Advanced automation and control

Optimization module

Prof. Davide M. Raimondo

Dipartimento di Ingegneria Industriale e dell'Informazione

Università degli Studi di Pavia

davide.raimondo@unipv.it

Acknowledgment: thanks to Andrea Pozzi, Diego Locatelli, Giacomo Saccani for their help with some of the slides

Course schedule

Two modules

- one part on **optimization and graphs (Raimondo)**
- one part on nonlinear systems (Ferrara)

Lectures

- Thursday (9-11)
- Friday (16-18)

Laboratories

- Dates to be announced

Course schedule

Website: http://sisdin.unipv.it/labsisdin/teaching/courses/ails/files/ails.php

- course schedule, slides, etc.

Office hours: by appointment

Dipartimento di Ingegneria Industriale e dell'Informazione
Davide M. Raimondo: floor F (davide.raimondo@unipv.it)

Textbook and exams

Textbooks

- W. L. Winston & M. Venkataramanan "Introduction to Mathematical Programming: Applications and Algorithms", 4th ed., Duxbury Press, 2002. ISBN: 0-534-35964-7
- S. Boyd & L. Vandenberghe, "Convex Optimization", Cambridge University Press, 2004, ISBN 0521833787
- C. Vercellis "Ottimizzazione: Teoria, metodi, applicazioni", McGraw-Hill, 2008. ISBN: 9788838664427

Exams: Closed-books closed-notes written exam on all course topics

The part on optimization & graphs lasts 2 hours. *No graphic or programmable calculators are allowed.*

Date/time/room on the website of the Faculty of Engineering

Registration to exams: Through the university website.

Why optimization?

Useful in many contexts

- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Why optimization?

Useful in many contexts

- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Identification

Further details in the course of IMAD (Prof. De Nicolao)

Objective: describe a system **behaviour** through a **mathematical model** starting from **data**.

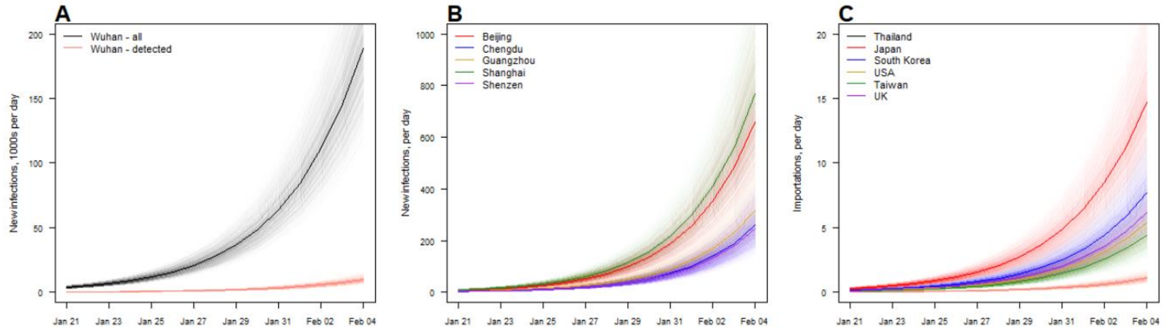
Identification

Further details in the course of IMAD (Prof. De Nicolao)

Objective: describe a system **behaviour** through a **mathematical model** starting from **data**.

Read JM et al. Novel coronavirus 2019-nCoV: early estimation of epidemiological parameters and epidemic predictions

Figure 3. Epidemic predictions for (**A**) Wuhan, (**B**) selected Chinese cities and (**C**) selected countries. Uncertainty in estimated model parameters is reflected by 500 repeated simulations with parameter values drawn randomly from the distribution of fit estimates.



Identification

Further details in the course of IMAD (Prof. De Nicolao)

Given

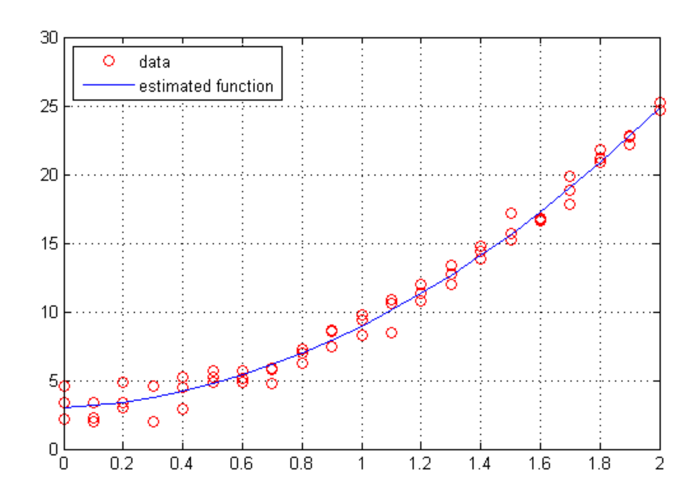
- a model structure, e.g. $y = \beta_1 + \beta_2 u + \beta_3 u^2$
- a set of input-output data $u_i, y_i, i = 1, \dots, m$

Find the value of parameters $\beta_1, \beta_2, \beta_3$ which provide the best match between model and experiments.

Since measurements are usually affected by noise, here we chose $m \gg 3$.

The problem can be stated as an **optimization:**

$$min_{\beta_1,\beta_2,\beta_3} \sum_{i=1}^{m} (\underbrace{y_i}_{\text{data}} - \underbrace{(\beta_1 + \beta_2 u_i + \beta_3 {u_i}^2)}_{\text{model}})^2$$



$$y_i = \beta_1 + \beta_2 u_i + \beta_3 u_i^2$$

Identification

Further details in the course of IMAD (Prof. De Nicolao)

Let generalize the previous problem. Consider now n parameters and use the vector notation

$$\beta = [\beta_1 \quad \beta_2 \quad \cdots \quad \beta_n]^T \qquad y = [y_1 \quad y_2 \quad \cdots \quad y_m]^T$$

The regressors (e.g. $1, u, u^2$) are predefined functions of the inputs. For each $i = 1, \dots, m$, we define X_i as the vector containing all the n regressors (e.g. X_i=[$1\ u_i\ {u_i}^2$]) and the matrix

$$\boldsymbol{X} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ X_{m1} & X_{m2} & \cdots & X_{mn} \end{bmatrix}$$

Then, we look for the parameters which provide the least square error $\boxed{\hat{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} ||\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}||^2}$

If prior knowledge is available, the problem above may be subject to constraints (e.g. $\boldsymbol{\beta} > 0$).

Identification: design of experiment

The Design of Experiment (DoE) procedure consists in designing an optimal input sequence (experiment) able to enhance the parameters identifiability and reduce the estimation error.

Process

Accurate estimation!

Optimal input choice

Identification: design of experiment

The optimal DOE is usually based on the **Fisher Information Matrix** $F^{\xi}(\phi) = S^{\xi}(\phi)^T C_y^{-1} S^{\xi}(\phi)$

- ϕ is the parameters vector
- ξ is the experiment (i.e. the input sequence)
- C_y is the covariance matrix of the measurements

The columns $i = 1, 2, \ldots, N_\phi$ of the sensitivity matrix $S^{\xi}(\phi)$ are given by

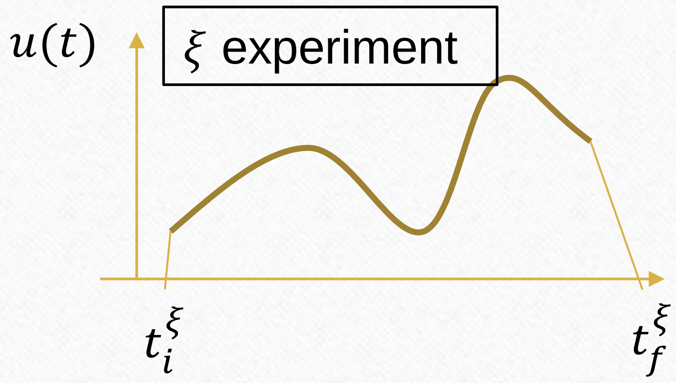
$$S_i^{\xi} = \frac{\mathbf{y}^{\xi}(\{\phi^1, \ldots\phi^i + h, \ldots\phi^{N_\phi}\}) - \mathbf{y}^{\xi}(\phi)}{h}$$

Identification: design of experiment

The Fisher matrix is a lower bound for the **parameters covariance matrix** C_ϕ^ξ: $C_\phi^\xi \geq F^\xi(\phi)^{-1}$

In order to **minimize the uncertainty on the estimated parameter** vector $\hat{\phi}$, we minimize for example the trace of the Fisher Matrix inverse

$$\boxed{\begin{gathered}\min_{u(\cdot)} \mathrm{Tr}\left(F^\xi(\hat{\phi})^{-1}\right) \\ x(t_i^\xi) = x_0^\xi \\ I_{min} \leq u(t) \leq I_{max} \\ h(x(t), u(t), \hat{\phi}) \leq 0\end{gathered}}$$



It is an optimization problem!

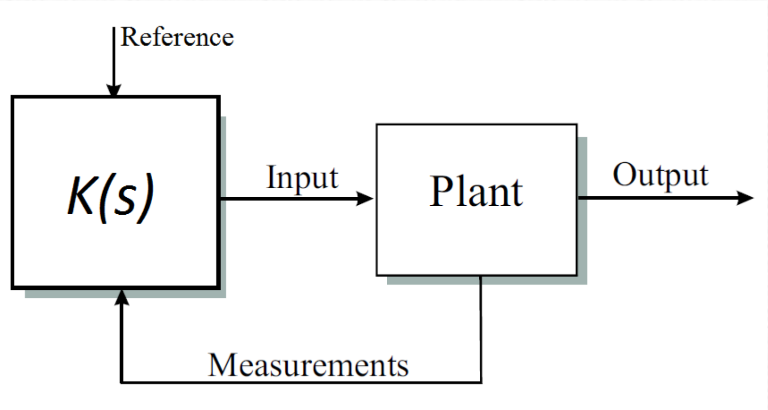
Why optimization?

Useful in many contexts

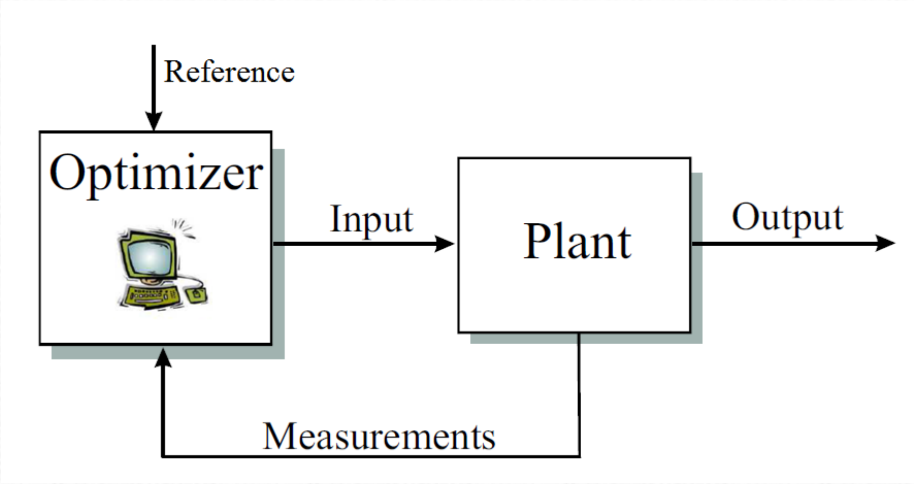
- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Control

Classic control



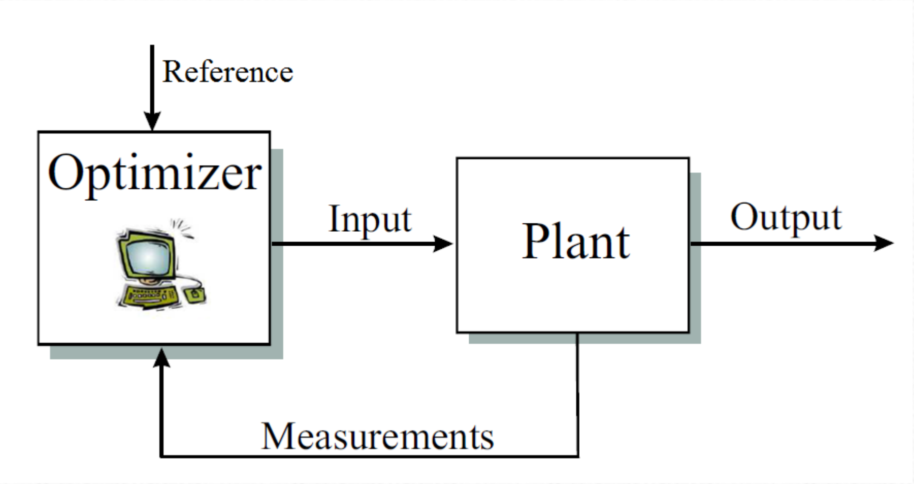
Optimization-based control

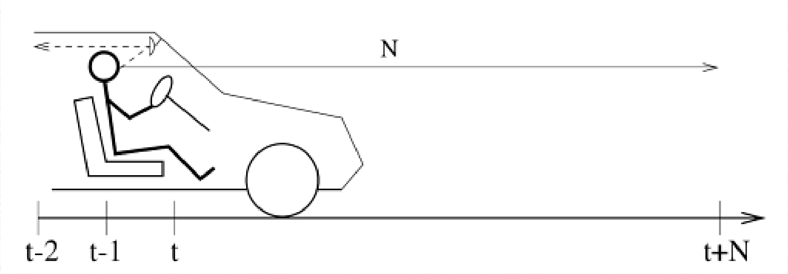


The classic controller is replaced by an **optimization** algorithm that runs on-line

Optimization-based control

Optimization-based control





The optimization uses **predictions** based on a **model** to optimize performance (e.g. minimize costs, maximize return of investment, etc.)

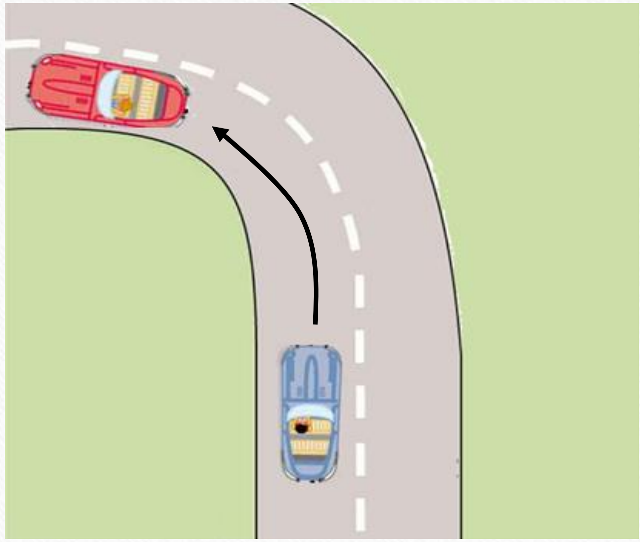
Optimization-based control

Driving a car

minimize (distance from desired path)

subject to constrains on:

- car dynamics
- distance from leading car
- speed limitations
- …



Further details in the course of Industrial Control (Prof. Lalo Magni)

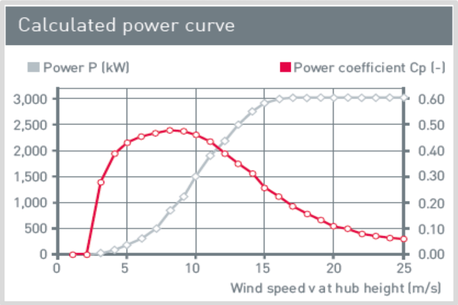
Why optimization?

Useful in many contexts

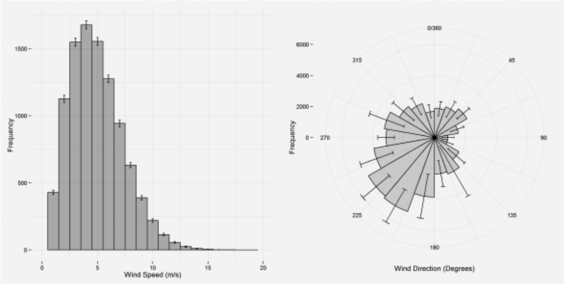
- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Optimal placement/sizing

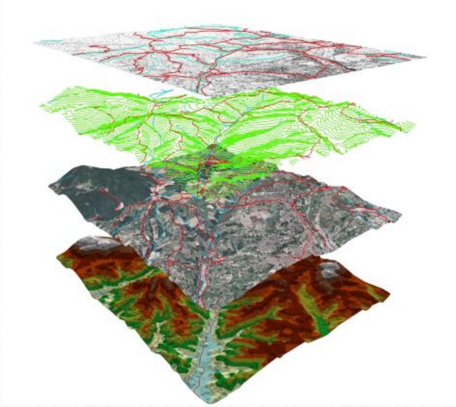
Choose the **number** and the **location** of a set of **wind turbines** in order to maximize the return of investment of a wind farm. Several elements need to be taken into account



Power Curve



Wind distribution

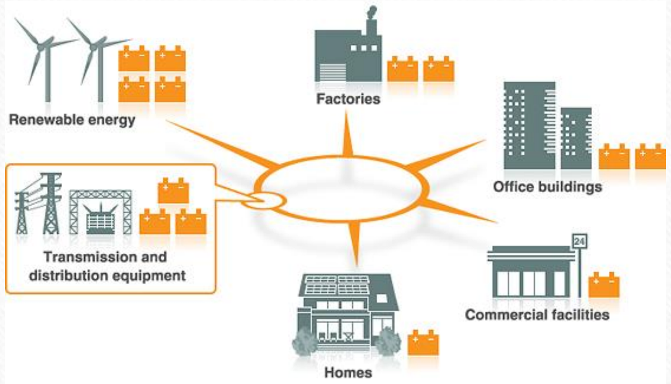


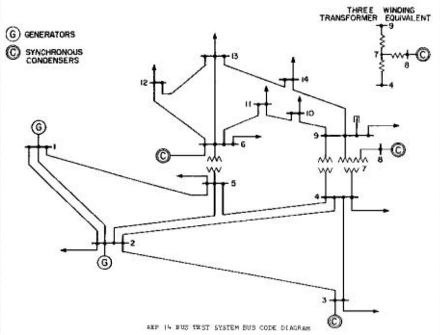
Geographic information

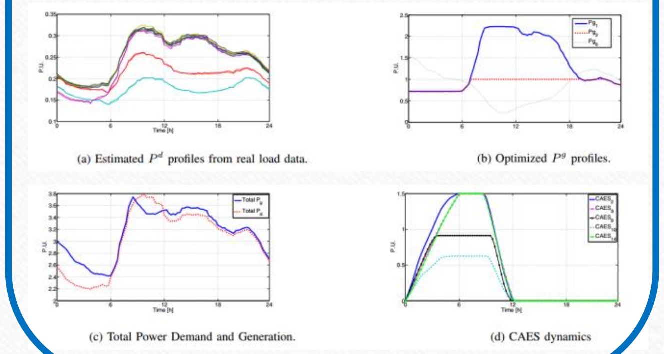
Wake effect

Optimal placement/sizing

Energy Storage Systems (ESS) can help to cope with intermittent availability of renewable sources. However, fixed, maintenance, and operating costs are a critical aspect that must be considered in the optimal positioning and sizing of these devices







Why optimization?

Useful in many contexts

- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Resources allocation

Demand Driven Employee Scheduling for the Swiss Market

C.N. Jones
Automatic Control Laboratory,
EPFL, Lausanne,
Switzerland,
colin.jones@epfl.ch

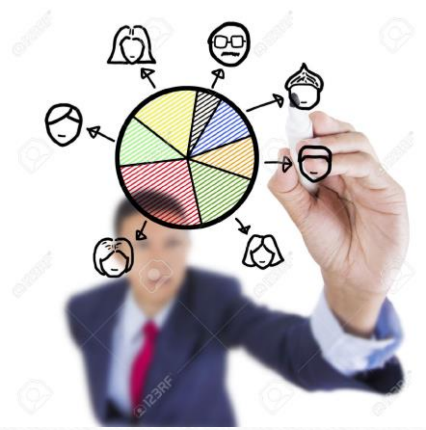
K. Nolde
Apex Optimization GmbH,
c/o Automatic Control Laboratory,
ETH Zurich, 8092 Zurich,
Switzerland,
nolde@control.ee.ethz.ch

June 24, 2013

1 Introduction

Standard practice for Swiss retail chains is to schedule employees so that the total number of workers present in the store is approximately constant during open hours. The number of shoppers, however, fluctuates throughout the day, which results in periods of under- and/or overstaffing that in turn reduces the effectiveness of the workforce. This paper reports on a new scheduling system that has been developed specifically for the Swiss market by Apex Optimization GmbH. The tool seeks to match expected customer demand to the number of sales staff by optimizing the shifts of the work force. The system has been successfully used by 38 small to mid-sized retail stores of the Migros chain of Switzerland over the past year, and the results of this initial implementation are reported here.

Schedules are computed on a weekly basis, one or more weeks in advance. Each week, the employees and/or store managers specify a wide range of store and employee-specific constraints through a web-based interface. The system then formulates a mixed-integer optimization problem in order to select a shift schedule that minimizes over- and under-staffing against a predicted customer demand profile, which has been estimated from past sales records.



Why optimization?

Useful in many contexts

- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

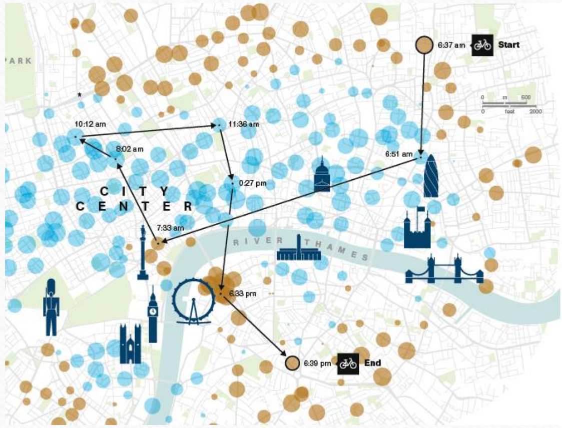
Routing/redistribution problems

EURO Journal on Transportation and Logistics
August 2013, Volume 2, Issue 3, pp 187–229

Static repositioning in a bike-sharing system: models and solution approaches

Authors   Authors and affiliations

Tal Raviv, Michal Tzur, Iris A. Forma



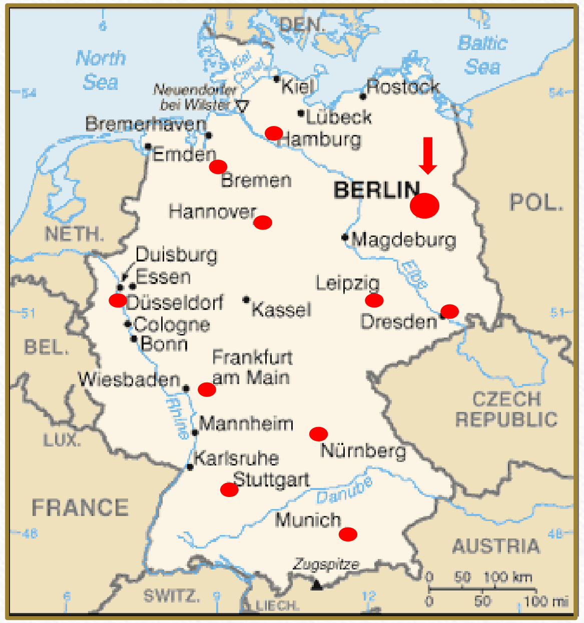
Abstract

Bike-sharing systems allow people to rent a bicycle at one of many automatic rental stations scattered around the city, use them for a short journey and return them at any station in the city. A crucial factor for the success of a bike-sharing system is its ability to meet the fluctuating demand for bicycles and for vacant lockers at each station. This is achieved by means of a *repositioning* operation, which consists of removing bicycles from some stations and transferring them to other stations, using a dedicated fleet of trucks. Operating such a fleet in a large bike-sharing system is an intricate problem consisting of decisions regarding the routes that the vehicles should follow and the number of bicycles that should be removed or placed at each station on each visit of the vehicles. In this paper, we present our modeling approach to the problem that generalizes existing routing models in the literature. This is done by introducing a unique convex objective function as well as time-related considerations. We present two mixed integer linear program formulations, discuss the assumptions associated with each, strengthen them by several valid inequalities and dominance rules, and compare their performances through an extensive numerical study. The results indicate that one of the formulations is very effective in obtaining high quality solutions to real life instances of the problem consisting of up to 104 stations and two vehicles. Finally, we draw insights on the characteristics of good solutions.

Routing/redistribution problems

The Travelling Salesman Problem

Given a list of cities and the distances between each pair of cities…

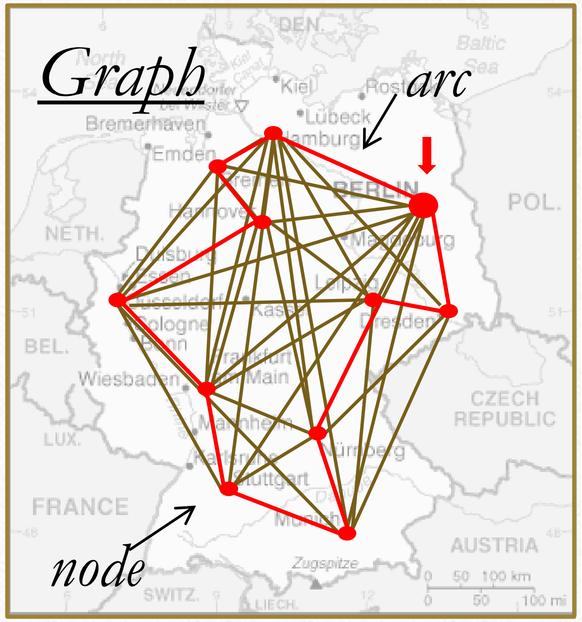


What is the shortest route that visits each city once and only once?

Routing/redistribution problems

The Travelling Salesman Problem

Given a list of cities and the distances between each pair of cities…



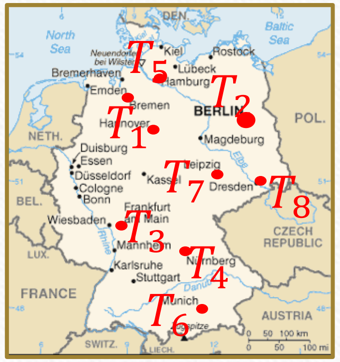
The objective function is the minimization of the cost of the path

Resource allocation + routing

Assume to have ***n*** **operators** that need to perform ***m*** **tasks** of **different duration** at **different locations**

The objective is to decide

- which and how many tasks to assign at each operator.
- for each operator in which order and over which route to perform the tasks.



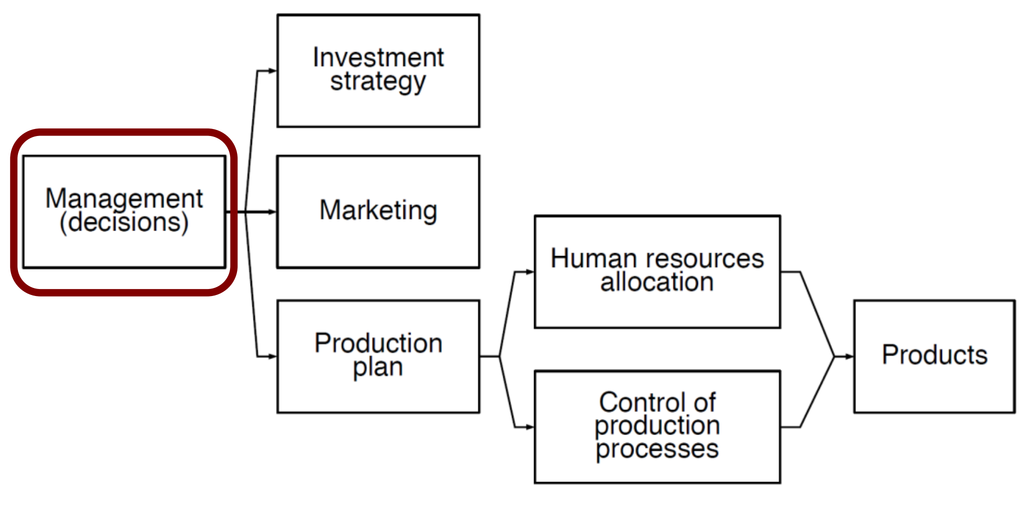
We aim to **minimize** the overall execution time subject to working hours constraints.

Why optimization?

Useful in many contexts

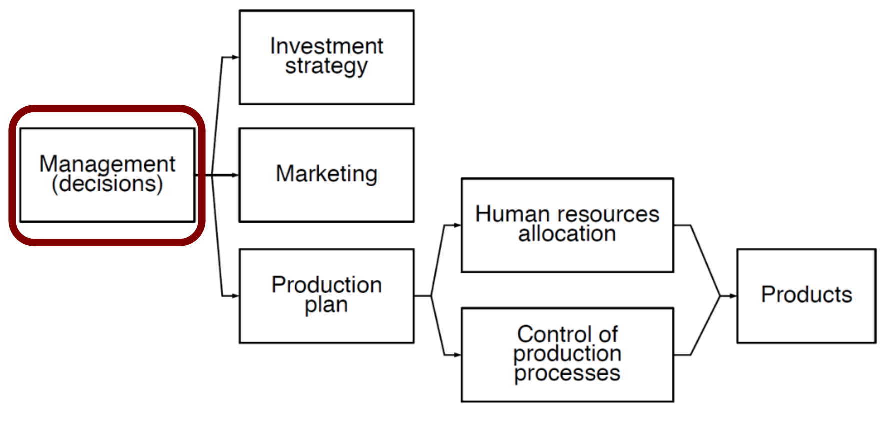
- Identification
- Control
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Planning of production processes



Management science: optimal decisions for complex problems

Planning of production processes



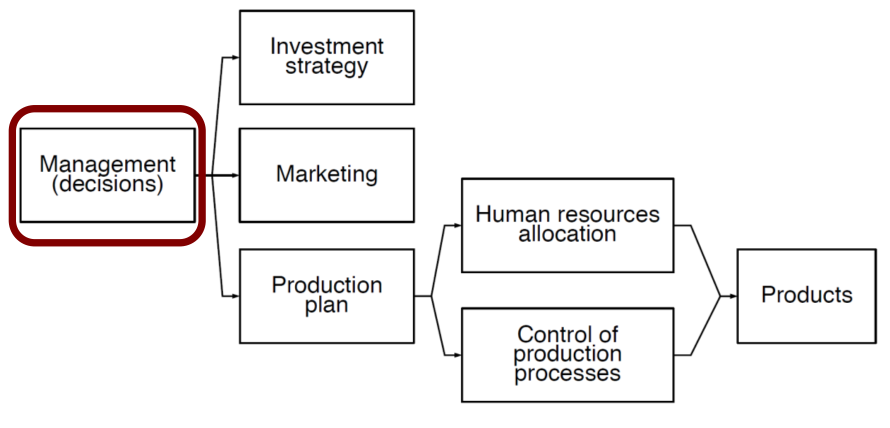
Management: decisions can be either "instinctive" or structured

- "Instinctive" decisions:
  - *Pros:* rapidity and flexibility
  - *Cons:* no quantitative model
    - limited number of the alternatives
    - limited understanding of decision criteria

Suboptimal decisions

Drawbacks can be extremely critical if decisions are complex (several alternatives / mutually dependent choices / limited resources)

Planning of production processes



Management: decisions can be either "instinctive" or structured

- Structured decisions (based on a quantitative model):
  - *Pros:*
    - Better understanding of the problem
      - consideration of all possible alternatives
      - precise decision criteria
    - optimal decisions can be tacken even for complex problems
  - *Cons:* getting a mathematical model of a decision problem might be time and resource consuming
    - trade-off between time/resources for decision-making and benefits of optimality. Very often optimality wins !

Example: product mix

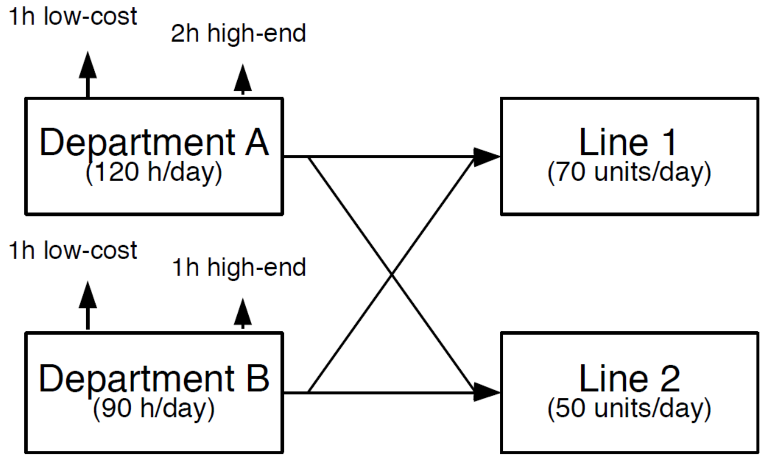
A company manifactures two radio models (low-cost and high-end) and produces two components

- Department A: antennas
  - no more than 120h hours of production per day
    - 1h of work for a low-cost antenna
    - 2h of work for a high-end antenna
- Department B: cases
  - no more than 90h hours of production per day
    - 1h of work for a low-cost case
    - 1h of work for a high-end case

The company has two assembly lines (1 radio=1 antenna + 1 case)

- Line 1: production of low-cost models. No more than 70 units/day
- Line 2: production of high-end models. No more than 50 units/day

Example: product mix

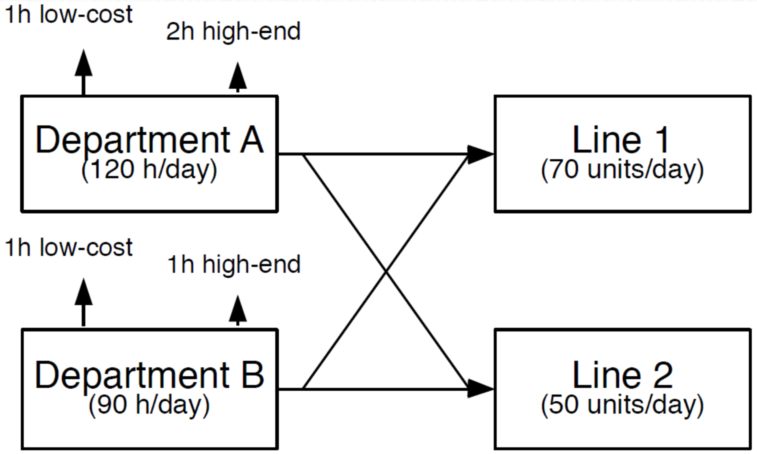


Profits: 20 Euro for a low-cost radio and 30 Euro for a high-end radio

Assuming the company will sell all the radios, which is the optimal number of units, for each model, that must be produced daily for maximizing the revenue?

Optimal daily production plan = mix of two products

Example: product mix



Instinctive (and greedy) manager: higher profits for high-end models ⇨ maximize their production (50 units/day)

Department A: 100h for high-end antennas (50 antennas) ⇨ 20h for low-cost antennas (20 antennas)

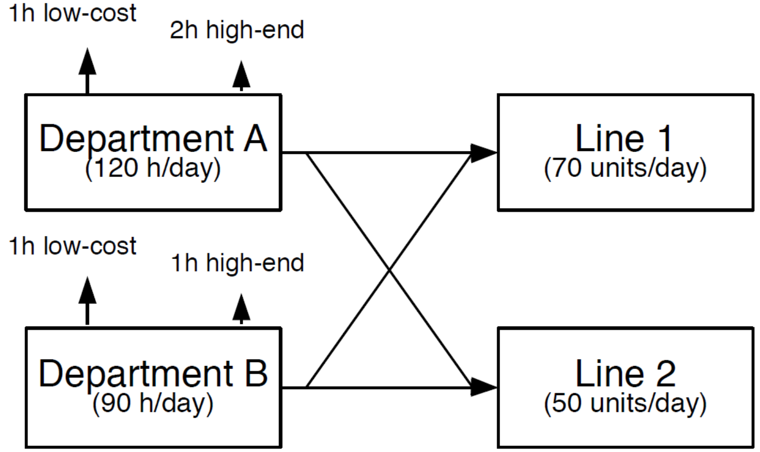
Department B: 50h for high-end cases (50 cases) ⇨ 20h for low-cost cases (20 cases)

Line 1: 20 low-cost radios per day

Line 2: 50 high-end radios per day

Daily profits: 20*20+50*30=1900 Euro. Is there any better solution ?

Example: product mix

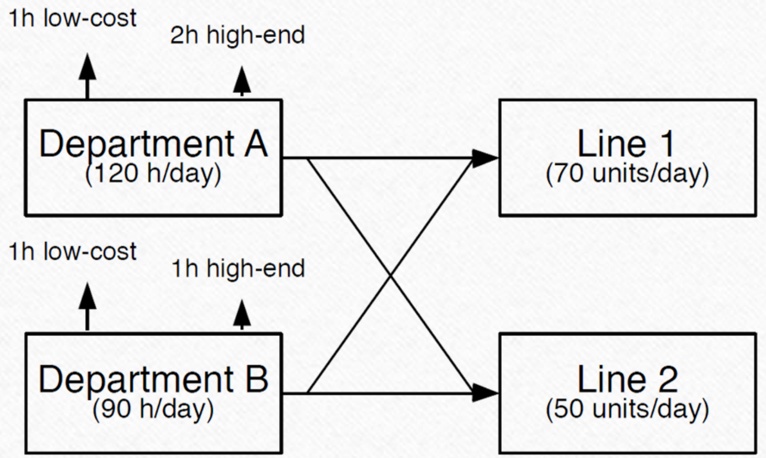


Smart manager: 60 low-cost models and 30 high-end models

Department A: 60h for high-end antennas (30 antennas) ⇨ 60h for low-cost antennas (60 antennas)
Department B: 30h for high-end cases (30 cases) ⇨ 60h for low-cost cases (60 cases)
Line 1: 60 low-cost radios per day
Line 2: 30 high-end radios per day

Daily profits: 60*20+30*30=2100 Euro

Example: product mix



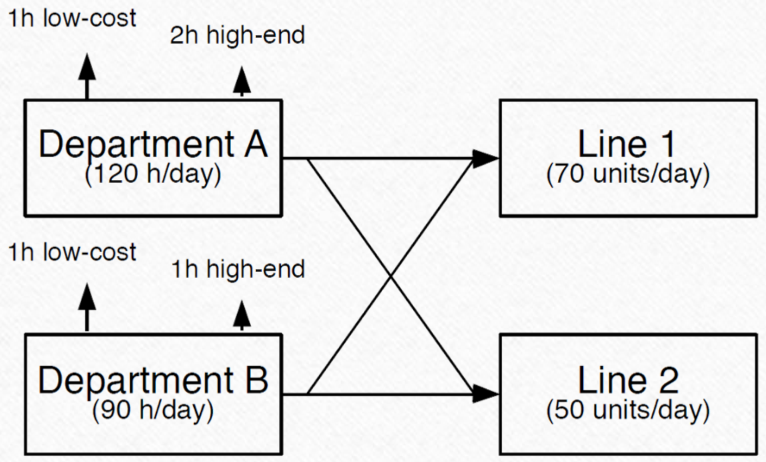
Decisions taken by the smart manager are **optimal** (profits cannot increase)

Smart manager: 60 low-cost models and 30 high-end models

- Department A: 60h for high-end antennas (30 antennas) ⇨ 60h for low-cost antennas (60 antennas)
- Department B: 30h for high-end cases (30 cases) ⇨ 60h for low-cost cases (60 cases)
- Line 1: 60 low-cost radios per day
- Line 2: 30 high-end radios per day

Daily profits: 60*20+30*30=2100 Euro

Example: product mix



How the manager came up with this plan ? How can we certify it is optimal ?

Smart manager: 60 low-cost models and 30 high-end models

Department A: 60h for high-end antennas (30 antennas) ⇨ 60h for low-cost antennas (60 antennas)
Department B: 30h for high-end cases (30 cases) ⇨ 60h for low-cost cases (60 cases)
Line 1: 60 low-cost radios per day
Line 2: 30 high-end radios per day

Daily profits: 60*20+30*30=2100 Euro

Optimization

Optimization

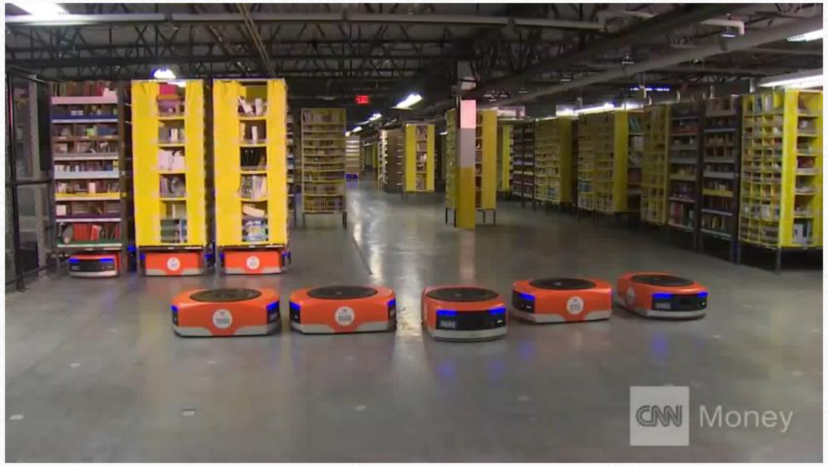
Mathematical formalization

+

optimization algorithms

Is it worth?

Optimization



Introduction to optimization

Optimization is also known as mathematical programming

- *Programming* means planning or building an action plan for solving a problem or taking a decision

- Optimization falls in the fields of operations research and management science

Introduction to optimization

Standard form of a **continuous** optimization problem:

$$
\begin{array}{ll}
\text{minimize} & f(x) \\
\text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\
 & h_i(x) = 0, \quad i = 1, \cdots, p
\end{array}
$$

Introduction to optimization

Standard form of a **continuous** optimization problem:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

Variables (optimization variables): $x = [x_1, \cdots, x_n]^\top$

Objective function (or cost): $f : \mathbb{R}^n \to \mathbb{R}$

Constraints: $g_i : \mathbb{R}^n \to \mathbb{R}, i = 1, \cdots, m$ $h_i : \mathbb{R}^n \to \mathbb{R}, i = 1, \cdots, p$ (if no constraints: **unconstrained problem)**

Feasible region: $X = \{x \in \mathbb{R}^n : g_1(x) \leq 0, \cdots, g_m(x) \leq 0, h_1(x) = 0, \cdots, h_p(x) = 0\}$

Feasible point (also said feasible solution): $x \in X$

Introduction to optimization

By convention, the standard form defines a minimization problem.

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

A **maximization problem** can be treated by negating the objective function.

$$\max_{x \in X} f(x) = -\min_{x \in X} -f(x)$$

optimal solutions are **the same** for both problems

Introduction to optimization

By convention, the standard form defines a minimization problem.

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \le 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

A **maximization problem** can be treated by negating the objective function.

$$\max_{x \in X} f(x) = -\min_{x \in X} -f(x)$$

optimal solutions are **the same** for both problems

Conversion from $\le$ to $\ge$ in the constraints

$$\{x \in \mathbb{R}^n : g(x) \ge 0\} = \{x \in \mathbb{R}^n : -g(x) \le 0\}$$

the feasible region does not change

Introduction to optimization

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

- $x^* \in X$ is an optimal solution

  (**global** minimum point) if

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

- $\bar{x} \in X$ is a local optimal solution

  (**local** minimum point) if

$$\exists \varepsilon > 0 : \forall x \in X, ||x - \bar{x}|| < \varepsilon \Rightarrow f(\bar{x}) \leq f(x)$$

Introduction to optimization

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \le 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

- $x^* \in X$ is an optimal solution

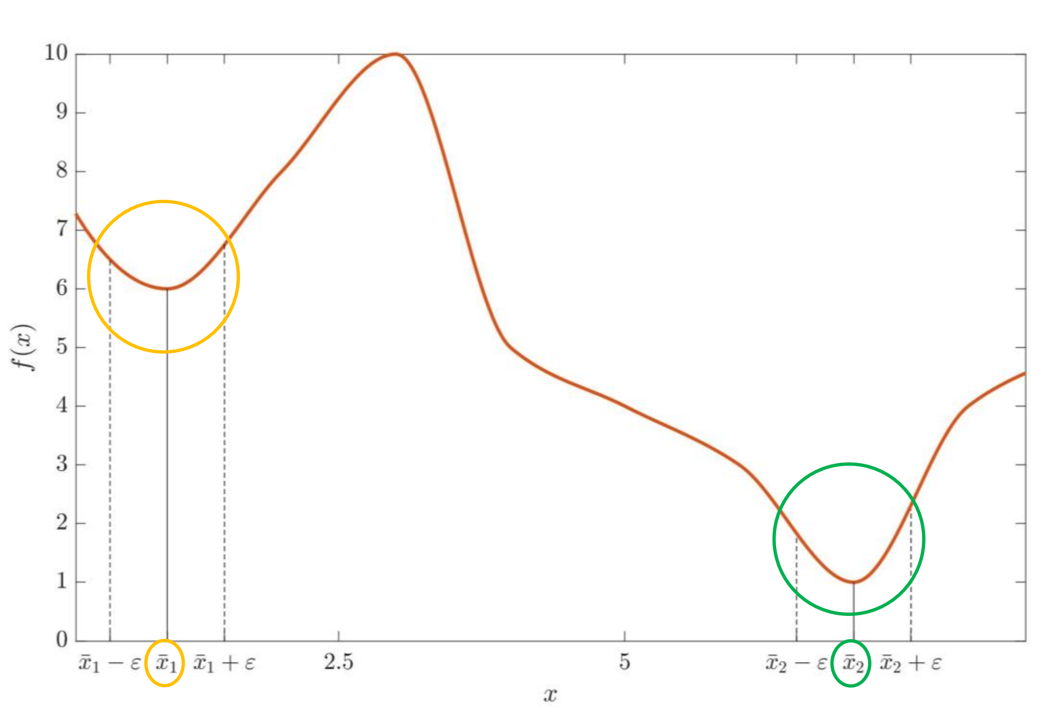
  (**global** minimum point) if

  $$f(x^*) \le f(x), \;\; \forall x \in X$$

- $\bar{x} \in X$ is a local optimal solution

  (**local** minimum point) if

$$\exists \varepsilon > 0 : \forall x \in X, ||x - \bar{x}|| < \varepsilon \Rightarrow f(\bar{x}) \le f(x)$$



In the figure: $\bar{x}_1$ and $\bar{x}_2$ are respectively a local and the global minimum point. $f(\bar{x}_2)$ is the **optimal cost**

Introduction to optimization

$$
\begin{array}{ll}
\text{minimize} & f(x) \\
\text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\
 & h_i(x) = 0, \quad i = 1, \cdots, p
\end{array}
$$

Optimal value (optimal cost)

$$p^* = \inf\{f(x) : g_1(x) \leq 0, \cdots, g_m(x) \leq 0, h_1(x) = 0, \cdots, h_p(x) = 0\}$$

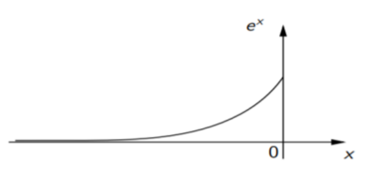
Introduction to optimization

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \le 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

Optimal value (optimal cost)

$$p^* = \inf\{f(x) : g_1(x) \le 0, \cdots, g_m(x) \le 0, h_1(x) = 0, \cdots, h_p(x) = 0\}$$

- In some cases, the basic problem can be
  - **infeasible** (if $X = \emptyset$) $p^* = \infty$
  - **unbounded** (if $\forall k < 0 \; \exists x \in X : f(x) < k$) $p^* = -\infty$
- Even if the basic problem is feasible and bounded, optimal solutions could
  - **exist and be not unique** (e.g. *f (x)* constant)
  - **not exist** e.g. $\min_{x \le 0} e^x \quad x \in \mathbb{R}$ $\Rightarrow$



Introduction to optimization

$$
\boxed{\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}}
$$

No easy way to solve the basic problem in its full generality!

- Need of numerical algorithms
- Often, only local optimal solutions can be computed

Convex optimization

Convex programming

A **convex optimization problem** is an optimization problem in which

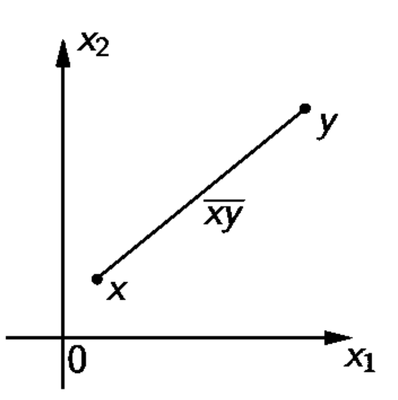
- the **feasible set** is a **convex set**
- the **objective function** is a **convex function.**

Convex set

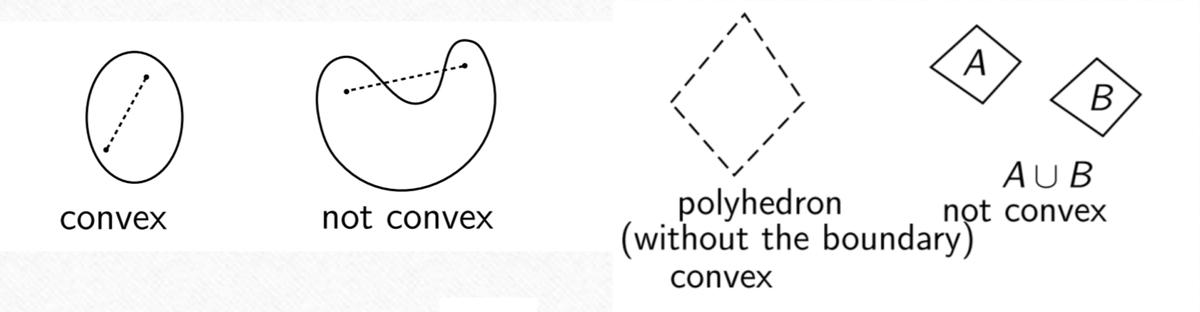
Definition: given two points $x, y \in \mathbb{R}^n$, the set

$$\overline{xy} = \{\lambda x + (1-\lambda)y : \lambda \in [0,1]\}$$

is a segment joining x and y



Definition: the set $X \subseteq \mathbb{R}^n$ is **convex** if $\forall x, y \in X$ one has $\overline{xy} \in X$

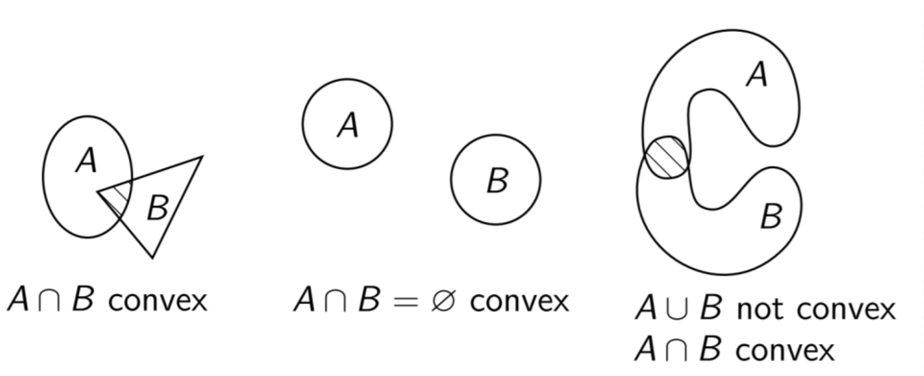


$\mathbb{R}^n$ is convex

Convex sets

Proposition (try to prove it at home): the intersection of two convex sets is a convex set.

Note: the proposition implies that the empty set is also a convex set.



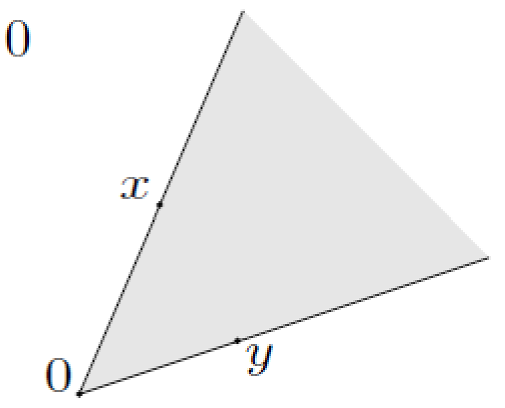
Attention: the union of two convex sets is not convex in general!

Convex cone

$X \subseteq \mathbb{R}^n$ is a **convex cone** if given any $x, y \in X$ and any $\lambda, \mu \geq 0$

$$\boxed{\lambda x + \mu y \in X}$$

Geometrically: $x, y \in X \rightarrow$ 'pie slice' between $x, y \subseteq X$



Note: in the definition of convex set, μ was set equal to $(1 - \lambda)$ (and therefore the sum of λ, μ was equal to 1). ➡ A convex cone is a convex set but not all convex sets are convex cones.

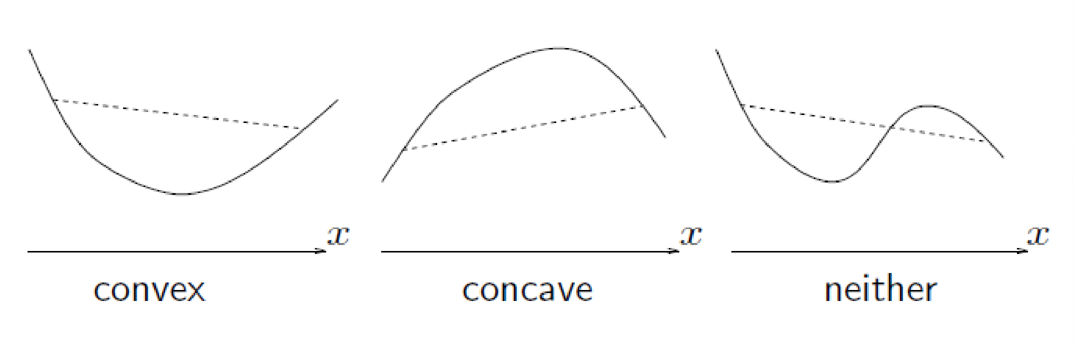
Convex functions

Definition: a function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $\mathbf{dom}(f)$ is convex and, for all $x, y \in \mathrm{dom}(f)$ and $\lambda \in [0, 1]$ one has $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$

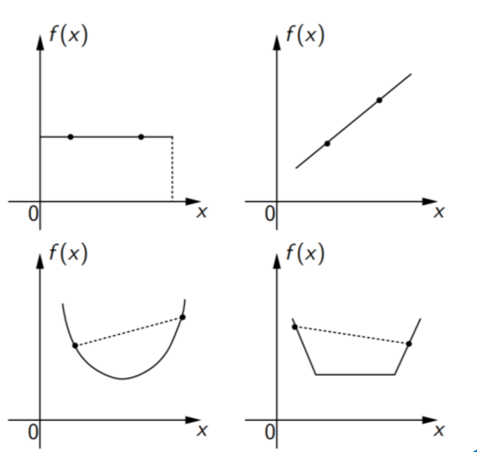
Convex functions

Definition: a function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $\mathbf{dom}(f)$ is convex and, for all $x, y \in \mathbf{dom}(f)$ and $\lambda \in [0,1]$ one has $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$

Note: f is **concave** if $-f$ is convex



Convex examples



Convex functions

All **norms** are convex

Norms in $\mathbb{R}^n$

Let $p \geq 1$ be a real number. The p-norm (also called l_p-norm) of vector $\mathbf{x} = (x_1, \cdots, x_n)$ is

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

Convex functions

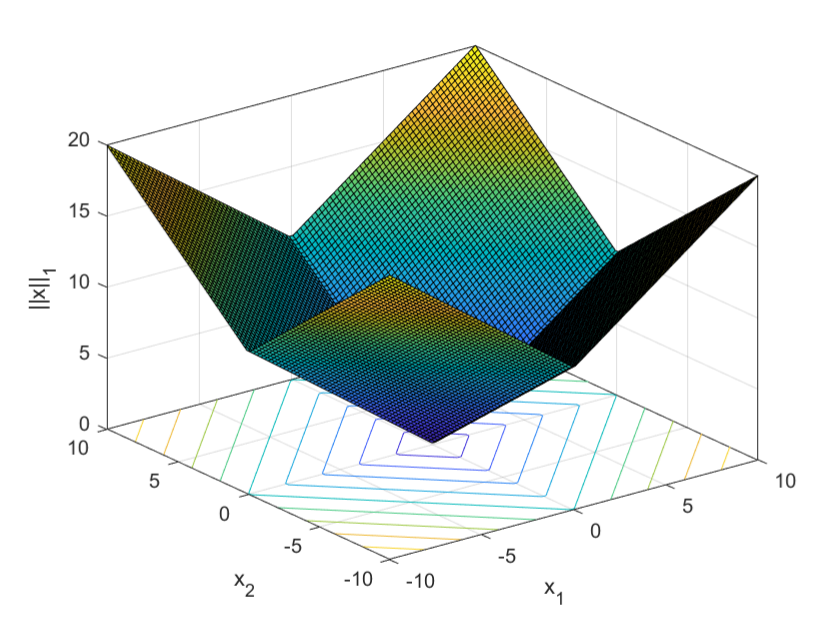
All **norms** are convex

Norms in $\mathbb{R}^n$

Let $p \geq 1$ be a real number. The p-norm (also called l_p-norm) of vector $\mathbf{x} = (x_1, \cdots, x_n)$ is

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

$$||x||_1 = \sum_{i=1}^{n} |x_i|$$



Convex functions

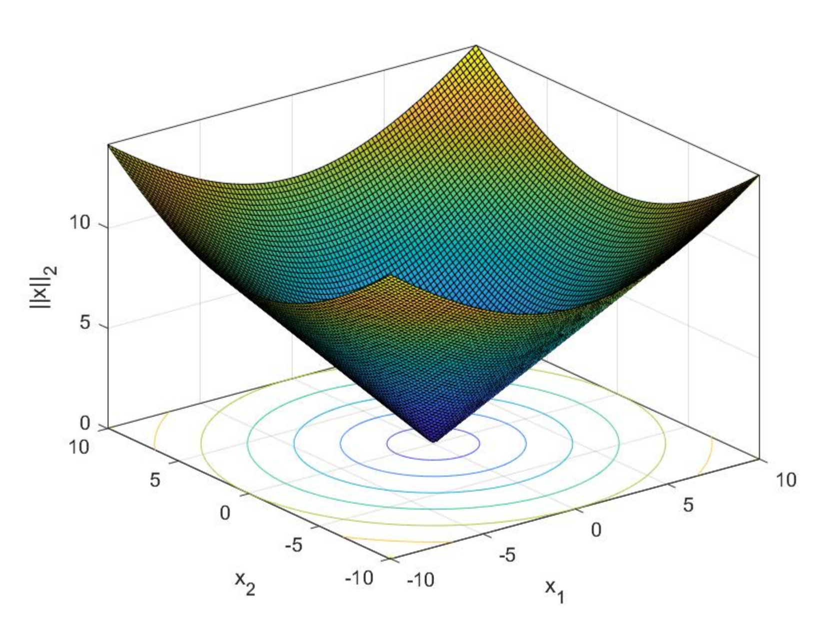
All **norms** are convex

Norms in $\mathbb{R}^n$

Let $p \geq 1$ be a real number. The p-norm (also called l_p-norm) of vector $\mathbf{x} = (x_1, \cdots, x_n)$ is

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

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



Convex functions

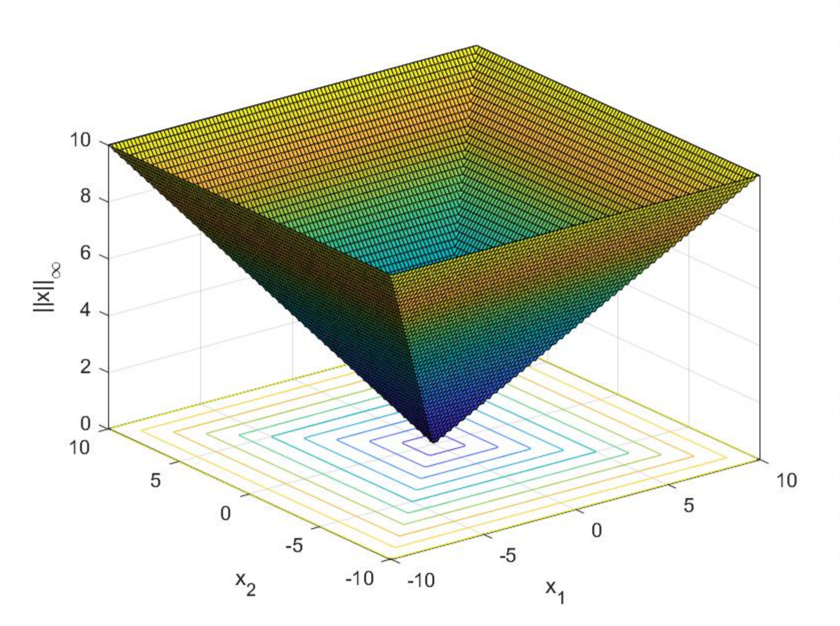
All **norms** are convex

Norms in $\mathbb{R}^n$

Let $p \geq 1$ be a real number. The p-norm (also called l_p-norm) of vector $\mathbf{x} = (x_1, \cdots, x_n)$ is

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

$$||x||_\infty = \max_i |x_i|$$



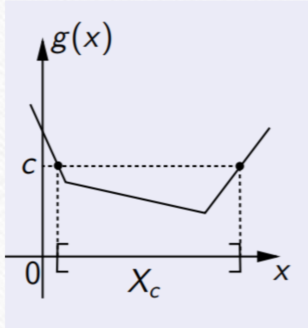
Convex functions and sets

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \le 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

If g_i are convex $i = 1, \dots, m$, and h_i are affine ($h_i = c^T x + b$), $i = 1, \dots, p$, what can we say about the feasible set?

Theorem: let $\boxed{g_i : \mathbb{R}^n \to \mathbb{R} \text{ be a convex function}}$ and take any $c \in \mathbb{R}$.

Then, the level set $X_c = \{x \in \mathbb{R}^n : g_i(x) \le c\}$ is convex.



Proof: Pick $x, y \in X_c$ and $\lambda \in [0, 1]$ and consider $z = \lambda x + (1 - \lambda) y$: we have to show that $z \in X_c$. From the convexity of g_i one has that $g_i(z) \le \lambda g_i(x) + (1 - \lambda) g_i(y)$. Since $x, y \in X_c$ one has

$$g_i(z) \le \lambda g_i(x) + (1 - \lambda) g_i(y) \le \lambda c + (1 - \lambda) c = c$$

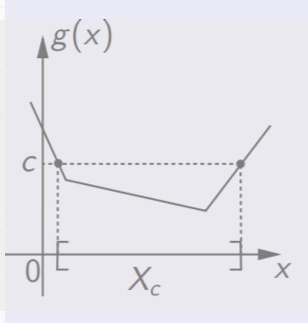
that implies $z \in X_c$.

Convex functions and sets

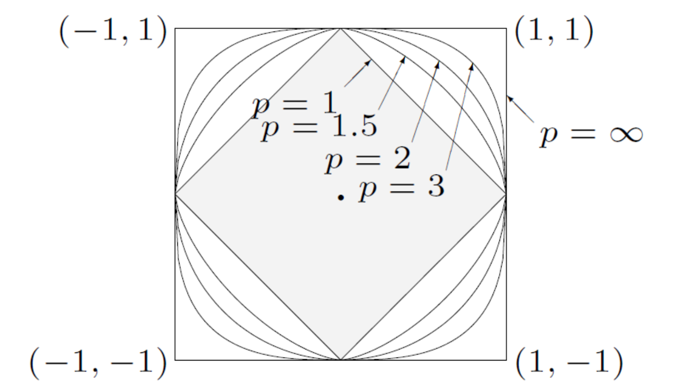
$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g_i(x) \leq 0, & i = 1, \cdots, m \\ & h_i(x) = 0, & i = 1, \cdots, p \end{array}$$

Theorem: let $\boxed{g_i : \mathbb{R}^n \to \mathbb{R} \text{ be a convex function}}$ and take any $c \in \mathbb{R}$.

Then, the <u>level set</u> $X_c = \{x \in \mathbb{R}^n : g_i(x) \leq c\}$ is convex.



The norm ball $B = \{x : ||x - x_c||_p \leq 1\}$ is a convex set



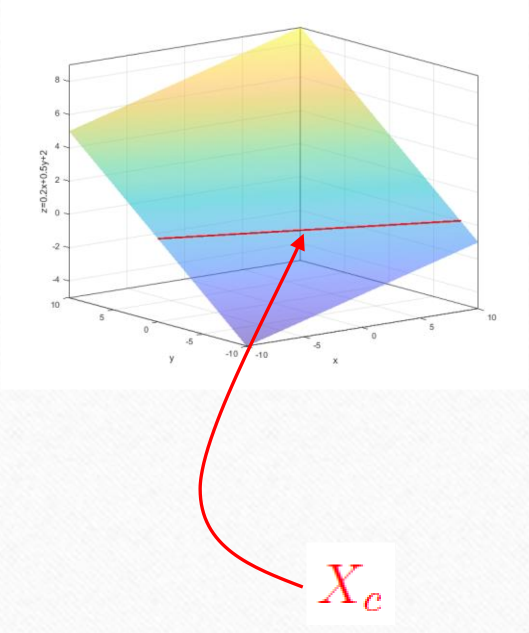
Convex functions and sets

$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g_i(x) \le 0, & i = 1, \cdots, m \\ & h_i(x) = 0, & i = 1, \cdots, p \end{array}$$

If g_i are convex $i = 1, \dots, m,$ and h_i are affine (h_i=$p^T x + q$), $i = 1, \dots, p,$ what can we say about the feasible set?

Theorem: let $\boxed{h_i : \mathbb{R}^n \to \mathbb{R} \text{ be an affine function}}$ (h_i=$p^T x + q$)

and take any $c \in \mathbb{R}$. Then, the set $X_c = \{x \in \mathbb{R}^n : h_i(x) = c\}$ is convex.



Proof: Pick $x, y \in X_c$ and $\lambda \in [0, 1]$ and consider $z = \lambda x + (1 - \lambda)y$: we have to show that $z \in X_c$. Since $x, y \in X_c$ one has

$$\begin{aligned} h_i(z) &= h_i(\lambda x + (1 - \lambda)y) \\ &= p^\mathsf{T}(\lambda x + (1 - \lambda)y) + q \\ &= \lambda p^\mathsf{T}(x) + (1 - \lambda)p^\mathsf{T}(y) + q \\ &= \lambda(h_i(x) - q) + (1 - \lambda)(h_i(y) - q) + q \\ &= \lambda(c - q) + (1 - \lambda)(c - q) + q = c \end{aligned}$$

that implies $z \in X_c$.

Convex functions and sets

Key corollary

Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

If g_i are convex $i = 1, \dots, m$, and h_i are affine ($h_i = p^T x + q$), $i = 1, \dots, p$ then the feasible region is convex. **Moreover, if *f(x)* is also convex, then the optimization problem is convex.**

Proof: the proof follows from the previous theorem and the fact that convexity is preserved by intersection.

Convex programming

A **convex optimization problem** is an optimization problem in which

- the **feasible set** is a **convex set**
- the **objective function** is a **convex function.**

Remark: the optimization problem $\{\max f(x): g_i(x) \leq 0, i = 1, \ldots, m,\ h_i(x) = 0, i = 1, \ldots, p\}$ is not a convex program even if f, g_i are convex and h_i are affine. Indeed, it is equivalent to

$$\{-min -f(x): g_i(x) \leq 0, i = 1, \ldots, m,\ h_i(x) = 0, i = 1, \ldots, p\}$$

where the function $-f(x)$ is concave.

Notable exception: $f(x)$ **linear** is both **convex and concave**

Fundamental theorem of convex programming

Important property of convex programs

Theorem: consider the following convex programming problem

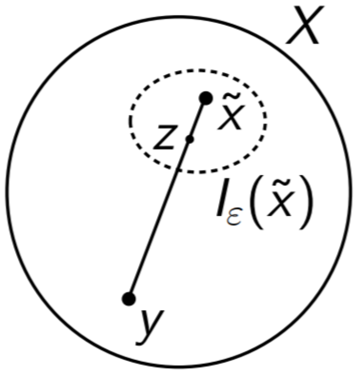
$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

and denote with X the feasible set. If $\tilde{x} \in X$ is a local optimal solution for the problem above, then $\tilde{x}$ is a (global) optimal solution.

Proof of the theorem

The goal is to show $f(\tilde{x}) \leq f(y)\ \forall y \in X$.
Fix $y \in X$, $y \neq \tilde{x}$ and let $I_\epsilon(\tilde{x})$ be a neighborhood of $\tilde{x}$ such that $z \in I_\epsilon(\tilde{x}) \Rightarrow f(\tilde{x}) \leq f(z)$. Pick $z \in X$ such that $z \in \overline{\tilde{x}y}$, $z \in I_\epsilon(\tilde{x})$ and $z \neq \tilde{x}$.
Such a z exists because

$$z = \lambda\tilde{x} + (1-\lambda)y$$



and

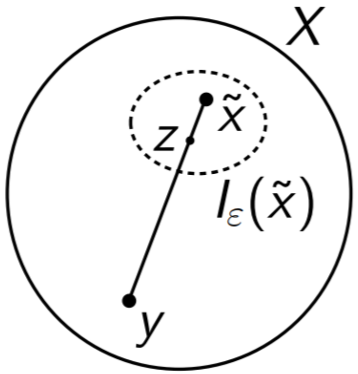
- choosing λ sufficiently close to 1 guarantees $z \in I_\epsilon(\tilde{x})$
- choosing $\lambda \neq 1$ guarantees $z \neq \tilde{x}$

Proof of the theorem

Then,

$$f(\tilde{x}) \underbrace{\leq}_{\text{local optimizer}} f(z) = f(\lambda\tilde{x} + (1-\lambda)y) \leq$$

$$\underbrace{\leq}_{f \text{ convex}} \lambda f(\tilde{x}) + (1-\lambda)f(y)$$

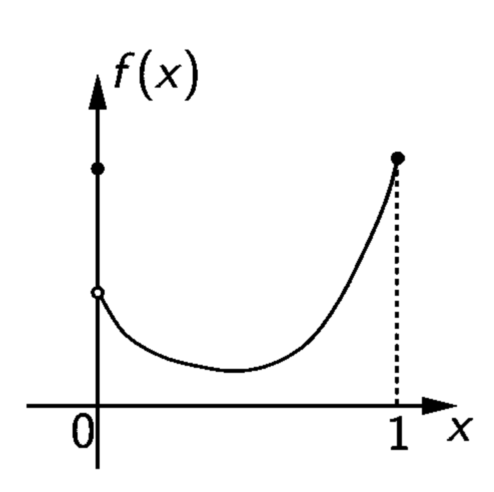


From the last inequality one has

$$(1-\lambda)f(\tilde{x}) \leq (1-\lambda)f(y) \underbrace{\Rightarrow}_{\lambda \neq 1} f(\tilde{x}) \leq f(y)$$

Convexity and smoothness

A **convex** function $f : X \to \mathbb{R}$, $X \subseteq \mathbb{R}^n$ is **continuous** in the **interior** of X



Continuity is needed! If we don't have it → not convex

If we do have it, then, how do we check convexity?

Differentiable convex functions

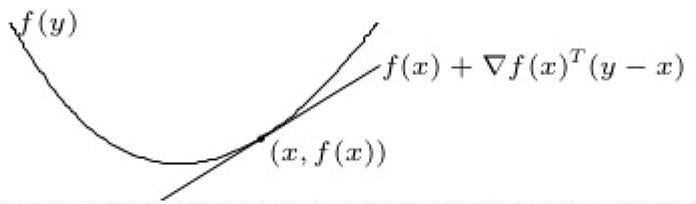
Gradient of $f : \mathbb{R}^n \to \mathbb{R}$: $\nabla f(x) = \left[\frac{\partial f}{\partial x_1} \quad \frac{\partial f}{\partial x_2} \quad \cdots \quad \frac{\partial f}{\partial x_n} \right]^\intercal$ evaluated at x

First order Taylor approximation at x_0: $f(x) \simeq f(x_0) + \nabla f(x_0)^\intercal (x - x_0)$

First order condition: for f differentiable (i.e. its gradient exists at each point of **dom** f, which is open) f is convex if and only if **dom** f is convex and

$$f(y) \geq f(x) + \nabla f(x)^\intercal (y - x)$$

holds for all $x, y \in$ **dom** f.



Differentiable convex functions of one real variable

(i.e. not empty and not reduced to a point)

Given a non-trivial interval $\mathrm{I} \subseteq \mathbb{R}$ and a function $f: \mathrm{I} \to \mathbb{R}$, differentiable in the interior of I, f is convex in I if and only if f' is an increasing function in I

(i.e. when $x_1 < x_2$ then $f(x_1) \leq f(x_2)$)

This condition can be verified more easily in practice than the one in the previous slide (see the examples).

Twice differentiable convex functions

Hessian of a twice differentiable function: $\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$ evaluated at x

Second order Taylor approximation at x_0: $f(x) \simeq f(x_0) + \nabla f(x_0)^\intercal (x - x_0) + \frac{1}{2}(x - x_0)^\intercal \nabla^2 f(x_0)(x - x_0)$

Second order condition: for f twice differentiable, f is convex $\boxed{\text{if and only if}}$

for all $x \in \mathbf{dom} f$, $\nabla^2 f(x) \succcurlyeq 0$

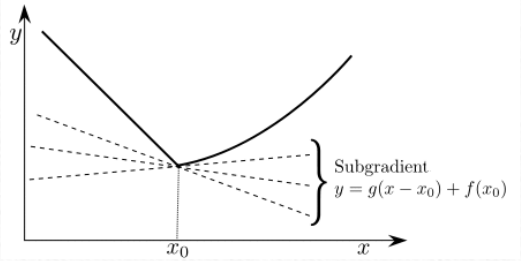
Continuous but **not differentiable** convex multi-variable functions

Non-differentiable functions do not have gradients at each point of the domain, but the existence of a supporting hyperplane can be used to check convexity.

The vector $g \in \mathbb{R}^n$ corresponding to a supporting hyperplane is called subgradient.

Definition: The subgradient of $f: \mathrm{I} \to \mathbb{R}$, $\mathrm{I} \subseteq \mathbb{R}^n$, at $x \in \mathrm{I}$ is a vector $g \in \mathbb{R}^n$, such that

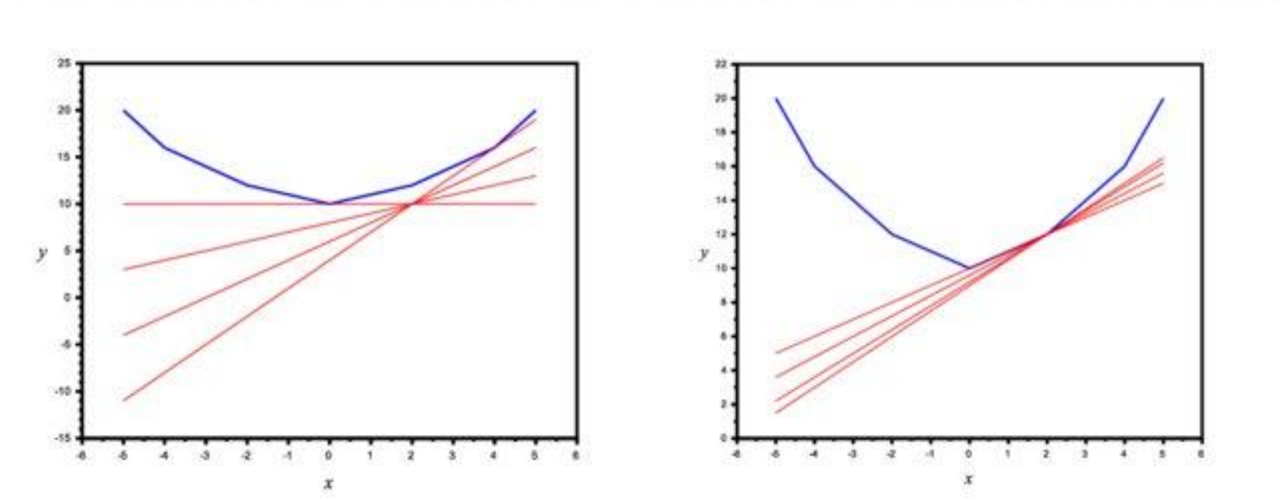
$$f(y) \geq f(x) + g^T (y - x), \forall y \in \mathrm{I}$$



The set of all subgradients is called the subdifferential of the function at x.

Continuous but **not differentiable** convex multi-variable functions

A function $f: \mathrm{I} \to \mathbb{R}$ is convex if and only if it has a non-empty subdifferential for any $x \in \mathrm{I}$.



The theorem establishes that a function is **convex** if and only if a subgradient exists at every point

Continuous but **not differentiable** convex functions of one real variable

Consider a non-trivial interval $\mathrm{I} \subseteq \mathbb{R}$ and a function $f: \mathrm{I} \to \mathbb{R}$, continuous in the interior of I. If f is convex in I, then, the limits

$$\lim_{x \to x_0^{\mp}} \frac{f(x) - f(x_0)}{x - x_0}$$

exist for all $x_0 \in \mathrm{I}$ [1]. In particular, if x_0 is inside the domain I, then both left (l_-) and right (l_+) limits exist, are finite and such that $l_- \leq l_+$

It extends the concept of f' being an increasing function to the case of non differentiable f.

[1] For the extremes only one of them makes sense.

Classification of convex optimization problems

Linear Program (LP): **affine** objective and constraint functions

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

$$\Rightarrow$$

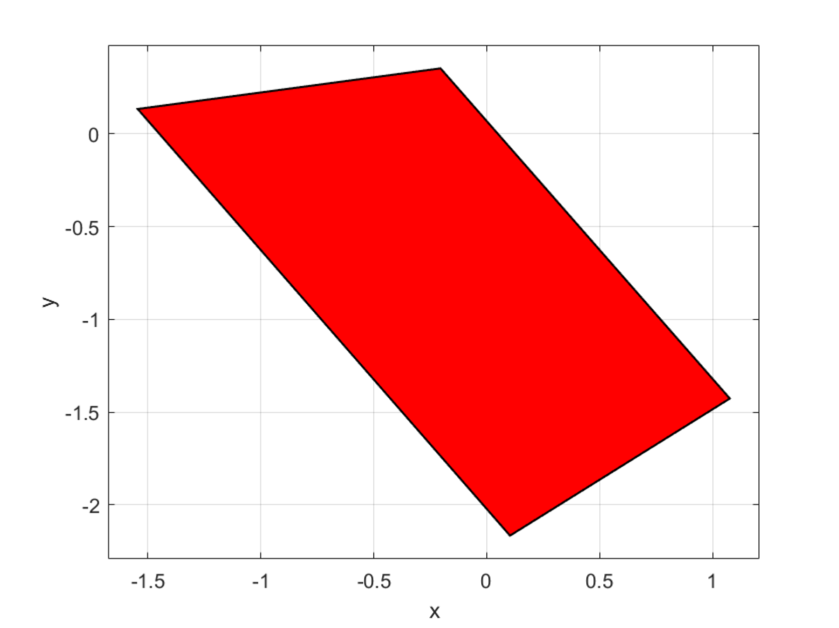
$$\begin{array}{ll} \text{minimize} & c^\intercal x + d \\ \text{subject to} & \boxed{\begin{array}{l} Gx - h \leq 0 \\ Ax - b = 0 \end{array}} \end{array}$$

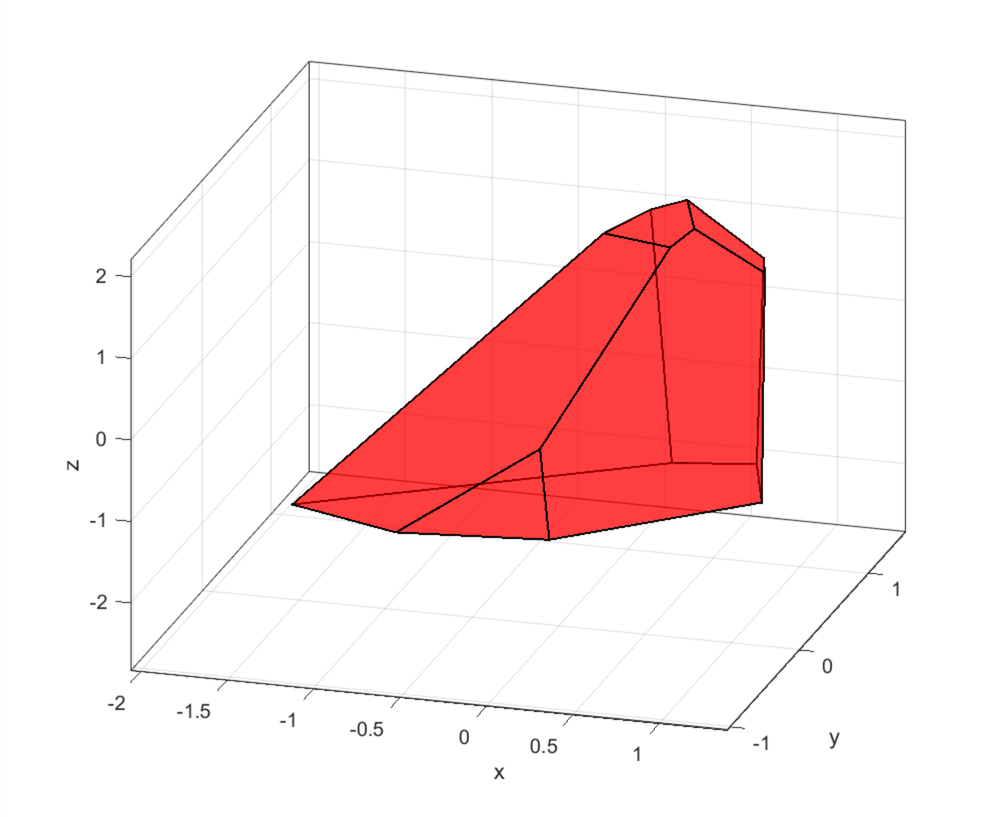
Constraints expressed in matricial form

$$\boxed{\begin{array}{ll} G \in \mathbb{R}^{m \times n} & h \in \mathbb{R}^{m} \\ A \in \mathbb{R}^{p \times n} & b \in \mathbb{R}^{p} \end{array}}$$

Classification of convex optimization problems

Example of LP constraints





Classification of convex optimization problems

Quadratic Program (QP): **quadratic** objective function and affine constraint functions

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array}$$

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} x^\intercal P x + q^\intercal x + r \\ \text{subject to} & Gx - h \leq 0 \\ & Ax - b = 0 \end{array}$$

Not always convex!! **For convexity** it is required that matrix ***P*** **is positive semidefinite.**

Same constraints of the LP.

More general than LP (a QP with P=0 is an LP).

Classification of convex optimization problems

Quadratically Constrained Quadratic Program (QCQP)

Quadratic objective function **quadratic** inequality constraints and **affine** equality constraints

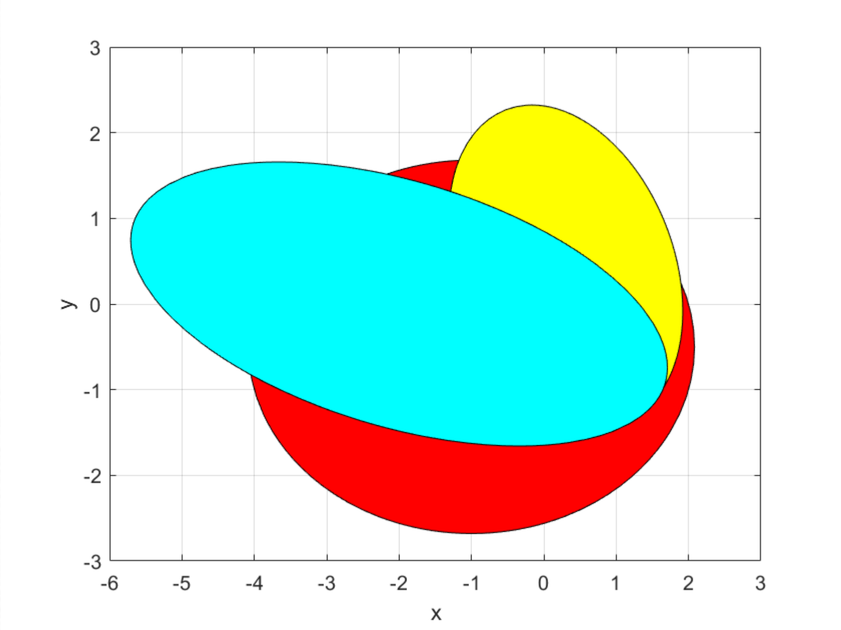
$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \cdots, m \\ & h_i(x) = 0, \quad i = 1, \cdots, p \end{array} \Rightarrow \begin{array}{ll} \text{minimize} & \frac{1}{2} x^\intercal P_0 x + q_0^\intercal x + r_0 \\ \text{subject to} & \frac{1}{2} x^\intercal P_i x + q_i^\intercal x + r_i \leq 0, \quad i = 1, \cdots, m \\ & Ax - b = 0 \end{array}$$

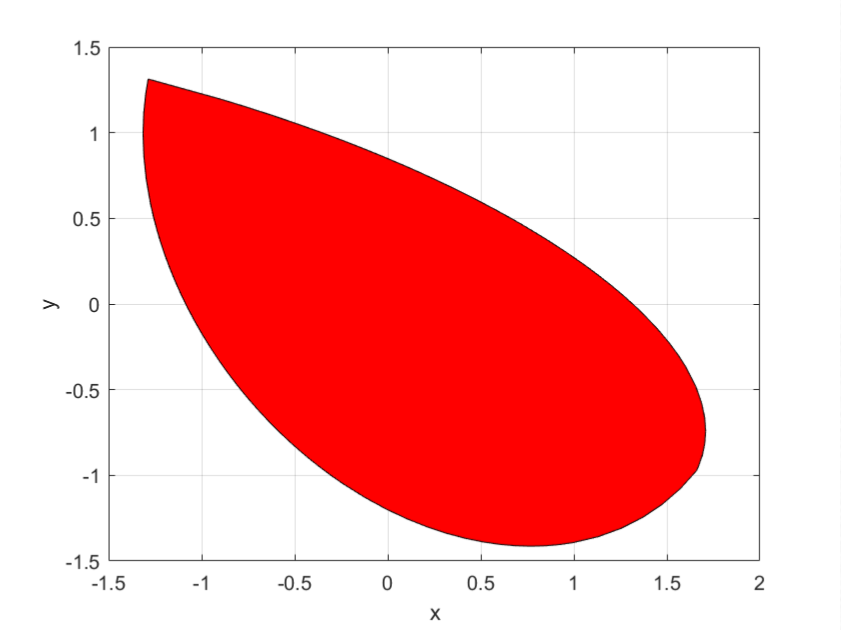
Convex if matrices $P_0, P_i, i = 1, \ldots, m$ **are positive semidefinite.**

More general than QP (it is a QP if P_i=0, $i = 1, \ldots, m$).

Classification of convex optimization problems

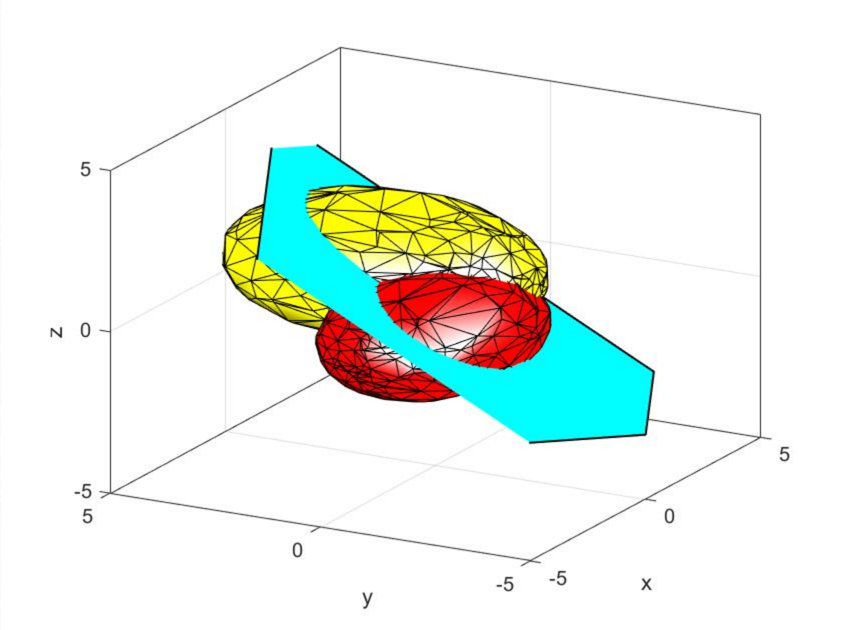
Examples of convex QCQP constraints: 2-dimensions

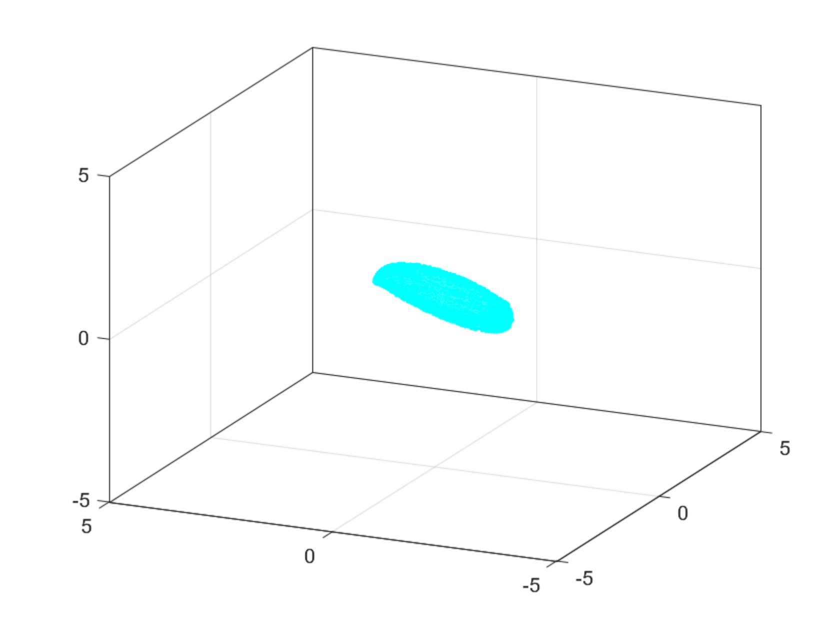




Classification of convex optimization problems

Examples of convex QCQP constraints: 3-dimensions





Classification of convex optimization problems

Second-Order Cone Programming (SOCP)

Linear cost and **second-order cone constraints** and **affine** equality constraints

$$
\begin{array}{ll}
\text{minimize} & f(x) \\
\text{subject to} & g_i(x) \le 0, \quad i = 1, \cdots, m \\
& h_i(x) = 0, \quad i = 1, \cdots, p
\end{array}
\quad \Rightarrow \quad
\begin{array}{ll}
\text{minimize} & f^\intercal x \\
\text{subject to} & ||G_i x + h_i||_2 - r_i^\intercal x - s_i \le 0, \quad i = 1, \cdots, m \\
& Ax - b = 0
\end{array}
$$

It is always convex

OK.. but what is the meaning of «second-order cone» constraints?

Classification of convex optimization problems

Norm cone

The norm cone $C = \{(x, t) : ||x||_p \leq t\}$ is a convex set

The second-order cone is the norm cone for the Euclidean norm $|| \cdot ||_2$

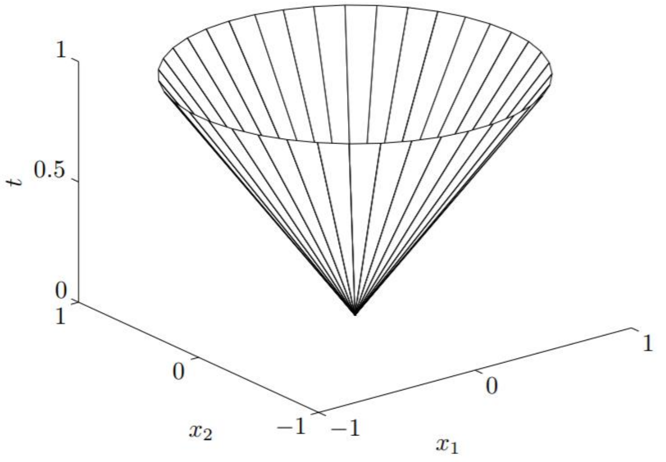


Figure 2.10 $\boxed{\text{Boundary}}$ of second-order cone in $\mathbf{R}^3$, $\{(x_1, x_2, t) \mid (x_1^2 + x_2^2)^{1/2} \leq t\}$.

$\boxed{\textbf{Note: } x_1, x_2, t \text{ are all variables}}$

Classification of convex optimization problems

Second-order cone constraints

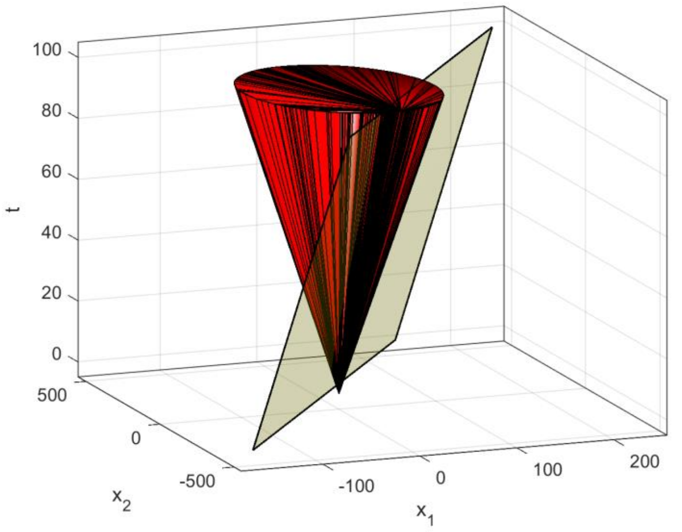
$$||G_i x + h_i||_2 - r_i^\intercal x - s_i \leq 0 \;\Rightarrow\; ||G_i x + h_i||_2 \leq r_i^\intercal x + s_i$$

$$||G_i x + h_i||_2 \leq t$$

$$t = r_i^\intercal x + s_i$$

It is a second order norm cone where variable t has been restricted to be $t = r_i^\intercal x + s_i$

The feasible set is given by the projection onto the original coordinates x of the **intersection** between the **cone** and the **equality constraint**.



Classification of convex optimization problems

Second-order cone constraints

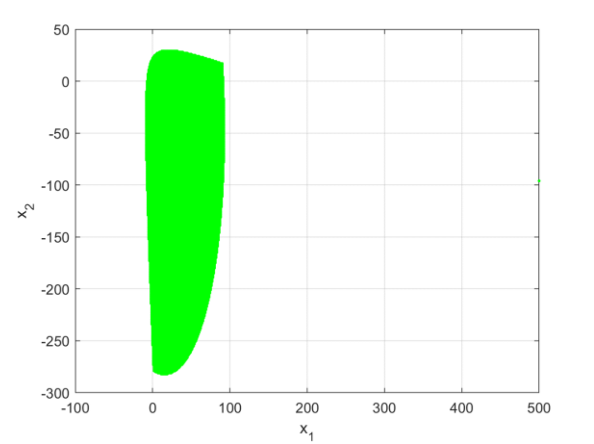
$$\|G_i x + h_i\|_2 - r_i^\intercal x - s_i \le 0 \;\Rightarrow\; \|G_i x + h_i\|_2 \le r_i^\intercal x + s_i$$

$$\|G_i x + h_i\|_2 \le t$$

$$t = r_i^\intercal x + s_i$$

It is a second order norm cone where variable t has been restricted to be $t = r_i^\intercal x + s_i$

The feasible set is given by the projection onto the original coordinates x of the **intersection** between the **cone** and the **equality constraint**.



Classification of convex optimization problems

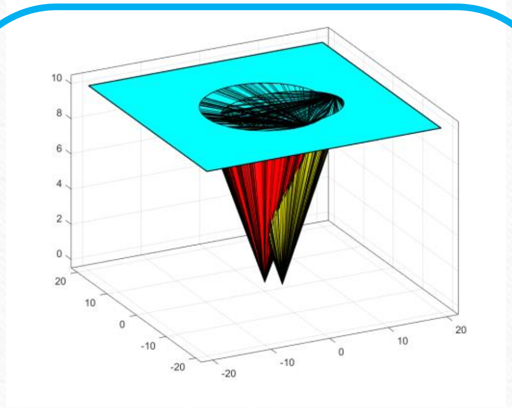
Second-Order Cone Programming (SOCP)

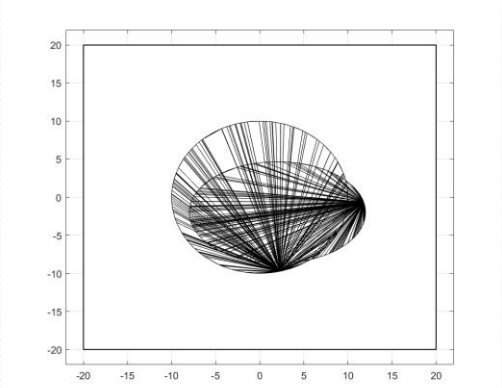
Let recall the SOCP formulation

$$
\begin{array}{ll}
\text{minimize} & f^\intercal x \\
\text{subject to} & ||G_i x + h_i||_2 - r_i^\intercal x - s_i \leq 0, \quad i = 1, \cdots, m \\
& Ax - b = 0
\end{array}
$$

More general constraints than a QCQP. It is a QCQP when r_i is equal to 0.

There is also a way to formulate the quadratic cost of QP/QCQP in the SOCP formulation!





Classification of convex optimization problems

Semi-Definite Program (SDP): it is always a **convex** program

Linear cost, **positive semi-definite cone** constraints and **affine** equality constraints

$$\begin{array}{ll} \text{minimize} & \operatorname{trace} CX \\ \text{subject to} & \operatorname{trace} A_i X = b_i, \quad i = 1, \cdots, m \\ & X \succcurlyeq 0 \end{array}$$

- The variable X is in the set of $n \times n$ symmetric matrices

$$\mathbb{S}^n = \{A \in \mathbb{R}^{n\times n} : A = A^\intercal\}$$

- $X \succcurlyeq 0$ means X is positive semidefinite
- The feasible set is the intersection of an affine set with a convex cone, in this case the positive semidefinite cone

$$\{X \in \mathbb{S}^n : X \succcurlyeq 0\}$$

$$\boxed{\operatorname{trace} CX = \sum_{i,j} c_{i,j} x_{i,j}}$$

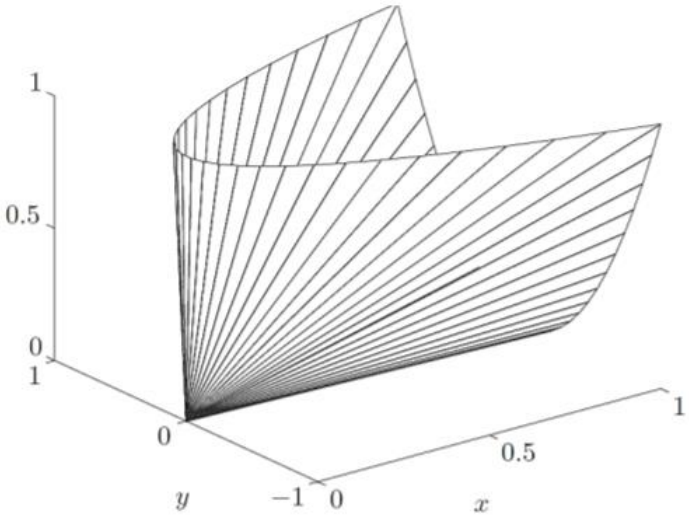
Linear cost w.r.t. the variables of the matrix X (the same holds for the equality constraints!)

Classification of convex optimization problems

Positive semi-definite cone constraints

Example *Positive semidefinite cone in* $\mathbf{S}^2$. We have

$$X = \begin{bmatrix} x & y \\ y & z \end{bmatrix} \in \mathbf{S}^2_+ \iff x \geq 0, \quad z \geq 0, \quad xz \geq y^2.$$

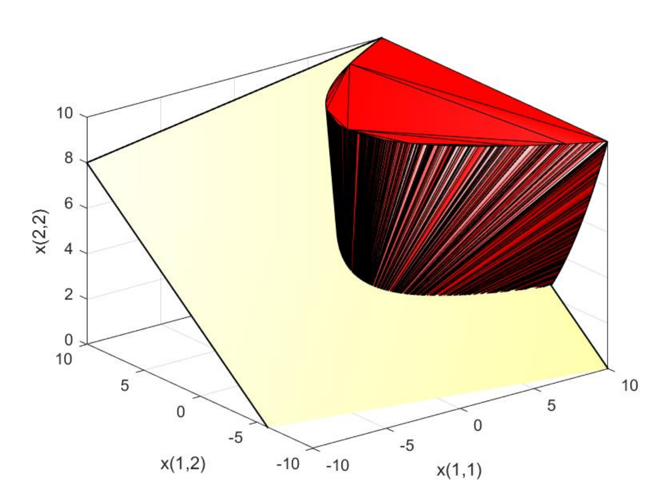


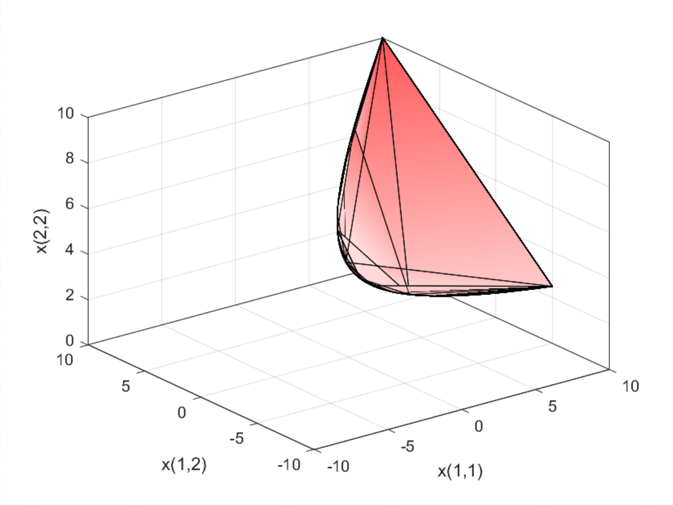
Boundary of positive semidefinite cone in $\mathbf{S}^2$.

Classification of convex optimization problems

Positive semi-definite cone constraints

A further example with also affine equality constraints





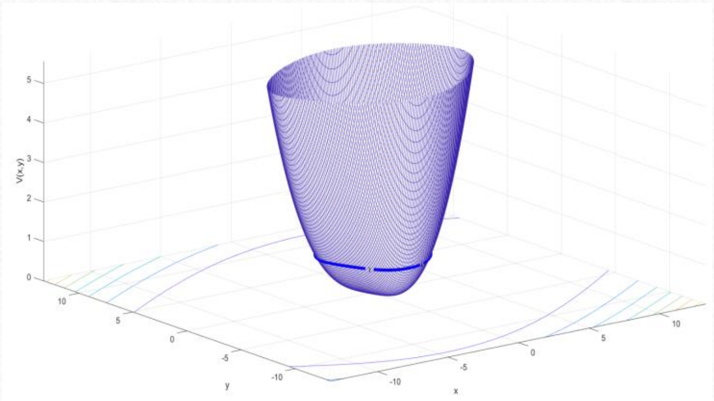
SDP has more general constraints than a SOCP.

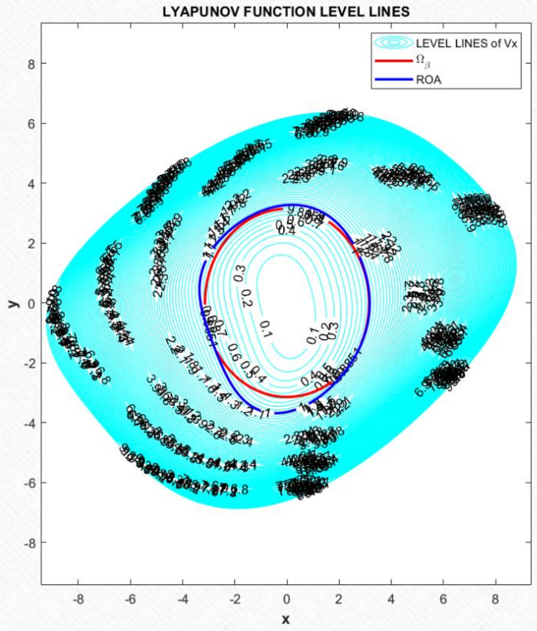
Classification of convex optimization problems

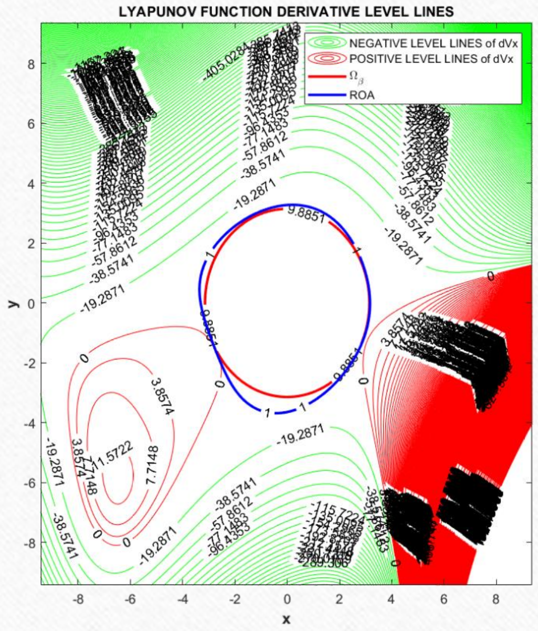
SDP programs can be used to find polynomial Lyapunov functions for polynomial systems!!

$$\begin{cases} \dot{x}(t) = -x(t) + y(t) \\ \dot{y}(t) = 0{,}1x(t) - 2y(t) - x(t)^2 - 0{,}1x(t)^3 \end{cases}$$

Quartic Lyapunov function

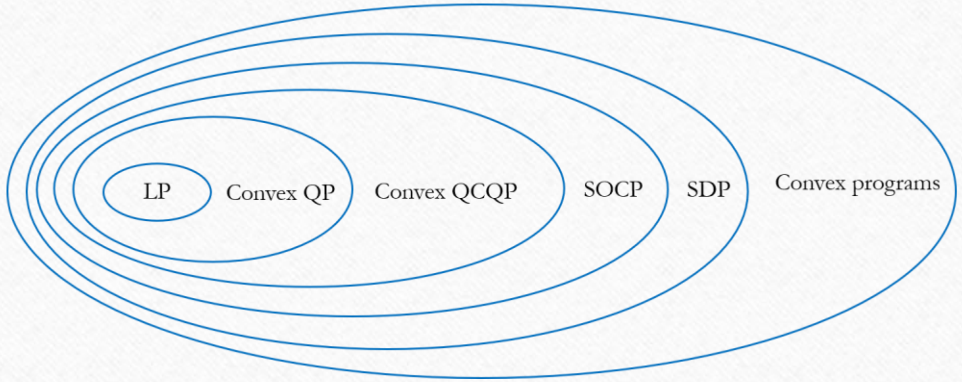






Classification of convex optimization problems

Semidefinite programming has recently emerged to prominence because it admits a new problem type previously unsolvable by convex optimization techniques and because it theoretically subsumes other convex types.



We can solve SDPs and their subsets efficiently with suitable methods!

Extensions (convexity is lost! Further details in future lectures)

If the variables must also verify $x \in \mathbb{Z}^n$ we have **integer** programming (**mixed-integer** programming if only a subset of the variables is constrained to integer values).

Checking convexity: examples

Example 1

Consider the following optimisation problem

$$
\begin{aligned}
\min_{x_1,x_2} \quad & 0.25x_1^2 + 9x_2^2 - 3x_1 \\
& x_1^2 + x_2^2 \leq 10 \\
& x_1^2 + x_2^2 \geq 3
\end{aligned}
$$

1. Indicate if the cost function is convex (motivate the answer).
2. Depict the feasibility domain of the problem.
3. Indicate if the optimisation problem is convex (motivate the answer).

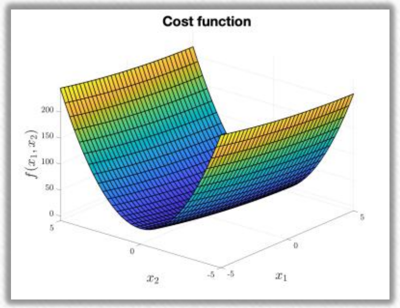
Checking convexity: examples

1. Indicate if the cost function is convex

Example 1
$$\min_{x_1,x_2} \quad 0.25x_1^2 + 9x_2^2 - 3x_1$$
$$x_1^2 + x_2^2 \leq 10$$
$$x_1^2 + x_2^2 \geq 3$$

The cost function is twice differentiable. Thus we can compute the Hessian matrix and check if it is semidefinite positive.

$$\nabla^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1\, \partial x_2} \\ \frac{\partial^2 f}{\partial x_2\, \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix} = \begin{bmatrix} 0.5 & 0 \\ 0 & 18 \end{bmatrix}$$



Since the eigenvalues of $\nabla^2 f$ are both positive ($\lambda_1 = 0.5, \lambda_2 = 18$), we can conclude that the Hessian matrix is positive definite and therefore the cost function is a convex function

Checking convexity: examples

1. Indicate if the cost function is convex

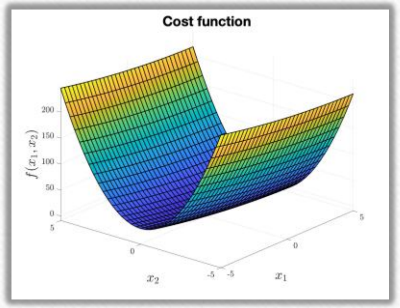
Example 1

$$\begin{aligned} \min_{x_1,x_2} \quad & 0.25x_1^2 + 9x_2^2 - 3x_1 \\ & x_1^2 + x_2^2 \leq 10 \\ & x_1^2 + x_2^2 \geq 3 \end{aligned}$$

Moreover we can see that the cost function is also a quadratic function, where

$$f(x) = x^T Q x + \; c^T x$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad Q = \begin{bmatrix} 0.25 & 0 \\ 0 & 9 \end{bmatrix} \qquad \text{c}= \begin{bmatrix} -3 \\ 0 \end{bmatrix}$$



And Q ≥ 0 thus proving convexity.

Checking convexity: examples

$$\begin{aligned} \min_{x_1,x_2} \quad & 0.25x_1^2 + 9x_2^2 - 3x_1 \\ & x_1^2 + x_2^2 \leq 10 \\ & x_1^2 + x_2^2 \geq 3 \end{aligned}$$

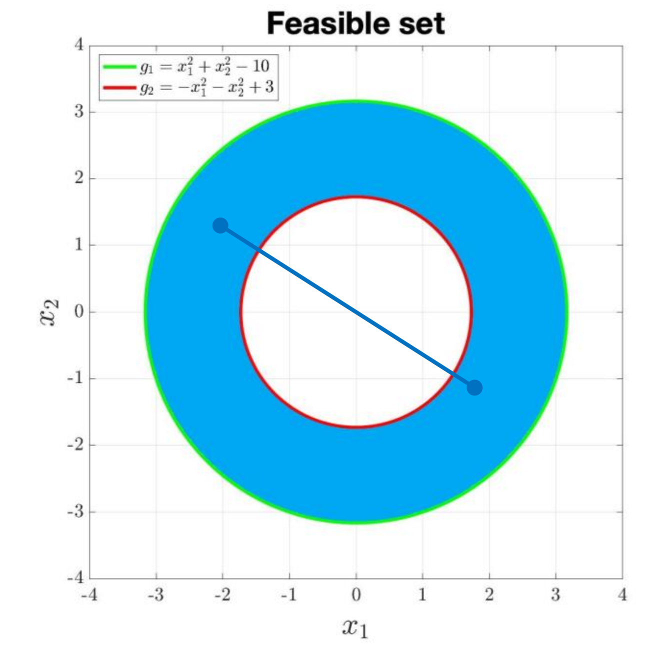
2. Depict the feasibility domain of the problem

Example 1. We now rewrite the inequality constraints in the standard form $g_i(\mathrm{x}) \leq 0$

$$\begin{cases} x_1^2 + x_2^2 \leq 10 \\ x_1^2 + x_2^2 \geq 3 \end{cases} = \begin{cases} x_1^2 + x_2^2 - 10 \leq 0 \\ -x_1^2 - x_2^2 + 3 \leq 0 \end{cases} = \begin{cases} g_1(x) \leq 0 \\ g_2(x) \leq 0 \end{cases}$$

In this example, one can rewrite $g_i(x) = x^T Q_i x + d$ with

$Q_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad Q_2 = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$ → $g_2(x)$ is not a convex function.



Convexity of $g_i(x)$ → convex feasible set. The viceversa is not guaranteed.
What can we do?

Looking at the figure and using the definition of convex set
we **see the set is not convex!**

Checking convexity: examples

3. Indicate if the optimisation problem is convex

Example 1

$$\begin{aligned} \min_{x_1,x_2} \quad & 0.25x_1^2 + 9x_2^2 - 3x_1 \\ & x_1^2 + x_2^2 \leq 10 \\ & x_1^2 + x_2^2 \geq 3 \end{aligned}$$

The optimisation problem is a convex problem if

- $f(x)$ is a convex function (**we minimize!**) and the feasible set is convex.

Since the feasibility domain is not convex, the optimisation problem is **NOT convex.**

- Note: it is a non-convex QCQP!

Checking convexity: examples

Example 2

Consider the following optimisation problem

$$\begin{array}{ll} \max_x & -f(x) \\ & \log(x) \leq 0 \\ & x \geq 0 \end{array}$$

where

$$f(x) = \begin{cases} x^2 & x \leq 0.5 \\ x - 0.25 & x \geq 0.5 \end{cases}$$

1 Indicate if the cost function is convex (motivate the answer).

2 Depict the feasibility domain of the problem.

3 Indicate if the optimisation problem is convex (motivate the answer).

Checking convexity: examples

$$\max_x \quad -f(x) \qquad f(x) = \begin{cases} x^2 & x \le 0.5 \\ x - 0.25 & x \ge 0.5 \end{cases}$$
$$\log(x) \le 0$$
$$x \ge 0$$

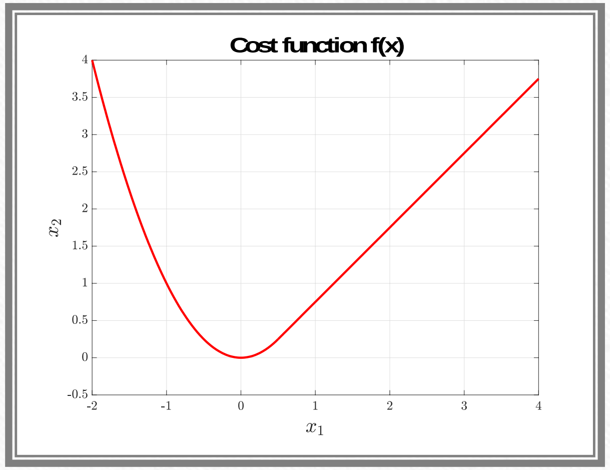
1. Indicate if the cost function is convex

Example 2

$$max_{x\in X} - f(x) \rightarrow min_{x \in X} f(x)$$

$f(x)$ is a continous function. Moreover f is differentiable:

$$\frac{d\ \ f(x)}{dx} = \begin{cases} 2x, & x \le 0.5 \\ 1, & x \ge 0.5 \end{cases} \qquad f_{-}'(0.5) = f_{+}'(0.5) = 1$$



Since $f'(\mathrm{x})$ is an increasing function in its domain → f is convex.

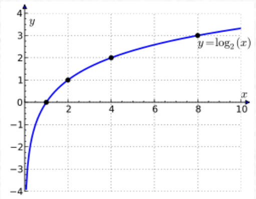
Checking convexity: examples

$$\max_x \quad -f(x) \qquad f(x) = \begin{cases} x^2 & x \le 0.5 \\ x - 0.25 & x \ge 0.5 \end{cases}$$
$$\log(x) \le 0$$
$$x \ge 0$$

2. Depict the feasibility domain of the problem

Example 2

$$\begin{cases} g_1(x) = \log(x) \le 0 \\ g_2(x) = -x \le 0 \end{cases}$$



Concave → Not convex

linear (thus convex)

Convexity of $g_i(x)$ → convex feasible set. The viceversa is not guaranteed.

However, if we analyze $g_1(x)$ we actually get, from 1 and 2

$$\begin{cases} 0 \le x \le 1 \\ x \ge 0 \end{cases}$$

The feasibility domain is the interval set $[0\ 1]$. An interval set is a convex set.

0 1

The feasible domain is convex!

Checking convexity: examples

3. Indicate if the optimisation problem is convex

Example 2

$$\begin{aligned} \max_x \quad & -f(x) \\ & \log(x) \le 0 \\ & x \ge 0 \end{aligned} \qquad f(x) = \begin{cases} x^2 & x \le 0.5 \\ x - 0.25 & x \ge 0.5 \end{cases}$$

- Cost function is a convex function (w.r.t. minimization)
- The feasibility domain is convex

⬇

Optimisation **problem is convex!**

Checking convexity: examples

Example 3

Consider the following optimisation problem

$$\begin{array}{ll} \min_x & f(x) \\ & \cos(x) = 0 \end{array}$$

where

$$f(x) = \begin{cases} x^2 & x \leq 0 \\ x^3 & x \geq 0 \end{cases}$$

1 Indicate if the cost function is convex (motivate the answer).

2 Depict the feasibility domain of the problem.

3 Indicate if the optimisation problem is convex (motivate the answer).

Checking convexity: examples

1. Indicate if the cost function is convex

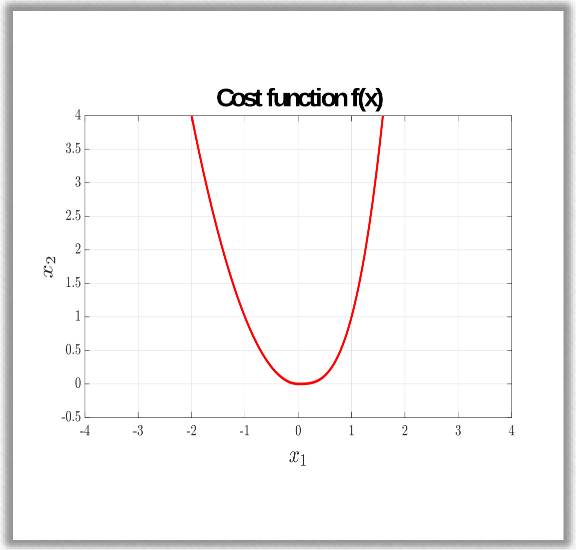
$$\min_x \quad f(x)$$
$$\cos(x) = 0$$

$$f(x) = \begin{cases} x^2 & x \leq 0 \\ x^3 & x \geq 0 \end{cases}$$

Example 3

$f(x)$ is a continous function $\forall x \in X \equiv \mathbb{R}$ and differentiable

$$\frac{df(x)}{dx} = \begin{cases} 2x\ , & x \leq 0 \\ 3x^2, & x \geq 0 \end{cases} \qquad f'_-(0) = f'_+(0) = 0$$



Since f' is an increasing function we can conclude that f **is convex.**

Checking convexity: examples
2. Depict the feasibility domain

$$\min_x \quad f(x)$$
$$\cos(x) = 0$$

$$f(x) = \begin{cases} x^2 & x \leq 0 \\ x^3 & x \geq 0 \end{cases}$$

Example 3: in this example we have only equality constraints $\mathrm{h(x)} = 0$

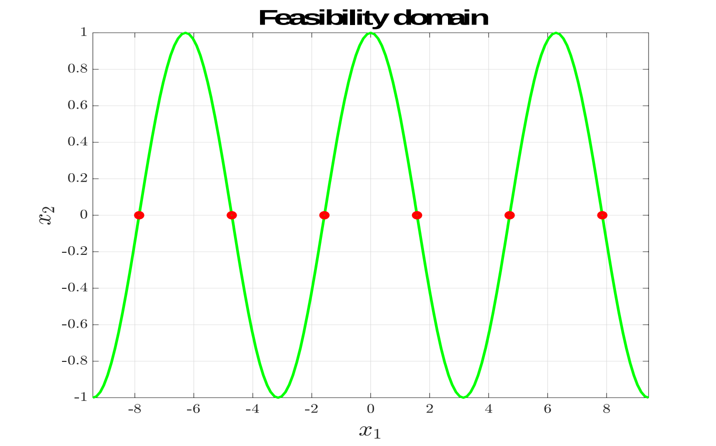
In the absence of inequality constraints, if $h_i(x)$ are affine → convex feasible set.

The viceversa is not guaranteed.

$\cos(x)$ is not affine! What can we do?

Let depict the set. The feasibility domain is in **1D** (the only variable is x) and is characterized by separated points (**red dots** in the figure).

Using the definition of convex set, every segment connecting two points of the set should be contained in it → **The set is not convex!**



Checking convexity: examples

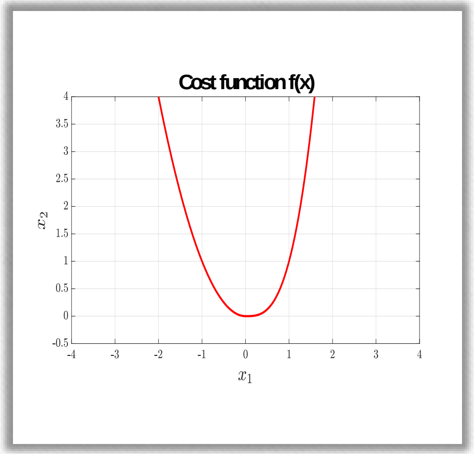
3. Indicate if the optimisation problem is convex

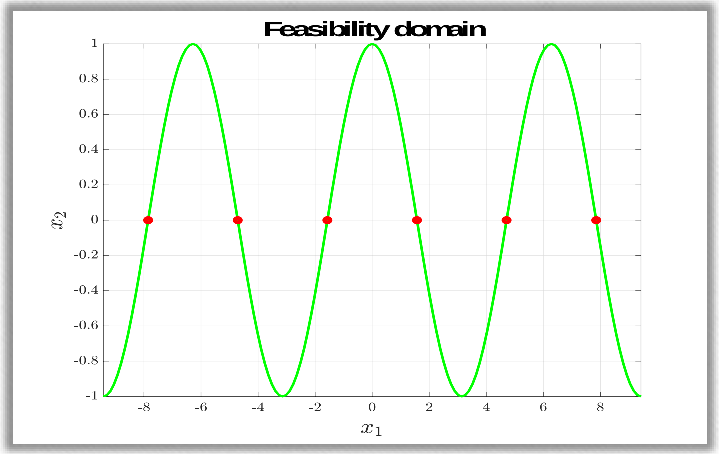
Example 3

$$\min_x \quad f(x)$$
$$\cos(x) = 0$$

$$f(x) = \begin{cases} x^2 & x \leq 0 \\ x^3 & x \geq 0 \end{cases}$$

Since the feasibility domain is not a convex set, the optimisation **problem is NOT convex!**





Checking convexity: examples

Example 4

Consider the following optimisation problem:

$$\begin{aligned} min \quad & f(x) \\ & \cos(x) = 0 \end{aligned}$$

$$f(x) = \begin{cases} x^2, & x \leq 0 \\ x^3 - 1, & x \geq 0 \end{cases}$$

1 Indicate if the cost function is convex (motivate the answer).

2 Depict the feasibility domain of the problem.

3 Indicate if the optimisation problem is convex (motivate the answer).

Checking convexity: example

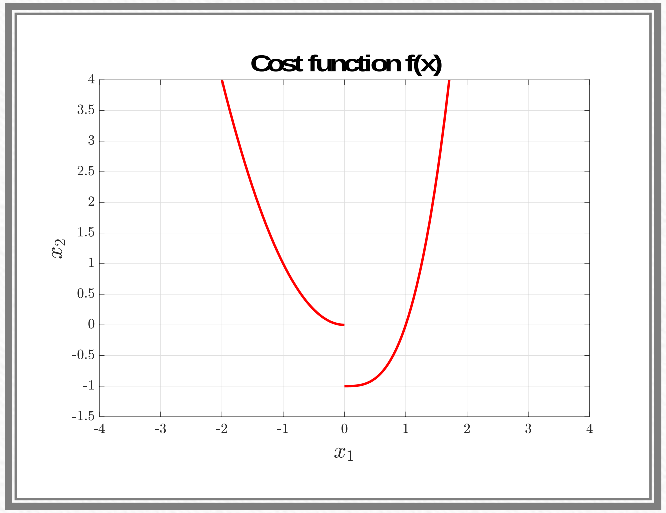
Example 4

$$\min \ f(x) \quad \text{s.t.} \ \cos(x) = 0 \qquad f(x) = \begin{cases} x^2, & x \le 0 \\ x^3 - 1, & x \ge 0 \end{cases}$$

$f(x)$ is a discontinous function → cost function **NOT CONVEX**

The feasibility domain is the same of Example 3 → **NOT convex**

Since both the cost function and the feasibility set are not convex, the **optimization problem is not convex!**



Checking convexity: examples

Example 5

Consider the following linear optimisation problem:

$$\max \quad 24x_1 + 18x_2$$

$$\begin{aligned} s.t. \quad x_1 + x_2 &\leq 40 \\ 4x_1 + 2x_2 &\leq 132 \\ 2x_1 + 4x_2 &\leq 140 \end{aligned}$$

1 Indicate if the cost function is convex (motivate the answer).

2 Depict the feasibility domain of the problem.

3 Indicate if the optimisation problem is convex (motivate the answer).

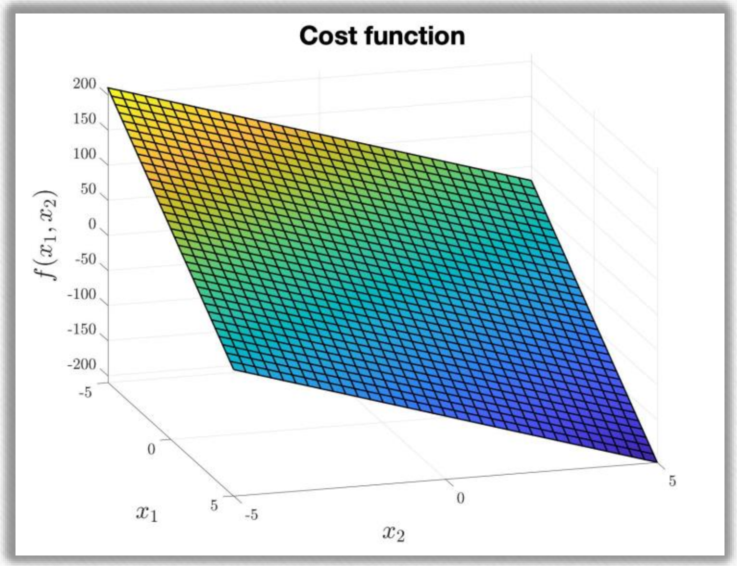
Checking convexity: examples

1. Indicate if the cost function is convex

Example 5

Since the optimisation problem is a maximisation problem we have to **convert it into a minimisation problem**:

$$\begin{aligned} \min \quad & -24x_1 - 18x_2 \\ s.t. \quad & x_1 + x_2 \leq 40 \\ & 4x_1 + 2x_2 \leq 132 \\ & 2x_1 + 4x_2 \leq 140 \end{aligned}$$



Checking convexity: examples

1. Indicate if the cost function is convex

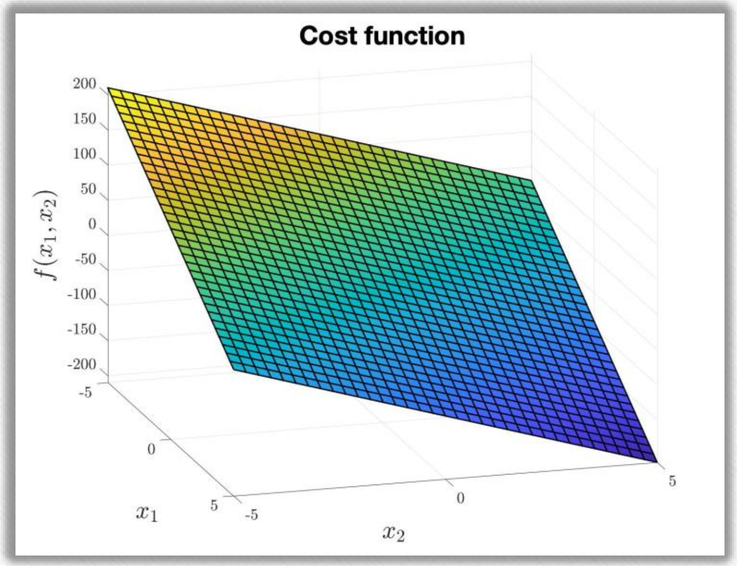
Example 5 $f(x_1, x_2) = -24x_1 - 18x_2$

The cost function is continous in its domain. It is a linear function in the variables x_1, x_2 and **therefore convex.**

Since it is twice differentiable, we could also compute the Hessian matrix

$$\nabla^2 f(x_1, x_2) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Its eigenvalues are null ($\lambda_{1,2} = 0$) → the Hessian is positive semidefinite → **the cost function is a convex function**



(To prove convexity, we could also check if $f'(x)$ is an increasing function)

Checking convexity: examples

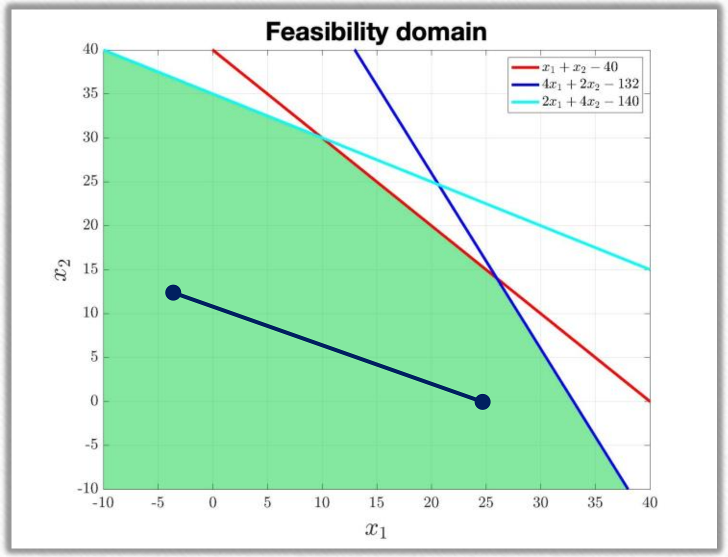
2. Depict the feasibility domain
3. Indicate if the optimisation problem is convex

$$\begin{aligned} \min \quad & -24x_1 - 18x_2 \\ s.t. \quad & x_1 + x_2 - 40 \leq 0 \\ & 4x_1 + 2x_2 - 132 \leq 0 \\ & 2x_1 + 4x_2 - 140 \leq 0 \end{aligned}$$

Example 5: the inequality functions $g_i(x)$ are affine → they are convex→ **the feas. set is convex!**

The feasible set is the **intersection** (in green) of the halfspaces defined by the single constraints (halfspaces).

Looking at the pic, **the set is convex**: the segment connecting any two points inside the green region is entirely contained in the set.



Convex feasibility domain + convex cost function and minimization problem → **The problem is CONVEX!**

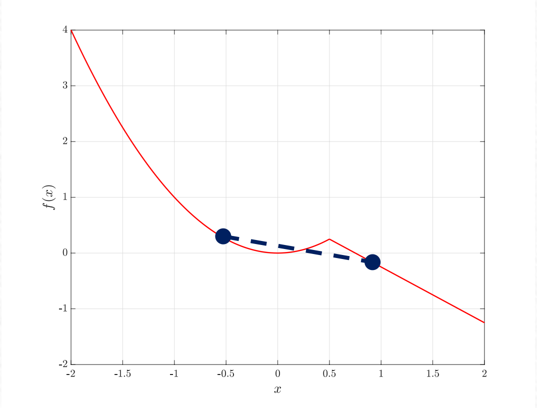
Checking convexity: examples

What if the cost function is not differentiable?

Example 6:

$$\min_{x} \quad f(x)$$
$$\text{subject to} \quad \log(x) \leq 0$$
$$x \geq 0$$

$$f(x) = \begin{cases} x^2, & x \leq 0.5 \\ -x + 0.75, & x \geq 0.5 \end{cases}$$



Differently from Example 2, f is not differentiable in its domain

$$\frac{d\ f(x)}{dx} = \begin{cases} 2x, & x \leq 0.5 \\ -1, & x \geq 0.5 \end{cases} \qquad f_-'(0.5) = 1 \neq f_+'(0.5) = -1$$

So that we cannot rely on conditions based on differentiability for checking convexity

However, we can use the condition for continuous but not differentiable functions → convex if $f_-' \leq f_+'$

It does not hold for $x = 0.5$! Therefore, the **function is not convex!** (Visible in this case also graphically)

Checking convexity: examples

What if the cost function is not differentiable?

Example 7:

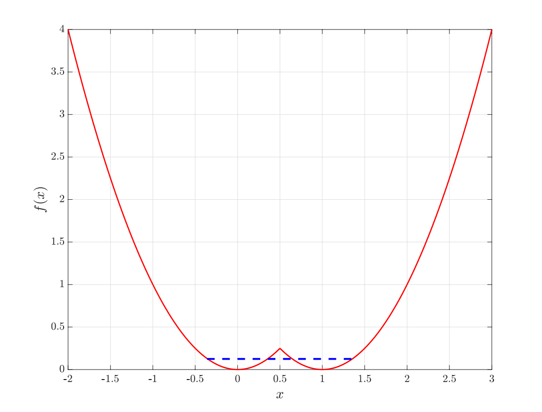
$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & \log(x) \leq 0 \\ & x \geq 0 \end{aligned}$$

$$f(x) = \begin{cases} x^2, & x \leq 0.5 \\ x^2 - 2x + 1, & x \geq 0.5 \end{cases}$$

Still, f is not differentiable in its domain

$$\frac{d\ f(x)}{dx} = \begin{cases} 2x, & x \leq 0.5 \\ 2x - 2, & x \geq 0.5 \end{cases}$$

$$f_{-}'(0.5) = 1 \neq f_{+}'(0.5) = -1$$



As before we cannot rely on conditions based on differentiability for checking convexity and we have to use the condition for continuous but not differentiable functions → convex if $f_{-}' \leq f_{+}'$

It does not hold for $x = 0.5$! Therefore, the **function is not convex!** (Visible in this case also graphically)

Checking convexity: examples

What if the cost function is not differentiable?

Example 8:

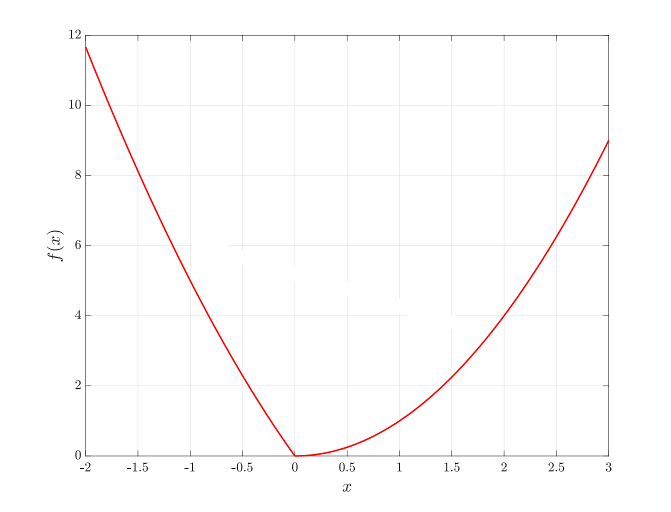
$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & \log(x) \leq 0 \\ & x \geq 0 \end{aligned}$$

$$f(x) = \begin{cases} \frac{5}{6}x^2 - \frac{25}{6}x, & x \leq 0 \\ x^2, & x \geq 0 \end{cases}$$

f is not differentiable in its domain

$$\frac{d\ \ f(x)}{dx} = \begin{cases} \frac{5}{3}x - \frac{25}{6}, & x \leq 0 \\ 2x, & x \geq 0 \end{cases}$$

$$f_-{}'(0) = -\frac{25}{6} \neq f_+{}'(0) = 0$$



We cannot rely on conditions based on differentiability for checking convexity!!

Checking convexity: examples

What if the cost function is not differentiable?

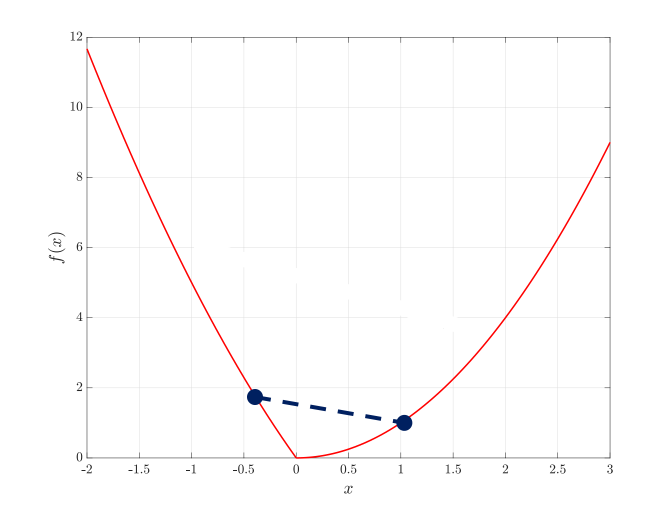
$$f(x) = \begin{cases} \frac{5}{6}x^2 - \frac{25}{6}x, & x \le 0 \\ x^2, & x \ge 0 \end{cases}$$

Example 8:

We can use the condition for continuous but not differentiable functions → convex if $f_-' \le f_+'$

It does hold for $x = 0$! $f_-'(0) = -\frac{25}{6} \le f_+'(0) = 0$

Is this enough? No. The condition needs to hold **everywhere**.



However, since

- for $x < 0$ the function is convex
- for $x > 0$ the function is convex
- for $x = 0$ $f_-' \le f_+'$

$$\frac{d^2 f(x)}{dx^2} = \begin{cases} \frac{5}{3}, & x < 0 \\ 2, & x > 0 \end{cases}$$

One can conclude that the **function is convex!** (Visible in this case also graphically)