Advanced automation and control

Prof. Davide M. Raimondo - Prof. Antonella Ferrara
Dipartimento di Ingegneria Industriale e dell'Informazione
Università degli Studi di Pavia
davide.raimondo@unipv.it antonella.ferrara@unipv.it

Course schedule

Two modules

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

Lectures

- Thursday (9-11), Aula 8
- Friday (14-16), Aula 5

Visiting Prof. G. Ciaramella (a.y. 2018/2019)

Dates: 14/03, 15/03, 21/03, 22/03, 28/03, 29/03

Scripts will be provided after each class.

Laboratories (4)

- 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)
Antonella Ferrara: floor F (antonella.ferrara@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
- 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
Date/time/room on the website of the Faculty of Engineering
No graphic or programmable calculators are allowed

Registration to exams: Through the university website.

Why optimization?

Useful in many contexts

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

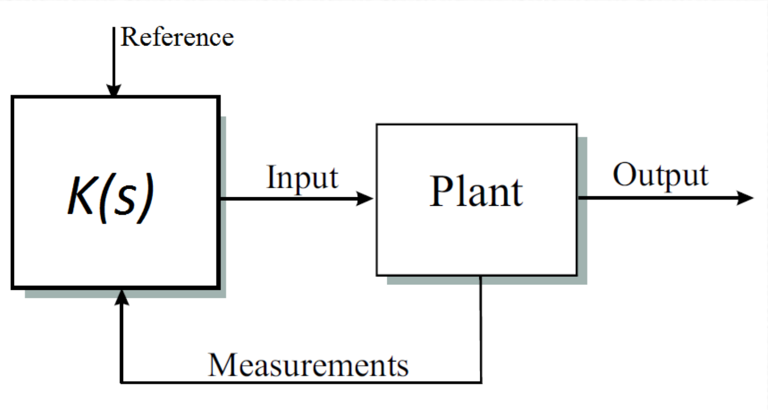
Why optimization?

Useful in many contexts

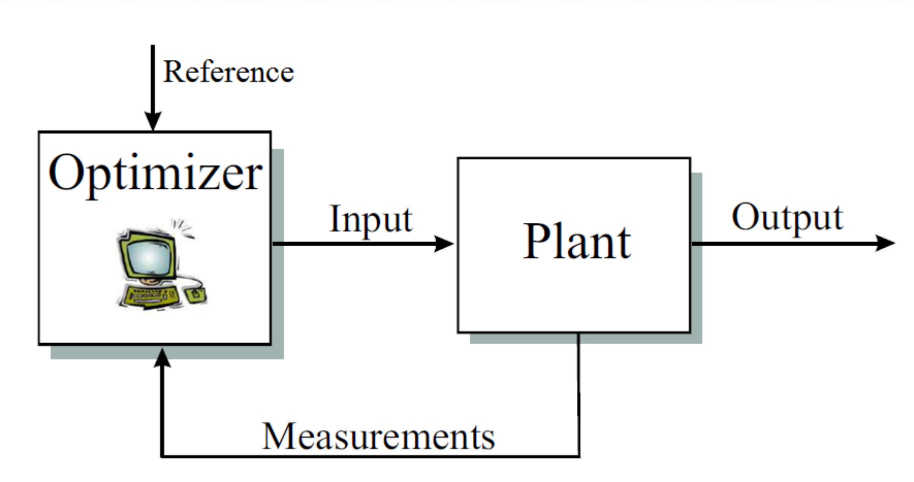
- Control
- Identification
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Control

Classic control



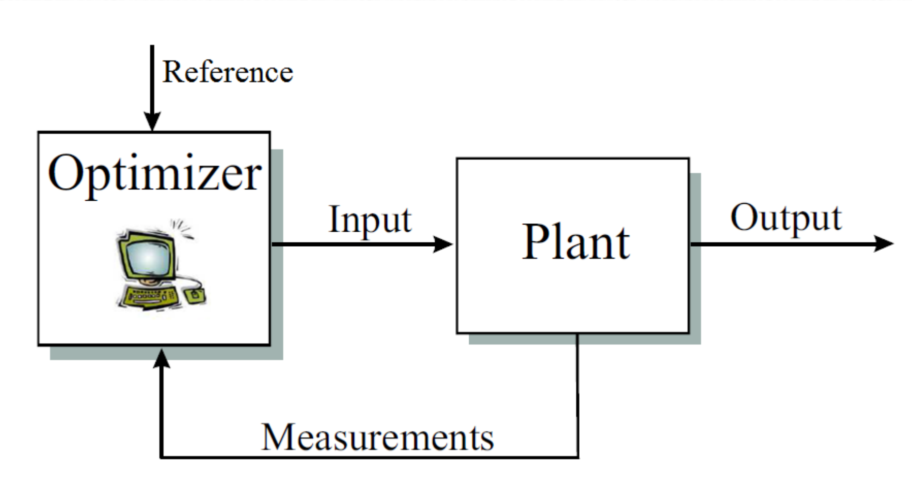
Optimization-based control

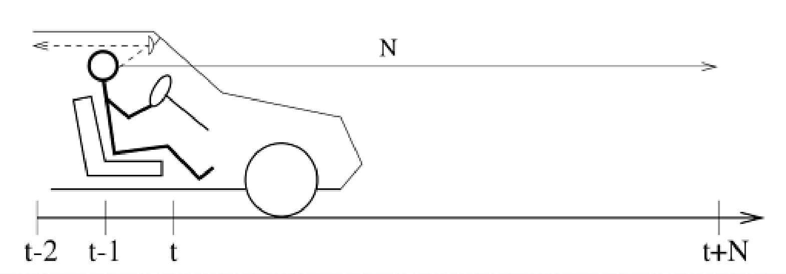


The classic controller is replaced by an **optimization** algorithm that runs on-line

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

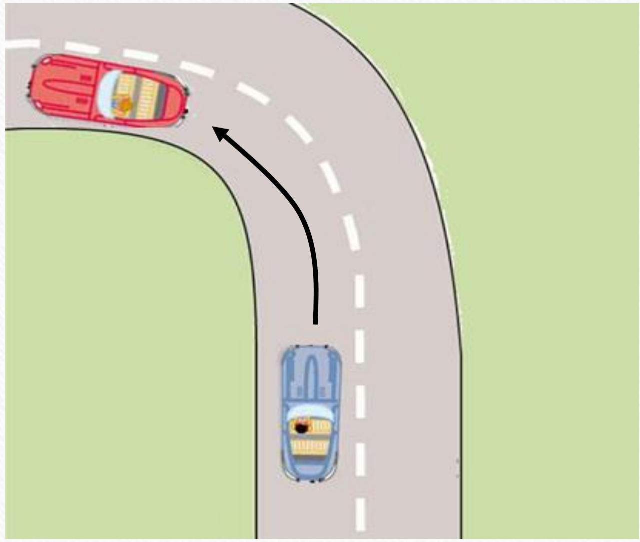
Driving a car

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

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

Identification

Given an input-output data set (u, y), consider the following system

$\sum_{j=1}^{n} X_{ij}\beta_j = y_i \ (i = 1, \cdots, m)$

with m linear equations in n $(m > n)$ unknown parameters β_j, $(j = 1, \cdots, n)$.

Let $X_{i1} = 1$, for all $i = 1, \cdots, m$. For each i, X_{ij} is a predefined function of the input u_i (i.e. $u_i, u_i^2, \cdots$).

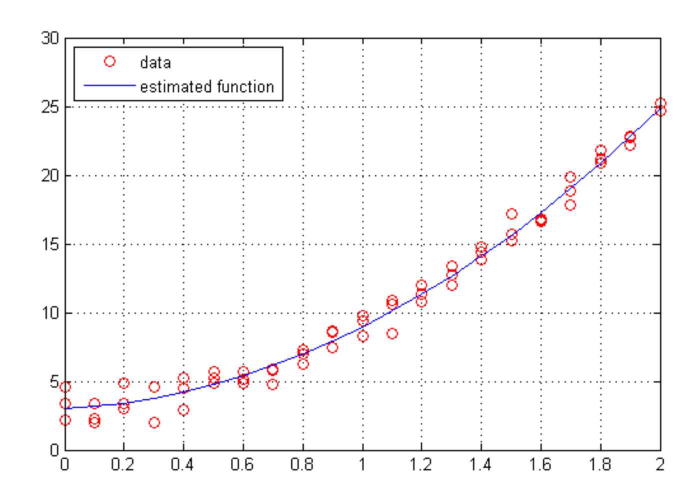
Define $\boldsymbol{X} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ X_{m1} & X_{m2} & \cdots & X_{mn} \end{bmatrix}$, $\boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}$, $\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$

Than, the problem above can be rewritten as $\boldsymbol{X\beta} = \boldsymbol{y}$.

Since the data is affected by noise the equality does not hold in general.

We aim to find the set of parameters which provides the least square error $\hat{\boldsymbol{\beta}} = \operatorname{argmin}_{\boldsymbol{\beta}} ||\boldsymbol{X\beta} - \boldsymbol{y}||^2$

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



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

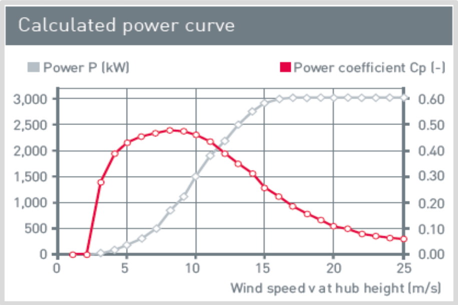
Why optimization?

Useful in many contexts

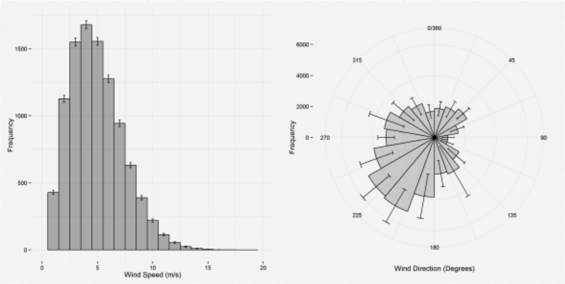
- Control
- Identification
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Optimal placement/sizing

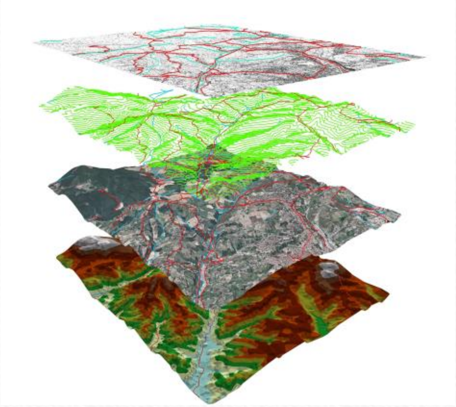
Choose the **number** and the **location** of a set of **wind turbines** in order to maximize the return of investment of a wind farm. Several elements need to be taken into account



Power Curve



Wind distribution

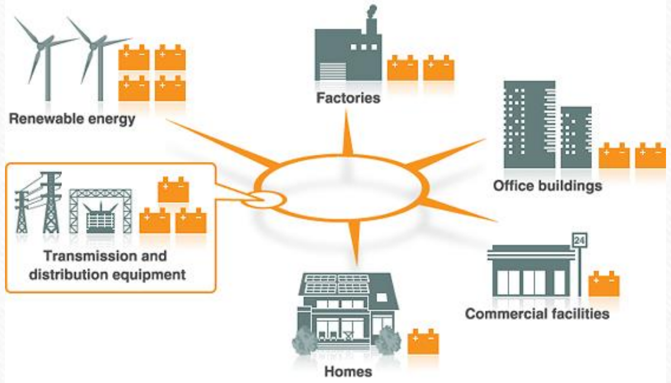


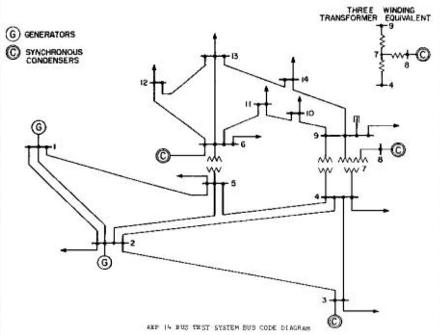
Geographic information

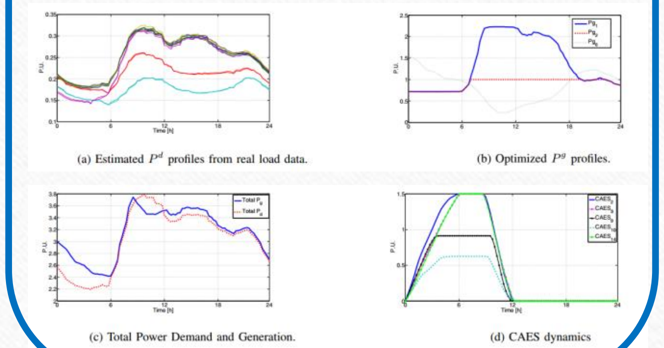
Wake effect

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







(a) Estimated P^d profiles from real load data.

(b) Optimized P^g profiles.

(c) Total Power Demand and Generation.

(d) CAES dynamics

Why optimization?

Useful in many contexts

- Control
- Identification
- 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

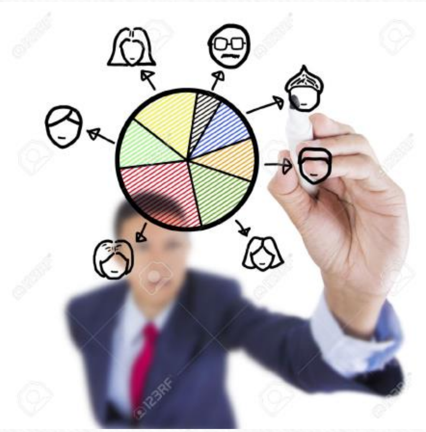
K. Nolde
Apex Optimization GmbH,
c/o Automatic Control Laboratory,
ETH Zurich, 8092 Zurich,
Switzerland,
nolde@control.ee.ethz.ch

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 over-staffing 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

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

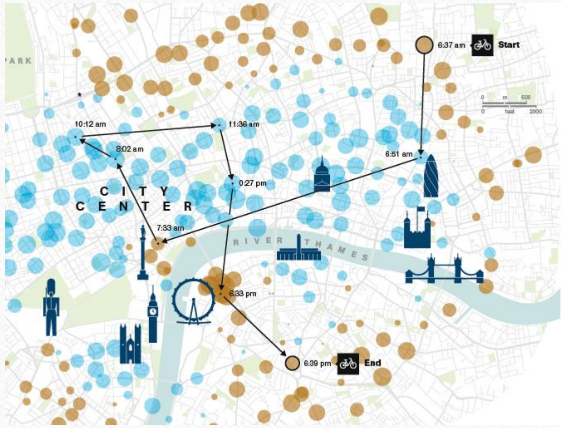
Routing/redistribution problems

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



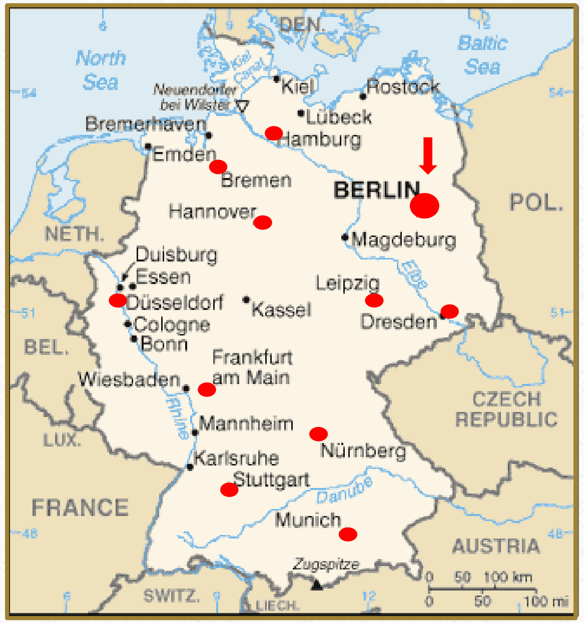
Abstract

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…

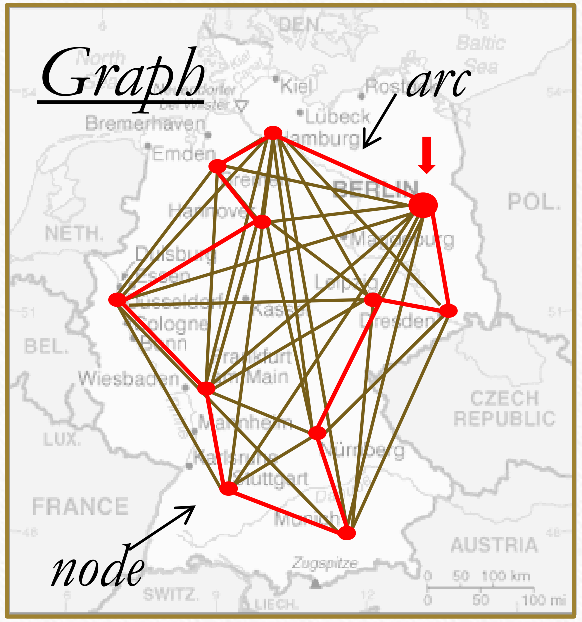


What is the shortest route that visits each city once and only once?

Routing/redistribution problems

The Travelling Salesman Problem

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



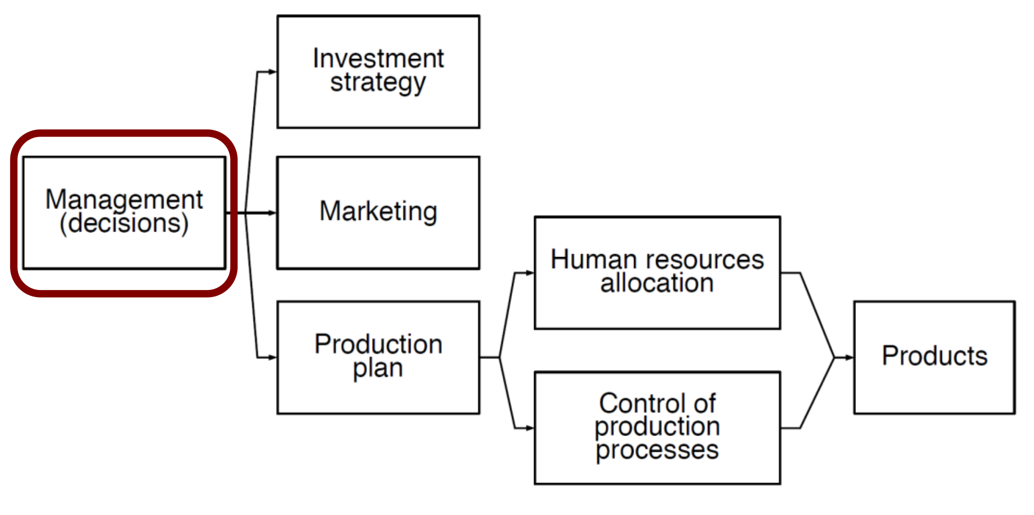
The objective function is the minimization of the cost of the path

Why optimization?

Useful in many contexts

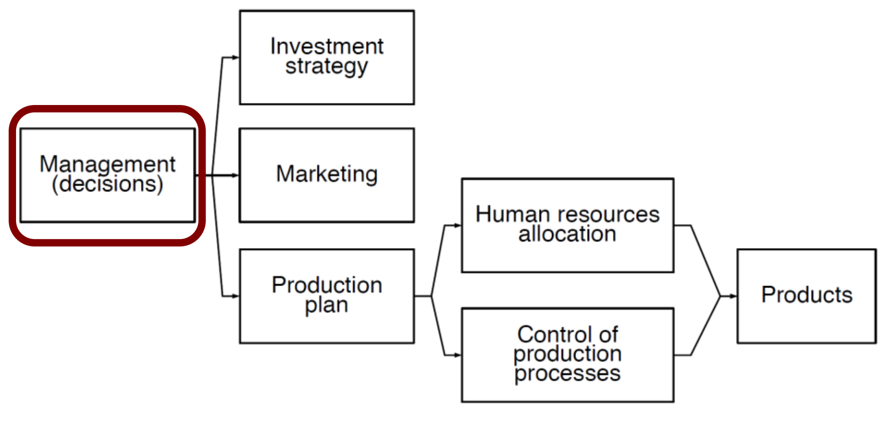
- Control
- Identification
- Management
  - Optimal placement/sizing
  - Resources allocation
  - Routing/redistribution problems
  - Planning of production processes
  - …

Planning of production processes



Management science: optimal decisions for complex problems

Planning of production processes



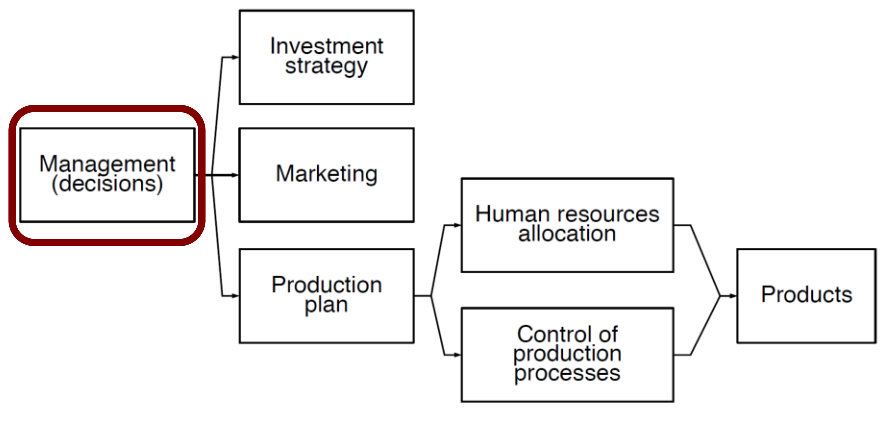
Management: decisions can be either "instinctive" or structured

- "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

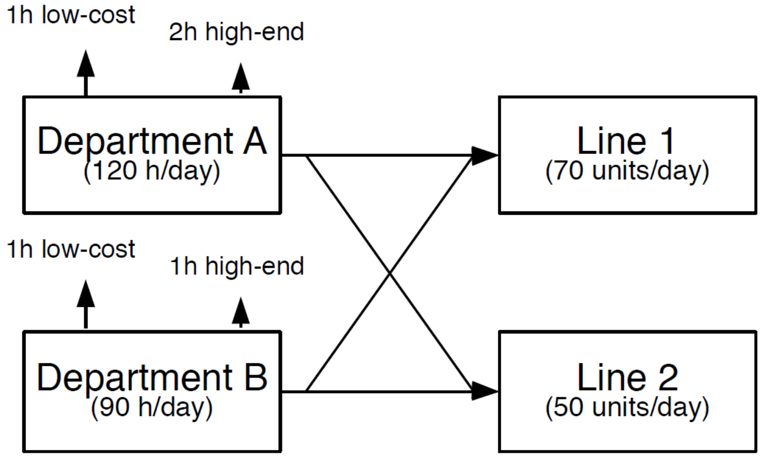
A company manifactures two radio models (low-cost and high-end) and produces two components

- 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

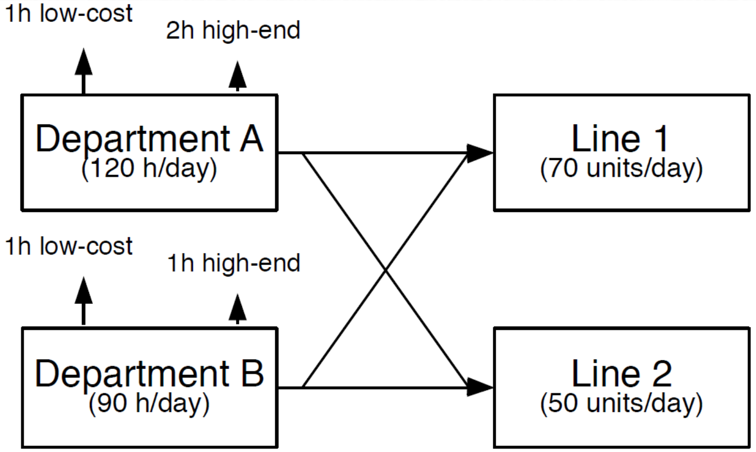


Profits: 20 Euro for a low-cost radio and 30 Euro for a high-end radio

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

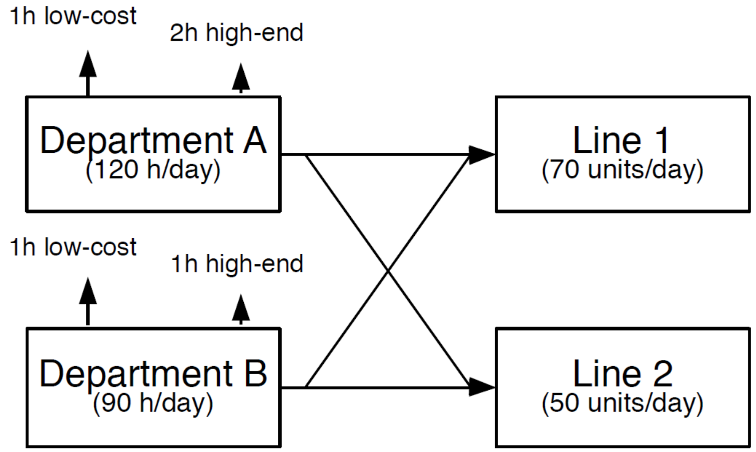


Instinctive (and greedy) manager: higher profits for high-end models ⇨ maximize their production (50 units/day)

- 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

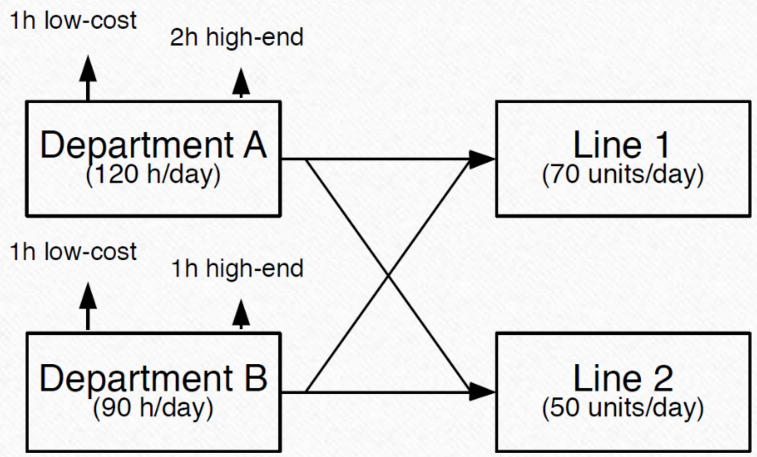


Smart manager: 60 low-cost models and 30 high-end models

- 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



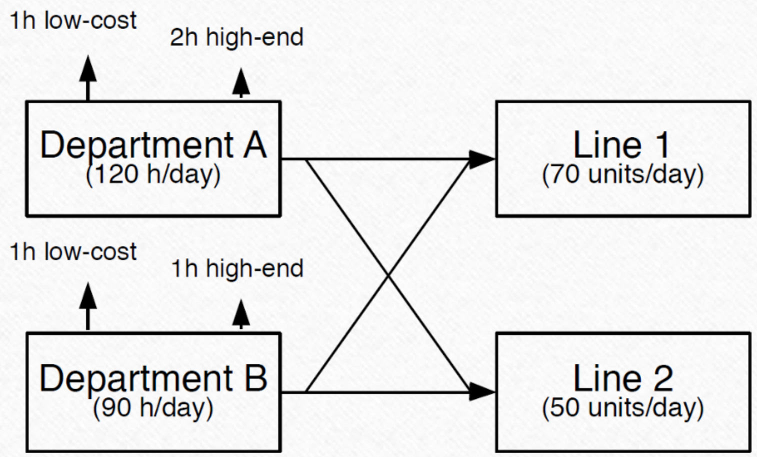
Decisions taken by the smart manager are **optimal** (profits cannot increase)

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



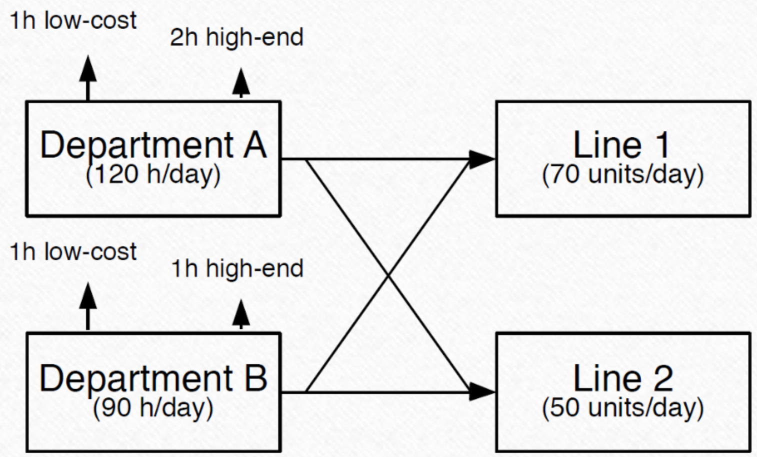
How the manager came up with this plan ? How can we certify it is optimal ?

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

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

Answers to both questions in the next lectures !

Optimization

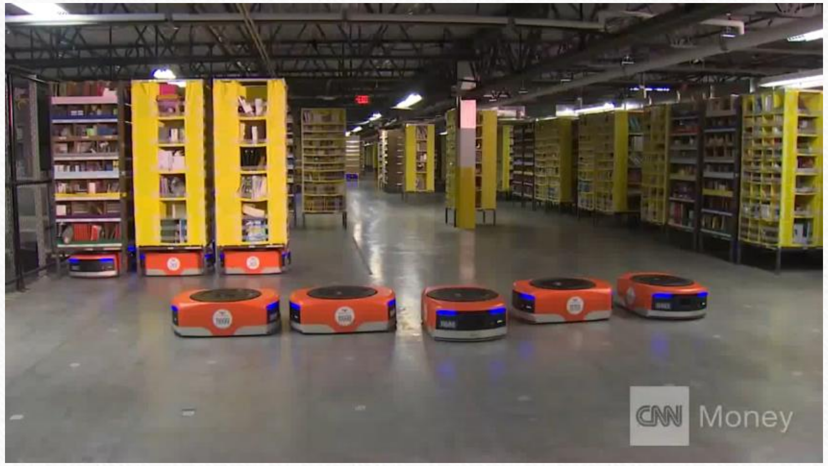
Mathematical formalization

+

Optimized 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

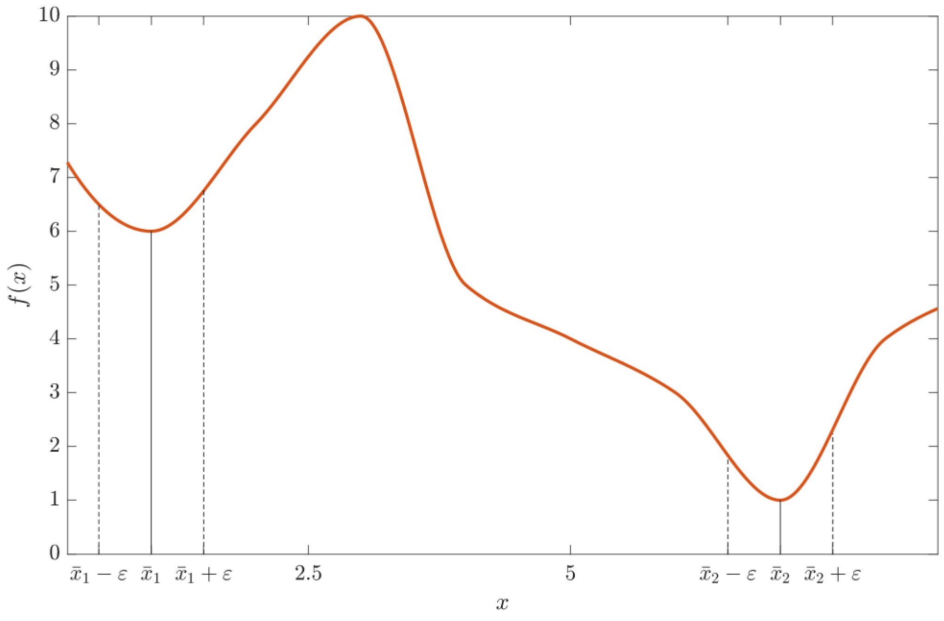
Introduction to optimization

Basic problem: $\displaystyle \min_{\substack{g_i(x)\leq 0 \\ i=1,2,\cdots,m}} f(x)$

- Variables: $x = [x_1, \cdots, x_n]^\top$
- Constraints: $g_i : \mathbb{R}^n \to \mathbb{R}, i = 1, 2, \cdots, m$
- Feasible region: $X = \{x \in \mathbb{R}^n : g_1(x) \leq 0, \cdots, g_m(x) \leq 0\}$
- Feasible solution or feasible point: $x \in X$
- Objective function (or cost): $f : X \to \mathbb{R}$

Introduction to optimization

Basic problem: $\min_{\substack{g_i(x)\leq 0 \\ i=1,2,\cdots,m}} f(x)$



- $x^* \in X$ 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

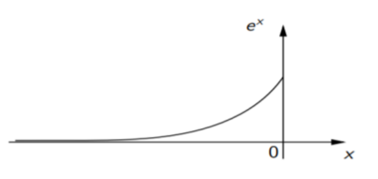
$$\exists \varepsilon > 0 : \forall x \in X, ||x-\bar{x}|| < \varepsilon \Rightarrow f(\bar{x}) \leq f(x)$$

In the figure: $\bar{x}_1$ and $\bar{x}_2$ are respectively a local and the global minimum point.

Introduction to optimization

Basic problem: $\displaystyle \min_{\substack{g_i(x)\leq 0 \\ i=1,2,\cdots,m}} f(x)$

- In some cases, the basic problem can be
  - **infeasible** (if $X = \emptyset$)
  - **unbounded** (if $\forall k < 0 \ \exists x \in X : f(x) < k$)
- 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. $\displaystyle \min_{x \leq 0} e^x \quad x \in \mathbb{R}$ ➡



Introduction to optimization

Basic problem: $$\min_{\substack{g_i(x)\leq 0 \\ i=1,2,\cdots,m}} \quad f(x)$$

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

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

Introduction to optimization

Maximum problems: $$\max_{\substack{g_i(x)\leq 0\\ i=1,2,\cdots,m}} f(x)$$

- The problem is **unbounded** i $\forall k > 0 \ \exists x \in X : f(x) > k$
- $x^* \in X$ is an optimal solution (**global** maximum point) if

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

- $\bar{x} \in X$ is a local optimal solution (**local** maximum point) if

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

Conversion in the basic problem form

- Conversion maximum/minimum: $\boxed{\max_{x \in X} f(x) = -\min_{x \in X} -f(x)}$

optimal solutions
are the same for
both problems

Conversion in the basic problem form

- Conversion maximum/minimum: $\boxed{\max_{x \in X} f(x) = -\min_{x \in X} -f(x)}$ ⇐ optimal solutions are the same for both problems

- Conversion from $\leq$ to $\geq$ in the constraints

$$\{x \in \mathbb{R}^n : g(x) \geq 0\} = \{x \in \mathbb{R}^n : -g(x) \leq 0\}$$ ⇐ the feasible region does not change

Conversion in the basic problem form

- Conversion maximum/minimum: $\boxed{\max_{x \in X} f(x) = -\min_{x \in X} -f(x)}$

optimal solutions are the same for both problems

- Conversion from $\leq$ to $\geq$ in the constraints

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

the feasible region does not change

- Conversion from equalities to inequalities in the constraints

$$\{x \in \mathbb{R}^n : g(x) = 0\} = \{x \in \mathbb{R}^n : g(x) \leq 0, g(x) \geq 0\}$$

An equality constraint is replaced by two inequality constraints

Classes of optimization problems

Basic problem: $\min_{\substack{g_i(x) \le 0 \\ i=1,2,\cdots,m}} f(x)$

- $f(x)$ is quadratic if $f(x) = x^\top Q x + c^\top x$ (Q matrix, c vector)
- $f(x)$ is linear if $f(x) = c^\top x$
- $f(x)$ is affine if $f(x) = c^\top x + b$ (b constant)

Classes of optimization problems

Basic problem: $\min\limits_{\substack{g_i(x)\leq 0\\ i=1,2,\cdots,m}} f(x)$

- $f(x)$ is quadratic if $f(x) = x^\top Q x + c^\top x$ (Q matrix, c vector)
- $f(x)$ is linear if $f(x) = c^\top x$
- $f(x)$ is affine if $f(x) = c^\top x + b$ (b constant)

Notable problems for which efficient algorithms exist

- $f(x)$ is convex and $g_i(x)$ are convex → **convex** programming
- if $f(x)$ is quadratic and $g_i(x)$ are affine → **quadratic** programming
- if $f(x)$ is linear and $g_i(x)$ are affine → **linear** programming

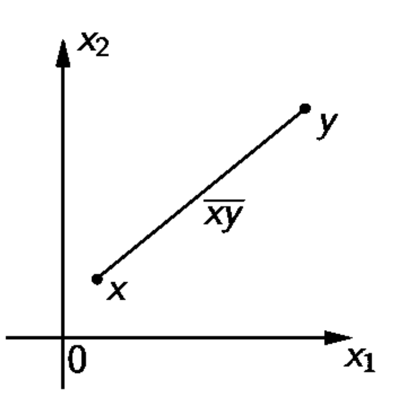
If the variables must also verify $x \in \mathbb{Z}^n$ we have **integer** programming (**mixed-integer** programming if only a subset of the variables is constrained to integer values)

Convex programming

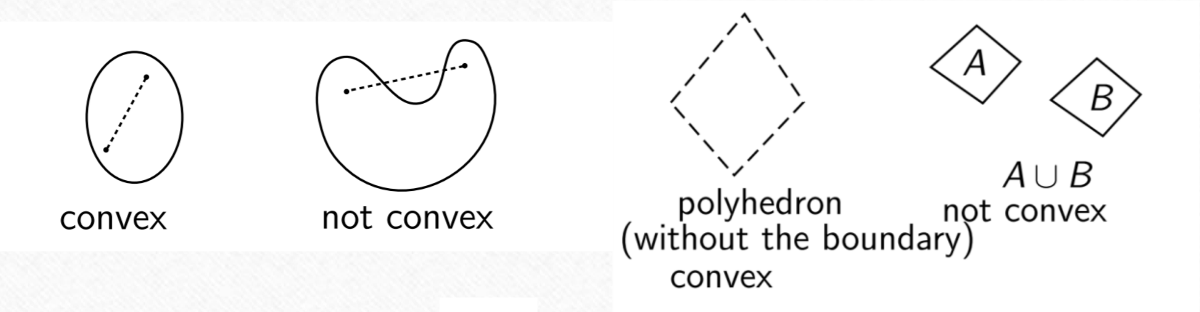
Definition: given two points $x, y \in \mathbb{R}^n$, the set

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

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 $\overline{xy} \in X$

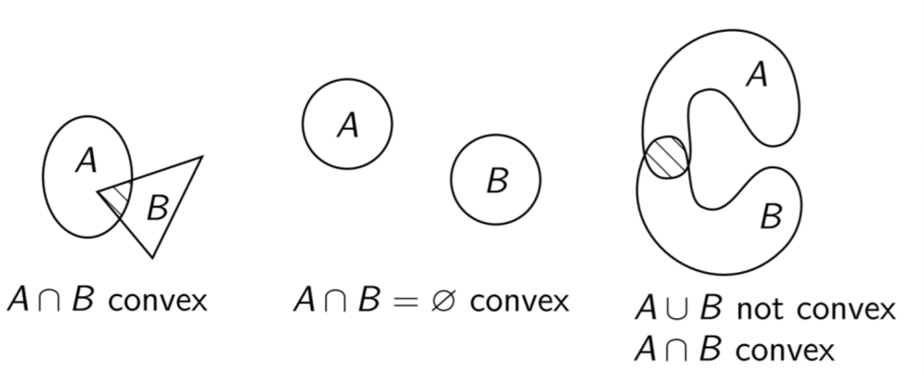


$\mathbb{R}^n$ is convex

Convexity and intersection

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.



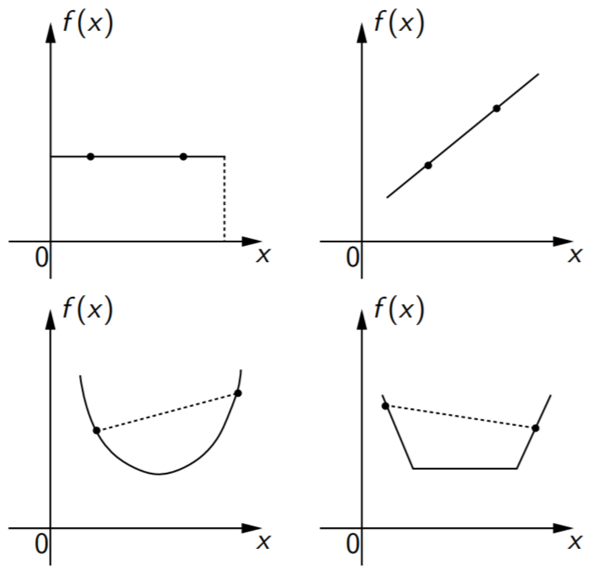
Attention: the union of two convex sets is not convex in general!

Convex functions

Definition: a function $f : X \to \mathbb{R}$ on a convex set $X \subseteq \mathbb{R}^n$ is convex if $\forall x, y \in X$ and $\lambda \in [0, 1]$ one has $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$

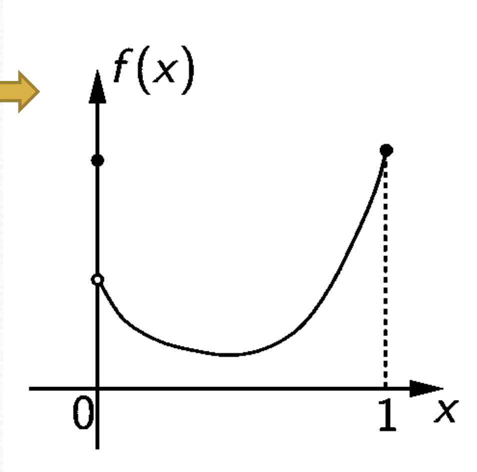
Convex functions

Definition: a function $f : X \to \mathbb{R}$ on a convex set $X \subseteq \mathbb{R}^n$ is convex if $\forall x, y \in X$ and $\lambda \in [0,1]$ one has $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y)$



Convexity and smoothness

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



Theorem – convexity test for smooth functions

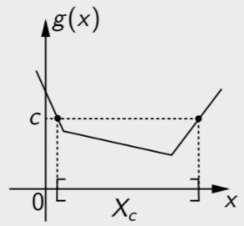
Let $X \subseteq \mathbb{R}^n$ be open and convex and let $f : X \to \mathbb{R}$ be a $\mathcal{C}^2$ function. Then, f is convex only if the Hessian matrix $H(x)$ is positive semidefinite $\forall x \in X$. In particular, if $X \subseteq \mathbb{R}^n$ and $f \in \mathcal{C}^2$, then f is convex only if

$$\frac{d^2 f(x)}{dx^2} \geq 0, \forall x \in X$$

Convex functions and sets

Theorem

Let $g : \mathbb{R}^n \to \mathbb{R}$ be a convex function and take $c \in \mathbb{R}$. Then, the level set $X_c = \{x \in \mathbb{R}^n : g(x) \leq c\}$ 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 one has that $g(z) \leq \lambda g(x) + (1-\lambda)g(y)$. Since $x, y \in X_c$ one has

$$g(z) \leq \lambda g(x) + (1-\lambda)g(y) \leq \lambda c + (1-\lambda)c = c$$

that implies $z \in X_c$.

Convex functions and sets

Key corollary

Consider the optimization problem

$$\min_{\substack{g_i(x)\leq 0\\ i=1,2,\ldots,m}} \quad f(x)$$

If functions $g_i(x)$, $i = 1, 2, \ldots, m$ are convex, then the feasibile region is convex.

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

In convex programming, the feasible region is convex

Fundamental theorem of convex programming

Theorem

If $\tilde{x} \in X$ is a *local optimal solution* for the convex programming problem

$$\{\min f(x) : g_i(x) \leq 0, i = 1, 2, \ldots, m\}$$

then $\tilde{x}$ is an *optimal solution*.

Remarks

Often one tries to transform a programming problem into a convex programming problem by performing suitable changes of variables

Fundamental theorem of convex programming

Remarks

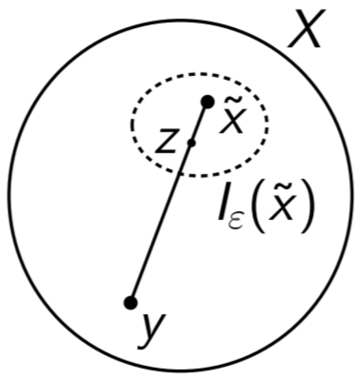
The optimization problem $\{\max f(x) : g_i(x) \leq 0, i = 1, 2, \ldots, m\}$ is not a convex programming problem even if f and g_i, $i = 1, 2, \ldots, m$ are convex. Indeed, it is equivalent to $\{-\min -f(x) : g_i(x) \leq 0, i = 1, 2, \ldots, m\}$ where the function $-f(x)$ is concave.

Notable exception: $f(x)$ linear.

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

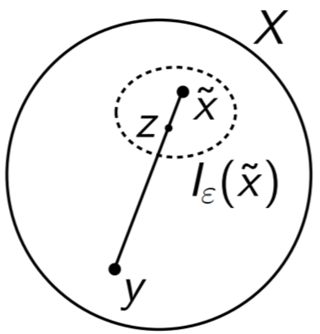
- choosing λ sufficiently close to 1 guarantees $z \in I_\epsilon(\tilde{x})$
- choosing $\lambda \neq 1$ guarantees $z \neq \tilde{x}$

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

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