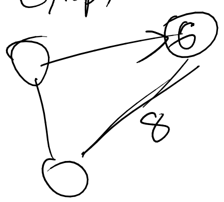
Extra notes in red (can be ignored)

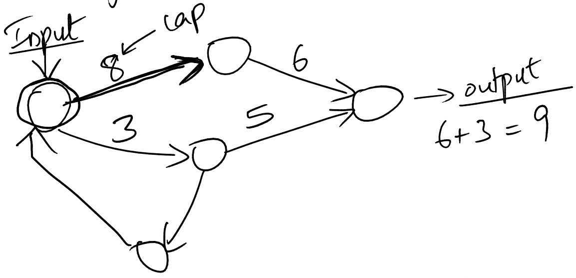
Graph



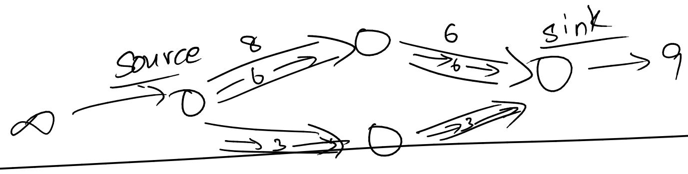
FLOWS

Network

- Directed
- Weighted → capacity



Flow Network

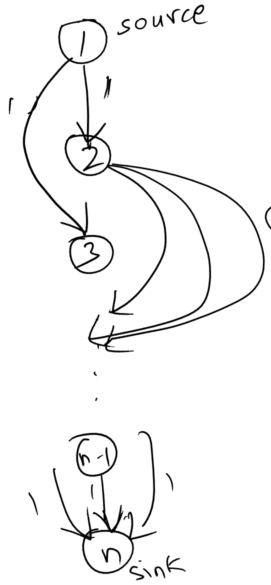


Algorithms

- Ford - Fulkerson
- Edmond - Karps
- • Dinics → $O(V^2E)$ or $O(fE)$
- Push - Relabel

FAST

Network → Dinics → Max flow (flow assignment)



$\leq n - 1$

$\frac{200^4}{2} = 8e8$

5e8

$O(VE)$

Iterations

- BFS $O(E)$
- DFS $O(E)$

↑ flow

★ $O(fE)$ ↳ maxflow

$f \sim 10^{18}$

$O(V^2E)$ better

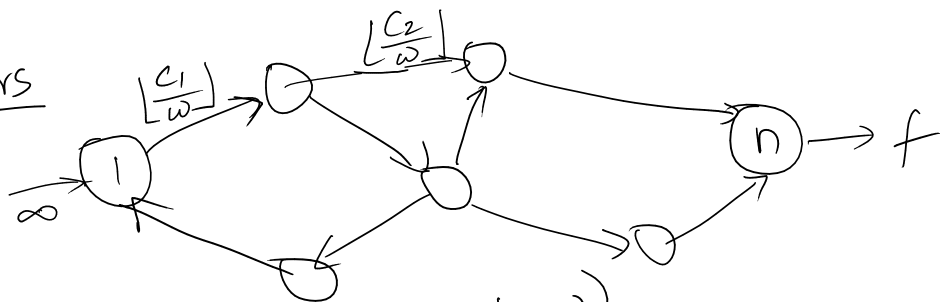
$\boxed{O(VE\log f)}$

★ Dinics + scaling

$\frac{10^7}{2^{70}} < \frac{1}{2}10^{-6}$

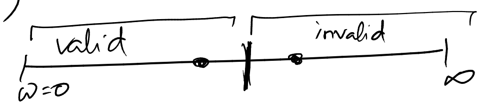
x bears

optimal ω (max)



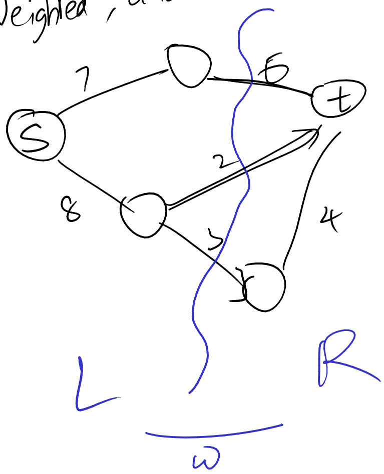
- Pick ω
  ↳ $f \geq x$?
  YES → Valid ω
  No → invalid

$O(\text{FLOW} \cdot \underbrace{\log \omega}_{\log 10^{12}})$

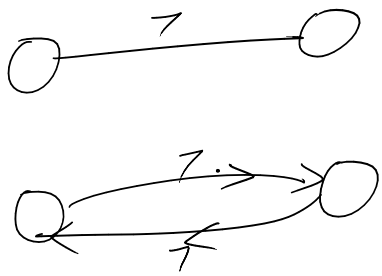


st-Min Cut

Weighted, undirected G



Min-cut-max-flow



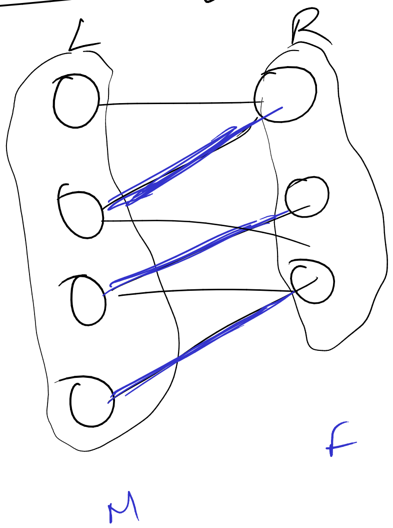
$\boxed{\text{cut} \geq \text{flow}}$

$c_i \geq f_i$

LP dual, gap = 0

max-flow(s,t) = s-t-min-cut

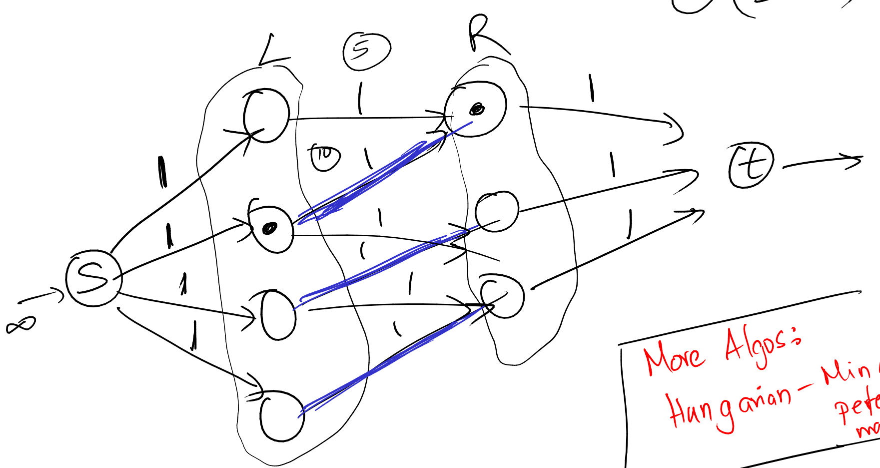
Bipartite Matching



max-matching = max-flow

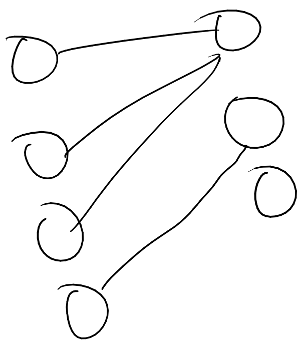
~aug

$O(E\sqrt{V})$



More Algos:

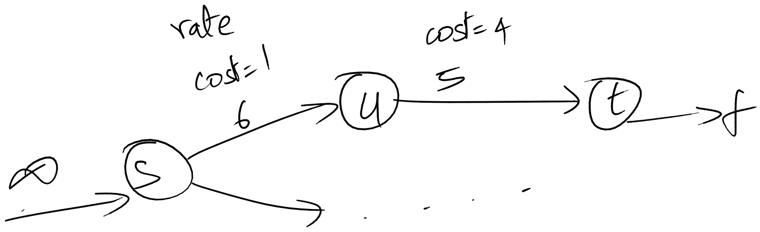
Hungarian – Min cost perfect matching



Similar: Hall's marriage theorem

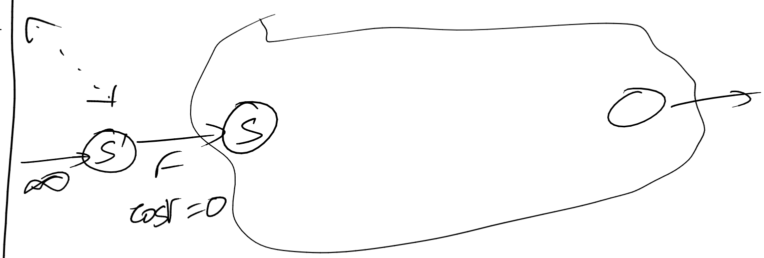
Min-cost-flow

cost ∈ R



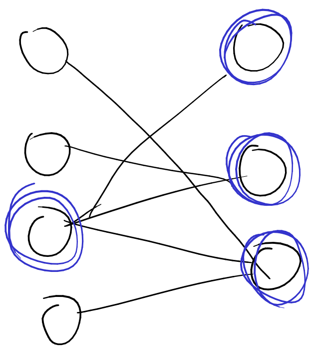
$G \to \boxed{\text{SPFA}} \to$ min cost c, max flow $f = F$

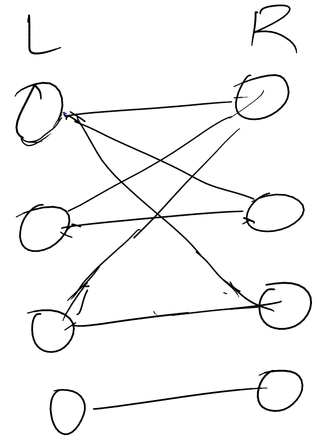
- $O(V^2E^2)$
- $O(fVE)$

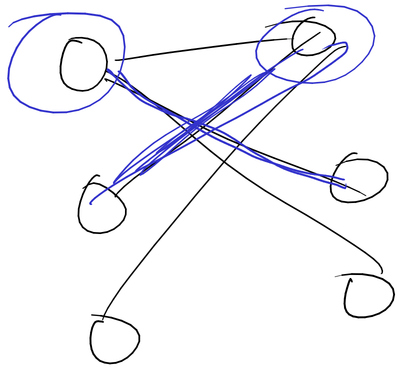


Min. Vertex Cover

(bip. graphs)







min. |vertex cover| = max |matching| ! (Konig's theorem)

(LP dual, gap = 0)

★ $\boxed{|VC| \geq |M|}$

intuition

On general graphs:
- Max matching – Blossom Algo $O(EV^2)$
- Min VC – NP Hard (LP duality gap > 0)

N = 5

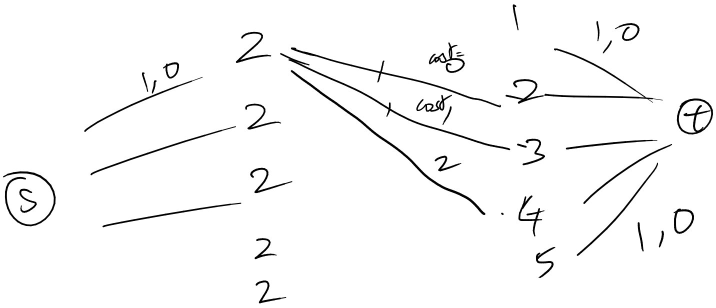
2 3 3 2 2

$\downarrow \quad \downarrow \quad \downarrow$

1 4 5

cost = 3

$x_i \leftrightarrow y_i$



$2 \leftrightarrow 3$

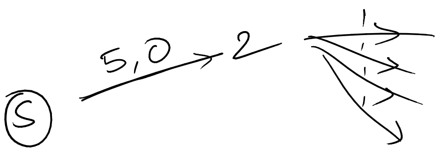
$3 \leftrightarrow 4$

$u \overset{?}{\leftrightarrow} v$

F.W. $O(n^3)$

Min cost flow

$O(EVf) = O(n^4)$



$n = 10^5, m = 10^5$

— Dilworth's theorem

min chains = max antichain

$\downarrow$

VC → Bip Mat.

?

$\downarrow$

Flows