DM545/DM871

Linear and Integer Programming

## Lecture 12

# Network Flows

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

# Outline

1. Well Solved Problems

2. (Minimum Cost) Network Flows

3. Application Example

# Outline

1. Well Solved Problems

2. (Minimum Cost) Network Flows

3. Application Example

# Separation problem

$\max\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in X\} \equiv \max\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in \text{conv}(X)\}$
$X \subseteq \mathbb{Z}^n$, $P$ a polyhedron $P \subseteq \mathbb{R}^n$ and $X = P \cap \mathbb{Z}^n$

**Definition (Separation problem for a COP)**

Given $\boldsymbol{x}^* \in P$; is $\boldsymbol{x}^* \in \text{conv}(X)$? If not find an inequality $\boldsymbol{ax} \leq \boldsymbol{b}$ satisfied by all points in $X$ but violated by the point $\boldsymbol{x}^*$.

(Farkas' lemma states the existence of such an inequality.)

# Properties of Easy Problems

Four properties that often go together:

**Definition**

(i) Efficient optimization property: $\exists$ a polynomial algorithm for $\max\{\boldsymbol{cx} : \boldsymbol{x} \in X \subseteq \mathbb{R}^n\}$

(ii) Strong duality property: $\exists$ strong dual D $\min\{w(\boldsymbol{u}) : \boldsymbol{u} \in U\}$ that allows to quickly verify optimality

(iii) Efficient separation problem: $\exists$ efficient algorithm for separation problem

(iv) Efficient convex hull property: a compact description of the convex hull is available

Example:

If explicit convex hull  strong duality holds
efficient separation property (just description of $\text{conv}(X)$)

Theoretical analysis to prove results about

- strength of certain inequalities that are facet defining
  2 ways
- descriptions of convex hull of some discrete $X \subseteq \mathbb{Z}^*$
  several ways, we see one next

Example

Let

$$X = \{(x, y) \in \mathbb{R}^m_+ \times \mathbb{B}^1 : \sum_{i=1}^{m} x_i \leq my, x_i \leq 1 \text{ for } i = 1, \ldots, m\}$$

$$P = \{(x, y) \in \mathbb{R}^m_+ \times \mathbb{R}^1 : x_i \leq y \text{ for } i = 1, \ldots, m, y \leq 1\}$$

.

Polyhedron $P$ describes $\text{conv}(X)$

# Totally Unimodular Matrices

When the LP solution to this problem

$$IP : \max\{\boldsymbol{c}^T\boldsymbol{x} : A\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \in \mathbb{Z}^n_+\}$$

with all data integer will have integer solution?

$$\left[\begin{array}{c:c:c:c} A_N & A_B & 0 & \boldsymbol{b} \\ \hdashline \boldsymbol{c}_N^T & \boldsymbol{c}_B^T & 1 & 0 \end{array}\right]$$

$$A_B x_B + A_N x_N = b$$

$\boldsymbol{x}_N = 0 \rightsquigarrow A_B\boldsymbol{x}_B = \boldsymbol{b}$,
$A_B$ $m \times m$ non singular matrix
$\boldsymbol{x}_B \geq 0$

Cramer's rule for solving systems of linear equations:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} e \\ f \end{bmatrix} \qquad x = \frac{\begin{vmatrix} e & b \\ f & d \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}} \qquad y = \frac{\begin{vmatrix} a & e \\ c & f \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}} \qquad \boldsymbol{x} = A_B^{-1}\boldsymbol{b} = \frac{A_B^{adj}\boldsymbol{b}}{\det(A_B)}$$

## Definition

- A square integer matrix $B$ is called unimodular (UM) if $\det(B) = \pm 1$
- An integer matrix $A$ is called totally unimodular (TUM) if every square, nonsingular submatrix of $A$ is UM

## Proposition

- *If $A$ is TUM then all vertices of $R_1(A) = \{\mathbf{x} : A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0\}$ are integer if $\mathbf{b}$ is integer*
- *If $A$ is TUM then all vertices of $R_2(A) = \{\mathbf{x} : A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq 0\}$ are integer if $\mathbf{b}$ is integer.*

Proof: if $A$ is TUM then $\left[A \,\vdots\, I\right]$ is TUM
Any square, nonsingular submatrix $C$ of $\left[A \,\vdots\, I\right]$ can be written as

$$C = \left[\begin{array}{c:c} B & 0 \\ \hdashline D & I_k \end{array}\right]$$

where $B$ is square submatrix of $A$. Hence $\det(C) = \det(B) = \pm 1$

## Proposition

*The transpose matrix $A^T$ of a TUM matrix $A$ is also TUM.*

## Theorem (Sufficient condition)

*An integer matrix $A$ is TUM if*

1. $a_{ij} \in \{0, -1, +1\}$ *for all* $i, j$
2. *each column contains at most two non-zero coefficients* ($\sum_{i=1}^{m} |a_{ij}| \leq 2$)
3. *if the rows can be partitioned into two sets* $I_1$, $I_2$ *such that:*
   - *if a column has 2 entries of same sign, their rows are in different sets*
   - *if a column has 2 entries of different signs, their rows are in the same set*

$$\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & -1 & -1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Proof: by induction

Basis: one matrix of one element $\{0, +1, -1\}$ is TUM

Induction: let $C$ be of size $k$.
If $C$ has column with all 0s then it is singular.
If a column with only one 1 then expand on that by induction
If 2 non-zero in each column then

$$\forall j : \sum_{i \in I_1} a_{ij} = \sum_{i \in I_2} a_{ij}$$

but then a linear combination of rows is zero and $\det(C) = 0$

Other matrices with integrality property:

- TUM
- Balanced matrices
- Perfect matrices
- Integer vertices

Defined in terms of forbidden substructures that represent fractionating possibilities.

**Proposition**

*A is always TUM if it comes from*

- *node-edge incidence matrix of undirected bipartite graphs (ie, no odd cycles)* ($I_1 = U, I_2 = V, B = (U, V, E)$)
- *node-arc incidence matrix of directed graphs* ($I_2 = \emptyset$)

Eg: Shortest path, max flow, min cost flow, bipartite weighted matching

# Summary

1. Well Solved Problems

2. (Minimum Cost) Network Flows

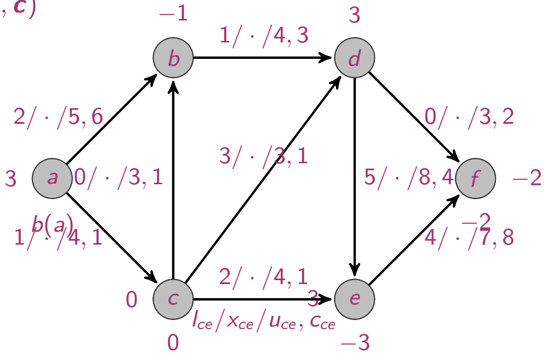
3. Application Example

# Outline

# Terminology

Network:
- directed graph $D = (V, A)$
- arc, directed link, from tail to head
- lower bound $l_{ij} > 0, \forall ij \in A$, capacity $u_{ij} \geq l_{ij}, \forall ij \in A$
- cost $c_{ij}$, linear variation (if $ij \notin A$ then $l_{ij} = u_{ij} = 0, c_{ij} = 0$)
- balance vector $b(i)$, $b(i) > 0$ supply node (source), $b(i) < 0$ demand node (sink, tank), $b(i) = 0$ transhipment node (assumption $\sum_i b(i) = 0$)

$N = (V, A, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{b}, \boldsymbol{c})$



# Network Flows

Flow $\boldsymbol{x} : A \to \mathbb{R}$

balance vector of $\boldsymbol{x}$: $b_{\boldsymbol{x}}(v) = \sum_{vu \in A} x_{vu} - \sum_{wv \in A} x_{wv}, \forall v \in V$

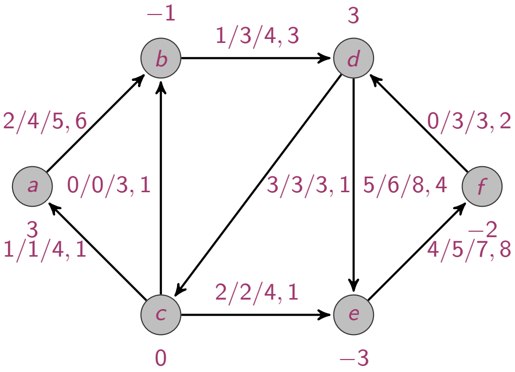
$$b_{\boldsymbol{x}}(v) \begin{cases} > 0 & \text{source} \\ < 0 & \text{sink/target/tank} \\ = 0 & \text{balanced} \end{cases}$$

(generalizes the concept of path with $b_{\boldsymbol{x}}(v) = \{0, 1, -1\}$)

feasible $\quad l_{ij} \leq x_{ij} \leq u_{ij}$, $b_{\boldsymbol{x}}(i) = b(i)$
cost $\quad \boldsymbol{c}^T\boldsymbol{x} = \sum_{ij \in A} c_{ij} x_{ij}$ (varies linearly with $\boldsymbol{x}$)

If $iji$ is a 2-cycle and all $l_{ij} = 0$, then at least one of $x_{ij}$ and $x_{ji}$ is zero.

# Example




Feasible flow of cost 109

# Minimum Cost Network Flows

Find cheapest flow through a network in order to satisfy demands at certain nodes from available supplier nodes.

**Variables:**

$$x_{ij} \in \mathbb{R}_0^+$$

**Objective:**

$$\min \sum_{ij \in A} c_{ij} x_{ij}$$

**Constraints:** mass balance + flow bounds

$$\sum_{j: ij \in A} x_{ij} - \sum_{j: ji \in A} x_{ji} = b(i) \quad \forall i \in V$$

$$l_{ij} \leq x_{ij} \leq u_{ij}$$

$$\begin{aligned} \min\ & \boldsymbol{c}^T \boldsymbol{x} \\ & N\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{l} \leq \boldsymbol{x} \leq \boldsymbol{u} \end{aligned}$$

N node arc incidence matrix

if flow of indivisible goods: under the assumption that all parameter values are integer (we can multiply if rational) the LP relaxation solution is integer.

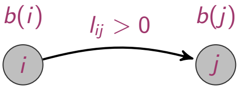
$$
\begin{array}{r|ccccccccc}
 & x_{e_1} & x_{e_2} & \dots & x_{ij} & \dots & x_{e_m} & & \\
 & c_{e_1} & c_{e_2} & \dots & c_{ij} & \dots & c_{e_m} & & \\
\hline
1 & 1 & . & \dots & . & \dots & . & = & b_1 \\
2 & . & . & \dots & . & \dots & . & = & b_2 \\
\vdots & \vdots & \ddots & & & & & = & \vdots \\
i & -1 & . & \dots & 1 & \dots & . & = & b_i \\
\vdots & \vdots & \ddots & & & & & = & \vdots \\
j & . & . & \dots & -1 & \dots & . & = & b_j \\
\vdots & \vdots & \ddots & & & & & = & \vdots \\
n & . & . & \dots & . & \dots & . & = & b_n \\
\hline
e_1 & 1 & & & & & & \leq & u_1 \\
e_2 & & 1 & & & & & \leq & u_2 \\
\vdots & \vdots & \ddots & & & & & \leq & \vdots \\
(i,j) & & & & 1 & & & \leq & u_{ij} \\
\vdots & \vdots & \ddots & & & & & \leq & \vdots \\
e_m & & & & & & 1 & \leq & u_m
\end{array}
$$

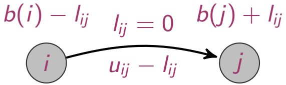
# Reductions/Transformations

## Lower bounds

Let $N = (V, A, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{b}, \boldsymbol{c})$

$$
\begin{aligned}
N' &= (V, A, \boldsymbol{l}', \boldsymbol{u}', \boldsymbol{b}', \boldsymbol{c}) \\
b'(i) &= b(i) - l_{ij} \\
b'(j) &= b(j) + l_{ij} \\
u'_{ij} &= u_{ij} - l_{ij} \\
l'_{ij} &= 0
\end{aligned}
$$




$$\boldsymbol{c}^T \boldsymbol{x}$$

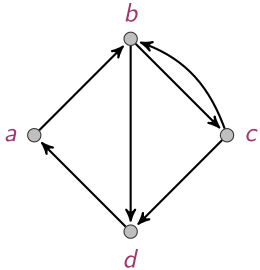
$$\boldsymbol{c}^T \boldsymbol{x}' + \sum_{ij \in A} c_{ij} l_{ij}$$

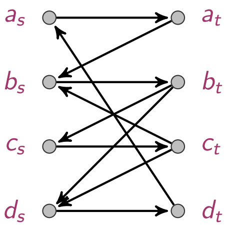
## Undirected arcs

## Vertex splitting

If there are bounds and costs of flow passing through vertices where $b(v) = 0$ (used to ensure that a node is visited):

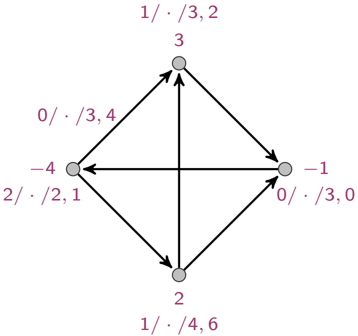
$N = (V, A, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{c}, \boldsymbol{l}^*, \boldsymbol{u}^*, \boldsymbol{c}^*)$

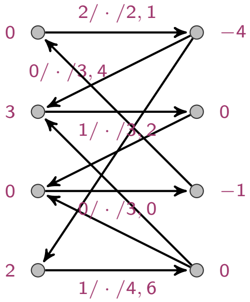



From $D$ to $D_{ST}$ as follows:

$$\begin{aligned}
\forall v \in V \quad &\rightsquigarrow v_s, v_t \in V(D_{ST}) \text{ and } v_s v_t \in A(D_{ST}) \\
\forall xy \in A(D) &\rightsquigarrow x_t y_s \in A(D_{ST})
\end{aligned}$$



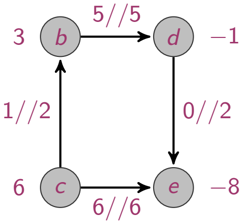



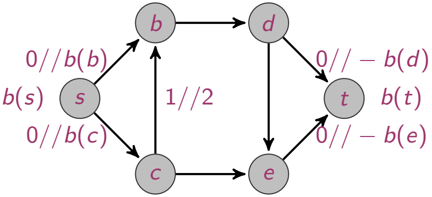

$$\forall v \in V \text{ and } v_s v_t \in A_{ST} \rightsquigarrow h'(v_s v_t) = h^*(v), \quad h^* \in \{l^*, u^*, c^*\}$$
$$\forall xy \in A \text{ and } x_t y_s \in A_{ST} \rightsquigarrow h'(x_t y_s) = h(x, y), \; h \in \{l, u, c\}$$

If $b(v) = 0$, then $b'(v_s) = b'(v_t) = 0$
If $b(v) < 0$, then $b'(v_s) = 0$ and $b'(v_t) = b(v)$
If $b(v) > 0$, then $b'(v_s) = b(v)$ and $b'(v_t) = 0$

**$(s,t)$-flow:**

$$b_{\mathbf{x}}(v) = \begin{cases} k & \textit{if } v = s \\ -k & \textit{if } v = t \\ 0 & \textit{otherwise} \end{cases}, \qquad |\mathbf{x}| = |b_{\mathbf{x}}(s)|$$




$$b(s) = \sum_{v:b(v)>0} b(v) = M$$
$$b(t) = \sum_{v:b(v)<0} b(v) = -M$$

$\exists$ feasible flow in $N$ $\iff$ $\exists$ $(s,t)$-flow in $N_{st}$ with $|x| = M$ $\iff$ max flow in $N_{st}$ is $M$

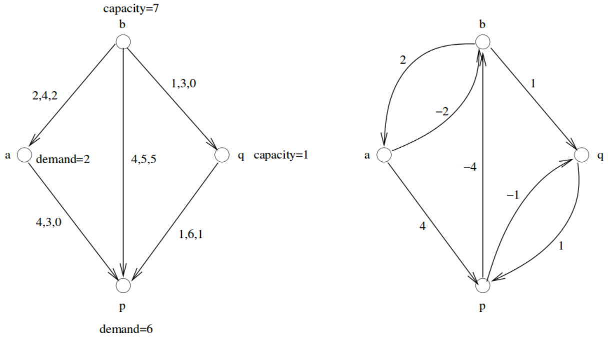
# Residual Network

**Residual Network $N(x)$:** given that a flow $x$ already exists, how much flow excess can be moved in $G$?

Replace arc $ij \in N$ with arcs:

| | residual capacity | cost |
|---|---|---|
| $ij$ : | $r_{ij} = u_{ij} - x_{ij}$ | $c_{ij}$ |
| $ji$ : | $r_{ji} = x_{ij}$ | $-c_{ij}$ |

$(N, \boldsymbol{c}, \boldsymbol{u}, \boldsymbol{x})$ $\qquad$ $(N(\boldsymbol{x}), \boldsymbol{r}, \boldsymbol{c}')$



# Special cases

**Shortest path problem** path of minimum cost from $s$ to $t$ with costs $\lesseqgtr 0$

$b(s) = 1, b(t) = -1, b(i) = 0$

if to any other node? $b(s) = (n-1), b(i) = 1, u_{ij} = n-1$

**Max flow problem** incur no cost but restricted by bounds

steady state flow from $s$ to $t$

$b(i) = 0 \ \forall i \in V, \qquad c_{ij} = 0 \ \forall ij \in A \qquad ts \in A$

$c_{ts} = -1, \qquad u_{ts} = \infty$

**Assignment problem** min weighted bipartite matching,

$|V_1| = |V_2|, A \subseteq V_1 \times V_2$

$c_{ij}$

$b(i) = 1 \ \forall i \in V_1 \qquad b(i) = -1 \ \forall i \in V_2 \qquad u_{ij} = 1 \ \forall ij \in A$

# Special cases

Transportation problem/Transhipment distribution of goods, warehouses-costumers

$|V_1| \neq |V_2|, \qquad u_{ij} = \infty$ for all $ij \in A$

$$\begin{aligned}
\min \sum c_{ij} x_{ij} & \\
\sum_i x_{ij} \geq b_j & \qquad \forall j \in V_2 \\
\sum_j x_{ij} \leq a_i & \qquad \forall i \in V_1 \\
x_{ij} \geq 0 &
\end{aligned}$$

if $\sum a_i = \sum b_i$ then $\geq$ / $\leq$ become $=$
if $\sum a_i > \sum b_i$ then add dummy tank nodes
if $\sum a_i < \sum b_i$ then infeasible

Multi-commodity flow problem ship several commodities using the same network, different origin destination pairs separate mass balance constraints, share capacity constraints, min overall flow

$$
\begin{aligned}
\min \textstyle\sum_k \boldsymbol{c}^k \boldsymbol{x}^k & \\
N\boldsymbol{x}^k \geq \boldsymbol{b}^k & \qquad \forall k \\
\textstyle\sum_k \boldsymbol{x}_{ij}^k \leq \boldsymbol{u}_{ij} & \qquad \forall ij \in A \\
0 \leq \boldsymbol{x}_{ij}^k \leq \boldsymbol{u}_{ij}^k &
\end{aligned}
$$

What is the structure of the matrix now? Is the matrix still TUM?

# Outline

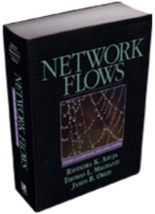
1. Well Solved Problems

2. (Minimum Cost) Network Flows

3. Application Example

# Ship loading problem

Plenty of applications. See Ahuja Magnanti Orlin, Network Flows, 1993

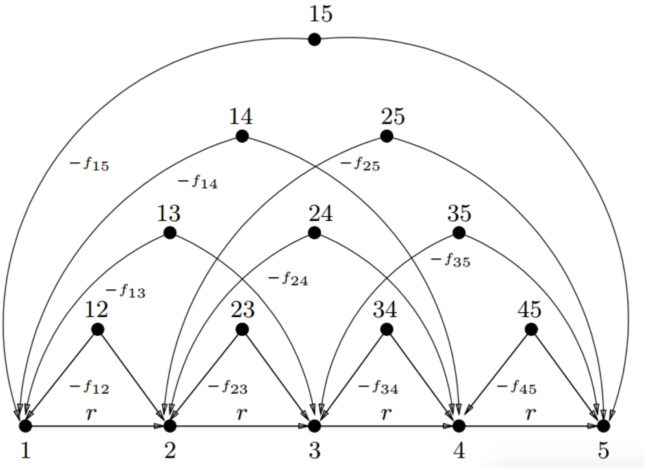



- A cargo company (eg, Maersk) uses a ship with a capacity to carry at most $r$ units of cargo.
- The ship sails on a long route (say from Southampton to Alexandria) with several stops at ports in between.
- At these ports cargo may be unloaded and new cargo loaded.
- At each port there is an amount $b_{ij}$ of cargo which is waiting to be shipped from port $i$ to port $j > i$
- Let $f_{ij}$ denote the income for the company from transporting one unit of cargo from port $i$ to port $j$.
- The goal is to plan how much cargo to load at each port so as to maximize the total income while never exceeding ship's capacity.

# Application Example: Modeling

- $n$ number of stops including the starting port and the terminal port.
- $N = (V, A, \boldsymbol{l} \equiv 0, \boldsymbol{u}, \boldsymbol{c})$ be the network defined as follows:
  - $V = \{v_1, v_2, ..., v_n\} \cup \{v_{ij} : 1 \leq i < j \leq n\}$
  - $A = \{v_1 v_2, v_2 v_3, ... v_{n-1} v_n\} \cup \{v_{ij} v_i, v_{ij} v_j : 1 \leq i < j \leq n\}$
  - capacity: $u_{v_i v_{i+1}} = r$ for $i = 1, 2, ..., n-1$ and all other arcs have capacity $\infty$.
  - cost: $c_{v_{ij} v_i} = -f_{ij}$ for $1 \leq i < j \leq n$ and all other arcs have cost zero (including those of the form $v_{ij} v_j$)
  - balance vector: $b(v_{ij}) = b_{ij}$ for $1 \leq i < j \leq n$ and the balance vector of $b(v_i) = -b_{1i} - b_{2i} - ... - b_{i-1,i}$ for $i = 1, 2, ..., n$

# Application Example: Modeling



# Application Example: Modeling

Claim: the network models the ship loading problem.

- suppose that $t_{12}, t_{13}, \ldots, t_{1n}, t_{23}, \ldots, t_{n-1,n}$ are cargo numbers, where $t_{ij}$ $(\leq b_{ij})$ is the amount of cargo the ship will transport from port $i$ to port $j$ and that the ship is never loaded above capacity.
- total income is

$$I = \sum_{1 \leq i < j \leq n} t_{ij} f_{ij}$$

- Let $\boldsymbol{x}$ be the flow in $N$ defined as follows:
  - flow on an arc of the form $v_{ij} v_i$ is $t_{ij}$
  - flow on an arc of the form $v_{ij} v_j$ is $b_{ij} - t_{ij}$
  - flow on an arc of the form $v_i v_{i+1}$, $i = 1, 2, \ldots, n-1$, is the sum of those $t_{ab}$ for which $a \leq i$ and $b \geq i + 1$.
- since $t_{ij}$, $1 \leq i < j \leq n$, are legal cargo numbers then $\boldsymbol{x}$ is feasible with respect to the balance vector and the capacity restriction.
- the cost of $\boldsymbol{x}$ is $-I$.

# Application Example: Modeling

- Conversely, suppose that $\boldsymbol{x}$ is a feasible flow in $N$ of cost $J$.
- we construct a feasible cargo assignment $s_{ij}, 1 \leq i < j \leq n$ as follows:
  - let $s_{ij}$ be the value of $\boldsymbol{x}$ on the arc $v_{ij}v_i$.
- income $-J$