Combinatorial Optimization Course

Min-Cost Flow Problem

Mohammad Hossein Bateni

February 13, 2005

Topics

- Problem Formulation
- Applications
- Primal-Dual Algorithms
- Hitchcock Problem
- Alphabeta Solution
- Min-Cost Circulation
- Equivalency

Min-Cost Flow Problem

- Given a graph G with edge capacities and edge cost, find the flow with size v with minimum cost.

Min-Cost Flow Problem

- Given a graph G with edge capacities and edge cost, find the flow with size v with minimum cost.

$$\begin{array}{rcl} \min & c^T f & \\ Af & = & -v_0 d \\ f & \leq & b \\ f & \geq & 0 \end{array}$$

where

$$d_i = \begin{cases} -1 & \text{if } i = s \\ +1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}$$

Applications

Caterer Problem

- She needs to supply r_i napkins for N succesive days.
- She can buy new napkins at p cents each.
- She can launder them at a fast laundry that takes m days and costs f cents a napkin.
- Or she can do that at a slow one which takes $n > m$ days and cost $s < f$ cents each.
- At the end of each day, she should determine how many to send to each laundry and how many new ones to buy so that she can satisfy the demand of the days
- Minimize the cost

Applications (Cont'd)

Capacitated Spanning Tree Problem

- We are give a complete undirected weighted graph G
- Each node has a traffic generation rate A_i
- And a central node s
- We want to force a maximum traffic B on edges
- Traffic is directed to central node
- Minimize the cost, where A_i is zero/one and $B = 1$

Combinatorializing the capacities

- Write it as **D**

Combinatorializing the capacities

- Write it as **D**

$$\begin{aligned} \max \quad & -c^T f \\ Af & \leq -v_0 d \\ f & \leq b \\ -f & \leq 0 \end{aligned}$$

Combinatorializing the capacities

- Write it as **D**

$$\begin{array}{rcl} \max & -c^T f & \\ Af & \leq & -v_0 d \\ f & \leq & b \\ -f & \leq & 0 \end{array}$$

- Writing the **DRP** we get

$$\begin{array}{rcll} \max & & -c^T f & \\ Af & = & 0 & \\ f & \leq & 0 & \text{for saturated edges} \\ f & \geq & 0 & \text{for empty edges} \\ f & \geq & -1 & \text{for all edges, beacuase } -c \leq 0 \end{array}$$

Algorithm CYCLE

- A feasible flow satisfying $Af = 0$ is called a **circulation**

Algorithm CYCLE

- A feasible flow satisfying $Af = 0$ is called a **circulation**
- We need to find a negative-cost circulation

Algorithm CYCLE

- A feasible flow satisfying $Af = 0$ is called a **circulation**
- We need to find a negative-cost circulation

THEOREM 1 *A flow is min-cost iff there is no negative-cost circulation.*

Algorithm

1. Use max-flow to find a flow of size v
2. **while** there is a negative-cost circulation **do**
   (a) Augment along it until this is no longer a circulation

- Finding the negative-weight cycle by Bellman-Ford in $\Theta(|V||E|)$

Algorithm CYCLE

- A feasible flow satisfying $Af = 0$ is called a **circulation**
- We need to find a negative-cost circulation

THEOREM 1 *A flow is min-cost iff there is no negative-cost circulation.*

Algorithm

1. Use max-flow to find a flow of size v
2. **while** there is a negative-cost circulation **do**
   (a) Augment along it until this is no longer a circulation

- Finding the negative-weight cycle by Bellman-Ford in $\Theta(|V||E|)$
- By Goldberg's algorithm in $O(\sqrt{|V|}|E| \log |C|)$

Algorithm Cycle

- A feasible flow satisfying $Af = 0$ is called a **circulation**
- We need to find a negative-cost circulation

Theorem 1 *A flow is min-cost iff there is no negative-cost circulation.*

Algorithm

1. Use max-flow to find a flow of size v
2. **while** there is a negative-cost circulation **do**
   (a) Augment along it until this is no longer a circulation

- Finding the negative-weight cycle by Bellman-Ford in $\Theta(|V||E|)$
- By Goldberg's algorithm in $O(\sqrt{|V|}|E| \log |C|)$
- Running time is $O(|V||E|^2|C||U|)$

Algorithm CYCLE

- A feasible flow satisfying $Af = 0$ is called a **circulation**
- We need to find a negative-cost circulation

THEOREM 1 *A flow is min-cost iff there is no negative-cost circulation.*

Algorithm

1. Use max-flow to find a flow of size v
2. **while** there is a negative-cost circulation **do**
   (a) Augment along it until this is no longer a circulation

- Finding the negative-weight cycle by Bellman-Ford in $\Theta(|V||E|)$
- By Goldberg's algorithm in $O(\sqrt{|V|}|E| \log |C|)$
- Running time is $O(|V||E|^2|C||U|)$
- If we always choose the minimum mean cycle it takes $O(|V|^2|E|^3 \log |V|)$

Combinatorializing the cost

- We shall write it as **P** and consider duals
- But it has lots of complexities
- Intuitively, we will increase the flow until reaching v
- We always choose the minimum weight augmenting path

  THEOREM 2 *Let f_1 be a flow of minimum cost with size v and let f_2 be a least cost augmentation. Then, $f_1 + f_2$ is the least cost flow with size $v + 1$.*

- Assume we don't have a negative weight at the beginning.

Algorithm

1. Initiazlize with $f = 0$
2. **while** f is less than v **do**
   (a) Augment along the least cost augmentation

Hitchcock Problem

- There are m sources which supply a_i units of something
- And n sinks which demand b_j units of that material
- Sending the output of source i to sink j has c_{ij} cost per unit
- We need to minimize the cost, satisfying all the demands
- We write it as **P** form

$$
\begin{aligned}
\min \ \textstyle\sum_{i,j} c_{ij} f_{ij} \\
\sum_{j=1}^{n} f_{ij} &= a_i \\
\sum_{i=1}^{m} f_{ij} &= b_j \\
f_{ij} &\geq 0
\end{aligned}
$$

- supposing that $\sum_i a_i = \sum_j b_j$

Hitchcock Problem (Cont'd)

- Dual is like this

$$\max \; w = \sum_{i=1}^{m} a_i \alpha_i + \sum_{j=1}^{n} b_j \beta_j$$

$$\begin{aligned} \alpha_i + \beta_j \quad &\leq \quad c_{ij} \\ \alpha_i, \beta_j \quad &\gtrless \quad 0 \end{aligned}$$

Hitchcock Problem (Cont'd)

- Dual is like this

$$\max \quad w = \sum_{i=1}^{m} a_i \alpha_i + \sum_{j=1}^{n} b_j \beta_j$$

$$\begin{aligned} \alpha_i + \beta_j &\leq c_{ij} \\ \alpha_i, \beta_j &\gtreqless 0 \end{aligned}$$

- An initial feasible solution is

$$\begin{aligned} \alpha_i &= 0 \\ \beta_j &= \min_{1 \leq i \leq m} \{c_{ij}\} \end{aligned}$$

Hitchcock Problem (Cont'd)

- Dual is like this

$$\max \ w = \sum_{i=1}^{m} a_i \alpha_i + \sum_{j=1}^{n} b_j \beta_j$$

$$\begin{aligned} \alpha_i + \beta_j &\leq c_{ij} \\ \alpha_i, \beta_j &\gtrless 0 \end{aligned}$$

- An initial feasible solution is

$$\begin{aligned} \alpha_i &= 0 \\ \beta_j &= \min_{1 \leq i \leq m} \{c_{ij}\} \end{aligned}$$

- Let $IJ = \{(i, j) : \alpha_i + \beta_j = c_{ij}\}$

Hitchcock Problem (Cont'd)

- **RP** is as follows

$$
\begin{aligned}
\min \ \zeta = \sum_{i=1}^{m+n} x_i^a & & \\
\sum_j f_{ij} + x_i^a &= a_i & \\
\sum_i f_{ij} + x_{m+j}^a &= b_j & \\
x_i^a &\geq 0 & \\
f_{ij} &\geq 0 & (i,j) \in IJ \\
f_{ij} &= 0 & (i,j) \notin IJ
\end{aligned}
$$

Hitchcock Problem (Cont'd)

- **RP** is as follows

$$\min \ \zeta = \sum_{i=1}^{m+n} x_i^a$$

$$\begin{aligned}
\sum_j f_{ij} + x_i^a &= a_i \\
\sum_i f_{ij} + x_{m+j}^a &= b_j \\
x_i^a &\geq 0 \\
f_{ij} &\geq 0 \qquad (i,j) \in IJ \\
f_{ij} &= 0 \qquad (i,j) \notin IJ
\end{aligned}$$

$$\zeta = \sum_i a_i + \sum_j b_j - 2 \sum_{(i,j)\in IJ} f_{ij}$$

Hitchcock Problem (Cont'd)

- **RP** is as follows

$$\begin{aligned} \min \ \zeta = \sum_{i=1}^{m+n} x_i^a \\ \sum_j f_{ij} + x_i^a &= a_i \\ \sum_i f_{ij} + x_{m+j}^a &= b_j \\ x_i^a &\geq 0 \\ f_{ij} &\geq 0 \qquad (i,j) \in IJ \\ f_{ij} &= 0 \qquad (i,j) \notin IJ \end{aligned}$$

$$\zeta = \sum_i a_i + \sum_j b_j - 2 \sum_{(i,j) \in IJ} f_{ij}$$

- So, we are maximizing the flow through admissible edges

Alphabeta

- We will reach an optimal flow for **RP** called **nonbreakthrough** situation

Alphabeta

- We will reach an optimal flow for **RP** called **nonbreakthrough** situation

$$\begin{aligned} I^* &= \{i : \text{source } i \text{ is labeled}\} \\ J^* &= \{j : \text{sink } j \text{ is labeled}\} \end{aligned}$$

$$\theta = \min_{\substack{i \in I^* \\ j \notin J^*}} \left[\frac{c_{ij} - \alpha_i - \beta_j}{2} \right]$$

Algorithm

1. choose α, β feasible in **D**
2. **while** flow is not maximum **do**
   (a) Solve the maximum flow in **RP** using only admissible edges
   (b) Find I^* and J^* and nonbreakthrough
   (c) Calculate θ and update α and β

Equivalency

- Hitchcock and Min-Cost Flow are equivalent

Equivalency

- Hitchcock and Min-Cost Flow are equivalent
  - Hitchcock can be solved using MCF

Equivalency

- Hitchcock and Min-Cost Flow are equivalent
  - Hitchcock can be solved using MCF
  - MCF can be solved using Hitchcock

Equivalency

- Hitchcock and Min-Cost Flow are equivalent
  - Hitchcock can be solved using MCF
  - MCF can be solved using Hitchcock
- Min-Cost Flow and Min-Cost Circulation are equivalent

Equivalency

- Hitchcock and Min-Cost Flow are equivalent
  - Hitchcock can be solved using MCF
  - MCF can be solved using Hitchcock
- Min-Cost Flow and Min-Cost Circulation are equivalent
  - MCC can be solved using MCF

Equivalency

- Hitchcock and Min-Cost Flow are equivalent
  - Hitchcock can be solved using MCF
  - MCF can be solved using Hitchcock
- Min-Cost Flow and Min-Cost Circulation are equivalent
  - MCC can be solved using MCF
  - MCF can be solved using MCC

The End