CPSC 490 – Problem Solving in Computer Science

Lecture 25: Practice with Flow, Flow with Demands

Jason Chiu and Raunak Kumar

2017/03/24

University of British Columbia

Problem 1 - Binary Search

Given undirected graph, assign direction to each edge so that maximum of in-degree of any vertex is minimized.

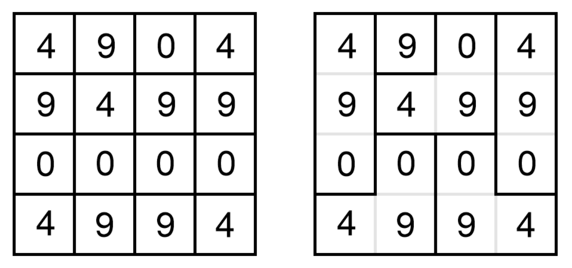
Problem 1 - Solution

Binary search for the minimum d, to determine if d is possible, form this flow graph

- For every vertex and every edge in original graph, add a vertex
- $s \to i$ with capacity 1, where i is edge
- $i \to u$ and $i \to v$ with capacity 1, where i is edge between u and v
- $u \to t$ with capacity d

Problem 2 - Where To Flow?

Cut out the maximum number of triominos with the sequence of digits $4 \to 9 \to 0$ (4 and 0 on each end and 9 in the middle)



Problem 2 - Solution

Connect $s \to 4 \to 9 \to 0 \to t$ with vertex capacity 1

Problem 3 - Image Segmentation

You are given an image.

It is represented as an undirected graph $G = (V, E)$. $v \in V$ is a pixel. $e = (u, v) \in E$ is a pair of pixels that are neighbours.

The goal is to classify pixels as foreground or background.

Problem 3 - Image Segmentation

X = foreground pixels, Y = background pixels.

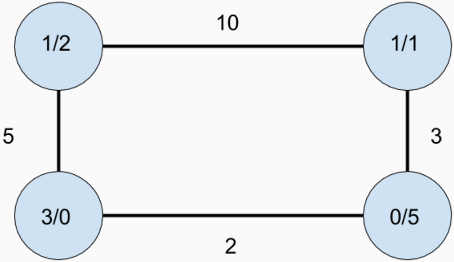
The input also contains:

- $\forall v \in V.\ a_v$: prize obtained when $v \in X$ (foreground)
- $\forall v \in V.\ b_v$: prize obtained when $v \in Y$ (background)
- $\forall e = (u, v) \in E.\ p_e$: penalty of classifying u and v in different parts

Find a classification that maximizes the following objective

$$\sum_{v \in X} a_v + \sum_{v \in Y} b_v - \sum_{e \in \delta(X)} p_e$$

Problem 3 - Image Segmentation



Each vertex v has a_v / b_v

Problem 3 - Solution

Transforming the Objective:

Maximizing

$$\sum_{v \in X} a_v + \sum_{v \in Y} b_v - \sum_{e \in \delta(X)} p_e$$

is the same as minimizing

$$\sum_{e \in \delta(X)} p_e - \sum_{v \in X} a_v - \sum_{v \in Y} b_v$$

Problem 3 - Solution

Transforming the Objective:

Since adding a constant does not change the solution, we can instead minimize the following:

$$\sum_{e \in \delta(X)} p_e + \sum_{v \in Y} a_v + \sum_{v \in X} b_v$$

Problem 3 - Solution

Transforming the graph:

First, convert from undirected to directed graph.
$\Rightarrow$ duplicate each edge, in either direction with the same penalty.

Problem 3 - Solution

Transforming the graph:

- Add a source s and a sink t.
- Add an edge (s, v) with capacity a_v.
- Add an edge (v, t) with capacity b_v.

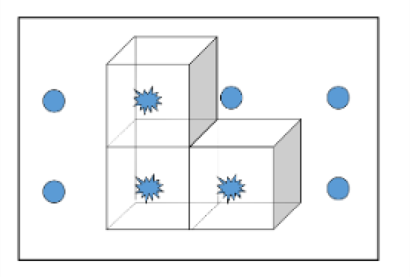
The capacity of the minimum cut in this flow network is identical to our objective function!

Problem 4 - 3D Geometry

Space is divided into $1 \times 1 \times 1$ sectors

Use as few 1×1 barriers as possible to box in the bad sectors!

You must place barriers axially aligned and at integer coordinates. Multiple boxes are allowed. Bad sectors are in the cube $[0, 10]^3$.



Source: ACM ICPC Pacific Northwest Regionals 2014

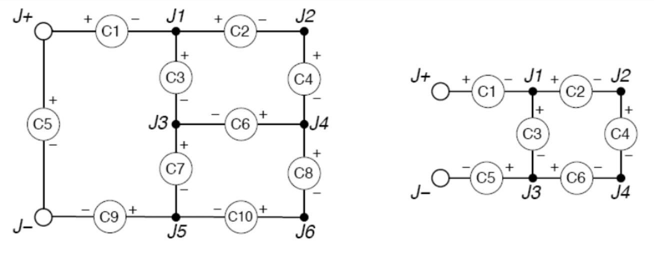
Problem 4 - Solution

This is a minimum cut problem. Observe that all bad sectors sits in $[0, 10]^3$ so optimal box is also inside this cube.

- Edges between adjacent sectors with capacity 1
- Edge from source $\rightarrow$ bad sector with capacity ∞
- Edge from every boundary sector of the $[0, 10]^3$ cube to sink with capacity 1

Problem 5 - (Pseudo-) Electrical Engineering

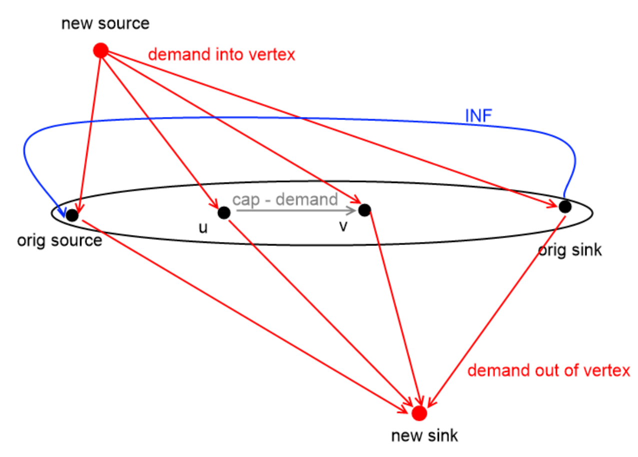
Each component in the circuit has a minimum current necessary to function. What's the minimum current you need to send from $+$ terminal to $-$ terminal?



Source: ACM ICPC Pacific Northwest Regionals 2008

Problem 5 - Solution

Each edge demands flow! Transform flow graph G into G' as follows:



Problem 5 - Solution

Each edge demands flow! Transform flow graph G into G' as follows:

- Add new source s' and new sink t'
- Add $s' \to u$ with net demand into u
- Add $u \to t'$ with net demand out of u
- Add $t \to s$ with infinite capacity
- Original edges have new capacity $=$ orig cap $-$ demand

Original graph has feasible flow $\Leftrightarrow$ max flow in new graph saturates source and destination edges.

How do we get the maximum flow?

Take this flow (for edges in original graph), add demands back in, and augment more flow <u>while keeping flow $\geq$ demand</u>.

Problem 6 - Averaged Cut

The average of a cut is the average of the capacities of the cut edges.

Given a flow graph, find the cut that has minimum average.

Hint: binary search

Andrew Stankevitch Contest 7 "G", 2004-2005 Summer Petrozavodsk Camp
`http://codeforces.com/gym/100204`

Next Class

Minimum Cost Maximum Flow