Introduction to Linear Programming

- Examples: resource optimization, the transportation problem, flow problems, portfolio design
- Generic form of linear programs, basic definitions, matrix notation
- General assumptions on problems that can be modeled with a linear program
- Miscellaneous topics: nonnegativity of variables, minimization and maximization, standard and canonical forms, transition between the two
- Notations and linear algebra: vectors, matrices, multiplication of matrices, the Euclidean space, linear independence, linear equations, basic solutions

Optimal Resource Allocation

- **Exercise:** a paper mill manufactures two types of paper, standard and deluxe
  - $\frac{1}{2}$ m^3 of wood is needed to manufacture 1 m^2 of paper (both standard or deluxe)
  - producing 1 m^2 of standard paper takes 1 man-hour, whereas 1 m^2 of deluxe paper requires 2 man-hours
  - every week 40 m^3 wood and 100 man-hours of workforce is available
  - the profit is 3 thousand USD per 1 m^2 of standard paper and 4 thousand USD per 1 m^2 of deluxe paper
- **Question:** how much standard and how much deluxe paper should be produced by the paper mill per week to maximize profits?

Modeling 1: Selecting Variables

- **Optimal Resource Allocation/Product Mix problem:** optimal allocation of resources in order to maximize production profit
- Choose two variables:
    - x_1: the quantity produced from standard paper [m^2]
    - x_2: the quantity produced from deluxe paper [m^2]
- For instance, $x_1 = 12$, $x_2 = 20$ means: 12 m^2 of standard and 20 m^2 of deluxe paper produced, for which the mill uses
    - $\frac{1}{2} * 12 + \frac{1}{2} * 20 = 16$ m^3 wood and
    - $1 * 12 + 2 * 20 = 52$ man-hours of workforce,
    - meanwhile realizing $3 * 12 + 4 * 20 = 116$ thousands USD profits

Modeling 2: Constraints

- **Resource constraint:** the available quantity of wood (40 m^3) limits the amount of paper that can be produced:

$$\frac{1}{2}x_1 + \frac{1}{2}x_2 \leq 40$$

- **Labor constraint:** the available workforce (100 man-hours) also limits the possible production mixes:

$$x_1 + 2x_2 \leq 100$$

- **Nonnegativity:**

$$x_1 \geq 0, \quad x_2 \geq 0$$

Modeling 3: The Objective Function

- The profits: $3x_1 + 4x_2$ [thousand USD]
- **Objective:** to maximize profits:

$$\max 3x_1 + 4x_2$$

in a way so that the amount of wood and workforce used does not exceed the available quantities

Linear Program

$$
\begin{array}{lrcrcr}
\max & 3x_1 & + & 4x_2 & & \\
\text{s.t.} & \frac{1}{2}x_1 & + & \frac{1}{2}x_2 & \leq & 40 \\
 & x_1 & + & 2x_2 & \leq & 100 \\
 & x_1, & & x_2 & \geq & 0
\end{array}
$$

Linear programs: Basic Definitions

- Maximization of the **objective function** that is a linear function of the **decision variables/activities**, or the minimization of a linear **cost function**
- The solution meets the **constraints** that are also linear functions of the variables
- The linear scaling constants are called **objective function coefficients** and **constraint coefficients**
- The combinations of variables x_1, x_2 that meet the constraints are called **feasible solutions** or **feasible points**
- The set of feasible solutions is called the **feasible region**
- The feasible solutions that maximize the objective function (minimize the cost function) are called the **optimal (feasible) solutions** (there can be more than one)
- Decision variables may be subject to nonnegativity or nonpositivity constraints

Modeling Assumptions

- A problem can be modeled and solved by a linear program only if the following assumptions all hold true
  - **Linearity:** the objective function and the constraints are the sums of linear products of the variables
    - **Proportionality:** each decision variable contributes to the objective function/constraints proportionally, independently from the value of other variables (there are no economies or returns to scale or discounts)
    - **Additivity:** the objective/constraints are the sums of the (linear) contributions of the variables (there is no substitution/interaction among the variables)
  - **Divisibility/Continuity:** the values of decision variables can be fractions
  - **Determinism:** the objective function and constraint coefficients are known constants

The General Form of Linear Programs

$$
\begin{array}{lccccccccc}
\max & c_1x_1 & + & c_2x_2 & + & \dots & + & c_nx_n & & \\
\text{s.t.} & a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & \leq & b_1 \\
 & a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & \leq & b_2 \\
 & \vdots & & \vdots & & \ddots & & \vdots & & \vdots \\
 & a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & \leq & b_m \\
 & & & & & & & & & \\
 & x_1, & & x_2, & & \dots, & & x_n & \geq & 0
\end{array}
$$

The General Form of Linear Programs

- m: the number of **rows**, i.e., the number of constraints
- n: the number of **columns**, i.e., the number of variables
- c_j: the objective coefficient for the j-th variable
- $\sum_{j=1}^{n} c_j x_j$: the objective/cost function
- $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$: the i-th constraint
    - a_{ij}: constraint coefficients
    - b_i: the i-th "right-hand-side" (RHS)

The Matrix Form of Linear Programs

- The constraint matrix ($m \times n$):

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

- The objective function/cost vector ($1 \times n$, row vector): $\boldsymbol{c}^T = [c_1 \; c_2 \; \dots c_n]$
- The RHS vector ($m \times 1$, column vector): $\boldsymbol{b} = [b_1 \; b_2 \; \dots \; b_m]^T$
- The vector of variables ($n \times 1$, column vector):

$$\boldsymbol{x} = [x_1 \quad x_2 \quad \dots \quad x_n]^T$$

The Matrix Form of Linear Programs

$$
\begin{array}{lcll}
\max & \begin{bmatrix} c_1 & c_2 & \ldots & c_n \end{bmatrix} \boldsymbol{x} & & \\
\text{s.t.} & \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{bmatrix} \boldsymbol{x} & \leq & \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \\
 & \boldsymbol{x} & \geq & \boldsymbol{0}
\end{array}
$$

$$
\begin{array}{rc}
\max & \boldsymbol{c}^T \boldsymbol{x} \\
\text{s.t.} & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b} \\
 & \boldsymbol{x} \geq \boldsymbol{0}
\end{array}
$$

Optimal Resource Allocation

$$\begin{aligned} \max \quad & 3x_1 + 4x_2 \\ \text{s.t.} \quad & \tfrac{1}{2}x_1 + \tfrac{1}{2}x_2 \leq 40 \\ & x_1 + 2x_2 \leq 100 \\ & x_1, x_2 \geq 0 \end{aligned}$$

$$\begin{aligned} \max \quad & \begin{bmatrix} 3 & 4 \end{bmatrix} \boldsymbol{x} \\ \text{s.t.} \quad & \begin{bmatrix} 1/2 & 1/2 \\ 1 & 2 \end{bmatrix} \boldsymbol{x} \leq \begin{bmatrix} 40 \\ 100 \end{bmatrix} \\ & \boldsymbol{x} \geq 0 \end{aligned}$$

Electric Power Transmission

- **Exercise:** an electricity company supplies 4 cities out of 3 power plants; the demand at each city is given but the capacity of the power plants is limited and the transmission loss increases proportionally with the distance between a city and a power plant
- **Task:** match cities to power plants in a way as to minimize the overall transmission loss

Electric Power Transmission

- **Transportation/transshipment problem:** matching demands to supplies in a way as to minimize loss

Plant		City			
	Capacity	City1	City2	City3	City4
Plant1	35	8	6	10	9
Plant2	50	9	12	13	7
Plant3	40	14	9	16	5
Demand		45	20	30	30

(Capacity: [GWh], Demand: [GWh], Cost: [million USD/GWh])

Modeling 1: Selecting Variables

- x_{ij}: the quantity of electricity to be transmitted from power plant i to city j [GWh]
- For example, x_{14} means the quantity of electricity to be transmitted from *Plant1* to *City4*

Modeling 2: Constraints

- **Supply constraints:** the total amount of electricity to be transmitted from power plant i cannot exceed its capacity:

$$\sum_{j=1}^{4} x_{ij} \leq \text{capacity}_i$$

- **Demand constraint:** the amount of electricity to be transmitted to city j must meet the demand and city j:

$$\sum_{i=1}^{3} x_{ij} = \text{demand}_j$$

- **Nonnegativity:** negative quantity of electricity cannot be transmitted:

$$x_{ij} \geq 0 \qquad \forall i \in \{1,2,3\},\ \forall j \in \{1,2,3,4\}$$

Modeling 3: The Cost Function

- The quantity of electricity transmitted from plant i to city j equals x_{ij}, the cost of which due to transmission losses:

$$\text{cost}_{ij} x_{ij}$$

- **Objective:** minimize the total cost:

$$\min \sum_{i=1}^{3} \sum_{j=1}^{4} \text{cost}_{ij} x_{ij}$$

in a way so as to x_{ij} meet the demand, supply, and nonnegativity constraints

Linear Program

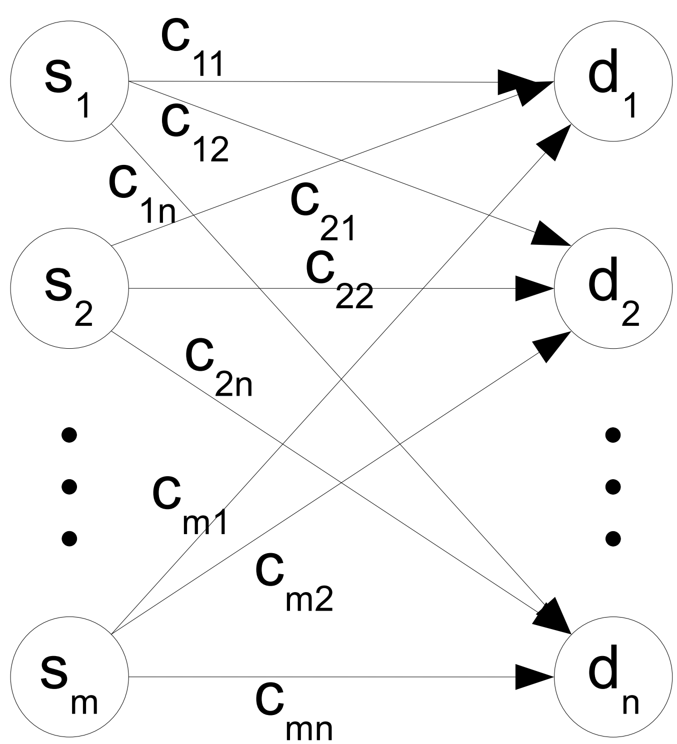
$$\min \quad 8x_{11} + 6x_{12} + 10x_{13} + 9x_{14} + 9x_{21} + 12x_{22} + 13x_{23} + 7x_{24} + 14x_{31} + 9x_{32} + 16x_{33} + 5x_{34}$$

$$\begin{aligned}
\text{s.t.} \quad x_{11} + x_{12} + x_{13} + x_{14} &\leq 35 \\
x_{21} + x_{22} + x_{23} + x_{24} &\leq 50 \\
x_{31} + x_{32} + x_{33} + x_{34} &\leq 40 \\
x_{11} + x_{21} + x_{31} &= 45 \\
x_{12} + x_{22} + x_{32} &= 20 \\
x_{13} + x_{23} + x_{33} &= 30 \\
x_{14} + x_{24} + x_{34} &= 30
\end{aligned}$$

$$x_{ij} \geq 0 \quad \forall i \in \{1, 2, 3\}, \quad \forall j \in \{1, 2, 3, 4\}$$

Transshipment Problem: General Form

- Given m **supply points**, where the capacity of the i-th supply is s_i
- Given n **demand points**, where the j-th demand is d_j
- And the cost of transmission of one unit of goods from the supply point i to demand point j is c_{ij}
- Minimize the total cost



Transshipment Problem: General Form

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} x_{ij} \leq s_i \qquad \forall i \in \{1, \ldots, m\}$$

$$\sum_{i=1}^{m} x_{ij} = d_j \qquad \forall j \in \{1, \ldots, n\}$$

$$x_{ij} \geq 0 \qquad \forall i \in \{1, \ldots, m\}, \ \forall j \in \{1, \ldots, n\}$$

Direction of Optimization

- If the objective function represents
  - profits: **maximization**
  - cost: **minimization**
- Conversion:

$$\max\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \le \boldsymbol{b}, \boldsymbol{x} \ge \boldsymbol{0}\} =$$
$$-1 * \min\{-\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \le \boldsymbol{b}, \boldsymbol{x} \ge \boldsymbol{0}\}$$

Constraints: Forms

- **Inequality and equality:** there are two types of constrains in the transportation problem
  - Supply constraint: $\sum_{j=1}^{4} x_{ij} \leq \text{capacity}_i$
  - Demand constraint: $\sum_{i=1}^{3} x_{ij} = \text{demand}_j$
- Conversion: inequality $\rightarrow$ equality
  - "$\leq$" type inequality: by **adding** an artificial **slack** variable

$$\sum_{j=1}^{n} a_{ij}x_j \leq b_i \Longleftrightarrow \sum_{j=1}^{n} a_{ij}x_j + x_{s_i} = b_i, \quad x_{s_i} \geq 0$$

Constraints: Forms

- Conversion: inequality $\rightarrow$ equality
  - "$\geq$" type inequality: by **subtracting** an artificial **slack** variable

$$\sum_{j=1}^{n} a_{ij}x_j \geq b_i \Longleftrightarrow \sum_{j=1}^{n} a_{ij}x_j - x_{s_i} = b_i, \quad x_{s_i} \geq 0$$

- Conversion: equality $\rightarrow$ inequality
  - a "$=$" type constraint can be substituted with a "$\leq$" type and a "$\geq$" type constraint

$$\sum_{j=1}^{n} a_{ij}x_j = b_i \Longleftrightarrow \sum_{j=1}^{n} a_{ij}x_j \leq b_i$$

$$\sum_{j=1}^{n} a_{ij}x_j \geq b_i$$

Nonnegativity

- In practice the variables are almost always constrained as nonnegative
- Substituting a nonpositive variable: $x_j = -x'_j$

$$\begin{array}{ccc} x_j \leq 0 & & x'_j \geq 0 \\ a_{ij}x_j & \Longleftrightarrow & -a_{ij}x'_j \\ c_j x_j & & -c_j x'_j \end{array}$$

- Substituting a free variable: $x_j = x'_j - x''_j$

$$\begin{array}{ccc} x_j \overset{\leq}{\underset{\geq}{=}} 0 & & x'_j \geq 0,\ x''_j \geq 0 \\ a_{ij}x_j & \Longleftrightarrow & a_{ij}(x'_j - x''_j) \\ c_j x_j & & c_j(x'_j - x''_j) \end{array}$$

The Canonical and the Standard Forms

	Minimization	Maximization
Standard form	$\min \boldsymbol{c}^T \boldsymbol{x}$ s.t. $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ $\boldsymbol{x} \geq \boldsymbol{0}$	$\max \boldsymbol{c}^T \boldsymbol{x}$ s.t. $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ $\boldsymbol{x} \geq \boldsymbol{0}$
Canonical form	$\min \boldsymbol{c}^T \boldsymbol{x}$ s.t. $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$ $\boldsymbol{x} \geq \boldsymbol{0}$	$\max \boldsymbol{c}^T \boldsymbol{x}$ s.t. $\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$ $\boldsymbol{x} \geq \boldsymbol{0}$

Logistics

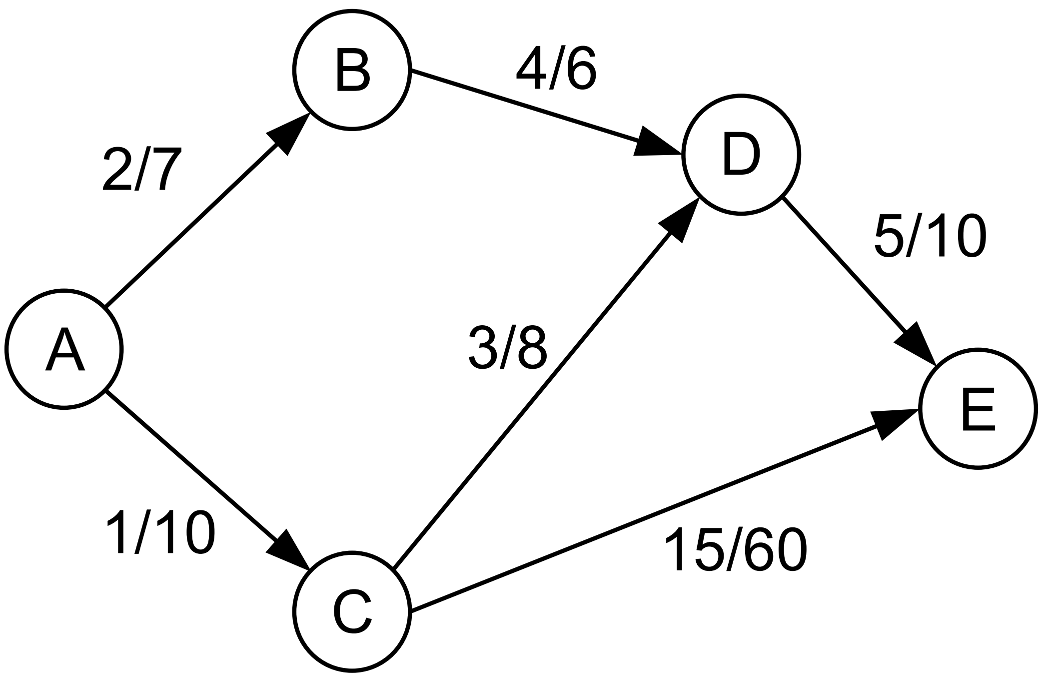
- **Exercise:** a transportation company can use the below list of freight ship routes between points A, B, C, D and E, each with limited capacity (u, [t]) and operating at specific price per tonne of cargo (c, [million USD/t])

c/u	A	B	C	D	E
A	-	2/7	1/10	-	-
B	-	-	-	4/6	-
C	-	-	-	3/8	15/60
D	-	-	-	-	5/10
E	-	-	-	-	-

- **Task:** to ship 15 tonnes of cargo from point A to point E at minimal transportation cost

Logistics

- **Flow problem:** generalization of the transportation problem
  - connection exists only between a subset of points
  - capacity of connections is limited
- Connections make up a capacitated graph $G(V, E)$, with V being the ports and E being the set of ship routes



Flow problem

- x_{ij}: quantity of goods to be transported between point i and j [t]
- x defines a **flow** on graph $G(V, E)$:
  - **capacity constraint:** $\forall (i, j) \in E : x_{ij} \leq u_{ij}$
  - **flow conservation:** the difference between the amount of flow entering point i and the amount of flow leaving it equals the difference of the demand and supply at i:

$$\forall i \in V : \sum_{j:(j,i) \in E} x_{ji} - \sum_{j:(i,j) \in E} x_{ij} = d_i$$

  - **nonnegativity:** $\forall (i, j) \in E : x_{ij} \geq 0$
- Total cost: $\sum_{(i,j) \in E} c_{ij} x_{ij}$

Logistics

$$
\begin{array}{llllllllll}
\min & 2x_{AB} & + x_{AC} & + 4x_{BD} & + 3x_{CD} & + 15x_{CE} & + 5x_{DE} & & \\
\text{s.t.} & -x_{AB} & - x_{AC} & & & & & = & -15 \\
& x_{AB} & & - x_{BD} & & & & = & 0 \\
& & + x_{AC} & & - x_{CD} & - x_{CE} & & = & 0 \\
& & & x_{BD} & x_{CD} & & - x_{DE} & = & 0 \\
& & & & & x_{CE} & + x_{DE} & = & 15 \\
& x_{AB} & & & & & & \leq & 7 \\
& & x_{AC} & & & & & \leq & 10 \\
& & & x_{BD} & & & & \leq & 6 \\
& & & & x_{CD} & & & \leq & 8 \\
& & & & & x_{CE} & & \leq & 60 \\
& & & & & & x_{DE} & \leq & 10 \\
& x_{AB}, & x_{AC}, & x_{BD}, & x_{CD}, & x_{CE}, & x_{DE} & \geq & 0
\end{array}
$$

Logistics: Matrix Form

$$N = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \quad d = \begin{bmatrix} -15 \\ 0 \\ 0 \\ 0 \\ 15 \end{bmatrix}$$

$$c^T = [\ 2 \ \ 1 \ \ 4 \ \ 3 \ \ 15 \ \ 5], \quad x = \begin{bmatrix} x_{AB} \\ x_{AC} \\ x_{BD} \\ x_{CD} \\ x_{CE} \\ x_{DE} \end{bmatrix}, \quad u = \begin{bmatrix} 7 \\ 10 \\ 6 \\ 8 \\ 60 \\ 10 \end{bmatrix}$$

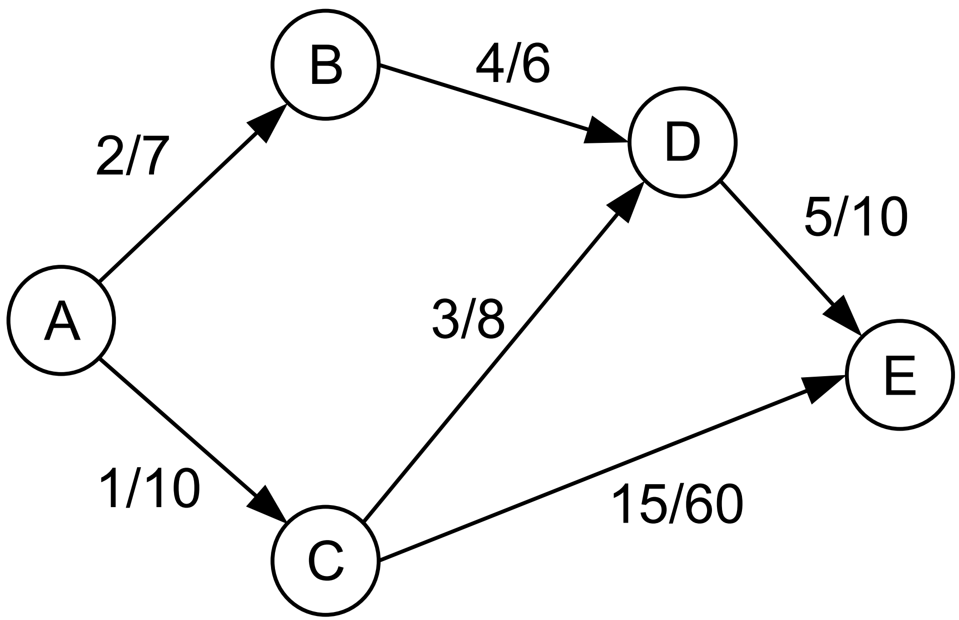
The Minimal Cost Flow Problem

$$\begin{aligned} \min \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{N}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \leq \boldsymbol{u} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

- Here the matrix $\boldsymbol{N}$ is representative for the graph G (see later)
  - node-arc incidence matrix
  - notation: $\boldsymbol{N}$

Logistics

- **Exercise:** this time not only one but two transportation companies provide transshipment service over the aforementioned ship routes: the first company carries 15 tonnes of goods from point A to point E whereas the second one ships 20 tonnes between C and E
- **Task:** compute the minimal-cost allocation of transshipment capacities, subject to ship capacities



Logistics

- x_{ij}^k: the amount of goods to be shipped from point i to point j by company k [t]
- The goods do not mix, therefore for each k the variables $x_{ij}^k : (i,j) \in E$ define a flow, satisfying (independently from other ks)
  - the flow conservation constraints: $\boldsymbol{N}\boldsymbol{x}^k = \boldsymbol{d}^k$
  - and nonnegativity: $\boldsymbol{x}^k \geq \boldsymbol{0}$
- The total quantity of cargo allocated to a particular ship cannot exceed its capacity: $\boldsymbol{x}^1 + \boldsymbol{x}^2 \leq \boldsymbol{u}$

Logistics

- For the first transportation company:

$$\boldsymbol{d}^1 = \begin{bmatrix} -15 \\ 0 \\ 0 \\ 0 \\ 15 \end{bmatrix}, \qquad \boldsymbol{x}^1 = \begin{bmatrix} x^1_{AB} \\ x^1_{AC} \\ x^1_{BD} \\ x^1_{CD} \\ x^1_{CE} \\ x^1_{DE} \end{bmatrix}$$

- For the second transportation company:

$$\boldsymbol{d}^2 = \begin{bmatrix} 0 \\ 0 \\ -20 \\ 0 \\ 20 \end{bmatrix}, \qquad \boldsymbol{x}^2 = \begin{bmatrix} x^2_{AB} \\ x^2_{AC} \\ x^2_{BD} \\ x^2_{CD} \\ x^2_{CE} \\ x^2_{DE} \end{bmatrix}$$

Multicommodity Flow Problem

$$\begin{aligned} \min \quad & \sum_{k \in K} \boldsymbol{c}^T \boldsymbol{x}^k \\ \text{s.t.} \quad & \boldsymbol{N}\boldsymbol{x}^k = \boldsymbol{d}^k \quad \forall k \in K \\ & \sum_{k \in K} \boldsymbol{x}^k \leq \boldsymbol{u} \\ & \boldsymbol{x}^k \geq \boldsymbol{0} \qquad \forall k \in K \end{aligned}$$

- K: the set of “goods” or commodities

Portfolio Design

- **Exercise:** the task is to design the financing for a 4 year construction project
  - construction works scheduled for the first year cost 2 million USD, 4 million in the second year, 8 million USD in the third and 5 million in the fourth year
  - costs will be covered by bonds (“debt instruments”) that must be paid back continuously for 20 years from the completion of the construction (“maturity is 20 years”)
  - the expected bond interest rate is 7% for the first year, 6% for the second year, then 6.5% and 7.5%
  - the money *not* spent on the construction can be invested into short-term securities, for which interest rates for the first year is 6%, 5.5% for the second year, and 4.5% for the third year (in the fourth year it is not worth investing)
- **Task:** choose the optimal portfolio

Portfolio Design

- Determine the optimal schedule for bond issuance and short-term investment
  - bonds must be paid back, interest rates depend on the year of issuance
  - a fixed portion of the income must be spent to cover construction costs
  - the rest can be invested into short-term securities to cover the bond coupons to be paid back
  - how to schedule bond issuance and short-term investment in each year in order to minimize the total cost of the construction, given the anticipated interest rates?

Portfolio Design

- $x_j : j = 1, \ldots, 4$: the quantity of bonds issued at the beginning of year j [million USD]
- $y_j : j = 1, \ldots, 3$: assets invested into short-term securities at the beginning of year j [million USD]
- In the first year, part of the income x_1 must be spent on the construction (2 million USD), the rest (y_1) can be invested

$$x_1 = 2 + y_1$$

- In the second year
  - income: from bond issuance x_2 plus the amount earned from previous year's investments with premium $(1.06 y_1)$
  - expenditure: construction work (4 million USD) plus additional short-term investment (y_2)

$$x_2 + 1.06 y_1 = 4 + y_2$$

Portfolio Design

- In the third year
  - income: from bond issuance x_3 plus the amount earned from previous year's investments with premium ($1.055y_2$)
  - expenditure: construction work (8 million USD) plus additional short-term investment (y_3)

$$x_3 + 1.055y_2 = 8 + y_3$$

- In the fourth year
  - income: from bond issuance x_4 plus the amount earned from previous year's investments with premium ($1.045y_3$)
  - expenditure: construction work (5 million USD), no new short-term investment (end of financial period)

$$x_4 + 1.045y_3 = 5$$

Portfolio Design

- **Objective:** minimize the premium (interest on bonds) that is to be paid back to investors
- Maturity is 20 years, according to the interest rate at the year of issuance
  - interest on the bonds issued in the first year: $(20 * 0.07)x_1$
  - interest on the bonds issued in the second year: $(20 * 0.06)x_2$
  - interest on the bonds issued in the third year: $(20 * 0.065)x_3$
  - interest on the bonds issued in the fourth year: $(20 * 0.075)x_4$

Portfolio Design

$$
\begin{aligned}
\min \quad & (20 * 0.07)x_1 + (20 * 0.06)x_2 + \\
& (20 * 0.065)x_3 + (20 * 0.075)x_4 \\
\text{s.t.} \quad & x_1 - y_1 = 2 \\
& 1.06y_1 + x_2 - y_2 = 4 \\
& 1.055y_2 + x_3 - y_3 = 8 \\
& 1.045y_3 + x_4 = 5 \\
& x_j \geq 0 \qquad \forall i \in \{1, \ldots, 4\} \\
& y_j \geq 0 \qquad \forall i \in \{1, \ldots, 3\}
\end{aligned}
$$

Applications of Linear Programming

- Management science/Operations research
  - resource allocation
  - portfolio design
- Logistics
  - transportation problems
  - production planning
- Telecommunications
  - network optimization
  - routing and traffic engineering

Notation: Vectors

- **Column** n**-vector:** $\boldsymbol{x} = [x_i] = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$, with **elements** x_i
- **Row** m**-vector:** $\boldsymbol{x}^T = [x_j] = \begin{bmatrix} x_1 & \ldots & x_m \end{bmatrix}$, elements (or components) x_j, notation $(.)^T$ is for **transposition**
- Vectors are sometimes also called **points**
- **Zero vector:** $\mathbf{0}$ and **1-vector** (each element equals 1): $\mathbf{1}$
- The i**-th canonical unit vector:** $\boldsymbol{e_i}$, $\boldsymbol{e_i}^T$

$$\boldsymbol{e_i}^T = \underbrace{\begin{bmatrix} 0 & 0 & \ldots & 1 & \ldots & 0 \end{bmatrix}}_{1 \text{ at position } i}$$

Operations on Vectors

- **Sum of vectors:** according to the Parallelogram Law

$$\boldsymbol{x} + \boldsymbol{y} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{bmatrix}$$

- **Multiplying a vector with a scalar:** $\lambda \boldsymbol{x} = [\lambda x_i]$
- The **scalar product** of a row n-vector and a column n-vector

$$\boldsymbol{x}^T \boldsymbol{y} = \sum_{i=1}^{n} x_i y_i$$

Linear Independence

- A vector $\boldsymbol{b}$ is a **linear combination** of vectors $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, …, $\boldsymbol{a}_k$, if $\boldsymbol{b} = \sum_{i=1}^{k} \lambda_i \boldsymbol{a}_i$ for some real scalars $\lambda_1, \lambda_2, \ldots, \lambda_k$
- Vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_k$ are **linearly independent**, if

$$\sum_{i=1}^{k} \lambda_i \boldsymbol{a}_i = \mathbf{0} \Rightarrow \forall i = \{1, 2, \ldots, k\} : \lambda_i = 0$$

- Real n-vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_k \in \mathbb{R}^n$ **span the vector space** $V \subseteq \mathbb{R}^n$, if each vector $\boldsymbol{b} \in V$ can be written as a linear combination of vectors $\boldsymbol{a}_i$
- The minimal set of n-vectors $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, …, $\boldsymbol{a}_k$ that span V is called a **basis** of V, and k is the **dimension** of V

Notation: Matrices

- A **mátrix** $\boldsymbol{A}$ of size $m \times n$

$$\boldsymbol{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a^1} \\ \boldsymbol{a^2} \\ \vdots \\ \boldsymbol{a^m} \end{bmatrix} = [\boldsymbol{a_1}, \boldsymbol{a_2}, \ldots, \boldsymbol{a_n}]$$

- A **zero matrix** is a matrix whose all elements are 0
- The $n \times n$ (canonical) **unit matrix**

$$\boldsymbol{I}_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

Operations on Matrices

- **Sum of matrices:** of $\boldsymbol{A}$ and $\boldsymbol{B}$ are $m \times n$ then

$$\boldsymbol{A} + \boldsymbol{B} = [a_{ij} + b_{ij}]$$

- **Matrix multiplication:** the multiple of matrix $\boldsymbol{A}$ ($m \times p$) and matrix $\boldsymbol{B}$ ($p \times n$) is matrix $\boldsymbol{C} = \boldsymbol{A}\boldsymbol{B}$ ($m \times n$)

$$c_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj} = \boldsymbol{a^i b_j}$$

- If $\boldsymbol{A}$ is $n \times n$ quadratic and there is matrix $\boldsymbol{A}^{-1}$ so that $\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}_n$, then $\boldsymbol{A}^{-1}$ is called the **inverse** of $\boldsymbol{A}$
- The maximal number of linearly independent rows (or columns) of $\boldsymbol{A}$ is called the **rank** of $\boldsymbol{A}$
- The inverse exists if all rows (or columns) of $\boldsymbol{A}$ are linearly independent ($\boldsymbol{A}$ is **nonsingular**)

Determinant

- The **determinant** of a real valued $n \times n$ matrix $\boldsymbol{A}$ is

$$\det \boldsymbol{A} = \sum_{i=1}^{n} a_{ij} A_{ij} \quad ,$$

  where a_{ij} is the element in position (i, j) of $\boldsymbol{A}$ and A_{ij} is the **cofactor** of a_{ij}, which is obtained by removing row i and column j from $\boldsymbol{A}$ and then taking the determinant multiplied by $(-1)^{i+j}$
- $\det \boldsymbol{A} = \det \boldsymbol{A}^T$
- if $\boldsymbol{B}$ is obtained by swapping two columns or rows of $\boldsymbol{A}$ then $\det \boldsymbol{B} = -\det \boldsymbol{A}$
- $\boldsymbol{A}$ is nonsingular if and only if $\det \boldsymbol{A} \neq 0$

Systems of Linear Equations

- Let $\boldsymbol{A}$ be an $m \times n$ matrix and let $\boldsymbol{b}$ be a column m-vector
- We seek a column n-vector $\boldsymbol{x}$ so that $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$
- If $\boldsymbol{b}$ is linearly independent from the column vectors of $\boldsymbol{A}$ then it cannot be written as the linear combination of $\boldsymbol{A}$'s columns: no solution exists
- Let $\operatorname{rank}(\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{A}, \boldsymbol{b}) = k$
- Reorder the rows of $\boldsymbol{A}$ and $\boldsymbol{b}$ so that the first k rows contain the linearly independent rows

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix}, \ \boldsymbol{b} = \begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \end{bmatrix} \quad ,$$

where $\boldsymbol{A}_1$ is $k \times n$ with $\operatorname{rank}(\boldsymbol{A}_1) = k$ and $\boldsymbol{A}_2$ is $(m-k) \times n$, and $\boldsymbol{b}_1$ is a column k-vector and $\boldsymbol{b}_2$ is a column $(m-k)$-vector

Systems of Linear Equations

- If $\boldsymbol{A}_1\boldsymbol{x} = \boldsymbol{b}_1$ then $\boldsymbol{A}_2\boldsymbol{x} = \boldsymbol{b}_2$ automatically holds, as the rows of $[\boldsymbol{A}_2\ \boldsymbol{b}_2]$ can be written as a linear combination of the rows of $[\boldsymbol{A}_1\ \boldsymbol{b}_1]$: we can safely ignore the rows of $\boldsymbol{A}_2\boldsymbol{x} = \boldsymbol{b}_2$
- Since $\mathrm{rank}(\boldsymbol{A}_1) = k$, we can select k linearly independent columns from $\boldsymbol{A}_1$
- Rearrange the columns of $\boldsymbol{A}_1$ so that the first k columns are linearly independent: $\boldsymbol{A}_1 = [\boldsymbol{B}\ \boldsymbol{N}]$ where
  - $\boldsymbol{B}$ is a $k \times k$ quadratic, nonsingular matrix, called the **basis matrix**
  - $\boldsymbol{N}$ is $k \times (n-k)$, called the (**nonbasic matrix**)
- Similarly, reorder the elements of $\boldsymbol{x}$ accordingly: $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_{\boldsymbol{B}} \\ \boldsymbol{x}_{\boldsymbol{N}} \end{bmatrix}$, where $\boldsymbol{x}_{\boldsymbol{B}}$ contains the variables corresponding to the columns of $\boldsymbol{B}$ and $\boldsymbol{x}_{\boldsymbol{N}}$ contains the rest

Systems of Linear Equations

- The rearranged system

$$[\boldsymbol{B}\ \boldsymbol{N}] \begin{bmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{bmatrix} = \boldsymbol{b}_1, \text{ vagyis } \boldsymbol{B}\boldsymbol{x}_{\boldsymbol{B}} + \boldsymbol{N}\boldsymbol{x}_{\boldsymbol{N}} = \boldsymbol{b}_1$$

- Multiply by the inverse of $\boldsymbol{B}$ from the left and rearrange

$$\boldsymbol{x}_{\boldsymbol{B}} = \boldsymbol{B}^{-1}\boldsymbol{b}_1 - \boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_{\boldsymbol{N}}$$

- If $k = n$ then we have a unique solution: $\boldsymbol{x}_{\boldsymbol{B}} = \boldsymbol{B}^{-1}\boldsymbol{b}_1$ (**basic solution)**
- If $k < n$ then $\boldsymbol{x}_{\boldsymbol{N}}$ can be chosen arbitrarily, the number of solutions is infinite

Systems of Linear Equations: Example

$$\begin{array}{rcrcrcrcl} & x_1 & + & 2x_2 & + & x_3 & - & 2x_4 & = & 10 \\ - & x_1 & + & 2x_2 & - & x_3 & + & x_4 & = & 6 \\ & & & x_2 & + & x_3 & & & = & 2 \end{array}$$

- In matrix form

$$\begin{bmatrix} 1 & 2 & 1 & -2 \\ -1 & 2 & -1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} 10 \\ 6 \\ 2 \end{bmatrix}$$

- We search for a basic solution: perform Gaussian Elimination on $(\boldsymbol{A}, \boldsymbol{b})$

$$\left[\begin{array}{rrrr|r} \boxed{1} & 2 & 1 & -2 & 10 \\ -1 & 2 & -1 & 1 & 6 \\ 0 & 1 & 1 & 0 & 2 \end{array}\right]$$

Systems of Linear Equations: Example

- Add the first row to the second one:

$$\left[\begin{array}{cccc|c} 1 & 2 & 1 & -2 & 10 \\ 0 & \boxed{4} & 0 & -1 & 16 \\ 0 & 1 & 1 & 0 & 2 \end{array}\right]$$

- Similarly

$$\left[\begin{array}{cccc|c} 1 & 0 & 1 & -\frac{3}{2} & 2 \\ 0 & 1 & 0 & -\frac{1}{4} & 4 \\ 0 & 0 & \boxed{1} & \frac{1}{4} & -2 \end{array}\right]$$

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & -\frac{7}{4} & 4 \\ 0 & 1 & 0 & -\frac{1}{4} & 4 \\ 0 & 0 & 1 & \frac{1}{4} & -2 \end{array}\right]$$

Systems of Linear Equations: Example

- The original system: $\boldsymbol{Ax} = \boldsymbol{b}$
- Choosind the first 3 columns as basis: $[\boldsymbol{B}\ \boldsymbol{N}] \begin{bmatrix} \boldsymbol{x_B} \\ \boldsymbol{x_N} \end{bmatrix} = \boldsymbol{b}$
- Multiply from the left with $\boldsymbol{B}^{-1}$

$$\boldsymbol{B}^{-1}[\boldsymbol{B}\ \boldsymbol{N}] \begin{bmatrix} \boldsymbol{x_B} \\ \boldsymbol{x_N} \end{bmatrix} = [\boldsymbol{I} \quad \boldsymbol{B}^{-1}\boldsymbol{N}] \begin{bmatrix} \boldsymbol{x_B} \\ \boldsymbol{x_N} \end{bmatrix} = \boldsymbol{B}^{-1}\boldsymbol{b}$$

- Substituting into the form $\boldsymbol{x_B} = \boldsymbol{B}^{-1}\boldsymbol{b} - \boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x_N}$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ -2 \end{bmatrix} - \begin{bmatrix} -\frac{7}{4} \\ -\frac{1}{4} \\ \frac{1}{4} \end{bmatrix} x_4$$

- x_4 can be chosen arbitrarily