



# CSE 421

## Introduction to Algorithms

Winter 2024

Lecture 20

Network Flow Applications

# Today's topics

- Network flow reductions
  - Multi source flow
  - Reviewer Assignment
- Baseball Scheduling
- Image Segmentation
- Task Selection
- Reading: 7.5, 7.6, 7.10-7.12

# Multi-source network flow

- Multi-source network flow
  - Sources  $s_1, s_2, \dots, s_k$
  - Sinks  $t_1, t_2, \dots, t_j$
- Solve with Single source network flow

# Resource Allocation: Assignment of reviewers

- A set of papers  $P_1, \dots, P_n$
- A set of reviewers  $R_1, \dots, R_m$
- Paper  $P_i$  requires  $A_i$  reviewers
- Reviewer  $R_j$  can review  $B_j$  papers
- For each reviewer  $R_j$ , there is a list of paper  $L_{j1}, \dots, L_{jk}$  that  $R_j$  is qualified to review

# Resource Allocation: Illegal Campaign Donations

- Candidates  $C_1, \dots, C_n$ 
  - Donate  $b_i$  to  $C_i$
- With a little help from your friends
  - Friends  $F_1, \dots, F_m$
  - $F_i$  can give  $a_{ij}$  to candidate  $C_j$
  - You can give at most  $M_i$  to  $F_i$

# Baseball elimination

- Can the Dinosaurs win the league?
- Remaining games:
  - AB, AC, AD, AD, AD, BC, BC, BC, BD, CD

	W	L
Ants	4	2
Bees	4	2
Cockroaches	3	3
Dinosaurs	1	5

A team **wins** the league if it has strictly more wins than any other team at the end of the season  
A team **ties** for first place if no team has more wins, and there is some other team with the same number of wins

# Baseball elimination

- Can the Fruit Flies win or tie the league?
- Remaining games:
  - AC, AD, AD, AD, AF, BC, BC, BC, BC, BC, BD, BE, BE, BE, BE, BF, CE, CE, CE, CF, CF, DE, DF, EF, EF

	W	L
Ants	17	12
Bees	16	7
Cockroaches	16	7
Dinosaurs	14	13
Earthworms	14	10
Fruit Flies	12	15

# Assume Fruit Flies win remaining games

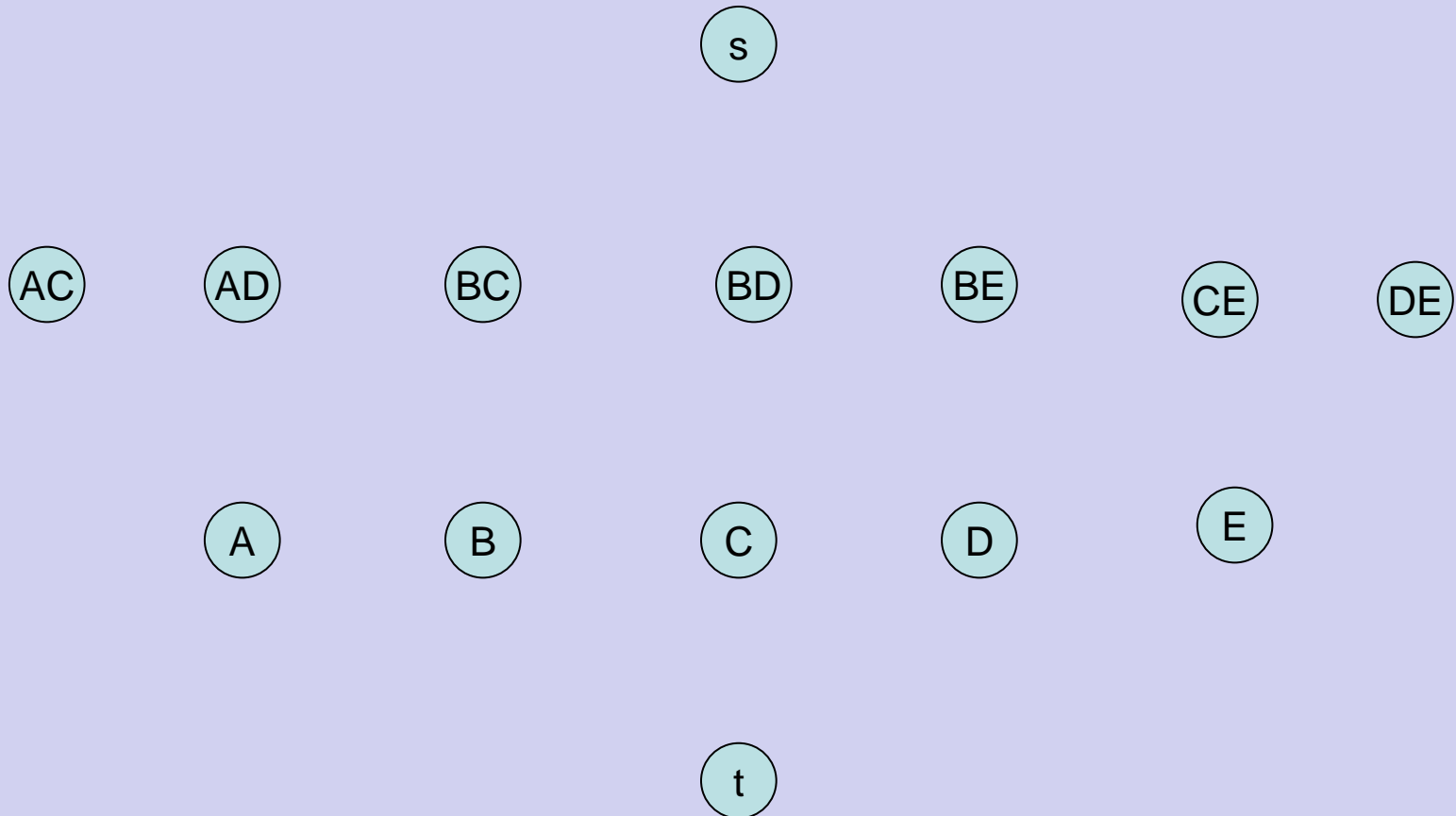
- Fruit Flies are tied for first place if no team wins more than 19 games
- Allowable wins
  - Ants (2)
  - Bees (3)
  - Cockroaches (3)
  - Dinosaurs (5)
  - Earthworms (5)
- 18 games to play
  - AC, AD, AD, AD, BC, BC, BC, BC, BC, BD, BE, BE, BE, BE, CE, CE, CE, DE

	W	L
Ants	17	13
Bees	16	8
Cockroaches	16	9
Dinosaurs	14	14
Earthworms	14	12
Fruit Flies	19	15



# Remaining games

AC, AD, AD, AD, BC, BC, BC, BC, BC, BD, BE, BE, BE, BE, CE, CE, CE, DE



# Minimum Cut Applications

- Image Segmentation
- Open Pit Mining / Task Selection Problem
- Reduction to Min Cut problem

$S, T$  is a cut if  $S, T$  is a partition of the vertices with  $s$  in  $S$  and  $t$  in  $T$

The capacity of an  $S, T$  cut is the sum of the capacities of all edges going from  $S$  to  $T$

# Image Segmentation



# Separate Lion from Savana



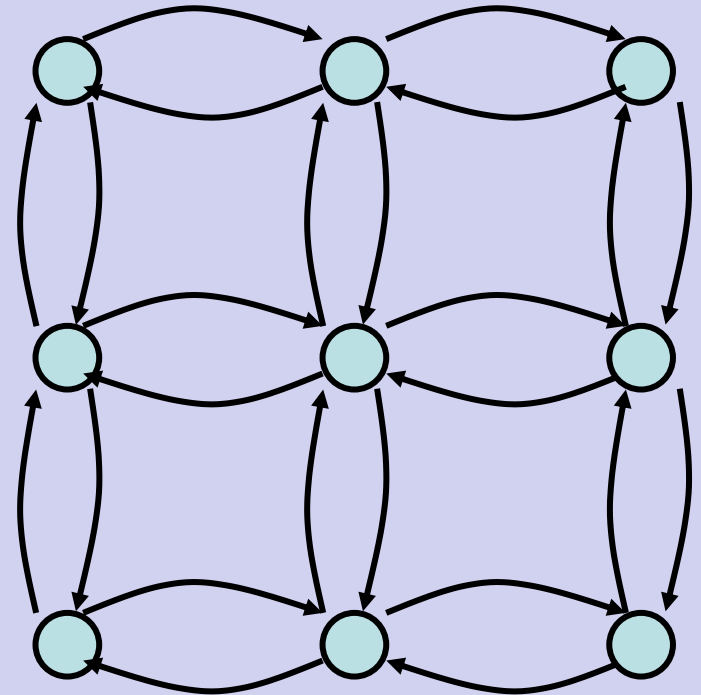
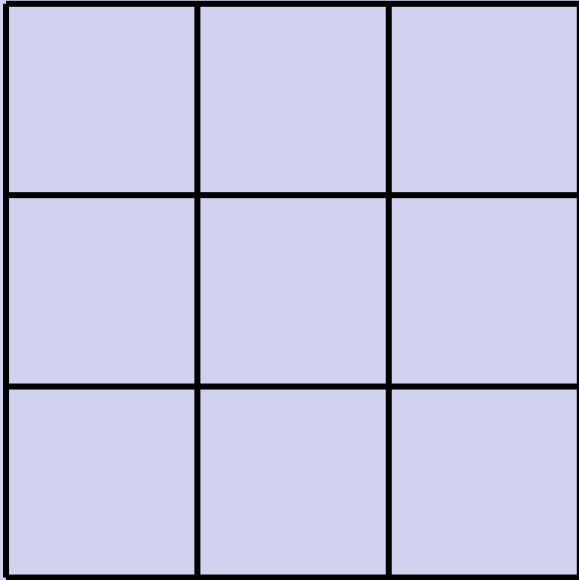


# Image analysis

- $a_i$ : value of assigning pixel  $i$  to the foreground
- $b_i$ : value of assigning pixel  $i$  to the background
- $p_{ij}$ : penalty for assigning  $i$  to the foreground,  $j$  to the background or vice versa
- $A$ : foreground,  $B$ : background
- $Q(A,B) = \sum_{\{i \text{ in } A\}} a_i + \sum_{\{j \text{ in } B\}} b_j - \sum_{\{(i,j) \text{ in } E, i \text{ in } A, j \text{ in } B\}} p_{ij}$

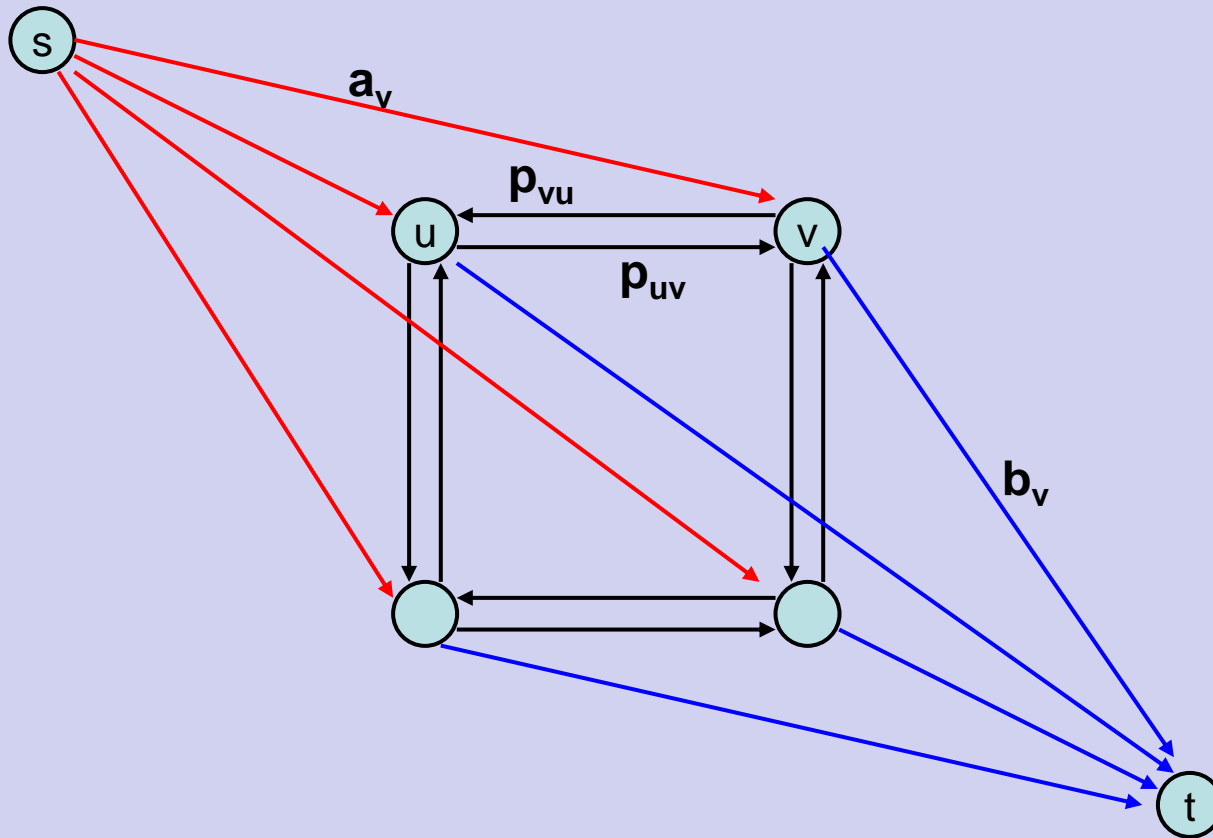
# Pixel graph to flow graph

s



t

# Mincut Construction





# Open Pit Mining



# Application of Min-cut

- Open Pit Mining Problem
- Task Selection Problem
- Reduction to Min Cut problem

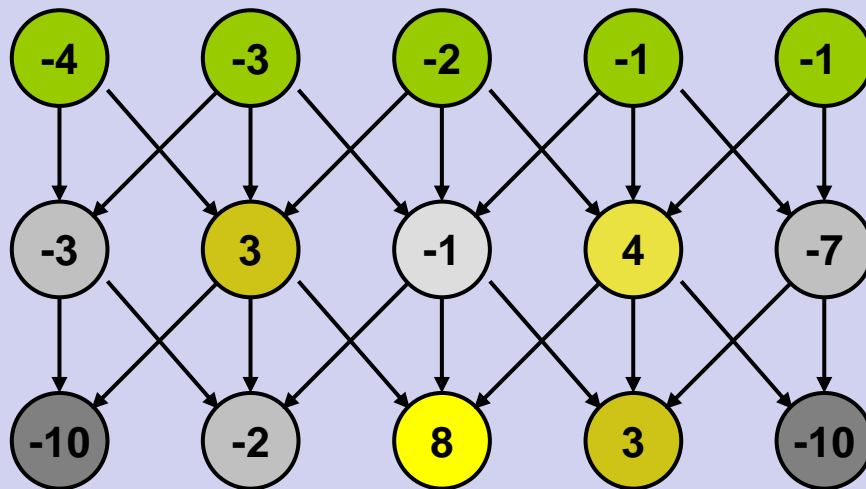
$S, T$  is a cut if  $S, T$  is a partition of the vertices with  $s$  in  $S$  and  $t$  in  $T$

The capacity of an  $S, T$  cut is the sum of the capacities of all edges going from  $S$  to  $T$

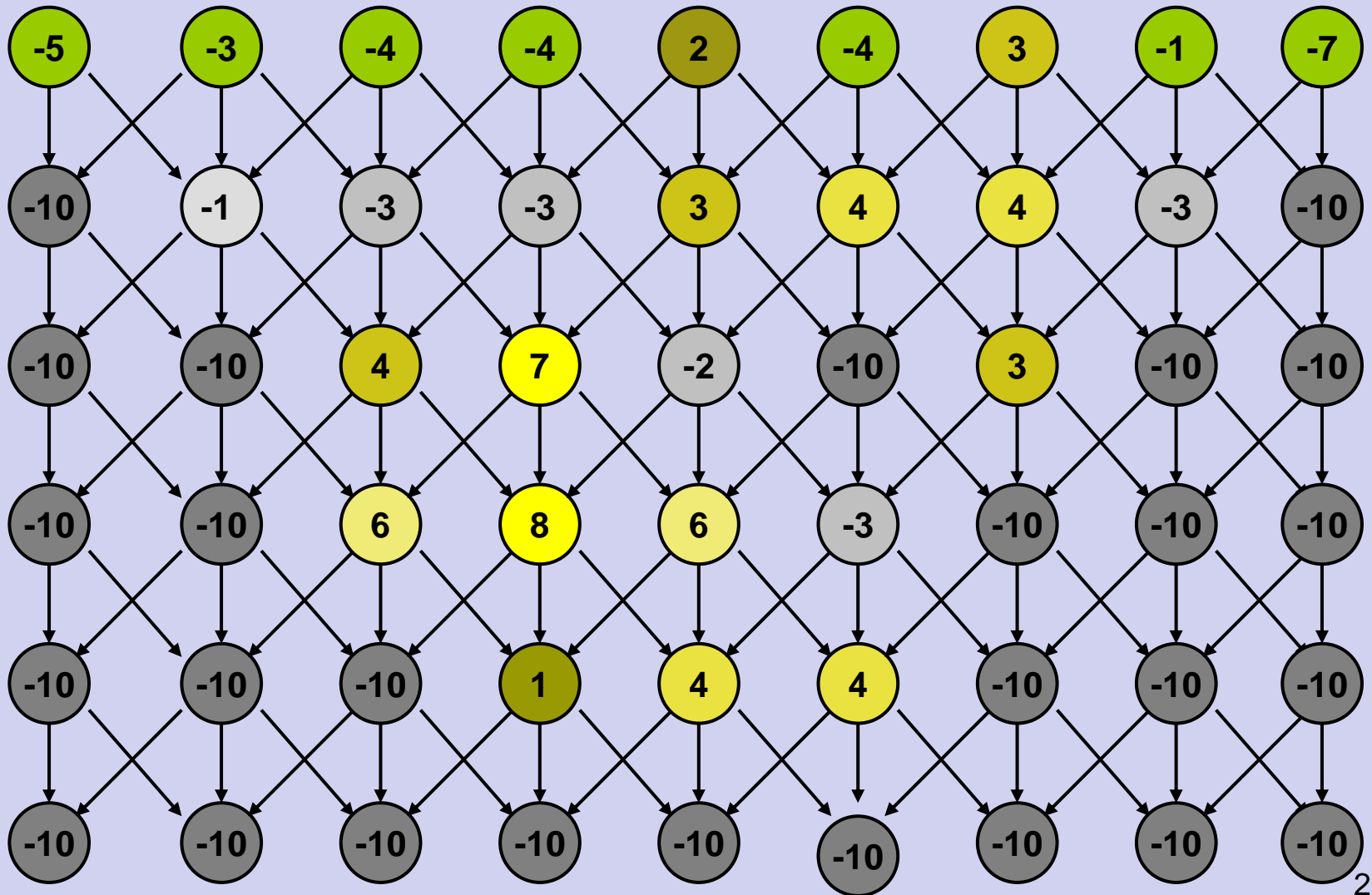
# Open Pit Mining

- Each unit of earth has a profit (possibly negative)
- Getting to the ore below the surface requires removing the dirt above
- Test drilling gives reasonable estimates of costs
- Plan an optimal mining operation

# Mine Graph

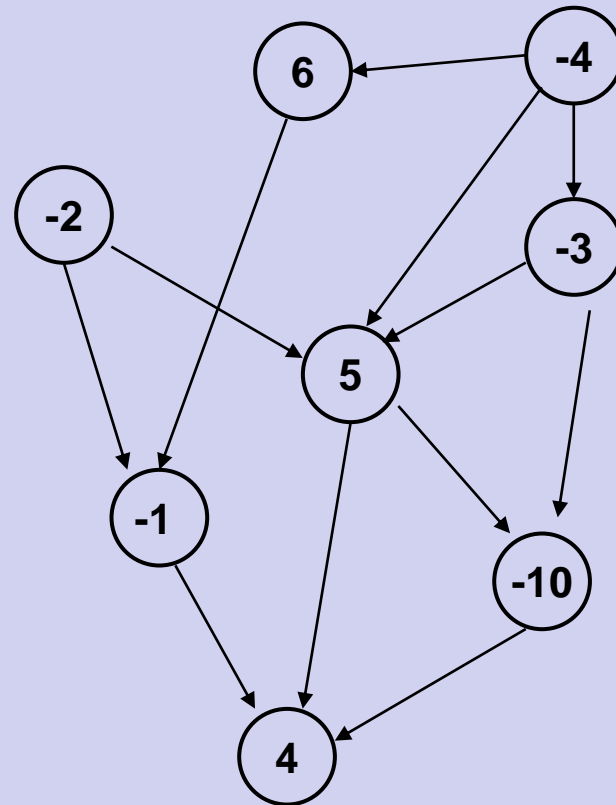


# Determine an optimal mine



# Generalization

- Precedence graph  $G=(V,E)$
- Each  $v$  in  $V$  has a profit  $p(v)$
- A set  $F$  is *feasible* if when  $w$  in  $F$ , and  $(v,w)$  in  $E$ , then  $v$  in  $F$ .
- Find a feasible set to maximize the profit

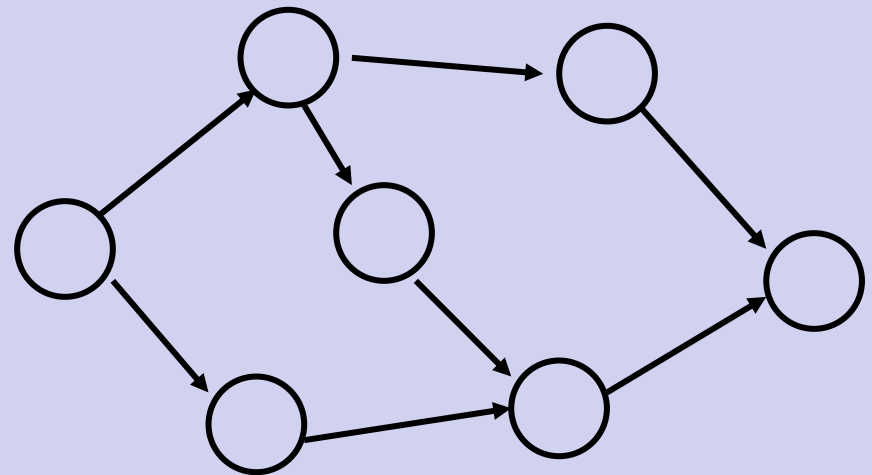


# Min cut algorithm for profit maximization

- Construct a flow graph where the minimum cut identifies a feasible set that maximizes profit

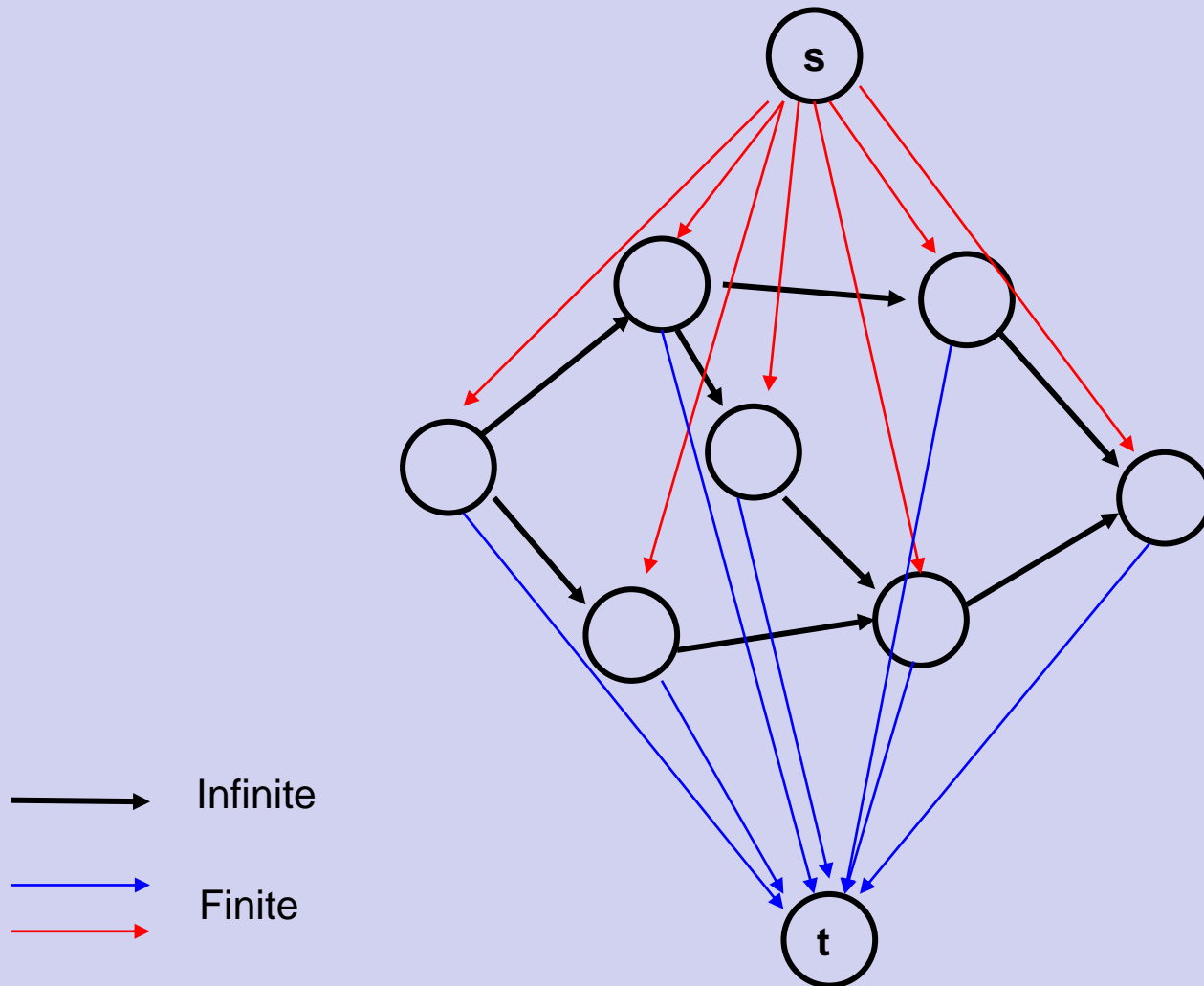
# Precedence graph construction

- Precedence graph  $G=(V,E)$
- Each edge in  $E$  has infinite capacity
- Add vertices  $s, t$
- Each vertex in  $V$  is attached to  $s$  and  $t$  with finite capacity edges



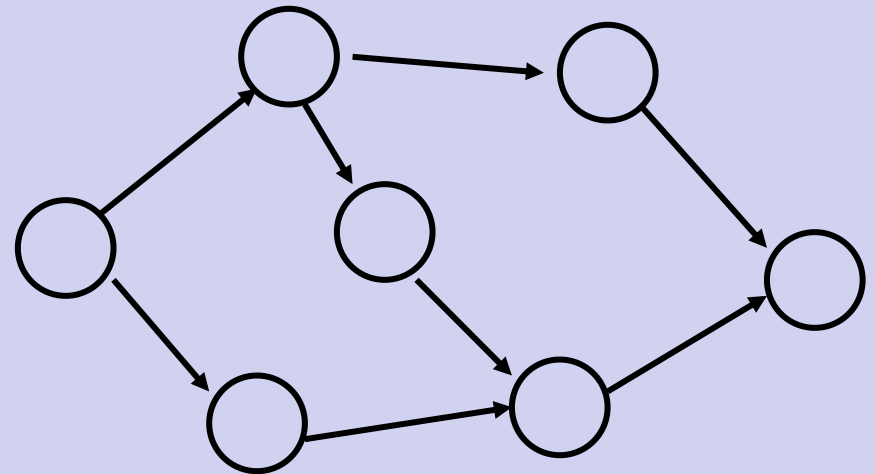


Find a **finite** value cut with at least two vertices on each side of the cut



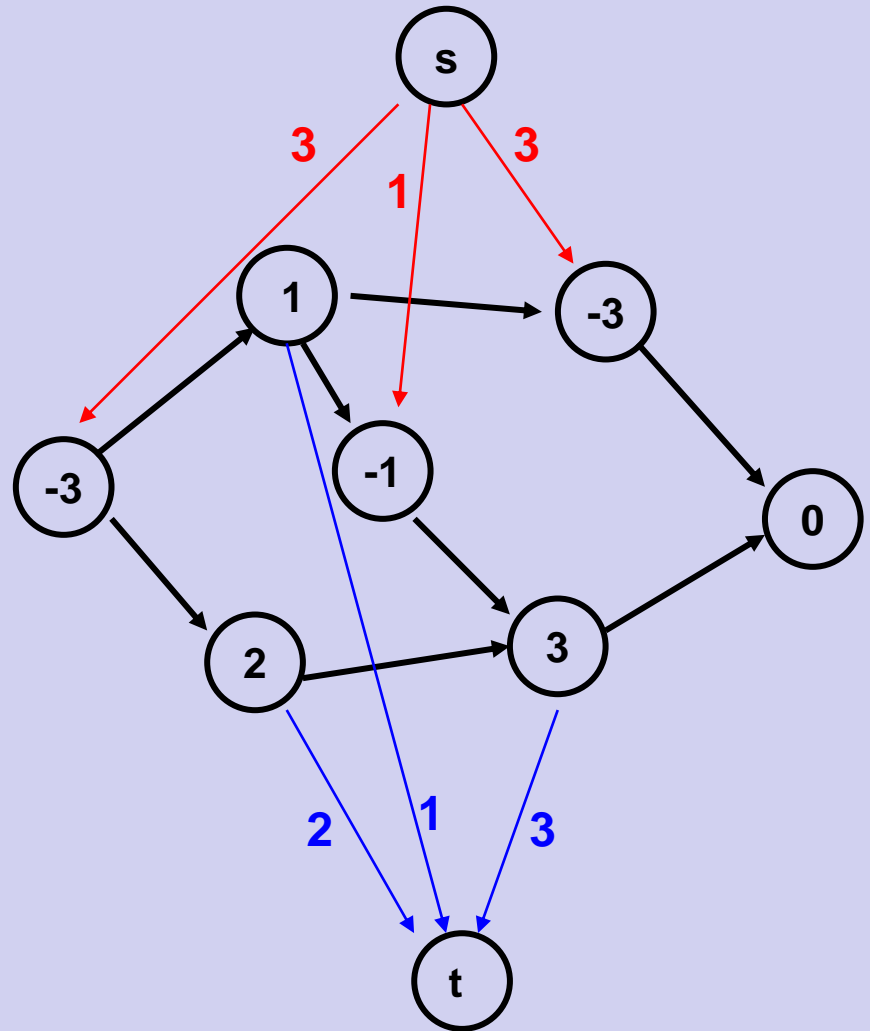
# The sink side of a finite cut is a feasible set

- No edges permitted from  $S$  to  $T$
- If a vertex is in  $T$ , all of its ancestors are in  $T$

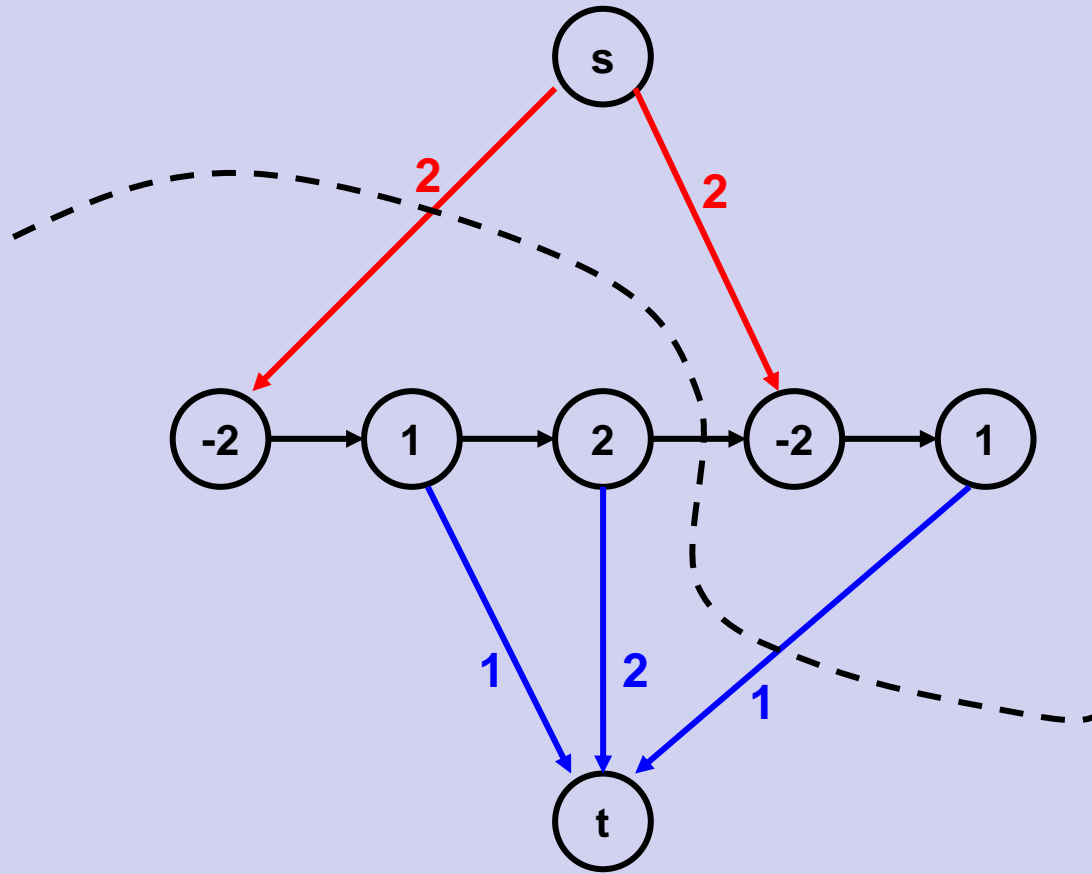


# Setting the costs

- If  $p(v) > 0$ ,
  - $\text{cap}(v,t) = p(v)$
  - $\text{cap}(s,v) = 0$
- If  $p(v) < 0$ 
  - $\text{cap}(s,v) = -p(v)$
  - $\text{cap}(v,t) = 0$
- If  $p(v) = 0$ 
  - $\text{cap}(s,v) = 0$
  - $\text{cap}(v,t) = 0$



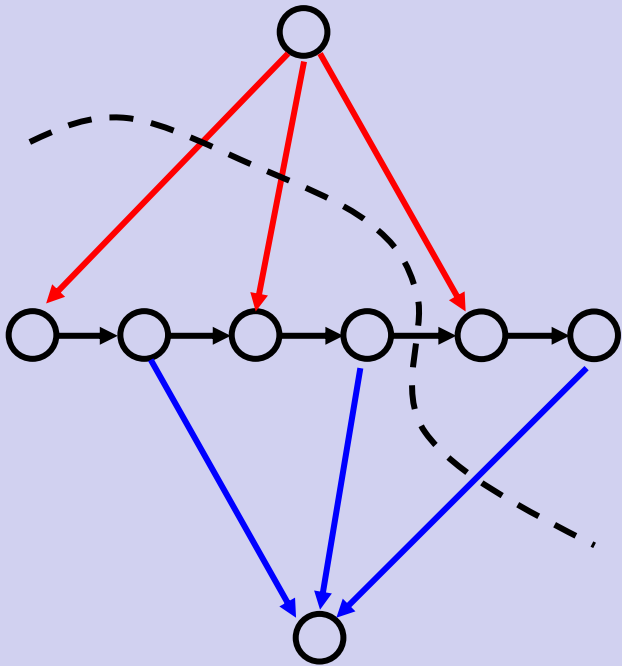
# Minimum cut gives optimal solution Why?



# Computing the Profit

- $\text{Cost}(W) = \sum_{\{w \text{ in } W; p(w) < 0\}} -p(w)$
- $\text{Benefit}(W) = \sum_{\{w \text{ in } W; p(w) > 0\}} p(w)$
- $\text{Profit}(W) = \text{Benefit}(W) - \text{Cost}(W)$
  
- Maximum cost and benefit
  - $C = \text{Cost}(V)$
  - $B = \text{Benefit}(V)$

Express  $\text{Cap}(S,T)$  in terms of  $B$ ,  $C$ ,  $\text{Cost}(T)$ ,  $\text{Benefit}(T)$ , and  $\text{Profit}(T)$



$$\begin{aligned}\text{Cap}(S,T) &= \text{Cost}(T) + \text{Ben}(S) = \text{Cost}(T) + \text{Ben}(S) + \text{Ben}(T) - \text{Ben}(T) \\ &= B + \text{Cost}(T) - \text{Ben}(T) = B - \text{Profit}(T)\end{aligned}$$