


# CSE 421
# Algorithms

Autumn 2019

Lecture 24

Network Flow Applications

# Announcements

- Homework 9: Due Wednesday, Nov 27
- Homework 10: Due Friday, Dec 6
- Final Exam: Monday, Dec 9, 2:30 PM

| | |
|---|---|
| Fri, Nov 22 | Net Flow Applications |
| Mon, Nov 25 | Net Flow Applications |
| Wed, Nov 27 | NP-Completeness |
| Fri, Nov 29 | Holiday |
| Mon, Dec 2 | NP-Completeness |
| Wed, Dec 4 | NP-Completeness |
| Fri, Dec 6 | Beyond NP-Completeness |

# Today's topics

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

Review

# Network Flow Definitions

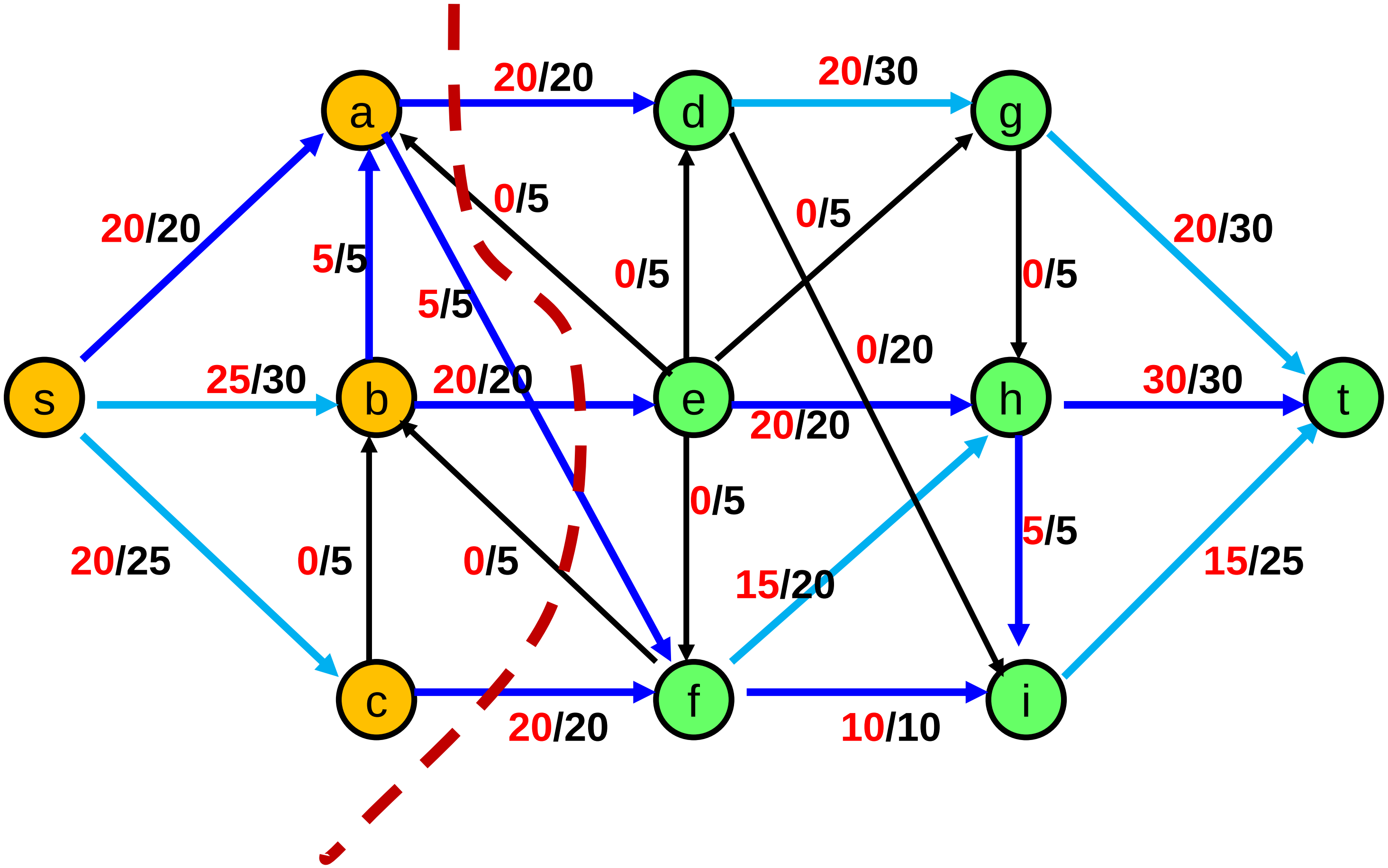
- Flowgraph: Directed graph with distinguished vertices s (source) and t (sink)
- Capacities on the edges, c(e) >= 0
- Problem, assign flows f(e) to the edges such that:
  - 0 <= f(e) <= c(e)
  - Flow is conserved at vertices other than s and t
    - Flow conservation: flow going into a vertex equals the flow going out
  - The flow leaving the source is a large as possible

# Key Ideas for Network Flow

- Residual Graph for a Flow
- Augmenting a flow
- Ford Fulkerson Algorithm
- Max Flow / Min Cut Theorem
- Practical Flow Algorithms
- Modelling problems as Network Flow or Minimum Cut

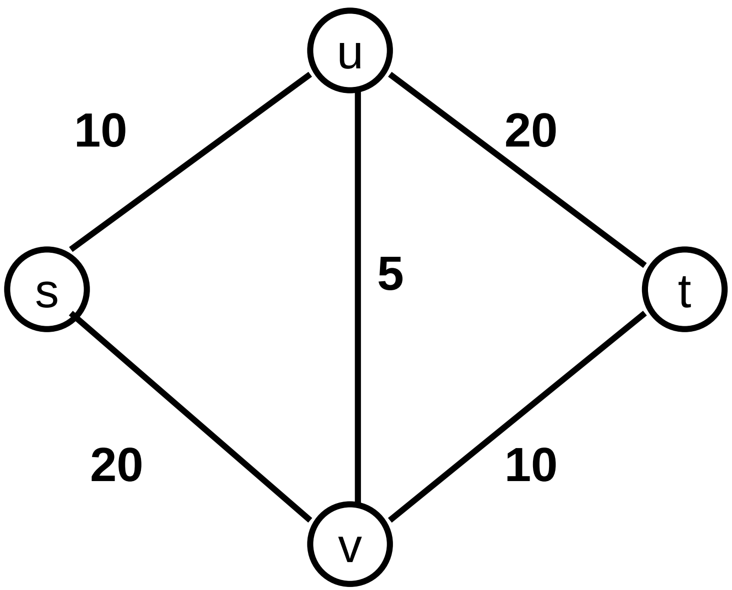
Review

# Max Flow / Min Cut



# Undirected Network Flow

- Undirected graph with edge capacities
- Flow may go either direction along the edges (subject to the capacity constraints)



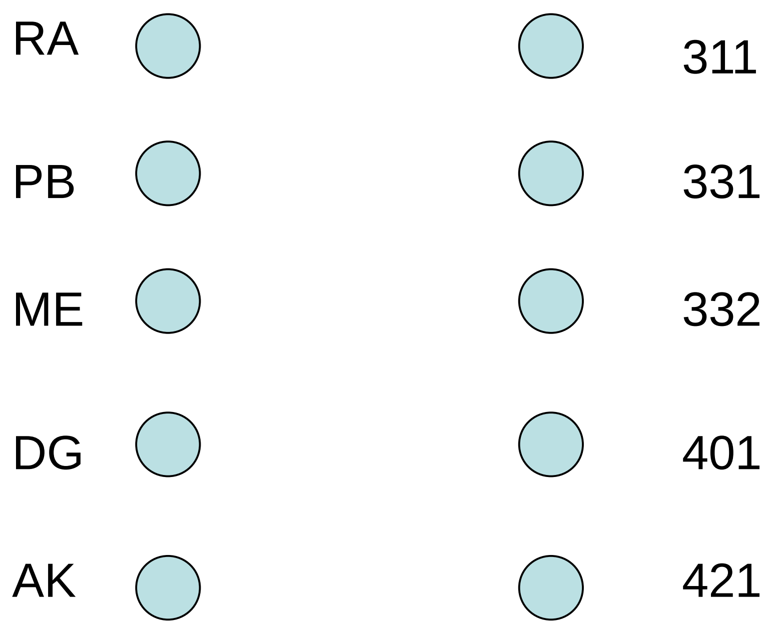

Construct an equivalent flow problem

# Bipartite Matching

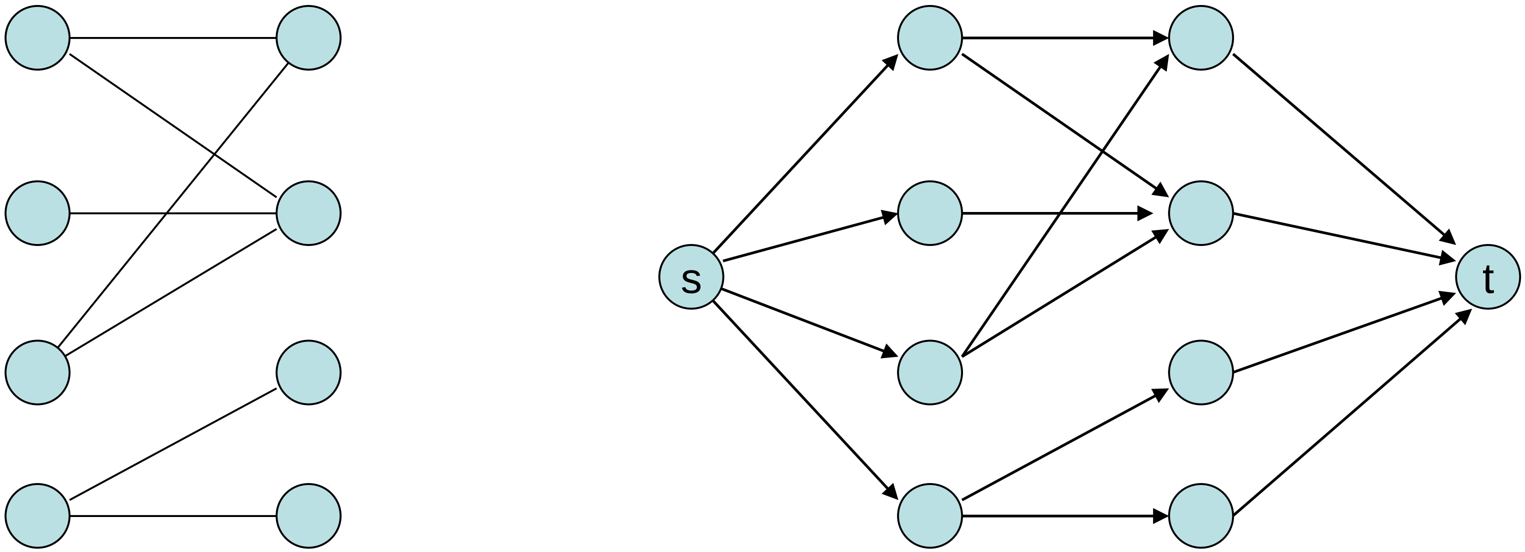
- A graph G=(V,E) is bipartite if the vertices can be partitioned into disjoints sets X,Y
- A matching M is a subset of the edges that does not share any vertices
- Find a matching as large as possible

# Application

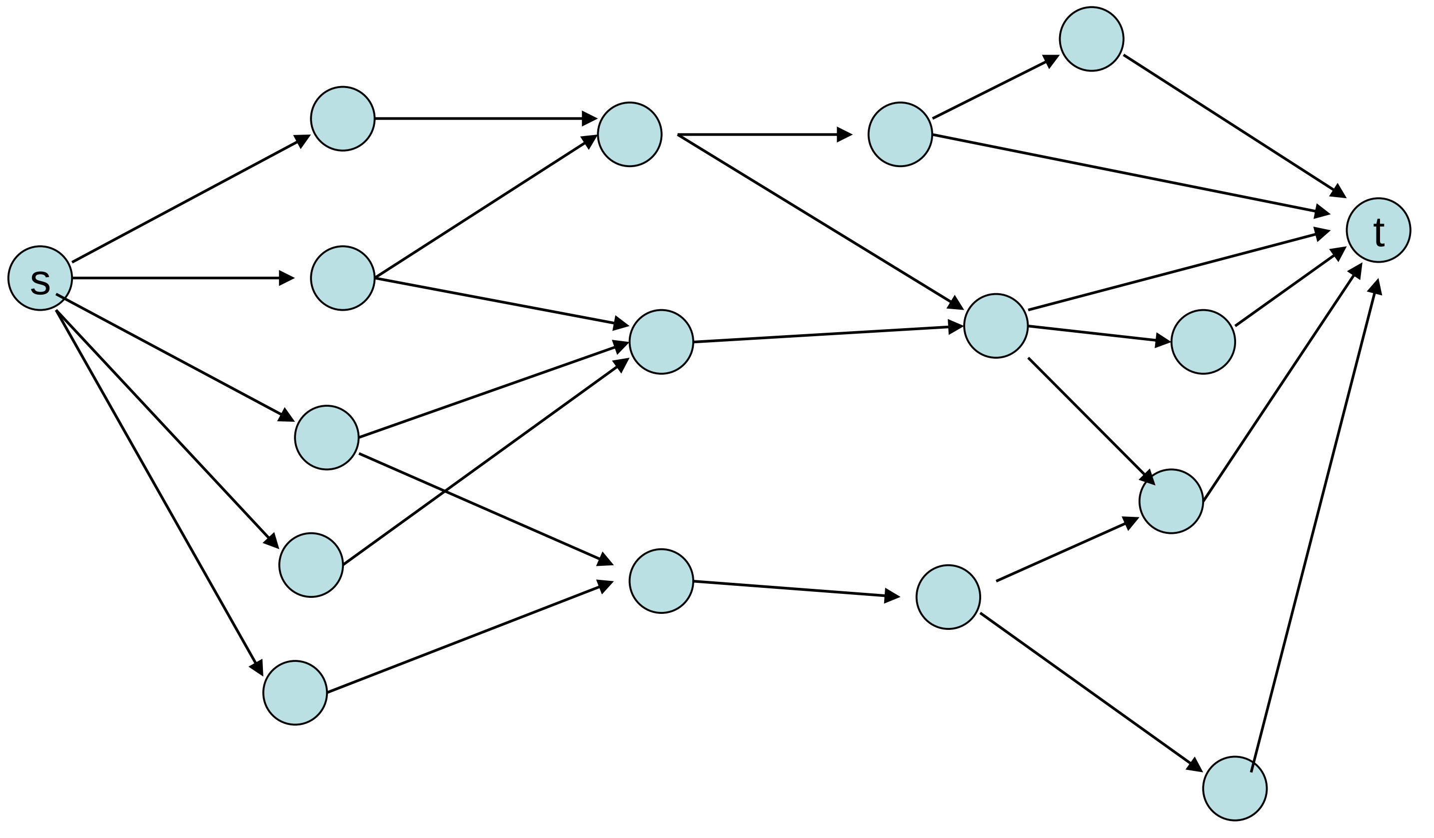
- A collection of teachers
- A collection of courses
- And a graph showing which teachers can teach which courses



# Converting Matching to Network Flow



# Finding edge disjoint paths




Construct a maximum cardinality set of edge disjoint paths

# 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

# 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

# Resource Allocation: Illegal Campaign Donations

- Candidates $C_i$, . . ., $C_n$
  - Donate $b_i$ to $C_i$
- With a little help from your friends
  - Friends $F_1$, . . ., $F_m$
  - $F_i$ 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 |

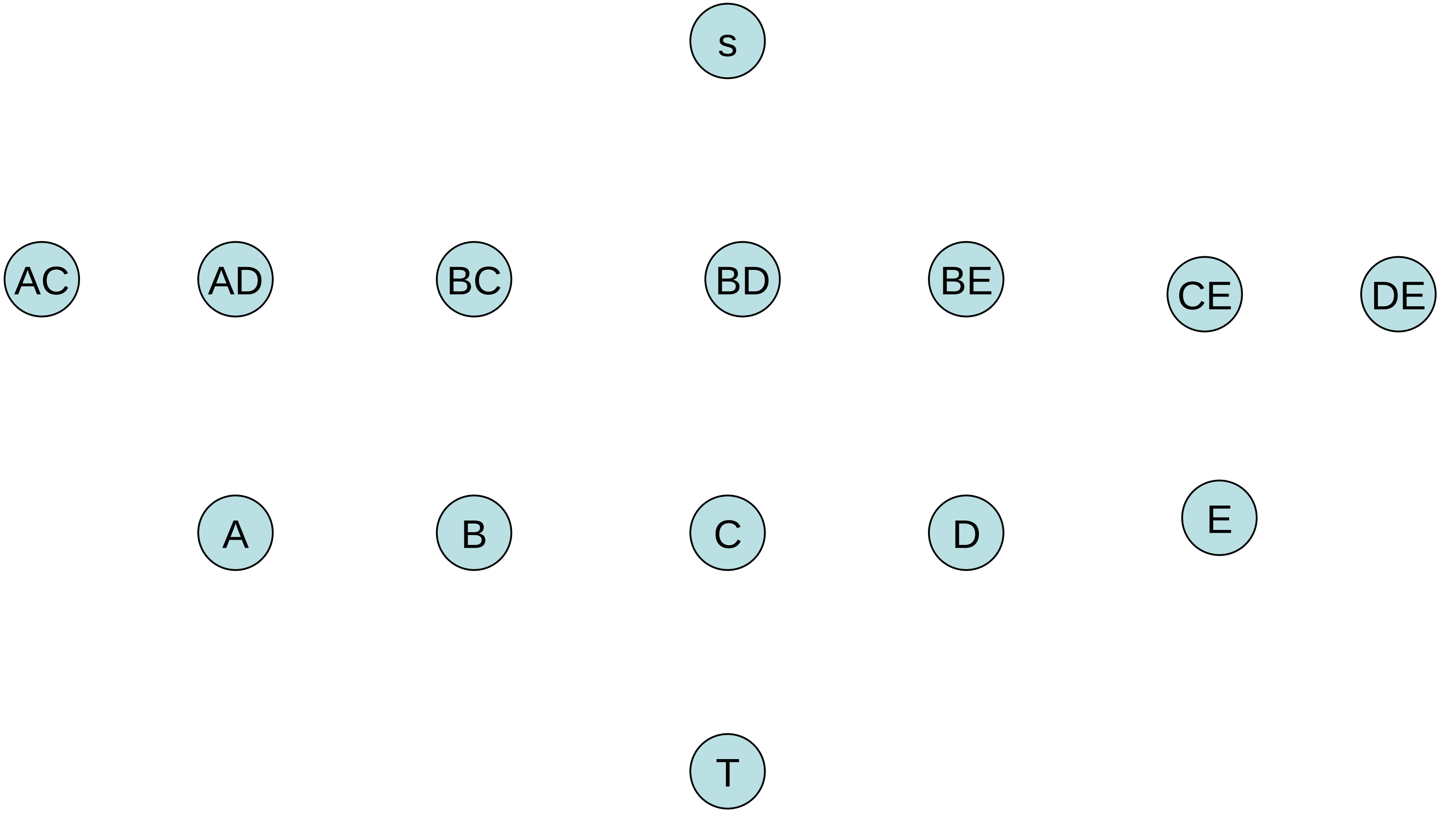
# Assume Fruit Flies win remaining games

- 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

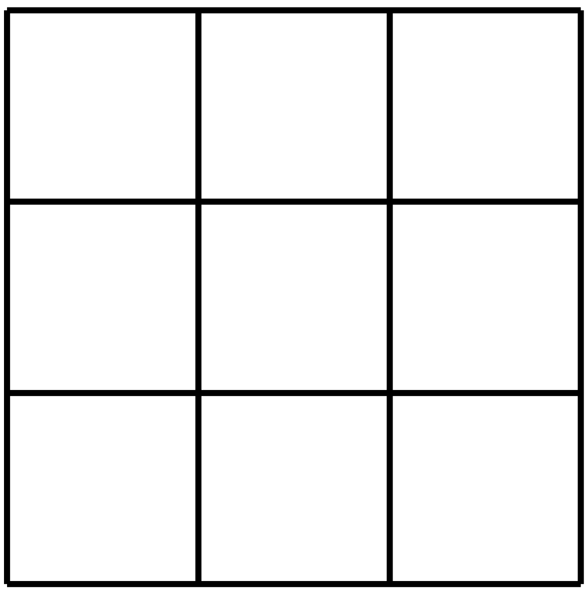
# Image Segmentation

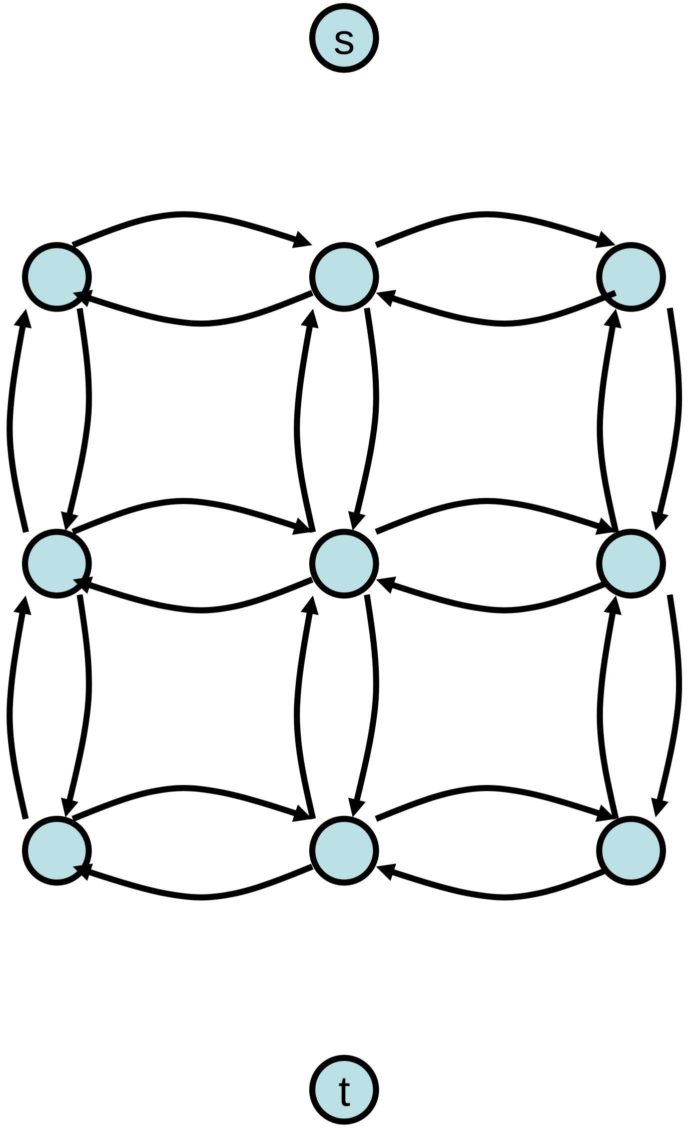
- Separate foreground from background

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

# Pixel graph to flow graph





# Mincut Construction