Well solved problems

Lorenzo Castelli, Università degli Studi di Trieste.

UNIVERSITÀ
DEGLI STUDI DI TRIESTE

Setting the context

A natural starting point in solving linear integer programs

$$(\boldsymbol{IP}) \quad \max\{\boldsymbol{cx} : \boldsymbol{Ax} \leq \boldsymbol{b}, \boldsymbol{x} \in \mathbb{Z}_+^{\boldsymbol{n}}\}$$

with integral data $(\boldsymbol{A}, \boldsymbol{b})$ is to ask when one will be so lucky that the linear programming relaxation

$$(\boldsymbol{LP}) \quad \max\{\boldsymbol{cx} : \boldsymbol{Ax} \leq \boldsymbol{b}, \boldsymbol{x} \in \mathbb{R}_+^{\boldsymbol{n}}\}$$

will have an optimal solution that is integral.

Total unimodularity

Definition

A matrix $\boldsymbol{A}$ is totally unimodular (TU) if every square submatrix of $\boldsymbol{A}$ has determinant $+\mathbf{1}$, $-\mathbf{1}$ or $\mathbf{0}$

Observation

If $\boldsymbol{A}$ is TU, $\boldsymbol{a_{ij}} \in \{+\mathbf{1}, -\mathbf{1}, \mathbf{0}\}$

Examples

Matrices not TU

$$A_1 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$$

$$|A_1| = 2$$

$$A_2 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

$$|A_2| = 2$$

TU matrix

$$A_3 = \begin{pmatrix} 1 & -1 & -1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$|A_3| = 0$$

Propositions

Proposition 1

A matrix $\boldsymbol{A}$ is TU if and only if

- the transpose matrix $\boldsymbol{A}^{\boldsymbol{T}}$ is TU
- the matrix $[\boldsymbol{A}|\boldsymbol{I}]$ is TU

Propositions

Proposition 2 (Sufficient condition)

A matrix $\boldsymbol{A}$ is TU if

(i) $a_{ij} \in \{+1, -1, 0\}$

(ii) Each column contains at most two nonzero coefficients, i.e, $\sum_{i=1}^{m} |a_{ij}| \leq 2$

(iii) There is a partition (M_1, M_2) of the M of rows such that each column j containing two nonzero coefficients satisfies $\sum_{i \in M_1} a_{ij} - \sum_{i \in M_2} a_{ij} = 0$.

Condition (***iii***) means that if the nonzeros are in row i and k, and if $a_{ik} = -a_{kj}$, then $\{i, k\} \in M_1$ or $\{i, k\} \in M_2$, whereas if $a_{ik} = a_{kj}, i \in M_1$ and $k \in M_2$ or vice versa.

Example - TU matrix

$$A = \begin{array}{cl} \begin{bmatrix} -1 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 \end{bmatrix} & \begin{array}{l} \textit{row } 1 \\ \textit{row } 2 \\ \textit{row } 3 \\ \textit{row } 4 \\ \textit{row } 5 \end{array} \end{array}$$

1. rows 1 and 3 are not in the same class
2. rows 2 and 3 are not in the same class
3. rows 1 and 4 are not in the same class
4. rows 2 and 5 are not in the same class
5. rows 1 and 2 are in the same class
6. rows 4 and 5 are in the same class
7. rows 2 and 4 are not in the same class

Hence $M_1 = \{1, 2\}$ and $M_2 = \{3, 4, 5\}$

Examples - TU matrices

Condition is sufficient

$$\begin{pmatrix} \mathbf{1} & -\mathbf{1} & -\mathbf{1} & \mathbf{0} \\ -\mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & -\mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{pmatrix}$$

1. rows 1 and 2 are in the same class
2. rows 1 and 3 are in the same class
3. rows 1 and 4 are in the same class
4. rows 2 and 3 are in the same class

$M_1 = \{1, 2, 3, 4\}$, $M_2 = \emptyset$

Condition is not necessary

$$\begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ -\mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{1} & -\mathbf{1} & \mathbf{0} \\ \mathbf{0} & -\mathbf{1} & -\mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{pmatrix}$$

Propositions

Proposition 3

The linear programming problem $\max\{\boldsymbol{cx} : \boldsymbol{Ax} \leq \boldsymbol{b}, \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}_{+}\}$ has an integer optimal solution for all integer vectors $\boldsymbol{b}$ for which it has a finite optimal value if and only if $\boldsymbol{A}$ is totally unimodular.

Minimum Cost Network Flow

Given a digraph $D = (V, A)$ with arc capacities h_{ij} for all $(i, j) \in A$, demands b_i (positive inflows or negative outflows) at each node $i \in V$, and unit flow costs c_{ij} for all $(i, j) \in A$, the minimum cost network flow problem is to find a feasible flow that satisfies all the demands at minimum cost. This has the formulation

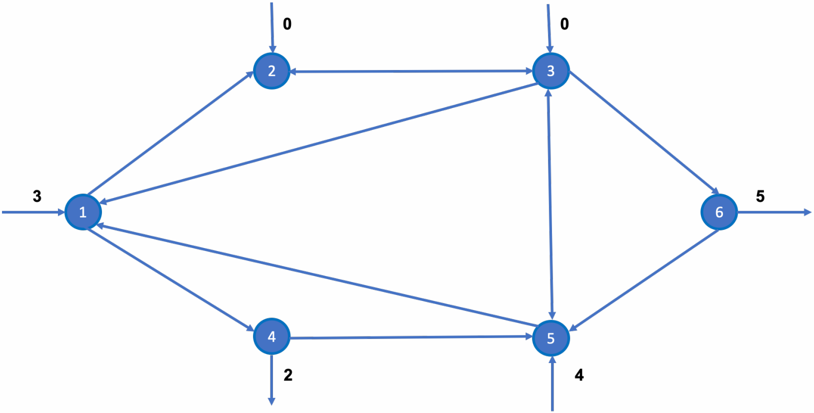
$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \tag{1}$$

$$\sum_{k \in V^+(i)} x_{ik} - \sum_{k \in V^-(i)} x_{ki} = b_i \text{ for } i \in V \tag{2}$$

$$0 \leq x_{ij} \leq h_{ij} \text{ for } (i, j) \in A \tag{3}$$

where x_{ij} denotes the flow in arc (i, j), $V^+(i) = \{k : (i, k) \in A\}$ and $V^-(i) = \{k : (k, i) \in A\}$

Minimum Cost Network Flow



Minimum Cost Network Flow

x_{12}	x_{14}	x_{23}	x_{31}	x_{32}	x_{35}	x_{36}	x_{45}	x_{51}	x_{53}	x_{65}		
1	1	0	-1	0	0	0	0	-1	0	0	=	3
-1	0	1	0	-1	0	0	0	0	0	0	=	0
0	0	-1	1	1	1	1	0	0	-1	0	=	0
0	-1	0	0	0	0	0	-1	0	0	0	=	-2
0	0	0	0	0	-1	0	-1	1	1	-1	=	4
0	0	0	0	0	0	-1	0	0	0	1	=	-5

The additional constraints are the capacity constraints $\mathbf{0} \leq \boldsymbol{x_{ij}} \leq \boldsymbol{h_{ij}}$

Propositions

Proposition 4

The constraint matrix $\boldsymbol{A}$ arising in a minimum cost network flow problem is totally unimodular.

Proof The matrix $\boldsymbol{A}$ is of the form $\begin{pmatrix} \boldsymbol{C} \\ \boldsymbol{I} \end{pmatrix}$ where $\boldsymbol{C}$ comes from the flow conservation constraints and $\boldsymbol{I}$ from the capacity constraints. Therefore it suffices to show that $\boldsymbol{C}$ is TU. The sufficient conditions of Proposition 2 are satisfied with $\boldsymbol{M_1} = \boldsymbol{M}$ and $\boldsymbol{M_2} = \emptyset$.

Key result

Corollary

In a minimum cost network flow problem, if the demands $\{b_i\}$ and the capacities $\{h_{ij}\}$ are integral

- Each extreme point is integral.
- The constraints (2) and (3) describe the convex hull of the integral feasible flows.

This corollary means that the linear relaxation of the minimum cost network flow problem always provides an integer solution provided that all capacities $\{h_{ij}\}$ and demands $\{b_i\}$ are integral.

Special minimum cost flows

The Shortest Path Problem Given a digraph $D = (V, A)$, two distinguished nodes $s, t \in V$, and non-negative arcs costs c_{ij} for $(i, j) \in A$, find a minimum cost $s - t$ path.

The Max Flow Problem Given a digraph $D = (V, A)$, two distinguished nodes $s, t \in V$, and non-negative capacities h_{ij} for $(i, j) \in A$, find a maximum flow from s to t path.

The Transportation Problem Let there be m suppliers and n consumers. The ith supplier can provide a_i units of a certain good and the jth consumer has a demand for b_j units. If c_{ij} is the cost to transport one unit of good from the ith supplier to the jth consumer, the problem is to transport the goods from the suppliers to the consumers at minimum cost.

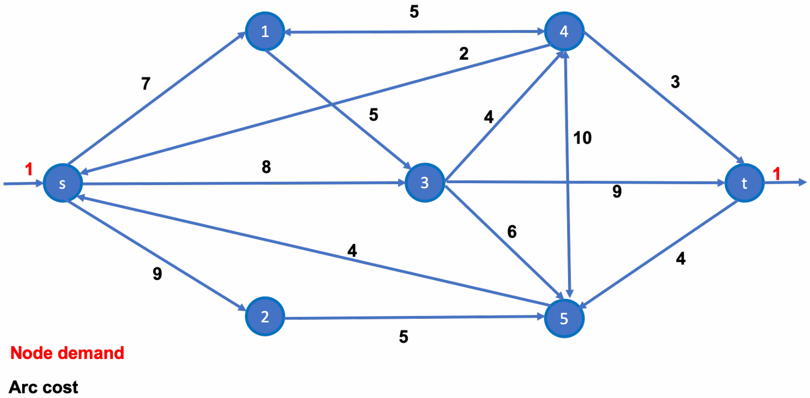
The Assignment Problem It is a special case of the transportation problem, where the number of suppliers is equal to the number of consumers, each supplier has unit supply, and each consumer has unit demand.

The Shortest Path Problem

If we set $b_s = 1$ and $b_t = -1$, only one unit of flow can move from s to t, and the problem is to find the sequence of arcs at minimum cost that this unit will traverse. An arc $(i, j) \in A$ if and only if $h_{ij} > 0$. Since we assume only integral values, $(i, j) \in A$ if and only if $h_{ij} \geq 1$. Since exactly one unit flows in the network, there is no need to explicitly include the capacity constraints.

Decision variables are such that $x_{ij} = 1$ if arc (i, j) is in the minimum cost (shortest) $s - t$ path. For the total unimodularity, an optimal solution is always integer. Therefore, we can write $x_{ij} \geq 0$.

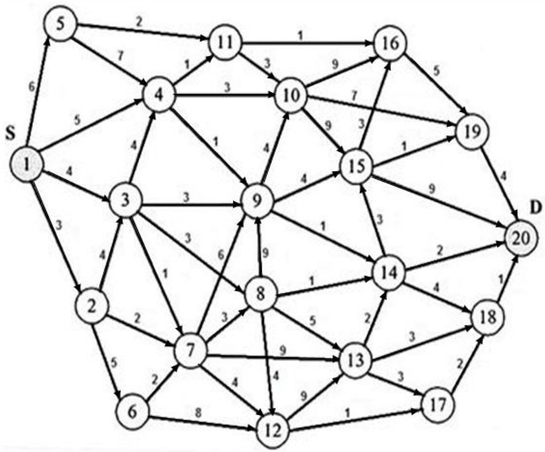
The Shortest Path Problem



The Shortest Path Problem

$$
\begin{aligned}
z = \min \sum_{(i,j)\in A} c_{ij} x_{ij} & & \\
\sum_{k\in V^+(i)} x_{ik} - \sum_{k\in V^-(i)} x_{ki} &= 1 & \text{for } i = s \\
\sum_{k\in V^+(i)} x_{ik} - \sum_{k\in V^-(i)} x_{ki} &= 0 & \text{for } i \in V \backslash \{s, t\} \\
\sum_{k\in V^+(i)} x_{ik} - \sum_{k\in V^-(i)} x_{ki} &= -1 & \text{for } i = t \\
x_{ij} \geq 0 \text{ for } (i, j) \in A & &
\end{aligned}
$$

The Shortest Path Problem



The Maximum Flow Problem

Adding a backward arc from t to s, the maximum $s - t$ flow problem can be formulated as

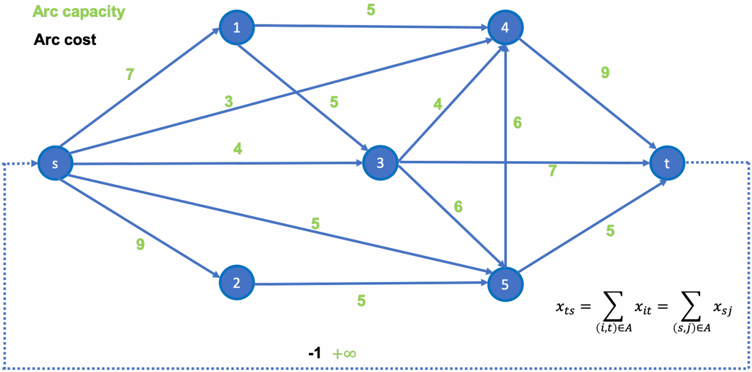
$$z = \max x_{ts}$$

$$\sum_{k \in V^{+}(i)} x_{ik} - \sum_{k \in V^{-}(i)} x_{ki} = 0 \qquad \text{for } i \in V$$

$$0 \leq x_{ij} \leq h_{ij} \qquad \text{for } (i,j) \in A$$

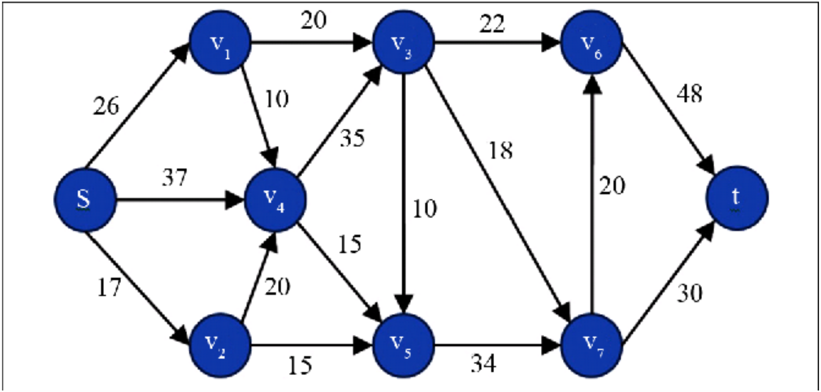
For the total unimodularity, an optimal solution is integer provided that all capacities h_{ij} are integral.

The Maximum Flow Problem



$$x_{ts} = \sum_{(i,t)\in A} x_{it} = \sum_{(s,j)\in A} x_{sj}$$

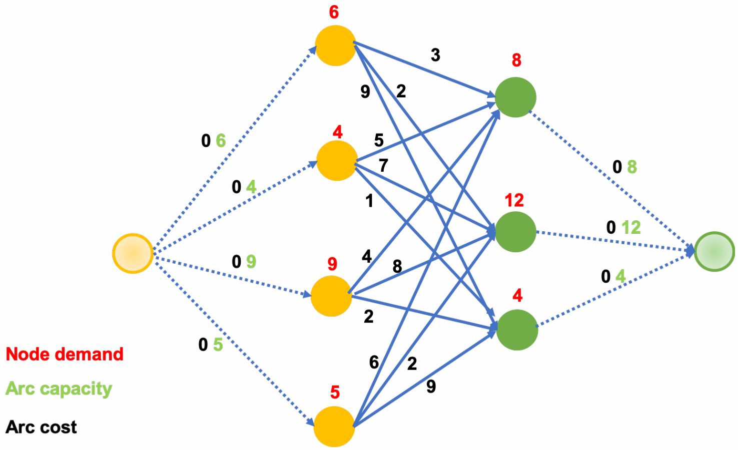
The Maximum Flow Problem



The Transportation Problem

It can be formulated as a minimum cost flow problem on a bipartite graph $D = (V_1 \cup V_2, A)$ where $V_1 = \{1, \ldots, m\}$ is the set of sources, $V_2 = \{1, \ldots, n\}$ is the set of sinks, and $A = \{(i, j) : i \in V_1, j \in V_2\}$. Without loss of generality, we assume there is an arc from each supply node to each demand node. The unit shipping cost from $i \in V_1$ to $j \in V_2$ is c_{ij}. If there is no arc from i to j, we take c_{ij} very large. Node $i \in V_1$ has a positive integral supply a_i and $j \in V_2$ has a positive integral demand b_j. The flow out of a source is required to equal its supply, and the flow into a sink must equal its demand. Thus a necessary condition for feasibility is $\sum_{i \in V_1} a_i = \sum_{j \in V_2} b_j$

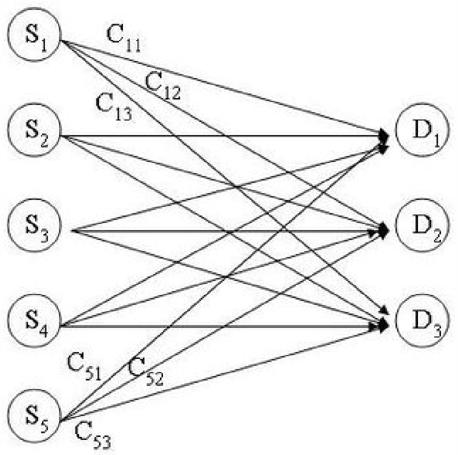
The Transportation Problem



The Transportation Problem

$$
\begin{aligned}
z = \min \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} x_{ij} & \\
\sum_{j \in V_2} x_{ij} = a_i & \qquad \text{for } i \in V_1 \\
\sum_{i \in V_1} x_{ij} = b_j & \qquad \text{for } j \in V_2 \\
x_{ij} \geq 0 & \qquad \text{for } (i, j) \in A
\end{aligned}
$$

The Transportation Problem

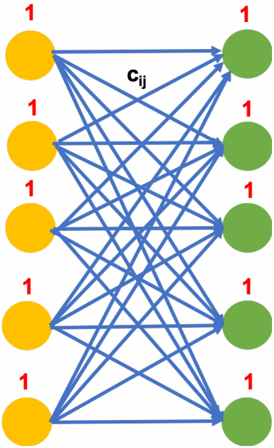


The Assignment Problem

When $a_i = b_j = 1$ for all i and j and $m = n$, we have the assignment problem

$$
\begin{aligned}
z = \min \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} x_{ij} & & \\
\sum_{j \in V_2} x_{ij} = 1 & \qquad & \text{for } i \in V_1 \\
\sum_{i \in V_1} x_{ij} = 1 & \qquad & \text{for } j \in V_2 \\
0 \leq x_{ij} \leq 1 & \qquad & \text{for } (i, j) \in A
\end{aligned}
$$

The Assignment Problem



The Assignment Problem