5800
Max Flows

mar 23/24 2022
shelat

Max flow

Min Cut

“Consider a rail network connecting two cities by way of a number of intermediate cities, where each link of the network has a number assigned to it representing its capacity. Assuming a steady state condition, find a maximal flow from one given city to the other.”

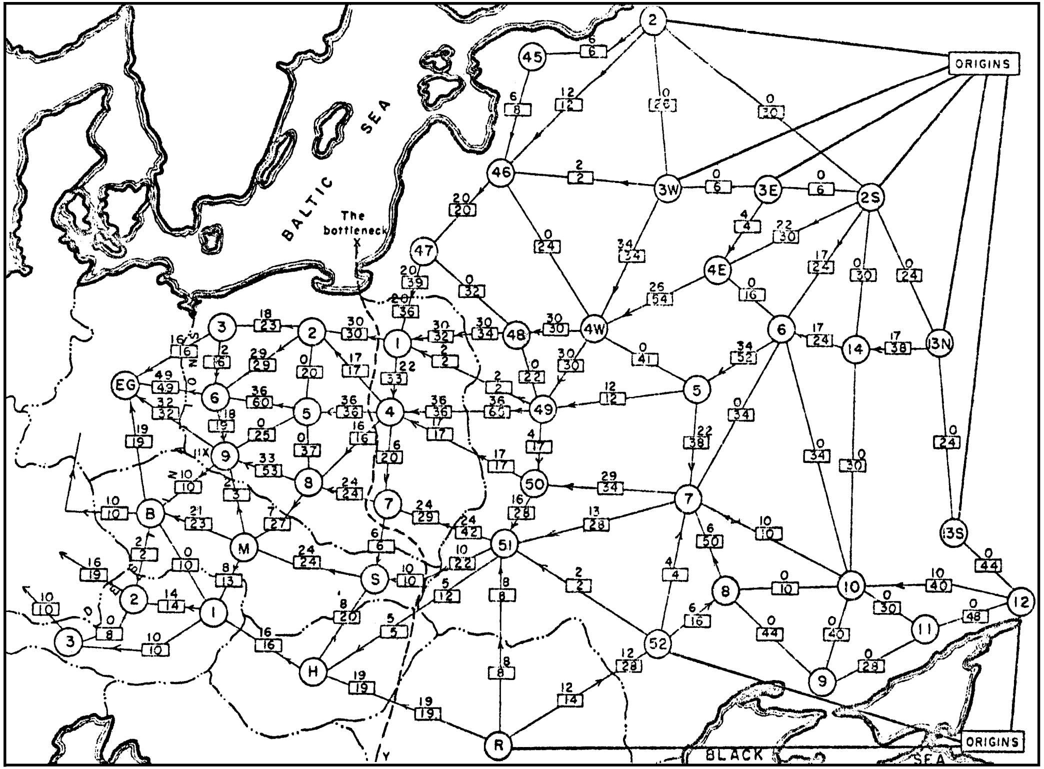


Figure 4 From Harris and Ross [3]: Schematic diagram of the railway network of the Western Soviet Union and East European countries, with a maximum flow of value 163,000 tons from Russia to Eastern Europe and a cut of capacity 163,000 tons indicated as ‘The bottleneck’

courtesy Alexander Schrijver

flow networks

$$G = (V, E)$$ $c : E \to \mathbb{N}$ capacities

source + sink: source $s \in V$ sink $t \in V$.

capacities:

flow networks

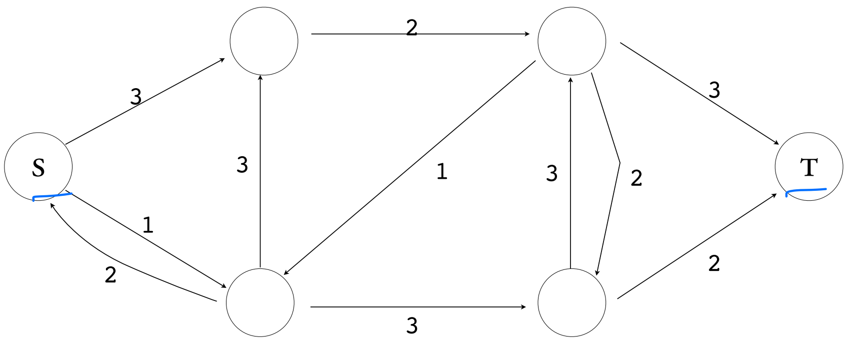
$$G = (V, E)$$

source + sink: node s, and t

capacities: $c(u, v)$

assumed to be 0 if no (u,v) edge

example



flow

A FLOW IS A MAP FROM EDGES TO NUMBERS: $f : E \to \mathbb{N}$ $\mathbb{R}^+$

CAPACITY CONSTRAINT: $f(e) \leq c(e)$ for every $e \in E$

FLOW CONSTRAINT: for every node $v \in V - \{s,t\}$

(inflow = outflow)

$$\underbrace{\sum_{w \in V} f(w,v)}_{\text{(inflow)}} = \underbrace{\sum_{w \in V} f(v,w)}_{\text{(outflow)}}$$

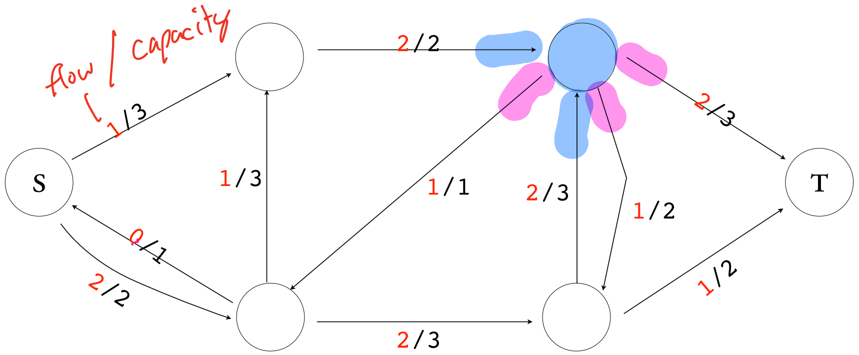
Net flow value

$$|f| = \sum_{v \in V} f(s,v) - \sum_{v \in V} f(v,s)$$

example

— flow

4 = 4



$|f| = (1+2) - 0 = 3.$

max flow problem

Given a graph $G = (V, E)$ and capacities $c : E \to \mathbb{N}$, compute

$$\operatorname{ARGMAX}_{f} \; |f|$$

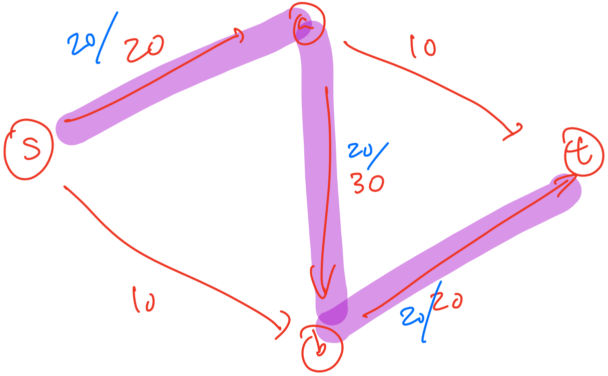
max flow problem

Given a graph $G = (V, E)$ and capacities $c : E \to \mathbb{N}$, compute

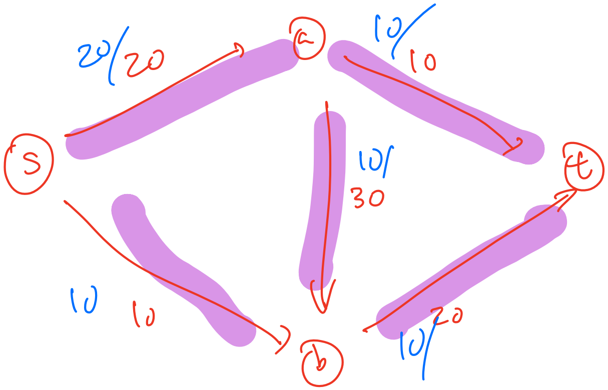
$$\operatorname{argmax}_f |f|$$

i.e., the maximum flow over all valid flows.

greedy solution?



$|f| = 20$



$|f| = 30$

hundreds of applications

bipartite matching
edge-disjoint paths
node-disjoint paths
scheduling
baseball elimination
resource allocations

will discuss many of these applications soon

Algorithms for max flow

(V,E) Residual graphs

given a graph G, and a flow f, we will define

$G_f = (V, E_f)$, the residual graph.

c_f

E_f: • include the edge $e \in E$

set the new capacity to be $c_f(e) = c(e) - f(e)$

• if $e = (u,v) \in E$ and $f(e) > 0$, then add the edge

(v,u) with $c_f(v,u) = f(e)$.

Residual graphs

$G_f = (V, E_f)$ A graph derived from G and a valid flow f.

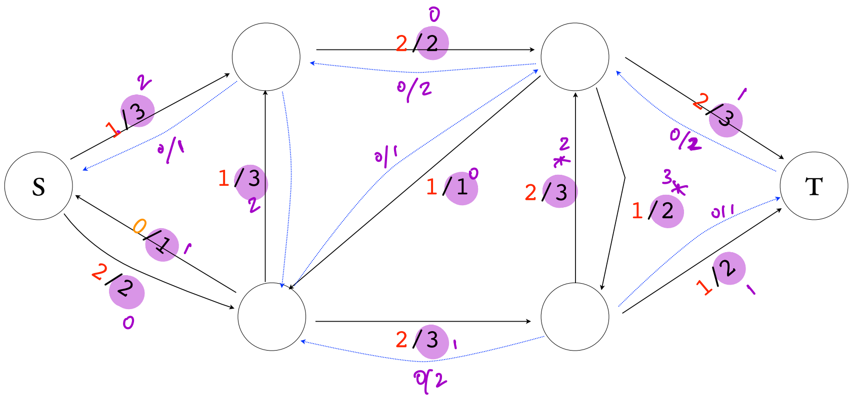
Residual graphs

$G_f = (V, E_f)$

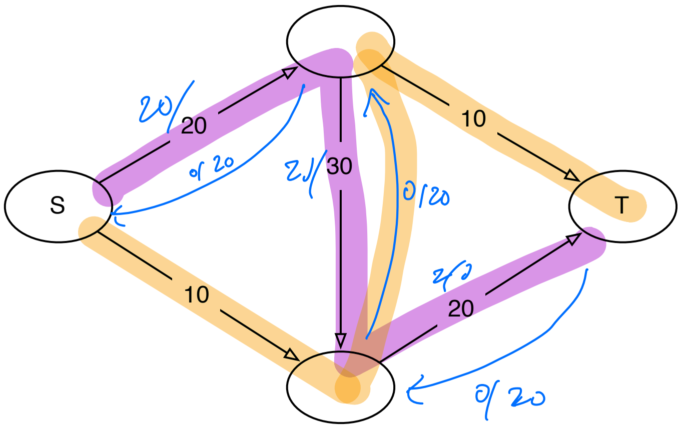
A graph derived from G and a valid flow f.

Same vertices, but difference edges:

example residual graph



why residual graphs ?



augmenting paths

DEF: any path from s to t in the residual graph

augmenting paths

DEF: A path from s to t in the residual graph G_f.

Ford-Fulkerson

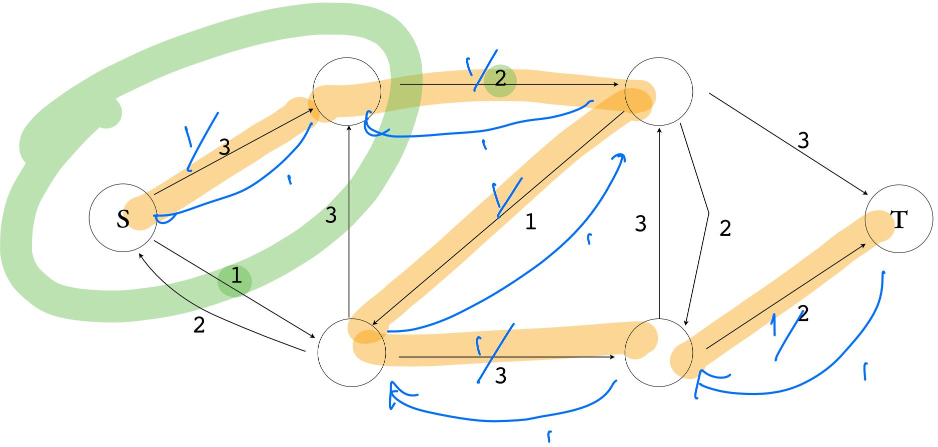
$G=(V,E)$, c

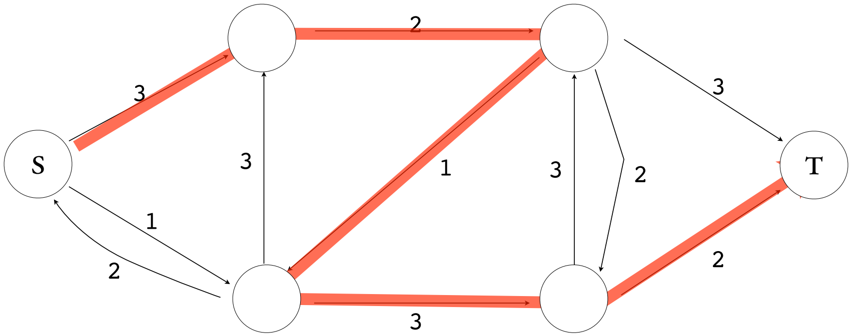
INITIALIZE $f(u,v) \leftarrow 0 \; \forall u,v$

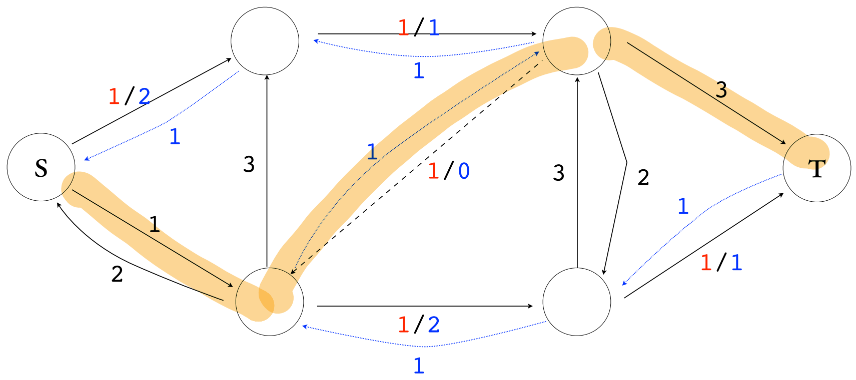
WHILE EXISTS AN AUGMENTING PATH p IN G_f

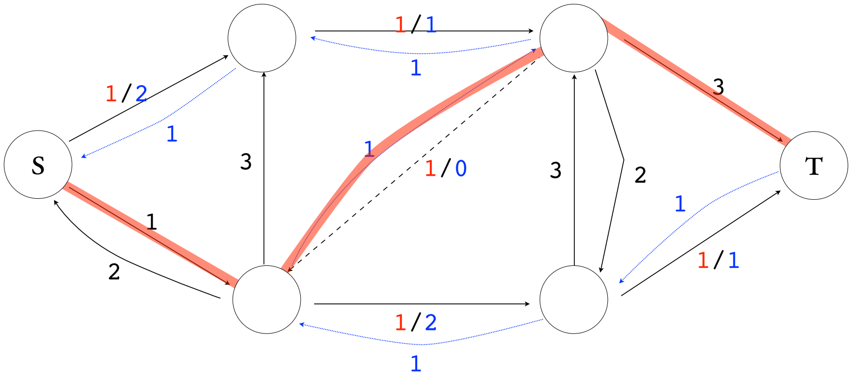
AUGMENT f WITH $c_f(p) = \min\limits_{(u,v)\in p} c_f(u,v)$

first augmenting path.

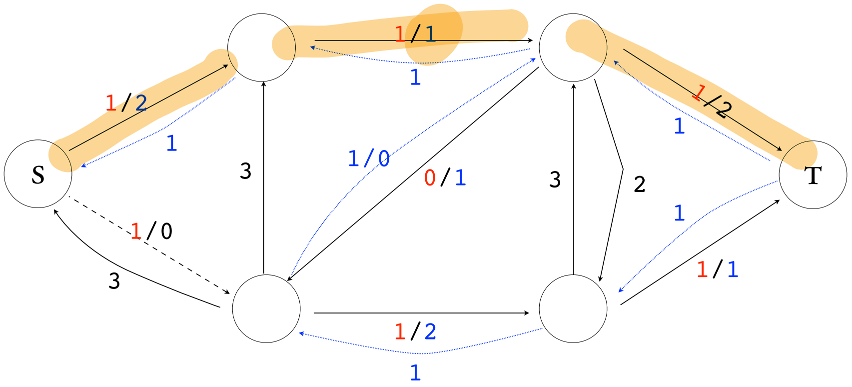



S
T
3
2
3
1
3
1
2
3
3
2
3
2

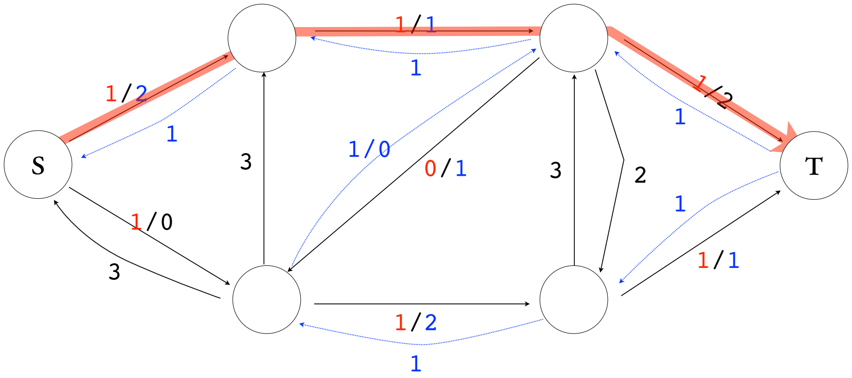

S
T
1 / 2
1
3
1 / 1
1
1
1 / 0
3
3
2
1
2
1
1 / 1
1 / 2
1



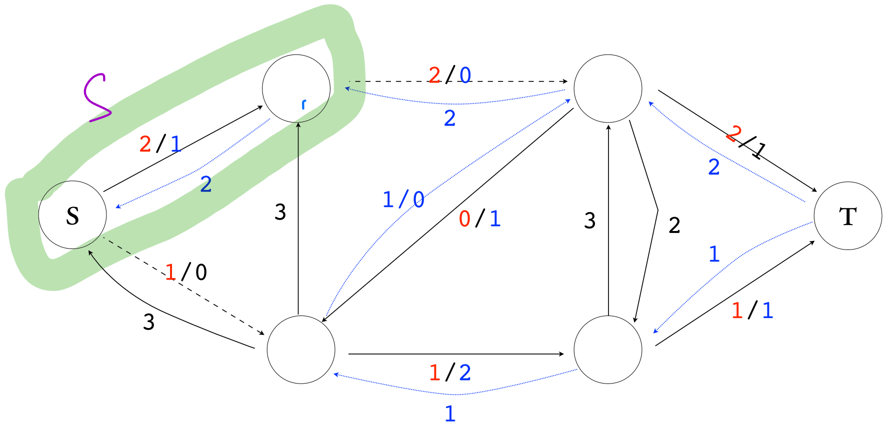
S
T
1/1
1
1/2
1
3
1
1/0
3
2
3
1
1
2
1/2
1
1/1



S
T
1/2
1
1/1
1
1/2
1
3
1/0
0/1
3
2
1
1/0
3
1/1
1/2
1



S
T
1/2
1
1/1
1
1/2
1
3
1/0
0/1
3
2
1/0
3
1
1/1
1/2
1



$|f| = 3$

FORD-FULKERSON

INITIALIZE $f(u,v) \leftarrow 0 \; \forall u,v$

WHILE EXISTS AN AUGMENTING PATH p IN G_f

AUGMENT f WITH $c_f(p) = \min_{(u,v)\in p} c_f(u,v)$

TIME TO FIND AN AUGMENTING PATH: $\Theta(E+V)$

BFS, DFS

NUMBER OF ITERATIONS OF WHILE LOOP:

$\Theta(E \cdot V) \cdot |f|$

Cuts

s-t cut

Def of a cut: partition of V into $(S, V-S)$ such that

$s \in S \quad t \in V-S$

cost of a cut:

$$||S, T|| =$$ sum of the capacities of the edges that cross the cut:

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

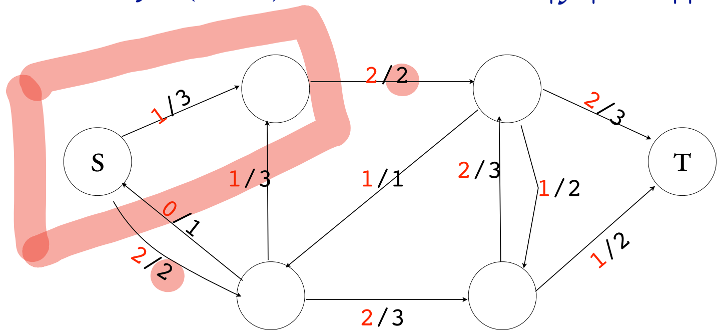
flow and cut

lemma: [min cut] for any $f, (S, T)$

flow $\leq$ cost of cut

$$|f| \leq ||S, T||$$

FOR ANY $f, (S, T)$ IT HOLDS THAT $|f| \leq ||S, T||$



EXAMPLE:

A property to remember

FOR ANY $f, (S,T)$ IT HOLDS THAT $|f| \leq ||S,T||$

PROOF: Consider some flow f.

① $$|f| = \sum_{v \in V} f(s,v) - \sum_{v \in V} f(v,s)$$

② Now consider the set $S = \{s, a, b, \ldots\}$

for all $u \in S - \{s\}$

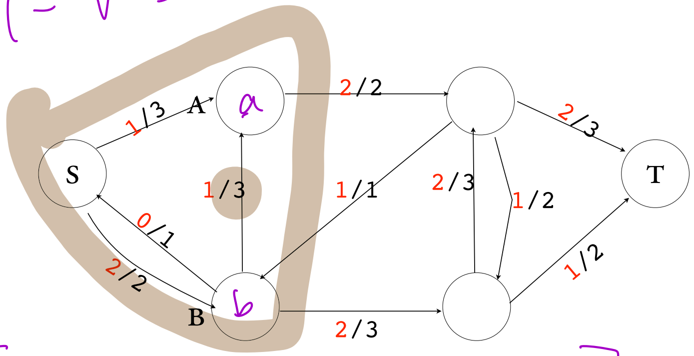
$$\sum_{v \in V} f(u,v) - \sum_{v \in V} f(v,u) = 0$$

by flow constraint

③ $$|f| = \sum_{v \in V} f(s,v) - \sum_{v \in V} f(v,s) + \sum_{u \in S - \{s\}} \Big(\underline{\sum_{v \in V} f(u,v) - \sum_{v \in V} f(v,u)} \Big)$$

④ $$|f| = \sum_{u \in S} \Big[\sum_{v \in V} f(u,v) - \sum_{v \in V} f(v,u) \Big]$$

$T = V - S$



$$|f| = \sum_{u\in S}\left[\sum_{v\in V} f(u,v) - \sum_{v\in V} f(v,u)\right]$$

these 2 terms cancel out.

$$= \sum_{u\in S}\left[\sum_{v\in T} f(u,v) + \overbrace{\sum_{v\in S} f(u,v) - \sum_{v\in S} f(v,u)} - \sum_{v\in T} f(v,u)\right]$$

$u=b$ $\quad v=a$ $\quad f(b,a)$

$u=a$ $\quad v=b$ $\quad -f(b,a)$

FOR ANY $f, (S, T)$ IT HOLDS THAT $|f| \leq ||S, T||$

(FINISHING PROOF)

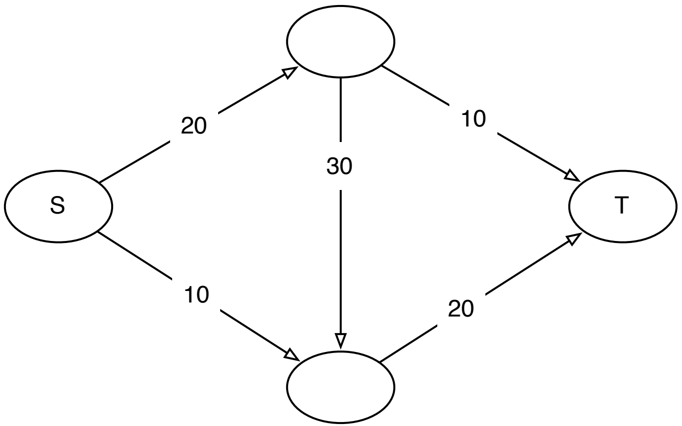
$$= \sum_{u \in S} \left[\sum_{v \in T} f(u,v) + \cancel{\sum_{v \in S} f(u,v) - \sum_{v \in S} f(v,u)} - \sum_{v \in T} f(v,u) \right]$$

$$|f| = \sum_{u \in S} \left[\sum_{v \in T} f(u,v) - \sum f(v,u) \right]$$

$$\leq \sum_{u \in S} \sum_{v \in T} f(u,v) \leq \sum_{u \in S} \sum_{v \in T} c(u,v) = ||S, T||$$

CAPACITY CONSTRAINT

why residual graphs ?



augmenting paths

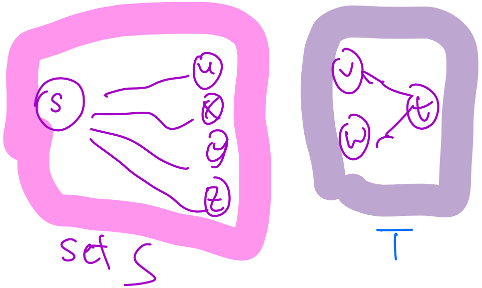
DEF:

Thm: max flow = min cut

$$\max_f |f| = \min_{S,T} ||S,T||$$

IF F IS A MAX FLOW, THEN GF HAS NO AUGMENTING PATHS.

Define the set $S = \{ v \mid \exists \text{ a path from } s \rightsquigarrow v \text{ in } G_f \text{ with } c_f(p) > 0 \}$

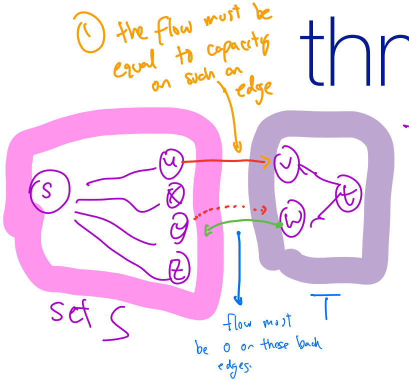


$T = V - S.$

① $s \in S$, and $t \in T$.

② (S, T) is therefore a cut.

thm: max flow = min cut



$$\max_{f} |f| = \min_{S,T} ||S,T||$$

① Consider some $u \in S$ and $v \in T$.

$f(u,v) = c(u,v)$. If not, then $c_f(u,v) > 0$ and so $v \in S$.

② $f(w,y) = 0$ for any $w \in T$ $y \in S$

If it were positive, then there would be a residual edge from y to w with $c_f(y,w) > 0$. So w would be in S!!

Finally,

$$|f| = \sum_{u \in S} \left[\sum_{v \in T} f(u,v) - \sum_{v \in T} f(v,u) \right]$$

$$= \sum_{u \in S} \overset{①}{\sum_{v \in T}} f(u,v) - \sum \overset{②}{\sum} f(v,u)$$

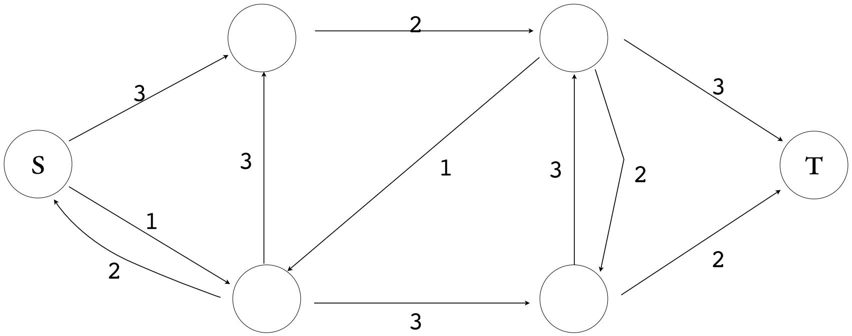
$$= \sum \sum c(u,v) - 0 = ||S,T||$$

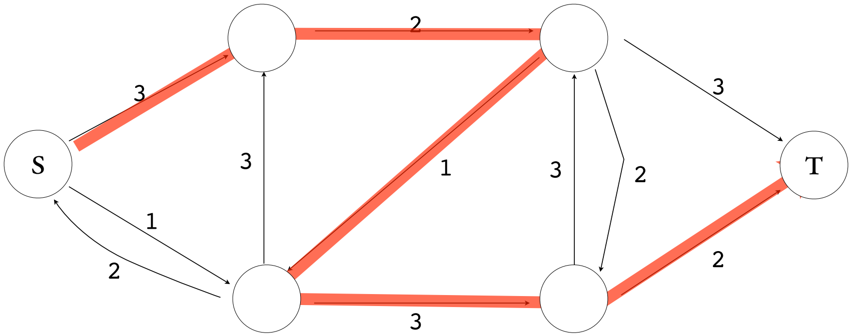
ford-fulkerson

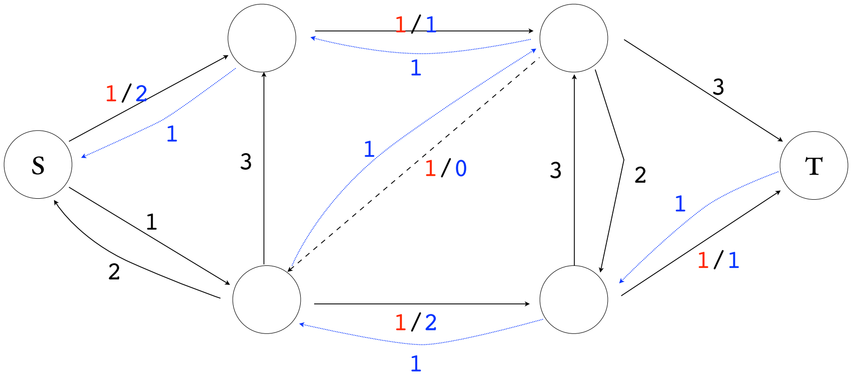
INITIALIZE $f(u,v) \leftarrow 0 \; \forall u,v$

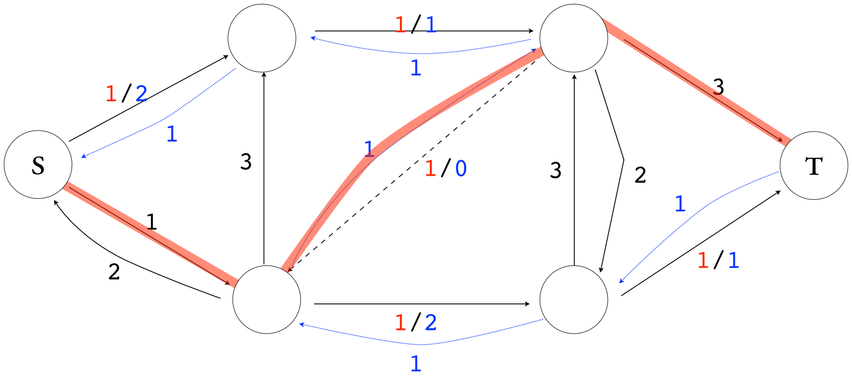
WHILE EXISTS AN AUGMENTING PATH p IN G_f

AUGMENT f WITH $c_f(p) = \min_{(u,v) \in p} c_f(u,v)$

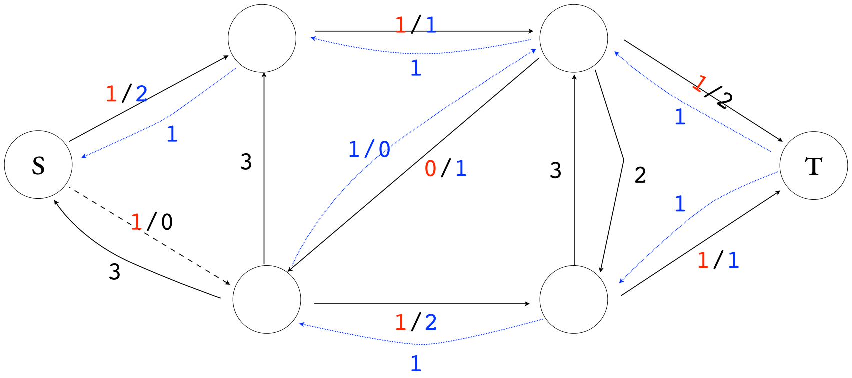

S
T
3
2
3
3
1
3
2
1
2
3
2

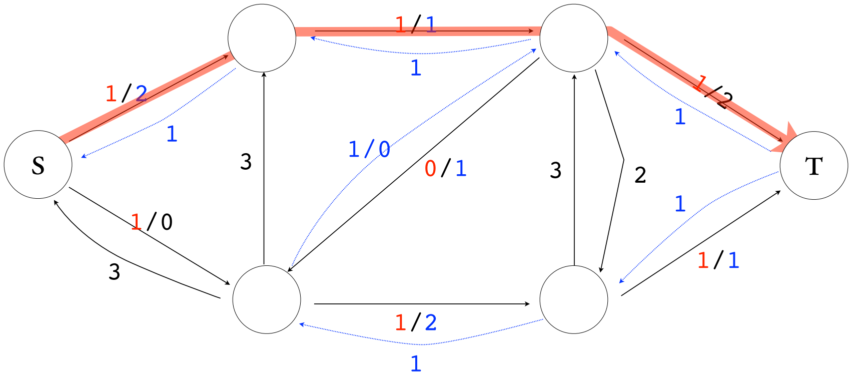

S
T
3
2
3
1
1
2
3
3
2
3
3
2


S
T
1/2
1
1/1
1
3
1
1/0
1
3
2
3
1
2
1/2
1
1
1/1

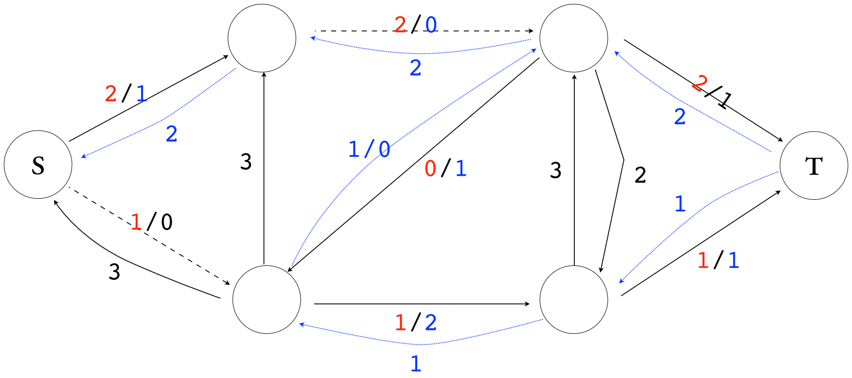


S
T
1/2
1
1
2
3
1/1
1
1
1/0
3
3
2
1
1/1
1/2
1


S
T
1/2
1
1/1
1
1/2
1
1/0
0/1
3
3
2
1/0
3
1
1
1/1
1/2
1



S
T
1/2
1
1/1
1
1/2
1
3
1/0
0/1
3
2
1
1/0
3
1/2
1
1/1



S
T
2 / 0
2
2 / 1
2
3
1 / 0
0 / 1
3
2
2 / 1
2
1
1 / 0
3
1 / 2
1
1 / 1

FORD-FULKERSON

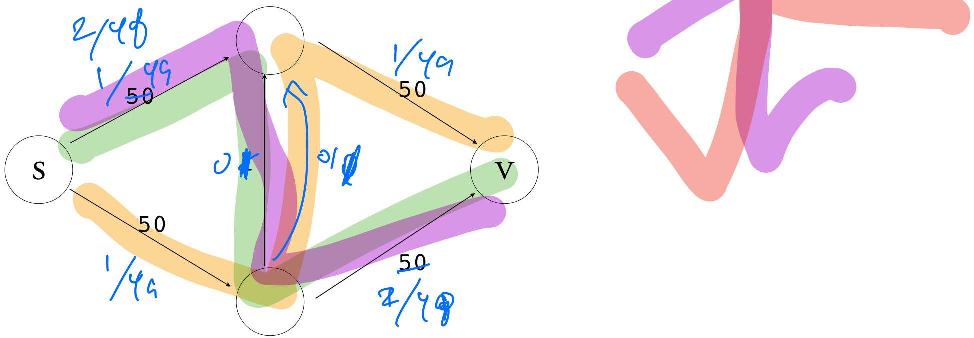
INITIALIZE $f(u,v) \leftarrow 0 \; \forall u,v$

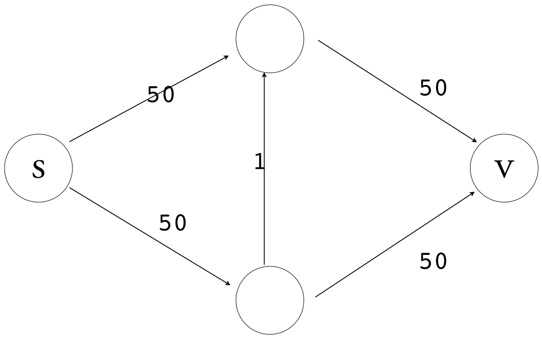
WHILE EXISTS AN AUGMENTING PATH p IN G_f

AUGMENT f WITH $c_f(p) = \min_{(u,v) \in p} c_f(u,v)$

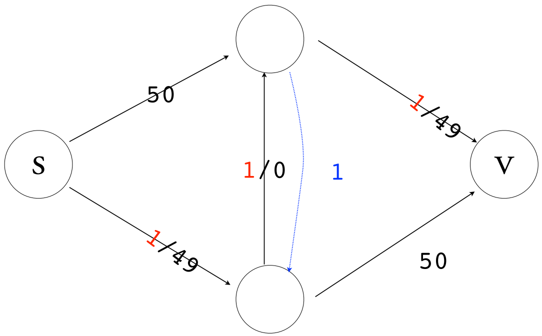
TIME TO FIND AN AUGMENTING PATH:

NUMBER OF ITERATIONS OF WHILE LOOP:

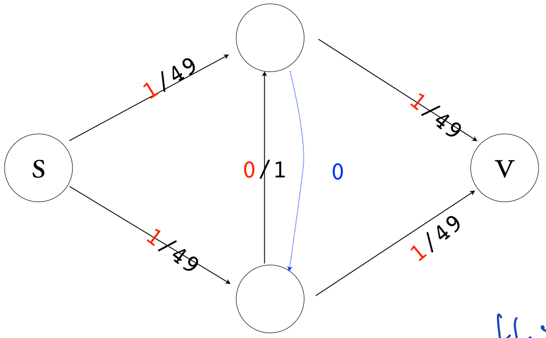

S
V
50
50
50
50
1/49
1/49
1/49



s
v
50
50
1
50
50



S
V
50
1/49
1/0
1
1/49
50

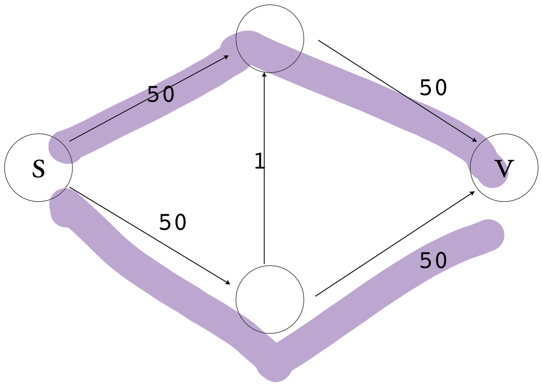


this could require

100

augmenting paths !!

root of the problem



Edmonds-Karp 2

$G=(V,E)$ a graph

f a flow on G.

choose path with fewest edges first. (BFS)

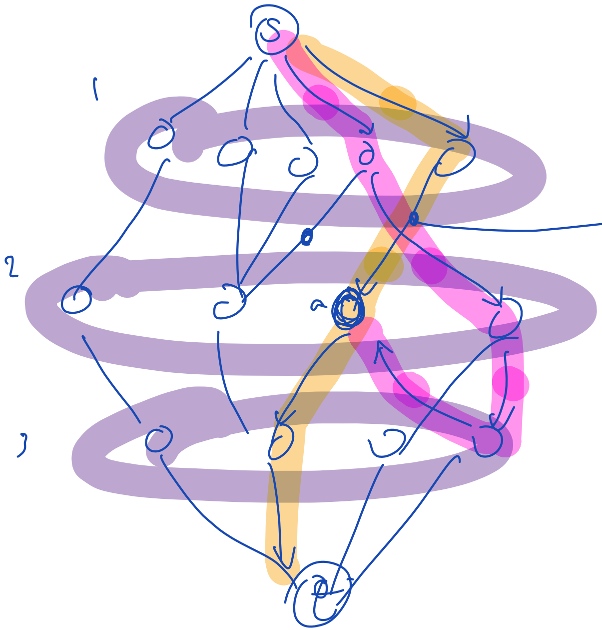
$\delta_f(s,v)$: fewest number of edges on a path from s to v in G_f

Can be computed in $\Theta(V+E)$ time using BFS.

$\delta_f(s, v)$ increases monotonically thru exec

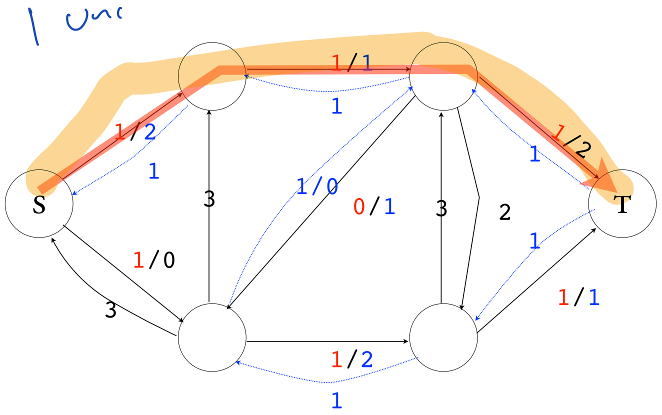
$\delta_{i+1}(v) \geq \underline{\delta_i(v)}$

G

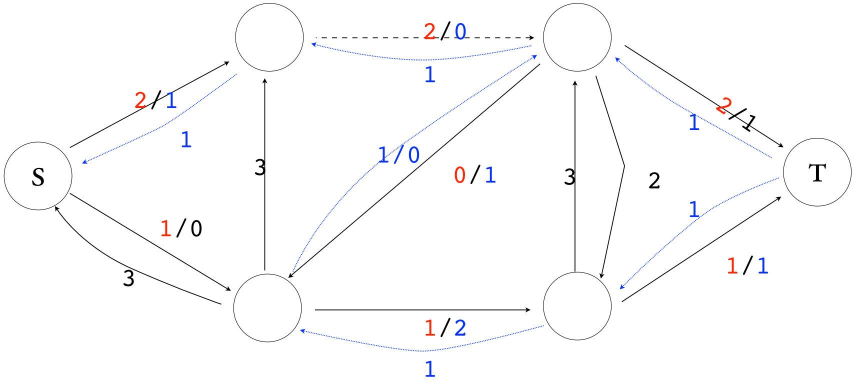


critical edge.

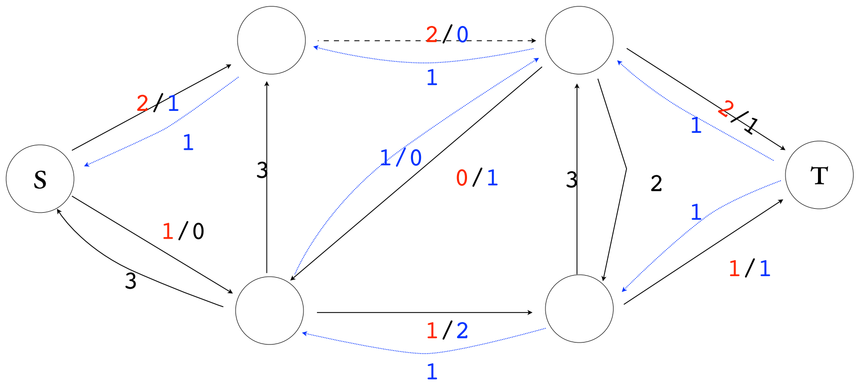
If removed, then $\delta_f(s,a)$ change from 2 to 4.



for every augmenting path, some edge is critical.



critical edges are removed in next residual graph.



key idea: how many times can an edge be critical?

We will show that an edge can only become critical $\frac{|V|}{2}$ times timeline of augmenting paths.

first time (u,v) is critical:

time i:

suppose was critical.

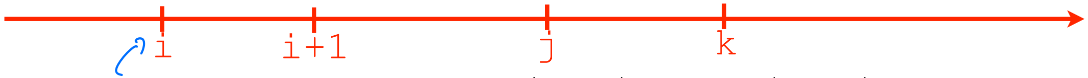
At this time

$$\delta_i(s,v) = \delta_i(s,u)+1$$

time i+1:

$$\delta_{i+1}(s,v) \geq \delta_i(s,u)+1$$

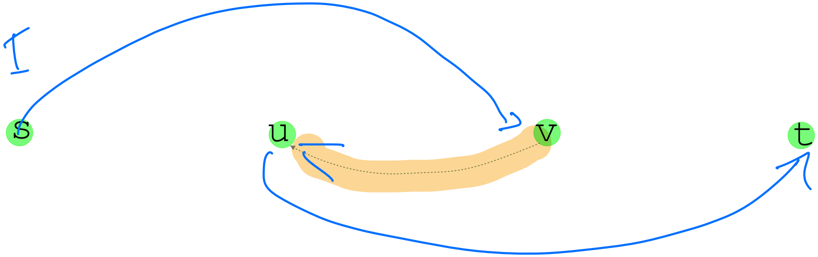
when edge (u,v) is added back



time i+1: (u,v) is critical: $$\delta_{i+1}(s,v) \geq \delta_i(s,u) + 1$$

time j: Edge (u,v) STRIKES BACK

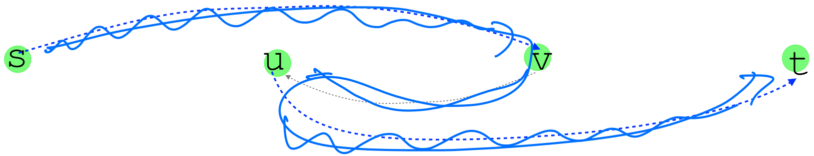
$$\delta_j(s,u) = \delta_j(s,v) + 1$$



because this path is also a shortest path.

time i+1: (u,v) is critical: $\delta_{i+1}(s,v) \geq \delta_i(s,u) + 1$

time j: Edge (u,v) STRIKES BACK



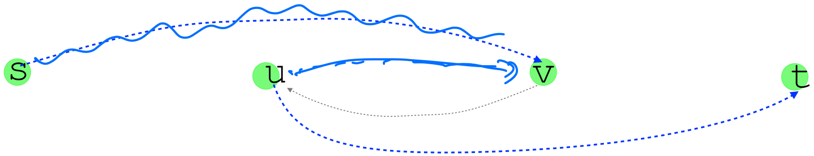
$$\delta_j(s,u) = \delta_j(s,v) + 1$$

2ND time that (u,v) was critical

time j: Edge (u,v) STRIKES BACK

$$\delta_{i+1}(s,v) \geq \delta_i(s,u) + 1$$

$$\delta_j(s,u) = \delta_j(s,v) + 1$$



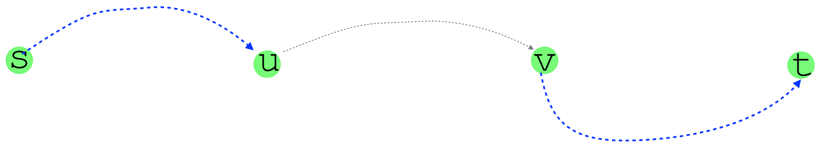
At time k,

$$\delta_k(s,u) \geq [\delta_{i+1}(s,u) + 1] + 1$$

$$\geq \delta_{i+1}(s,u) + 2$$

time k: RETURN OF THE (u,v) critical

$$\delta_k(s,u) \geq \delta_i(s,u) + 2$$



QUESTION: How many times can (u,v) be critical?

A shortest path can have at most V-1 edges.

$$\Rightarrow \left\lfloor \frac{|V|}{2} \right\rfloor$$

edge critical only $\frac{V}{2}$ times.

there are only E edges.

ergo, total # of augmenting paths: $O(EV)$

time to find an augmenting path: $O(E)$

total running time of E-K algorithm: $O(E^2V)$

FF	$O(E\|f^*\|)$
EK2	$O(E^2V)$
PUSH-RELABEL	$O(EV^2)$
FASTER PUSH-RELABEL	$O(V^3)$

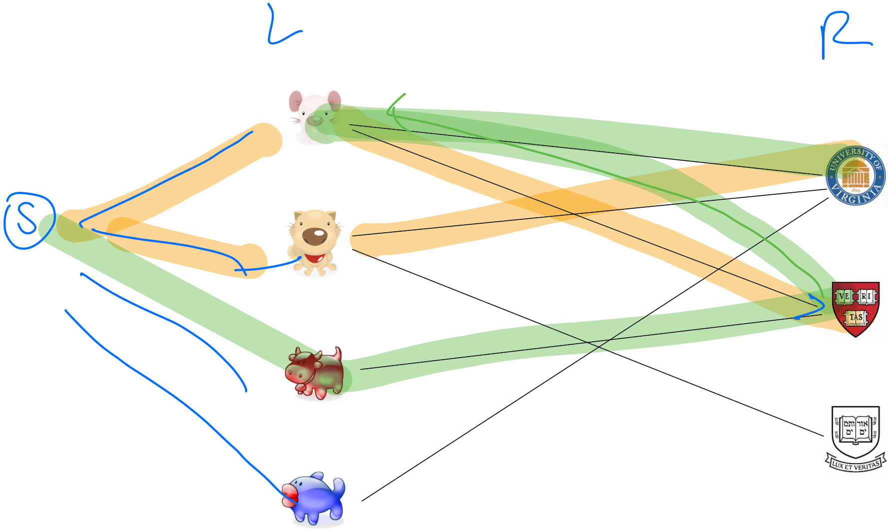
TARJAN

MAX capacity value.

GOLDBERG-RAO : $O\left(\min\{E^{2/3}, V^{1/2}\} \cdot E \cdot \log\left(\frac{V^2}{E}\right) \cdot \log(U)\right)$

Bipartite Matchings

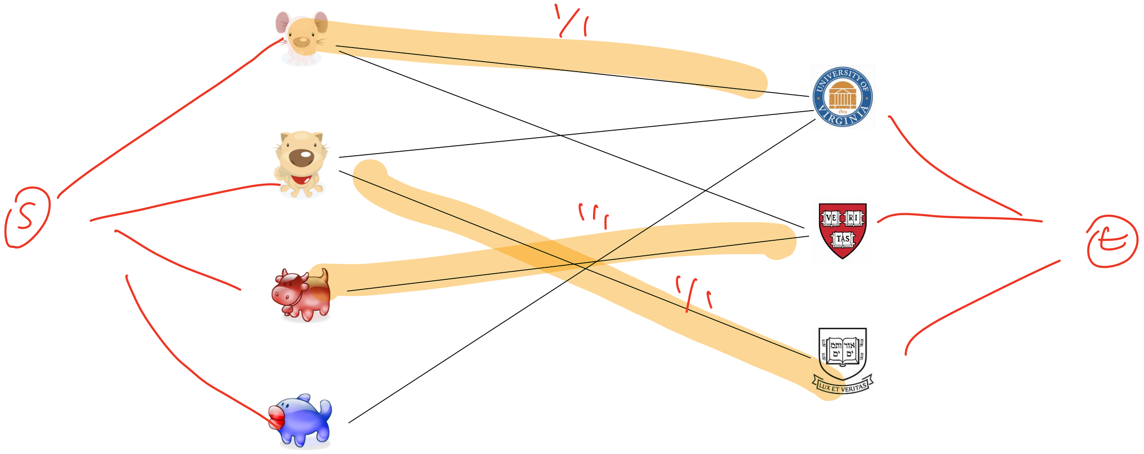
maximum bipartite matching



all edges
go between L to R

e
only 2
Students
can
match

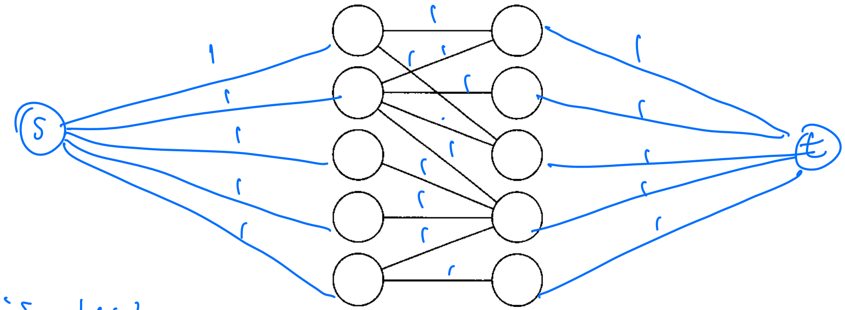
maximum bipartite matching



bipartite matching

PROBLEM: Given a bipartite graph $G=(L,R,E)$, find the largest set $M \subseteq E$ such that each node $v \in L$ or $v \in R$ is incident to at most one edge in M.

algorithm



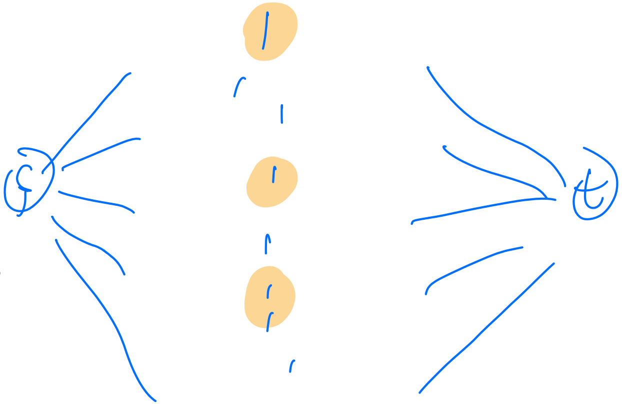
CALL This new graph G'.

Bipartite Matching(G) algorithm

1. MAKE NEW G' FROM INPUT G. CAPACITY 1.

2. RUN FF ON G'

3. OUTPUT ALL MIDDLE EDGES WITH FLOW F(E)=1.



correctness

IF G HAS A MATCHING OF SIZE K, THEN G' has a flow f, $|f| = K$.

Proof: Let M^* be a matching for G w/ $|M| = K$.

Construct a flow f for G' s.t. $|f| = K$.

$f(e) = 1$ if $e = (u,v) \in M$.

$f(s,u) = 1$ if $e = (u,v) \in M$

$f(v,t) = 1$ if $e = (u,v) \in M$

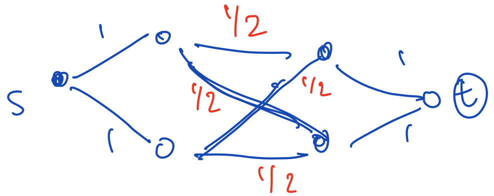
① This flow satisfies the capacity + flow constraints

② $|f| = K$ by inspection.

correctness

IF G' HAS A FLOW OF K, THEN G has a matching of size K.

<u>Proof idea:</u> add all of the edges from L to R that have $f(e)=1$ to the MATCHING. By flow constraint, each node on the left can only have inflow 1, and thus can only have 1 unit of outflow.



FF integrality theorem

IF CAPACITIES ARE ALL INTEGRAL, THEN the FF maxflow will have integral values.

Proof: By induction. At the start of FF, the flow f is 0 and therefore integral.

Suppose the flow was integral after i steps.

[ToDo at home: argue why on the next step, the flow will still be integral].

correctness

IF G' HAS A FLOW OF K, THEN G HAS K-MATCHING.

① At end of FF, the max flow for the graph G' is integral.

Define the set $M = \{ e \mid f(e) = 1 \text{ and } e = (x,y), x \in L, y \in R \}$

. Because $c(e) = 1$, by the capacity constraint

$f(e) \leq c(e)$, and so by integrality, $f(e) = 0$ or 1.

Thus for all $v \in L$, v is incident to at most one edge in M.

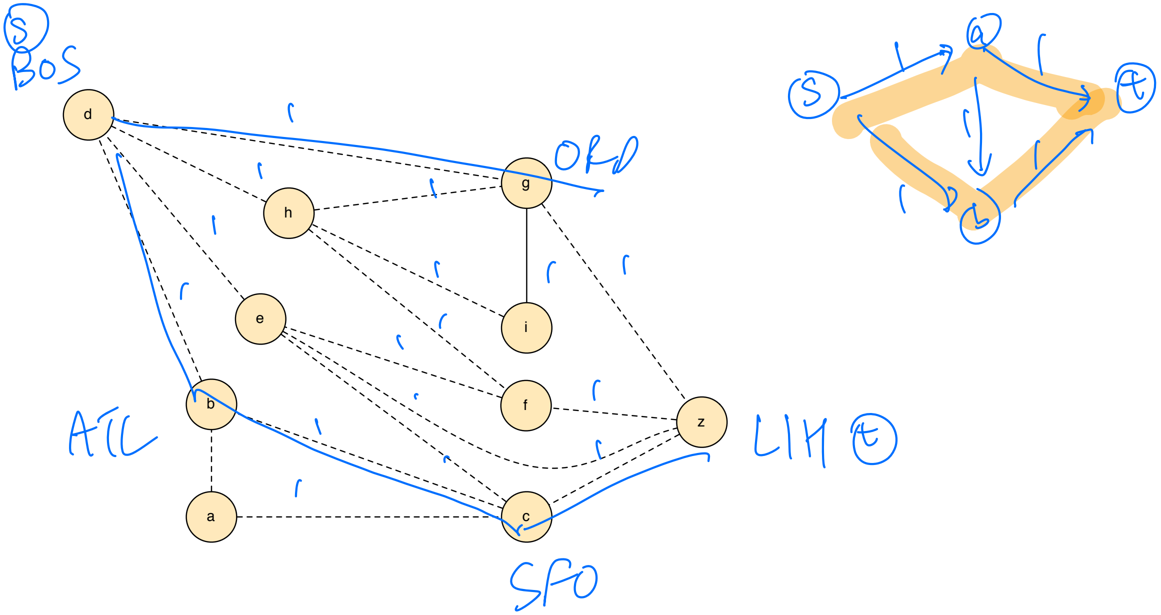
By the min cut, $|M| = k$. by defining S, T as we did before.

running time

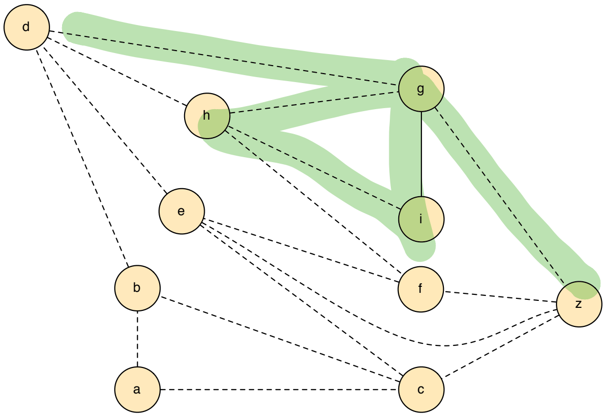
$O(E \cdot |f|)$ but we know that $|f| = O(V)$

$O(E \cdot V)$

edge-disjoint paths



algorithm



1. Compute max flow
2. Remove all edges with f(e) = 0.
3. Walk from s.
   1. If you reach a node you have visited before, erase flow along path
   2. If you reach t, add this path to your set, erase flow along path.

analysis

IF G HAS K DISJOINT PATHS, THEN

analysis

IF G' HAS A FLOW OF K, THEN

vertex-disjoint paths