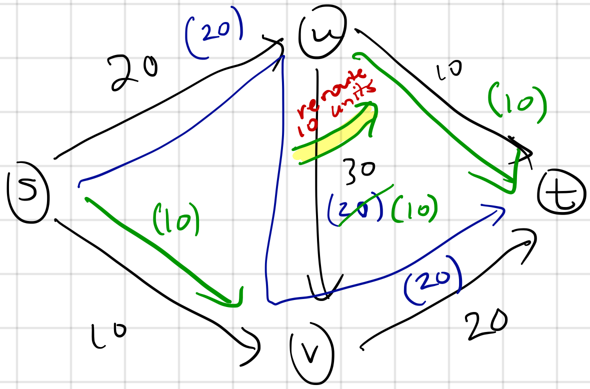
3 Mar 2019

The Ford–Fulkerson Algorithm



Reminder:

A flow assigns $f(e)$ to each edge e.

① $0 \le f(e) \le c(e)$

② $f^{out}(v) = f^{in}(v)$ $\forall v \neq s,t$

Maximize $f^{out}(s) \triangleq v(f)$ "value of f"

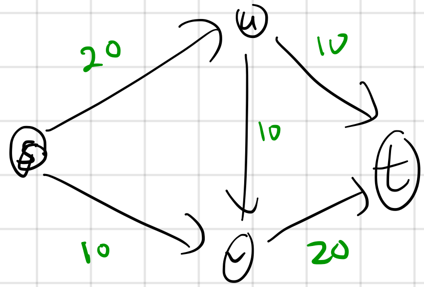
This initial flow sends 20 units from s to t.
Is that the maximum possible?

(A) Yes

(B) No, you can send 10 more units.

(C) No, you can send 20 more units.

(D) " " " " 30 " "



Being more precise about "rerouting"....
There are two ways we can modify flow on an edge $e=(u,v)$ without violating capacity constraints.

- Add more flow: but not more than $c(e) - f(e)$
- Subtract flow: but not more than $f(e)$.

With this in mind we make some definitions.

[1] Notation:
$$f(u,v) = \begin{cases} f(e) & \text{if } \exists \text{ edge } e=(u,v) \\ \emptyset & \text{if not.} \end{cases}$$

$$c(u,v) = \begin{cases} c(e) & \text{if } \exists \text{ edge } e=(u,v) \\ \emptyset & \text{if not.} \end{cases}$$

[2] Residual capacity of $e=(u,v)$ with respect to flow f.

$$C_f(e) = \overbrace{c(u,v) - f(u,v)}^{\text{potential to add flow}} + \overbrace{f(v,u)}^{\text{potential to subtract flow}}$$

on the opposite edge (v,u).

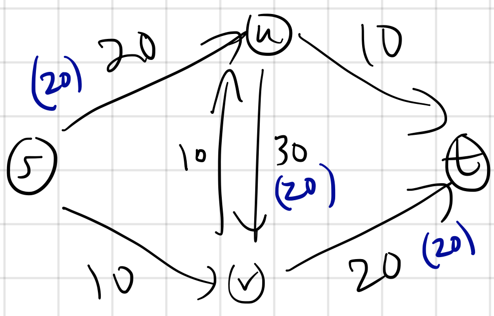
[2] Residual capacity of $e = (u,v)$ with respect to flow f.

$$c_f(e) = \underbrace{c(u,v) - f(u,v)}_{\text{potential to add flow}} + \underbrace{f(v,u)}_{\text{potential to subtract flow}}$$

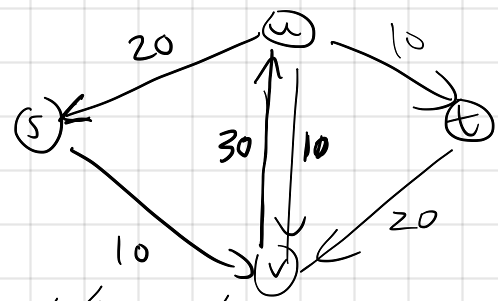
potential to subtract flow on the opposite edge (v,u).

[3] Residual graph of f is $G_f = \left(\text{vertices of } G,\ \text{edges } (u,v) \text{ with } c_f(u,v) > 0\right)$

Difference from K&T book: suppose f sends 20 units on path $s \to u \to v \to t$.

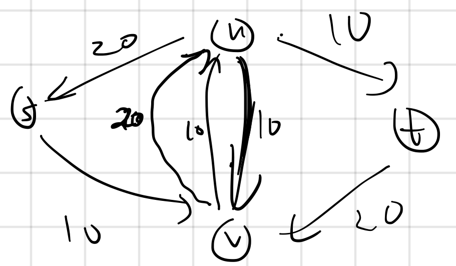


G_f



$c_f(u,s) = \cancel{c(u,s)}^{0} - \cancel{f(u,s)}^{0} + f(s,u) = 20$

K&T book



DEF. An augmenting path with respect to f is a path from s to t in G_f.

flow ↓, edge of G_f ↓ ↓, number ↓

increase (f, u, v, x): // increase "net flow" $f(u,v) - f(v,u)$ by x.
 if $f(v,u) \geq x$ $\{ f(v,u) \leftarrow f(v,u) - x \}$
 else $\{ f(u,v) \leftarrow f(u,v) + x - f(v,u)$; $f(v,u) \leftarrow 0 \}$

$f^{out}(u) - f^{in}(u)$ increases by x, $f^{out}(v) - f^{in}(v)$ decreases by x.

If $0 \leq x \leq c_f(u,v)$ then capacity constraints continue to be satisfied.

flow $\downarrow$ path from s to t in G_f

augment (f, P):

let $x = \min\{ c_f(e) : e \text{ an edge of } P \}$

for each $e = (u,v)$ in P:

increase (f, u, v, x)

// $f^{out}(s) - f^{in}(s)$ increases by x, $f^{out}(t) - f^{in}(t)$ decreases by x,
$f^{out}(v) - f^{in}(v)$ unchanged at every other v.
Capacity constraints remain satisfied.

FordFulkerson (G):

initialize $f(e) = 0$ for all e.

build G_f

while (G_f contains an augmenting path P):

augment (f, P)

recompute G_f

endwhile

output f.

Running time bound: each iteration increases $v(f)$ by at least 1. (Residual capacities are integers. Flow quantities $f(e)$ are integers.)

$v(f)$ can never exceed $\sum_{u \in V} c(s,u) \triangleq C$.

One loop iteration takes $O(m)$ time.

Worst-case running time of Ford-Fulkerson is $O(mC)$.