IA meeting 14/12/2020

The price of robustness

Bertsimas, Dimitris, and Melvyn Sim. "The price of robustness." Operations research 52.1 (2004): 35-53.

The price of robustness

Context

Quote from the case study by Ben-Tal and Nemirovski (2000):

*« In real-world applications of Linear Programming, one cannot ignore the possibility that **a small uncertainty** in the data can make the usual optimal solution completely **meaningless** from a **practical** viewpoint. »*

This observation raises the natural question of designing solution approaches that are **immune to data uncertainty**; that is, they are « **robust** ».

This paper designs a **new robust approach** that adresses the issue of **over-conservatism**.

The price of robustness

Data uncertainty in linear optimization

Linear optimization problem:

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}'\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \leqslant \mathbf{b} \\ & \mathbf{l} \leqslant \mathbf{x} \leqslant \mathbf{u}. \end{aligned}$$

Data uncertainty is in the matrix A.

The coefficients **a_ij** that are subjected to parameter uncertainty takes values according to a **symmetric distribution** with a mean equal to the nominal value a_ij in the interval [a_ij- â_ij, a_ij + â_ij].

Row i -> **J_i** coefficients subject to uncertainty

Gamma_i = parameter to adjust the robustness of the proposed method against the level of conservatism of the solution.

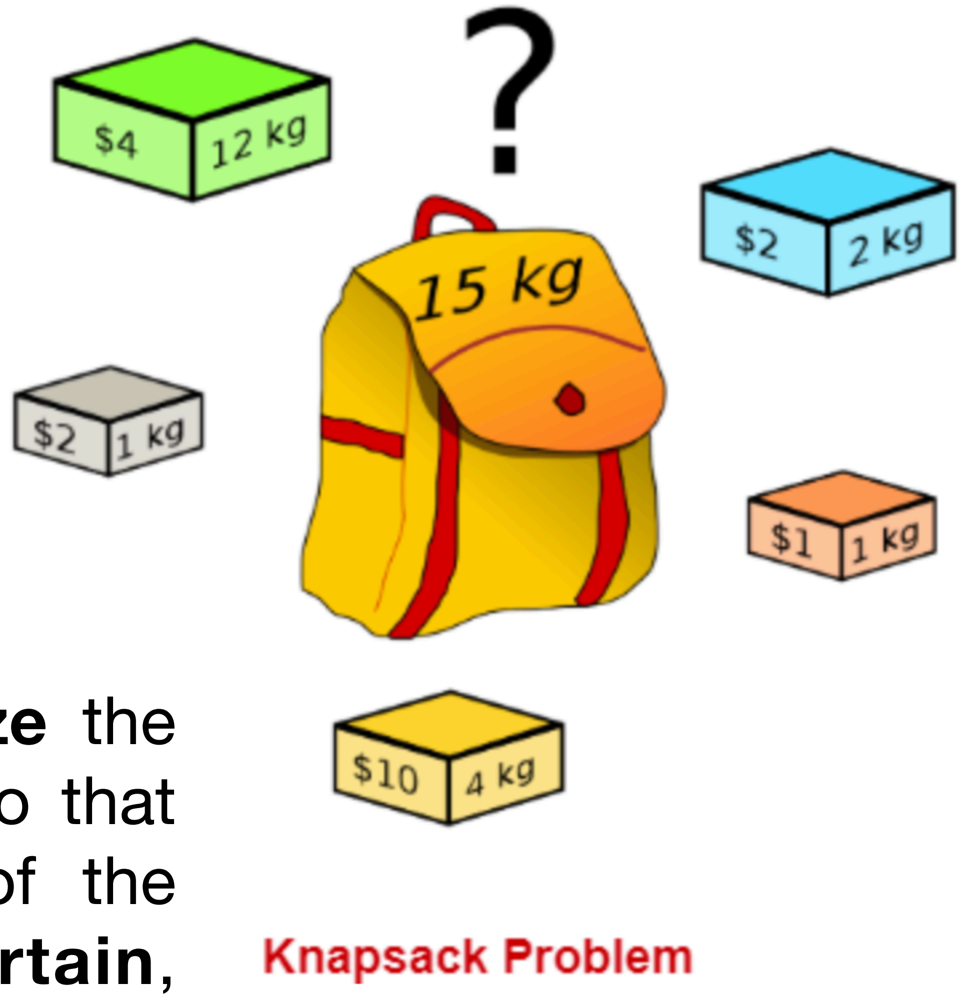
0 <= Gamma_i <= J_i -> **only a subset** of the coefficients will change in order to adversely affect the solution.
The higher Gamma_i, the more robust the solution is. With Gamma_i = J_i -> maximum protection.

The price of robustness

Zero-one knap sack problem (MILP)

MILP:

$$\begin{aligned} \text{maximize} \quad & \sum_{i \in N} c_i x_i \\ \text{subject to} \quad & \sum_{i \in N} w_i x_i \leqslant b \\ & x_i \in \{0, 1\}. \end{aligned}$$



An application of this problem is to **maximize** the total **value of goods** to be loaded on a cargo that has strict weight restrictions. The **weight** of the individual item is assumed to be **uncertain**, independent of other weights, and follows a symmetric distribution.

The price of robustness

Zero-one knap sack problem (MILP)

The zero-one knapsack problem is the following discrete optimization problem:

$$\max_{x_i} \sum_{1 \leq i \leq N} c_i x_i$$
$$\text{s.t.} \quad \sum_{1 \leq j \leq N} \omega_j x_j \leq b$$
$$x_j \in \{0, 1\}.$$

Let **J the set of uncertain parameters ωj**, with **0 ≤ |J| ≤ N**. The weights ωj with j ∈ J are subjected to parameter uncertainty takes values according to a symmetric distribution with a mean equal to the nominal value ωj in the interval **[ωj – ωˆj, ωj + ωˆj]**. The parameter to adjust the robustness of the approach is Γ, with **0 ≤ Γ ≤ |J| ≤ N**.

The price of robustness

Zero-one knap sack problem (MILP)

We assume **Γ takes only integer** values for the sake of simplicity.
Then, the **robust** zero-one knapsack problem is (**NON LINEAR**)

$$
\begin{aligned}
\max_{x_i} \quad & \sum_{1\le i\le N} c_i x_i \\
\text{s.t.} \quad & \sum_{1\le j\le N} \omega_j x_j + \max_{S\subseteq J, |S|=|J|} \left\{ \sum_{j\in S} \hat{\omega}_j x_j \right\} \le b \\
& x_j \in \{0, 1\}.
\end{aligned}
$$

Given a vector $x^\star$ the **protection function** is (worst case path)

$$
\beta(\boldsymbol{x}^\star, \Gamma) = \max_{S\subseteq J, |S|=|J|} \left\{ \sum_{j\in S} \hat{\omega}_j x_j \right\},
$$

The price of robustness

Zero-one knap sack problem (MILP)

$$\beta(\boldsymbol{x}^\star, \Gamma) = \max_{S \subseteq J, |S| = |J|} \Big\{ \sum_{j \in S} \hat{\omega}_j x_j \Big\},$$

and is equal to the following linear optimization problem that provides the **worst case scenario** given J and Γ

Primal

$$\begin{aligned} \max_{z_j} \quad & \sum_{j \in J} \hat{\omega}_j x_j^\star z_j \\ \text{s.t.} \quad & \sum_{j \in J} z_j \leq \Gamma \quad [z] \\ & 0 \leq z_j \leq 1 \quad j \in J \quad [p_j]. \end{aligned}$$

Dual

$$\begin{aligned} \min_{p_j, z} \quad & \sum_{j \in J} p_j + z\Gamma \\ \text{s.t.} \quad & p_j + z \geq \hat{\omega}_j x_j^\star \quad j \in J \\ & p_j \geq 0 \quad j \in J \\ & z \geq 0. \end{aligned}$$

By **strong duality** since the primal problem is feasible and bounded for 0 ≤ Γ ≤ |J|, then the dual problem is also feasible and bounded and their objective values coincide.

The price of robustness

Zero-one knap sack problem (MILP)

Finally, by substitution the robust zero-one knap sack problem is (**MILP**)

$$
\begin{aligned}
\max_{x_i} \quad & \sum_{1 \le i \le N} c_i x_i \\
\text{s.t.} \quad & \sum_{1 \le j \le N} \omega_j x_j + \sum_{j \in J} p_j + z\Gamma \le b \\
& p_j + z \ge \hat{\omega}_j x_j \quad j \in J \\
& p_j \ge 0 \quad j \in J \\
& x_j \in \{0, 1\}.
\end{aligned}
$$

The price of robustness

Zero-one knap sack problem: use case

Goal = maximize the total value of the goods but allow a maximum of **1% chance** of constraint violation.

Size N = **200**

Capacity limit b = 4 000

Nominal weight randomly chosen from the set {20, …, 29} with uncertainty equals to **10% of the nominal weights**.

Cost randomly chosen from the set {16, …, 77}

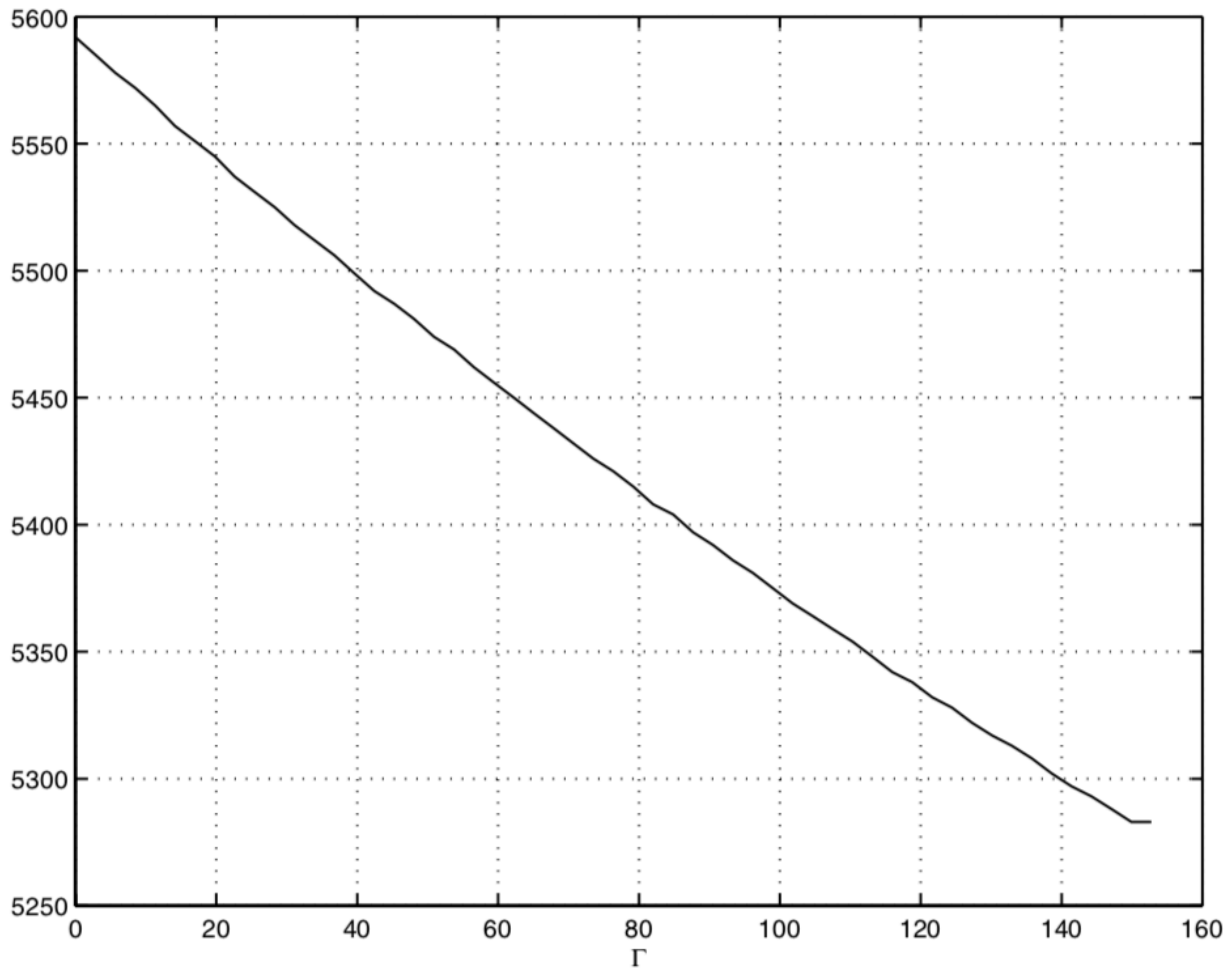
The price of robustness

Zero-one knap sack problem: use case

No protection -> 5 992

Full protection -> 5 283 (5.5% of reduction)

Optimal value of the robust knapsack formulation as a function of Γ.

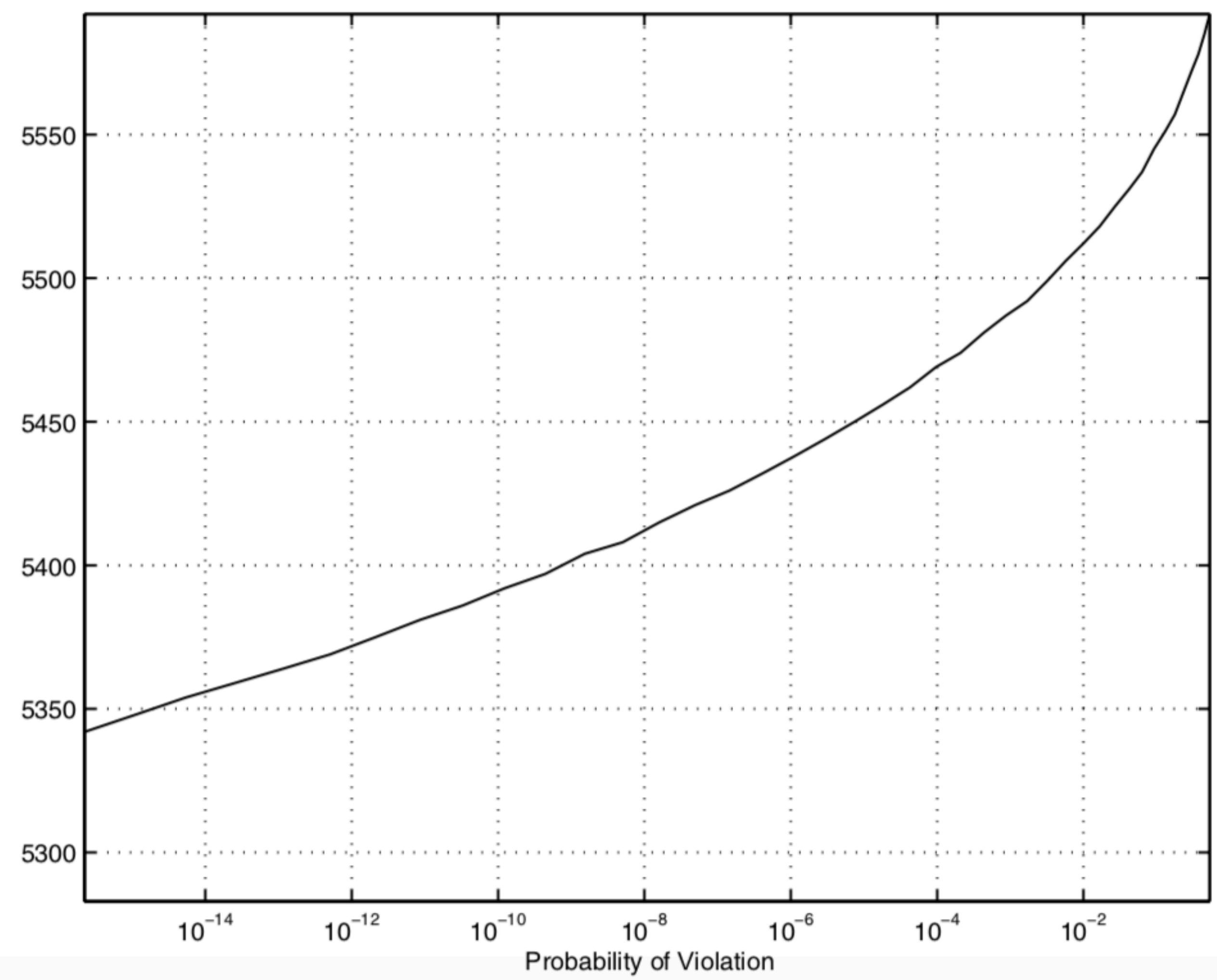


The price of robustness

Zero-one knap sack problem: use case

To have a probability guarantee of at most **0.57%** chance of constraint violation, the objective is reduced by **1.54%** for **Gamma = 37**.

Optimal value of the robust knapsack formulation as a function of the probability bound of constraint violation given in Equation (18).



The price of robustness

Zero-one knap sack problem: use case

Table 2. Results of robust knapsack solutions.

Γ	Probability Bound	Optimal Value	Reduction (%)
2.8	4.49×10^{-1}	5,585	0.13
14.1	1.76×10^{-1}	5,557	0.63
25.5	4.19×10^{-2}	5,531	1.09
36.8	5.71×10^{-3}	5,506	1.54
48.1	4.35×10^{-4}	5,481	1.98
59.4	1.82×10^{-5}	5,456	2.43
70.7	4.13×10^{-7}	5,432	2.86
82.0	5.04×10^{-9}	5,408	3.29
93.3	3.30×10^{-11}	5,386	3.68
104.7	1.16×10^{-13}	5,364	4.08
116.0	2.22×10^{-16}	5,342	4.47

This approach succeeds in **reducing the price of robustness:** it does not heavily penalize the objective function value in order to protect ourselves against constraint violation.