Pset 7

Darwin Do

April 26, 2022

(a) Darwin Do

(b) 919941748

(c) Collaborators: Graham Stodolski, Raffael Davila

(d) I have followed the academic integrity and collaboration policy

(e) Hours: 5

1 Strongly Independent Set

To prove that `SIS` is NP-hard, we show that `IS` $\leq_p$ `SIS` where `IS` is the `Independent-Set` problem as discussed in lecture.

Proof. Let $G = (V, E)$ be an undirected graph that is a valid input to the `IS` problem. We will form $G' = (V', E')$ by replacing every edge $(u, v) \in E$ with two edges, (u, w) and (w, v) where w is a new node. Then, we connect all the w nodes together.

If G is `yes` on the `IS` problem, then G' will also be `yes` on the `SIS` problem with the same k. This is because any nodes in the independent set S of size $\geq k$ will have a distance between each other of at least 2 by definition. Due to the adding of the (u, w) and (w, v) edges, this distance will become at least 4. So the nodes in S are a valid strong independent set in G' with size $\geq k$. Otherwise, if the candidate set S is `no` for the `IS` problem on G, G' does not have an independent set of size $\geq k$. There will be at least 2 vertices $u, v \in S$ that are connected and have a distance of 1 between each other. When we create the (u, w) (w, v) edge pairs in G', the distance between u and v in G' will be 2 and break the `SIS` principle.

If G' is `yes` on the `SIS` problem, then G will also be `yes` on the `IS` problem with the same k. Let S be the strong independent set of size k in G'. We know that there can be no w nodes in S as all the w nodes are connected to each other so you can reach any node from a w node within 2 edges. That means we can remove the w nodes from G' and re-connect the (u, v) nodes to get G. The nodes in the strong independent set S are the same nodes in the independent set for G since none of the vertices in S have a distance of 1 between them by definition of `SIS`. Otherwise, if the candidate set S is `no` for the `SIS` problem on G', G does not have an independent set of size $\geq k$. In this case, is at least one pair of nodes $u, v \in S$ whose distance is ≤ 2. If it is 1, then we have already violated the independent set principle. If the distance is 2, then we know the path from u to v is $(u, w) \to (w, v)$ by construction. Once we remove the w node and reconnect the (u, v), they will be neighbors and violate the independent set principle.

We have shown `IS` $\leq_p$ `SIS`. Since `IS` is NP-hard, `SIS` is NP-hard as well. $\square$

Now we show that `SIS` $\in NP$.

Proof. Let S be a candidate strongly independent set for graph G. For all $v \in S$, we perform a BFS from v that only goes 2 levels from v. If we encounter any other nodes in S from our 2-level BFS, we know that this candidate is not a strongly independent set. Otherwise if complete this process for all $v \in S$ and no elements of S are in the 2-level BFS, we know this is an strongly independent set.

This modified version of BFS is faster than BFS as it only goes 2 levels. The runtime of normal BFS is $O(|V| + |E|)$. We are running this modified version of BFS on every node in S, so the upper bound runtime is $O((|V| + |E|) \cdot |V|)$

which is polynomial to the graph. Verifying a candidate takes a polynomial amount of time so `SIS` is in NP. $\square$

`SIS` is in NP and is NP-hard so it is NP-complete.

2 Dominating Set

We first show that `DS` is NP-hard by showing that `VC` $\leq_p$ `DS` where `VC` is the `Vertex-Cover` problem as discussed in lecture.

Proof. Let $G = (V, E)$ be an undirected graph that is a valid input to the `VC` problem. We will form $G' = (V', E')$ by replacing every edge $(u, v) \in E$ with two edges, (u, w) and (w, v) where w is a new node. Then we connect all the vertices $v \in V$ together with edges.

If G is `yes` on the `VC` problem then there is a vertex set S with size $\leq k$ where every edge has at least one vertex in S. This set S will also be a valid dominating set in G'. We know that every edge in G is touching at least one node in S. By construction, that means every w node will be touching a node in S. Then since we connect all the nodes $v \in V$ together, we know all $v \in V$ is either a node in S or touching a S node. Every $v \in V'$ is either in S or touching a node in S so G' is a dominating set of size $\leq k$. Otherwise, if set S is `no` for `VC` on G, there will be at least one edge $(u, v) \in E$ where u and v are not in S. In G', the w node connecting to the u, v nodes will be connected to 2 nodes not in S breaking the dominating set principle.

If G' is `yes` on the `DS` problem, then G is also `yes` on the `VC` problem. Let S be the dominating set in G' of size $\leq k$. By definition, we know that every w node is a neighbor of at least 1 node in S.

When we remove the w node and the (v, w), (w, u) edges, we connect (v, u) again to form the edge in E. Since we know that at least u or v is in S, the (v, u) edge follows the vertex-cover principle. We do this step for all edges in in G' that are connected to a w node. For edges that are not connected to a w node, (instead connected to each $v \in V$), we simply remove these from G' to get G so we don't need to worry about them. Therefore the S set is a valid vertex cover of size $\leq k$. Otherwise, if set S is `no` for `DS` on G', there will be at least one node v^* who has 0 neighbors in S. We know $v^* \notin V$ as every $v \in V$ is connected to each other so all $v \in V$ is either in S or a neighbor of S since $|S| > 0$. That means v^* is a w node with edges (u, w) and (w, v) where v and u are not in S. When deleting the w nodes to form G, the (u, v) edge will violate the vertex-cover principle and S will not be a valid vertex-cover for G.

□

Now we show `DS` $\in NP$

Proof. Let S be a candidate dominating set for graph G with size $\leq k$. For all $v \in V$, we check if either v is in S or if one of the neighbors of v is in S. If this condition fails, we know that S is not a valid dominating set. Otherwise if none of the nodes fail this check, we know S is a valid dominating set of size $\leq k$ on G.

The worst-case runtime for this is $O(|V|^2)$ as we iterate through all the nodes and look at each node's neighbors. This is polynomial time so `DS` is in NP. □

`DS` is in NP and is NP-hard so it is NP-complete.