CS 3510 — Algorithms, Spring 2022

Homework 4A Solutions

1 True or False

a.) If A is NP-Complete, and B reduces to A, then B is NP-Complete.

Solution. `False`. Since A is NP-Complete, it means that every problem in NP can be reduced to A. By showing that B reduces to A, we get no information about B. If we want to show B is NP-Hard, we want to reduce $A \to B$. Further, a reduction only shows NP-Hardness, not NP-Complete.

b.) If A is in P, then A reduces to 3SAT.

Solution. `True`. 3SAT is NP-Complete, which means that every problem in NP reduces to it. Since $P \subseteq NP$ and $A \in$ P, then A reduces to 3SAT.

c.) If 3SAT reduces to A, then Independent Set can also be reduced to A.

Solution. `True`. If 3SAT reduces to A. Since 3SAT is NP-Complete, then every problem in NP reduces to A. Hence, Independent Set reduces to A.

d.) If A reduces to B and B reduces to C, then A reduces to C.

Solution. `True`. If A reduces to B, we can solve A with a solver for B. Since B reduces to C, then we can construct a solver for B using the solver for C. Hence, we can solve A with a solver for B where B is constructed using a solver for C. Hence, A reduces to C if A reduces to B and B to C.

e.) Solutions to NP-Hard problems can be verified in polynomial time.

Solution. `False`. It is known to be the case there are problems in NP-Hard that are not in NP. Hence, this is false.

2 Kite

Theorem. *Kite is NP-Complete.*

Proof. To show Kite is NP-Complete, we must show Kite is NP and NP-Hard. To show Kite is NP, we must construct a verifier that can take in the original problem and a candidate solution to determine if the solution is correct. in polynomial time.

For some input subgraph of G with parameter k, we check if the subgraph has $k-1$ vertices with degree $k-1$ (vertices part of the clique connected to all other vertices of the star), one vertex with degree k (connected to all other vertices of the clique as well as the tail). We must also check there are $k-1$ vertices with degree 2 (vertices that are part of the tail) and one vertex with degree 1 (tail end). We can iterate through the subgraph to check the number of neighbors of each vertex to calculate their degree wihch takes polynomial time If we find too many vertices or too few vertices of each of the different degrees, we can return `False`. We also need to check the set S is a subset of V which can also be done in polynomial time. If both requirements are satisfied, return `True`.

Next, we must show Kite is NP-Hard. We reduce from Clique. We want to design some mapping reduction f that takes the inputs of Clique and converts them to inputs of Kite. The mapping reduction f takes input $G = (V, E), k$ where k is the size of the clique and should output another graph G', k' such that the solution to this input on the Kite problem is the same solution to the Clique problem with inputs G, k.

We construct G' by taking every vertex $v \in V$ and attaching a series of vertices connected to each other of length g. We also set $k' = k$. This takes polynomial time as we are simply created g vertices for each vertex. Creating the g vertices is $O(1)$ and there are $O(|V|)$ vertices to add to. Hence, this reduction takes polynomial time.
Proof. There is a k kite on G' if and only if there is a k clique on G.
($\implies$) A k kite on G' implies a k clique on G. If there is a k kite in G', there is also a k clique in G', but the reduction of $G \to G'$ only added a string of vertices so there must have been a k clique in the graph graph G.

($\impliedby$) A k clique on G implies a k kite on G'. Since the reduction of $G \to G'$ adds a string of k vertices onto every vertex, then the original k clique in G turns into a kite on G' as we have a k clique connected to a k long tail.

Hence, since we have shown Kite is NP-Hard and NP, Kite is NP-Complete. $\square$

3 Exact 4-SAT

Theorem. *Exact 4-SAT is NP-Complete.*

Proof. To show Exact 4-SAT is NP-Complete, we must show Exact 4-SAT is NP and NP-Hard. To show Exact 4-SAT is NP, we can build a verifier that simply takes the original boolean formula and an assignment evaluating the formula with the assignments. If the formula evaluates to true, it returns true, otherwise false.

Next, we must show Exact 4-SAT is NP-Hard. We reduce from 3-SAT. We want to design some mapping reduction f that takes the inputs of 3-SAT and converts them to inputs of Exact 4-SAT.

Since the literals must be unique, we must force any sized 3-SAT clause to be in the form of Exacg 4-SAT. Hence, we must show three different cases:

1. For each clause of form $(a \vee b \vee c)$, we create $(a \vee b \vee c \vee z) \wedge (a \vee b \vee c \vee \neg z)$.
2. For each clause of form $(a \vee b)$, we create $(a \vee b \vee y \vee z) \wedge (a \vee b \vee \neg y \vee z) \wedge (a \vee b \vee y \vee \neg z) \wedge (a \vee b \vee \neg y \vee \neg z)$
3. For each clause of form (a), we create $(a \vee x \vee y \vee z) \wedge (a \vee x \vee y \vee \neg z) \wedge (a \vee x \vee \neg y \vee z) \wedge (a \vee x \vee \neg y \vee \neg z) \wedge (a \vee \neg x \vee y \vee z) \wedge (a \vee \neg x \vee y \vee \neg z) \wedge (a \vee \neg x \vee \neg y \vee z) \wedge (a \vee \neg x \vee \neg y \vee \neg z)$

This reduction requires to create a fixed number of clauses (8 at max) for each clause. Hence, this takes $O(m)$ time where m is the number of clauses. Hence, this is a polynomial reduction. This completes the reduction, which we now must prove.

Proof. $\exists$ an assignment in F (3-SAT input) if and only if $\exists$ an assignment in F'

$(\implies)$ If there is a satisfying assignment in F, then each of the clauses has a true literal making each clause true in F making each clause in F' also true.

$(\impliedby)$ If there is a satisfying assignment in F', this must mean each of the clauses is satisifed by only any of the original literlas. If only x, y, z were used to satisfy a clause, they would not satisfy another one of the corresponding clauses as a corresponding clause relies on the negation of x, y, or z. Hence, a satisfying assignment in F' implies the existence of a satisfying assignment for F.

Hence, since have shown this is NP and NP-Hard, Exact 4-SAT is NP-Complete. $\square$