Network Flow Problems

<u>**Network**</u> N: <u>**digraph**</u> $N = (V, E)$ with

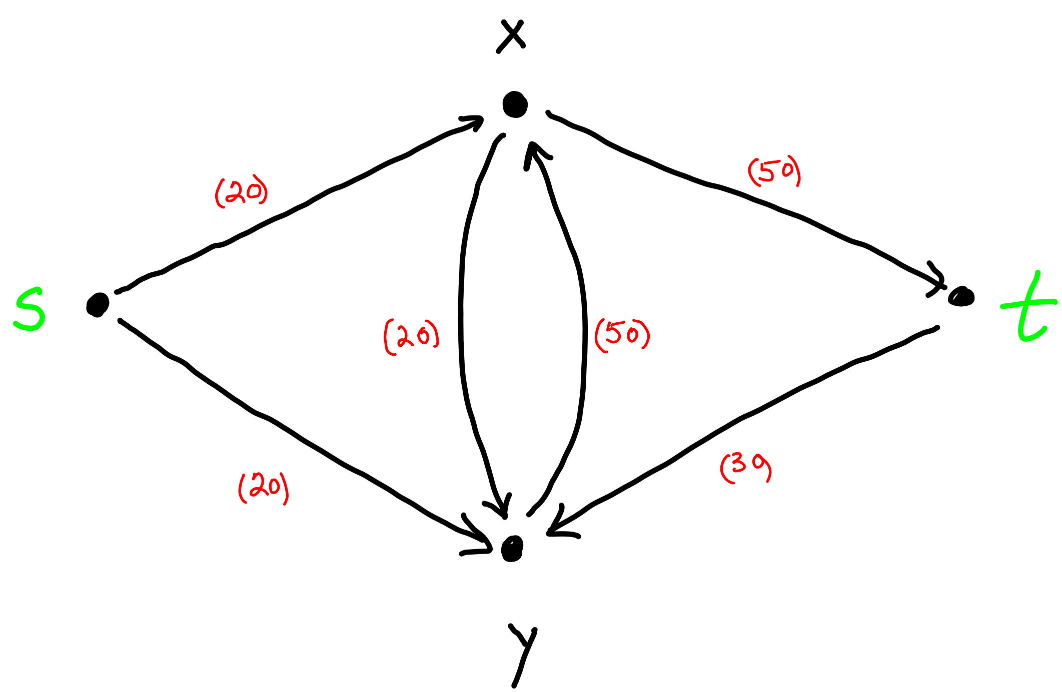
<u>**capacity**</u> $c(e) \geq 0$ on each edge,

<u>**source**</u> $s \in V$, and

<u>**sink**</u> $t \in V$.

<u>**Flow**</u> on N: assignment $f(e)$ to each edge.

Network N :



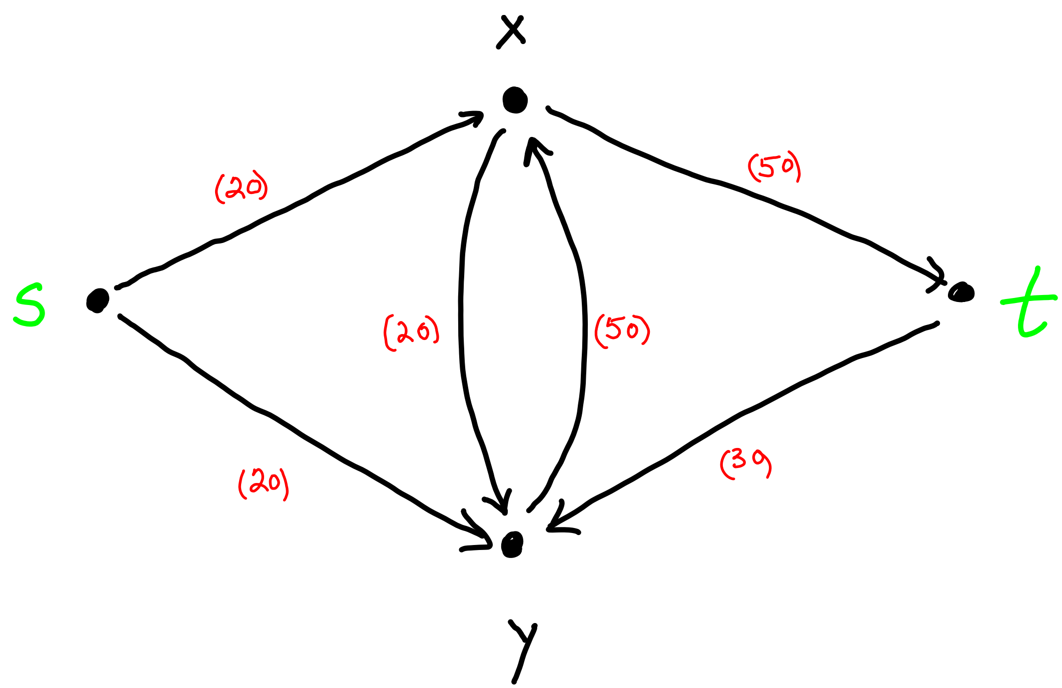
<u>Network</u> N: <u>digraph</u> $N = (V, E)$ with

<u>capacity</u> $c(e) \geq 0$ on each edge,

<u>source</u> $s \in V$, and

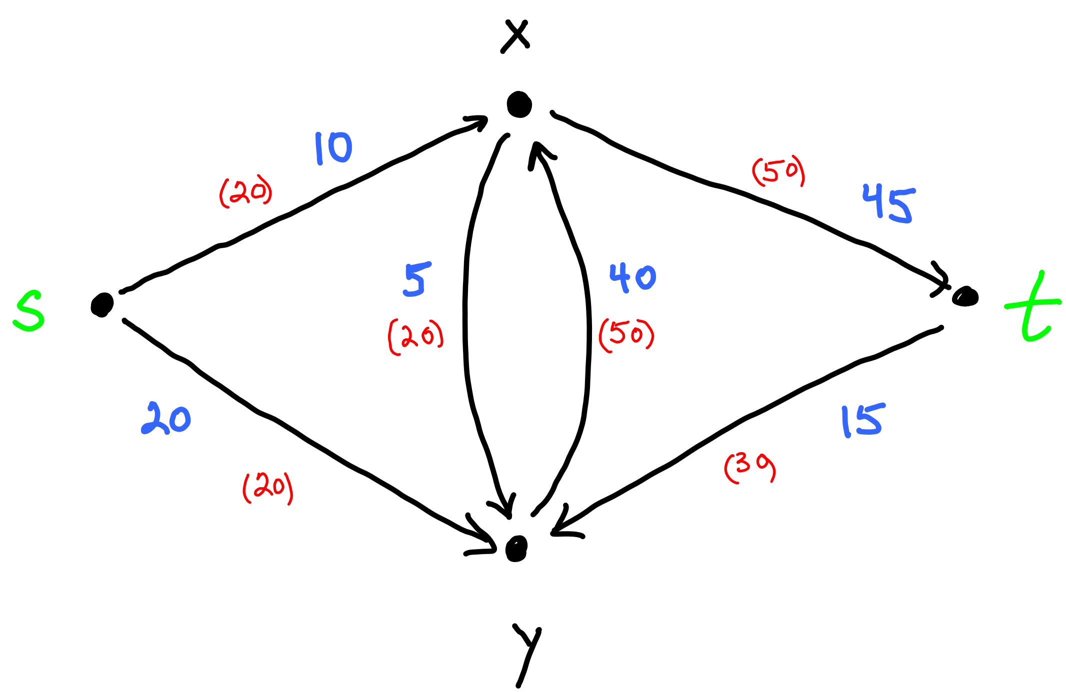
<u>sink</u> $t \in V$.

Network N :



Network N :

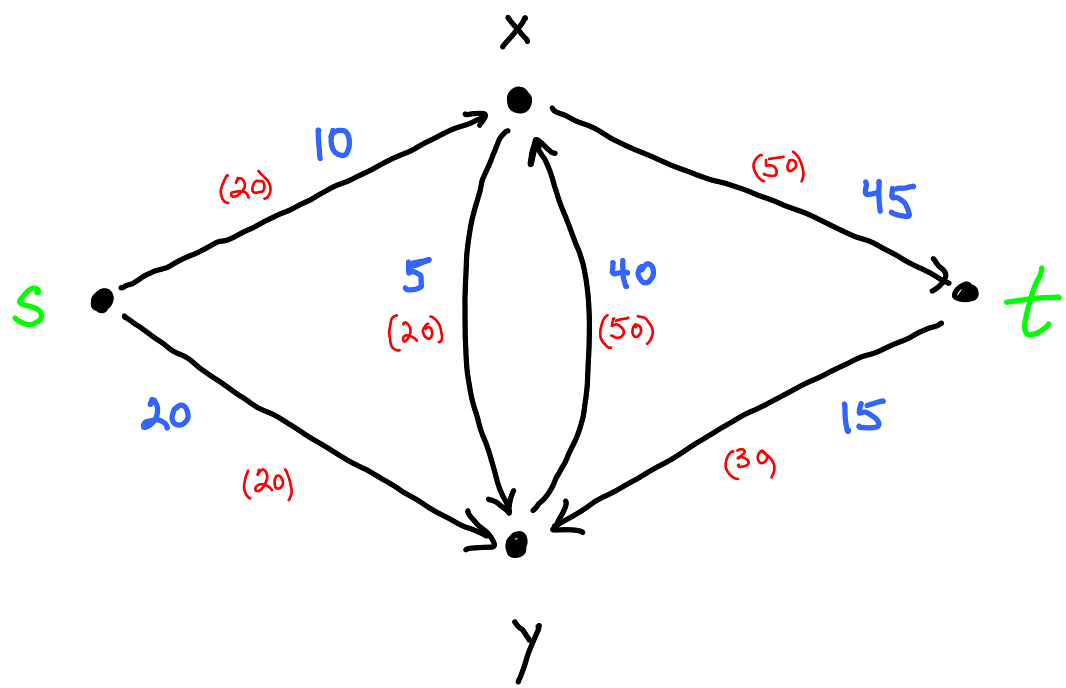
flow f



<u>Flow</u> on N: assignment $f(e)$ to each edge.

Network N :

flow f

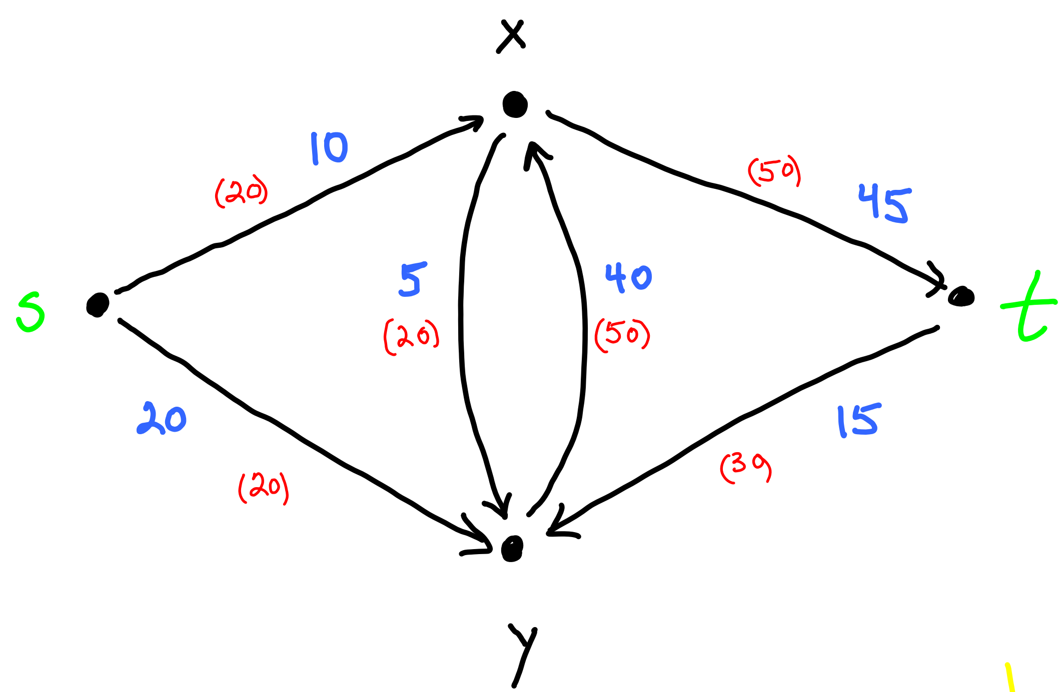


$f^+(v)$ = sum of flow on edges leaving v

$f^-(v)$ = sum of flow on edges entering v

Network N :

flow f



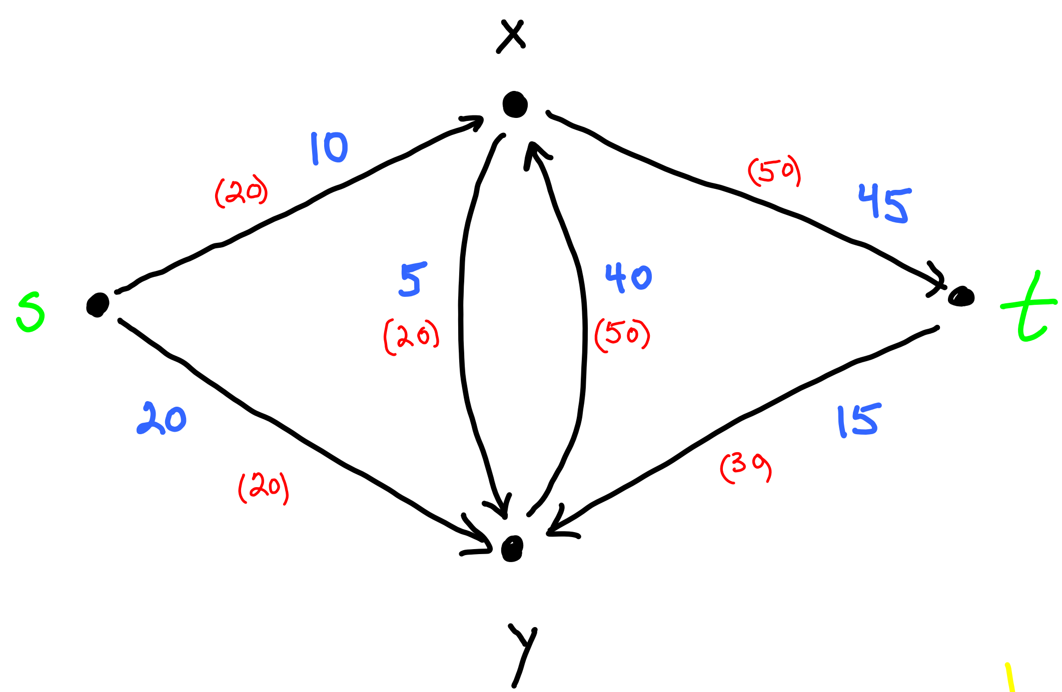
	f^+	f^-
x	50	50
y	40	40
s	30	0
t	15	45

$f^+(v)$ = sum of flow on edges leaving v

$f^-(v)$ = sum of flow on edges entering v

Network N :

flow f



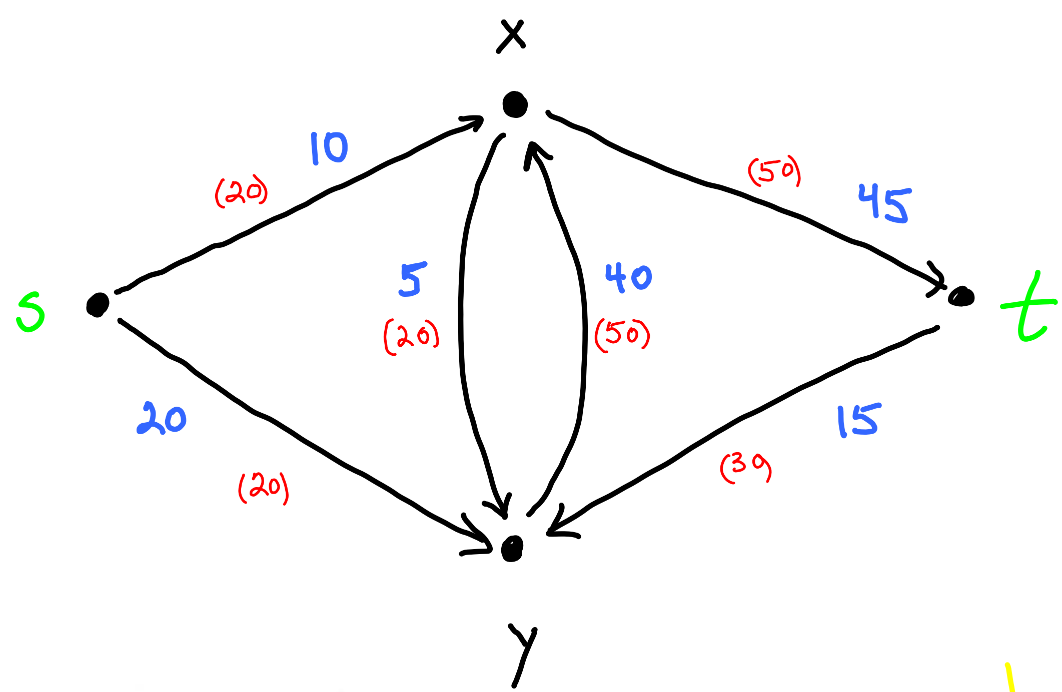
	f^+	f^-
x	50	50
y	40	40
s	30	0
t	15	45

$f^+(v)$ = sum of flow on edges leaving v

$f^-(v)$ = sum of flow on edges entering v

Network N:

flow f



Flow f is __feasible__ if both

$0 \le f(e) \le c(e) \quad \forall e \in E$

NON-NEGATIVITY CONSTRAINT

CAPACITY CONSTRAINT

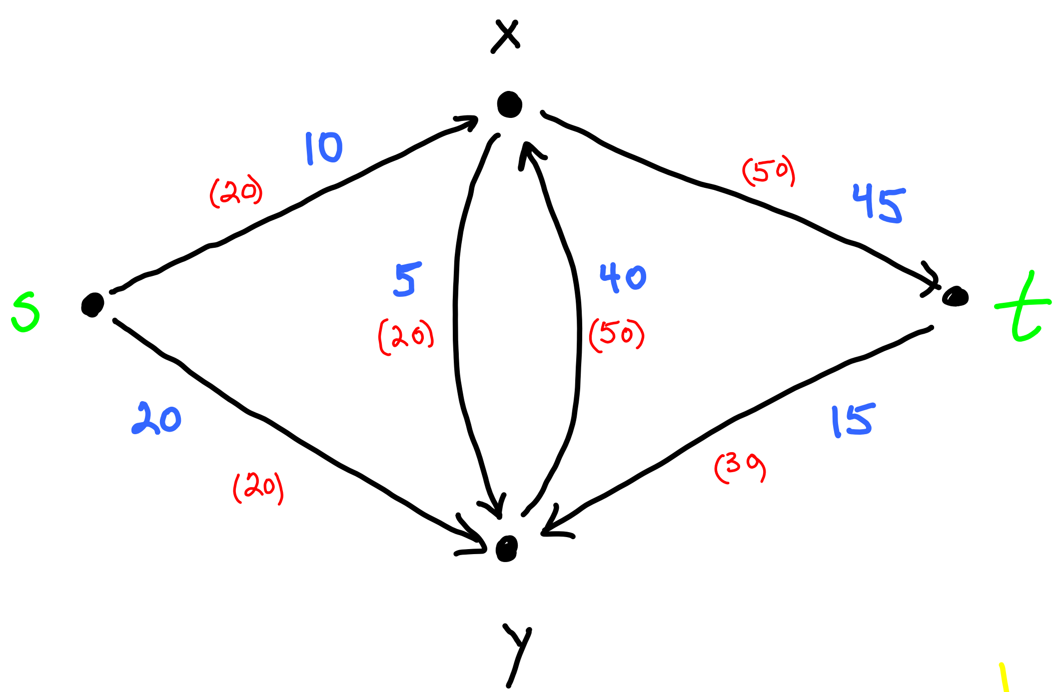
$f^+(v) = f^-(v) \quad \forall v \in V - \{s,t\}$

CONSERVATION OF FLOW

	f^+	f^-
x	50	50
y	40	40
s	30	0
t	15	45

Network N :

flow f



	f^+	f^-
x	50	50
y	40	40
s	30	0
t	15	45

Value of flow, f :

$$\text{val}(f) = f^-(t) - f^+(t)$$

$$= 45 - 15 = 30$$

Maximum Flow Problem : Find a feasible flow of maximum value.

Main tools :
- flow augmenting paths
- source/sink cuts (directed edge cuts)

Preview :
- If N has an f-augmenting path, f is not a max. flow
- The value of any feasiblle flow is less than or equal to the capacity of any source/sink cut.
- The value of the maximum flow is equal to the capacity of the minimum source/sink cut.

N network, f feasible flow

p - <u>undirected</u> path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

$\underline{tolerance(p)} = \min\{\epsilon(e) \mid e \text{ on } p\}$

Example:

$$p: \quad s \xrightarrow{\overset{(20)}{10}} x \xrightarrow{\overset{(20)}{0}} y \xleftarrow{20} t$$

N network, f feasible flow

p - <u>undirected</u> path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

<u>$tolerance(p)$</u> $= \min\{\epsilon(e) \mid e \text{ on } p\}$

Example:

$$p: \quad s \xrightarrow{\overset{(20)}{10}} x \xrightarrow{\overset{(20)}{0}} y \xleftarrow{20} t$$

$20-10 = 10 \qquad 20-0 = 20$

N network, f feasible flow

p - undirected path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

$\underline{tolerance(p)} = \min\{\epsilon(e) \mid e \text{ on } p\}$

Example:

$$p: \quad s \xrightarrow{\overset{(20)}{10}} x \xrightarrow{\overset{(20)}{0}} y \xleftarrow{20} t$$

$$20-10 = 10 \qquad 20-0 = 20 \qquad 20$$

N network, f feasible flow

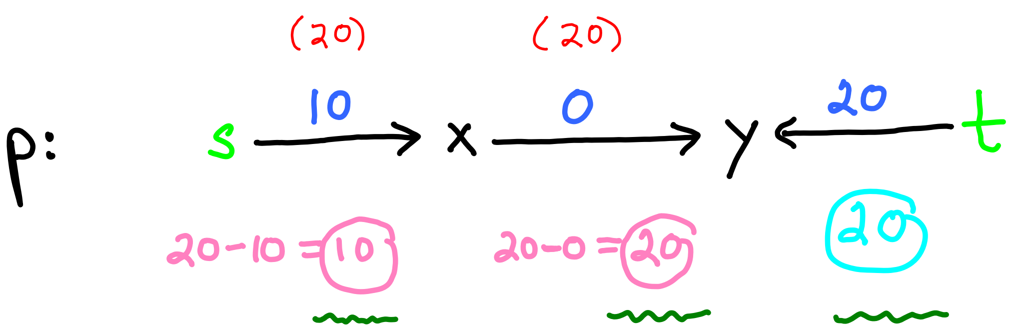
p - undirected path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

$tolerance(p) = \min\{\epsilon(e) \mid e \text{ on } p\}$

Example:



tolerance(p) = 10

N network, f feasible flow

p - undirected path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

$$tolerance(p) = \min\{\epsilon(e) \mid e \text{ on } p\}$$

Example:

f

$$p: \quad s \xrightarrow[+10]{\overset{(20)}{10}} x \xrightarrow[+10]{\overset{(20)}{0}} y \xleftarrow[-10]{20} t$$

$tolerance(p) = 10$

add to forward edges
subtract from backward edges

N network, f feasible flow

p - undirected path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

$tolerance(p) = \min\{\epsilon(e) \mid e \text{ on } p\}$

Example:

$$p: \quad f \quad s \xrightarrow[+10]{\substack{(20)\\ 10}} x \xrightarrow[+10]{\substack{(20)\\ 0}} y \xleftarrow[-10]{20} t$$

$tolerance(p) = 10$

$$p: \quad \boxed{f'} \quad s \xrightarrow{\substack{(20)\\ 20}} x \xrightarrow{\substack{(20)\\ 10}} y \xleftarrow{10} t$$

$\boxed{f'}$ still feasible (why?)

and

$val(f') = val(f) + 10$

N network, f feasible flow

p - undirected path in N. Define

$\epsilon(e) = c(e) - f(e)$ on forward edges of p.

$\epsilon(e) = f(e)$ on all backward edges of p.

$tolerance(p) = \min\{\epsilon(e) \mid e \text{ on } p\}$

Example:

f

$$p: \quad s \xrightarrow[+10]{(20)\ 10} x \xrightarrow[+10]{(20)\ 0} y \xleftarrow[-10]{20} t \qquad tolerance(p) = 10$$

f-augmenting path: an undirected s,t-path, p, with $tolerance(p) > 0$.

Lemma 4.3.5. If p is an f-augmenting path with tolerance z, then increasing flow by z along forward edges of p and decreasing flow by z along backward edges of p produces a feasible flow, f' with $val(f') = val(f) + z$.

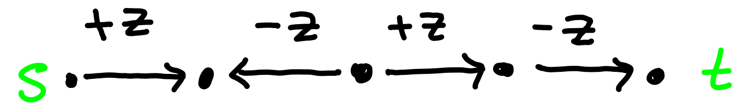
$$s \bullet \xrightarrow{+z} \bullet \xleftarrow{-z} \bullet \xrightarrow{+z} \bullet \xrightarrow{+z} \bullet\ t$$

Proof. By definition, f' satisfies non-negativity and capacity constraint.

$val(f') = val(f) + z$ since on p:

$$\left\{ \begin{array}{l} \cdots\cdots\cdots \bullet \xrightarrow{+z} \bullet\ t \quad \text{or} \\ \cdots\cdots\cdots \bullet \xleftarrow{-z} \bullet\ t \end{array} \right.$$

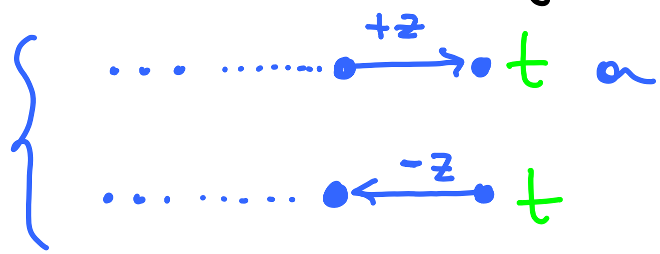
Lemma 4.3.5. If p is an f-augmenting path with tolerance z, then increasing flow by z along forward edges of p and decreasing flow by z along backward edges of p produces a feasible flow, f' with $val(f') = val(f) + z$.



Proof. By definition, f' satisfies non-negativity and capacity constraint.

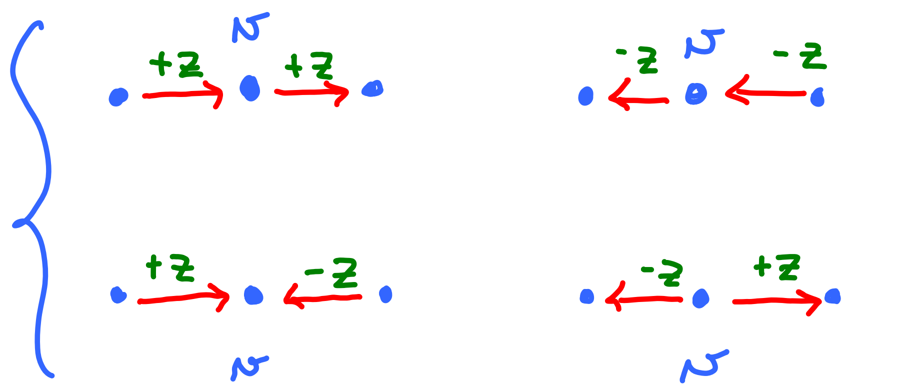
$val(f') = val(f) + z$ since

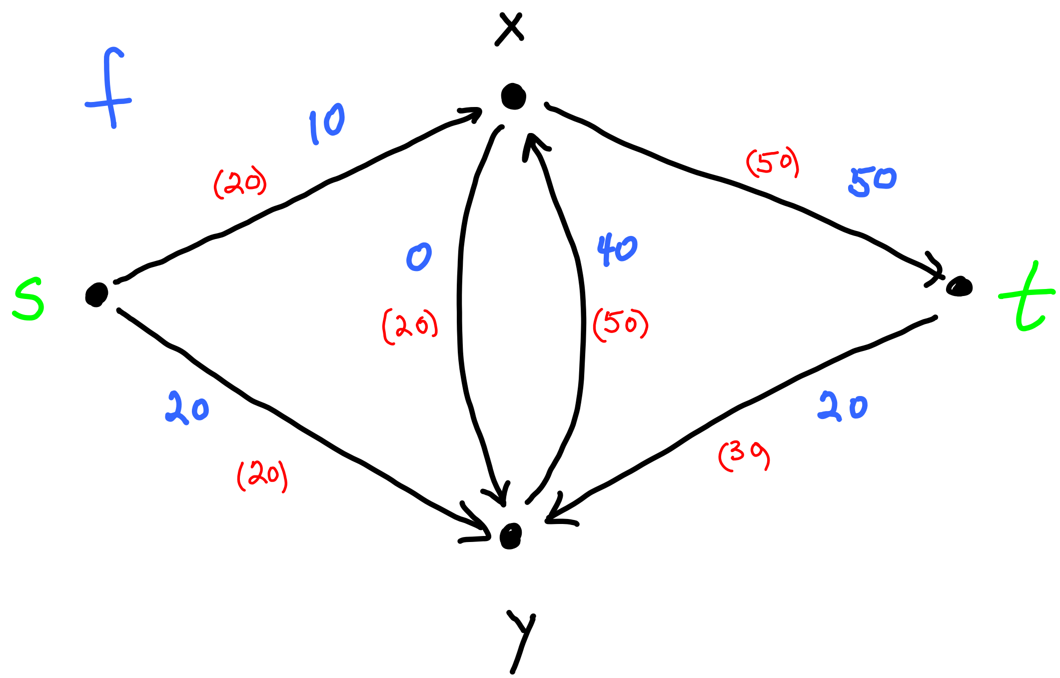
on p:



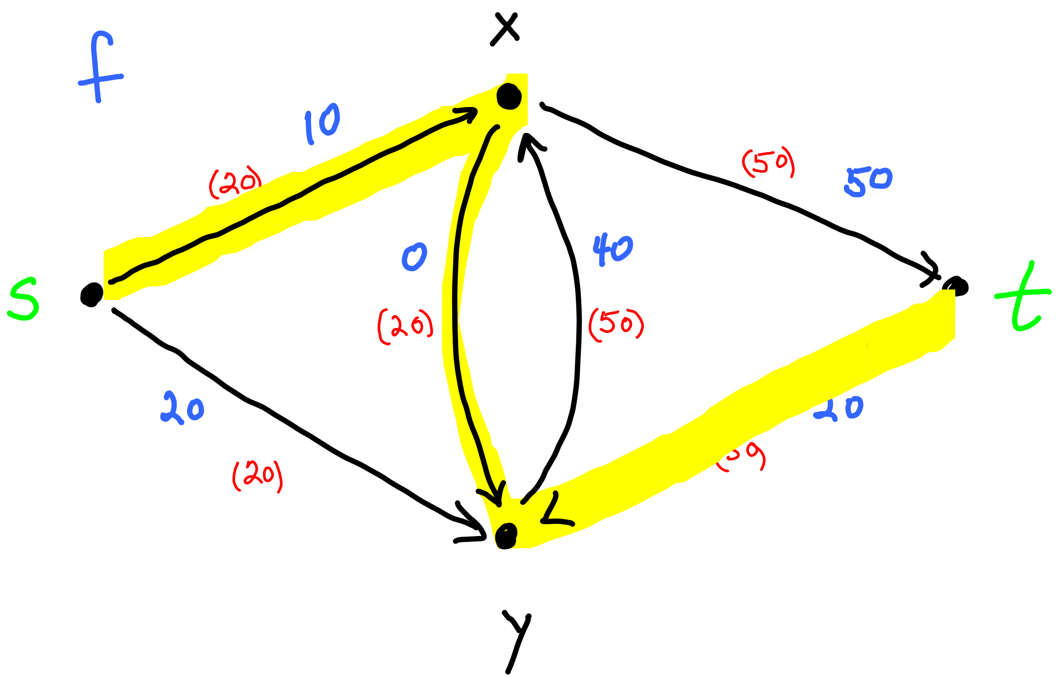
So, it remains to show flow is conserved at internal vertices of p.

There are 4 cases:



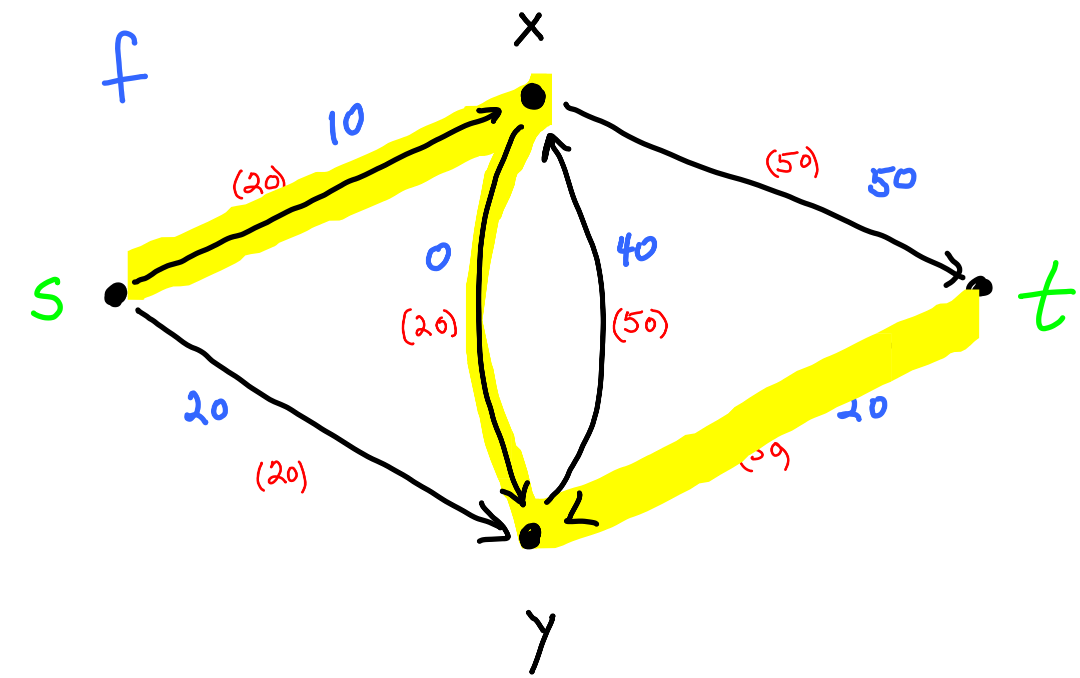


val (f) = 30



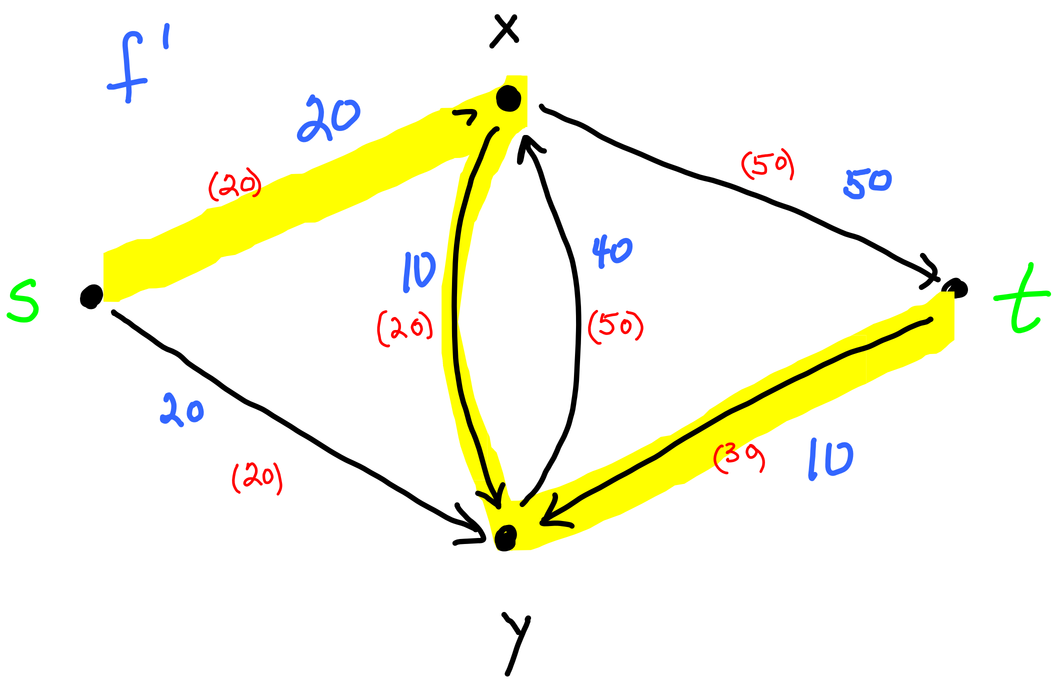
val (f) = 30

tolerance (ρ) = 10

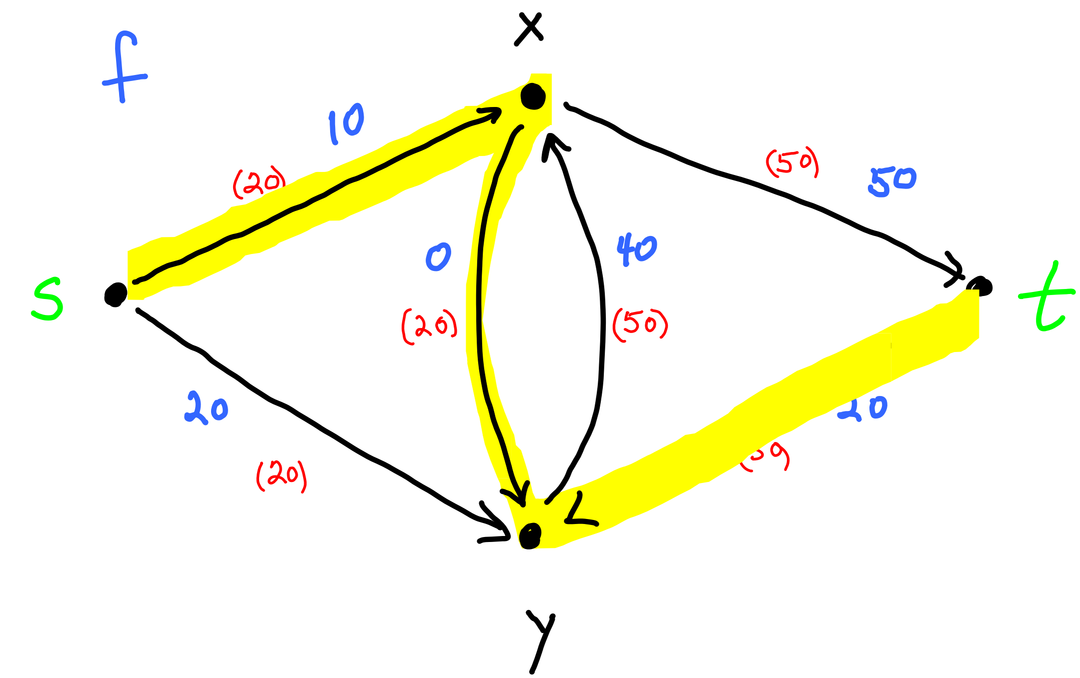


$val(f) = 30$

$tolerance(p) = 10$

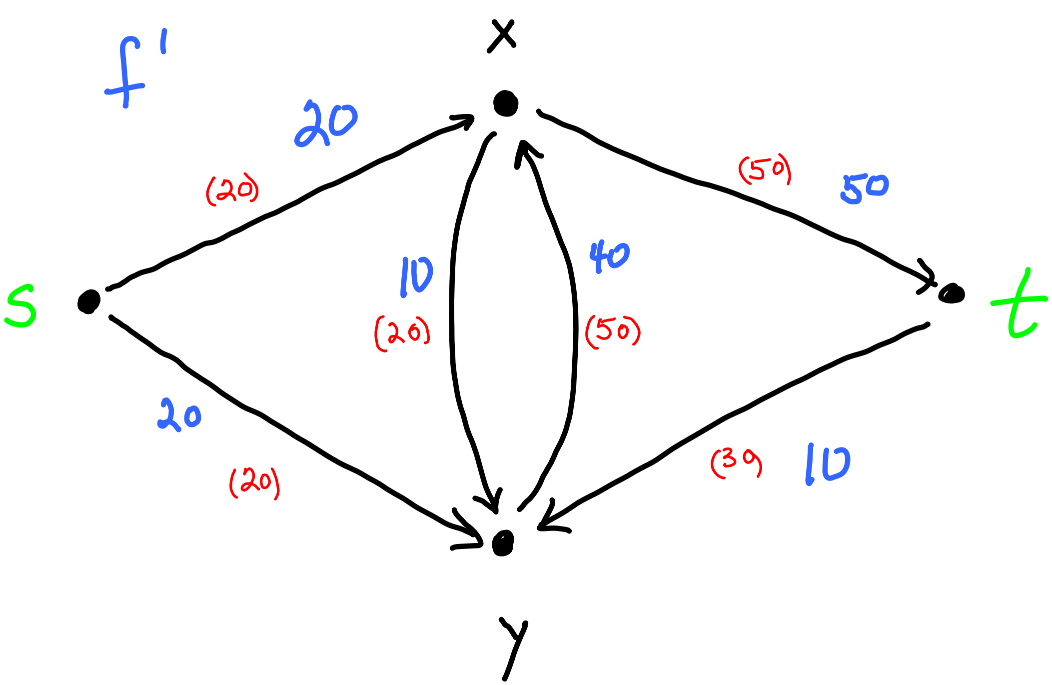


$val(f') = 40$



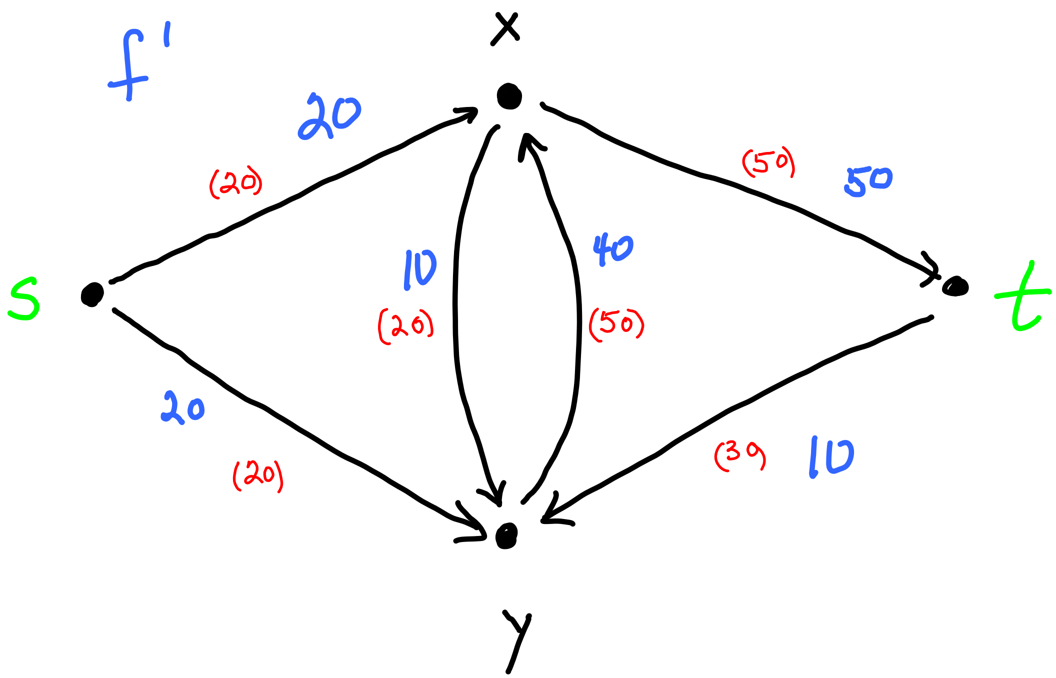
val(f) = 30

tolerance(p) = 10



val(f') = 40

Now . . .



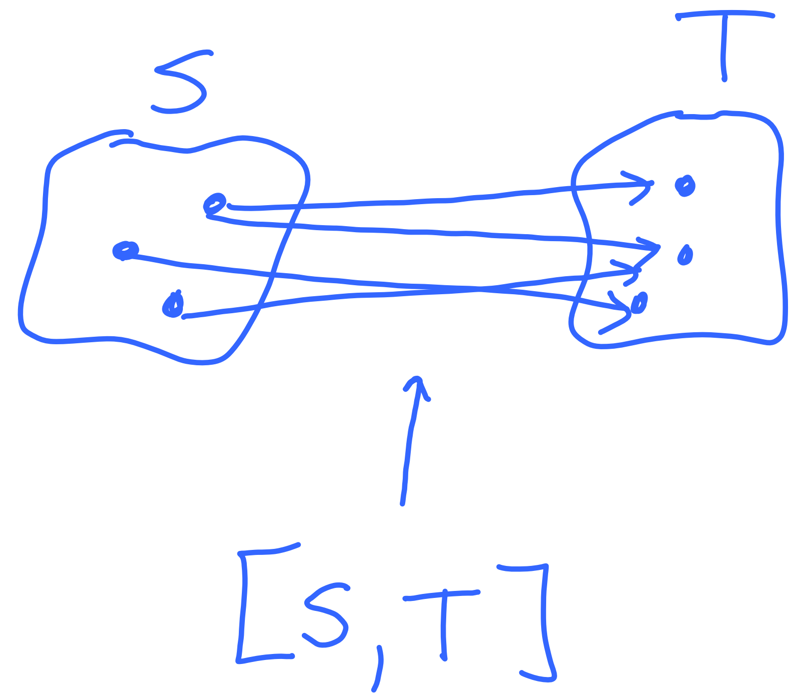
val (f') = 40

Is f maximum?

How to know?

Def. Let S and T be two sets of vertices in *digraph* D.

Then $[S,T]$ is the set of all edges in D whose tail is in S and head is in T.

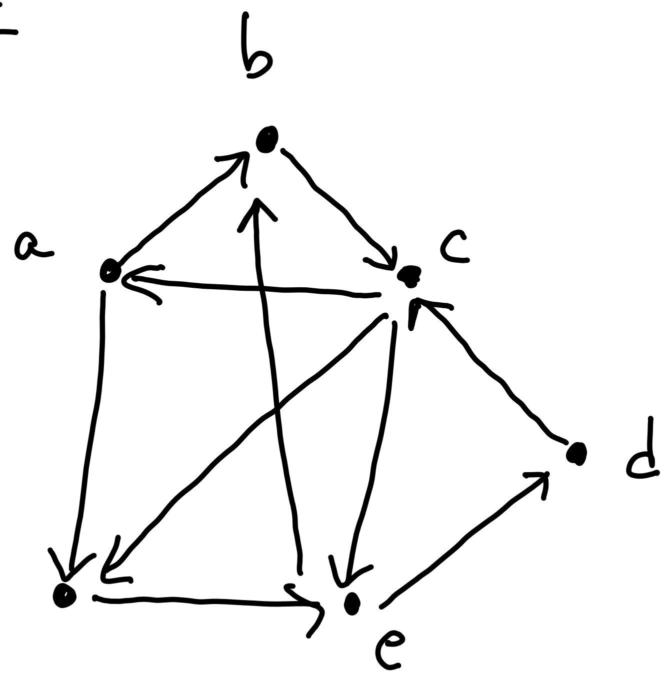


<u>Def</u>. Let S and T be two sets of vertices in <u>digraph</u> D.

Then $[S, T]$ is the set of all edges in D whose tail is in S and head is in T.

<u>Example</u>

D



Def. Let S and T be two sets of vertices in digraph D.

Then $[S,T]$ is the set of all edges in D whose tail is in S and head is in T.

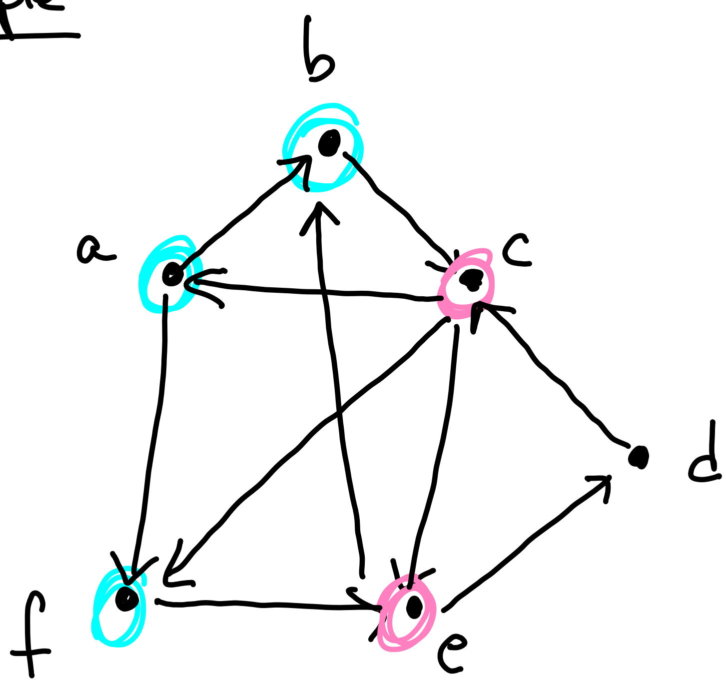
Example

Let

$S = \{a, b, f\}$

$T = \{c, e\}$

D



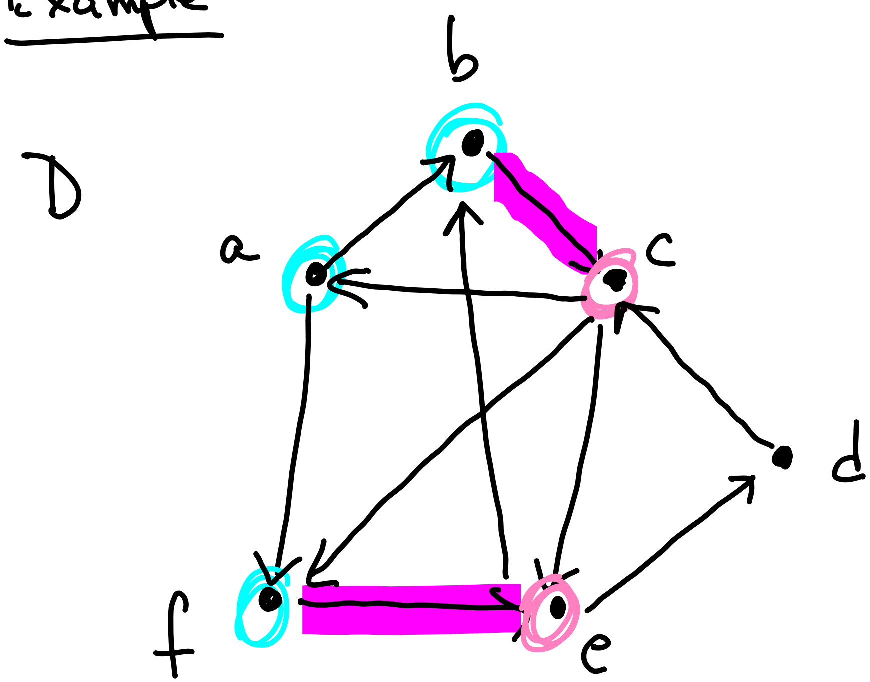
Then

$[S, T] = \{ \qquad \}$

Def. Let S and T be two sets of vertices in digraph D.

Then $[S, T]$ is the set of all edges in D whose tail is in S and head is in T.

Example



Let

$S = \{a, b, f\}$

$T = \{c, e\}$

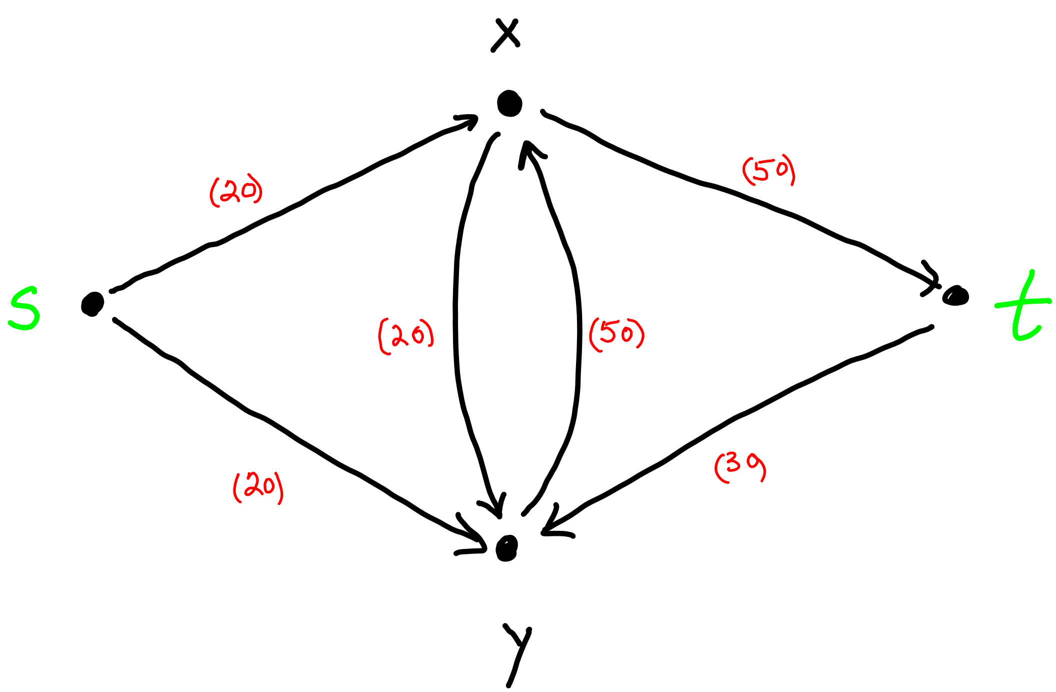
Then

$[S, T] = \{bc, fe\}$

In a network, N, a <u>source/sink cut</u> is a set of edges

$[S,T]$, where $\boxed{s \in S,\ t \in T}$ and $\boxed{S \cup T = V(N) \quad S \cap T = \emptyset}$

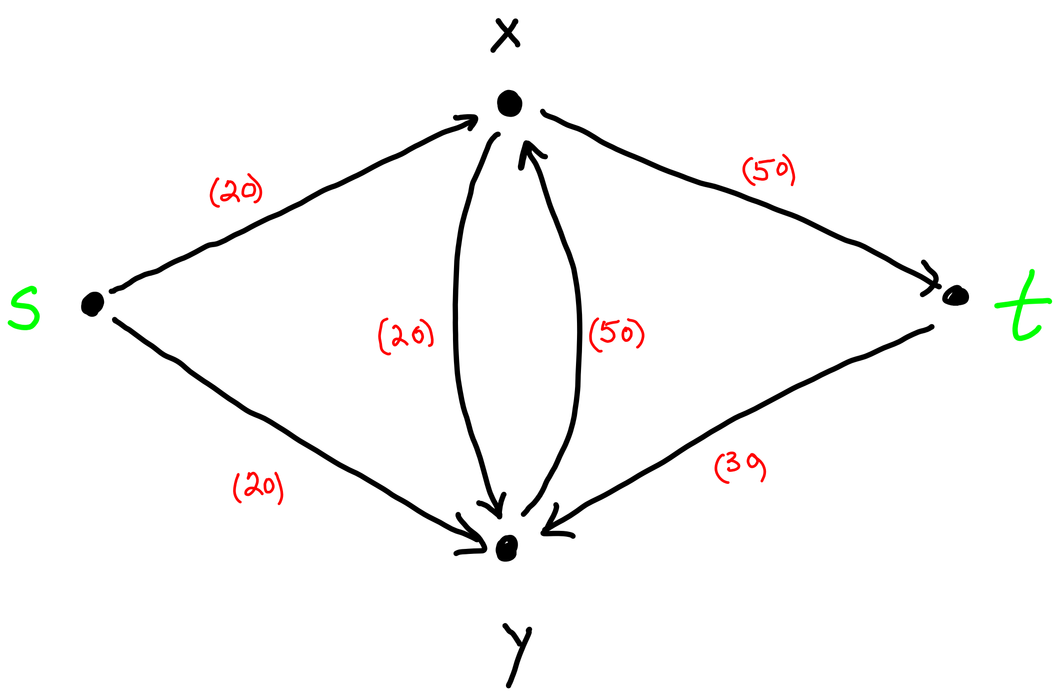
The capacity of a cut, $c(S,T)$, is the sum of its edge capacities



In a network, N, a source/sink cut is a set of edges

$[S,T]$, where $\boxed{s \in S,\ t \in T}$ and $\boxed{S \cup T = V(N) \quad S \cap T = \emptyset}$

The capacity of a cut, $c(S,T)$, is the sum of its edge capacities



source/sink cut	capacity
$[\{s\}, \{x, y, t\}]$	40
$[\{s, x\}, \{y, t\}]$	90
$[\{s, y\}, \{x, t\}]$	70
$[\{s, x, y\}, \{t\}]$	50

Source-sink cut $[S,T]$

Define:

$$f^+(S) = \sum_{e \in [S,T]} f(e) \qquad f^-(S) = \sum_{e \in [T,S]} f(e)$$

$$f^+(T) = f^-(S) \qquad f^-(T) = f^+(S)$$

4.3.8 Let f be any feasible flow in N

Let $[S,T]$ be any source/sink cut.

☆

Then:

$$\text{val}(f) \le \text{cap}(S,T)$$

Proof outline: Show

$$\text{val}(f) = \sum_{v \in T} \left(f^-(v) - f^+(v) \right) \qquad \text{(a)}$$

$$= f^-(T) - f^+(T) \qquad \text{(b)}$$

$$= f^+(S) - f^-(S) \qquad \text{(c)}$$

$$\le \text{cap}(S,T) \qquad \text{(d)}$$

(a) $\mathrm{val}(f) = \sum_{v \in T} \boxed{f^-(v) - f^+(v)}$

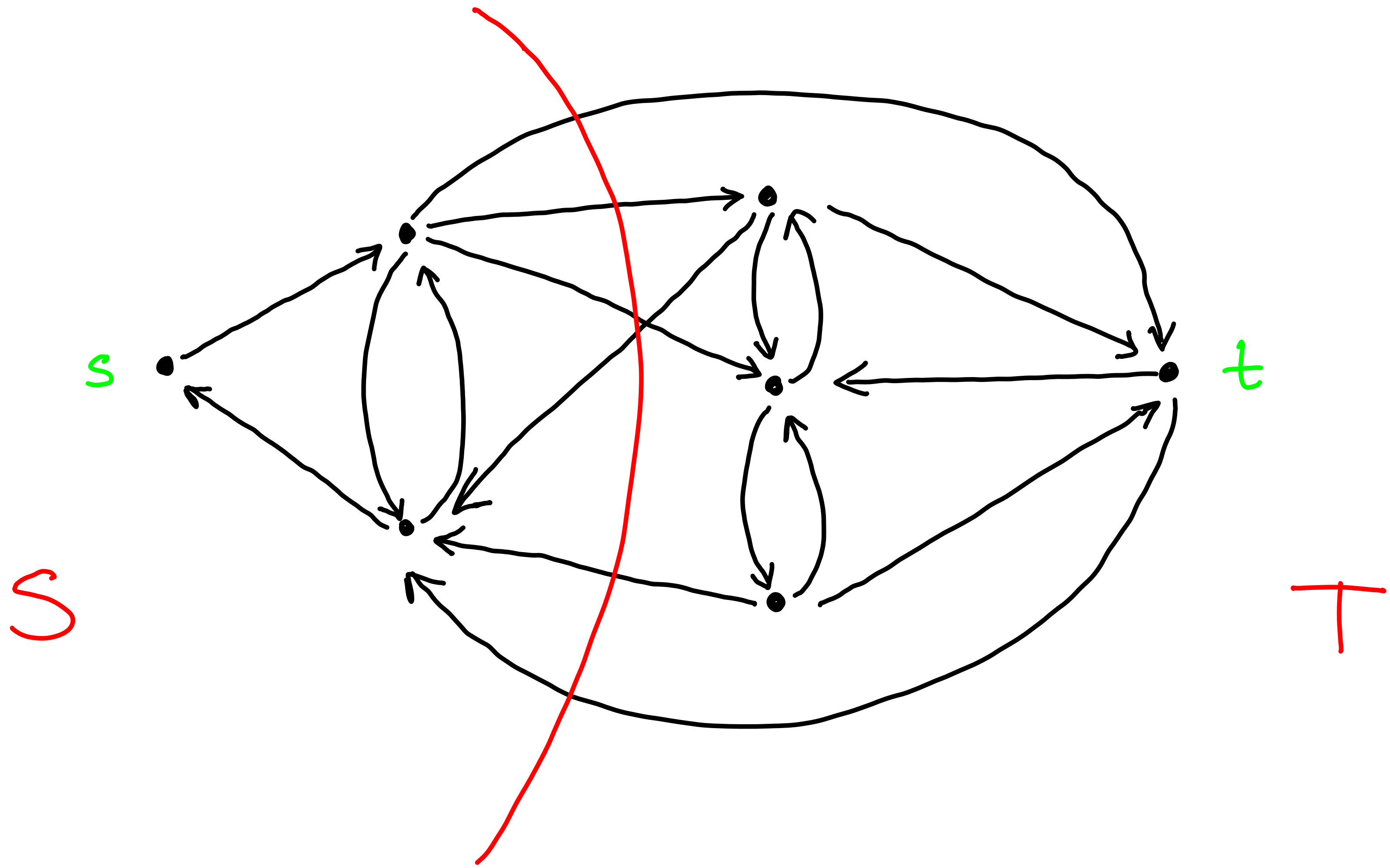
by conservation of flow, this is 0, unless $v = t$

$= f^-(t) - f^+(t)$

by definition

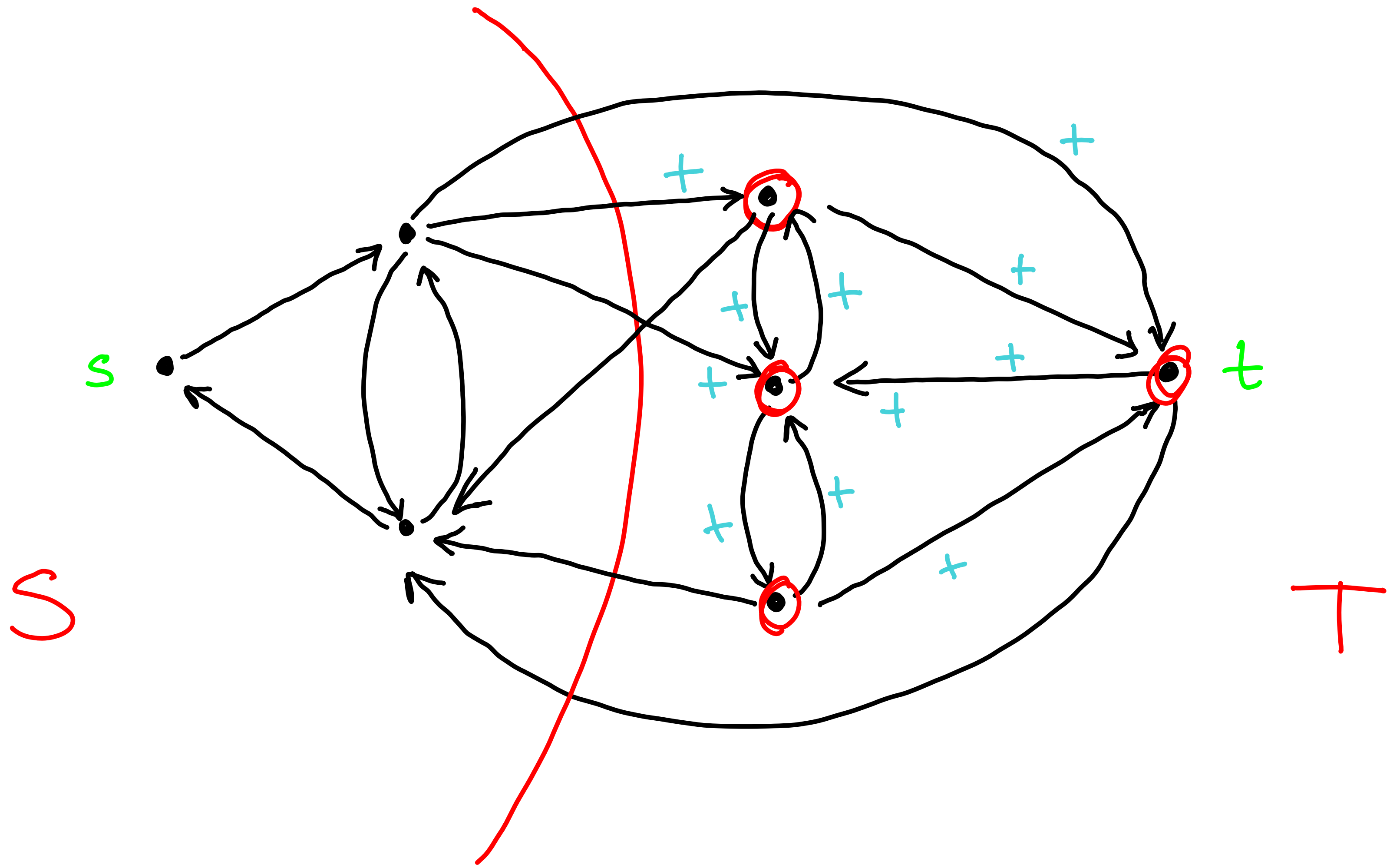
✓

(b) $\displaystyle\sum_{v \in T} \left(f^-(v) - f^+(v) \right) = f^-(T) - f^+(T)$



(b) $\sum_{v \in T} \underline{f^-(v)} - f^+(v) = f^-(T) - f^+(T)$

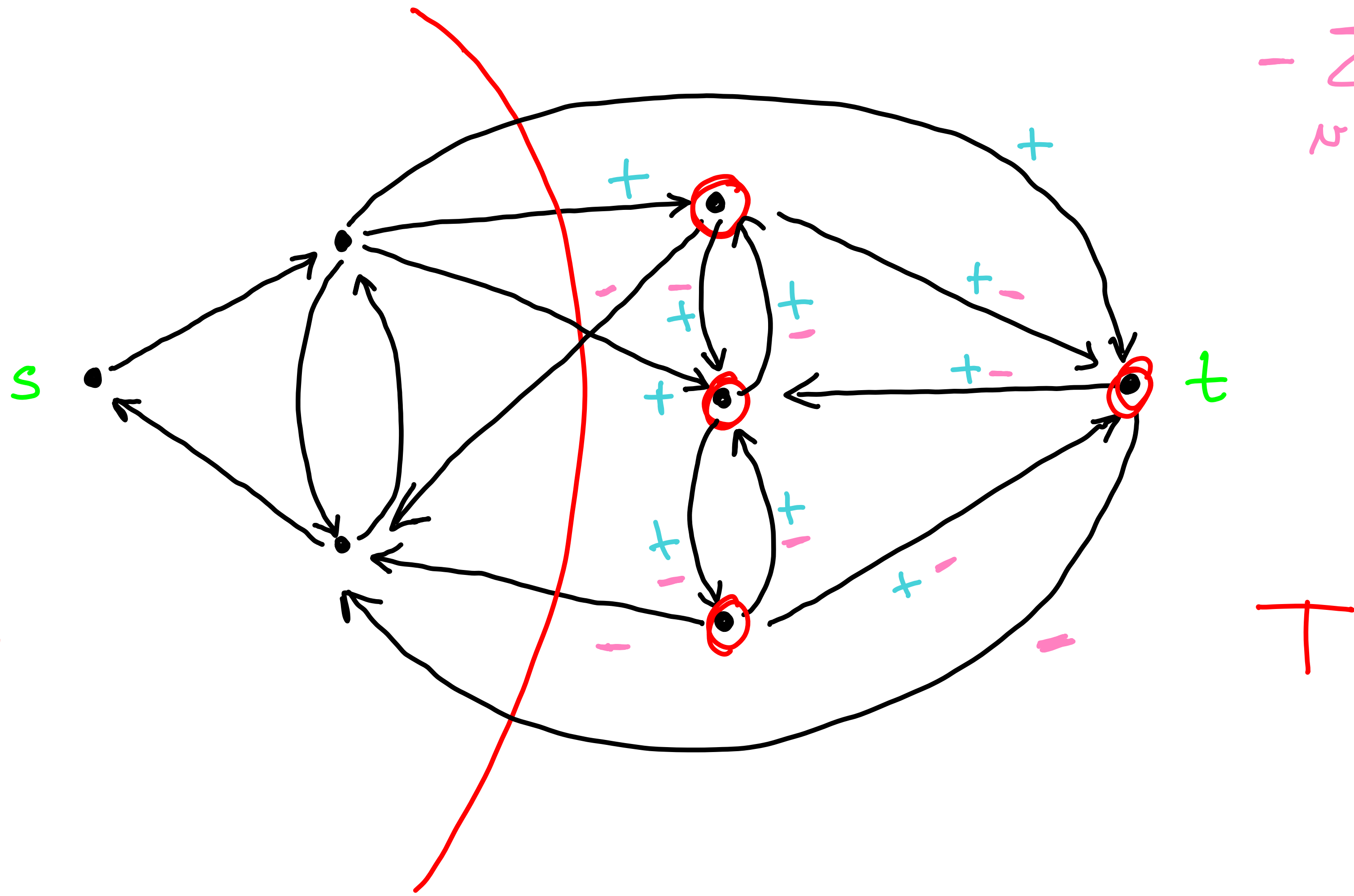
$\sum_{v \in T} f^-(v)$



(b) $\sum_{v \in T} \underline{f^-(v)} - \underline{f^+(v)} = f^-(T) - f^+(T)$

$\sum_{v \in T} f^-(v)$

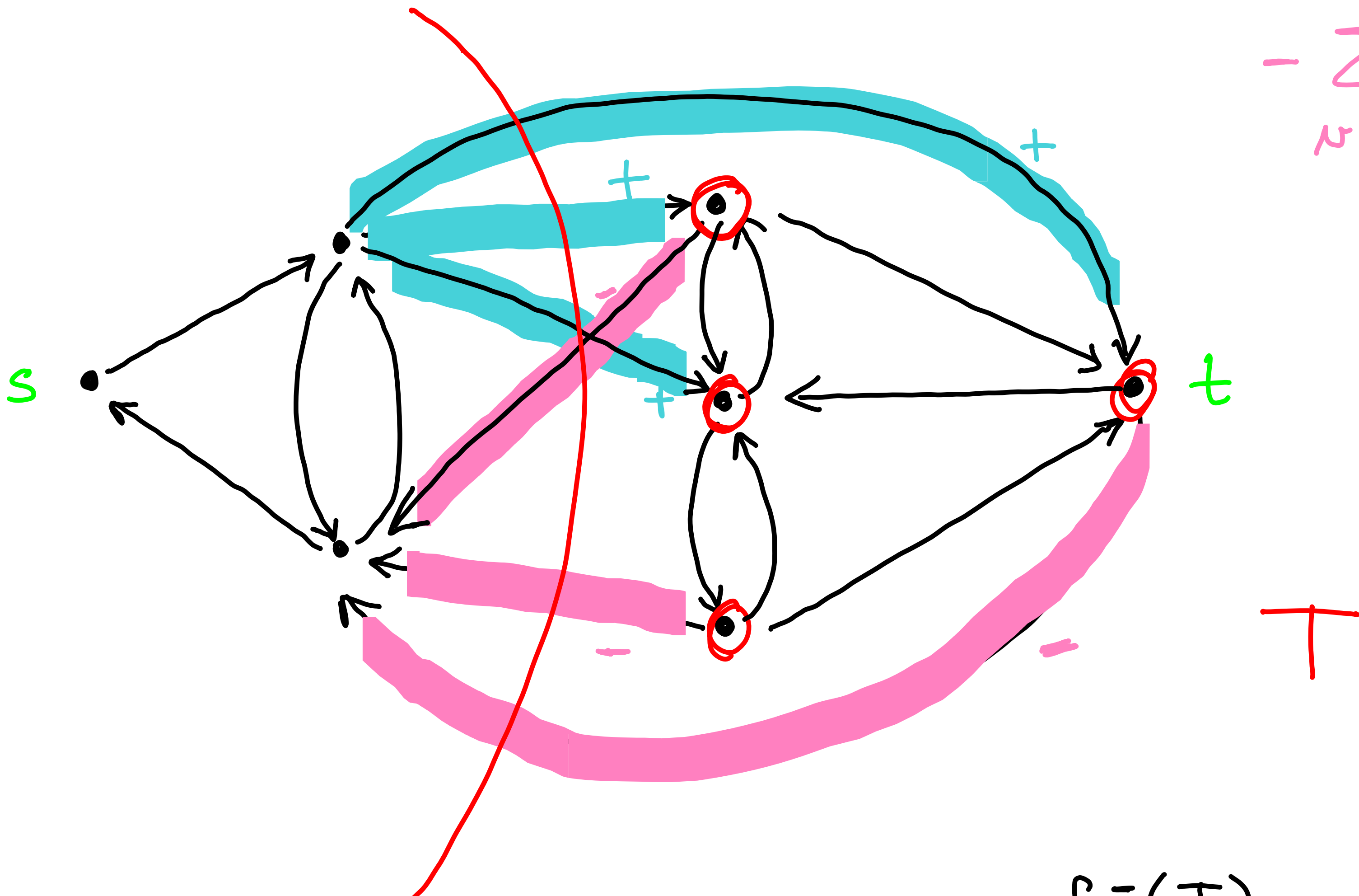
$- \sum_{v \in T} f^+(v)$



(b) $\sum_{v \in T} f^-(v) - f^+(v) = f^-(T) - f^+(T)$

$\sum_{v \in T} f^-(v)$

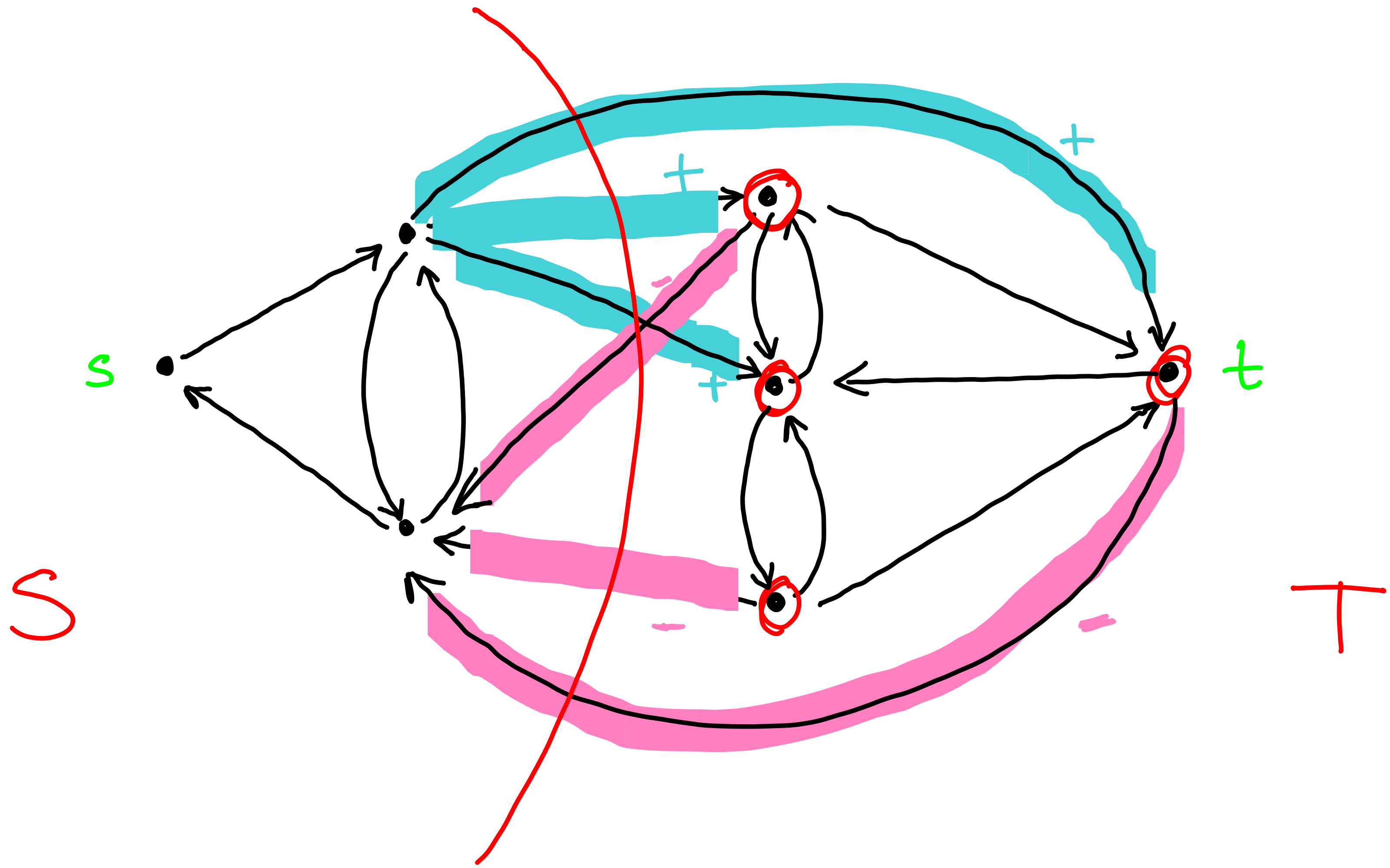
$- \sum_{v \in T} f^+(v)$



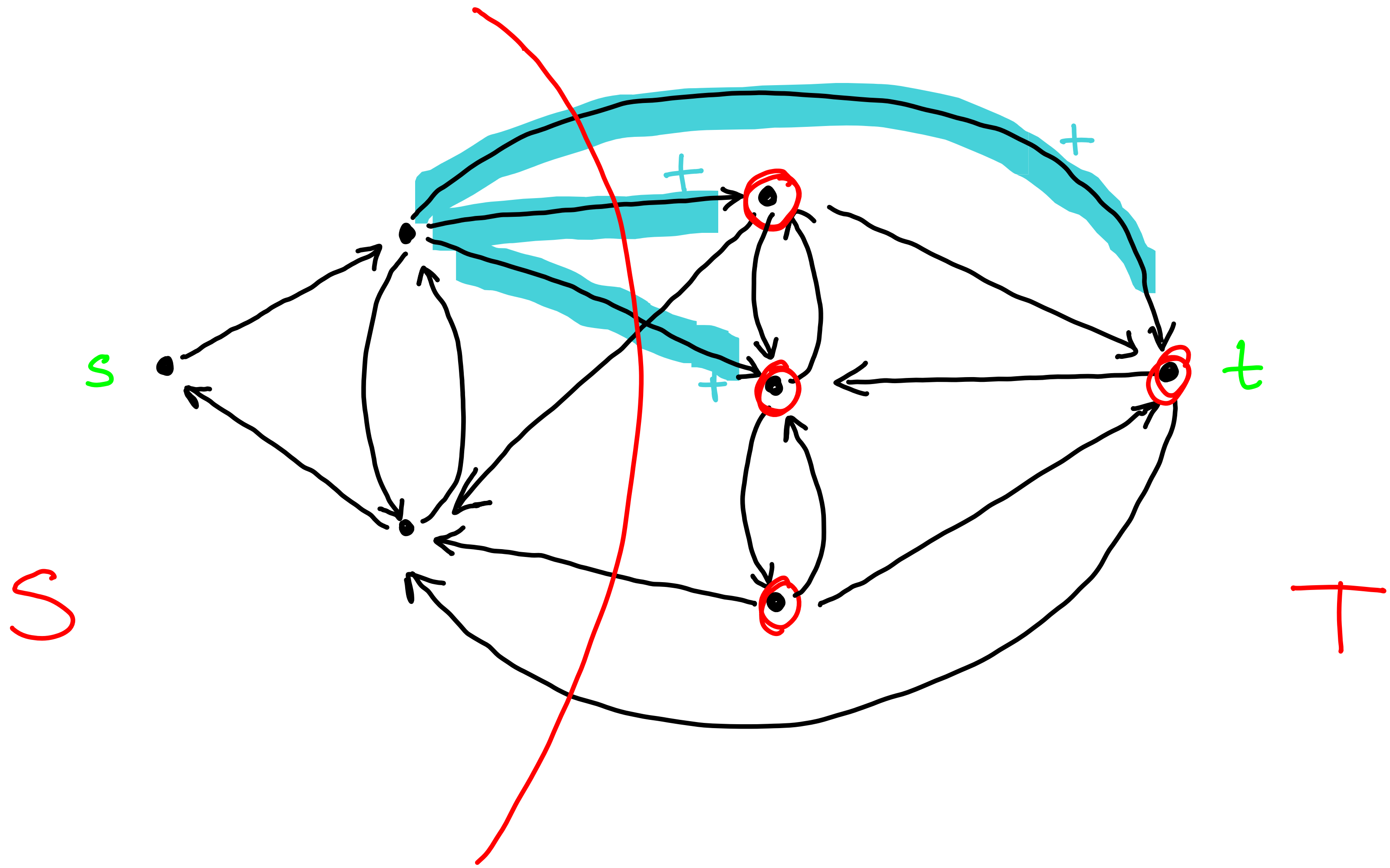
$= f^-(T) - f^+(T)$

after cancellation

(c) $f^-(T) - f^+(T) = f^+(S) - f^-(S)$



(d) $f^{+}(S) - f^{-}(S) \leq f^{+}(S) \leq \text{cap}(S,T)$



By 4.3.8 $\text{val}(f) \leq \text{Cap}(S,T)$

So, if $\text{val}(f) = \text{Cap}(S,T)$ then

f must be a maximum flow and

$[S,T]$ must be a minimum source/sink cut.

But, do there always exist such f and $[S,T]$ satisfying

$$\text{val}(f) = \text{Cap}(S,T) \quad ?$$

Thm. 4.3.11

<u>**Max-flow Min-cut Theorem**</u> [Ford-Fulkerson 1956] In every network, the maximum value of a feasible flow equals the minimum capacity of a source/sink cut.

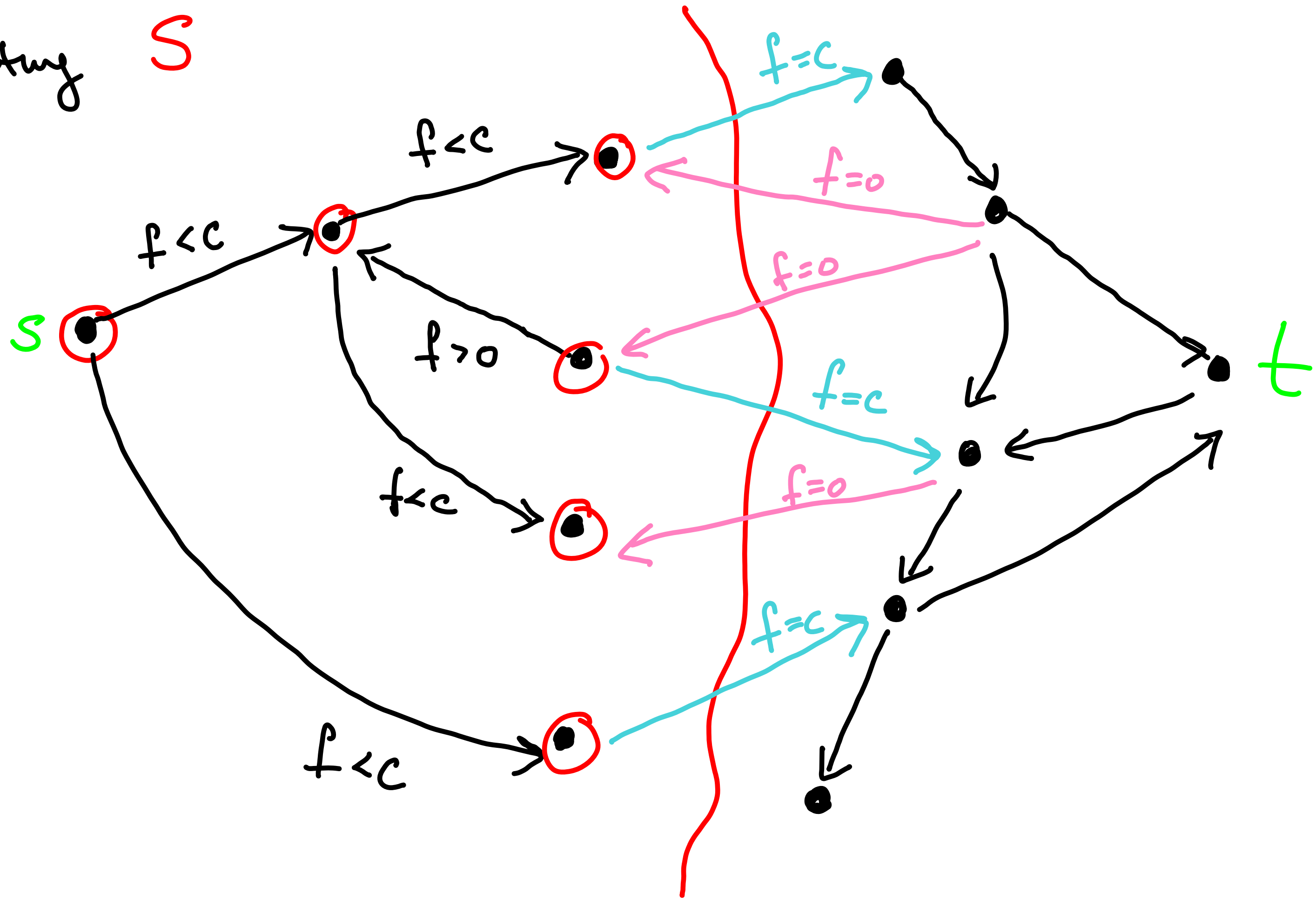
Max-flow Min-cut Theorem [Ford-Fulkerson 1956] In every network, the maximum value of a feasible flow equals the minimum capacity of a source/sink cut.

Proof Suppose f is a maximum flow. Then no f-augmenting paths.

Construct a cut as follows.

Let S consist of s and all vertices reachable from s by paths of positive tolerance. (See figure)

Constructing S



Max-flow Min-cut Theorem [Ford-Fulkerson 1956] In every network, the maximum value of a feasible flow equals the minimum capacity of a source/sink cut.

Proof Suppose f is a maximum flow. Then no f-augmenting paths.

Construct a cut as follows.

Let S consist of s and all vertices reachable from s by paths of positive tolerance. (See figure)

Then $t \notin S$ (why?)

So $[S, V-S]$ is a source/sink cut.

By 4.3.8 proof (a-c)

$$val(f) = f^+(S) - f^-(S).$$

Max-flow Min-cut Theorem [Ford-Fulkerson 1956] In every network, the maximum value of a feasible flow equals the minimum capacity of a source/sink cut.

Proof Suppose f is a maximum flow. Then no f- augmenting paths.

Construct a cut as follows.

Let S consist of s and all vertices reachable from s by paths of positive tolerance. (See figure)

Then $t \notin S$ (Why?)

So $[S, V-S]$ is a source/sink cut.

By 4.3.8 proof (a-c)

$$\boxed{val(f) = f^+(S) - f^-(S).}$$

Now

$f(e) = c(e)$ for $e \in [S, V-S]$

so $f^+(S) = cap(S, V-S)$

$f(e) = 0$ for $e \in [V-S, S]$.

so $f^-(S) = 0$

and $\boxed{val(f) = cap(S, V-S)}$

Ford-Fulkerson labeling algorithm

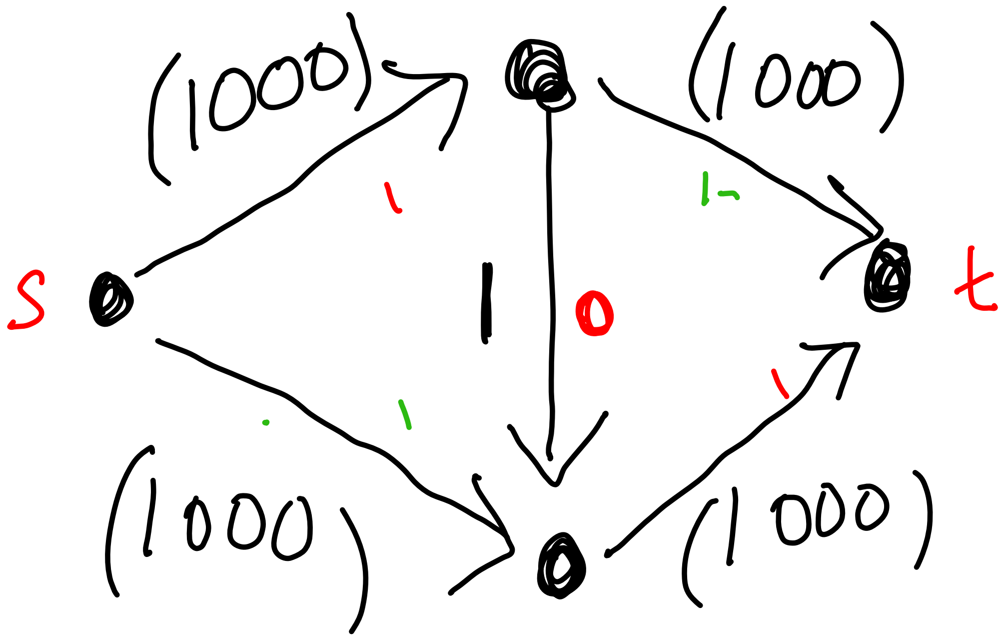
$f \leftarrow$ any feasible flow (e.g. all 0)

$S \leftarrow$ set of vertices reachable from s by paths of positive tolerance

while $t \in S$ do (Since there is an augmenting path)

use augmenting path to make f larger

By the proof of the Max-flow Min-cut theorem, upon termination, f is a max. flow and $[S, V-S]$ a min cut.



val f = 2

tolerance 1

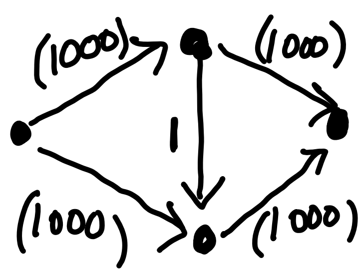
Questions:

Will the algorithm always terminate?

- See p. 180 of text.

How many iterations could it take in the worst case for a graph with n vertices and m edges?

Consider what might happen with this graph



(It is not polynomial in the size of the graph!)

Who showed that the network flow problem could be solved in polynomial time, regardless of the capacities?