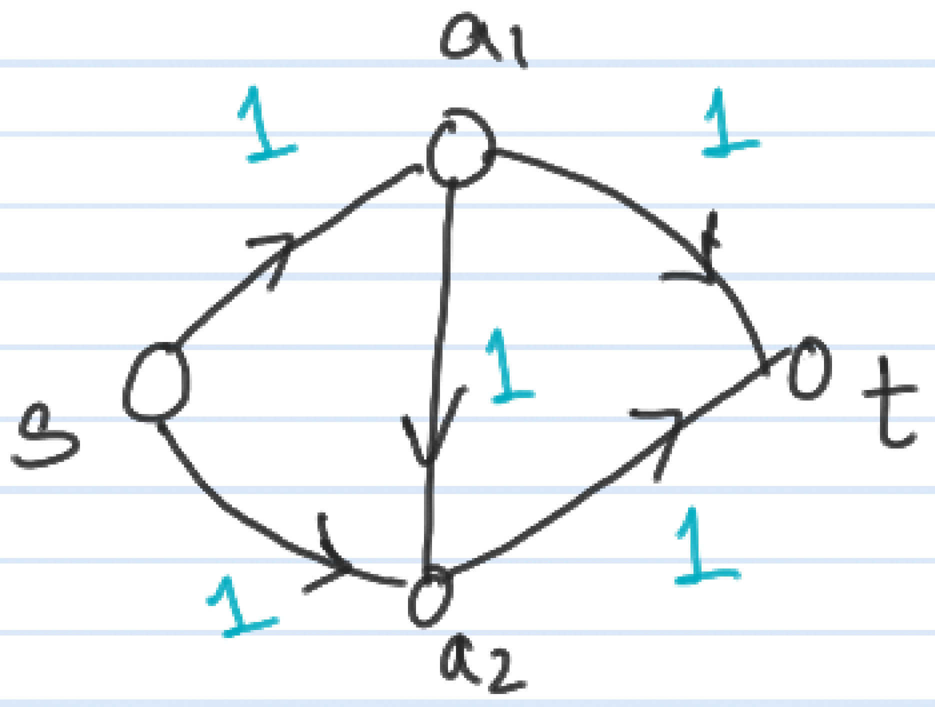
CS6130 : Advanced Graph Algorithms

Maximum Flow Problem

- what is a flow n/w ?
- what is flow ?
- How to compute a maximum flow ?

Flow Network and flows

A directed graph $G(V, E)$ with designated
- source (s) and destination (t)
- each edge has a capacity $c(e) \geq 0$



Flow in a n/w is a **function** $f: E \to R^+$ s.t

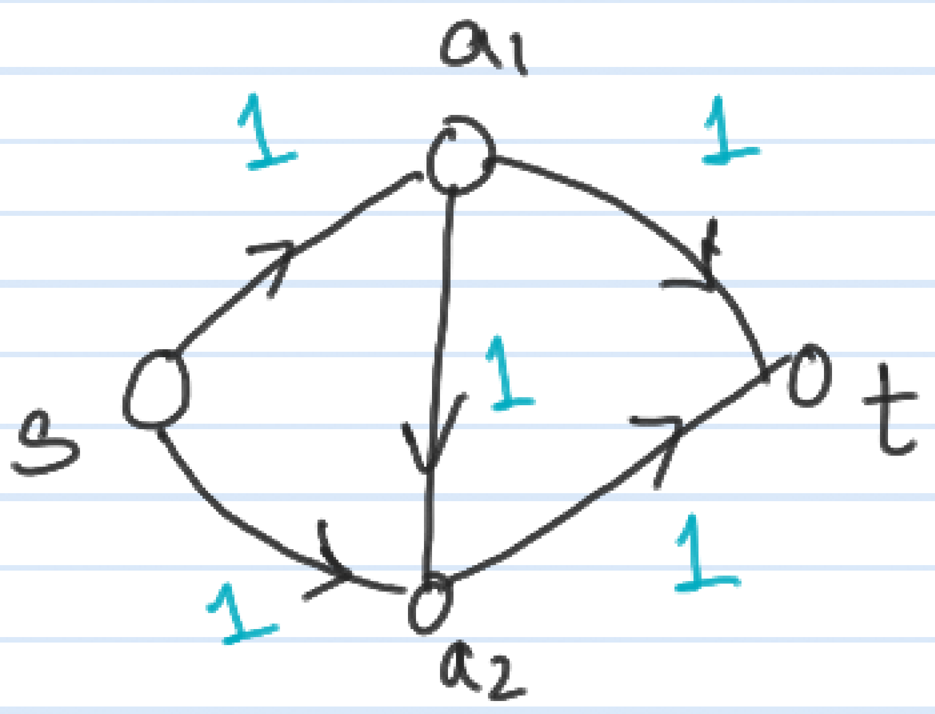
(1) capacity constraint $\forall e : f(e) \leq c(e)$

(2) Flow conservation: $\forall u \in V \setminus \{s, t\}$
$f_{in}(u) = f_{out}(u)$

Flow Network and flows

A directed graph $G(V,E)$ with designated

- source (s) and destination (t)
- each edge has a capacity $c(e) \geq 0$



$f_{in}(a_1) = f(s, a_1)$

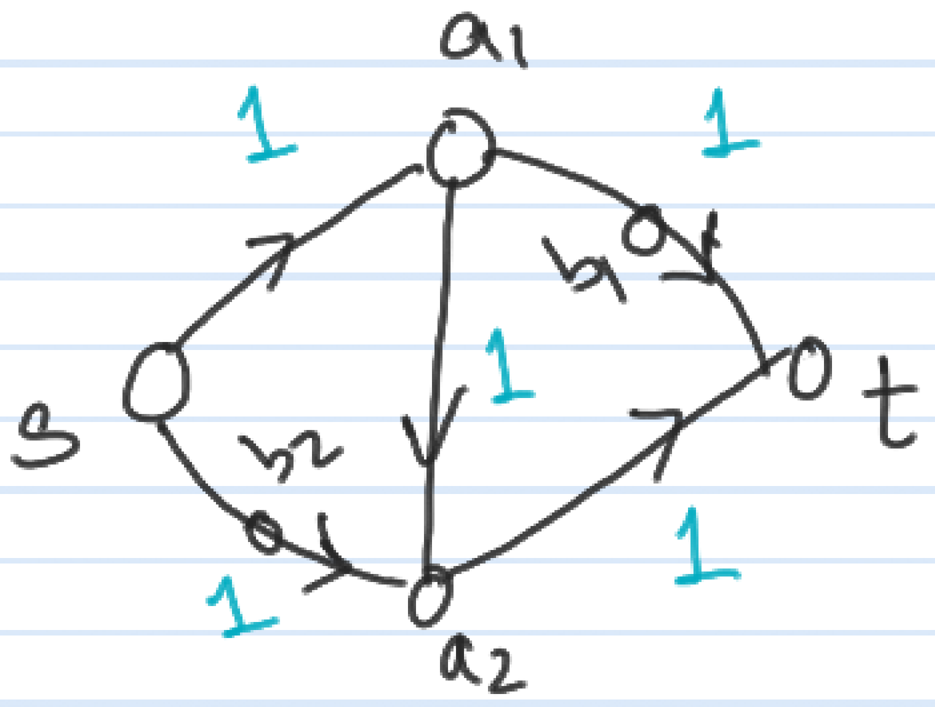
$f_{out}(a_1) =$

$f_{in}(a_2) = f(s, a_2) + f(a_1, a_2)$

$f_{out}(a_2) =$

Flow Network and flows

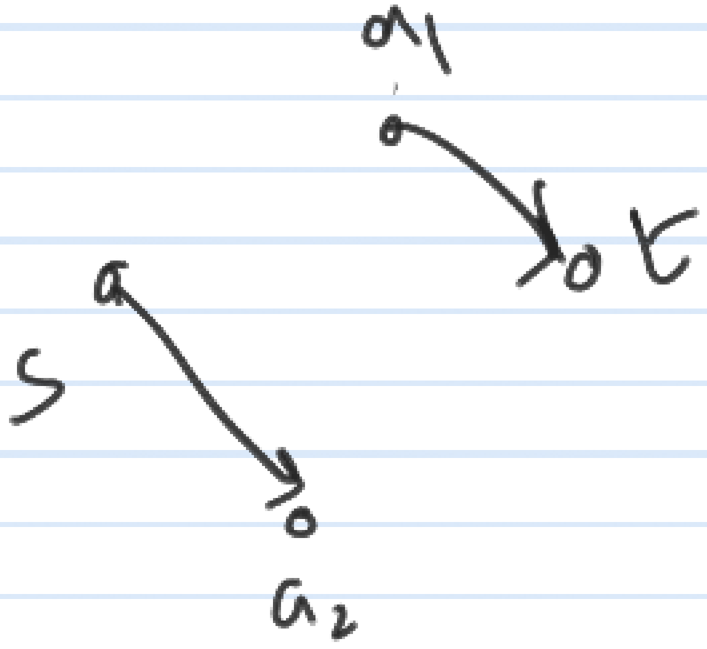
A directed graph $G(V,E)$ with designated
- source (s) and destination (t)
- each edge has a capacity $c(e) \geq 0$



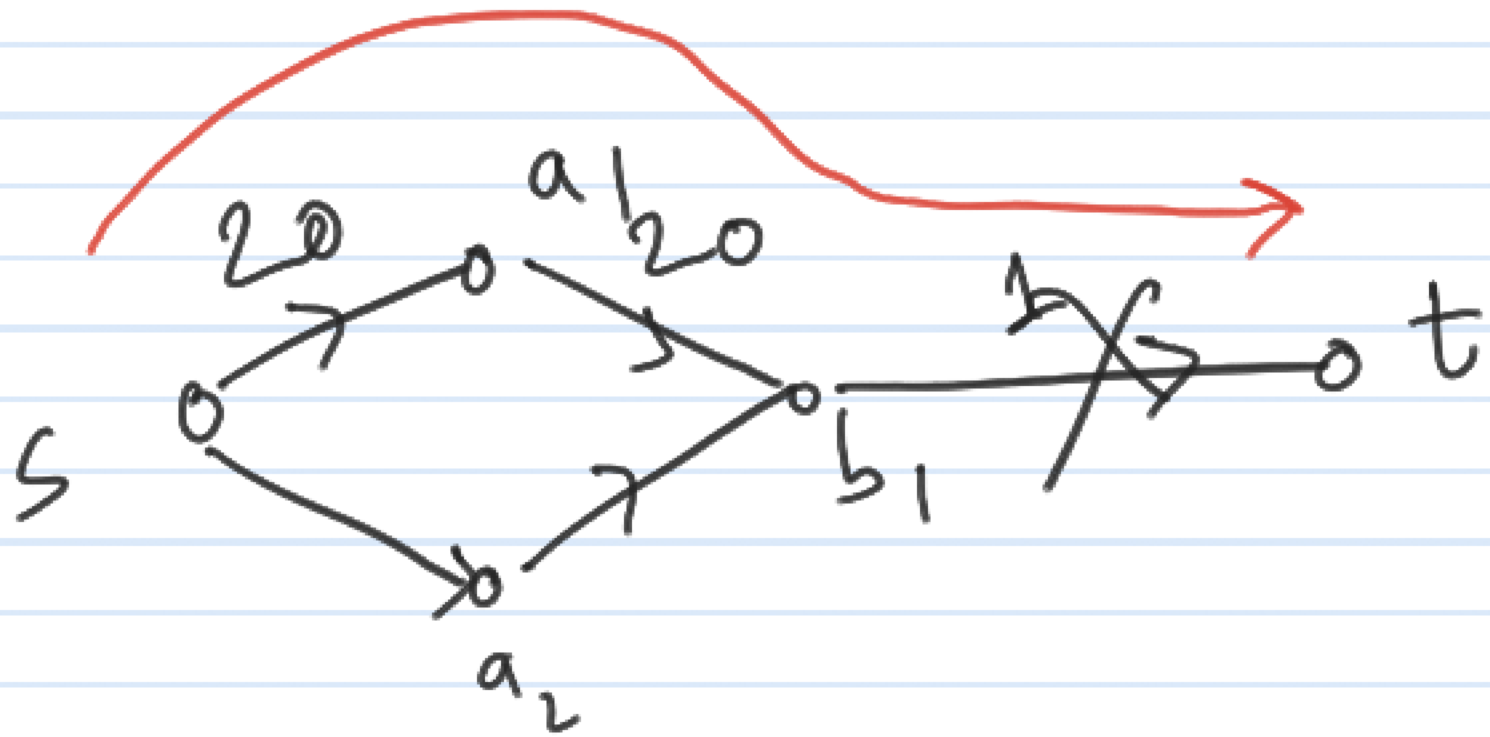
Flow f is a function that satisfies capacity and conservation

value of flow f: $|f|$

$|f| = f_{out}(s)$

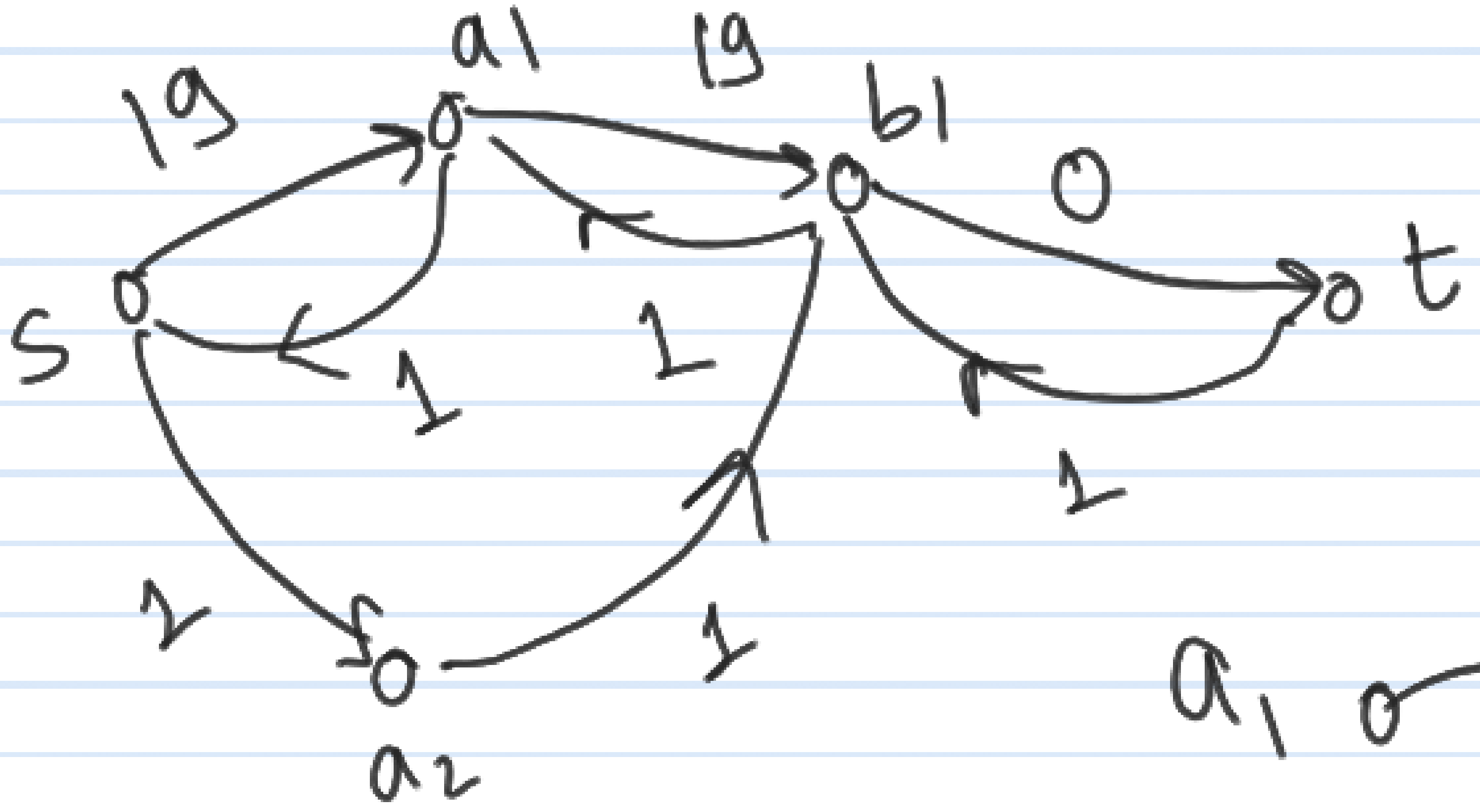


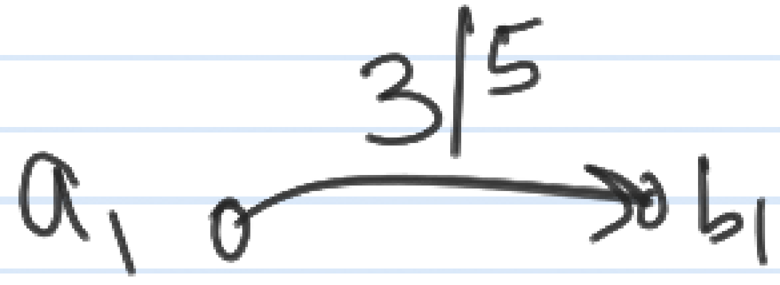
Example worked out



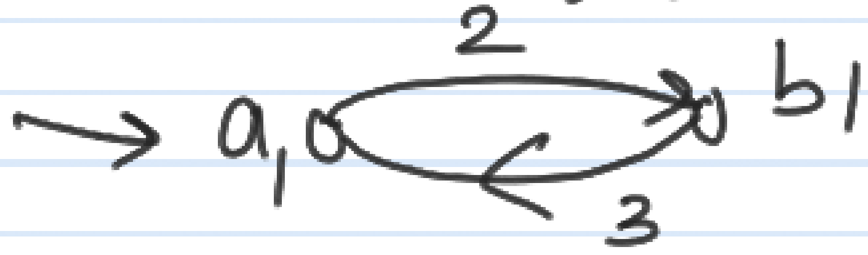
while ∃ an s,t path in current n/w

- let p be an s,t path
- send as much flow as possible along p

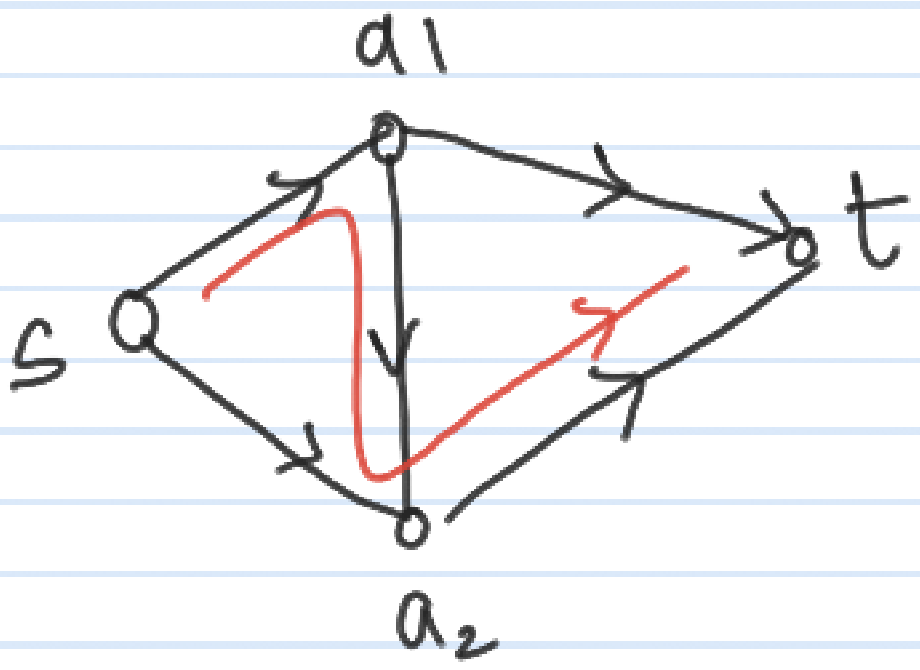




- modify n/w



Residual Network



$$f(s,a_1) = f(a_1\ a_2) = f(a_2\ t) = 1$$

$$f(s,a_2) = f(a_1\ t) = 0$$

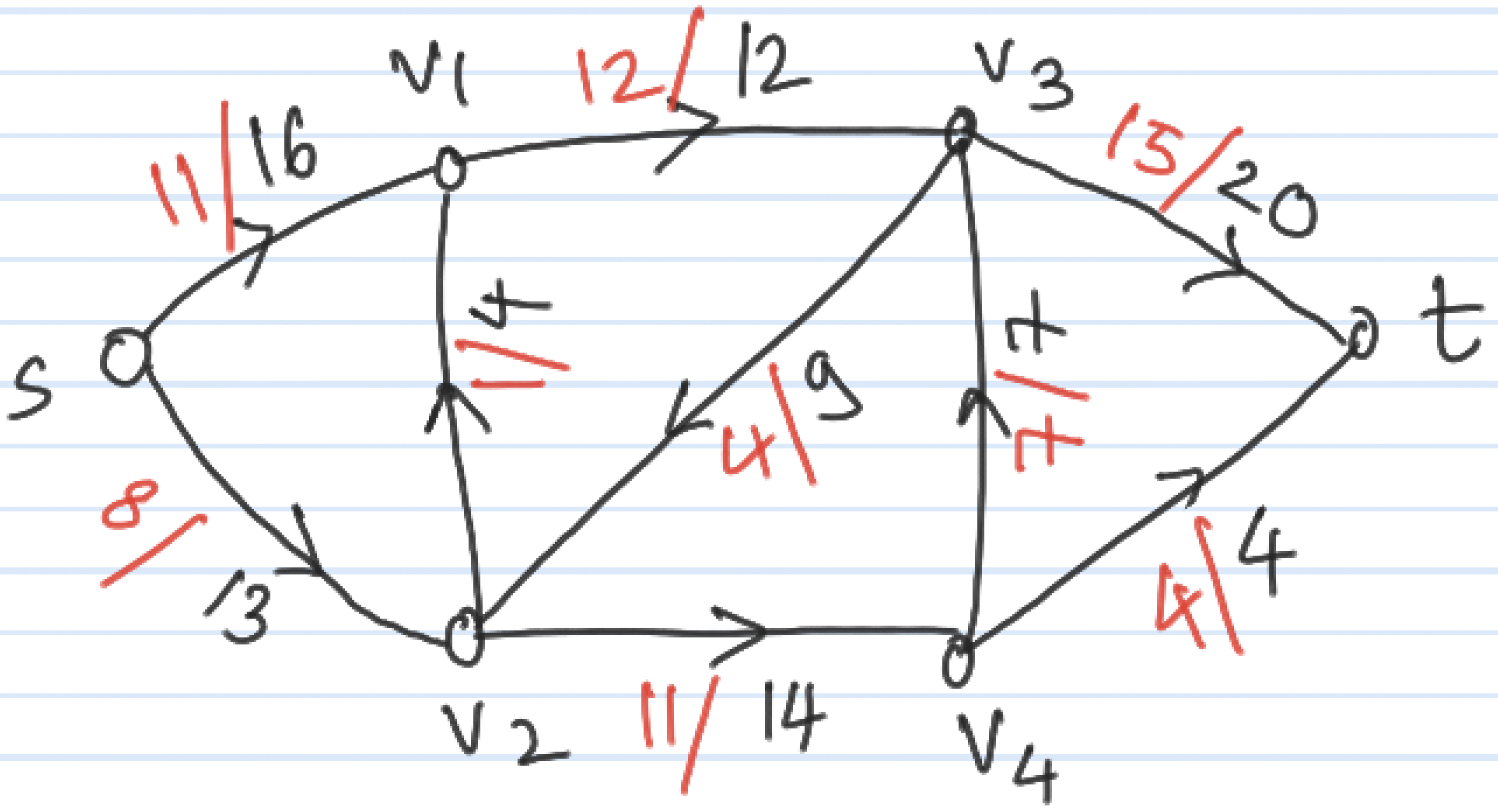
$c_f(a_1, a_2) = 0$

$c_f(a_2, a_1) = 1$

Residual capacity (defined for $V \times V$)

$$c_f(u,v) = \begin{cases} c(u,v) - f(u,v) & \text{if } (u,v) \in E \\ f(v,u) & \text{if } (v,u) \in E \\ 0 & \text{otherwise.} \end{cases}$$

Residual Network



G with flow f

Residual capacities

$c_f(v_1 v_3) = 0$

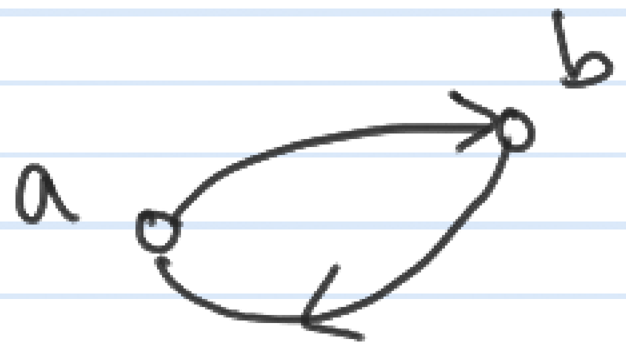
$c_f(v_2 v_3) = 4$

$c_f(v_3 v_2) = 5$

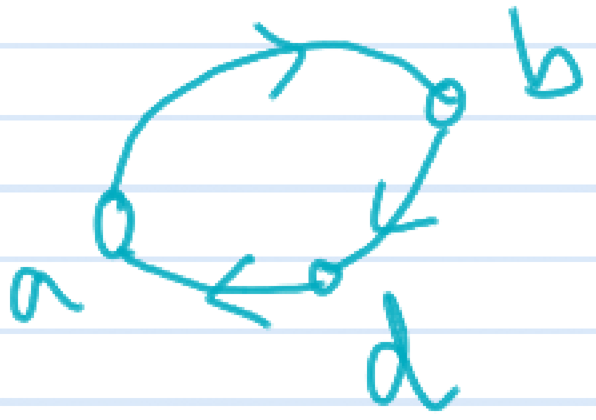
$c_f(s, v_3) = 0$

$G_f = (V, \{E : c_f(e) > 0\}$

Antiparallel Edges



antiparallel edges



What if given n/w has antiparallel edges?

$G \longrightarrow G'$

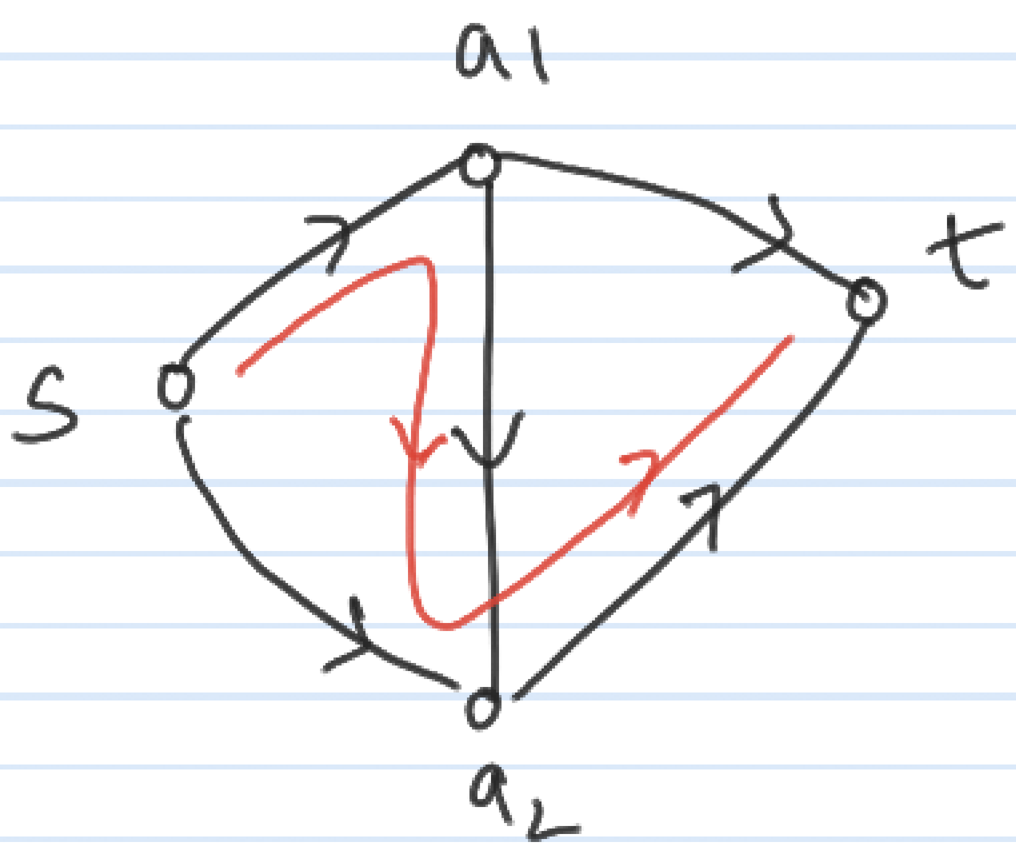
$f \qquad f'$

$|f| = |f'|$

$f_{out}(s) - f_{in}(s)$

$|f| = f_{out}(s) - f_{in}(s)$

Augmenting Flow

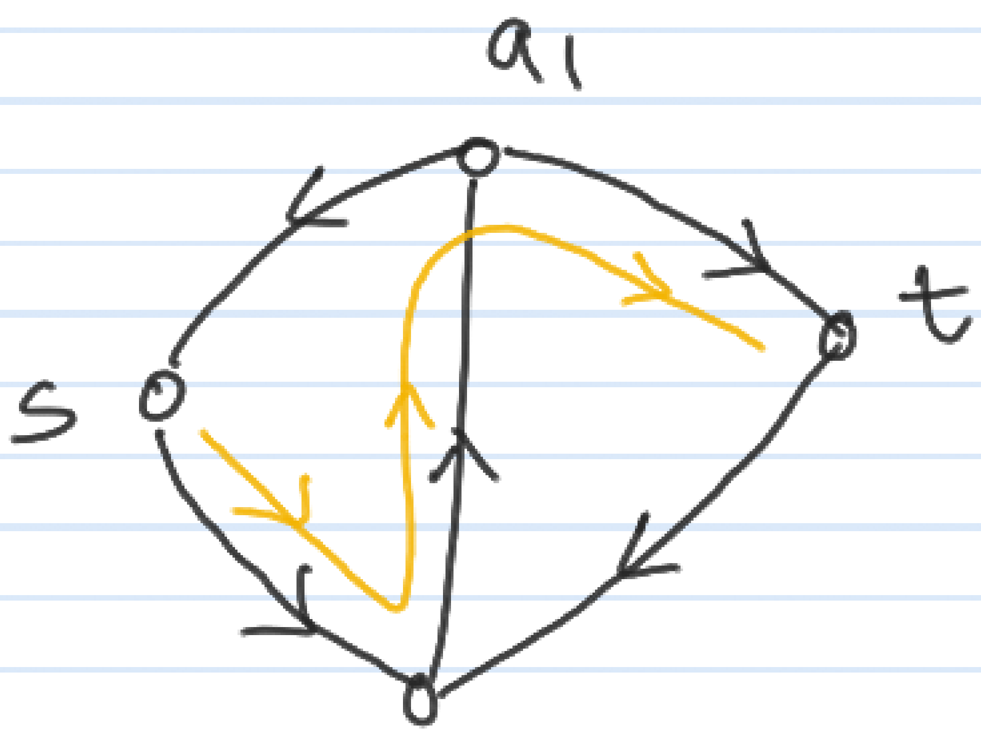


G along with flow f

f

$f''(s, a_1) = 1$

$f''(a_1, a_2) = 1$



Residual n/w G_f

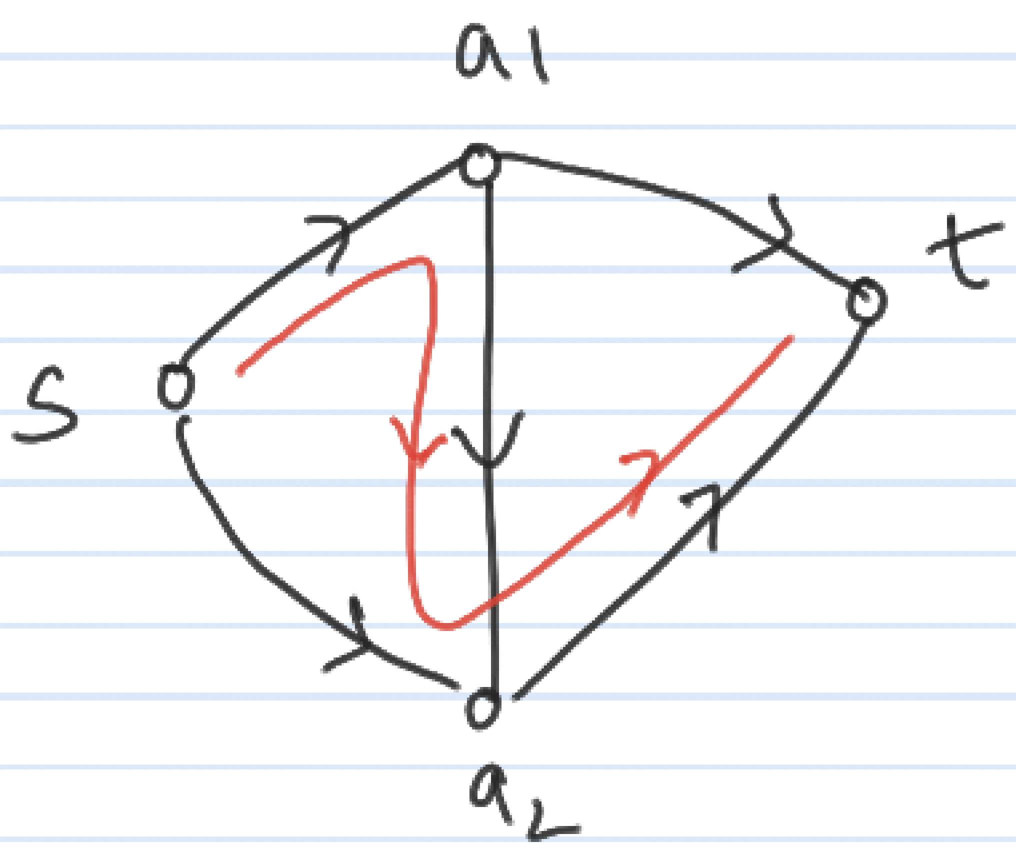
Flow f' in G_f

$f'(s, a_2) = f'(a_2 a_1) = f'(a_1 t) = 1$

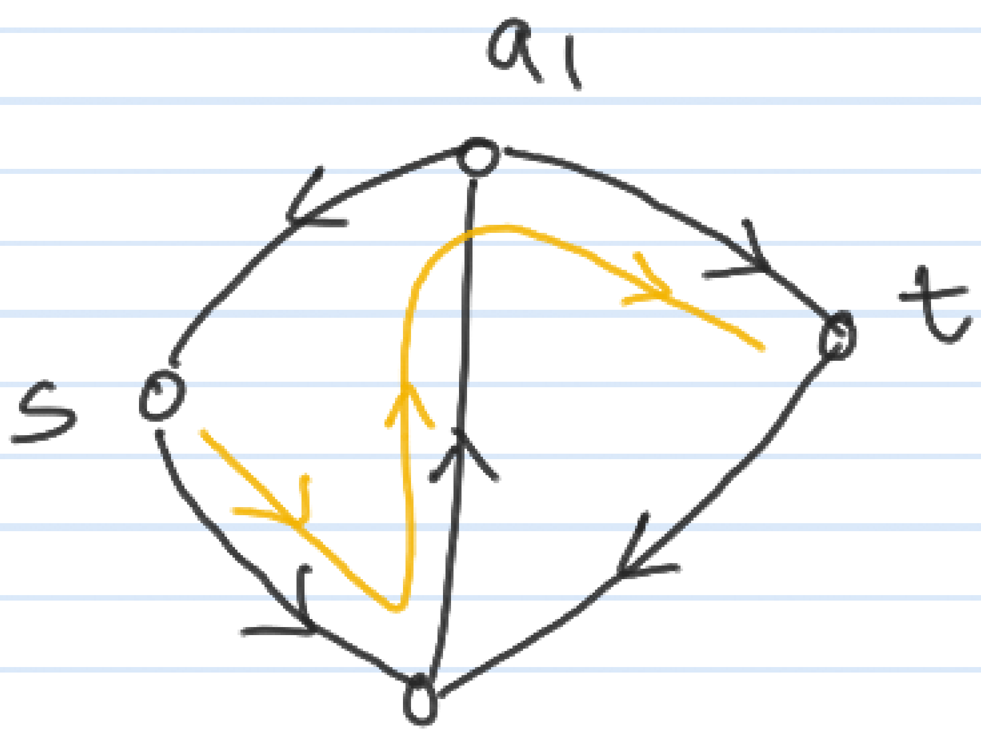
Need to combine f and f'

- why?

Augmenting Flow



G along with flow f



Residual n/w G_f

Flow f' in G_f

Need to combine f and f'
- why?

$$(f \uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u) \quad \text{for } (u,v) \in E.$$

$$= 0 \quad \text{otherwise.}$$

Augmented Flow

$$(f\uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u) \qquad \text{if } (u,v) \in E$$

$$= 0 \qquad \text{otherwise}$$

$f\uparrow f'$: is a function

Argue that it is a valid flow function

$$(f\uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u)$$

$$\geq f(u,v) + f'(u,v) - f'(u,v)$$

$$\geq f(u,v) \geq 0$$

$f'(v,u) \leq c_f(v,u)$

$c_f(v,u) = f(u,v)$

Augmented Flow

$$(f \uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u) \quad \text{if } (u,v) \in E$$
$$= 0 \quad \text{otherwise}$$

$f \uparrow f'$: is a function

Argue that it is a valid flow function

$$(f \uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u)$$
$$\leq f(u,v) + f'(u,v)$$
$$\leq c(u,v)$$

$f'(u,v) \leq c_f(u,v)$

$c_f(u,v) =$
$c(u,v) - f(u,v)$

Augmented Flow

$(f \uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u)$ if $(u,v) \in E$

$= 0$ otherwise

$f \uparrow f'$: is a function

Argue that it is a valid flow function

$(f \uparrow f')(u,v)$:

(1) ≥ 0

(2) $\leq c(u,v)$

(3) satisfies flow conservation

$|f \uparrow f'| = |f| + |f'|$

(complete the proof)

Ford Fulkerson Algorithm

Input : G, s, t, $c(e)$

start with $f(e) = 0$ for all e

while $\exists$ an aug path p in G_f

- aug f along p to obtain f''
- set $f = f''$

end while

Ford Fulkerson Algorithm

Input : G, s, t, $c(e)$

start with $f(e) = 0$ for all e

while $\exists$ an aug path p in G_f

- aug f along p to obtain f''
- set $f = f''$

end while

$\boxed{|f^*| \times O(m+n)}$

Need to establish

1) Termination

2) Running time

\# of iterations × time for an iteration

3) Optimality