Name: Daniel Kim
ID: 102353420

CSCI 3104, Algorithms **Profs. Chen & Grochow**
Problem Set 8 – Due Thurs Apr 2 11:55pm **Spring 2020, CU-Boulder**

Advice 1: For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.

Advice 2: Informal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

Instructions for submitting your solutions:

- All submissions must be typed.
- You should submit your work through the **class Canvas page** only.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please allot at least as many pages per problem (or subproblem) as are allotted in this template.

Quicklinks: 1a 1b 2a 2b

Name: Daniel Kim
ID: 102353420

1. *Consider the following flow network, with each edge labeled by its capacity:*

`Network.pdf`

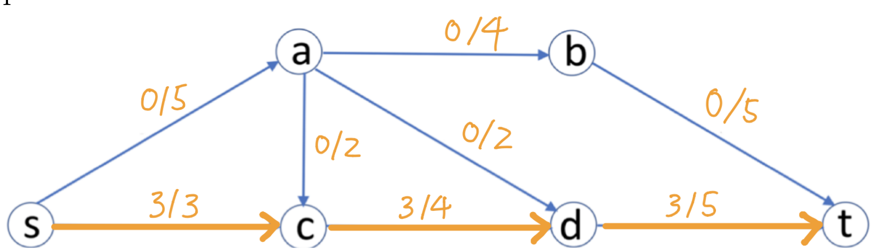
(a) Using the Ford-Fulkerson algorithm, compute the maximum flow that can be pushed from $s \to t$. **You must use** $s \to a \to d \to t$ **as your first flow-augmenting path.**

In order to be eligible for full credit, you must include the following:

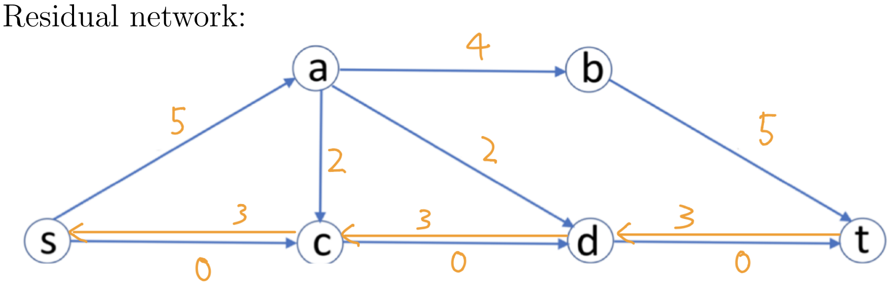
- The residual network for each iteration, including the residual capacity of each edge.
- The flow augmenting path for each iteration, including the amount of flow that is pushed through this path from $s \to t$.
- The updated flow network **after each iteration**, with flows for each directed edge clearly labeled.
- The maximum flow being pushed from $s \to t$ after the termination of the Ford-Fulkerson algorithm.

$s \to c \to d \to t$. is my first augmenting path. The bottleneck capacity of this path will be $min\{sc, cd, dt\} = 3$. Therefore, if we send 3 unit of flow through this path, the flow augments the path.

Updated flow network:

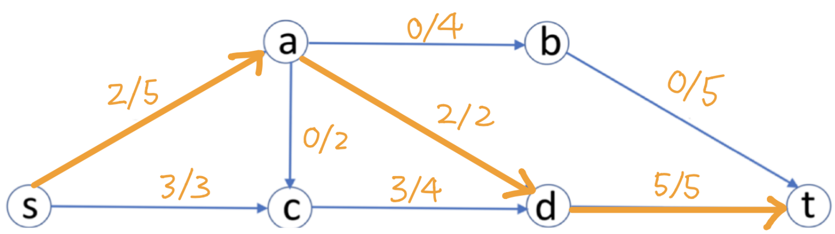


Residual network:

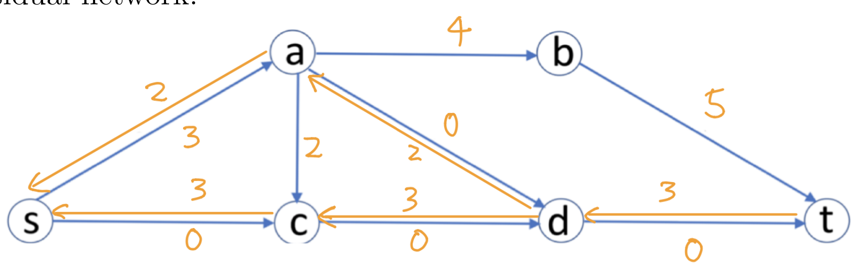


$s \to a \to d \to t$ is my second augmenting path. The bottleneck capacity of this path will be $min\{sa, ad, dt\} = 2$. Therefore, if we send 2 unit of flow through this path, the flow augments the path.

Updated flow network:

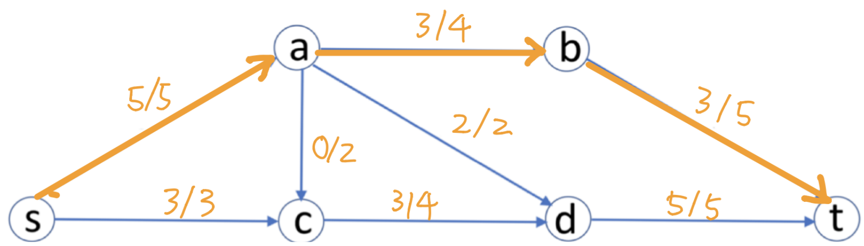


Residual network:

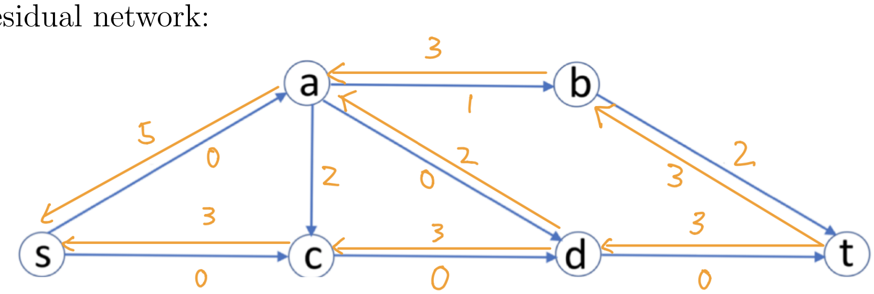


$s \to a \to b \to t$ is my third augmenting path. The bottleneck capacity of this path will be $min\{sa, ab, bt\} = 3$. Therefore, if we send 3 unit of flow through this path, the flow augments the path.

Updated flow network:



Residual network:



These full forward edges (sa, sc) will block all paths from s to t. Therefore, using augmenting path theorem, the current flow 8 is the max flow and there will be no more augmenting paths.

(b) The Ford-Fulkerson algorithm will terminate when there is no longer an augmenting path on the residual network. At this point, you can find a minimum capacity cut. Indicate this cut and its capacity, and verify if max flow min cut theorem holds.

First, let's say we partition the vertices into two disjoint sets. One of them being only the source node, and the other one being the rest of the nodes. Next, the flow from $s \to t$ is the flow of the cut. So $5 + 3 = 8$ will be the capacity. Therefore, the max flow min cut theorem holds.

2. In this problem, we seek to generalize the max-flow problem from class to allow for multiple sources and sinks. Given a directed graph $G = (V, E)$ with capacity $c(u, v) > 0$ for each edge $(u, v) \in E$ and demand $r(v)$ at each vertex $v \in V$, a routing of flow is a function f such that

   - **(capacity contraint)** for all $(u, v) \in E$, $0 \leq f(u, v) \leq c(u, v)$, and
   - **(flow conservation)** for all $v \in V$,

   $$\sum_{u:(u,v)\in E} f(u, v) - \sum_{u:(v,u)\in E} f(v, u) = r(v),$$

   i.e., the total incoming flow minus the total outgoing flow at vertex v is equal to $r(v)$.

   We note that $r(v)$ can be positive, negative, or 0, just as in the max-flow setting from class. In particular, note that:

   - The vertex v is a **source** vertex precisely if $r(v) < 0$.
   - The vertex v is a **sink** vertex precisely if $r(v) > 0$.
   - The vertex v is neither a source nor a sink precisely if $r(v) = 0$.

   These conditions are the same as in the single-source, single-sink max-flow model from class.

   (a) Show how to find a routing or determine that one does not exist by reducing to a maximum flow problem. In other words, given a graph, c and r as above, construct a related graph H with capacities $c' \colon E(H) \to \mathbb{R}$ and $s, t \in V(H)$ such that you can use the maximum $s \to t$ flow on H to determine whether a routing exists in G. Hint: You may want to take a look at Chapter 7.5 of the recommended text by Kleinberg and Tardos to get some insight.

First we need to add a source s and to every source nodes, you insert a directed edge from s whose demand $r(v)$ that are less than zero. For capacity from inserted edges, we set it equal to the absolute value of demand $r(v)$ of the source nodes. Next we need to add a new sink t with a directed edge added to every sink nodes whose $r(v)$ is non negative (bigger than 0). For capacity, we set it equal to the value of demand $r(v)$. To answer the question, in order to find a routing in the original graph, we need to let the source s bring as much flow for the multiple sources. In addition, we need to let sink t absorb as much flow for the multiple sinks.

In the new graph, the capacity constraint remains the same and except for sourse s and sink t, each source and sink should satisfy the "flow in equals flow out". $f_i = \sum_{(v)\in V} f(s_i, v) - \sum_{(v)\in V} f(v, s_i)$ is the value of a flow in the original graph. To add on, $f(s, s_i) = f_i$ satisfies the capacity constraint and flow conservation because of the new single-source flow network. $f_i : (f_1 + f_2 + ... + f_i)$ is the flow for the multiple-source flow network. Because there are no other edges coming into s in the single-source network, the flow is $\sum_{i=1} f(s, s_i)$. In addition, because $f(s, s_i)$ is equal to f_i and positive, they are equivalent.

(b) Suppose that additionally you are given a lower bound $l(u, v) > 0$ at each edge (u, v), and we are looking for a routing f satisfying

- **(lower bounds)** for all $(u, v) \in E$, $f(u, v) \geq l(u, v)$,

in addition to the **capacity constraint** and the **flow conservation**. Show how to find such a routing or determine that one does not exist by reducing to a maximum flow problem. Hint: You may want to solve this problem by reducing it to (a). Think about how to modify the graph and demands to equivalently enforce lower bounds $l(u, v)$ on the flows.

To reduce to part a), we can modify the graph. Between every edge (u, v), we add vertices a and b. In other words, the edges from u to v will become $(u, a), (a, b)$, and (b, v). To modify the demands, we set $r(a) = l(u, v)$ and $r(b) = -l(u, v)$. By doing this, the capacity of each new edge will match the capacity of original (u, v). In addition, because we set $r(a) = l(u, v)$, we can see if the lower bound satisfy the flow through (u, v). Finally, we can re-modify into the new graph as part a). The routing exists if the demands of all of the vertices are executed.

COLLABORATED WITH:
RUIJIANG MA