CMPS 102 — Fall 2018 — Homework 4

*"I have read and agree to the collaboration policy." - **Kevin Wang***

Collaborators: *None*

Solution to Problem 4: One Edge, One Unit

Given:

- a directed graph $G = (V, E)$ with n nodes and m edges
- a capacity $c_e \in \mathbb{Z}^+$ on each edge $e \in E$
- source $s \in V$ and sink $t \in V$

... assume there is a maximum acyclic s–t flow, f, in G.

Let H be a modified graph of G, where a specific edge $e' \in E$ is picked and its capacity $c_{e'}$ reduced by 1 unit. Find the max flow of H.

Observation: If we decrease $c_{e'}$ by 1 unit, then the max flow of H is either the same as f or also decreased by 1 unit.

> **Case 1:** The flow of edge e was smaller than the capacity of edge e: $f_e < c_e$.
>
> > The max flow in G, f, is also the max flow in H because $f_{e'} \leq c_{e'}$ when $c_{e'} = c_e - 1$.
>
> **Case 2:** The flow of edge e was equal to the capacity of edge e: $f_e = c_e$.
>
> > By reducing $c_{e'}$, the flow becomes invalid as $f_{e'} > c_{e'}$ when $c_{e'} = c_e - 1$.

To fix the invalidity, we reduce $f_{e'}$ by 1 unit as well. Then find a simple u–s path and a simple t–v path and route 1 unit of flow along the path t–v–u–s to balance the in-flow and out-flow of e'. The new flow is 1 unit less than f and is in H as well.
Observation: An augumenting path from s–t has a flow of at most 1 unit.

Finally, we check if there are any augmenting paths from s–t using **Ford-Fulkerson**.

> **Case 1:** There is an augmenting path, p, with max flow $f_p = 1$
>
> > We can increase the flow of the edges by 1 unit, resulting in the same max-flow as G.
>
> **Case 2:** No augmenting path exists.
>
> > The max flow of H is less than f by 1 unit.

Algorithm 1 Finds the max flow f' of H

MAX-FLOW:
if $f_e < c_e$ **then**
  Return f
else
  [If: $f_e = c_e$*]*
  Initialize the residual network $H_{f'}$
  Search for augumenting path p (Ford-Fulkerson)
  if $p \neq$ null **then**
    Return $f' + f_p$
  else
    Return f'
  end if
end if

Time Complexity: $O(m + n)$
Initializing the residual graph takes time $O(m + n)$. Ford Fulkerson runs in time $O(C \cdot (m + n))$ where C is the capacity of the cut from the source node. Total run time is $O((C + 1) \cdot (m + n))$.

Space Complexity: $O(m + n)$
Typical space complexity for a graph is $O(m + n)$.