfied. This is same as the SAT-verification. Moreover, we can count the number of satisfied clauses and check if it is equal to $\mathrm{m}-2$ in linear time.
Next, we show that SAT $\leq$ p SAT': Construction: Add four more clauses $x_1, x_2, \neg x_1, \neg x_2$ to the original SAT instance.

Claim: CNF formula obtained for SAT', F' has an assignment which satisfies SAT' iff CNF formula of SAT, F has an assignment which satisfies SAT.

$\Rightarrow$) if F has an assignment which satisfies SAT, then F' has an assignment which satisfies SAT'
Proof: If an assignment $x_1 \ldots x_n$ satisfies F, it satisfies exactly two of the four extra clauses, giving exactly $m+2$, which is m ' -2 satisfied clauses for F'.

$\Leftarrow$) if F' has an assignment which satisfies SAT', then F has an assignment which satisfies SAT
Proof: By construction, the only unsatisfied clauses for F must be one of x_1 or $\neg x_1$ and one of x_2 or $\neg x_2$, so all the original m clauses are satisfied.

Rubric (25 pts)

- 5 pt: state the problem is in NP and mention how to verify it in polynomial time.
- 5 pt: Choose a NP-complete problem and say that we need to prove that chosen problem $\leq_p$ SAT'. Other reduction can also get 3 points. Note: wrong direction will get 0 .
- 15 pt: Show the polynomial reduction with details and its proof including: 5 pt: instance construction, 10 pt: prove the claim. Each direction worths 5 pts.

Problem 4 (25 pts)

Show that Vertex Cover is still NP-complete even when all vertices in the graph are restricted to have even degree.

Solution:

To show that VCE (Vertex Cover with only Even-degree vertices) is NP-Complete, we first show that it's in NP. The certifier for VCE is the same as the certifier for VC (Vertex Cover). It takes an undirected graph and a set of vertices as input. It verifies if the union of all edges touching the set of vertices covers all the graph edges. The certifier runs in polynomial time.

We next show that VC $\leq_p$ VCE.

Let $\langle G, k\rangle$ be an input instance for VC, where $G = (V, E)$. Because each edge in E contributes a count of 1 to the degree of each of the two vertices it connects, the sum of the degrees of all vertices in G is $2|E|$, an even number. Hence, there is an even number of vertices in G that have odd degrees. Let U be the subset of vertices with odd degrees in G.

Based on $\langle G, k\rangle$ for VC, we construct an instance $\langle \bar{G}, k+2\rangle$ for VCE, where $\bar{G} = (V_0, E_0)$, $V_0 = V \cup \{p, p_1, p_2\}$, $E_0 = E \cup \{(p, p_1), (p, p_2), (p_1, p_2)\} \cup \{(p, v) \mid v \in U\}$. That is, we make a triangle with three new vertices p, p_1, p_2, and then connect p to all vertices in U. The degree of every vertex in V_0 is even, making $\langle \bar{G}, k+2\rangle$ a valid instance for VCE.

Below we prove that $\langle G, k\rangle$ is satisfiable for VC if and only if $\langle \bar{G}, k+2\rangle$ is satisfiable for VCE.

$\Rightarrow$: If G has a vertex cover of size at most k, then $\bar{G}$ has a vertex cover of size at most $k+2$.

Proof: Denote the vertex cover for G as S. Then $S \cup \{p, p_1\}$ is a vertex cover for $\bar{G}$ as S covers the edges in G, $\{p, p_1\}$ covers the triangle, $\{p\}$ covers the edges connecting p and U.

$\Leftarrow$: If $\bar{G}$ has a vertex cover of size at most $k+2$, then G has a vertex cover of size at most k.

Proof: Denote the vertex cover for $\bar{G}$ as $\bar{S}$. $\bar{S}$ should consist of vertices from the triangle and vertices from G. $\bar{S}$ should include at least two vertices from the triangle to cover it. Removing triangle vertices from $\bar{S}$, we will get a vertex cover for G.

Rubric (25 pts)

- 5 pts: Show that VCE is in NP.
- 10 pts: Choose a NPC problem and show a polynomial reduction algorithm to convert an instance of the NPC problem to an instance of VCE.
- 10 pts: Prove that the instance of the NPC problem is satisfiable if and only if the corresponding instance of the VCE problem is satisfiable (5 pts for $\Rightarrow$ and 5 pts for $\Leftarrow$). The proof should include a polynomial algorithm to convert a solution to NPC to a solution to VCE.

Practice Problems

Problem 5 (25 pts)

(Kleinberg and Tardos, Chapter 8, Exercise 5)

Consider a set $A = \{a_1, \ldots, a_n\}$ and a collection $B_1, B_2, \ldots, B_m$ of subsets of A (i.e., $B_i \subseteq A$ for each i).

We say that a set $H \subseteq A$ is a *hitting set* for the collection $B_1, B_2, \ldots, B_m$ if H contains at least one element from each B_i—that is, if $H \cap B_i$ is not empty for each i (so H "hits" all the sets B_i).

We now define the *Hitting Set Problem* as follows. We are given a set $A = \{a_1, \ldots, a_n\}$, a collection $B_1, B_2, \ldots, B_m$ of subsets of A, and a number k. We are asked: Is there a hitting set $H \subseteq A$ for $B_1, B_2, \ldots, B_m$ so that the size of H is at most k?

Prove that Hitting Set is NP-complete.

Solution:
To show that Hitting Set is NP-Complete, we first show that it's in NP. For an instance of Hitting Set, given a set of nodes H, we can verify if $|H| \leq k$ and $H \cap B_i \neq \emptyset$ for $i = 1, 2, \ldots, m$ in polynomial time.

We next show that Vertex Cover $\leq_p$ Hitting Set.

Let $\langle G, k \rangle$ be an input instance for Vertex Cover, where $G = (V, E)$. We construct an instance $\langle V, \{B_1, B_2, \ldots, B_{|E|}\}, k \rangle$ for Hitting Set. The subset collection $\{B_1, B_2, \ldots, B_{|E|}\}$ is constructed in the way that for the i-th edge $(u, v) \in E$, we have $B_i = \{u, v\}$.

Below we prove that $\langle G, k \rangle$ is satisfiable for Vertex Cover if and only if $\langle V, \{B_1, B_2, \ldots, B_{|E|}\}, k \rangle$ is satisfiable for Hitting Set.

$\Rightarrow$: If G has a vertex cover of size at most k, then there exists a hitting set $H \subseteq V$ of size at most k for $\{B_1, B_2, \ldots, B_{|E|}\}$.

Proof: Denote the vertex cover for G as S, then $|S| \leq k$ and for every edge $(u, v) \in E$, we have $u \in S \vee v \in S$. Therefore, $S \cap B_i \neq \emptyset$ for $i = 1, 2 \ldots, m$, and thus S is a hitting set that satisfies the constraints.

$\Leftarrow$: If there exists a hitting set $H \subseteq V$ of size at most k for $\{B_1, B_2, \ldots, B_{|E|}\}$, then G has a vertex cover of size at most k.

Proof: As H is the hitting set for $\{B_1, B_2, \ldots, B_{|E|}\}$, we have $H \cap B_i \neq \emptyset$ for $i = 1, 2 \ldots, m$. That is, for each $(u, v) \in E$, $H \cap \{u, v\} \neq \emptyset$. Therefore, H is a vertex cover for G.

Rubric (25 pts)

- 5 pts: Show that Hitting Set is in NP.
- 10 pts: Choose a NPC problem and show a polynomial reduction algorithm to convert an instance of the NPC problem to an instance of Hitting Set.
- 10 pts: Prove that the instance of the NPC problem is satisfiable if and only if the corresponding instance of the Hitting Set problem is satisfiable (5 pts for $\Rightarrow$ and 5 pts for $\Leftarrow$). The proof should include a polynomial algorithm to convert a solution to NPC to a solution to Hitting Set.