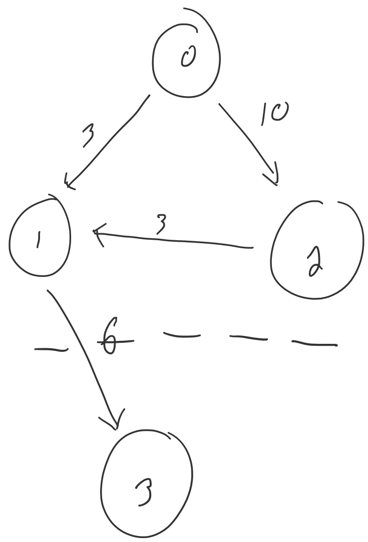
CS/ECE 473 Fall 2020 | Ryan Prendergast (ryanp4)
Homework 6 Problem 3 | Noah Watson (nwatson3)
Lawson Probasco (lawsonp2)

Let $G = (V, E)$ be a flow network with integer edge capacities. We have seen algorithms that compute a minimum $s - t$ cut via maximum flow. For the problem below assume that you only have black box access to an algorithm that given G and nodes s, t outputs a minimum cut between s and t.

(a) Describe a simple example of a flow-network G and two nodes s, t such that there are two distinct $s - t$ minimum cuts with the same capacity but different number of edges in the cuts.

Solution: On the next page, we show two minimum (s,t) cuts of a graph with a different number of edges. We define s to be 0 and t to be 3. ■

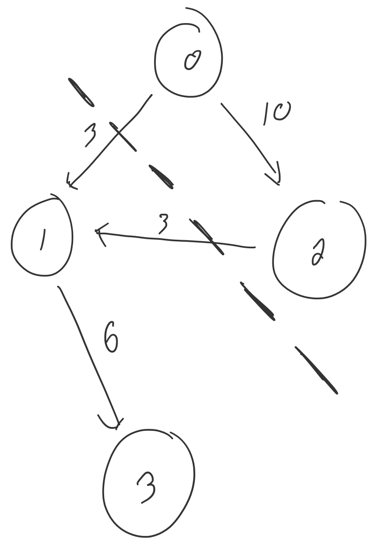


$S = \{0, 1, 2\}$

$T = \{3\}$

Capacity = 6

Edges = 1



$S = \{0, 2\}$

$T = \{1, 3\}$

Capacity = 6

Edges = 2

CS/ECE 473 Fall 2020
Homework 6 Problem 3

Ryan Prendergast (ryanp4)
Noah Watson (nwatson3)
Lawson Probasco (lawsonp2)

(b) Given G and s, t and an integer k describe an algorithm that checks whether G has a minimum cut with at most k edges.

Solution: We used as inspiration for our solution the modifications described here: https://cs.stackexchange.com/questions/115159/minimum-cut-with-minimum-number-of-edges.

We break up this problem into three parts.

- First, we modify the flow network G into G' such that the minimum cut in G' is equivalent to the minimum cut in G with the minimum number of edges. We define $G' = (V', E')$ as a flow network with the same vertices as G, $V' = V$, and the same edges $E' = E$. We then modify the edge capacities $C'(e)$ as follows:

  $$C'(e) = C(e) * (|E| + 1) + 1$$

  We note that a cut in G of capacity m corrpesponds to a cut in G' with capacity $m * (|E| + 1) + l$ where l is the number of edges in the cut.

  Any min cut in G' will be a min cut in G. Assume we have a min cut in G of size m, with corrpesponding size in G' of $m*(|E|+1)+l$. If you take any cut capacity in G of size $m+1$ (and all larger capacities must be at least this large since all capacities are integers), the cut capacity in G' will be $(m+1)*(E+1)+l'$. $l' \leq |E|$, so this new capacity will be larger than the cut capacity corrpesonding to the min cut in G, $m*(|E|+1)+l$.

  We note further that any min cut in G' will be a min cut in G with the fewest possible number of cut edges. We defined our capacity function such that the cut capacity in G' increases by 1 for every edge it cuts, thus a cut with fewer edges will have a smaller capacity.

- Second, we use our black box algorithm to find a min cut in G'. The cut it provides gives us a minimum cut in G with the minimum number of edges in it. Call this number of edges l.

- Finally, we check to see if G has a min cut with at most k edges. If $l \leq k$, then we return True. Otherwise, we return False.

Analysis We construct the graph G' in $O(V + E)$ time. We call the runtime of the black box algorithm BB, where BB is a O function according to V and E. We run the black box algorithm once, and our third step runs in constant time, so we say the whole routine runs in $O(V + E + BB)$ time.

■