Randomized Techniques for Design Under Uncertainty *

Giuseppe Carlo Calafiore

Dipartimento di Automatica e Informatica
Politecnico di Torino, Italy.
Tel.: +39-011-564.7071; Fax: +39-011-564.7099
E-mail: giuseppe.calafiore@polito.it

Abstract

Engineering design problems can often be cast as numerical optimization prgrams where the designer goal is to minimize a cost index (or maximize a utility index), subject to a set of constraints on the decision variables. In particular, a class of design problems that have both a theoretical and a practical relevance are those where the minimization objective and the constraints are convex functions of the variables. In this case, the design problem turns out to be *computationally tractable*, i.e. suitable numerical codes exist that permit to determine an optimal solution with a computational effort that grows gracefully with the problem size.

However, in practice, the problem data are often *uncertain*, and hence the design needs not only be optimal, but also *guaranteed*, or *robust*, against the uncertainty. Unfortunately, it has been proven that convex problems in which uncertainty is present are very hard to solve. In this note, we discuss a novel technique based on uncertainty randomization that permits to overcome this difficulty. This technique is both simple in its implementation and backed by a rigorous theoretical analysis. In the author's opinion, these two features make this methodology appealing as a general computational tool for robust design.

1 Introduction

The main motivation for studying robustness problems in engineering comes from the fact that the actual system (a "plant," or in general a problem involving physical data) upon which the engineer should act, is realistically not fixed but rather it entails some level of uncertainty. For instance, the data that characterizes an engineering problem typically depends on the value of physical parameters. If measurements of these parameters are

*This work is supported by MIUR under the FIRB project "Learning, randomization and guaranteed predictive inference for complex uncertain systems."

performed, say, on different days or under different operating conditions, it is likely that we will end up not with a single (nominal) problem representation D, but rather with a family $D(\delta)$ of possible problems, where $\delta \in \Delta$ represents the vector of uncertain parameters that affect the data, and Δ is the admissible set of variation of the parameters. Any sensible design should in some way take into account the variability in the problem data, i.e. it should be *robust* with respect to the uncertainty.

To formalize our setup, we consider specifically design problems that may be expressed in the form of minimization of a linear objective subject to convex constraints:

$$\min_{x \in \mathcal{X}} \quad c^T x \quad \text{subject to:}$$
$$f_i(x, \delta) \leq 0, \quad i = 1, \ldots, m$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ is a convex set, and the functions $f_i(x, \delta)$ that define the constraints are convex in the decision variable x, for any given value of the uncertainty $\delta \in \Delta$. Without loss of generality, we can actually consider problems with a single constraint function

$$f(x, \delta) \doteq \max_{1,\ldots,m} f_i(x, \delta)$$

(the maximum of convex functions is still convex), which yields a prototype problem in the form

$$\min_{x \in \mathcal{X}} \quad c^T x \quad \text{subject to:}$$
$$f(x, \delta) \leq 0.$$

Notice that in this problem statement we remained voluntarily vague as to the meaning of robustness. We shall next define and briefly discuss three different ways in which this robustness can be intended.

2 Three robust design paradigms

2.1 Worst-case design

A first paradigm is a worst-case one, in which we seek a design x that satisfies the constraints *for all possible realizations* of the uncertain parameter δ. In formal terms, the design problem becomes

$$\min_{x \in \mathcal{X}} \quad c^T x \quad \text{subject to:} \qquad (1)$$
$$f(x, \delta) \leq 0, \quad \forall \delta \in \Delta.$$

A worst-case design may be necessary in cases when the violation of a constraint is associated to an unacceptable cost or it has disastrous consequences. It is a "pessimistic" paradigm in which the designer tries to be guaranteed against all possible odds. For the same reason, this kind of design tends to be conservative, since it takes into account also very rare events that may never realize in practice. Also, from a computation point of view, worst-case design problems are, in general, computationally hard. Even under the convexity assumptions the convex optimization problem (1) entails a usually infinite number of constraints. This class of problem goes under the name of robust convex programs, which are known to be NP-hard, see for instance [1, 6].

2.2 Probabilistic design

A second paradigm for robustness is a probabilistic one. In this setup, we add further structure on the problem, assuming that δ is a random variable with assigned probability distribution over Δ. Then, the probabilistic design objective is to determine a parameter x that satisfies the constraints up to a given high level of probability, $p \in (0, 1)$. Formally, the optimization problem takes the form:

$$\min_{x\in\mathcal{X}} \quad c^T x \quad \text{subject to:} \tag{2}$$

$$\text{Prob}\{f(x, \delta) \leq 0\} \geq p. \tag{3}$$

This design formulation alleviates the pessimism inherent in the worst-case design, but still gives rise to a numerically untractable problem. In fact, even under the convexity assumption, problem (2) can be extremely hard to solve exactly in general. This is due to the fact that the probability in the so-called "chance constraint" (3) can be hard to compute explicitly and, more fundamentally, to the fact that the restriction imposed on x by (3) is in general *nonconvex*, even though $f(x, \delta)$ is convex in x, see for instance [4, 7, 9].

2.3 Sampled scenarios design

Finally, we define a third approach to robustness, which is the scenario approach: let $\delta^{(1)}, \ldots, \delta^{(N)}$ be N independent and identically distributed random samples of the uncertainty $\delta \in \Delta$, extracted according to some assigned probability distribution. Each sample corresponds to a different realization (scenario) of the uncertain parameters upon which the problem data depend. If the problem solver task is to devise a once and for all fixed policy that performs well on the actual (unknown) problem, a sensible strategy would be to design this policy such that it performs well on all the collected scenarios. This is of course a well-established technique which is widely used in practical problems, and it is for instance the standard way in which uncertainty is dealt with in difficult financial planning problems, such as multi-stage stochastic portfolio optimization, see e.g. [5].

We hence define the *scenario design problem* as:

$$\min_{x\in\mathcal{X}} \quad c^T x \quad \text{subject to:} \tag{4}$$

$$f(x, \delta^{(i)}) \leq 0, \quad i = 1, \ldots, N. \tag{5}$$

We readily notice that the scenario problem is a standard convex problem with a finite number of constraints, and therefore it is generally solvable efficiently by numerical techniques, such as interior point methods.

While simple and effective in practice, the scenario approach also raises interesting theoretical questions. First, it is clear that a design that is robust for given scenarios is not robust in the worst-case sense, unless the considered scenarios actually contain all possible realizations of the uncertain parameters. Also, satisfaction of the constraints for the considered scenarios does not a-priori enforce the probabilistic constraint (3). Then, it becomes natural to ask what is the relation between robustness in the scenario sense and the probabilistic robustness. It turns out that a design based on scenarios actually guarantees a specified level of probabilistic robustness, provided that the number N of scenarios is chosen properly.

3 Properties of the scenario design

In this section we analyze in further detail the properties of the solution of the scenario-robust design problem (4). The first results on sampling-based convex optimization appeared recently in the paper [3], whereas an important refinement of these results is given in [2]. This section is based on the results contained in these references.

Denote with $\hat{x}_N$ the optimal solution of (4), assuming that the problem is feasible and the solution is attained. Notice that problem (4) is certainly feasible whenever the worst-case problem (1) is feasible, since the former involves a subset of the constraints of the latter. Notice further that the optimal solution $\hat{x}_N$ is a random variable, since it depends on the sampled random scenarios $\delta^{(1)}, \ldots, \delta^{(N)}$.

The following key result establishes the connection between the scenario approach and the probabilistic approach to robust design.

Theorem 1 (Scenario optimization) *Let $p, \beta \in (0,1)$ be given probability levels, and let $\hat{x}_N$ denote the optimal solution[1] of problem (4), where the number N of scenarios has been selected so that*

$$N \geq \frac{2}{1-p} \ln \frac{1}{\beta} + 2n + \frac{2n}{1-p} \ln \frac{2}{1-p}. \tag{6}$$

Then, it holds with probability at least $1-\beta$ that

$$\text{Prob}\{f(x,\delta) \leq 0\} \geq p.$$

What it is claimed in the above theorem is that if the number of scenarios is selected according to the bound (6), then the optimal solution returned by the scenario-robust design has, with high probability $1-\beta$, a guaranteed level p of probabilistic robustness. An important feature of bound (6) is that the probability level β enters it under a logarithm,

[1]Here, we assume for simplicity that the optimal solution exists and it is unique. These hypotheses can be removed without harming the result, see [3].

and therefore β may be chosen very small without substantially increasing the required number of samples. For instance, setting $\beta = 10^{-9}$, we have the bound

$$N \geq \frac{41.5}{1-p} + 2n + \frac{2n}{1-p} \ln \frac{2}{1-p}.$$

4 Discussion

Seeking robustness in design by considering different scenarios has long been a common practice in engineering. However, the application of the sampling technique was mainly driven by heuristics, and no general result was previously available to answer the key question: "how many scenarios are needed to guarantee some given level of robustness"?

The methodology that we briefly illustrated here actually answers in a rigorous way this fundamental question. The result from Theorem 1 states that the number of required scenarios scales gracefully with the problem dimension n and with the probabilistic levels. Convex optimization based on sampled scenarios is thus a simple, rigorous and efficient way to achieve robustness in design.

In a broader perspective, methods based on sampling and randomization have proved to be effective in solving in a relaxed sense problems that are otherwise hard to attack by means of classical deterministic techniques. Specific applications of randomized techniques in the domain of dynamic systems and robust control are extensively discussed in the recent monograph [8]. A comprehensive and up-to-date account of general probabilistic optimization methods, along with many pointers to the literature, is instead available in the edited monograph [4].

References

[1] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

[2] G. Calafiore and M.C. Campi. A new bound on the generalization rate of sampled convex programs. In *Proc. IEEE Conf. on Decision and Control*, Paradise Island, Bahamas, 2004.

[3] G. Calafiore and M.C. Campi. Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, 102(1):25–46, 2005.

[4] G. Calafiore and F. Dabbene (editors). *Probabilistic and Randomized Methods for Design under Uncertainty.* Springer, London, to appear: 2005.

[5] G.B. Dantzig and G. Infanger. Multi-stage stochastic linear programs for portfolio optimization. *Annals of Operations Research*, 45:59–76, 1993.

[6] L. El Ghaoui, F. Oustry, and H. Lebret. Robust solutions to uncertain semidefinite programs. *SIAM J. Optimization*, 9(1):33–52, 1998.

[7] A. Prékopa. *Stochastic Programming.* Kluwer, 1995.

[8] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems.* Communications and Control Engineering Series. Springer-Verlag, London, 2004.

[9] S. Uryasev (editor). *Probabilistic Constrained Optimization: Methodology and Applications.* New York, 2000.