Advanced Algorithms and Data structures
Assignment One

Magnus Goltermann (xzb187), Thomas Busk-Jepsen (tnr653)
Mia Rahlff Pedersen(bvx284)

December 4, 2024

1 26.1-1*

Since the two vertices that are being split still have the same in/out flow, the total max flow will remain the same.
To show that splitting an edge in a flow network will yeild the same network we will demonstrate that for any flow f in G there is a corresponding flow f' in G'
If we consider and define that in G the edge (u, v) have capacity $c(u, v)$. Then we define the two new edges as a vertex is placed in the middle. Namely (u, x) with capacity $c(u, v)$ and (x, v) also with capacity $c(u, v)$.
We can then define the flow in the network as $f'(u, x) = f(u, v)$ and $f'(x, v) = f(u, v)$. And all the other flows remain the same, thus meaning $f' = f$ for all other flows.
Thereby the network will be the same as it still both the capacity and the flow constraints are followed and the in and outflows are the same.

2 26.1-4*

We need to show that both the capacity constraint and the flow conservation are kept. First, we can show that the capacity constraint is kept by

$$0 \leq \alpha f_1(u, v) + (1 - \alpha) f_2(u, v) \leq \alpha c(u, v) + (1 - \alpha) c(u, v) = c(u, v) \quad (1)$$

Then we show the the flow conservation by summing up all the flows

$$\sum_{v \in V} (\alpha f_1(v, u) + (1 - \alpha) f_2(v, u)) = \sum_{v \in V} (\alpha f_1(u, v) + (1 - \alpha) f_2(u, v)) \quad (2)$$

We need to show that both the capacity constraint and the flow conservation are kept. First, we can show that the capacity constraint is kept by

$$0 \leq \alpha f_1(u, v) + (1 - \alpha) f_2(u, v) \leq \alpha c(u, v) + (1 - \alpha) c(u, v) = c(u, v) \quad (3)$$

Then we show the the flow conservation by summing up all the flows

$$\sum_{v \in V} (\alpha f_1(v,u) + (1-\alpha) f_2(v,u)) = \sum_{v \in V} (\alpha f_1(u,v) + (1-\alpha) f_2(u,v)) \tag{4}$$

Breaking up the sums for the two flows we have

$$\alpha \sum_{v \in V} f_1(v,u) + (1-\alpha) \sum_{v \in V} f_2(v,u) = \alpha \sum_{v \in V} f_1(u,v) + (1-\alpha) \sum_{v \in V} f_2(u,v) \tag{5}$$

And since both flows have flow conservation, the linear combination of them must also hold.

3 26.1-7

If we have a network with vertex capacity, we can still have the same maximum flow. This we can do if we for each edge we add a new vertex where the vertex constraint on that vertex is the same as the edge constraint on the edge it replaced. This means that we will have $|V| + |E|$ vertexes and $2|E|$ edges.

4 26.2-2

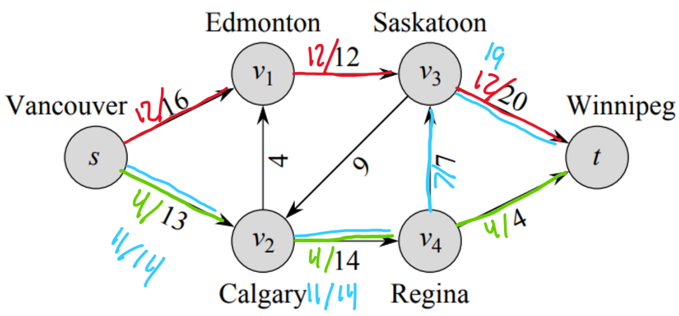
The flow across the cut is the flow there is right now crossing the cut, therefore we get the flow as 11+1-4+7+4=19.
The capacity of the cut is the total amount of flow possible between the two parts given by the cut, we therefore get 16+4+7+4=31

5 26.2-4

The minimum cut corresponding to the maximum flow is $S = \{s, v1, v2, v4\}$ and $T = \{t, v3\}$. The augmentation that cancels flow is c) as here we use a back edge to move flow.

6 26.2-3

To run the execution of the Edmonds-Karp algorithm we will do each augmentation with a breadth-first search and take the augmentation that reaches the sink first. Our run of the algorithm can be seen below:



(a)

To begin with, we run a breadth-first search this is illustrated as the red line, here we can add 12 to the flow. Then we run breath first again as the green, here we can increase the flow by 4. Finally, we can increase the flow by 7 seen as the blue line. When we run breath first search the last run then we cannot add anything to the flow as the capacity is filled.

7 26.2-7*

To see that f_p is a flow we need to prove that it complies with both the capacity and the flow constraints.
First we know that the flow in the augmented path complies with the capacity constraints as it is defined as the minimum residual capacity along the path, as this holds for every edge on the path we know we will still comply with the capacity constraint, thus we have still always have

$$c(u,v) \geq c_f(p) > 0 \tag{6}$$

In regards to the flow constraint then we know that the only vertexes that are given flow are the ones along the path, and that flow will go directly to the sink as it is an augmenting path.

To see that f_p is a flow we need to prove that it complies with both the capacity and the flow constraints.
First we know that the flow in the augmented path complies with the capacity constraints as it is defined as the minimum residual capacity along the path, as this holds for every edge on the path we know we will still comply with the capacity constraint, thus we have still always have

$$c(u,v) \geq c_f(p) > 0 \tag{7}$$

To demonstrate that f_p satisfies the flow constraint we will look at two cases, vertexes on the path p and vertices not on the path.
If we consider a vertex that is not on the path $u \notin p$ the flow $f_p(u,v)$ is zero for all edges as the flow is zero for any edges not on the path. Therefore the flow

conservation is trivially satisfied for all vertexes not on p
For vertices on the augmenting path we will consider the vertex u on the path p. By definition the augmenting path f_p assigns the same flow $c_f(p)$ to all edges on p. For u this means that the flow entering through the precising edge equals the flow exiting though the next edges. Meaning that the flow into u can be represented by:

$$\sum_{v \in V} f_p(v,u) = c_f(p).$$

And the flow out of the vertex u can be described as:

$$\sum_{v \in V} f_p(u,v) = c_f(p),$$

This means that the flow into u subtracted with the flow out of u, so the net flow can be described as:

$$\sum_{v \in V} f_p(u,v) - \sum_{v \in V} f_p(v,u) = c_f(p) - c_f(p) = 0.$$

8 26.2-9*

The flow conservation property will still hold, since we still just flow more both in and out, but the same into and out of every vertex. When looking at the capacity constraint, we see that the capacity of a given vertex can be exceeded, since both f and f' can flow up to $c(u,v)$ on an edge, and thus will $f \uparrow f'$ flow up to $2c(u,v)$ which does not hold the capacity constraint. The computed flow $f \uparrow f'$ is the pointwise sum of f and f', meaning

$$(f \uparrow f')\,(u,v) = f(u,v) + f'(u,v)$$

As for the flow conservation property, it states

$$\forall v \in V \backslash \{s,t\}; \sum_{U \in v} f(u,v) = \sum_{w \in V} f(v,w)$$

Under the assumption that both f and f' satisfy this property, adding these, we have

$$\sum_{(u,v) \in E} (f(u,v) + f'(u,v)) = \sum_{(v,w) \in E} (f(v,w) + f'(v,w))$$

Hence $f \uparrow f'$ is satisfied.

As for the capacity constraint

$$\forall u,v \in V : 0 \leq f(u,v) \leq c(u,v)$$

the augmented flow will be as follows

$$(f \uparrow f')\,(u,v) = f(u,v) + f'(u,v) = c(u,v) + c(u,v) = 2c(u,v)$$

Which exceeds the capacity, violating the constraint.

9 26.3-2

We start with the base case, where f is set to 0, therefor we have $u, v, (fu, v) = 0$, which is an integer. For the induction hypothesis we have that, after n iterations the flow on every edge is an integer and the total flow is an integer. For the inductive step, we have that after $k+1$ iterations, the residual capacity is set to the minimum, which by our induction hypothesis is an integer. In our while loop we augment the flow for both forward and backward edges. Since the residual capacity is an integer, and previous flow values are integers, the updated flow values remain integers.