COMPUTATIONAL METHODS IN OPTIMIZATION

Sep 19, 2023

Today: → Logistics
→ Intro to linear programming

Schedule:

Lectures (Mondays 8.30am - 10am
labs exercises (Tuesdays: 3.45pm - 5.15pm

Until mid-October:
Lectures + Exercises
⇒ Exam (70% of the grade)

Afterwards:
Some lectures, as many labs as possible
⇒ Course project (30% of the grade)

INTRODUCTION AND BASICS OF LINEAR PROGRAMMING

↳ Goal of the course: Study algorithms that work for solving real-world problems

Key components

- Modeling phase: writing the problem as a (convex) "simple" optimization problem
- Algorithmic phase: design numerical algorithms to solve the problem
- Implementation phase: Python, comparison with existing libraries
- Possibly: Loop between the phases!

Today's optimization solvers (for nice convex programs):

→ Can solve problems with 1000s / 10^6s of variables (depending on the problem class) in seconds on a laptop!

Ex: Gondzio & Grothey (2005) solved a quadratic program from finance with

1 010 000 000 variables

and 353 000 000 constraints

They did that by

→ Relying a powerful class of algorithms (Interior-Point Methods)

→ Exploiting the problem structure

(1) Linear programming

↳ Most basic optimization model possible, and the most successful one

Ex) DCOPF (DC Optimal Power Flow)

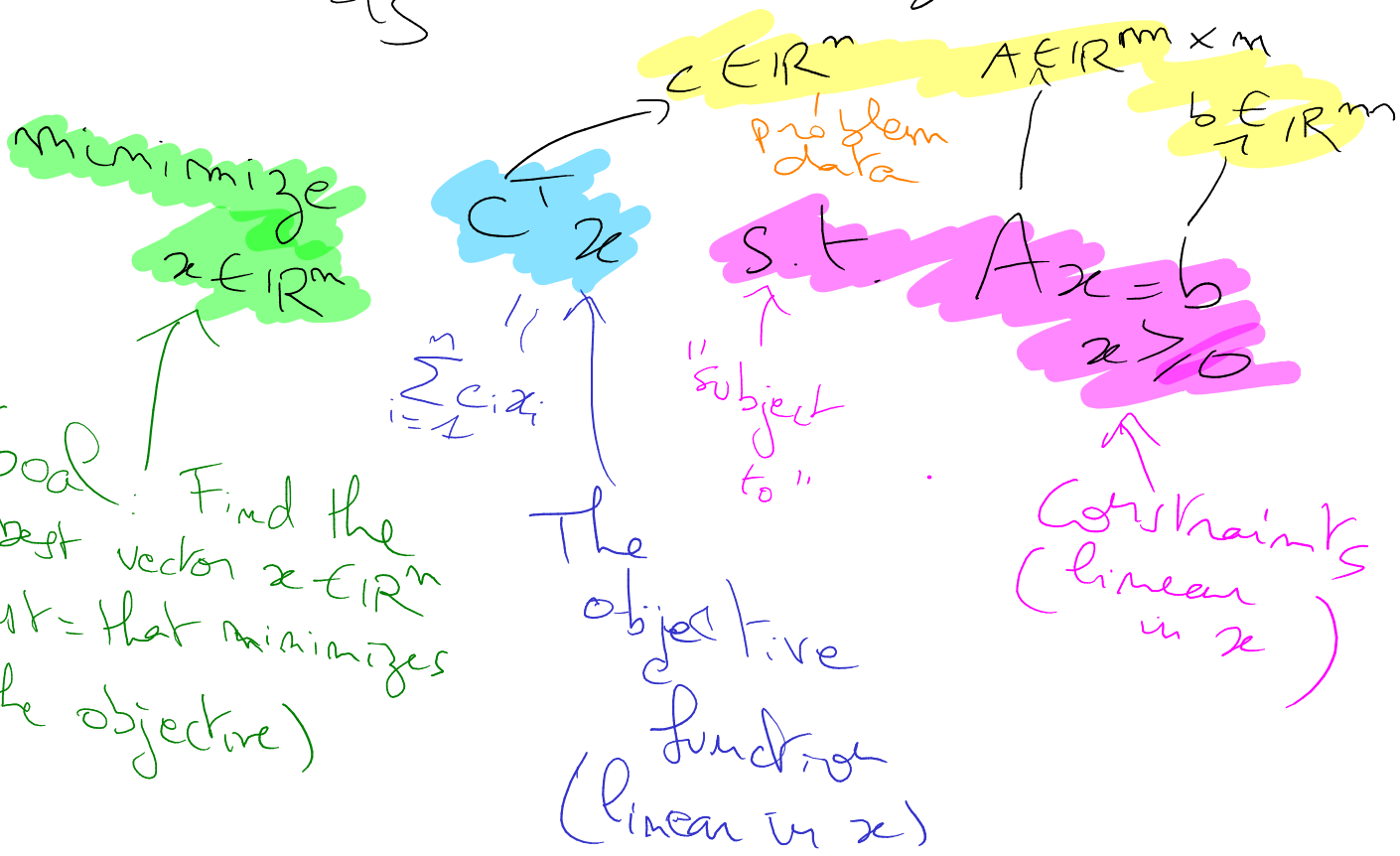
Linear program { Solved every 15 minutes on the US power grid

⇒ Describes how power is distributed (and produced) in the network

⇒ Relaxation of the real problem, solution adjusted by human expertise

Linear program (LP)

An LP in standard form is written as



↳ What about maximization?

$\Rightarrow$ Minimize $c^T x$ s.t. $Ax = b$, $x \geq 0$
$x \in \mathbb{R}^n$

and

Maximize $-c^T x$ s.t. $Ax = b$, $x \geq 0$
$x \in \mathbb{R}^n$

have the same solution set

Solution set

$$\underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \{ c^T x \text{ s.t. } Ax = b, x \geq 0 \} \subseteq \mathbb{R}^n$$

is the set of solutions to

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \; c^T x \text{ s.t. } Ax = b, x \geq 0$$

↳ What about $Cx \leq d$ as a constraint?

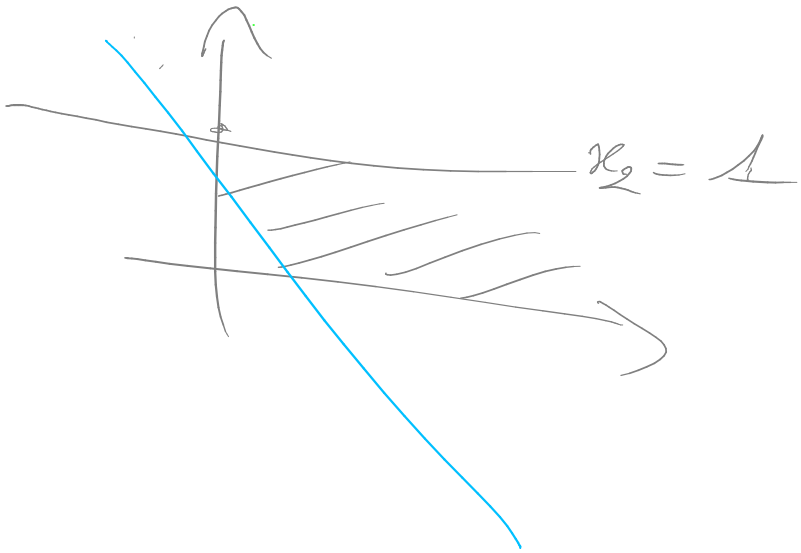
$C \in \mathbb{R}^{l \times n}$, $d \in \mathbb{R}^l$

($a \in \mathbb{R}^l$, $b \in \mathbb{R}^l$, $a \leq b \iff a_i \leq b_i$)

$Cx \leq d$ can be replaced by

$$\begin{cases} Cx + z = d \\ z \geq 0 \end{cases} \text{ for } z \in \mathbb{R}^l$$

(adds variables z but it respects the standard form)



$x_2 = 1$

$x_1, x_2 \geq 0$

$\begin{cases} x_2 = 1 \\ x_1, x_2 \leq 0 \end{cases}$ → ∅

↳ What about $x_i \geq l_i > 0$

$x_i \geq l_i \Rightarrow y_i = x_i - l_i \geq 0$

(Replace x_i by y_i as a variable)

NB: All these "tricks" / reformulation techniques are typically done by optimization solvers automatically

② Solving an LP (Linear Program)

→ Commercial solvers:

Gurobi, CPLEX, Mosek (+ XPRESS, LocalSolver)

(+ Excel Solver)
Google Sheets

→ Academic, open-source
Codes:
CVXpy (goes beyond LP)
HiGHs (default solver
for LP in SciPy since 2022)

→ Historically:

→ Before 1940s: by hand?

→ 1940s: George Dantzig
The simplex method

Important observation (among others)
When you solve a linear
program

(P) $\underset{x \in \mathbb{R}^n}{\text{minimize}}\ c^T x$ s.t. $Ax = b$
$x \geq 0$

You are actually trying to solve
two linear programs at once!

⇒ One is (P) (primal problem)

⇒ The other one is called the dual of (P) and is defined as:

(D) $\underset{\substack{y \in \mathbb{R}^m \\ s \in \mathbb{R}^n}}{\text{maximize}} \; b^T y \quad \text{s.t.} \; A^T y + s = c, \; s \geq 0$

Same data than that defining (P)

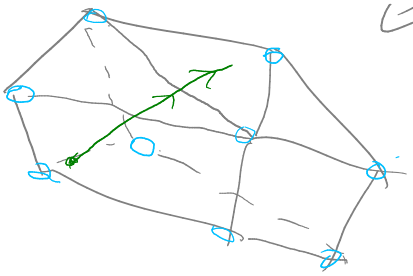
(D) is also a linear program

⇒ The solution sets of (P) and (D) are related to one another

↳ 1980s: Karmakar published a paper on interior-point methods for LPs

↳ 2000s: All major codes rely on IPMs + thousands of research papers

minimize $c^T x$ s.t. $Ax = b,\ x \geq 0$



○ possible solutions (explored by the simplex method)

→ Trajectory of IPMs (inside containing the set of