15 Mar 2021 The Ford–Fulkerson Algorithm

Announcements

1. Prelim 1 is being given in two shifts.
   - Thurs 3/18 8:00 pm EDT
   - Fri 3/19 5:00 pm EDT

   Separate questions on the two versions, large numbers taking each, separately curved.

   You should have gotten email from hg38@cornell.edu with time slot.

   Time limit: 2 hrs, 15 minutes ← for scanning, uploading.
   Length: 5 questions, ≤ 2 write-an-algo
   ≤ 1 additional write-a-para, 2 multi-part short ans.

2. Review session today 5:00 pm.
   See announcement on Ed.

3. Pinned post on Ed contains prelim review materials.

Recall:

1. Flow network G (dir graph)

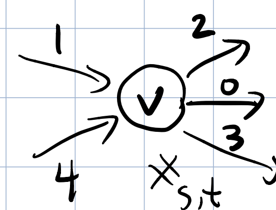
s,t (source, sink vert)

$c: E \to \mathbb{R}_{>0}$ (capacities)

For today, integers.

Flow: $f: E \to \mathbb{R}_{\geq 0}$

satisfies conservation



Capacity $0 \leq f(e) \leq c(e)$ $\forall$ edge e

2. Residual graph G_f has $V(G_f) = V(G)$

$$E(G_f) = \{(u,v) \mid f(u,v) < c(u,v)\}$$ forward

$$\cup \{(u,v) \mid f(v,u) > 0\}$$ backward

Residual capacity of an edge.

$$c_f(e) = \begin{cases} c(e) - f(e) & \text{if } e=(u,v) \text{ forward} \\ f(v,u) & \text{if } e=(u,v) \text{ backward} \end{cases}$$

3. Augmenting path: path from s to t in G_f.

Augment (f, P):

1. Find "bottleneck value"

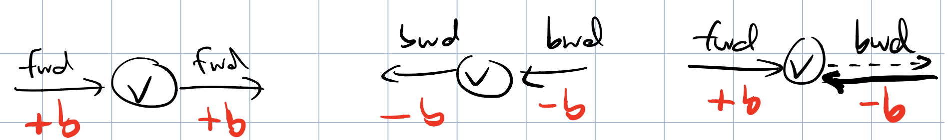
$$b(f, P) = \min\{c_f(e) \mid e \in P\}$$

2. Update flow on P.

$$f'(e) = \begin{cases} f(e) + b(f, P) & \text{if } e \in P \text{ forward} \\ f(e) - b(f, P) & \text{if } e \in P \text{ backward} \end{cases}$$

$$f'(e) = f(e) \qquad \text{if } e \notin P.$$

If f is a flow, so is f'.



Choice of $b(f, P)$ designed to ensure that augmenting P never violates capacity constraints.

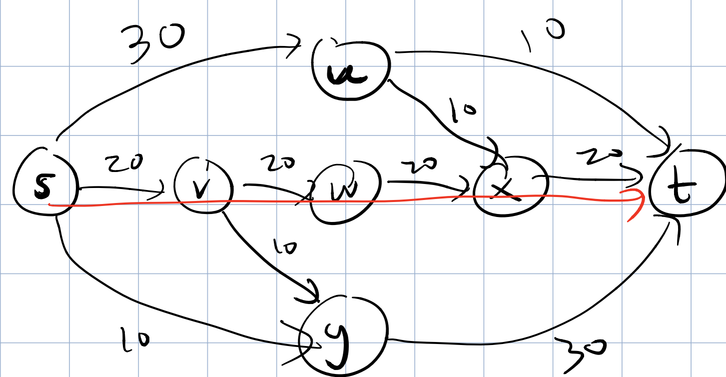
FORD-FULKERSON (G)

initialize $f(e) = \emptyset$ for every $e \in E$.

repeat {

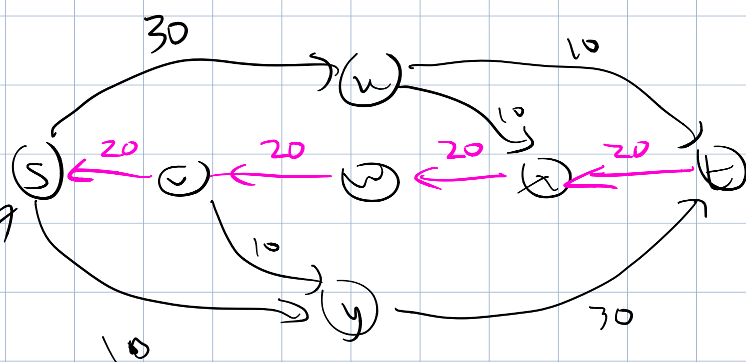
construct residual graph G_f.

search for augmenting path P.
if P found, augment (f, P)
} until not augmenting path exists in G_f
output f.



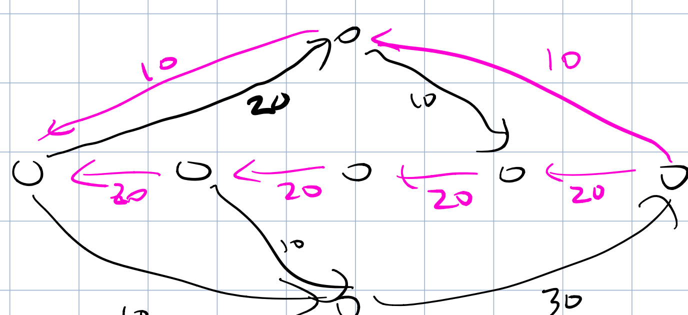
$P_1 = s \to v \to w \to x \to t$

Residual Graphs



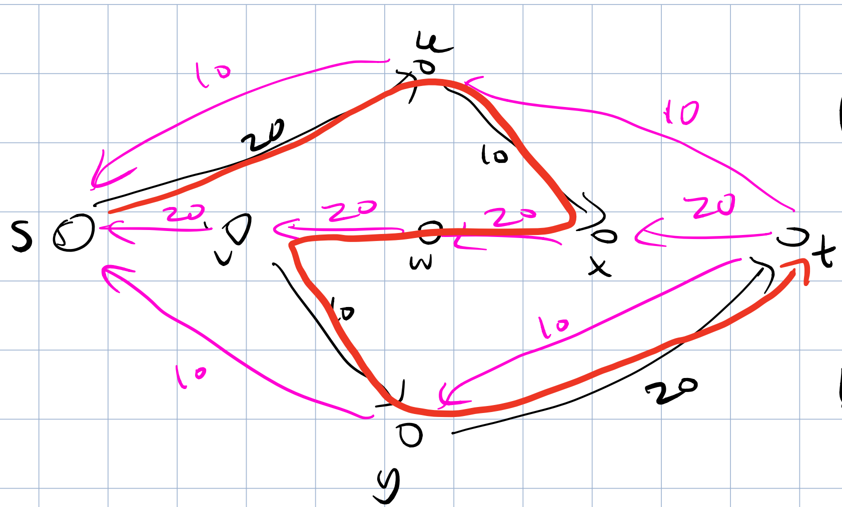
$P_2 : s \to u \to t$

$b(f, P_2) = 10$



$P_3 : s \to y \to t$

$b(f, P_3) = 10$

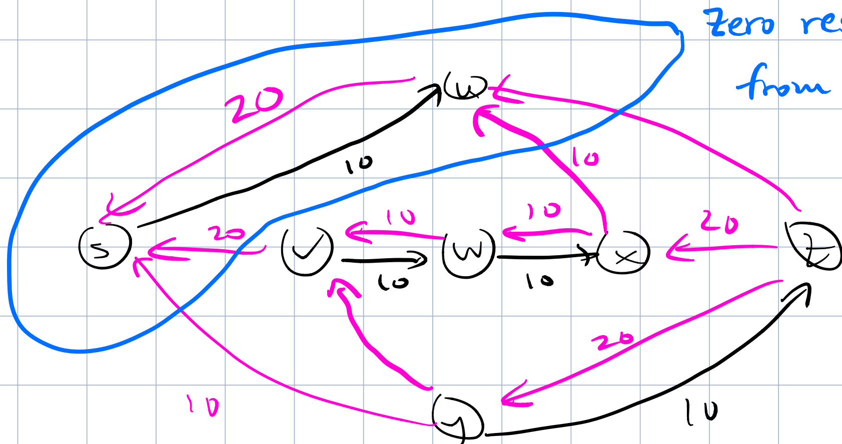


P_4: $s \to u \to x$

$w \to v$

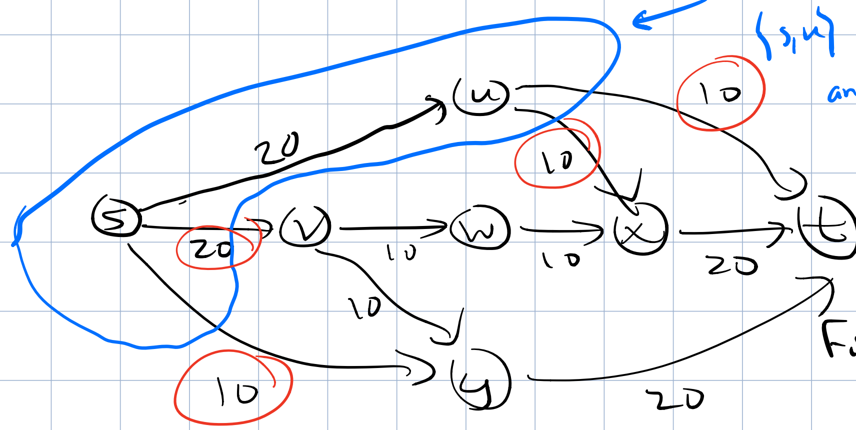
$y \to t$

$b(f, P_4) = 10$



Zero residual capacity from $\{s,u\}$ to $\{v,w,x,y,t\}$.

All the edges leaving $\{s,u\}$ are saturated and no flow is coming into $\{s,u\}$.



Final Flow.

Is this really a maximum flow?

Combined capacity of all edges leaving $\{s,u\}$ is 50, so no flow can send more than 50 from s to t.

Def. An s-t cut is a partition of $V(G)$ into sets A, B such that $s \in A$, $t \in B$. The capacity of the cut is $c(A,B)$

$$c(A,B) = \sum_{\substack{e=(u,v) \\ u \in A,\ v \in B}} c(e).$$ ← Total capacity of edges from A to B.

Theorem (Max-Flow Min-Cut Theorem)

For every flow network, the maximum flow value equals the minimum cut capacity. Furthermore the Ford-Fulkerson alg is guaranteed to find a max flow. (If edges have integer capacities.)

Proof. We will prove $\forall$ flow f, $\forall$ cut A,B

(*) $$v(f) \leq c(A,B).$$

which shows max flow $\leq$ min cut.

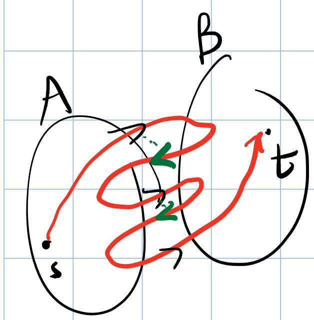
Then we'll show if $f^* =$ output of F-F alg and $A^* = \{$vertices reachable from s in $G_{f^*}\}$
$B^* = V - A^*$

then A^*, B^* is an (s,t)-cut and

(**) $$v(f^*) = c(A^*, B^*)$$

which shows max-flow $\geq$ min-cut.

For arbitrary flow f and cut (A, B)



$v(f)$ = total flow leaving s

= (flow from A to B) − (flow from B to A)

[by flow conservation]

..... will elaborate if time

$\leq c(A, B)$ [by capacity constraints]

For the output of Ford-Fulkerson, f^*:
we know G_{f^*} has no path from s to t.

s belongs here

$\therefore \quad A^* = \{\text{vert reachable from } s \text{ in } G_{f^*}\}$

$B^* = V - A^*$

t belongs here

No G_{f^*} edges from A^* to B^*.

$$c(A^*, B^*) = \sum_{\substack{e=(u,v) \\ u \in A^*,\, v \in B^*}} c(e) \overset{①}{=} \sum_{\substack{e \in (u,v) \\ u \in A^*,\, v \in B^*}} f^*(e)$$

(2)

Flow from B^* to A^* is zero.
(Else backward edge $A^* \to B^*$ in G_{f^*}.)

$$v(f^*) = (\text{flow from } A^* \text{ to } B^*) - (\text{flow from } B^* \text{ to } A^*)$$

$$= c(A^*, B^*) - \emptyset$$

$$= c(A^*, B^*)$$