Author's Accepted Manuscript

A practical guide to robust optimization

Bram L. Gorissen, İhsan Yanıkoğlu, Dick den Hertog

Cite this article as: Bram L. Gorissen, İhsan Yanıkoğlu, Dick den Hertog, A practical guide to robust optimization, *Omega,* http://dx.doi.org/10.1016/j.omega.2014.12.006

A Practical Guide to Robust Optimization

Bram L. Gorissen[†] • İhsan Yanıkoğlu[‡*] • Dick den Hertog[†]

[†]Center for Economic Research (CentER), Tilburg University, 5000 LE Tilburg, The Netherlands
[‡]Industrial Engineering Department, Özyeğin University, İstanbul, Turkey

b.l.gorissen@uvt.nl • ihsan.yanikoglu@ozyegin.edu.tr • d.denhertog@uvt.nl

Robust optimization is a young and active research field that has been mainly developed in the last 15 years. Robust optimization is very useful for practice, since it is tailored to the information at hand, and it leads to computationally tractable formulations. It is therefore remarkable that real-life applications of robust optimization are still lagging behind; there is much more potential for real-life applications than has been exploited hitherto. The aim of this paper is to help practitioners to understand robust optimization and to successfully apply it in practice. We provide a brief introduction to robust optimization, and also describe important do's and don'ts for using it in practice. We use many small examples to illustrate our discussions.

Key words: robust optimization; adjustable robust optimization

JEL classification: C00, C15, C44, C61, C63

1. Introduction

Real-life optimization problems often contain uncertain data. Data can be inherently stochastic/random or it can be uncertain due to errors. The reasons for data errors could be measurement/estimation errors that come from the lack of knowledge of the parameters of the mathematical model (e.g., the uncertain demand in an inventory model) or could be implementation errors that come from the physical impossibility to exactly implement a computed solution in a real-life setting. There are two approaches to deal with data uncertainty in optimization, namely robust and stochastic optimization. *Stochastic optimization* (SO) has an important assumption, i.e., the true probability distribution of uncertain data has to be known or estimated. If this condition is met and the reformulation of the uncertain optimization problem is computationally tractable, then SO is the methodology to solve the uncertain optimization problem at hand. For details on SO, we refer to Prékopa (1995), Birge and Louveaux (2011) and Ruszczyński and Shapiro (2003), but the list of references can be easily extended.

Robust optimization (RO), on the other hand, does not assume that probability distributions are known, but instead it assumes that the uncertain data resides in a so-called uncertainty set. Additionally, basic versions of RO assume "hard" constraints, i.e., constraint violation cannot be allowed for any realization of the data in the uncertainty set. RO is popular because of its computational

* Corresponding author

tractability for many classes of uncertainty sets and problem types. For a detailed overview of the RO framework, we refer to Ben-Tal et al. (2009), Ben-Tal and Nemirovski (2008) and Bertsimas et al. (2011).

Although the first published study (Soyster 1973) dates back to 1970s, RO is a relatively young and active research field, and has been mainly developed in the last 15 years. There have been many publications that show the value of RO in many fields of application including finance (Lobo 2000), energy (Bertsimas et al. 2013b, Babonneau et al. 2010), supply chain (Ben-Tal et al. 2005, Lim 2013), healthcare (Fredriksson et al. 2011), engineering (Ben-Tal and Nemirovski 2002), scheduling (Yan and Tang 2009), marketing (Wang and Curry 2012), etc. Indeed, the RO concepts and techniques are very useful for practice, since they are tailored to the information at hand and leads to tractable formulations. It is therefore remarkable that real-life applications are still lagging behind; there is much more potential for real-life applications than has been exploited hitherto.

In this paper we give a concise description of the basics of RO, including so-called *adjustable* RO for multi-stage optimization problems. Moreover, we extensively discuss several items that are important when applying RO, and that are often not well understood or incorrectly applied by practitioners. Several important do's and don'ts are discussed, which may help the practitioner to successfully apply RO. We use many small examples to illustrate our discussions.

The remainder of the paper is organized as follows. Section 2 gives a concise introduction to RO. Section 3-10 discuss several important practical issues in more detail. An important ingredient of RO is the so-called *uncertainty set*, which is the set of values for the uncertain parameters that are taken into account in the robust problem. This set has to be specified by the user, and Section 3 presents several ways how to construct uncertainty sets. Section 4 discusses an important technical aspect in *adjustable* RO. In practice, multi-stage problems may contain adjustable variables that are integer. Section 5 proposes a RO method that deals with such integer variables. Section 6 warns that the robust versions of equivalent deterministic problems are not necessarily equivalent, and gives the correct robust counterparts. Section 7 discusses that equality constraints containing uncertain parameters needs a special treatment. In RO one optimizes for the worst case scenario, i.e., for the worst case values of the uncertain parameters. Although this statement is true, the way it is formulated is often misunderstood. In Section 8, we therefore clarify how it should be formulated. It is important to test how robust the final solution is, and to compare it to, e.g., the nominal solution. Section 9 discusses how to assess the robustness performance of a given solution via a simulation study. Section 10 shows that RO applied in a folding horizon setting may yield better solutions than *adjustable* RO for multi-stage problems. Section 11 summarizes our conclusions, and indicates future research topics.

2. Introduction to robust optimization

In this section we first give a brief introduction to RO, and then we give a procedure for applying RO in practice. The scopes of the following sections are also presented in this section.

2.1. Robust optimization paradigm

For the sake of exposition, we use an uncertain linear optimization problem, but we point out that most of our discussions in this paper can be generalized for other classes of uncertain optimization problems. The "general" formulation of the uncertain linear optimization problem is as follows:

$$\min_{\mathbf{x}} \{\mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d}\}_{(\mathbf{c}, \mathbf{A}, \mathbf{d}) \in \mathcal{U}}, \tag{1}$$

where $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{d} \in \mathbb{R}^m$ denote the uncertain coefficients, and $\mathcal{U}$ denotes the user specified uncertainty set. The "basic" RO paradigm is based on the following three assumptions (Ben-Tal et al. 2009, p. xii):

A.1. All decision variables $\mathbf{x} \in \mathbb{R}^n$ represent "here and now" decisions: they should get specific numerical values as a result of solving the problem before the actual data "reveals itself".

A.2. The decision maker is fully responsible for consequences of the decisions to be made when, and only when, the actual data is within the prespecified uncertainty set $\mathcal{U}$.

A.3. The constraints of the uncertain problem in question are "hard", i.e., the decision maker cannot tolerate violations of constraints when the data is in the prespecified uncertainty set $\mathcal{U}$.

In addition to the "basic" assumptions, we may assume without loss of generality that: 1) the objective is certain; 2) the constraint right-hand side is certain; 3) $\mathcal{U}$ is compact and convex; and 4) the uncertainty is constraint-wise. Below, we explain the technical reasons of why these four assumptions are not restrictive.

E.1. Suppose the objective coefficients ($\mathbf{c}$) are uncertain and (say) these coefficients reside in the uncertainty set $\mathcal{C}$:

$$\min_{\mathbf{x}} \max_{\mathbf{c} \in \mathcal{C}} \{\mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d} \quad \forall \mathbf{A} \in \mathcal{U}\}.$$

Without loss of generality we may assume that the uncertain objective of the optimization problem can be equivalently reformulated as certain (Ben-Tal et al. 2009, p. 10):

$$\min_{\mathbf{x}, \, t} \{t : \mathbf{c}^\top \mathbf{x} - t \leq 0 \quad \forall \mathbf{c} \in \mathcal{C}, \ \mathbf{A}\mathbf{x} \leq \mathbf{d} \quad \forall \mathbf{A} \in \mathcal{U}\},$$

using a reformulation and the additional variable $t \in \mathbb{R}$.

E.2. The second assumption is not restrictive because the uncertain right-hand side of a constraint can always be translated to an uncertain coefficient by introducing an extra variable $x_{n+1} = -1$.

E.3. The uncertainty set $\mathcal{U}$ can be replaced by its convex hull conv($\mathcal{U}$), i.e., the smallest convex set that includes $\mathcal{U}$, because testing the feasibility of a solution with respect to $\mathcal{U}$ is equivalent to taking the supremum of the left hand side of a constraint over $\mathcal{U}$, which yields the same optimal objective value if the maximization is conv($\mathcal{U}$). For details of the formal proof and the compactness assumption, see (Ben-Tal et al. 2009, pp. 12–13).

E.4. To illustrate that robustness with respect to $\mathcal{U}$ can always be formulated constraint-wise, consider a problem with two constraints and with uncertain parameters d_1 and d_2: $x_1 + d_1 \leq 0$, $x_2 + d_2 \leq 0$. Let $\mathcal{U} = \{\mathbf{d} \in \mathbb{R}^2 : d_1 \geq 0,\ d_2 \geq 0,\ d_1 + d_2 \leq 1\}$ be the uncertainty set. Then, $\mathcal{U}_i = [0, 1]$ is the projection of $\mathcal{U}$ on d_i. It is easy to see that robustness of the i-th constraint with respect to $\mathcal{U}$ is equivalent to robustness with respect to $\mathcal{U}_i$, i.e., the uncertainty in the problem data can be modelled *constraint-wise*. For the general proof, see (Ben-Tal et al. 2009, pp. 11–12).

For uncertain *nonlinear* optimization problems, the assumptions are also without loss of generality, except the third basic assumption [E.3].

If we assume $\mathbf{c} \in \mathbb{R}^n$ and $\mathbf{d} \in \mathbb{R}^m$ are certain, then the robust reformulation of (1) that is generally referred to as the *robust counterpart* (RC) problem is given as follows:

$$\min_{\mathbf{x}}\{\mathbf{c}^\top \mathbf{x} :\ \mathbf{A}(\boldsymbol{\zeta})\mathbf{x} \leq \mathbf{d} \quad \forall \boldsymbol{\zeta} \in \mathcal{Z}\}, \tag{2}$$

where $\mathcal{Z} \subset \mathbb{R}^L$ denotes the user specified *primitive* uncertainty set. A solution $\mathbf{x} \in \mathbb{R}^n$ is called *robust feasible* if it satisfies the uncertain constraints $[\mathbf{A}(\boldsymbol{\zeta})\mathbf{x} \leq \mathbf{d}]$ for all realizations of $\boldsymbol{\zeta} \in \mathcal{Z}$.

As it is mentioned above, and explained in [E.4], we may focus on a single constraint, since the uncertainty is constraint-wise in RO. A single constraint taken out of (2) can be modeled as follows:

$$(\mathbf{a} + \mathbf{P}\boldsymbol{\zeta})^\top \mathbf{x} \leq d \quad \forall \boldsymbol{\zeta} \in \mathcal{Z}. \tag{3}$$

In the left-hand side of (3), we use a factor model to formulate a single constraint of (2) as an affine function $\mathbf{a} + \mathbf{P}\boldsymbol{\zeta}$ of the *primitive* uncertain parameter $\boldsymbol{\zeta} \in \mathcal{Z}$, where $\mathbf{a} \in \mathbb{R}^n$ and $\mathbf{P} \in \mathbb{R}^{n \times L}$. One of the most famous example of such a factor model is the 3-factor model of Fama and French (1993), which models different type of assets as linear functions of a limited number of uncertain economic factors. To point out, the dimension of the general uncertain parameter $\mathbf{P}\boldsymbol{\zeta}$ is often much higher than that of the primitive uncertain parameter $\boldsymbol{\zeta}$ (i.e., $n \gg L$).

2.2. Solving the robust counterpart

Notice that (3) contains infinitely many constraints due to the *for all* ($\forall$) quantifier imposed by the worst case formulation, i.e., it seems intractable in its current form. There are two ways to deal with this. The first way is to apply robust reformulation techniques to exclude the *for all* ($\forall$) quantifier. If

deriving such a robust reformulation is not possible, then the second way is to apply the adversarial approach. In this section, we describe the details of these two approaches.

We start with the first approach, which consists of three steps. The result will be a computationally tractable RC of (3), which contains a finite number of constraints. Note that this reformulation technique is one of the main techniques in RO (Bertsimas et al. 2011).

We illustrate the three steps of deriving the RC based on a polyhedral uncertainty set:

$$\mathcal{Z} = \{\boldsymbol{\zeta} : \mathbf{D}\boldsymbol{\zeta} + \mathbf{q} \geq 0\},$$

where $\mathbf{D} \in \mathbb{R}^{m \times L}, \boldsymbol{\zeta} \in \mathbb{R}^{L}$, and $\mathbf{q} \in \mathbb{R}^{m}$.

Step 1 (*Worst case reformulation*). Notice that (3) is equivalent to the following worst case reformulation:

$$\mathbf{a}^{\top}\mathbf{x} + \max_{\boldsymbol{\zeta} : \mathbf{D}\boldsymbol{\zeta} + \mathbf{q} \geq 0} (\mathbf{P}^{\top}\mathbf{x})^{\top}\boldsymbol{\zeta} \leq d. \tag{4}$$

Step 2 (*Duality*). We take the dual of the inner maximization problem in (4). The inner maximization problem and its dual yield the same optimal objective value by strong duality. Therefore, (4) is equivalent to

$$\mathbf{a}^{\top}\mathbf{x} + \min_{\mathbf{w}}\{\mathbf{q}^{\top}\mathbf{w} : \mathbf{D}^{\top}\mathbf{w} = -\mathbf{P}^{\top}\mathbf{x}, \ \mathbf{w} \geq \mathbf{0}\} \leq d. \tag{5}$$

Step 3 (*RC*). It is important to point out that we can omit the minimization term in (5), since it is sufficient that the constraint holds for at least one $\mathbf{w}$. Hence, the final formulation of the RC becomes

$$\exists \ \mathbf{w} : \ \mathbf{a}^{\top}\mathbf{x} + \mathbf{q}^{\top}\mathbf{w} \leq d, \quad \mathbf{D}^{\top}\mathbf{w} = -\mathbf{P}^{\top}\mathbf{x}, \quad \mathbf{w} \geq \mathbf{0}. \tag{6}$$

Note that the constraints in (6) are linear in $\mathbf{x} \in \mathbb{R}^{n}$ and $\mathbf{w} \in \mathbb{R}^{m}$.

Table 1 presents the tractable robust counterparts of an uncertain linear optimization problem for different classes of uncertainty sets. These robust counterparts are derived using the three steps that are described above. However, we need conic duality instead of LP duality in Step 2 to derive the tractable robust counterparts for the conic uncertainty set; see the fourth row of Table 1. Similarly, to derive the tractable RC for an uncertainty region specified by general convex constraints, i.e., in the fifth row of Table 1, we need Fenchel duality in Step 2; see Rockafellar (1997) for details on Fenchel duality, and Ben-Tal et al. (2014) for the formal proof of the associated RC reformulation. Notice that each RC constraint has a positive safeguard in the constraint left-hand side, e.g., $\|\mathbf{P}^{\top}\mathbf{x}\|_{1}$, $\|\mathbf{P}^{\top}\mathbf{x}\|_{2}$, and $\mathbf{q}^{\top}\mathbf{w}$; see the tractable RCs in the third column of Table 1. These safeguards represent the level of robustness that we introduce to the constraints. Note the equivalence between the robust counterparts of the polyhedral and the conic uncertainty set when K is the nonnegative orthant.

Nonlinear problems. Table 1 focuses on uncertain linear optimization problems, i.e., both linear in the decision variables and the uncertain parameters. Notice that, different than the presented results,

Table 1 Tractable reformulations for the uncertain constraint $[(\mathbf{a} + \mathbf{P}\boldsymbol{\zeta})^\top \mathbf{x} \leq d \quad \forall \boldsymbol{\zeta} \in \mathcal{Z}]$

for different types of uncertainty sets

Uncertainty	$\mathcal{Z}$	Robust Counterpart	Tractability
Box	$\|\boldsymbol{\zeta}\|_\infty \leq 1$	$\mathbf{a}^\top \mathbf{x} + \|\mathbf{P}^\top \mathbf{x}\|_1 \leq d$	LP
Ellipsoidal	$\|\boldsymbol{\zeta}\|_2 \leq 1$	$\mathbf{a}^\top \mathbf{x} + \|\mathbf{P}^\top \mathbf{x}\|_2 \leq d$	CQP
Polyhedral	$\mathbf{D}\boldsymbol{\zeta} + \mathbf{q} \geq 0$	$\begin{cases} \mathbf{a}^\top \mathbf{x} + \mathbf{q}^\top \mathbf{w} \leq d \\ \mathbf{D}^\top \mathbf{w} = -\mathbf{P}^\top \mathbf{x} \\ \mathbf{w} \geq \mathbf{0} \end{cases}$	LP
Cone (closed, convex, pointed)	$\mathbf{D}\boldsymbol{\zeta} + \mathbf{q} \in K$	$\begin{cases} \mathbf{a}^\top \mathbf{x} + \mathbf{q}^\top \mathbf{w} \leq d \\ \mathbf{D}^\top \mathbf{w} = -\mathbf{P}^\top \mathbf{x} \\ \mathbf{w} \in K^* \end{cases}$	Conic Opt.
Convex cons.	$h_k(\boldsymbol{\zeta}) \leq 0 \quad \forall k$	$\begin{cases} \mathbf{a}^\top \mathbf{x} + \sum_k u_k h_k^*\left(\frac{\mathbf{w}^\mathbf{k}}{u_k}\right) \leq d \\ \sum_k \mathbf{w}^\mathbf{k} = \mathbf{P}^\top \mathbf{x} \\ \mathbf{u} \geq \mathbf{0} \end{cases}$	Convex Opt.

(∗) h^* is the convex conjugate function, i.e, $h^*(x) = \sup_y\{x^\top y - h(y)\}$; and K^* is the dual cone of K;
Table taken from Ben-Tal et al. (2014)

the original uncertain optimization problem can be nonlinear in the optimization variables and/or the uncertain parameters; for more detailed treatment of such problems that are nonlinear in the uncertain parameters, we refer to Ben-Tal et al. (2009, pp. 383–388) and Ben-Tal et al. (2014).

Adversarial approach. If the robust counterpart cannot be written as or approximated by a tractable reformulation, we advocate to perform the so-called *adversarial approach.* The adversarial approach starts with a finite set of scenarios $S_i \subset \mathcal{Z}_i$ for the uncertain parameter in constraint i. For example, at the start, S_i only contains the nominal scenario. Then, the robust optimization problem, which has a finite number of constraints since $\mathcal{Z}_i$ has been replaced by S_i, is solved. If the resulting solution is robust feasible, we have found the robust optimal solution. If that is not the case, we can find a scenario for the uncertain parameter that makes the last found solution infeasible, e.g., we can search for the scenario that maximizes the infeasibility. We add this scenario to S_i, and solve the resulting robust optimization problem, and so on. For a more detailed description, we refer to Bienstock and Özbay (2008). It appeared that this simple approach often converges to optimality in a few number of iterations. The advantage of this approach is that solving the robust optimization problem with S_i instead of $\mathcal{Z}_i$ in each iteration, preserves the structure of the original optimization problem. Only constraints of the same type are added, since constraint i should hold for all scenarios in S_i. This approach could be faster than reformulating, even for polyhedral uncertainty sets. See Bertsimas et al. (2014) for a comparison. Alternatively, if the probability distribution of the uncertain parameter is known, one may also use the *randomized sampling* of the uncertainty set proposed by Calafiore and Campi (2005). The randomized approach substitutes the infinitely many robust constraints with a finite set of constraints that are randomly sampled. It is shown that such a randomized approach

is an accurate approximation of the original uncertain problem provided that a sufficient number of samples is drawn; see Campi and Garatti (2008, Theorem 1).

Pareto efficiency. Iancu and Trichakis (2014) discovered that "the inherent focus of RO on optimizing performance only under worst case outcomes might leave decisions un-optimized in case a non worst case scenario materialized". Therefore, the "classical" RO framework might lead to a Pareto inefficient solution; i.e., an alternative robust optimal solution may guarantee an improvement in the objective or slack size for (at least) one scenario without deteriorating it in other scenarios. Given a robust optimal solution, Iancu and Trichakis propose optimizing a new problem to find a solution that is Pareto efficient. In this new problem, the objective is optimized for a scenario in the interior of the uncertainty set, e.g., for the nominal scenario, while the worst case objective is constrained to be not worse than the robust optimal objective value. For more details on Pareto efficiency in robust linear optimization we refer to Iancu and Trichakis (2014).

2.3. Adjustable robust optimization

In multistage optimization, the first assumption [A.1] of the RO paradigm, i.e., the decisions are "here and now", can be relaxed. For example, the amount a factory will produce next month is not a "here and now" decision, but a "wait and see" decision that will be taken based on the amount sold in the current month. Some decision variables can therefore be adjusted at a later moment in time according to a decision rule, which is a function of (some or all part of) the uncertain data. The *adjustable* RC (ARC) is given as follows:

$$\min_{\mathbf{x},\mathbf{y}(\cdot)}\{\mathbf{c}^\top\mathbf{x}:\ \mathbf{A}(\boldsymbol{\zeta})\mathbf{x}+\mathbf{B}\mathbf{y}(\boldsymbol{\zeta})\leq\mathbf{d}\quad\forall\boldsymbol{\zeta}\in\mathcal{Z}\},\tag{7}$$

where $\mathbf{x}\in\mathbb{R}^n$ is the first-stage "here and now" decision that is made before $\boldsymbol{\zeta}\in\mathbb{R}^L$ is realized, $\mathbf{y}\in\mathbb{R}^k$ denotes the second-stage "wait and see" decision that can be adjusted according to the actual data, and $\mathbf{B}\in\mathbb{R}^{m\times k}$ denotes a certain coefficient matrix (i.e., fixed recourse).

However, ARC is a complex problem unless we restrict the function $\mathbf{y}(\boldsymbol{\zeta})$ to specific classes; see Ben-Tal et al. (2009, Ch. 14) for details. In practice, $\mathbf{y}(\boldsymbol{\zeta})$ is often approximated by affine (or linear) decision rules:

$$\mathbf{y}(\boldsymbol{\zeta}):=\mathbf{y^0}+\mathbf{Q}\boldsymbol{\zeta},\tag{8}$$

because they yield computationally tractable *affinely* ARC (AARC) reformulations, where $\mathbf{y^0}\in\mathbb{R}^k$ and $\mathbf{Q}\in\mathbb{R}^{k\times L}$ are the coefficients in the decision rule, which are to be optimized. Eventually, the tractable reformulation of the constraints in (7):

$$\min_{\mathbf{x},\mathbf{y^0},\mathbf{Q}}\{\mathbf{c}^\top\mathbf{x}:\ \mathbf{A}(\boldsymbol{\zeta})\mathbf{x}+\mathbf{B}\mathbf{y^0}+\mathbf{B}\mathbf{Q}\boldsymbol{\zeta}\leq\mathbf{d}\quad\forall\boldsymbol{\zeta}\in\mathcal{Z}\}$$

can be derived in a similar vein by applying the three steps that are described above, since the problem is affine in the uncertain parameter $\boldsymbol{\zeta}$, and the decision variables $\mathbf{x}, \mathbf{y^0}$, and $\mathbf{Q}$.

To point out, ARO is less conservative than the classic RO approach, since it yields more flexible decisions that can be adjusted according to the realized portion of data at a given stage. More precisely, ARO yields optimal objective values that are at least as good as that of the standard RO approach. In addition, aside from introducing additional variables and constraints, the AARC does not introduce additional computational complexity to that of RO with fixed recourse, and it can be straightforwardly adopted to the classic RO framework, i.e., the AARC is a computationally tractable approach. Moreover, affine decision rules are optimal or near optimal for many practical cases; e.g., inventory management (Bertsimas et al. 2010). Last but not least, ARO has many applications in real-life, e.g., supply chain management (Ben-Tal et al. 2005), project management (Ben-Tal et al. 2009, Ex. 14.2.1), engineering (Ben-Tal and Nemirovski 2002), and so on.

Notice that we adopt affine decision rules in the ARC, but it is important to point out that tractable ARC reformulations for nonlinear decision rules also exist for specific classes; we refer to Ben-Tal et al. (2009, Ch. 14.3) and Georghiou et al. (2014).

Integer adjustable variables. A parametric decision rule, like the linear one in (8), cannot be used for integer adjustable variables, since we have then to enforce that the decision rule is integer for all $\boldsymbol{\zeta} \in \mathcal{Z}$. In Section 5 we propose a general way for dealing with adjustable integer variables similar to Bertsimas and Caramanis (2010). However, much more research is needed.

2.4. Robust optimization procedure

Now having introduced the general notation in RO and *adjustable* RO (ARO), we can give a procedure for applying RO in practice; see Table 2.

In the remainder of this paper, we describe the most important items at each step of this procedure. The associated section(s) for each step are reported in the last column of Table 2.

3. Choosing the uncertainty set

In this section we describe different possible uncertainty sets and their advantages and disadvantages. Often one wants to make a trade-off between robustness against each physical realization of the uncertain parameter and the size of the uncertainty set. A box uncertainty set that contains the full range of realizations for each component of $\boldsymbol{\zeta}$ is the most robust choice and guarantees that the constraint is never violated, but on the other hand there is only a small chance that all uncertain parameters take their worst case values. This has led to the development of smaller uncertainty sets that still guarantee that the constraint is "almost never" violated. Such guarantees are inspired by chance constraints, which are constraints that have to hold with at least a certain probability. Often the underlying probability distribution is not known, and one seeks a distributionally robust solution.

One application of RO is to provide a tractable safe approximation of the chance constraint in such cases, i.e., a tractable formulation that guarantees that the chance constraint holds:

$$\text{if } x \text{ satisfies } \mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x} \le d \quad \forall \boldsymbol{\zeta} \in \mathcal{U}_\varepsilon, \text{ then } x \text{ also satisfies } \mathbb{P}_{\boldsymbol{\zeta}}(\mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x} \le d) \ge 1 - \varepsilon. \tag{9}$$

For $\varepsilon = 0$, a chance constraint is a traditional robust constraint. The challenge is to determine the set $\mathcal{U}_\varepsilon$ for other values of ε. We distinguish between uncertainty sets for uncertain parameters and for uncertain probability vectors.

For uncertain parameters, many results are given in (Ben-Tal et al. 2009, Chapter 2). The simplest case is when the only knowledge about $\boldsymbol{\zeta}$ is that $||\boldsymbol{\zeta}||_\infty \le 1$. For this case, the box uncertainty set is the only set that can provide a probability guarantee (of $\varepsilon = 0$). When more information becomes available, such as bounds on the mean or variance, or knowledge that the probability distribution is symmetric or unimodal, smaller uncertainty sets become available. Ben-Tal et al. (2009, Table 2.3) list seven of these cases. Probability guarantees are only given when $||\boldsymbol{\zeta}||_\infty \le 1$, $\mathbb{E}(\boldsymbol{\zeta}) = \mathbf{0}$ and the components of $\boldsymbol{\zeta}$ are independent. We mention the uncertainty sets that are used in practice when box uncertainty is found to be too pessimistic. The first is an ellipsoid (Ben-Tal et al. 2009, Proposition 2.3.1), possibly intersected with a box (Ben-Tal et al. 2009, Proposition 2.3.3):

$$\mathcal{U}_\varepsilon = \{\boldsymbol{\zeta} : ||\boldsymbol{\zeta}||_2 \le \Omega \quad ||\boldsymbol{\zeta}||_\infty \le 1\}, \tag{10}$$

Table 2

Practical RO procedure		Ref. section(s)
Step 0:	Solve the nominal problem.	-
Step 1:	**a)** Determine the uncertain parameters. **b)** Determine the uncertainty set.	§ 3
Step 2:	Check robustness of the nominal solution. **IF** the nominal solution is robust "enough" **THEN** stop.	§ 9
Step 3:	**a)** Determine the adjustable variables. **b)** Determine the type of decision rules.	§§ 5 §§ 4, 10
Step 4:	Formulate the robust counterpart.	§§ 6, 7, 8
Step 5:	Solve the (adjustable) robust counterpart via an exact or approximate tractable reformulation, or via the adversarial approach.	§ 2
Step 6:	Check quality of the robust solution. **IF** the solution is "too conservative" **THEN** go to **Step 1b** or **Step 3**.	§ 9

where $\varepsilon = \exp(-\Omega^2/2)$. The second is a polyhedral set (Ben-Tal et al. 2009, Proposition 2.3.4), called budgeted uncertainty set or the "Bertsimas and Sim" uncertainty set (Bertsimas and Sim 2004):

$$\mathcal{U}_\varepsilon = \{\boldsymbol{\zeta} : ||\boldsymbol{\zeta}||_1 \le \Gamma \quad ||\boldsymbol{\zeta}||_\infty \le 1\}, \tag{11}$$

where $\boldsymbol{\zeta} \in \mathbb{R}^L$, and $\varepsilon = \exp(-\Gamma^2/(2L))$. A stronger bound is provided in (Bertsimas and Sim 2004). This set has the interpretation that (integer) Γ controls the number of elements of $\boldsymbol{\zeta}$ that may deviate from their nominal values. (10) leads to better objective values for a fixed ε compared to (11), but gives rise to a CQP for an uncertain LP while (11) results in an LP and is therefore more tractable from a computational point of view.

Bandi and Bertsimas (2012) propose uncertainty sets based on the central limit theorem. When the components of $\boldsymbol{\zeta} \in \mathbb{R}^L$ are independent and identically distributed with mean μ and variance σ^2, the uncertainty set is given by:

$$\mathcal{U}_\varepsilon = \left\{\boldsymbol{\zeta} : \left|\sum_{i=1}^{L} \zeta_i - L\mu\right| \le \rho\sqrt{n}\sigma\right\},$$

where ρ controls the probability of constraint violation $1 - \varepsilon$. Bandi and Bertsimas also show variations on $\mathcal{U}_\varepsilon$ that incorporate correlations, heavy tails, or other distributional information. The advantage of this uncertainty set is its tractability, since the robust counterpart of an LP with this uncertainty set is also LP. A disadvantage of this uncertainty set is that it is unbounded for $L > 1$, since one component of $\boldsymbol{\zeta}$ can be increased to an arbitrarily large number (while simultaneously decreasing a different component). This may lead to intractability of the robust counterpart or to trivial solutions. In order to avoid infeasibility, it is necessary to define separate uncertainty sets for each constraint, where the summation runs only over the elements of $\boldsymbol{\zeta}$ that appear in that constraint. Alternatively, it may help to take the intersection of $\mathcal{U}_\varepsilon$ with a box.

Bertsimas et al. (2013a) show how to construct uncertainty sets based on historical data and statistical tests. An advantage compared to the aforementioned approach, is that it requires fewer assumptions. For example, independence of the components of $\boldsymbol{\zeta}$ is not required. They have included a section with practical recommendations, which is too extensive to discuss here.

Bertsimas and Brown (2009) show how to construct uncertainty sets based on coherent risk measures that model the preferences of the decision maker, and on past observations of the uncertain data. Starting from a representation theorem of coherent risk measures, they make a natural link to RO. For a specific class of risk measures, this gives rise to polyhedral uncertainty sets.

We now focus on uncertain probability vectors, i.e,. $\mathcal{U} \subset \Delta^{L-1} = \{\mathbf{p} \in \mathbb{R}^L : \mathbf{p} \ge \mathbf{0}, \sum_{i=1}^{L} p_i = 1\}$. These appear, e.g., in a constraint on expected value or variance. Ben-Tal et al. (2013) construct uncertainty sets based on ϕ-divergence. The ϕ-divergence between the vectors $\mathbf{p}$ and $\mathbf{q}$ is:

$$I_\phi(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{L} q_i \phi\left(\frac{p_i}{q_i}\right),$$

where ϕ is the (convex) ϕ-divergence function. Let $\mathbf{p}$ denote a probability vector and let $\mathbf{q}$ be the vector with observed frequencies when N items are sampled according to $\mathbf{p}$. Under certain regularity conditions,

$$\frac{2N}{\phi''(1)} I_\phi(\mathbf{p}, \mathbf{q}) \xrightarrow{d} \chi^2_{L-1} \text{ as } N \to \infty.$$

This motivates the use of the following uncertainty set:

$$\mathcal{U}_\varepsilon = \{\mathbf{p} : \mathbf{p} \geq \mathbf{0}, \quad \mathbf{e}^\top \mathbf{p} = 1, \quad \frac{2N}{\phi''(1)} I_\phi(\mathbf{p}, \hat{\mathbf{p}}) \leq \chi^2_{L-1;1-\varepsilon}\},$$

where $\hat{\mathbf{p}}$ is an estimate of $\mathbf{p}$ based on N observations, and $\chi^2_{L-1;1-\varepsilon}$ is the $1-\varepsilon$ percentile of the χ^2 distribution with $L-1$ degrees of freedom. The uncertainty set contains the true $\mathbf{p}$ with (approximate) probability $1-\varepsilon$. Ben-Tal et al. (2013) give many examples of ϕ-divergence functions that lead to tractable robust counterparts.

An alternative to ϕ-divergence is using the Anderson-Darling test to construct the uncertainty set (Ben-Tal et al. 2014, Ex. 15).

For multistage problems, Goh and Sim (2010) show how to construct uncertainty sets based on, e.g., the covariance matrix or bounds on the expected value. They show how these sets can be used for optimizing piecewise-linear decision rules.

We conclude this section by pointing out a mistake that is sometimes made regarding the interpretation of the uncertainty set. Sometimes the set $\mathcal{U}$ in (9) is constructed such that it contains the true parameter with probability ε. This provides a much stronger probability guarantee than one expects. For example, let the uncertain parameter $\boldsymbol{\zeta} \in \mathbb{R}^L$ with $L = 10$ have independent components with mean 0 and variance 1. Then, $\boldsymbol{\zeta}^\top \boldsymbol{\zeta} \sim \chi^2_L$, so the set $\mathcal{U}_\varepsilon = \{\boldsymbol{\zeta} : ||\boldsymbol{\zeta}||_2 \leq \sqrt{\chi^2_{L;1-\varepsilon}}\}$ contains $\boldsymbol{\zeta}$ with probability $1-\varepsilon$. Consequently, the logical implication (9) holds. However, the probability in (9) is much larger than ε, since the constraint also holds for the "good" realizations of the uncertain parameter outside the uncertainty set. For example, the singleton $\mathcal{U}_\varepsilon = \{\mathbf{0}\}$ satisfies $\mathbb{P}(\boldsymbol{\zeta} \in \mathcal{U}_\varepsilon) = 0$, but the probability on the right hand side of (9) is 0.5 if the components of $\boldsymbol{\zeta}$ are independent and have a median of $\mathbf{0}$. To see this, note that $\mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x}$ is an affine function of $\boldsymbol{\zeta}$, and therefore has a median of $d = \mathbf{a}(\mathbf{0})^\top \mathbf{x}$. So, the probability that $\mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x}$ becomes larger than d equals the probability that it becomes smaller. In order to construct the correct set $\mathcal{U}_\varepsilon$, we first write the explicit chance constraint. Since $(\mathbf{a} + \mathbf{P}\boldsymbol{\zeta})^\top \mathbf{x} \leq d$ is equivalent to $(\mathbf{a})^\top \mathbf{x} + (\mathbf{P}^\top \mathbf{x})^\top \boldsymbol{\zeta} \leq d$, and since the term $(\mathbf{P}^\top \mathbf{x})^\top \boldsymbol{\zeta}$ follows a normal distribution with mean 0 and standard deviation $||\mathbf{P}^\top \mathbf{x}||_2$, the chance constraint can explicitly be formulated as $(\mathbf{a})^\top \mathbf{x} + z_{1-\varepsilon} ||\mathbf{P}^\top \mathbf{x}||_2 \leq d$, where $z_{1-\varepsilon}$ is the $1-\varepsilon$ percentile of the normal distribution. This is the robust counterpart of the original linear constraint with ellipsoidal uncertainty and a radius of $z_{1-\varepsilon}$. The value $z_{1-\varepsilon} = 9.3$ coincides with $\varepsilon \approx 7.0 \cdot 10^{-21}$. So, while one thinks to construct a set that makes the constraint hold in 50% of the cases, the set actually makes the constraint hold in almost all cases. To make the chance constraint hold with probability $1-\varepsilon$, the radius of the ellipsoidal uncertainty set is $z_{1-\varepsilon}$ instead of $\sqrt{\chi^2_{L;1-\varepsilon}}$. These only coincide for $L = 1$.

4. Linearly adjustable robust counterpart: linear in what?

Tractable examples of decision rules used in ARO are linear (or affine) decision rules (AARC) (Ben-Tal et al. 2009, Chapter 14) or piecewise linear decision rules (Chen et al. 2008); see also Section 2.3. The AARC was introduced by Ben-Tal et al. (2004) as a computationally tractable method to handle adjustable variables. In the following constraint:

$$(\mathbf{a} + \mathbf{P}\boldsymbol{\zeta})^\top \mathbf{x} + \mathbf{b}^\top \mathbf{y} \le d \qquad \forall \boldsymbol{\zeta} \in \mathcal{Z},$$

$\mathbf{y}$ is an adjustable variable whose value may depend on the realization of the uncertain $\boldsymbol{\zeta}$, while $\mathbf{b}$ does not depend on $\boldsymbol{\zeta}$ (fixed recourse). There are two different AARCs for this constraint:

AARC 1. $\mathbf{y}$ is linear in $\boldsymbol{\zeta}$ (e.g., see Ben-Tal et al. (2004) and Ben-Tal et al. (2009, Chapter 14)),

or

AARC 2. $\mathbf{y}$ is linear in $\mathbf{a} + \mathbf{P}\boldsymbol{\zeta}$ (e.g., see Roelofs and Bisschop (2012, Chapter 20.4)).

Note that AARC 2 is as least as conservative as AARC 1, since the linear transformation of $\boldsymbol{\zeta} \mapsto \mathbf{a} + \mathbf{P}\boldsymbol{\zeta}$ can only lead to loss of information, and that both methods are equivalent if the linear transformation is injective on $\mathcal{Z}$. The choice for a particular method may be influenced by four factors: (i) the availability of information. An actual decision cannot depend on $\boldsymbol{\zeta}$ if $\boldsymbol{\zeta}$ has not been observed. (ii) The number of variables in the final problem. AARC 1 leads to $|\boldsymbol{\zeta}|$ extra variables compared to the RC, whereas AARC 2 leads to $|\mathbf{a}|$ extra variables. (iii) Simplicity for the user. Often the user observes model parameters instead of the primitive uncertainty vector. (iv) For state or analysis variables one should always use the least conservative method.

The practical issue raised in the first factor (availability of information) has been addressed with a information base matrix $\mathbf{P}$. Instead of being linear in $\boldsymbol{\zeta}$, $\mathbf{y}$ can be made linear in $\mathbf{P}\boldsymbol{\zeta}$. We give one example where uncertain demand is observed. Suppose there are two time periods and three possible scenarios for demand time period one and two, namely $(10, 10)^\top$, $(10, 11)^\top$ and $(11, 11)^\top$. So, the uncertainty set of the demand vector is the convex hull of these scenarios: $\{\mathbf{P}\boldsymbol{\zeta} : \boldsymbol{\zeta} \in \mathcal{Z}\}$ where $\mathbf{P}$ is the matrix with the scenarios as columns and $\mathcal{Z} = \Delta^2 = \{\boldsymbol{\zeta} \in \mathbb{R}^3 : \sum_{\ell=1}^3 \zeta_\ell = 1, \boldsymbol{\zeta} \ge 0\}$ is the standard simplex in $\mathbb{R}^3$. If the observed demand for time period one is 10, it is not possible to distinguish between $\boldsymbol{\zeta} = (1, 0, 0)^\top$ and $\boldsymbol{\zeta} = (0, 1, 0)^\top$. So, a decision for time period two can be modeled either as AARC 1 with $\mathbf{P} = (1, 1, 0)$ or as AARC 2. The latter leads to a decision rule that is easier to interpret, since it directly relates to previously observed demand.

5. Adjustable integer variables

Ben-Tal et al. (2009, Chapter 14) use parametric decision rules for adjustable continuous variables. However, their novel techniques "generally" cannot be applied for adjustable integer variables. In the

literature two alternative approaches have been proposed. Bertsimas and Georghiou (2013) introduced an iterative method to treat adjustable *binary* variables as piecewise constant functions. The approach by Bertsimas and Caramanis (2010) is different and is based on splitting the uncertainty region into smaller subsets, where each subset has its own binary decision variable (see also Vayanos et al. (2011), and Hanasusanto et al. (2014)). In this section, we briefly show this last method to treat adjustable integer variables, and show how the average behavior can be improved. We use the following notation for the general RC problem:

$$(\text{RC1}) \ \max_{\mathbf{x,y,z}} \ c(\mathbf{x,y,z})$$
$$\text{s.t. } \mathbf{A}(\boldsymbol{\zeta})\,\mathbf{x} + \mathbf{B}(\boldsymbol{\zeta})\,\mathbf{y} + \mathbf{C}(\boldsymbol{\zeta})\,\mathbf{z} \le \mathbf{d}, \quad \forall \boldsymbol{\zeta} \in \mathcal{Z},$$

where $\mathbf{x} \in \mathbb{R}^{n_1}$ and $\mathbf{y} \in \mathbb{Z}^{n_2}$ are "here and now" variables, i.e., decisions on them are made before the uncertain parameter $\boldsymbol{\zeta}$, contained in the uncertainty set $\mathcal{Z} \subseteq \mathbb{R}^{L}$, is revealed; $\mathbf{z} \in \mathbb{Z}^{n_3}$ is a "wait and see" variable, i.e., the decision on $\mathbf{z}$ is made after observing (part of) the value of the uncertain parameter. $\mathbf{A}(\boldsymbol{\zeta}) \in \mathbb{R}^{m_1 \times n_1}$ and $\mathbf{B}(\boldsymbol{\zeta}) \in \mathbb{R}^{m_2 \times n_2}$ are the uncertain coefficient matrices of the "here and now" variables. Notice that the integer "wait and see" variable $\mathbf{z}$ has an uncertain coefficient matrix $\mathbf{C}(\boldsymbol{\zeta}) \in \mathbb{R}^{m_3 \times n_3}$. So, unlike the "classic" parametric method, this approach can handle uncertainties in the coefficients of the integer "wait and see" variables. For the sake of simplicity, we assume that the uncertain coefficient matrices to be linear in $\boldsymbol{\zeta}$ and, without loss of generality, $c(\mathbf{x,y,z})$ is the certain linear objective function.

To model the *adjustable* RC (ARC) with integer variables, we first divide the given uncertainty set $\mathcal{Z}$ into m disjoint, excluding the boundaries, subsets ($\mathcal{Z}_i, i = 1, \ldots, m$):

$$\mathcal{Z} = \bigcup_{i \in \{1, \ldots, m\}} \mathcal{Z}_i,$$

and we introduce additional integer variables $\mathbf{z_i} \in \mathbb{Z}^{n_3}$ $(i = 1, \ldots, m)$ that model the decision in $\mathcal{Z}_i$. Then, we replicate the uncertain constraint and the objective function in (RC1) for each $\mathbf{z_i}$ and the uncertainty set $\mathcal{Z}_i$ as follows:

$$(\text{ARC1}) \ \max_{\mathbf{x,y,Z},t} \ t$$
$$\text{s.t. } c(\mathbf{x,y,z_i}) \ge t \qquad\qquad \forall i \in \{1, \ldots, m\} \qquad (12)$$
$$\mathbf{A}(\boldsymbol{\zeta})\,\mathbf{x} + \mathbf{B}(\boldsymbol{\zeta})\,\mathbf{y} + \mathbf{C}(\boldsymbol{\zeta})\,\mathbf{z_i} \le \mathbf{d} \qquad \forall \boldsymbol{\zeta} \in \mathcal{Z}_i, \forall i \in \{1, \ldots, m\}.$$

Note that (ARC1) is more flexible than the non-adjustable RC (RC1) in selecting the values of integer variables, since it has a specific decision $\mathbf{z_i}$ for each subset $\mathcal{Z}_i$. Therefore, (ARC1) yields a robust optimal objective that is at least as good as (RC1).

Pareto inefficiency (see Section 2.2) is also an issue in (ARC1). Hence, we must take the individual performance of the m subsets into account to have a better understanding of the general performance of (ARC1). Similar to Iancu and Trichakis (2014), we apply a reoptimization procedure to improve the average performance of (ARC1). More precisely, we first solve (ARC1) and find the optimal objective t^*. Then, we solve the following problem:

$$
\text{(re-opt)} \max_{\mathbf{x},\mathbf{y},\mathbf{Z},\mathbf{t}} \sum_{i \in \{1,\ldots,m\}} t_i
$$
$$
\text{s.t. } t_i \geq t^* \qquad\qquad\qquad\qquad \forall i \in \{1,\ldots,m\}
$$
$$
c(\mathbf{x},\mathbf{y},\mathbf{z_i}) \geq t_i \qquad\qquad\qquad \forall i \in \{1,\ldots,m\}
$$
$$
\mathbf{A}(\boldsymbol{\zeta})\,\mathbf{x} + \mathbf{B}(\boldsymbol{\zeta})\,\mathbf{y} + \mathbf{C}(\boldsymbol{\zeta})\,\mathbf{z_i} \leq \mathbf{d} \qquad \forall \boldsymbol{\zeta} \in \mathcal{Z}_i, \forall i \in \{1,\ldots,m\},
$$

that optimizes (i.e., maximizes) the slacks in (12), while the worst case objective value t^* remains the same. Note that t_i's are the additional variables associated with the objectives values of the subsets; (re-opt) mimics a multi-objective optimization problem that assigns equal weights to each objective, and finds Pareto efficient robust solutions.

Example

Here we compare the optimal objective values of (RC1), (ARC1), and (ARC1) with (re-opt) via a toy example. For the sake of exposition, we exclude continuous variables in this example. The non-adjustable RC is given as follows:

$$
\max_{(w,\mathbf{z}) \in \mathbb{Z}_+^3} 5w + 3z_1 + 4z_2
$$
$$
\text{s.t. } (1 + \zeta_1 + 2\zeta_2)w + (1 - 2\zeta_1 + \zeta_2)z_1 + (2 + 2\zeta_1)z_2 \leq 18 \quad \forall \boldsymbol{\zeta} \in \text{Box} \qquad (13)
$$
$$
(\zeta_1 + \zeta_2)w + (1 - 2\zeta_1)z_1 + (1 - 2\zeta_1 - \zeta_1)z_2 \leq 16 \qquad \forall \boldsymbol{\zeta} \in \text{Box},
$$

where $\text{Box} = \{\zeta : -1 \leq \zeta_1 \leq 1, -1 \leq \zeta_2 \leq 1\}$ is the given uncertainty set, and w, z_1, and z_2 are nonnegative integer variables. In addition, we assume that z_1 and z_2 are adjustable on ζ_1; i.e., the decision on these variables is made after ζ_1 is being observed. Next, we divide the uncertainty set into two subsets:

$$
\mathcal{Z}_1 = \{(\zeta_1, \zeta_2) : -1 \leq \zeta_1 \leq 0, -1 \leq \zeta_2 \leq 1\}
$$
$$
\mathcal{Z}_2 = \{(\zeta_1, \zeta_2) : 0 \leq \zeta_1 \leq 1, -1 \leq \zeta_2 \leq 1\}.
$$

Then ARC of (13) is:

$$
\text{(Ex:ARC)} \max_{t,w,\mathbf{Z}} t
$$
$$
\text{s.t. } 5w + 3z_1^i + 4z_2^i \geq t \qquad\qquad\qquad\qquad \forall i \in \{1,\ldots,m\}
$$
$$
(1 + \zeta_1 + 2\zeta_2)w + (1 - 2\zeta_1 + \zeta_2)z_1^i + (2 + 2\zeta_1)z_2^i \leq 18 \quad \forall \boldsymbol{\zeta} \in \mathcal{Z}_i, \forall i \in \{1,\ldots,m\}
$$
$$
(\zeta_1 + \zeta_2)w + (1 - 2\zeta_1)z_1^i + (1 - 2\zeta_1 - \zeta_1)z_2^i \leq 16 \qquad \forall \boldsymbol{\zeta} \in \mathcal{Z}_i, \forall i \in \{1,\ldots,m\},
$$

where $t \in \mathbb{R}$, $w \in \mathbb{Z}_+$, $\mathbf{Z} \in \mathbb{Z}_+^{2 \times m}$, and $m = 2$ since we have two subsets. Table 3 presents the optimal solutions of RC and ARC problems.

Table 3   RC vs ARC

Method	Obj.	w	$\mathbf{z}$
RC	29	1	$(z_1, z_2) = (4, 3)$
ARC	31	0	$(z_1^1, z_2^1, z_1^2, z_2^2) = (0, 8, 5, 4)$

The numerical results show that using the adjustable reformulation we improve the objective value of the non-adjustable problem by 7%. On the other hand, if we assume that z_1 and z_2 are adjustable on ζ_2 (but not on ζ_1), and we modify the uncertainty subsets $\mathcal{Z}_1$ and $\mathcal{Z}_2$ accordingly, then RC and ARC yield the same objective 29. This shows that the value of information of ζ_1 is higher than that of ζ_2.

Next we compare the average performance of ARC and the second stage optimization problem (re-opt) that is given by:

$$\max_{\mathbf{t}, w, \mathbf{Z}} \sum_{i \in \{1, \ldots, m\}} t_i$$

$$\text{s.t.} \quad 5w + 3z_1^i + 4z_2^i \geq t_i, \quad t_i \geq t^* \qquad \qquad \forall i \in \{1, \ldots, m\}$$

$$(1 + \zeta_1 + 2\zeta_2)w + (1 - 2\zeta_1 + \zeta_2)z_1^i + (2 + 2\zeta_1)z_2^i \leq 18 \qquad \forall \boldsymbol{\zeta} \in \mathcal{Z}_i, \forall i \in \{1, \ldots, m\}$$

$$(\zeta_1 + \zeta_2)w + (1 - 2\zeta_1)z_1^i + (1 - 2\zeta_1 - \zeta_2)z_2^i \leq 16 \qquad \forall \boldsymbol{\zeta} \in \mathcal{Z}_i, \forall i \in \{1, \ldots, m\},$$

where $\mathbf{t} \in \mathbb{R}^m$. For changing the number of subsets, we again split the uncertainty sets ($\mathcal{Z}_i, i = 1, \ldots, m$) on ζ_1 but not on ζ_2. The numerical results are presented in Table 4.

Table 4   ARC vs re-opt for varying number of subsets

# Subsets	Worst Case Obj. Values per Subset		W.-C. Average	
	ARC	re-opt	ARC	re-opt
1	29	29	29	29.0
2	(32, 31*)	(34, 31*)	31.5	32.5
3	(33, 30*, 32)	(49, 30*, 35)	31.6	38.0
4	(33, 31*, 32, 32)	(64, 34, 31*, 54)	32	45.7
5	(33, 30*, 30*, 32, 32)	(80, 40, 30*, 33, 66)	31.4	49.8
8	(32, 32, 32, 34, 31*, 33, 33, 33)	(128, 64, 40, 34, 31* 36, 54, 108)	32.5	61.8
10	(32, 32, 32, 32, 34, 31*, 33, 33, 33, 33)	(160, 80, 52, 40, 34, 31*, 33, 45, 66, 135)	32.5	64.3

($*$) denotes the worst case (w.-c.) objective value over all subsets

The first column of the table presents the number of subsets used in ARC, and we assume that the domain of ζ_1 is divided into equally sized intervals (e.g., if the number of subsets is three, then the intervals are $-1 \leq \zeta_1 \leq -0.33, -0.33 \leq \zeta_1 \leq 0.33$, and $0.33 \leq \zeta_1 \leq 1$). The second column reports objective values of the subproblems associated with the subsets in ARC. The third column presents the objective values of the subproblems when we apply (re-opt). The fourth and fifth columns show the averages of the results in columns two and three. As anticipated, intra row comparisons show that ARC and (re-opt) yield the same worst case performance for a fixed number of subsets, and (re-opt) yields significantly better average performances than ARC. Moreover, the average performance improves with the number of subsets. Notice that the average performance of the RC solution is not reported in Table 4 because it has the same average performance, that is 29, for any given number of subsets. Nevertheless, it is important to see the significant average objective value improvement made by ARC with (re-opt) for the "fixed" performance of the RC. Last but not least, the optimal objective value 31, which is obtained when the number of subsets is two, four, eight and ten in Table 4, is the global optimal of the ARC; for details on optimality see the following section where this example will be revisited.

Optimality

To quantify how far is the optimal objective value (t^*) of (ARC1) from that of the best possible solution, we need to define an efficient lower bound (or an upper bound for a maximization problem) for the best objective. One way of finding such a bound is by solving (RC1), where the uncertainty set $\mathcal{Z}$ is replaced with a finite subset (denoted by $\hat{\mathcal{Z}}$), and where each scenario from $\hat{\mathcal{Z}}$ has a separate second-stage decision (Hadjiyiannis et al. 2011, Bertsimas and Georghiou 2013, Postek 2013). The optimal objective value of such a formulation is always a lower bound for the best possible objective value, since it is an optimal (i.e., not restricted to an affine form) adjustable solution for a smaller uncertainty set. More precisely, the lower bound problem is given as follows:

$$
\begin{aligned}
\text{(BRC)} \quad \min_{\mathbf{x}, \mathbf{y}, \mathbf{z}^{(\zeta)}, t^{\text{lb}}} \quad & t^{\text{lb}} \\
\text{s.t.} \quad & c(\mathbf{x}, \mathbf{y}, \mathbf{z}^{(\zeta)}) \leq t^{\text{lb}} && \forall \zeta \in \hat{\mathcal{Z}} \\
& \mathbf{A}(\boldsymbol{\zeta})\,\mathbf{x} + \mathbf{B}(\boldsymbol{\zeta})\,\mathbf{y} + \mathbf{C}(\boldsymbol{\zeta})\,\mathbf{z}^{(\zeta)} \leq \mathbf{d} && \forall \zeta \in \hat{\mathcal{Z}}
\end{aligned}
$$

where $\hat{\mathcal{Z}}$ is a finite subset of $\mathcal{Z}$, as explained above. Now the question that has to be answered is: how to construct $\hat{\mathcal{Z}}$ efficiently? Postek (2013) proposes to first find the optimal solution of (ARC1) for a given number of subsets, and then taking $\hat{\mathcal{Z}}$ as uncertain parameters that maximizes the left-hand side of at least one constraint. For additional details on improving the lower bound we refer to Postek (2013, Chapter 4.2).

Example (Ex:ARC) revisited. The solution of (Ex:ARC) for two subsets (i.e., $m = 2$) is given in the second row of Table 3. The associated finite "worst case" subset for this solution is $\hat{\mathcal{Z}} = \{(0,1),(0,-1)\}$, and the upper bound for the best possible worst case objective is $t^{\text{ub}} = 31$ (this is obtained by solving the upper bound reformulation of (BRC) for $\hat{\mathcal{Z}}$). Therefore, the optimal objective value of (Ex:ARC) is bounded above by 31 for any given number of subsets. So, the solution found for two subsets (see Table 4) is optimal w.r.t. the worst case.

Tractability

It is important to point out that our adjustable reformulation and the "non-adjustable" RC have the same "general" mathematical complexity, but the adjustable reformulation increases the number of variables and constraints by a factor m (the number of subsets), so that if the number of integer variables is high (say a few hundreds) then the resulting adjustable RC may be intractable. Dividing the main uncertainty set $\mathcal{Z}$ into more subsets $\mathcal{Z}_i$ may improve the objective value by giving more freedom in making adjustable decisions, but the decision maker should make the tradeoff between optimality and computational complexity.

6. Robust counterparts of equivalent deterministic problems are not necessarily equivalent

In this section we show that the robust counterparts of equivalent deterministic problems are not always equivalent. The message in this section is thus that one has to be careful with reformulating optimization problems, since the corresponding robust counterparts may not be the same.

Let us start with a few simple examples. The first one is similar to the example in Ben-Tal et al. (2009, p. 13). Consider the following constraint:

$$(2 + \zeta)x_1 \leq 1,$$

where ζ is an (uncertain) parameter. This constraint is equivalent to:

$$\begin{cases} (2 + \zeta)x_1 + s = 1 \\ s \geq 0. \end{cases}$$

However, the robust counterparts of these two constraint formulations, i.e.

$$(2 + \zeta)x_1 \leq 1 \quad \forall \zeta \; : \; |\zeta| \leq 1, \tag{14}$$

and

$$\begin{cases} (2 + \zeta)x_1 + s = 1 \quad \forall \zeta \; : \; |\zeta| \leq 1 \\ s \geq 0, \end{cases} \tag{15}$$

in which the uncertainty set for ζ is the set $\{\zeta : |\zeta| \le 1\}$, are not equivalent. It can easily be verified that the feasible set for robust constraint (14) is: $x_1 \le 1/3$, while for the robust constraint (15) this is $x_1 = 0$. The reason why (14) and (15) are not equivalent is that by adding the slack variable, the inequality becomes an equality that has to be satisfied for all values of the uncertain parameter, which is very restrictive. The general message is therefore: *do not introduce slack variables in uncertain constraints, unless they are adjustable like in Kuhn et al. (2011), and avoid uncertain equalities.*

Another example is the following constraint:

$$|x_1 - \zeta| + |x_2 - \zeta| \le 2,$$

which is equivalent to:

$$\begin{cases} y_1 + y_2 \le 2 \\ y_1 \ge x_1 - \zeta \\ y_1 \ge \zeta - x_1 \\ y_2 \ge x_2 - \zeta \\ y_2 \ge \zeta - x_2. \end{cases}$$

However, the robust versions of these two formulations, namely:

$$|x_1 - \zeta| + |x_2 - \zeta| \le 2 \quad \forall \, \zeta \, : \, |\zeta| \le 1, \tag{16}$$

and:

$$\begin{cases} y_1 + y_2 \le 2 \\ y_1 \ge x_1 - \zeta & \forall \zeta \, : \, |\zeta| \le 1 \\ y_1 \ge \zeta - x_1 & \forall \zeta \, : \, |\zeta| \le 1 \\ y_2 \ge x_2 - \zeta & \forall \zeta \, : \, |\zeta| \le 1 \\ y_2 \ge \zeta - x_2 & \forall \zeta \, : \, |\zeta| \le 1, \end{cases} \tag{17}$$

are not equivalent. Indeed, it can easily be checked that the set of feasible solutions for (16) is $(\theta, -\theta)$, $-1 \le \theta \le 1$, but the only feasible solution for (17) is $\mathbf{x} = (0, 0)$. The reason for this is that in (17) the uncertainty is split over several constraints, and since the concept of RO is constraint-wise, this leads to different problems, and thus different solutions. The following linear optimization reformulation, however, is equivalent to (16):

$$\begin{cases} x_1 - \zeta + x_2 - \zeta \le 2 & \forall \, \zeta \, : \, |\zeta| \le 1 \\ x_1 - \zeta + \zeta - x_2 \le 2 & \forall \, \zeta \, : \, |\zeta| \le 1 \\ \zeta - x_1 + x_2 - \zeta \le 2 & \forall \, \zeta \, : \, |\zeta| \le 1 \\ \zeta - x_1 + \zeta - x_2 \le 2 & \forall \, \zeta \, : \, |\zeta| \le 1. \end{cases} \tag{18}$$

The general rule therefore is: *do not split the uncertainty in one constraint over more constraints, unless the uncertainty is disjoint.* In particular do not use "definition variables" if this leads to such a splitting of the uncertainty.

In the remainder we give a general treatment of some often used reformulation tricks to reformulate nonlinear problems into linear ones, and discuss whether the robust counterparts are equivalent or not.

- **Maximum function.** Consider the following constraint:

$$\mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x} + \max_k \mathbf{b_k}(\boldsymbol{\zeta})^\top \mathbf{x} \le d(\boldsymbol{\zeta}) \quad \forall \boldsymbol{\zeta} \in \mathcal{Z},$$

where $\boldsymbol{\zeta} \in \mathcal{Z}$ is the uncertain parameter, and $\mathbf{a}(\boldsymbol{\zeta})$, $\mathbf{b_k}(\boldsymbol{\zeta})$, and $d(\boldsymbol{\zeta})$ are parameters that depend linearly on $\boldsymbol{\zeta}$. A more conservative reformulation for this constraint is:

$$\begin{cases} \mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x} + z \le d(\boldsymbol{\zeta}) & \forall \boldsymbol{\zeta} \in \mathcal{Z} \\ z \ge \mathbf{b_k}(\zeta)^\top \mathbf{x} & \forall k, \forall \boldsymbol{\zeta} \in \mathcal{Z}, \end{cases}$$

since the uncertainty is split over more constraints. The exact reformulation is:

$$\mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x} + \mathbf{b_k}(\boldsymbol{\zeta})^\top \mathbf{x} \le d(\boldsymbol{\zeta}) \quad \forall k, \forall \boldsymbol{\zeta} \in \mathcal{Z}.$$

Note that in many cases we have "a sum of max":

$$\mathbf{a}(\boldsymbol{\zeta})^\top \mathbf{x} + \sum_i \max_k \mathbf{b_{ik}}(\boldsymbol{\zeta})^\top \mathbf{x} \le d(\boldsymbol{\zeta}) \quad \forall \boldsymbol{\zeta} \in \mathcal{Z}.$$

Important examples that contain such constraints are production-inventory problems. We refer to Gorissen and den Hertog (2012) for an elaborate treatment on exact and approximate reformulations of such constraints.

- **Absolute value function.** Note that $|x| = \max\{x, -x\}$, and hence this is a special case of the max function, treated above.

- **Linear fractional program.** Consider the following robust linear fractional problem:

$$\begin{cases} \min_{\mathbf{x}} \max_{\boldsymbol{\zeta} \in \mathcal{Z}} \frac{\alpha(\boldsymbol{\zeta}) + \mathbf{c}(\boldsymbol{\zeta})^\top \mathbf{x}}{\beta(\boldsymbol{\zeta}) + \mathbf{d}(\boldsymbol{\zeta})^\top \mathbf{x}} \\ \text{s.t.} \sum_j a_{ij} x_j \ge b_i \quad \forall i \\ \mathbf{x} \ge \mathbf{0}, \end{cases} \tag{19}$$

where $\boldsymbol{\alpha}(\boldsymbol{\zeta})$, $\mathbf{c}(\boldsymbol{\zeta})$, $\boldsymbol{\beta}(\boldsymbol{\zeta})$, and $\mathbf{d}(\boldsymbol{\zeta})$ are parameters that depend linearly on $\boldsymbol{\zeta}$. Moreover, we assume that $\beta(\boldsymbol{\zeta}) + \mathbf{d}(\boldsymbol{\zeta})^\top \mathbf{x} > 0$, for all feasible $\mathbf{x}$ and for all $\boldsymbol{\zeta} \in \mathcal{Z}$. For the non-robust version one can use the Charnes-Cooper transformation that is proposed by Charnes and Cooper (1962) to obtain an equivalent linear optimization problem. However, if we apply this transformation to the robust version, we obtain:

$$\begin{cases} \min_{\mathbf{y},t} \max_{\boldsymbol{\zeta} \in \mathcal{Z}} \alpha(\boldsymbol{\zeta})t + \mathbf{c}(\boldsymbol{\zeta})^\top \mathbf{y} \\ \text{s.t.} \ \beta(\boldsymbol{\zeta})t + \mathbf{d}(\boldsymbol{\zeta})^\top \mathbf{y} = 1 \quad \forall \boldsymbol{\zeta} \in \mathcal{Z} \\ \quad \sum_j a_{ij} y_j \ge b_i t \quad \forall i \\ \quad \mathbf{y} \ge \mathbf{0}, \ t \ge 0, \end{cases}$$

which is not equivalent to (19) since the uncertainty in the original objective is now split over the objective and a constraint. A better way to deal with such problems is to solve the robust linear problem

$$
\begin{cases}
\min_{\mathbf{x}} \max_{\boldsymbol{\zeta} \in \mathcal{Z}} \left[\alpha(\boldsymbol{\zeta}) + \mathbf{c}(\boldsymbol{\zeta})^{\top} \mathbf{x} - \lambda \left(\beta(\boldsymbol{\zeta}) + \mathbf{d}(\boldsymbol{\zeta})^{\top} \mathbf{x} \right) \right] \\
\text{s.t. } \sum_{j} a_{ij} x_j \geq b_i \\
\quad \mathbf{x} \geq \mathbf{0},
\end{cases}
$$

for a fixed value of λ, and then find the minimal value of λ for which this optimization problem still has a non positive optimal value. One can use for example binary search on λ to do this. For a more detailed treatment of robust fractional problems we refer to Gorissen (2014).

- **Product of binary variables.** Suppose that a robust constraint contains a product of binary variables, say xy, with $x, y \in \{0, 1\}$. Then one can use the standard way to linearize this:

$$
\begin{cases}
z \leq x \\
z \leq y \\
z \geq x + y - 1 \\
z \geq 0,
\end{cases}
$$

and replace xy with z. One can use this reformulation since the added constraints do not contain uncertain parameters.

- **Product of binary and continuous variable.** A product of a binary and a continuous variable that occurs in a robust constraint can also be reformulated in linear constraints, in a similar way as above. However, note that in the following robust constraint:

$$
\mathbf{a}(\boldsymbol{\zeta})^{\top} \mathbf{x} + z \mathbf{b}(\boldsymbol{\zeta})^{\top} \mathbf{x} \leq d(\boldsymbol{\zeta}) \quad \forall \boldsymbol{\zeta} \in \mathcal{Z},
$$

where $z \in \{0, 1\}$, one cannot use the standard trick:

$$
\begin{cases}
\mathbf{a}(\boldsymbol{\zeta})^{\top} \mathbf{x} + zy \leq d(\boldsymbol{\zeta}) & \forall \boldsymbol{\zeta} \in \mathcal{Z} \\
y \geq \mathbf{b}(\boldsymbol{\zeta})^{\top} \mathbf{x} & \forall \boldsymbol{\zeta} \in \mathcal{Z},
\end{cases}
\tag{20}
$$

and then linearize zy. This is not possible since in (20) the uncertainty is split over different constraints. A correct reformulation is:

$$
\begin{cases}
\mathbf{a}(\boldsymbol{\zeta})^{\top} \mathbf{x} + \mathbf{b}(\boldsymbol{\zeta})^{\top} \mathbf{x} \leq d(\boldsymbol{\zeta}) + M(1 - z) & \boldsymbol{\zeta} \in \mathcal{Z} \\
\mathbf{a}(\boldsymbol{\zeta})^{\top} \mathbf{x} \leq d(\boldsymbol{\zeta}) + Mz & \boldsymbol{\zeta} \in \mathcal{Z},
\end{cases}
\tag{21}
$$

where M is a sufficiently big number.

- **K out of N constraints should be satisfied.** Suppose the restriction is that at least K out of the N robust constraints

$$
\mathbf{a_i}(\boldsymbol{\zeta})^{\top} \mathbf{x} \leq d_i(\boldsymbol{\zeta}) \quad \forall \boldsymbol{\zeta} \in \mathcal{Z}
\tag{22}
$$

should be satisfied, where $i \in \{1, \ldots, N\}$. Then one can use the standard way

$$\begin{cases} \mathbf{a_i}(\boldsymbol{\zeta})^\top \mathbf{x} \le d_i(\boldsymbol{\zeta}) + M(1 - z_i) & \forall \boldsymbol{\zeta} \in \mathcal{Z}, \forall i \\ \sum_i z_i \ge K \\ z_i \in \{0, 1\} & \forall i, \end{cases}$$

where M is a sufficiently big number. However, if the restriction is that $\forall \boldsymbol{\zeta} \in \mathcal{Z}$ at least K out of the N constraints should be satisfied (notice the difference with (22)), then the above constraint-wise formulation is not equivalent and is overly conservative. We do not see how to model such a constraint correctly. Maybe an adversarial approach could be used for such constraints.

- **If-then constraint.** Since an "if-then constraint" can be modeled as an at least 1 out of 2 constraints, the above remarks hold.

Up to now we only described linear optimization examples. Similar examples can be given for conic and nonlinear optimization. In Lobo et al. (1998) for example, many optimization problems are given that can be modeled as conic quadratic programming problems. However, for many of them it holds that the corresponding robust counterparts are not the same. This means that if an optimization problem is conic quadratic representable, then the robust counterparts are not automatically the same, and hence in such cases the robust optimization techniques for CQP cannot be used.

7. How to deal with equality constraints?

Equality constraints containing uncertain parameters should be avoided as much as possible, since often such constraints restrict the feasibility region drastically or even lead to infeasibility. Therefore, the advice is: *do not use slack variables unless they are adjustable, since using slack variables leads to equality constraints*; see Ben-Tal et al. (2009, Chapter 2). However, equality constraints containing uncertain parameters cannot always be avoided. There are several ways to deal with such uncertain equality constraints:

- In some cases it might be possible to convert the equality constraints into inequality constraints. An illustrating example is the transportation problem: the demand constraints can either be formulated as equality constraints or as inequality constraints. The structure of the problem is such that at optimality these inequalities are tight.

- The equality constraints can be used to eliminate variables. This idea is mentioned in Ben-Tal et al. (2009). However, several questions arise. First of all, after elimination of variables and after the resulting problem has been solved, it is unclear which values to take for the eliminated variables, since they also depend on the uncertain parameters. This is no problem if the eliminated variables are adjustable variables or state/analysis variables, since there is no need to know their optimal values. A good example is the production-inventory problem for which one can easily eliminate

the state or analysis variables indicating the inventory in different time periods. See, e.g., Ben-Tal et al. (2009). Secondly, suppose the coefficients with respect to the variables that will be eliminated contain uncertain parameters. Eliminating such variables leads to problems that contain non-linear uncertainty, which are much more difficult to solve. To illustrate this, let us consider the following two constraints of an optimization problem:

$$\zeta_1 x_1 + x_2 + x_3 = 1, \quad x_1 + x_2 + \zeta_2 x_3 \leq 5,$$

in which ζ_1 and ζ_2 are uncertain parameters. Suppose that x_1, x_2 and x_3 are all adjustable in ζ_1. Then there are three options for elimination:

1. *Elimination of x_1*. Let us assume that $\zeta_1 = 0$ is not in the uncertainty set. By substituting $x_1 = (1 - x_2 - x_3)/\zeta_1$ the inequality becomes:

$$\left(1 - \frac{1}{\zeta_1}\right) x_2 + \left(\zeta_2 - \frac{1}{\zeta_1}\right) x_3 \leq 5 - \frac{1}{\zeta_1}.$$

   The disadvantage of eliminating x_1 is thus that the uncertainty in the inequality becomes non-linear.

2. *Elimination of x_2*. By substituting $x_2 = 1 - \zeta_1 x_1 - x_3$ the inequality becomes:

$$(1 - \zeta_1)x_1 + (\zeta_2 - 1)x_3 \leq 4,$$

   which is linear in the uncertain parameters.

3. *Elimination of x_3*. By substituting $x_3 = 1 - \zeta_1 x_1 - x_2$ the inequality becomes:

$$(1 - \zeta_1\zeta_2)x_1 + (1 - \zeta_2)x_2 \leq 5 - \zeta_2,$$

   which is nonlinear in the uncertain parameters. We conclude that from a computational point of view it is more attractive to eliminate x_2.

It is important to note that different choices of variables to eliminate may lead to different optimization problems.

• If the constraint contains state or analysis variables one could make these variables adjustable and use decision rules, thereby introducing much more flexibility. One can easily prove that when the coefficients for such variables in the equality constraint do not contain uncertain parameters and the equality constraint is linear in the uncertain parameters, then using linear decision rules for such variables is equivalent to eliminating these variables. To be more precise: suppose the linear equality constraint is

$$\mathbf{q}(\boldsymbol{\zeta})^\top \mathbf{x} + y = r,$$

where $\mathbf{q}(\boldsymbol{\zeta})$ is linear in $\boldsymbol{\zeta}$, and y is an state or analysis variable (without loss of generality we assume the coefficient for y is 1). Then it can easily be proven that substituting $y = r - \mathbf{q}(\boldsymbol{\zeta})^\top \mathbf{x}$ everywhere in the problem is equivalent to using a linear decision rule for y. To reduce the number of extra variables, it is therefore better to eliminate such variables.

8. On maximin and minimax formulations of RC

In this section, we consider an uncertain LP of the following general form:

$$\text{(LP)} \quad \max_{\mathbf{x} \geq \mathbf{0}} \{\mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d}\},$$

where without loss of generality $\mathbf{A}$ is the uncertain coefficient matrix that resides in the uncertainty set $\mathcal{U}$. So the general RC is given by

$$\text{(R-LP)} \quad \max_{\mathbf{x} \geq \mathbf{0}} \{\mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d} \ \forall \mathbf{A} \in \mathcal{U}\}.$$

Here we show that (R-LP) can be reformulated as:

$$\text{(RF)} \quad \min_{\mathbf{A} \in \mathcal{U}} \max_{\mathbf{x} \geq \mathbf{0}} \{\mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d}\},$$

if the uncertainty is constraint-wise; however if this condition is not met, then (RF) may not be equivalent to (R-LP).

Remark 1 *This shows that the statement "RO optimizes for the worst case $\mathbf{A}$" is too vague. Also the maximin reformulation:*

$$\max_{\mathbf{x} \geq \mathbf{0}} \min_{\mathbf{A} \in \mathcal{U}} \{\mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d}\},$$

is usually not equivalent to (R-LP). This is because we can almost always find an $\mathbf{x} \geq 0$ such that no $\mathbf{A} \in \mathcal{U}$ exists for which $\mathbf{A}\mathbf{x} \leq \mathbf{d}$; therefore, we minimize over an empty set, and have $+\infty$ for the maximin objective. Also when $\mathbf{x}$ is selected such that at least one feasible $\mathbf{A}$ exists (e.g., see Falk (1973)), it is easy to find examples where both formulations are not equivalent.

To show (R-LP)=(RF) when the uncertainty is constraint-wise, we first take the dual of the (inside) maximization problem of (RF) $[\max_{\mathbf{x} \geq \mathbf{0}} \mathbf{c}^\top \mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{d}]$. Then, substituting the dual with the primal (maximization) problem in (RF) gives:

$$\text{(OC-LP)} \quad \min_{\mathbf{A} \in \mathcal{U}, \mathbf{y} \geq \mathbf{0}} \{\mathbf{d}^\top \mathbf{y} : \mathbf{A}^\mathbf{T} \mathbf{y} \geq \mathbf{c}\},$$

where val(OC-LP) = val(RF) holds under regularity conditions at optimality. Note that the constraints of (RF) can be formulated as $[\mathbf{a_i^T}\mathbf{x} \leq d_i, \forall \mathbf{a_i} \in \mathcal{U}_i, i = 1, \ldots, m]$, if the uncertainty is constraint-wise. Beck and Ben-Tal (2009, Theorem 3.1) and Soyster and Murphy (2013) show that (OC-LP)—which is the *optimistic counterpart of the dual problem*—is equivalent to the dual of the general robust counterpart (R-LP), and val(OC-LP) = val(R-LP) holds for constraint-wise uncertainty, and disjoint $\mathcal{U}_i$'s. Therefore, val(R-LP) = val(RF) is also satisfied when the uncertainty is constraint-wise. However, if (some of) the uncertainty at (some of) the constraints are dependent in (R-LP), then we may not sustain the associated equivalence. The following example shows such a situation.

Example

Consider the following toy RC example in which the uncertainty is not constraint-wise:

$$\text{(RC-Toy)} \ \max_{\mathbf{y}} \ y_1 + y_2$$
$$\text{s.t. } a_1 y_1 \le 1, \quad a_2 y_2 \le 1 \ \forall \mathbf{a} \in \mathbb{R}^2 : ||\mathbf{a}||_2 \le 1,$$

where two constraints of the problem are dependent on each other via the ellipsoidal uncertainty set $[\mathbf{a} \in \mathbb{R}^2 : ||\mathbf{a}||_2 \le 1]$. The robust reformulation of the (RC-Toy) is as follows:

$$\text{(RF-Toy)} \ \min_{\mathbf{a}:||\mathbf{a}||_2 \le 1} \ \max_{\mathbf{y}} \ y_1 + y_2$$
$$\text{s.t. } a_1 y_1 \le 1, \quad a_2 y_2 \le 1,$$

and the optimistic counterpart (OC) of the problem is

$$\text{(OC-Toy)} \ \min_{\mathbf{x} \ge \mathbf{0}, \, \mathbf{a}:||\mathbf{a}||_2 \le 1} \ x_1 + x_2$$
$$\text{s.t. } a_1 x_1 = 1, \quad a_2 x_2 = 1.$$

(RC-Toy) attains an optimal objective value of 2, whereas the (RF-Toy)'s optimal objective value is $2\sqrt{2}$. Therefore, the robust reformulation (RF-Toy) is not equivalent to the general RC problem (RC-Toy) in this situation. However, val(RF-Toy) = val(OC-Toy) from duality.

9. Quality of robust solution

In this section we describe how to assess the quality with respect to robustness of a solution based on a simulation study. We first identify four focus points for performing a Monte Carlo experiment, and conclude with two statistical tests that can be used to compare two solutions.

Choice of the uncertainty set. For a comparison between different solutions, it is necessary to define an uncertainty set $\mathcal{U}$ that is used for evaluation. This set should reflect the real-life situation. The uncertainty set that is used for optimization may be different than the set for evaluation. For example, an ellipsoidal set may be used to reduce the conservatism when the real-life uncertainty is a box, while still maintaining a large probability of constraint satisfaction (Ben-Tal et al. 2009, p. 34).

Choice of the probability distribution. A simulation requires knowledge of the probability distribution on the uncertainty set. If this knowledge is ambiguous, it may be necessary to verify whether the simulation results are sensitive with respect to changes in this distribution. For example, Rozenblit (2010) performs different simulations, each based on a probability distribution with a

different skewness level.

Choice of the sampling method. For univariate random variables it is computationally easy to draw a random sample from any given distribution. For multivariate random variables rejection sampling can be used, but it may be inefficient depending on the shape of the uncertainty set, e.g., for an uncertainty set with no volume. A more efficient method for sampling from an arbitrary continuous probability distribution is "hit and run" sampling (Bélisle et al. 1993). An R package for uniform hit and run sampling from a convex body is also available.

Choice of the performance characteristics. From a mathematical point of view there is no difference between uncertainty in the objective and uncertainty in the constraints since an uncertain objective can always be reformulated as a certain objective and an uncertain constraint. However, the distinction between an uncertain objective and an uncertain constraint is important for the interpretation of a solution. First, we look at the effects of adjustable RO and reformulations, then we present the performance characteristics.

Effect of adjustable RO. When one or more "wait and see" variables are modeled as adjustable variables, uncertain parameters may enter the objective function. In that case the performance characteristics for uncertainty in the objective become applicable.

Effect of reformulations. Reformulations are sometimes necessary to end up with a tractable model. The evaluation should be based on the original model, since reformulations introduce additional constraints whose violation is not necessarily a problem. Take for example an inventory model that has constraints on variables that indicate the cost at a certain time period (e.g., constraints (23) and (24)). These constraints have been introduced to model the costs in the objective function. A violation of these constraints does not render the solution infeasible but does affect the objective value (i.e., the costs of carrying out the solution).

Performance characteristics for uncertainty in the constraints. For an uncertain constraint $f(\mathbf{a}, \boldsymbol{\zeta}) \leq 0$ for all $\boldsymbol{\zeta}$ in $\mathcal{Z}$, the violation is $\max\{0, f(\mathbf{a}, \boldsymbol{\zeta})\}$. Meaningful statistics are the probability on positive violation and the distribution of the violation (average, worst case, standard deviation) under the condition that the violation is positive. When multiple constraints are uncertain, these statistics can be computed per constraint. Additionally, the average number of violated constraints can be reported.

There is a clear trade-off between the objective value and constraint violations. The difference between the worst case objective value of the robust solution and the nominal objective value of the nominal solution is called the *price of robustness* (PoR) (Bertsimas and Sim 2004). It is useful if the objective is certain, since in that case PoR is the amount that has to be paid for being robust against constraint violations. We observe that PoR is also used when the objective is uncertain. We discourage this, since it compares the nominal solution in case there is no uncertainty with the robust solution where the worst case occurs, so it compares two different scenarios.

Performance characteristics for uncertainty in the objective. Uncertainty in the objective affects the performance of a solution. For every simulated uncertainty vector, the actual objective value can be computed. One may be interested in the worst case, but also in the average value or the standard deviation. For a solution that is carried out many times, reporting the average performance is justified by the law of large numbers. The worst case may be more relevant when a solution is carried out only once or a few times, e.g., when optimizing a medical treatment plan for a single patient. These numbers show what objective value to expect, but they do not provide enough information about the quality of a solution since a high standard deviation is not necessarily undesirable. A robust solution is good when it is close to the *perfect hindsight* (PH) solution. The PH solution is the solution that is obtained by optimizing the decision variables for a specific uncertainty vector as if it is fully known beforehand. This has to be done for every simulated uncertainty vector, and yields an utopia solution. The PH solution may have a large variation, causing a high variation of good solutions as well.

Performance characteristics for any problem. Regardless of whether the uncertainty is in the objective or in the constraints, the mean and associated standard deviation of the difference between the actual performance of a solution and the PH solution are useful for quantifying the quality of a solution. The mean difference between the PH solution and a fully robust solution is defined as the *price of uncertainty* (PoU) by Ben-Tal et al. (2005). It is the maximum amount that a company should invest for reducing the level of uncertainty, e.g., by using more accurate forecasting techniques. It can also be interpreted as the regret of choosing a certain solution rather than the PH solution. Alternative names for PoU are "cost of robustness" (Gregory et al. 2011) or "price of robustness" (Ben-Tal et al. 2004), which are less descriptive than "price of uncertainty" and may cause confusion with price of robustness from (Bertsimas and Sim 2004). A low mean PoU and a low standard deviation characterize a good solution.

Subtracting the mean objective value of the nominal solution from the mean value of a robust solution yields the *actual price of robustness* (APoR) (Rozenblit 2010). APoR can be interpreted as

the expected price that has to be paid for using the robust solution rather than the nominal solution, which is negative if RO offers a solution that is better on average. PoR equals APoR when uncertainty only occurs in the constraints.

For multistage problems one may also follow a folding horizon (FH) approach. With FH in each stage where a part of the uncertain parameter is observed, that information is used to optimize for the remaining time periods. This is done by taking the original optimization problem, fixing the decision variables for previous stages, and fixing the elements of the uncertain parameter that have been observed. This allows a fair comparison between a dynamic solution (e.g., created by the AARC) and a static solution (e.g., the nominal solution) when in real-life the static solution is reoptimized in every stage.

Comparing two solutions. We provide several comparison criteria and provide the corresponding statistical test to verify whether one solution is better than another solution. The tests will be demonstrated in Section 10. We will assume that the data for the statistics test is available as n pairs (X_i, Y_i) $(i = 1, 2, \ldots, n)$, where X_i and Y_i are performance characteristics in the i'th simulation. For uncertainty in the objective, they can be objective values whereas for uncertainty in the constraints they can be the numbers of constraint violations or the sizes of the constraint violations. We assume that (X_i, Y_i) and (X_j, Y_j) are independent if $i \neq j$, and that smaller values are better. When a conjecture for a test is based on the outcome of a simulation study, the statistical test must be performed with newly generated data to avoid statistical bias. While for the statistical tests it is not necessary that X_i and Y_i are based on the same simulated uncertainty vector $\boldsymbol{\zeta}$, it increases the power of the test since X_i and Y_i will be positively correlated. This reduces the variance of the difference: $\mathrm{Var}(X_i - Y_i) = \mathrm{Var}(X_i) + \mathrm{Var}(Y_i) - 2\,\mathrm{Cov}(X_i, Y_i)$, which is used in the following tests:

• The sign test for the median validates the null hypothesis that the medians of the distributions of X_i and Y_i are equal. This tests the conjecture that the probability that one solution outperforms the other is larger than 0.5.

• The t-test for the mean validates the null hypothesis that the means of the distributions of X_i and Y_i are equal. This tests the conjecture that one solution outperforms the other in long run average behavior.

10. RC may take better "here and now" decisions than AARC

A linear decision rule is a linear approximation of a more complicated decision rule. It dictates what to do at each stage as a linear function of observed uncertain parameters, but it is not guaranteed to be the optimal strategy. Every time a decision has to be made it is possible to either follow the linear decision rule, or to reoptimize the AARC for the remaining time periods based on everything that

is observed up till then. We will refer to the latter as the AARC-FH, where FH stands for folding horizon. Ben-Tal et al. (2005) compare the AARC with the AARC-FH, and show that the latter produces better solutions on average. A comparison that involves AARC-FH assumes that there is time to reoptimize. It is therefore natural to also make a comparison with the RC-FH, where the RC is solved for the full time horizon and re-optimized for the remaining time period every time a part of the uncertain parameters is unveiled. On average, the RC-FH may outperform the AARC (Cohen et al. 2007, Rozenblit 2010).

In the remainder of this section we will evaluate both the average and the worst case performance of the nominal solution with FH, the RC-FH and the AARC-FH. A comparison between RC-FH and AARC-FH is new, and shows which model takes the best "here and now" decisions.

We first give an example for the worst case performance. Consider a warehouse that transfers one good. The current inventory is $x_0 = 5$, the holding costs per time period are $h = 1$, the backlogging costs per time period are $b = 2$. In the first period, any nonnegative (not necessarily integer) amount can be ordered while in the second period the maximum order quantity is $q_2^{max} = 3$. Let $T = \{1, 2\}$, let q_t be the order quantity in time period t, and let c_t denote the costs associated with time period t. The ending inventory can be returned to the supplier without penalty fee at time period three. The optimization problem can be formulated as:

$$\min \quad \sum_{t \in T} c_t$$

$$\text{s.t.} \quad c_t \geq (x_0 + \sum_{i=1}^{T} q_i - d_i)h \qquad \forall t \in T \qquad (23)$$

$$c_t \geq -(x_0 + \sum_{i=1}^{T} q_i - d_i)b \qquad \forall t \in T \qquad (24)$$

$$q_2 \leq q_2^{max}$$

$$q_t \in \mathbb{R}_+ \qquad \forall t \in T.$$

Suppose the demand vector **d** is uncertain but is known to reside in a ball around **5** with radius 5. We will use this uncertainty set both for optimization and for evaluation.

For this small example, it is possible to approximate the worst case costs for an FH approach as a function of the "here and now" decision q_1 as follows. For each q_1 in a range of values, we have randomly drawn 100 uncertain demand vectors from the boundary of the uncertainty set. For each demand vector we have computed the inventory level at the beginning of the second time period $(= x_0 + q_1 - d_1)$. Based on this inventory level, we reoptimized the order quantity for the second time period, where d_2 was assumed to reside in the interval $[5 - r, 5 + r]$ with $r = \sqrt{25 - (d_1 - 5)^2}$ (so

Figure 1   Approximation of the total worst case costs for an FH strategy as a function of the initial order quantity q_1.
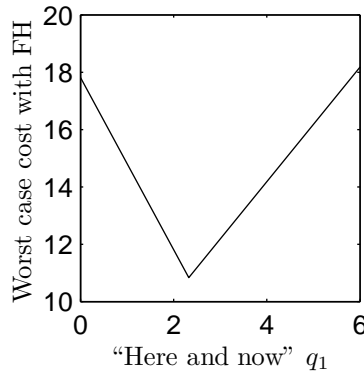
that the full $\mathbf{d}$ vector is in a ball around $\mathbf{5}$ with radius 5). Then we computed the total costs over both time periods. The maximum total costs over all 100 demand vectors approximates the worst case costs with the FH approach, and is depicted in Figure 1. From this picture it becomes clear that the optimal order quantity for the first time period is approximately 2.3, which has a worst case performance of 10.8.

We have solved the model for the full time horizon with the RC, with the AARC (where c_1 and c_2 are adjustable on the full $\mathbf{d}$, and q_2 is adjustable on d_1), and as a certain problem with $\mathbf{d} = \mathbf{5}$. The nominal solution gives $q_1 = 0$, the RC gives $q_1 \approx 4.4$, while the AARC yields $q_1 \approx 5.3$, leading to worst case costs of the FH approach of 17.8, 14.9 and 16.8, respectively. So, the RC takes the best "here and now" decision with respect to the worst case performance. It may be paradoxical that the AARC yields a worse solution than the RC, since the feasible region of the AARC includes the RC solution. However, neither of the two optimize the right objective function. Both approximate the objective value using (static or adjustable) auxiliary variables c_t. While AARC indeed has a better objective value than RC, the solution is not better for the original objective function.

We also perform a comparison on a more realistic problem, which is the retailer-supplier flexible commitment problem by Ben-Tal et al. (2005). At the starting time, the retailer commits himself to ordering certain quantities in later months. These commitments are flexible, i.e., deviations are allowed at a penalty cost. The objective is to minimize the total costs for the retailer, consisting of ordering costs (minus the salvage value), holding costs, backlogging costs, penalty costs associated with deviating from the commitments, and costs for variability in the commitments. The article provides two data sets for twelve time periods, A12 and W12, which we also use in our optimization and comparison.

In this problem the retailer faces an uncertain demand. Following Ben-Tal et al. (2005) we consider box uncertainty where the demand may deviate up to $\rho\%$ around the nominal value. For the

simulation we draw demand vectors uniformly from this box region. For these demand vectors the nominal solution, RC and AARC are carried out completely. For the FH approach, the reoptimization is performed after each time period based on previously observed demand. 500 demand vectors were generated for each data set and each uncertainty level ρ, and the same demand vectors were used for all models. In addition, the PH solution was computed for each of these demand vectors.

The simulation results are listed in Tables 5 and 6. The PH solution always performs best, but cannot be used in practice, and hence, we ignore it for now. For data set A12, the nominal solution with FH results in the lowest average costs. This means that the nominal solution takes better "here and now" decisions than RC and AARC. Moreover, the RC-FH has lower average costs than the AARC-FH, so also the RC takes better "here and now" decisions than AARC. The advantage of the nominal FH solution compared to RC-FH and AARC-FH increases when the uncertainty set becomes larger. The same conclusions can be drawn for data set W12, except that the nominal solution is the best solution and FH leads to higher mean costs, and except that AARC (without folding horizon) outperforms RC. The Nominal-FH in Table 5 is an example of the statement in Section 9 that a high standard deviation is not necessarily bad. Its standard deviation is higher than that of AARC-FH but this is explained by the high standard deviation of PH. In Table 6 we see that Nominal-FH has a lower mean and standard deviation than AARC-FH, which means that Nominal-FH is closer to the PH solution than AARC-FH. For data set A12 and $\rho = 10\%$, it is not clear whether RC outperforms AARC. We now demonstrate the two statistical tests from Section 9 on this data set, each based on 100 newly generated uncertainty vectors, to test whether RC outperforms AARC. The null hypothesis that both solutions perform equally on average is rejected ($p = 4.7 \cdot 10^{-12}$), and also the null hypothesis that the medians of the RC and AARC are equal is rejected ($p = 2.8 \cdot 10^{-8}$). These results show that the AARC is not necessarily better than the RC and support the statement in Section 9 that a simulation is required for comparing solutions. As mentioned in Section 2.2, RO may provide solutions that are not Pareto efficient when multiple optimal solutions exist. A different optimal solution to the RC or AARC may yield completely different simulation results, rendering our conclusions useless. We have therefore generated solutions that are robust optimal, and additionally, are optimal with respect to the nominal demand trajectory. Although these solutions were different, the conclusions remain unaffected.

Table 5   Simulated mean (std) costs for the retailer-supplier flexible commitment problem.

	A12		W12	
	$\rho = 10\%$	$\rho = 50\%$	$\rho = 10\%$	$\rho = 50\%$
Nominal	816 (35)	999 (217)	12799 (742)	16002 (3634)
RC	861 (16)	1080 (81)	13907 (275)	21564 (1415)
AARC	874 (4)	1030 (105)	13313 (35)	18577 (179)
Nominal-FH	807 (21)	967 (156)	12839 (288)	16276 (1447)
RC-FH	829 (6)	1030 (47)	13110 (56)	17578 (309)
AARC-FH	881 (4)	1063 (65)	13312 (35)	18577 (178)
PH	785 (18)	826 (95)	12187 (204)	12950 (1024)

Table 6   Simulated mean (std) PoU for the retailer-supplier flexible commitment problem.

	A12		W12	
	$\rho = 10\%$	$\rho = 50\%$	$\rho = 10\%$	$\rho = 50\%$
Nominal	30 (25)	173 (156)	653 (597)	3053 (2979)
RC	76 (29)	253 (141)	1720 (466)	8615 (2370)
AARC	89 (16)	228 (78)	1126 (231)	5627 (1160)
Nominal-FH	22 (9)	140 (75)	653 (144)	3326 (703)
RC-FH	44 (16)	204 (103)	924 (207)	4628 (1042)
AARC-FH	95 (16)	237 (125)	1125 (232)	5527 (1161)

11. Conclusion

In this paper, we give a concise introduction and a general procedure that shall be helpful for using RO in practice. Additionally, we give several practical insights and hints in applying RO. Examples of such practical insights are: the robust reformulations of equivalent deterministic optimization problems may not be equivalent; in multi-stage optimization problems, re-optimizing the given problem at each stage using static RO or nominal data may outperform solutions provided by ARO; and the actual probability guarantee of an uncertainty set is often higher than the probabilistic guarantee that is approximated by using a safe approximation technique. We also discuss many practical issues to apply RO in a successful and convincing way. Examples are: how to choose the uncertainty set; what is the right interpretation of "RO optimizes for the worst case"; and should the decision rule used in ARO be a function of the final or the primitive uncertainty? Moreover, we propose ideas on how to deal with equality constraints and integer adjustable variables, and on how to compare the robustness characteristics of two solutions. We have provided many numerical examples to illustrate our insights and discussions.

References

F. Babonneau, J.-P. Vial, and R. Apparigliato.   Robust optimization for environmental and energy planning.   In *Uncertainty and Environmental Decision Making*, pages 79–126. Springer, 2010.

http://dx.doi.org/10.1007/978-1-4419-1129-2_3.

C. Bandi and D. Bertsimas. Tractable stochastic analysis in high dimensions via robust optimization. *Mathematical Programming*, 134(1):23–70, 2012.

A. Beck and A. Ben-Tal. Duality in robust optimization: primal worst equals dual best. *Operations Research Letters*, 37(1):1–6, 2009.

C. J. P. Bélisle, H. E. Romeijn, and R. L. Smith. Hit-and-run algorithms for generating multivariate distributions. *Mathematics of Operations Research*, 18(2):255–266, 1993.

A. Ben-Tal and A. Nemirovski. Robust optimization–methodology and applications. *Mathematical Programming*, 92(3):453–480, 2002.

A. Ben-Tal and A. Nemirovski. Selected topics in robust convex optimization. *Mathematical Programming*, 112(1):125–158, 2008.

A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

A. Ben-Tal, B. Golany, A. Nemirovski, and J.-P. Vial. Retailer-supplier flexible commitments contracts: A robust optimization approach. *Manufacturing & Service Operations Management*, 7(3):248–271, 2005.

A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton Series in Applied Mathematics. Princeton University Press, 2009. http://sites.google.com/site/robustoptimization/.

A. Ben-Tal, D. den Hertog, A. M. B. De Waegenaere, B. Melenberg, and G. R. Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

A. Ben-Tal, D. den Hertog, and J.-P. Vial. Deriving robust counterparts of nonlinear uncertain inequalities. *Mathematical Programming*, Online First, 2014.

D. Bertsimas and D. B. Brown. Constructing uncertainty sets for robust linear optimization. *Operations Research*, 57(6):1483–1495, 2009.

D. Bertsimas and C. Caramanis. Finite adaptability in multistage linear optimization. *IEEE Transactions on Automatic Control*, 55(12):2751–2766, 2010.

D. Bertsimas and A. Georghiou. Design of near optimal decision rules in multistage adaptive mixed-integer optimization. *Optimization Online*, Sept, 2013.

D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

D. Bertsimas, D. A. Iancu, and P. A. Parrilo. Optimality of affine policies in multistage robust optimization. *Mathematics of Operations Research*, 35(2):363–394, 2010.

D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011.

D. Bertsimas, V. Gupta, and N. Kallus. Data-driven robust optimization. *arXiv preprint*, 1401.0212, 2013a.

D. Bertsimas, E. Litvinov, X. A. Sun, J. Zhao, and T. Zheng. Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Transactions on Power Systems*, 28(1):52–63, 2013b.

D. Bertsimas, I. Dunning, and M. Lubin. Reformulations versus cutting planes for robust optimization: A computational and machine learning perspective. *Optimization Online*, April, 2014.

D. Bienstock and N. Özbay. Computing robust basestock levels. *Discrete Optimization*, 5(2):389–414, 2008.

J. R. Birge and F. V. Louveaux. *Introduction to Stochastic Programming*. Springer, 2011. http://dx.doi.org/10.1007/978-1-4614-0237-4.

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

M. C. Campi and S. Garatti. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230, 2008.

A. Charnes and W. W. Cooper. Programming with linear fractional functionals. *Naval Research Logistics Quarterly*, 9(3-4):181–186, 1962.

X. Chen, M. Sim, P. Sun, and J. Zhang. A linear decision-based approximation approach to stochastic programming. *Operations Research*, 56(2):344–357, 2008.

I. Cohen, B. Golany, and A. Shtub. The stochastic time-cost tradeoff problem: A robust optimization approach. *Networks*, 49(2):175–188, 2007.

J. E. Falk. A linear max–min problem. *Mathematical Programming*, 5(1):169–188, 1973.

E. F. Fama and K. R. French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56, 1993.

A. Fredriksson, A. Forsgren, and B. Hårdemark. Minimax optimization for handling range and setup uncertainties in proton therapy. *Medical Physics*, 38(3):1672–1684, 2011.

A. Georghiou, W. Wiesemann, and D. Kuhn. Generalized decision rule approximations for stochastic programming via liftings. *Mathematical Programming*, Online First, 2014.

J. Goh and M. Sim. Distributionally robust optimization and its tractable approximations. *Operations Research*, 58(4):902–917, 2010.

B. L. Gorissen. Robust fractional programming. *Journal of Optimization Theory and Applications*, Online First, 2014.

B. L. Gorissen and D. den Hertog. Robust counterparts of inequalities containing sums of maxima of linear functions. *European Journal of Operational Research*, 227(1):30–43, 2012.

C. Gregory, K. Darby-Dowman, and G. Mitra. Robust optimization and portfolio selection: The cost of robustness. *European Journal of Operational Research*, 212(2):417–428, 2011.

M. J. Hadjiyiannis, P. J. Goulart, and D. Kuhn. A scenario approach for estimating the suboptimality of linear decision rules in two-stage robust optimization. In *50th IEEE Conference on Decision and Control and European Control Conference*, pages 7386–7391, 2011. http://dx.doi.org/10.1109/CDC.2011.6161342.

G. A. Hanasusanto, D. Kuhn, and W. Wiesemann. Two-stage robust integer programming. *Optimization Online*, March, 2014.

D. A. Iancu and N. Trichakis. Pareto efficiency in robust optimization. *Management Science*, 60(1):130–147, 2014.

D. Kuhn, W. Wiesemann, and A. Georghiou. Primal and dual linear decision rules in stochastic and robust optimization. *Mathematical Programming*, 130(1):177–209, 2011.

S. Lim. A joint optimal pricing and order quantity model under parameter uncertainty and its practical implementation. *Omega*, 41(6):998–1007, 2013.

M. S. Lobo. *Robust and convex optimization with applications in finance.* PhD thesis, Stanford University, 2000. http://sousalobo.com/thesis/thesis.pdf.

M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284(1):193–228, 1998.

K. Postek. Adjustable robust optimization by splitting the uncertainty set. Master's thesis, Tilburg University, the Netherlands, 2013.

A. Prékopa. *Stochastic Programming.* Kluwer Academic Publishers, 1995. http://dx.doi.org/10.1007/978-94-017-3087-7.

R. T. Rockafellar. *Convex analysis.* Princeton University Press, 1997.

M. Roelofs and J. Bisschop. AIMMS 3.12: The Language Reference, Paragon Decision Technology BV, 2012.

M. Rozenblit. Robust optimization of a multi-period production planning problem under uncertainty. Master's thesis, Technion – Israel Institute of Technology, 2010.

A. P. Ruszczyński and A. Shapiro. *Stochastic Programming (handbooks in operations research and management science).* Elsevier, 2003.

A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5):1154–1157, 1973.

A. L. Soyster and F. H. Murphy. A unifying framework for duality and modeling in robust linear programs. *Omega*, 41(6):984–997, 2013.

P. Vayanos, D. Kuhn, and B. Rustem. Decision rules for information discovery in multi-stage stochastic programming. In *50th IEEE Conference on Decision and Control and European Control Conference*, pages 7368–7373, 2011. http://dx.doi.org/10.1109/CDC.2011.6161382.

X. Wang and D. J. Curry. A robust approach to the share-of-choice product design problem. *Omega*, 40(6):818–826, 2012.

S. Yan and C.-H. Tang. Inter-city bus scheduling under variable market share and uncertain market demands. *Omega*, 37(1):178–192, 2009.