Theory Assignment-3: ADA Winter-2023

Shivoy Arora (2021420) Chehak Malhotra (2021141)

---

1. The towns and villages of the Island of Sunland are connected by an extensive rail network. Doweltown is the capital of Sunland. Due to a deadly contagious disease, recently, few casualties have been reported in the village of Tinkmoth. To prevent the disease from spreading to Doweltown, the Ministry of Railway of the Sunland wants to completely cut down the rail communication between Tinkmoth and Doweltown. For this, they wanted to put traffic blocks between pairs of rail stations that are directly connected by railway track. It means if there are two stations x and y that are directly connected by railway line, then there is no station in between x and y in that particular line. If a traffic block is put in the track directly connecting x and y, then no train can move from x to y. To minimize expense (and public notice), the authority wants to put as few traffic blocks as as possible. Note that traffic blocks cannot be put in a station, it has to be put in a rail-track that directly connects two stations.

   Formulate the above as a flow-network problem and design a polynomial-time algorithm to solve it. Give a precise justification of the running time of your algorithm.

Ans: We are **assuming** in this question that the railway track is bi-directed, i.e., if there is are two stations x, y which has a railway line in between then a train can either go from x station to y station or from y station to x station.

(a) We can formulate this problem as a undirected graph $G(V, E)$ where V is the set of vertices where each station is a vertex and E is the set of edges where each edge is a railway line between two stations with max capacity as 1.

$$V = \{x|\text{x is a station}\}$$

$$E = \{(x, y, 1)|\text{there is a railway line between x and y}\}$$

We will be representing the station of Tinkmoth as S and the station of Doweltown as T.

Now, we will convert this graph $G(V, E)$ into a flow network $F(V', E')$ with flow function $f : E'(F) \leftarrow \mathbb{Z}^+$.

We will assume the vertex S to be the source and vertex T to be the sink.

All the edges from S will be directed outwards from S and all the edges from T will be directed inwards to T. All the other edges from x to y will be replace by two edges, one directed from x to y and the other directed from y to x.

Now, we will replace each vertex x with two vertices x_{in} and x_{out}, connected by an edge from x_{in} to x_{out} with capacity ∞. All the incoming edges of x will be connected with x_{in}, all the outgoing edges of x will be connected with x_{out}.

(b) To solve the problem, we will find the max-flow of the network using the **ford-fulkerson's algorithm** but in this algorithm we will make a small change in which, while finding a path from S to T, and there are two choices to go from station x to station y, which are connect by a regular edge and a residual edge then residual edge will be given preference.

The max-flow value that we will get from the algorithm will be the minimum cut value for the problem according to the **max-flow min-cut theorem**.

In the problem the min-cut value means the minimum number of traffic blocks required to be put in the railway network to prevent the disease from spreading to T from S, i.e., from Tinkmoth to Doweltown, since the initial capacity of each edge was 1.

To find the edges on which the traffic blocks are to be put, i.e., the railway tracks that would be block, we will use **dfs** or **bfs** on the residual graph to find the edges that are reachable from T in the residual graph.

And all the edges that connect the reachable edges to the unreachable edges are in the min-cut, i.e., those railway lines would be blocked.

(c) The time complexity of the algorithm is $O(|E| \times f)$, where f is the max-flow value of the network as this algorithm essentially runs the ford-fulkerson's algorithm with a small change. And then uses dfs or bfs to find the edges the minimum s-t cut, whose time complexity is $O(|E| + |V|)$, which essentially is $O(|E|)$ as $|V| = O(|E|)$.

So overall complexity is $O(|E| \times f) + O(|E|)$, which equals to $O(|E| \times f)$.