CS-602 - Design of Problem Solvers

Chapter 6 - Linear Programming for Optimization Problems

Dr. Mahdi Khemakhem

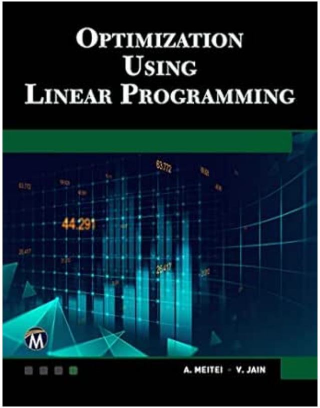
Department of Computer Science
College of Computer Engineering and Science
Prince Sattam bin Abdulaziz University

AY - 2025/2026

Outline

Materials

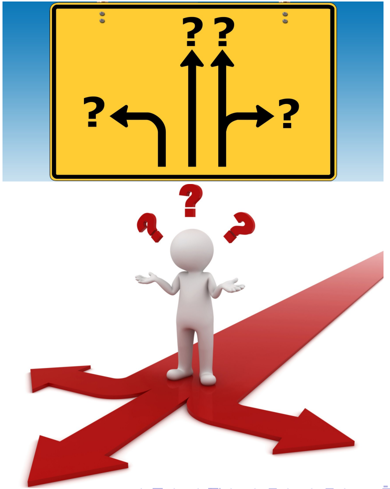
- **Textbook:** "Optimization Using Linear Programming" by *A. J. Metei and Veena Jain*
- **⇒ Read Chapters 1 and 2 for this chapter's material**



Outline

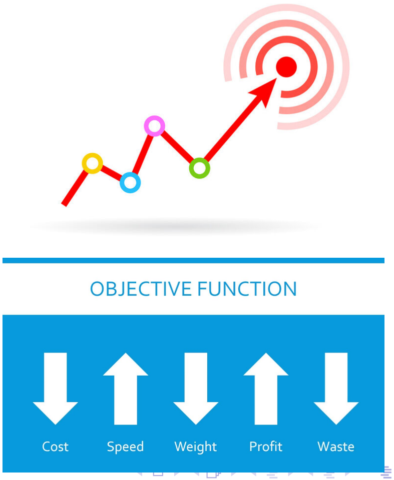
What is an Optimization Problem?

- An **optimization problem (OP)** is a computational problem in which the object is to find **the best of all possible solutions**.
- More formally, an **OP** is to find a solution in the **feasible region** which has the minimum (or maximum) value of the **objective function**.



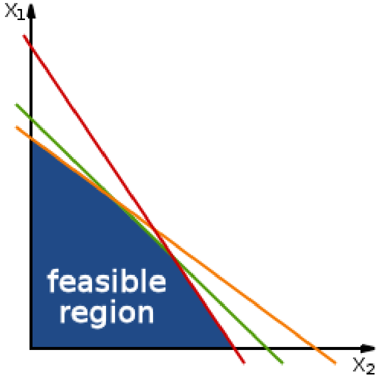
What is an Objective Function?

- An **objective function (OF)** is a function associated with an **OP** which determines **how good a solution is**.
- An **OF** should be defined using **decision variables**.



What is a Feasible Region?

- A **feasible region** is the **set of all possible solutions** of an **OP**.
- In mathematical optimization, a **feasible region**, **feasible set**, **search space**, or **solution space** is the **set of all possible points** (sets of values of the choice variables) of an **OP** that satisfy the problem's **constraints**.



What are Constraints?

- **Constraints** are **logical conditions** that a **solution** to an **OP must satisfy**.
- **Constraints** reflect real-world limits on **production capacity**, **market demand**, **available funds**, and so on.
- **Constraints** should be defined using **decision variables**.

What is a Decision Variable?

- A **decision variable** is an **unknown-value variable** associated to an **OP**.
- **Decision variable** has **a domain**, which is a compact representation of the **set of all possible values for the variable**.
- **Decision variable types** are references to objects whose exact nature depends on the underlying **optimizer of a model**.
- **Decision variables** are a set of quantities to be determined for solving the **OP**; i.e., the problem is solved when the best values of the **decision variables** have been identified.
- They are called **decision variables because the problem is to decide what value each variable should take**.

Optimization Problem Requirements

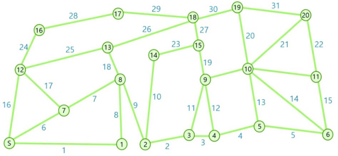
An **optimization problem** definition **requires** a prior definitions of:

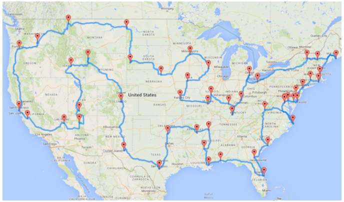
1. A set of **decision variable(s)**.
2. An **objective function** using defined **decision variable(s)** to precise how good solution is.
3. A set of **constraints(s)** using defined **decision variable(s)** to determine the **feasible region** containing all **OP**'s possible solutions.

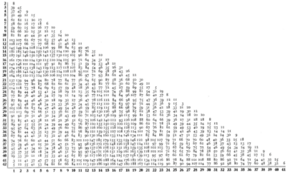
Solving an **optimization problem** consists to **finding** the **better solution(s)** in the **feasible region** according to the defined **objective function** and the **constraint(s)** set **limitations**.

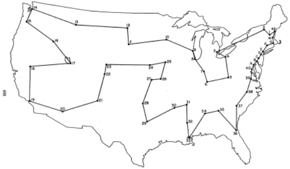
Traveling Salesman Problem (1/2)

The most popular combinatorial optimization problem is the **Traveling Salesman Problem (TSP)**. It can be defined as follows: given n cities and a distance matrix $d_{n,n}$, where each element d_{ij} represents the distance between the cities i and j, find a tour that minimizes the total distance. A tour visits each city exactly once (Hamiltonian cycle).





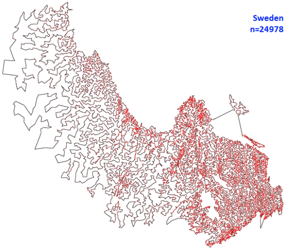




Traveling Salesman Problem (2/2)

- **The size of the search space is $n!$.**
- Table below shows the **combinatorial explosion** of the number of solutions regarding the number of cities.
- **Unfortunately, enumerating exhaustively all possible solutions is impractical for moderate and large instances.**

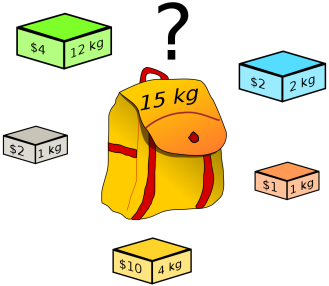
Number of Cities n	**Size of the Search Space**
5	120
10	3,628,800
75	2.5×10^{109}



Knapsack Problem

The **Knapsack Problem (KP)** can be defined as follows. Given a set of N objects. Each object i has a specified weight w_i and a specified profit p_i. Given a capacity C, which is the maximum total weight of the knapsack.

The KP consists of finding a subset of the objects whose total weight is at most equal to the knapsack capacity and whose total profit is maximized.



Outline

Classical Optimization Models

- **Optimization problems are encountered in many domains**: science, engineering, management, and business.
- An **optimization problem** may be defined by the **couple** (S, f), where S represents the **set of feasible solutions**[1], and $f : S \to \mathbb{R}$ the **objective function**[2] **to optimize**.
- The **objective function** f **assigns** to every solution $s \in S$ of the **search space** a **real number indicating its value**.
- The **objective function** f **allows to define** a total order relation between any pair of solutions in the search space.

[1] **Sometimes named: Feasible Region, or Feasible Set, or Search Space, or Solution Space**
[2] **Sometimes named: Cost, Utility, or Fitness Function.**

Global Optimum

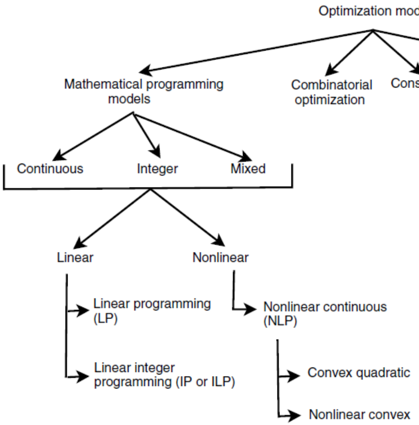
A solution $s^* \in S$ is a **global optimum** if it **has a better objective function**[3] **than all solutions of the search space**, that is, $\forall s \in S, f(s^*) \leq f(s)$.

- The **main goal** in **solving** an **optimization problem** is to **find** a **global optimal solution** s^*.
- Many **global optimal solutions may exist** for a given problem.
- Hence, to get more alternatives, the problem may also be defined as finding all global optimal solutions.

[3] **We suppose without loss of generality a minimization problem. Maximizing an objective function f is equivalent to minimizing $-f$.**

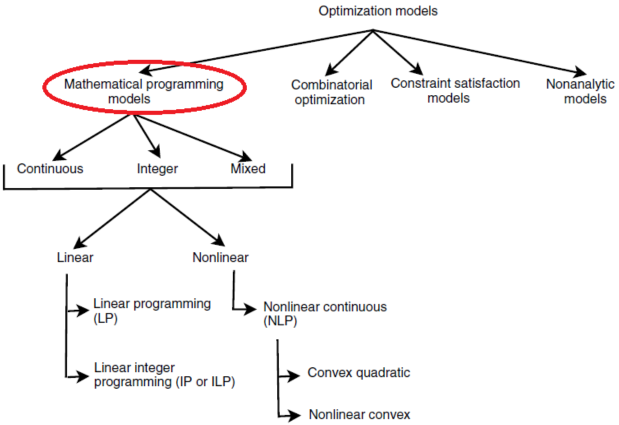
Different Families of Optimization Models

- Different **families of optimization models** are **used** in practice to **formulate** and **solve decision-making problems**.



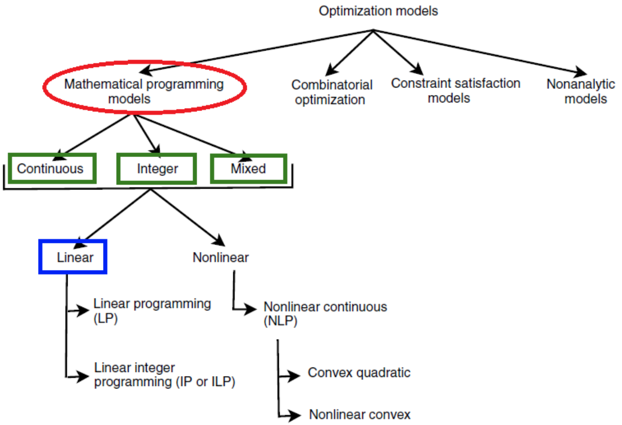
Different Families of Optimization Models

- Different **families of optimization models** are **used** in practice to **formulate** and **solve decision-making problems**.
- The most successful models are based on **mathematical programming**.



Different Families of Optimization Models

- Different **families of optimization models** are **used** in practice to **formulate** and **solve decision-making problems**.
- The most successful models are based on **mathematical programming**.
- The most common **mathematical programming models** are **linear** with **continuous**, or **integer**, or **mixed decision variables**.



Outline

LP Model Definition

- A commonly used model in **mathematical programming** is **linear programming (LP)**.
- **Linear programming (LP) model** or **Linear Optimisation** may be defined as the problem of **maximizing** or **minimizing** a **linear function** that is **subjected to linear constraints**.
- **The constraints** may be **equalities** or **inequalities**.
- **Linear programming problems** are an important class of optimisation problems, that **helps to find the feasible region** and **optimise the solution in order to have the highest or lowest value of the function**.
- The **basic components of the LP** are as follows:
  - Data
  - Decision Variables
  - Constraints
  - Objective Function(s)

LP Model Characteristics (1/2)

The **six characteristics** of the **linear programming model** are as follows:

1. Decision Variables
   - **Unknown-value variables in a specified domain.**
   - They are expected to be **estimated as an output of the LP model solution**.
   - For any problem, the first step is to **identify the decision variables and their domains**.
2. Objective Function
   - Determines the **objective of the LP problem** that generally aims to either **maximize** and/or **minimize** some quantitative value.
   - **Mathematical function** that evaluates the **amount** by which **each decision variable** would contribute to the net present value.
3. Constraints
   - The **problem restrictions** or **limitations** that **deny to reach an infinite profit or a zero cost**.
   - They should be **expressed in the mathematical form (inequalities and/or equalities)**.

LP Model Characteristics (2/2)

4 Linearity

- The **relationship between two or more variables** of the **objective function** and the **constraints must be linear**. It means that the degree of the variable is one.
- **Products of decision variables are denied.**

5 Finiteness

- There should be **finite** and **infinite input** and **output** numbers.
- In case, if the **objective function** or a **constraint has infinite factors**, the **optimal solution is not feasible**.

6 Non-negativity

- Each **decision variable value should be positive or zero**.
- **It should not be a negative value**.

LP Expanded Form

A Linear Programming (LP) model, with n **decision variables** and m **constraints**, can be formulated as follows:

$$\textbf{Min (or Max) } Z = \sum_{i=1}^{n} c_i x_i$$

subject to **(s.t.)**:

$$\sum_{i=1}^{n} a_{1i} x_i \leq (\text{or } \geq \text{ or } =) b_1$$

$$\sum_{i=1}^{n} a_{2i} x_i \leq (\text{or } \geq \text{ or } =) b_2$$

$$\vdots$$

$$\sum_{i=1}^{n} a_{ji} x_i \leq (\text{or } \geq \text{ or } =) b_j$$

$$\vdots$$

$$\sum_{i=1}^{n} a_{mi} x_i \leq (\text{or } \geq \text{ or } =) b_m$$

$$\forall i \in \{1, 2, \ldots, n\}, x_i \in \mathbb{R}^+ (\text{or } \mathbb{N} \text{ or } \{0, 1\})$$

LP Compact/Matrix Form

A Linear Programming (LP) model, with n **decision variables** and m **constraints**, can be **compactly** formulated as follows:

$$\textbf{Min (or Max) } Z = c.x$$

subject to **(s.t.)**:

$$A.x \begin{matrix} \leq \\ \geq \\ = \end{matrix} b$$

$$\forall x_i \in x, x_i \in \mathbb{R}^+ (\text{or } \mathbb{N} \text{ or } \{0,1\})$$

With: $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$, $c = [c_1, c_2, \dots, c_n]$, $b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$, $A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$,

where x is a vector of n **continuous (or integer, or mixed) decision variables**, and c and b (resp. A) are constant vectors (resp. matrix) of coefficients.

First Simple LP Model (1/3)

A given company synthesizes two products $Prod_1$ and $Prod_2$ based on two kinds of raw materials M_1 and M_2. The below table presents the daily available raw materials for M_1 and M_2, and for each product $Prod_i$ the used amount of raw materials and the profit. The objective consists in finding the most profitable amounts of product mix $Prod_1$ and $Prod_2$.

	Usage for $Prod_1$	**Usage for** $Prod_2$	**Material Availability**
M_1	6	4	**24**
M_2	1	2	**6**
Profit	**$5**	**$4**	

First Simple LP Model (2/3)

- **Decision variables:** Two unknown positive real variables X_1 and X_2 that determine, respectively, the amounts of $Prod_1$ and $Prod_2$. A negative output of a product is not possible, therefore variables X_1 and X_2 must be positive. $X_1, X_2 \in \mathbb{R}^+$
- **Objective function:** Maximize the total profit obtained by X_1 of $Prod_1$ and X_2 of $Prod_2$.
  $MaxZ = 5X_1 + 4X_2$
- **Constraints:**
  1. **Material M_1 availability:** Each unit of product $Prod_1$ produced requires 6 units of material M_1. Each unit of product $Prod_2$ produced requires 4 units of material M_1. Only 24 units of material M_1 are available. $6X_1 + 4X_2 \leq 24$
  2. **Material M_2 availability:** Each unit of product $Prod_1$ produced requires 1 unit of material M_2. Each unit of product $Prod_2$ produced requires 2 units of material M_2. Only 6 units of material M_2 are available. $1X_1 + 2X_2 \leq 6$
- **Feasible region:** The set of all solutions **satisfying all constraints simultaneously**.
- **Optimal Solution:** further than satisfying all constraints simultaneously, it should **provide the maximum value Z of the objective function**.

First Simple LP Model (3/3)

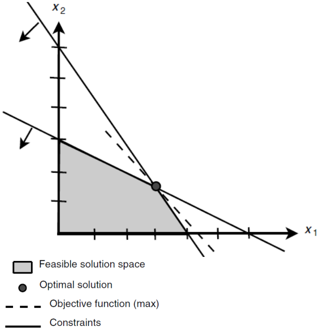
The model of this problem may be formulated as an LP mathematical program:

$$
\begin{aligned}
&Max Z = 5X_1 + 4X_2\\
&\textbf{(s.t.)}:\\
&\qquad 6X_1 + 4X_2 \leq 24\\
&\qquad 1X_1 + 2X_2 \leq 6\\
&\qquad X_1, X_2 \in \mathbb{R}^+
\end{aligned}
$$

$$x = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix},\ c = [5, 4],$$

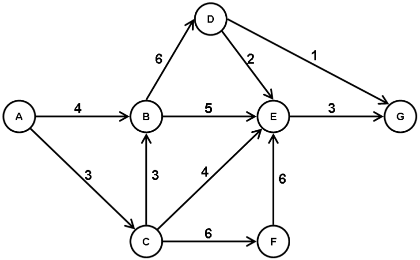
$$b = \begin{bmatrix} 24 \\ 6 \end{bmatrix},\ A = \begin{bmatrix} 6 & 4 \\ 1 & 2 \end{bmatrix}$$

- **The figure on the right illustrates the graphical interpretation of the model.**
- **Each constraint can be represented by a line.**
- **The objective function is an infinity of parallel lines.**
- **The optimum solution will always lie at an extreme point.**
- **The optimal solution is $(X_1 = 3, X_2 = 1.5)$ with a maximum profit of $Z = 21$.**



Computer Network (1/2)

Given an IP network where we consider only one direction of transmission indicated by arrows. The number on each link indicates the time taken by an IP packet (in seconds) to be transferred over that link. **What is the fastest path to forward an IP packet from router A to router G?**



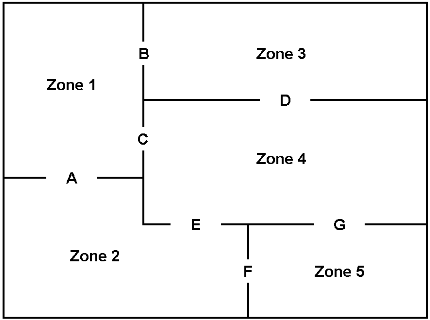
Computer Network (2/2)

- **Decision variables:** Unknown binary variables that determine for each link if it is taken or not by the packet when forwording it from A to G.
  $X_{AB}, X_{AC}, X_{BD}, X_{BE}, X_{CB}, X_{CE}, X_{CF}, X_{DE}, X_{DG}, X_{EG}, X_{FE} \in \{0, 1\}$
- **Objective function:** Minimize the total time of the path from A to G according to the taken links and their required times.
  $MinZ = 4X_{AB} + 3X_{AC} + 6X_{BD} + 5X_{BE} + 3X_{CB} + 4X_{CE} + 6X_{CF} + 2X_{DE} + X_{DG} + 3X_{EG} + 6X_{FE}$
- **Constraints:**
  1. **Starting point:** IP packet must start from router A. $X_{AB} + X_{AC} = 1$
  2. **Ending point:** IP packet must arrive to router G. $X_{DG} + X_{EG} = 1$
  3. **Flow conservation:** If IP packet enters to B, it must leave it. $X_{AB} + X_{CB} = X_{BD} + X_{BE}$. Same thing for router C, D, E and F. $X_{AC} = X_{CB} + X_{CE} + X_{CF}$, $X_{BD} = X_{DE} + X_{DG}$, $X_{BE} + X_{CE} + X_{DE} + X_{FE} = X_{EG}$, $X_{CF} = X_{FE}$
- **Feasible region:** The set of all solutions **satisfying all constraints simultaneously**.
- **Optimal Solution:** further than satisfying all constraints simultaneously, it should **provide the minimum value Z of the objective function**.

Mobile Phone Coverage (1/2)

A mobile phone company is newly installed in a country whose plan is presented in the figure beside. The transmitting antennas can be placed in sites A,B,...,G located on the common borders of the different areas of the country. An antenna placed on a given site can cover the two zones whose common borders shelter this site.

The company's goal is to ensure, at the lowest cost, the coverage of each zone with at least one antenna, while covering zone 4 with at least two antennas.



Mobile Phone Coverage (2/2)

- **Decision variables:** Unknown binary variables that determine for each site (from A,B,...,G) if it will contain an antenna or note. $X_A, X_B, X_C, X_D, X_E, X_F, X_G \in \{0, 1\}$
- **Objective function:** Minimize the total number of placed antennas in all sites. $MinZ = X_A + X_B + X_C + X_D + X_E + X_F + X_G$
- **Constraints:**
  1. **Zone 1 requirement:** Zone 1 must be covered by at least 1 antenna placed on one of its border. $X_A + X_B + X_C \geq 1$
  2. **Zone 2 requirement:** Same thing for Zone 2. $X_A + X_E + X_F \geq 1$
  3. **Zone 3 requirement:** Same thing for Zone 3. $X_B + X_D \geq 1$
  4. **Zone 5 requirement:** Same thing for Zone 5. $X_F + X_G \geq 1$
  5. **Zone 4 requirement:** Zone 4 must be covered by at least 2 antennas placed on one of its border. $X_C + X_D + X_E + X_G \geq 2$
- **Feasible region:** The set of all solutions **satisfying all constraints simultaneously**.
- **Optimal Solution:** further than satisfying all constraints simultaneously, it should **provide the minimum value Z of the objective function**.

Part-time Student-job Working Hours (1/2)

Mohamed is a college student who works two jobs on campus.

- He must work for **at least 5 hours per week at the library** and **2 hours per week as a tutor**, but he is **not allowed to work more than 20 hours per week** total.
- Mohamed gets **\$7 per hour at the library** and **\$10 per hour at tutoring** and he want to make at last \$170 during the current week.
- He **prefers working at the library** though, so he wants to have **at least as many library hours as tutoring hours**.

What is the minimum number of hours he can work at each job this week to meet his goals and preferences?

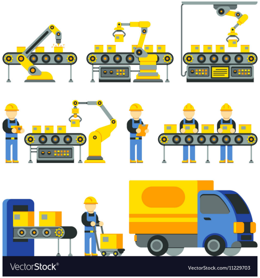
Part-time Student-job Working Hours (2/2)

- **Decision variables:** Unknown variables that determine the number of working hours at the library and the number of working hours at tutoring. $X_L, X_T \in \mathbb{N}$: number of working hours at library and at tutoring, respectively.
- **Objective function:** Minimize the total number of working hours for this week. $MinZ = X_L + X_T$
- **Constraints:**
  1. **Library requirement:** Mohamed must work at least 5 hours per week at library. $X_L \geq 5$
  2. **Tutoring requirement:** Mohamed must work at least 2 hours per week at tutoring. $X_T \geq 2$
  3. **Maximum number of working hours :** Mohamed is not allowed to work more than 20 hours per week. $X_L + X_T \leq 20$
  4. **Budget goal:** Mohamed need to make at least \$170 during the current week, while he gets \$7 per hour at the library and \$10 per hour at tutoring. $7X_L + 10X_T \geq 170$
  5. **Preferences:** Mohamed wants to have at least as many library hours as tutoring hours. $X_L \geq X_T$
- **Feasible region:** The set of all solutions **satisfying all constraints simultaneously**.
- **Optimal Solution:** further than satisfying all constraints simultaneously, it should **provide the minimum value Z of the objective function**.

Factory Production (1/2)

A factory produces two types of building materials.

- **Sale price: Product A = $140/ton, Product B=$160/ton.**
- During production, a **special ingredient X** is added.
- **Each ton of product A or B produced requires 2 , 4 cubic meters** of ingredient X, respectively. **Only 28 cubic meters** of ingredient X are available in production per week.
- **The worker** who produces the materials can work **up to 50 hours/week. The machine** producing the materials is able to construct a ton of product at a time, **while the process lasts 5 hours.**
- **The finished products** are stored in bins: **8 tons of product A and 6 tons of product B.**



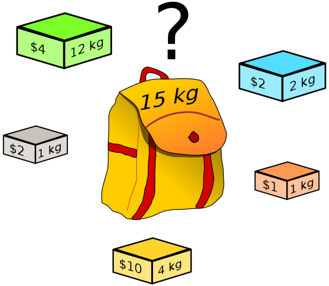
The purpose of **solving the problem** is to **determine the quantity of product A and of product B** that can be produced **every week** in order to achieve **maximization of the total weekly profit**.

Factory Production (2/2)

- **Decision variables:** Unknown variables that determine the quantity of product A and of product B to produce each week. $X_A, X_B \in \mathbb{R}^+$: number of weekly produced tons of product A and B, respectively.
- **Objective function:** Maximize the total profit per week. $MaxZ = 140X_A + 160X_B$
- **Constraints:**
  1. **Ingredient availability:** Each ton of product A produced requires 2 cubic meters of ingredient X. Each ton of product B produced requires 4 cubic meters of ingredient X. Only 28 cubic meters of ingredient X are available per week. $2X_A + 4X_B \leq 28$
  2. **Total Production time:** The worker who produces products A and B can work up to 50 hours/week. The machine that produces products A and B is able to construct a ton of product at a time, while the process lasts 5 hours. $5X_A + 5X_B \leq 50$
  3. **Storage availability for A:** The bins stored at maximum 8 tons of product A. $X_A \leq 8$
  4. **Storage availability for B:** The bins stored at maximum 6 tons of product B. $X_B \leq 6$
- **Feasible region:** The set of all solutions **satisfying all constraints simultaneously**.
- **Optimal Solution:** further than satisfying all constraints simultaneously, it should **provide the maximum value Z of the objective function**.

Knapsack Problem

The Knapsack Problem (KP) can be defined as follows. Given a set of N objects. Each object i has a specified weight w_i and a specified profit p_i. Given a capacity C, which is the maximum total weight of the knapsack. The KP consists in finding a subset of the objects whose total weight is at most equal to the knapsack capacity and whose total profit is maximized



$$\textbf{Max } Z = \sum_{i=1}^{n} p_i x_i$$

subject to **(s.t.)**:

$$\sum_{i=1}^{n} w_i x_i \leq C$$

$$\forall i \in \{1, 2, \ldots, n\}, x_i \in \{0, 1\})$$

Outline

Microsoft Excel Solver

- **Microsoft Excel Solver** is an add-in program in **Microsoft Excel** that can be used for optimization.
- It is able to solve many types of **linear programming problems** e.g., continuous, integer, and binary. It is user-friendly and easy to use.
- To use **Microsoft Excel Solver**, you need to define the **objective function**, **decision variables**, and **constraints** in an Excel worksheet. Then, you can use the Solver add-in to find the optimal solution.
- For a practical example of using Microsoft Excel Solver to solve an optimization problem, please refer to the following video tutorial: (**Click here to watch the video tutorial**)

Outline

1. Optimization Problems
2. Optimization Models
3. Linear Programming Model
4. Microsoft Excel Solver
5. Homework

Homework

1. Download the following pdf file that contains a set of small optimization problems. (Click to download)
2. Follow the instructions, model the problems mathematically, and solve them using Microsoft Excel Solver.
3. Submit:
   - The pdf file after writing the linear program for each problem.
   - Four Excel files, each one for a problem solved using Microsoft Excel Solver.

End of Chapter 6