

# Optimisation under Uncertainty

## Lecture 3/4

Fabricio Oliveira

Systems Analysis Laboratory  
Department of Mathematics and Systems Analysis

Aalto University  
School of Science

August 16, 2023

# Outline of this lecture

Introduction

Robust optimisation

Adjustable robust optimisation

# Outline of this lecture

Introduction

Robust optimisation

Adjustable robust optimisation

# What is robust optimisation

An alternative paradigm for taking uncertainty into account:

- ▶ Permeated by the notion of **worst-case**;
- ▶ Control of the degree of **conservatism**;
- ▶ Parallels with **chance constraints** and **risk measures**.

# What is robust optimisation

An alternative paradigm for taking uncertainty into account:

- ▶ Permeated by the notion of **worst-case**;
- ▶ Control of the degree of **conservatism**;
- ▶ Parallels with **chance constraints** and **risk measures**.

In robust optimisation, **feasibility** is the key concern

- ▶ Can be extended to objective function performance requirements;
- ▶ May or may not be **scenario-based**;
- ▶ Static v. adaptable: the presence of **recourse decisions**;
- ▶ Exception: distributionally robust optimisation.

# Robust optimisation approaches

The key notion in robust optimisation is tied to that of an **uncertainty set**

- ▶ The “region”  $U$  within the uncertainty support  $\Xi$  within which **parameter realisation** does not turn the solution infeasible;
- ▶ Tractability is closely tied to the **geometry** of such uncertainty sets.

# Robust optimisation approaches

The key notion in robust optimisation is tied to that of an **uncertainty set**

- ▶ The “region”  $U$  within the uncertainty support  $\Xi$  within which **parameter realisation** does not turn the solution infeasible;
- ▶ Tractability is closely tied to the **geometry** of such uncertainty sets.

$$\min_x c^\top x$$

$$\text{s.t.: } Ax \leq b$$

$$x \in X$$



$$\min_x c^\top x$$

$$\text{s.t.: } A(\eta)x \leq b, \forall \eta \in U \subseteq \Xi$$

$$x \in X$$



$$\min_x c^\top x$$

$$\text{s.t.: } \max_{\eta \in U \subseteq \Xi} A(\eta)x \leq b$$

$$x \in X.$$

# Outline of this lecture

Introduction

Robust optimisation

Adjustable robust optimisation



## Robust counterparts

Let  $\tilde{a}_{ij} \in J_i$  be the **uncertain elements** in the matrix  $A_{m \times n}$

- ▶ Random variables  $\tilde{a}_{ij}$  with “central value”  $a_{ij}$  and “maximum deviation”  $\hat{a}_{ij}$ ;
- ▶ symmetric and with bounded support  $\tilde{a}_{ij} \in [a_{ij} - \hat{a}_{ij}, a_{ij} + \hat{a}_{ij}]$ .

---

<sup>1</sup>Assuming, w.l.g.,  $a_{ij} \geq 0, \forall i \in [m], \forall j \in [n]$

## Robust counterparts

Let  $\tilde{a}_{ij} \in J_i$  be the **uncertain elements** in the matrix  $A_{m \times n}$

- ▶ Random variables  $\tilde{a}_{ij}$  with “central value”  $a_{ij}$  and “maximum deviation”  $\hat{a}_{ij}$ ;
- ▶ symmetric and with bounded support  $\tilde{a}_{ij} \in [a_{ij} - \hat{a}_{ij}, a_{ij} + \hat{a}_{ij}]$ .

Let  $\eta_{ij} = \frac{(\tilde{a}_{ij} - a_{ij})}{\hat{a}_{ij}}$ . Thus  $\eta_{ij} \in [-1, 1]$  and follows the **same distribution** as  $\tilde{a}_{ij}$ , but centred in zero and scaled.

Our robust counterpart<sup>1</sup> is the following **bilevel** problem:

$$\begin{aligned} \min_x \quad & c^\top x \\ \text{s.t.} \quad & a_{ij}x_j + \max_{\eta_i \in U_i} \left\{ \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j \right\} \leq b_i, \forall i \in [m] \\ & x_j \geq 0, \forall j \in [n]. \end{aligned} \tag{RC}$$

---

<sup>1</sup>Assuming, w.l.g.,  $a_{ij} \geq 0, \forall i \in [m], \forall j \in [n]$

# Uncertainty set geometries

Box uncertainty set [Soyster, 1973]

- ▶ **Maximum** protection level;
- ▶ All parameters take their worst-possible value;
- ▶ Simple, but highly conservative.

The **uncertainty set** is

$$U_i = \{\eta_i : \|\eta_i\|_1 \leq |J_i|\} \equiv \{\eta_{ij} : |\eta_{ij}| \leq 1, \forall j \in J_i\}$$

# Uncertainty set geometries

Box uncertainty set [Soyster, 1973]

- ▶ **Maximum** protection level;
- ▶ All parameters take their worst-possible value;
- ▶ Simple, but highly conservative.

The **uncertainty set** is

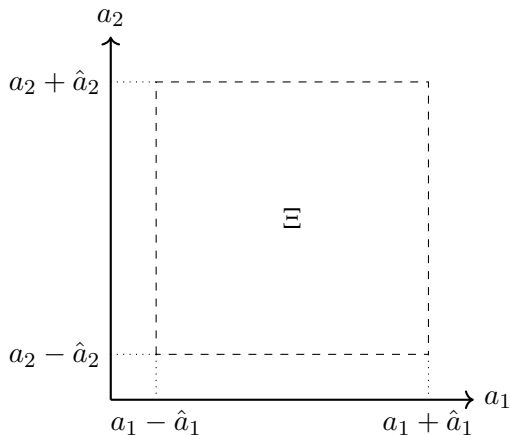
$$U_i = \{\eta_i : \|\eta_i\|_1 \leq |J_i|\} \equiv \{\eta_{ij} : |\eta_{ij}| \leq 1, \forall j \in J_i\}$$

The **lower-level problem** becomes

$$\max_{\eta_i \in U_i} \left\{ \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j : |\eta_{ij}| \leq 1, \forall j \in J_i \right\} = \sum_{j \in J_i} \hat{a}_{ij} x_j.$$

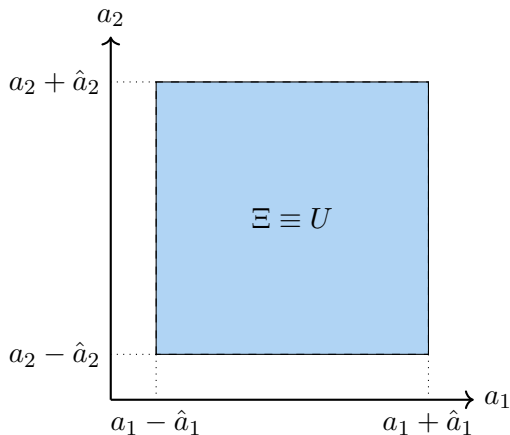
# Uncertainty set geometries

Box uncertainty set [Soyster, 1973]



# Uncertainty set geometries

Box uncertainty set [Soyster, 1973]



# Uncertainty set geometries

Ellipsoidal uncertainty set [Ben-Tal and Nemirovski, 1999]

- ▶ Softens extreme-case protection;
- ▶ Parametrically controlled;
- ▶ Leads to smooth sets;
- ▶ (MI)SOCPs which are more computationally demanding.

The uncertainty set is

$$U_i = \{\eta_i : \|\eta_i\|_2 \leq \Gamma\} \equiv \left\{ \eta_{ij} : \sum_{j \in J_i} \eta_{ij}^2 \leq \Gamma^2, \forall j \in J_i \right\}$$

The lower-level problem becomes

$$\max_{\eta_i \in U_i} \left\{ \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j : \sum_{j \in J_i} \eta_{ij}^2 \leq \Gamma^2, \forall j \in J_i \right\}.$$

# Uncertainty set geometries

Ellipsoidal uncertainty set [Ben-Tal and Nemirovski, 1999]

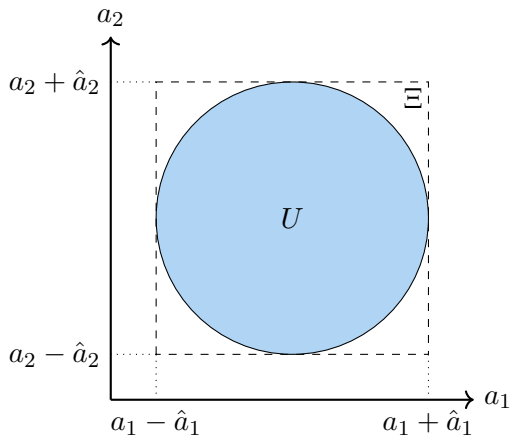
Again, this uncertainty set has a **closed-form** solution:

$$\begin{aligned} & \max_{\eta_i \in U_i} \left\{ \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j : \sum_{j \in J_i} \eta_{ij}^2 \leq \Gamma^2, \forall j \in J_i \right\} \\ &= \max_{\eta_i \in U_i} \left\{ \sqrt{\left( \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j \right)^2} : \sum_{j \in J_i} \eta_{ij}^2 \leq \Gamma^2, \forall j \in J_i \right\} \\ &= \max_{\eta_i \in U_i} \left\{ \sqrt{\left( \sum_{j \in J_i} \eta_{ij} \right)^2 \left( \sum_{j \in J_i} \hat{a}_{ij} x_j \right)^2} : \sum_{j \in J_i} \eta_{ij}^2 \leq \Gamma^2, \forall j \in J_i \right\} \\ &= \Gamma \sqrt{\sum_{j \in J_i} \hat{a}_{ij}^2 x_j^2} \end{aligned}$$



# Uncertainty set geometries

Ellipsoidal uncertainty set [Ben-Tal and Nemirovski, 1999]



# Uncertainty set geometries

Polyhedral uncertainty set [Bertsimas and Sim, 2004]

- ▶ Allows for controlling conservatism;
- ▶ Retains problem complexity;
- ▶ **Budget of uncertainty** lacks interpretability.

The **uncertainty set** is

$$U_i = \{\eta_i : \|\eta_i\|_1 \leq \Gamma_i\} \equiv \left\{ \eta_{ij} : \sum_{j \in J_i} \eta_{ij} \leq \Gamma_i, \forall j \in J_i \right\}.$$

# Uncertainty set geometries

Polyhedral uncertainty set [Bertsimas and Sim, 2004]

- ▶ Allows for controlling conservatism;
- ▶ Retains problem complexity;
- ▶ **Budget of uncertainty** lacks interpretability.

The **uncertainty set** is

$$U_i = \{\eta_i : \|\eta_i\|_1 \leq \Gamma_i\} \equiv \left\{ \eta_{ij} : \sum_{j \in J_i} \eta_{ij} \leq \Gamma_i, \forall j \in J_i \right\}.$$

The **lower-level problem** becomes

$$\max_{\eta_i \in U_i} \left\{ \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j : \sum_{j \in J_i} \eta_{ij} \leq \Gamma_i, 0 \leq \eta_{ij} \leq 1, \forall j \in J_i \right\}.$$

# Uncertainty set geometries

Polyhedral uncertainty set [Bertsimas and Sim, 2004]

In this case, the lower-level problem does not admit a closed form. However, it is a **linear program**.

- ▶ Strong duality (primal-dual equivalence) is available;
- ▶ True for any **convex**<sup>2</sup> lower-level problem.

---

<sup>2</sup>Satisfying some constraint qualification.

# Uncertainty set geometries

Polyhedral uncertainty set [Bertsimas and Sim, 2004]

In this case, the lower-level problem does not admit a closed form.  
However, it is a **linear program**.

- ▶ Strong duality (primal-dual equivalence) is available;
- ▶ True for any **convex**<sup>2</sup> lower-level problem.

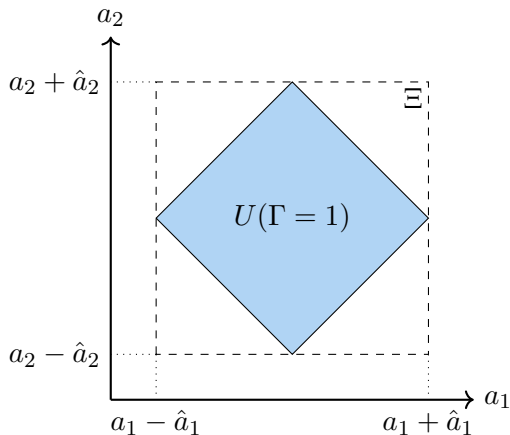
$$\begin{array}{ll} \max_{\eta_i \in U_i} & \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j \\ \text{s.t.:} & \sum_{j \in J_i} \eta_{ij} \leq \Gamma_i \quad (\pi_i) \\ & 0 \leq \eta_{ij} \leq 1, \quad (p_{ij}) \quad \forall j \in J_i \end{array} \quad \Rightarrow \quad \begin{array}{ll} \min_{\pi_i, p_i} & \Gamma_i \pi_i + \sum_{j \in J_i} p_{ij} \\ \text{s.t.:} & \pi_i + p_{ij} \geq \hat{a}_{ij} x_j, \quad \forall j \in J_i \\ & p_{ij} \geq 0, \quad \forall j \in J_i \\ & \pi_i \geq 0. \end{array}$$

---

<sup>2</sup>Satisfying some constraint qualification.

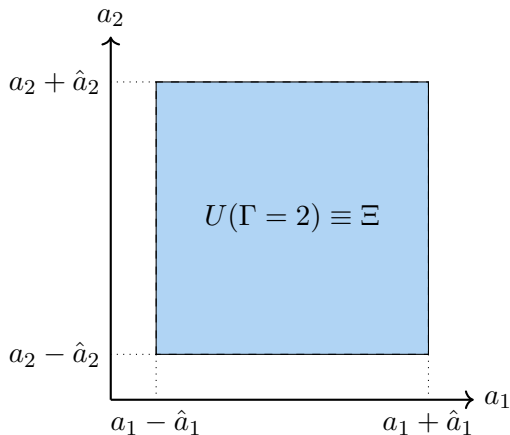
# Uncertainty set geometries

Polyhedral uncertainty set [Bertsimas and Sim, 2004]



# Uncertainty set geometries

Polyhedral uncertainty set [Bertsimas and Sim, 2004]



## Robust counterparts

Let us consider a knapsack problem of the form

$$\begin{aligned} \min. \quad & c^\top x \\ \text{s.t.} \quad & \sum_{j \in [n]} a_j x_j \leq b \\ & 0 \leq x_j \leq 1, \quad \forall j \in [n]. \end{aligned}$$



## Robust counterparts

Let us consider a knapsack problem of the form

$$\begin{aligned} \min. \quad & c^\top x \\ \text{s.t.:} \quad & \sum_{j \in [n]} a_j x_j \leq b \\ & 0 \leq x_j \leq 1, \quad \forall j \in [n]. \end{aligned}$$

The **robust counterparts** for the previous uncertainty sets are:

### Box

$$\begin{aligned} \min. \quad & c^\top x \\ \text{s.t.:} \quad & \sum_{j \in [n]} (a_j + \hat{a}_{ij}) x_j \leq b \\ & 0 \leq x_j \leq 1, \quad \forall j \in [n]. \end{aligned}$$

### Ellipsoid

$$\begin{aligned} \min. \quad & c^\top x \\ \text{s.t.:} \quad & \sum_{j \in [n]} a_j x_j + \Gamma \sqrt{\sum_{j \in [n]} (\hat{a}_j x_j)^2} \leq b \\ & 0 \leq x_j \leq 1, \quad \forall j \in [n]. \end{aligned}$$

## Robust counterparts

Let us consider a knapsack problem of the form

$$\begin{aligned} \min. \quad & c^\top x \\ \text{s.t.:} \quad & \sum_{j \in [n]} a_j x_j \leq b \\ & 0 \leq x_j \leq 1, \quad \forall j \in [n]. \end{aligned}$$

The **robust counterparts** for the previous uncertainty sets are:

### Polyhedral

$$\begin{aligned} \min. \quad & c^\top x \\ \text{s.t.:} \quad & \sum_{j \in [n]} a_j x_j + \Gamma \pi + \sum_{j \in [n]} p_j \leq b \\ & \pi + p_j \geq \hat{a}_j x_j, \quad \forall j \in [n] \\ & 0 \leq x_j \leq 1, p_j \geq 0, \quad \forall j \in [n] \\ & \pi \geq 0. \end{aligned}$$

## On constraint violation probabilities

Arguably, [Bertsimas & Sim \(2004\)](#) raised attention to robust optimisation with “the price of robustness”.

- ▶ The [price](#) refers to the optimality traded off for feasibility guarantees;
- ▶ Quantifying these trade-offs can be done:
  1. Using [theoretical](#) bounds;
  2. Via [simulating](#) solution performance.
- ▶ In my own experience, theoretical bounds are often loose.

## On constraint violation probabilities

Arguably, [Bertsimas & Sim \(2004\)](#) raised attention to robust optimisation with “the price of robustness”.

- ▶ The [price](#) refers to the optimality traded off for feasibility guarantees;
- ▶ Quantifying these trade-offs can be done:
  1. Using [theoretical](#) bounds;
  2. Via [simulating](#) solution performance.
- ▶ In my own experience, theoretical bounds are often loose.

For example, [Bertsimas and Sim, 2004] show the probability of violation of constraint  $i \in [m]$  to be

$$P^{\text{vio}} = P \left( a_{ij}x_j + \max_{\eta_i \in U_i} \left\{ \sum_{j \in J_i} \eta_{ij} \hat{a}_{ij} x_j \right\} > b_i \right) \leq e^{\frac{-\Gamma^2}{2|J_i|}},$$

# Outline of this lecture

Introduction

Robust optimisation

Adjustable robust optimisation

# Multi-stage robust optimisation

We focus on 2-stage adjustable robust optimisation (ARO) problems:

$$\begin{aligned} \min \quad & c^\top x + \max_{\xi \in U \subset \Xi} \min_y q^\top y(\xi) \\ \text{s.t.:} \quad & Ax = b, \ x \geq 0 \\ & Tx + Wy(\xi) = h(\xi), \ \forall \xi \in U \subset \Xi \\ & y(\xi) \geq 0, \ \forall \xi \in \Xi. \end{aligned} \tag{ARO}$$

- ▶ Only **RHS** uncertainty:  $T(\xi) = T$ ,  $W(\xi) = W$ , and  $q(\xi) = q \ \forall \xi \in \Xi$ ;
- ▶ Assumption often necessary to eliminate **quadratic** dependence between  $\xi$  and decision variables;
- ▶ Not necessary if the uncertainty set is **discrete** and **finite** (scenarios)

## A side note: mini-max, minimum regret and related

If the uncertainty set is a finite and discrete set of scenarios, we have that

$$\begin{aligned} \min_x \quad & c^\top x + \max_{s \in U} q_s^\top y_s \\ \text{s.t.:} \quad & T_s x + W_s y_s \leq h_s, \quad \forall s \in U \\ & x \in X \end{aligned}$$

is a **tractable** ARO [Mulvey et al., 1995].

## A side note: mini-max, minimum regret and related

If the uncertainty set is a finite and discrete set of scenarios, we have that

$$\begin{aligned} \min_x. \quad & c^\top x + \max_{s \in U} q_s^\top y_s \\ \text{s.t.:} \quad & T_s x + W_s y_s \leq h_s, \quad \forall s \in U \\ & x \in X \end{aligned}$$

is a **tractable** ARO [Mulvey et al., 1995]. Variants include:

### Minimax

$$\begin{aligned} \min_x. \quad & c^\top x + \theta \\ & \theta \geq q_s^\top y_s, \quad \forall s \in U \\ \text{s.t.:} \quad & T_s x + W_s y_s \leq h_s, \quad \forall s \in U \\ & x \in X \end{aligned}$$

### Min-regret

$$\begin{aligned} \min_x. \quad & c^\top x + \theta \\ & \theta \geq q_s^\top y_s - q_s^\top y_s^*, \quad \forall s \in U \\ \text{s.t.:} \quad & T_s x + W_s y_s \leq h_s, \quad \forall s \in U \\ & x \in X \end{aligned}$$

where  $y_s^*$  is optimal for  $s \in U$ .



## Affinely adjustable robust optimisation [Ben-Tal et al., 2004]

One idea for modelling adjustability is using **affine policies**:

- ▶ Replace  $y(\xi)$  with  $\alpha + \beta\xi$ ;
- ▶  $h(\xi)$  is assumed **affinely dependent** on  $\xi$ , e.g.:  $h(\xi) = h + \hat{h}\xi$ .

## Affinely adjustable robust optimisation [Ben-Tal et al., 2004]

One idea for modelling adjustability is using **affine policies**:

- ▶ Replace  $y(\xi)$  with  $\alpha + \beta\xi$ ;
- ▶  $h(\xi)$  is assumed **affinely dependent** on  $\xi$ , e.g.:  $h(\xi) = h + \hat{h}\xi$ .

Then ARO becomes:

$$\begin{aligned} \min_{x, \alpha, \beta} \quad & c^\top x + \theta \\ \text{s.t.:} \quad & \theta \geq q^\top (\alpha + \beta\xi), \quad \forall \xi \in U \\ & Ax = b, \quad x \geq 0 \\ & Tx + W(\alpha + \beta\xi) \leq h(\xi), \quad \forall \xi \in U \\ & y(\xi) \geq 0, \quad \forall \xi \in \Xi. \end{aligned}$$

## Affinely adjustable robust optimisation [Ben-Tal et al., 2004]

One idea for modelling adjustability is using **affine policies**:

- ▶ Replace  $y(\xi)$  with  $\alpha + \beta\xi$ ;
- ▶  $h(\xi)$  is assumed **affinely dependent** on  $\xi$ , e.g.:  $h(\xi) = h + \hat{h}\xi$ .

Then ARO becomes:

$$\begin{aligned} \min_{x, \alpha, \beta} \quad & c^\top x + \theta \\ \text{s.t.:} \quad & \theta \geq q^\top (\alpha + \beta\xi), \quad \forall \xi \in U \\ & Ax = b, \quad x \geq 0 \\ & Tx + W(\alpha + \beta\xi) \leq h(\xi), \quad \forall \xi \in U \\ & y(\xi) \geq 0, \quad \forall \xi \in \Xi. \end{aligned}$$

Similar to the static case, computational **tractability** can be achieved:

- ▶ Requires that  $U$  is a box or ellipsoidal set;
- ▶ For a practical example, see [Ben-Tal et al., 2005]

## Adjustable robust optimisation

An **alternative** approach: looking closer at the inner problem as a **bilevel** optimisation problem.

Let us restate our ARO in a simplified notation. For that, let

- ▶  $X = \{x \in \mathbb{R}^{n_1} : Ax = b, x \geq 0\};$
- ▶  $Y = \{y \in \mathbb{R}^{n_2} : y \geq 0\};$
- ▶ Uncertainty in **RHS only**, with  $h(\xi) = h - \hat{h}\xi$ , and  $\xi \in [\underline{\xi}, \bar{\xi}]$

## Adjustable robust optimisation

An **alternative** approach: looking closer at the inner problem as a **bilevel** optimisation problem.

Let us restate our ARO in a simplified notation. For that, let

- ▶  $X = \{x \in \mathbb{R}^{n_1} : Ax = b, x \geq 0\};$
- ▶  $Y = \{y \in \mathbb{R}^{n_2} : y \geq 0\};$
- ▶ Uncertainty in **RHS only**, with  $h(\xi) = h - \hat{h}\xi$ , and  $\xi \in [\underline{\xi}, \bar{\xi}]$

Then we have that ARO is equivalent to

$$\min_{x \in X} c^\top x + \mathcal{Q}(x), \quad (\text{ARO})$$

where

$$\mathcal{Q}(x) = \left\{ \max_{\xi \in U} \min_{y \in Y} q^\top y : Tx = (h - \hat{h}\xi) - Wy \right\}.$$

## Adjustable robust optimisation & CCG

Let us assume that an **oracle** is available such that, for a given  $\bar{x} \in X$  it evaluates  $\mathcal{Q}(x)$  and returns associated  $(\bar{\xi}, \bar{y})$ , if they exist.

## Adjustable robust optimisation & CCG

Let us assume that an **oracle** is available such that, for a given  $\bar{x} \in X$  it evaluates  $\mathcal{Q}(x)$  and returns associated  $(\bar{\xi}, \bar{y})$ , if they exist.

In addition, let us assume that the uncertainty set is **finitely** representable:

- ▶ Scenarios, but an **intractable amount** of them (e.g., samples, or data)
- ▶ **Polyhedral** set (finite extreme points and rays)

# Adjustable robust optimisation & CCG

Let us assume that an **oracle** is available such that, for a given  $\bar{x} \in X$  it evaluates  $\mathcal{Q}(x)$  and returns associated  $(\bar{\xi}, \bar{y})$ , if they exist.

In addition, let us assume that the uncertainty set is **finitely** representable:

- Scenarios, but an **intractable amount** of them (e.g., samples, or data)
- **Polyhedral** set (finite extreme points and rays)

In that case, we can employ **column-and-constraint generation** (CCG) [Zeng and Zhao, 2013] to solve ARO:

Main problem  $M^k$ :  $\bar{x}^{k+1}$

$$\min_{x, y, \theta} c^\top x + \theta$$

$$\theta \geq q^\top y_l, \quad l \in [k]$$

$$x \in X$$

$$Tx = h - \hat{h}\bar{\xi}_l - Wy_l, \quad l \in [k]$$

$$y_l \in Y, \quad l \in [k].$$

Oracle  $\mathcal{Q}(\bar{x}^{k+1})$ :  $\bar{\xi}^{k+1}$

$$\max_{\xi \in U} \min_{y \in Y} q^\top y$$

$$\text{s.t.: } T\bar{x}^{k+1} = (h - \hat{h}\xi) - Wy.$$



# Adjustable robust optimisation & CCG

In summary, the CCG method can be stated as

1. **Initialisation.**  $LB = -\infty$ ,  $UB = \infty$ ,  $k = 0$ .

# Adjustable robust optimisation & CCG

In summary, the CCG method can be stated as

1. **Initialisation.**  $LB = -\infty$ ,  $UB = \infty$ ,  $k = 0$ .
2. **Solve the main problem  $M^k$ .** Let  $\underline{z}^k = c^\top x^k + \theta^k$ , where  $\operatorname{argmin} M^k = (x^k, \theta^k, (y_l^k)_{l=1}^k)$ . Make  $LB = \underline{z}^k$ .

# Adjustable robust optimisation & CCG

In summary, the CCG method can be stated as

1. **Initialisation.**  $LB = -\infty$ ,  $UB = \infty$ ,  $k = 0$ .
2. **Solve the main problem  $M^k$ .** Let  $\underline{z}^k = c^\top x^k + \theta^k$ , where  $\operatorname{argmin} M^k = (x^k, \theta^k, (y_l^k)_{l=1}^k)$ . Make  $LB = \underline{z}^k$ .
3. **Solve  $Q(x^k)$ .** Let  $\operatorname{argmin} Q(x^k) = (\bar{\xi}^{k+1}, \bar{y}^{k+1})$ , if it exists. Let  $\bar{z}^k = c^\top x^k + Q(x^k)$ . Make  $UB = \min \{UB, \bar{z}^k\}$ . **If  $UB = LB < \epsilon$ , return  $x^k$ .**

# Adjustable robust optimisation & CCG

In summary, the CCG method can be stated as

1. **Initialisation.**  $LB = -\infty$ ,  $UB = \infty$ ,  $k = 0$ .
2. **Solve the main problem  $M^k$ .** Let  $\underline{z}^k = c^\top x^k + \theta^k$ , where  $\operatorname{argmin} M^k = (x^k, \theta^k, (y_l^k)_{l=1}^k)$ . Make  $LB = \underline{z}^k$ .
3. **Solve  $Q(x^k)$ .** Let  $\operatorname{argmin} Q(x^k) = (\bar{\xi}^{k+1}, \bar{y}^{k+1})$ , if it exists. Let  $\bar{z}^k = c^\top x^k + Q(x^k)$ . Make  $UB = \min \{UB, \bar{z}^k\}$ . **If  $UB = LB < \epsilon$ , return  $x^k$ .**
4. **Add column and constraints to  $M^k$ .** If  $Q(x^k)$  is feasible, create columns  $y_{k+1}$  and, together with the constraints

$$\theta \geq q^\top y_{k+1} \tag{1}$$

$$Tx = h - \hat{h}\bar{\xi}_{k+1} - Wy_{k+1}, \quad y_{k+1} \in Y, \tag{2}$$

add them to  $M^k$ , forming  $M^{k+1}$ . Make  $k = k + 1$  and return to [Step 2](#). If  $Q(x^k)$  is not feasible, then only (2) is created.

# Adjustable robust optimisation & CCG

## Practical remarks

Essentially, CCG for ARO is a **delayed-generation** approach of the min-max formulation

- ▶ Can thus be useful when **too many scenarios** are available;
- ▶ Convergence relies on a **finiteness argument** on the uncertainty set.

# Adjustable robust optimisation & CCG

## Practical remarks

Essentially, CCG for ARO is a **delayed-generation** approach of the min-max formulation

- ▶ Can thus be useful when **too many scenarios** are available;
- ▶ Convergence relies on a **finiteness argument** on the uncertainty set.

CCG can be seen as a **primal equivalent** to Benders decomposition

- ▶ One can use the same column generation approach in the context of the L-shaped method [Van Slyke and Wets, 1969];
- ▶ This can help as a way to **transmit** “recourse information” to the main problem.

# Adjustable robust optimisation & CCG

On solving  $\mathcal{Q}(x)$

Recall that  $\mathcal{Q}(x)$  is of the form

$$\mathcal{Q}(x) = \max_u q^\top y$$

$$\text{s.t.: } u \in \mathcal{U}$$

$$y \in \operatorname{argmin}_y q^\top y$$

$$\text{s.t.: } Tx = h - \hat{h}\xi - Wy$$

$$y \in Y.$$

# Adjustable robust optimisation & CCG

On solving  $\mathcal{Q}(x)$

Recall that  $\mathcal{Q}(x)$  is of the form

$$\begin{aligned}\mathcal{Q}(x) = & \max_u q^\top y \\ \text{s.t.: } & u \in \mathcal{U} \\ & y \in \operatorname{argmin}_y q^\top y \\ & \text{s.t.: } Tx = h - \hat{h}\xi - Wy \\ & y \in Y.\end{aligned}$$

This is a **bilevel model** and can be solved using dedicated methods.

- ▶ Most techniques rely on **posing optimality conditions** of the lower-level problem to yield an **equivalent single-level** (tractable) problem;
- ▶ Thus, **lower-level convexity** (plus CQ) is often a requirement.



# Adjustable robust optimisation & CCG

On solving  $\mathcal{Q}(x)$





Example: assume that  $Y = \mathbb{R}_+^{n_2}$ . We can use strong duality to reformulate the lower-level problem, obtaining

$$\begin{aligned}\mathcal{Q}(x) = & \max_{\xi, \pi} (h - \hat{h}\xi - Tx)^\top \pi \\ \text{s.t.: } & \pi^\top W \leq q^\top \\ & \xi \in U.\end{aligned}$$

$\mathcal{Q}(x)$  is solvable, if:

1.  $\xi$  is integer or has a discrete domain, since  $\xi^\top \pi$  can be reformulated exactly (e.g., [Rintamäki et al., 2023]);
2. if  $-(\hat{h}\xi)^\top \pi + (h - Tx)^\top \pi$  is a **concave bilinear** function in  $\pi$  and  $\xi$ ;
3. if applying a **global solver** (e.g., Gurobi's spatial branch-and-bound method) is feasible from a computational standpoint.

# References I

-  Ben-Tal, A., Golany, B., Nemirovski, A., and Vial, J.-P. (2005).  
Retailer-supplier flexible commitments contracts: A robust optimization approach.  
*Manufacturing & Service Operations Management*, 7(3):248–271.
-  Ben-Tal, A., Goryashko, A., Guslitzer, E., and Nemirovski, A. (2004).  
Adjustable robust solutions of uncertain linear programs.  
*Mathematical programming*, 99(2):351–376.
-  Ben-Tal, A. and Nemirovski, A. (1999).  
Robust solutions of uncertain linear programs.  
*Operations research letters*, 25(1):1–13.
-  Bertsimas, D. and Sim, M. (2004).  
The price of robustness.  
*Operations research*, 52(1):35–53.

## References II



Mulvey, J. M., Vanderbei, R. J., and Zenios, S. A. (1995).

Robust optimization of large-scale systems.

*Operations research*, 43(2):264–281.



Rintamäki, T., Oliveira, F., Siddiqui, A. S., and Salo, A. (2023).

Achieving emission-reduction goals: Multi-period power-system expansion under short-term operational uncertainty.

*IEEE Transactions on Power Systems*.



Soyster, A. L. (1973).

Convex programming with set-inclusive constraints and applications to inexact linear programming.

*Operations research*, 21(5):1154–1157.

## References III



Van Slyke, R. M. and Wets, R. (1969).

L-shaped linear programs with applications to optimal control and stochastic programming.

*SIAM journal on applied mathematics*, 17(4):638–663.



Zeng, B. and Zhao, L. (2013).

Solving two-stage robust optimization problems using a column-and-constraint generation method.

*Operations Research Letters*, 41(5):457–461.