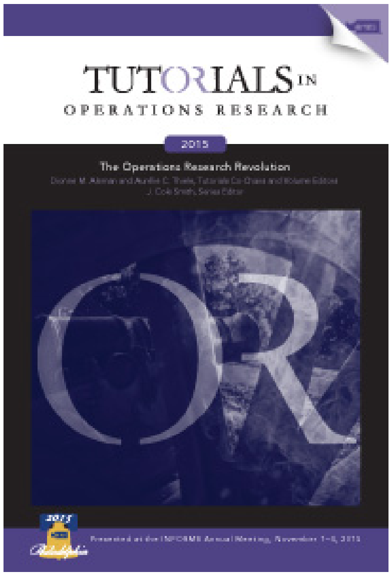


INFORMS TutORials in Operations Research

Robust Multistage Decision Making

Erick DelageDan A. Iancu

TUTORIALS IN OPERATIONS RESEARCH
INFORMS 2015

 | ISBN 978-0-9843378-8-0
http://dx.doi.org/10.1287/educ.2015.0139

Robust Multistage Decision Making

Erick Delage
Department of Decision Sciences, HEC Montréal, Montréal, Québec H3T 2A7, Canada,
erick.delage@hec.ca

Dan A. Iancu
Graduate School of Business, Stanford University, Palo Alto, California 94305,
daniancu@stanford.edu

Abstract Testifying to more than 10 years of academic and practical developments, this tutorial attempts to provide a succinct yet unified view of the robust multistage decision-making framework. In particular, the reader should better understand (1) the distinction between static versus fully or partially adjustable decisions, (2) the root of tractability issues, (3) the connection to robust dynamic programming, (4) some motivation for using simple policies, especially in terms of optimality, (5) how time consistency issues can arise, and (6) some relevant applications.

Keywords robust optimization; sequential decision making; time consistency; dynamic programming

1. Introduction

The key underlying philosophy behind the robust optimization modeling paradigm is that, in many practical situations, a complete stochastic description of the uncertainty may not be available. Instead, one may only have information with less detailed structure, such as bounds on the magnitude of the uncertain quantities or rough relations linking multiple unknown parameters. In such cases, one may be able to describe the unknowns by specifying a set in which all realizations should lie, the so-called *uncertainty set*. The decision maker then seeks to ensure that the constraints in the problem remain feasible for *any* possible realization while optimizing an objective that protects against the worst possible outcome.

In its original form, proposed by Soyster [71] and Falk [48], robust optimization was mostly concerned with linear programming problems in which the data were inexact. Because of the columnwise structure of the uncertainty considered, the robust optimization problem amounted to taking the worst case for each parameter; because this was very conservative, its adoption by the operations research community was therefore limited until new research efforts in the late 1990s devised approaches to control for the degree of conservatism of the solution. Papers by Ben-Tal and Nemirovski [7, 8, 9, 10], Ben-Tal et al. [14], El-Ghaoui and Lebret [46], and El-Ghaoui et al. [47], followed by those of Bertsimas and Sim [26, 27], and Bertsimas et al. [35], considerably generalized the earlier framework by extending it to other classes of convex optimization problems beyond linear programming (quadratic, conic, and semidefinite programs), as well as more complex descriptions of the uncertainty (intersections of ellipsoidal uncertainty sets, uncertainty sets with budgets of uncertainty, etc.). A key feature of these papers was that the uncertainty set was centered at the nominal value of the uncertain parameters and that the size of the set could be controlled by the decision maker to capture his level of aversion to ambiguity. Throughout these papers, the key emphases were on

1. *tractability*, understanding the circumstances under which a nominal problem with uncertain data can be formulated as a tractable (finite-dimensional, convex) optimization problem and characterizing the complexity of solving this resulting *robust counterpart*; and

2. *the degree of conservatism and probabilistic guarantees*, understanding when the robust counterpart constitutes an exact or a safe (i.e., conservative) reformulation of the robust problem, as well as developing models that can control the degree of conservatism of the robust solution. For instance, in cases when the uncertainty is truly stochastic, one can design uncertainty sets ensuring that any feasible solution for the robust problem is feasible with high probability for the original stochastic problem.

Although robust optimization was initially developed for static problems with parameters of unknown but fixed value, it quickly emerged as a high-potential technique to handle both decision-making problems, where parameters are random and obey imprecisely known distributions, and dynamic decision-making problems, where the decision maker is able to adjust his strategy to information revealed over time.

Ben-Tal et al. [16] were the first[1] to discuss robust multistage decision problems, opening the field to numerous other papers either dealing with theoretical concepts or applying the framework to practical problems, such as inventory management (e.g., Ben-Tal et al. [15], Bertsimas and Thiele [28], Bienstock and Özbay [37]), facility location and transportation (e.g., Baron et al. [5]), scheduling (e.g., Lin et al. [60], Mittal et al. [64], Yamashita et al. [78]), dynamic pricing and revenue management (e.g., Adida and Perakis [1], Perakis and Roels [66], Thiele [72]), project management (e.g., Wiesemann et al. [75]), energy generation and distribution (e.g., Lorca and Sun [61], Zhao et al. [80]), or portfolio optimization (e.g., Bertsimas and Pachamanova [25], Ceria and Stubbs [39], Goldfarb and Iyengar [52], Pınar and Tütüncü [67], Tütüncü and Koenig [74]). We refer the reader to the review papers by Bertsimas et al. [29] and Gabrel et al. [50] and the book by Ben-Tal et al. [13] for a more thorough discussion and additional applications.

Whereas robust static decision making is now relatively well understood, robust multistage decision making continues to be a cutting-edge research area that provides a way to model uncertainty that is well suited to the uncertainty at hand and offers the potential for both strategic insights into the optimal policy and computational tractability. As a testimony to more than 10 years of academic and practical developments revolving around the robust multistage decision-making framework, this tutorial attempts to provide a succinct and unified view of the methodology while highlighting potential pitfalls and indicating several open questions. In particular, our objectives with the tutorial are to

1. provide tools for identifying a static versus a fully or partially adjustable decision variable (§2),
2. highlight some tractability issues related to the formulation (§3),
3. clarify the connection to robust dynamic programming (§4),
4. provide motivations for using simple policies (§5), and finally
5. illustrate how time consistency issues might arise (§6).

[1] In the context of multistage decision making, we should note that a parallel stream of work, focusing on similar notions of robustness, also existed for several decades in the field of dynamical systems and control. Witsenhausens early thesis [76] and subsequent paper [77] first formulated problems of state estimation with a set-based membership description of the uncertainty, and the thesis by Bertsekas [18] and paper by Bertsekas and Rhodes [20] considered the problem of deciding under what conditions the state of a dynamical system affected by uncertainties is guaranteed to lie in specific ellipsoidal or polyhedral tubes (the latter two references showed that, under some conditions, control policies that are linear in the states are sufficient for such a task). The literature on robust control received a tremendous speed-up in the 1990s, with contributions from numerous groups (e.g., Doyle et al. [44], Fan et al. [49]), resulting in two published books on the topic (Dullerud and Paganini [45], Zhou and Doyle [81]). Typically, in most of this literature, the main objective was to design control laws that ensured the dynamical system remained stable under uncertainty, and the focus was on coming up with computationally efficient procedures for synthesizing such controllers.

The target audience for our tutorial comprises primarily academics interested in learning more about robust multistage decision making, as well as practitioners seeking a gentle introduction to the framework. The tutorial assumes basic knowledge of the robust optimization paradigm for static problems, as well as basic familiarity with concepts in dynamic optimization (dynamic programming, Bellman principle, etc.).

2. When to Worry About Adjustable Decisions

We start our discussion by considering a simple inventory problem in which a retailer needs to order some goods in order to satisfy demand from his customers while incurring the lowest total cost, which is the sum of the ordering, holding, and backlogging costs over a finite time horizon. In a deterministic setting, this might take the shape of the following convex optimization problem:

$$\begin{aligned} \underset{x_t,\, y_t}{\text{minimize}} \quad & \sum_{t=1}^{T} \big(c_t x_t + h_t (y_{t+1})^+ + b_t(-y_{t+1})^+\big) \\ \text{s.t.} \quad & y_{t+1} = y_t + x_t - d_t, \quad \forall\, t, \\ & 0 \le x_t \le M_t, \quad \forall\, t, \\ & y_1 = a, \end{aligned} \tag{1}$$

where $x_t \in \mathbb{R}$ captures the number of goods ordered at time t and received by time $t+1$; y_t is the number of goods in stock at the beginning of time t, with $y_1 = a$ as the initial inventory (given); d_t is the demand for the retailer's goods between times t and $t+1$; and c_t, h_t, and b_t denote the per-unit ordering, holding (i.e., storage), and backlogging costs, respectively.[2] Finally, $(y)^+ := \max(0, y)$ denotes the positive part of y.

It is well known that this deterministic problem can be reformulated as a linear program:

$$\begin{aligned} \underset{x_t,\, y_t,\, s_t^+,\, s_t^-}{\text{minimize}} \quad & \sum_{t=1}^{T} \big(c_t x_t + h_t s_t^+ + b_t s_t^-\big) \\ \text{s.t.} \quad & y_{t+1} = y_t + x_t - d_t, \quad \forall\, t, \\ & s_t^+ \ge 0, s_t^- \ge 0, \quad \forall\, t, \\ & s_t^+ \ge y_{t+1}, \quad \forall\, t, \\ & s_t^- \ge -y_{t+1}, \quad \forall\, t, \\ & 0 \le x_t \le M_t, \quad \forall\, t, \\ & y_1 = a, \end{aligned}$$

where $s_t^+ \in \mathbb{R}$ and $s_t^- \in \mathbb{R}$ capture the amount of goods held in storage and the backlogged customer demands during stage t, respectively.[3]

Consider now a setting where the future customer demand is uncertain. More precisely, we assume that each d_t depends on a set of "primitive" uncertainty drivers z, which are

[2] For simplicity, we assume that orders are received instantaneously (i.e., with zero lead time) and that the cost of any remaining inventory or backlog following stage $T+1$ is accounted for in h_T and b_T.

[3] Note that, for the purposes of this deterministic formulation, one could also use a single decision variable $s_t \ge 0$, constrained to satisfy $s_t \ge y_{t+1}, s_t \ge -y_{t+1}$. We retain separate decision variables for the positive and negative parts of y_{t+1} for clarity.

only known to belong to an uncertainty set $\mathcal{Z}$. A naïve but incorrect approach to *robustify* this problem would be to simply state the robust counterpart as

$$\begin{aligned} \underset{x_t,\, y_t,\, s_t^+,\, s_t^-}{\text{minimize}} \quad & \sum_{t=1}^{T} (c_t x_t + h_t s_t^+ + b_t s_t^-) \\ \text{s.t.} \quad & y_{t+1} = y_t + x_t - d_t(z), \quad \forall\, z \in \mathcal{Z}, \forall\, t, \\ & s_t^+ \ge 0,\, s_t^- \ge 0, \quad \forall\, t, \\ & s_t^+ \ge y_{t+1}, \quad \forall\, t, \\ & s_t^- \ge -y_{t+1}, \quad \forall\, t, \\ & 0 \le x_t \le M_t, \quad \forall\, t, \\ & y_1 = a, \end{aligned}$$

where we simply robustified each constraint that involved uncertain parameters by enforcing that the constraint should hold for any value of the uncertainty.

Unfortunately, there are obvious issues with this formulation. The first one is the infeasibility of the constraint

$$y_{t+1} = y_t + x_t - d_t(z), \quad \forall\, z \in \mathcal{Z},$$

because neither the x nor the y variables are allowed to depend on the uncertainty z in this *equality* constraint. Indeed, the above constraint is equivalent to

$$y_{t+1} = y_t + x_t - \tilde{d}, \quad \forall\, \tilde{d} \in \left\{ \tilde{d} \in \mathbb{R} \mid \exists\, z \in \mathcal{Z}, \tilde{d} = d_t(z) \right\}.$$

Unless the implied uncertainty set for $\tilde{d}$ is an interval of zero length (i.e., no uncertainty about d_t), which is unlikely, this constraint is impossible to satisfy.

A simple way of resolving the above issue is to replace $y_t := y_1 + \sum_{t'=1}^{t} (x_{t'} - d_{t'})$ in the deterministic problem *before* deriving the robust counterpart. This would lead to the deterministic model

$$\begin{aligned} \underset{x_t,\, s_t^+,\, s_t^-}{\text{minimize}} \quad & \sum_{t} (c_t x_t + h_t s_t^+ + b_t s_t^-) \\ \text{s.t.} \quad & s_t^+ \ge 0, s_t^- \ge 0, \quad \forall\, t, \\ & s_t^+ \ge y_1 + \textstyle\sum_{t'=1}^{t} x_{t'} - d_{t'}, \quad \forall\, t, \\ & s_t^- \ge -y_1 + \textstyle\sum_{t'=1}^{t} d_{t'} - x_{t'}, \quad \forall\, t, \\ & 0 \le x_t \le M_t \quad \forall\, t, \end{aligned} \tag{2}$$

where $y_1 = a$.

The robust counterpart of this model now takes the form

$$\begin{aligned} \underset{x_t,\, s_t^+,\, s_t^-}{\text{minimize}} \quad & \sum_{t=1}^{T} (c_t x_t + h_t s_t^+ + b_t s_t^-) \\ \text{s.t.} \quad & s_t^+ \ge 0,\, s_t^- \ge 0, \quad \forall\, t, \\ & s_t^+ \ge y_1 + \textstyle\sum_{t'=1}^{t} x_{t'} - d_{t'}(z), \quad \forall\, z \in \mathcal{Z}, \forall\, t, \\ & s_t^- \ge -y_1 + \textstyle\sum_{t'=1}^{t} d_{t'}(z) - x_{t'}, \quad \forall\, z \in \mathcal{Z}, \forall\, t, \\ & 0 \le x_t \le M_t, \quad \forall\, t. \end{aligned} \tag{3}$$

When $\mathcal{Z}$ is bounded, a solution to this model necessarily exists since one can set $x_t = 0$ for all t. Unfortunately, this model still suffers from two important pitfalls, which could easily mislead a practitioner to conclude that robust solutions are necessarily overly conservative.

First, there is the fact that this model makes an important assumption about what type of policy is used. Namely, it assumes that the decision x_t is predefined at time $t = 0$ and never modified, even though some information about early demand might be obtained. This is acceptable when all orders must be placed with suppliers at time $t = 0$ and the contract prevents the retailer to make future modification to these orders. In general, however, it might be possible to adjust a new order placed so as to exploit the available demand information. In this case, we need to allow x_t to be adjustable with respect to $\{d_{t'}\}_{t'=1}^{t-1}$. We will show how this can be done shortly, after considering the second pitfall.

To understand the second issue, which is more subtle, it helps to consider a problem instance in which there is only one stage, $y_1 = 0$, $c_1 = 0.5$, and $h_1 = b_1 = 1$. In this context, we might be interested in the robust counterpart of the deterministic model:

$$\begin{aligned}\underset{x_1}{\text{minimize}} \quad & 0.5x_1 + (x_1 - d_1)^+ + (x_1 + d_1)^+ \\ \text{s.t.} \quad & 0 \le x_1 \le 2,\end{aligned}$$

for a case when $d_1 \in [0, 2]$. Since the problem only has one stage, it is reasonable to assume that no new information will be obtained by the time that the order is implemented. The robust counterpart of this problem instance using the model described in (3) would take the following form:

$$\begin{aligned}\underset{x_1, s_1^+, s_1^-}{\text{minimize}} \quad & 0.5x_1 + s_1^+ + s_1^- \\ \text{s.t.} \quad & s_1^+ \ge 0,\ s_1^- \ge 0, \\ & s_1^+ \ge x_1 - d_1, \quad \forall\, d_1 \in [0, 2], \\ & s_1^- \ge -x_1 + d_1, \quad \forall\, d_1 \in [0, 2], \\ & 0 \le x_1 \le 2.\end{aligned}$$

One can easily verify that the optimal solution here suggests $x_1^* = 0$, ${s_1^+}^* = 0$, and ${s_1^-}^* = 2$ with an optimal value of 2. The $s_1^+ \ge x_1 - d_1, \forall\, d_1 \in [0, 2]$ constraint protects against a worst-case demand of $d_1 = 0$, and the $s_1^- \ge -x_1 + d_1, \forall\, d_1 \in [0, 2]$ constraint protects against a worst-case demand of $d_1 = 2$. Indeed, it is the case that if $x_1 = 0$, then the worst-case scenario would be that a demand of two units occurs and leads to a backlog cost of 2. However, is this truly the best that one can do to reduce worst-case inventory costs?

Consider, for instance, the solution $x_1^{**} = 1$, which would lead to two equivalent worst-case scenarios: (a) a demand of zero units and (b) a demand of two units. Both scenarios lead to a total cost of 1.5, which is smaller than the worst-case cost of $x_1^* = 0$, which is 2.

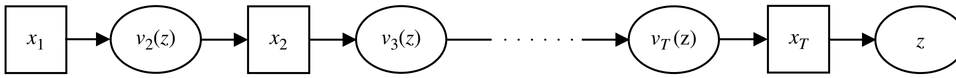
So why did the robust counterpart model obtained from problem (3) not provide the best solution in terms of worst-case cost? The reason lies in the fact that in problem (2), s_t^+ and s_t^- are not authentic decision variables but rather *auxiliary* decision variables that are employed by the linearization scheme that serves to evaluate the objective function. Put another way, the linearization of the piecewise linear terms, which introduces the s_1^+ and s_1^- variables, leads to different worst-case demand values for each constraint of problem (3), although these values cannot be achieved simultaneously. Indeed, the true robust counterpart takes the following form:

$$\begin{aligned}\underset{x_1}{\text{minimize}} \quad & \sup_{d_1 \in [0, 2]} 0.5x_1 + (y_1 + x_1 - d_1)^+ + (-y_1 - x_1 + d_1)^+ \\ \text{s.t.} \quad & 0 \le x_1 \le 1.\end{aligned}$$

This model can in some sense be linearized but only as a *two-stage* problem:

$$\begin{aligned}\underset{x_1}{\text{minimize}} \quad & \sup_{d_1 \in [0, 2]} 0.5x_1 + h(x_1, d_1) \\ \text{s.t.} \quad & 0 \le x_1 \le 2,\end{aligned}$$

FIGURE 1. Chronology of executions x_i, observations v_i, and z.



where

$$h(x_1,d_1) := \min_{s_1^+,s_1^-} \quad \{s_1^+ + s_1^-\}$$
$$\text{s.t.} \quad s_1^+ \geq 0,\, s_1^- \geq 0,$$
$$s_1^+ \geq x_1 - d_1,$$
$$s_1^- \geq -x_1 + d_1.$$

Note that in this linearized formulation, s_1^+ and s_1^- are allowed to depend on the instance of d_1 that is studied. Namely, with $x_1^{**}=1$, they will take the values $s_1^+(d_1) := (1-d_1)^+$ and $s_1^-(d_1) := (d_1-1)^+$. This is different from optimization problem (3), which yielded $x_1^*=0$ because the choice of s_1^+ and s_1^- was forced to be made before the realization of d_1 was known.

Remark 1. This discrepancy arises primarily because the original objective function in the inventory example is nonlinear (actually nonconcave) in d, i.e., given by a sum of maxima of (linear) functions, as per (1). The conservativeness in the reformulation of such objectives has been a topic of active research. For an overview, we direct the interested reader to Bertsimas et al. [33, 34], Gorissen and den Hertog [53], Iancu et al. [56], and Ardestani-Jaafari and Delage [2].

Takeaway message: When robustifying a linear program that involves either (1) decisions that are implemented at different points in time (such as x_t) or (2) auxiliary decision variables whose sole purpose is to aid in the reformulation/computation of the objective value or the validation of a constraint (such as s_t^+ and s_t^-), one must carefully identify the timing of the sequence of decisions and observations and allow any decisions that can be adjustable to depend on (some of) the uncertain quantities in order to take high-quality decisions. This can be done through the adjustable robust counterpart framework introduced in Ben-Tal et al. [16], which we discuss next. Note that in what follows, a decision that can incorporate information revealed so far is referred as an *adjustable decision*, whereas the actual function that maps the revealed information to the action that is implemented is referred as a *decision rule.*

3. The Adjustable Robust Counterpart Model

As seen in the above inventory problem, before developing a robust optimization model, it is important to clearly lay out the chronology of decisions and observations, as portrayed in Figure 1.

Note that in Figure 1, we represent decisions implemented at time t as x_t, and observations made between times $t-1$ and t are denoted by v_t (for *visual* evidence). The observation v_t is a function of z, the underlying uncertainty that affects the entire decision problem. Finally, after the terminal decision x_T is implemented, one can observe the realized uncertain vector z in its entirety to evaluate the objective function and assess whether all the constraints were met.

To be precise, consider a deterministic sequential decision problem that can be written as

$$\underset{\{x_t\}_{t=1}^T}{\text{minimize}} \quad \sum_{t=1}^T c_t^\top x_t + d$$

$$\text{s.t.} \quad \sum_{t=1}^T a_{jt}^\top x_t \leq b_j, \quad \forall\, j=1,\ldots,J,$$

where for each t, the vector $x_t \in \mathbb{R}^n$ describes the different decisions (continuous variables) that will be implemented at time t, $c_t \in \mathbb{R}^n$ describes the per-unit cost generated from each decision, and $a_{jt} \in \mathbb{R}^n$ describes the (per-unit) amount of resources of type j used among a total set of b_j units. In practice, a number of reasons might motivate considering any of the parameters c_t, a_{jt}, or b_j as uncertain. In this case, one must consider the following robust formulation:

$$\text{(ARC)} \quad \underset{x_1, \{x_t(\cdot)\}_{t=2}^T}{\text{minimize}} \; \sup_{z \in \mathcal{Z}} \left\{ c_1(z)^\top x_1 + \sum_{t=2}^T c_t(z)^\top x_t(v_t(z)) + d(z) \right\}$$

$$\text{s.t.} \quad a_{j1}(z)^\top x_1 + \sum_{t=2}^T a_{jt}(z)^\top x_t(v_t(z)) \le b_j(z), \quad \forall z \in \mathcal{Z}, \forall j = 1, \ldots, J, \tag{4}$$

where for each t and each j, the functions $c_t(z)$, $d(z)$, $a_{jt}(z)$, and $b_j(z)$ are affine functions of $z \in \mathbb{R}^m$, which is considered the fundamental source of uncertainty, where $v_t\colon \mathbb{R}^m \to \mathbb{R}^\nu$ is a mapping[4] from the uncertainty space to the (partial) observation space $\mathbb{R}^\nu$ made at time t, and where x_t is a mapping from the space of observations $\mathbb{R}^\nu$ to the space of actions $\mathbb{R}^n$ implemented at time t. The fact that each x_t is no longer a vector but rather a mapping is important because it enables the decision maker to react differently depending on the realized observation. Of course, this flexibility comes at the price of significant computational challenges.

Remark 2. The most famous example of observation mapping is one that simply reveals at each stage an additional subset of the terms in z. This is often referred to as the property that *z is progressively revealed*. Mathematically speaking, let z be composed of $T-1$ vectors $\{z_t\}_{t=1}^{T-1}$, with $z_t \in \mathbb{R}^{m'}$ such that $z := [z_1^\top \; z_2^\top \; \cdots \; z_{T-1}^\top]^\top$. One considers z to be *progressively revealed* if $v_t(z) = z_{[t-1]} = [z_1^\top \; z_2^\top \; \cdots \; z_{t-1}^\top]^\top$. In this context, one can consider that v_t is a linear mapping $v_t(z) := V_t z$, where the observation matrix $V_t \in \mathbb{R}^{\nu \times m}$, with $\nu := m$, is described as follows:

$$V_t := \begin{bmatrix} \mathbf{I}_{(t-1)m' \times (t-1)m'} & \mathbf{0}_{(t-1)m' \times (T-t)m'} \\ \mathbf{0}_{(T-t)m' \times (t-1)m'} & \mathbf{0}_{(T-t)m' \times (T-t)m'} \end{bmatrix}.$$

In de Ruiter et al. [43], the authors also discuss extensively the notion of *inexactly revealed data*, referring to the idea that at each stage of time it is not $z_{[t-1]}$ that is revealed but rather a noisy measurement $\hat{z}_{[t-1]} \approx z_{[t-1]}$. This framework can also be represented using a linear observation mapping after augmenting the uncertainty space $z \in \mathcal{Z}$ to $(\hat{z}, z) \in \mathcal{U}$, where $\mathcal{U}$ captures the relation between z and $\hat{z}$ (e.g., $\|z - \hat{z}\|_2 \le \gamma$) and for which $\{z \in \mathbb{R}^m \mid \exists \hat{z}, (\hat{z}, z) \in \mathcal{U}\} = \mathcal{Z}$. It is then possible to model the progressively revealed measurements using

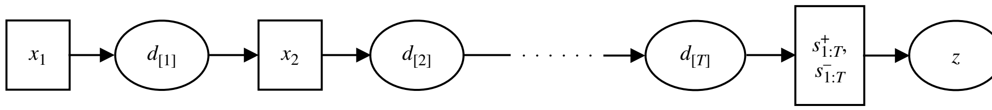
$$V_t := \begin{bmatrix} \mathbf{I}_{(t-1)m' \times (t-1)m'} & \mathbf{0}_{(t-1)m' \times (T-t)m'} & \mathbf{0}_{(t-1)m' \times m}, \\ \mathbf{0}_{(T-t)m' \times (t-1)m'} & \mathbf{0}_{(T-t)m' \times (T-t)m'} & \mathbf{0}_{(T-t)m' \times m} \end{bmatrix},$$

such that $v_t([\hat{z}^\top \; z^\top]^\top) := V_t [\hat{z}^\top \; z^\top]^\top = \hat{z}_{[t-1]}$.

Example 1. Considering the inventory problem presented in §2, one might consider that at each point of time, the inventory manager is able to observe all of the prior demand before placing an order for the next stage. This could apply, for instance, to the case when the entire unmet demand from customers is backlogged, so censoring of observations never arises. One might then define the sequence of decision variables and observations according to Figure 2.

[4] Note that although we will assume that this mapping is linear, the methods presented in §5.3 could be used to handle nonlinear mappings.

FIGURE 2. Timing of decisions and observations in an inventory problem.



Note that in Figure 2, we made explicit that s_t^+ and s_t^- are auxiliary variables that are adjustable with respect to the full uncertainty vector $d_{[T]} = d$. In fact, once $d_{[T]}$ is revealed, the uncertainty can be considered reduced to zero; hence the role of z in this chronology is somewhat artificial. We are thus left with the following multistage adjustable robust counterpart model:

$$\begin{aligned}
\underset{x_1,\{x_t(\cdot)\}_{t=2}^T,\{s_t^+(\cdot),s_t^-(\cdot)\}_{t=1}^T}{\text{minimize}} \quad & \sup_{d\in\mathcal{U}}\left\{c_1x_1 + \sum_t (c_tx_t(d_{[t-1]}) + h_ts_t^+(d) + b_ts_t^-(d))\right\}\\
\text{s.t.} \quad & s_t^+(d)\geq 0, s_t^-(d)\geq 0, \quad \forall d\in\mathcal{U}, \forall t,\\
& s_t^+(d)\geq y_1 + \sum_{t'=1}^t x_{t'}(d_{[t'-1]}) - d_{t'}, \quad \forall d\in\mathcal{U}, \forall t,\\
& s_t^-(d)\geq -y_1 + \sum_{t'=1}^t d_{t'} - x_{t'}(d_{[t'-1]}), \quad \forall d\in\mathcal{U}, \forall t,\\
& 0\leq x_t(d_{[t'-1]})\leq M_t, \quad \forall d\in\mathcal{U}, \forall t,
\end{aligned} \tag{5}$$

where $\mathcal{U}\subseteq\mathbb{R}^T$ captures the set of potential demand vectors.

As explained in Ben-Tal et al. [16], in most cases the adjustable robust counterpart is computationally intractable (NP-hard). Below, we provide the proof of the NP-hardness result to illustrate proof techniques that arise in robust optimization.

Proposition 1. *Solving problem (4) is NP-hard even when $v_t(z)=z$ and $\mathcal{Z}$ is polyhedral.*

Proof. This result is obtained by showing that the NP-complete 3-SAT problem can be reduced to verifying whether the optimal value of the following problem is greater than or equal to zero:

$$\begin{aligned}
\underset{x(\cdot)}{\text{minimize}} \quad & \sup_{z\in[0,1]^m}\sum_{i=1}^N (x_i(z)-1)\\
\text{s.t.} \quad & x_i(z)\geq a_{i,k}^\top z + b_{i,k}, \quad \forall z\in\mathcal{Z}, \forall i=1,\ldots,N, \forall k=1,\ldots,K,
\end{aligned} \tag{6}$$

where $z\in\mathbb{R}^m$ is the uncertain vector, x_i: $\mathbb{R}^m\to\mathbb{R}$ is a second-stage decision vector, and $a_{i,k}\in\mathbb{R}^m$ and $b_{i,k}\in\mathbb{R}$ are known parameters of the model. Note that the above problem is the adjustable robust counterpart of

$$\begin{aligned}
\underset{x\in\mathbb{R}^N}{\text{minimize}} \quad & \sum_{i=1}^N (x_i-1)\\
\text{s.t.} \quad & x_i\geq b_{i,k}, \quad \forall i=1,\ldots,N, \forall k=1,\ldots,K,
\end{aligned}$$

in a case where each $b_{i,k}$ is uncertain and each x_i is fully adjustable.

3-SAT problem. Let W be a collection of disjunctive clauses $W=\{w_1,w_2,\ldots,w_N\}$ on a finite set of variables $V=\{v_1,v_2,\ldots,v_m\}$ such that $|w_i|=3$ $\forall i\in\{1,\ldots,N\}$. Let each clause be of the form $w=v_i\vee v_j\vee \bar{v}_k$, where $\bar{v}$ is the negation of v. Is there a truth assignment for V that satisfies all the clauses in W?

Given an instance of the 3-SAT problem, we can attempt to verify whether the optimal value of the following problem is greater than or equal to zero:

$$\begin{aligned} \max_{z} \quad & \sum_{i=1}^{N} (h_i(z) - 1) \\ \text{s.t.} \quad & 0 \le z_j \le 1, \quad \forall j = 1, \ldots, m, \end{aligned} \tag{7}$$

where $z \in \mathbb{R}^m$ and where $h_i(z) := \max\{z_{j_1}, z_{j_2}, 1 - z_{j_3}\}$ if the ith clause is $w_i = v_{j_1} \vee v_{j_2} \vee \bar{v}_{j_3}$. It is straightforward to confirm that $\{z \in \mathbb{R}^m \mid 0 \le z_j \le 1, \forall j\}$ is a polyhedron and that each $h_i(z)$ can be expressed as $h_i(z) := \max_k (a_{i,k}^\top z + b_{i,k})$. Hence, problem (7) can be expressed in the form of problem (6). Finally, the answer to the 3-SAT problem is positive if and only if the optimal value of an instance of problem (6) achieves an optimal value greater than or equal to zero. □

The fact that the problem of computing the robust adjustable counterpart is NP-hard motivates the use of other techniques to address robust multistage decision making. Section 4 discusses the connection with robust dynamic programming, and §5 provides an overview of simple policies that can be employed.

4. Connections with Robust Dynamic Programming

Similarly to stochastic multistage decision models, one can attempt to circumvent some of the computational difficulties affecting the robust adjustable problem by using the paradigm of dynamic programming (DP) (see Bertsekas [19] for a general overview; Iyengar [57] and Nilim and El Ghaoui [65] for a treatment of Markov decision problems with finite state spaces and uncertain transition probabilities; and Ben-Tal et al. [15], Bertsimas et al. [33], and Iancu et al. [56] for models closer to the ones considered here). To this end, it is instructive to first rewrite the problem as a sequential min-max game between the decision maker and nature. To provide a concrete example and ground our discussion, recall the inventory model of Example 1, and note that the problem of deciding the optimal robust ordering quantities x_t so as to minimize the worst-case ordering and holding/backlogging costs can be rewritten as the following sequential decision problem:

$$\begin{aligned} \min_{0 \le x_1 \le M_1} \Big[c_1 x_1 + \max_{d_1 \in \mathcal{U}_1(\varnothing)} \Big[h_1 (y_2)^+ + b_1 (-y_2)^+ + \min_{0 \le x_2 \le M_2} \Big[c_2 x_2 + \max_{d_2 \in \mathcal{U}_2(d_1)} \Big[h_2 (y_3)^+ \\ + b_2(-y_3)^+ + \cdots + \min_{0 \le x_t \le M_t} \Big[c_T x_T + \max_{d_T \in \mathcal{U}_T(d_{[T-1]})} \Big[h_T (y_{T+1})^+ + b_T(-y_{T+1})^+ \Big] \Big] \ldots \Big] \Big] \Big] \Big], \end{aligned}$$

where $y_{t+1} = y_t + x_t - d_t$ and where

$$\mathcal{U}_t(d_{[t-1]}) := \big\{ d \in \mathbb{R} \colon \exists \xi \in \mathbb{R}^{T-t} \text{ such that } [d_{[t-1]}^\top \;\; d \;\; \xi^\top]^\top \in \mathcal{U} \big\}, \quad \forall\, d_{[t-1]} \in \mathbb{R}^{t-1}$$

denotes the set of demand values in stage t that are consistent with the realized sequence of observations $d_{[t-1]}$ and the original uncertainty set $\mathcal{U}$, while, similarly, $\mathcal{U}_1(\varnothing) := \{d \in \mathbb{R}\colon \exists \xi \in \mathbb{R}^{T-1}, [d \;\; \xi^\top]^\top \in \mathcal{U}\}$, and we still have $y_1 = a$. Note that this DP formulation explicitly states the rule that the decision maker should use when updating the uncertainty set based on past observations—namely, that future feasible values of the uncertainties should be "conditioned" on past ones through the set $\mathcal{U}_t(d_{[t-1]})$ at time t (similar to the conditioning operation for random variables, in stochastic optimization). We return to discuss this further in §6, where we highlight some time consistency issues that might arise when this rule is not followed.

In this context, it can be readily seen that the *state* of the system at time t is given by

$$S_t := [y_t \;\; d_{[t-1]}^\top]^\top = [y_t \;\; d_1 \;\; d_2 \;\; \cdots \;\; d_{t-1}]^\top \in \mathbb{R}^t,$$

with the recursive dynamical equation $y_{t+1} = y_t + x_t - d_t$. Moreover, letting $J_t^*(S_t)$ denote the optimal *value function* at time t, the typical DP Bellman recursion can be rewritten as

$$J_t^*(S_t) = \min_{0 \le x_t \le M_t} \Big[c_t x_t + \max_{d_t \in \mathcal{U}_t(d_{[t-1]})} \big[h_t (y_{t+1})^+ + b_t(-y_{t+1})^+ + J_{t+1}^*(S_{t+1}) \big] \Big],$$

where $S_{t+1} := [y_t + x_t - d_t \;\; d_1 \;\; d_2 \;\; \cdots \;\; d_{t-1} \;\; d_t]^\top$, and $J_{T+1}(S) = 0, \forall\, S$. Several remarks concerning this DP formulation are in order.

First, note that for a general uncertainty set $\mathcal{U} \subseteq \mathbb{R}^T$, the state S_t is a t-dimensional vector, as full knowledge of the history of observations up to stage t is required in order to describe the uncertainty set for the subproblem over stages $t, t+1, \ldots, T$. Since the complexity of the underlying Bellman recursions explodes with the number of state variables (Bertsekas [19]), this severely limits the practical applicability of the DP framework for finding exact policies and value functions. This manifestation of the well-known "curse of dimensionality" is consistent with the hardness result concerning general adjustable policies in Proposition 1. Thus, in practice, one would have to either solve the recursions numerically, e.g., by multiparametric programming (Bemporad et al. [6]), or resort to approximate DP techniques (Bertsekas [19], Powell [68]), sampling (Calafiore and Campi [38]), or other methods.

It is important to note that, when the uncertainty sets possess additional structure, a reduction in the DP state space may be possible. For instance, if the uncertainty set in our inventory model is given by a hypercube (e.g., $\mathcal{U} = \times_{t=1}^{T} [\underline{d}_t, \bar{d}_t]$ for some $\underline{d}_t \le \bar{d}_t, \forall\, t$), the on-hand inventory is a sufficient state; i.e., $S_t = y_t$ (Ben-Tal et al. [15], Bertsimas et al. [33]). Similarly, for the budgeted uncertainty set of Bertsimas and Sim [27] and Bertsimas and Thiele [28],

$$\mathcal{U} = \big\{ d \in \mathbb{R}^T \colon \exists z \in [0,1]^T \text{ such that } \|z\|_\infty \le 1, \|z\|_1 \le \Gamma, d_t = \bar{d}_t + \hat{d}_t z_t, \forall\, t \in \{1, \ldots, T\} \big\},$$

where $\bar{d}, \hat{d} \in \mathbb{R}^T$ are given data, and $\Gamma \ge 0$ is the budget of uncertainty, it can be readily checked that a sufficient state is given by the two-dimensional vector $S_t = [y_t \;\; \sum_{\tau=1}^{t-1} |z_\tau|]^T$.[5] This reduction in the state space—enabled by imposing additional structure on the uncertainty set—may carry several benefits, by (i) enabling computational tractability when the overall dimension of the state remains small, as well as (ii) affording a characterization of certain structural properties of the optimal policy or value function, which can be insightful in practice. To provide a concrete example of the latter, note that when $\mathcal{U}$ is a hypercube, the model resembles a classical model in inventory management, for which it is known that the optimal ordering policy follows a *modified base-stock* rule (Kasugai and Kasegai [58]),

$$x_t = \min\big(M_t, \max(0, \theta_t - y_t)\big). \tag{8}$$

Here, θ_t is the optimal base-stock value; thus, in the absence of any order capacities (i.e., $M_t = \infty$), the optimal policy is to raise the inventory position to θ_t whenever the initial level y_t is below this threshold and to not order otherwise. The presence of bounds simply truncates these orders at the capacity M_t. Furthermore, such a DP formulation could also be used to prove useful comparative statics, such as the fact that the optimal order quantity x_t decreases in the initial inventory y_t while the inventory position after ordering, i.e., $x_t + y_t$, increases in y_t—results that may hold more generally (see Iancu et al. [56] for a discussion and more references).

Our final remark concerns a more subtle point, which often goes unnoticed in the context of robust multistage decision problems: although a DP solution approach is always *sufficient* to solve our problem, it is generally not *necessary*, and it may impose unnecessarily stringent

[5] Similar arguments can be used to reduce the ellipsoidal uncertainty sets of Ben-Tal et al. [15] or the budgeted uncertainty sets of Bandi and Bertsimas [4] to a two-dimensional state for our inventory problem. We omit the details for brevity.

requirements on the solution process. To better understand and appreciate this point, it is illustrative to consider a very simple, two-stage instance of our inventory model, where no ordering and no costs are incurred in the first stage, and demand is zero in the second stage (the ideas presented readily generalize to arbitrary multistage decision problems and uncertainty sets). More precisely, assume that $T=2$, $y_1=x_1=0$, $h_1=b_1=0$, $M_2=\infty$, $d_1\in[\underline{d}_1,\bar{d}_1]$, and $d_2=0$. Then, omitting the time indices, the decision problem can be rewritten as

$$\max_{d\in[\underline{d},\,\bar{d}]}\ \min_{0\le x} f(d,x), \tag{9}$$

where $f(d,x):=cx+h(-d+x)^+ + b(d-x)^+$. Let J^* denote the optimal value in the max-min problem above. In this context, according to (8), a DP approach would yield the *Bellman-optimal* policy $x^*(d)=\max(0,\theta^*-d)$, where θ^* depends only on problem parameters. By definition, we readily have that

$$J^*=\max_{d\in[\underline{d},\,\bar{d}]} f(d,x^*(d)).$$

Thus, the policy $x^*(d)$ is clearly *sufficient* for the purposes of finding the min-max optimal value J^*. Furthermore, and consistent with the Bellman optimality criterion, the costs in the subproblem of stage $t=2$ (which are identical with $f(d,x)$ here) are minimized for *any* realization of d when the policy $x^*(d)$ is followed.

However, is the policy $x^*(d)$ *necessary* for optimality, i.e., to achieve the overall cost J^* in the multistage decision problem, from the perspective of stage $t=0$? As it turns out, the answer is emphatically *negative*; moreover, there exists an arbitrary number of policies that achieve the same worst-case optimal value J^*. To see this, let us define the following set of policies:

$$\mathcal{X}^{\text{wc}}:=\{x\colon [\underline{d},\bar{d}]\to\mathbb{R}^+\colon\ f(d,x(d))\le J^*,\ \forall\, d\in\mathcal{U}\}.$$

By construction, any policy $x\in\mathcal{X}^{\text{wc}}$ is feasible in (9) and achieves the same optimal worst-case value as the Bellman-optimal policy $x^*(d)$, i.e., $J^*=\max_{d\in[\underline{d},\,\bar{d}]} f(d,x(d))$, since

$$\max_{d\in[\underline{d},\,\bar{d}]} f(d,x(d))\le J^*=\max_{d\in[\underline{d},\,\bar{d}]} f(d,x^*(d))\le \max_{d\in[\underline{d},\,\bar{d}]} f(d,x(d)).$$

We thus refer to $\mathcal{X}^{\text{wc}}$ as the set of *worst-case optimal* policies in the robust dynamic problem (9).

It is worth noting that $\mathcal{X}^{\text{wc}}$ is clearly nonempty, since $x^*\in\mathcal{X}^{\text{wc}}$. Furthermore, since $f(d,x)$ is convex in x for any fixed d, $\mathcal{X}^{\text{wc}}$ is a convex set. We claim that $\mathcal{X}^{\text{wc}}$ generally contains an infinite number of policies. For instance, consider the following *affine* policy:

$$x^{\text{aff}}(d)=x^*(\underline{d})+\frac{x^*(\bar{d})-x^*(\underline{d})}{\bar{d}-\underline{d}}(d-\underline{d}),$$

obtained by linearly interpolating $x^*(d)$ at the extreme points of $\mathcal{U}$. Since both $f(d,x^*(d))$ and $f(d,x^{\text{aff}}(d))$ are convex in d, and since $x^{\text{aff}}(d)=x^*(d),\ \forall\, d\in\{\underline{d},\bar{d}\}$, we immediately have that

$$\max_{d\in[\underline{d},\,\bar{d}]} f(d,x^{\text{aff}}(d))=\max_{d\in\{\underline{d},\,\bar{d}\}} f(d,x^{\text{aff}}(d))=\max_{d\in\{\underline{d},\,\bar{d}\}} f(d,x^*(d))=\max_{d\in[\underline{d},\,\bar{d}]} f(d,x^*(d))=J^*.$$

Thus, the affine policy x^{aff} is worst-case optimal and generally different from the Bellman-optimal (piecewise-affine) policy $x^*(d)$. Furthermore, any policy of the form $\lambda x^*(d)+(1-\lambda)x^{\text{aff}}(d)$ is also worst-case optimal for any $\lambda\in[0,1]$.

The degeneracy that we have uncovered here is by no means an exception but rather a general fact, intrinsic in any robust multistage decision model. In fact, the concept of worst-case optimality introduced above can be readily extended to more general settings, with multiple decision stages, more complex uncertainty sets, and/or more complicated dynamics; we omit those formulations for conciseness and focus on the key insights and implications here (see Bertsimas et al. [33], Iancu et al. [56] for more details). Intuitively, in any sequential min-max game against an adversary ("nature"), when the adversary does not play the optimal (i.e., worst-case) response at a particular stage, the decision maker also has the freedom to play a slightly suboptimal response, provided that the overall objective is not deteriorated beyond the optimal worst-case value. This is exactly the requirement defining any policy $x \in \mathcal{X}^{\mathrm{wc}}$, and it is a *necessary* condition for worst-case optimality. By contrast, the Bellman optimality criterion is much more stringent, as it requires the decision maker to follow an optimal policy for the ensuing subproblems even when the adversary plays suboptimally.

It is worth noting that this concept (and degeneracy) is unique to *robust* multistage decision making. In particular, for a general stochastic multistage decision problem with an objective involving expected outcomes, any optimal policy must satisfy the Bellman optimality criterion in any state and at any time, since doing otherwise would immediately translate into a worse performance in expectation.

At this point, the reader might pause and question the relevance of the degeneracy: we thus conclude our discussion by highlighting a key positive outcome, as well as a potential pitfall and a fix thereof. First, note that the presence of degeneracy can be helpful for a decision maker, since it may give rise to worst-case optimal policies with "simple structure"—for instance, a static or an affine policy (such as x^{aff}). Such policies are attractive in terms of practical implementation as well as computationally, as one may exploit the simple structure to devise tractable algorithms for finding them. We return to this issue in more detail in §5.2, where we explore several nontrivial implications of this concept in robust multistage decision problems.

From a different standpoint, the degeneracy inherent in $\mathcal{X}^{\mathrm{wc}}$ may come across as unattractive, since it may generate inefficiencies in the decision process. In particular, note that any policy $x \in \mathcal{X}^{\mathrm{wc}}$ with $x \neq x^*$ has the property that $f(d, x(d)) \geq f(d, x^*(d))$; i.e., x generates costs at least as large as $x^*(d)$. In particular, barring exceptional circumstances where the Bellman-optimal policy is itself degenerate, this implies that the worst-case optimal policy $x \in \mathcal{X}^{\mathrm{wc}}$ is Pareto-dominated by x^*, leading to justified objections concerning the actual enforcement of policy x. The key to resolving this issue is to recognize that the presence of degeneracy in the (worst-case) dynamic policies should be useful only as an existential result, and any computational procedure based on it should be implemented in a shrinking horizon fashion (Iancu et al. [56]). More precisely, suppose we have information that a particular class of tractable dynamic policies is worst-case optimal, and we have a black-box algorithm for generating one such policy. Then, in stage $t = 1$, one could use this algorithm to generate a set of worst-case policies $\{x_1, x_2, x_3, \ldots, x_T\}$ but only implement the first-stage action x_1, which is a constant and equal to the Bellman optimal action corresponding to the first stage. Once the first-stage uncertainties are materialized, the black-box algorithm can be again used to generate a new set of worst-case optimal actions, of which only the first (constant) action is implemented. By repeating this process in a shrinking horizon fashion, one can effectively reconstruct the decisions taken by a Bellman-optimal policy on the particular sample path of uncertain values, which is the best possible course of action.

5. Simple Policies and Their Optimality

5.1. Static Policies

The simplest type of policy to consider is a static one, whereby all future decisions are constant and independent of the intermediate observations. Such policies do not increase the underlying complexity of the decision problem, and often they result in tractable robust counterparts (Ben-Tal et al. [13]).

Despite their simplicity, static policies often have surprisingly good performance in practice and are known to be in fact *optimal* in several cases. Such an example is described in Ben-Tal and Nemirovski [8], in the context of a generic linear optimization problem with *row-wise* (i.e., *constraintwise*) uncertainty:

$$\begin{aligned} \underset{x(\cdot)}{\text{minimize}} \quad & \sup_{z\in\mathcal{Z}} [c(z_0)^\top x(z) + d(z_0)] \\ \text{s.t.} \quad & a_j(z_j)^\top x(z) \le b_j(z_j), \quad \forall\, z \in \mathcal{Z}, \forall\, j = 1, \ldots, J. \end{aligned} \tag{10}$$

In such a model, the uncertainty vector can be partitioned into $J+1$ blocks, i.e., $z = [z_0^\top \; z_1^\top \; \cdots \; z_J^\top]^\top$ and $\mathcal{Z} = \mathcal{Z}_0 \times \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_J$, such that the data in the objective depend solely on $z_0 \in \mathcal{Z}_0$ (in an affine fashion), whereas the data in the jth constraint depend solely on $z_j \in \mathcal{Z}_j$ (also in an affine fashion). Under these conditions, a static policy $x(z) = x$ is optimal for problem (10) (see Ben-Tal and Nemirovski [8] as well as Ben-Tal et al. [13, Chapter 14] for more details). This result has been recently extended in Bertsimas et al. [30] for uncertain packing problems, where the authors linked the optimality of static policies to the convexity of a particular transformation of the uncertainty set.

In typical multistage decision problems, it is likely that the conditions discussed above do not hold. For instance, that is the case even in our motivating example in inventory management, as problem (5) does not have a row-wise structure (the demand from a particular stage affects multiple constraints). In such circumstances, a natural question concerns the performance guarantees associated with static policies—for an overview of the latest results, we direct interested readers to Bertsimas and Goyal [24], Goyal and Lu [54], and references therein.

5.2. Affine Decision Rules

The paper by Ben-Tal et al. [16] was the first to suggest using affine decision rules to approximate the adjustable robust counterpart (ARC) problem presented in (4). The multistage version of this approximation scheme is better known under the name of affinely adjustable robust counterpart (AARC) and seeks the optimal solution to the following model:

$$\text{(AARC)} \qquad \begin{aligned} \underset{\{x_t\}_{t=1}^T, \{X_t\}_{t=2}^T}{\text{minimize}} \quad & \sup_{z\in\mathcal{Z}} \left\{ c_1(z)^\top x_1 + \sum_{t=2}^T c_t(z)^\top (x_t + X_t v_t(z)) + d(z) \right\} \\ \text{s.t.} \quad & a_{j1}(z)^\top x_1 + \sum_{t=2}^T a_{jt}(z)^\top (x_t + X_t v_t(z)) \le b_j(z), \\ & \begin{cases} \forall\, z \in \mathcal{Z}, \\ \forall\, j = 1, \ldots, J, \end{cases} \end{aligned} \tag{11}$$

where each adjustable policy $x_t(\cdot)$ was replaced with an affine decision rule representation $x_t(\bar{v}) := x_t + X_t \bar{v}$ and where the optimization is now made over the finite-dimensional space spanned by the set of decision vectors $x_t \in \mathbb{R}^n$ and decision matrices $X_t \in \mathbb{R}^{n\times\nu}$. To obtain a tractable reformulation of this model, one needs the following assumption.

Assumption 1. *The ARC model has fixed recourse and all observations are linear functions of z. Mathematically speaking, we make the following two assumptions*:

(1) *For all $t = 2, \ldots, T$ and all $j = 1, \ldots, J$, the affine mappings $c_t(z)$ and $a_{jt}(z)$ are constant; i.e., $c_t(z) = c_t$ and $a_{jt}(z) = a_{jt}$.*

(2) *For all $t = 2, \ldots, T$, the observations $v_t(\cdot)$ can be described as $v_t(z) := V_t z$ for some $V_t \in \mathbb{R}^{\nu\times m}$.*

Proposition 2. *Let the ARC model (4) satisfy Assumption 1 and the uncertainty set $\mathcal{Z}$ be a bounded polyhedron. Then the AARC model (11) can be described as the robust linear program:*

$$\begin{aligned} \underset{\{x_t\}_{t=1}^T, \{X_t\}_{t=2}^T}{\text{minimize}} \quad & \sup_{z\in\mathcal{Z}} \left\{ c_1(z)^\top x_1 + \sum_{t=2}^{T} c_t^\top (x_t + X_t V_t z) + d(z) \right\} \\ \text{s.t.} \quad & a_{j1}(z)^\top x_1 + \sum_{t=2}^{T} a_{jt}^\top (x_t + X_t V_t z) \le b_j(z), \quad \forall z \in \mathcal{Z}, \forall j = 1, \dots, J, \end{aligned} \tag{12}$$

and it can therefore be reformulated as a linear program with a finite number of decisions and a finite number of constraints. Furthermore, the optimal affine decision rules and optimal objective value obtained from the AARC model provide a conservative approximation of the ARC model; i.e., the optimal affine decision rules are implementable in ARC and achieve the optimal objective value obtained by problem (12), thus providing an upper bound on the true optimal value of ARC.[6]

Proof. The proof for this proposition is fairly straightforward. Problem (12) is obtained simply after replacing the observation mapping $v_t(\cdot)$ with its linear definition. One can then establish that the objective function and each constraint involve affine functions of the decision variables and of the perturbation z, hence making it an instance of a robust linear program for which we can obtain a tractable reformulation (see Ben-Tal and Nemirovski [8] for a description of the fundamental theory involved in obtaining this reformulation). Perhaps as important is the fact that once the optimal solution is obtained in terms of $\{x_t^*\}_{t=1}^T$ and $\{X_t^*\}_{t=2}^T$, it is possible to construct a decision rule $x_t(\bar{v}) := x_t^* + X_t^* \bar{v}_t$ that will satisfy all constraints of the ARC model and for which the objective value is reduced to

$$\sup_{z\in\mathcal{Z}} \left\{ c_1(z)^\top x_1^* + \sum_{t=2}^{T} c_t^\top (x_t^* + X_t^* V_t z) + d(z) \right\},$$

which is exactly the optimal value of the AARC model. □

Example 2. Looking back at the inventory problem presented in Example 1 where we were trying to identify ordering strategies robust to demand uncertainty, as portrayed by $d \in \mathcal{U} \subset \mathbb{R}^T$, we observe that the multistage ARC model (5) satisfies Assumption 1. Namely, the recourse is fixed since all x_t are only multiplied by coefficients that are certain and the observations are linear functions of the uncertain vector d. The affinely adjustable robust counterpart of this inventory model can be presented as (with the initial inventory y_1 given)

$$\begin{aligned} \underset{x_1, \{x_t, X_t\}_{t=2}^T, \{s_t^+, S_t^+\}_{t=1}^T, \{s_t^-, S_t^-\}_{t=1}^T}{\text{minimize}} \quad & \sup_{d\in\mathcal{U}} \left\{ c_1 x_1 + \sum_{t=1}^{T} \left[c_t(x_t + X_t V_t d) + h_t(s_t^+ + S_t^+ d) \right.\right. \\ & \qquad \left.\left. + b_t(s_t^- + S_t^- d) \right] \right\} \\ \text{s.t.} \quad & s_t^+ + S_t^+ d \ge 0, \; s_t^- + S_t^- d \ge 0, \quad \forall d \in \mathcal{U}, \forall t, \\ & s_t^+ + S_t^+ d \ge y_1 + \sum_{t'=1}^{t} \left[x_{t'} + X_{t'} V_{t'} d - d_{t'} \right], \\ & \qquad \forall d \in \mathcal{U}, \forall t, \end{aligned}$$

[6] Note that this upper bound could potentially be infinite when there are no affine decision rules that are feasible, even in cases where a finite worst-case value is achievable in the actual robust multistage decision problem.

$$s_t^- + S_t^- d \geq -y_1 + \sum_{t'=1}^{t} \big[d_{t'} - (x_{t'} + X_{t'} V_{t'} d)\big], \quad \forall d \in \mathcal{U}, \forall t,$$

$$0 \leq x_t + X_t V_t d \leq M_t, \quad \forall d \in \mathcal{U}, \forall t, \tag{13}$$

where $X_t \in \mathbb{R}^{1\times T}$, $S_t^+ \in \mathbb{R}^{1\times T}$, $S_t^- \in \mathbb{R}^{1\times T}$, and where V_t is such that $V_t d = [d_1 \ \cdots \ d_{t-1} \ 0 \ \cdots \ 0]$.[7]

For cases without fixed recourse, Ben-Tal et al. [16] suggested several approximation techniques, using tools derived from linear systems and control theory. In Ben-Tal et al. [11, 12], the same approach was extended to multistage linear dynamical systems affected by uncertainty, and tractable exact or approximate reformulations were presented, which allow for the computation of affine decision rules.

Affine decision rules, which are strictly more flexible than static policies, have been found to deliver excellent performance in a variety of applications (see, e.g., Adida and Perakis [1], Babonneau et al. [3], Ben-Tal et al. [15], Mani et al. [62]). Ben-Tal et al. [15] performed simulations in the context of a supply chain contracting problem, and the authors found that in only 2 of 300 instances were the affine decision rules suboptimal (in fact, Ben-Tal et al. [13, Chapter 14] contains a slight modification of the model in Ben-Tal et al. [15], for which the authors find that in *all* tested instances, the affine class is optimal). By implementing affine decision rules both in the primal and dual formulations, Kuhn et al. [59] also investigated the optimality gap of such decision rules in a related application.

However, despite ample evidence of empirical success, we are only aware of a small set of results that characterizes the optimality of such decision rules. One such example is a linear optimization problem with fully adjustable decisions and a simplex uncertainty set (see Ben-Tal et al. [13, Lemma 14.3.6]). More formally, consider a two-stage adjustable decision problem:

$$\begin{aligned} \underset{x(\cdot)}{\text{minimize}} \quad & \sup_{z\in\mathcal{Z}} [c^\top x(z) + d(z)] \\ \text{s.t.} \quad & a_j^\top x(z) \leq b_j(z), \quad \forall z \in \mathcal{Z}, \forall j = 1,\ldots,J, \end{aligned}$$

where $d(z)$ and $b_j(z)$ depend affinely on the uncertainty. It is known that if $\mathcal{Z} \subseteq \mathbb{R}^m$ is a simplex, then an affine decision rule $x + Xz$ is optimal. The intuition behind this result is straightforward since affine decision rules allow sufficient degrees of freedom to match an arbitrary policy at the extreme points of the uncertainty set. More precisely, for any arbitrary policy $x^*(z)$, one can always find a vector x and a matrix X such that $x^*(z) = x + Xz$, $\forall z \in \text{ext}(\mathcal{Z})$. Since the adversary always seeks to maximize linear functions of z (under affine decision rules), it is optimal to choose $z \in \text{ext}(\mathcal{Z})$, which then preserves optimality.

A second instance where affine decision rules are known to be optimal is an inventory model closely related to our motivating example 1. In particular, Bertsimas et al. [33] show that, if the demands in successive stages are independent and only known to belong to certain intervals, i.e., if $\mathcal{U} = \times_{t=1}^{T} [\underline{d}_t, \bar{d}_t]$, then the AARC formulation in (13) recovers the optimal worst-case value (these results have been extended to more general uncertainty sets and cost structures; see Iancu et al. [56] for details). We note that this result is surprising on two fronts, since the AARC model generally introduces *two* layers of potential suboptimality: (1) by restricting the replenishment policies x_t to be affine decision rules and (2) by restricting the reformulation variables s_t^+ and s_t^- to be affine, instead of piecewise affine.[8] In the context

[7] Note that the size of this AARC model could be reduced by accounting for the fact that the observation vector v_t is smaller for smaller t. For simplicity of presentation, we choose to leave it this way, as we understand that some terms of X_t will always be multiplied to zero and can therefore be set arbitrarily.

[8] It can be readily checked that, under affine decision rules x_t, the auxiliary variables depend in a piecewise affine fashion on the observations, e.g., $s_t^+(d) = \max(0, y_1 + \sum_{t'=1}^{t} [x_{t'} + X_{t'} V_{t'} d - d_{t'}])$.

of the dynamic programing formulation in §4, these results confirm that affine decision rules can be worst-case optimal for an inventory model with a hyperrectangle uncertainty set, despite not being Bellman optimal; this is possible only for the robust multistage model and not for the stochastic multistage counterpart, where piecewise affine decision rules are required (see Bertsimas et al. [33] for a discussion).

5.3. Piecewise Affine Decision Rules

It is to be expected that in the general case, applying affine decision rules will generate suboptimal policies compared to fully adjustable ones (see, e.g., Bertsimas and Goyal [23], Kuhn et al. [59] for discussions about the degree of suboptimality of affine decision rules in two-stage linear problems). Thus, one might be tempted to investigate whether tighter conservative approximations can be obtained by employing more sophisticated (yet tractable) decision rules, in particular nonlinear ones. Below, we show how this can be achieved for piecewise affine decision rules by employing affine decision rules on a lifted version of the uncertainty set.

We will employ a class of piecewise affine decision rules described as

$$x_t(v_t(z);) := \bar{x}_t + \bar{X}_t v_t(z) + \sum_{k=1}^{\nu} \big[\theta_{tk}^{+} \max(0; v_{tk}(z)) + \theta_{tk}^{-} \max(0; -v_{tk}(z))\big],$$

where $v_{tk}(z)$ captures the kth observation in z at time t. When the observation mapping is linear, we can assume, without loss of generality, that z is progressively revealed so that each $v_{tk}(z) = z_{i(t,k)}$ for some mapping $i\colon \mathbb{N}\times\mathbb{N}\to\mathbb{N}$. Under this assumption, a piecewise affine decision rule can be expressed as

$$x_t\big(v_t(z); \bar{x}_t, \bar{X}_t^{+}, \bar{X}_t^{-}\big) := \bar{x}_t + \bar{X}_t^{+} V_t z^{+} + \bar{X}_t^{-} V_t z^{-},$$

where $z_i^{+} := \max(0; z_i)$ and $z_i^{-} := \max(0; -z_i)$ and where we omit to include the adjustment $X_t V_t z$ since it can be replicated as $X_t V_t z^{+} - X_t V_t z^{-}$. In this formulation, the decision rule is actually affine with respect to the vector $[z^\top\, z^{+\top}\, z^{-\top}]^\top$. Hence, optimizing such piecewise affine decision rules is equivalent to applying affine decision rules to the lifted uncertainty set

$$\mathcal{Z}' := \big\{(z, z^{+}, z^{-}) \in \mathbb{R}^{3m} \mid z \in \mathcal{Z}, z_i^{+} = \max(0; z_i), z_i^{-} = \max(0; -z_i), \forall\, i = 1, \ldots, m\big\}.$$

Specifically, we are interested in solving the lifted AARC model:

$$\text{(LAARC)} \quad \begin{aligned} \underset{\{x_t\}_{t=1}^{T}, \{X_t^{+}, X_t^{-}\}_{t=2}^{T}}{\text{minimize}} \quad \sup_{(z, z^{+}, z^{-}) \in \mathcal{Z}'} &\Big\{ c_1(z)^\top x_1 \\ &+ \sum_{t=2}^{T} c_t(z)^\top (x_t + X_t^{+} V_t z^{+} + X_t^{-} V_t z^{-}) + d(z) \Big\} \\ \text{s.t.} \quad a_{j1}(z)^\top x_1 + \sum_{t=2}^{T} &a_{jt}(z)^\top (x_t + X_t^{+} V_t z^{+} + X_t^{-} V_t z^{-}), \\ &\le b_j(z), \quad \begin{cases} \forall\, (z, z^{+}, z^{-}) \in \mathcal{Z}', \\ \forall\, j = 1, \ldots, J. \end{cases} \end{aligned}$$

The difficulty is that $\mathcal{Z}'$ is not a convex polyhedron so that even with fixed recourse, an infinite number of constraints such as

$$a_{j1}(z)^\top x_1 + \sum_{t=2}^{T} a_{jt}^\top (x_t + X_t^{+} V_t z^{+} + X_t^{-} V_t z^{-}) \le b_j(z), \quad \forall\, (z, z^{+}, z^{-}) \in \mathcal{Z}'$$

cannot be reformulated in a tractable manner as a single constraint through a direct application of duality theory for linear programs.

One promising direction to design a tractable solution comes from the following proposition.

Proposition 3. *Given that Assumption 1 is satisfied, the lifted AARC model is equivalent to*

$$\begin{aligned}
&\underset{\{x_t\}_{t=1}^T,\{X_t^+,X_t^-\}_{t=2}^T}{\text{minimize}} \quad \sup_{(z,z^+,z^-)\in\mathcal{Z}''} \left\{ c_1(z)^\top x_1 + \sum_{t=2}^T c_t^\top (x_t + X_t^+ V_t z^+ + X_t^- V_t z^-) + d(z) \right\} \\
&\qquad\qquad \text{s.t.} \quad a_{j1}(z)^\top x_1 + \sum_{t=2}^T a_{jt}^\top (x_t + X_t^+ V_t z^+ + X_t^- V_t z^-) \le b_j(z), \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \begin{cases} \forall (z,z^+,z^-)\in\mathcal{Z}'', \\ \forall j = 1,\ldots,J, \end{cases}
\end{aligned} \tag{14}$$

where $\mathcal{Z}'' := \mathrm{ConvexHull}(\mathcal{Z}')$*. Therefore, if the convex hull of* $\mathcal{Z}'$ *can be described with a finite number of linear constraints, then the lifted AARC model can be solved efficiently.*

Proof. This proof simply relies on the fact that since $\mathcal{Z}'' \supseteq \mathcal{Z}'$, any feasible solution of problem (14) is necessarily feasible for the LAARC model. Alternatively, since the functions involved in each constraint are linear in (z, z^+, z^-), if we take a feasible solution to LAARC and verify feasibility in problem (14), a worst-case realization for each constraint necessarily occurs at one of the vertices of $\mathcal{Z}''$, which, by construction, were members of $\mathcal{Z}'$. This indicates that any feasible solution of LAARC is also feasible in problem (14). Hence, the feasible sets of both problems are equivalent. Furthermore, a similar argument, based on the linearity of functions involved, can be used to establish that both objective functions are equivalent. We can thus conclude that the set of optimal solutions and the optimal value of both problems are equivalent. □

This result indicates that the decision maker might be able to obtain a tractable reformulation of the LAARC model if he can identify a good representation for the convex hull of $\mathcal{Z}'$. In this regard, the following proposition might prove useful.

Proposition 4. *Let* $\mathcal{Z} \subseteq [-B, B]^m$ *for some* $B > 0$*. Then, the uncertainty set* $\mathcal{Z}'$ *can be represented as*

$$\mathcal{Z}' = \left\{ (z,z^+,z^-) \in \mathbb{R}^{3m} \;\middle|\; \exists u^+ \in \{0,1\}^m,\, u^- \in \{0,1\}^m, \quad \begin{array}{l} z \in \mathcal{Z} \\ z = z^+ - z^- \\ 0 \le z^+ \le B u^+ \\ 0 \le z^- \le B u^- \\ u^+ + u^- = 1 \end{array} \right\}.$$

Proof. Given any $z \in \mathcal{Z}$, since for any $i = 1, \ldots, m$, $u_i^+ + u_i^- = 1$, and the u's are binary, we necessarily have that either $z_i^+ > 0$ or $z_i^- > 0$. Hence, if $z_i > 0$, then it is necessary that $z_i^+ = z_i$ and $z_i^- = 0$, whereas if $z_i < 0$, it is necessary that $z_i^+ = 0$ and $z_i^- = -z_i$. Finally, if $z_i = 0$, then the only option is for $z_i^+ = z_i^- = 0$. This is exactly the behavior described by $\mathcal{Z}'$. □

This proposition is interesting for two reasons. First, by relaxing the binary constraints on u^+ and u^-, we instantly obtain a tractable outer approximation of ConvexHull($\mathcal{Z}'$). In particular, this idea will be especially effective with the budgeted uncertainty set, i.e., the uncertainty set parameterized by a budget of uncertainty, first introduced in Bertsimas and Sim [27]. Second, this representation of $\mathcal{Z}'$ provides us with a way of performing the worst-case analysis of a fixed piecewise affine decision rule.

Corollary 1. *Given a set of piecewise affine decision rules* $\{x_t\}_{t=1}^T$ *and* $\{X_t^+, X_t^-\}_{t=2}^T$*, let* $\mathcal{Z}$ *be a bounded polyhedron such that* $\mathcal{Z} \subseteq [-B, B]^m$*. Then, one can verify the feasibility with respect to any* j*th constraint of the LAARC model,*

$$a_{j1}(z)^\top x_1 + \sum_{t=2}^T a_{jt}^\top \big(x_t + X_t^+ V_t z^+ + X_t^- V_t z^-\big) \le b_j(z), \quad \forall (z, z^+, z^-) \in \mathcal{Z}'',$$

where $\mathcal{Z}'' := \mathrm{ConvexHull}(\mathcal{Z}')$*, by solving the following mixed integer linear program:*

$$\begin{aligned} \underset{z,\, z^+,\, z^-,\, u^+,\, u^-}{\text{maximize}} \quad & a_{j1}(z)^\top x_1 + \sum_{t=2}^T a_{jt}^\top (x_t + X_t^+ V_t z^+ + X_t^- V_t z^-) - b_j(z) \\ \text{s.t.} \quad & z \in \mathcal{Z}, \\ & z = z^+ - z^-, \\ & 0 \le z^+ \le B u^+, \\ & 0 \le z^- \le B u^-, \\ & u^+ + u^- = 1, \\ & u^+ \in \{0,1\}^m,\ u^- \in \{0,1\}^m, \end{aligned}$$

to obtain $(z^*, {z^+}^*, {z^-}^*)$ *and by verifying that the optimal value is lower than or equal to zero. In the situation that the piecewise affine decision rule is infeasible, then the constraint*

$$a_{j1}(z^*)^\top x_1 + \sum_{t=2}^T a_{jt}^\top \big(x_t + X_t^+ V_t {z^+}^* + X_t^- V_t {z^-}^*\big) \le b_j(z^*)$$

separates the current piecewise affine decision rule from the set of such decision rules that are feasible with respect to the j*th constraint of the LAARC model.*

We now present the tractable representation of $\mathrm{ConvexHull}(\mathcal{Z}')$ presented in Ben-Tal et al. [13] for the case where $\mathcal{Z}$ is the budgeted uncertainty set.

Proposition 5. *Let* $\mathcal{Z}$ *be the budgeted uncertainty set. Then the uncertainty set* Convex-Hull($\mathcal{Z}'$) *can be represented using the following tractable form:*

$$\mathrm{ConvexHull}(\mathcal{Z}') = \left\{ (z, z^+, z^-) \in \mathbb{R}^{3m} \;\middle|\; \begin{array}{l} z^+ + z^- \le 1 \\ \sum_{i=1}^m z_i^+ + z_i^- \le \Gamma \\ z = z^+ - z^- \\ 0 \le z^+ \\ 0 \le z^- \end{array} \right\}.$$

Although we refer the reader to Ben-Tal et al. [13, Chapter 14.3.2] for a complete proof of a more general result that involves *absolutely symmetric convex functions*, it is worth noting that the above set is obtained by applying fractional relaxation of a slightly modified version of the set proposed in Proposition 4. Specifically, the representation proposed in Proposition 5 can be seen as equivalent to the set

$$\left\{ (z,z^+,z^-)\in\mathbb{R}^{3m} \;\middle|\; \exists u^+\in[0,1]^m, u^-\in[0,1]^m, \quad \begin{array}{l} \|z\|_\infty \le 1 \\ \|z\|_1 \le \Gamma \\ z = z^+ - z^- \\ z^+ + z^- \le 1 \\ \sum_{i=1}^m z_i^+ + z_i^- \le \Gamma \\ 0\le z^+ \le u^+ \\ 0 \le z^- \le u^- \\ u^+ + u^- = 1 \end{array} \right\},$$

which is obtained by adding two types of valid inequalities, $z^+ + z^- \le 1$ and $\sum_{i=1}^m (z_i^+ + z_i^-) \le \Gamma$, to the representation described in Proposition 4 before relaxing the binary constraints on u^+ and u^-. This process is well known to produce a tighter polyhedral outer approximation of feasible sets that involve integer variables. Whereas, in general, there is no guarantee that adding valid inequalities to a representation involving binary variables will lead to an exact representation of ConvexHull($\mathcal{Z}'$), it can be used to identify tighter conservative approximations of any instance of the LAARC problem (see Ardestani-Jaafari and Delage [2] for a related discussion).

Example 3. Consider our inventory management problem in which we wish to optimize a decision rule that is piecewise affine with respect to the positive and negative deviations of each demand parameter. In particular, at time $t=2$, we would like to design a decision rule that is parameterized as

$$x_2(d_1) := \bar{x}_2 + \bar{x}_2^+ \max\big(0; (d_1 - \bar{d}_1)/\hat{d}_1\big) + \bar{x}_2^- \max\big(0; (\bar{d}_1 - d_1)/\hat{d}_1\big)$$

such that we increase the order by $\bar{x}_2^+$ units per *normalized* unit of demand above the nominal amount and increase it by $\bar{x}_2^-$ units per normalized unit below the nominal amount. This can be done by designing decision rules on the lifted space $(z^+, z^-)\in\mathbb{R}^T\times\mathbb{R}^T$ such that $d_i := \bar{d}_i + \hat{d}_i(z_i^+ - z_i^-)$ for all i. Namely, at time $t=2$, the decision rule becomes

$$x_2(z_1^+, z_1^-) := \bar{x}_2 + \bar{x}_2^+ z_1^+ + \bar{x}_2^- z_1^-.$$

An efficient representation for the convex hull of the lifted uncertainty space in terms of (z^+, z^-) was described in Proposition 5 in the case of the budgeted uncertainty set. This leads to the following lifted AARC:

$$\begin{aligned}
\underset{\substack{x_1, \{x_t, X_t^+, X_t^-\}_{t=2}^T, \\ \{r_t, R_t^+, R_t^-\}_{t=1}^T, \\ \{s_t, S_t^+, S_t^-\}_{t=1}^T}}{\text{minimize}} \quad & \sup_{(z^+,z^-)\in\mathcal{Z}''} \Big\{ c_1 x_1 + \sum_{t=2}^T c_t\big(x_t + X_t^+ z_{[t-1]}^+ + X_t^- z_{[t-1]}^-\big) \\
& \qquad + h_t(r_t + R_t^+ z^+ + R_t^- z^-) + b_t(s_t + S_t^+ z^+ S_t^- z^-) \Big\} \\
\text{s.t.} \quad & r_t + R_t^+ z^+ + R_t^- z^- \ge 0, \quad s_t + S_t^+ z^+ S_t^- z^- \ge 0, \quad \forall (z^+,z^-)\in\mathcal{Z}'', \forall t, \\
& r_t + R_t^+ z^+ + R_t^- z^- \ge y_1 + \sum_{t'=1}^t x_t + X_t^+ z_{[t-1]}^+ + X_t^- z_{[t-1]}^- - d_{t'}(z^+,z^-), \\
& \qquad \forall (z^+,z^-)\in\mathcal{Z}'', \forall t, \\
& s_t + S_t^+ z^+ S_t^- z^- \ge -y_1 + \sum_{t'=1}^t d_{t'}(z^+,z^-) - (x_t + X_t^+ z_{[t-1]}^+ + X_t^- z_{[t-1]}^-), \\
& \qquad \forall (z^+,z^-)\in\mathcal{Z}'', \forall t, \\
& 0 \le x_t + X_t^+ z_{[t-1]}^+ + X_t^- z_{[t-1]}^- \le M_t, \quad \forall (z^+,z^-)\in\mathcal{Z}'', \forall t,
\end{aligned}$$

where $d_t(z^+, z^-) := \bar{d}_j + \hat{d}_j(z_j^+ - z_j^-)$ and

$$\mathcal{Z}'' := \left\{ (z^+, z^-) \in \mathbb{R}^{2m} \,\middle|\, z^+ \geq 0,\, z^- \geq 0,\, z^+ + z^- \leq 1, \sum_i z_i^+ + z_i^- \leq \Gamma \right\}.$$

Remark 3. One can trace back the use of lifted affine decision rules to the work of Chen and Zhang [41], Chen et al. [42], and Goh and Sim [51], which suggests using so-called *segregated linear decision rules*. Several other types of nonlinear decision rules have also been proposed in the literature. For instance, Bertsimas and Caramanis [21] and Bertsimas et al. [31] suggest using piecewise constant decision rules, which can be found by simulation and convex optimization. Similarly, Chatterjee et al. [40] consider arbitrary functional forms of the disturbances, and they show how the coefficients parameterizing the decision rule can be found by solving convex optimization problems for specific types of p-norm constraints on the controls. A similar approach is taken in Skaf and Boyd [70], where the authors also consider arbitrary functional forms for the policies and show how, for a problem with convex state-control constraints and convex costs, such policies can be found by convex optimization, combined with Monte Carlo sampling to enforce constraint satisfaction. Finally, Bertsimas et al. [34] propose using polynomial decision rules, which can be found by solving tractable semidefinite programming problems. The book by Ben-Tal et al. [13, Chapter 14] also contains a thorough review of several other classes of decision rules and a discussion of cases when sophisticated decision rules can actually improve performance over the affine ones.

6. Should One Worry About Time Consistency?

In the context of multistage decision making under uncertainty, time (or dynamic) consistency is an axiomatic property that requires a decision maker's stated preferences over future courses of action to remain consistent with the actual preferred actions when planned-for contingencies arise. To understand this concept, consider a hypothetical investor who decides—according to his or her own risk preferences—that it would be optimal to invest \$100 in company A's stock tomorrow if its share price were to rise above \$3. If tomorrow the share price does rise above \$3, and the investor decides—upon reconsidering his or her preferred actions—that \$100 is no longer an optimal investment amount, the investor's preferences would be *inconsistent*. For more background information on time consistency, the reader can refer to Shapiro et al. [69, Chapter 6].

Time consistency is intrinsically related to the formulation of the multistage decision problem, particularly to the way in which the decision maker's current-day preferences relate to future preferences. In broad terms, a decision maker who always formulates and solves a multistage problem via dynamic programming, thus ensuring that the actions satisfy the Bellman principle of optimality, will always behave in a consistent fashion. However, in the context of robust optimization, a subtler issue that may affect the consistency in preferences has to do with the way in which uncertainty sets are updated dynamically and as the time comes for future actions to be implemented. In particular, one needs to understand how a robust multistage decision model implicitly assumes a specific updating rule that must be followed in order to avoid giving rise to dynamic inconsistencies in the decision maker's preferences (and actions). This issue is illustrated below.

Consider a two-stage inventory problem with an initial ordering cost of \$1 per unit and a larger second-stage ordering cost of \$4 per unit. We also assume that there are no holding costs and that backlog costs are only charged in the final stage at a cost of \$10 per unit. In each stage, demand is expected to be 1 unit with a possible deviation of up to 1 unit.

To control the conservativeness of the solution, it is decided that total upward deviation should be bounded by 1. This gives rise to the following multistage ARC:

$$\begin{aligned}\underset{x_1, x_2(\cdot), s(\cdot)}{\text{minimize}} \quad & \sup_{d\in\mathcal{U}}\big\{x_1 + 4x_2(d_1) + 10s(d)\big\} \\ \text{s.t.} \quad & s(d) \geq 0, \quad \forall\, d \in \mathcal{U}, \\ & s(d) \geq d_1 + d_2 - x_1 - x_2(d_1), \quad \forall\, d \in \mathcal{U}, \\ & x_1 \geq 0, \\ & x_2(d_1) \geq 0, \quad \forall\, d \in \mathcal{U}, \end{aligned} \tag{15}$$

where $\mathcal{U} := \{d \in \mathbb{R}^2 \mid (d_1 - 1)^+ + (d_2 - 1)^+ \leq 1\}$. One can easily confirm that an optimal robust policy consists of ordering three units at time $t = 1$ and nothing at time $t = 2$. Under this policy, the worst-case total cost of \$3 occurs for any realization of the pair (d_1, d_2) in $\mathcal{U}$. Intuitively, the policy is optimal since we wish to protect against the pair $(2, 1)$, which would require us to produce three units so as to avoid the large backlog cost, yet there is no reason to delay the purchase since the cost is lower at time $t = 1$ and there is no holding cost.

The idea that we wish to highlight here is that the optimality of the policy we just identified relies entirely on the assumption that the decision maker acts in a *time-consistent* fashion and solves the decision problem following the dynamic programming paradigm highlighted in §4. In particular, once the first-stage decision x_1 is implemented and the demand d_1 is observed, the decision maker updates the uncertainty set for the second stage depending on d_1, so that the optimization problem solved in the second stage is

$$\begin{aligned}\underset{x_2, s(\cdot)}{\text{minimize}} \quad & \sup_{d_2\in\mathcal{U}_2(d_1)}\big\{4x_2 + 10s(d_2)\big\} \\ \text{s.t.} \quad & s(d_2) \geq 0, \quad \forall\, d_2 \in \mathcal{U}_2(d_1), \\ & s(d_2) \geq d_1 + d_2 - x_1 - x_2, \quad \forall\, d_2 \in \mathcal{U}_2(d_1), \\ & x_2 \geq 0. \end{aligned}$$

Here, $\mathcal{U}_2(d_1) := \{d_2 \in \mathbb{R} \mid (d_1, d_2) \in \mathcal{U}\}$ captures the *slice* of $\mathcal{U}$ when the first-stage demand d_1 was observed. In particular,

$$\begin{aligned}\mathcal{U}_2(0) &:= [0, 2], \\ \mathcal{U}_2(1) &:= [0, 1].\end{aligned}$$

Although this rule for updating the uncertainty set is implicitly assumed in a robust multistage model, it may be more or less applicable from a modeling standpoint, depending on the particular application. We provide two examples in which this updating rule comes across as more or less realistic, and we comment on the potential pitfalls when this rule is violated.

- *Time-consistent situation*: Consider the owner of a coffee stand that is allowed to operate for one morning in the lobby of a hotel. The owner plans on selling coffee during the 7 A.M–11 A.M period and possibly replenishing with fresh coffee at 9 A.M. Based on the hotel's occupancy level and his prior experience, he estimates that about 100 cups of coffee (one unit) might be purchased during the 7 A.M–9 A.M interval, and about 100 cups (one unit) might be purchased during the 9 A.M–11 A.M interval. He also considers it extremely unlikely that more than 300 cups of coffee (three units) would be needed in a single morning (e.g., since that happens to be the maximum number of guests at the hotel, and very few individuals buy two cups of coffee in the morning). This circumstance motivates an uncertainty set of the form in (15), and it suggests that it may be reasonable to not order more coffee even after having sold 200 cups (two units) during the 7 A.M–9 A.M interval.

• *Time-inconsistent situation*: Consider the same coffee stand owner that instead plans to move his stand at 9 A.M to a different nearby hotel that has similar occupancy. In this context, it might still be reasonable to initially assume when opening the stand at 7 A.M that no more than 300 cups of coffee (three units) would be needed the whole morning (possibly with the argument that if demands at the two hotels are independent, it would be unlikely that they are both significantly above their expected amounts), it seems unreasonable to assume that having sold more than 200 cups (two units) in the first hotel by 9 A.M implies without any doubt that no more than 100 cups (one unit) are needed for customers at the second hotel. Instead, the owner may be tempted to believe that there might still be enough customers to sell up to 200 cups (two units) in the second hotel and thus might make an order that departs from what his original optimal policy suggested.

This second scenario would give rise to time inconsistency in preferences, in the sense that at time $t=2$, the decision would actually be taken with respect to a robust optimization model using an uncertainty set that is inconsistent with $\mathcal{U}_2(d_1)$. In our example, this would be $\tilde{\mathcal{U}}_2(d_1) := [0,2]$. Looking back at the robust first-stage decision $x_1 = 3$, this would mean that if $d_1 = 2$, then the second ordering quantity would be decided so as to minimize $\max_{d_2\in[0,2]}[3+4x_2+10(3+x_2-2-d_2)]$, which would imply ordering one additional unit to avoid the excessive backlog cost. Under this scenario, the total cost ends up being $3+4\cdot 1 = 7$. Yet, it is clear that if the extra unit had been purchased in the initial stage, then the total cost would have been 4 for this scenario, and always lower than this amount as long as the policy implemented at the second stage was $x_2(d_1) := (d_1+2-4)^+$.

One can actually show that the decisions $x_1 = 4$ and $x_2(d_1) := (d_1+2-4)^+$ are optimal according to the bilevel problem:

$$\begin{aligned}
\underset{x_1,\,x_2(\cdot),s(\cdot)}{\text{minimize}} \quad & \sup_{d\in\mathcal{U}}\big\{x_1+4x_2(d_1)+10s(d)\big\}\\
\text{s.t.} \quad & s(d)\ge 0, \quad \forall\, d\in\mathcal{U},\\
& s(d)\ge d_1+d_2-x_1-x_2(d_1), \quad \forall\, d\in\mathcal{U},\\
& x_1\ge 0,\\
& x_2(d_1)\in \underset{x_2'\ge 0}{\text{argmin}} \max_{d_2'\in\tilde{\mathcal{U}}_2(d_1)} \big\{c_2x_2'+b(d_1+d_2'-x_1-x_2')^+\big\}, \quad \forall\, d_1\in\mathcal{U}_1,
\end{aligned}$$

where $\mathcal{U}_1$ is the projection of the set $\mathcal{U}$ over the first component, $\mathcal{U}_1 := \{d\in\mathbb{R} \mid \exists d_2, (d,d_2)\in\mathcal{U}\}$. This model resolves time inconsistency by explicitly acknowledging that the second-stage decision must be consistent with respect to the uncertainty set that will be employed in the second stage, once d_1 is observed. Such bilevel optimization problems reduce to our multistage ARC when $\tilde{\mathcal{U}}_2(d_1) = \mathcal{U}_2(d_1) := \{d_2\in\mathbb{R} \mid (d_1,d_2)\in\mathcal{U}\}$.

This discussion highlights the importance of fully specifying the way in which uncertainty sets are updated depending on intermediate observations; such specification critically governs the objectives of the robust decision problem at future time stages. When the updating is done by "slicing" the uncertainty set, as in §4—which is analogous to the typical conditioning in stochastic optimization—the decision maker will always act in a time-consistent fashion and will never desire changing the dynamic policies once contingencies that have been planned for actually arise.

7. Conclusions and Future Directions

Despite the advances in robust multistage optimization, several important directions remain underexplored. One such example concerns the development of exact algorithms for identifying the optimal first-stage decision in a multistage ARC. Although the inherent NP-hardness of the problem might be a little discouraging, some valuable procedures based on Benders' decomposition and column-constraint generation methods have been proposed for two-stage problems and applied quite successfully to a number of applications (e.g., in Bertsimas et al. [36], Thiele et al. [73], Zhao and Zeng [79]). However, it is still unclear how such methods can be extended to problems with a longer horizon.

Another important issue concerns the development of results concerning the optimality (or degree of suboptimality) guarantees for particular simple functional forms of policies, such as affine or piecewise affine with few (i.e., polynomially many) pieces.[9] As discussed in §§4 and 5.2, such results would have to leverage the degeneracy inherent in robust multistage decision making, which allows policies that are worst-case optimal without necessarily being Bellman optimal. New theoretical tools would have to be developed that can exploit this degeneracy without making use of dynamic programming to solve the underlying model (as the latter paradigm would always enforce the strict requirement of Bellman optimality).

A third direction of study concerns problems where the assumption of fixed recourse is not satisfied or, alternatively, where some of the adjustable decisions are constrained to be integral. Indeed, in the former case, the AARC problem (12) no longer involves robust biaffine constraints. One might be able to obtain semidefinite programming reformulations by using methods for polynomial optimization problems (as demonstrated in Bertsimas et al. [34]), yet it is still unclear how effective these methods actually are for problems of reasonable size. Note that a similar issue also arises when the decision model involves nonlinear constraints such as

$$g(x_1, x_2, \ldots, x_T) \le 0, \quad \forall\, z \in \mathcal{Z},$$

where $g(\cdot)$ is jointly convex, which gives rise to hard separation problems when employing affine decision rules.

The case of integer adjustable decisions has attracted more attention recently. In particular, such models have been initially discussed by Bertsimas and Caramanis [21], who prove intractability and propose approximations based on sampling. Their work has been extended by Hanasusanto et al. [55], who consider binary adaptable policies with K contingency plans, as well as by Bertsimas and Gheorghiou [22], who adopt the following specific parameterization of an adjustable binary decision y:

$$y(z) = \begin{cases} 1 & \text{if } \max\{\bar{y}_1^\top z, \bar{y}_2^\top z, \ldots, \bar{y}_P^\top z\} - \max\{\underline{y}_1^\top z,\ \underline{y}_2^\top z, \ldots,\ \underline{y}_P^\top z\} \le 0, \\ 0 & \text{otherwise,} \end{cases}$$

where $\{\bar{y}_i,\ \underline{y}_i\}_{i=1}^P$ are chosen by the decision maker. Despite this progress, the theory of adjustable decisions with integrality constraints lacks the same solid foundations as its counterpart for continuous decisions, and results are lacking concerning the optimality of simple parameterizations for important applications.

A final example of future research directions might also involve the design of uncertainty sets for multistage problems. Several techniques have been proposed for constructing uncertainty sets in single-stage models, using statistical hypothesis testing (see, e.g., Ben-Tal et al. [17], Bertsimas et al. [32]). Although some results exist in the literature for characterizing temporal correlations between different uncertain parameters (see, e.g., Lorca and Sun [61], Miao et al. [63]), still relatively little is known on how to effectively control the level of conservativeness of such dynamic uncertainty sets in a wide variety of practical settings.

Acknowledgments

The authors gratefully acknowledge Aurélie Thiele and Daniel Kuhn for the constructive feedback they provided in the final stage of development of this tutorial. E. Delage gratefully acknowledges the support from the Canada Research Chairs program.

[9] The latter class is known to be optimal for several types of models, but it typically involves an exponential number of pieces, which makes it impractical.

References

[1] E. Adida and G. Perakis. A robust optimization approach to dynamic pricing and inventory control with no backorders. *Mathematical Programming* 107(1–2):97–129, 2006.

[2] A. Ardestani-Jaafari and E. Delage. Robust optimization of sums of piecewise linear functions with application to inventory problems. Cahiers du GERAD G-2015-19, GERAD–HEC Montréal, Montréal, 2015.

[3] F. Babonneau, J.-P. Vial, and R. Apparigliato. Robust optimization for environmental and energy planning. J. A. Filar and A. Haurie, eds. *Uncertainty and Environmental Decision Making*, International Series in Operations Research and Management Science, Vol. 138. Springer, New York, 79–126, 2010.

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

[5] O. Baron, J. Milner, and H. Naseraldin. Facility location: A robust optimization approach. *Production and Operations Management* 20(5):772–785, 2011.

[6] A. Bemporad, F. Borrelli, and M. Morari. Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on Automatic Control* 48(9):1600–1606, 2003.

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

[8] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters* 25(1):1–13, 1999.

[9] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming, Series A* 88(3):411–424, 2000.

[10] A. Ben-Tal and A. Nemirovski. Robust optimization—Methodology and applications. *Mathematical Programming* 92(3):453–480, 2002.

[11] A. Ben-Tal, S. Boyd, and A. Nemirovski. Control of uncertainty-affected discrete time linear systems via convex programming. Working paper, Massachusetts Institute of Technology, Cambridge, 2005.

[12] A. Ben-Tal, S. Boyd, and A. Nemirovski. Extending scope of robust optimization: Comprehensive robust counterparts of uncertain problems. *Mathematical Programming* 107(1):63–89, 2006.

[13] A. Ben-Tal, L. El-Ghaoui, and A. Nemirovski. *Robust Optimization*, Princeton Series in Applied Mathematics. Princeton University Press, Princeton, NJ, 2009.

[14] A. Ben-Tal, A. Nemirovski, and C. Roos. Robust solutions of uncertain quadratic and conic-quadratic problems. *SIAM Journal on Optimization* 13(2):535–560, 2002.

[15] 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.

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

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

[18] D. Bertsekas. Control of uncertain systems with a set-membership description of the uncertainty. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, 1971.

[19] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, MA, 2001.

[20] D. P. Bertsekas and I. B. Rhodes. On the minimax reachability of target tubes. *Automatica* 7(2):233–247, 1971.

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

[22] D. Bertsimas and A. Gheorghiou. Design of near optimal decision rules in multistage adaptive mixed-integer optimization. *Operations Research* 63(3):610–627, 2015.

[23] D. Bertsimas and V. Goyal. On the power and limitations of affine policies in two-stage adaptive optimization. *Mathematical Programming* 134(2):1–41, 2010.

[24] D. Bertsimas and V. Goyal. On the power of robust solutions in two-stage stochastic and adaptive optimization problems. *Mathematics of Operations Research* 35(2):284–305, 2010.

[25] D. Bertsimas and D. Pachamanova. Robust multiperiod portfolio management in the presence of transaction costs. *Computers and Operations Research* 35(1):3–17, 2008.

[26] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming* 98(1–3):49–71, 2003.

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

[28] D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations Research* 54(1):150–168, 2006.

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

[30] D. Bertsimas, V. Goyal, and B. Lu. A tight characterization of the performance of static solutions in two-stage adjustable robust linear optimization. *Mathematical Programming* 150(2):281–319, 2015.

[31] D. Bertsimas, V. Goyal, and X. A. Sun. A geometric characterization of the power of finite adaptability in multistage stochastic and adaptive optimization. *Mathematics of Operations Research* 36(1):24–54, 2011.

[32] D. Bertsimas, V. Gupta, and N. Kalus. Data-driven robust optimization. Working paper, Massachusetts Institute of Technology, Cambridge, 2013.

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

[34] D. Bertsimas, D. A. Iancu, and P. Parrilo. A hierarchy of near-optimal policies for multistage adaptive optimization. *IEEE Transactions on Automatic Control* 56(12):2809–2824, 2011.

[35] D. Bertsimas, D. Pachamanova, and M. Sim. Robust linear optimization under general norms. *Operations Research Letters* 32(6):510–516, 2004.

[36] 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, 2013.

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

[38] G. Calafiore and M. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control* 51(5):742–753, 2006.

[39] S. Ceria and R. Stubbs. Incorporating estimation errors into portfolio selection: Robust portfolio construction. *Journal of Asset Management* 7(2):109–127, 2006.

[40] D. Chatterjee, P. Hokayem, and J. Lygeros. Stochastic model predictive control with bounded control inputs: A vector space approach. *IEEE Transactions on Automatic Control* 56(11):2704–2710, 2011.

[41] X. Chen and Y. Zhang. Uncertain linear programs: Extended affinely adjustable robust counterparts. *Operations Research* 57(6):1469–1482, 2009.

[42] 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.

[43] F. J. C. T. de Ruiter, A. Ben-Tal, R. C. M. Brekelmans, and D. Den Hertog. Adjustable robust optimization with decision rules based on inexact revealed data. CentER Discussion Paper 2014-003, CentER, Tilburg University, Tilburg, The Netherlands, 2014.

[44] J. Doyle, K. Glover, P. Khargonekar, and B. Francis. State-space solutions to standard H_2 and HA_∞ control problems. *IEEE Transactions on Automatic Control* 34(8):831–847, 1989.

[45] G. E. Dullerud and F. Paganini. *A Course in Robust Control Theory: A Convex Approach.* Springer, New York, 2005.

[46] L. El-Ghaoui and H. Lebret. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications* 18(4):1035–1064, 1997.

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

[48] J. E. Falk. Exact solutions of inexact linear programs. *Operations Research* 24(4):783–787, 1976.

[49] M. K. H. Fan, A. Tits, and J. Doyle. Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics. *IEEE Transactions on Automatic Control* 36(1):25–38, 1991.

[50] V. Gabrel, C. Murat, and A. Thiele. Recent advances in robust optimization and robustness: An overview. *European Journal of Operational Research* 235(3):471-483, 2014.

[51] J. Goh and M. Sim. Distributionally robust optimization and its tractable approximations. *Operations Research* 58(4, Part 1):902–917, 2010.

[52] D. Goldfarb and G. Iyengar. Robust portfolio selection problems. *Mathematics of Operations Research* 28(1):1–38, 2003.

[53] 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, 2013.

[54] V. Goyal and B. Lu. On the adaptivity gap in two-stage robust linear optimization under uncertain constraints. Working paper, Columbia University, New York, 2015.

[55] G. Hanasusanto, D. Kuhn, and W. Wiesemann. K-Adaptability in two-stage robust binary programming. *Operations Research* 63(4):877–891, 2015.

[56] D. A. Iancu, M. Sharma, and M. Sviridenko. Supermodularity and affine policies in dynamic robust optimization. *Operations Research* 61(4):941–956, 2013.

[57] G. Iyengar. Robust dynamic programming. *Mathematics of Operations Research* 30(2):257–280, 2005.

[58] H. Kasugai and T Kasegai. Characteristics of dynamic maximin ordering policy. *Journal of the Operations Research Society of Japan* 3(1):11–26, 1960.

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

[60] X. Lin, S. L. Janak, C. A. Floudas. A new robust optimization approach for scheduling under uncertainty: I. bounded uncertainty. *Computers and Chemical Engineering* 28(6–7):1069–1085, 2004.

[61] A. Lorca and X. A. Sun. Adaptive robust optimization with dynamic uncertainty sets for multi-period economic dispatch under significant wind. *IEEE Transactions on Power Systems* 30(4):1702–1713, 2015.

[62] M. Mani, A. Sing, and M. Orshansky. Joint design-time and post-silicon minimization of parametric yield loss using adjustable robust optimization. *Proceedings of the 2006 IEEE/ACM International Conference on Computer-Aided Design*. ACM, New York, 19–26, 2006.

[63] W. Miao, H. Yin, D. Du, and J. Han. Robust optimization model for a class of uncertain linear programs. B. Chen, M. Paterson, and G. Zhang, eds. *Combinatorics, Algorithms, Probabilistic and Experimental Methodologies*, Lecture Notes in Computer Science, Vol. 4614. Springer, Berlin, 384–395, 2007.

[64] S. Mittal, A. S. Schulz, and S. Stiller. Robust appointment scheduling. K. Jansen, J. Rolim, N. Devanur, and C. Moore, eds. *Proceedings of the 17th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems/18th International Workshop on Randomization and Computation*, Leibniz International Proceedings in Informatics, Vol. 28. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Wadern, Germany, 356–370, 2014.

[65] A. Nilim and L. El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research* 53(5):780–798, 2005.

[66] G. Perakis and G. Roels. Robust controls for network revenue management. *Manufacturing & Service Operations Management* 12(1):56–76, 2010.

[67] M. Ç. Pınar and R. H. Tütüncü. Robust profit opportunities in risky financial portfolios. *Operations Research Letters* 33(4):331–340, 2005.

[68] W. B. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality.* John Wiley & Sons, Hoboken, 2007.

[69] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory.* SIAM, Philadelphia, 2009.

[70] J. Skaf and S. Boyd. Nonlinear Q-design for convex stochastic control. *IEEE Transactions on Automatic Control* 54(10):2426–2430, 2009.

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

[72] A. Thiele. Multi-product pricing via robust optimisation. *Journal of Revenue and Pricing Management* 8(1):67–80, 2009.

[73] A. Thiele, T. Terry, and M. Epelman. Robust linear optimization with recourse. Technical report, University of Michigan, Ann Arbor, 2010.

[74] R. Tütüncü and M. Koenig. Robust asset allocation. *Annals of Operations Research* 132(1–4): 157–187, 2004.

[75] W. Wiesemann, D. Kuhn, and B. Rustem. Robust resource allocations in temporal networks. *Mathematical Programming* 135(1–2):437–471, 2012.

[76] H. Witsenhausen. Minimax control of uncertain systems. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 1966.

[77] H. Witsenhausen. Sets of possible states of linear systems given perturbed observations. *IEEE Transactions on Automatic Control* 13(5):556–558, 1968.

[78] D. S. Yamashita, V. A. Armentano, and M. Laguna. Robust optimization models for project scheduling with resource availability cost. *Journal of Scheduling* 10(1):67–76, 2007.

[79] L. Zhao and B. Zeng. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters* 41(5):457–461, 2013.

[80] C. Zhao, J. Wang, J. P. Watson, and Y.Guan. Multi-stage robust unit commitment considering wind and demand response uncertainties. *IEEE Transactions on Power Systems* 28(3): 2708–2717, 2013.

[81] K. Zhou and J. C. Doyle. *Essentials of Robust Control.* Prentice-Hall, Upper Saddle River, NJ, 1998.