Robust Optimization Seminar

Berk Öztürk

03/05/2024

Introductory Remarks

- About me:
  - Optimizer / Operations Research Analyst at MITRE
    - What I do: optimization, data science, simulation and modeling.
  - PhD, MIT AeroAstro, 2022

    "Global and Robust Optimization for Engineering Design"

    Advisor: Prof. Dimitris Bertsimas.

    TAed Robust Optimization in Spring 2021.

- Please feel free to ask questions during the talk. I will be asking you questions too!

[Paraphrasing mercilessly from lecture notes of MIT's 15.094 RO course, taught Spring 2021 by Dimitris Bertsimas and Dick den Hertog (and TAed by me).]

What to expect from today

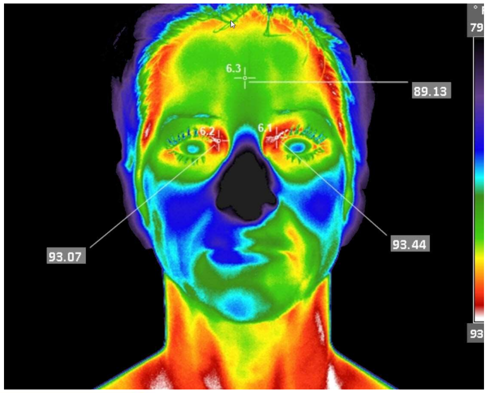
- Theory of robust optimization (RO), which
  - turns stochastic optimization problems into deterministic ones,
  - is general and practical for a range of decision problems,
  - Provides guarantees of constraint satisfaction under uncertainty.
- Some demonstrations of the need for RO, and derivations of key concepts.
- Suggestions about how you can use RO in your own work.
- A real-time demonstration of a facility location problem under uncertainty.

Parametric uncertainty is ubiquitous in decision/design problems.

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x,u) \le 0,\ \forall u \in \mathcal{U},\ i = 1,\ldots,n \end{aligned}$$

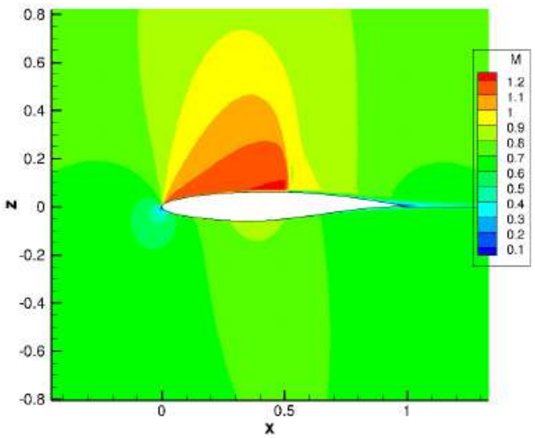
Decision variables (→ x)

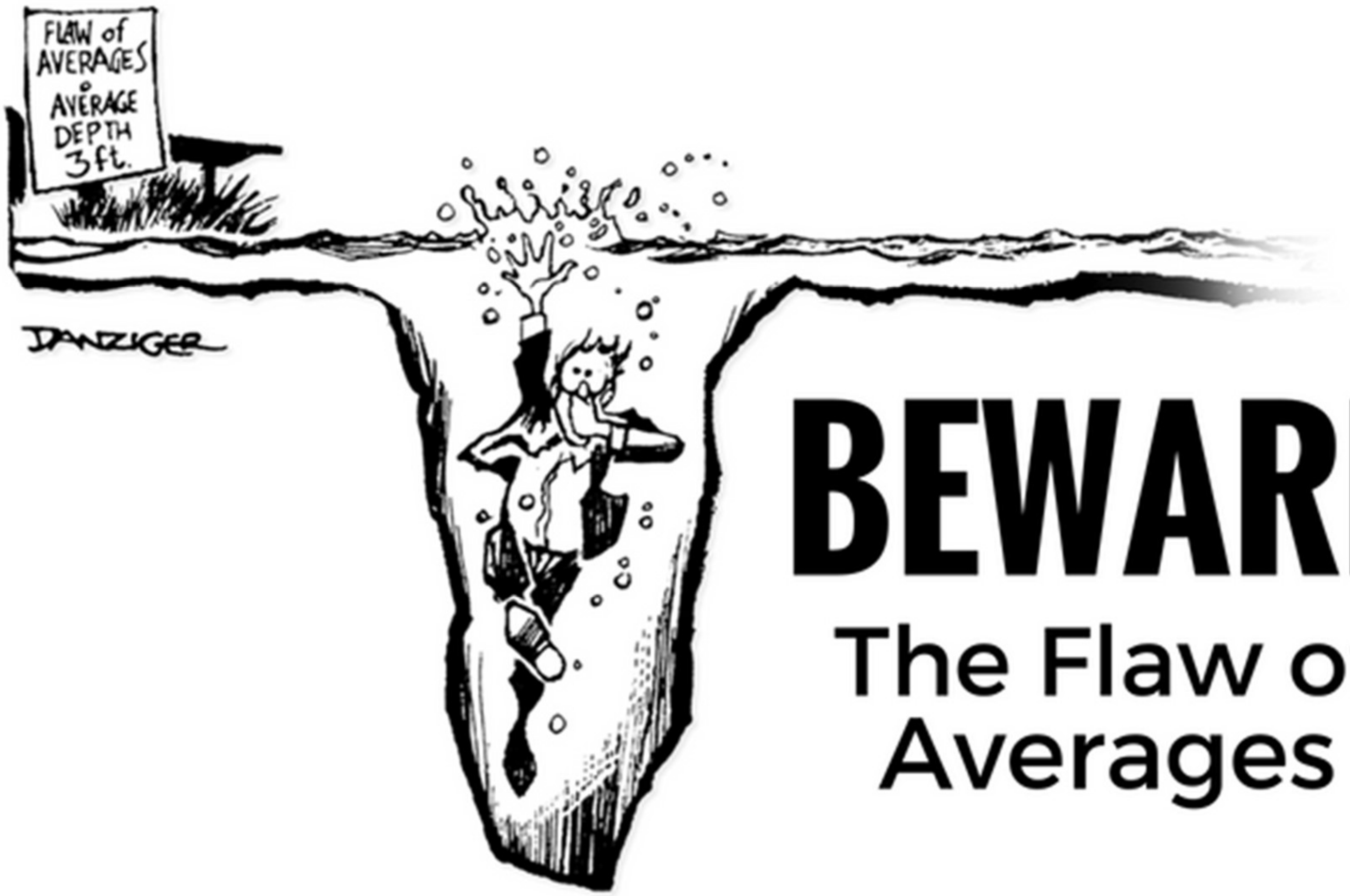
Uncertain parameters (→ u)



Uncertainty arises from:

- Measurement errors,
  - Blood pressure, temperature …
- Estimation/prediction errors,
  - Demand, truncation error …
- Implementation errors,
  - Voltages, engineering tolerances …





BEWARE

The Flaw of Averages

^ “Flaw of Averages”, Sam Savage, 2012.

Here's a scenario* to avoid…

$$
\begin{aligned}
\bar{a}^T x = &-15.79081x_{826} - 8.598819x_{827} - 1.88789x_{828} - 1.362417x_{829} - 1.526049x_{830} \\
&-0.031883x_{849} - 28.725555x_{850} - 10.792065x_{851} - 0.19004x_{852} - 2.757176x_{853} \\
&-12.290832x_{854} + 717.562256x_{855} - 0.057865x_{856} - 3.785417x_{857} - 78.30661x_{858} \\
&-122.163055x_{859} - 6.46609x_{860} - 0.48371x_{861} - 0.615264x_{862} - 1.353783x_{863} \\
&-84.644257x_{864} - 122.459045x_{865} - 43.15593x_{866} - 1.712592x_{870} - 0.401597x_{871} \\
&+x_{880} - 0.96049x_{898} - 0.946049x_{916} \\
&\geq b = 23.387405
\end{aligned}
$$

Suppose: accuracy is 0.1%:

$$
(*) \qquad \left| a_i^{true} - \bar{a}_i \right| \leq 0.001 \left| \bar{a}_i \right| .
$$

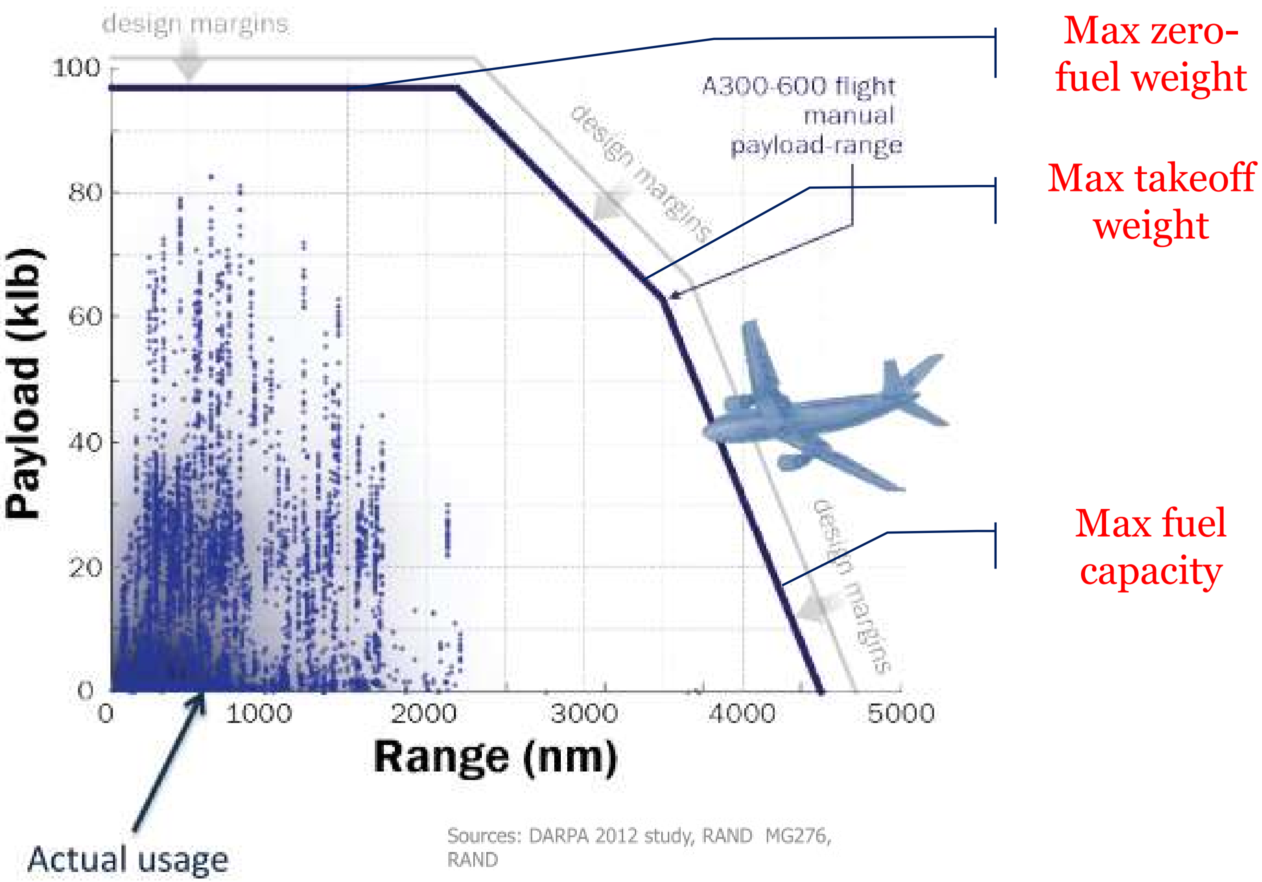
Worst case: the constraint can be violated by as much as 450%:

$$
\min_{a^{true} \text{satisfies } (*)} \left(a^{true}\right)^T \bar{x} - b = -128.8 \approx -4.5b.
$$

Optimization under uncertainty (OuU) is critical to protect against adverse outcomes!

*From 15.094 lecture notes, 2021.

My original motivation: Legacy aerospace design methods do not adequately consider the risk-performance tradeoff.



Sources: DARPA 2012 study, RAND MG276, RAND

There is no such thing as a free lunch.
Conservative margins leave performance on the table.

How about:

- Technological capabilities?
- Manufacturing quality?
- Regulatory environment?

Why don’t we tackle the following problem directly?

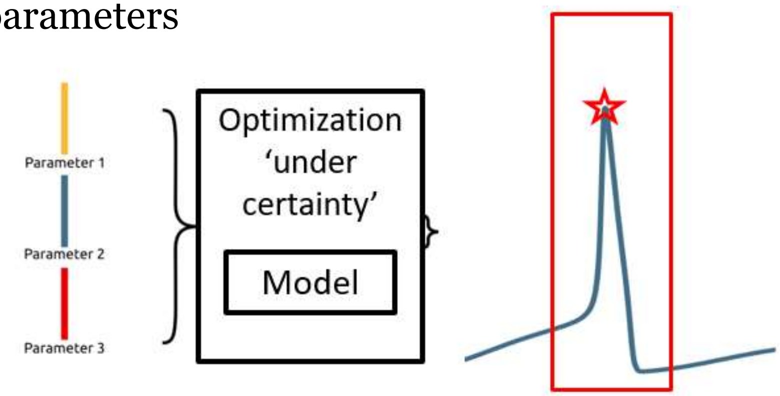
$$\begin{aligned} \min \;& f_0(x) \\ \text{s.t. }\; & f_i(x,u) \leq 0, \; \forall u \in \mathcal{U}, \; i = 1, \ldots, n \end{aligned}$$

Note: there is no uncertainty in the objective function, without loss of generality (wlog).

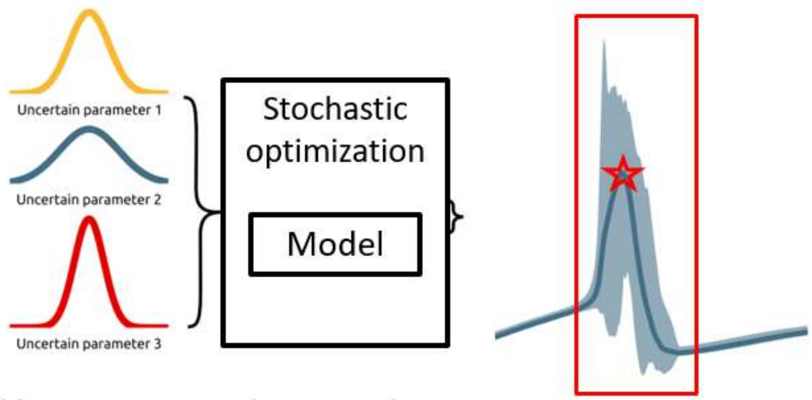
Answer: infinitely-many constraints can be generated with the infinite number of possible realizations of u.

One method to approximate the problem is through stochastic optimization.

Fixed parameters



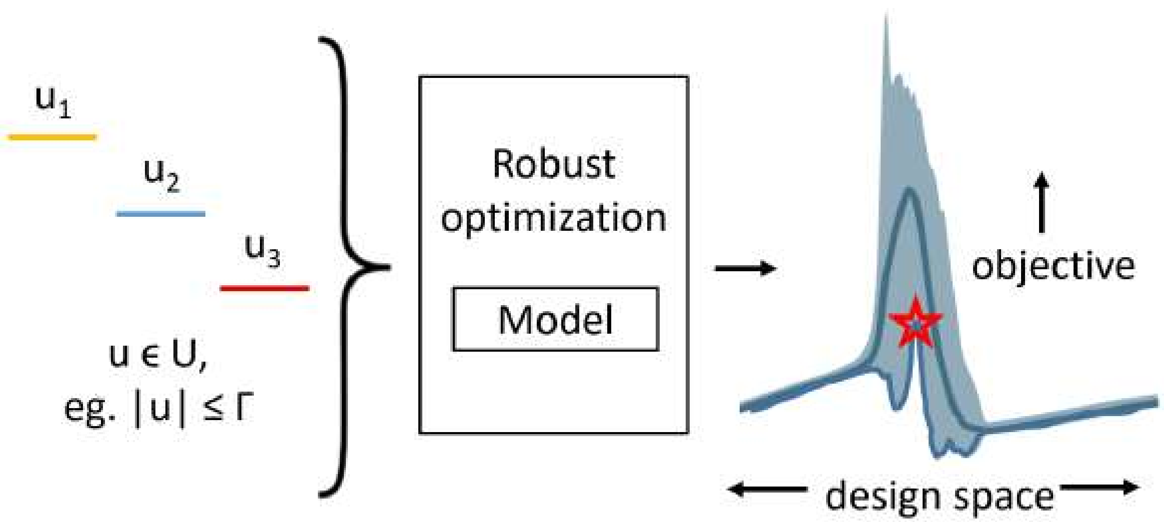
Parameter distributions



- Problem: Even low-D problems with nice distributions cause computational trouble. **The Curse of Dimensionality.**
  - Jargon: Propagating distributions through constraints via convolutions and high-dimensional quadrature is computationally expensive.

RO is an alternative for optimization under uncertainty that is tractable and deterministic.

$$\begin{aligned} &\min \; f_0(x) \\ &\text{s.t.} \; \max_{u \in \mathcal{U}} f_i(x,u) \leq 0, \; i = 1, \ldots, n \end{aligned}$$



RO makes sure all constraints are feasible for all parameter outcomes from an uncertainty set, while minimizing the worst-case objective.

Important notes and intuitions

- Robust optimization problem is robustified *constraint-wise*:
  - Each constraint protects against **all** possible outcomes, allowing different uncertain outcomes for each constraint.

$$\begin{aligned} \min \ & f_0(x) \\ \text{s.t.} \ & \max_{u \in \mathcal{U}} f_i(x, u) \leq 0, \ i = 1, \ldots, n \end{aligned}$$

- Thus, depending on the structure of the uncertainty, RO can be conservative, or not!
- RO makes mild assumptions about what the uncertainty set can be.
- RO is deterministic. I will demonstrate in a second!

Robust (Linear) Optimization Theory

Constraint under uncertainty

$$\underbrace{(a + Pz)}_{\text{Uncertain coefficients}}{}^T \underbrace{x}_{\text{Decision variables}} \leq b, \ \forall z \in \mathcal{Z},$$

can be transformed via the max,

$$\max_{z \in \mathcal{Z}} (a + Pz)^T x \leq b.$$

z is *adversarial* to x, so cannot solve in one shot except through reformulation.

Say our uncertainty comes from a box, i.e. bounded by the inf-norm hypercube… can we find a *robust counterpart?*

$$\max_{z:\|z\|_\infty \le \rho} (\bar{a} + Pz)^T x \le b,$$

$$\bar{a}^T x + \max_{z:\|z\|_\infty \le \rho} (P^T x)^T z \le b,$$

Math identity: Separating

$$\bar{a}^T x + \max_{z:|z_i| \le \rho} \sum_i (P^T x)_i z_i \le b,$$

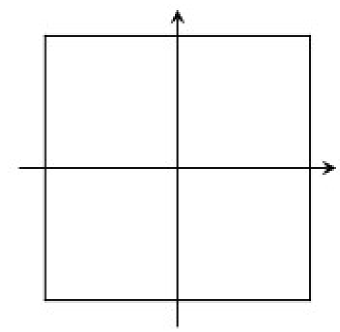
Math identity: Rearranging

$$\bar{a}^T x + \rho \sum_i |(P^T x)_i| \le b,$$

Optimization: In this case intuition.

Math identity: Rewriting as norm

$$\boxed{\bar{a}^T x + \rho \|P^T x\|_1 \le b.}$$



$p = \infty$

Robust counterpart does not contain $z \in Z$!

Practical RLO questions (1/3)

$(a + Pz)^T x \leq b, \forall \{z : ||z||_\infty \leq \rho\}$ became $a^T x + \rho ||P^T x||_1 \leq b$.

Questions:

1. The robust counterpart is what kind of optimization problem?
2. What is the complexity of this optimization problem? (Hint: can you estimate the number of constraints? Assume $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$.)

Practical RLO questions (2/3)

$(a + Pz)^T x \leq b, \forall \{z : ||z||_\infty \leq \rho\}$ became $a^T x + \rho ||P^T x||_1 \leq b$.

Questions:

1. The robust counterpart is what kind of optimization problem? **A linear optimization problem!**
2. What is the complexity of this optimization problem? (Hint: can you estimate the number of constraints? Assume $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$.) m **extra variables and** $2m$ **linear constraints are required, on top of the** n **original variables and constraint.**

Practical RLO questions (3/3)

$(a + Pz)^T x \leq b, \forall z : ||z||_\infty \leq \rho$ became $a^T x + \rho ||P^T x||_1 \leq b$.

Let's explicitly show the m extra variables and $2m$ linear constraints that make up the robust counterpart:

$$
\begin{aligned}
a^T x + \rho \sum_{i \in |m|} \delta_i \leq b, \\
\delta_i \geq P^T x, \forall i \in |m|, \\
\delta_i \geq -P^T x, \forall i \in |m|, \\
\delta \in \mathbb{R}^m.
\end{aligned}
$$

Takeaway: Added constraints can substantially affect the tractability of the robust counterpart!

Let's try another. How about polyhedral uncertainty?

Constraint with uncertain z in a polyhedron:

$$(a + Pz)^T x \leq b, \forall z : Dz \leq d. \tag{1}$$

Take the worst-case,

$$a^T x + \max_{z:\ Dz \leq d} (P^T x)^T z \leq b. \tag{2}$$

Use the dual!

$$\max_{z:\ Dz \leq d} (P^T x)^T z = \min_{D^T y = P^T x, y \geq 0} d^T y. \tag{3}$$

Replace max with min:

$$a^T x + \min_{D^T y = P^T x, y \geq 0} d^T y \leq b. \tag{4}$$

Rearrange:

$$a^T x + d^T y \leq b, D^T y = P^T x, y \geq 0. \tag{5}$$

Robust counterparts summary

- Started with infinite-dimensional problem under uncertainty.
- Took the max of the left-hand-side with respect to uncertain parameters.
- Did some math magic to end with deterministic optimization, devoid of uncertain parameters!

Note: You do not have to derive these yourself, unless your uncertainty lies in an exotic set. Many robust counterparts exist in the literature.

Cautions:

- Explicit robust counterparts may or may not exist (but there are workarounds).
- If they exist, it's not obvious that they are tractable (but very often they are).

Some examples of robust counterparts for linear constraints

$$(\bar{a} + Pz)^T x \leq b, \quad \forall z \in Z.$$

Unc. set	Z	Robust Counterpart	Tractability
Box	$\|z\|_\infty \leq \rho$	$\bar{a}^T x + \rho \|P^T x\|_1 \leq b$	LO
Ellipsoidal	$\|z\|_2 \leq \rho$	$\bar{a}^T x + \rho \|P^T x\|_2 \leq b$	CQO
Polyhedral	$Dz \leq d$	$\begin{cases} \bar{a}^T x + d^T y \leq b \\ D^T y = P^T x \\ y \geq 0 \end{cases}$	LO
Budget	$\begin{cases} \|z\|_\infty \leq \rho \\ \|z\|_1 \leq \Gamma \end{cases}$	$\bar{a}^T x + \rho \|y\|_1 + \Gamma \|P^T x - y\|_\infty \leq b$	LO

(convex quadratic optimization)

(linear optimization)

My personal favorite. Linear and tractable.

New research allows for finding robust counterparts for almost any combination constraint and set.

- For convex constraints and sets, it's called Reformulation-Perspectification.
- If you are comfortable with mathematics of convexity, please check out the paper:
  - Bertsimas, Dimitris, et al. "Robust convex optimization: A new perspective that unifies and extends." *Mathematical Programming* 200.2 (2023): 877-918.
- For uncertainty sets with discrete partitions, see below:
  - Bertsimas, Dimitris, and Iain Dunning. "Multistage robust mixed-integer optimization with adaptive partitions." Operations Research 64.4 (2016): 980-998.

Sometimes, we don’t even know we are using robustness, when we are!

Anyone use LASSO for linear regression (LR)? What's the benefit of LASSO over unregularized linear regression?

$$\min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||_q + \rho||\beta||_r \quad \text{vs.} \quad \min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||_q$$

LASSO is robust regression (not sparse). (1/2)

Let's derive it!

$$\min_{\beta} \max_{\Delta \mathbf{X} \in \mathcal{U}} ||\mathbf{y} - (\mathbf{X} + \Delta \mathbf{X})\beta||_2. \tag{1}$$

Let's assume that $\mathcal{U}$ is feature-wise uncorrelated:

$$\mathcal{U} := \left\{ (\delta_1, \ldots, \delta_m) : ||\delta_i||_2 \leq c_i,\ \forall i \in |m| \right\}. \tag{2}$$

Separate, with a relaxation:

$$\min_{\beta} \max_{(\delta_1,\ldots,\delta_m):||\delta_i||_2 \leq c_i} ||\mathbf{y} - (\mathbf{X} + (\delta_1, \ldots, \delta_m))\beta||_2 \leq \min_{\beta} \left\{ ||\mathbf{y} - \mathbf{X}\beta||_2 + \max_{(\delta_1,\ldots,\delta_m):||\delta_i||_2 \leq c_i} \sum_{i=1}^{m} ||\beta_i \delta_i||_2 \right\}. \tag{3}$$

*H. Xu, C. Caramanis, and S. Mannor. Robust regression and Lasso. http://arxiv.org/abs/0811.1790v1, 2008.
Also submitted to NeurIPS.

LASSO is robust regression (not sparse). (2/2)

Recall our upper bound on the original robust optimization problem:

$$\min_{\beta}\left\{||\mathbf{y}-\mathbf{X}\beta||_2 + \max_{(\delta_1,\ldots,\delta_m):||\delta_i||_2\leq c_i} \sum_{i=1}^{m} ||\beta_i\delta_i||_2\right\} \quad (1)$$

Let's add an additional restriction on the δ_i's:

$$\delta_i := -c_i \text{sign}(\beta_i)u, \text{ where } u = \begin{cases} \frac{\mathbf{y}-\mathbf{X}\beta}{||\mathbf{y}-\mathbf{X}\beta||_2}, & \text{if } \mathbf{y}\neq\mathbf{X}\beta \\ \text{any vector with unit l2} - \text{norm otherwise} \end{cases} \quad (2)$$

This is a type of induced norm that still obeys $||\delta_i||_2 \leq c_i$. It also reduces to

$$||\beta_i\delta_i||_2 = ||-\beta_i c_i \text{sign}(\beta_i)u||_2 = c_i|\beta_i|. \quad (3)$$

There is a little more derivation in the paper*, but LASSO is *equivalent* to the above robust optimization problem under the induced norm:

$$\min_{\beta}||\mathbf{y}-\mathbf{X}\beta||_2+\rho||\beta||_1 \leftrightarrow \min_{\beta}||\mathbf{y}-\mathbf{X}\beta||_2+\sum_{i=1}^{m} c_i|\beta_i|, \text{ when } c_i=\rho, \forall i\in[m]! \quad (4)$$

*H. Xu, C. Caramanis, and S. Mannor. Robust regression and Lasso. http://arxiv.org/abs/0811.1790v1, 2008. Also submitted to NeurIPS.

How can you use robust optimization in your work?

Option 1 (medium-hard difficulty): Optimize directly using the robust counterpart.

- Steps
  - Write your optimization problem with no uncertainty.
  - Determine your uncertainty set, ideally using some data or model.
  - Look up the robust counterpart online (or derive it 😬).
  - Rewrite the constraints with uncertainty using the robust counterpart (and additional auxiliary variables and constraints).
  - Solve.
- (+) Fast, deterministic, one-shot solutions.
- (-) Can still be computationally difficult. Lots of work up front to write robust counterparts.

Option 2 (easy-medium difficulty): Use adversarial robust optimization.

- Steps:
  - Solve the nominal problem, with no uncertainty
  - Repeat:
    - For each constraint:
      - Solve the inner maximization problem for worst-case parameters u from the uncertainty set, for fixed decisions x.
      - If constraints are violated, use those parameters to add cuts (i.e. perturb the original constraints and add them to the existing constraints).
      - Terminate if no cuts are added!
    - Re-solve the problem for x with all accumulated cuts.
- (+) Can be applied to any mixed-integer convex optimization problem, without taking the robust counterpart, under any uncertainty set.
- (-) Computationally expensive. Check out this paper for a comparison: Bertsimas, Dimitris, Iain Dunning, and Miles Lubin. "Reformulation versus cutting-planes for robust optimization: A computational study." Computational Management Science 13 (2016): 195-217.

Facility Location Demo

(can find on my GitHub,
RОdemos/homeworks/HW4 at master · 1ozturkbe/RОdemos · GitHub)

Problem setup (1/4)

- N = 10 possible facility locations, with supply capacity limits.
- M = 50 customers, with demand of 50.
- Question we are answering: How many facilities should we construct, to serve customer demand while minimizing cost? (A MILO!)
  - Cost = facility fixed costs + transport costs (distance * flow).
- Customers may be served by multiple facilities.
- There is budget uncertainty on demand: $\begin{cases} ||d||_\infty \leq \rho = 1, \\ ||d||_1 \leq \Gamma = 5. \end{cases}$

Problem setup (2/4)

$$\min_{\boldsymbol{y}(\cdot),\boldsymbol{x}} \quad \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}y_{ij} + \sum_{i=1}^{n} f_i x_i$$

Objective with variable and fixed costs

$$s.t. \quad \sum_{i=1}^{n} y_{ij} \geq d_j, \quad \boxed{\forall d \in \mathcal{D},} \ \forall j \in [m],$$

Customer-wise supply and demand constraint

$$\sum_{j=1}^{m} y_{ij} \leq s_i x_i, \quad \forall i \in [n],$$

Facility-wise capacity constraint

$$y_{ij} \geq 0, \quad \forall i \in [n], \ \forall j \in [m],$$

Transport variables

$$\boldsymbol{x} \in \{0,1\}^n.$$

Facility location variables

Problem setup (3/4)

$$
\begin{aligned}
\min_{\boldsymbol{y}(\cdot),\boldsymbol{x}} \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}y_{ij} + \sum_{i=1}^{n} f_i x_i \\
s.t. \quad & \sum_{i=1}^{n} y_{ij} \geq d_j, \quad \boxed{\forall d \in \mathcal{D},} \; \forall j \in [m], \\
& \sum_{j=1}^{m} y_{ij} \leq s_i x_i, \quad \forall i \in [n], \\
& y_{ij} \geq 0, \quad \forall i \in [n], \; \forall j \in [m], \\
& \boldsymbol{x} \in \{0,1\}^n.
\end{aligned}
$$

Question: Notice anything that could be problematic with this formulation?

Uncertainty is spread over many constraints, thus very conservative. (This will be partially mitigated by adaptivity, so stay tuned!)

Problem setup (4/4)

$$
\begin{aligned}
\min_{\boldsymbol{y}(\cdot),\boldsymbol{x}} \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}y_{ij} + \sum_{i=1}^{n} f_i x_i + \lambda \left\| \sum_{i=1}^{n} y_{ij} - d_j, \forall j \in |m| \right\|_2, \forall d \in \mathcal{D}, \\
s.t. \quad & \sum_{j=1}^{m} y_{ij} \le s_i x_i, \quad \forall i \in [n], \\
& y_{ij} \ge 0, \quad \forall i \in [n],\ \forall j \in [m], \\
& \boldsymbol{x} \in \{0,1\}^n
\end{aligned}
$$

$$
\begin{aligned}
\min_{\boldsymbol{y}(\cdot),\boldsymbol{x}} \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}y_{ij} + \sum_{i=1}^{n} f_i x_i \\
s.t. \quad & \sum_{i=1}^{n} y_{ij} \ge d_j, \quad \forall d \in \mathcal{D},\ \forall j \in [m], \\
& \sum_{j=1}^{m} y_{ij} \le s_i x_i, \quad \forall i \in [n], \\
& y_{ij} \ge 0, \quad \forall i \in [n],\ \forall j \in [m], \\
& \boldsymbol{x} \in \{0,1\}^n.
\end{aligned}
$$

Question: Can you tell me THE BEST way to reduce conservativeness in this formulation?

Put the uncertainty in the objective through the Lagrangian! Now have a convex quadratic objective with a convex uncertainty set. But we will assume we NEED to satisfy demand exactly.

Addressed via four methods.

- Nominal (no uncertainty)
- Robust (static transportation decision rules)
- Adaptive (affine transportation decision rules w.r.t uncertainty)
  - We fix facility locations x before demand is realized, then make an affine transportation decision based on that realized demand d.

$$\{y = u + Vd,\ d_j = \bar{d}_j + (Pz)_j,\ \forall\ ||z||_\infty \leq \rho, ||z||_1 \leq \Gamma\}.$$

- Adversarial adaptive (affine transportation decision rules w.r.t uncertainty, via cuts)

Quick intro to adaptive robustness

$$\max_{d \in D} A(d)x + By(d) \leq b. \tag{1}$$

A is an uncertain matrix, x are the first-stage decisions, and $y(d)$ are the decision rules, which we will assume are linear for our application: $y(d) = u + Vd$. Assuming budget uncertainty, taking the robust counterpart has the same process!

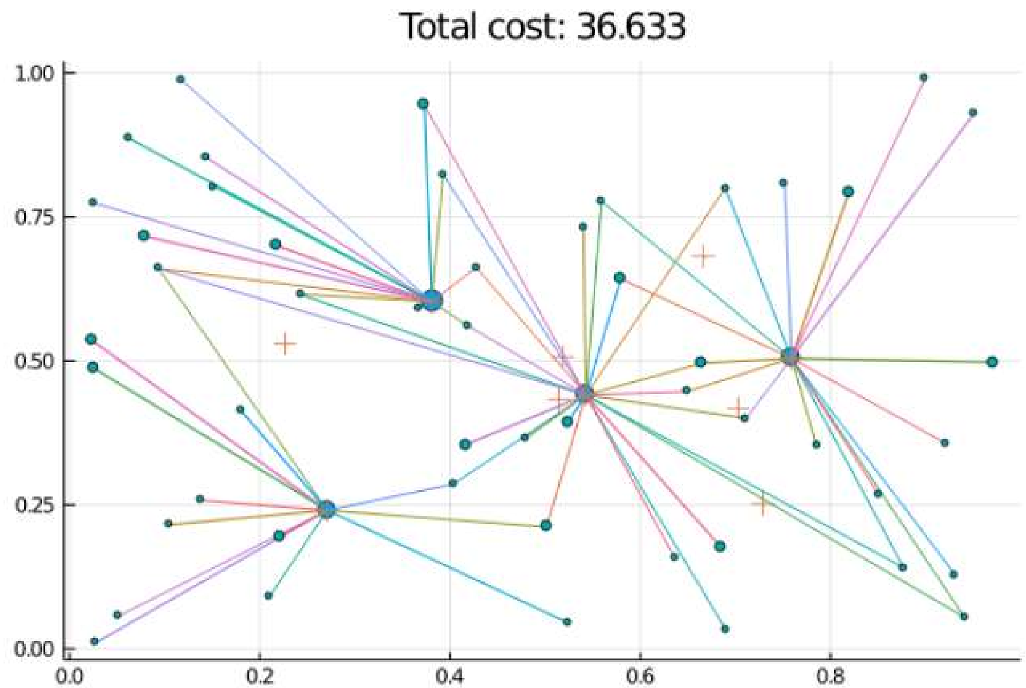
$$\max_{d \in D} (a + Pd)^T x + B^T(u + Vd) \leq b. \tag{2}$$

$$a^T x + B^T u + \max_{d \in D} d^T P^T x + d^T BV^T \leq b. \tag{3}$$

If d is in the budget set, i.e. $||d||_\infty \leq \rho$, and $||d||_1 \leq \Gamma$, we can simply use the robust counterpart table to rewrite as

$$a^T x + B^T u + \rho||\xi||_\infty + \Gamma||\xi||_1 \leq b, \; \xi = P^T x + BV^T. \tag{4}$$

Adaptive RO has high computational cost (especially depending on solver).



Solution time:
Adaptive takes 40s with Gurobi, adversarial 2 hours or so.

With GLPK, 40 minutes adaptive, adversarial >24 hours.

Scales of problems

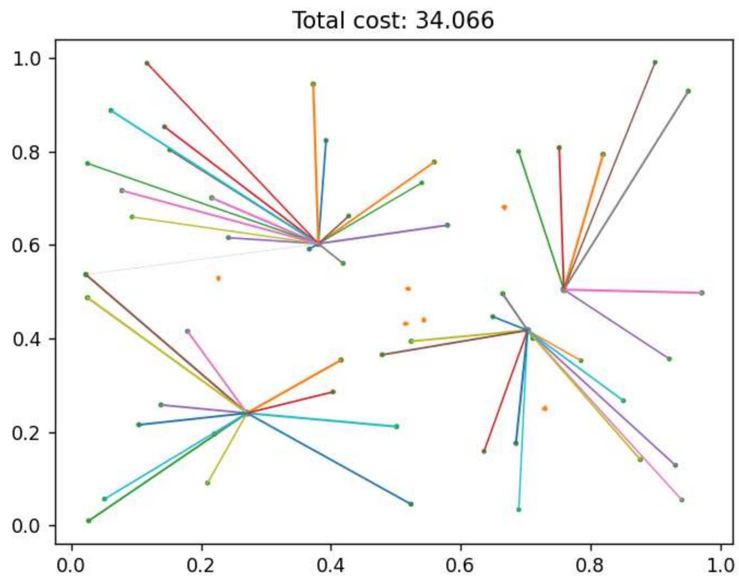
```
julia> include("nominal.jl")
GLPK Simplex Optimizer 5.0
60 rows, 510 columns, 1010 non-zeros
INTEGER OPTIMAL SOLUTION FOUND
Facility cost: 21.509166299898814
Transportation cost: 12.55731068395753
  0.522297 seconds (397.70 k allocations: 33.663 MiB, 32.59% compilation time)
```

```
julia> @time include("robust.jl")
GLPK Simplex Optimizer 5.0
10060 rows, 5560 columns, 23560 non-zeros
INTEGER OPTIMAL SOLUTION FOUND
Facility cost: 34.35670583607979
Transportation cost: 16.2223369008492
plt = nothing
  3.054248 seconds (1.15 M allocations: 78.755 MiB, 0.54% gc time, 10.28% compilation time)
```

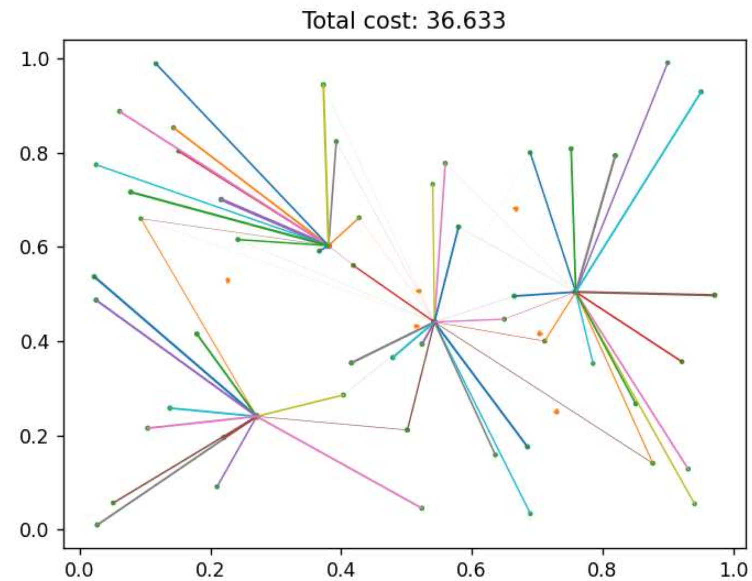
```
julia> include("adaptive.jl")
GLPK Simplex Optimizer 5.0
112761 rows, 82172 columns, 455032 non-zeros
INTEGER OPTIMAL SOLUTION FOUND
2308.518484 seconds (3.48 M allocations: 196.539 MiB, 0.00% gc time, 0.00% compilation time)
Facility cost: 21.427802592678976
Transportation cost: 13.355109872484778
```

~40 minutes

The price of robustness can be low!



Nominal (no uncertainty)



Adaptive

We only increased our cost by 7.5%, to protect against a huge amount demand uncertainty (more precisely up to 14% of total across the 50 customers)!

Engineering Design under Uncertainty via Robust Optimization (from PhD thesis work)

Application: aircraft design that captures important multidisciplinary tradeoffs.

- Unmanned, gas-powered aircraft
- Without uncertainty: 176 variables and 154 constraints
- Monolithic: optimizes aircraft and flight trajectory concurrently through disciplined signomial programming (nonconvex) form

Wing

- Structure
- Fuel volume
- Profile drag
- Stall constraint

Fuselage

- Fuel and payload
- Profile drag

Engine

- Data-based sizing
- Lapse rate
- BSFC fits
- T/O and TOC constraints

First, we determine uncertain parameters and variances, based on engineering intuition.

Table 3.1: Parameters and uncertainties (increasing order)

Parameters	Description	Value	% Uncert. (3σ)
e	span efficiency	0.92	3
μ	air viscosity (SL)	1.78×10^{-5} kg/(ms)	4
ρ	air density (SL)	1.23 kg/m^3	5
$C_{L,\max}$	stall lift coefficient	1.6	5
k	fuselage form factor	1.17	10
$C_{f,\mathrm{ref}}$	reference fuselage skin friction factor	0.455	10
ρ_{p}	payload density	1.5 kg/m^3	10
N_{ult}	ultimate load factor	3.3	15
$V_{\min}$	takeoff speed	35m/s	20
W_{p}	payload weight	3000 N	20
$W_{\mathrm{coeff,strc}}$	wing structural weight coefficient	2×10^{-5} 1/m	20
$W_{\mathrm{coeff,surf}}$	wing surface weight coefficient	60 N/m^2	20

The uncertainty is defined by box and ellipsoidal sets.

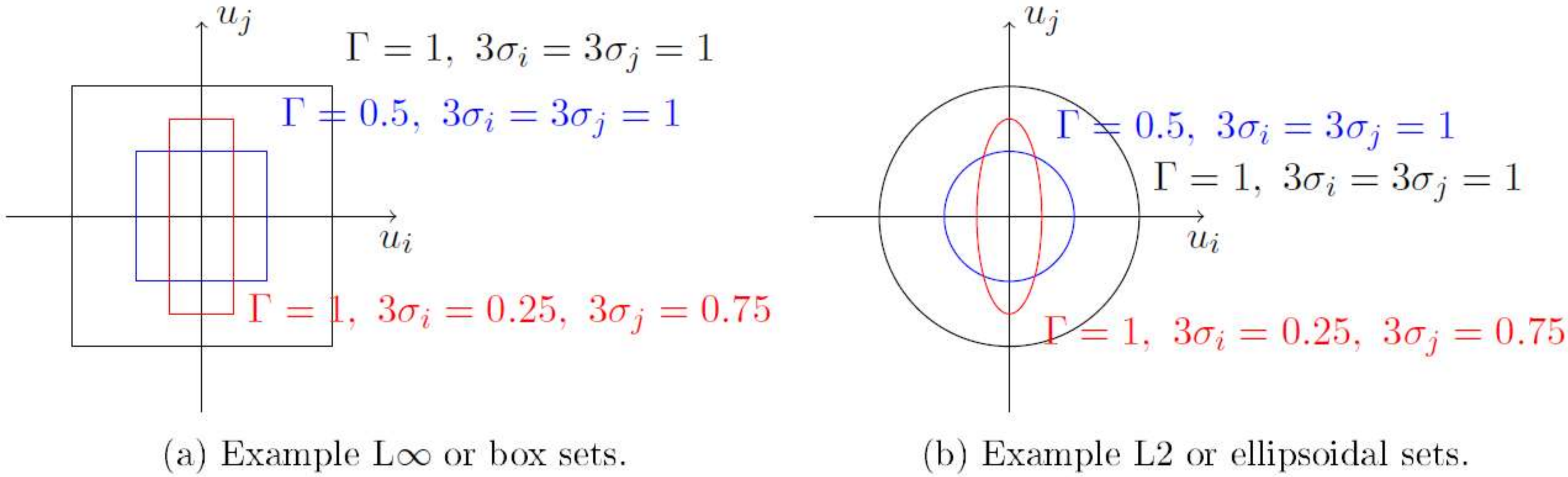
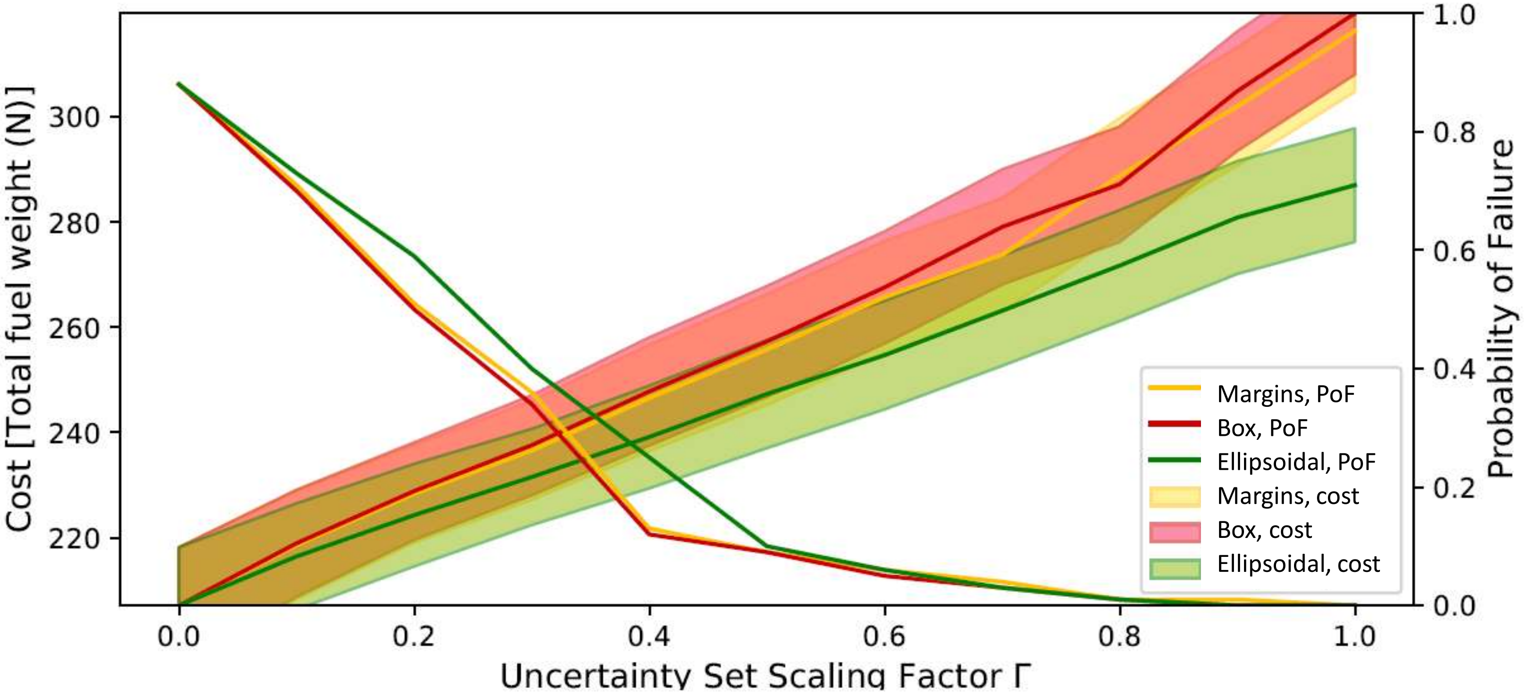


Figure 3-5: Γ defines the overall size of norm uncertainty sets, while 3σ defines the relative size of the set in each uncertain parameter.

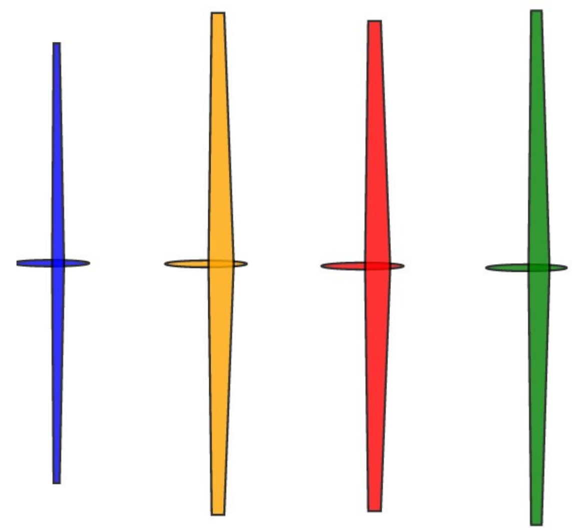
Primary result: RO mitigates probability of constraint violation under uncertain outcomes,



and is less conservative than designs with margins.

Table 2: SP Aircraft Optimization Results, for $\Gamma = 1$

Free variable	Description	Units	No Uncert.	Margins	Box	Elliptical
L/D	mean lift-to-drag ratio	-	33.6	23.6	25.1	27.7
AR	aspect ratio	-	24.6	13.3	13.0	16.3
Re	Reynolds number	-	1.54×10^6	2.65×10^6	3.03×10^6	250×10^6
S	wing planform area	m^2	13.6	32.8	32.0	28.1
V	mean flight velocity	m/s	41.6	37.3	38.9	38.4
T_{flight}	time of flight	hr	20.1	22.4	21.4	21.7
W_{w}	wing weight	N	2830	4760	4800	4480
$W_{\text{w,strc}}$	wing structural weight	N	2010	4760	2670	2620
$W_{\text{w,surf}}$	wing skin weight	N	820	2170	2120	1860
W_{fuse}	fuselage weight	N	250	314	288	279
$V_{\text{f,avail}}$	total fuel volume	m^3	0.0759	0.146	0.154	0.136
$V_{\text{f,fuse}}$	fuselage fuel volume	m^3	0.0394	0	0	0.0159
$V_{\text{f,wing}}$	wing fuel volume	m^3	0.0365	0.167	0.154	0.120
Objective metric	**Description**	**Units**	**No Uncert.**	**Margins**	**Box**	**Elliptical**
Objective	total fuel weight	N	608	1170	1240	1090
E[Objective]	expected total fuel weight	N	572	964	976	856
σ[Objective]	std. dev. of fuel weight	N	9	32	32	29
P[failure]	probability of failure	%	94	0	0	0



*100 MC simulations over 3σ truncated Gaussians

References / resources

ROdemos

https://github.com/1ozturkbe/ROdemos

This repository contains a number of practical and tractable examples of robust optimization (RO), applied to problems as diverse as experimental design and multicommodity network flows. They are written in Julia, using the JuMP modeling environment. The models accept a variety of optimizers compatible with JuMP, but Gurobi is the default, available with a free academic license.

Good papers on this topic

- Bertsimas, Dimitris, Iain Dunning, and Miles Lubin. "Reformulation versus cutting-planes for robust optimization: A computational study." Computational Management Science 13 (2016): 195-217.
- Bertsimas, D., & Sim, M. (2004). The Price of Robustness. *Operations Research*, *52*(1), 35–53. https://doi.org/10.1287/opre.1030.0065
- Bertsimas, D., Brown, D. B., & Caramanis, C. (2011). Theory and Applications of Robust Optimization. *SIAM Review*, *53*(3), 464–501. https://doi.org/10.1515/9781400831050
- Bertsimas, D., Hertog, D. den, & Pauphilet, J. (2019). *Probabilistic guarantees in Robust Optimization*. *September*, 1–27.
- Ben-Tal, A., & Nemirovski, A. (1999). Robust solutions of uncertain linear programs. *Operations Research Letters*. https://doi.org/10.1016/S1470-2045(03)01230-0
- Bertsimas, D., Brown, D. B., & Brown, D. B. (2009). *Constructing Uncertainty Sets for Robust Linear Optimization*. *September 2019*. https://doi.org/10.1287/opre.1080.0646
- Bertsimas, Dimitris, et al. "Robust convex optimization: A new perspective that unifies and extends." *Mathematical Programming* 200.2 (2023): 877-918.
- H. Xu, C. Caramanis, and S. Mannor. Robust regression and Lasso. http://arxiv.org/abs/0811.1790v1, 2008. Also submitted to NeurIPS.

Mediocre papers on this topic

- Öztürk, B., & Saab, A. (2021). Optimal Aircraft Design Decisions Under Uncertainty Using Robust Signomial Programming. *AIAA Journal*, *59*(5), 1–13. https://doi.org/10.2514/1.j058724