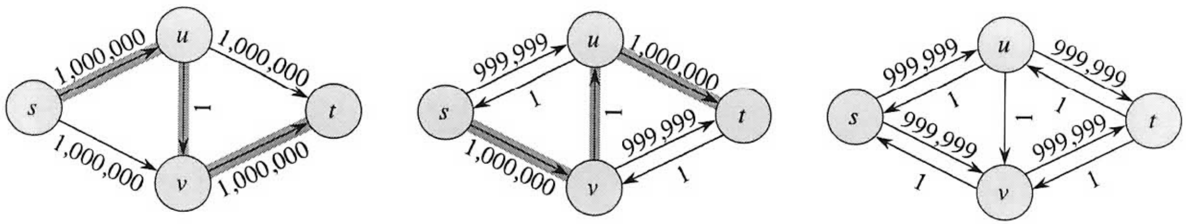
Edmonds-Karp

use BFS to find augmenting paths


s
u
v
t
1,000,000
1,000,000
1,000,000
1,000,000
1
999,999
1
1,000,000
999,999
1
1,000,000
999,999
999,999
1
1
999,999
1
999,999
1
1

(a) (b) (c)

Ford-Fulkerson method

1. $f :=$ zero flow
2. Construct residual graph G_f
3. Find an augmenting path p & construct f_p
4. $f := f \uparrow f_p$

Repeat until no augmenting paths are found

flow value of max flow

How many iterations do we need?

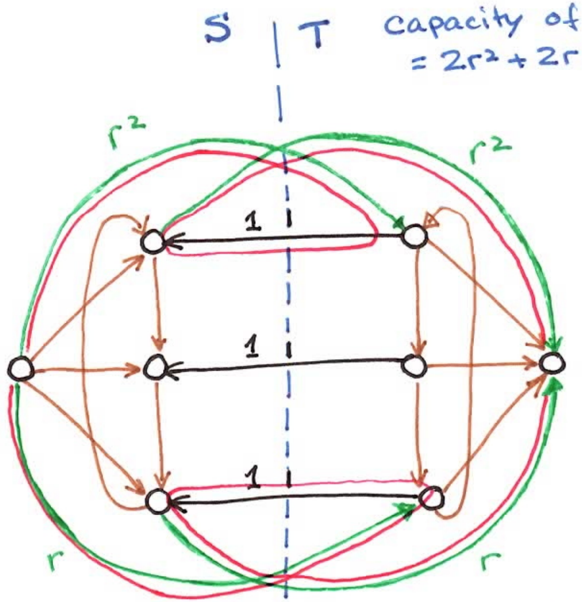
- If capacities are integers $|f^*|$ must be an integer since min cut is an integer.
- Each iteration increases $|f|$ by 1, at least.
- # of iterations $\leq |f^*|$

not good, could be large

The strange case of a

FLOW NETWORK WITH IRRATIONAL CAPACITY

capacity of min cut
$= 2r^2 + 2r = 2(r^2+r) = 2$

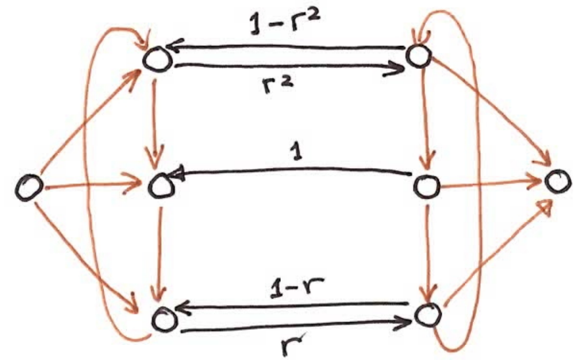


$\rightarrow$ brown edges have capacity $10 = \infty$

$r = \frac{\sqrt{5}-1}{2}$ is the root of $x^2 + x - 1 = 0$

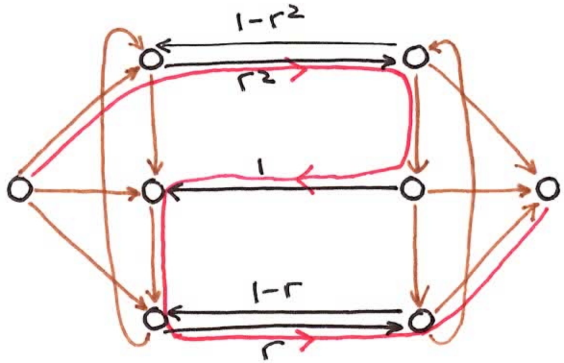
= golden ratio

send r^2 and r units thru 2 augmenting paths



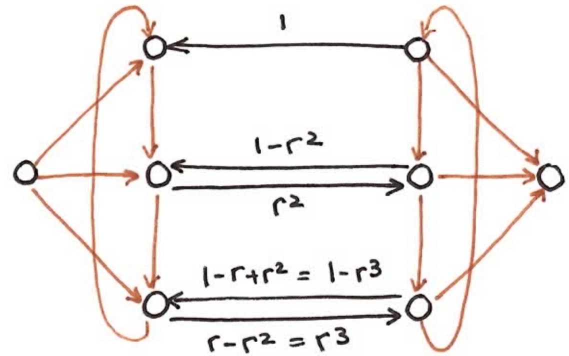
residual graph
some reverse edges
not shown

flow $= r^2 + r = 1$



send r² units thru augmenting path

residual graph
reverse of brown edges not shown



$r^2 + r - 1 = 0$

$\Rightarrow r^2 = 1 - r$

$\Rightarrow r^3 = r - r^2$

$\Rightarrow 1 - r^3 = 1 - r + r^2$

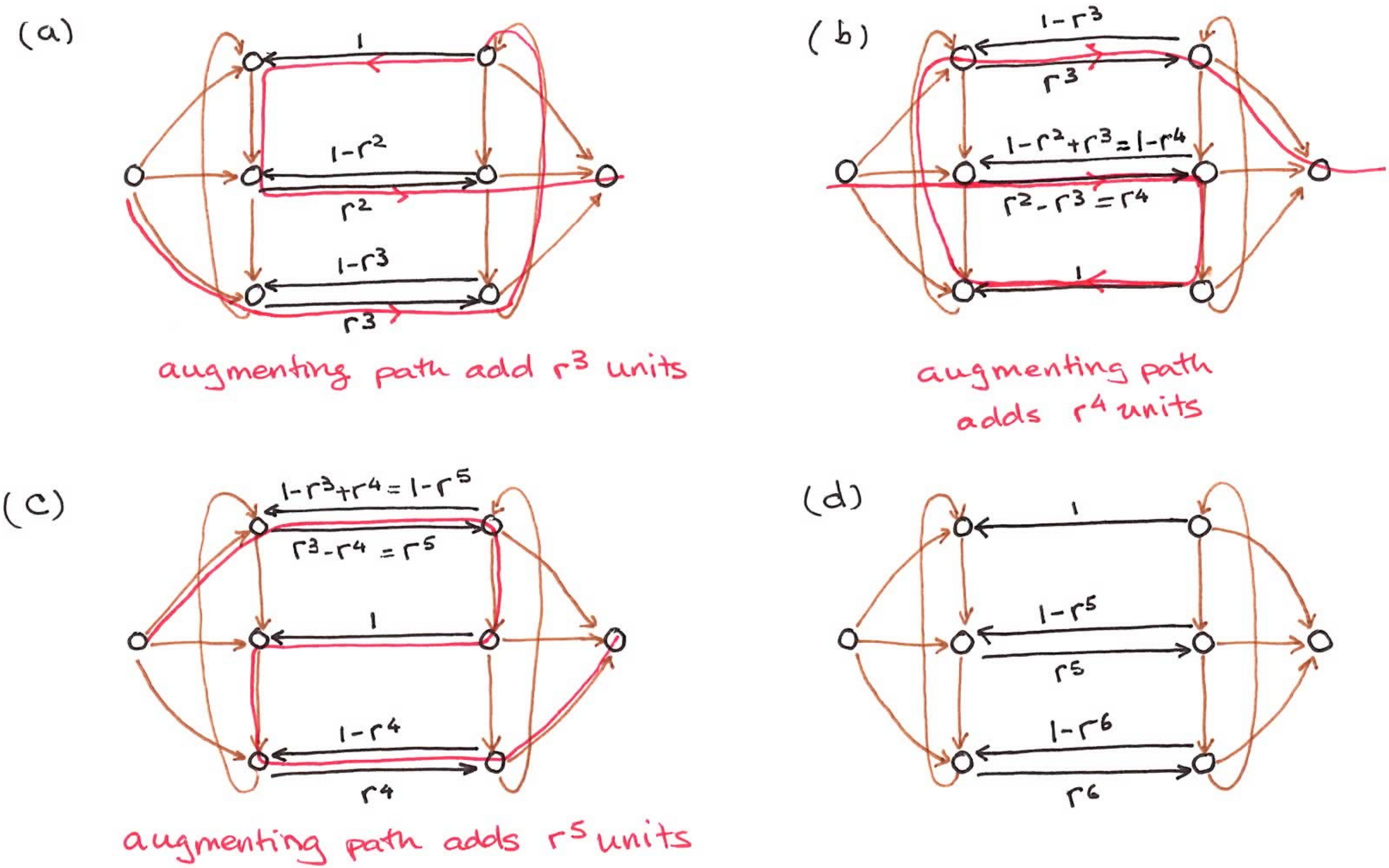
Some facts about $r = \frac{\sqrt{5}-1}{2}$

$$\frac{x}{1} = \frac{1}{1+x}$$

- called the golden ratio
- solution to $x^2 + x - 1 = 0$
- approximately 0.6180339887499

- $r^{i+2} = r^i - r^{i+1}$
- $\frac{1}{1-r} = r + 2 = \sum_{i=0}^{\infty} r^i$
- $r^2 + r^3 + r^4 + \cdots = 1$

just simple arithmetic


(a)
1
1-r²
r²
1-r³
r³
augmenting path add r³ units
(b)
1-r³
r³
1-r²+r³=1-r⁴
r²-r³=r⁴
1
augmenting path adds r⁴ units
(c)
1-r³+r⁴=1-r⁵
r³-r⁴=r⁵
1
1-r⁴
r⁴
augmenting path adds r⁵ units
(d)
1
1-r⁵
r⁵
1-r⁶
r⁶

Flow constructed:

- initial 2 augmenting paths added r^2 and r units
- i^{th} augmenting path adds r^{i-1} units
- total flow $= (r^2 + r) + r^2 + r^3 + r^4 + r^5 + \cdots$

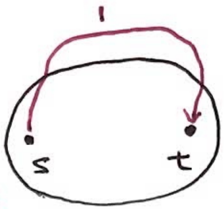
$$= \quad 1 \quad + \quad 1$$

$$= 2$$

because $r + 2 = \frac{1}{1-r} = \sum_{i=0}^{\infty} r^i$

$$= 1 + r + \underbrace{r^2 + r^3 + \cdots}$$

- Max flow attained after an infinite # of augmenting paths.
- Might not converge to max flow:



add an extra edge from s to t
with capacity 1. Now max flow is 3.

Moral of the story?

Stick with integer capacities!

Edmonds-Karp algorithm

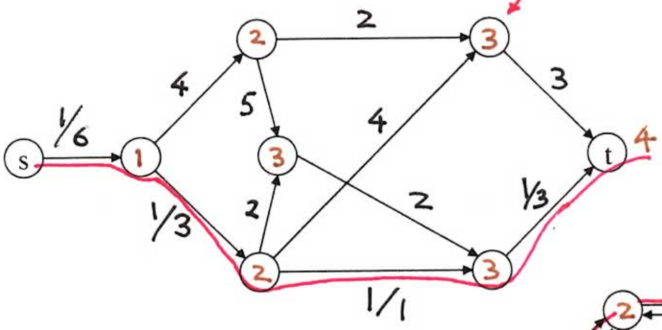
- use BFS to find augmenting path with fewest # of edges in Ford-Fulkerson.
- uses $O(VE)$ iterations
- $O(V+E)$ time per iteration
- Total time $O(VE^2) \approx O(V^5)$

  still better than $O(|f^*|)$

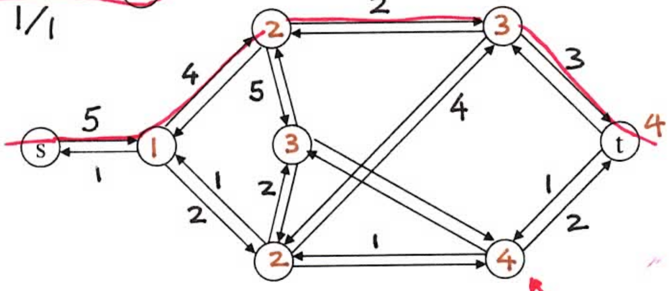
  since $V \le E$ — for dense graphs
- Slow [$O(V^3)$ algorithms exist.]

Edmonds-Karp Example

link distance from s

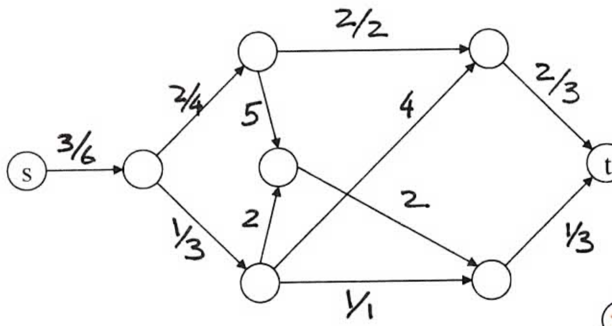


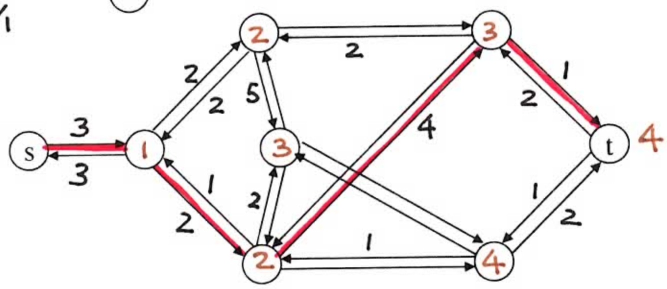
$C_p = 1$



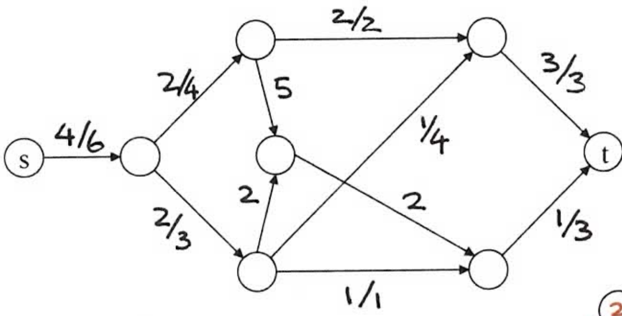
$C_p = 2$

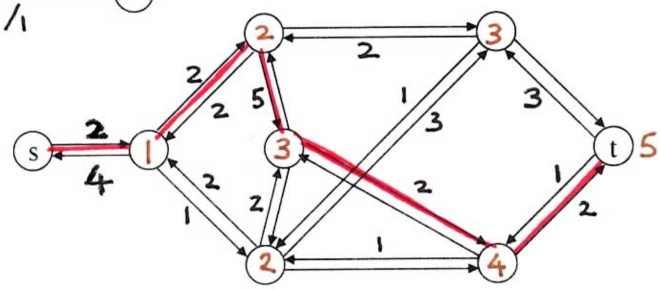
distance increased



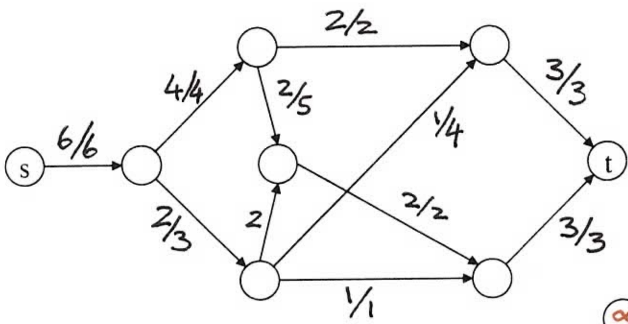


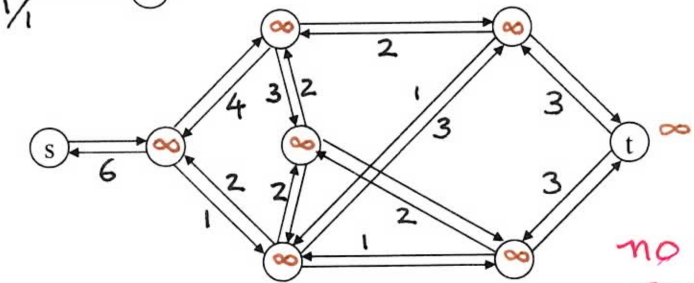
$C_p = 1$





$C_p = 2$





no more augmenting paths

Edmonds-Karp Running Time

depends on flow f

residual graph

$\delta_f(s,v)$ = minimum # of edges in a path from s to v in G_f

= link distance from s to v in G_f

= $\delta_f(v)$ since we only use $\delta_f(s,-)$

can stay equal

Lemma: For each $v \in V - \{s,t\}$, $\delta_f(v)$ increases monotonically after each flow augmentation using BFS.

Intuition: As flow is augmented, more and more edges in G_f "disappear". Hence, the link distances get larger & larger.

Proof of Lemma: (by contradiction)

Let $f' = f \uparrow f_P$ be the first augmentation that causes $\delta_f()$ to decrease for some vertex v. I.e., $\delta_{f'}(v) < \delta_f(v)$. 1

Out of all such v, pick the one with smallest $\delta_{f'}()$. 2

Let p' be a min. link path in $G_{f'}$ from s to v.

$$p': s \rightsquigarrow u \to v \rightsquigarrow t$$

Then, $\delta_{f'}(v) = \delta_{f'}(u) + 1$ 3 since p' is a min. link path

By 2, $\delta_{f'}(u) \geq \delta_f(u)$. 4

if $\delta_{f'}(u) < \delta_f(u)$, then we would have picked u instead of v

not f'

Two cases: either $(u,v) \in E_f$ or not.

Case 1: If $(u,v) \in E_f$ then

$$\delta_f(v) \le \delta_f(u) + 1$$ triangle inequality

$$\le \delta_{f'}(u) + 1$$ by 4 $\delta_{f'}(u) \ge \delta_f(u)$.

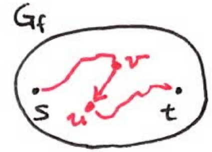
$$= \delta_{f'}(v)$$ by 3

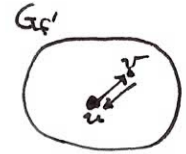
$\therefore$ $\delta_f(v) \le \delta_{f'}(v)$

But, this contradicts the assumption that $\delta_{f'}(v) < \delta_f(v)$ in 1.

Case 2: Suppose $(u,v) \notin E_f$. That means $(u,v) \in E_{f'}$ reappeared after augmentation $f' = f \uparrow f_p$.

So augmenting path p in G_f must have directed flow from v to u.





$$p: \quad s \rightsquigarrow v \rightarrow u \rightsquigarrow t$$

Note: p is a min. link path.

Then, $\delta_f(v) = \delta_f(u) - 1$ min. link path

$\qquad \leq \delta_{f'}(u) - 1$ by 4 $\delta_{f'}(u) \geq \delta_f(u)$

$\qquad = (\delta_{f'}(v) - 1) - 1$ by 3 $\delta_{f'}(v) = \delta_{f'}(u) + 1$

$\qquad = \delta_{f'}(v) - 2$

So, $\delta_f(v) \leq \delta_{f'}(v) - 2$, but this contradicts $\delta_{f'}(v) < \delta_f(v)$. by 1

END OF LEMMA ☒

Claim: Edmonds-Karp takes at most $O(VE)$ iterations.

Let $f' = f \uparrow f_p$ be an augmentation.

$c_f(p) = c_f(u,v)$ of some edge (u,v) on p.

(u,v) "disappears"

We send $c_f(p)$ units thru p, so edge $(u,v) \notin E_{f'}$.
We say that edge (u,v) is critical.

In later iterations, edge (u,v) can reappear in a residual graph if some augmenting path sends flow from v to u.

We need to count the # of times an edge can reappear.

Subclaim: Each time an edge (u,v) reappears, the link distance of u increases by at least 2.

Suppose (u,v) disappears after augmentation $f_1' = f_1 \uparrow f_{p_1}$ and reappears after augmentation $f_2' = f_2 \uparrow f_{p_2}$.

Then (u,v) must be on path p_1. So, $\delta_{f_1}(v) = \delta_{f_1}(u) + 1$.
Also (v,u) must be on path p_2. So, $\delta_{f_2}(u) = \delta_{f_2}(v) + 1$.

By previous lemma, $\delta_{f_1}(v) \le \delta_{f_2}(v)$.

Lemma says link distance increases monotonically.

$$\delta_{f_1}(u) + 1 = \delta_{f_1}(v) \le \delta_{f_2}(v) = \delta_{f_2}(u) - 1$$

$$\Rightarrow \delta_{f_1}(u) + 2 \le \delta_{f_2}(u)$$

END OF SUBCLAIM ☒

Thus, each edge (u,v) can be critical no more than $V/2$ times.
Otherwise, link distance from s to $v > V-1$.

So, total # of iterations $\leq \frac{V}{2} \cdot E = O(VE)$.

END OF CLAIM $\boxtimes$

Time per iteration = time for BFS + constructing G_f + updating flow
$= O(V+E)$
$= O(E)$ since $V \leq E$.

Total running time $= O(VE) \cdot O(E) = O(VE^2)$

Max Flow running times:

Edmonds-Karp $O(VE^2) \approx O(V^5)$ for dense graphs $E = \Theta(V^2)$.

Max Flow by Scaling $O(E^2 \log C)$

C = max edge capacity

Dinitz's Algorithm $O(V^2E) \approx O(V^4)$

level graphs, amortized analysis

MPM Algorithm $O(V^3)$

MPM = Malhorta, Pramodh-Kumar & Maheshwari

level graphs, Fibonacci Heaps.

Preflow-Push $O(V^3)$

preflows do not conserve flow.