The Voice of Optimization

Bartolomeo Stellato

joint work with Dimitris Bertsimas

MIT MANAGEMENT SLOAN SCHOOL

UC Berkeley, 25 Oct 2019

Inventory management

$$
\begin{array}{ll}
\text{minimize} & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\
\text{subject to} & x_{t+1} = x_t + u_t - d_t \\
& x_0 = x_{\text{init}} \\
& 0 \leq u_t \leq M
\end{array}
$$

Inventory management

$$
\begin{aligned}
\text{minimize} \quad & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\
\text{subject to} \quad & x_{t+1} = x_t + u_t - d_t \\
& x_0 = x_{\text{init}} \\
& 0 \leq u_t \leq M
\end{aligned}
$$

Inventory

Inventory management

$$
\begin{aligned}
\text{minimize} \quad & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\
\text{subject to} \quad & x_{t+1} = x_t + u_t - d_t \\
& x_0 = x_{\text{init}} \\
& 0 \leq u_t \leq M
\end{aligned}
$$

Inventory (x_{t+1})

Order (u_t)

Inventory management

$$\begin{array}{ll} \text{minimize} & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\ \text{subject to} & x_{t+1} = x_t + u_t - d_t \\ & x_0 = x_{\text{init}} \\ & 0 \le u_t \le M \end{array}$$

Inventory (x_{t+1})

Order (u_t)

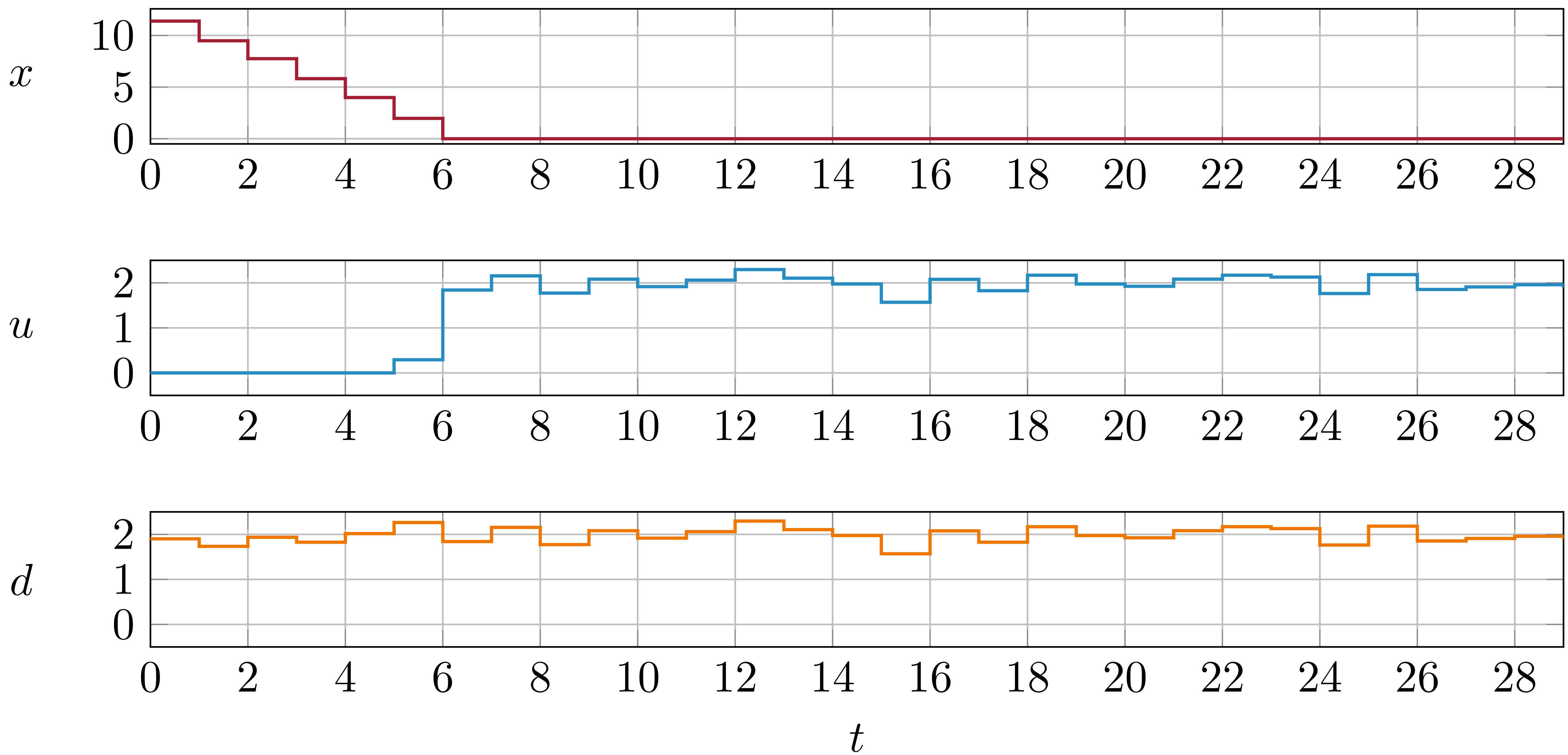
Demand (d_t)

Inventory management

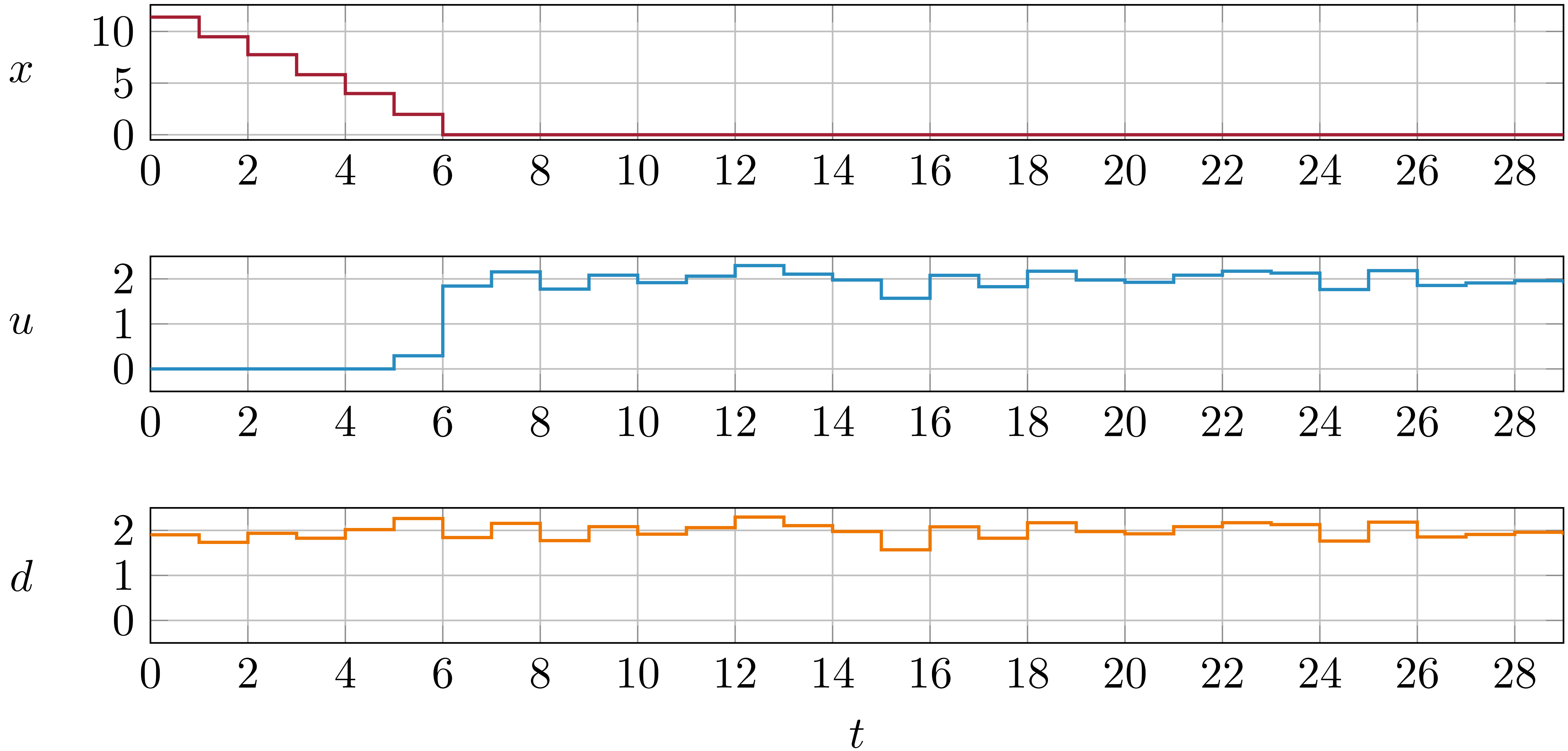
$$
\begin{array}{ll}
\text{minimize} & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\
\text{subject to} & x_{t+1} = x_t + u_t - d_t \\
& x_0 = x_{\text{init}} \\
& 0 \leq u_t \leq M
\end{array}
$$

Parameters

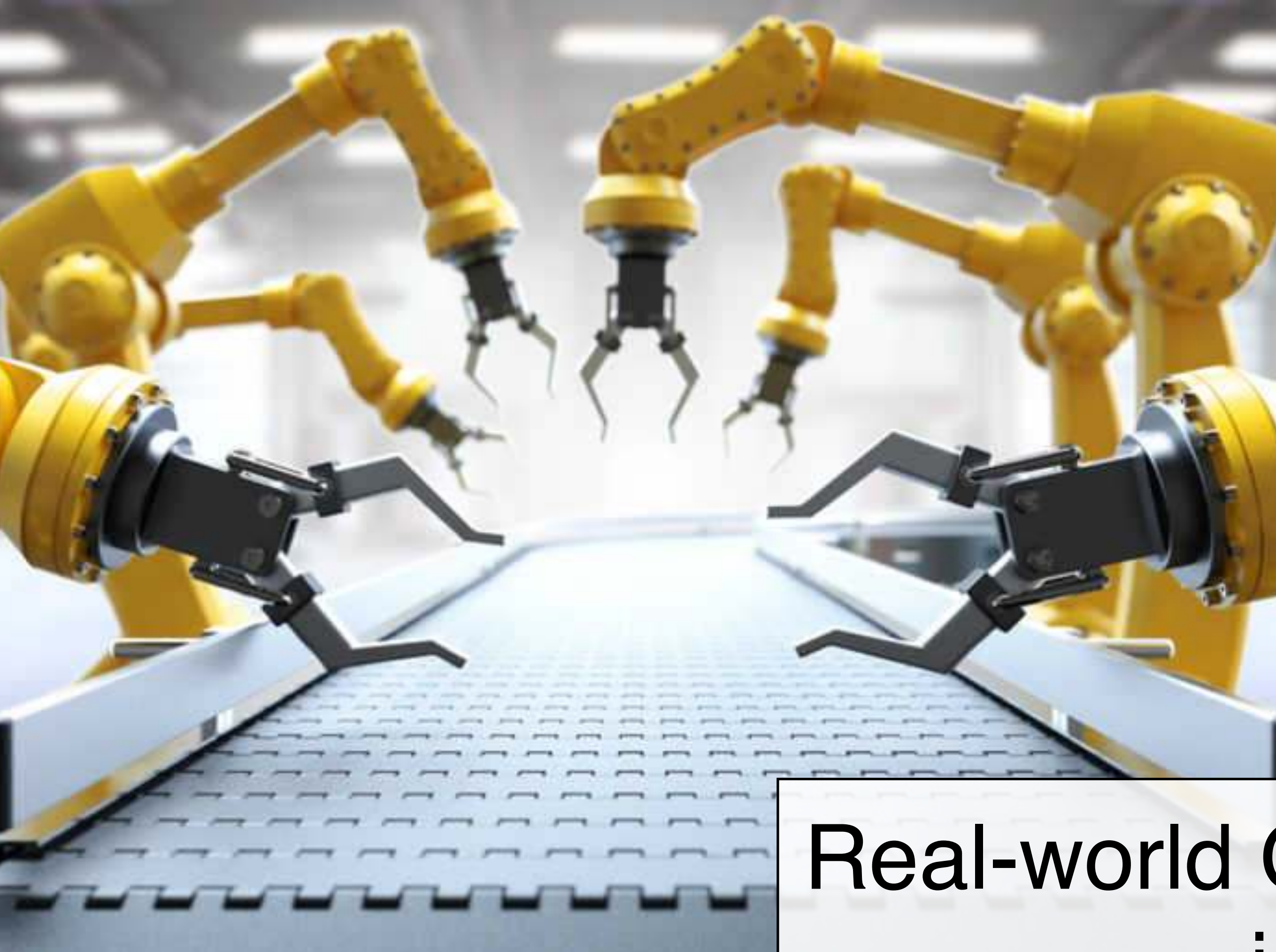
How do x and u depend on the parameters?



How do x and u depend on the parameters?



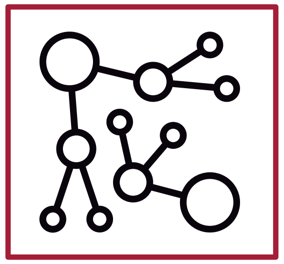
x_{init} ? d_t ?



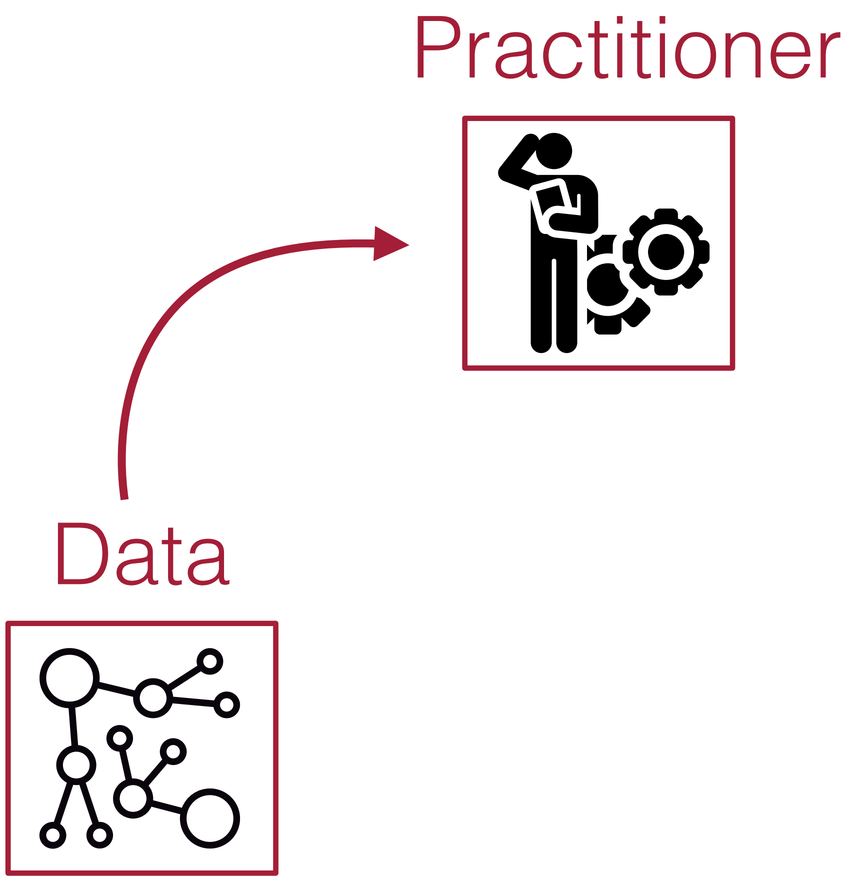
Real-world Optimization
is
Parametric

The decision-making workflow

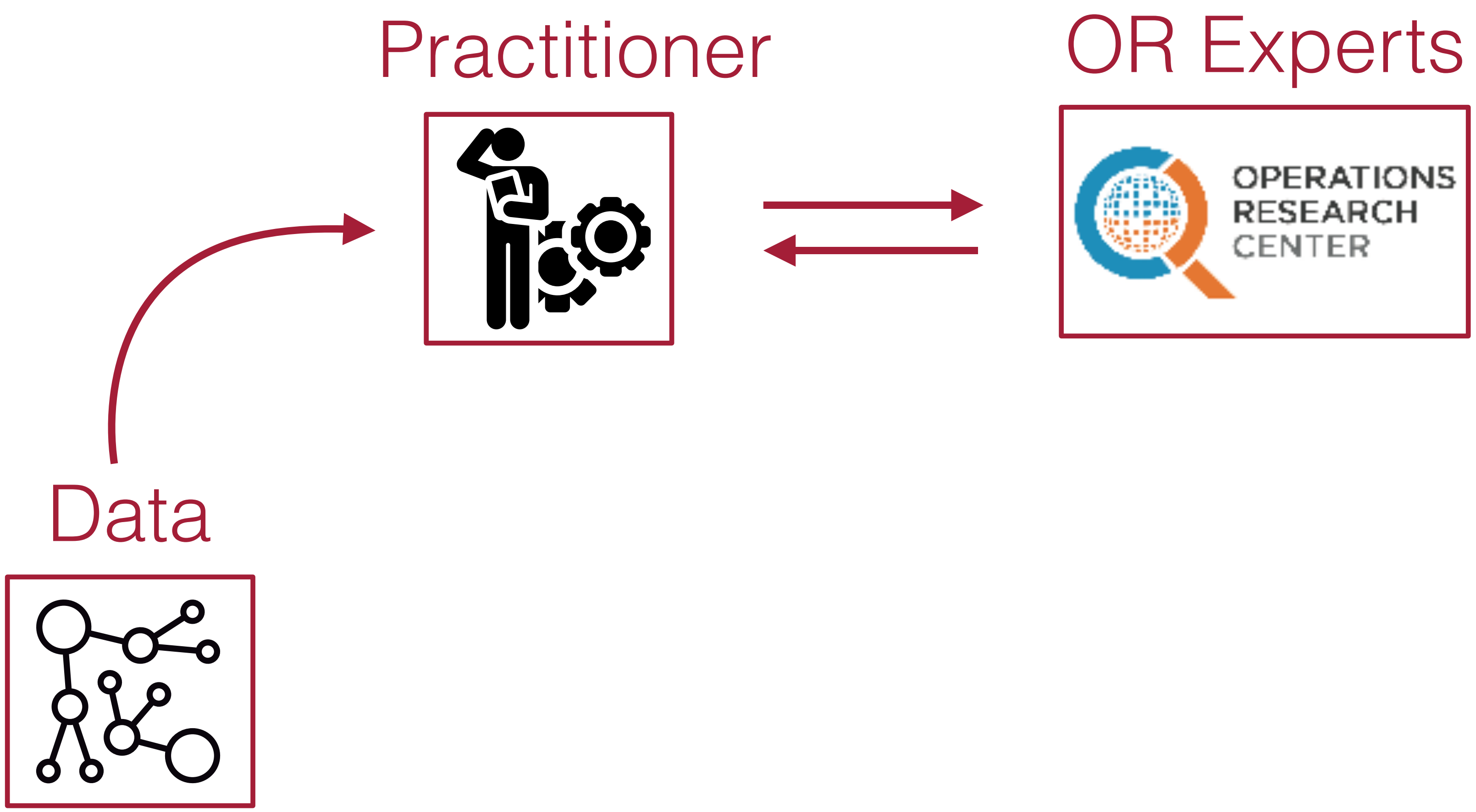
Data



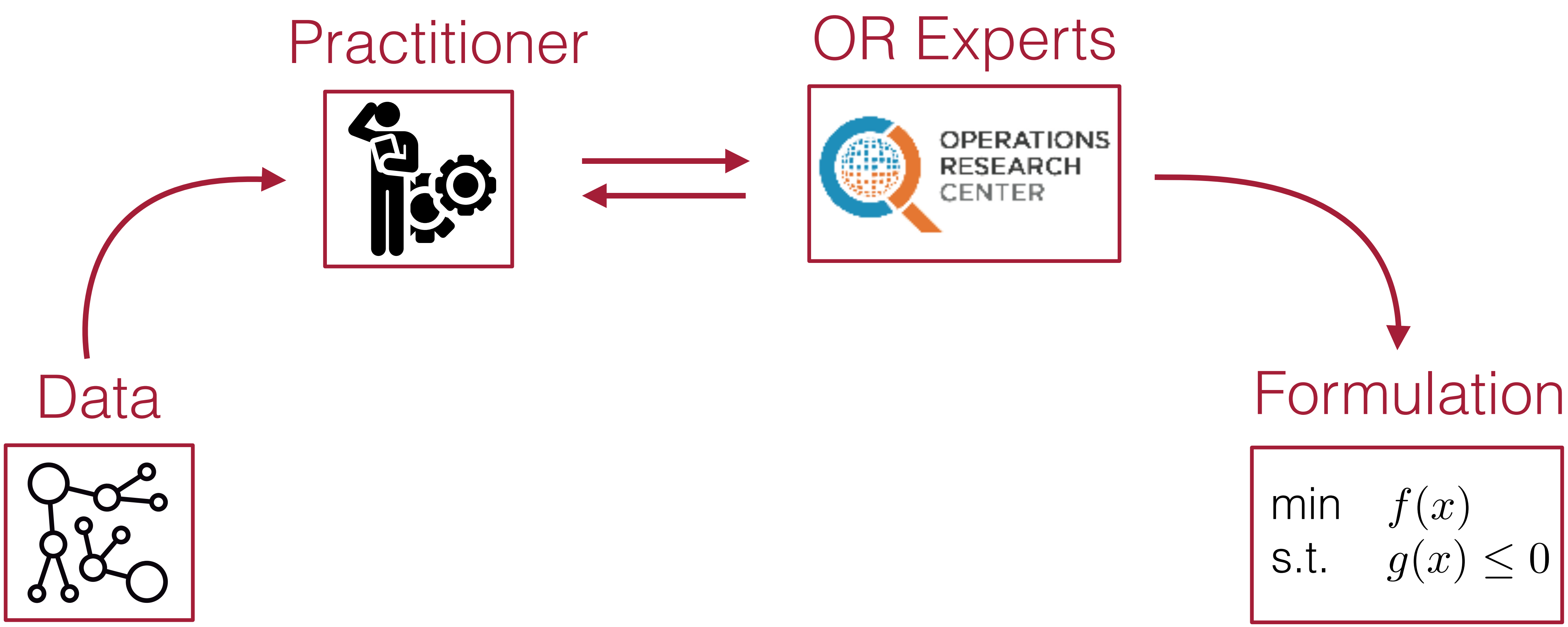
The decision-making workflow



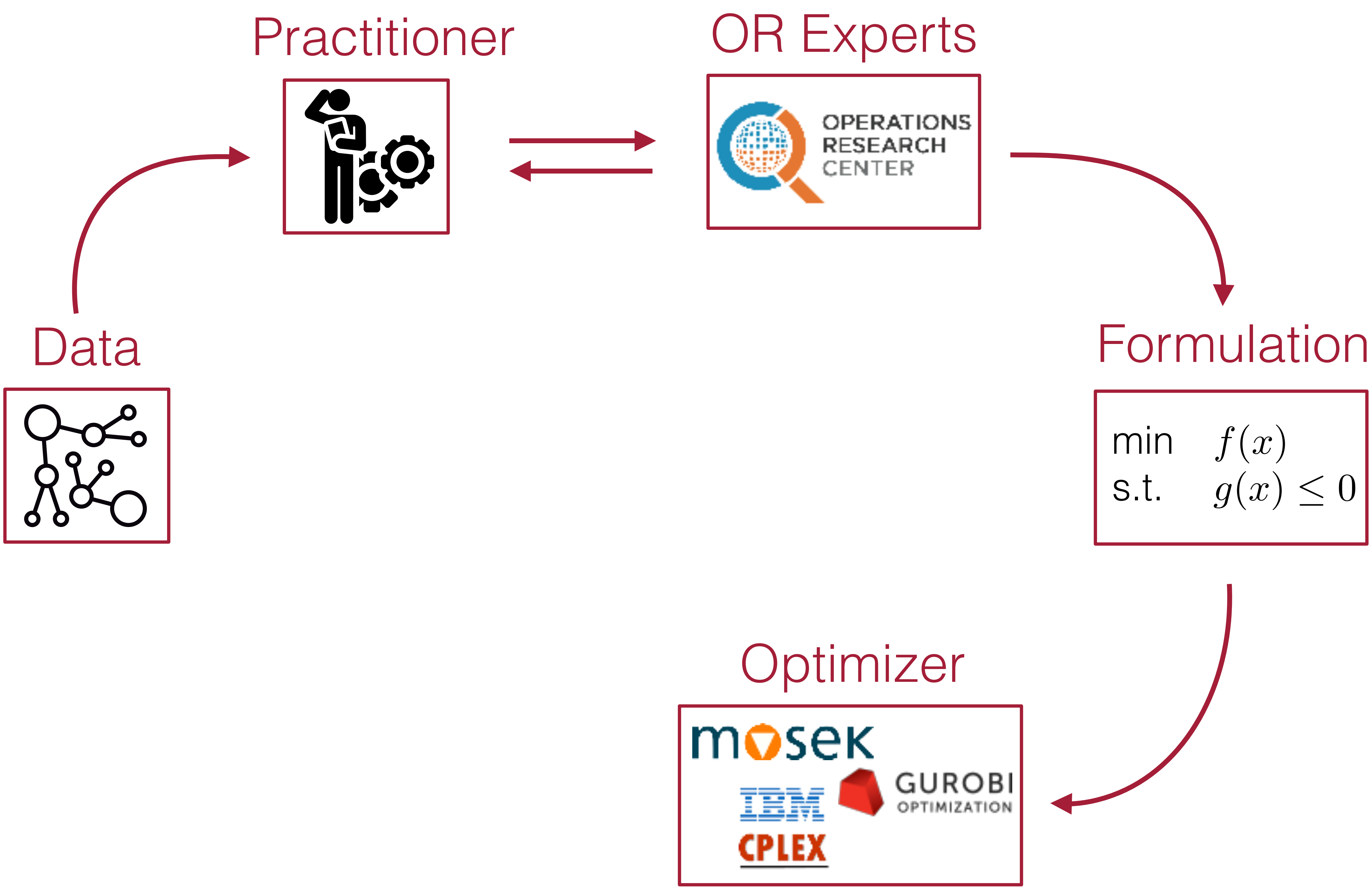
The decision-making workflow



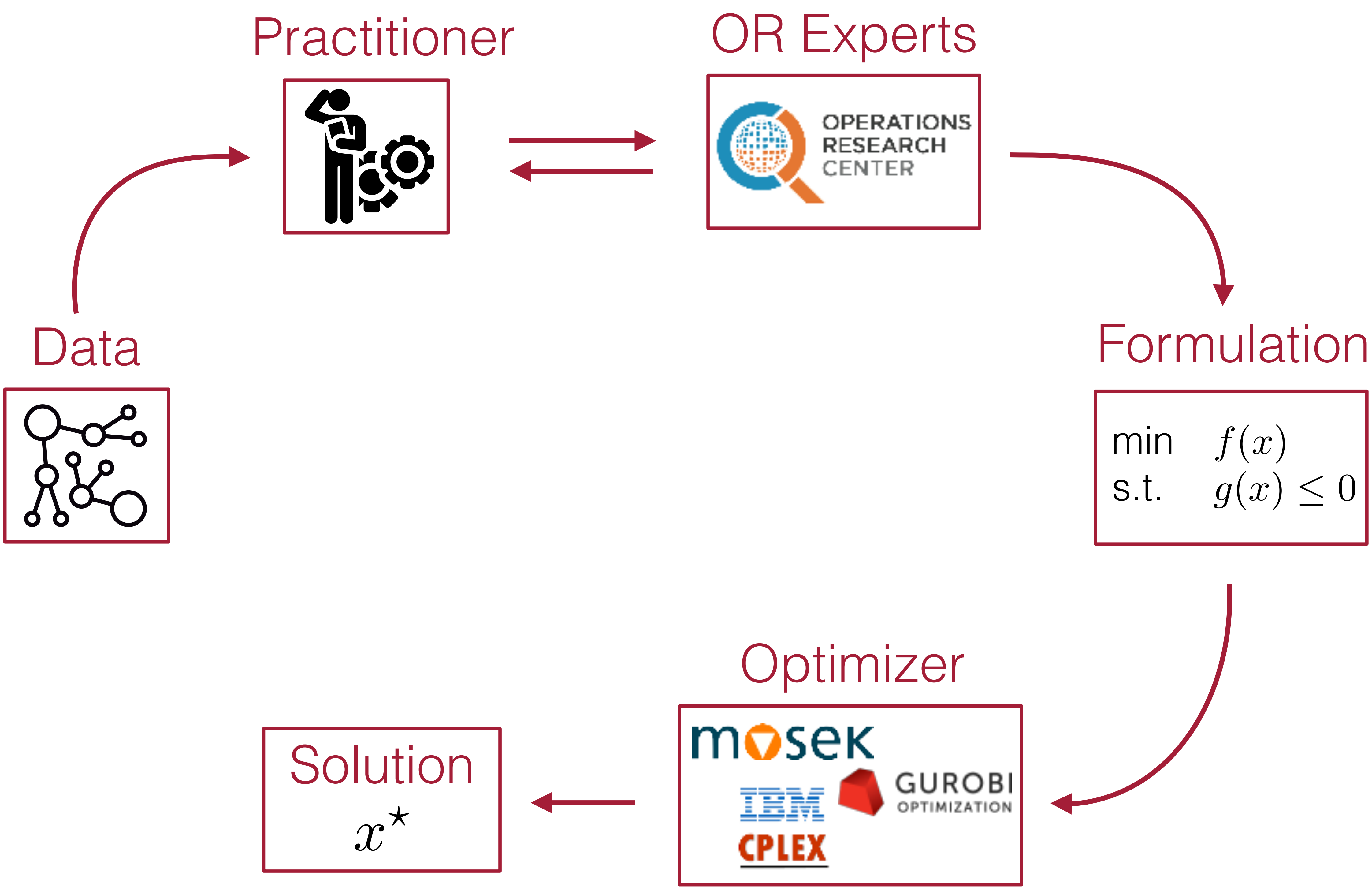
The decision-making workflow



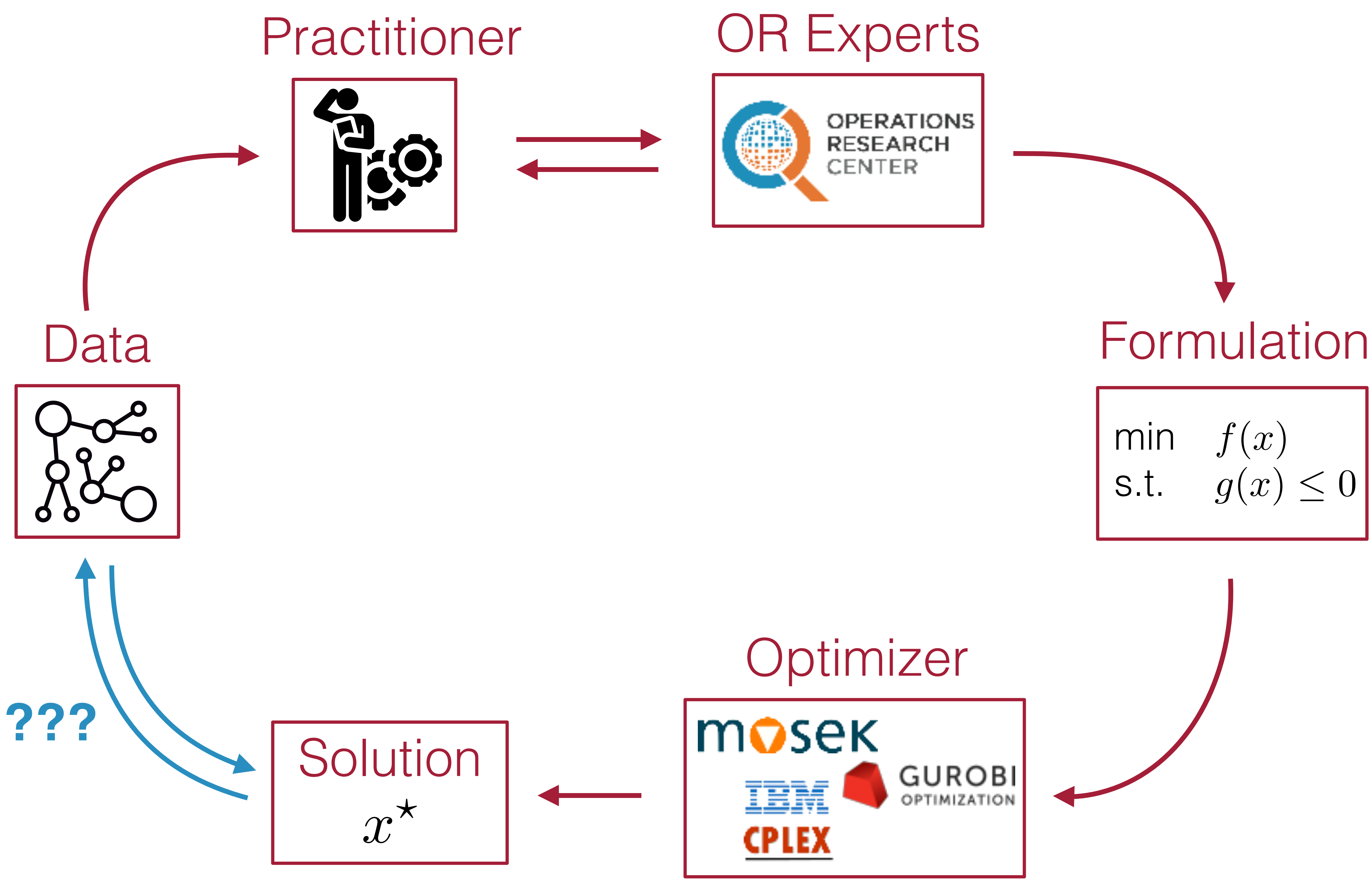
The decision-making workflow



The decision-making workflow

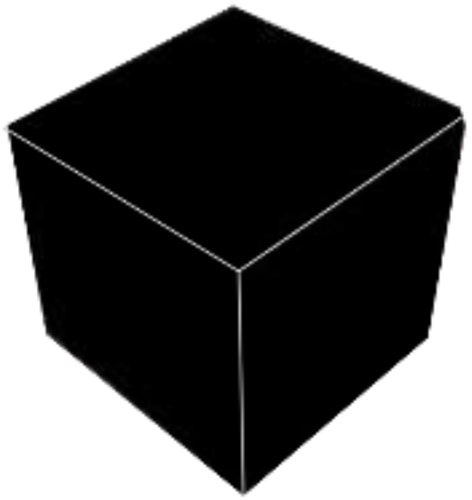


The decision-making workflow

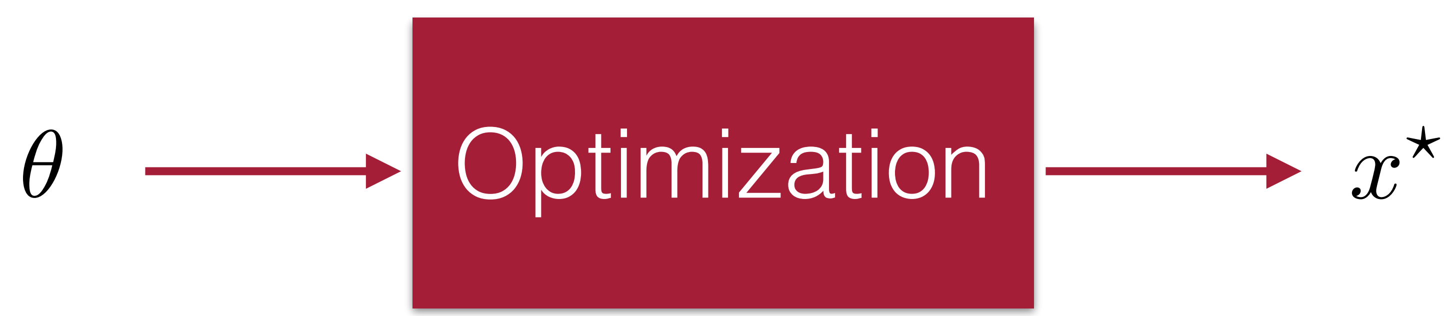


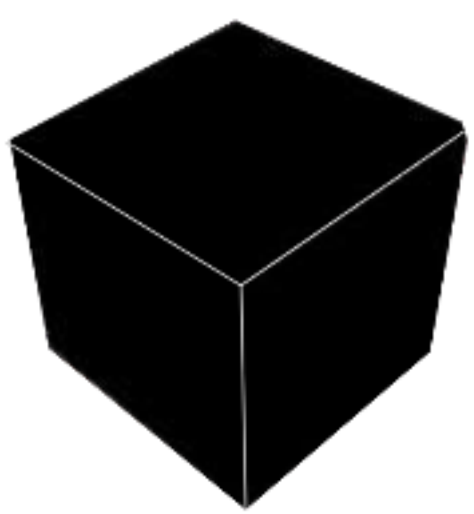
Traditional optimization

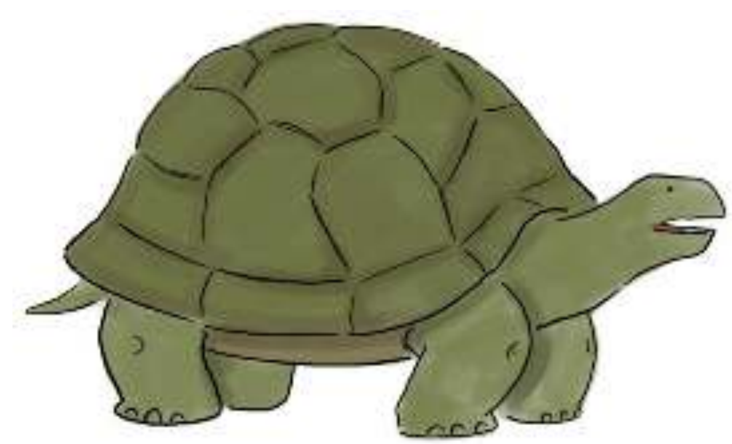
Traditional optimization



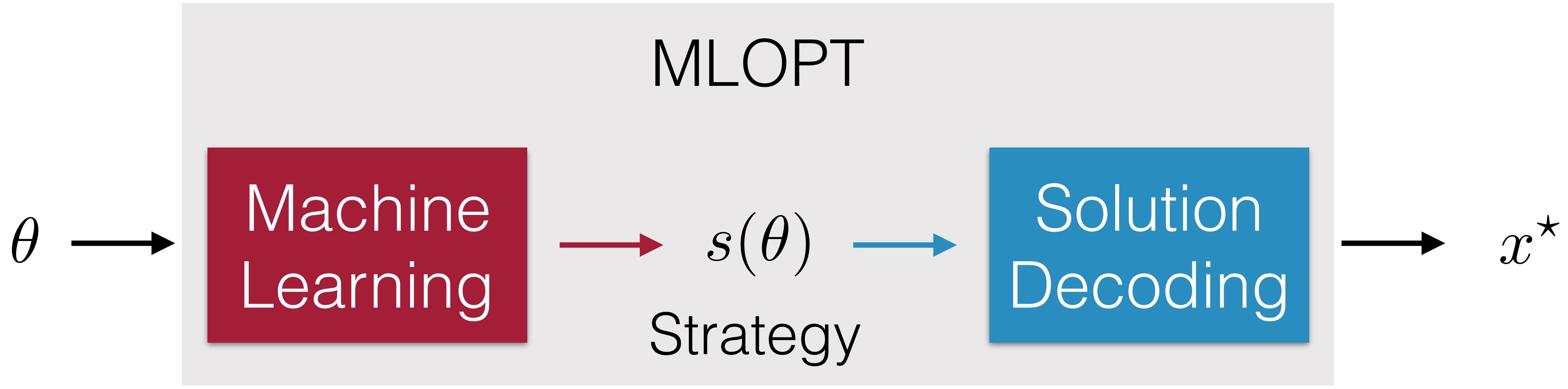
Traditional optimization



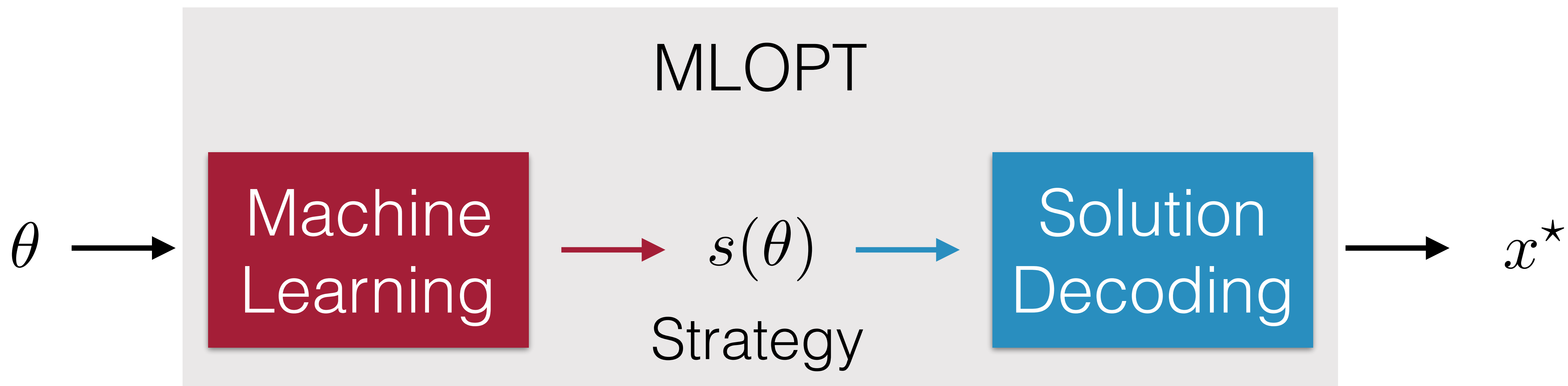


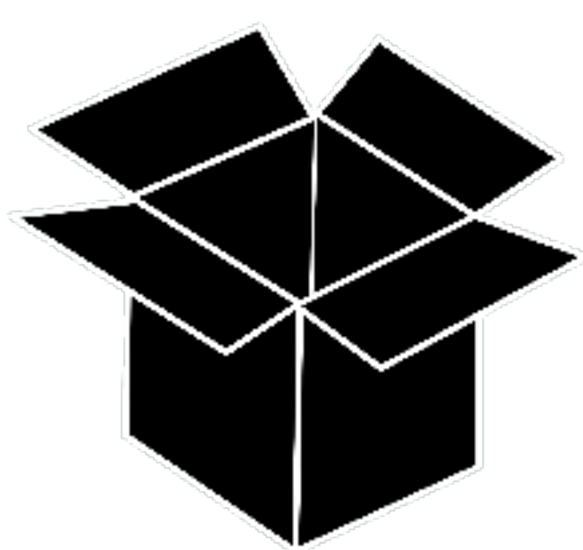


Machine Learning Optimizer

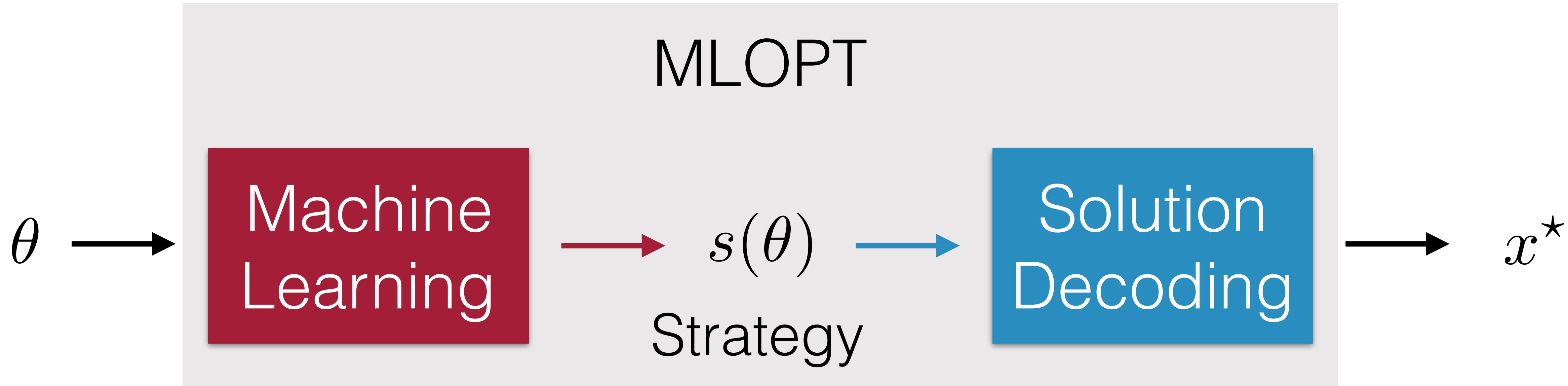


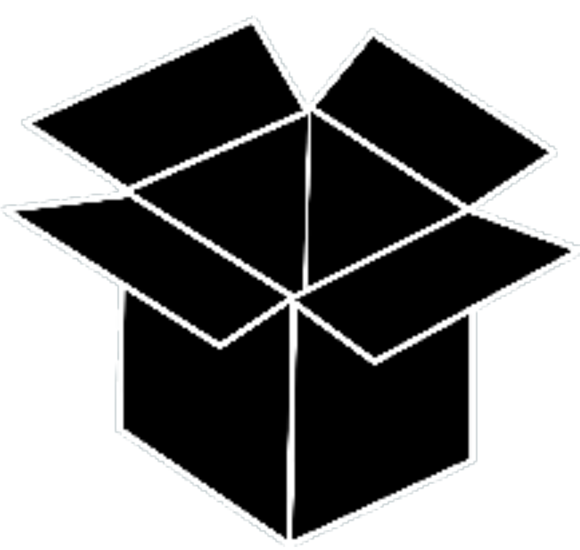
Machine Learning Optimizer





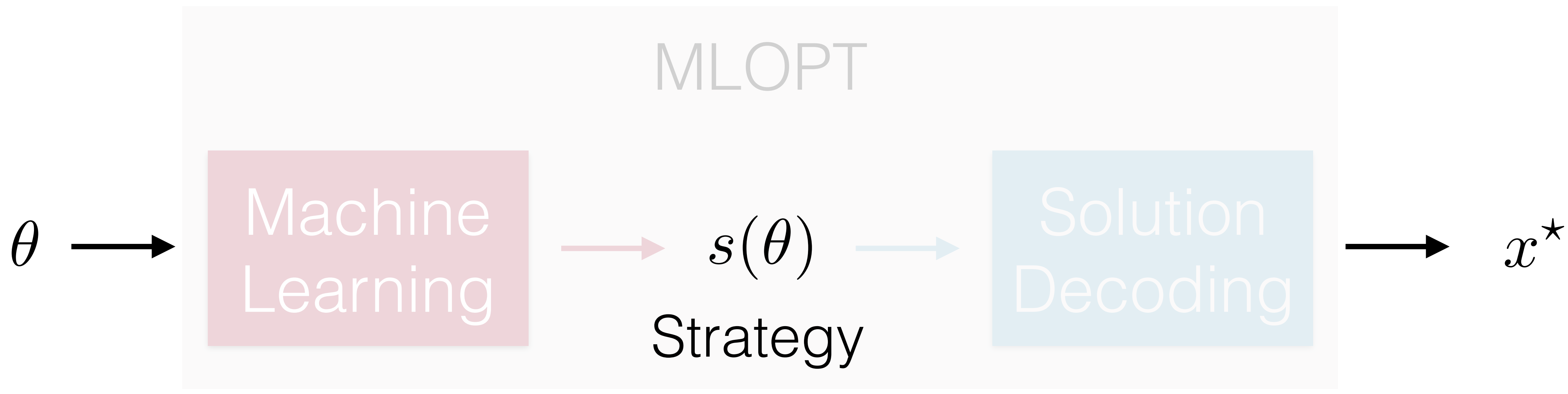
Machine Learning Optimizer





Optimal strategies

Strategy

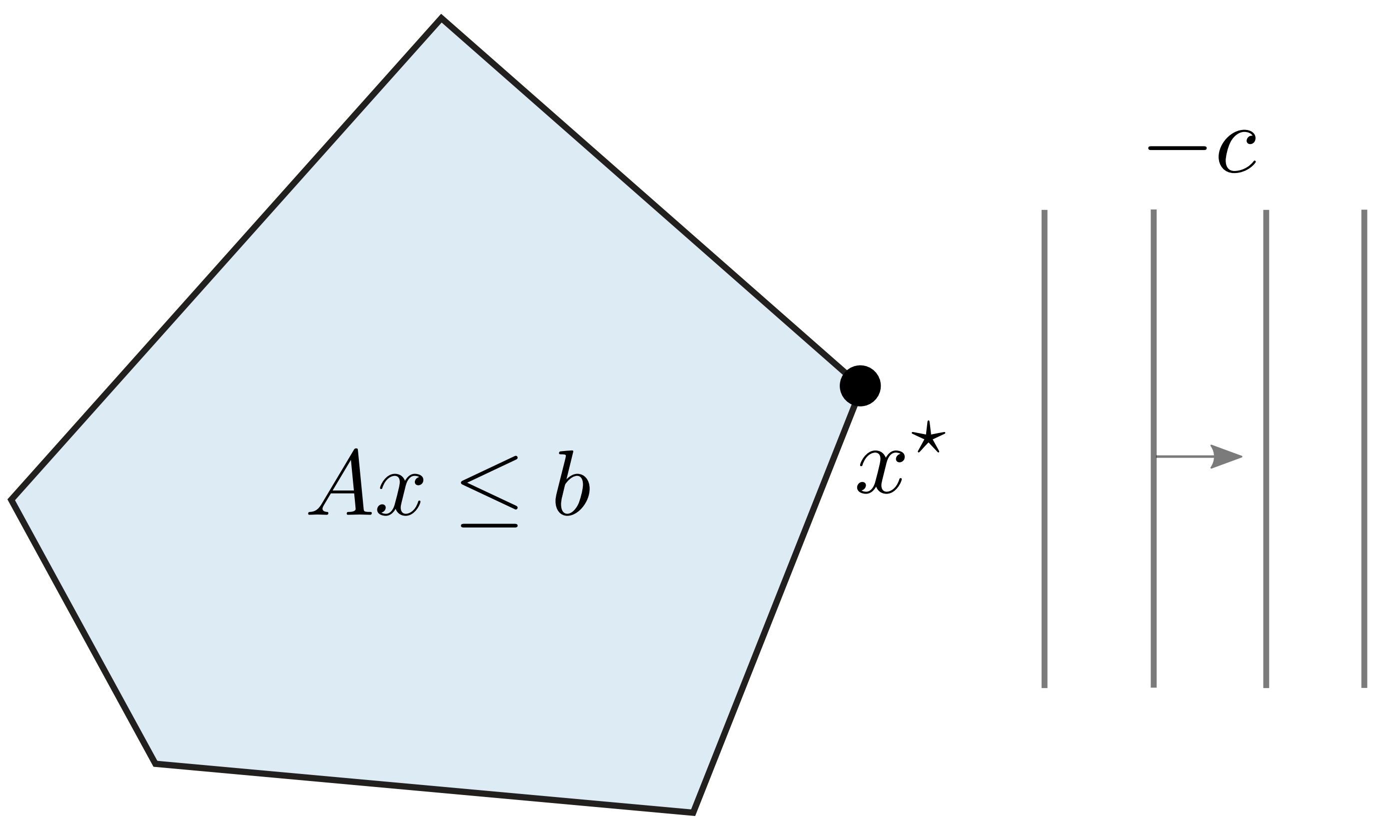


The complete information to efficiently get the optimal solution

$$s(\theta)$$

Linear optimization

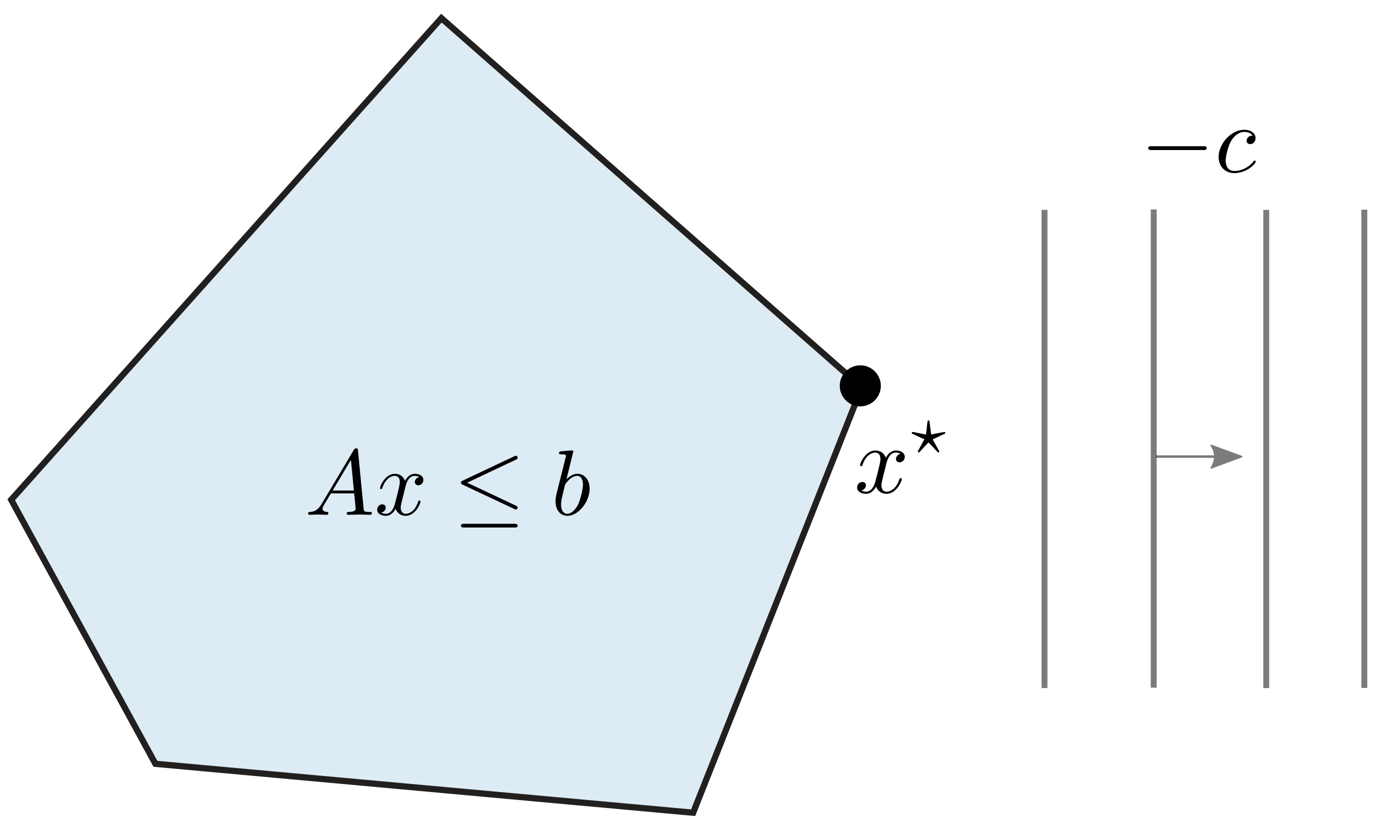
$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \end{array}$$



Linear optimization

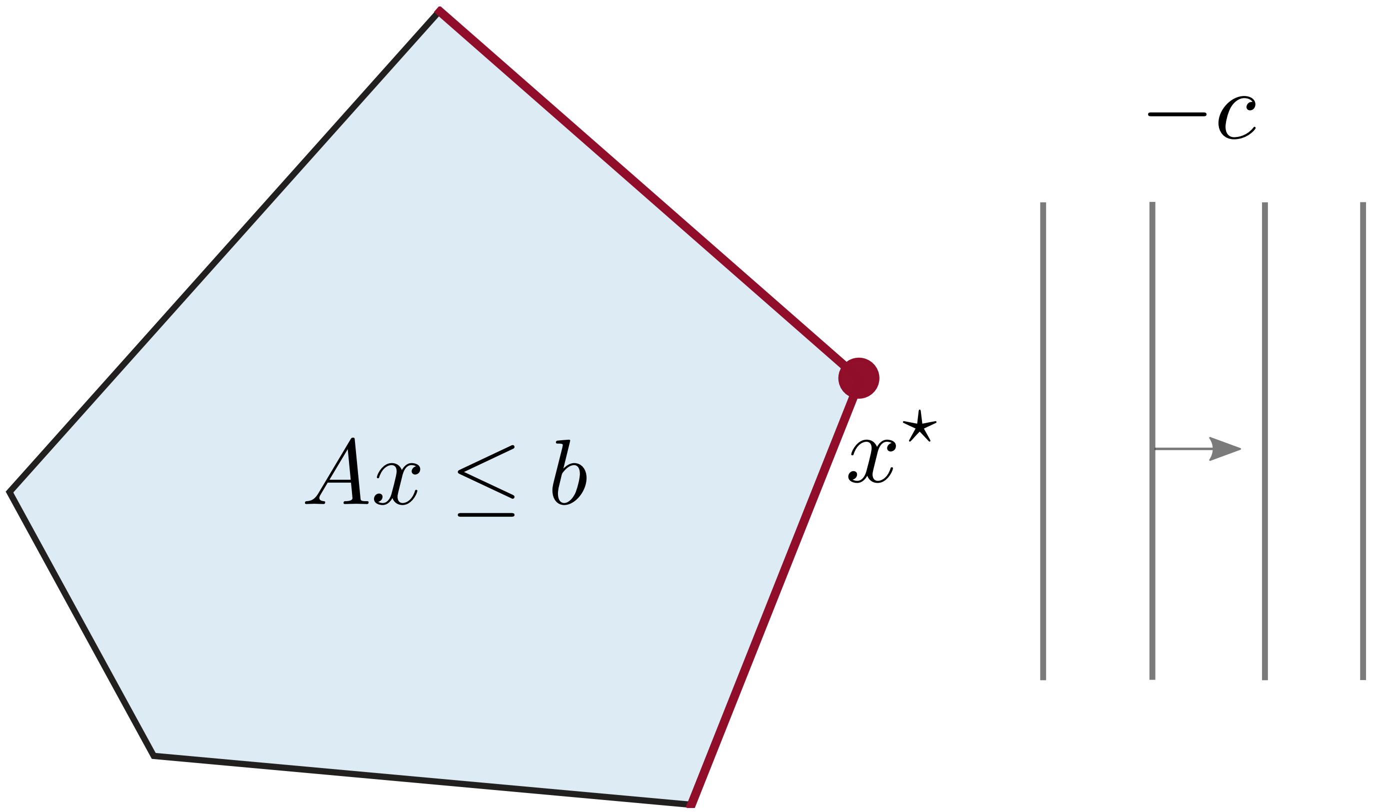
$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \end{array}$$

Parameters



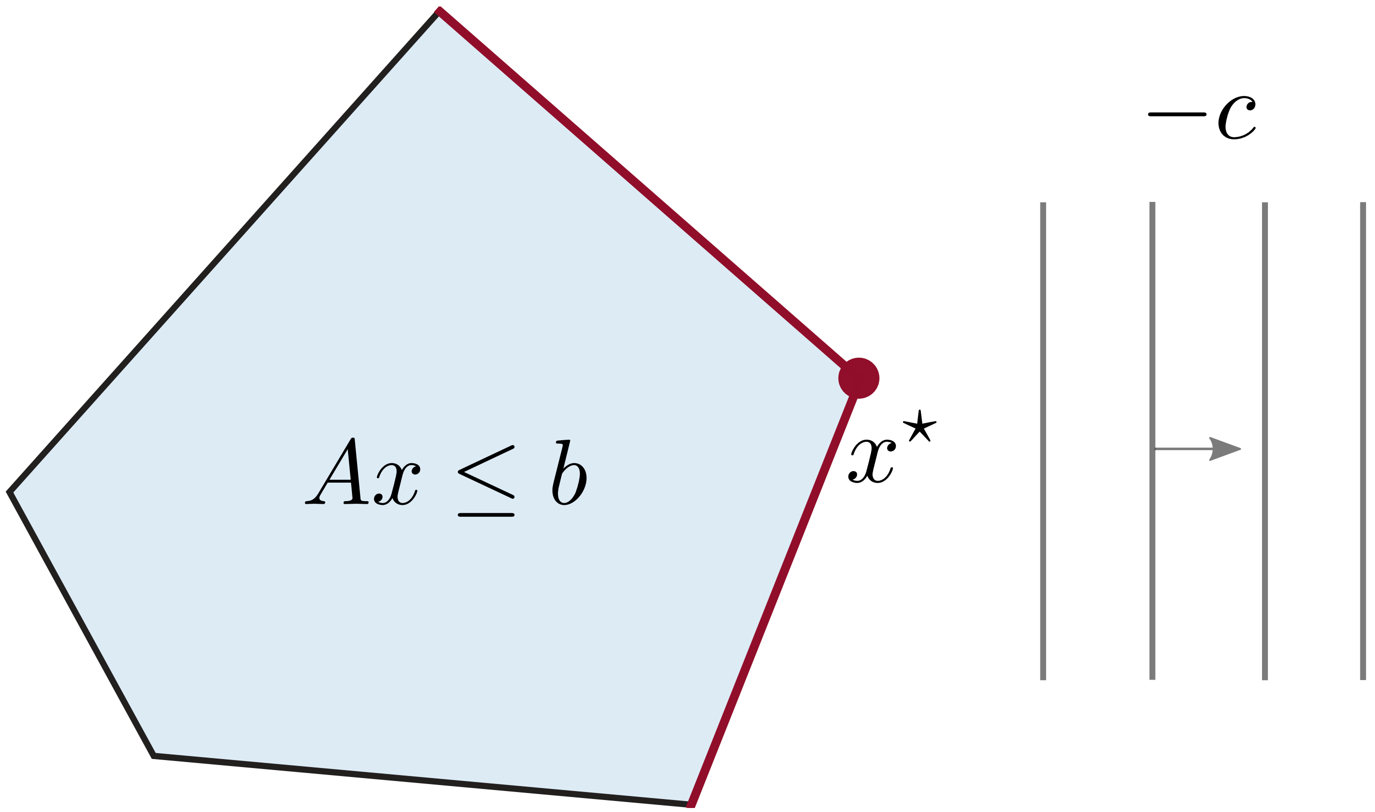
Tight constraints

$$\mathcal{T}(\theta) = \{i \mid A_i(\theta)x^\star = b_i(\theta)\}$$



Tight constraints

$$\mathcal{T}(\theta) = \{i \mid A_i(\theta)x^\star = b_i(\theta)\}$$

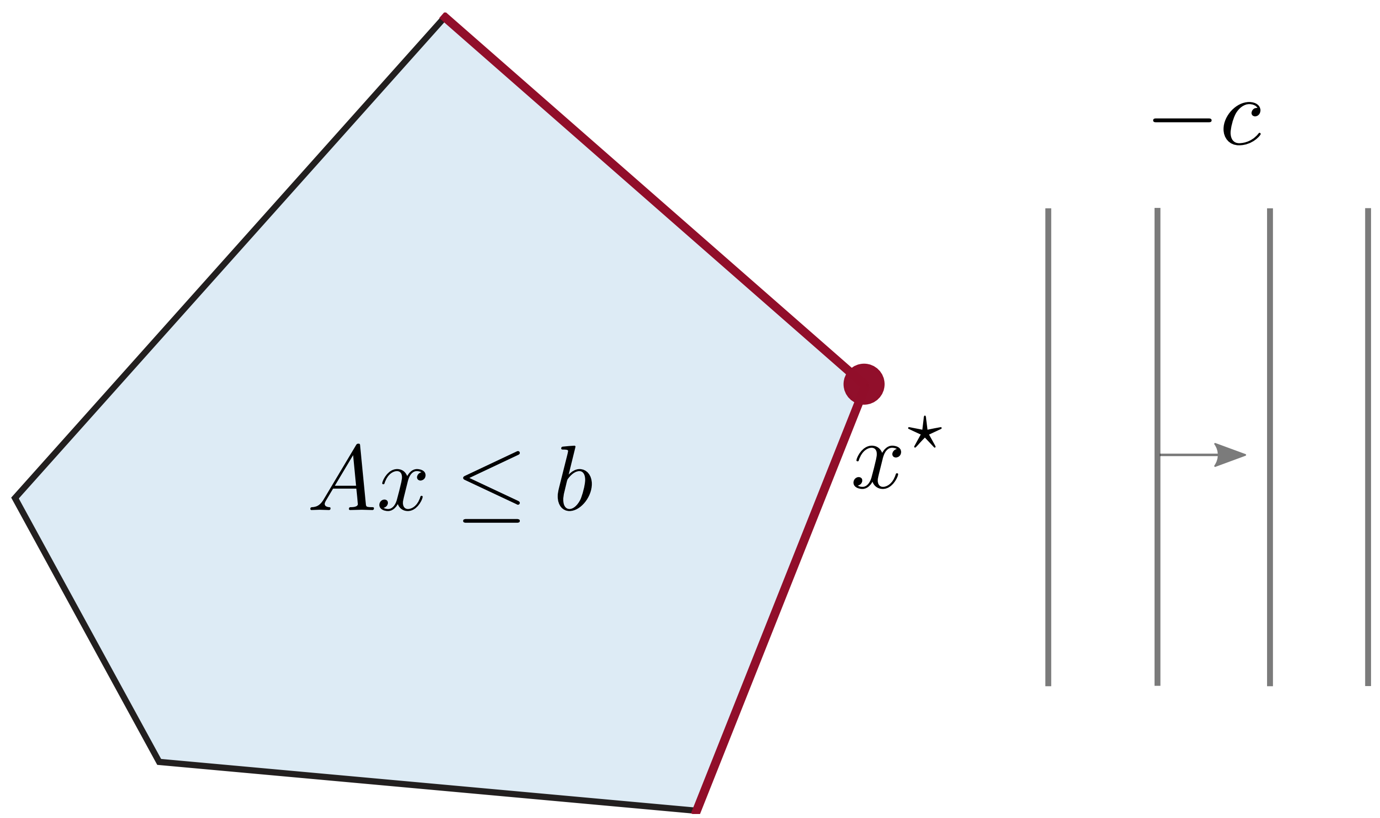


$|\mathcal{T}(\theta)| = \#$ variables

if non-degenerate

Tight constraints

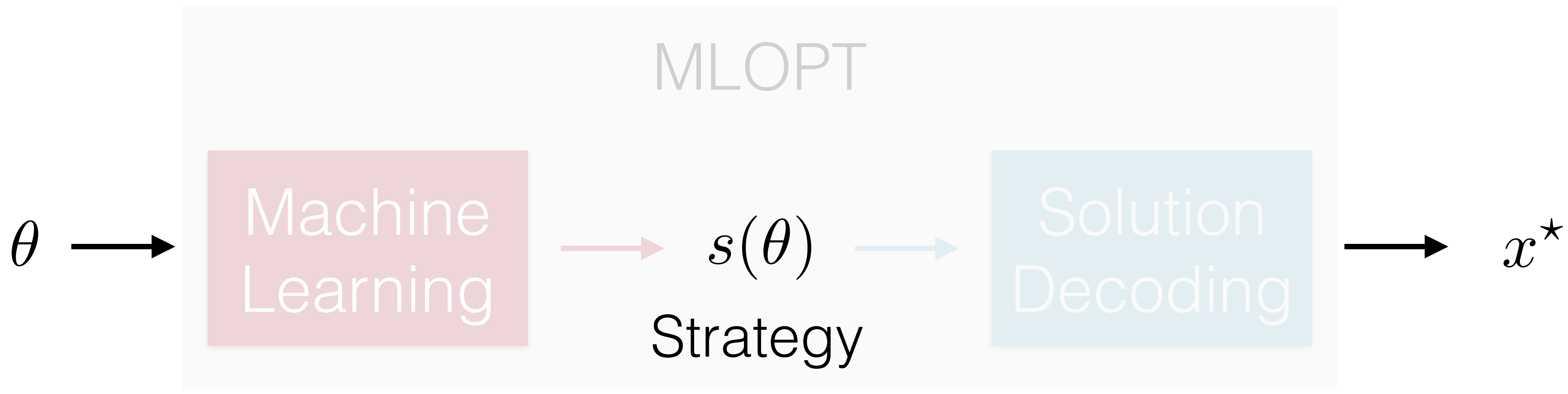
$$\mathcal{T}(\theta) = \{i \mid A_i(\theta)x^\star = b_i(\theta)\}$$



$|\mathcal{T}(\theta)| = \#$ variables — if non-degenerate

$|\mathcal{T}(\theta)| \ll \#$ constraints — in general

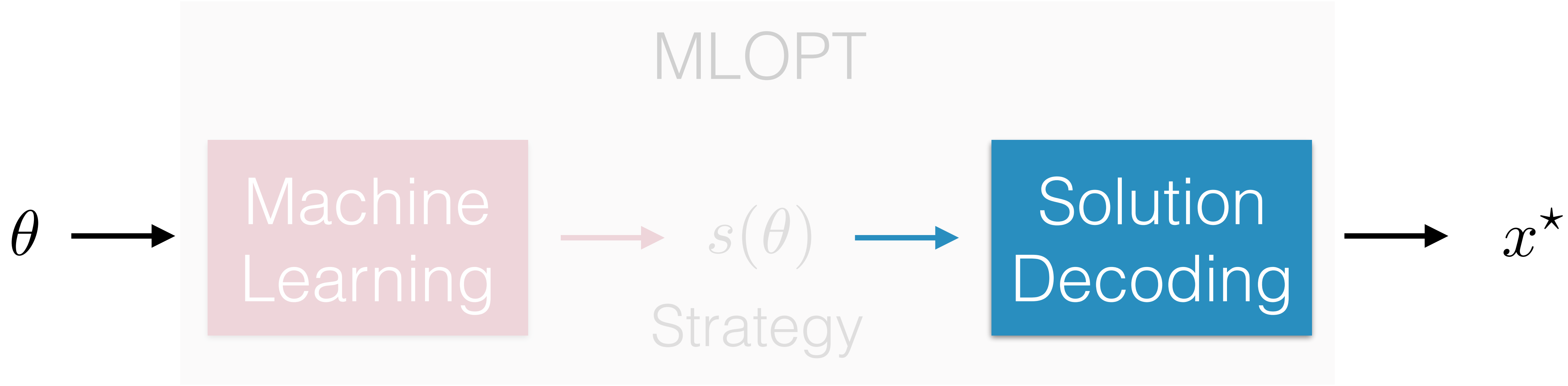
Strategy



The complete information to efficiently get the optimal solution

$$s(\theta) = \mathcal{T}(\theta)$$

Computing the solution from the optimal strategy



Computing the solution from the optimal strategy

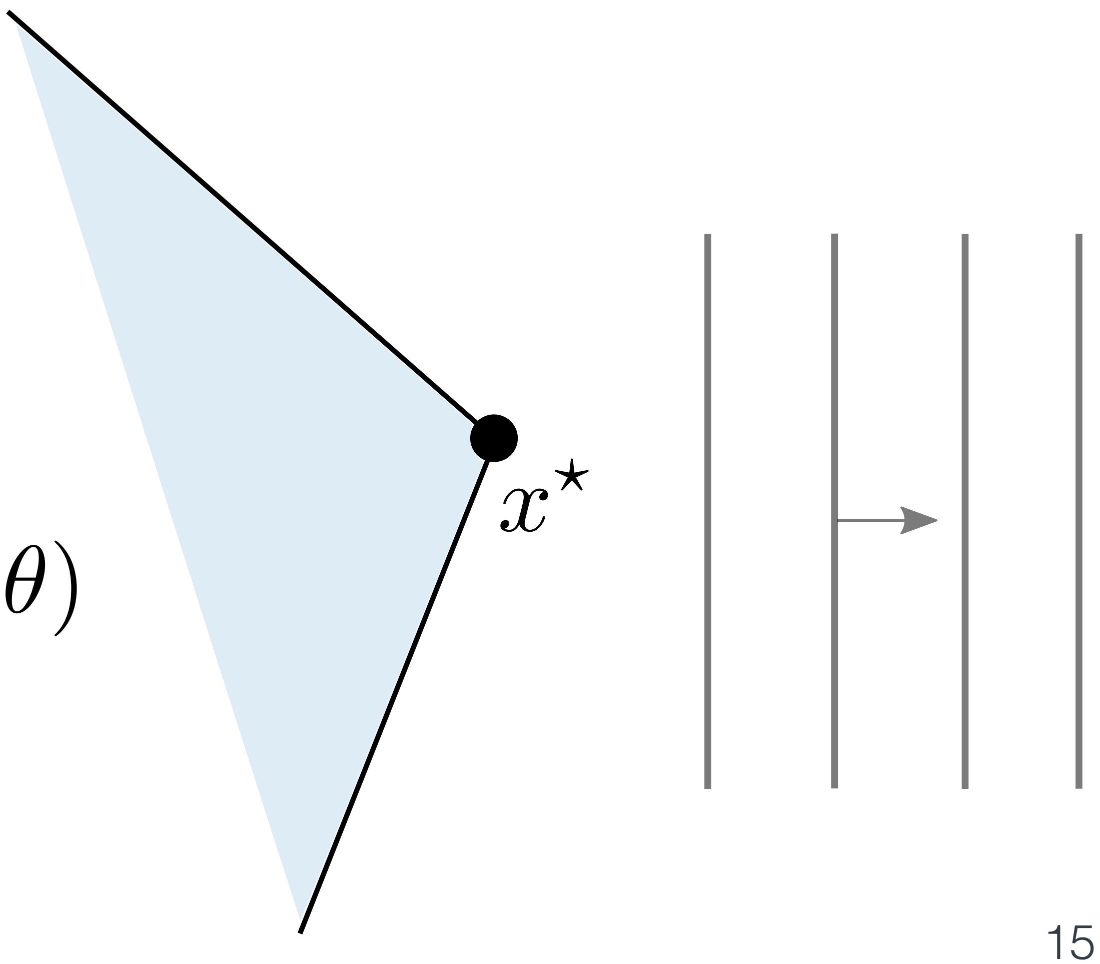
$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \end{array}$$

Computing the solution from the optimal strategy

$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \end{array}$$

Strategy $s(\theta) = \mathcal{T}(\theta)$

$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A_i(\theta)x = b_i(\theta) \quad \forall i \in \mathcal{T}(\theta) \end{array}$$



Inventory management

$$\begin{aligned}\text{minimize} \quad & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\ \text{subject to} \quad & x_{t+1} = x_t + u_t - d_t \\ & x_0 = x_{\text{init}} \\ & 0 \le u_t \le M\end{aligned}$$

Inventory management

$$\begin{array}{ll} \text{minimize} & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\ \text{subject to} & x_{t+1} = x_t + u_t - d_t \\ & x_0 = x_{\text{init}} \\ & 0 \leq u_t \leq M \end{array}$$

Inventory

Inventory management

$$\begin{aligned} \text{minimize} \quad & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\ \text{subject to} \quad & x_{t+1} = x_t + u_t - d_t \\ & x_0 = x_{\text{init}} \\ & 0 \le u_t \le M \end{aligned}$$

Inventory

Order

Inventory management

$$\begin{aligned}\text{minimize} \quad & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\ \text{subject to} \quad & x_{t+1} = x_t + u_t - d_t \\ & x_0 = x_{\text{init}} \\ & 0 \le u_t \le M\end{aligned}$$

Inventory (x_{t+1})

Order (u_t)

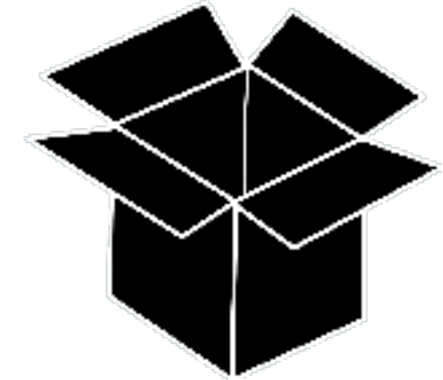
Demand (d_t)

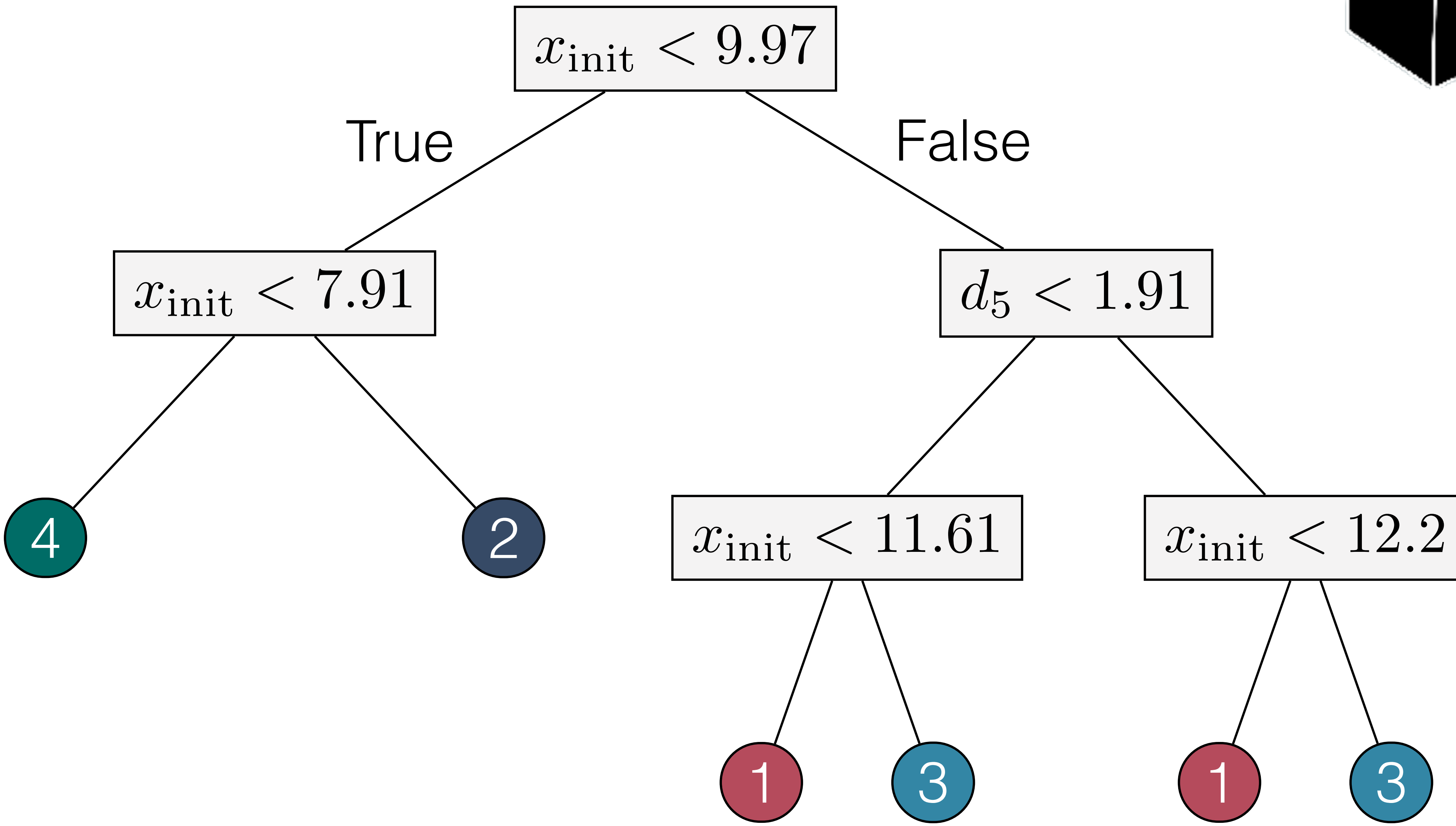
Inventory management

$$
\begin{array}{ll}
\text{minimize} & \sum_{t=0}^{T-1} h(x_t) + o(u_t) \\
\text{subject to} & x_{t+1} = x_t + u_t - d_t \\
& x_0 = x_{\text{init}} \\
& 0 \leq u_t \leq M
\end{array}
$$

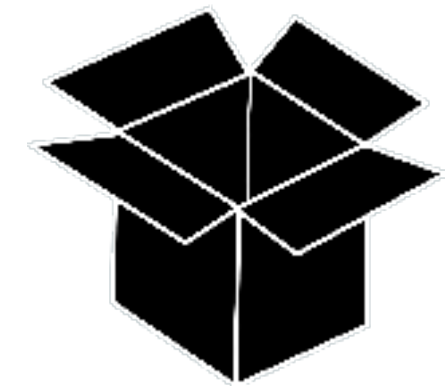
Parameters

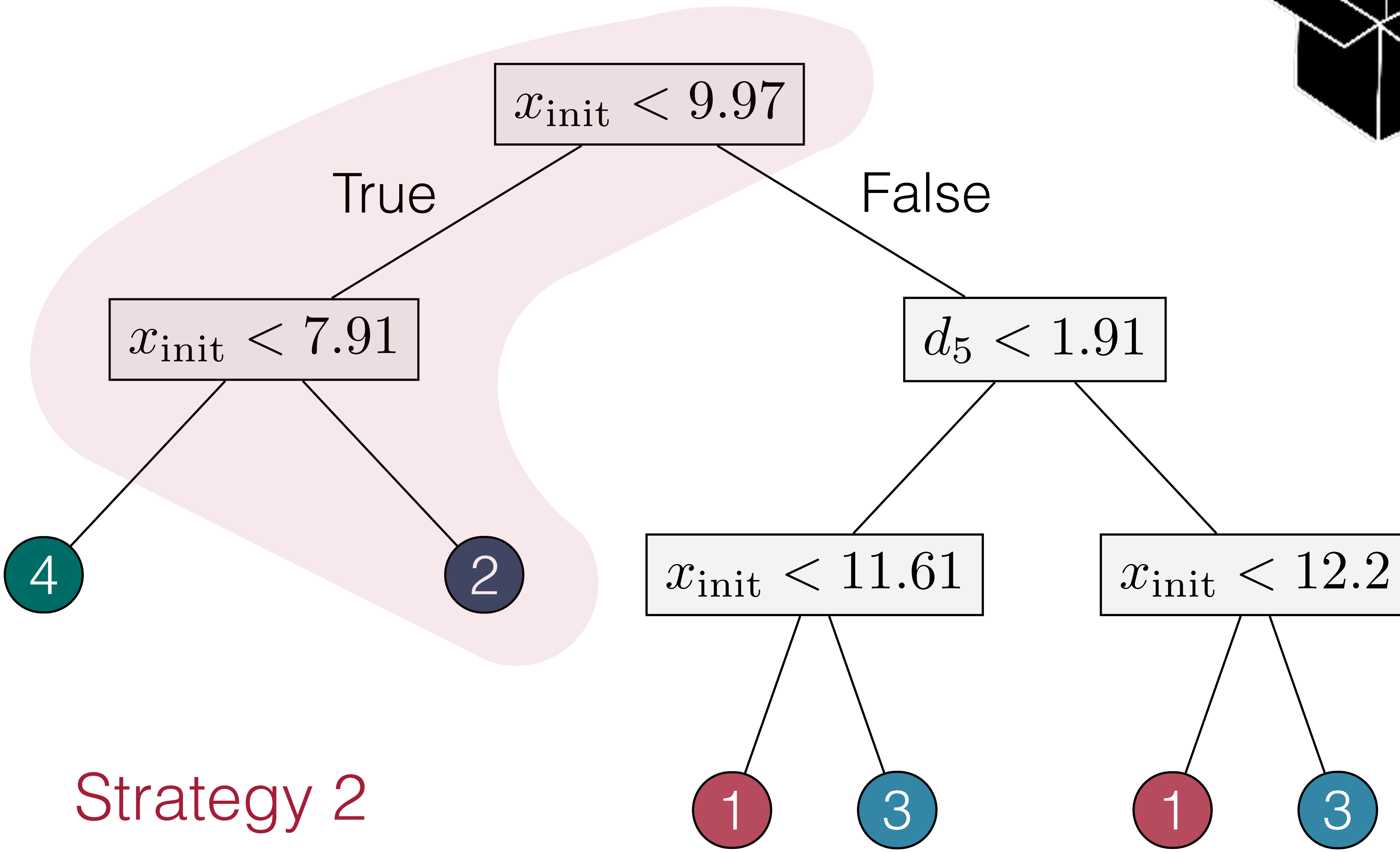
Strategy selection





Strategy selection





Strategy 2

$$u_t = 0 \qquad t \leq 4$$

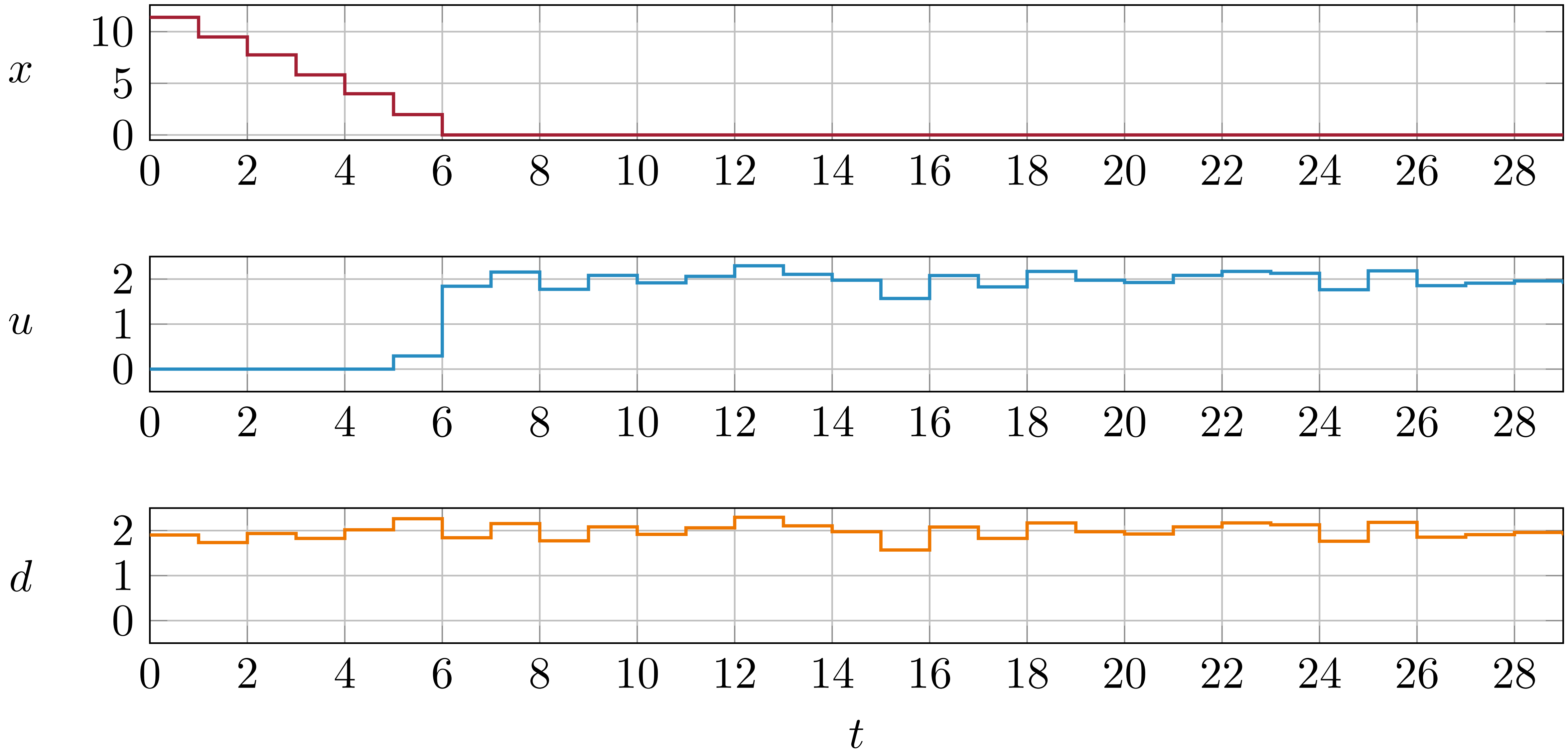
$$0 \leq u_t \leq M \qquad t > 4$$

Strategies for inventory

Strategy 2

$$u_t = 0 \qquad t \leq 4$$

$$0 \leq u_t \leq M \qquad t > 4$$



Mixed-integer optimization

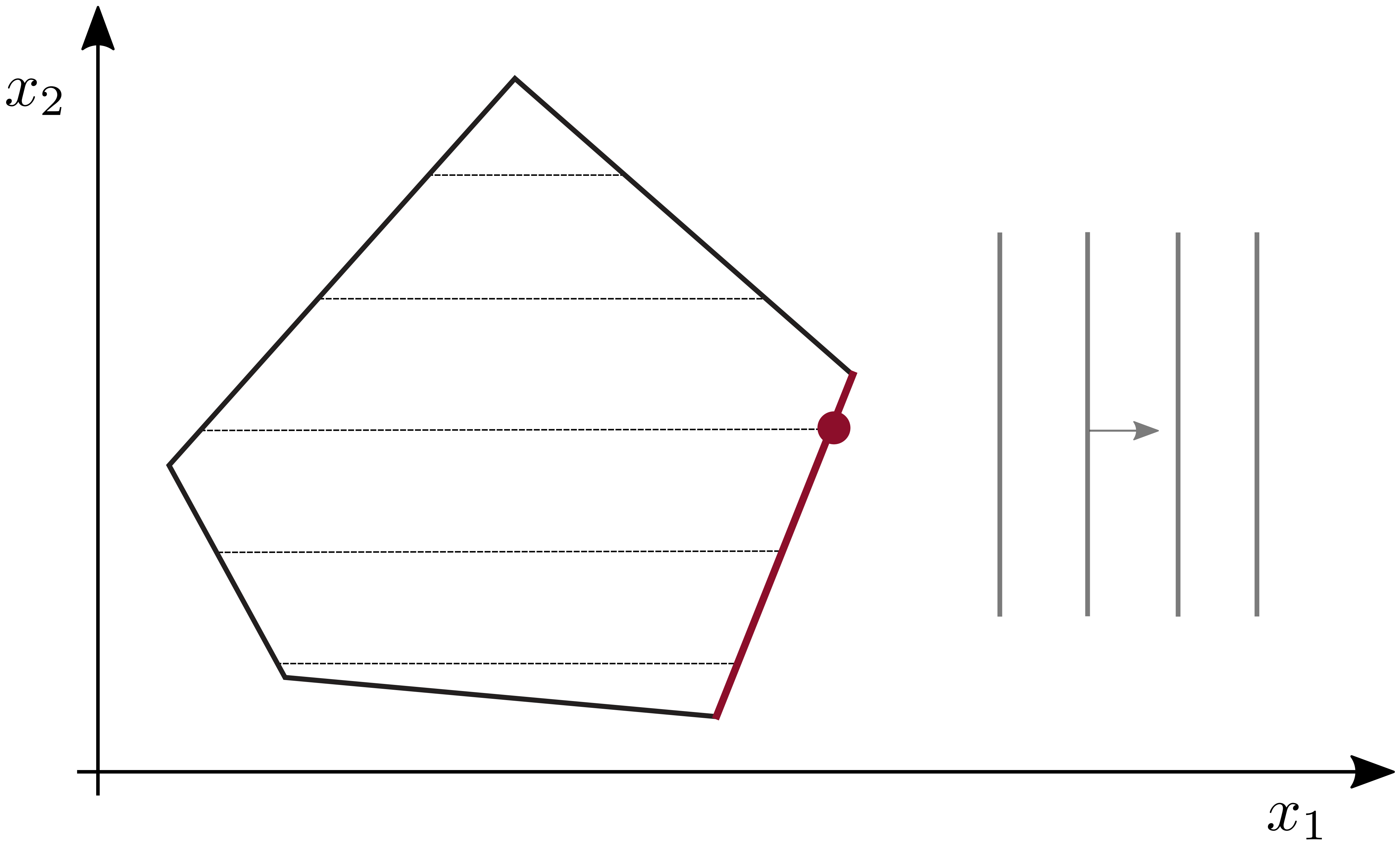
$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \\ & x_{\mathcal{I}} \in \mathbf{Z}^d \end{array}$$

Mixed-integer optimization

$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \\ & x_{\mathcal{I}} \in \mathbf{Z}^d \end{array}$$

Integers

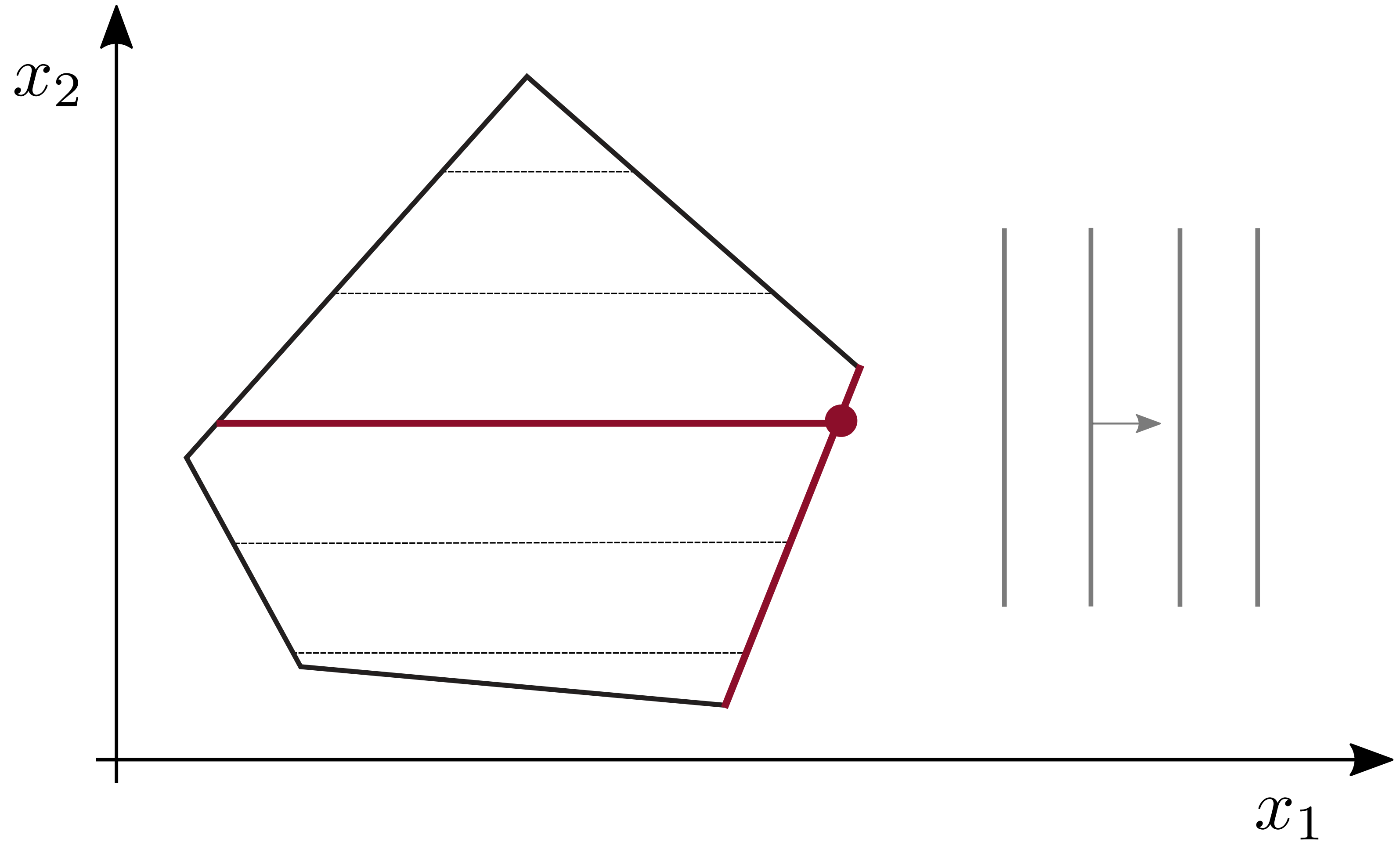
Tight constraints are not enough



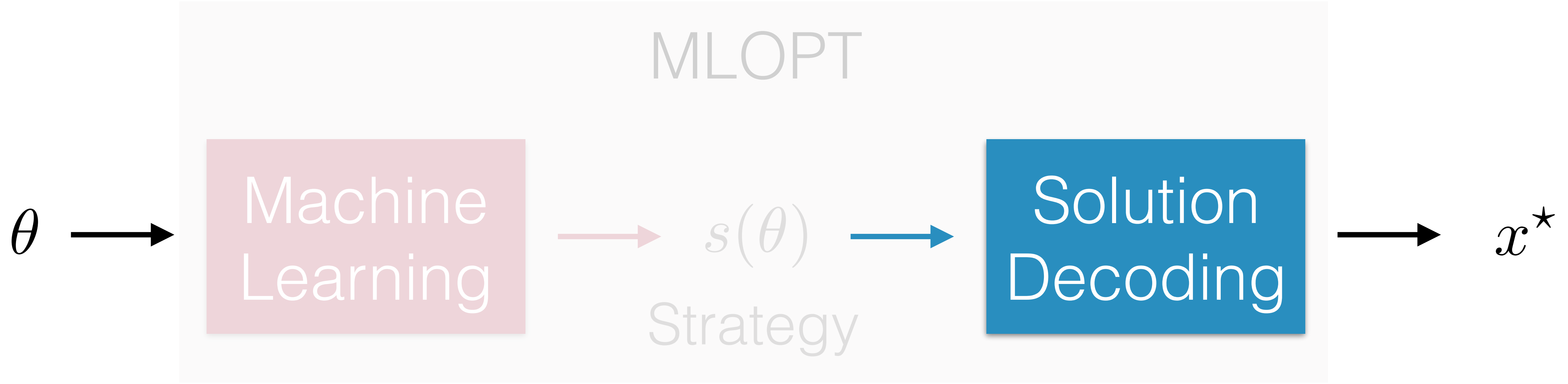
Strategy

$$s(\theta) = (\mathcal{T}(\theta), x^{\star}_{\mathcal{I}}(\theta))$$

Integer variables



Computing the solution from the optimal strategy



Computing the solution from the optimal strategy

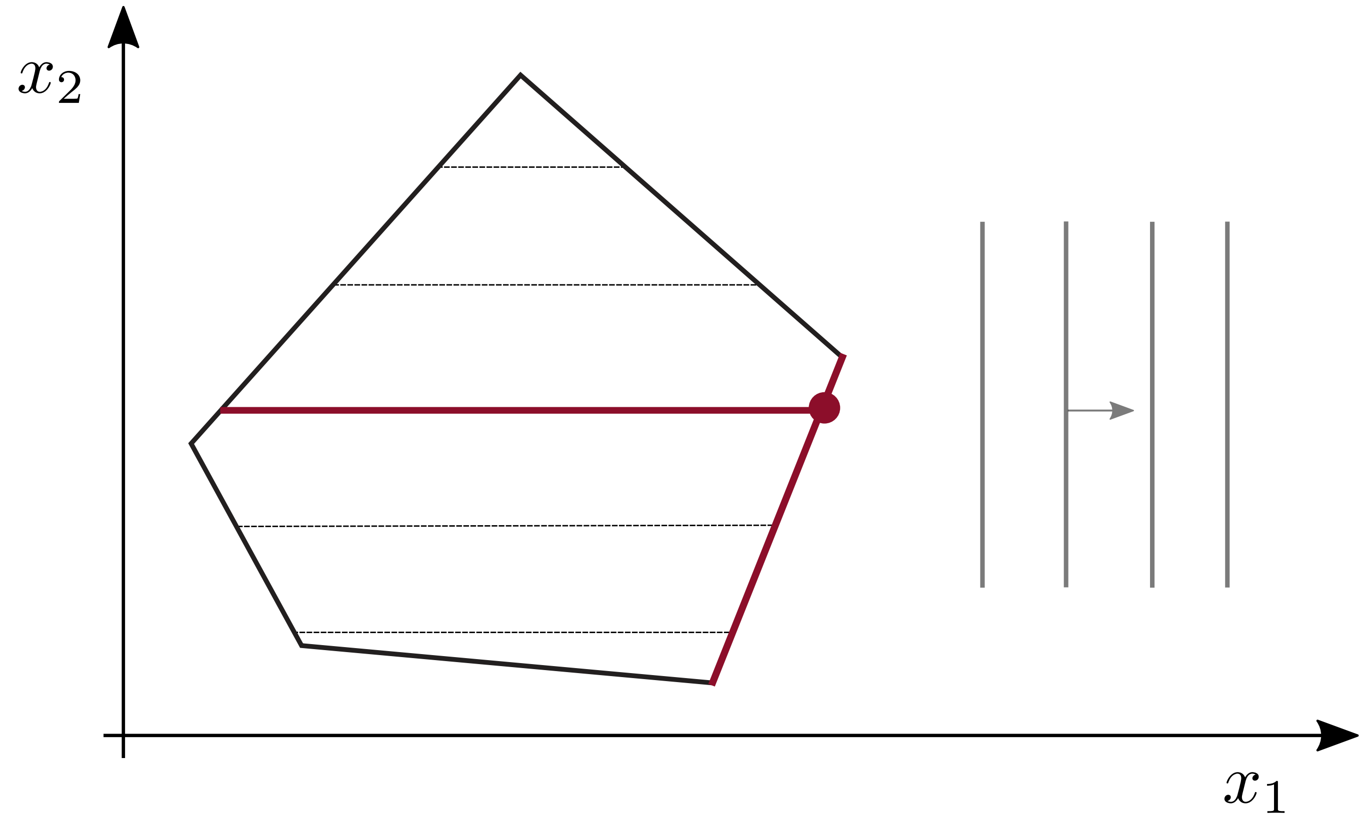
$$\begin{array}{ll} \text{minimize} & c(\theta)^T x \\ \text{subject to} & A(\theta)x \leq b(\theta) \\ & x_{\mathcal{I}} \in \mathbf{Z}^d \end{array}$$

Computing the solution from the optimal strategy

minimize	$c(\theta)^T x$
subject to	$A(\theta)x \leq b(\theta)$
	$x_{\mathcal{I}} \in \mathbf{Z}^d$

Strategy

$$s(\theta) = (\mathcal{T}(\theta), x^\star_{\mathcal{I}}(\theta))$$

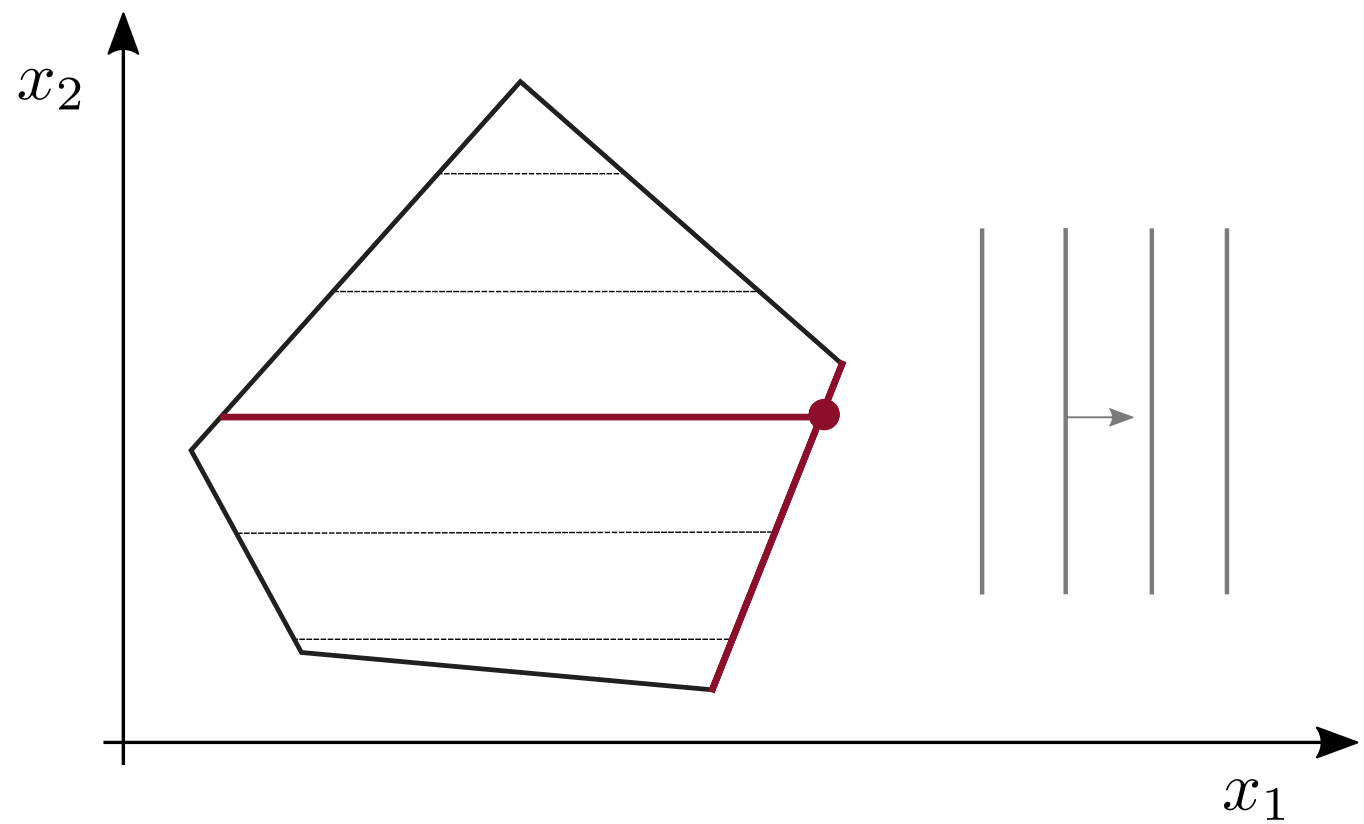


Computing the solution from the optimal strategy

$$
\begin{array}{ll}
\text{minimize} & c(\theta)^T x \\
\text{subject to} & A(\theta)x \leq b(\theta) \\
& x_{\mathcal{I}} \in \mathbf{Z}^d
\end{array}
$$

Strategy

$$s(\theta) = (\mathcal{T}(\theta), x^\star_{\mathcal{I}}(\theta))$$



$$
\begin{array}{lll}
\text{minimize} & c(\theta)^T x & \\
\text{subject to} & A_i(\theta)x = b_i(\theta) & \forall i \in \mathcal{T}(\theta) \\
& x_{\mathcal{I}} = x^\star_{\mathcal{I}}(\theta) &
\end{array}
$$

Integers

Knapsack

maximize $c^T x$

subject to $a^T x \leq b$

$0 \leq x \leq u$

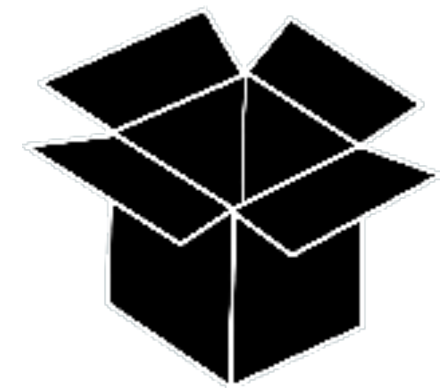
$x \in \mathbf{Z}^n$

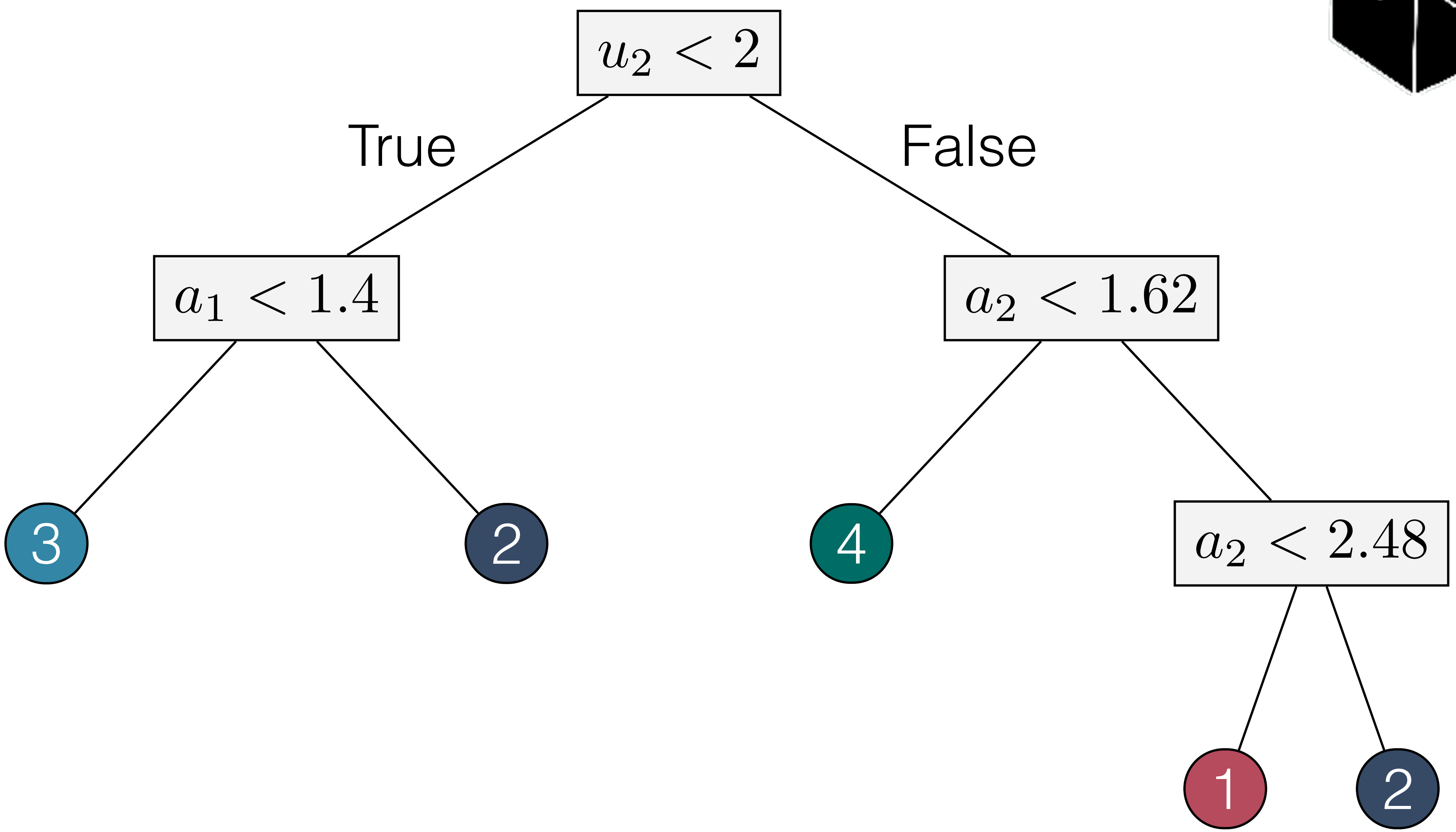
Knapsack

$$
\begin{array}{ll}
\text{maximize} & c^T x \\
\text{subject to} & a^T x \leq b \\
 & 0 \leq x \leq u \\
 & x \in \mathbf{Z}^n
\end{array}
$$

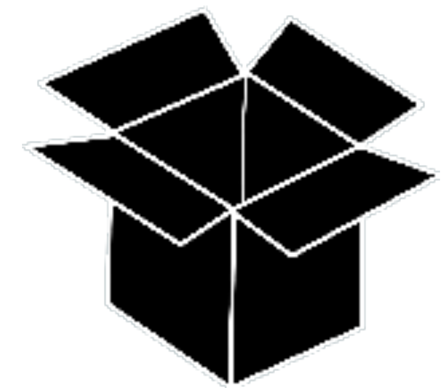
Parameters

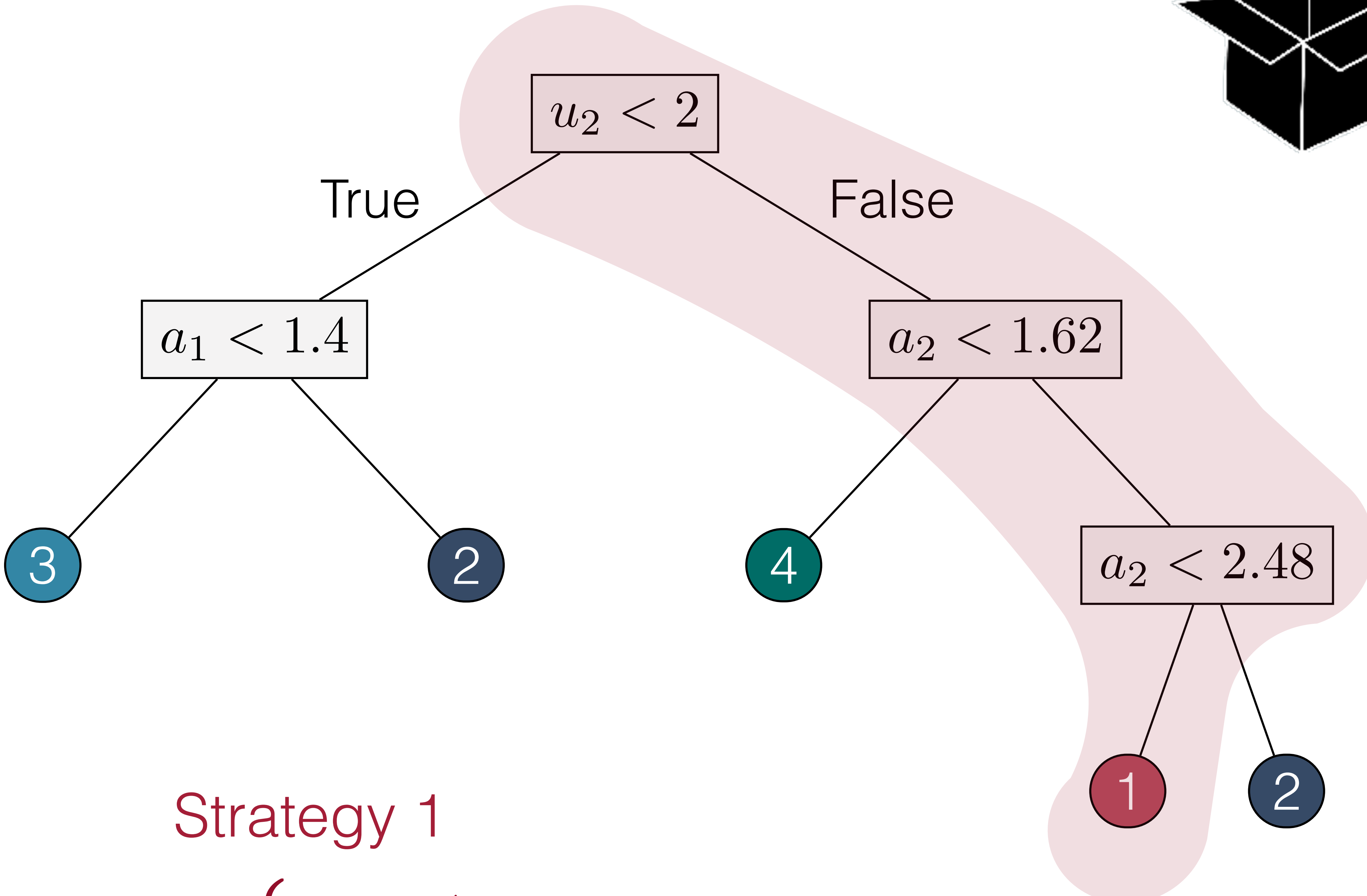
Strategy selection





Strategy selection



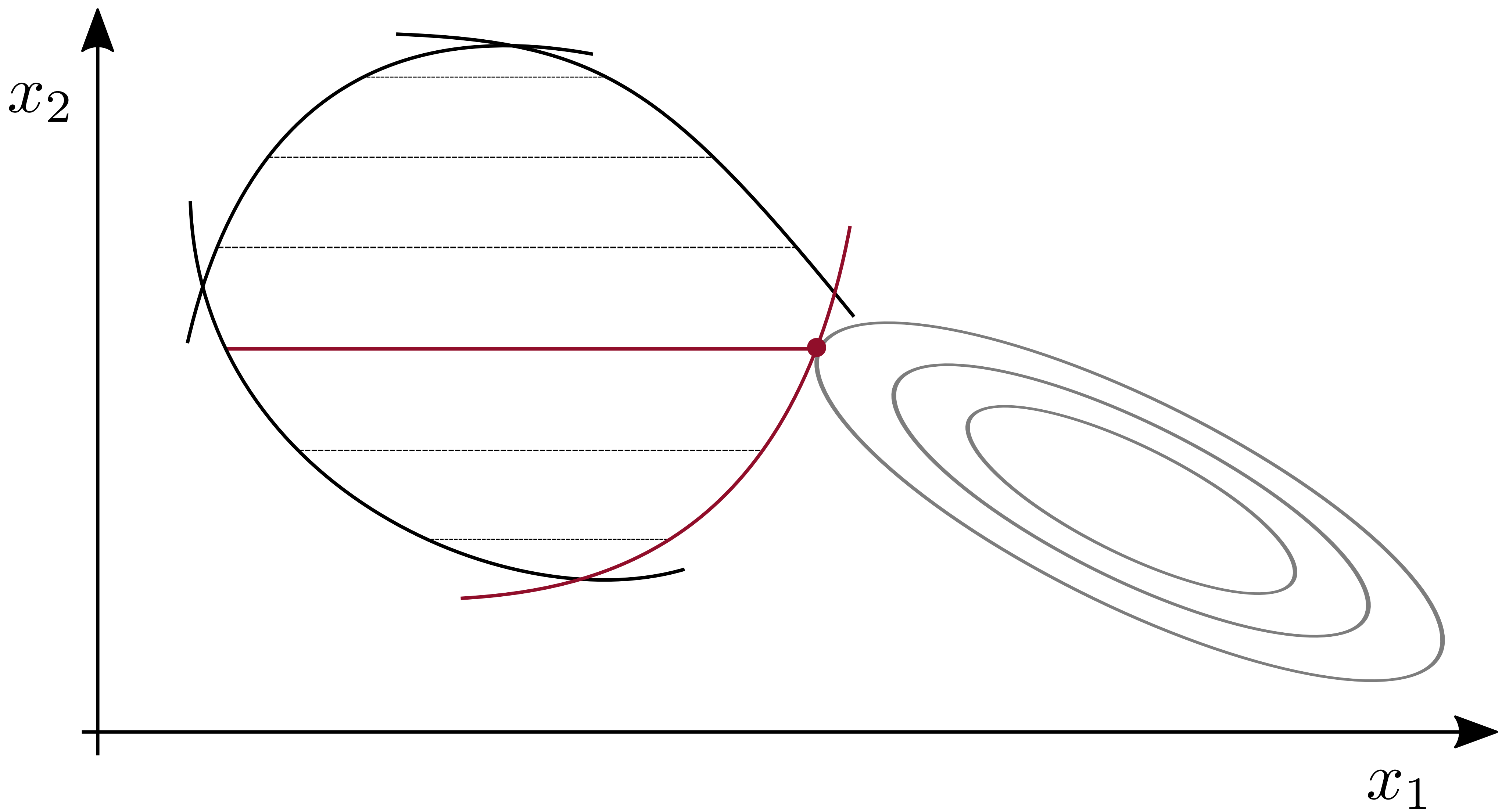


Strategy 1

$$x_i = \begin{cases} 0 & i \neq 2 \\ 2 & i = 2 \end{cases}$$

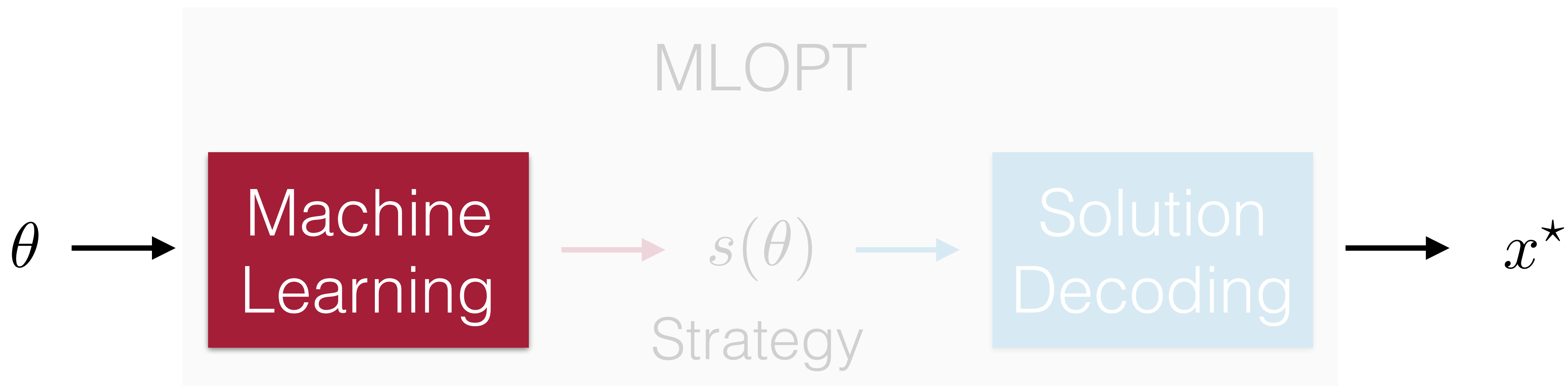
Mixed-integer convex optimization

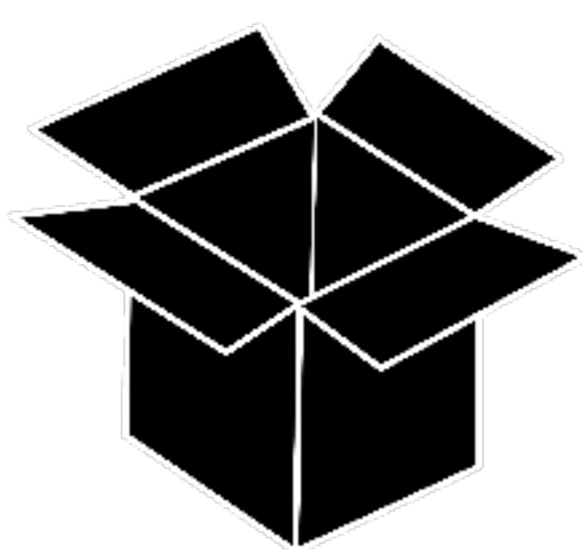
$$\begin{array}{ll} \text{minimize} & f(\theta, x) \\ \text{subject to} & g(\theta, x) \leq 0 \\ & x_{\mathcal{I}} \in \mathbf{Z}^d \end{array}$$



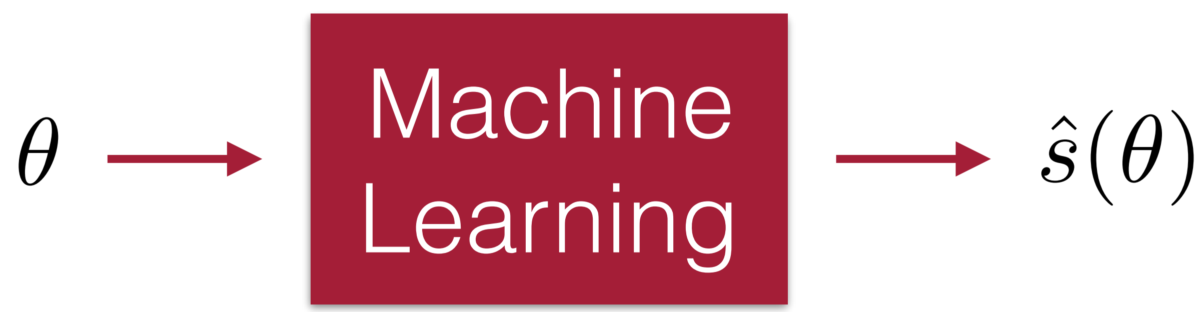
Learning the strategies

Machine Learning Optimizer

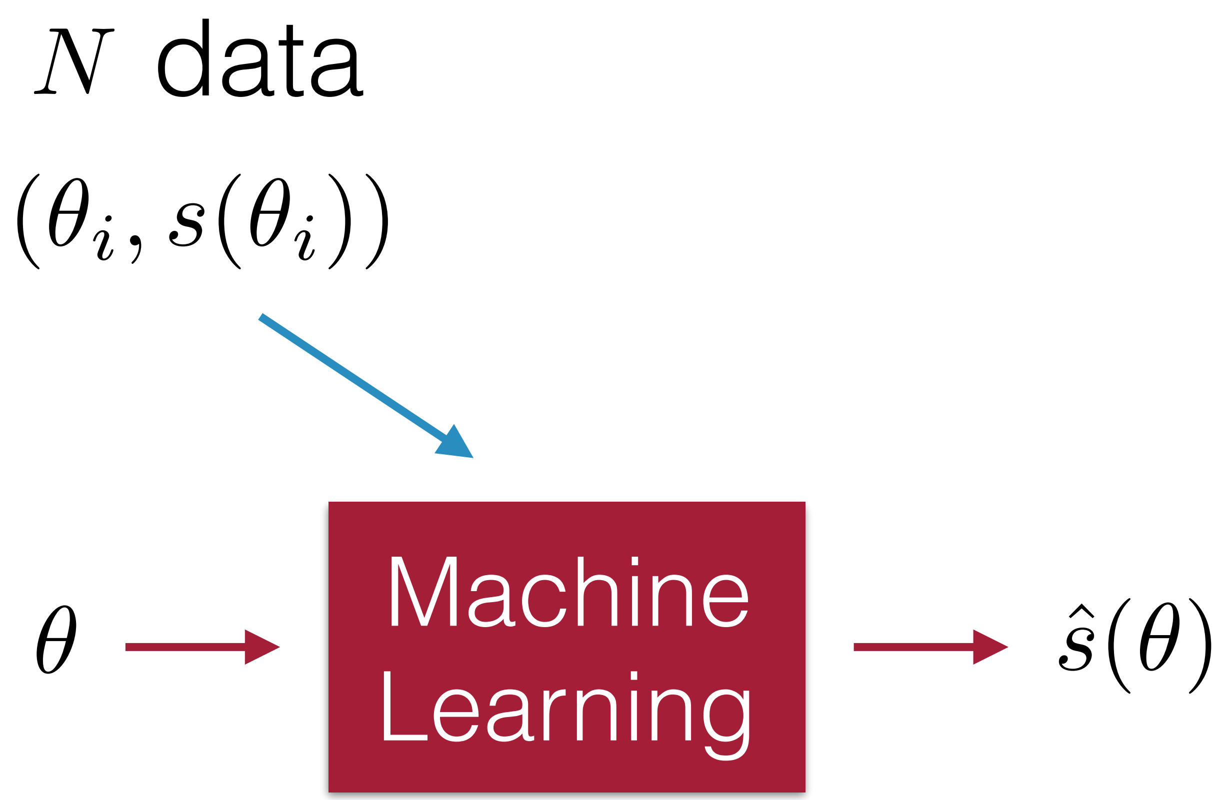




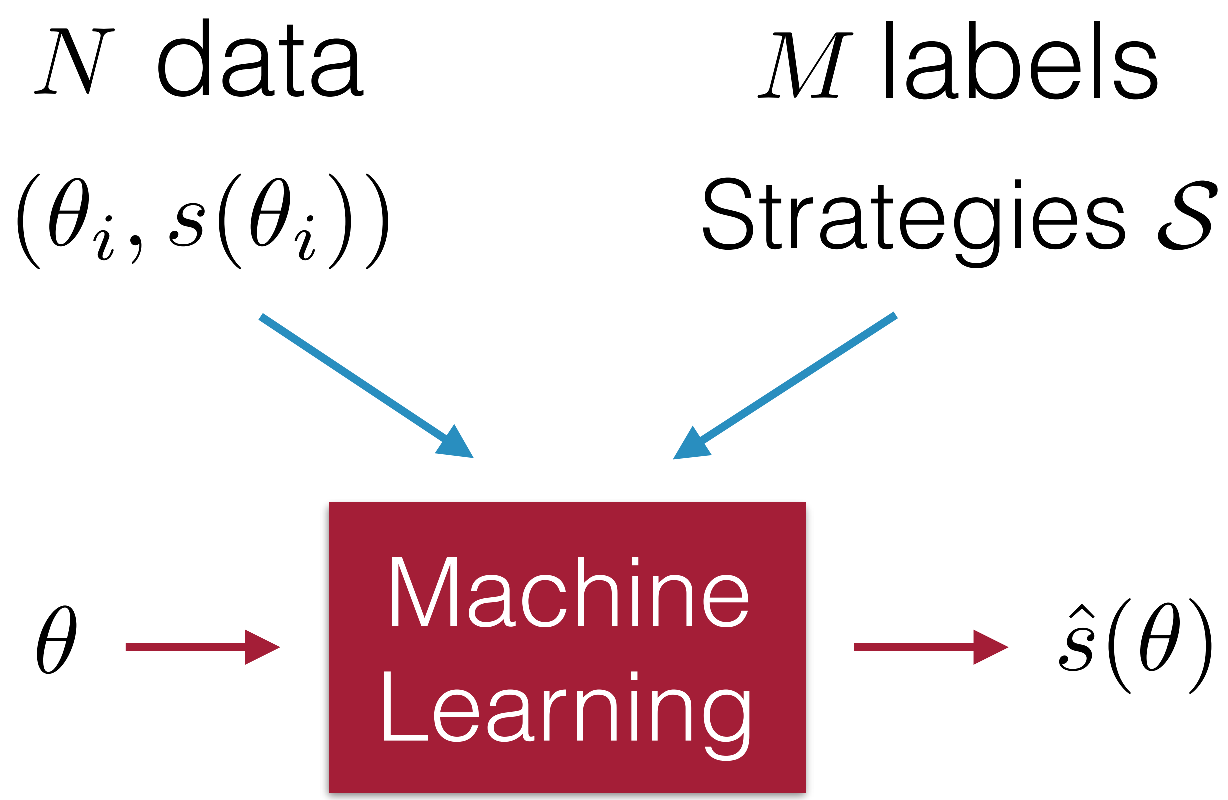
Selecting strategies is a classification problem



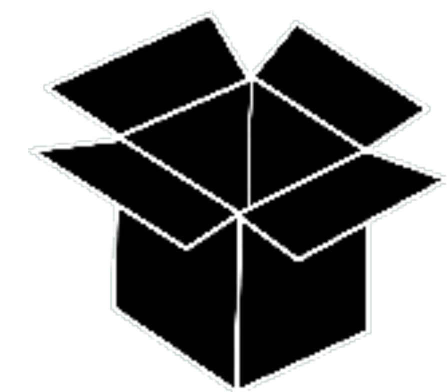
Selecting strategies is a classification problem

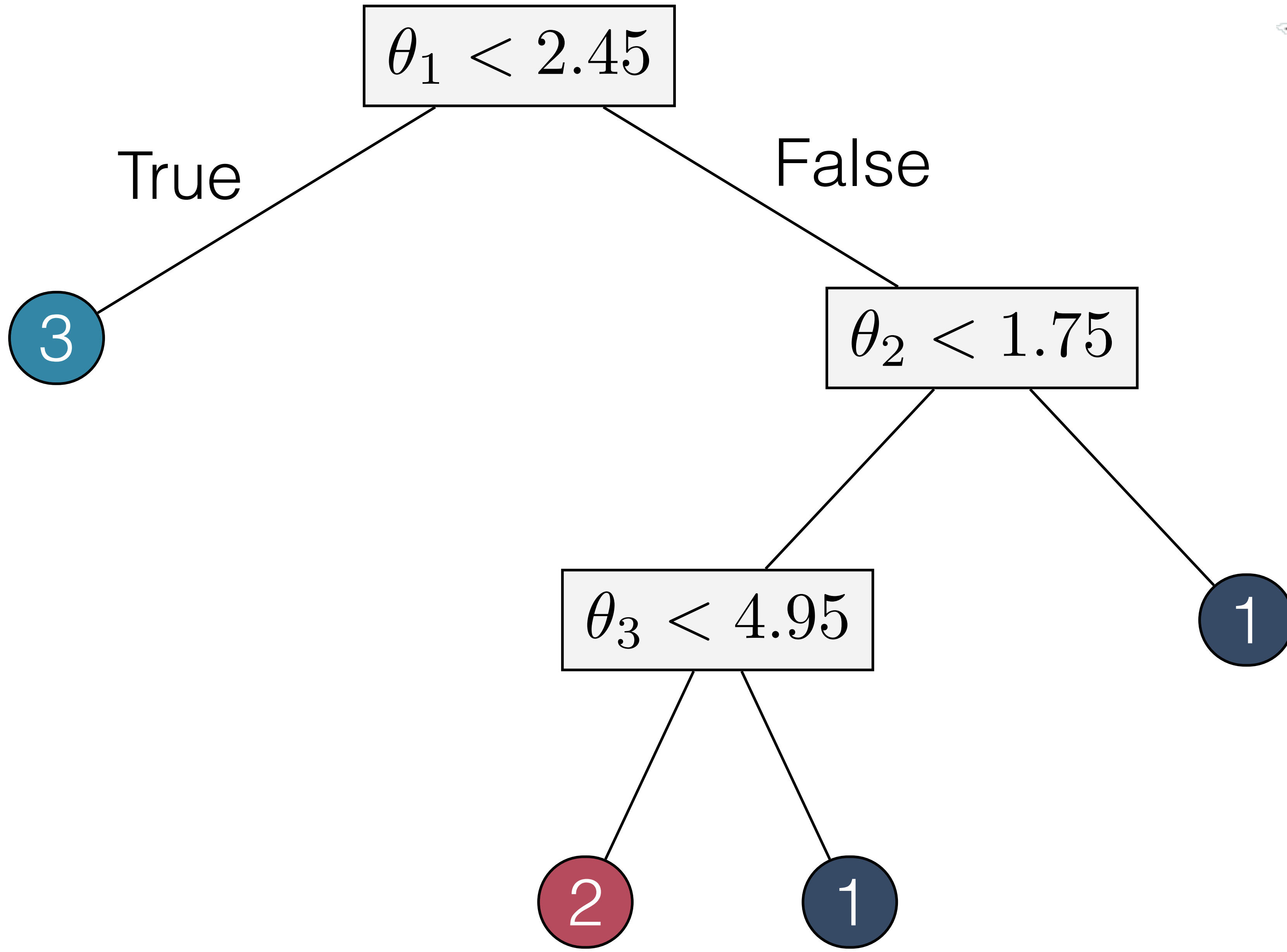


Selecting strategies is a classification problem



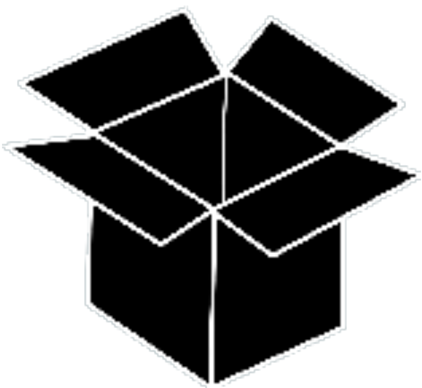
Optimal Classification Trees (OCT)

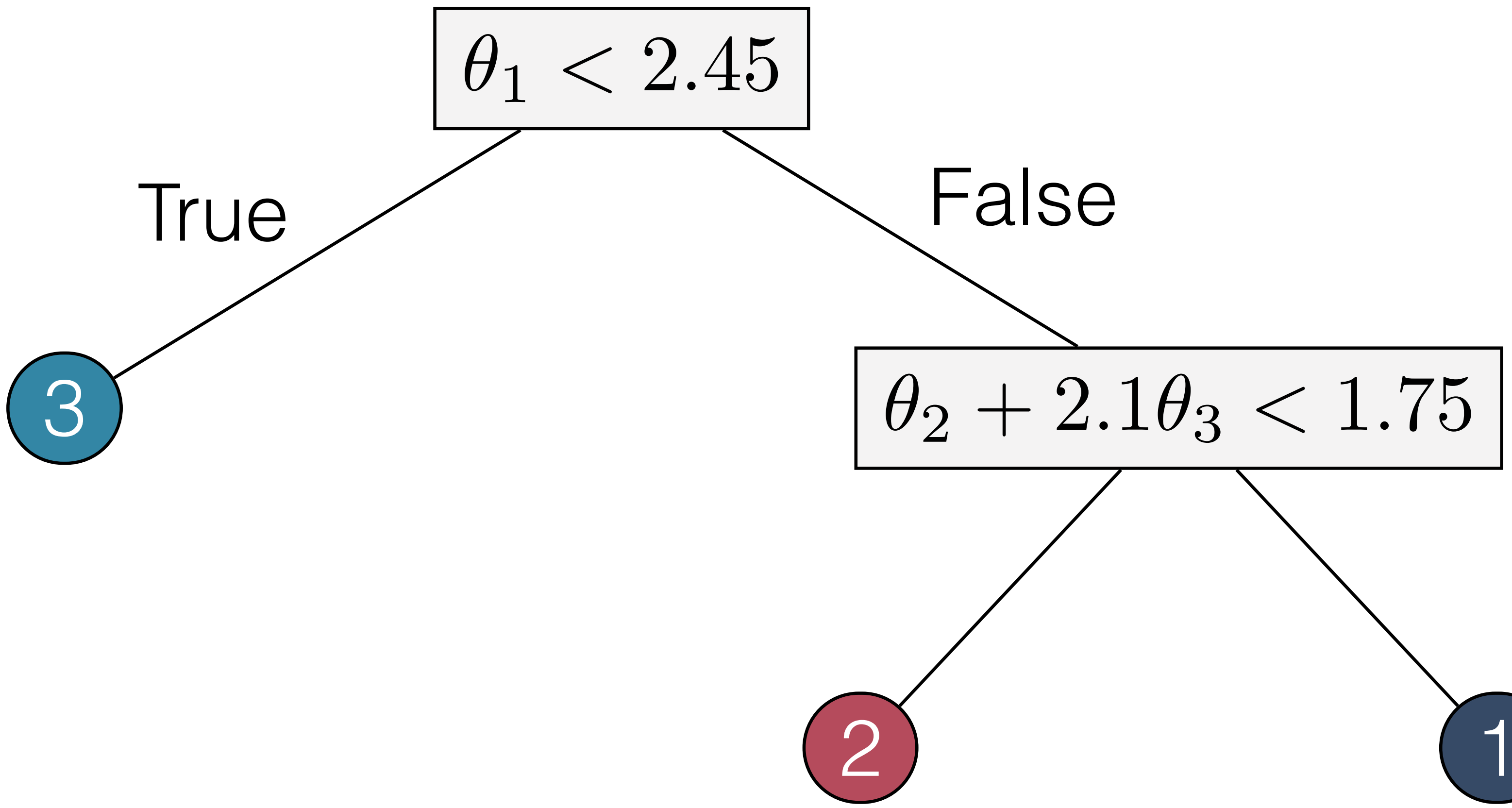




[Bertsimas and Dunn (2017)]

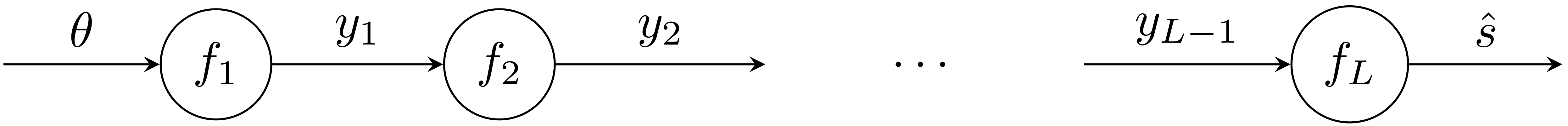
Optimal Classification Trees with Hyperplanes (OCT-H)





[Bertsimas and Dunn (2017)]

Neural Networks (NN)



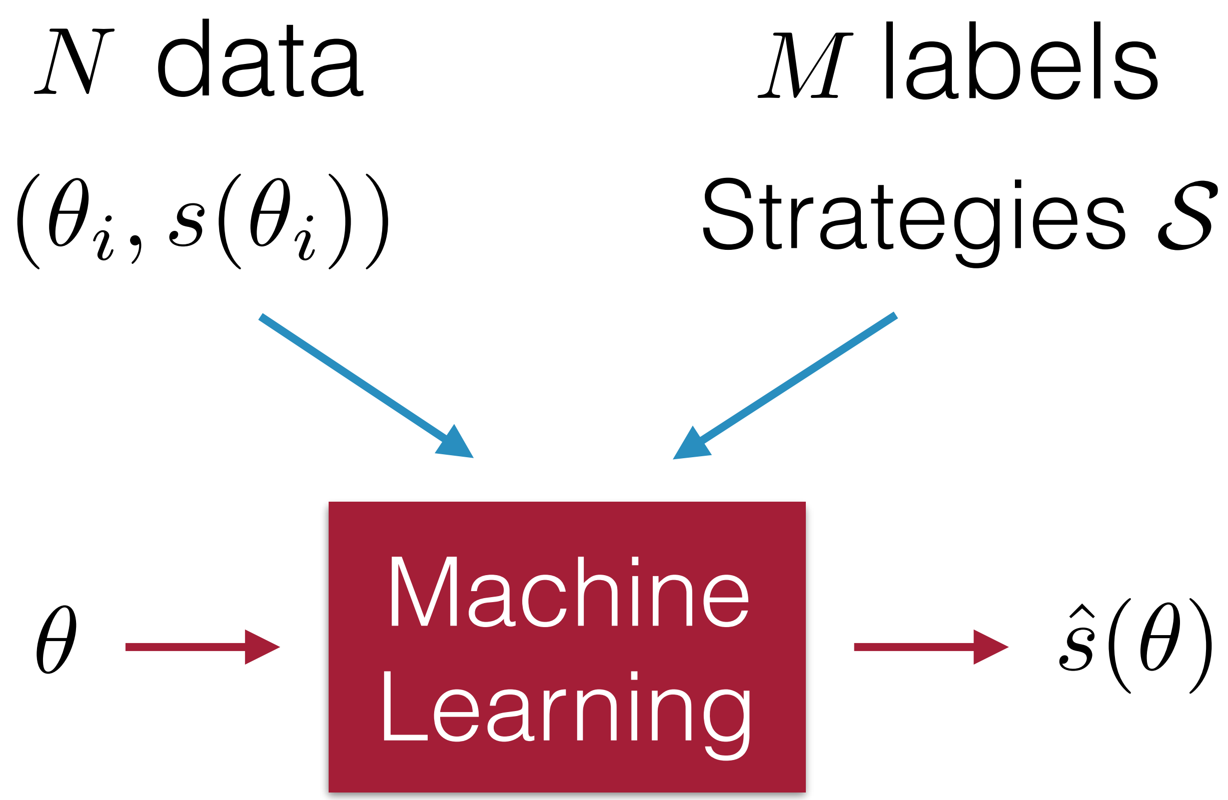
Single layer

$$y_l = f(y_{l-1}) = (W_l y_{l-1} + b_l)_+$$

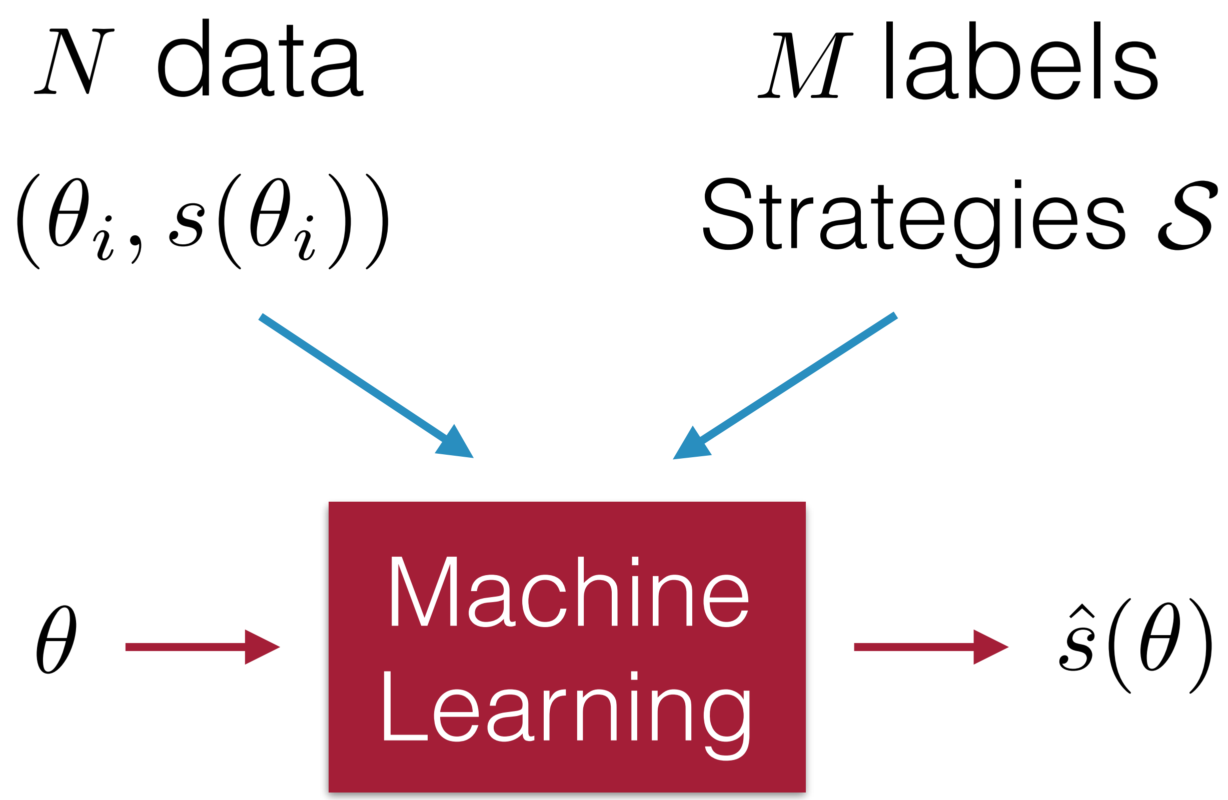
PyTorch

Strategies exploration

Classification problem

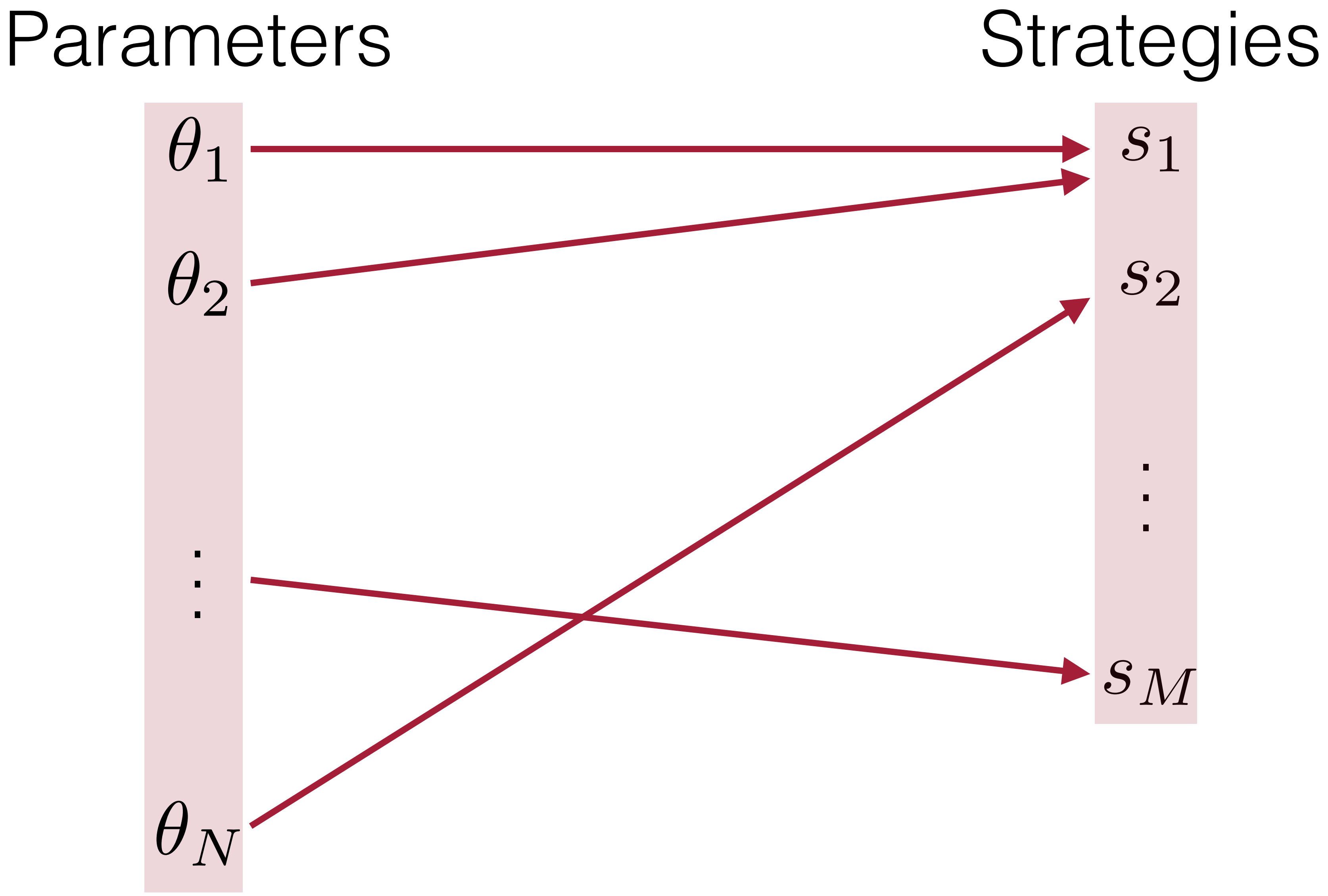


Classification problem

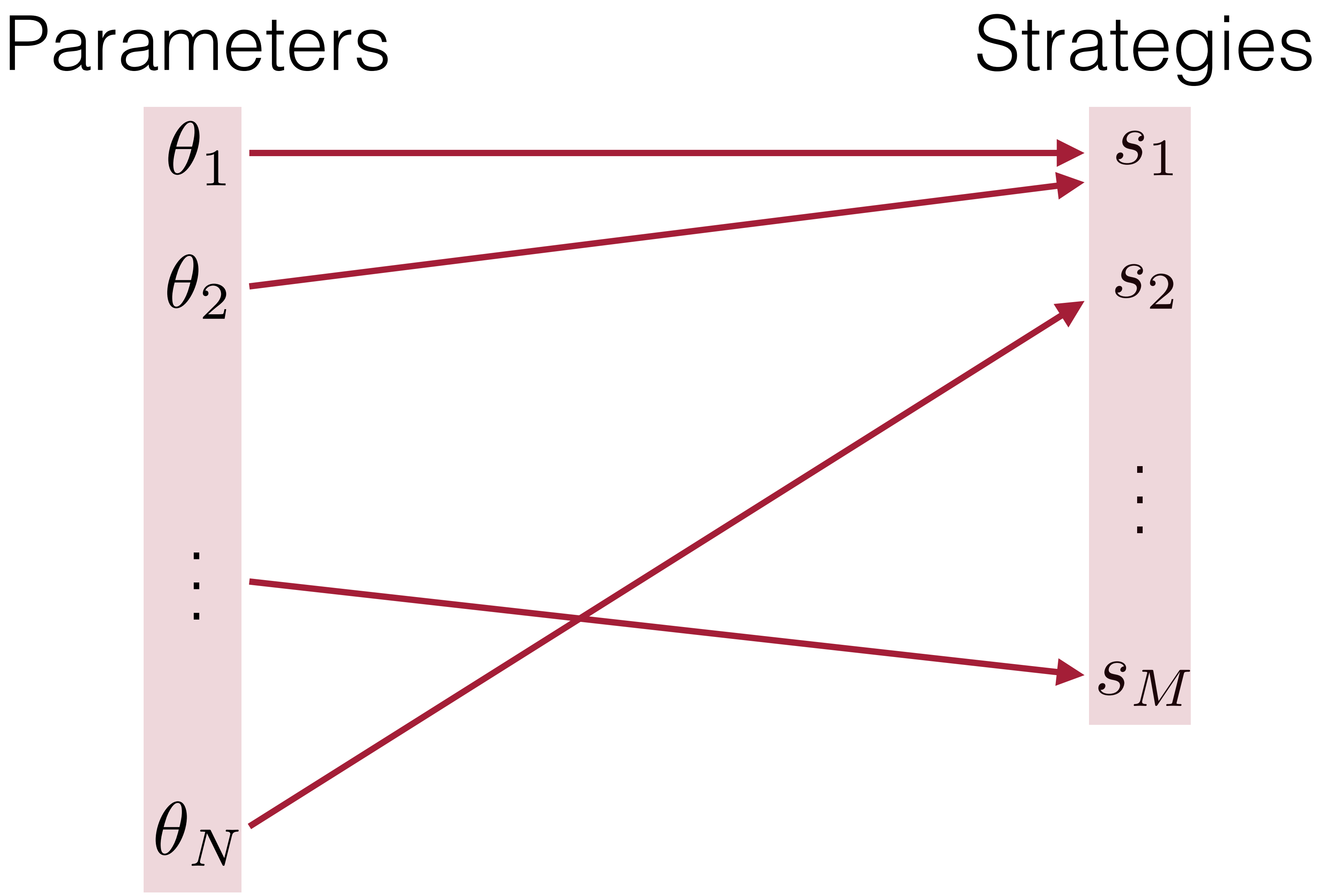


Have we seen enough data?

Will we find new strategies?



Will we find new strategies?



θ_{N+1}?

Alan Turing already knew that

Decoded the **Enigma Machine**
in
World War II

Only some words (labels) needed

Good-Turing estimator

s_1	12 times
s_2	45 times
$\vdots$	$\vdots$
s_M	2 times

Good-Turing estimator

s_1	12 times
s_2	45 times
$\vdots$	$\vdots$
s_M	2 times

Probability of unseen strategies

$$GT = \frac{N_1}{N} \approx \mathbf{P}(\theta_{N+1} \in \mathbf{R}^p \mid s(\theta_{N+1}) \notin \mathcal{S}(\Theta_N))$$

Good-Turing estimator

s_1	12 times
s_2	45 times
$\vdots$	$\vdots$
s_M	2 times

strategies appeared once

Probability of unseen strategies

$$GT = \frac{N_1}{N} \approx \mathbf{P}(\theta_{N+1} \in \mathbf{R}^p \mid s(\theta_{N+1}) \notin \mathcal{S}(\Theta_N))$$

Good-Turing estimator

s_1	12 times
s_2	45 times
$\vdots$	$\vdots$
s_M	2 times

\# strategies appeared once

Probability of unseen strategies

$$GT = \frac{N_1}{N} \approx \mathbf{P}(\theta_{N+1} \in \mathbf{R}^p \mid s(\theta_{N+1}) \notin \mathcal{S}(\Theta_N))$$

\# samples

Good-Turing estimator

s_1	12 times
s_2	45 times
$\vdots$	$\vdots$
s_M	2 times

strategies appeared once

Probability of unseen strategies

$$GT = \frac{N_1}{N} \approx \mathbf{P}(\theta_{N+1} \in \mathbf{R}^p \mid s(\theta_{N+1}) \notin \mathcal{S}(\Theta_N))$$

samples

Bound

$$\mathbf{P}(\theta_{N+1} \in \mathbf{R}^p \mid s(\theta_{N+1}) \notin \mathcal{S}(\Theta_N)) \leq GT + c\sqrt{(1/N)\ln(3/\beta)}$$

Simple sampling scheme

Bound

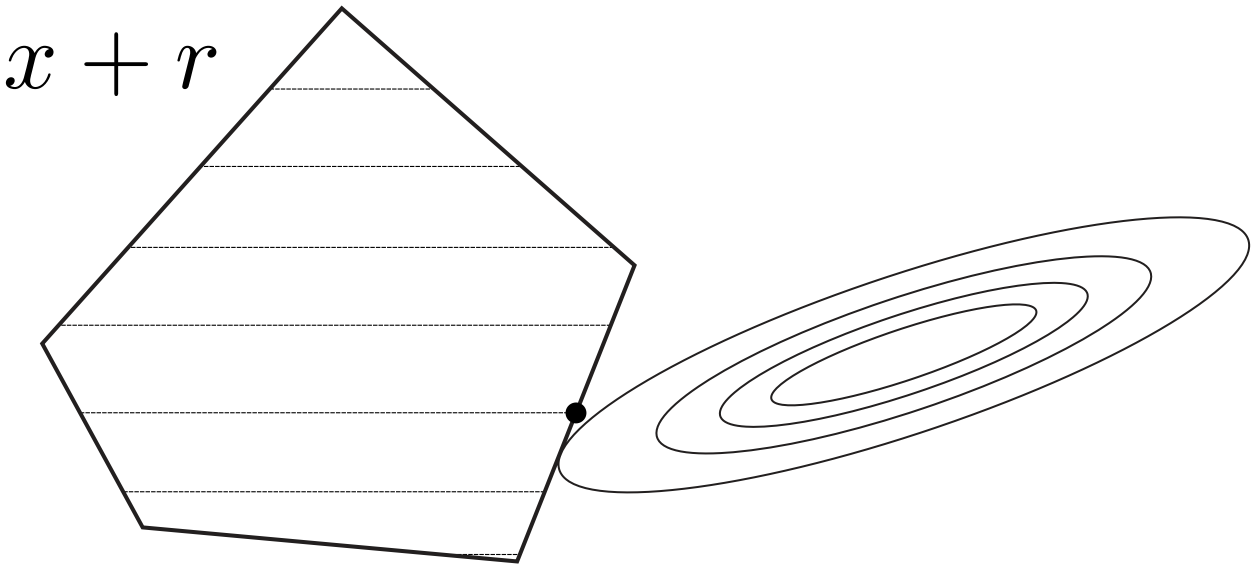
Repeat until $GT + c\sqrt{(1/N)\ln(3/\beta)} \leq \epsilon$

1. sample θ_i
2. compute $s(\theta_i)$
3. update estimator $GT = \dfrac{N_1}{N}$

Speedups

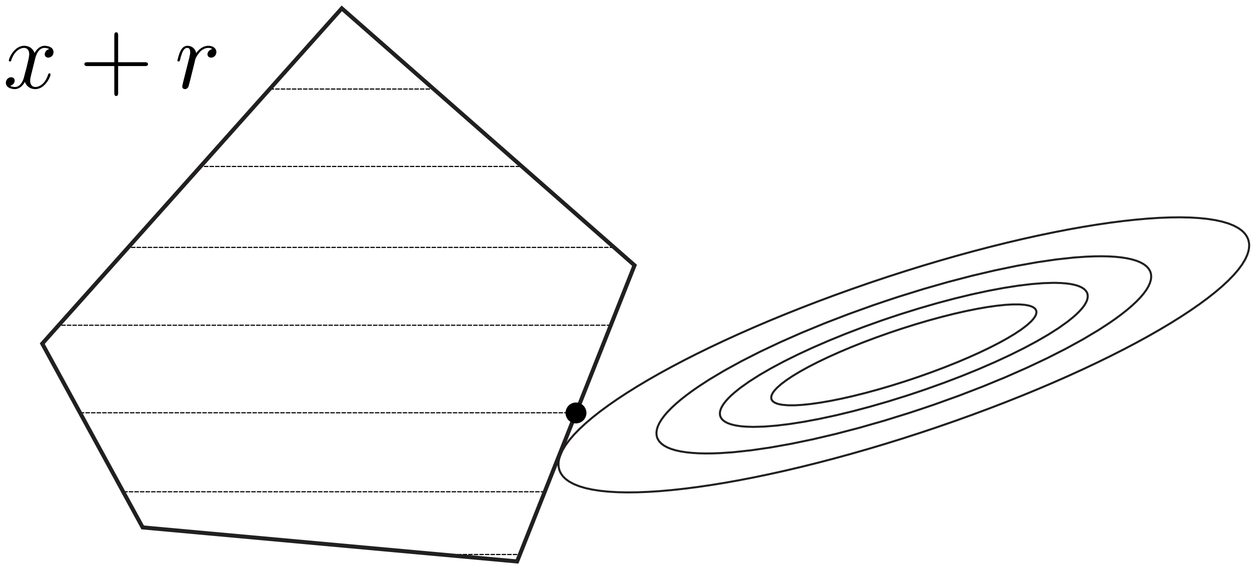
Mixed-integer quadratic optimization

minmimize	$(1/2)x^T P x + q^T x + r$
subject to	$Ax \leq b$
	$x_{\mathcal{I}} \in \mathbf{Z}^d$



Mixed-integer quadratic optimization

$$\begin{array}{ll} \text{minmimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Ax \leq b \\ & x_{\mathcal{I}} \in \mathbf{Z}^d \end{array}$$



Strategy

$$\begin{array}{ll} \text{minmimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & A_{\mathcal{T}(\theta)} x = b_{\mathcal{T}(\theta)} \\ & x_{\mathcal{I}} = x^\star_{\mathcal{I}}(\theta) \end{array}$$

Equality Constrained QP

Quick solution

KKT System

$$\begin{bmatrix} P & A_{\mathcal{T}(\theta)}^T & I_{\mathcal{I}}^T \\ A_{\mathcal{T}(\theta)} & & \\ I_{\mathcal{I}} & & 0 \end{bmatrix} \begin{bmatrix} x \\ \nu \end{bmatrix} = \begin{bmatrix} -q \\ b_{\mathcal{T}(\theta)} \\ x_{\mathcal{I}}^{\star}(\theta) \end{bmatrix}$$

Quick solution

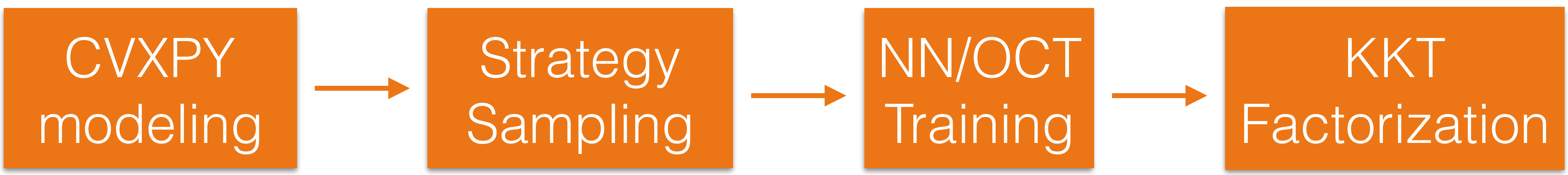
KKT System

$$\begin{bmatrix} P & A_{\mathcal{T}(\theta)}^T & I_{\mathcal{I}}^T \\ A_{\mathcal{T}(\theta)} & & \\ I_{\mathcal{I}} & 0 & \end{bmatrix} \begin{bmatrix} x \\ \nu \end{bmatrix} = \begin{bmatrix} -q \\ b_{\mathcal{T}(\theta)} \\ x_{\mathcal{I}}^\star(\theta) \end{bmatrix}$$

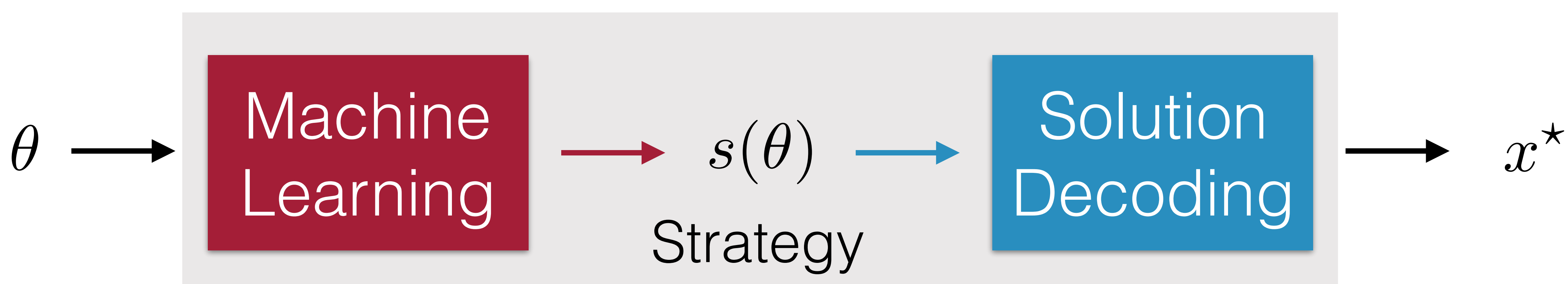
$\mathcal{NP}$-hard problem → Linear System

MLOPT: Machine Learning Optimizer

Offline Learning

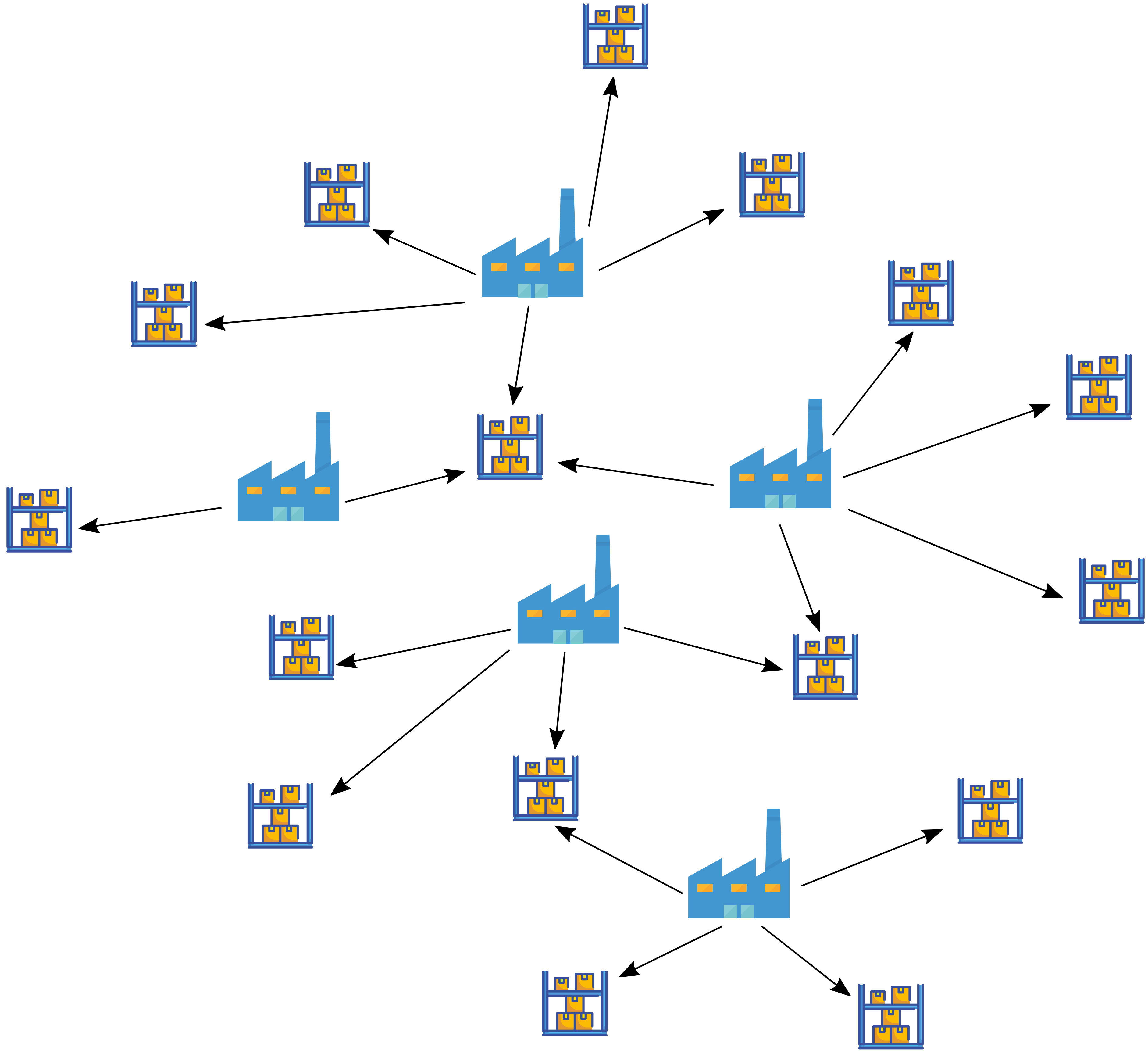


Online Optimization



Benchmarks

Facility location



Facility location

minmimize $\sum_{i \in F} \sum_{j \in W} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i$

subject to $\sum_{i \in F} x_{ij} \geq d_j \quad \forall j \in W$

$$\sum_{j \in W} x_{ij} \leq s_i y_i, \quad \forall i \in F$$

$$x_{ij} \geq 0, y_i \in \{0, 1\}$$

Facility location

Facility cost

minmimize $\sum_{i \in F} \sum_{j \in W} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i$

subject to $\sum_{i \in F} x_{ij} \geq d_j \quad \forall j \in W$

$\sum_{j \in W} x_{ij} \leq s_i y_i, \quad \forall i \in F$

$x_{ij} \geq 0, y_i \in \{0, 1\}$

Facility

Facility location

Shipment cost — Facility cost

minmimize $$\sum_{i \in F} \sum_{j \in W} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i$$

subject to $$\sum_{i \in F} x_{ij} \geq d_j \quad \forall j \in W$$

$$\sum_{j \in W} x_{ij} \leq s_i y_i, \quad \forall i \in F$$

$$x_{ij} \geq 0, y_i \in \{0, 1\}$$

Shipment (x_{ij}) — Facility (y_i)

Facility location

Shipment cost

Facility cost

minmimize $$\sum_{i \in F} \sum_{j \in W} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i$$

subject to $$\sum_{i \in F} x_{ij} \geq d_j \quad \forall j \in W$$

Demand

$$\sum_{j \in W} x_{ij} \leq s_i y_i, \quad \forall i \in F$$

$$x_{ij} \geq 0, y_i \in \{0, 1\}$$

Shipment

Facility

Facility location

Shipment cost — Facility cost

minmimize $$\sum_{i \in F} \sum_{j \in W} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i$$

subject to

$$\sum_{i \in F} x_{ij} \geq d_j \quad \forall j \in W$$ Demand

$$\sum_{j \in W} x_{ij} \leq s_i y_i, \quad \forall i \in F$$ Supply

$$x_{ij} \geq 0, y_i \in \{0, 1\}$$

Shipment (x_{ij}) — Facility (y_i)

Facility location

Shipment cost | Facility cost

minmimize $\sum_{i \in F} \sum_{j \in W} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i$

subject to $\sum_{i \in F} x_{ij} \geq d_j \quad \forall j \in W$ Demand

$\sum_{j \in W} x_{ij} \leq s_i y_i, \quad \forall i \in F$ Supply

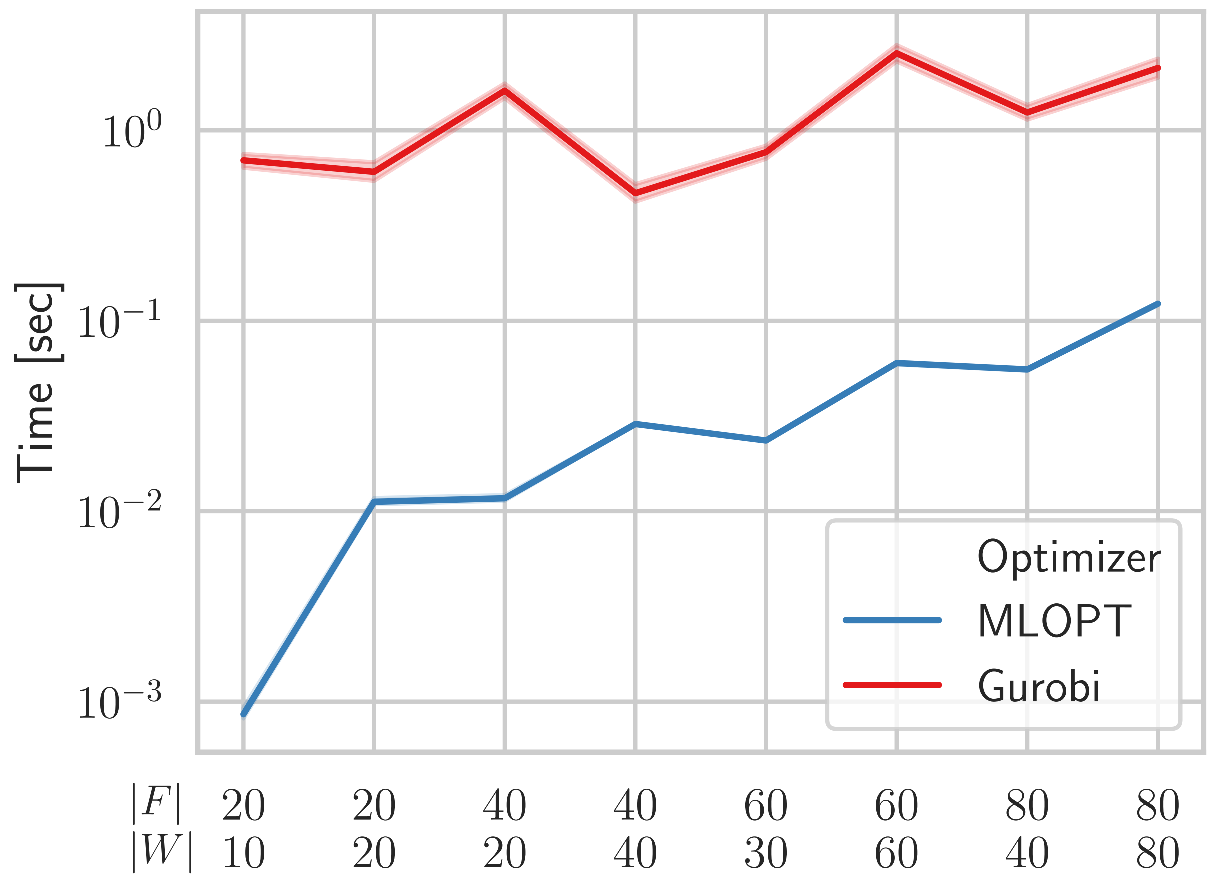
$x_{ij} \geq 0, y_i \in \{0, 1\}$

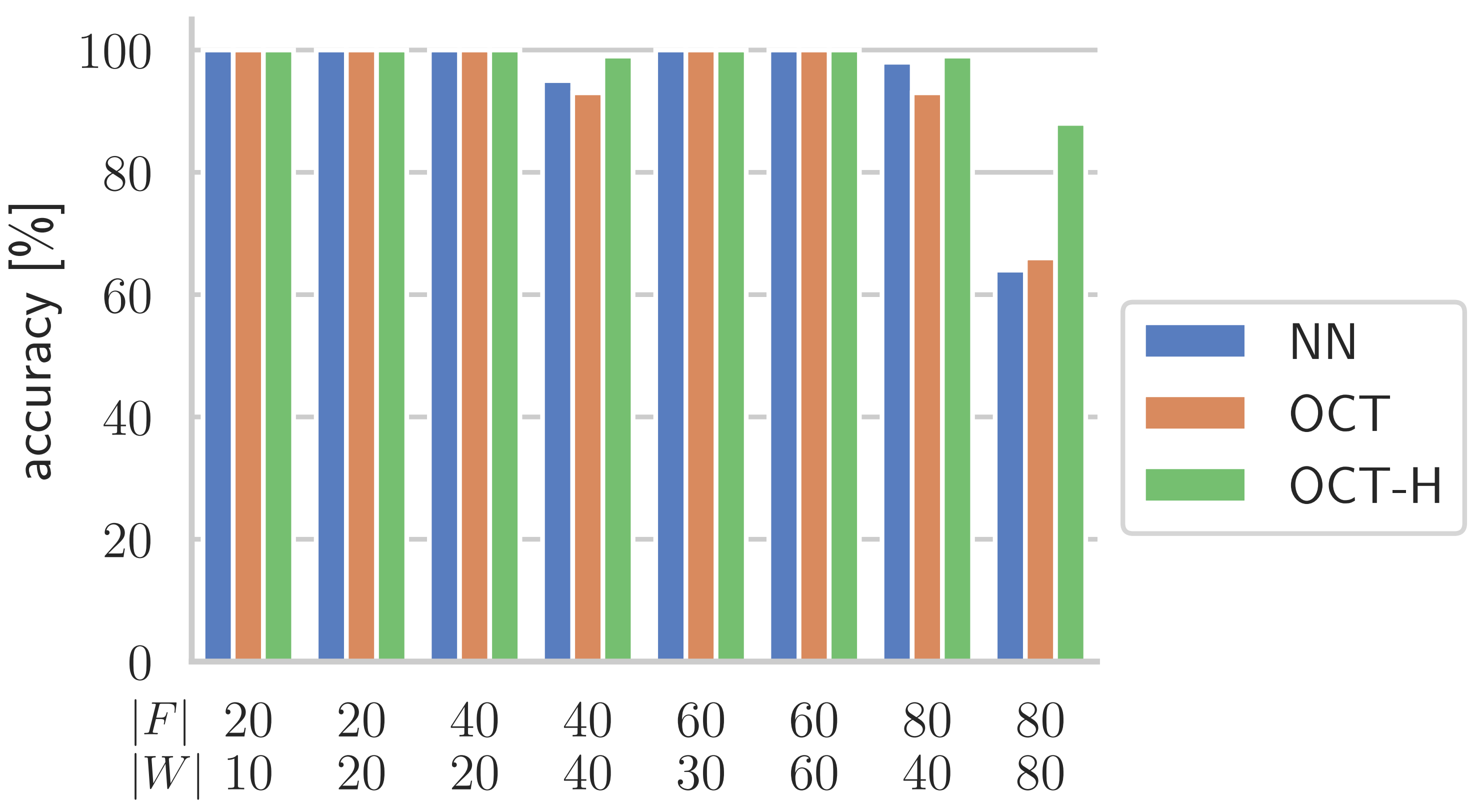
Parameters

Shipment

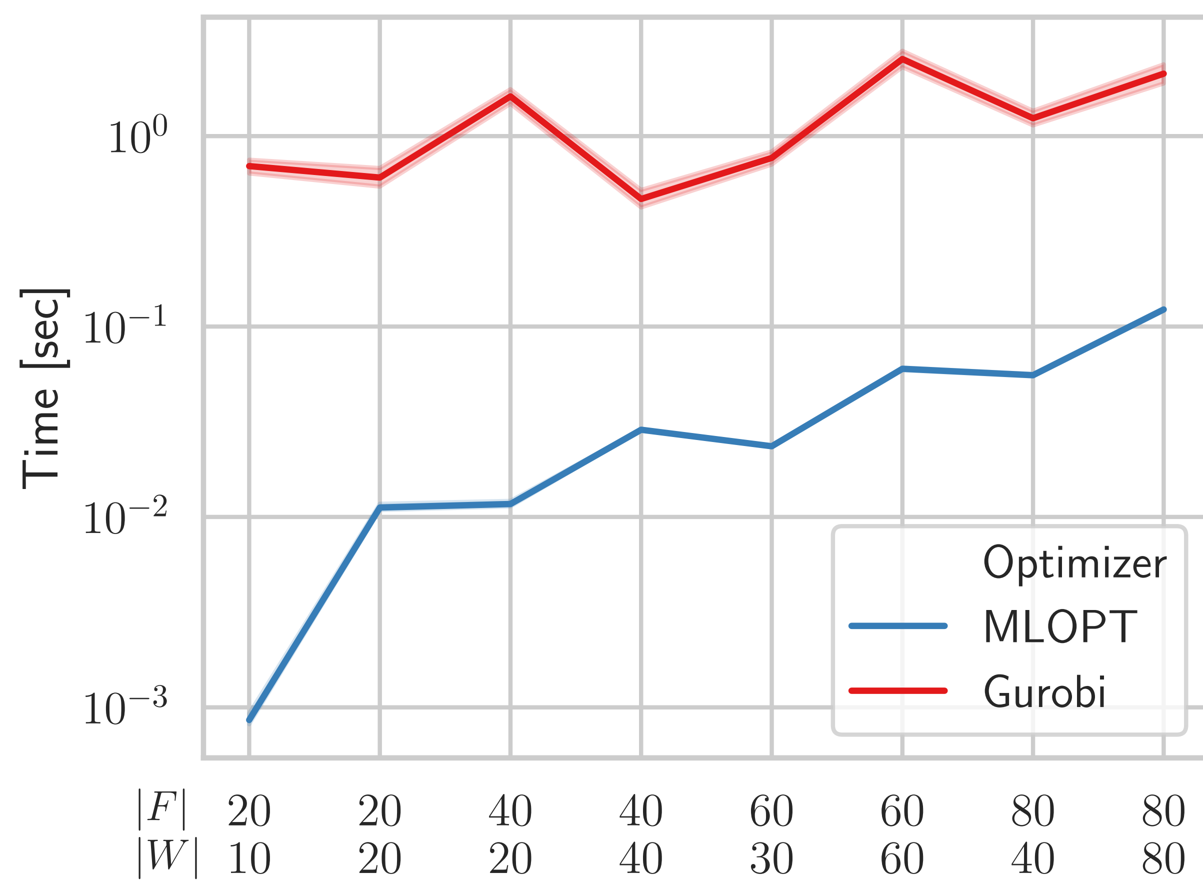
Facility

Facility location results: high speed and accuracy

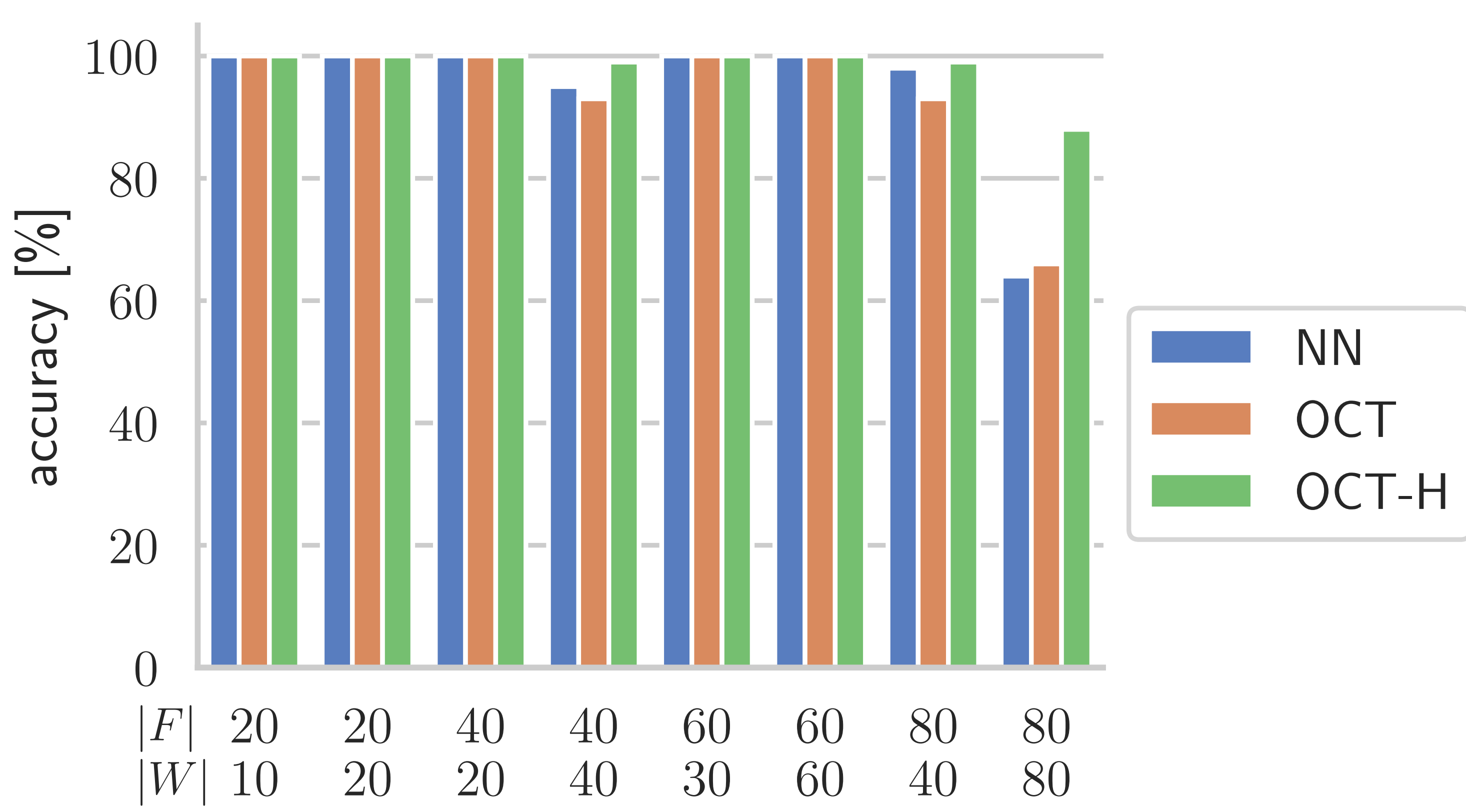




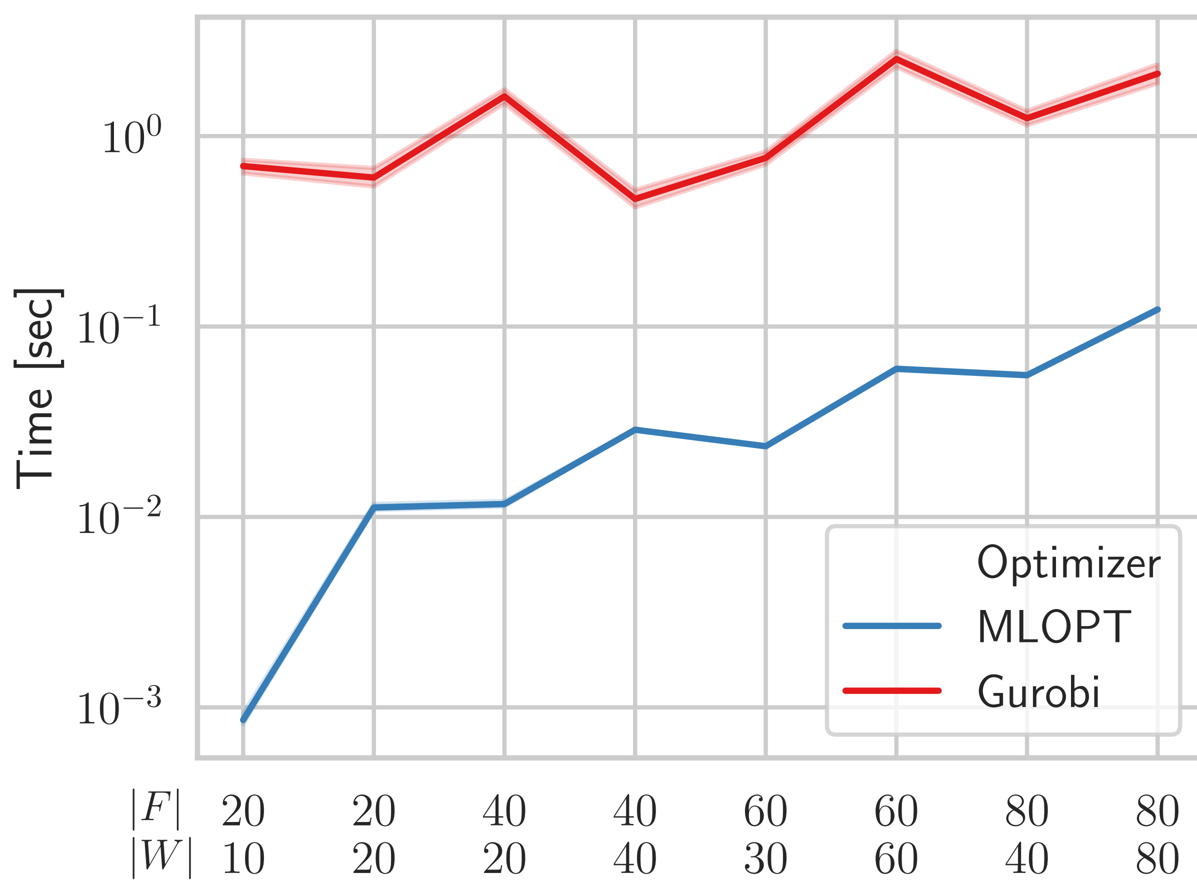
Facility location results: high speed and accuracy



10-100x faster

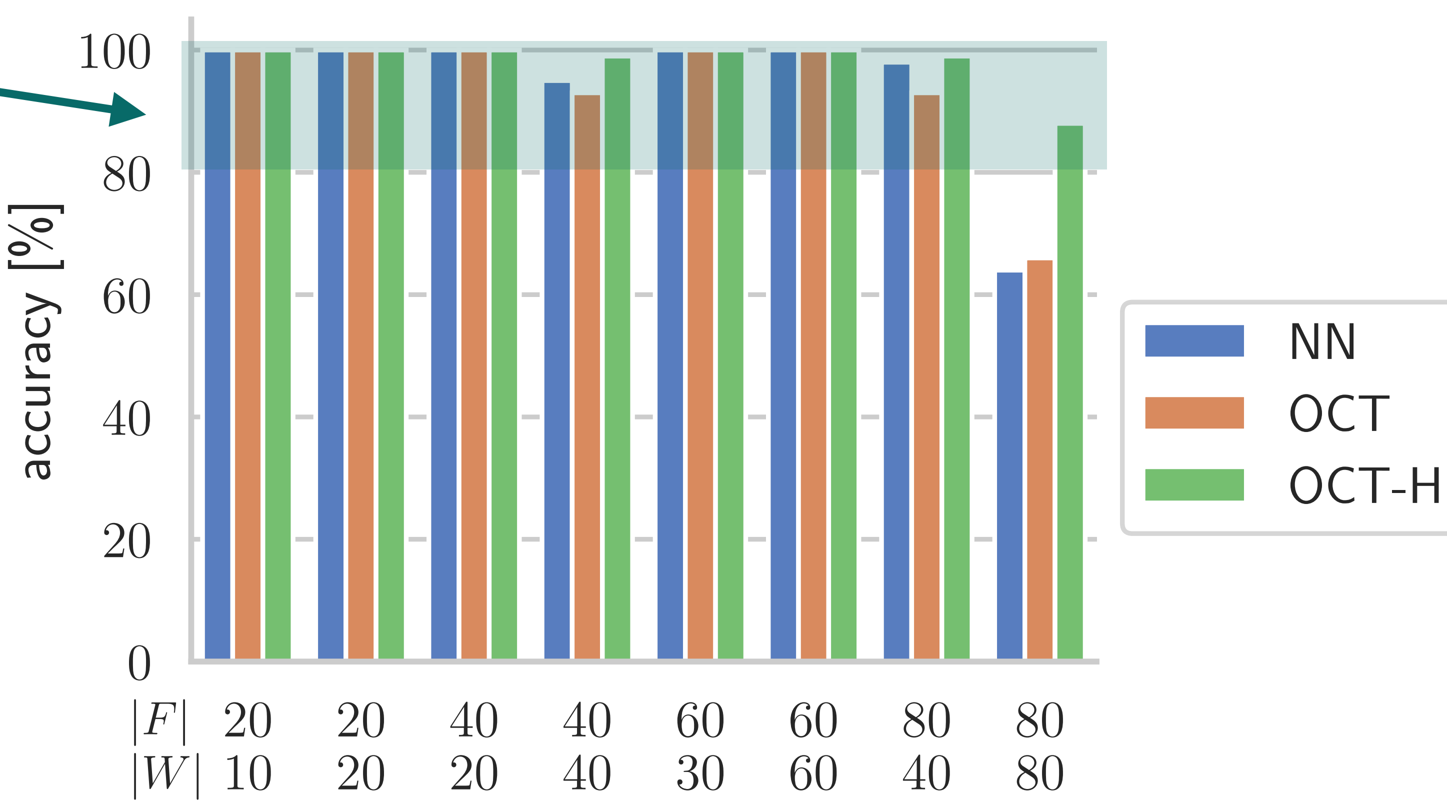


Facility location results: high speed and accuracy



10-100x faster

High accuracy



Sparse portfolio optimization

$$
\begin{array}{ll}
\text{maximize} & r^T w - \gamma w^T \Sigma w \\
\text{subject to} & \mathbf{1}^T w = 1 \\
& \mathbf{card}(w) \leq c
\end{array}
$$

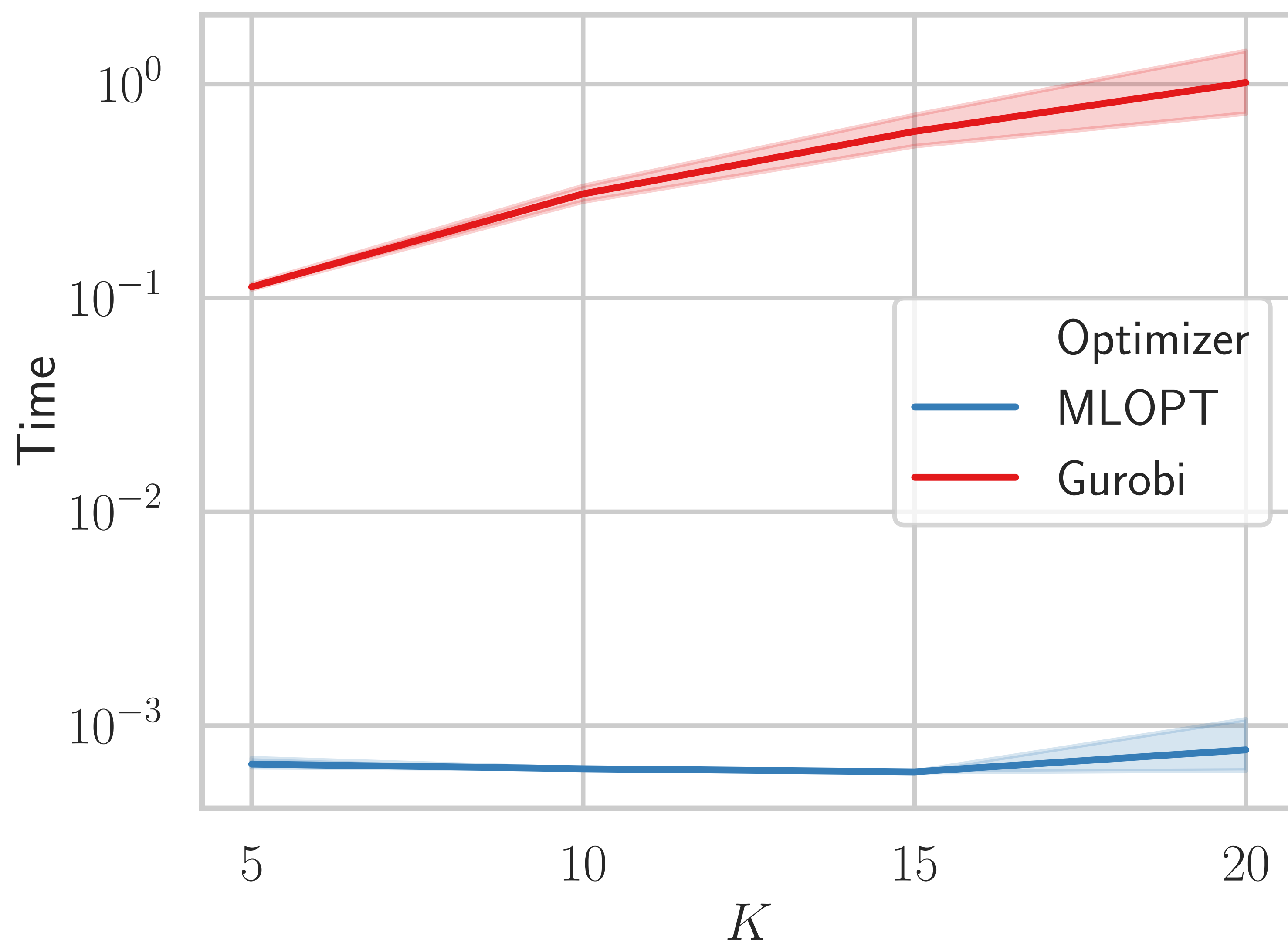
k-factor Risk Model

$$\Sigma = F \Sigma^k F^T + D$$

Parameters

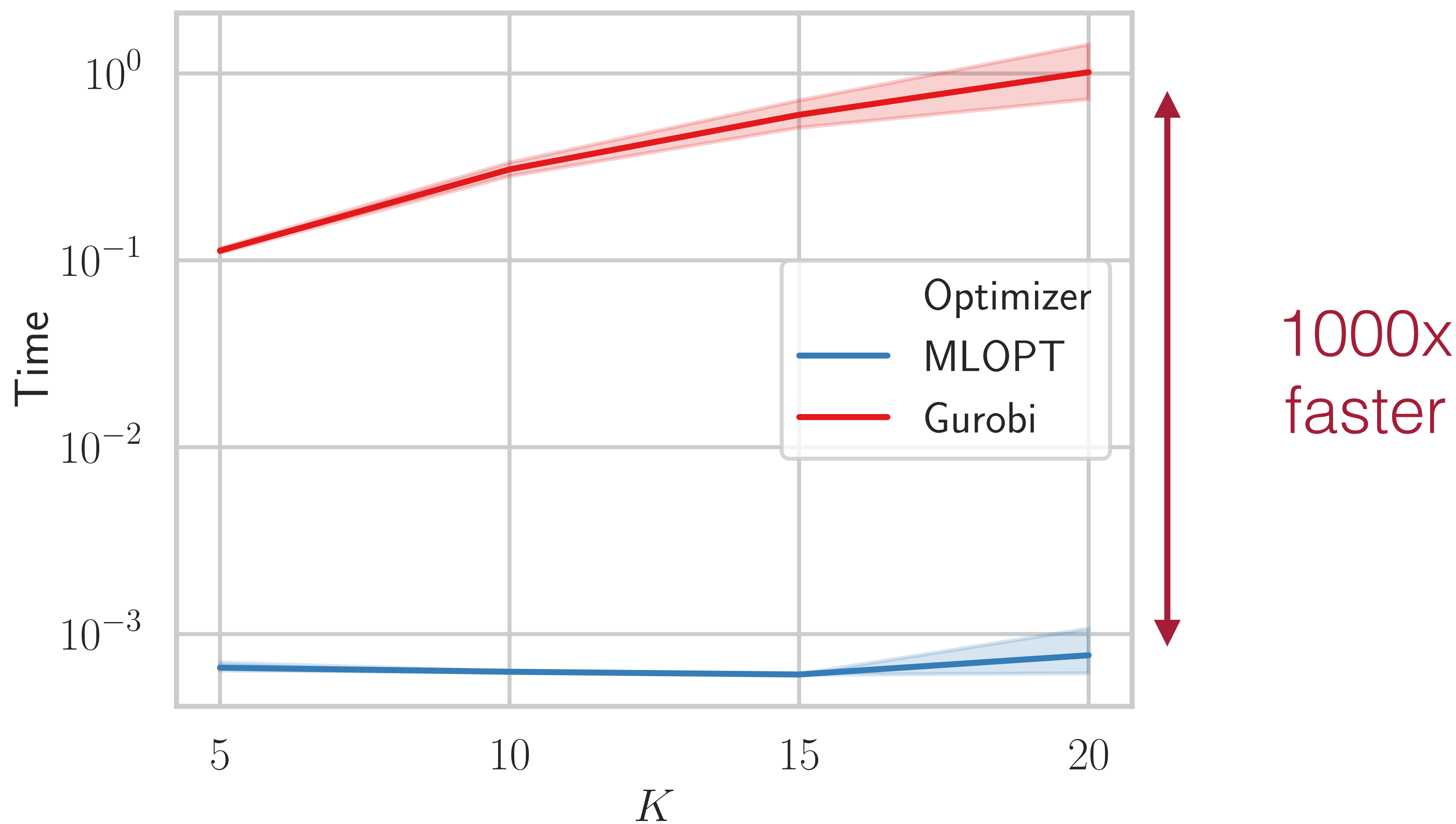
$$\theta = (r, \Sigma^k, F, D)$$

S&P100 backtesting: high speed and accuracy



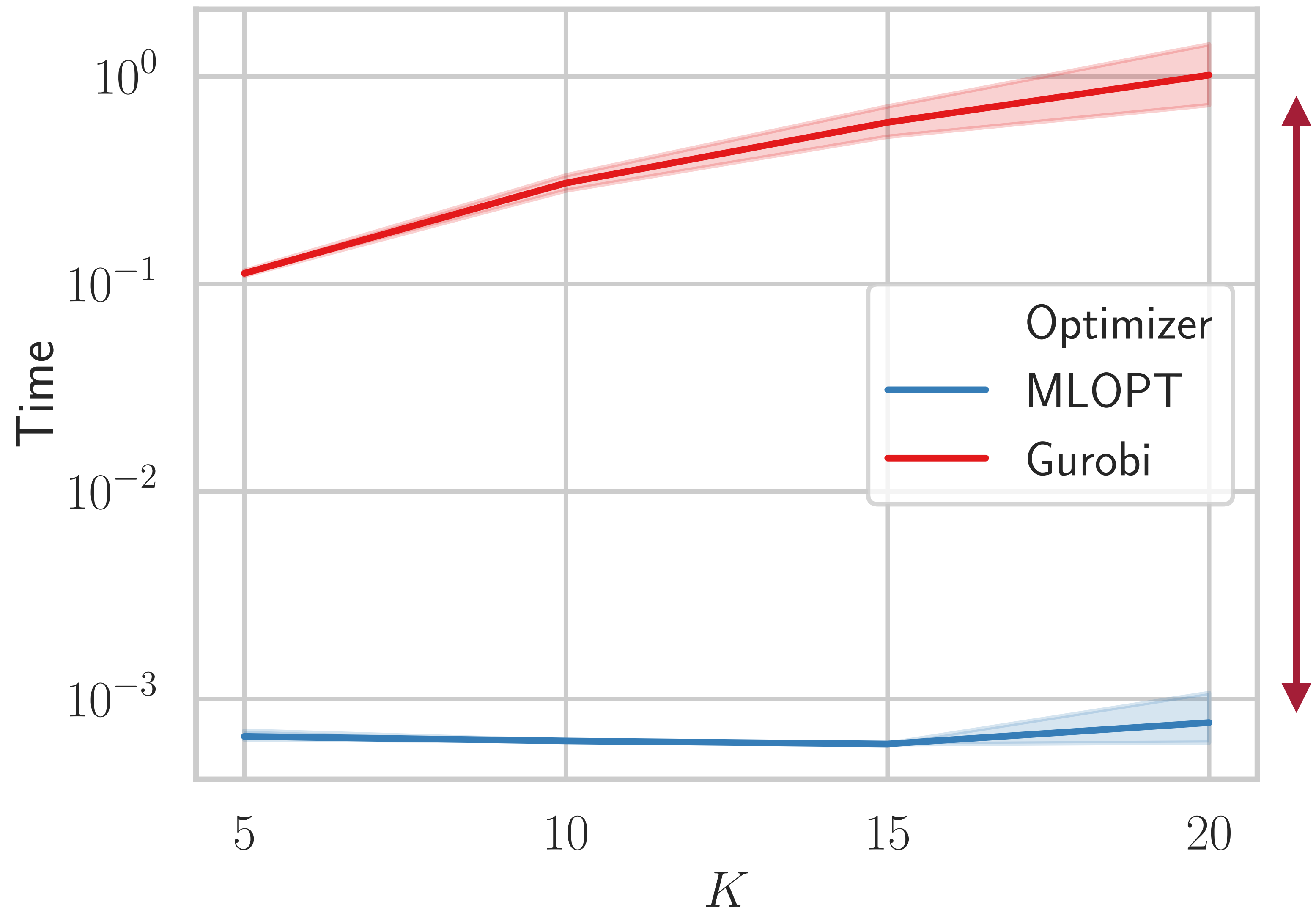
K	M	acc [%]	suboptimality	infeasibility
5	1027	99.11	9.88×10^{-3}	2.25×10^{-9}
10	1083	99.21	7.01×10^{-3}	8.19×10^{-7}
15	1120	99.30	1.92×10^{-2}	2.68×10^{-7}
20	1209	99.50	3.54×10^{-3}	4.12×10^{-7}

S&P100 backtesting: high speed and accuracy



K	M	acc [%]	suboptimality	infeasibility
5	1027	99.11	9.88×10^{-3}	2.25×10^{-9}
10	1083	99.21	7.01×10^{-3}	8.19×10^{-7}
15	1120	99.30	1.92×10^{-2}	2.68×10^{-7}
20	1209	99.50	3.54×10^{-3}	4.12×10^{-7}

S&P100 backtesting: high speed and accuracy

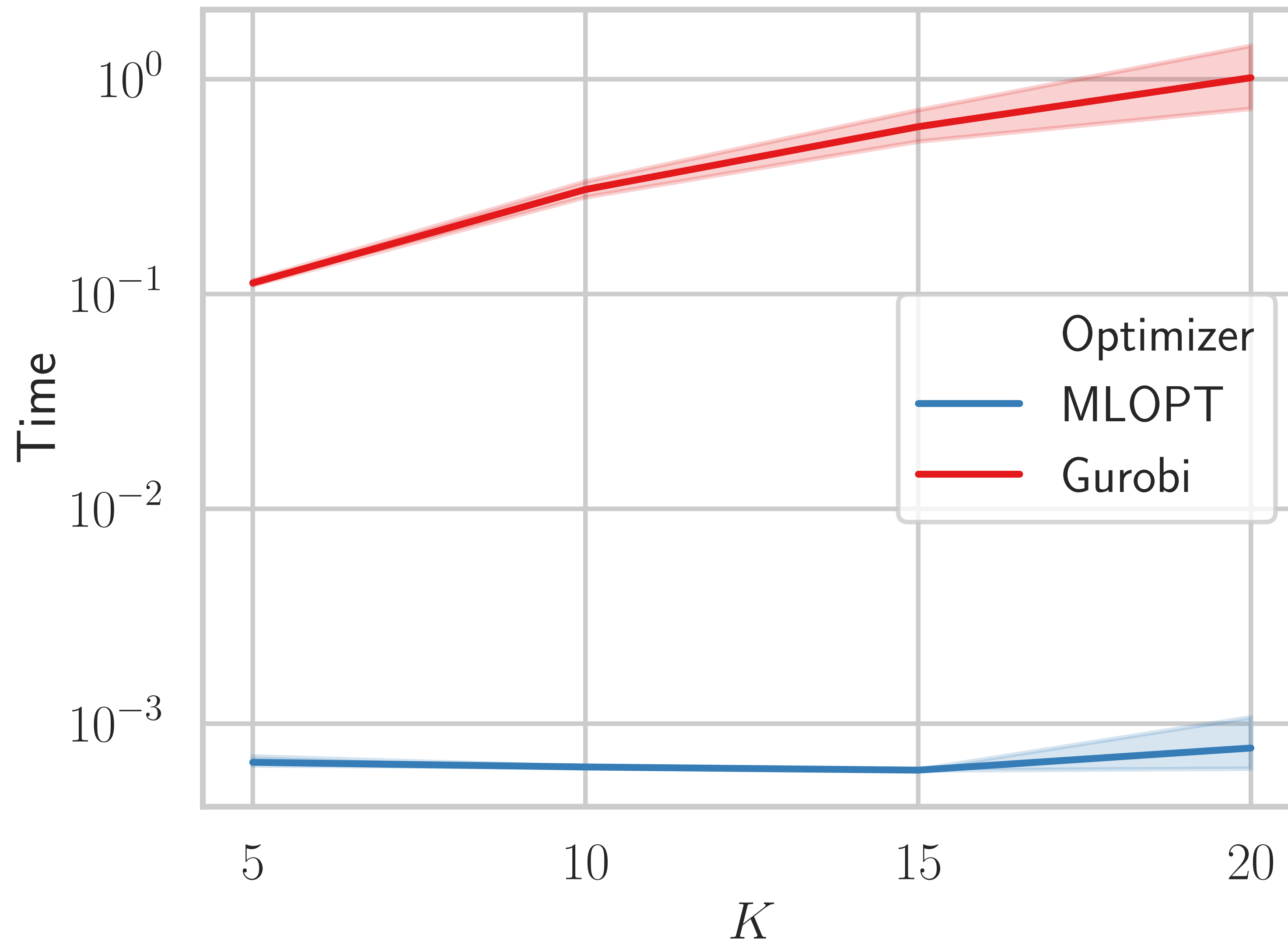


1000x faster

Low suboptimality and infeasibility

K	M	acc [%]	suboptimality	infeasibility
5	1027	99.11	9.88×10^{-3}	2.25×10^{-9}
10	1083	99.21	7.01×10^{-3}	8.19×10^{-7}
15	1120	99.30	1.92×10^{-2}	2.68×10^{-7}
20	1209	99.50	3.54×10^{-3}	4.12×10^{-7}

S&P100 backtesting: high speed and accuracy



1000x faster

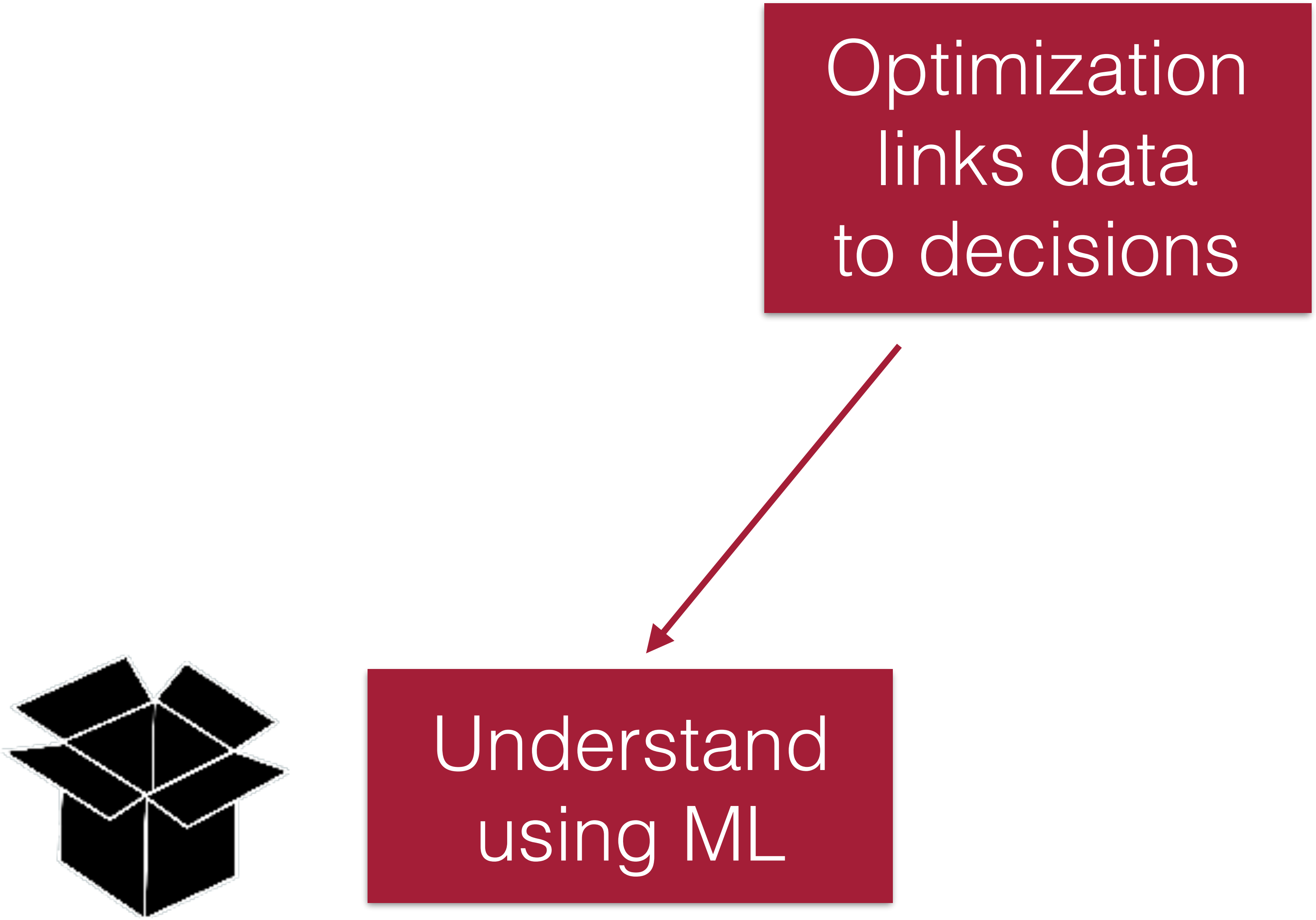
Low suboptimality and infeasibility

High accuracy

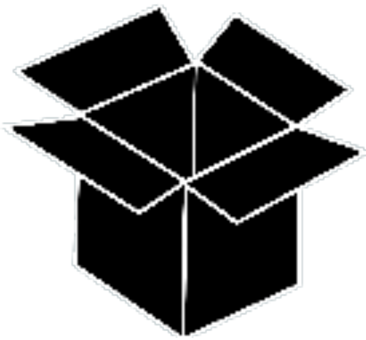
K	M	acc [%]	suboptimality	infeasibility
5	1027	99.11	9.88×10^{-3}	2.25×10^{-9}
10	1083	99.21	7.01×10^{-3}	8.19×10^{-7}
15	1120	99.30	1.92×10^{-2}	2.68×10^{-7}
20	1209	99.50	3.54×10^{-3}	4.12×10^{-7}

Conclusions

Optimization
links data
to decisions


Optimization links data to decisions
Understand using ML

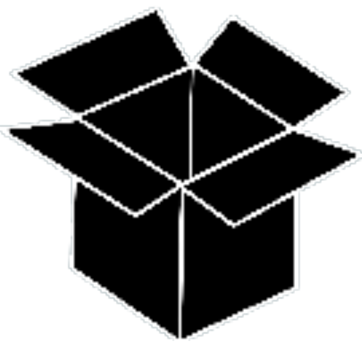
Optimization links data to decisions



Understand using ML

Solve really fast

Optimization links data to decisions



Understand using ML

Solve really fast

Bertsimas, D., Stellato, B., *The Voice of Optimization*, arXiv:1812.09991

Bertsimas, D., Stellato, B., *Online Mixed-Integer Optimization in Milliseconds*, arXiv:1907.02206

stellato@mit.edu

@b_stellato

Backup

Sampling scheme

Algorithm 1 Strategies exploration

given $\epsilon, \beta, \Theta = \emptyset, \mathcal{S} = \emptyset, u = \infty$
for $k = 1, \ldots,$ **do**
  Sample θ_k and compute $s(\theta_k)$ ▷ Sample parameter and strategy.
  $\Theta \leftarrow \Theta \cup \{\theta_k\}$ ▷ Update set of samples.
  if $s(\theta_k) \notin \mathcal{S}$ **then**
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{s(\theta_k)\}$ ▷ Update strategy set if new strategy found
  end if
  if $G + c\sqrt{(1/k)\ln(3/\beta)} \leq \epsilon$ **then** ▷ Break if bound less than ϵ
    break
  end if
end for
return $k, \Theta, \mathcal{S}$