Effects of Uncertain Requirements on the Architecture Selection Problem

Antonio Pugliese[a] , Roshanak Nilchiani[a], Somayeh Moazeni[a,*]

[a]*Stevens Institute of Technology, 1 Castle Point on Hudson, Hoboken, New Jersey 07030, USA*

Abstract

The problem of identifying a specific design or architecture that allows to satisfy all the system requirements becomes more difficult when uncertainties are taken into account. When a requirement is subject to uncertainty there are a number approaches available to the system engineer, each one with its own benefits and disadvantages. Classical robust optimization is one of the attractive approaches for optimization under uncertainty, since it allows to select the design that will perform better in the worst case scenario. In this framework uncertainty is described deterministically by uncertainty sets, which directly impact characteristics of the robust counterpart problem and quality of the robust solution. In other words, depending on the way uncertainty sets are defined, there is a chance that the problem becomes infeasible, meaning that no architecture satisfies all the system requirements in presence of uncertainty. This paper shows the effect of uncertainties on the feasibility of the problem introducing a measure of infeasibility.

Keywords: Requirements; Uncertainty; Robust Optimization

1. Introduction

The selection of a specific design or architecture for a system is an important component of the systems engineering process. Given a set of requirements, capturing objectives and constraints, there may be several designs that will satisfy them at various levels. More precisely, the designs that fully satisfy the constraints, the suitable designs, will perform differently in achieving the objectives. The purpose of the selection process is then to identify a design, or designs, that better achieve the objectives, thus maximizing the utility for the stakeholders.

When the constraints are well defined and the suitability of a design can be precisely measured, the search for an optimal configuration is about finding the maximum of the objective function. However, when the constraints are

* Corresponding author. Tel.: +1-201-216- 8723
E-mail address: smoazeni@stevens.edu

subject to uncertainty, both the modeling and optimization should take into account these uncertainties. One of these modeling and solution strategies is the robust approach, which tries to select a design with the optimum of the objective function under the worst-case scenario when all the uncertainties act upon the constraints in the direction of shrinking the design space. Selecting a design with the robust optimization approach means to build a system that will work under all scenarios with a performance at least as expected, and likely better than expected, when not all of the uncertain events happen simultaneously. This approach is preferred for the design of those systems in which failing a requirement cannot be tolerated and acceptable under any circumstances.

In this paper, two concerns regarding the classical robust optimization, due to its significant reliance on pre-specified sets for uncertainties, are rigorously discussed. First, we show that, the robust counterpart problem may become infeasible, even when a feasible design exists for many realized values of the uncertain inputs. Thus, in such cases, models explaining uncertainties in the requirements must be re-adjusted and re-specified in order to attain a feasible design space. In addition, even when the robust counterpart problem is feasible and a robust solution can be found, the system's performance at the robust solution might not be satisfactory and may be significantly worse than the objective function value at the nominal design. This conservativeness of the approach has been previously noted in the literature[1,2]. Whence to reduce conservativeness of the robust solution, similar to the previous case, the uncertainty sets need to be redefined.

Second, we provide an argument that for an effective adjustment, uncertainty sets of all constraints and the objective function must be taken into account simultaneously, in accordance with the claim that pairwise requirement comparison does not help in conflict resolution[3]. Specifying the parameters individually might be ineffective. Highlighting these uncertainty sets-related shortcomings of the classical robust optimization can motivate further research to develop more favorable approaches in the systems engineering process to obtain a robust solution.

2. Literature Review

The problem of design selection has been studied extensively in the recent past. At MIT this problem is labeled as *Tradespace Exploration*[4]. The tradespace exploration paradigm allows for the search of an optimal configuration, or to be more precise, a set of Pareto optimal configurations that take into account the utility of the system and its cost[5]. This approach is based on the consideration that typically space systems are evaluated using four metrics: cost, time, quality, and risk[6]. A different approach considers also the market potential of the design, taking into account competitor products[7], and the possibility of generating a product line out of a certain design, thus looking for flexibility in the design[8]. Both these methods focus mainly on the evaluation of design alternatives, and the selection of the best ones, without thinking whether the design space in which they are searching is in fact an optimal one.

The shape of the design space is defined by requirements representing constraints that the system needs to satisfy. The work of Salado on conflicting requirements[9] explains how some sets of requirements can define design spaces that have no solution, and that it is necessary to consider all the requirements at the same time, thus showing that pairwise comparison can hide the conflict. This model assumes that requirements are exact, and there is no uncertainty about their formulation.

Uncertainties are often considered at early phases of the system design[10]. Several efforts have been made towards a standardization of the type of uncertainties to be considered[11,12]. When uncertainties are considered, one of the attractive approaches to address optimization under uncertainty is the (classical) robust optimization[13,14]. In robust optimization, uncertainty is defined deterministically through uncertainty sets, which include all or most possible realizations of the uncertain inputs. Robust optimization, then, offers a solution which has the best worst performance when inputs belong in the given uncertainty sets. Most of the current literature on robust optimization is devoted to the applications of robust optimization or to develop tractable algorithms to solve the minimax robust counterpart problem, both under the assumption that prior uncertainty sets are given. This critical assumption about the decision environment is, however, a large burden on the decision maker. Determining uncertainty sets is a difficult task. Firstly, there are not many statistical techniques and studies available on constructing the uncertainty sets. Secondly, as it is shown in[15], the robust solution and its objective function value can be very sensitive to changes in parameters of the uncertainty sets.

3. Methodology

In this section, we outline the model of data uncertainty considered in this paper. The design goals are represented as an objective function that will be maximized. Suitability (or feasibility) of designs are defined by a set of constraints. For the purpose of illustration, we concentrate on constraints and objectives which are linear in design parameters with (well-defined) deterministic coefficient matrix. Here, we assume that solely the coefficients of the objective function and the entries of the right-hand-side vector are subject to uncertainty. In reality, not all the system requirements will be linear, but this assumption simplifies the model and allows us to show the effects of uncertainty.

3.1. Robust optimization and design selection problem

An uncertain linear programming problem with a deterministic coefficient matrix $A \in \mathbb{R}^{m\times n}$ is of the form:

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & \tilde{c}^T x \\ \text{s.t.} \quad & Ax \leq \tilde{b} \end{aligned} \tag{1}$$

where $\tilde{c}$ and $\tilde{b}$ are n-vector and an m-vector, whose entries are subject to uncertainty, and represent respectively the importance that stakeholders give to the various variables and the values of the constraints. Throughout, superscript ^{T} denotes transpose of a vector or a matrix. Without loss of generality in the following discussion, we assume that all of the $\tilde{b}_i$s are uncertain.

Following the common practice in robust optimization[13], we assume that the uncertain inputs, $\tilde{c}$ and $\tilde{b}$, depend on a set of primitive independent uncertainties $\{\tilde{z}_l\}_{l=1}^N$, where $(\tilde{z}_1, \cdots, \tilde{z}_N)$belongs in a set defined through a vector norm $\|\cdot\|$. The choice of norm may depend on the statistical distribution assumption of uncertain parameters or a subjective opinion of the decision maker or the model user.

$$\begin{aligned} \mathcal{U}_{b_i}(\Omega_i) &\stackrel{\text{def}}{=} \left\{ \tilde{b}_i = b_i^{(0)} + \sum_{l=1}^{N} \Delta b_i^{(l)} \cdot \tilde{z}_l : \tilde{z} = (\tilde{z}_1, \cdots, \tilde{z}_N) \in \mathbb{R}^N, \|\tilde{z}\| \leq \Omega_i \right\}, \qquad i = 1, \cdots, m \\ \mathcal{U}_{c}(\Omega_c) &\stackrel{\text{def}}{=} \left\{ \tilde{c} = c^{(0)} + \sum_{l=1}^{N} \Delta c^{(l)} \cdot \tilde{z}_l : \tilde{z} = (\tilde{z}_1, \cdots, \tilde{z}_N) \in \mathbb{R}^N, \|\tilde{z}\| \leq \Omega_c \right\} \end{aligned} \tag{2}$$

Here, $c^{(0)}$ and $b^{(0)} = \left(b_1^{(0)}, \cdots, b_m^{(0)}\right)^T$ are the nominal values of the data. Denote $\Delta b_i = \left(b_i^{(1)}, \cdots, b_i^{(N)}\right)$ and $\Delta c = \left(\Delta c^{(1)}, \cdots, \Delta c^{(N)}\right)$. We assume that the directions of data perturbation are non-negative, $\Delta b_i \geq 0$ and $\Delta c \geq 0$. The parameters Ω_c and Ω_i are referred to as the *budgets of uncertainty* for the objective function and the constraints, respectively.

Let a_i be the ith column of A^T. It is easy to see that the robust counterpart of the constraint $a_i^T x \leq \tilde{b}_i$with respect to the uncertainty set $\mathcal{U}_{b_i}(\Omega_i)$ will be

$$a_i^T x \leq b_i^{(0)} - \Omega_i \|\Delta b_i\|^* \tag{3}$$

where $\|\cdot\|^*$ denotes the dual norm given by $\|u\|^* = \sup\{u^T x : \|x\| \leq 1\}$.

Budget of uncertainty parameters are usually specified subjectively by the decision maker or affected by several elements of institutional, social, physical, economical, and environmental limitations. These are the design margins that are used by the engineers to take into account manufacturing or integration uncertainties, or the monetary budget margins that program managers use as a buffer in order to overcome funding uncertainties. The parameter Ω_c is mainly to control the degree of conservatism of the robust solution. As Ω_c increases, one expects that the robust optimal value decreases. The role of the budgets of uncertainty, Ω_is, are to adjust robustness of the proposed solution. Bertisimas and Sim[2] suggest to select these parameters by establishing probabilistic guarantee for feasibility under reasonable probabilistic assumptions on uncertain inputs. Some attempts to determine uncertainty sets through risk measures have been made[16,17]. However, these methods assume that the support of the risk measure is given and is still performed constraint-wise.

Determining proper values for the parameters Ω_c and Ω_i is often a challenging task. An improper assignment of these parameters may result in an infeasible robust problem; whence the model user remains with no solution. Even when the robust counterpart problem is feasible, the stakeholders may be unsatisfied by the level of optimality or robustness of the proposed solution. Thus, to cure infeasibility or to reduce conservativeness of an obtained robust solution, these parameters must be modified jointly. These issues are addressed further in Sections 3.2 and 3.3.

3.2. Measure of infeasibility

We are now going to introduce a measure to quantify infeasibility of a system of linear inequalities. The quantification of the infeasibility in the design selection problem, based only on its requirements, has the goal of measuring by how much the constraints need to be adjusted in order to have a non-empty design space. Define, the *measure of infeasibility* of the system $Ax \leq \tilde{b}$, as below:

$$D(A,\tilde{b}) \stackrel{\text{def}}{=} \max\left\{0, \min_{x\in\mathbb{R}^n} \max_{i=1,\cdots,m} \frac{a_i^T x - b_i^{(0)} + \Omega_i \|\Delta b_i\|^*}{\|\Delta b_i\|^*}\right\}$$

This measure is scale-independent, meaning that for every positive $\lambda \in \mathbb{R}^m$, we have $D\big(\mathrm{Diag}(\lambda)A, \mathrm{Diag}(\lambda)\tilde{b}\big) = D(A,\tilde{b})$. Furthermore, $0 \leq D(A,\tilde{b}) \leq \max_{i=1,\cdots,m} \Omega_i$. Moreover, $D\big(A,\tilde{b}\big) = 0$ if and only if the feasible region of the robust counterpart problem, associated with the uncertainty sets $\mathcal{U}_{b_i}(\Omega_i)$, is non-empty, meaning that a robust solution exists. However, when $D\big(A,\tilde{b}\big) > 0$ there is no $x \in \mathbb{R}^n$ which satisfies inequalities (3), for all $i = 1,\cdots,m$. In other words, there is no design that satisfies all the constraints simultaneously, together with their margins.

The min-max problem in the definition of $D(A,\tilde{b})$ can be computed efficiently by solving the following linear programming problem:

$$\begin{aligned} \min_{x\in\mathbb{R}^n,\ \psi\in\mathbb{R}} \quad & \psi \\ \text{s.t.} \quad & a_i^T x - \|\Delta b_i\|^* \psi \leq b_i^{(0)} - \Omega_i \|\Delta b_i\|^*, \ \ i = 1,\cdots,m \end{aligned}$$

Thus, if $D\big(A,\tilde{b}\big) > 0$, $D(A,\tilde{b})$ is the minimum decrease in the budgets of uncertainty which guarantees a feasible robust counterpart problem. More precisely, the system

$$a_i^T x \leq b_i^{(0)} - (\Omega_i - \psi)\|\Delta b_i\|^*, \ \ i = 1,\cdots,m$$

is always feasible. It is worth mentioning that the definition of $D\big(A,\tilde{b}\big)$ can be extended to the case when A is also subject to uncertainty, by replacing $a_i^T x$ with $\max_{a_i \in \mathcal{U}_{a_i}} a_i^T x$, where $\mathcal{U}_{a_i}$ is the uncertainty set of the elements in a_i.

3.3. Design space and a non-zero measure of infeasibility

When $D(A, \tilde{b}) > 0$, the classical robust optimization would not propose any solution to the model user leaving the problem as an infeasible one. However, consider an uncertain problem in which the number of primitive uncertainties equals one, i.e., $N = 1$, and $\tilde{z}$ is randomly uniformly distributed. Thus each $\tilde{b}_i$ is uniformly distributed in the interval $[b_i^{(0)} - \Omega_i \Delta b_i, b_i^{(0)} + \Omega_i \Delta b_i\,]$. Whence, we have

$$\begin{aligned}\Pr(Ax \leq \tilde{b} \text{ is feasible}) = \; & \Pr(Ax \leq \tilde{b} \text{ is feasible and } b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for all } i) \\ & + \Pr(Ax \leq \tilde{b} \text{ is feasible and } b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i > \tilde{b}_i, \text{for some } i) \\ & \geq \Pr(Ax \leq \tilde{b} \text{ is feasible and } b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for all } i)\end{aligned} \tag{4}$$

Using the notion of conditional probability, the probability in (4) can be rewritten as:

$$\begin{aligned}& \Pr(b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for } i = 1, \cdots, m) \; \times \\ & \Pr(Ax \leq \tilde{b} \text{ is feasible} \mid b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for } i = 1, \cdots, m)\end{aligned} \tag{5}$$

Since the system $a_i^T x \leq b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, i = 1, \cdots, m$ is always feasible, the conditional probability (5) equals 1. Hence,

$$\begin{aligned}& \Pr(Ax \leq \tilde{b} \text{ is feasible and } b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for } i = 1, \cdots, m) = \\ & \Pr(b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for } i = 1, \cdots, m)\end{aligned}$$

Using this equality along with inequality (4), we arrive at:

$$\begin{aligned}\Pr(Ax \leq \tilde{b} \text{ is feasible}) \; & \geq \Pr(b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i, \text{for } i = 1, \cdots, m) \\ & = \prod_{i=1}^{m} \Pr(b_i^{(0)} - (\Omega_i - D(A, \tilde{b}))\Delta b_i \leq \tilde{b}_i) \\ & = \prod_{i=1}^{m} \left(1 - \frac{D(A, \tilde{b})}{2\Omega_i}\right)\end{aligned}$$

where the last equality comes from the assumption that $\tilde{b}_i$ is uniformly distributed with support $[b_i^{(0)} - \Omega_i \Delta b_i, b_i^{(0)} + \Omega_i \Delta b_i\,]$. Now assume $\Omega_i = 1$ for all $i = 1, \cdots, m$. Thus, we get

$$\Pr(Ax \leq \tilde{b} \text{ is feasible}) \geq \left(1 - \frac{D(A, \tilde{b})}{2}\right)^m \tag{6}$$

Let $D(A, \tilde{b}) = 0.001$. Thus when the optimization problem (1) has 20 constraints, i.e., $m = 20$, inequality (6) yields to

$$\Pr(Ax \leq \tilde{b} \text{ is feasible}) \geq 0.99$$

This indicates that although the classical robust methodology considers the problem infeasible and proposes no robust solution, the underlying system is feasible with probability more than 99%; in other words, the problem is feasible for many realizations of the uncertain inputs. This is where robust optimization fails in the search of a suitable design. The assumption that all the constraints need to be satisfied at the same time, in order to have a robust system, will reject a large amount of designs that can easily satisfy $m-1$ uncertain requirements, or even m requirements in case a correction to the margin is applied.

A similar result can be observed when the entries of $\tilde{b}$ are independently and *normally* distributed. Let $\tilde{b}_i$ be normally distributed with mean $b_i^{(0)}$and standard deviation $\sigma\Delta b_i$. Table 1 presents lower bounds on the feasibility probability of the system $Ax \leq \tilde{b}$, when $\sigma = 0.25$. This table shows that the system remains feasible with a high probability even for a fairly large number of constraints.

Table 1 - Lower bounds on the feasibility probability of the system $Ax \leq \tilde{b}$ for normally distributed $\tilde{b}$. Here, $\sigma = 0.25$, $\Omega_i = 1$ for $i = 1, \cdots, m$, and $D(A, \tilde{b}) = 0.001$.

m	$\prod_{i=1}^{m} \Pr\left(b_i^{(0)} - (\Omega_i - D(A,\tilde{b}))\Delta b_i \leq \tilde{b}_i\right)$
1500	0.95283
500	0.98402
100	0.99678
25	0.99919
15	0.99951

In such cases that the problem is feasible for many realizations of the uncertain parameters, one expects that the chosen budgets of uncertainty are capable to offer a robust solution. This expectation, however, may not be fulfilled unless these parameters are determined jointly and based on a systematic method. In the next section, we discuss that the parameters determining the uncertainty sets must be reassigned (to make the robust counterpart problem feasible) simultaneously. This is in particular important, when we are dealing with large number of constraints in the optimization problem.

3.4. Adjustment of parameters defining the uncertainty sets

One of the shortcomings of the classical robust optimization, frequently reported in the literature, is its conservative nature, in the sense that too much of optimality for the nominal problem may be lost in order to ensure robustness[1]. One of the remedies suggested in the literature to find a less conservative robust solution has been to use different vector norms for a fixed budget of uncertainty. Since norms are related to each other:

$$\|z\|_\infty \leq \|z\|_2 \leq \|z\|_1 \leq \sqrt{N}\|z\|_2 \leq N\|z\|_\infty$$

a smaller norm results in a bigger uncertainty set and (most likely) a more conservative robust solution. For example, Ben-Tal and Nemirovski[1] suggest to use ellipsoidal norm instead of $\|\cdot\|_\infty$. Bertsimas and Sim[2] propose to use $\|z\|_\Gamma = \max \left\{\frac{1}{\Gamma}\|z\|_1, \|z\|_\infty\right\}$, to control the level of conservatism in the robust solution and to obtain less conservative solutions compared to the Soyster's method[18].

However, when the coefficient matrix A is not subject to uncertainty, inequality (3) shows that for a chosen norm $\|\cdot\|_s$ and the budget of uncertainty Ω_i , there exists some budget of uncertainty $\widehat{\Omega}_i$, which along with another norm $\|\cdot\|_t$, describes the same robust feasible region. Indeed, it is enough to set $\widehat{\Omega}_i = \Omega_i \frac{\|\Delta b_i\|_s^*}{\|\Delta b_i\|_t^*}$, where $\|\Delta b_i\|_s^*$ and $\|\Delta b_i\|_t^*$

are dual norms of $\|\cdot\|_s$ and $\|\cdot\|_t$, respectively. For example, if $\|\cdot\|_s$ is $\|\cdot\|_\Gamma$ and $\|\cdot\|_t$ is $\|\cdot\|_2$, we should assign

$$\widehat{\Omega}_i = \Omega_i \frac{\|\Delta b_i\|_s^*}{\|\Delta b_i\|_t^*} = \Omega_i \frac{\max\left(\frac{1}{\Gamma}\|\Delta b\|_1, \|\Delta b\|_\infty\right)}{\|\Delta b\|_2}$$

Hence, when the coefficient matrix defining the feasible region, A, is deterministic, the norm describing the uncertainty can be fixed and the conservativeness can be solely controlled through the budgets of uncertainty. An inappropriate selection of these parameters, then, tends to result in an over conservative solution, if feasible at all.

To obtain a less conservative solution, the budgets of uncertainty, for both the objective function and constraints, need to be updated. However, this readjustment may not necessarily be effective, when it is done individually. As an example, consider a linear programming problem with four constraints and two decision variables (x, y):

$$\begin{array}{cl} \max\limits_{x,y\,\in\,\mathbb{R}} & \tilde{c}_1 x + \tilde{c}_2 y \\ \text{s.t.} & \begin{array}{l} \tilde{b}_1 \le x \le 1, \\ \tilde{b}_2 \le y \le 1 \end{array} \end{array}$$

where $\tilde{b}_1 \in [-2, 0]$, $\tilde{b}_2 \in [-2, 0]$ with nominal values $b_1^{(0)} = -1$ and $b_2^{(0)} = -1$. Further,

$$\mathcal{U}_c(\Omega_c) = \left\{ \begin{pmatrix} \tilde{c}_1 \\ \tilde{c}_2 \end{pmatrix} = \begin{pmatrix} c_1^{(0)} \\ c_2^{(0)} \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tilde{z}_1 + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tilde{z}_2 \,:\, \|(\tilde{z}_1, \tilde{z}_2)\|_1 \le \Omega_c \right\}.$$

Here, $(c_1^{(0)}, c_2^{(0)}) = (1,1)$. The nominal problem will be

$$\begin{array}{cl} \max\limits_{x,y\,\in\,\mathbb{R}} & x + y \\ \text{s.t.} & \begin{array}{l} -1 \le x \le 1, \\ -1 \le y \le 1 \end{array} \end{array}$$

Thus the unique nominal solution is $x = -1$ and $y = -1$, in which case the nominal optimal objective function value equals -2. The robust counterpart problem is

$$\begin{array}{cl} \min\limits_{x,y\,\in\,\mathbb{R}} \max\limits_{(\tilde{c}_1,\tilde{c}_2)^T \in\, \mathcal{U}_c(\Omega_c)} & \tilde{c}_1 x + \tilde{c}_2 y \\ \text{s.t.} & \begin{array}{l} \tilde{b}_1 \le x \le 1, \\ \tilde{b}_2 \le y \le 1 \end{array} \end{array}$$

Thus the unique robust solution is $x = 0$ and $y = 0$, and the robust optimal objective value will be 0. This solution does not depend on Ω_c. Thus the robust optimal objective function value cannot get improved even when the model user updates Ω_c; unless the interval uncertainty sets for $\tilde{b}_1$ and $\tilde{b}_2$, also get adjusted. However, if in this example we update both the uncertainty set for $\tilde{b}_1$ and the budget of uncertainty Ω_c, to $\tilde{b}_1 \in [-2, -0.5]$ and $\Omega_c = 0.5$, then the unique robust solution will be $x = -0.5$ and $y = 0$. Thus the robust optimal objective function value is decreased to -0.25. Therefore, improvement can only be achieved when the budgets of uncertainty are updated simultaneously.

This example shows the need for an approach to balancing the optimality and robustness, in that parameters of the uncertainty sets for the constraints as well as for the objective function are updated jointly and not individually. Translating this result in systems engineering words, we see that the value delivered to the stakeholders cannot be maximized by only solving the uncertainties related to their utility functions, which means that working only on defining more precise system performances, and getting the stakeholders to sign the most precise contracts in order to cancel any uncertainty on what are their real needs, is not going to help. There is a need to work on the uncertainties that relate to the constraints of the problem, otherwise the optimal design will not be reached. Therefore, a robust optimization approach should be equipped with some pre-processing, to infer as much useful information as possible about the structure of the feasible region polytope and to detect contributing constraints and factors to the robustness of the problem.

4. Conclusions

The classical robust optimization approach to deal with parameter uncertainty in optimization problems heavily relies on the description of the given uncertainty sets. In this paper, we address several shortcomings of the approach due to this dependence. Improper values for the parameters specifying the uncertainty sets may make the robust counterpart problem infeasible, while the problem has feasible points for many realizations of the uncertain inputs. Furthermore, individual adjustment of these parameters, to make the problem feasible or less conservative, can be ineffective. Thus updating the budgets of uncertainty must be done simultaneously. The discussion also highlights the importance of developing efficient techniques for data pre-processing before adopting the classical robust optimization framework, devising methods for determining uncertainty sets jointly, and proposing approaches less dependent on prespecified uncertainty sets.

References

1. Ben-Tal, A. and Nemirovski, A. (2000). Robust solutions of linear programming problems contaminated with uncertain data. Mathematical Programming, 88:411–424.
2. Bertsimas, D. and Sim, M. (2004). The price of robustness. Operations Research, 52(1):35–53.
3. Salado, A. and Nilchiani, R. (2013). The concept of order of conflict in requirements engineering.
4. Ross, A. M., Diller, N. P., Hastings, D. E., and Warmkessel, J. M. (2002). Multiattribute tradespace exploration in space system design. In International Astronautical Congress.
5. Ross, A. M. and Hastings, D. E. (2005). The tradespace exploration paradigm.
6. Ross, A. M., Hastings, D. E., and Diller, N. P. (2003). Multi-attribute tradespace exploration with concurrent design for space system conceptual design. In st Aerospace Sciences Meeting, AIAA2003-1328. Reno, NV: January, pages 6–9.
7. Li, H. and Azarm, S. (2000). Product design selection under uncertainty and with competitive advantage. Journal of Mechanical Design, 122(4):411–418.
8. Li, H. and Azarm, S. (2002). An approach for product line design selection under uncertainty and competition. Journal of Mechanical Design, 124(3):385–392.
9. Salado, A. and Nilchiani, R. (2014). The concept of problem complexity. Procedia Computer Science, 28:539–546.
10. de Weck, O., Eckert, C. M., John, C. P., et al. (2007). A classification of uncertainty for early product and system design. Guidelines for a Decision Support Method Adapted to NPD Processes.
11. Salado, A., Nilchiani, R., and Efatmaneshnik, M. (2012). Taxonomy and categorization of uncertainties in space systems with an application to the measurement of the value of adaptability. In AIAA SPACE 2012 Conference & Exposition.
12. McManus, H. and Hastings, D. (2005). A framework for understanding uncertainty and its mitigation and exploitation in complex systems. In INCOSE International Symposium, volume 15, pages 484–503. Wiley Online Library.
13. Ben-Tal, A., Ghaoui, L. E., and Nemirovski, A. (2009). Robust Optimization. Princeton University Press.
14. Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. SIAM Review, 53(3):464–501.
15. Moazeni, S., Coleman, T. F., and Li, Y. (2013). Regularized robust optimization for optimal portfolio execution. Computational Optimization and Application, 55(2):341–377.
16. Chen, X., Sim, M., and Sun, P. (2007). A robust optimization perspective on stochastic programming. Operations Research, 55(6):1058–1071.
17. Bertsimas, D. and Brown, D. B. (2009). Constructing uncertainty sets for robust linear optimization. Operations Research, 57(6):1483–1495.
18. Soyster, A. L. (1973). Convex programming with set-inclusive constraints and applications to inexact linear programming. Operations Research, 21:1154–1157.