COL351: Assignment-4

Abhinav Jain
2019CS10322
Himanshu Gaurav Singh
2019CS10358

November 13, 2021

Problem 1

Let $G = (V, E)$ be a directed graph with source s and $T = \{t_1, ..., t_k\} \subseteq V$ be a set of terminals. For any $X \subseteq E$, let $r(X)$ denote the number of vertices $v \in T$ that remains reachable from s in $G - X$. Give an $O(|T||E|)$ time algorithm to find a set X of edges that minimizes the quantity $r(X) + |X|$.

Answer Consider an auxilary graph $H(V_h, E_h)$ formed by adding a sink node t to graph $G(V, E)$ i.e $V_h = V \cup \{t\}$ and $E_h = E \cup \{\{t_1, t\}, \{t_2, t\}, ..., \{t_k, t\}\}$. We construct the flow network by assigning the capacities of all edges of H as unity. Using the Ford-Fulkerson algorithm we compute the $s - t$ max-flow (say F) and the corresponding min-cut (say (A, B)). We claim that the minimum value of the quantity $r(X) + |X|$ is equal to the max-flow in H and a set X for which the minimum is achieved is the set of edges in G from A to B.

Algorithm 1 $SOLVER(G, s, T = \{t_1, ..., t_k\})$

Construct the auxilary graph $H = (G.V \cup \{t\}, G.E \cup \{\{t_1, t\}, \{t_2, t\}, ..., \{t_k, t\}\}$
Assign capacity 1 *to each edge of* H
$F, (A, B) = Ford - Fulkerson(H, s, t)$ Ford-Fulkerson return an integer maximum flow F and a min-cut (A,B)
$X = \{\}$
for $(u, v) \in H.E$ **do**
  if $u \in A$ *and* $v \in B$ **then**
    $X \leftarrow X \cup \{(u, v)\}$
  end if
end for
return $X \cap G.E$

Proof of correctness Suppose the max-flow value is F. In that case, F vertices out of $t_1, t_2, ..t_k$ receive flow. Since the edge capacities are all integers (and all equal to 1), we can assume that the flow function for which max flow value F is achieved is such that the flow through each edge is either 0 or 1. Observe that since all edge capacities of H are unity, each edge with non-zero flow can participate in providing flow to exactly one vertex among the F terminals with non-zero flow. Thus, the existence of an $s - t$ flow with value F would imply the existence of F edge-disjoint paths from s to t. **−1**
Claim : For all possible edge-sets X, the value of the quantity $r(X) + |X|$ is greater than or equal to F.

Proof. Consider those terminals among $t_1, t_2, .., t_k$ from which the incoming flow to sink t is 1. Observe that there will be exactly F such terminals. Let T_f be the set of all those terminals.

Consider an arbitrary X. If $|X| > F$, then we are done since that would imply $r(X) + |X| > F$. Assume $|X| = t \leq F$. From **1**, there exist F edge disjoint paths from s to each vertex in T_f. Since $|X| \leq F$, out of these F paths there will be atleast $F - |X|$ paths such that no edge of the path belongs to X. The terminals in T_f corresponding to these paths will still be reachable in $G - X$. Hence, we can conclude that $r(X) \geq F - t$. Therefore, $r(X) + |X| \geq F$. □

Now we will construct a set X_0 for which the quantity $r(X_0) + |X_0|$ is equal to F. Then by above claim, it will follow that X_0 is the required set and the minimum value is F.

Claim If the set of cut-edges corresponding to the max-flow in H is X_f, then $X_0 = X_f \cap G.E$ is the required set.

Proof. The set X_f will be a subset of the edges of H, it will contain some edges that belong to G and possibly some of the edges that were added while constructing H from G(the edges to the sink t). Thus, X_f can be written as $X = X_0 \cup X_t$, where $X_t = \{(t_j, t)\ for\ some\ js\}$. Consider the corresponding cut-sets be A and B.
Claim Only those vertices t_i in T for which the edge $\{t_i, t\}$ belongs to X_t will be in set A.
Consider a vertex t_i for which $\{t_i, t\}$ belongs to X_t. Since $\{t_i, t\}$ is a cut edge t_i must be in set A. Now consider a vertex t_i for which $\{t_i, t\}$ does not belong to X_t. For the sake of contradiction assume that it belongs to set A. Since it belongs to set A there is a path from s to t_i in the residual graph. Since the edge $\{t_i, t\}$ does not belong to X_t , the flow through the edge is 0. Since the flow through the edge is 0, there will be a edge $\{t_i, t\}$ of capacity 1 in residual graph. Hence we achieved at a contradiction as there will an path from s to t via t_i in the residual graph. Therefore t_i must be in B.
Since (A, B) is a cut, observe that in the graph $H - X_f$ there will be no path from s to t. Observe that only those vertices t_i in T for which the edge $\{t_i, t\}$ belongs to X_t will be in set A. Thus only these vertices in the set T will be reachable from s in the graph $G - X_g$. Hence $r(X_g)$ will be $|X_t|$ and therefore $r(X_g) + |X_g| = |X_g| + |X_t| = |X_f|$ which is equal to F.
Hence the proof is complete.

□

Time complexity Observe that the time required to construct H from G is $O(|V| + |E| + |T|)$. The time complexity of the ford-fulkerson algorithm is $O(max - flow * (|E| + |V|))$. Observe that in our case the max flow is bounded by $|T|$ as the max incoming flow to the sink node t can is $|T|$. Since $|V| \leq |E|$, the time complexity of our algorithm is $O(|T||E|)$.

1 Problem 2

1.0.1 1

The following algorithm is a polynomial time verifier for a given instance $I = (U, A_1, A_2, .., A_m)$ of the problem and the proposed solution S.

Algorithm 2 $VERIFIER(U, A_1, ..A_m, S)$

```
for i in 1, 2, ..m do
    flag ← False
    for s ∈ S do
        if s ∈ A_i then
            flag ← True
        end if
    end for
    if not flag then
        return False
    end if
end for
return True
```

Observe that the above algorithm returns **True** if and only if for all A_i, $A_i \cap S \neq \phi$. Also, the complexity of the above algorithm is $O(|S|(|A_1| + |A_2| + .. + |A_m|))$ which is polynomial in the size of the input. Hence, the problem is in NP.

1.0.2 2

We construct a mapping from all instances of the vertex cover problem to instances of the hitting set problem as follows.
For an arbitrary instance of the vertex cover problem (G, k), consider $U = G.V$, $A(= \{A_1, A_2, ..A_m\}) = \{\{u, v\}, (u, v) \in G.E\}$.
In other words, the universe U is the set of all vertices of G and A_i is the two element subset of U corresponding to the $i-$th edge in G.

Lemma 1.1. *We claim that G has a vertex cover of size k if and only if there is a hitting set of size k corresponding to $(U, A_1, ..A_m)$ as defined above.*

Proof. Consider a vertex cover $V_0 = v_1, v_2, ..v_k$ of G. From the definition of vertex cover, for all (u, v) in $G.E$, one of either u or v lies in V_0. In other words, for all A_i as defined above, $A_i \cap V_0 \neq \phi$. Thus, V_0 is a hitting set corresponding to the input $(U, A_1, ..A_m)$.
For the converse, assume that for $(U, A_1, ..A_m)$ as defined above, there exists a hitting set $V_0 = v_1, ..v_k$. From the definition of hitting set, for all A_i, $A_i \cap V_0 \neq \phi$. Using the definition of A_i we can conclude that for all $(u, v) \in G.E$, one of u or v is in V_0. Thus, all edges have atleast one endpoint belonging to V_0. Thus, by definition, V_0 is a vertex cover of G. $\square$

Observe that $U, A_1, ..A_m$ can be constructed from G in polynomial time. Hence, there exists a polynomial time algorithm that maps instances of vertex cover to instances of hitting set. Therefore, vertex cover is polynomial time reducible to hitting set. Since, vertex cover is NP-complete, hitting set is also NP-complete.

Problem 3

1.0.3 1

The following algorithm is a polynomial time verifier for a given instance G of the problem and the proposed solution S.

Algorithm 3 $VERIFIER(G, S)$

```
Construct the induced graph H = G[G.V − S]
Perform DFS over H
if there exists any back − edge in H or |S| ≤ k then
    return False
else
    return True
end if
```

The existence of any back-edge in H would imply the existence of a cycle and hence the algorithm returns false. The absence of any back edge would imply that H is a forest and hence the algorithm returns true. Thus, the above algorithm is indeed a correct verifier. The runtime for the algorithm is $O(n+m)(O(n+m)$ for construction of H and DFS and $O(m)$ for checking any back-edge) which is polynomial in the size of the input. Thus, the problem is in NP.

1.0.4 2

To reduce the vertex-cover problem to the feedback-set problem, we construct a mapping from all instances of the vertex-cover problem to instances of the feedback-set problem as follows.
For an arbitrary instance of the vertex cover problem (G, k), consider another undirected graph H constructed as follows:

1. Begin $H \leftarrow \phi(empty\ graph)$
2. Add all edges and vertices of G to H.
3. For each edge e of G, add a new vertex v_e to H and add two edges from v_e to the endpoints of e in H.

Finally, $H.V = G.V \cup V_e$ such that $V_e = \{v_{e_1}, v_{e_2}, ..v_{e_m}\}$ where $e_1, e_2, ..e_m$ are edges of G and $H.E = G.E \cup E'$ where $E' = \bigcup_{i=1}^{m} \{(v_{e_i}, e_i.x), (v_{e_i}, e_i.y)\}$ where $e.x$ and $e.y$ are the endpoints of e.

Lemma 1.2. *We claim that G has a vertex cover of size atmost k if and only if there is a feedback set of size at most k for the graph H constructed from G as above.*

Proof. Consider a vertex cover $V_0 = \{v_0, v_1, v_2, ..v_k\}$ of G. We prove the following claim.
Claim V_0 is a feedback set for H.

Proof. We need to prove that the subgraph induced by the vertices remaining in H after removing vertices belonging to V_0(say $H'(= H[H.V - V_0])$) does not contain a cycle. The edge set of H is $G.E \cup E'$ as defined above. Consider any edge e of H that lies in $G.E$. Since V_0 is a vertex cover of G, one of the end points of e lies in V_0. Thus, H' does not contain e. Since e was chosen arbitrarily, no edge of G is present in H'. $-\mathbf{1}$
We proceed by contradiction. Consider any cycle C in H'. From **1**, no edge of G is present in H', hence all the edges of C belong to E', all which have one of the endpoints belonging to V_e as defined in the construction of H. Thus, C has at least one vertex(say v_{e_0}) which belongs to the set V_e. Consider the edge e_0 of G corresponding to v_{e_0}. Since V_0 is a vertex cover of G, one of the endpoints of e_0 must be present in V_0 and hence that endpoint will be absent in H'. This would imply that v_{e_0} is connected to atmost one vertex in H. This contradicts that it lies on C, otherwise it would have atleast two neighbours in C. Hence, there cannot be any cycle in H'. □

From the above claim, we can conclude that the existence of a vertex cover of size atmost k implies the existence of a feedback set of size atmost k in H.
For the converse, consider any feedback set F of size k for the graph H. We prove the following claim.
Claim Given F of size k, there exists a vertex cover V' of G of size atmost k.

Proof. Consider the vertices of H that belong to V_e. Each of them lies on a 3-membered cycle $\{V_e, e.x, e.y\}$ where $e.x, e.y$ are endpoints of e. Since F is a feedback set, atleast one vertex of each such cycle must lie in F. We construct V' as follows. For each such cycle (corresponding to each edge e in G), if the corresponding vertex in F is one of the endpoints of e, we add it to V', otherwise, we add one of the endpoints of e to V'(if it is already not present in V'). Observe that for each edge of G, we have a corresponding three-membered cycle as defined above. Thus, from the construction of V', we can conclude that one of the endpoints for each edge in G lies in V'. Also, since we have added at most one vertex in V' for each vertex in F, there are atmost k vertices in V'. Thus, V' is a vertex-cover of G of size atmost k.

□

From the above two claims, we can conclude that G has a vertex cover of size atmost k if and only if there is a feedback set of size at most k for the graph H constructed from G as above. □

Observe that H can be constructed from G in time linear in the number of vertices and edges of G, thus, polynomial in the size of the input. Hence, using the lemma proved above, we can conclude that there exists a mapping that reduces the vertex cover problem to the feedback set problem in poylnomial time. We know that the vertex-cover problem is NP-complete. Hence, we can conclude that the feedback-set problem is also NP-complete.