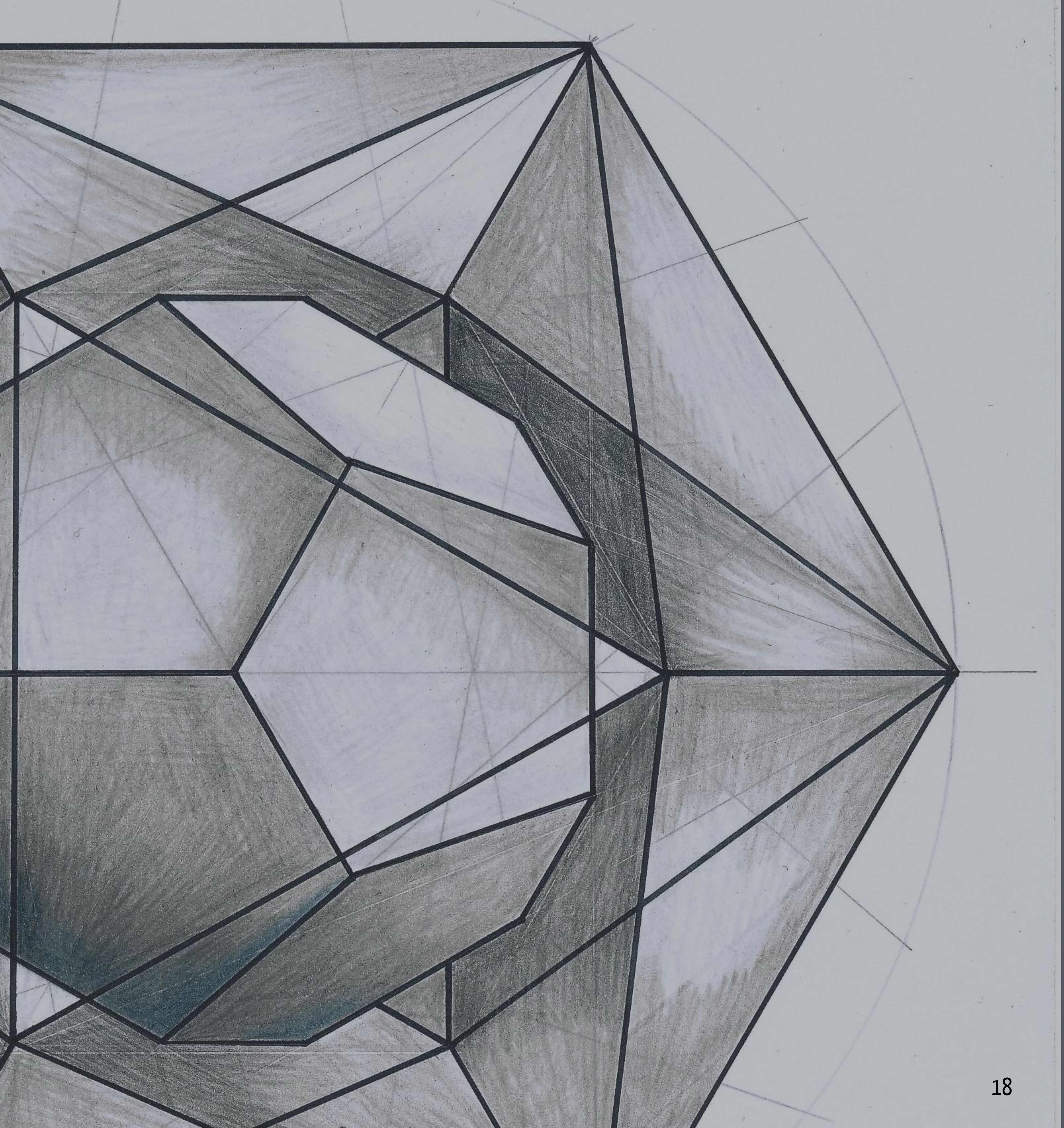
Mathematical optimization: introduction and application to water management

Sophie Demassey
https://sofdem.github.io/



mathematical optimization

mathematical program

$$\min f(x) : g(x) \leq 0,\ x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}$$

$f : \mathbb{R}^n \mapsto \mathbb{R}$ objective

$g : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^m$ constraints

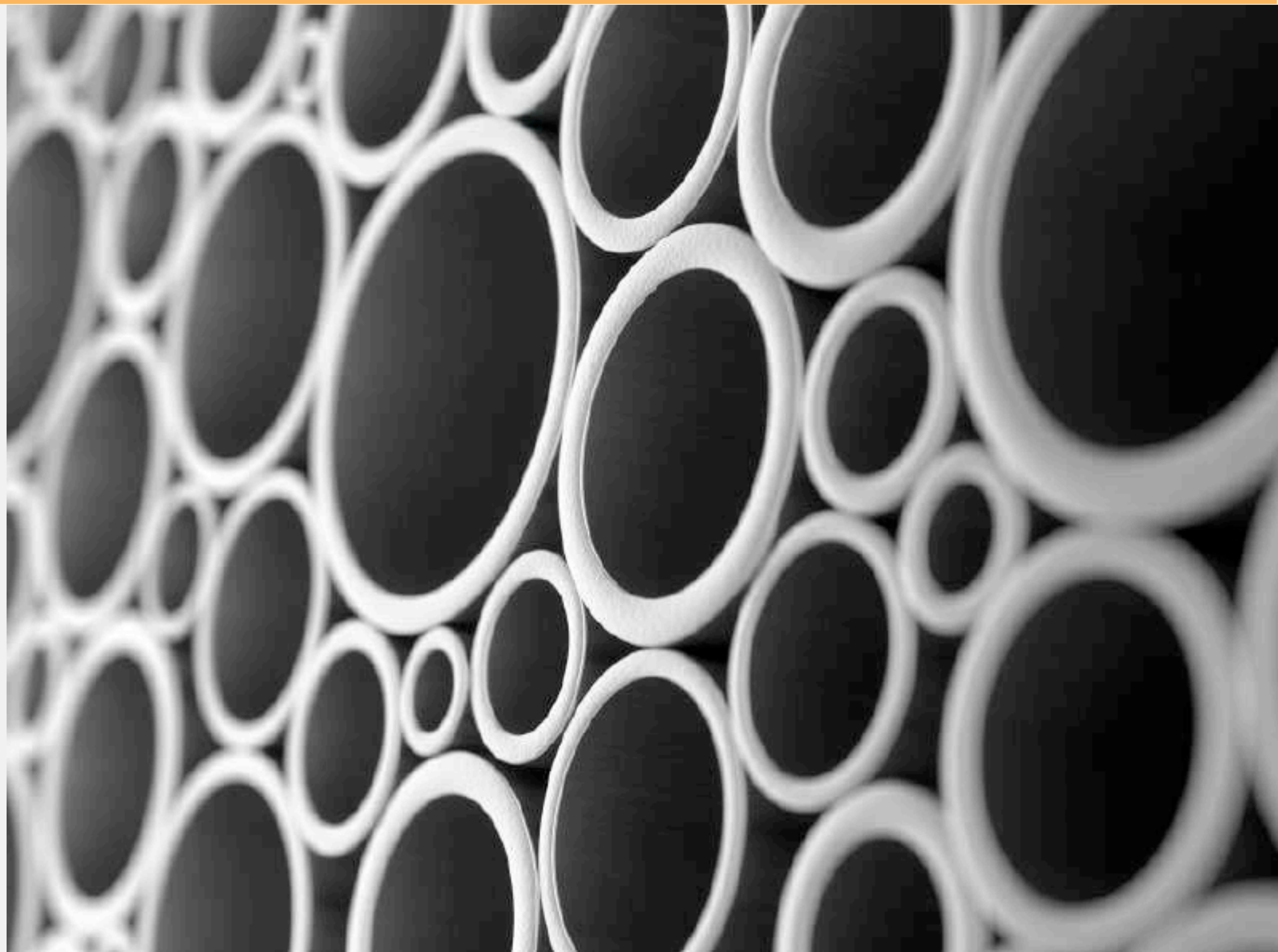
$x \in \mathbb{R}^n$ variables / solution

my first MP

Pipe Sizing

Choose the diameter of two water distribution pipes
to maximize the total rate of flow, within a budget of 180 euros, given that:

- 1st pipe: maximum diameter = 40cm, cost=3 euros/cm, avg rate=3u/cm
- 2nd pipe: maximum diameter = 60cm, cost=2 euros/cm, avg rate=5u/cm



variables: diameters for the pipes (in cm)
constraints: bounds and budget
objective: maximize flow rate

$$\max\ 3x_1 + 5x_2 :$$
$$0 \le x_1 \le 40,\ 0 \le x_2 \le 60$$
$$3x_1 + 2x_2 \le 180$$

my first MP

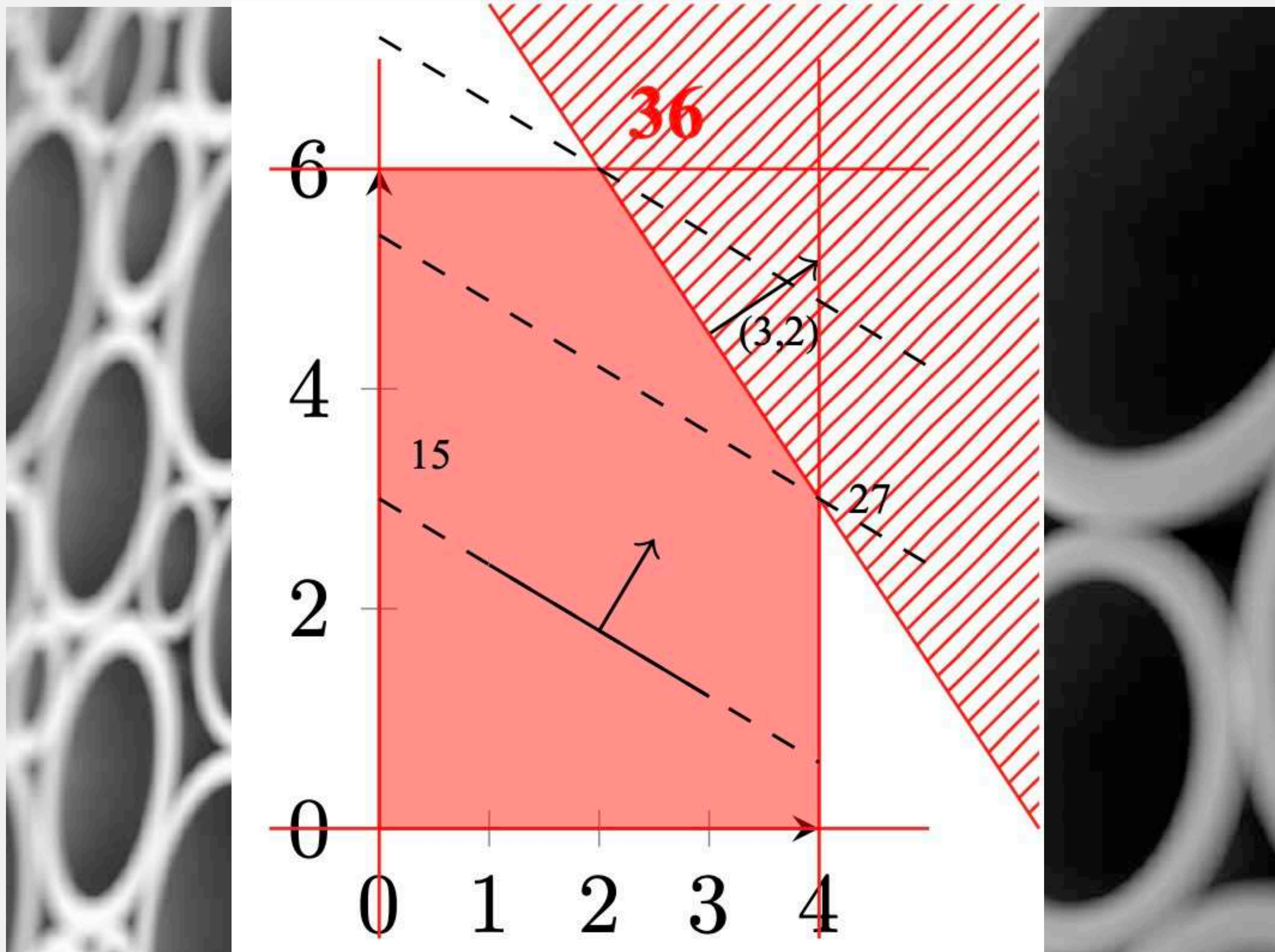
$$\max\ 3x_1 + 5x_2 :$$
$$0 \leq x_1 \leq 40,\ 0 \leq x_2 \leq 60$$
$$3x_1 + 2x_2 \leq 180$$

variables: diameters for the pipes (in cm)
constraints: bounds and budget
objective: max flow rate

linear case: graphical solution

- constraints define half-spaces in $\mathbb{R}^2$
- intersection = poyhedron = feasible solutions
- solutions of cost p: point in line $3x_1 + 5x_2 = p$
- optimal solution: corner on the highest line
- $x = (20{,}60)$ cost $p = 3 * 20 + 5 * 60 = 360$

automatization: the simplex algorithm



mathematical program

$$\min f(x) : g(x) \leq 0,\ x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}$$

well-solved classes:

f, g linear $p = 0$	linear programming
f convex, $g \equiv 0, p = 0$	unconstrained optimization
f, g smooth convex $p = 0$	convex programming
f, g linear $p > 1$	mixed integer linear programming

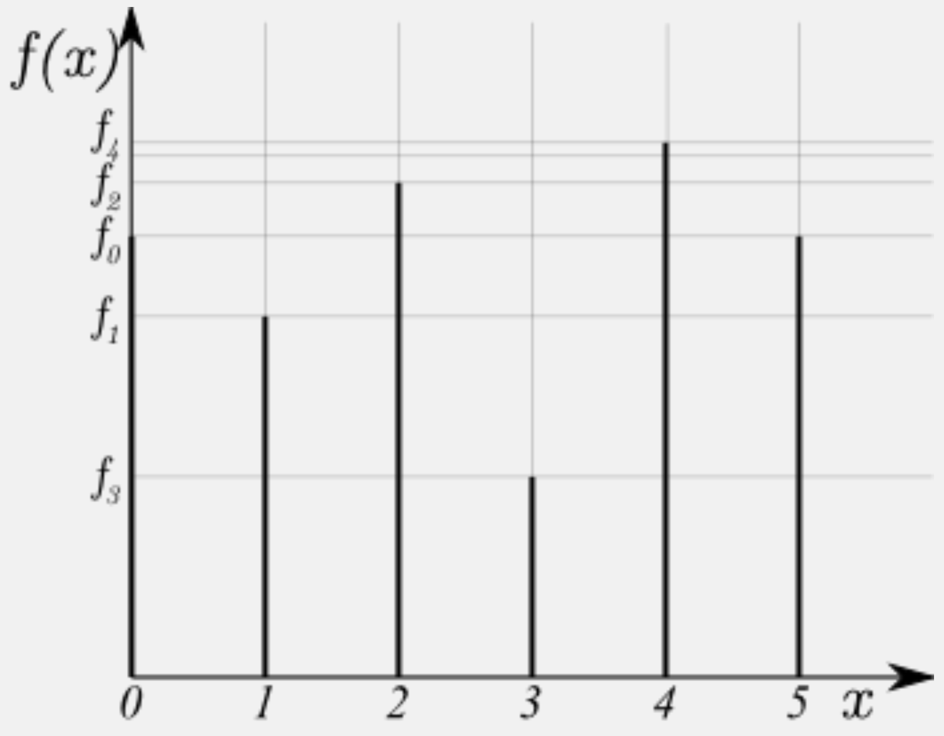
Mixed Integer Linear Program

covers discrete decisions: off/on status $x \in \{0,1\}$, operation level $l \in \{0,1,\ldots,N\}$

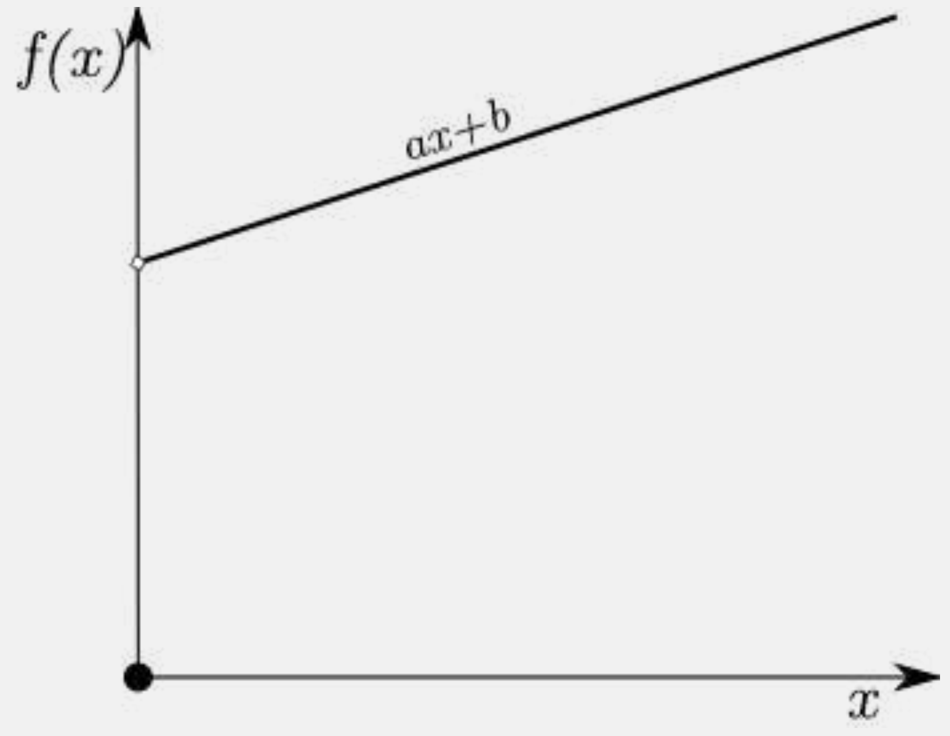
covers logical relations: $l \leq N(1-x)$ level is 0 if status is on: $x = 1 \implies l = 0$

covers nonlinear relations: $l = \sum_{i=0}^{N} i x_i,\ y = \sum_{i=0}^{N} f_i x_i,\ 1 = \sum_{i=0}^{N} x_i,\ x_i \in \{0,1\}\, \forall i \in \{0,\ldots,N\}$

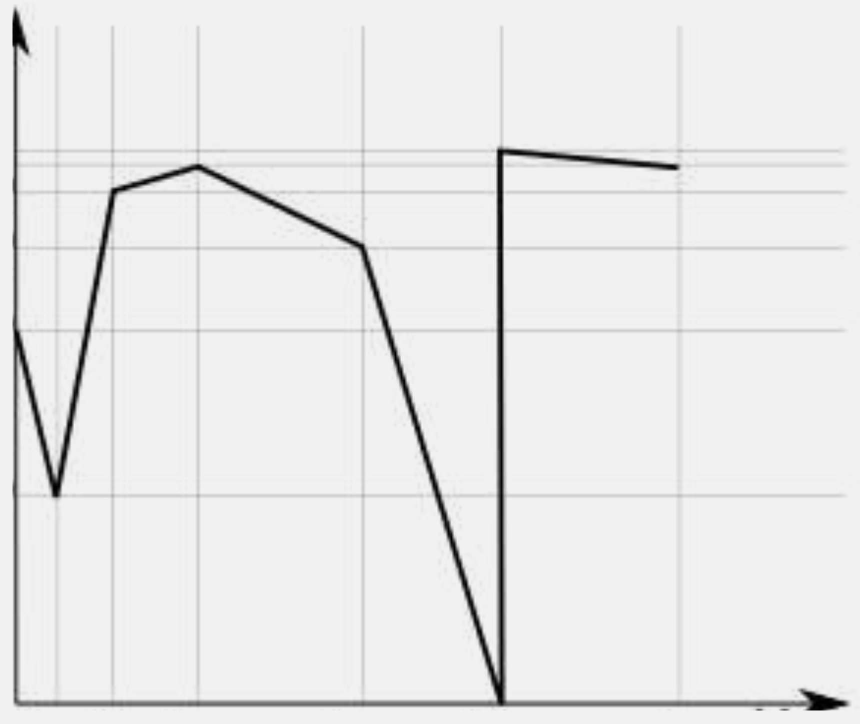
$y = f(l)$ a discrete function



discrete



setup



piecewise linear

my first MILP

Groundwater abstraction (identical pumps)

Choose places to install pumps within a finite set J of candidates and minimize the global cost, given:

- installation cost c_j and average flow rate q_j of a pump at place $j \in J$
- limited total abstraction rate: lower $\underline{Q}$ and upper $\overline{Q}$ limits
- at most 3 pumps installed, no 2 pumps on places a and $b \in J$

variables: $x_j \in \{0,1\}$ number of pumps installed at $j \in J$

$$min \ \sum c_j x_j :$$

$$\underline{Q} \leq \sum q_j x_j \leq \overline{Q}, \sum x_j \leq 3$$

$$x_a + x_b \leq 1, \ x \in \{0,1\}^J$$

variant MILP

Groundwater abstraction (distinct available pumps)

Assign available pumps taken from a finite set K:

- installation cost c_k and flow rate q_k now depends on pump $k \in K$
- limited abstraction $\underline{Q}, \overline{Q}$

$$\min \sum_k \sum_j c_k x_{jk} :$$

$$\underline{Q} \leq \sum_k \sum_j q_k x_{jk} \leq \overline{Q}$$

$$\sum_j x_{jk} \leq 1 \quad \forall k \in K$$

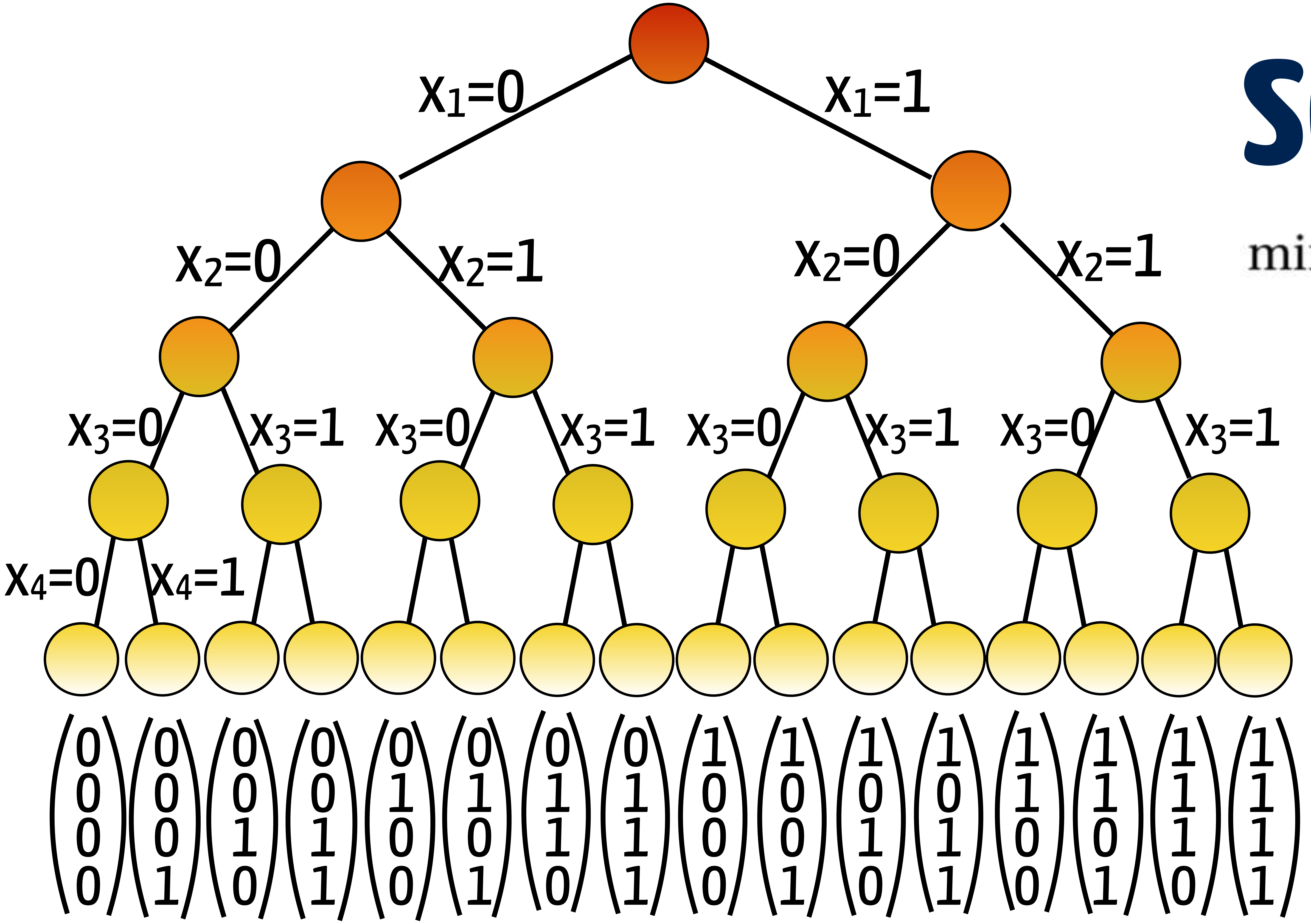
$$\sum_k x_{jk} \leq 1 \quad \forall j \in J$$

$$x_{jk} \in \{0,1\} \quad \forall j \in J, k \in K$$

variables: $x_{jk} = 1$ if pump $k \in K$ installed at place $j \in J$

solving MILP

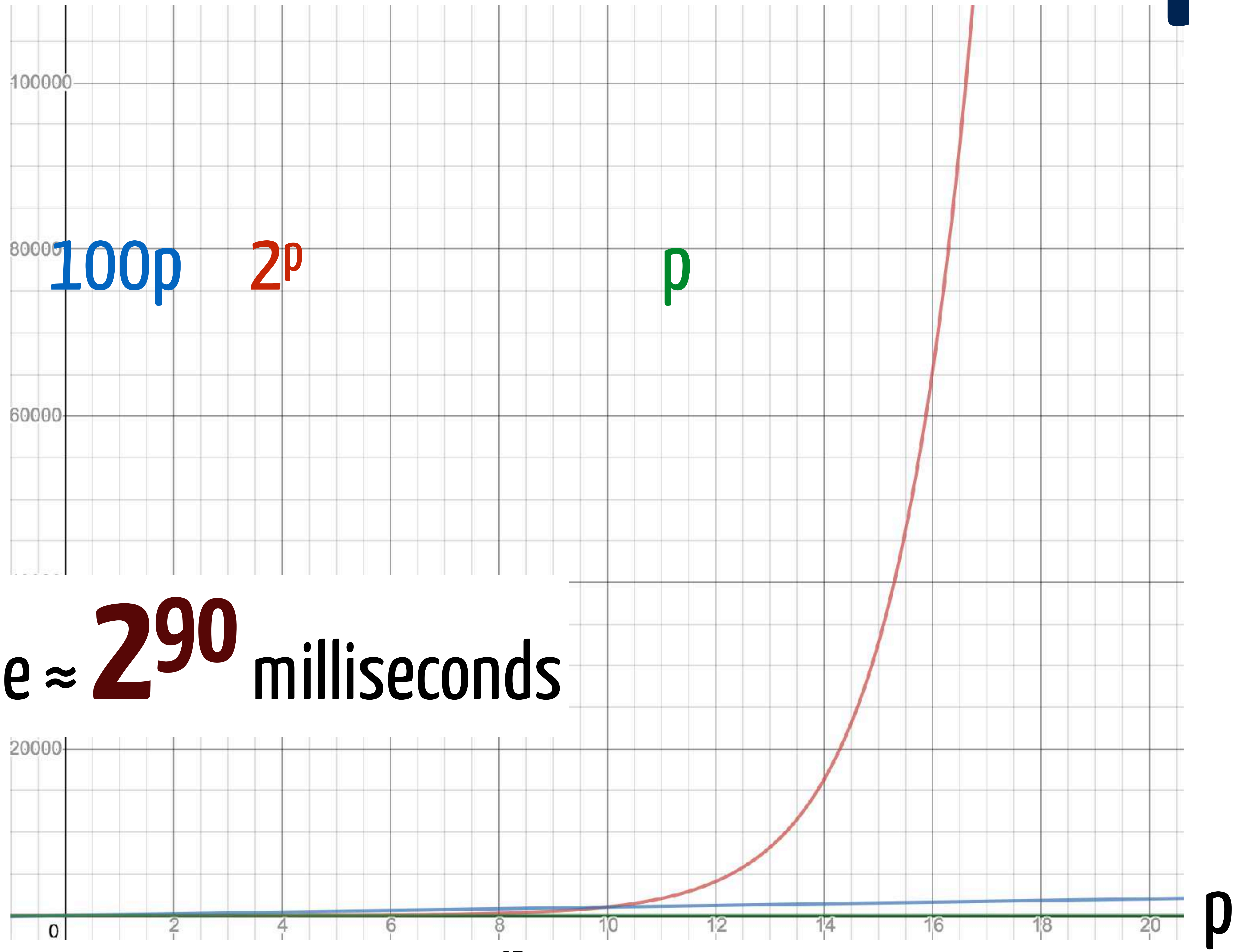
$\min\{cx \mid Ax \geq b, x \in \{0,1\}^p \times \mathbb{R}^{n-p}\}$



Complete enumeration = 2^p LPs to solve

Combinatorial explosion

complexity



age of the universe ≈ 2^{90} milliseconds

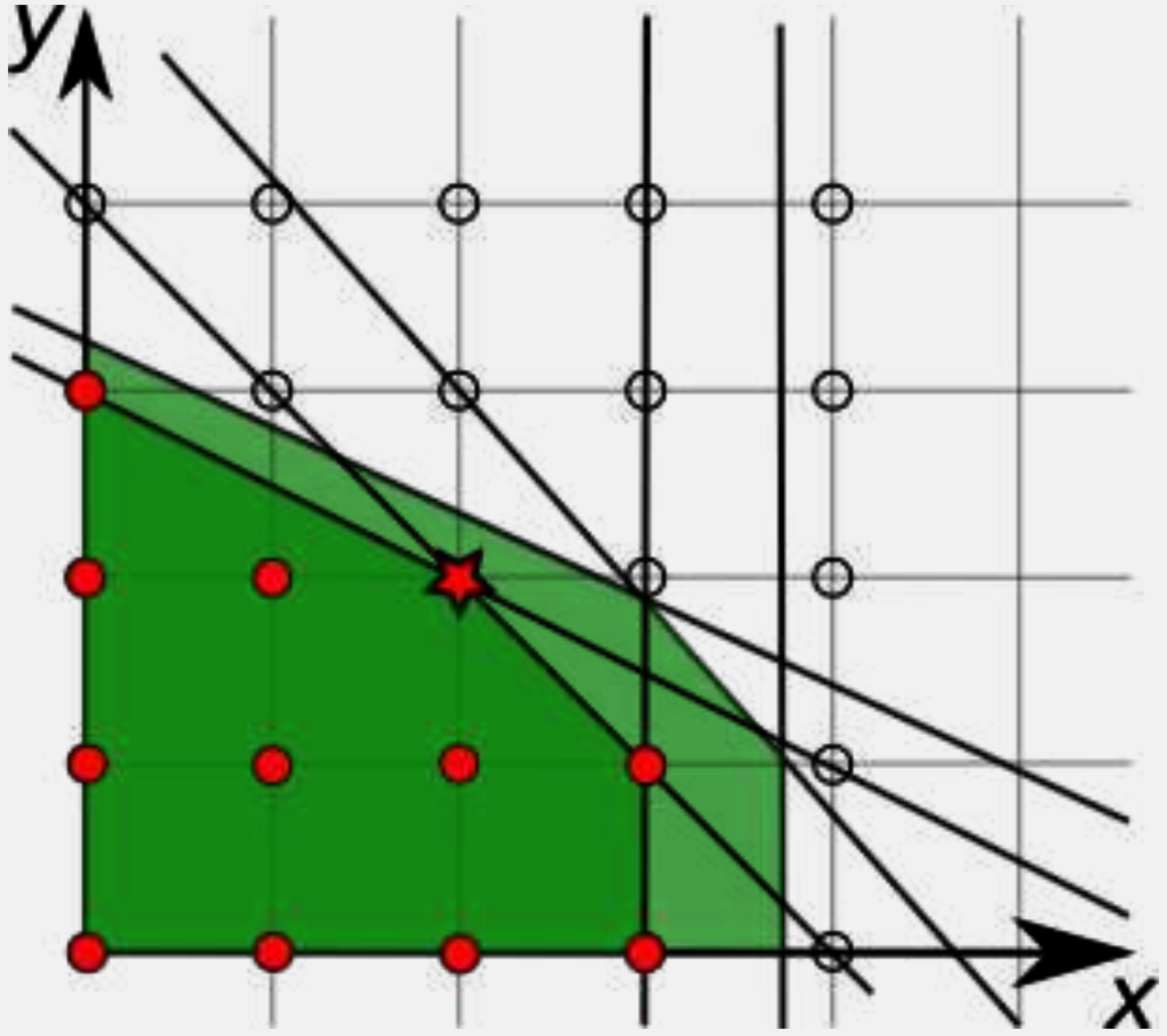
MILP algorithms

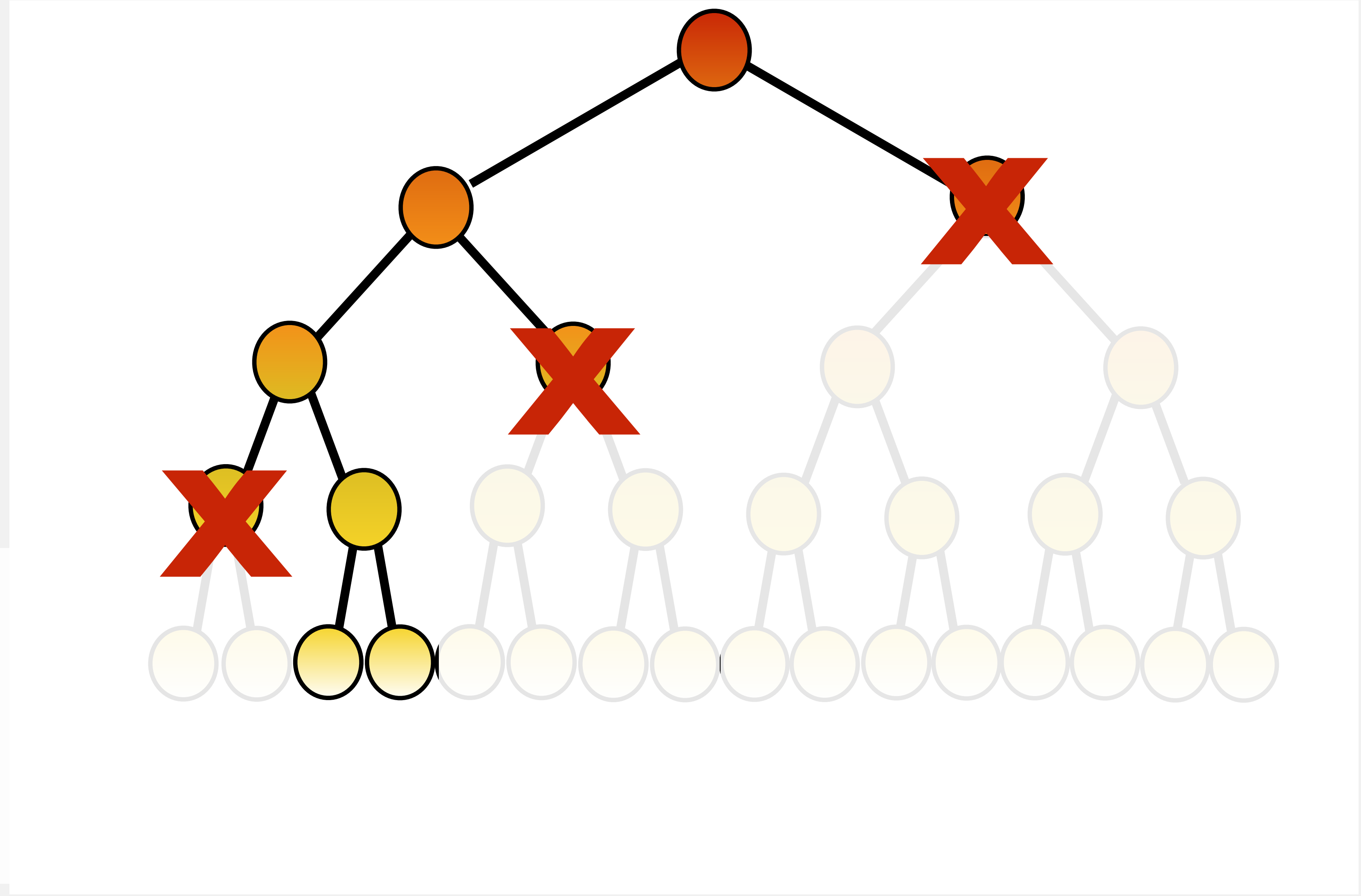
- based on the LP relaxation (ex: simplex algorithm)

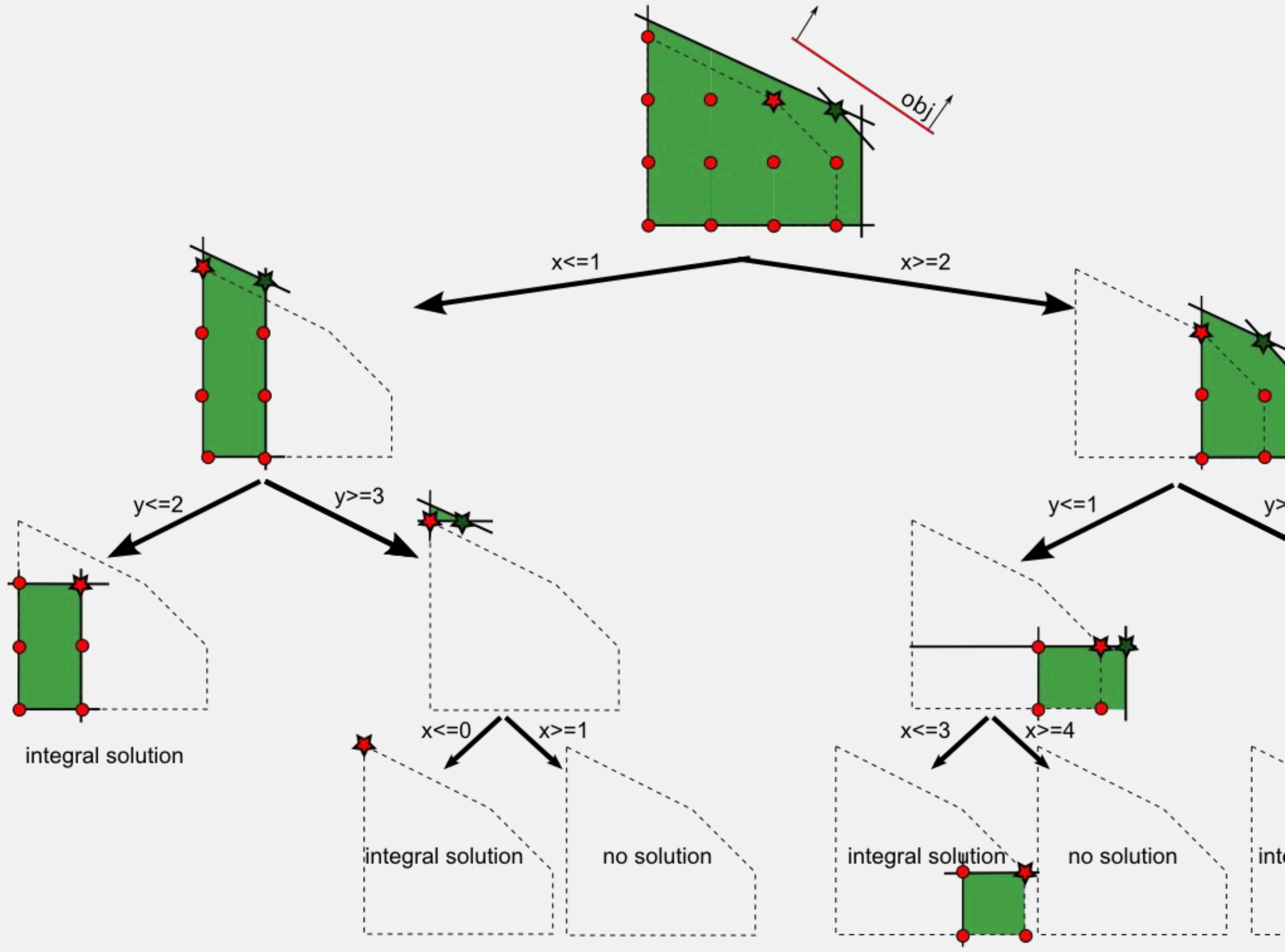
cutting-plane algorithm – evaluate, refine, iterate

branch-and-bound – separate (on discrete variables), estimate, backtrack/iterate

branch-and-cut – refine then estimate

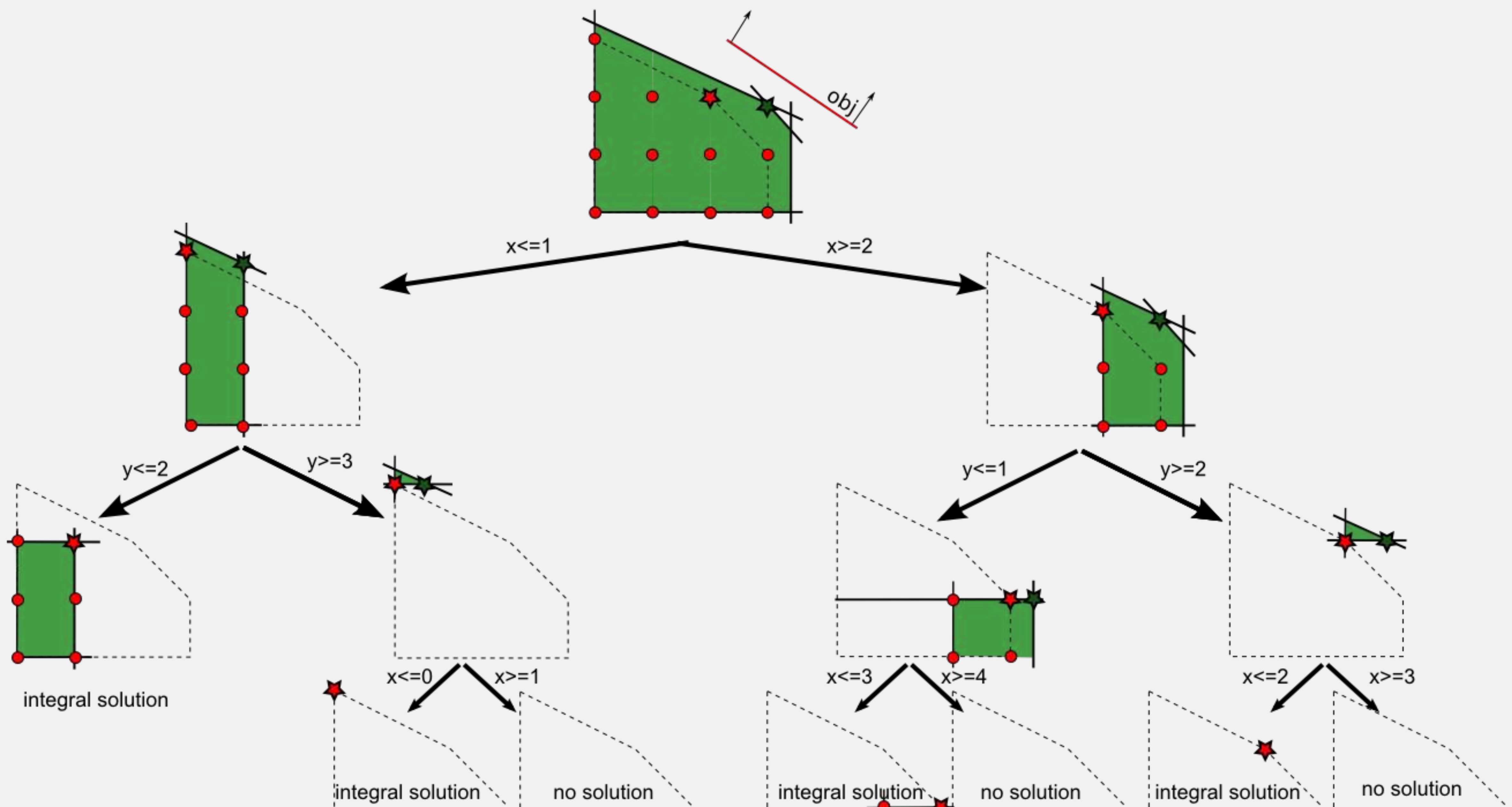






MILP solvers

- MILP solvers (**gurobi, cplex, glpk, mosek,...**) offer sophisticated and efficient implementations of branch-and-cut algorithms
- no algorithm to develop, just one model to give in input
- input: text format (**lp**), modelling langage (**gams, ampl**), library (**pyomo, matlab**) or solver API for different languages (**python, C, java,...**)




```
\ Model PowerGen0
\ LP format - for model browsing. Use MPS format to
Minimize
  9 p[A,0] + 4.5 p[A,1] + 8.28 p[B,0] + 4.14 p[B,1]
   + 16.5 p[C,0] + 8.25 p[C,1]
Subject To
 pmin[A,0]: - 850 x[A,0] + p[A,0] >= 0
 pmin[A,1]: - 850 x[A,1] + p[A,1] >= 0
 pmin[B,0]: - 1250 x[B,0] + p[B,0] >= 0
 pmin[B,1]: - 1250 x[B,1] + p[B,1] >= 0
 pmin[C,0]: - 1500 x[C,0] + p[C,0] >= 0
 pmin[C,1]: - 1500 x[C,1] + p[C,1] >= 0
 load[0]: p[A,0] + p[B,0] + p[C,0] >= 15000
 load[1]: p[A,1] + p[B,1] + p[C,1] >= 30000
Bounds
 x[A,0] <= 12
 x[A,1] <= 12
 x[B,0] <= 10
 x[B,1] <= 10
 x[C,0] <= 5
 x[C,1] <= 5
Generals
 x[A,0] x[A,1] x[B,0] x[B,1] x[C,0] x[C,1]
End
```

GUROBI
OPTIMIZATION

Exercise: code

- gurobi + python = gurobipy
- Gurobi is a commercial solver freely available for students and academics
- limited version available on Google Colab
- code examples as Jupyter Notebooks available at:
  https://www.gurobi.com/jupyter_models/

linear regression + LAD*

https://www.gurobi.com/jupyter_models/curve-fitting/

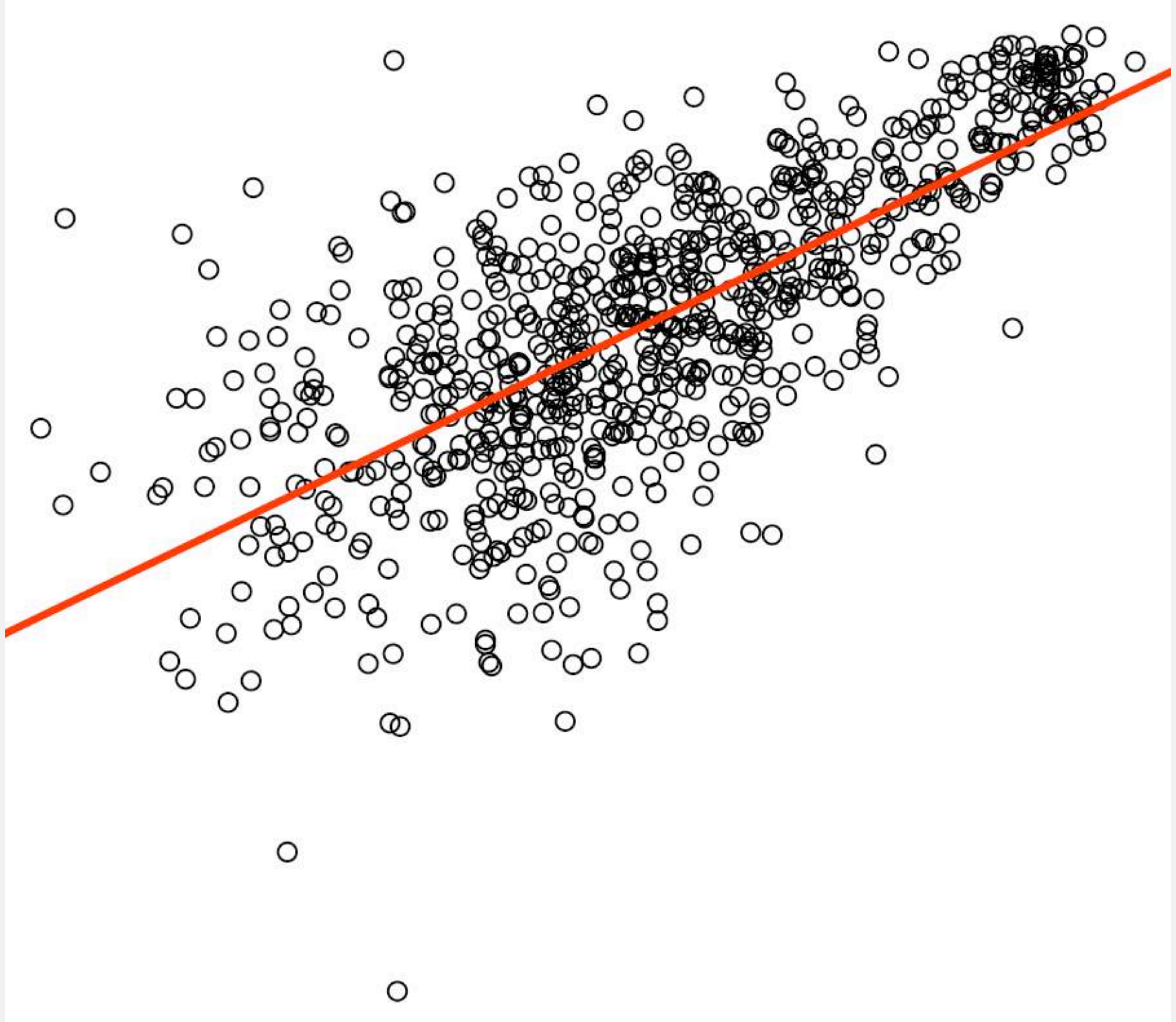
$$min\ \|d\|_1 = \sum_i |d_i| :$$

$$d_i = a + bx_i - y_i \quad \forall i$$

$$d_i \in \mathbb{R} \quad \forall i$$

$$a \in \mathbb{R},\ b \in \mathbb{R}$$

model is not linear !



LP:

$$min \sum_i u_i + v_i :$$

$$u_i - v_i = a + bx_i - y_i \quad \forall i$$

$$u_i \geq 0, v_i \geq 0 \quad \forall i$$

$$a \in \mathbb{R},\ b \in \mathbb{R}$$

LAD*: Least Absolute Deviation

LAD

$$\min \sum_i u_i + v_i :$$
$$u_i - v_i = a + bx_i - y_i \quad \forall i$$
$$u_i \geq 0, v_i \geq 0 \quad \forall i$$
$$a \in \mathbb{R},\ b \in \mathbb{R}$$

https://www.gurobi.com/jupyter_models/curve-fitting/

```
import gurobipy as gp
from gurobipy import GRB
model = gp.Model('CurveFitting')
obs, x, y = gp.multidict(\{('1'): [0,1], ('2'): [0.5,0.9]\})
u = model.addVars(obs, name="u")
v = model.addVars(obs, name="v")
a = model.addVar(lb=-GRB.INFINITY, name="a")
b = model.addVar(lb=-GRB.INFINITY, name="b")
d = model.addConstrs((b*x[i]+a+v[i]-u[i]==y[i] for i in obs))
model.setObjective(u.sum('*') + v.sum('*'))
model.optimize()
print(f"y = \{b.x:.4f\} x + (\{a.x:.4f\})")
```

exercise

- consider this second model for the absolute function:

$$\min |d| = \min s : s \geq d, s \geq -d, s \geq 0$$

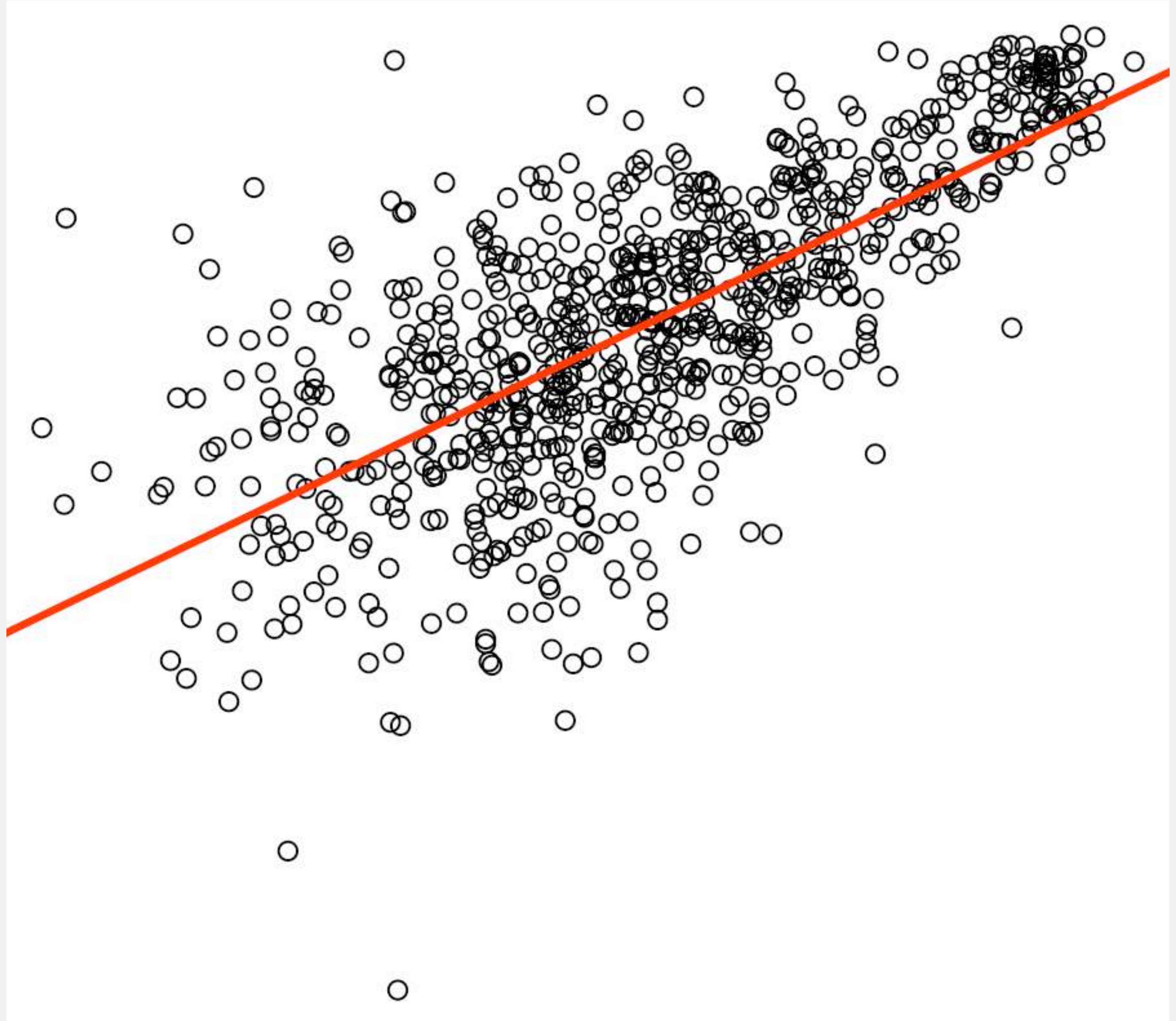
change the LAD model and code accordingly

- minimize the worst (highest) absolute deviation:

$$\min \|d\|_\infty = \min \max_i |d_i|$$

- note that:

$$\min \|d\|_\infty = \min s : s \geq d_i, s \geq -d_i \ \forall i, s \geq 0$$



Uncapacitated Facility Location Problem

$$\min \sum_{j=1}^{n} c_j x_j + \sum_{j=1}^{n} \sum_{i=1}^{m} d_{ij} y_{ij}$$

$$\text{s.t.} \sum_{j=1}^{n} y_{ij} = 1 \qquad i = 1..m$$

or $\sum_{j=1}^{n} y_{ij} \geq 1$ (if d positive)

$$y_{ij} \leq x_j \qquad j = 1..n,\ i = 1..m$$

$$x_j \in \{0,1\} \qquad j = 1..n$$

$$y_{ij} \in \{0,1\} \qquad j = 1..n,\ i = 1..m$$

Input:

- n facility locations
- m customers
- cost c_j to open facility j
- cost d_{ij} to serve customer i from facility j

Output:

a mimimum (opening and service) cost assignment of customers to facilities ?

x_j is location j open ? y_{ij} is customer i served from j ?

MILP perks

declarative
equations, not algorithms

performance
sophisticated solvers

versatile
covers logic & nonlinear

optimality
primal-dual bounds

large-scale
decomposition methods

flexible
general-purpose format & solvers

MILP perks*

declarative
equations, not algorithms
*good model ?

performance
sophisticated solvers
*still NP-hard: scale to some extent
(or consider LP)

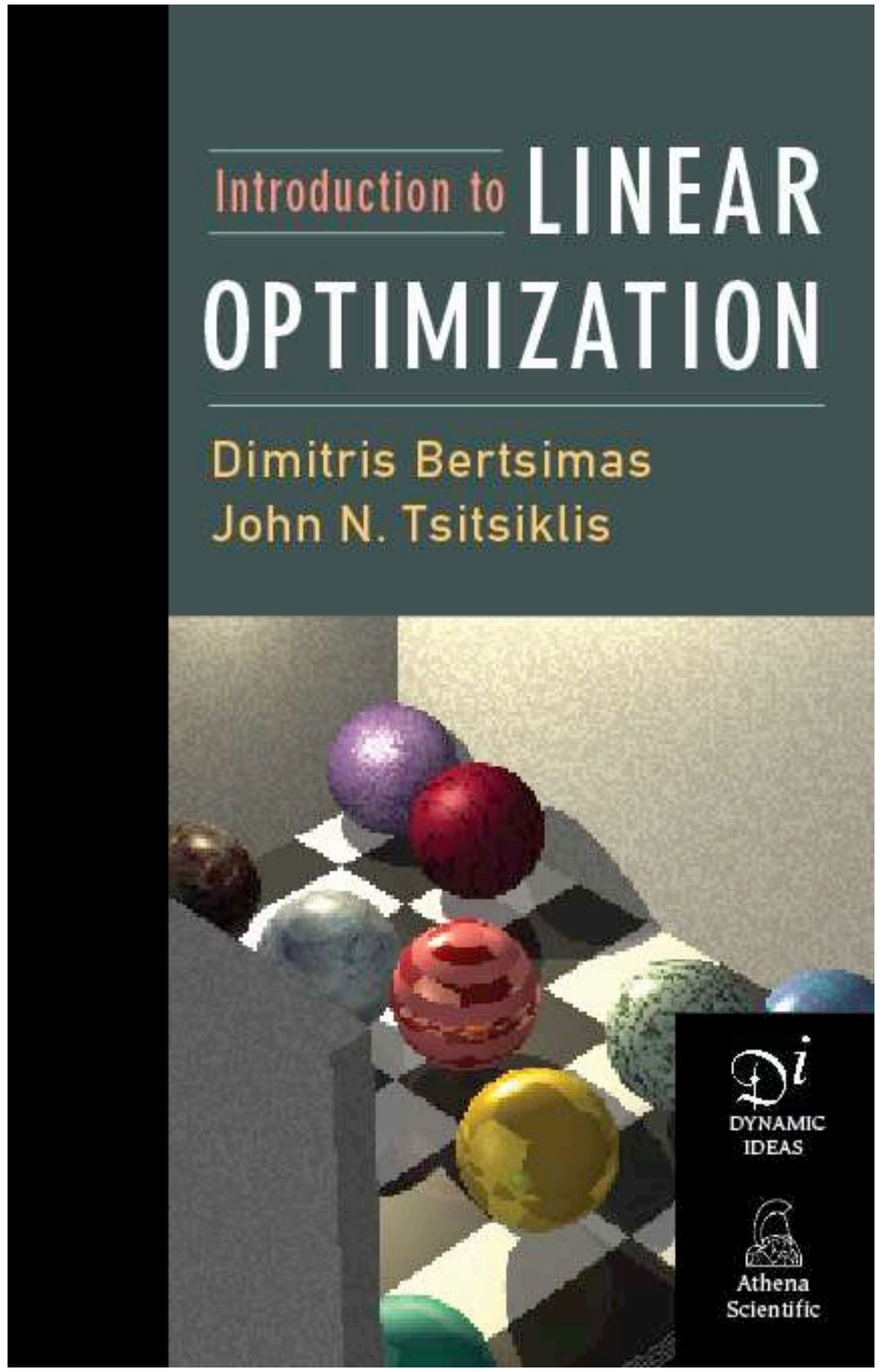
versatile
covers logic & nonlinear
*approximation
(or consider MINLP)

optimality
primal-dual bounds

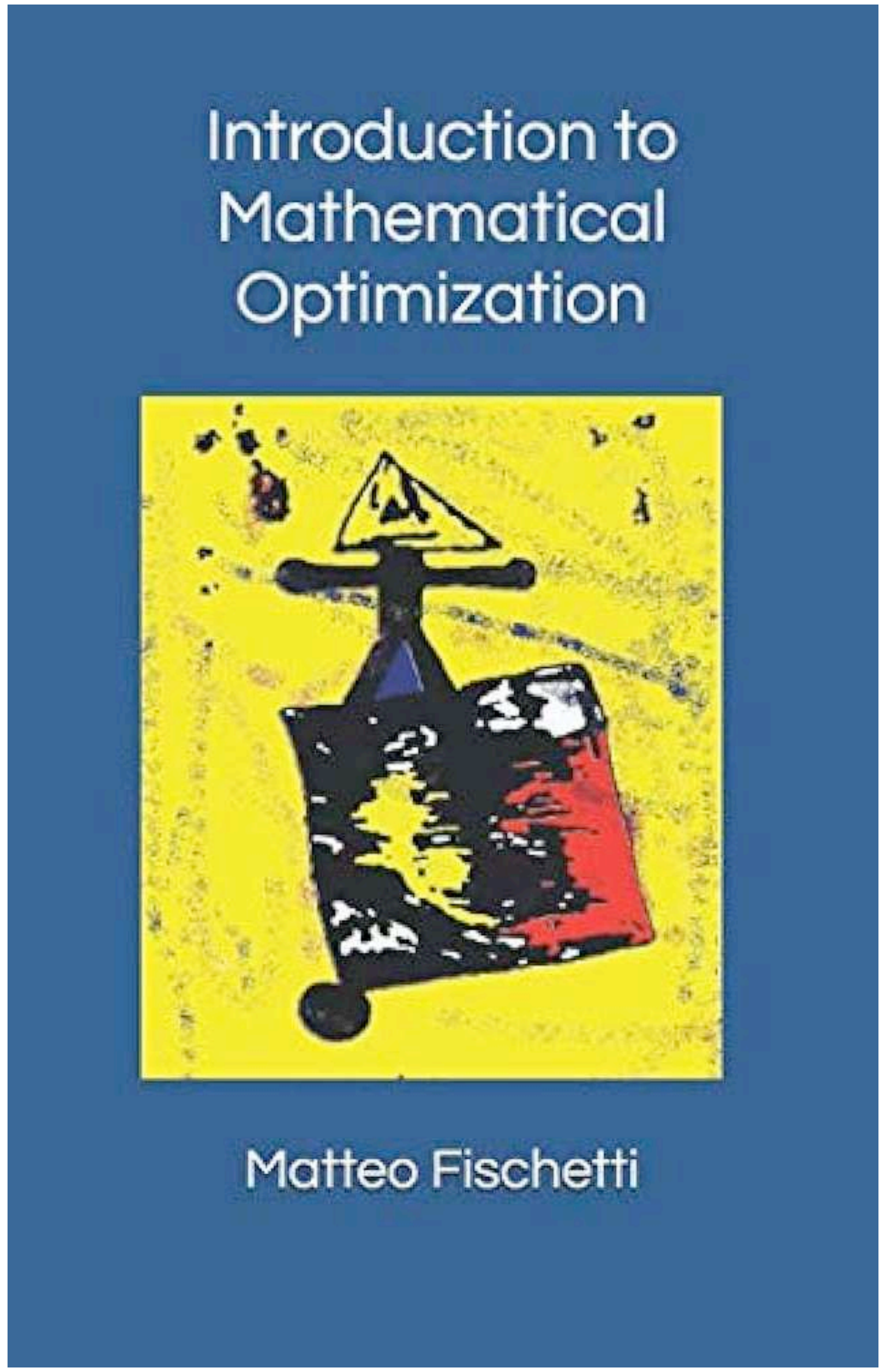
large-scale
decomposition methods
*algorithmic challenge

flexible
general-purpose format & solvers
*generic ≠ best

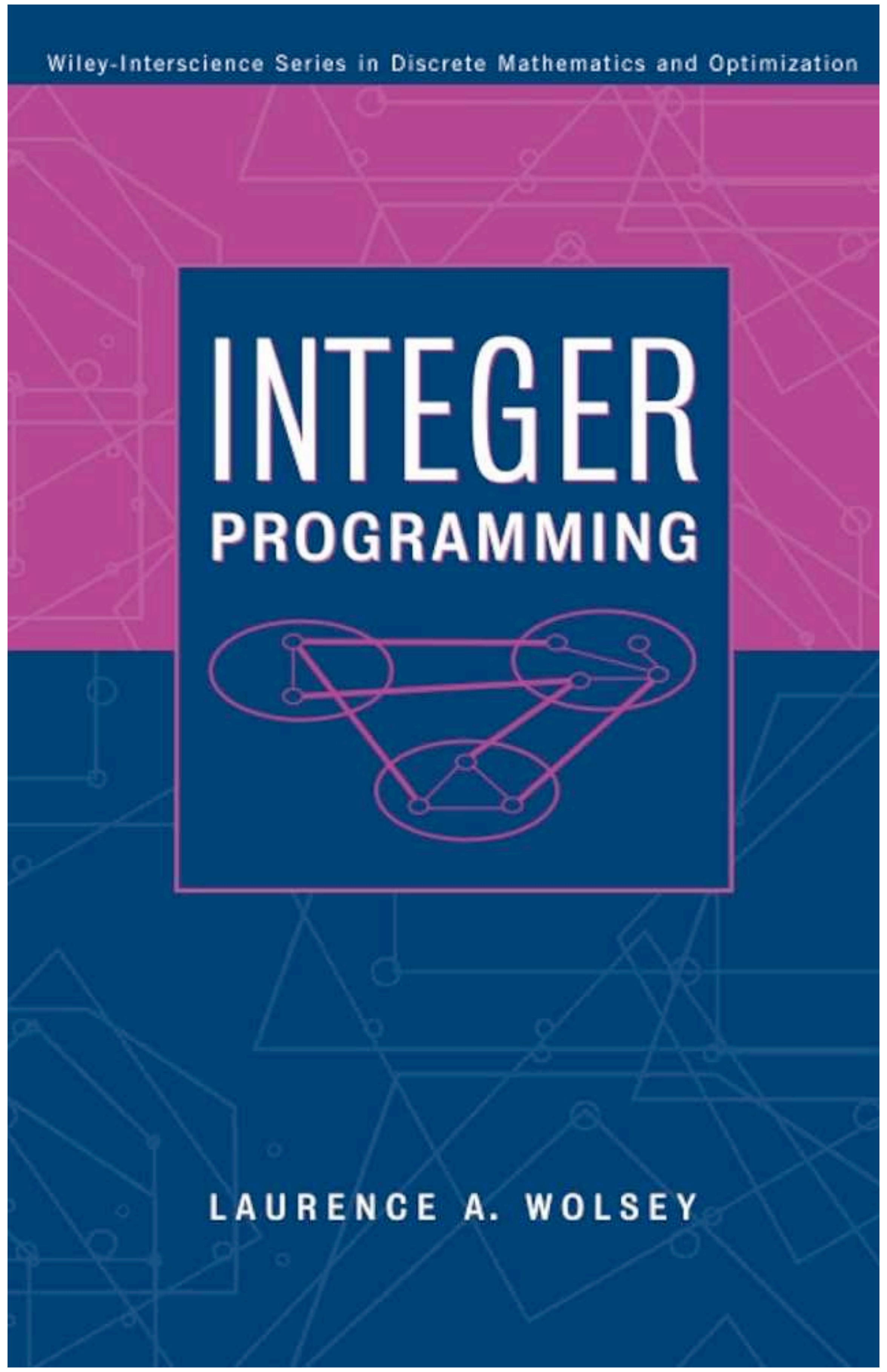
D. Bertsimas & J. Tsitsiklis 1997



M. Fischetti 2019



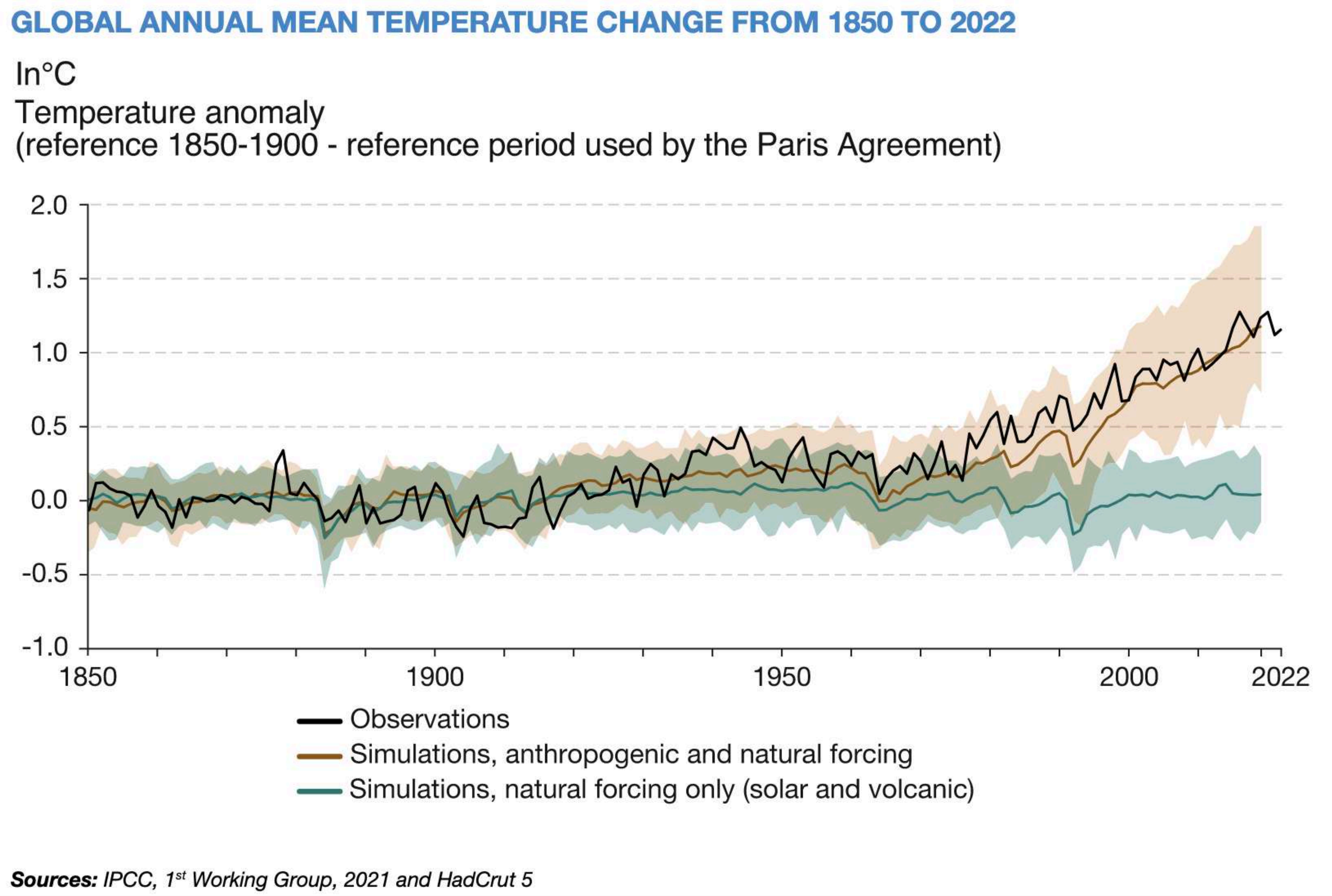
L. Wolsey 1998



climate change & decision aid

global warming

https://www.statistiques.developpement-durable.gouv.fr



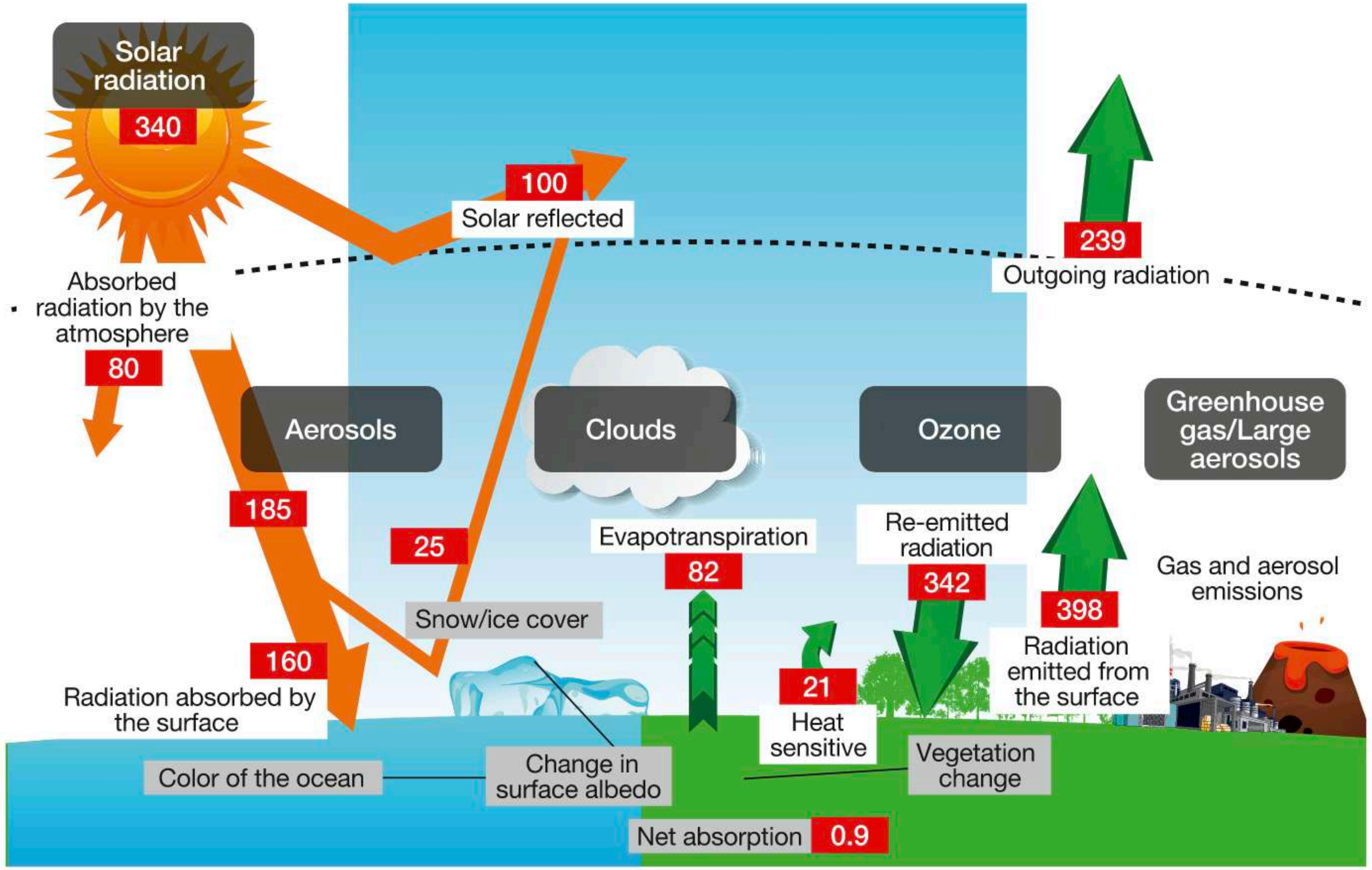
- World: glacier melt, sea-level rise acceleration
- France : increased rate of warming, rainfall deficit, number of forest fires, restrictions on water use, etc.

greenhouse gases

https://www.statistiques.developpement-durable.gouv.fr

THE NATURAL GREENHOUSE EFFECT AND ITS DISRUPTION BY HUMAN ACTIVITIES

Current energy flows in W/m²



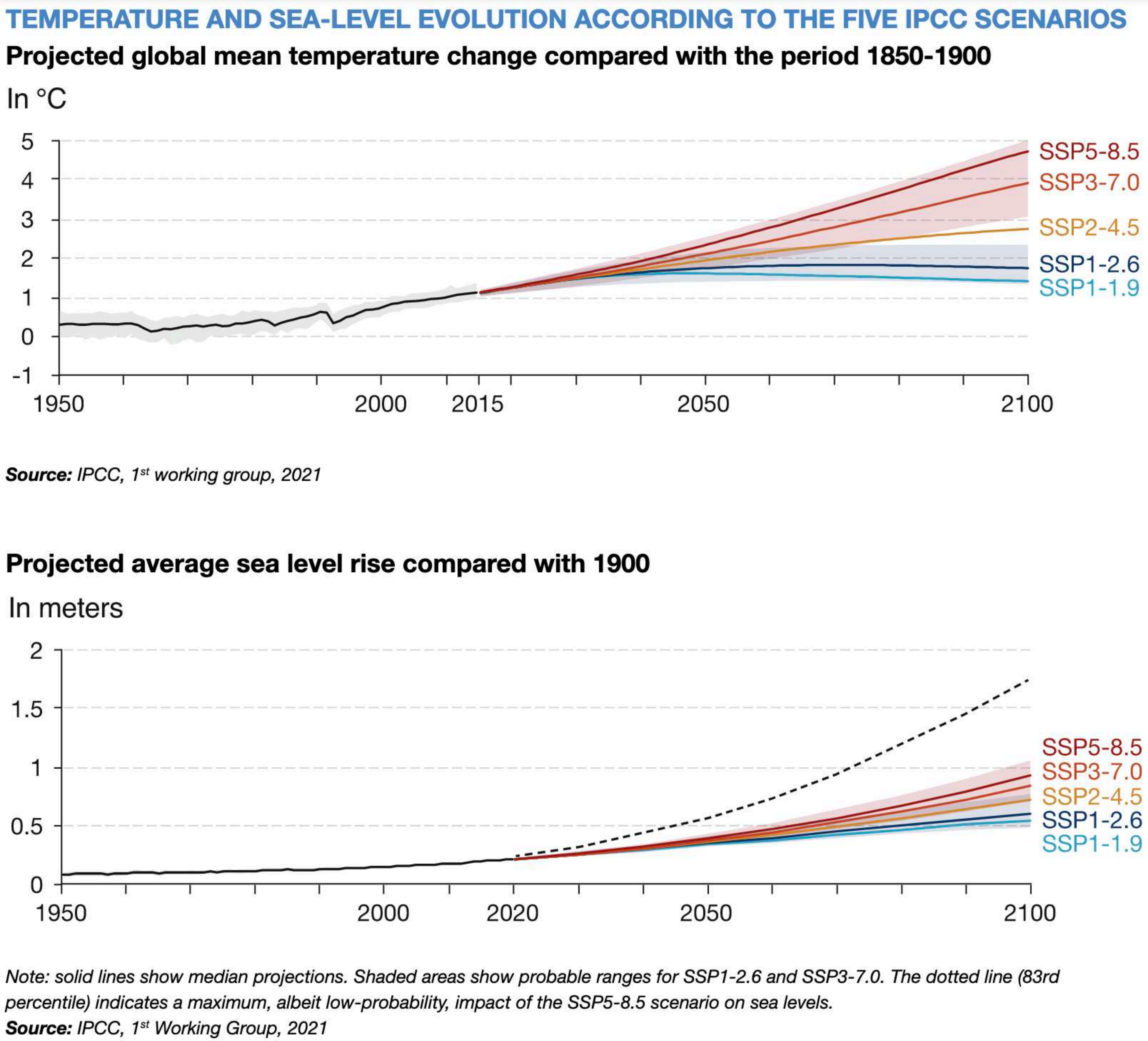
Note: the Earth constantly receives energy from the sun. The part of this energy that is not reflected by the atmosphere, such as clouds or the earth's surface (oceans and continents), is absorbed by the earth's surface, which heats up by absorbing it. In return, surfaces and the atmosphere emit infrared radiation, the hotter the surface, the more intense the radiation. Some of this radiation is absorbed by certain gases and clouds, then re-emitted towards the surface, helping to warm it. This phenomenon is known as the greenhouse effect.

***Sources:** from Météo-France; IPCC, 1st Working Group, 2021*

- Human activities have little impact on water vapor concentration, but a strong impact on the other GHGs.
- CO_2 is the GHG with the lowest global warming potential, but the highest effective contribution due to the large quantities emitted.
- Since pre-industrial era, land and ocean reservoirs have absorbed more than half of anthropogenic emissions. The remaining emissions persist in the atmosphere.

low carbon policy

https://www.statistiques.developpement-durable.gouv.fr



- EU objectives: reducing its net emissions by at least 55% between 1990 and 2030, achieving climate neutrality by 2050 at the latest.

- decarbonation: reducing anthropic emissions primarily in the highest emitting sectors : electricity production (42% world CO_2 emissions in 2020), transportation (22%), industry (20%).

implementing the transition

levers

- sobriety : reduce needs
- efficiency : reduce resource consumption for a same need
- sustainability : reduce non-renewable resources consumption

Example of residential heat : limit the set temperature; isolate; solar vs. fossil energy; electricity/heat cogeneration; storage.

solutions

new technologies and/or better processes
techno-solutionism and/or decision aid ?

Ex: install photovoltaic panels (PV), heat pump, new insulating materials,…
then how to choose them, size them, arrange them, plan them, manage them,
according to: heating needs, budget, physical constraints, GHG emissions, lifespan, etc. ?

water optimization

water is

a commodity, a resource, an environment

drinking water
wastewater
rain, ice, surface water, ground water
fresh, brackish, saline water
irrigation water
source of hydropower (river, tide, wave)
vector of pumped-storage hydroelectricity
steam to generate heat and energy
water for cooling or cleaning
water for processing (fracking, diluting, drilling)
storms, floods, droughts, mudflows, tsunamis
subject to thermal, chemical pollution
related to climate change, climate variability
wetlands, rain forests, oceans, coasts and rivers

to process

extract,
supply,
treat,
produce,
irrigate,
desalinate,
purify,
drain,
heat,
blend,
store,
pump,
flow,
preserve,
measure,
prevent,
control

in small/large systems

urban networks
sewers
desalination plants
farms
power systems
hydropower plants
thermal plants
industries
municipalities
pumps, turbines
aquifers
drainage basins
ecosystems
world

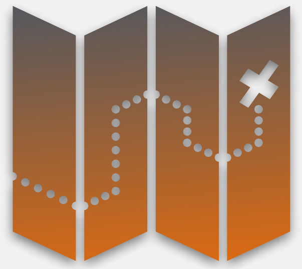
water optimization

Operational

organize the process

select elements to operate
assign operation level
allocate resources
schedule operations
position elements

Tactical



design the system

select elements to dimension, maintain
assign dimension, equipment
plan resources and times

often discrete decisions
nonlinear physical dynamics
minimize an economic/social/ecological cost

study cases

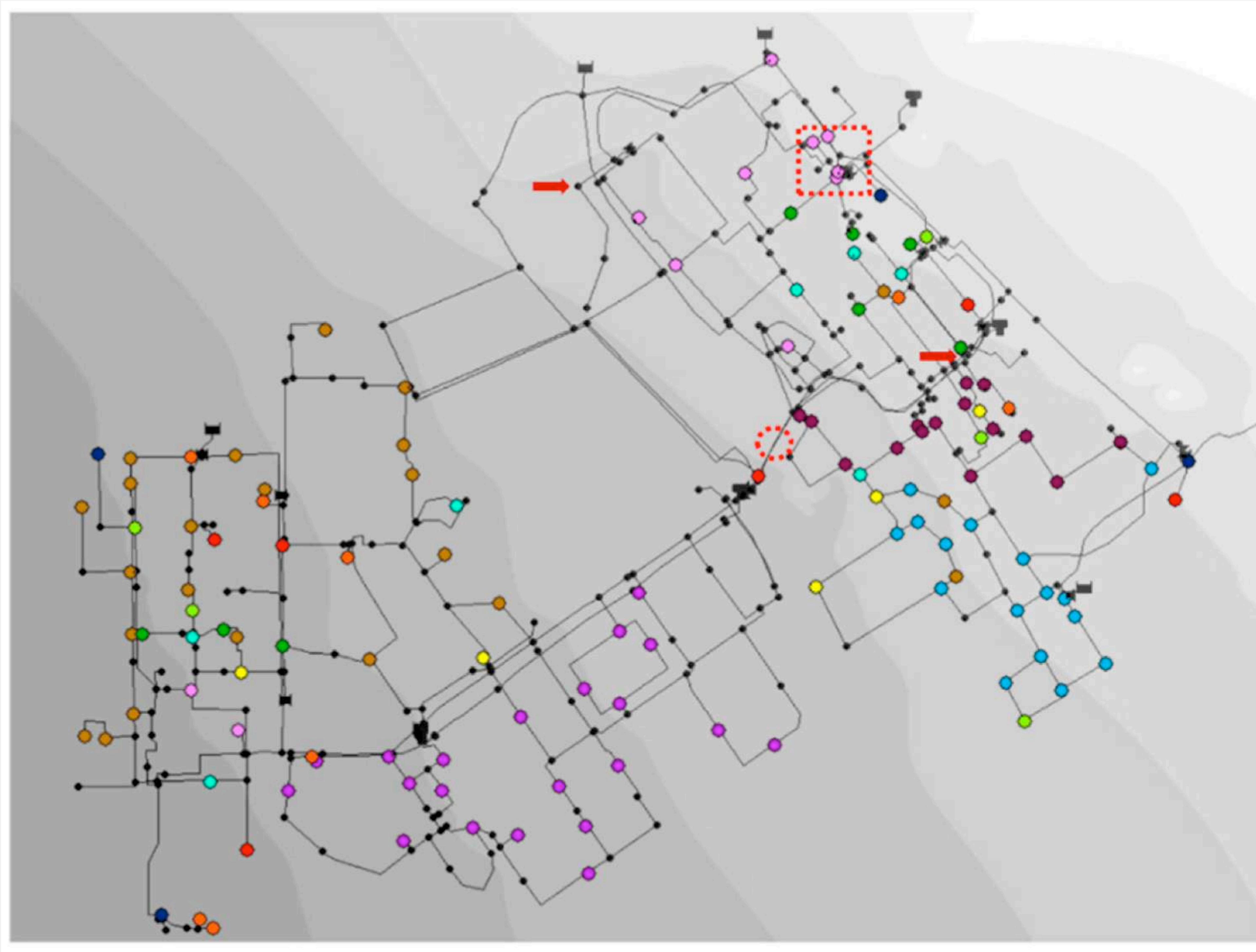
urban water networks

groundwater abstraction

hydroelectricity production

Urban networks

ex1: pipe sizing



select the size of the pipes in a gravity-fed network
to satisfy the demand at each delivery node
while minimizing the installation costs

finite catalog of pipes: size ↗ capacity ↗ cost ↗

ex1: pipe sizing

assign a size k to each pipe a: $x_{ak} = 1$ (otherwise $x_{ak} = 0$)

hydraulic equilibrium between flows q and heads h, v in the selected network

$$\min_{x,q,h} \sum_a \sum_k c_{ak} x_{ak}$$

$$\text{s.t.} \; x_{ak} = 0 \implies q_{ak} = v_{ak} = 0 \qquad \forall a \in A, k \in K$$

$$\sum_k x_{ak} = 1, h_i - h_j = \sum_k v_{ak} \qquad \forall a = (i,j) \in A$$

$$(q_{AK}, h_S) \in NAP(D_S, H_R, \phi_{AK(x)}).$$

bilevel program or
simulation-based genetic algorithm

convex MINLP or approximate MILP
+ branch-and-bound

ex1: pipe sizing

convex MINLP reformulation

$$\min_{x,q,h} \sum_a \sum_k c_{ak} x_{ak}$$

$$\text{s.t.} x_{ak} = 0 \implies q_{ak} = v_{ak} = 0 \qquad \forall a \in A, k \in K$$

$$\sum_k x_{ak} = 1, h_i - h_j = \sum_k v_{ak} \qquad \forall a = (i,j) \in A$$

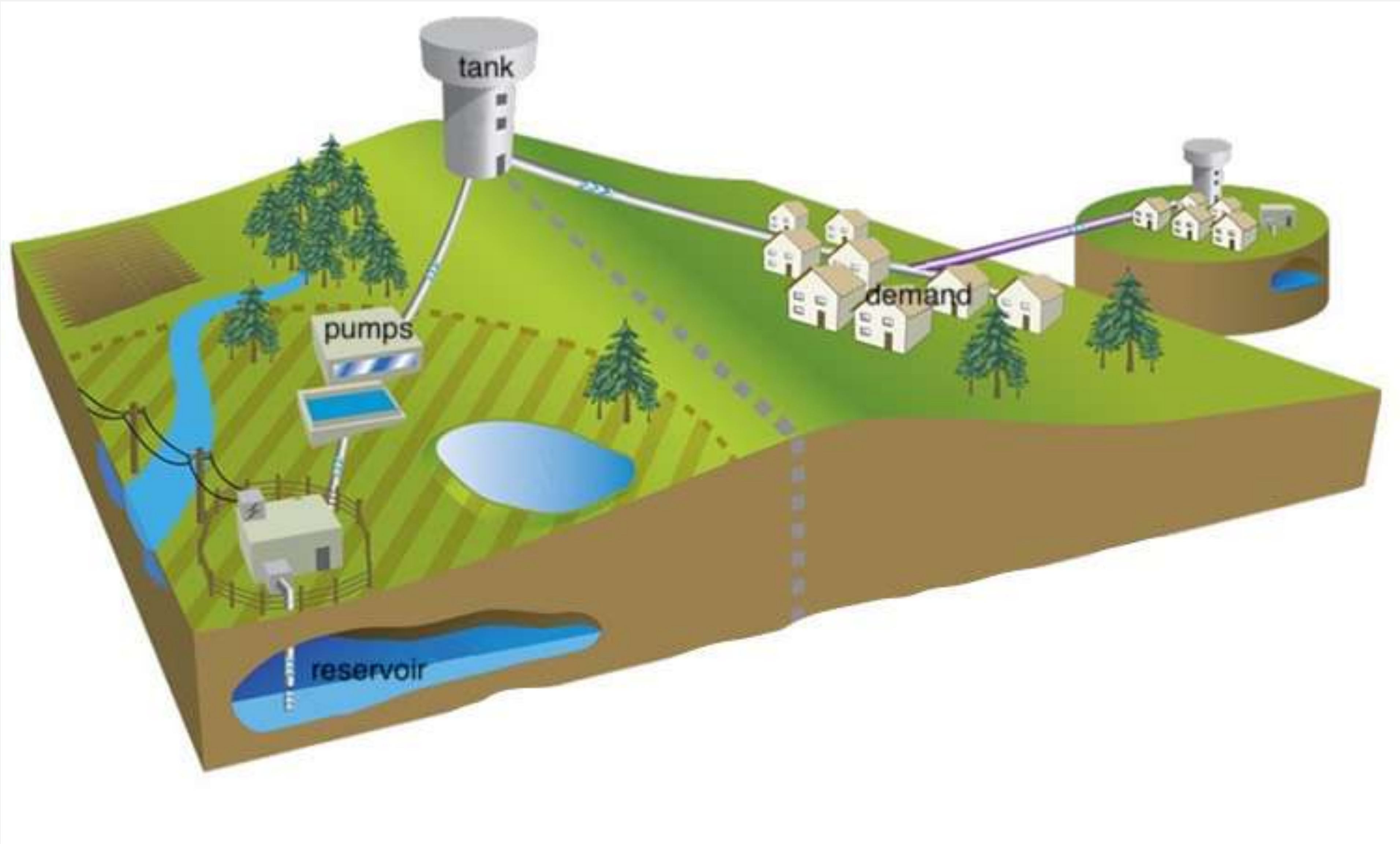
$$\sum_{ak} E_{as} q_{ak} = D_s \qquad \forall s \in S$$

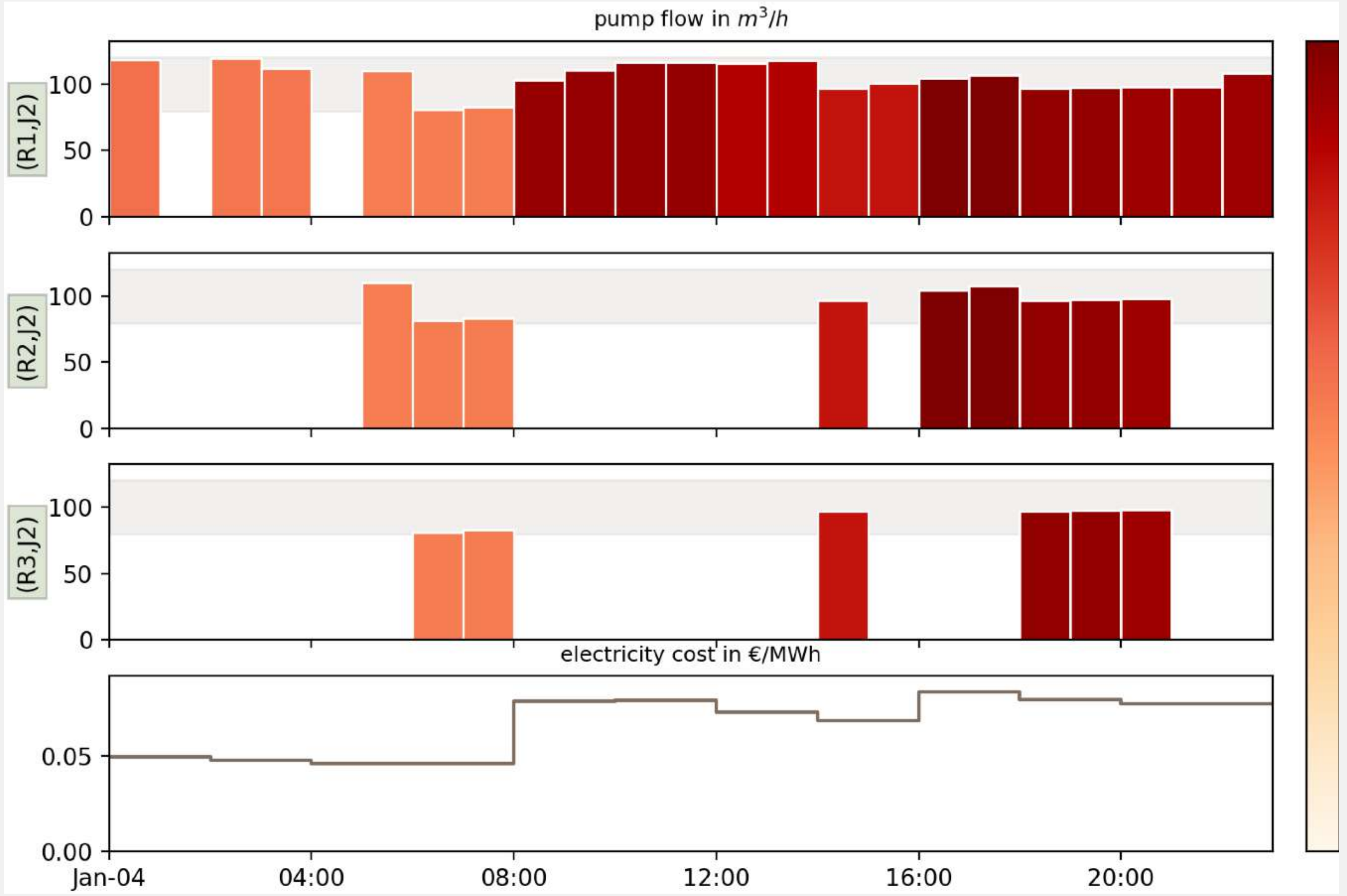
$$\sum_{ak} \left(f_{ak}(q_{ak}) + f^*_{ak}(v_{ak}) \right) + H_R^\top q_R + D_S^\top h_S \leq 0 \qquad (SD)$$

[Demassey. Strong duality reformulation for bilevel optimization over nonlinear flow networks. 2023]

ex2: pump scheduling
(load shifting in pressurized networks)

schedule pumps and valves in a pressurized network on a time horizon
to satisfy the varying demand at each delivery node
and the capacity of the water tanks
while minimizing the electricity bill





ex2: pump scheduling

activate pump/valve a at time t: $x_{at} = 1$ (otherwise $x_{at} = 0$)

hydraulic equilibrium between flows q and heads h, v in the active network

limit the water tank level H

$$\min \sum_a \sum_t c^0_{at} x_{at} + c^1_{at} q_{at}$$

$$\text{s.t.} (q_{At}, h_{St}) \in NAP(D_{St}, H_{Rt}, \phi_{A(x_t)}) \qquad \forall t \in T$$

$$x_{at} = 0 \implies q_{at} = 0 \qquad \forall a \in A, t \in T$$

$$H_{R(t+1)} = H_{Rt} + s_R^\top q_{Rt} \qquad \forall t \in T$$

$$\underline{H}_{Rt} \leq H_{Rt} \leq \overline{H}_{Rt} \qquad \forall t \in T.$$

additional complexity: temporal inter-dependency

[Demassey Strong duality reformulation for bilevel optimization over nonlinear flow networks. 2023]

water network optimization

(drinking, waste, irrigation)

decisions

dimension
renovation
extension
sectorization
scheduling operations
scheduling maintenance
place equipments and controllers
calibrate hydraulic models

concerns

demand: standard, worst-case, emergency
network topology
energy consumption
leakage, over-pressure
flow conservation
pressure-flow relation
chlorine consumption
water quality, treatment
storage capacity
resilience to failures or storms
sewer overflow

[Bello, et al. Solving Management Problems in Water Distribution Networks: A Survey of Approaches and Mathematical Models. Water 2019]
[Mala-Jetmarova, Sultanova, Savic. Lost in Optimisation of Water Distribution Systems? A Literature Review of System Design. Water 2018]

Groundwater

credit: nasa.gov

ex3: sustainable abstraction

place pumps and plan pumping
to prevent aquifer depletion (then land subsidence or seawater intrusion)
and quality degradation (temperature, salinity)
while maximizing the abstraction value

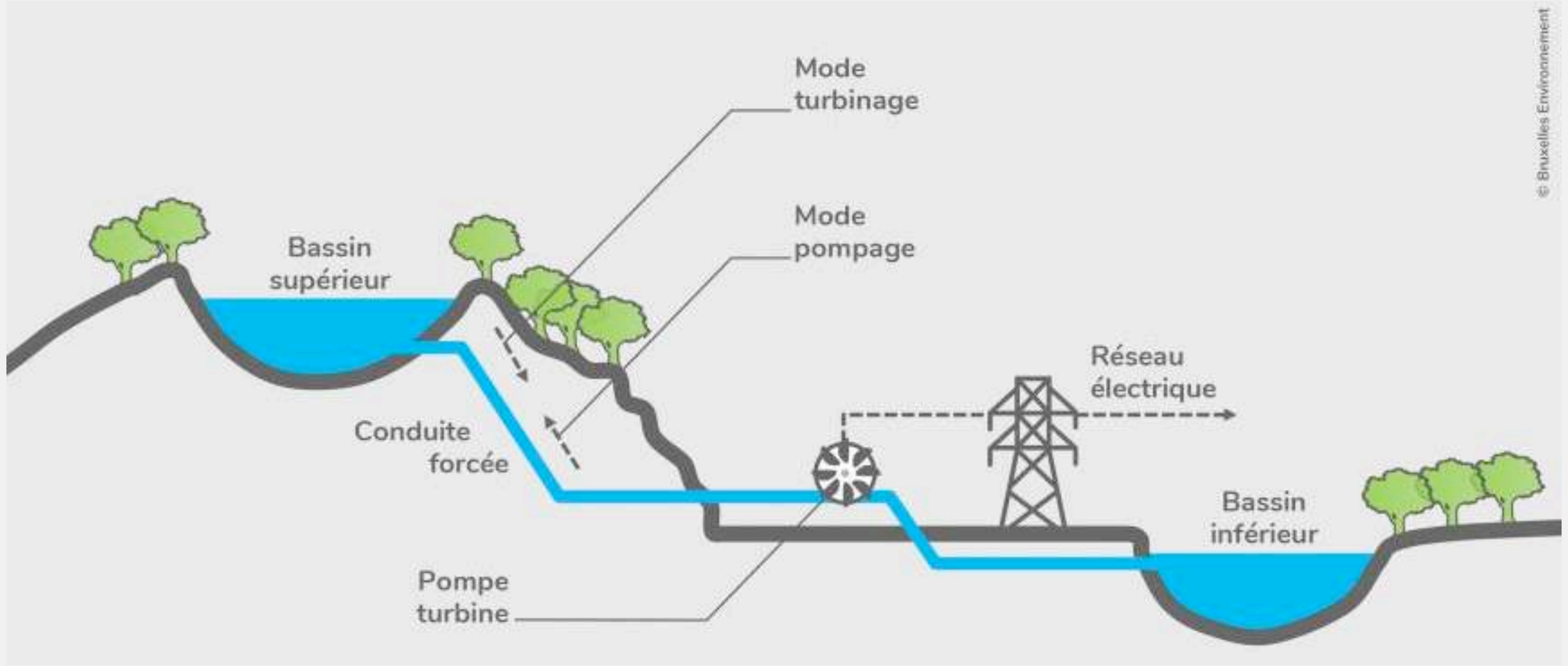
strong uncertainties (aquifer recharge rate), approximate dynamics(quality) and sustainability models

[Hassan et al. Mapping the optimization of groundwater abstraction research: A bibliometric review in the context of South Asian region. Heliyon 2023]

Hydropower

ex4: hydro unit commitment

schedule pumps and turbine
to ensure flow conservation
and maintain reservoir level in their limits
w.r.t strategic constraints (load balance, ramp, irrigation)
while maximizing the power production value



(lagrangian) subproblem of day-to-day unit commitment encompassing national power systems

ex4: hydro unit commitment

flow q_{it}, volume v_{it}, power production/consumption p_{it} in plant i at time t

nonlinear flow-power relation ϕ (turbine), disjunctive flow domains

volume conservation and limits in reservoirs

$$\max \sum_{i \in I} \sum_{t \in T} \lambda_{it} p_{it} \qquad (1)$$

$$p_{it} = \Phi(q_{it}, v_{it}) \qquad \forall t, \forall i \qquad (2)$$

$$v_{it} = v_{i(t-1)} + I_{it} + \Delta T(-q_{it} + \sum_{r \in I_i^+} q_{r(t-1)} - \sum_{r \in I_i^-} q_{r(t-1)}) \qquad \forall t, \forall i \qquad (3)$$

$$q_{it} \in \{Q_i^-\} \cup \{0\} \cup [\underline{Q}_i, \overline{Q}_i] \qquad \forall t, \forall i \qquad (4)$$

$$\underline{V}_i \leq v_{it} \leq \overline{V}_i \qquad \forall t, \forall i \qquad (5)$$

[Taktak & d'Ambrosio. An overview on mathematical programming approaches for the deterministic unit commitment problem in hydro valleys. Energy Sys 2017]

conclusion

- huge diversity of water systems & processes
- management involves decision involves optimization, e.g. maximize sustainability
- mathematical optimization as a low-tech solution (except computation and data acquisition) to get as much out of existing investments
- uncertain forecasts, intricated systems, nonlinear dynamics, fuzzy objectives: trade-off between accurate models and efficient algorithms

future

- modelling sustainability accurately
- from simulation (*what if*) to optimization (*what should*)
- short/long-term model coupling: time-scale reconciliation