Robust combinatorial optimization with polyhedral uncertainty

Michael Poss
based on joint work with Marin Bougeret, Jérémy Omer, Artur Pessoa

October 2, 2024

Table of Contents

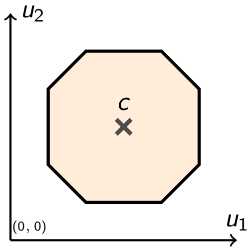
Given:

- $\mathcal{Y} \subseteq \{0,1\}^n$: feasibility set of an optimization problem
- $U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$: uncertainty polytope
- c: nominal cost vector

Solve:

$$\min_{y \in \mathcal{Y}} \max_{u \in U} \; u^\top y \qquad \text{(Min-Max)}$$

Example of U:



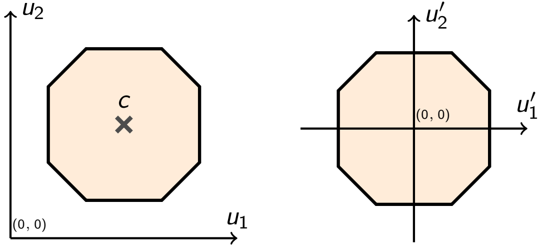
Given:

- $\mathcal{Y} \subseteq \{0,1\}^n$: feasibility set of an optimization problem
- $U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$: uncertainty polytope
- c: nominal cost vector

Solve:

$$\min_{y \in \mathcal{Y}} \max_{u \in U} (c + u')^\top y \qquad \text{(Min-Max)}$$

Example of U:



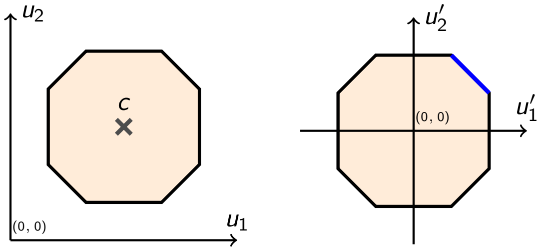
Given:

- $\mathcal{Y} \subseteq \{0,1\}^n$: feasibility set of an optimization problem
- $U = \{u \in \mathbb{R}^n \mid \mathbf{Au} \leq \mathbf{b}, 0 \leq u \leq d\}$: uncertainty polytope
- c: nominal cost vector

Solve:

$$\min_{y \in \mathcal{Y}} \max_{u \in U} (c + u')^\top y \qquad \text{(Min-Max)}$$

Example of U:



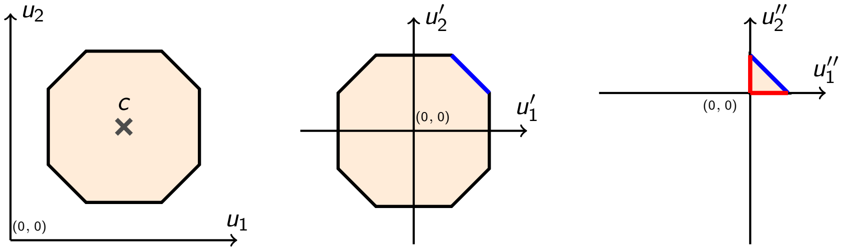
Given:

- $\mathcal{Y} \subseteq \{0,1\}^n$: feasibility set of an optimization problem
- $U = \{u \in \mathbb{R}^n \mid \mathbf{Au} \leq \mathbf{b}, \mathbf{0} \leq \mathbf{u} \leq d\}$: uncertainty polytope
- c: nominal cost vector

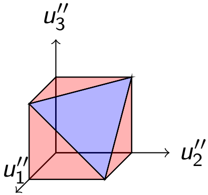
Solve:

$$\min_{y \in \mathcal{Y}} \max_{u \in U} \, (c' + u'')^\top y \qquad \text{(Min-Max)}$$

Example of U:



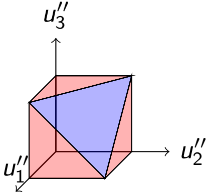
A 3D example



Translated description:

$U'' = \{u'' \in \mathbb{R}^3 \mid u_1'' + u_2'' + u_3'' \leq 2, 0 \leq u'' \leq 1\}$

A 3D example

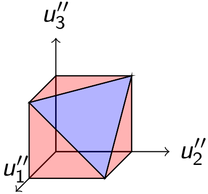


Translated description:
$U'' = \left\{ u'' \in \mathbb{R}^3 \mid u_1'' + u_2'' + u_3'' \leq 2, 0 \leq u'' \leq 1 \right\}$

"Natural" description: 8 symmetric copies of the polytope
$U = \left\{ u \in \mathbb{R}^3 \mid c_i - 1 \leq u_i \leq c_i + 1,\ \forall i \in [3], \quad \sum_{i=1}^{3} |u_i - c_i| \leq 2 \right\}$

A 3D example

Translated description:
$U'' = \left\{ u'' \in \mathbb{R}^3 \mid u''_1 + u''_2 + u''_3 \leq 2, 0 \leq u'' \leq 1 \right\}$



"Natural" description: 8 symmetric copies of the polytope

$$U = \left\{ u \in \mathbb{R}^3 \mid c_i - 1 \leq u_i \leq c_i + 1,\ \forall i \in [3], \quad \sum_{i=1}^{3} |u_i - c_i| \leq 2 \right\}$$

$$= \left\{ u \in \mathbb{R}^3 \mid c_i - 1 \leq u_i \leq c_i + 1,\ \forall i \in [3], \quad \sum_{i=1}^{3} (u_i - c_i) \leq 2, \quad \sum_{i=1}^{3} (-u_i + c_i) \leq 2, \right.$$

$$\sum_{i \in I} (u_i - c_i) - u_j + c_j \leq 2,\ \forall I \cup j = [3],$$

$$\left. \sum_{i \in I} (-u_i + c_i) + u_j - c_j \leq 2,\ \forall I \cup j = [3] \right\}$$

Hardness I: the 2-scenarios case is hard

Theorem

The robust selection problem is NP-hard even when $|U| = 2$*.*

Proof.

$\square$

Hardness I: the 2-scenarios case is hard

Theorem

The robust selection problem is NP-hard even when $|U| = 2$.

Proof.

1. SELECTION PROBLEM: $\min_{S \subseteq N, |S|=p} \sum_{i \in S} u_i$

□

Hardness I: the 2-scenarios case is hard

Theorem

The robust selection problem is NP-hard even when $|U| = 2$.

Proof.

1. SELECTION PROBLEM: $\min\limits_{S \subseteq N, |S| = p} \sum\limits_{i \in S} u_i$
2. ROBUST SEL. PROB.: $\min\limits_{S \subseteq N, |S| = p} \max\limits_{u \in U} \sum\limits_{i \in S} u_i$

□

Hardness I: the 2-scenarios case is hard

Theorem

The robust selection problem is NP-hard even when $|U| = 2$.

Proof.

1. SELECTION PROBLEM: $\min_{S \subseteq N, |S|=p} \sum_{i \in S} u_i$
2. ROBUST SEL. PROB.: $\min_{S \subseteq N, |S|=p} \max_{u \in U} \sum_{i \in S} u_i$
3. PARTITION PROBLEM: $\min_{S \subseteq N, |S|=|N|/2} \max \left(\sum_{i \in S} a_i, \sum_{i \in N \setminus S} a_i \right)$

□

Hardness I: the 2-scenarios case is hard

Theorem

The robust selection problem is NP-hard even when $|U| = 2$.

Proof.

1. SELECTION PROBLEM: $\min_{S \subseteq N, |S| = p} \quad \sum_{i \in S} u_i$
2. ROBUST SEL. PROB.: $\min_{S \subseteq N, |S| = p} \quad \max_{u \in U} \quad \sum_{i \in S} u_i$
3. PARTITION PROBLEM: $\min_{S \subseteq N, |S| = |N|/2} \quad \max\left(\sum_{i \in S} a_i, \sum_{i \in N \setminus S} a_i \right)$
4. Reduction: $p = \frac{|N|}{2}$, and $U = \{u^1, u^2\}$ such that

$$u_i^1 = a_i \qquad \text{and} \qquad u_i^2 = \frac{2}{|N|} \sum_k a_k - a_i$$

$$\Rightarrow \quad \max_{u \in U} \sum_{i \in S} u_i = \max\left(\sum_{i \in S} a_i, \sum_{i \in N \setminus S} a_i \right)$$

□

Hardness II: polytope is more general than 2 scenarios

Corollary

The robust selection problem is NP-hard **when the number of rows of** A **is part of the input**.

Proof.

□

Hardness II: polytope is more general than 2 scenarios

Corollary

The robust selection problem is NP-hard **when the number of rows of** A **is part of the input**.

Proof.

1. $y^* \in \arg\min_{y \in \mathcal{Y}} \max_{u \in U} (c+u)^\top y \Leftrightarrow y^* \in \arg\min_{y \in \mathcal{Y}} \max_{u \in \text{ext}(U)} (c+u)^\top y$

□

Hardness II: polytope is more general than 2 scenarios

Corollary

The robust selection problem is NP-hard **when the number of rows of A is part of the input**.

Proof.

1. $y^* \in \arg\min_{y \in \mathcal{Y}} \max_{u \in U} (c+u)^\top y \Leftrightarrow y^* \in \arg\min_{y \in \mathcal{Y}} \max_{u \in \text{ext}(U)} (c+u)^\top y$
2. $\{u^1, u^2\}$ is hard $\implies$ $\text{conv}(\{u^1, u^2\})$ is hard.

□

Hardness II: polytope is more general than 2 scenarios

Corollary

The robust selection problem is NP-hard **when the number of rows of** *A* **is part of the input**.

Proof.

1. $y^* \in \arg\min_{y \in \mathcal{Y}} \max_{u \in U} (c+u)^\top y \Leftrightarrow y^* \in \arg\min_{y \in \mathcal{Y}} \max_{u \in \text{ext}(U)} (c+u)^\top y$
2. $\{u^1, u^2\}$ is hard $\implies$ $\text{conv}(\{u^1, u^2\})$ is hard.
3. $\text{conv}(\{u^1, u^2\})$ can be described by $2n+2$ inequalities

□

Constant number of rows

$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$, s denotes the number of rows of A

Theorem

Solving (Min-Max) *amounts to solve* $O(n^s)$ *problems of the form*
$\min_{y \in \mathcal{Y}} \bar{c}^\top y$

Proof.

Constant number of rows

$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$, s denotes the number of rows of A

Theorem

Solving (Min-Max) *amounts to solve* $O(n^s)$ *problems of the form* $\min_{y \in \mathcal{Y}} \bar{c}^\top y$

Proof.

1. dualize the maximization over $u \implies$ dual variables $(\alpha, \pi) \in D(y)$

□

Constant number of rows

$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$, s denotes the number of rows of A

Theorem

Solving (Min-Max) *amounts to solve* $O(n^s)$ *problems of the form* $\min_{y \in \mathcal{Y}} \bar{c}^\top y$

Proof.

1. dualize the maximization over $u \implies$ dual variables $(\alpha, \pi) \in D(y)$
2. we can construct a set $|D|^* \in O(n^s)$ such that:

$$\bigcup_{x \in \{0,1\}^n} \text{Proj}_\alpha \Big[\text{ext}(D(y)) \Big] \subseteq D^*$$

$\implies$ enumerate $\alpha \in D^*$

□

Constant number of rows

$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$, s denotes the number of rows of A

Theorem

Solving (Min-Max) *amounts to solve* $O(n^s)$ *problems of the form* $\min_{y \in \mathcal{Y}} \bar{c}^\top y$

Proof.

1. dualize the maximization over $u \implies$ dual variables $(\alpha, \pi) \in D(y)$
2. we can construct a set $|D|^* \in O(n^s)$ such that:

$$\bigcup_{x \in \{0,1\}^n} \text{Proj}_\alpha \Big[\text{ext}(D(y)) \Big] \subseteq D^*$$

$\implies$ enumerate $\alpha \in D^*$

3. variables π_k can be substituted by $\max(y_k - f(\alpha), 0)$

□

Constant number of rows

$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}$, s denotes the number of rows of A

Theorem

Solving (Min-Max) *amounts to solve* $O(n^s)$ *problems of the form* $\min_{y \in \mathcal{Y}} \bar{c}^\top y$

Proof.

1. dualize the maximization over $u \implies$ dual variables $(\alpha, \pi) \in D(y)$
2. we can construct a set $|D|^* \in O(n^s)$ such that:

$$\bigcup_{x \in \{0,1\}^n} \text{Proj}_\alpha \Big[\text{ext}(D(y)) \Big] \subseteq D^*$$

$\implies$ enumerate $\alpha \in D^*$

3. variables π_k can be substituted by $\max(y_k - f(\alpha), 0)$
4. $\max(y_k - f(\alpha)) = y_k \max(1 - f(\alpha)) + (1 - y_k) \max(-f(\alpha), 0)$

□

Extension I: non-linear objective

The previous results extend to:

$$\min_{y\in\mathcal{Y}} \max_{u\in U} (c+u)^\top f(x)$$

where $f_k(x) \in \{0, F_k\}$ for each $y \in \mathcal{Y}$.

Examples

- max-cut with uncertain weights:

$$\max_{y\in\mathcal{Y}} \min_{w\in U} \sum_{\{i,j\}\in E} w_{ij} y_i (1-y_j),$$

with $F_{ij} = 1$.

Extension I: non-linear objective

The previous results extend to:

$$\min_{y \in \mathcal{Y}} \max_{u \in U} (c + u)^\top f(x)$$

where $f_k(x) \in \{0, F_k\}$ for each $y \in \mathcal{Y}$.

Examples

- max-cut with uncertain weights:

$$\max_{y \in \mathcal{Y}} \min_{w \in U} \sum_{\{i,j\} \in E} w_{ij} y_i (1 - y_j),$$

with $F_{ij} = 1$.

- minimizing weight of falling jobs

$$\min_{y \in \mathcal{Y}} \max_{w \in U} \sum_{i \in J} w_i U_i(y),$$

with $U_i(y) = 1$ iff job i fails in schedule y.

Extension II: constraint uncertainty

$$\begin{aligned} \min \quad & c^\top y \\ \text{s.t.} \quad & (a+u)^\top y \le h, \quad \forall u \in U \\ & y \in \mathcal{Y} \end{aligned} \tag{1}$$

Define:

$$\widetilde{\mathcal{Y}} = \mathcal{Y} \cap \{(a+u)^\top y \le h, \ \forall u \in U\}$$

$$\widetilde{\mathcal{Y}}(\alpha) = \mathcal{Y} \cap \{a(\alpha)^\top y \le h\}$$

Theorem

$$\widetilde{\mathcal{Y}} = \bigcup_{\alpha \in D^*} \widetilde{\mathcal{Y}}(\alpha)$$

Extension II: constraint uncertainty

$$\begin{aligned} \min \quad & c^\top y \\ \text{s.t.} \quad & (a+u)^\top y \le h, \quad \forall u \in U \\ & y \in \mathcal{Y} \end{aligned} \tag{1}$$

Define:

$$\widetilde{\mathcal{Y}} = \mathcal{Y} \cap \{(a+u)^\top y \le h, \ \forall u \in U\}$$

$$\widetilde{\mathcal{Y}}(\alpha) = \mathcal{Y} \cap \{a(\alpha)^\top y \le h\}$$

Theorem

$$\widetilde{\mathcal{Y}} = \bigcup_{\alpha \in D^*} \widetilde{\mathcal{Y}}(\alpha)$$

Corollary

Solving (1) amounts to solve $O(n^s)$ problems of the form

$$\begin{aligned} \min \quad & c^\top y \\ s.t. \quad & \bar{a}^\top y \le h, \quad \forall u \in U \\ & y \in \mathcal{Y} \end{aligned}$$

Extension II: constraint uncertainty - D.-W. reformulation

$$\mathcal{Y} = \overbrace{\mathcal{Y}^M}^{master} \cap \overbrace{\mathcal{Y}^P}^{pricing}$$

Extension II: constraint uncertainty - D.-W. reformulation

$$\mathcal{Y} = \overbrace{\mathcal{Y}^M}^{master} \cap \overbrace{\mathcal{Y}^P}^{pricing}$$

$$\widetilde{\mathcal{Y}}^P = \mathcal{Y}^P \cap \{(a+u)^\top y \leq h,\ \forall u \in U\}$$

Master:

$$\begin{aligned} \min \quad & c^\top \left(\sum_s \lambda_s x_s \right) \\ \text{s.t.} \quad & \sum_s \lambda_s x_s \in \mathcal{Y}^M \\ & \lambda \in \Lambda \end{aligned}$$

Pricing: $\min \left\{ c^\top y \mid x \in \widetilde{\mathcal{Y}}^P \right\}$

Extension II: constraint uncertainty - D.-W. reformulation

$$\mathcal{Y} = \overbrace{\mathcal{Y}^M}^{master} \cap \overbrace{\mathcal{Y}^P}^{pricing}$$

$$\widetilde{\mathcal{Y}}^P = \mathcal{Y}^P \cap \{(a+u)^\top y \le h,\ \forall u \in U\}$$

Master:

$$\begin{aligned} \min \quad & c^\top \left(\sum_s \lambda_s x_s \right) \\ \text{s.t.} \quad & \sum_s \lambda_s x_s \in \mathcal{Y}^M \\ & \lambda \in \Lambda \end{aligned}$$

Pricing: $\min \left\{ c^\top y \mid x \in \widetilde{\mathcal{Y}}^P \right\}$

$$\widetilde{\mathcal{Y}}^P(\alpha) = \mathcal{Y}^P \cap \{a(\alpha)^\top y \le h\}$$

Master:

$$\begin{aligned} \min \quad & c^\top \left(\sum_{\alpha \in D^*} \sum_s \lambda_s(\alpha) x_s \right) \\ \text{s.t.} \quad & \sum_{\alpha \in D^*} \sum_s \lambda_s(\alpha) x_s \in \mathcal{Y}^M \\ & \sum_{\alpha \in D^*} \lambda(\alpha) \in \Lambda \end{aligned}$$

Pricing: $\min \left\{ c^\top y \mid x \in \widetilde{\mathcal{Y}}^P(\alpha) \right\}$

See Pessoa et al. [2021] for pre-processing techniques reducing the number of $\alpha \in D^*$

Table of Contents

Step by step definition of the problem

Note: for the sake of simplicity, we consider

$$\Delta = \left\{ \delta \in \mathbb{R}^n \mid \sum_{i \in [n]} \delta_i \leq 1, 0 \leq \delta \leq 1 \right\}$$

See Omer et al. [2024] for the generalization to U.

Decision dependent information discovery (DDID)

Before choosing y, one may ask for q deviations to be revealed:

- worst-case realization of those q deviations is considered,
- y is chosen after those q observations.

$\Rightarrow$DDID **seeks for the optimal choice of the q observations.**

Step by step definition of the problem

Note: for the sake of simplicity, we consider

$$\Delta = \left\{ \delta \in \mathbb{R}^n \mid \sum_{i \in [n]} \delta_i \leq 1, 0 \leq \delta \leq 1 \right\}$$

See Omer et al. [2024] for the generalization to U.

Decision dependent information discovery (DDID)

Before choosing y, one may ask for q deviations to be revealed:

- worst-case realization of those q deviations is considered,
- y is chosen after those q observations.

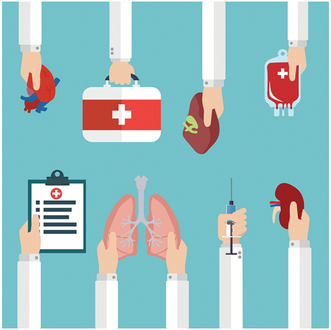
$\Rightarrow$DDID **seeks for the optimal choice of the q observations.**

$$z^{\mathrm{DDID}} = \min_{w \in \mathcal{W}} \max_{\bar{\delta} \in \Delta} \min_{y \in \mathcal{Y}} \max_{\delta \in \Delta(w, \bar{\delta})} \sum_{i \in [n]} (c_i + d_i \delta_i) y_i, \qquad (\mathrm{DDID})$$

- $\mathcal{W} = \{ w \in \{0,1\}^n \mid \sum_i w_i = q \}$,
- $\Delta(w, \bar{\delta}) = \left\{ \delta \in \Delta \mid w \circ \delta = w \circ \bar{\delta} \right\}$: deviations cannot change once observed.

Applications

Any planning process were one may investigate the parameters before taking the actual decision:



organ transplant (e.g. kidney exchange)
$\mathcal{Y} = \{\text{short cycles}, \text{short paths}\}$
Carvalho et al. [2021]

underground works
$\mathcal{Y} = \{\text{trees}, \ldots\}$
Focke et al. [2020]

Many more applications in Vayanos et al. [2020]

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i \boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}.$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}$.

- How to solve the following ?

$$\max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i$$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}.$

- How to solve the following ?

$$\max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i$$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}$.

- How to solve the following ?

$$\max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i$$

- And the following ?

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i \boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}.$

- How to solve the following ?

$$\max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i \boldsymbol{\delta}_i)\boldsymbol{y}_i$$

- And the following ?

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\delta})} \sum_{i\in[n]} (c_i + d_i \boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}$.

- How to solve the following ?

$$\max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i$$

- And the following ?

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\delta})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Theorem ([Bertsimas and Sim, 2003])

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i = \min_{\ell\in[n]} \Gamma d^{\ell} + \min_{\boldsymbol{y}\in\mathcal{Y}} \sum_{i\in[n]} c_i^{\ell}\boldsymbol{y}_i,$$

where d^ℓ and c^ℓ, $\ell \in [n]$, follow simple formulas.

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}.$

- So the following is easy:

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}.$

- So the following is easy:

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

- What about the following ?

$$\max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Robust combinatorial optimization

Let's decompose

$$\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

Remember: $\Delta = \left\{\delta \in [0,1]^n \mid \sum_{i\in[n]} \delta_i \leq \Gamma\right\}$.

- So the following is easy:

$$\min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

- What about the following ?

$$\max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i.$$

$\Rightarrow$ This is our main contribution. We study

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}}\in\Delta} \min_{\boldsymbol{y}\in\mathcal{Y}} \max_{\boldsymbol{\delta}\in\Delta(\boldsymbol{w},\bar{\boldsymbol{\delta}})} \sum_{i\in[n]} (c_i + d_i\boldsymbol{\delta}_i)\boldsymbol{y}_i,$$

under two assumptions:

$$\mathcal{Y} \subseteq \{0,1\}^n \qquad \text{and} \qquad \text{conv}(\mathcal{Y}) = \mathcal{P}$$

Linear formulation of the adversary problem I

For a given choice of observations $\boldsymbol{w} \in \mathcal{W}$, the (outer) adversary problem chooses the revealed deviations $\bar{\boldsymbol{\delta}}$:

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \Delta(\boldsymbol{w}, \bar{\boldsymbol{\delta}})} \sum_{i \in [n]} (c_i + d_i \delta_i) \boldsymbol{y}_i,$$

Linear formulation of the adversary problem I

For a given choice of observations $\boldsymbol{w} \in \mathcal{W}$, the (outer) adversary problem chooses the revealed deviations $\bar{\boldsymbol{\delta}}$:

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \Delta(\boldsymbol{w}, \bar{\boldsymbol{\delta}})} \sum_{i \in [n]} (c_i + d_i \boldsymbol{\delta}_i) \boldsymbol{y}_i,$$

Let us "simplify" $\Delta(\boldsymbol{w}, \bar{\boldsymbol{\delta}})$ to $\bar{\Delta}$:

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \bar{\Delta}} \sum_{i \in [n]} (\bar{c}_i + \bar{d}_i \boldsymbol{\delta}_i) \boldsymbol{y}_i,$$

where:

- $\bar{c}_i$ and $\bar{\Gamma}$ are affine functions of $\bar{\boldsymbol{\delta}}$
- $\bar{\Delta} = \left\{ \boldsymbol{\delta} \in [0,1]^n \mid \sum_i \boldsymbol{\delta}_i \leq \bar{\Gamma} \right\}$
- $\bar{d}$ is independent of $\bar{\boldsymbol{\delta}}$

Linear formulation of the adversary problem I

For a given choice of observations $\boldsymbol{w} \in \mathcal{W}$, the (outer) adversary problem chooses the revealed deviations $\bar{\boldsymbol{\delta}}$:

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \Delta(\boldsymbol{w}, \bar{\boldsymbol{\delta}})} \sum_{i \in [n]} (c_i + d_i \boldsymbol{\delta}_i) \boldsymbol{y}_i,$$

Let us "simplify" $\Delta(\boldsymbol{w}, \bar{\boldsymbol{\delta}})$ to $\bar{\Delta}$:

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \bar{\Delta}} \sum_{i \in [n]} (\bar{c}_i + \bar{d}_i \boldsymbol{\delta}_i) \boldsymbol{y}_i,$$

where:

- $\bar{c}_i$ and $\bar{\Gamma}$ are affine functions of $\bar{\boldsymbol{\delta}}$
- $\bar{\Delta} = \left\{ \boldsymbol{\delta} \in [0,1]^n \mid \sum_i \boldsymbol{\delta}_i \leq \bar{\Gamma} \right\}$
- $\bar{d}$ is independent of $\bar{\boldsymbol{\delta}}$

1. Epigraphic formulation:

$$\begin{aligned} \Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \quad & \boldsymbol{\omega} \\ \text{s.t.} \quad & \boldsymbol{\omega} \leq \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \bar{\Delta}} \sum_{i \in [n]} (\bar{c}_i + \bar{d}_i \boldsymbol{\delta}_i) \boldsymbol{y}_i \end{aligned}$$

Linear formulation of the adversary problem II

1. **Epigraphic formulation:**

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \leq \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \bar{\Delta}} \sum_{i \in [n]} (\bar{c}_i + \bar{d}_i \boldsymbol{\delta}_i) \boldsymbol{y}_i$$

2. **B&S theorem:** rhs is equivalent to solving $n+1$ independent problems.

$$\max_{\bar{\boldsymbol{\delta}} \in \Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \leq \min_{\ell \in [n]_0} \bar{\Gamma} \bar{d}^{\ell} + \min_{\boldsymbol{y} \in \mathcal{Y}} \sum_i \bar{c}_i^{\ell} \boldsymbol{y}_i,$$

Linear formulation of the adversary problem II

1. Epigraphic formulation:

$$\Phi(\boldsymbol{w}) = \max_{\bar{\boldsymbol{\delta}} \in \Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \leq \min_{\boldsymbol{y} \in \mathcal{Y}} \max_{\boldsymbol{\delta} \in \bar{\Delta}} \sum_{i \in [n]} (\bar{c}_i + \bar{d}_i \boldsymbol{\delta}_i) \boldsymbol{y}_i$$

2. B&S theorem: rhs is equivalent to solving $n + 1$ independent problems.

$$\max_{\bar{\boldsymbol{\delta}} \in \Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \leq \min_{\ell \in [n]_0} \bar{\Gamma} \bar{d}^{\ell} + \min_{\boldsymbol{y} \in \mathcal{Y}} \sum_i \bar{c}_i^{\ell} \boldsymbol{y}_i,$$

$\Leftrightarrow$

$$\max_{\bar{\boldsymbol{\delta}} \in \Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \leq \bar{\Gamma} \bar{d}^{\ell} + \min_{\boldsymbol{y} \in \mathcal{Y}} \sum_i \bar{c}_i^{\ell} \boldsymbol{y}_i, \quad \ell \in [n]_0$$

Linear formulation of the adversary problem III

2. **B&S theorem**: rhs is equivalent to solving $n+1$ independent problems.

$$\begin{aligned} \max_{\bar{\delta}\in\Delta} \quad & \boldsymbol{\omega} \\ \text{s.t.} \quad & \boldsymbol{\omega} \leq \bar{\Gamma}\bar{d}^{\ell} + \min_{\boldsymbol{y}\in\mathcal{Y}} \sum_i \bar{c}_i^{\ell} \boldsymbol{y}_i, \quad \ell \in [n]_0 \end{aligned}$$

Linear formulation of the adversary problem III

2. **B&S theorem**: rhs is equivalent to solving $n+1$ independent problems.

$$\max_{\bar{\delta}\in\Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \le \bar{\Gamma}\bar{d}^{\ell} + \min_{\boldsymbol{y}\in\mathcal{Y}} \sum_i \bar{c}_i^{\ell} \boldsymbol{y}_i, \quad \ell \in [n]_0$$

3. **Dualization**: use that $\text{conv}(\mathcal{Y}) = \mathcal{P}$ to dualize the minimization.

$$\Phi(\boldsymbol{w}) = \max_{\bar{\delta}\in\Delta} \quad \boldsymbol{\omega}$$
$$\text{s.t.} \quad \boldsymbol{\omega} \le \bar{\Gamma}\bar{d}^{\ell} + \max_{\pi,\lambda} b^T \boldsymbol{\lambda}_{\ell} - \sum_{i\in[n]} \boldsymbol{\pi}_{\ell,i}, \ \ell \in [n]_0$$
$$\text{s.t.} (B_{\cdot,i})^T \boldsymbol{\lambda}_{\ell} - \boldsymbol{\pi}_{\ell,i} \le \bar{c}_i^{\ell}, \forall \ell \in [n]_0, \ \forall i$$
$$\boldsymbol{\lambda}_{\ell}, \boldsymbol{\pi}_{\ell} \ge 0, \ \forall \ell \in [n]_0$$

Linear formulation of the adversary problem III

2. **B&S theorem**: rhs is equivalent to solving $n+1$ independent problems.

$$\begin{aligned} \max_{\bar{\delta}\in\Delta} \quad & \boldsymbol{\omega} \\ \text{s.t.} \quad & \boldsymbol{\omega} \leq \bar{\Gamma}\bar{d}^{\ell} + \min_{\boldsymbol{y}\in\mathcal{Y}} \sum_i \bar{c}_i^{\ell} \boldsymbol{y}_i, \quad \ell \in [n]_0 \end{aligned}$$

3. **Dualization**: use that $\text{conv}(\mathcal{Y}) = \mathcal{P}$ to dualize the minimization.

$$\begin{aligned} \Phi(\boldsymbol{w}) = \max_{\bar{\delta}\in\Delta} \quad & \boldsymbol{\omega} \\ \text{s.t.} \quad & \boldsymbol{\omega} \leq \bar{\Gamma}\bar{d}^{\ell} + \max_{\pi,\lambda} b^T \boldsymbol{\lambda}_{\ell} - \sum_{i\in[n]} \boldsymbol{\pi}_{\ell,i}, \; \ell \in [n]_0 \\ & \qquad \text{s.t.} (B_{\cdot,i})^T \boldsymbol{\lambda}_{\ell} - \boldsymbol{\pi}_{\ell,i} \leq \bar{c}_i^{\ell}, \forall \ell \in [n]_0, \; \forall i \\ & \qquad \boldsymbol{\lambda}_{\ell}, \boldsymbol{\pi}_{\ell} \geq 0, \; \forall \ell \in [n]_0 \end{aligned}$$

$$\Leftrightarrow \quad \begin{aligned} \Phi(\boldsymbol{w}) = \max_{\bar{\delta}\in\Delta} \quad & \boldsymbol{\omega} \\ \text{s.t.} \quad & \boldsymbol{\omega} \leq \bar{\Gamma}\bar{d}^{\ell} + b^T \boldsymbol{\lambda}_{\ell} - \sum_{i\in[n]} \boldsymbol{\pi}_{\ell,i}, \; \ell \in [n]_0 \\ & (B_{\cdot,i})^T \boldsymbol{\lambda}_{\ell} - \boldsymbol{\pi}_{\ell,i} \leq \bar{c}_i^{\ell}, \forall \ell \in [n]_0, \; \forall i \\ & \boldsymbol{\lambda}_{\ell}, \boldsymbol{\pi}_{\ell} \geq 0, \; \forall \ell \in [n]_0 \end{aligned}$$

Linear formulation of the adversary problem IV

We have reformulated $\min_{\boldsymbol{w}\in\mathcal{W}} \Phi(\boldsymbol{w})$ as:

$$
\begin{aligned}
\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \quad & \boldsymbol{\omega} \\
\text{s.t.} \quad & \boldsymbol{\omega} \leq \bar{\Gamma}\bar{d}^{\ell} + b^T \boldsymbol{\lambda}_{\ell} - \sum_{i\in[n]} \boldsymbol{\pi}_{\ell,i},\ \ell \in [n]_0 \\
& (B_{\cdot,i})^T \boldsymbol{\lambda}_{\ell} - \boldsymbol{\pi}_{\ell,i} \leq \bar{c}_i^{\ell}, \forall \ell \in [n]_0,\ \forall i \\
& \boldsymbol{\lambda}_{\ell}, \boldsymbol{\pi}_{\ell} \geq 0,\ \forall \ell \in [n]_0
\end{aligned}
$$

Linear formulation of the adversary problem IV

We have reformulated $\min_{\boldsymbol{w}\in\mathcal{W}} \Phi(\boldsymbol{w})$ as:

$$
\begin{aligned}
\min_{\boldsymbol{w}\in\mathcal{W}} \max_{\bar{\boldsymbol{\delta}}\in\Delta} \quad & \boldsymbol{\omega} \\
\text{s.t.} \quad & \boldsymbol{\omega} \le \bar{\Gamma}\bar{d}^{\ell} + b^T\boldsymbol{\lambda}_\ell - \sum_{i\in[n]} \boldsymbol{\pi}_{\ell,i},\ \ell\in[n]_0 \\
& (B_{\cdot,i})^T\boldsymbol{\lambda}_\ell - \boldsymbol{\pi}_{\ell,i} \le \bar{c}_i^{\ell}, \forall \ell\in[n]_0,\ \forall i \\
& \boldsymbol{\lambda}_\ell, \boldsymbol{\pi}_\ell \ge 0,\ \forall \ell\in[n]_0
\end{aligned}
$$

The formulation of the DDID is finally obtained by

- dualization of inner maximization problem,
- linearization of all the hidden products with the $\boldsymbol{w}$.

Main result: Compact MILP formulation for DDID

$$
\begin{aligned}
\min \quad & \sum_{\ell \in [n]_0} \left(r \alpha_\ell \boldsymbol{u}_\ell + \sum_{i \in [n]} c_i \boldsymbol{y}_{\ell,i} + \sum_{i \in [n]} \beta_{\ell,i} \boldsymbol{y}^0_{\ell,i} \right) + \sum_{i \in [n]} d_i \boldsymbol{\sigma}_i \\
\text{s.t.} \quad & \sum_{\ell \in [n]_0} \boldsymbol{u}_\ell = 1 \\
& a_i \boldsymbol{\sigma}_i \geq -a_i \sum_{\ell \in [n]_0} \alpha_\ell \boldsymbol{u}_\ell + \sum_{\ell \in [n]_0} \boldsymbol{y}_{\ell,i} - (1 - \boldsymbol{w}_i), & \forall i \in [n] \\
& B \boldsymbol{y}_\ell \geq \boldsymbol{u}_\ell b, & \forall \ell \in [n]_0 \\
& \boldsymbol{y}_{\ell,i} \leq \boldsymbol{u}_\ell, & \forall \ell \in [n]_0, i \in [n] \\
& \boldsymbol{y}^0_{\ell,i} \geq \boldsymbol{y}_{\ell,i} - \boldsymbol{w}_i, & \forall \ell \in [n]_0, i \in [n] \\
& \boldsymbol{w} \in \mathcal{W}, \boldsymbol{u}, \boldsymbol{y}, \boldsymbol{y}^0, \boldsymbol{\sigma} \geq 0.
\end{aligned}
$$

Selection problem

$\mathcal{Y} = \{\text{select } p = n/10 \text{ items among } n \text{ to minimize total cost}\}$

Vayanos et al. [2020]: K-adaptability heuristic: decision-maker must chooses K strategies $y^1, \ldots, y^K$ among which one is chosen after revealing $\bar{\delta}$.

- our own implementation
- budget uncertainty ($L = 1$)
- K-adaptability reformulations grow linearly in L and K

n	K	T	gap
10	2	7	10
	3	24	11
15	2	46	7
	3	3275	9

(a) K-adaptability MILP reformulation.

n	T	gap
10	6	0.14
20	10	0.26
30	26	0.39
40	49	0.16
50	160	0.22

(b) Our exact MILP reformulation.

Note: times are in centiseconds

Orienteering problem

$\mathcal{Y} = \{$elementary paths with maximum time constraint$\}$

- CB : exact algorithm: column and constraint generation to solve the adversary problem embedded in combinatorial Benders' cut algorithm [Paradiso et al., 2022].
- BP : our branch-and-price algorithm

instance	Opt		Solved at root
	CB	BP	BP
TS2N10	**100%**	**100%**	89%
TS1N15	**100%**	**100%**	90%
TS3N16	**100%**	**100%**	83%
TS2N19	76%	**100%**	73%
TS1N30	41%	96%	61%
TS3N31	37%	90%	80%

Conclusion

$$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}, \qquad L \text{ is the number of rows of } A$$

Take-away messages

- MIN-MAX: not harder than nominal problem if L is constant
- DDID: If $\text{conv}(\mathcal{Y}) = \mathcal{P}$ is a compact polyhedron and L is constant
    - Computing Φ is a compact LP
    - Solving DDID is a compact MILP

Conclusion

$$U = \{u \in \mathbb{R}^n \mid Au \leq b, 0 \leq u \leq d\}, \qquad L \text{ is the number of rows of } A$$

Take-away messages

- MIN-MAX: not harder than nominal problem if L is constant
- DDID: If $\text{conv}(\mathcal{Y}) = \mathcal{P}$ is a compact polyhedron and L is constant
  - Computing Φ is a compact LP
  - Solving DDID is a compact MILP

Perspectives

- Complexity: it is not known yet whether DDID can be NP-Hard when the nominal problem is polynomial $\Rightarrow$ thesis of Xiaoyu Chen
- Numerically: decomposition is promising when $\text{conv}(\mathcal{Y}) \neq \mathcal{P}$

Table of Contents

What is open science?

Assuming our research is useful:

- Make the results of science available to everybody
- Help disseminating science to society

Open data ☺

Benchmarks (MIPlib, QPlib, netlib...), real data from industrial applications

Open code ☹

☹☹ Most published algorithms are not reproducible!

☺ Some journals enforce reproducibility (IJOC, MPC, OJMO, OR)

Open publications ☹

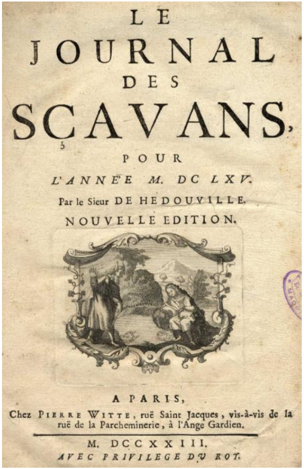
☺ Optimization online, arxiv

☹ Many papers not available online

☹☹ Ever-increasing publication fees (cost $\sim$ 100M€ annually in France)

The beginnings ...

Back in the days, publishing was expensive!



LE JOURNAL DES SCAVANS,

POUR L'ANNÉE M. DC LXV.

Par le Sieur DE HEDOUVILLE.

NOUVELLE EDITION.

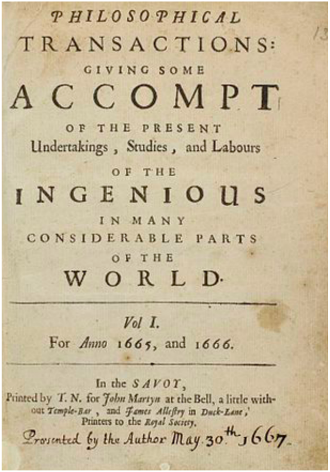
A PARIS,

Chez PIERRE WITTE, ruë Saint Jacques, vis-à-vis de la ruë de la Parcheminerie, à l'Ange Gardien.

M. DCC XXIII.

AVEC PRIVILEGE DU ROY.

Paris, January 5, *1665*



PHILOSOPHICAL TRANSACTIONS:

GIVING SOME ACCOMPT OF THE PRESENT Undertakings, Studies, and Labours OF THE INGENIOUS IN MANY CONSIDERABLE PARTS OF THE WORLD.

Vol I.

For *Anno* 1665, and 1666.

In the SAVOY,

Printed by T. N. for *John Martyn* at the Bell, a little without *Temple-Bar*, and *James Allestry* in *Duck-Lane*, Printers to the *Royal Society*.

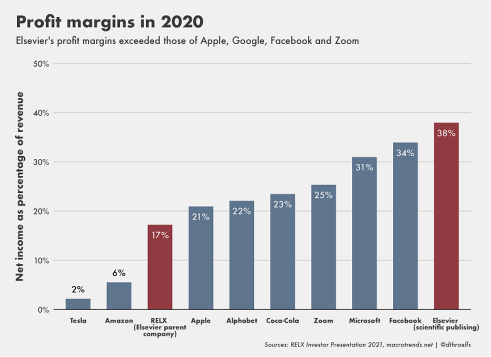
Presented by the Author May 30th 1667

London, March 6, *1665*

Internet and LaTeX

- The advent of electronic publishing and dissemination via the Web and the use of LaTeXreduced significantly the operating costs of publishers
- Led to nearly **open** and **free** publications?
- In fact no:

source: PhD comics



Note: Publishers ask at least 2000€ for Open Access. Its real cost varies between 3€and 800€ (when heavy typesetting is needed).

The uprising

Sir Tim Gowers,
Fields Medal 1998

17062 Researchers Taking a Stand. ***See the list***

Academics have protested against Elsevier's business practices for years with little effect. These are some of their objections:

1. They charge exorbitantly high prices for subscriptions to individual journals.
2. In the light of these high prices, the only realistic option for many libraries is to agree to buy very large "bundles", which will include many journals that those libraries do not actually want. Elsevier thus makes huge profits by exploiting the fact that some of their journals are essential.
3. They support measures such as SOPA, PIPA and the ~~Research Works Act~~, that aim to restrict the free exchange of information.

http://www.thecostofknowledge.com/

Taken from the presentation of Marie Farge

Examples of fair journals/conferences

Machine learning, Artificial intelligence

- JAIR
- Journal of Machine Learning Research (JMLR)
- Transactions on Machine Learning Research (TMLR)

Graph theory, algorithmics

- Advances in Combinatorics
- TheoretiCS
- Theory of Computing
- Innovations in Graph Theory

And many conferences: LIPICs (mainly theoretical CS), and more . . .

Examples of fair journals/conferences

Machine learning, Artificial intelligence

- JAIR
- Journal of Machine Learning Research (JMLR)
- Transactions on Machine Learning Research (TMLR)

Graph theory, algorithmics

- Advances in Combinatorics
- TheoretiCS
- Theory of Computing
- Innovations in Graph Theory

And many conferences: LIPICs (mainly theoretical CS), and more . . .

Mathematical optimization

- Open Journal of Mathematical Optimization (OJMO)
  - Area editors: G. Bayraksan, R. Luke, J. Malick, S. Pokutta
  - indexed in most databases, Q2 at scimago
  - fast track for short papers
  - enforces reproducibility!

References I

Dimitris Bertsimas and Melvyn Sim. Robust discrete optimization and network flows. *Math. Program.*, 98(1-3):49–71, 2003. doi: 10.1007/s10107-003-0396-4. URL https://doi.org/10.1007/s10107-003-0396-4.

Margarida Carvalho, Xenia Klimentova, Kristiaan Glorie, Ana Viana, and Miguel Constantino. Robust models for the kidney exchange problem. *INFORMS J. Comput.*, 33(3):861–881, 2021. doi: 10.1287/IJOC.2020.0986. URL https://doi.org/10.1287/ijoc.2020.0986.

Jacob Focke, Nicole Megow, and Julie Meißner. Minimum spanning tree under explorable uncertainty in theory and experiments. *ACM J. Exp. Algorithmics*, 25: 1–20, 2020. doi: 10.1145/3422371. URL https://doi.org/10.1145/3422371.

Jérémy Omer, Michael Poss, and Maxime Rougier. Combinatorial Robust Optimization with Decision-Dependent Information Discovery and Polyhedral Uncertainty. *Open Journal of Mathematical Optimization*, 5:5, 2024. doi: 10.5802/ojmo.33. URL https://ojmo.centre-mersenne.org/articles/10.5802/ojmo.33/.

Rosario Paradiso, Angelos Georghiou, Said Dabia, and Denise Tönissen. Exact and approximate schemes for robust optimization problems with decision dependent information discovery. *arXiv preprint arXiv:2208.04115*, 2022.

Artur Alves Pessoa, Michael Poss, Ruslan Sadykov, and François Vanderbeck. Branch-cut-and-price for the robust capacitated vehicle routing problem with knapsack uncertainty. *Oper. Res.*, 69(3):739–754, 2021. doi: 10.1287/opre.2020.2035. URL https://doi.org/10.1287/opre.2020.2035.

Phebe Vayanos, Angelos Georghiou, and Han Yu. Robust optimization with decision-dependent information discovery. *arXiv preprint arXiv:2004.08490*, 2020.