The Maximum Flow Prob

15-750
3/22/19

Def A flow network is

1) $G=(V,E)$ directed (oriented)
2) Edge capacities $c: V \times V \to \mathbb{R}$
   st $c(u,v) \geq 0$
   eg $(u,v) \notin E$ then $c(u,v)=0$
3) $s \neq t \in V$ $s \equiv$ source & $t \equiv$ sink

Def $f: V \times V \to \mathbb{R}$ is a flow for network G if

1) Capacity constraints: $f(u,v) \leq c(u,v)$
2) Skewed symmetric: $f(u,v) = -f(v,u)$
3) Flowin = Flowout for $u \in V-\{s,t\}$

$$\sum_{v \in V} f(u,v) = 0$$

Def Netflow $\equiv |f| = \sum_{v \in V} f(s,v)$

The Maximum Flow Prob

Input: Flow-Network $G=(V,E), s, t, c$
Output: Flow f with maximum net-flow.

Residual Network

Consider Network $G=(V,E,c)$ & flow f

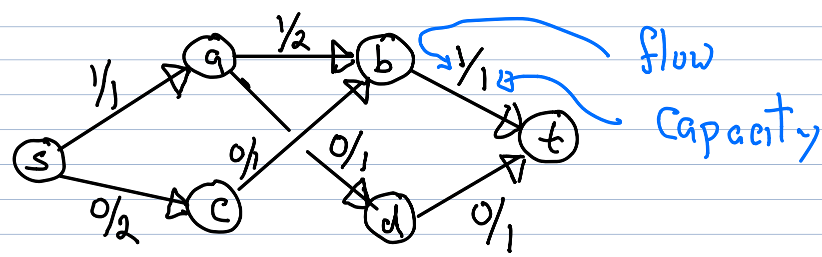
Def Residual Capacity: $c_f(u,v) = c(u,v) - f(u,v)$

Residual Network:

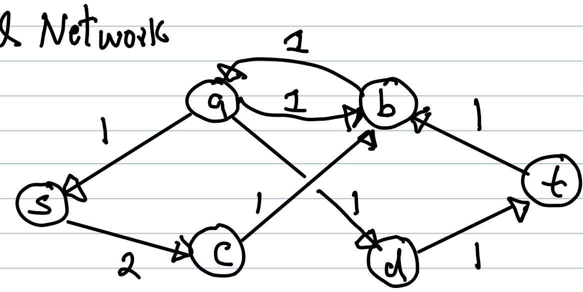
Edges $E_f = \{(u,v) \in V^2 \mid c_f(u,v) > 0\}$

$$G_f = (V, E_f, c_f)$$

Ex Network & flow



Residual Network



Ford-Fulkerson Method

F-F-Method (G, s, t, c)

1) Initialize flow f to zero.
2) While $\exists$ augmenting Path p in G_f

   Let f_p be max flow on p.

   Set $f = f_p + f$
3) Return f.

<u>Def</u> P is an <u>aug path</u> if P is a path in G_f using edges with positive capacities.

<u>Lemma</u> f flow on G, G_f the residual

a) f' is flow on G_f iff $f+f'$ flow on G.

b) f' is max flow on G_f iff $f+f'$ is max flow on G

c) $|f+f'| = |f| + |f'|$ (f' has no flow into s)

<u>proof</u> a) $f'(e) \le c_f(e)$ iff $f'(e) \le c(e) - f(e)$

iff $(f'+f)e \le c(e)$

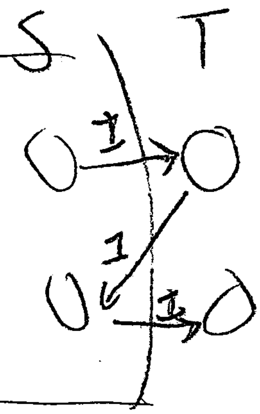
S, T cuts

Def $S, T \subseteq V$ is a <u>cut</u> if

1) $S \cap T = \emptyset$ & $S \cup T = V$
2) $s \in S$ & $t \in T$

$$Cap(S,T) \equiv \sum_{u \in S, v \in T} c(u,v)$$

eg



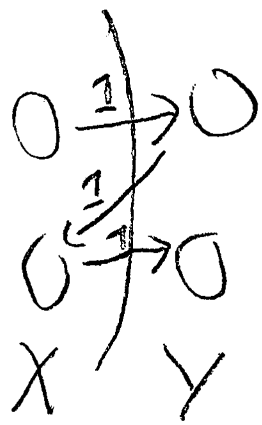
Cap = 2

---

f flow

$$f(X,Y) \equiv \sum_{u \in X} \sum_{v \in Y} f(u,v)$$

eg



flow = 1

$f(S,T) \equiv$ net flow from S to T

$|f| \equiv f(s, V-s)$

<u>Lemma</u> If f is a flow on G & (S,T) is a cut then $|f| = f(S,T)$.

<u>Pf</u> By induction on $|S|$

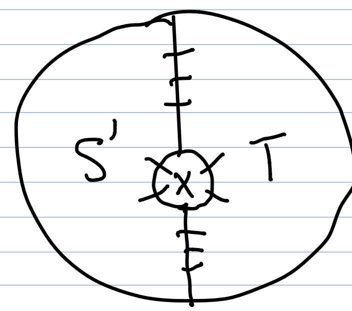
$|S|=1$ done by def.

Assume true sets of size less than S.

i.e. $|f| = f(S',T')$ where $S = S' \cup \{x\}$

Thus $S = S' \cup \{x\}$, $T = T' \setminus \{x\}$, $T' = T \cup \{x\}$

We get



$f(S,T) = f(S',T) + f(x,T)$

$f(S',T') = f(S',T) + f(S',x)$

$|f| = f(S',T')$

$= f(S',T) + f(x,T)$

$= f(S,T) - f(x,T) + f(x,T)$

$= f(S,T)$ flowin = flowout of X.

Cor (S,T) cut then $|f| \leq c(S,T)$

since $f(S,T) \leq c(S,T)$

Thm Max-flow = Min-Cut

i.e. The following are equivalent

1) f is a max-flow
2) G_f contains no augmenting path.
3) $\exists$ cut (S,T) s.t. $|f| = Cap(S,T)$

We will show 1) $\Rightarrow$ 2) $\Rightarrow$ 3) $\Rightarrow$ 1)

1) $\Rightarrow$ 2) ie $\neg 2) \Rightarrow \neg 1)$

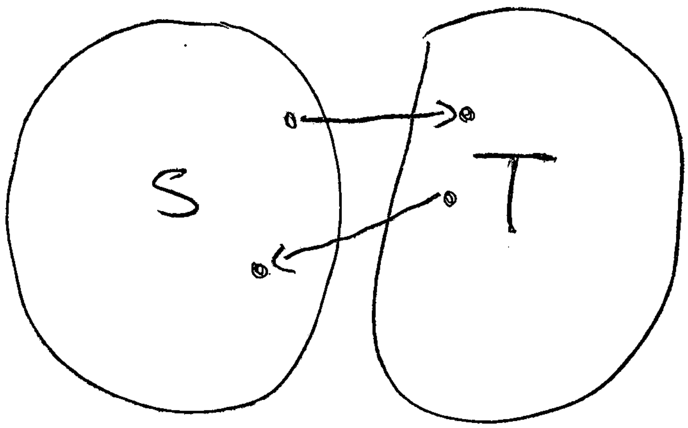
$\exists$ augmenting path $\Rightarrow$ f is not maximum

2) $\Rightarrow$ 3)

Let $S \subseteq V \equiv$ reachable vertices from s in G_f

$T = V - S$ note: $t \in T$ by 2).

In G



1) f saturates all edges from S to T

2) f does not use edges from T to S

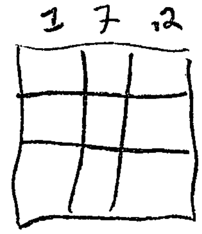
$\therefore |f| = f(S,T) = cap(S,T)$

3) $\Rightarrow$ 1) clear

An Image Segmentation Prob

Foreground / Background Prob

Input: 1) Pixel Image eg



2) Affinity graph. $G=(V,E)$

weights $P_{ij} \equiv$ similarity of pixel V_i & V_j

3) $b_j \equiv$ likelihood V_j in background

$a_j \equiv$ " " foreground

Output: Partition A, B of V s.t.

$$\max_{A,B} g(A,B) = \sum_{i \in A} a_i + \sum_{i \in B} b_i - \sum_{\substack{i \in A \\ j \in B \\ (i,j) \in E}} P_{ij}$$

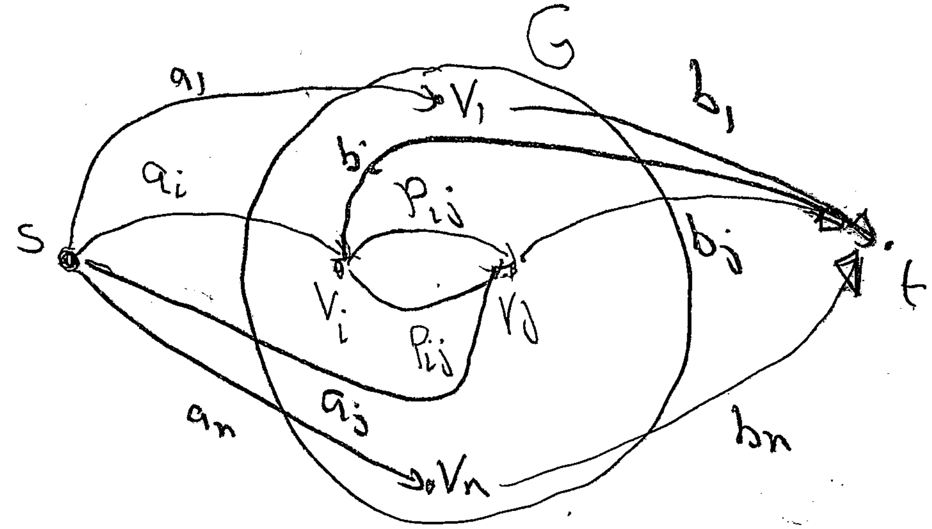
Change to Min Prob.

$Q \equiv \sum_i (a_i + b_i)$ then

$$g(A,B) = Q - \sum_{i \in A} b_i - \sum_{i \in B} a_i - \sum P_{ij}$$

Goal: $$\text{Min } g'(A,B) = \sum_{i \in A} b_i + \sum_{i \in B} a_i + \sum_{\substack{i \in A \\ j \in B}} P_{ij}$$

Make a flow-Network:

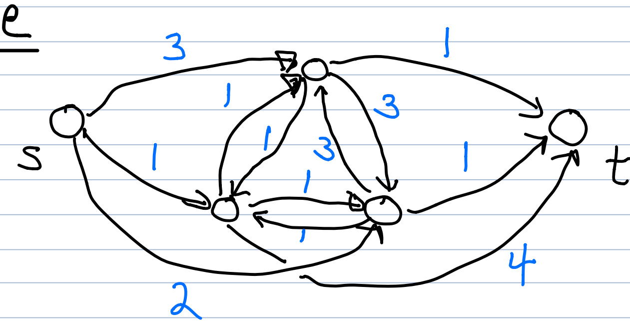


Claim If A, B is a cut then $s \in A$ & $t \in B$

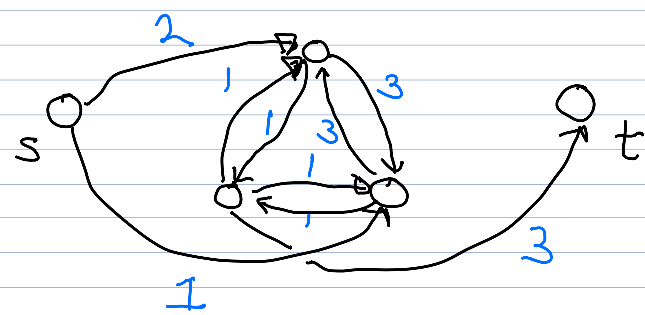
$$C(A,B) = g'(A,B).$$

Thus Min Cut is a Max Segmentation.

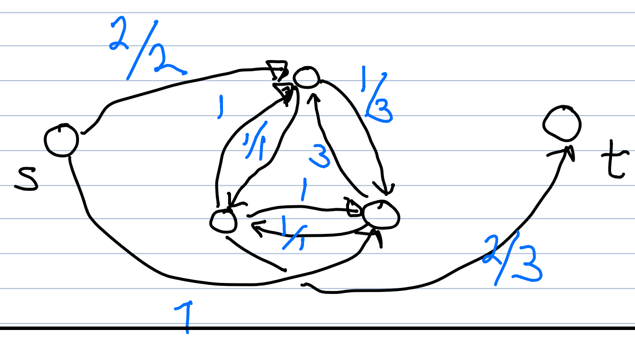
Example



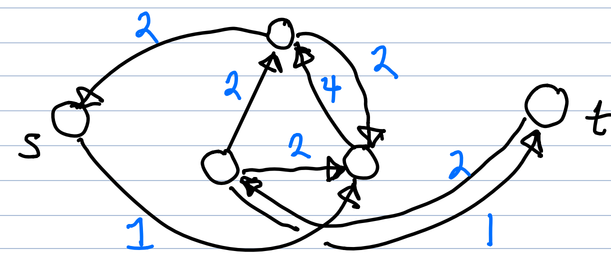
Note We may assume that $a_i = 0$ or $b_i = 0$



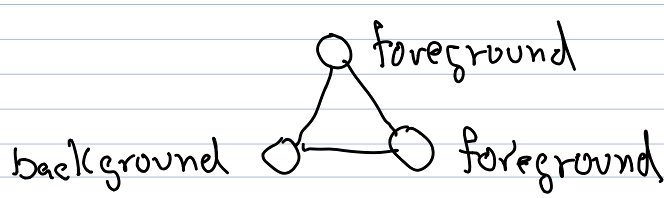
Consider f



Residual



Thus



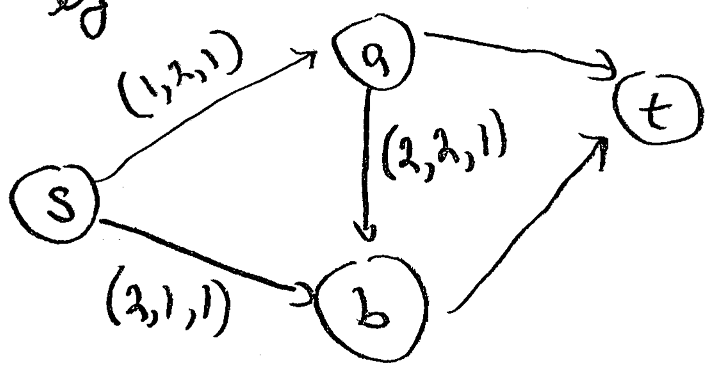
Dynamic Networks

Input: Network: $G=(V,E)$

Discreet time: $0 \le t \le 2$

Capacities: $C_{ij}^t \equiv$ cap from a_i to a_j at time t.

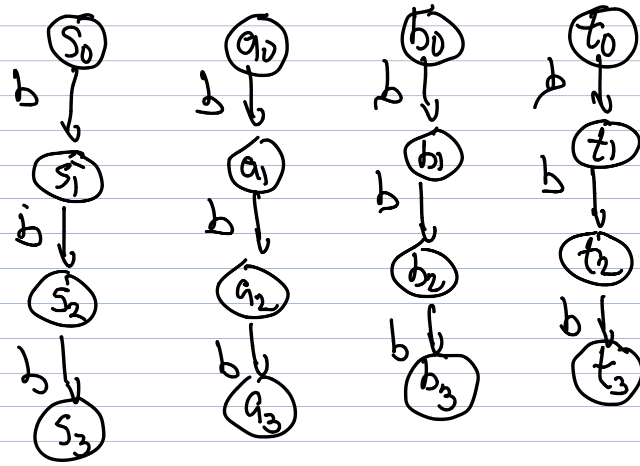
eg



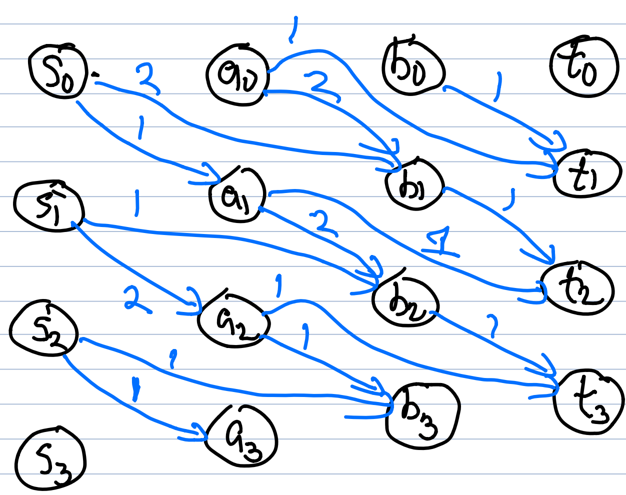
Buffer size: b_a^t

Idea Make 4 copies of static network and add edges between copies.

Add buffer edges



Add transition edges.



Solve max flow from s_0 to t_3 on network of buffer & transition edges.