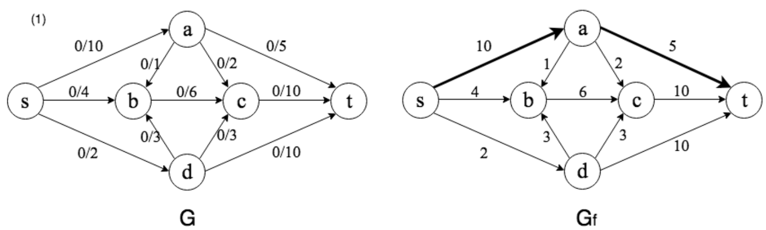
HW3 - Theoretical Part

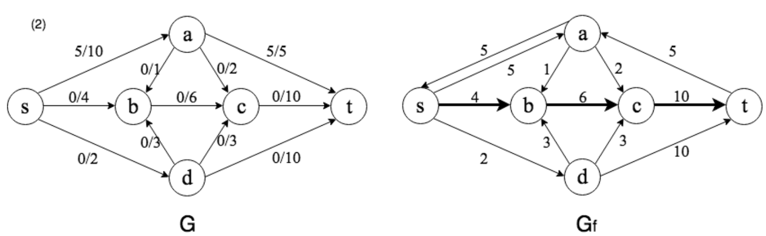
Xinyuan Cao
Uni: xc2461

November 9, 2018

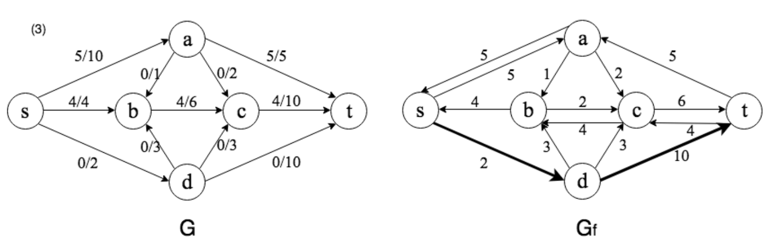
1. I'll run the Ford-Fulkerson algorithm as follows:



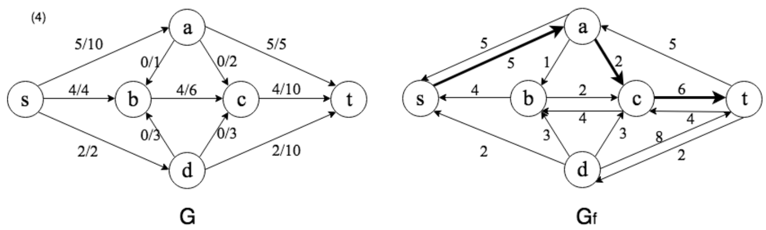
Choose $s-a-t$, $C(P)=5$.



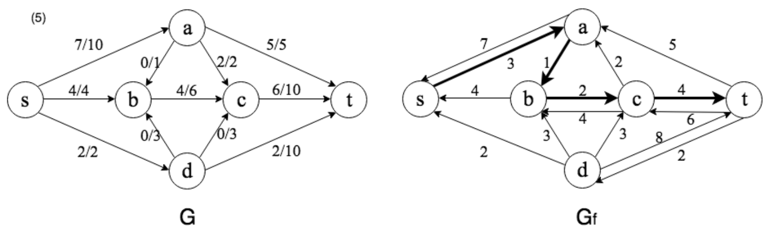
Choose $s-b-c-t$, $C(P)=4$.



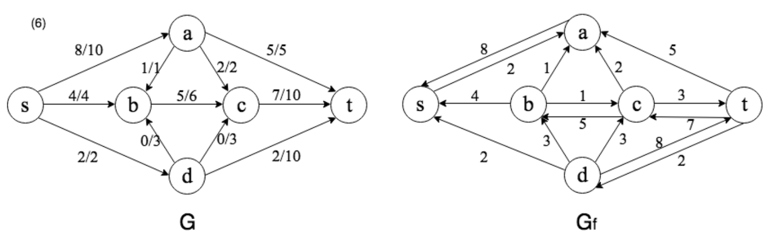
Choose $s-d-t$, $C(P)=2$.



Choose $s - a - c - t$, $C(P) = 2$.



Choose $s - a - b - c - t$, $C(P) = 1$.



There is no $s - t$ path in G_f.

So we have the maximum flow of the graph is $8 + 4 + 2 = 14$. The flow reaching the maximum is shown in graph G in step(6).

The $s - t$ cut (S, T) reaches the minimum, where $S = \{s, a\}$, $T = \{b, c, d, t\}$ and $C(S, T) = 14$. #

2. (a) If f is a feasible circulation. For $\forall v \in V$, from *demand constraints* we know

$$d_v = f^{in}(v) - f^{out}(v) = \sum_{u:(u,v)\in E} f((u,v)) - \sum_{x:(v,x)\in E} f((v,x))$$

Sum up all the vertices.

$$\begin{aligned}\sum_{v\in V} d_v =& \sum_{v\in V}\sum_{u:(u,v)\in E} f((u,v)) - \sum_{v\in V}\sum_{x:(v,x)\in E} f((v,x)) \\ =& \sum_{e\in E} f(e) - \sum_{e\in E} f(e) \\ =& 0\end{aligned}$$

So we have a necessary condition for a feasible circulation with demands to exist that the sum of all nodes' demand is equal to 0. That is, $\sum_{v\in V} d_v = 0$.

(b) Firstly we want to derive a *flow network* G' given a *flow network with demands* $G = (V, E, c)$ as follows:
- Add a source s.
- Add a sink t.
- Add (s, x) edges for all $x \in S$, and assign the capacity $c_{(s,x)} = -d_x$.
- Add (y, t) edges for all $y \in T$, and assign the capacity $c_{(y,t)} = d_y$.

So we get the derived flow network G'.

Claim 1 *The derived network $G' = (V', E', c')$ is a flow network.*

Proof The original network G is a flow network, so the original part $G = (V, E, c)$ satisfies all the requirement of flow network.

i. G is a directed graph, and the edges we add also has direction. So G' is also a directed graph.
ii. For each edge $e \in E$, it has a non-negative integer capacity. The edges we add to G' has the capacity of $|d_v|$. Given d_v is an integer, we have all the edges in G' have non-negative integer capacities.
iii. The node s is a source as there are only edges out of s. And t is a sink as there are only edges into t.
iv. For an edge $(u, v) \in E'$. If $(u, v) \in E$, then $(v, u) \notin E$. Also only add edges connecting to either s or t, so $(v, u) \notin E'$. For edge $(s, x) \in E'$, there is no edge entering s. So $(x, s) \notin E'$. Similarly for $(y, t) \in E'$, we have $(t, y) \notin E'$.
v. For each node $v \in V' - \{s, t\} = V$. $\exists x \in S, y \in T$ s.t. v on the path $x - v - y$. (x and y can be the same as v, then the path $x - v$ or $v - y$ becomes a node). So v is on the path $s - x - v - y - t$.
Hence we know the network we construct is a flow network. #

Claim 2 *The reduction is efficient.*

Proof The time complexity of reduction is $O(n)$, so it's linear time. Therefore we know the reduction is efficient. #

So from here we have efficiently transfer the problem of finding a feasible circulation with demands to the problem of solving max flow. And Claim 3 shows the equivalence of the problems.

Claim 3 *Finding a feasible circulation with demands is equivalent to solving Max Flow.*

To prove this we need to prove two things:
1. For any feasible circulation in G, we can construct a flow f' in G' with value equals to $\sum_{x\in S} -d_x$;
2. Given a max flow f' in G'. If its values equals to $\sum_{x\in S} -d_x$, then we can construct a feasible circulation in G, otherwise there is no feasible circulation in G.

Proof [$\Rightarrow$] For any feasible circulation with demands f in G. Construct an integral flow f' as follow: For edge $e \in E$, $f'(e) = f(e)$. For edge $e = (s, x)$ where $x \in S$, $f'(e) = -d_x$. For edge $e = (y, t)$ where $y \in T$, $f'(e) = d_y$.
The edge $e \in E$ naturally meets the capacity constraints. For other edges e we have $f'(e) = c_e$, also satisfies the capacity constraints.
For each $v \in V' - \{s, t\} = V$. From the demand constraints we have $f^{in}(v) - f^{out}(v) = d_v$. So if $v \in V - S \bigcup T$, we have $d_v = 0$. Therefore $f^{in}(v) = f^{out}(v)$. And there is no edges added connecting it, so $f'^{in}(v) = f'^{out}(v)$. If $v \in S$, we only add an edge (s, v) connecting it. So we have $f'^{in}(v) = f^{in}(v) + f'((s, v)) = f^{in}(v) - d_v = f^{out}(v) + d_v - d_v = f^{out}(v) = f'^{out}(v)$. If $v \in T$, we only add (v, t) connecting it. Hence we get $f'^{out}(v) = f^{out}(v) + f((v, e)) = f^{in}(v) - d_v + d_v = f^{in}(v) = f'^{in}(v)$. So we know f' satisfies flow conservation constraints.
Then we have $|f'| = \sum_{\text{e out of s}} f'(e) = -\sum_{x\in S} d_x$. This is a max flow because when we consider the cut $(S*, T*)$ with $S* = \{s\}$, $T* = V - \{s\}$, the capacity of $(S*, T*) = \sum_{\text{e out of s}} c_e = \sum_{x\in S}(-d_x) = -\sum_{x\in S} d_x$ which is an upper bound of $|f'|$. Therefore the f' we get is a max flow, with value $= \sum_{x\in S} -d_x$.

[$\Leftarrow$] Let f' be a max flow in G'. If the $|f'| = \sum_{x\in S} -d_x$, then we'll construct a feasible circulation in G. For $e \in E$, let $f(e) = f'(e)$. And we will check f is a circulation with demands.
For each $e \in E$, we have $0 \le f'(e) \le c_e$. So $0 \le f(e) \le c_e$. And so f satisfies the capacity constraints.
Given the necessary condition that $\sum_{v\in V} d_v = 0$, we have $\sum_{x\in S} d_x + \sum_{y\in T} d_y = 0$. So $|f'| = \sum_{x\in S} -d_x = \sum_{y\in T} d_y$. From the definition we know that $|f'| =$

$\sum_{x \in S} f'((s,x)) = \sum_{y \in T} f'((y,t))$. Because $f'((s,x)) \leq -d_x$ and $f'((y,t)) \leq d_y$. So we know that f' achieve this maximum iff $\forall x \in S$, $f'((s,x)) = -d_x$.For any $v \in S$, in G' we have $f'^{in}(v) = f'^{out}(v)$, where $f'^{in}(v) = f^{in}(v) + f'((s,v)) = f^{in}(v) - d_x$. So $f^{in}(v) - f^{out}(v) = d_v$. Similarly we can prove that demand constraints are correct for nodes $\in T$. For any $v \in V - S - T$, in G' from flow conservation we know $f'^{in}(v) = f'^{out}(v)$. So in G we have $f^{in}(v) = f^{out}(v) + 0 = f^{out}(v) + d_v$. So we know that demand constraints hold. And the function f we get is a feasible circulation with demands.

On the other hand, if the value of max flow f' in G' is not equal to $\sum_{x \in S} -d_x$, then it will be less than $\sum_{x \in S} -d_x$ (because of flow conservation constraints). Assume that there is a feasible circulation f. Then from the first half proof we know that there exists a flow g' in G' so that $|g'| = \sum_{x \in S} -d_x$. Hence $|g'| > |f'|$, causing a contradiction to f' is a max flow. So this time there is no feasible circulation. #

From claim 1,2,3 we know that we can reduce the problem of finding a feasible circulation with demands to Max Flow in polynomial time. So the time complexity of finding the feasible circulation is : O(mnU) = O(mn) (Ford-Fulkerson), or can improved to $O(m\sqrt{n})$.

3. (a) Given a network flow $G = (V, E, c)$, the source s and the sink t. we'd love to reduce the problem of max flow to a min-cost flow problem. Transfer G to the flow network with supplies $G' = (V', E', c')$ as follows:

- $\forall v \in V$, set $s_v = 0$.
- $\forall e = (i, j) \in E$, set the cost $a_{ij} = 0$.
- Add an edge (t, s), and set its cost $a_{ts} = -1$ and its capacity $c_{ts} = \infty$.

Claim 4 *G' is a flow network with supplies.*

It can be proved easily by definition.

Then I want to prove that given a feasible flow f' in G' that minimizes the total cost of the flow, we can construct a solution of max flow in G. Let f be a $s - t$ flow in G. For each edge $e \in E$, let $f(e) = f'(e)$. Then I'll prove its a max flow in G.

First we check the capacity constraints. Since G' has the same capacity as G for the edges in E, and f' satisfies the edge capacity constraints. Naturally we have f satisfies capacity constraints.

Secondly we check the flow conservation. For all $v \in V - \{s, t\}$, we have $f'^{out}(v) - f'^{in}(v) = s_v = 0$. So $f'^{out}(v) = f'^{in}(v)$. Hence we have $f^{out}(v) = f^{in}(v)$. It shows f satisfies flow conservation constraints.

Lastly we'll show f is a max flow. Assume that we have another $s - t$ flow g in G that has larger value than f. So $\sum_{\text{e out of s}} g(e) > \sum_{\text{e out of s}} f(e)$. Then for G', define a flow g' in G' so that $g'(e) = g(e)$ for $e \in E$, $g'((t, s)) = \sum_{\text{e out of s}} g(e)$. Because g satisfies capacity constraints and flow conservation, we only need to check s and t. Here

$$g'^{out}(s) - g'^{in}(s) = \sum_{\text{e out of s}} g(e) - g'((t, s)) = 0$$

$$g'out(t) - g'^{in}(t) = g'((t, s)) - \sum_{\text{e into t}} g(e) = \sum_{\text{e out of s}} g(e) - \sum_{\text{e out of s}} g(e) = 0$$

It holds because g is an $s - t$ flow. Also the capacity of (t, s) is infinity, so the capacity constraints also hold. So g' satisfies edge capacity constraints and node supplies. Then we check the total cost of the flow. As the cost of edges in E is equal to 0, only edge (t, s) matters the total cost. The total cost of g' is

$$T(g') = \sum_{e \in E'} a_e f(e) = a_{ts} g'((t, s)) = -g'((t, s)) = -\sum_{\text{e out of s}} g(e)$$

Similarly to ensure node supplies, we have

$$f'((t, s)) = \sum_{\text{e out of s}} f(e)$$

So we can have the total cost of f' is

$$T(f') = -\sum_{\text{e out of s}} f(e)$$

Because $\sum_{\text{e out of s}} g(e) > \sum_{\text{e out of s}} f(e)$, we have $T(g') < T(f')$, which causes a contradiction to f' minimize the total cost of the flow. Therefore there is no other $s-t$ flow in G that has larger value than f. That is, f is a max flow. #

Here we efficiently reduce max flow problem to min-cost flow problem since the reduction is polynomial time.

(b) The variables are nodes and edges and our constraints are:

$$\forall v \in V, f^{out}(v) - f^{in}(v) = s_v$$

$$\forall e \in E, 0 \leq f(e) \leq c_e$$

Our object is to find f so as to minimize the total cost $\sum_{(i,j)\in E} a_{ij} f_{ij}$. Therefore we get a linear program as below:

$$\begin{aligned} \min_{f} \quad & \sum_{(i,j)\in E} a_{ij} f_{ij} \\ \text{s.t.} \quad & \forall v \in V, f^{out}(v) - f^{in}(v) = s_v \\ & \forall e \in E, 0 \leq f(e) \leq c_e \end{aligned}$$

4. Now I'll list all the possible cutting methods of a 3m roll to the four types of rolls. Ensure that for each cutting methods, the remains is not long enough to become another rolls.

Index	# 135cm	# 108cm	# 93cm	# 42cm
1	2	0	0	0
2	1	1	0	1
3	1	0	1	1
4	1	0	0	3
5	0	2	0	2
6	0	1	2	0
7	0	1	1	2
8	0	1	0	4
9	0	0	3	0
10	0	0	2	2
11	0	0	1	4
12	0	0	0	7

Denote x_i as the number of 3m rolls using the i_{th} cutting method. ($1 \le i \le 12$, $x_i \ge 0$)

The constraints are:

$$
\begin{aligned}
2x_1 + x_2 + x_3 + x_4 &\geq 97 \\
x_2 + 2x_5 + x_6 + x_7 + x_8 &\geq 610 \\
x_3 + 2x_6 + x_7 + 3x_9 + 2x_{10} + x_{11} &\geq 395 \\
x_2 + x_3 + 3x_4 + 2x_5 + 2x_7 + 4x_8 + 2x_{10} + 4x_{11} + 7x_{12} &\geq 211
\end{aligned}
$$

The objective is:

$$\min_{x_i \geq 0} \sum_{1 \leq i \leq 12} x_i$$

So we get a linear program as follows:

$$
\begin{aligned}
\max_{x_i \geq 0} \quad & -\sum_{1 \leq i \leq 12} x_i \\
\text{s.t.} \quad & -2x_1 - x_2 - x_3 - x_4 \leq -97 \\
& -x_2 - 2x_5 - x_6 - x_7 - x_8 \leq -610 \\
& -x_3 - 2x_6 - x_7 - 3x_9 - 2x_{10} - x_{11} \leq -395 \\
& -x_2 - x_3 - 3x_4 - 2x_5 - 2x_7 - 4x_8 - 2x_{10} - 4x_{11} - 7x_{12} \leq -211
\end{aligned}
$$

After we get the result of the linear program, we should first round x_i to integers that still satisfying the constraints and maximize the objective function. Then we get a series of integers x_i. So we will use x_i rolls with width 3m to cut as the i_{th} method listing in the table.

For example, if we get $x_1 = 100$, then we will use 100 3m rolls, each of them will be cut into 2 rolls of width 135cm. #