

Network Flows - Final

Do 4 of the following 6 problems. You are permitted to use the text (AMO), your notes and any material handed out in class. The exam time is 2 hours and 30 minutes. GOOD LUCK!!!

Academic integrity is expected of all students at all times, whether in the presence or absence of members of the faculty. Understanding this, I declare that I shall not give, use, or receive unauthorized aid in this examination.

Name (PRINT CLEARLY), and Signature:

Problems to be graded:

1). (a). A FEEDBACK EDGE SET of an undirected graph $G = (N, E)$ is a subset of edges $E' \subseteq E$ whose removal results in an acyclic graph $G' = (N, E \setminus E')$. Given a graph with positive edge weights $w_{ij} > 0$, describe a polynomial time algorithm that finds a minimum weight feedback edge set. Recall, this problem is NP Complete for directed graphs.

(b). For a graph $G = (N, E)$ define $conn(G)$ the number of connected components of G . Prove that $conn(G) + m \geq n$ for every graph G . $|N| = n$, and $|E| = m$.

Comment: parts (a) and (b) are not related.

2). We say that an undirected graph has a *strongly connected orientation* if its edges can be directed so that the resulting graph is strongly connected (meaning that for every two nodes i, j there is a directed path from i to j and from j to i). A *bridge* in a graph G is an edge whose removal disconnects G .

Prove: A graph $G = (N, E)$ has a strongly connected orientation if and only if G is connected and has no bridge. (Ideally, your proof should be constructive. It should describe an efficient algorithm that gives a strongly connected orientation, if one exists).

3). Let $G = (N, R, B)$ be a connected (multi) graph, where N are the nodes, R is a set of red edges and B is a set of blue edges. Suppose that for every node $i \in N$ we have that $deg_R(i) = deg_B(i)$, where $deg_R(i)$ is the degree of node i in $G(R) = (N, R)$ (and $deg_B(i)$ is the degree of node i in $G(B) = (N, B)$).

(a). Prove that G has an Euler cycle.

(b). Prove that G has an Euler cycle in which the edges of R and B alternate (i.e., when traversing the cycle, we walk along a red edge then blue then red etc.).

4). We are given a directed graph $D = (N, A)$ with (nonnegative) upper bounds on the flow and nodes $s, t \in N$. An arc is *upward critical* if increasing the capacity on this arc (strictly) increases the max flow from s to t . An arc is *downward critical* if decreasing the capacity on this arc (strictly) decreases the max flow from s to t .

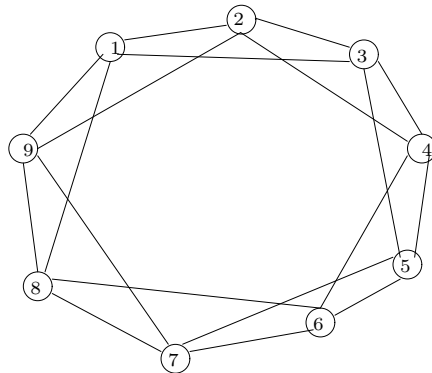
(a). Does every network have an upward critical arc?

- (b). Describe an algorithm to identify all upward critical arcs. Your algorithm should be faster than solving m max flow problems!
- (c). Does every network have an downward critical arc?
- (d). Is the set of upward critical arcs the same as the set of downward critical arcs?

5). For each of the following prove or give a counterexample:

- (a). Given a graph G with distinct edge costs, the edge of minimum cost in some cycle must be part of the minimum spanning tree.
- (b). Given a graph G with distinct edge costs, the shortest path between s and t is unique.
- (c). Given a directed graph G with distinct capacities, the minimum cut (S, \bar{S}) is unique.
- (d). Given a directed graph G with nonnegative arc lengths, and a node s . The tree of shortest paths from s remains the same if a constant $B > 0$ is added to the lengths of all arcs (s, i) (for all nodes i).
- (e). Given a directed graph G with nonnegative arc lengths, and a node s . The tree of shortest paths from s remains the same if for some node $j \neq s$, and some constant $B > 0$ is added to the lengths of all arcs (j, i) (for all nodes i).
- (f). Given a min cost flow problem on directed graph G with nonnegative arc costs, and capacities, the min cost flow x^* remains the same if a constant B is added to all edge costs.

6). Given a Hamilton tour on a set of n nodes, $\{(v_i, v_{i+1}) \mid i = 1, \dots, n\}$, The *double tour* is the set of edges of the tour *and* their shortcuts $\{(v_i, v_{i+2}) \mid i = 1, \dots, n\}$ (all indices are modulo n). See the example below for 9 nodes.



The DOUBLE TOUR PROBLEM (DTSP) is: Given a graph G with nonnegative costs on the edges, find a Double Tour of minimum cost. We assume our graph is complete, undirected and that the costs satisfy the triangle inequality.

Describe an approximation algorithm for the DOUBLE TOUR PROBLEM. You should clearly describe an algorithm that runs in polynomial time, then prove that the Double Tour has cost at most some constant times the cost of an optimal Double Tour, for all instances of the problem. You may use any of the approximations of TSP discussed in class.