Davie Truong

I have read and agree to the collaboration policy. Davie Truong

Homework Heavy

CMPS 102 – Spring 2017 – Homework 3

Solution to problem 1

The algorithm consists of a step in which we reduce the current graph g into a graph g’ that can run Ford-Fulkerson. This will return the path with the maximum flow.

Constructing the algorithm (create graph g’ and run Ford-Fulkerson)

Turn every node in g into two separate nodes Vin and Vout, the edge being the node capacity Cv >= 0. The edges connected in g, connects the corresponding (Vin, Vout) pairs, which will have an edge of infinite capacity. The resulting graph g’, now has the correct format to run Ford-Fulkerson.

Proof of Correctness:

Any flow in g’ entering any node must go through the edges of (Vin, Vout) and it cannot exceed Cv. The remaining edges have an infinite capacity, meaning there can’t be a flow bottleneck problem. Ford- Fulkerson was proved in class. The max flow min cut theorem hold because Cv is the max flow of every node/edge, thus there exist a cut whose capacity will equal the flow.

Runtime:

The algorithm runs in O(mnC) time because it needs to go through every edge, vertex, and C amount of augmentations.

Theorem: The algorithm terminates in at most v(f\*) <= nC iterations.

Space Complexity:

O(V+E) to hold the adjacency list representation

The analogue of the s-t cut in a node-capacitated network can be discovered by running the algorithm. It will return the equivalent path for the node capacitated network. The capacity of our object is the max Cv flow value from the source to the connecting cities.