Problem 1: DPV 8.8 Exact 4-SAT

In the EXACT 4SAT problem, the input is a set of clauses, each of which is a disjunction of exactly four literals, and such that each variable occurs at most once in each clause. The goal is to find a satisfying argument, if one exists. Prove that EXACT 4SAT is NP-complete.

Solution.

First we show that EXACT 4SAT is in NP. For a formula f, given an assignment σ, we can verify in constant time if a given clause has at least one literal satisfied, and we can verify all clauses in f are satisfied by σ in $O(m)$ time.

Now we show: 3SAT $\rightarrow$ EXACT 4SAT. Suppose we have a 3SAT input formula f with n variables and m clauses. We will define an input formula f' for EXACT 4SAT. Suppose C is a clause in f and the length is 3, so say $C = (x \vee y \vee z)$. We will create a new variable w and replace C by a pair of clauses: $C' = (x \vee y \vee z \vee w) \wedge (x \vee y \vee z \vee \overline{w})$. Notice that this pair of clauses C' is satisfiable iff C is satisfiable. If C is of length 2, say $C = (x \vee y)$ then we add two new variables w_1, w_2 and we replace C by 4 clauses: $C' = (x \vee y \vee w_1 \vee w_2) \wedge (x \vee y \vee w_1 \vee \overline{w_2}) \wedge (x \vee y \vee \overline{w_1} \vee w_2) \wedge (x \vee y \vee \overline{w_1} \vee \overline{w_2})$. Once again, these 4 clauses C' are satisfiable iff C is satisfiable. Similarly, if C has length 1, then we add 3 new variables and we replace C by 8 clauses. Note that f' will have $O(m)$ clauses and $O(n)$ variables. As we have argued on a clause-by-clause basis, f is satisfiable if and only if f' is satisfiable. Moreover, given a satisfying assignment to f', we immediately get a satisfying assignment to f by just ignoring the extra variables we added. This reduction replaces each clause in f with at most 8 easily computed clauses, and so is polynomial. Therefore, EXACT 4SAT is NP-complete.

Problem 2: DPV 8.14 Clique+IS

Prove that the following problem is NP-complete: given an undirected graph $G = (V, E)$ and an integer k, return a clique of size k as well as an independent set of size k, provided both exist.

Solution.

We will call the above problem the CLIQUE-IS problem. First, we show that CLIQUE-IS is in NP. For an input G and k, given a potential clique S and independent set T, we can verify in $O(n^2)$ time that all pairs of vertices in S are connected and S is a clique, that no pairs in T are connected and hence T is an independent set, and that $|S|$ and $|T|$ are both $= k$.

Now, we will reduce a known NP-complete problem to CLIQUE-IS. Specifically, we will show that CLIQUE $\rightarrow$ CLIQUE-IS. Recall that in the CLIQUE problem, one is given a graph G and a number k, and the answer is whether the graph has a clique of size $\geq k$ or not. Note that if a graph has a clique of size $\geq k$ then it has a clique of size $= k$, so it suffices to determine if there is a clique of size $= k$ in the input graph G.

Consider an input to the CLIQUE problem with a graph G and a parameter k. We will define a graph G' to run the CLIQUE-IS problem on. To create G', we add a set I of k new vertices to G, there are no new edges added. This forms the new graph G'. Note that G' always contains an independent set I of size $= k$. Moreover, this set I is not included in any cliques in G' since there are no edges from I. Hence, G' has a clique and an independent set of size $= k$ if and only if G has a clique of size $= k$. Therefore, if we can solve the CLIQUE-IS problem on G' for parameter k then we can solve the CLIQUE problem on G for parameter k. This completes the reduction and proves that CLIQUE-IS is NP-Complete.

Problem 3: DPV 8.19 Kite

A kite is a graph on an even number of vertices, say $2n$, in which n of the vertices form a clique and the remaining n vertices are connected in a "tail" that consists of a path joined to one of the vertices of the clique. Given a graph and a goal g, the KITE problem asks for a subgraph which is a kite and which contains $2g$ nodes. Prove that KITE is NP-complete.

Solution.

First we show that KITE is in NP. For an input graph $G = (V, E)$ and a potential solution $S \subseteq V$, we check if S is a kite of size g as follows. First, check the degrees of the vertices of S within the subgraph S in order to partition S into sets K and P, where K is a clique of size g (vertices have degree $g - 1$ except for one with degree g that connects to the "tail") and P is a path of length g (vertices have degree 2 except for the end of the tail which has degree 1). This can be done in $O(n)$ time per vertex and hence $O(n^2)$ total time.

Now, we will reduce a known NP-complete problem to KITE. Specifically, we will show that CLIQUE $\to$ KITE. Suppose that $G = (V, E)$ and k is an input for Clique. To create the input for KITE, we set $g = k$ and create a new graph G' in the following way. We start with G. For every vertex $v \in V$, we create g new vertices connected by a path, and the head of the path is connected to v. Thus, in G' every vertex has a path of length g hanging off of it.

We claim that G has a clique of size k iff G' has kite of size $2g$. If G has a clique C of size k, then it is clear that in G' we have a kite of size $2g$ by using the same set C and the path hanging off any one of the vertices in C. In the other direction, given a kite in G' of size $2g$, this contains a clique S on g vertices, this clique must exist in G as well since the only vertices added into G' had degree 2. Hence, given a solution to KITE in G' there is a solution to CLIQUE in G. This completes the reduction and proves that KITE is NP-Complete.