

(1) (5 points) Design a linear-time algorithm that tests whether a given flow is a maximum flow. The input to the algorithm is a flow network G and a flow f in G ; the output should be an answer to the question, “Is f a maximum flow?” You may assume that G has integer edge capacities and that f is an integer-valued flow. Your algorithm’s running time should be $O(m + n)$, where m and n are the number of edges and vertices in G , respectively.

Solution

We can solve this problem by first building a residual graph and then checking if there is some simple path from the source S to sink T by running DFS. If such a path exists, we know that this flow is not a maximum flow since we can find an augmenting path which would increase the flow value to something higher. We will output No. If such a path does not exist, this is equivalent to the terminating condition in Ford Fulkerson algorithm which basically means that the flow given to us was indeed a maximum flow. We will output Yes.

Proof of correctness

We can show correctness by the fact that if this flow was not the maximal flow, we would definitely find a path from S to T and using that augmenting path, find a flow with a greater value. If there is no such path, this is equivalent to the terminating condition in Ford Fulkerson. Hence assuming the correctness of FF algorithm, this algorithm will work correctly.

Running Time

Building the residual graph takes $O(m + n)$ time. Searching for a path using DFS takes $O(m)$ time. Hence the total time is $O(m + n)$.