Announcements

- HW and section question solutions all on canvas
  - hw6 updated with more solutions
- sample prelim question solutions will on canvas Saturday
- Prelim 2: Tuesday March 24th in Statler auditorium
  - Topics: stable matching, flows and applications and NP-completeness
  - Information sheet on topics and sample question in canvas

- Section next week prelim review: it is optional, and feel free to attend the section of your choice

- HW8 will be divide and conquer, due Friday April 10

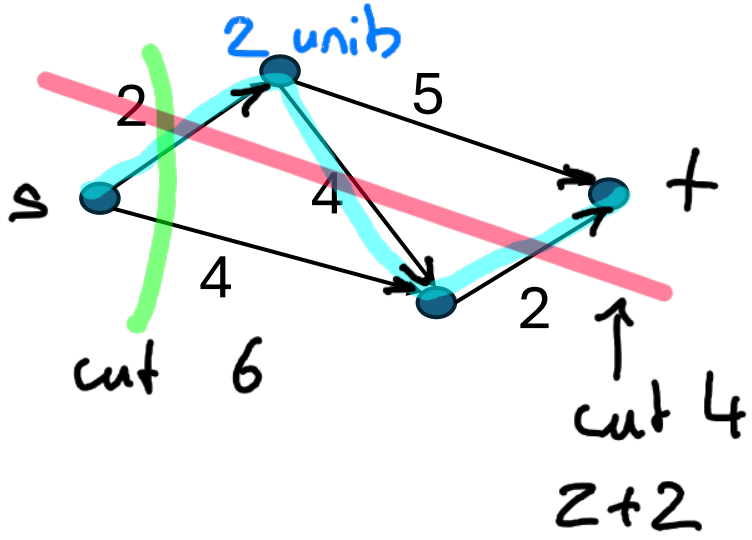
Prelim Review part I: stable matching & HW3

hw3 mean 9.5 median 10

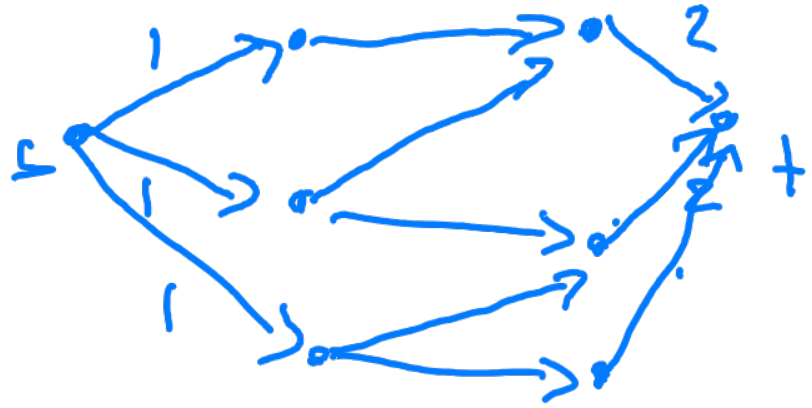
- Gale-Shapley
  - know basic properties

Max flow and min cuts and applications (hw4)

know Ford-Fulkerson, residual graphs & properties

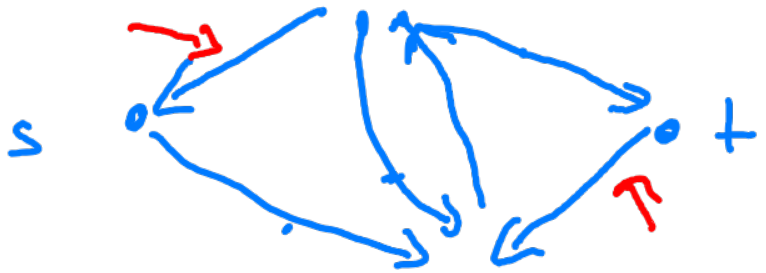


fair scheduling



Nok (s,v) or (w,t) edge never used backwards

residual graph



$\Rightarrow$ flow on these monotone increasing

marked edges never used backwards

Max flow and min cuts and applications (hw5)

will ask alg & run time only, no proof

flow

interview scheduling

(i,j) edge if j can interview i

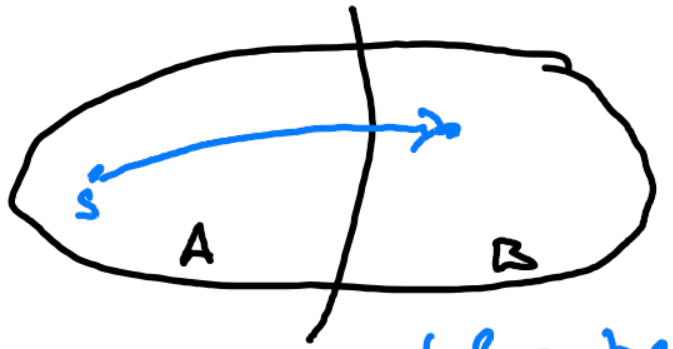
add available times

add middle nodes

candidates - times they offer

cut application

- project scheduling
- two building



cost for being in B can go here

Common mistake

running time $O(C \cdot m)$

FF $C = \sum_{e=(s,v)} c_e$ $m = \#\text{edges}$, use params in input

NP-completeness

OK to use: in class, hw, section, book
list posted: canvas & Ed

- ✓ Independent Set — very indep set — k small easy (e.g. k=1,2)
- ✓ Vertex Cover — monotone SAT — k huge easy
- ✓ Hamiltonian Path/Cycle — sequencing
- ✓ Subset Sum — huge numbers
- SAT and 3-SAT — generic, but likely harder

Steps X NP-c.
1. NP
2. select Y
3. construct $Y \leq X$
4 yes Y => yes X
5 yes X => yes Y

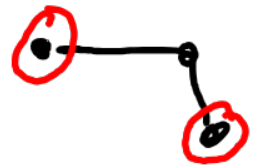
NP-completeness (hw 6-7 comments)

HW 7 ✓ vertex cover, monotone SAT
✓SAT
✓subset sum

HW 6 for X NP-complete

1. X in NP need hint, alg, time
2. choose Y NP-complete
3. reduction $Y \leq_p X$ & time it takes
4. Y yes $\Rightarrow$ X yes
   X yes $\Rightarrow$ Y yes

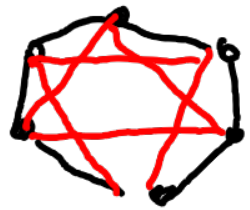
Very Indep Set



not very indep

backwards

VIS $\leq_p$ IS



VIS indep

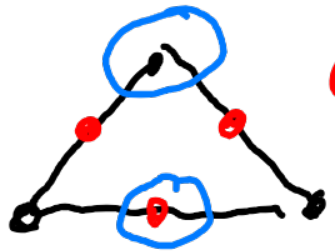
add edges (v,w) if G' G has (v,u) & (u,w)

S VIS in G $\Leftrightarrow$ S Indep G

IS $\leq$ VIS right direction

G input for IS

---->



G'

input for VIS

add node middle of edges

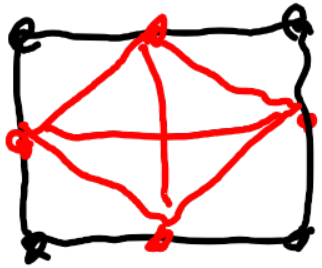
issue

S indep in G $\Rightarrow$ S Very independent G'

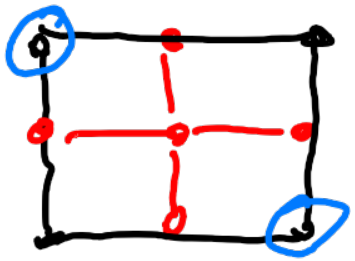
not working

Solutions

A



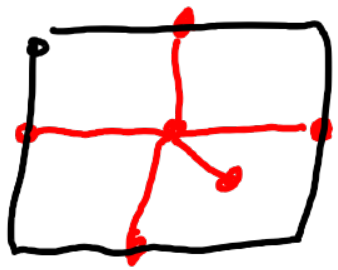
F



k+1

not working

B



k+1

works well

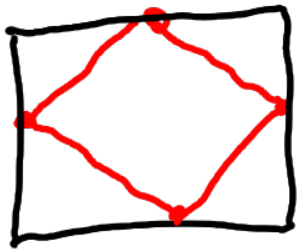
1. fully connect

⇐ picking new node only when $k=1$

max one red node

S indep G ⇒ S + extra one indep in G'

C



works well, but harder to prove