

CSE 421 Algorithms

Richard Anderson

Lecture 24

Network Flow Applications

Today's topics

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

Multi-source network flow

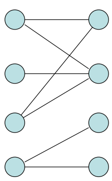
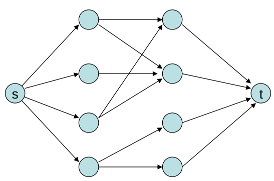
- Multi-source network flow
  - Sources $s_1, s_2, \ldots, s_k$
  - Sinks $t_1, t_2, \ldots, t_j$
- Solve with Single source network flow

Review

Bipartite Matching

- A graph G=(V,E) is bipartite if the vertices can be partitioned into disjoints sets X,Y
- A matching M is a subset of the edges that does not share any vertices
- Find a matching as large as possible

Converting Matching to Network Flow

Resource Allocation: Assignment of reviewers

- A set of papers $P_1, \ldots, P_n$
- A set of reviewers $R_1, \ldots, R_m$
- 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 $L_{j1}, \ldots, L_{jk}$ that R_j is qualified to review

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

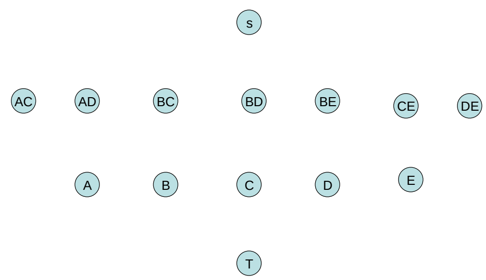
Assume Fruit Flies win remaining games

- 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

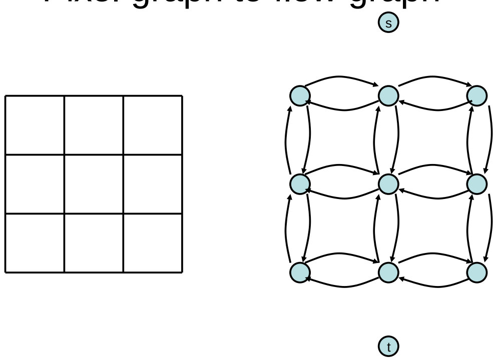
Image Segmentation

- Separate foreground from background

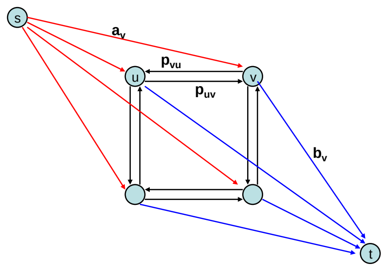
Image analysis

- 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
- $Q(A,B) = \Sigma_{\{i \text{ in } A\}} a_i + \Sigma_{\{j \text{ in } B\}} b_j - \Sigma_{\{(i,j) \text{ in } E, \text{ i in } A, \text{ j in } B\}} p_{ij}$

Pixel graph to flow graph



Mincut Construction



Open Pit Mining

Application of Min-cut

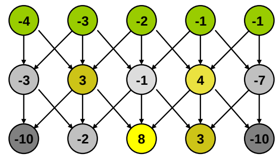
- Open Pit Mining Problem
- Task Selection Problem
- Reduction to Min Cut problem

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

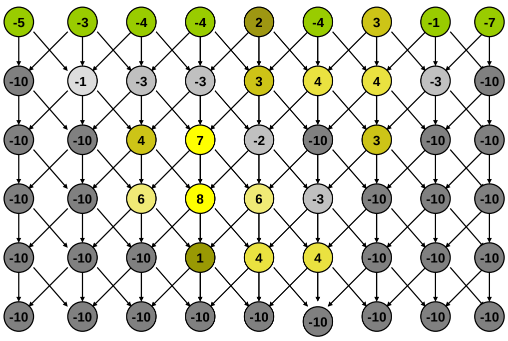
Open Pit Mining

- 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

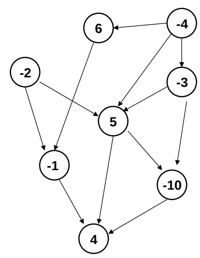


Determine an optimal mine



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

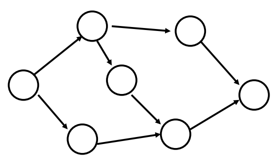


Min cut algorithm for profit maximization

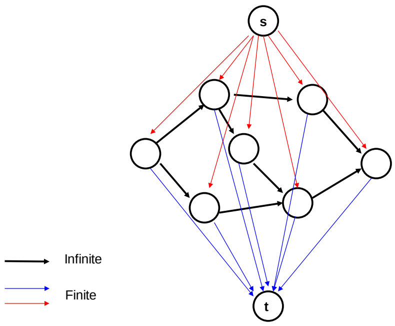
- Construct a flow graph where the minimum cut identifies a feasible set that maximizes profit

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

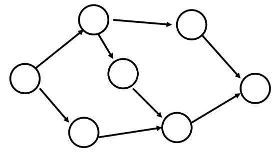


Find a finite value cut with at least two vertices on each side of the cut



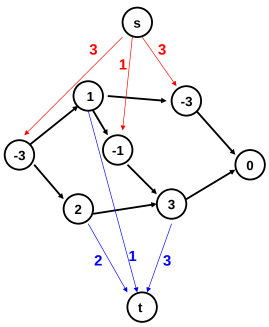
The sink side of a finite cut is a feasible set

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

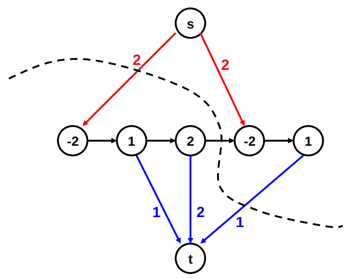


Setting the costs

- If p(v) > 0,
  - cap(v,t) = p(v)
  - cap(s,v) = 0
- If p(v) < 0
  - cap(s,v) = -p(v)
  - cap(v,t) = 0
- If p(v) = 0
  - cap(s,v) = 0
  - cap(v,t) = 0



Minimum cut gives optimal solution Why?

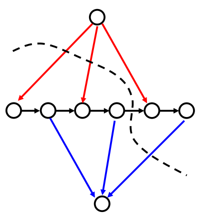


Computing the Profit

- Cost(W) = $\Sigma_{\{w \text{ in } W;\ p(w) < 0\}}$-p(w)
- Benefit(W) = $\Sigma_{\{w \text{ in } W;\ p(w) > 0\}}$ p(w)
- Profit(W) = Benefit(W) – Cost(W)

- Maximum cost and benefit
  - C = Cost(V)
  - B = Benefit(V)

Express Cap(S,T) in terms of B, C, Cost(T), Benefit(T), and Profit(T)



Cap(S,T) = Cost(T) + Ben(S) = Cost(T) + Ben(S) + Ben(T) – Ben(T)
= B + Cost(T) – Ben(T) = B – Profit(T)