DDA5002 Optimization
Lecture 1 Course Introduction
Optimization Basics
Linear Program Modeling

Yuang Chen

School of Data Scienc
The Chinese University of Hong Kong, Shenzhen

September 3, 2025

Outline

Outline

What is Operations Research?

- Operations Research: a **quantitative** approach to **decision making** based on the scientific method of problem solving.
- Operations Research is the scientific approach to execute decision making, which consists of:
  - The art of mathematical modeling of complex situations.
  - The science of the development of solution techniques used to solve these models.
  - The ability to effectively communicate the results to the decision maker.

History of Operations Research

- Operation research origins in World War II for military service - planning and scheduling problems for Air Force.
- Urgent need to allocate resources at efficient manner.
- British and US called large number of scientists from discipline were asked to do research on military operation.
- Developed effective method to locate radar (Britain Air Battle).
- Developed a better method to manage convoy and antisubmarine operation(North Atlantic).
- Developed a method to utilize resources efficiently (resource cost reduced one half)
- Invented a branch of operations research called linear programming in 1947 at the pentagon.

Development of Operations Research

- Success of OR in the war spurred interest in outside the military (business, industry and government)
- Two factors played a major for rapid growth of OR
  - Continuous contribution by scientist's to improve the techniques of OR
  - Computer Revolution

Operations Research Models

OR
Optimization
Stochastic System / Probability
Simulation

Stochastic Models

- Discrete-Time Markov Chains
- Continuous-Time Markov Chains
- Queuing Theory
- Stochastic Optimization (Markov Decision Process)
- Bayesian Optimization
- Simulation

Deterministic Models

- Linear Programming
- Integer Programming
- Convex Optimization
- Nonlinear Programming
- Combinatorial Optimization

Some Major Applications of OR

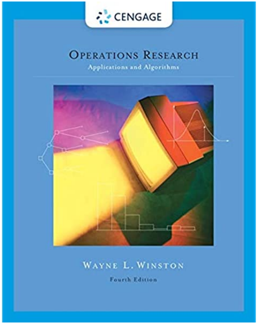
- Manufacturing and service sciences
- Supply chain management
- Policy modeling and public sector work
- Revenue management
- Financial engineering
- Game theory
- Transportation
- Healthcare
- Some people feel machine learning is an application of OR.

Terminology

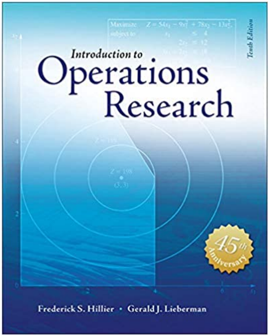
- The British/Europeans refer to “Operational Research”, the Americans to “Operations Research” - but both are often shortened to just "OR".
- Another term used for this field is “Management Science” ("MS"). In U.S. OR and MS are combined together to form "OR/MS" or "ORMS".
- Yet other terms sometimes used are “Industrial Engineering” ("IE") and “Decision Science” ("DS").
- The Institute for Operations Research and the Management Sciences (INFORMS) is an international society for practitioners in the fields of operations research, management science, and analytics.

Operations Research Textbooks

Operations Research: Applications and Algorithms by Wayne Winston



Introduction to Operations Research by Frederick S Hillier and Gerald J. Lieberman



INFORMS John von Neumann Theory Prize

- Awarded annually to individuals (or occasionally groups) for fundamental and sustained contributions to theory in operations research and management sciences.
- Recognized as the highest honor in the field, often referred to as the "Nobel Prize" of Operations Research.
- 6 out of 50 recipients have also won the Nobel Prize in Economics.
- ~~Three~~ recipients are affiliated with SDS/CUHKSZ:

Four
  - Stephen Boyd
  - Yurii Nesterov
  - Yinyu Ye
  - Jim Dai (see: `https://mp.weixin.qq.com/s/SXv3pvYjAFzuF9Y7SNoaqQ`)

CUHKSZ = OR University?

- All faculty members in the SDS at CUHKSZ are assigned to one of three areas: Operations Research, Statistics, or Computer Science.
- Guess why many people like to say: CUHKSZ is known as the OR University.

Outline

Optimization

- Optimization is the core mathematical engine of OR.
- It deals with decision making problem by formulating and solving mathematical problems.
- A typical optimization model involves minimizing or maximizing an objective function subject to constraints.
- Example: How do I get to classroom in the morning?
  - Alternatives: Possible decisions: driving, taking the bus, taking a taxi, walking, biking, or using skateboard
  - Goal: Objective: minimize time, maximize happiness, minimize cost, minimize social contacts, minimize carbon footprint
  - Restrictions: Constraints: budget limitation, time limitation, traffic consideration, route restriction

Summary of Optimization Problem

- Components of optimization:
  - Decision
  - Objective
  - Constraints
- Optimization concerns choosing a *decision* (or decisions) to *optimize* certain *objectives* while subject to certain *constraints*
- Optimize could mean *maximize* or *minimize* depending on the problem context.

Why Optimization?

- "Optimization" comes from the same root as "optimal", which means best. The purpose of optimization is to achieve the "best" to a set of prioritized criteria or constraints.
- When you optimize something, you are "making it best". When you make a decision, you are optimizing. Every decision-making question is essentially an optimization problem.
- Optimization problems underlie nearly everything we do in real life. A few examples: manufacturing, production, inventory control, transportation, scheduling, network flow, finance, energy system, mechanics, economics, optimal control, marketing, policy making.

Who Uses Optimization?

Why Study Optimization?

- Career development: job in data science/machine learning/AI requires basic knowledge in optimization, advanced machine learning scientist (i.e., computer vision) uses a lot of optimization.
- Research development: a lot of (actually almost all) science, engineering, and business/financial research involve different levels of optimization models.
- Until the end, all machine learning/AI models are optimization models. AI research is essentially solving an difficult optimization model.

Program vs. Optimization Problem

- A "program" or "mathematical program" is an optimization problem with a finite number of variables and constraints written out using explicit mathematical (algebraic) expressions.
- The word "program" / "programming" means "plan" / "planning."
- Early applications of optimization arose in planning resource allocations (especially in defense) and gave rise to "programming" to mean optimization (predates computer programming).
- We will use "program" / "programming" and "optimization problem" / "optimization" interchangeably.

Three Well-Known Optimization Solvers

- CPLEX: The CPLEX Optimizer was named for the simplex method as implemented in the C programming language at first. Today, it supports other types of mathematical optimization and offers interfaces other than C. CPLEX is actively developed by IBM.
- GuRoBi: Zonghao Gu, Edward Rothberg and Robert Bixby developed GuRoBi in 2008. Bixby was also the founder of CPLEX, while Rothberg and Gu led the CPLEX development team for nearly a decade.
- COPT: Cardinal Optimizer is a mathematical optimization solver for large-scale optimization problems. COPT supports solving all kinds of optimization and is among the fastest solvers for various optimization problems.

Outline

Course Information

6 - 7:20 class
7:20 - 7:30 break
7:30 - 8:50 class

- Instructor: Dr. Yuang Chen
- Class time: Wednesday 6 – 9 pm
- Location: Teaching B 103
- Office hours: Thursday 10 - 11 am or by appointments
- Office: Dao Yuan Building 520
- Email: ychen@cuhk.edu.cn

Tutorials

- Tutorial T01: Monday, 6 – 7pm, Teaching B 103.
- Tutorial T02: Monday, 7 – 8pm, Teaching B 103.
- Tutorial T03: Monday, 8 – 9pm, Teaching B 103.
- Tutorial mainly talks about coding on Python/MATLAB with COPT.
- Tutorials start from the second week!

Teaching Assistants

TA	Email	Office Hour	Location
Bingwei Zhang	119010422@link.cuhk.edu.cn	Wednesday 4–5pm	RB 302
Lisu Wang	224040346@link.cuhk.edu.cn	Wednesday 4:30–5:30pm	Zhixin 410
Yuqi Fei	225040475@link.cuhk.edu.cn	Friday 1–2pm	Zhixin 410
Qingwei Zhang	225040457@link.cuhk.edu.cn	Thursday 4–5pm	Zhixin 410
Ding Luo	224040043@link.cuhk.edu.cn	Tuesday 3–4pm	Daoyuan 501

What You Need to Know About DDA 5002

- DDA 5002 is a centralized course with 3 sessions this semester, taught by Prof. Haoxiang Yang (leading instructor), Prof. Junchi Yang, and myself.
- All students across the 3 sessions are graded together, so you are competing with everyone in the course.
- Master in AIR, data science, statistics, FE take this class together.
- DDA 5002 is fundamentally a mathematics course. The course assumes prior knowledge of calculus and linear algebra. Because of the accelerated pace, there will be little time for review, you are expected to master background material.
- There is no official textbook. You should primarily rely on the lecture slides. Reference books are available, but their notations, depth, difficulty, and logic differ from our course.

What This Course Will Study?

We study and solve many important optimization problems (linear program, integer program, convex optimization, nonlinear optimization, and unconstrained optimization).

- Modeling of optimization models
- Structures and properties of optimization models
- Algorithms to solve optimization models

Problem $\rightarrow$ Modeling $\rightarrow$ Structure/Properties $\rightarrow$ Algorithms

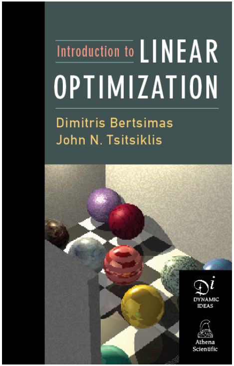
Course Pre-requisites

- The class involves many real-world applications, but the class content is a **mathematical class**.
- **You need fundamental knowledge in calculus and linear algebra**. If you are not familiar with calculus (taking derivatives) and linear algebra (equations in matrix form), the class will be difficult for you!

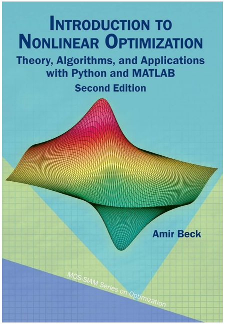
Reference Books

linear

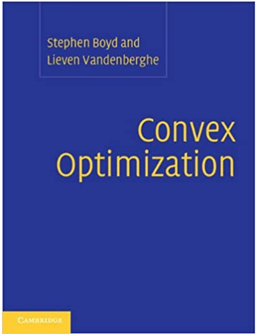
nonlinear



Introduction to Linear Optimization by D. Bertsimas and J. Tsitsiklis



Introduction to Nonlinear Optimization. Theory, Algorithms, and Applications with MATLAB by Amir Beck



Convex Optimization by S. Boyd and L. Vandenberghe

Grading

- Homework: 30%
  - 6 HWs in the semester.
  - **Late homework submission will be not be accpeted!**
- Midterm: 35% (Time: Nov 2, Sunday, 10–12 ~~am~~)
- Final 35% (Time TBA)

Pm

10am - 12Pm

Topics Covered

1. Optimization Basics
2. ~~General Modeling~~
3. Linear Program Modeling
4. Geometry of LP
5. Simplex Method
6. LP Duality
7. Sensitivity Analysis
8. Integer Program Modeling and Properties
9. Branch-and-Bound Alogirthm
10. Optimality Conditions
11. Lagrangian Dual and KKT Conditions
12. Convexity
13. Algorithms (Gradient Descent, Newton, Projected Gradient)

LP

IP

Midtem

non Linear

Outline

Optimization Generic Formulation

Mathematically, an optimization problem is usually represented as:

Generic Formulation

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in X \end{array}$$

max. / min. f(x)
s.t. x ∈ X

- x: decision variable or optimization variable
- $f(\cdot)$: objective function
- X: feasible region/set (constraints)
- $x \in X$: a feasible solution (satisfies all constraints)
- Sometimes, we express the problem using the abstract format: $\min\{f(x) : \ x \in X\}$.

$$\min_{x \in X} f(x)$$

$$\min_{x} \; f(x) \quad \text{s.t.} \quad x \in X$$

$$\min_{x} \{ f(x) : x \in X \}$$

decisions

Mathematical Formulation

Mathematical Formulation

Constraints

$$
\begin{aligned}
\text{minimize} \quad & f(\boldsymbol{x}) \\
\text{subject to} \quad & g_i(\boldsymbol{x}) \le 0, \quad \forall i = 1, 2, ..., m \\
& h_j(\boldsymbol{x}) = 0, \quad \forall j = 1, 2, ..., p
\end{aligned}
$$

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

- Optimization variables: $\boldsymbol{x} = (x_1, x_2, ..., x_n) \in \mathbb{R}^n$
- Objective function: $f \colon \mathbb{R}^n \to \mathbb{R}$ Scalar function
- Inequality constraints functions: $g_i \colon \mathbb{R}^n \to \mathbb{R}$, $j = 1, 2, ..., m$
- Equality constraints functions: $h_j \colon \mathbb{R}^n \to \mathbb{R}$, $j = 1, 2, ..., p$
- Sometimes, we write feasible region as $X = \{\boldsymbol{x} : \boldsymbol{g}(\boldsymbol{x}) \le 0,\ \boldsymbol{h}(\boldsymbol{x}) = 0\}$, where

$$\boldsymbol{g}(\boldsymbol{x}) = \begin{bmatrix} g_1(\boldsymbol{x}) \\ \vdots \\ g_m(\boldsymbol{x}) \end{bmatrix} \quad \text{and} \quad \boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} h_1(\boldsymbol{x}) \\ \vdots \\ h_p(\boldsymbol{x}) \end{bmatrix}$$

Optimization Problem

Mathematical Formulation

$$\begin{aligned} \text{minimize} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & g_i(\boldsymbol{x}) \leq 0, \quad \forall i = 1, 2, ..., m \\ & h_j(\boldsymbol{x}) = 0, \quad \forall j = 1, 2, ..., p \end{aligned}$$

Our goal is to find the optimal solution $\boldsymbol{x}^*$ such that the objective function $f(\boldsymbol{x}^*)$ is the smallest among all $\boldsymbol{x}$ vectors that satisfy the constraints. We call $f(\boldsymbol{x}^*)$ the optimal objective function value (optimal value).

Remark

- The problem may be infeasible: you cannot find any vectors that satisfy all constraints.
- The optimal solution can be more than one.
- In our class, we don't allow strict inequality constraints.

X

$g(x) < 0$

$g(x) > 0$

Terminologies Summary

- Feasible solution (point): a decision that satisfies all constraints ($\boldsymbol{x} \in X$)
- Feasible region (set): the set of all feasible solutions (X).
- Optimal solution: a feasible solution that attains an objective value that is as good as any other feasible point ($\boldsymbol{x}^*$).
- Optimal value: The objective value of any optimal solution ($f(\boldsymbol{x}^*)$).

Minimization vs Maximization

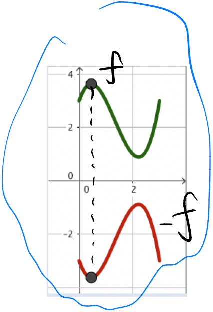
Without loss of generality, it is sufficient to consider a minimization objective since:

Optimal Value $= \max_x\{f(x) : x \in X\} \equiv - \min_x\{-f(x) : x \in X\}$

$\min_x\{f(x) : x \in X\} = - \max_x\{-f(x) : x \in X\}$

Example:

$$\max\{4 - x^2 + (x-1)^3 : 0 \le x \le 3\}$$
$$\equiv -\min\{-4 + x^2 - (x-1)^3 : 0 \le x \le 3\}$$



Thus, to develop the theory we will only consider minimization problems. When solving problems, we can use the actual min or max objective as needed.

Definition of argmin and argmax

- The **argmin** and **argmax** are operations used to find the optimal solutions where the problem reaches its minimum or maximum.
- **Definition of** `argmin`**:**

optimal solution $=$ $$\arg\min_{x \in X} f(x) = \{x^* \in X \mid f(x^*) \le f(x) \ \ \forall x \in X\}$$

- **Definition of** `argmax`**:**

$$\arg\max_{x \in X} f(x) = \{x^* \in X \mid f(x^*) \ge f(x) \ \ \forall x \in X\}$$

- **Example:**
  - For $f(x) = (x-2)^2$ over $X = \mathbb{R}$:

$$\min_{x} (x-2)^2 = 0$$

$$\arg\min_{x \in \mathbb{R}} f(x) = \{2\}.$$

In-class Exercise

Compare the following optimization problems

1. $\max \; f(x)$
2. $\min \; f(x)$
3. $\max \; -f(x)$
4. $\min \; -f(x)$
5. $-\max \; f(x)$
6. $-\min \; f(x)$
7. $-\max \; -f(x)$
8. $-\min \; -f(x)$

- How are the optimal objective values related among these problems?
- How are the optimal solutions related among these problems?

$Y_1 = \min. f(x)$ $\quad$ $X_1 = \arg\min. f(x)$

$Y_2 = \min. -f(x)$ $\quad$ $X_2 = \arg\min. -f(x)$

$Y_3 = \max. f(x)$ $\quad$ $X_3 = \arg\max. f(x)$

$Y_4 = \max. -f(x)$ $\quad$ $X_4 = \arg\max -f(x)$

$Y_5 = -\min. f(x)$ $\quad$ $X_5 = -\arg\min. f(x)$

$Y_6 = -\min. -f(x)$ $\quad$ $X_6 = -\arg\min -f(x)$

$Y_7 = -\max. f(x)$ $\quad$ $X_7 = -\arg\max f(x)$

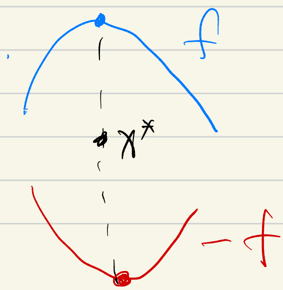
$Y_8 = -\max. -f(x)$ $\quad$ $X_8 = -\arg\max -f(x)$

$Y_1 = -Y_5 = Y_8 = -Y_4$ $\quad$ $X_1 = -X_5 = X_4 = -X_8$

$Y_2 = -Y_6 = Y_7 = -Y_3$ $\quad$ $X_2 = -X_6 = X_3 = -X_7$

$$\arg\min. f(x) = \arg\max. -f(x)$$

$$\arg\max. f(x) = \arg\min. -f(x)$$



Outline

Classifications

- Unconstrained optimization: No constraints. Otherwise constrained optimization.
- Linear optimization (LP): Constraints and objective function are linear in the decision variables.
- Nonlinear optimization (NLP): Either some of the constraints or the objective function is nonlinear.
- Continuous optimization: All decision variables take continuous values. ↳ default
- Integer/Discrete optimization (IP): Some of the decision variables have to be integers or discrete.

Remarks

- Sometimes, an NLP can be equivalently transformed to an LP.
- Sometimes, an IP can be equivalently transformed to a continuous optimization problem.

Linear optimization is the most well-studied and the easiest optimization problem.

- Nonlinear optimization and integer optimization could be significantly harder than LP.
- Therefore, in many cases, people strive to find LP formulations for problems.

In the first half of the semester, we will focus on linear optimization, then we will discuss integer and nonlinear optimization in the second half of the semester.

Outline

Infeasible Problem

Feasible $X \neq \emptyset$ $\{x : x \leq 1\}$
Infeasible $X = \emptyset$ $\{x : x \leq 1, x \geq 2\}$

Mathematical formulation

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x \in X \end{aligned}$$

- Any $x \in X$ is a feasible solution of the optimization problem.
- If $X = \phi$; then no feasible solutions exist, and the problem is said to be infeasible.
- The problem $\min\{3x + 2y : \underbrace{x + y \leq 1, x \geq 2, y \geq 2}_{\emptyset}\}$ is infeasible.

Unbounded Problem

Mathematical formulation

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x \in X \end{aligned}$$

- The optimization problem is unbounded, if there are feasible solutions with arbitrarily small objective values (for minimization problem).
- Formally, the problem is unbounded if there exists a sequence of feasible solutions $\{x^i\} \in X$ such that $\lim_{i \to \infty} f(x^i) = -\infty$.
- An unbounded problem must be feasible.
- The problem $\min\{x : \underline{x \leq 1}\}$ is unbounded.

Optimal Solution Exists

Mathematical formulation

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x \in X \end{aligned}$$

- A feasible solution x^* is an **optimal solution** of the optimization problem if

$$f(x^*) \leq f(x) \quad \forall x \in X$$

- The objective value corresponding to an optimal solution (if it exists) is called the **optimal (objective function) value** of the optimization problem.

$x^* = 1$

- The problem $\min\{x : x \geq 1\}$ has one unique solution. The problem $\min\{x : x \geq 1, y \leq 2\}$ has infinite number of optimal solutions.

$x^* = 1,\ y^* \leq 2$

For an OPT Problem, it can have

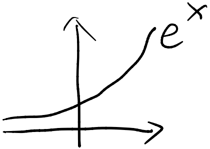
$\underline{0 - \infty}$ optimal solns.

For an OPT Problem with at least one optimal soln, it can have $\underline{\quad 1 \quad}$ optimal obj fun value.

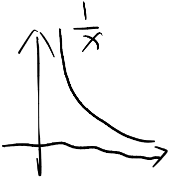
e.g. $x^* \neq x^{**} \neq x^{***}$: optimal solns

$f(x^*) = f(x^{**}) = f(x^{***})$: optimal value.

Optimal Solution Cannot be Achieved/Attained



- $\min\{e^x : x \in \mathbb{R}\}$
- $\min\{\frac{1}{x} : x \geq 0\}$



Feasible and Bounded but optimal soln is not attained.

Four Outcomes of Optimization Problem

Mathematical Formulation

Constraint : g(x)<0

not allowed

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x \in X \end{aligned}$$

1. Infeasible: $X = \phi$
2. Unbounded: $\exists \{x^i\} \in X,$ s.t. $f(x^i) \to -\infty$
3. Feasible and bounded but the ~~minimizer~~ optimal soln is not achieved (attained)
4. An optimal solution x^* exists

Existence of Optimal Solutions: Weierstrass Theorem

Definition

A function f is **continuous** if for all convergent sequences $\{x^i\} \subseteq dom(f) : \lim_{i\to\infty} x^i = x^0$ such that $\lim_{i\to\infty} f(x^i) = f(x^0)$.
A set X is **closed** if for all convergent sequences $\{x^i\} \subseteq X$ such that $\lim_{i\to\infty} x^i = x^0 \in X$.
A set X is **bounded** if $\exists M > 0, ||x|| \leq M, \forall x \in X$.

Weierstrass Theorem

For an optimization problem, if the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is continuous, and the feasible region $X \in \mathbb{R}^n$ is nonempty, closed, bounded, then the problem has an optimal solution.

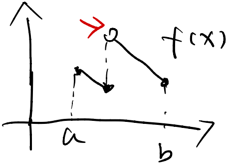
1 2 3 4

1 + 2 + 3 + 4 ⇒ optimal soln exists

True or False Exercise

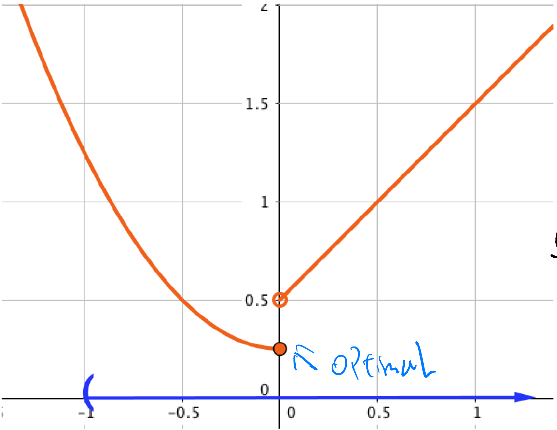
min. x

s.t. $x \geq 0$



- The problem: $\max\{f(x) : x \in [a, b]\}$ has an optimal solution. F
- Any optimization problem whose feasible region is unbounded cannot have an optimal solution. F
- For an optimization problem, if the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is not continuous, and the feasible region $X \in \mathbb{R}^n$ is nonempty, open, unbounded, then the problem has no optimal solution. F

Sufficient But Not Necessary



$f(x)$

min. $f(x)$

s.t. $x > -1$

Infimum (inf) in Optimization

- The infimum of a function $f(x)$ over a feasible region X is the greatest lower bound of $f(x)$.
- It represents the smallest value $f(x)$ can approach, but not necessarily attain.
- Formally:

$$\inf_{x \in X} f(x) = \sup\{y \in \mathbb{R} : f(x) \geq y, \forall x \in X\}.$$

Relationship to Minimum:

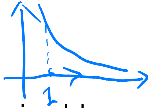
- If $f(x)$ attains its minimum in X, then:

$$\inf_{x \in X} f(x) = \min_{x \in X} f(x).$$

- Otherwise, $\inf_{x \in X} f(x)$ is strictly less than the values $f(x)$ achieves.

Example:

- For $f(x) = \frac{1}{x}$ over $X = [1, \infty)$:



$$\inf_{x \in X} f(x) = 0 \quad \text{(not attained by any } x \in X\text{)}.$$

Supremum (sup) in Optimization

- The `supremum` of a function $f(x)$ over a feasible region X is the least upper bound of $f(x)$.
- It represents the largest value $f(x)$ can approach, but not necessarily attain.
- Formally:

$$\sup_{x \in X} f(x) = \inf\{y \in \mathbb{R} : f(x) \leq y, \forall x \in X\}.$$

Relationship to Maximum:

- If $f(x)$ attains its maximum in X, then:

$$\sup_{x \in X} f(x) = \max_{x \in X} f(x).$$

- Otherwise, $\sup_{x \in X} f(x)$ is strictly greater than the values $f(x)$ achieves.

Example:

- For $f(x) = -\frac{1}{x}$ over $X = [1, \infty)$:

$$\sup f(x) = 0 \quad \text{(not attained by any } x \in X\text{)}.$$

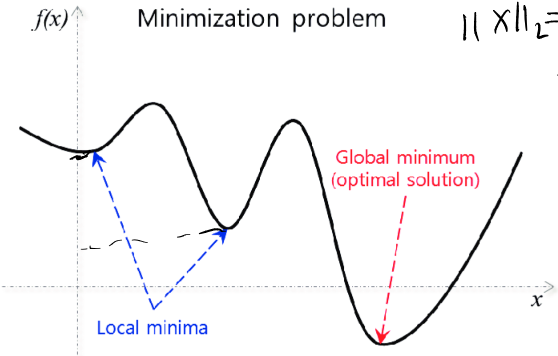
Global v.s. Local Optimal Solution

Definition

$x^* \in X$ is a **global optimal solution** if $f(x^*) \leq f(x)\ \forall x \in X$.
$\hat{x} \in X$ is a **local optimal solution** if
$\exists \epsilon > 0$ s.t. $f(\hat{x}) \leq f(x)\ \forall x \in X \cap B(\hat{x}, \epsilon)$ where
$B(\hat{x}, \epsilon) = \{x : ||x - \hat{x}|| \leq \epsilon\}$ is the ϵ-neighborhood of $\hat{x}$.

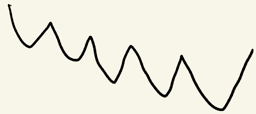
$$||x||_2 = \sqrt{x_1^2 + \cdots + x_n^2}$$

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



Local v.s. Global Optimal Solutions Remarks

- Every global optimal solution is a local optimal solution, but not vice versa.
- The objective function value at different local optimal solutions may be different.
- The objective function value at all global optimal solutions must be the same.
- Typical optimization algorithms are designed to find local optimal solutions (at best).
- We want to find a type of optimization problems that all local optimal solutions are global $\Rightarrow$ Convex Optimization



For an OPT problem with at least one global optimal solution, it can have

≥ 1 local optimal soln,

≥ 1 local optimal value,

≥ 1 global optimal soln,

$= 1$ global optimal value.

Strict Local/Global Optimal Solutions

- x^* is a **local optimal solution** if $x^* \in X$ and there exists $\varepsilon > 0$ such that
$$f(x) \geq f(x^*) \quad \text{for all } x \in X \cap B_\varepsilon(x^*).$$
- x^* is a **strict local optimal solution** if $x^* \in X$ and there exists $\varepsilon > 0$ such that
$$f(x) > f(x^*) \quad \text{for all } x \in (X \cap B_\varepsilon(x^*)) \setminus \{x^*\}.$$
- x^* is a **global optimal solution** if $x^* \in X$ and
$$f(x) \geq f(x^*) \quad \text{for all } x \in X.$$
- x^* is a **strict global optimal solution** if $x^* \in X$ and
$$f(x) > f(x^*) \quad \text{for all } x \in X \setminus \{x^*\}.$$

Minimum, Minimizer, Minima and Related Concepts

- **Minimum**: The smallest value of the objective function.

  ↳ optimal value

$$\min_{x \in X} f(x)$$

  It is a *value*, not a solution (x).

- **Minimizer**: A point $x^* \in X$ where the objective function attains its minimum value.

  ↳ optimal sols

$$f(x^*) = \min_{x \in X} f(x)$$

- **Minima**: Plural form of minimum. "Minima" should mean multiple minimum values. However, in optimization, "minima" commonly refers to multiple (local or global) minimizers where the objective function attains a minimum value. → set of optimal sols
- Concepts of **maximum**, **maximizer**, and **maxima** are defined similarly for the largest value of the function.

Outline

Ingredients of a linear program

A Linear program (or a linear optimization model) is composed of:

- Variables:

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$$ Continuous

- A linear objective function:

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} c_i x_i = \boldsymbol{c}^\top \boldsymbol{x}. = C_1X_1 + C_2X_2 + \cdots + C_nX_n$$

- Linear constraints:

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1$$
$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \geq b_2$$
$$a_{31}x_1 + a_{32}x_2 + \cdots + a_{3n}x_n = b_3$$

$\Rightarrow$

$$\boldsymbol{a}_1^\top \boldsymbol{x} \leq b_1$$
$$\boldsymbol{a}_2^\top \boldsymbol{x} \geq b_2$$
$$\boldsymbol{a}_3^\top \boldsymbol{x} = b_3$$

→ Scalar

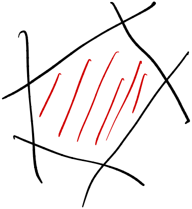
$$a_1 = \begin{bmatrix} a_{11} \\ a_{12} \\ \vdots \\ a_{1n} \end{bmatrix} \quad a_2 = \begin{bmatrix} a_{21} \\ \vdots \\ a_{2n} \end{bmatrix} \quad a_3 = \begin{bmatrix} a_{31} \\ \vdots \\ a_{3n} \end{bmatrix}$$

Linear Program

$$
\begin{aligned}
\min \quad & \boldsymbol{c}^\top \boldsymbol{x} \\
\text{s.t.} \quad & \boldsymbol{a}_i^\top \boldsymbol{x} \geq \boldsymbol{b}_i, \quad i = 1, 2, .., m
\end{aligned}
$$

- Linear objective function
- n continuous decision variables
- m linear constraints
- Optimize a linear function over a polyhedron
- Matrix form

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



$$
\begin{aligned}
\min \quad & \boldsymbol{c}^\top \boldsymbol{x} \\
\text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}
\end{aligned}
$$

LP Example in Matrix

$$
\begin{aligned}
\min \quad & 3x_1 + x_2 \\
\text{s.t.} \quad & x_1 + 2x_2 \geq 2 \\
& 2x_1 + x_2 \geq 3 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

In the matrix form, we can define

$$
\boldsymbol{c} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \quad \boldsymbol{A} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \quad \boldsymbol{b} = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
$$

Matrix form

$$
\begin{aligned}
\min \quad & \boldsymbol{c}^\top \boldsymbol{x} \\
\text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\
& \boldsymbol{x} \geq \boldsymbol{0}
\end{aligned}
$$

Why LP?

- Many important optimization problems can be modeled or approximated as LPs.
- Elegant and essentially complete mathematical theory.
- Powerful algorithms for solving very large scale LPs.
- LP is a key subroutine in methods for many general optimization problems.
- **LP has a very wide applications (almost all areas): transportation, manufacturing, long term planning, financial investment, revenue management, supply chain, medicine, healthcare, telecommunication, and more.**

Outline

Production Problem - Data Specific

A company needs to decide the amount of each product to produce, in order to maximize the profit.

	Steel	Iron	Copper	Profit
Alloy 1	1	0	1	\$1
Alloy 2	0	2	1	\$2
Resources	100	200	150	

Step 1: Decision variables

x_1: # of Alloy 1 to Produce

x_2: # of Alloy 2 to Produce

Step 2: objective function

$$\max.\quad x_1 + 2x_2$$

Step 3: Constraints

$$\begin{aligned} \text{s.t.}\quad x_1 \quad\quad &\le 100 \\ 2x_2 &\le 200 \\ x_1 + x_2 &\le 150 \end{aligned}$$

Step 4: Varible types

$$x_1, x_2 \ge 0$$

Production Problem - Data Independent

- n products, m raw materials
- c_j: profit of product j
- b_i: available units of material i
- a_{ij}: number of units required of material i in producing product j

Goal: decide the optimal quantity for producing each product with largest profit

Decision Variable:

x_j: # of Product j to Produce

objective:

$$\max. \sum_{j=1}^{n} c_j x_j$$

Constraints:

$$\text{s.t.} \quad a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \le b_m$$

Variable type:

$$x_j \ge 0, \quad \forall j = 1, \dots, n$$

$$\left\{ \begin{array}{l} \sum_{j=1}^{n} a_{1j} x_j \le b_1 \\ \vdots \\ \sum_{j=1}^{n} a_{mj} x_j \le b_m \end{array} \right. \Rightarrow \underbrace{\sum_{j=1}^{n} a_{ij} x_j \le b_i, \ \forall i = 1, \dots, m}_{}$$

↳ budget limitation

Diet Problem - Data Specific

Suppose we have five types of food and two critical nutrient requirements to satisfy, and we want to do so with the minimum cost.

Critical nutrient requirements:

- Nutrient #1: 21 units
- Nutrient #2: 12 units

Food type	**Nutrient #1** (units/kg)	**Nutrient #2** (units/kg)	**Cost ($/kg)**
1	2	0	20
2	0	1	10
3	3	2	31
4	1	2	11
5	2	1	12

x_i : # of feed i to eat, $i = 1, \cdots, 5$

$$\min.\ 20x_1 + 17x_2 + 31x_3 + 11x_4 + 12x_5$$

$$\text{s.t.}\quad 2x_1 + 3x_3 + 1x_4 + 2x_5 \geq 21$$

$$x_2 + 2x_3 + 2x_4 + x_5 \geq 12$$

$$x_1, x_2, \cdots, x_5 \geq 0$$

Diet Problem - Data Independent

$j = 1, \cdots, n$

$i = 1, \cdots, m$

- c_j: unit cost of food type j
- b_i: requirement for nutrient i
- a_{ij}: density of nutrient i in food type j

Goal: decide the optimal quantity of food to eat to satisfy nutrient requirements with lowest cost

x_j: # of food j to eat

$$\min. \sum_{j=1}^{n} c_j x_j$$

$$\text{s.t.} \quad \begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\geq b_1 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &\geq b_m \end{aligned}$$

$$x_j \geq 0, \quad \forall j = 1, \cdots, n$$

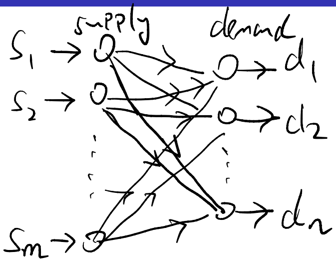
$$\underbrace{\sum_{j=1}^{n} a_{ij} x_j \geq b_i, \quad \forall i = 1, \cdots, m}_{\text{demand requirement}}$$

Golden Rule in Formulating Optimization Problems

The golden rule: Identify and formulate the three components:

- Decision $\rightsquigarrow$ Decision variables $\boldsymbol{x}$.
- Objective $\rightsquigarrow$ Objective function $f(\boldsymbol{x})$.
- Constraints $\rightsquigarrow$ Constraint functions: inequality / equality constraints.

Transportation



- m plants, n warehouses
- s_i: supply of i^{th} plant $i = 1, .., m$
- d_j: demand of j^{th} warehouse $j = 1, ...n$
- c_{ij}: cost of transportation from i to j

Goal: decide the optimal units of transportation from supply plant i to warehouse j with the lowest cost

x_{ij} : # items transport from i to j

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

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} \le S_i, \quad \forall i = 1, \cdots, m$$

budget

$$\sum_{i=1}^{m} x_{ij} \ge d_j, \quad \forall j = 1, \cdots, n$$

demand

$$x_{ij} \ge 0$$

Sorting

- Given n numbers: $c_1, c_2, ..., c_n$
- Order statistic: $c_{(1)} \leq c_{(2)} \leq ... \leq c_{(n)}$

Goal: sort the numbers in a nondecreasing order

$$2 = \left.\begin{array}{ll} \min. & 2x_1 + 3x_2 \\ \text{s.t.} & x_1 + x_2 = 1 \\ & 0 \le x_1, x_2 \le 1 \end{array}\right\} \Rightarrow x_1^* = 1,\ x_2^* = 0$$

$$5 = \left.\begin{array}{ll} \min. & 2x_1 + 3x_2 + 4x_3 \\ \text{s.t.} & x_1 + x_2 + x_3 = 2 \\ & 0 \le x_1, x_2, x_3 \le 1 \end{array}\right\} \Rightarrow x_1^* = 1,\ x_2^* = 1,\ x_3^* = 0$$

$$\begin{array}{ll} \min. & c_1 x_1 + c_2 x_2 + \cdots + c_n x_n = \sum_{i=1}^{n} c_i x_i \\ \text{s.t.} & \sum_{i=1}^{n} x_i = 1, 2, 3, \cdots, n-1 \\ & 0 \le x_i \le 1, \ \forall i = 1, \cdots, n \end{array}$$

Scheduling

- Hospital wants to make weekly nightshift for its nurses
- D_j: demand for nurses on day j, $j = 1, .., 7$
- Every nurse works 5 days in a row

Goal: hire minimum number of nurses to satisfy all demands

x_j : # nurses working on day j

$$x_j \geq d_j \quad , \forall j = 1, \cdots, 7$$

e.g. $x_1 = 20$

$x_2 = 30$

bad decision variable

Airline Revenue Management

- Before deregulation, carriers were only allowed to fly certain routes (e.g., Northwest, Eastern, Southwest). Fares were determined by the Civil Aeronautics Board (CAB) based on mileage and other costs (CAB no longer exists).
- After deregulation (1978), any carrier can fly anywhere, fares are determined by the carrier and market dynamics.
- Economics of the Airline Industry:
  - Huge sunk and fixed costs. Very low variable costs per passenger (e.g., $10 or less).
  - Highly competitive market environment.
  - Near-perfect information and negligible cost of information.
  - Highly perishable inventory (e.g., unsold seats lose value after departure).
  - Result: Airlines implement multiple fare structures. Dynamic pricing strategies to maximize revenue.

Ticketing Problem

- n routes
- 2 classes: Q class, Y class
- Revenue: r_i^Q, r_i^Y on route $i = 1, ..n$
- Capacities: C_i on route $i = 1, ..n$
- Expected demand: D_i^Q, D_i^Y on route $i = 1, ..n$

Goal: find open tickets in each class on each route with maximized revenue

Capacity Expansion

- D_t: forecast demand for electricity at year t
- E_t: existing capacity (in oil) available at t
- c_t: cost to construct 1 MW power using coal capacity
- n_t: cost to construct 1MW using nuclear capacity
- No more than 20% nuclear
- Coal plants last 20 years
- Nuclear plants last 15 years
- Consider a T-year time horizon

Goal: find the optimal coal and nuclear capacity for each year with lowest total costs

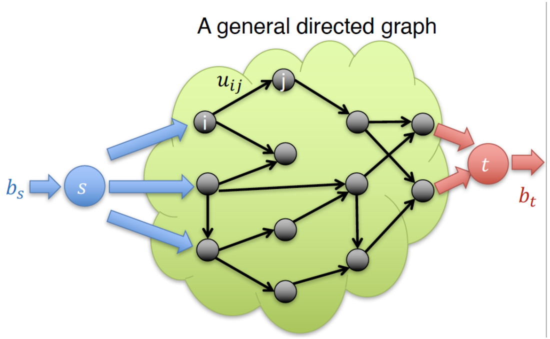
Graph

- Undirected Graph $G = \{N, E\}$: a set of N nodes and a set of E undirected edges (unordered pair (i, j))
- Directed Graph $G = \{N, A\}$: a set of N nodes and a set of A directed arc (ordered pair (i, j))

Min-cost Network Flow Problem

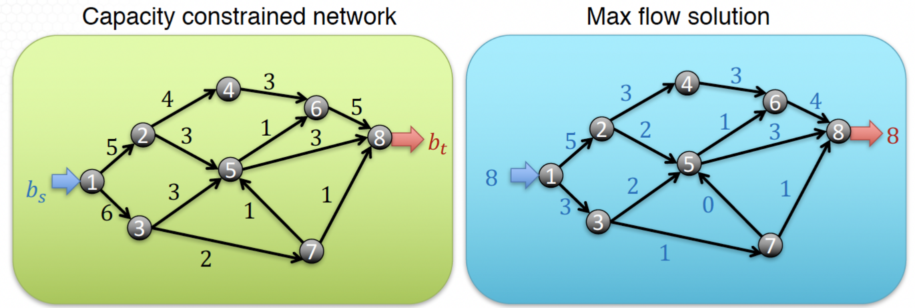
- A directed network with at least one supply node ($b_i > 0$), at least one demand node ($b_i < 0$), and transshipment nodes ($b_i = 0$).
- Flow through an arc (ij) is allowed only in the direction indicated by the arrowhead, where the maximum amount of flow is given by the capacity of that arc (u_{ij}).
- The cost of the flow through each arc is proportional to the amount of that flow, where the cost per unit flow is known (c_{ij}).
- The objective is to minimize the total cost of sending the available supply through the network to satisfy the given demand.

Maximum Flow Problem

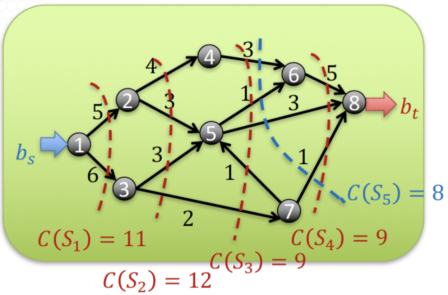


- The capacity of an edge (u_{ij}) is the maximum amount of flow that can pass through an edge.
- The sum of the flows entering a node must equal the sum of the flows exiting that node, except for the source (s) and the sink (t).
- Question: What's the largest supply b_s can be transported from source to sink through the network with limited arc capacity?

Concrete Example



Minimum Cut Problem



A s-t **cut** S is a subset of nodes
Such that $s \in S$ and $t \notin S$

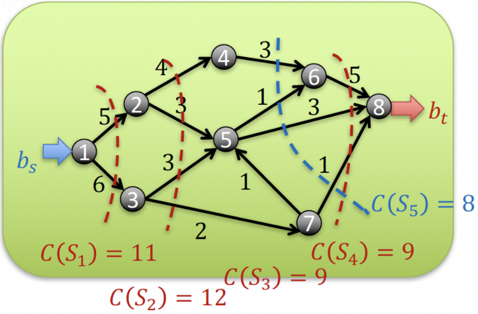
So a **cut** S is a separation of
Source node from target node

Capacity of a cut S is the total capacity
of arcs that cross from S to its complement
Denoted as $C(S) := \sum_{(i,j)\in A, i\in S, j\notin S} c_{ij}$

A million-dollar question:
Can you find a cut with minimum capacity?

Minimum Cut = Max Flow

Minimum cut = $C(S_5) = 8$ = Max Flow



Is this a Coincidence?

Not at all! There is a deep theory behind it – LP duality.

Max-flow and min-cut are two LPs dual to each other.

Intuitively, it makes sense too.