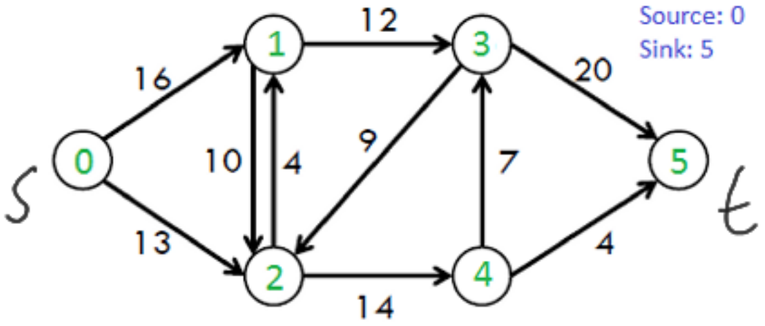
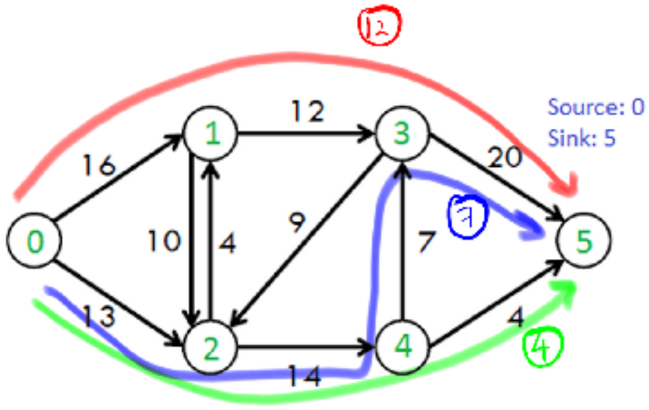
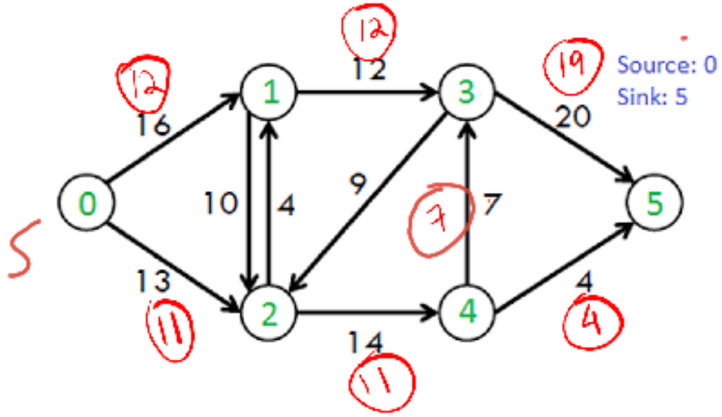
Lecture 12: Flows and LPs

- Algorithm (FF)
  - Max Flow Min Cut
- Min Cost max flows
- LPs

$G = (V, E)$







1. $0 \le f_e \le c_e$
2. flow conservation
3. no flow entering s, no flow leaving t

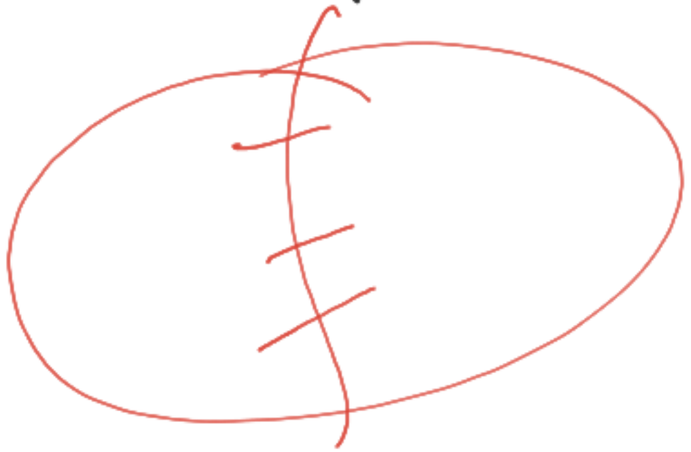
$\forall$ flows f

$\forall$ cuts C

$$\underline{\mathrm{val}(f)} = \sum_{\substack{\text{Paths } P \\ \text{in decomp}}} \phi(P) \leq \cancel{\sum_{\substack{P \\ \text{in decomp.}}} \mathrm{bottleneck}(P)}$$

$$\leq \sum_{\text{edges } e \in C} \left(\sum_{P : P \ni e} \phi(P) \right) \leq \sum_{\text{edges } e \in C} \mathrm{cap}(e) \leq \cancel{\underbrace{\sum_{P} \text{edge in } C \text{ hitting } P}} \leq \underline{\cancel{\mathrm{value}(C)}}$$

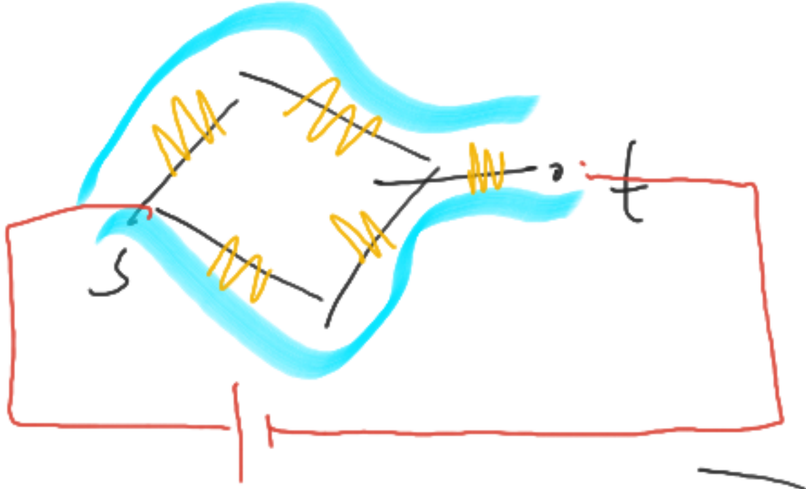
s-t Cut set of edges that "hits" every s,t path



Thm [Menger]: $\exists$ (max) flow f^* $\exists$ (min) cut C^* st $\text{value}(f^*) = \text{value}(C^*)$

- if all caps are integers then f^* can also be integer valued
- found in poly time. (always)

poly $(n, m, \underbrace{\log U_{max}})$



Extension

Sps each edge has a cost c_e

Want a flow f^*

that has ① max value, and

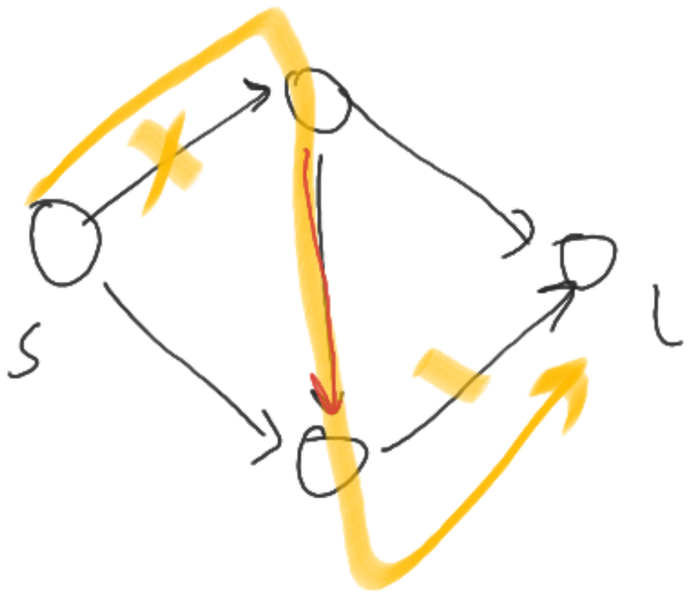
② min cost among all max value flows.

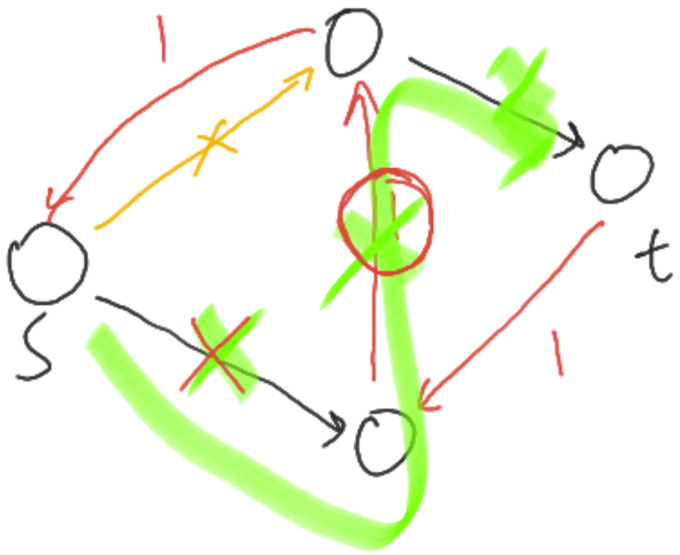
Thm: can also find this in poly time.

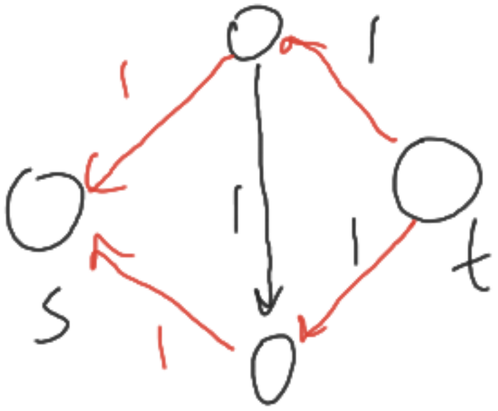
if caps are integers, then $\exists$ a integer valued f^* sat'ing

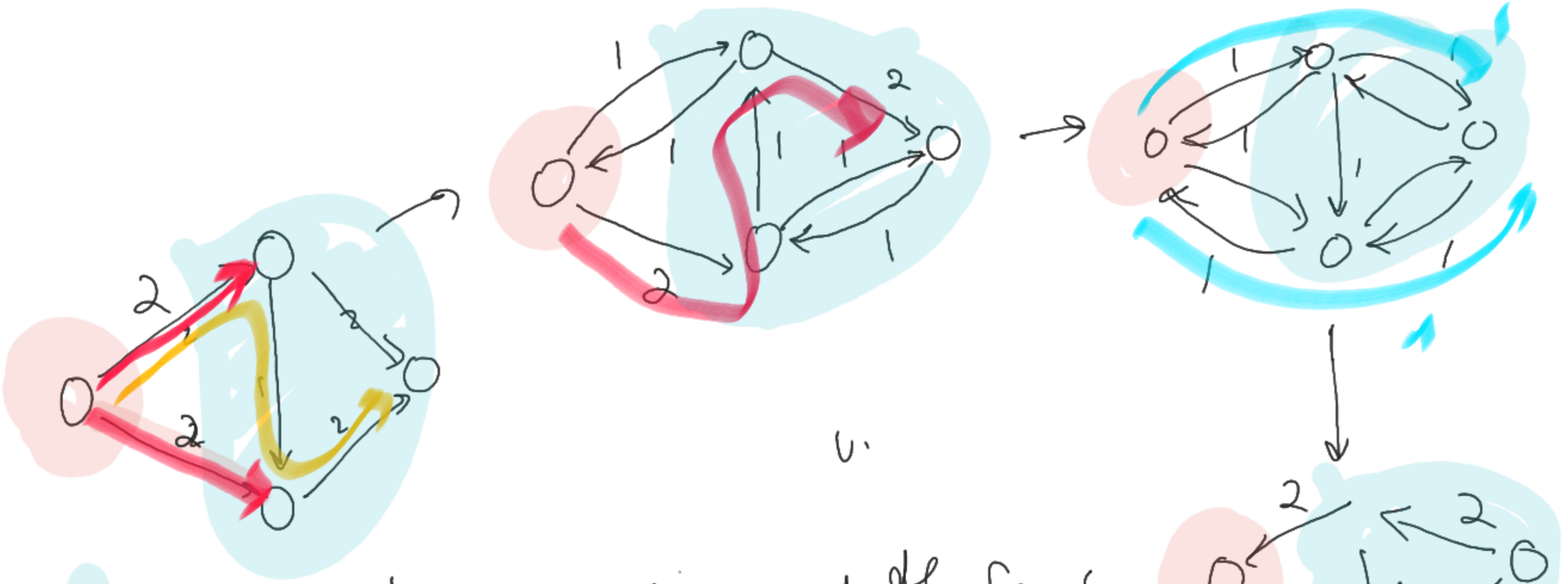
Give an algo for max flow. (outline the MFMC proof).

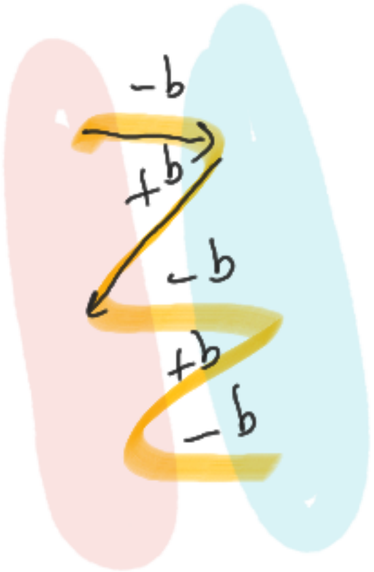
Ford - Fulkerson











Let L = all vertices reachable from s

R = all the rest

<u>Claim</u>: original edges going from L to R form a min cut.

Linear Programs:

+ Powerful (gen. Min Cost Max flow)
+ Versatile
+ Easy to use
+ ~~Machine w~~
- may not return integer sol^{n}s (even if input $\in$ Integers)
- slower (maybe)

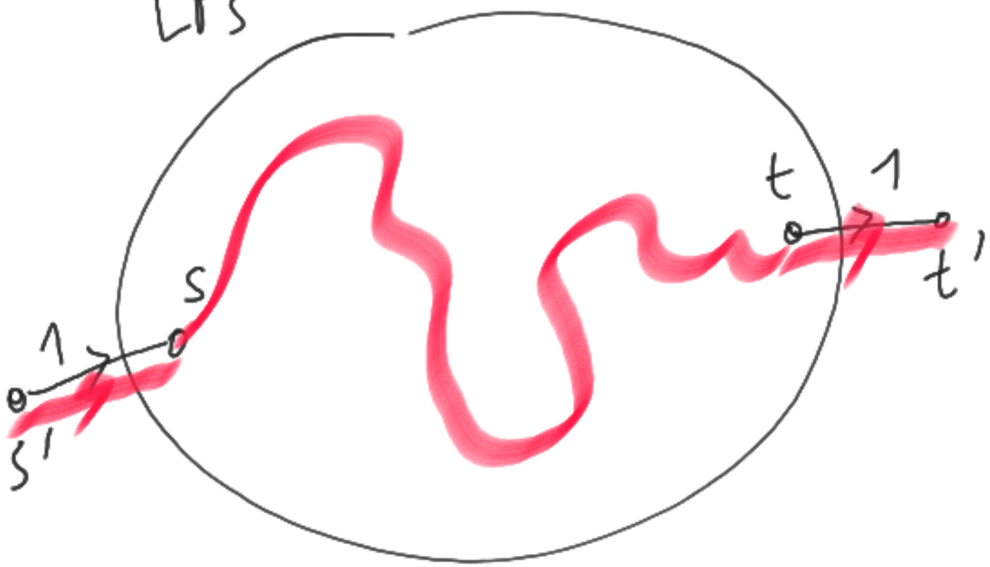
Shortest Paths

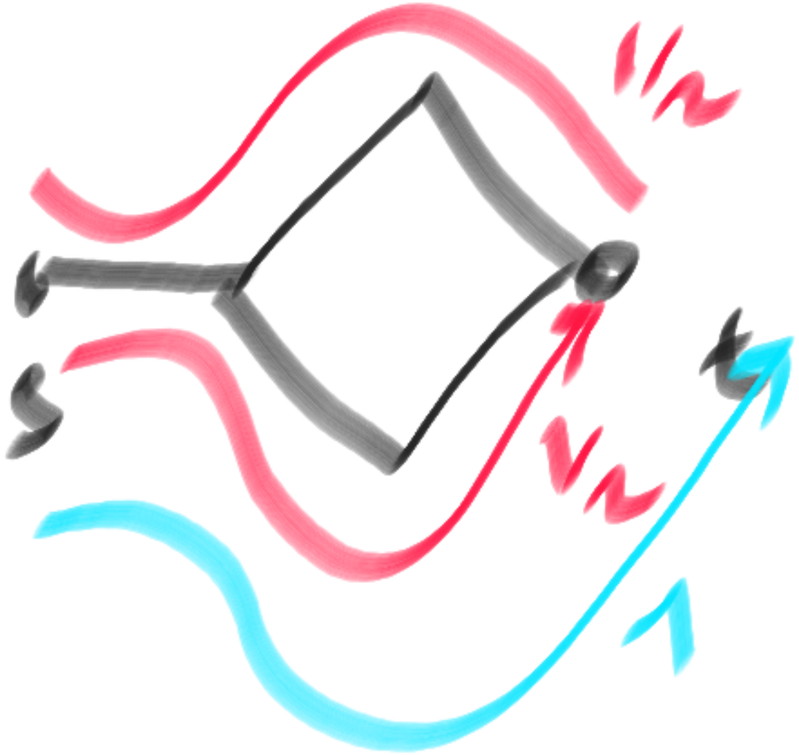
↓

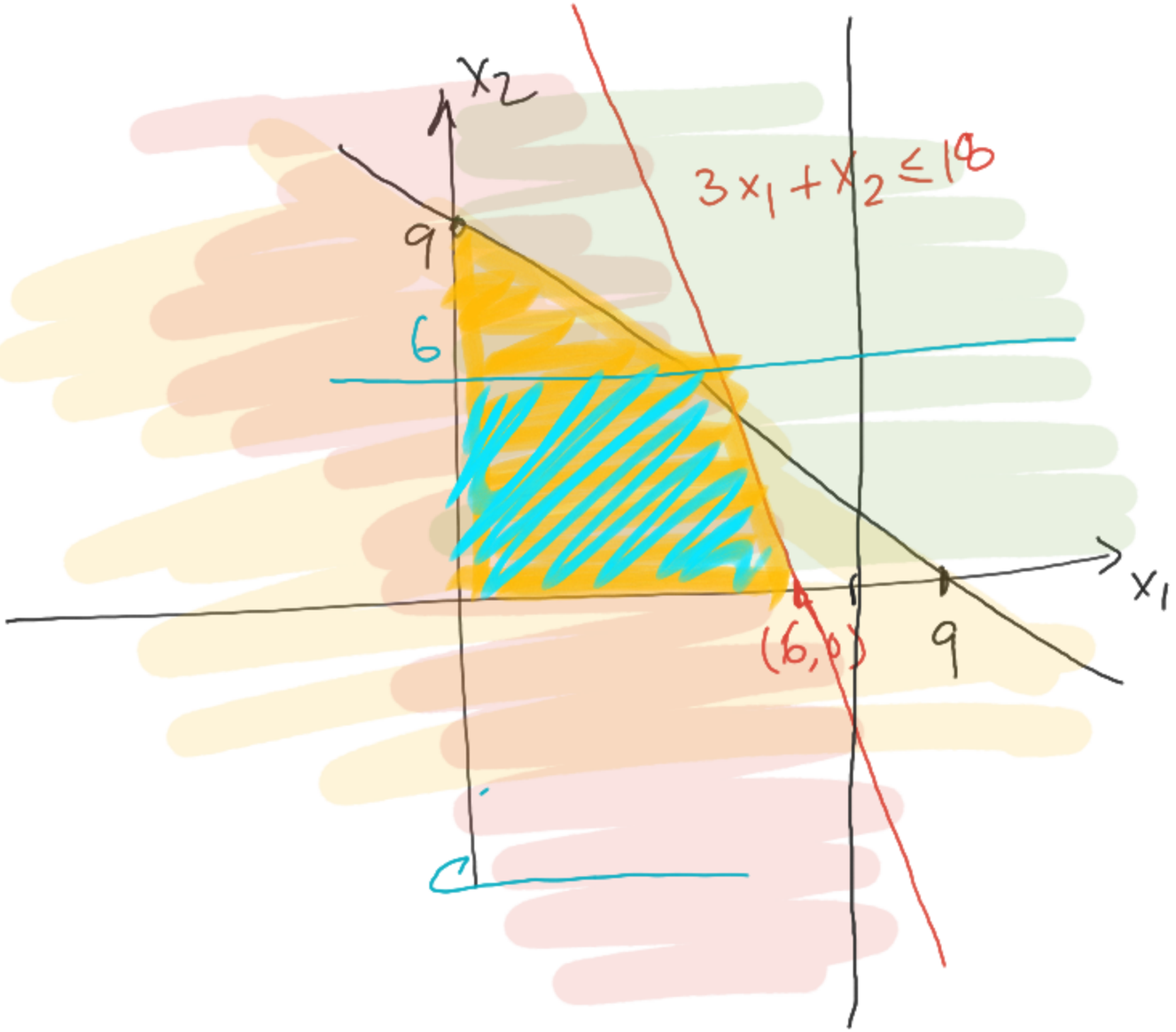
Min Cost Max flow.

↙

LPs




1/2
1/2
1


x_2
$3x_1 + x_2 \leq 18$
9
6
x_1
(6,0)
9