

Flow Report

Daniel Odenbrand

March 31, 2016

Results

My implementation successfully computes a flow of 163 on the input file.

The bottleneck becomes

28-30, 29-30, 31-41, 39-44, 39-45, 39-46, 40-41, 40-44, 38-46

Implementation details

I use a straightforward implementation of Ford-Fulkerson's flow algorithm as described in Tardos, Klein, *Algorithm Design*, chap. 7.1.

I use DFS to find an augmenting path.

The running time is $O(mf)$ where f is maximum flow since every edge can be used to push 1 flow every time we augment.

I used a graph containing Nodes and Edges. All edges are directed and to represent the undirected graph an edge is added in both directions between two nodes. The capacity is set on the edge and altered whenever flow is pushed (in both directions, thus remaking the start graph into the residual graph). The Nodes keep a vector of edges while each edge has a start-node and an end-node (in addition to its capacity). All objects are pointers to avoid copying and storing large amounts of data.