Design of Near Optimal Decision Rules in Multistage Adaptive Mixed-Integer Optimization

Design of Near Optimal Decision Rules in Multistage Adaptive Mixed-Integer Optimization

Dimitris Bertsimas * Angelos Georghiou †

September 2, 2014

Abstract

In recent years, decision rules have been established as the preferred solution method for addressing computationally demanding, multistage adaptive optimization problems. Despite their success, existing decision rules (a) are typically constrained by their a priori design and (b) do not incorporate in their modeling adaptive binary decisions. To address these problems, we first derive the structure for optimal decision rules involving continuous and binary variables as piecewise linear and piecewise constant functions, respectively. We then propose a methodology for the optimal design of such decision rules that have finite number of pieces and solve the problem using mixed-integer optimization. We demonstrate the effectiveness of the proposed methods in the context of two multistage inventory control problems. We provide global lower bounds and show that our approach is (i) practically tractable and (ii) provides high quality solutions that outperform alternative methods.

Keywords. Adaptive Optimization, Decision Rules, Mixed-integer optimization.

1 Introduction

Robust optimization has emerged as the leading modeling paradigm for optimization problems under uncertainty. Its success stems from its ability to immunize problems against perturbations in model parameters, while preserving computational tractability, see Ben-Tal *et al.* [8] and Bertsimas and Sim [14]. The classical robust optimization framework involves only *here-and-now* decisions, i.e., decisions that must be selected before any of the uncertain parameters are observed. In recent years, adaptive robust optimization problems have attracted considerable interest. Such problems incorporate additional

*Sloan School of Management and Operations Research Center, Massachusetts Institute of Technology, USA, E-mail: dbertsim@mit.edu.

†Process Systems Engineering Laboratory, Massachusetts Institute of Technology, USA, E-mail: angelosg@mit.edu.

adaptive, *wait-and-see* decisions, which can be chosen after the parameter realizations are revealed. As such, these *decision rules* are modeled as functions of the observed uncertain parameter realizations.

Adaptive optimization problems have been shown to be in general computationally intractable from a theoretical point of view, see Shapiro and Nemirovski [33]. A drastic but very effective simplification in gaining computational tractability proposed in Ben-Tal *et al.* [6], is to restrict the space of admissible adaptive continuous decisions to those that admit a linear structure with respect to the random parameters. Specifically, given uncertainty $\boldsymbol{\xi} \in \Xi$, the linear decision rule for an adaptive continuous variable $x(\cdot)$ is given by $x(\boldsymbol{\xi}) = \boldsymbol{x}^\top \boldsymbol{\xi}$, $\boldsymbol{\xi} \in \Xi \subseteq \mathbb{R}^k$. This linear decision rule has attracted considerable interest in recent years, since its simple structure enables scalability to multistage models. Even though linear decision rules are known to be optimal for the linear quadratic regulator problem, see Anderson and Moore [1], some one-dimensional robust control problems, see Bertsimas *et al.* [12], and some vehicle routing problems, see Gounaris *et al.* [26], decision rules generically sacrifice a significant amount of optimality in return for scalability. Indeed, Bertsimas and Goyal [11] have shown that the worst-case cost associated with linear decision rules can be $O(\sqrt{k})$ suboptimal when applied to two-stage robust optimization problems with k uncertain parameters. The aforementioned work also shows that the optimal decision rule for adaptive optimization problems is piecewise linear. This motivates the use of non-linear decision rules.

There is a plethora of non-linear decision rules appearing in the literature. Under a non-linear decision rule, the adaptive decision $x(\cdot)$ is given by $x(\boldsymbol{\xi}) = \boldsymbol{x}^\top L(\boldsymbol{\xi})$, where $L : \mathbb{R}^k \to \mathbb{R}^{k'}$, $k \leq k'$, is a non-linear operator, mapping the uncertain vector $\boldsymbol{\xi}$ to $L(\boldsymbol{\xi})$ in a higher dimensional space $L(\Xi)$. Using the re-parameterization $\boldsymbol{\xi}' = L(\boldsymbol{\xi})$, the adaptive decision rule $x(\boldsymbol{\xi}) = \boldsymbol{x}^\top L(\boldsymbol{\xi})$ can be expressed as a linear decision rule $x(\boldsymbol{\xi}') = \boldsymbol{x}^\top \boldsymbol{\xi}'$ for all $\boldsymbol{\xi}' \in L(\Xi)$. The choice of the non-linear operator $L(\cdot)$ defines the non-linear structure of the decision rule. This formulation was first introduced by Chen and Zhang [21] creating simple piecewise linear decision rule structures. The framework was later extended to cover general non-linear decision rules by Goh and Sim [24] and Georghiou *et al.* [23] in the realm of stochastic programming. Bertsimas *et al.* [13] also proposed the use of polynomial decision rules for adaptive robust optimization problems, an idea that was later refined by Bampou and Kuhn [3] in the context of stochastic programming. The strict advantage of the aforementioned approximations is that the resulting decision rule problems can be cast as conic optimization problems, whose size grows only polynomially with the input parameters. Nevertheless, there are two main disadvantages associated with these non-linear decision rules: (*i*) The non-linear structure of the decision rule $L(\cdot)$ needs to be decided a priori by the user; (*ii*) An exact reformulation of the decision rule problem into a finite dimensional conic optimization problem can only be done efficiently for certain classes of non-linear decision rules and special structure of the uncertainty set Ξ, see Georghiou *et al.* [23] for more details. This restricts the choices of admissible decision rules, and can lead to the same worst-case performance as with linear decision rules.

In contrast to the large effort dedicated in developing decision rules for continuous decisions, very little effort is devoted in developing approximations for binary adaptive decisions. In most of the literature, all binary decisions are treated as here-and-now decisions leading to conservative estimates for the optimal solution. To the best of our knowledge, the work of Bertsimas and Caramanis in [10] is the only framework that can address the binary decisions in an adaptive robust optimization setting. To simulate the recourse nature of the binary decisions, the authors proposed to partition the uncertainty set into finite subsets, assigning a binary decision for each subset. Although this idea is very effective when the adaptive problem in hand has a small number of time stages, the exponential growth in the number of binary decisions and constraints renders the method unsuitable for multistage problems with large number of stages.

The goal of this paper is to design piecewise linear and piecewise constant decision rules for continuous and binary decisions, respectively, in the context of multistage adaptive optimization problems. The proposed decision rules overcome the shortfalls of existing approaches, while maintaining in part the desirable tractability properties of existing methods. The main contributions of this paper may be summarized as follows.

1. We derive the structure for the optimal decision rules involving both continuous and binary variables as piecewise linear and piecewise constant functions, respectively.

2. We propose a methodology for the optimal design of such piecewise linear continuous and piecewise linear binary decision rules that have fixed number of pieces. To the best of our knowledge, this is the first decision rule that can address binary adaptive decisions. In contrast with the non-linear continuous decision rules in the literature, the structure of the proposed decision rules is not decided a priori, but rather is decided endogenously through the solution of a sequence of mixed-integer optimization problems.

3. We demonstrate the effectiveness of the proposed methods in the context of two multistage inventory control problems. We provide global lower bounds using the approach in Hadjiyiannis *et al.* [27] and show that our approach is (*i*) practically tractable and (*ii*) provides high quality solutions that outperform alternative methods.

The rest of this paper is organized as follows. In Section 2, we outline our approach for continuous and binary adaptive decision in the context of one-stage adaptive optimization problems involving only recourse decisions. Section 3 discusses the lower bounding technique proposed in Hadjiyiannis *et al.* [27].

In Section 4, we extent the proposed approach to multistage adaptive optimization problems and in Section 5, we present our computational results. Finally, Section 6 provides some concluding remarks.

Notation We denote scalar quantities by lowercase, non-bold face symbols and vector quantities by lowercase, boldface symbols, e.g., $x \in \mathbb{R}$ and $\boldsymbol{x} \in \mathbb{R}^n$, respectively. Similarly, scalar and vector valued functions will be denoted by, $x(\cdot) \in \mathbb{R}$ and $\boldsymbol{x}(\cdot) \in \mathbb{R}^n$, respectively. Matrices are denoted by uppercase symbols , e.g., $A \in \mathbb{R}^{n \times m}$. Vector $\boldsymbol{e} \in \mathbb{R}^k$ denotes the vector of ones. We denote by $\mathcal{R}$ the space of all real-valued functions from $\mathbb{R}^k$ to $\mathbb{R}$ and $\mathcal{B}$ the space of binary functions from $\mathbb{R}^k$ to $\{0, 1\}$.

2 Design of Piecewise Linear Decision Rules

In this section, we present our approach for one-stage adaptive optimization problems. Given matrices $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{m \times q}$, $C \in \mathbb{R}^{m \times k}$, vectors $\boldsymbol{c} \in \mathbb{R}^n$, $\boldsymbol{d} \in \mathbb{R}^q$ and an uncertainty set Ξ for the uncertain quantity $\boldsymbol{\xi} \in \mathbb{R}^k$, we are interested to choose functions $\boldsymbol{x}(\cdot) = (x_1(\cdot), \ldots, x_n(\cdot))^\top$, $x_i(\cdot) \in \mathcal{R}$ and $\boldsymbol{y}(\cdot) = (y_1(\cdot), \ldots, y_q(\cdot))^\top$, $y_i(\cdot) \in \mathcal{B}$ in order to solve:

$$
\begin{array}{ll}
\text{minimize} & \max\limits_{\boldsymbol{\xi} \in \Xi} \boldsymbol{c}^\top \boldsymbol{x}(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\xi}) \\
\text{subject to} & x_i(\cdot) \in \mathcal{R},\, i = 1, \ldots, n, \\
& y_i(\cdot) \in \mathcal{B},\, i = 1, \ldots, q, \\
& A\boldsymbol{x}(\boldsymbol{\xi}) + B\boldsymbol{y}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi.
\end{array}
\tag{1}
$$

Here, we assume that Problem (1) has a non-empty and bounded feasible region and the uncertainty set Ξ is a nonempty, convex and compact polyhedron

$$
\Xi = \left\{ \boldsymbol{\xi} \in \mathbb{R}^k : W\boldsymbol{\xi} \geq \boldsymbol{g},\, \xi_1 = 1 \right\}, \tag{2}
$$

where $W \in \mathbb{R}^{l \times k}$ and $\boldsymbol{g} \in \mathbb{R}^l$. The parameter ξ_1 is set equal to 1 without loss of generality as it allows us to represent affine functions of the non-degenerate outcomes $(\xi_2, \ldots, \xi_k)$ in a compact manner as linear functions of $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_k)$.

Problem (1) involves a continuum of decision variables and inequality constraints. Therefore, in order to make the problem amenable to numerical solutions, there is a need for suitable functional approximations for $\boldsymbol{x}(\cdot)$ and $\boldsymbol{y}(\cdot)$.

2.1 Design of Piecewise Linear Decision Rules for Real-Valued Decisions

To present our approach we consider first a special case of Problem (1) involving only real-valued recourse decisions:

$$\begin{array}{ll} \text{minimize} & \max_{\xi \in \Xi} \boldsymbol{c}^\top \boldsymbol{x}(\boldsymbol{\xi}) \\ \text{subject to} & x_i(\cdot) \in \mathcal{R},\ i = 1, \ldots, n, \\ & A\boldsymbol{x}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi. \end{array} \tag{3}$$

Bemporad *et al.* [4] showed that the optimal decision rules for Problem (3) are piecewise linear continuous functions.

Let $\mathcal{PC}$ be the space of piecewise linear continuous functions from $\mathbb{R}^k$ to $\mathbb{R}$ that admit the following structure:

$$x(\boldsymbol{\xi}) = \max\{\overline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{x}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{x}}_P^\top \boldsymbol{\xi}\}, \tag{4}$$

for some $P \in \mathbb{N}_+$ and $\overline{\mathbf{x}}_p, \underline{\mathbf{x}}_p \in \mathbb{R}^k$, $p \in \mathcal{P} := \{1, \ldots, P\}$.

Theorem 1 *Every piecewise linear continuous function defined over a compact set Ξ is an element of $\mathcal{PC}$.*

Proof Given any piecewise linear convex function $x : \mathbb{R}^k \to \mathbb{R}$, it can be written as the point-wise supremum of a finite collection of P linear functions:

$$x(\boldsymbol{\xi}) = \sup_{p=1,\ldots,P} \mathbf{x}_p^\top \boldsymbol{\xi},$$

for some $P \in \mathbb{N}_+$ and $\mathbf{x}_p \in \mathbb{R}^k$. Since the uncertainty set Ξ in (3) is compact, we can replace the supremum with the maximum operator, thus verifying that any piecewise linear convex function can be written in the form $x(\boldsymbol{\xi}) = \max\{\mathbf{x}_1^\top \boldsymbol{\xi}, \ldots, \mathbf{x}_P^\top \boldsymbol{\xi}\}$.

Hempel *et al.* [28, Lemma 4] states that for every piecewise linear continuous function $x : \mathbb{R}^k \to \mathbb{R}$ defined over a convex polyhedral Ξ, there exists $P \in \mathbb{N}_+$ and two piecewise linear continuous and convex functions $\overline{x}(\boldsymbol{\xi}) = \max\{\overline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{x}}_P^\top \boldsymbol{\xi}\}$ and $\underline{x}(\boldsymbol{\xi}) = \max\{\underline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{x}}_P^\top \boldsymbol{\xi}\}$ such that $x(\boldsymbol{\xi}) = \overline{x}(\boldsymbol{\xi}) - \underline{x}(\boldsymbol{\xi})$. This verifies that the space $\mathcal{PC}$ coincides with the space of all piecewise linear continuous functions defined over a compact set Ξ. ∎

The implication of Theorem 1 is that *for every piecewise linear continuous function, there exists $P \in \mathbb{N}_+$ such that the function can be written in the form* (4). Since the optimal decision rules for Problem (3) are piecewise linear continuous functions, see [4, Theorem 2], the optimal solution of Problem (3) is an element of $\mathcal{PC}$. Nevertheless, it will be computationally intractable to optimize over decisions that have arbitrarily many linear pieces. To this end, we restrict the space of admissible solutions from $\mathcal{PC}$ to $\mathcal{PC}(P)$, where $\mathcal{PC}(P)$ is the space of piecewise linear continuous functions where the number of linear pieces $P \in \mathbb{N}_+$ is fixed. This functional restriction yields the following semi-infinite problem, which for fixed P, involves only a finite number of decision variables, and an infinite number of constraints:

$$\begin{aligned} \text{minimize} \quad & \max_{\xi \in \Xi} \boldsymbol{c}^\top \boldsymbol{x}(\boldsymbol{\xi}) \\ \text{subject to} \quad & x_i(\cdot) \in \mathcal{PC}(P),\ i = 1, \ldots, n, \\ & A\boldsymbol{x}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi. \end{aligned} \tag{5}$$

The number of linear pieces P allows the decision maker to control the trade-off between tractability and optimality. A large P will result in greater flexibility, and thus achieving near optimal solutions, while setting $P = 1$ corresponds to the highly tractable linear decision rules. We emphasize that any feasible solution in (5) is also feasible in (3), since for any number of linear pieces P the inclusion $\mathcal{PC}(P) \subseteq \mathcal{PC}$ is valid. Therefore, the optimal value of Problem (5) provides an *upper bound* on the true optimal value of Problem (3).

The proposed approach can also be seen as a generalization of the work presented in Gorissen and den Hertog [25]. Their work, addresses the following type of constraints:

$$\mathbf{x}_1^\top \boldsymbol{\xi} + \sum_{i=1}^{n} \max\{\mathbf{x}_{i,1}^\top \boldsymbol{\xi}, \ldots, \mathbf{x}_{i,P}^\top \boldsymbol{\xi}\} \leq 0 \qquad \forall \boldsymbol{\xi} \in \Xi.$$

These constraints define a convex feasible region as they involve summation of maxima, each of which has a positive sign. The authors present exact reformulations for this class of constraints, as well as approximations through linear and polynomial decision rules. We emphasize that this type of constraints can be cast as an instance of (5), using appropriate matrices A and C and the proposed decision rules.

2.2 Design of Piecewise Linear Decision Rules for Binary Decisions

We now concentrate our attention to the binary recourse decisions and to the corresponding special case of Problem (1) involving only binary recourse decisions:

$$\begin{aligned} \text{minimize} \quad & \max_{\xi \in \Xi} \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\xi}) \\ \text{subject to} \quad & y_i(\cdot) \in \mathcal{B},\ i = 1, \ldots, q, \\ & B\boldsymbol{y}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi. \end{aligned} \tag{6}$$

Let $\mathcal{PB}$ be the space of piecewise linear binary functions from $\mathbb{R}^k$ to $\{0, 1\}$ that admit the following structure:

$$y(\boldsymbol{\xi}) = \begin{cases} 1, & \max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} \leq 0, \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

for some $P \in \mathbb{N}_+$ and $\overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p \in \mathbb{R}^k$, $p \in \mathcal{P} = \{1, \ldots, P\}$.

Theorem 2 *Every piecewise linear binary function defined over a compact set Ξ is an element of $\mathcal{PB}$.*

Proof From Theorem 1, we know that any piecewise linear continuous function defined on Ξ can be written in the form $\max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}\}$, for some $P \in \mathbb{N}_+$ and $\overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p \in \mathbb{R}^k$. Since a function $y(\cdot) \in \mathcal{PB}$ is just the projection of such piecewise linear continuous function on $\{0, 1\}$, we can conclude that $\mathcal{PB}$ coincides with the space of all piecewise linear binary functions defined over a compact set Ξ. ∎

As before, the implication of Theorem 2 is that *for every piecewise linear binary function, there exists $P \in \mathbb{N}_+$ such that the function can be written in the form* (7). Even though $\mathcal{PB}$ coincides with the space of piecewise linear binary functions, there are two main difficulties associated with its computational implementation: (i) As in the case of $\mathcal{PC}$, it is computationally intractable to optimize over decision rules that have an arbitrarily large number of linear pieces and (ii) it is not clear how one will implement in an optimization problem the open set defined by the complement of $\max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} \leq 0$. To this end, we define the finite counterpart of $\mathcal{PB}$. Let $\epsilon > 0$, and let $\mathcal{PB}(P)$ be the space of piecewise linear binary functions from $\mathbb{R}^k$ to $\{0, 1\}$ that have $P \in \mathbb{N}_+$ linear pieces and admit the following structure:

$$y(\boldsymbol{\xi}) = \begin{cases} 1, & \max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} \leq 0, \\ 0, & \max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} \geq \epsilon, \end{cases} \tag{8}$$

for some $\overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p \in \mathbb{R}^k$, $p \in \mathcal{P} = \{1, \ldots, P\}$. The constant ϵ, which for practical purposes will be defined as $\epsilon = 10^{-5}$, allows to express the open set $\max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}\} > 0$, within the context of an optimization problem. We note that for the small interval defined by ϵ, the decision rule can take arbitrarily the value 0 or 1.

Restricting the space of admissible solutions from $\mathcal{PB}$ to $\mathcal{PB}(P)$, yields the following semi-infinite problem, which for fixed number of linear pieces P, involves only a finite number of decision variables, and an infinite number of constraints:

$$\begin{aligned} \text{minimize} \quad & \max_{\xi \in \Xi} \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\xi}) \\ \text{subject to} \quad & y_i(\cdot) \in \mathcal{PB}(P),\ i = 1, \ldots, q, \\ & B\boldsymbol{y}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi. \end{aligned} \tag{9}$$

As in the case of Problems (5) and (3), any feasible solution in (9) is also feasible in (6), since for any number of linear pieces P the inclusion $\mathcal{PB}(P) \subseteq \mathcal{PB}$ is valid. Therefore, the optimal value of of Problem

(9) provides an *upper bound* on the true optimal value of Problem (6).

Remark 1 *Note that the functional spaces $\mathcal{PC}$ and $\mathcal{PB}$ do not necessarily contain the optimal solution of Problem* (1)*. This is the case as in the presence of binary decision rules, the optimal real-valued decision can be discontinuous. Nevertheless, $\mathcal{PC}$ and $\mathcal{PB}$ contain functions that are arbitrarily close to the optimal solution of* (1)*.*

2.3 Solution Method

Combining the two decision rules, and introducing an epigraph formulation, results in the following semi-infinite optimization problem, whose optimal value provides an upper bound for the optimal value of (1):

$$\begin{aligned} Z(\Xi) \;=\; &\underset{\tau\in\mathbb{R}}{\text{minimize}} && \tau \\ &\text{subject to} && x_i(\cdot)\in\mathcal{PC}(P),\ i=1,\dots,n, \\ &&& y_i(\cdot)\in\mathcal{PB}(P),\ i=1,\dots,q, \\ &&& \left.\begin{array}{l} \boldsymbol{c}^\top\boldsymbol{x}(\boldsymbol{\xi})+\boldsymbol{d}^\top\boldsymbol{y}(\boldsymbol{\xi})\le\tau \\ A\boldsymbol{x}(\boldsymbol{\xi})+B\boldsymbol{y}(\boldsymbol{\xi})\le C\boldsymbol{\xi} \end{array}\right\}\ \forall\boldsymbol{\xi}\in\Xi. \end{aligned} \tag{10}$$

Here, the optimal value of (10) is denoted by $Z(\Xi)$ to emphasize the dependence on the set Ξ. The key advantage of the proposed decision rules compared with the non-linear decision rules expressed by $x(\boldsymbol{\xi})=\boldsymbol{x}^\top L(\boldsymbol{\xi})$ in [5, 13, 23, 24], is that the structure of the decision rule is designed by the optimization problem. Moreover, as we will demonstrate in Section 2.3, reformulation of the problem into a finite dimensional optimization problem does not depend on the structure of the decision rule and the uncertainty set, unlike the decision rules discussed in [5, 23, 24], see [23, Section 4] for an extensive discussion. As we will demonstrate in Section 5, the latter characteristic plays an important role in the performance of the proposed decision rules.

There are two main difficulties associated with Problem (10):

1. It involves an infinite number of constraints as a result of the continuous structure of the uncertainty set Ξ;

2. The non-linear structure of $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$ results in non-convex constraints both with respect to decision variables and the uncertain parameters $\boldsymbol{\xi}$.

A typical solution strategy employed in robust optimization [5, 7] for reformulating the infinite number of constraints, utilizes duality arguments. Unfortunately, the non-convex structure of the constraints prohibits the use of these arguments.

An alternative approach for solving semi-infinite problems is to employ a cutting plane algorithm, first proposed in Blankenship and Falk [17]. Instead of solving a problem involving infinite number of constraints, the algorithm iteratively identifies a finite subset of binding scenarios $\widehat{\Xi} \subset \Xi$ in Problem (10), and solves the corresponding finite dimensional optimization problem. The overall algorithm converges to an optimal solution of (10) but not necessarily in finite number of iterations. This type of solution method directly addresses the semi-infinite structure of (10), but perhaps most importantly, it allows us to reformulate the non-linear constraints associated with $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$ and express them as mixed-integer linear constrains. In this section, we overview the cutting plane algorithm [17], and in the following section we provide reformulations for the constraints associated with $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$.

For a finite set $\widehat{\Xi} \subset \Xi$, we define the following finite dimensional variant of Problem (10):

$$
\begin{aligned}
Z(\widehat{\Xi}) \;=\; \underset{\tau \in \mathbb{R}}{\text{minimize}} \quad & \tau \\
\text{subject to} \quad & x_i(\cdot) \in \mathcal{PC}(P),\ i = 1, \dots, n, \\
& y_i(\cdot) \in \mathcal{PB}(P),\ i = 1, \dots, q, \\
& \left.\begin{aligned} & \boldsymbol{c}^\top \boldsymbol{x}(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\xi}) \le \tau \\ & A\boldsymbol{x}(\boldsymbol{\xi}) + B\boldsymbol{y}(\boldsymbol{\xi}) \le C\boldsymbol{\xi} \end{aligned}\right\} \forall \boldsymbol{\xi} \in \widehat{\Xi}.
\end{aligned}
\tag{11}
$$

As before, $Z(\widehat{\Xi})$ is the optimal value of Problem (11) and $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ denotes its optimal solution. As we will show in Section 2.4, Problem (11) can be reformulated as a mixed-integer linear optimization problem. Before we introduce the cutting plane algorithm, notice that for any finite subset $\widehat{\Xi} \subset \Xi$, Problem (11) underestimates the optimal value of Problem (10), i.e.,

$$
Z(\widehat{\Xi}) \;\le\; Z(\Xi).
$$

Therefore, to achieve robust feasibility, it is important to check if there exists scenario realizations $\boldsymbol{\xi} \in \Xi \backslash \widehat{\Xi}$, for which solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ violates any of the $m+1$ constraints in (10). To this end, we formulate $m+1$ auxiliary problems that will determine if solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ is robustly feasible, or return a scenario $\boldsymbol{\xi}$ for which the constraint is violated.

The first auxiliary problem is associated with the epigraph formulation of the objective function.

$$
\begin{aligned}
Q_0(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*) = \;\text{maximize} \quad & \boldsymbol{c}^\top \boldsymbol{x}^*(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}^*(\boldsymbol{\xi}) - \tau \\
\text{subject to} \quad & \boldsymbol{\xi} \in \Xi, \\
& x_i^*(\cdot) \in \mathcal{PC}(P),\ i = 1, \dots, n, \\
& y_i^*(\cdot) \in \mathcal{PB}(P),\ i = 1, \dots, q.
\end{aligned}
\tag{12a}
$$

Constraints $x^*(\cdot) \in \mathcal{PC}(P)$ and $y^*(\cdot) \in \mathcal{PB}(P)$ ensure that the fixed decision rules $x^*(\cdot)$ and $y^*(\cdot)$ maintain their structure, i.e., vectors $\overline{\mathbf{x}}_p^*, \underline{\mathbf{x}}_p^*$, $p \in \mathcal{P}$ and $\overline{\mathbf{y}}_p^*, \underline{\mathbf{y}}_p^*$, $p \in \mathcal{P}$ are fixed, while optimizing over $\boldsymbol{\xi}$. If a solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ is feasible in constraint $\boldsymbol{c}^\top \boldsymbol{x}^*(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}^*(\boldsymbol{\xi}) \leq \tau^*$ for all $\boldsymbol{\xi} \in \Xi$, then the optimal value $Q_0(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ will be less than or equal to zero. If, however, $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ is not robustly feasible, then the optimal value $Q_0(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ will be strictly positive, with the corresponding optimal solution $\boldsymbol{\xi}_0^*$ constituting a scenario for which $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ violated the underlying constraint. The same idea is used for the remaining constraints of Problem (10). Indeed, for each constraint $[A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) \leq C\boldsymbol{\xi}]_j$, $j = 1, \ldots, m$, we formulate the following auxiliary problems:

$$
\begin{aligned}
Q_j(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot)) = \ \ & \text{maximize} && [A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) - C\boldsymbol{\xi}]_j \\
& \text{subject to} && \boldsymbol{\xi} \in \Xi, \\
& && x_i^*(\cdot) \in \mathcal{PC}(P),\ i = 1, \ldots, n, \\
& && y_i^*(\cdot) \in \mathcal{PB}(P),\ i = 1, \ldots, q.
\end{aligned} \tag{12b}
$$

Here, the notation $[A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) - C\boldsymbol{\xi}]_j$ refers to the j^{th} entry in vector $A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) - C\boldsymbol{\xi}$. As we will demonstrate in Section 2.4, all auxiliary problems can be reformulated as mixed-integer linear optimization problems. We note that in the case where an auxiliary problem has more than one optimal solutions, without loss of generality, one can use the optimal solution returned by the mixed-integer linear solver.

The structure of the cutting plane algorithm is summarised as follows:

Algorithm 1 (Cutting plane [17])

Step 1: *Initialize: set $i = 1$ and choose $\widehat{\Xi}_1 \subset \Xi$.*

Step 2: *Solve the mixed-integer linear Problem (11) parameterised by $\widehat{\Xi}_i$, and set the optimal solution to $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$. If the problem is infeasible, then no feasible solution to Problem (10) exists and the algorithm terminates.*

Step 3: *Solve the $m + 1$ auxiliary, mixed-integer linear optimization Problems (12) parameterised by $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$, and let their optimal solutions to be $\boldsymbol{\xi}_j^*$, $j = 0, \ldots, m$, respectively. Also, let $V_i := \{\boldsymbol{\xi}_j^* :\ \text{if } Q_j > 0, j = 0, \ldots, m\}$.*

Step 4: *Check if the solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ satisfies all constraints, i.e., if all the following inequalities hold:*

$$
\begin{aligned}
& Q_0(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*) \leq 0 \\
& Q_j(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot)) \leq 0, \qquad j = 1, \ldots, m.
\end{aligned}
$$

If this is true, then the algorithm terminates and the optimal solution is $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$*, with the set of binding scenarios being* $\widehat{\Xi}^* = \widehat{\Xi}_i$*. Otherwise, set* $\widehat{\Xi}_{i+1} = \widehat{\Xi}_i \cup V_i$ *and* $i = i + 1$ *and return to* **Step 2**.

For the remainder of the paper, we will refer to scenarios in set $\widehat{\Xi}^*$ as the critically binding scenarios associated with the optimal solution of Problem (10). In Algorithm 1, the choice of $\widehat{\Xi}_1$ is not crucial and can be any arbitrary finite subset of Ξ. However, if the user has a priori knowledge on the position of the critically binding scenarios for the problem in hand, these scenarios can be added to $\widehat{\Xi}_1$, accelerating the convergence of the algorithm. For example, in a problem involving only real-valued adaptive decisions, one can first solve the linear decision rule problem and use the associated critically binding scenarios as a starting point in a non-linear decision rule problem.

Algorithm 1 explicitly assumes that all optimization problems involved are solved to global optimality. Under this condition, at termination, the algorithm will provide the *best piecewise linear decision rules* for Problem (10), i.e.,

$$Z(\widehat{\Xi}^*) \;=\; Z(\Xi).$$

This equality relation is valid despite the fact that Problem (11) is a relaxation of Problem (10). This is the case as the finite collection of scenarios $\widehat{\Xi}^*$ contains the worst-case scenarios associated with the optimal decision rule of (10).

In practice, solving all optimization problems to global optimality can be computationally challenging. Thus, ensuring that a solution is robustly feasible becomes the first priority. In this case, for given decisions $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$, a user can choose to terminate the optimization of the auxiliary Problems (12), if the objective value becomes strictly positive. This indicates that a realization $\boldsymbol{\xi}$ is found for which $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ violates some constraint in (10). If in addition, at every iteration Problem (11) is solved to global optimality, then the algorithm will still provide the best piecewise linear decision rules for Problem (10). This approach might require more iterations until the algorithm terminates but can potentially accelerate the solution method.

In the case where Problem (11) is not solved to global optimality in every iteration, at termination the algorithm can potentially provide a suboptimal solution to (10). Therefore, the solution obtained at termination can be a conservative approximation to (10), i.e.,

$$Z(\Xi) \;\leq\; Z(\widehat{\Xi}^*).$$

We emphasize that even if a suboptimal solution is achieved, the algorithm guarantees that all decisions are robustly feasible in (10) and consequently robustly feasible in (1).

2.3.1 Alternative Termination Criteria

In this section, we provide alternative termination criteria for Algorithm 1. Algorithm 1 can be computationally cumbersome for some problem instances, as it may require a large number of iterations, and consequently a large number of scenarios, until robust feasibility is achieved. To address this, we introduce a relaxation for the termination criteria in **Step 4**, by alternatively checking if a solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ satisfies the following inequality:

$$p := \mathbb{P}\left\{\boldsymbol{\xi} \in \Xi : \bigcup_{j=1}^{m}\left\{[A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) > C\boldsymbol{\xi}]_j\right\} \bigcup \left\{\boldsymbol{c}^\top \boldsymbol{x}^*(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}^*(\boldsymbol{\xi}) > \tau^*\right\}\right\} \leq \delta \qquad (13)$$

where $\mathbb{P}$ is the uniform distribution supported on Ξ, and $\delta \in [0, 1)$ is the violation level defined by the user. We will refer to p as the violation probability. If one sets $\delta = 0$, then (13) ensures robust feasibility in Problem (10), while the degree of violation increases as the value of δ increases.

Evaluating analytically the violation probability p, can be difficult for arbitrary constraints and uncertainty set Ξ. Therefore, we are forced to estimate this quantity using simulation, by computing the empirical violation probability $\widehat{p}_N$ as follows:

$$\widehat{p}_N := \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left(\bigcup_{j=1}^{m}\left\{[A\boldsymbol{x}^*(\boldsymbol{\xi}_i) + B\boldsymbol{y}^*(\boldsymbol{\xi}_i) > C\boldsymbol{\xi}_i]_j\right\} \bigcup \left\{\boldsymbol{c}^\top \boldsymbol{x}^*(\boldsymbol{\xi}_i) + \boldsymbol{d}^\top \boldsymbol{y}^*(\boldsymbol{\xi}_i) > \tau^*\right\}\right)$$

where $\mathbf{1}(\cdot)$ is the indicator function and N is the number of samples used. The samples $\boldsymbol{\xi}_i \in \Xi$ can be generated using techniques discussed in Calafiore and Dabbene [19].

A natural question that arises is: What is the number of samples N needed to accurately estimate p, from the values of the empirical violation probability $\widehat{p}_N$? To answer this, we use Hoeffding's inequality [29], which provides a lower bound on N. This idea was discussed in Calafiore and Campi [18]. The Hoeffding's inequality states that for a fixed solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$,

$$\mathbb{P}_N\left\{|\widehat{p}_N - p| \leq \widehat{\epsilon}\right\} \geq 1 - 2\exp(-2\widehat{\epsilon}^2 N), \quad \text{where} \quad N \geq \frac{\log(2/\widehat{\beta})}{2\widehat{\epsilon}^2},$$

which ensures that $|\widehat{p}_N - p| \leq \widehat{\epsilon}$ with confidence greater than $1 - \widehat{\beta}$. Here, $\mathbb{P}_N$ is the empirical uniform distribution constructed using the N i.i.d. samples, while $\widehat{\epsilon} \in [0, 1)$ and $\widehat{\beta} \in [0, 1)$ are fixed constants defined by the user. In other words, by computing the empirical violation probability $\widehat{p}_N$ using at least $N \geq \log(2/\widehat{\beta})/(2\widehat{\epsilon}^2)$ samples, we are guarantied to be within $\widehat{\epsilon}$ of the exact violation probabilities p, with confidence greater than $1 - \widehat{\beta}$. We emphasize that N can be potentially large depending on the choice of $\widehat{\epsilon}$ and $\widehat{\beta}$. Nevertheless, calculating the empirical violation probability using N samples is not a computationally intensive process as it does not involve an optimization problem.

The alternative termination criterion to **Step 4** is given as follows:

Step 4′: Given fixed $\delta \in [0,1)$, $\widehat{\epsilon} \in [0,1)$ and $\widehat{\beta} \in [0,1)$, generate $N \geq \log(2/\widehat{\beta})/(2\widehat{\epsilon}^2)$ i.i.d. samples from the uniform distribution supported on Ξ and check if the solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ satisfies the following inequality:

$$\frac{1}{N}\sum_{i=1}^{N}\mathbf{1}\left(\bigcup_{j=1}^{m}\left\{[A\boldsymbol{x}^*(\boldsymbol{\xi}_i) + B\boldsymbol{y}^*(\boldsymbol{\xi}_i) > C\boldsymbol{\xi}_i]_j\right\}\bigcup\left\{\boldsymbol{c}^\top\boldsymbol{x}^*(\boldsymbol{\xi}_i) + \boldsymbol{d}^\top\boldsymbol{y}^*(\boldsymbol{\xi}_i) > \tau^*\right\}\right) \leq \delta$$

If the inequality hold, then the algorithm terminates and the optimal solution is $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$, with the set of binding scenarios being $\widehat{\Xi}^* = \widehat{\Xi}_i$. Otherwise, set $\widehat{\Xi}_{i+1} = \widehat{\Xi}_i \cup V_i$ and $i = i+1$ and return to **Step 2**.

We define **Algorithm 2** that has **Step 1-3** from **Algorithm 1** and **Step 4′**.

To illustrate the implications of **Step 4′**, we present an example for fixed parameters $\delta, \widehat{\epsilon}$ and $\widehat{\beta}$.

Example 1 *Set the violation level* $\delta = 1\%$*, and let* $\widehat{\epsilon} = 0.005$ *and* $\widehat{\beta} = 10^{-4}$ *such that* $N \geq 198070$*. If for a given solution* $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ *the empirical violation probability is, say,* $\widehat{p}_N = 70\%$*, then Hoeffding's inequality ensures that the exact violation probability* p *lays within*

$$69.5\% \leq p \leq 70.5\%,$$

with a confidence greater 99.99%*. Since* $\delta = 1\%$*, we require that the algorithm terminates when at least* 99% *feasibility is achieved. Therefore, we can conclude that solution* $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$ *does satisfy* (13)*, thus a violating scenario is determined using the auxiliary Problems* (12)*, and Problem* (11) *is resolved using an updated* $\widehat{\Xi}$*.*

2.3.2 Probabilistic Guarantees for the Convergence Properties of Algorithm 2

In this section, we provide probabilistic guarantees for the convergence properties of Algorithm 2 by providing an alternative scenario selection procedure. Given a solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$, **Step 3** of Algorithm 2 generates scenarios by solving the auxiliary Problems (12). From a theoretical standpoint, this scenario selection does not guarantee that the algorithm terminates in a finite number of iterations. This is particularly true if Problem (10) involves binary recourse decisions. To address this, we introduce an alternative scenario selection procedure which ensures that Algorithm 2 will terminate in a finite number of steps under the termination criteria **Step 4′** with a high confidence level.

Before we introduce the alternative scenario selection, a natural question that arises is: What is the number of samples N needed such that the solution of Problem (11) satisfies constraint (13) with high confidence, without necessarily applying Algorithm 2? To answer this, we use the following result from statistical learning theory discussed in Anthony and Biggs [2], and was first utilized in the optimization literature by de Farias and Van-Roy [22] and Caramanis [20].

Proposition 1 *Given fixed* $\delta \in [0, 1)$ *and* $\beta \in [0, 1)$*, construct the finite set* $\widetilde{\Xi}_N \subseteq \Xi$ *containing* N *i.i.d. samples from the uniform distribution supported on* Ξ *such that*

$$N \geq \frac{4}{\delta}\left(\mathcal{V}_{x(\cdot),y(\cdot),\tau}\ln\left(\frac{12}{\delta}\right)+\ln\left(\frac{2}{\beta}\right)\right), \tag{14}$$

where $\mathcal{V}_{x(\cdot),y(\cdot),\tau} = 2(1+2P(n+q)k)\log_2\Big(4e(m+1)(2P)(n+q)\Big)$*. Then, any feasible solution of Problem* (11) *satisfies constraint* (13) *with probability at least* $1-\beta$*.*

Proof From [20, Proposition 4.6] we have that any feasible solution of Problem (11) satisfies constraint (13) with probability at least $1-\beta$, provided that N i.i.d. samples are drawn from the uniform distribution supported on Ξ with

$$N \geq \frac{4}{\delta}\left(\mathcal{V}\ln\left(\frac{12}{\delta}\right)+\ln\left(\frac{2}{\beta}\right)\right). \tag{15}$$

Here, $\mathcal{V}$ is the VC-dimension of set $C := \{C_{x(\cdot),y(\cdot),\tau} : x_i(\cdot) \in \mathcal{PC}(P), i = 1,\dots,n,\ y_i(\cdot) \in \mathcal{PB}(P), i = 1,\dots,q,\ \tau \in \mathbb{R}\}$, where

$$C_{x(\cdot),y(\cdot),\tau} := \left\{\boldsymbol{\xi} \in \Xi : \bigcup_{j=1}^{m}\left\{[A\boldsymbol{x}(\boldsymbol{\xi}) + B\boldsymbol{y}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi}]_j\right\} \bigcup \left\{\boldsymbol{c}^\top\boldsymbol{x}(\boldsymbol{\xi}) + \boldsymbol{d}^\top\boldsymbol{y}(\boldsymbol{\xi}) \leq \tau\right\}\right\}.$$

It is generally a hard problem to calculate exactly the VC-dimension of set C. Nevertheless, since $\bigcup_{j=1}^{m}\{[A\boldsymbol{x}(\boldsymbol{\xi})+B\boldsymbol{y}(\boldsymbol{\xi}) \leq C\boldsymbol{\xi}]_j\} \bigcup\{\boldsymbol{c}^\top\boldsymbol{x}(\boldsymbol{\xi})+\boldsymbol{d}^\top\boldsymbol{y}(\boldsymbol{\xi}) \leq \tau\}$ can be expressed as a Boolean formula involving standard logical connectives, Vidyasagar [34, Theorem 4.6] provides the following upper bound on the VC-dimension of C:

$$VC\text{-dimension}(C) \leq \mathcal{V}_{x(\cdot),y(\cdot)} = 2(1+2P(n+q)k)\log_2\Big(4\mathrm{e}(m+1)(2P)(n+q)\Big),$$

where the constant e is the Euler's number. We therefore conclude that the bound (14) is a conservative approximation to (15) and thus the result follows. ∎

From Proposition 1, we see that the number of samples needed such that the solution of Problem (11) satisfies constraint (13) with high confidence, grows polynomially with respect to the dimension of the uncertainty set, k, the number of continuous and binary decision rules, $(n+q)$, and the number of linear pieces in the decision rule, $2P$. In addition, since the number of constraints, m, and the confidence level, β, appears in the logarithm, their size does not contribute significantly to the final sample size.

Nevertheless, for a small choice of $\delta \in [0,1)$, the number of samples can render Problem (11) intractable. To this end, we formulate $m+1$ variants of the auxiliary Problems (12) that identify a critical subset of these scenarios. For fixed $\delta \in [0,1)$, $\beta \in [0,1)$ and solution $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$, the $m+1$ auxiliary problems are defined as follows:

$$
\begin{aligned}
Q'_0(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*) = \quad & \text{maximize} && \boldsymbol{c}^\top \boldsymbol{x}^*(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}^*(\boldsymbol{\xi}) - \tau \\
& \text{subject to} && \boldsymbol{\xi} \in \widetilde{\Xi}_N \\
& && x_i^*(\cdot) \in \mathcal{PC}(P),\ i = 1, \dots, n, \\
& && y_i^*(\cdot) \in \mathcal{PB}(P),\ i = 1, \dots, q,
\end{aligned}
\tag{16a}
$$

where $\widetilde{\Xi}_N$ is finite subset of Ξ containing N i.i.d. samples from the uniform distribution supported on Ξ, generated using bound (14). Similarly, for each constraint $[A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) \le C\boldsymbol{\xi}]_j,\ j = 1, \dots, m,$ we have

$$
\begin{aligned}
Q'_j(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot)) = \quad & \text{maximize} && [A\boldsymbol{x}^*(\boldsymbol{\xi}) + B\boldsymbol{y}^*(\boldsymbol{\xi}) - C\boldsymbol{\xi}]_j \\
& \text{subject to} && \boldsymbol{\xi} \in \widetilde{\Xi}_N, \\
& && x_i^*(\cdot) \in \mathcal{PC}(P),\ i = 1, \dots, n, \\
& && y_i^*(\cdot) \in \mathcal{PB}(P),\ i = 1, \dots, q.
\end{aligned}
\tag{16b}
$$

Problems (16) are difficult to solve due to the discrete nature of $\widetilde{\Xi}_N$. Nevertheless, one can solve Problems (16), by evaluating and sorting the objective function for each $\widetilde{\Xi}_N$. This can be done efficiently as the number of samples grow polynomially with the input parameters and it does not require to solve a mixed-integer linear optimization problem.

The alternative scenario selection procedure to **Step 3** is given as follows:

Step 3a′: Solve the $m+1$ auxiliary Problems (16) parameterized by $(\boldsymbol{x}^*(\cdot), \boldsymbol{y}^*(\cdot), \tau^*)$, and let their optimal solutions to be $\boldsymbol{\xi}_j^*$, $j = 0, \dots, m$, respectively. Also, let $V_i := \{\boldsymbol{\xi}_j^* :\ \text{if } Q'_j > 0, j = 0, \dots, m\}$.

We define **Algorithm 3** that has **Step 1-2** from **Algorithm 1** together with **Step 3′** and **Step 4′**. Note also, that the scenario selection **Step 3′** can be used together with **Step 3** for improved scenario selection. We close this section by emphasizing that since $\widetilde{\Xi}_N$ contains only finitely many scenarios, in the worst-case, **Algorithm 3** will converge when all scenarios in $\widetilde{\Xi}_N$ are added to $\widehat{\Xi}^*$.

2.4 Mixed-Integer Linear Optimization Formulations

In this section, we show that Problems (11), (12), have mixed-integer linear optimization reformulations. We now present three equivalent mixed-integer linear reformulations for each functional space $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$. These reformulations use Big-M, special ordered sets (SOS) and indicator constraints (IC), respectively, see Bertsimas and Weismantel [15] for more details. In particular, we will use special ordered sets of type 1 (SOS1) and type 2 (SOS2). Special ordered sets of type 1 ensure that in a set of variables, say $z_1, \dots, z_n$, at most one variable may be nonzero. This will be denoted by $\mathrm{SOS1}(z_1, \dots, z_n)$. Similarly, special ordered sets of type 2 ensure that in a set of variables, at most two consecutive variables may be nonzero and this will be denoted by $\mathrm{SOS2}(z_1, \dots, z_n)$. Indicator constraints express relationships among variables by identifying binary variables to control whether or not specified linear constraints are active. For example, the indicator constraint $z = 1 \iff \boldsymbol{x}^\top \boldsymbol{\xi} \le 0$, ensures that the inequality constraint $\boldsymbol{x}^\top \boldsymbol{\xi} \le 0$ is satisfied if and only if the binary variable z is equal to 1. SOS and indicator constraints are supported in many commercial and non-commercial optimization software such as IBM ILOG CPLEX [30] and SCIP [32]. Choosing which formulation to use depends on the problem structure and the efficiency of the optimization software used.

The functional spaces $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$ can be parameterised in terms of the decision variables $\overline{\mathbf{x}}_p, \underline{\mathbf{x}}_p$, $p \in \mathcal{P}$ and $\overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p$, $p \in \mathcal{P}$, and the uncertain vector $\boldsymbol{\xi}$. Indeed, with slight abuse of notation, $\mathcal{PC}(P)$ can be equivalently written as

$$\mathcal{PC}(P, \overline{\mathbf{x}}_p, \underline{\mathbf{x}}_p, \boldsymbol{\xi}) = \left\{ x \in \mathbb{R} \ : \ x = \max\{\overline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \dots, \overline{\mathbf{x}}_P^\top \boldsymbol{\xi}\} - \max\{\underline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \dots, \underline{\mathbf{x}}_P^\top \boldsymbol{\xi}\} \right\}, \tag{17}$$

and $\mathcal{PB}(P)$ can be equivalently written as

$$\mathcal{PB}(P, \overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p, \boldsymbol{\xi}) = \left\{ y \in \{0,1\} : \psi(\cdot) \in \mathcal{PC}(P, \overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p, \boldsymbol{\xi}), \ \ y = \left\{ \begin{array}{ll} 1, & \psi(\boldsymbol{\xi}) \le 0, \\ 0, & \psi(\boldsymbol{\xi}) \ge \epsilon, \end{array} \right. \right\}. \tag{18}$$

The following proposition provides three alternative formulations for the set of piecewise linear continuous decision rules $\mathcal{PC}(P)$.

Proposition 2 *The feasible region given by* (17) *is equivalent to:*

$$\mathcal{PC}_{\textit{Big-M}}: \left\{ \begin{array}{l} x \in \mathbb{R} \ : \ \exists\, (\overline{\boldsymbol{z}}, \overline{x}), (\underline{\boldsymbol{z}}, \underline{x}) \in \{0,1\}^P \times \mathbb{R}, \ \textit{such that } x = \overline{x} - \underline{x}, \ \boldsymbol{e}^\top \overline{\boldsymbol{z}} = \boldsymbol{e}^\top \underline{\boldsymbol{z}} = 1, \\ \qquad \left. \begin{array}{ll} \overline{x} \ge \overline{\mathbf{x}}_p^\top \boldsymbol{\xi}, & \underline{x} \ge \underline{\mathbf{x}}_p^\top \boldsymbol{\xi}, \\ \overline{x} \le \overline{\mathbf{x}}_p^\top \boldsymbol{\xi} + M(1 - \overline{z}_p), & \underline{x} \le \underline{\mathbf{x}}_p^\top \boldsymbol{\xi} + M(1 - \underline{z}_p), \end{array} \right\} \forall p \in \mathcal{P} \end{array} \right\}, \tag{19}$$

given a sufficiently large constant $M \in \mathbb{R}_+$;

$$\mathcal{PC}_{\boldsymbol{SOS}}: \left\{ \begin{array}{l} x \in \mathbb{R} \ : \ \exists(\overline{\boldsymbol{z}}, \overline{\boldsymbol{\chi}}, \overline{x}), (\underline{\boldsymbol{z}}, \underline{\boldsymbol{\chi}}, \underline{x}) \in \{0,1\}^P \times \mathbb{R}_+^P \times \mathbb{R}, \\ \qquad \textit{such that } x = \overline{x} - \underline{x},\ \boldsymbol{e}^\top \overline{\boldsymbol{z}} = \boldsymbol{e}^\top \underline{\boldsymbol{z}} = 1,\ \boldsymbol{e}^\top \overline{\boldsymbol{\chi}} \le M,\ \boldsymbol{e}^\top \underline{\boldsymbol{\chi}} \le M, \\ \qquad\qquad \left. \begin{array}{ll} \overline{x} \ge \overline{\mathbf{x}}_p^\top \boldsymbol{\xi}, & \underline{x} \ge \underline{\mathbf{x}}_p^\top \boldsymbol{\xi}, \\ \overline{x} \le \overline{\mathbf{x}}_p^\top \boldsymbol{\xi} + \overline{\chi}_p, & \underline{x} \le \underline{\mathbf{x}}_p^\top \boldsymbol{\xi} + \underline{\chi}_p, \\ \mathrm{SOS1}(\overline{z}_p, \overline{\chi}_p), & \mathrm{SOS1}(\underline{z}_p, \underline{\chi}_p), \end{array} \right\} \forall p \in \mathcal{P}, \end{array} \right\}, \tag{20}$$

given a sufficiently large constant $M \in \mathbb{R}_+$;

$$\mathcal{PC}_{\boldsymbol{IC}}: \left\{ \begin{array}{l} x \in \mathbb{R} \ : \ \exists\,(\overline{\boldsymbol{z}}, \overline{x}), (\underline{\boldsymbol{z}}, \underline{x}) \in \{0,1\}^P \times \mathbb{R},\ \textit{such that } x = \overline{x} - \underline{x},\ \boldsymbol{e}^\top \overline{\boldsymbol{z}} = \boldsymbol{e}^\top \underline{\boldsymbol{z}} = 1, \\ \qquad\qquad \left. \begin{array}{l} \overline{z}_p = 1 \iff \overline{x} = \overline{\mathbf{x}}_p^\top \boldsymbol{\xi}, \\ \underline{z}_p = 1 \iff \underline{x} = \underline{\mathbf{x}}_p^\top \boldsymbol{\xi}, \end{array} \right\} \forall p \in \mathcal{P}. \end{array} \right\}. \tag{21}$$

Proof We first introduce a simplified version of (17) involving a single max function:

$$\widehat{\mathcal{PC}}(P, \mathbf{x}_p, \boldsymbol{\xi}) = \left\{ x \in \mathbb{R} \ : \ x = \max\{\mathbf{x}_1^\top \boldsymbol{\xi}, \ldots, \mathbf{x}_P^\top \boldsymbol{\xi}\} \right\}. \tag{22}$$

The feasible region (22) is known to be equivalent to

$$\widehat{\mathcal{PC}}_{\mathbf{Big\text{-}M}}(P, \mathbf{x}_p, \boldsymbol{\xi}) = \left\{ \begin{array}{l} x \in \mathbb{R} \ : \ \exists \boldsymbol{z} \in \{0,1\}^P,\ \text{such that } \boldsymbol{e}^\top \boldsymbol{z} = 1, \\ \qquad\qquad \left. \begin{array}{l} x \ge \mathbf{x}_p^\top \boldsymbol{\xi}, \\ x \le \mathbf{x}_p^\top \boldsymbol{\xi} + M(1 - z_p), \end{array} \right\} \forall p \in \mathcal{P} \end{array} \right\}, \tag{23}$$

given a sufficiently large constant $M \in \mathbb{R}$, see [25, Section 2]. It is therefore easy to see that (17) can be expressed as (19), with the variables $(\overline{\boldsymbol{z}}, \overline{x})$ being used to reformulate $\overline{x} = \max\{\overline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \ldots, \overline{\mathbf{x}}_P^\top \boldsymbol{\xi}\}$ and $(\underline{\boldsymbol{z}}, \underline{x})$ being used to reformulate $\underline{x} = \max\{\underline{\mathbf{x}}_1^\top \boldsymbol{\xi}, \ldots, \underline{\mathbf{x}}_P^\top \boldsymbol{\xi}\}$.

To show that (17) can be expressed as (20), we first show that the feasible region (22) is equivalent to

$$\widehat{\mathcal{PC}}_{\mathbf{SOS}}(P, \mathbf{x}_p, \boldsymbol{\xi}) = \left\{ \begin{array}{l} x \in \mathbb{R} \ : \ \exists(\boldsymbol{z}, \boldsymbol{\chi}) \in \{0,1\}^P \times \mathbb{R}_+^P,\ \text{such that } \boldsymbol{e}^\top \boldsymbol{z} = 1,\ \boldsymbol{e}^\top \boldsymbol{\chi} \le M \\ \qquad\qquad \left. \begin{array}{l} x \ge \mathbf{x}_p^\top \boldsymbol{\xi}, \\ x \le \mathbf{x}_p^\top \boldsymbol{\xi} + \chi_p, \\ \mathrm{SOS1}(z_p, \chi_p), \end{array} \right\} \forall p \in \mathcal{P} \end{array} \right\}. \tag{24}$$

given a sufficiently large constant $M \in \mathbb{R}$. Notice that (24) share similarities with (23). Indeed, the constraints $x \le \mathbf{x}_p^\top \boldsymbol{\xi} + M(1 - z_p)$, $p \in \mathcal{P}$ used to underestimate the max function in (22), are replaced by constraints $x \leq \mathbf{x}_p^\top \boldsymbol{\xi} + \chi_p$, $p \in \mathcal{P}$ together with $\boldsymbol{e}^\top \boldsymbol{\chi} \leq M$ and $\text{SOS1}(z_p, \chi_p)$, $p \in \mathcal{P}$. Given a feasible point (22) and assuming that there exists $M \in \mathbb{R}_+$ such that $\mathbf{x}_i^\top \boldsymbol{\xi} \leq M + \mathbf{x}_p^\top \boldsymbol{\xi}$ for all $i, p \in \mathcal{P}$, we can verify that

$$\begin{aligned} &\text{if } z_p = 1 \implies \chi_p = 0 \text{ and } x = \mathbf{x}_p^\top \boldsymbol{\xi}, \\ &\text{if } z_p = 0 \implies \chi_p \geq 0 \text{ and } x \leq \mathbf{x}_p^\top \boldsymbol{\xi} + \chi_p, \end{aligned}$$

Therefore, any feasible point in (22) is feasible in (20). The result that any feasible point in (20) is feasible in (22) follows trivially. Using similar arguments as before, we deduce that the feasible region of (17) can be expressed as (20).

To show that (17) can be expressed as (21), we first show that the feasible region (22) is equivalent to

$$\widehat{\mathcal{PC}}_{\mathbf{IC}}(P, \mathbf{x}_p, \boldsymbol{\xi}) = \left\{ \; x \in \mathbb{R} \; : \exists \boldsymbol{z} \in \{0,1\}^P, \; \boldsymbol{e}^\top \boldsymbol{z} = 1, z_p = 1 \iff x = \mathbf{x}_p^\top \boldsymbol{\xi}, \; \forall p \in \mathcal{P}. \; \right\}. \tag{25}$$

It is easy to see that $x = \max\{\mathbf{x}_1^\top \boldsymbol{\xi}, \ldots, \mathbf{x}_P^\top \boldsymbol{\xi}\}$ is equivalent to $x = \mathbf{x}_p^\top \boldsymbol{\xi}$ for exactly one $p \in \mathcal{P}$. This is equivalently expressed as the indicator constraints $z_p = 1 \iff x = \mathbf{x}_p^\top \boldsymbol{\xi}$, $\forall p \in \mathcal{P}$ together with $\boldsymbol{e}^\top \boldsymbol{z} = 1$. Using similar arguments as before, we deduce that the feasible region of (17) can be expressed as (21). ∎

Note that for fixed $\boldsymbol{\xi} \in \widehat{\Xi}$, the constraint $x(\boldsymbol{\xi}) \in \mathcal{PC}(P)$ in Problem (11) can be reformulate using (19), (20), (21), thus allowing to optimize over vectors $\overline{\mathbf{x}}_p, \underline{\mathbf{x}}_p \in \mathbb{R}^k$, $p \in \mathcal{P}$. Similarly, for fixed vectors $\overline{\mathbf{x}}_p^*, \underline{\mathbf{x}}_p^*$, $p \in \mathcal{P}$, the constraint $x^*(\cdot) \in \mathcal{PC}(P)$ in Problems (12) can be reformulate using (19), (20), (21), thus allowing to optimize over $\boldsymbol{\xi} \in \Xi$.

The following proposition provides three alternative formulations for the set of piecewise linear binary decision rules $\mathcal{PB}(P)$.

Proposition 3 *The feasible region given by* (18) *is equivalent to:*

$$\mathcal{PB}_{\textit{Big-M}} : \left\{ \begin{array}{l} y \in \{0,1\} \; : \; \psi(\boldsymbol{\xi}) \in \mathcal{PC}_{\textit{Big-M}}(P, \overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p, \boldsymbol{\xi}), \\ \hfill \textit{such that } \epsilon + \underline{M} y \leq \psi(\boldsymbol{\xi}) \leq \overline{M}(1-y) \end{array} \right\}, \tag{26}$$

given sufficiently large constant $\overline{M} \in \mathbb{R}_+$ *and sufficiently small constant* $\underline{M} \in \mathbb{R}_-$;

$$\mathcal{PB}_{\textit{SOS}} : \left\{ \begin{array}{l} y \in \{0,1\} \; : \; \exists \phi, \chi \in \mathbb{R}_+ \textit{ such that } \psi(\boldsymbol{\xi}) \in \mathcal{PC}_{\textit{SOS}}(P, \overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p, \boldsymbol{\xi}), \\ \hfill \textit{with } \epsilon - \phi \leq \psi(\boldsymbol{\xi}) \leq \chi, \; \text{SOS2}(\phi, y, 1-y, \chi), \; \phi + \chi \leq M \end{array} \right\}, \tag{27}$$

given sufficiently large constants $M \in \mathbb{R}_+$;

$$\mathcal{PB}_{\boldsymbol{IC}}: \left\{ y \in \{0,1\} \ : \ \psi(\boldsymbol{\xi}) \in \mathcal{PC}_{\boldsymbol{IC}}(P, \overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p, \boldsymbol{\xi}), \ \ y = 1 \iff \psi(\boldsymbol{\xi}) \leq 0 \right\}. \tag{28}$$

Proof We first show that any feasible point in (18) is feasible in (26). Under the assumption that there exists constants $\overline{M} \in \mathbb{R}_+$ and $\underline{M} \in \mathbb{R}_-$ such that $\epsilon + \underline{M} \leq \psi(\boldsymbol{\xi}) \leq \overline{M}$, we can verify that

$$\begin{aligned} &\text{if } y = 1 \implies \epsilon + \underline{M} \leq \psi(\boldsymbol{\xi}) \leq 0, \\ &\text{if } y = 0 \implies \epsilon \leq \psi(\boldsymbol{\xi}) \leq \overline{M}. \end{aligned}$$

Therefore, any feasible point in (18) is feasible in (26). The result that any feasible point in (26) is feasible in (18) follows trivially.

To show that any feasible point in (18) is feasible in (27), we assume that there exists constant $M \in \mathbb{R}_+$ such that $\epsilon + M \leq \psi(\boldsymbol{\xi}) \leq M$. Therefore,

$$\begin{aligned} &\text{if } y = 1 \implies \chi = 0 \ \text{ and } \ \epsilon - \phi \leq \psi(\boldsymbol{\xi}) \leq 0, \\ &\text{if } y = 0 \implies \phi = 0 \ \text{ and } \ \epsilon \leq \psi(\boldsymbol{\xi}) \leq \psi, \end{aligned}$$

and thus, any feasible point in (18) is feasible in (27). The result that any feasible point in (26) is feasible in (18) follows trivially. Finally, the equivalence between the feasible regions (18) and (28), follows from definition (7). ∎

Note that for fixed $\boldsymbol{\xi} \in \widehat{\Xi}$, the constraint $y(\boldsymbol{\xi}) \in \mathcal{PB}(P)$ in Problem (11) can be reformulate using (26), (27), (28), thus allowing to optimize over vectors $\overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p \in \mathbb{R}^k$, $p \in \mathcal{P}$. Similarly, for fixed vectors $\overline{\mathbf{y}}_p^*, \underline{\mathbf{y}}_p^*$, $p \in \mathcal{P}$, the constraint $y^*(\cdot) \in \mathcal{PB}(P)$ in Problems (12) can be reformulate using (26), (27), (28), thus allowing to optimize over $\boldsymbol{\xi} \in \Xi$.

We end this section by pooling together the central insights of Section 2. If Problem (1) has a non-empty and bounded feasible region, and a polyhedral uncertainty set, then the cutting plane Algorithms 1, 2 or 3 can be applied to Problem (10). At each iteration of the algorithms, Problems (11) and (12) can be reformulated into mixed-integer linear optimization problems using (19), (20), (21) for the continuous decision rules and (26), (27), (28) for the binary decision rules. In particular, if one uses reformulations (19) and (26), then Problem (11) has a total of $1 + (3|\widehat{\Xi}| + 2Pk)n + (3|\widehat{\Xi}| + 2Pk)q$ continuous variables, $|\widehat{\Xi}|2Pn + |\widehat{\Xi}|(1+2P)q$ binary variables and $(m+1) + |\widehat{\Xi}|(3+4P)n + |\widehat{\Xi}|(5+4P)q$ constraints, where $|\widehat{\Xi}|$ indicates the cardinality of $\widehat{\Xi}$. Similarly, each of the Problems (12) has a total of $k + 3n + 3q$ continuous variables, $2Pn + (1+2P)q$ binary variables and $l + (3+4P)n + (5+3P)q$ constraints. The number of decision variables and constraints in the reformulations of Problems (11) and (12) using (20), (21) and (27), (28) is of similar order.

3 Lower Bounds on the Optimal Adaptive Solution

In this section, we survey the main results from the literature for assessing the quality of the decision rules, and discuss how we can utilize the ideas in Hadjiyiannis *et al.* [27] in conjunction with the proposed decision rules.

The functional spaces $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$ restrict the feasible region of the continuous and binary decisions, thus providing an upper bound on the optimal solution of (1). Although one can improve the quality of the decision rule by increasing the number of linear pieces at the expense of computational tractability, it is not clear what is the absolute loss of optimality compared to the true optimal solution of Problem (1). There are a number of papers devoted to assessing this loss of optimality for adaptive optimization problems *involving only continuous decisions*. These papers can be divided to *a priori* and *a posteriori* assessments.

An a priori result was proposed by Bertsimas and Goyal in [11]. Here, the authors proved that for any problem instance (3), the worst-case cost associated with the linear decision rules can be of the order $O(\sqrt{k})$ suboptimal, where k is the number of random parameters in the problem. This result is also valid for non-linear decision rules. In addition, Bertsimas and Bidkhori [9] show that the quality of linear decision rules is directly linked with the geometric properties of the uncertainty set.

A posteriori results were developed in [23, 27, 31]. In this context, the label "a posteriori" means that the resulting quality measure is specific for each problem instance. To measure the loss of optimality, Kuhn *et al.* [31] proposed to apply the linear decision rule approximation to the dual version of (3). Their work was motivated by the fact that any feasible solution to the dual problem provides a progressive approximation to the optimal value of (3). Since the dual of (3) shares similar structure as its primal problem, applying linear decision rules to the dual adaptive decisions results to a progressive approximation. The corresponding optimal value constitutes a lower bound on the optimal value of (3). An estimation of the quality for any upper bounding approximation can therefore be obtained by comparing the gap between the upper and lower bounds. If this gap is small, then the upper bounding approximation can be thought as nearly optimal. On the other hand if the gap is big, then linear decision rules are potentially suboptimal, and the user should consider using a richer class of decision rules. Utilizing the dual version of (3) was later refined in Georghiou *et al.* [23], where it was extended for classes of non-linear decision rules. The extra functional flexibility offered by non-linear decision rules translated to improved lower bounds, giving improved estimates for the loss of optimality. An alternative a posteriori bound proposed by Hadjiyiannis *et al.* [27] generates a lower bound by solving the scenario counterpart of Problem (3). Since the scenario counterpart is constructed using a finite subset of the constraints in (3), its optimal solution will overestimate the optimal solution of (3). As before, the comparison between the upper bound and this lower bound can be used as an indication for the loss of optimality.

We now present an adaptation of the idea in Hadjiyiannis *et al.* [27], for problems involving both continuous and binary decisions and discuss its implications. The scenario counterpart of Problem (1) is given by the following optimization problem:

$$
\begin{aligned}
S(\widehat{\Xi}) \;=\; \text{minimize} \quad & \max_{\xi \in \widehat{\Xi}} \boldsymbol{c}^\top \boldsymbol{x}(\boldsymbol{\xi}) + \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\xi}) \\
\text{subject to} \quad & \left.\begin{array}{l} \boldsymbol{x}(\boldsymbol{\xi}) \in \mathbb{R}^n,\ \boldsymbol{y}(\boldsymbol{\xi}) \in \{0,1\}^q \\ A\boldsymbol{x}(\boldsymbol{\xi}) + B\boldsymbol{y}(\boldsymbol{\xi}) \le C\boldsymbol{\xi} \end{array}\right\} \quad \forall \boldsymbol{\xi} \in \widehat{\Xi}.
\end{aligned} \tag{29}
$$

Here, $\widehat{\Xi}$ is a finite subset of the uncertainty set Ξ and $S(\widehat{\Xi})$ denotes the optimal value of Problem (29). Since (29) is a relaxation of the original problem, the inequality $S(\widehat{\Xi}) \le S(\Xi)$ is always valid for any $\widehat{\Xi} \subset \Xi$. The quality of lower bound from (29) depends on the choice of the finite set $\widehat{\Xi}$. Bertsimas and Goyal [11] have shown that if (1) involves only continuous decisions and set Ξ_{ext} consists of the extreme points of Ξ, then the optimal value of scenario counterpart coincide with (1), i.e., $S(\Xi_{\text{ext}}) = S(\Xi)$. Of course, for most practical problems, one cannot enumerate the extreme points of Ξ as these can be exponentially many. Furthermore, if (1) involves both continuous and binary adaptive decisions, then the equality $S(\Xi_{\text{ext}}) = S(\Xi)$, does not necessarily hold. This is demonstrated in the following example.

Example 2 *Consider the following instance of Problem* (1)*:*

$$
\begin{aligned}
\textit{minimize} \quad & \tau \\
\textit{subject to} \quad & y(\cdot) \in \mathcal{B},\ \tau \in \mathbb{R}, \\
& \left.\begin{array}{r} y(\boldsymbol{\xi}) - \xi_2 \le \tau \\ y(\boldsymbol{\xi}) \ge \xi_2 \end{array}\right\} \quad \forall \boldsymbol{\xi} \in \Xi,
\end{aligned} \tag{30}
$$

where $\Xi = \left\{\xi \in \mathbb{R}^2 : \xi_1 = 1,\ -\frac{1}{2} \le \xi_2 \le \frac{1}{2}\right\}$*. It can be easily seen that the optimal value of* (30) *equals to* 1*, while the scenario counterpart involving the extreme scenarios* $(1, -\frac{1}{2})^\top$ *and* $(1, \frac{1}{2})^\top$ *results to the optimal value of* $\frac{1}{2}$*. Nevertheless, the optimal value of the scenario counterpart constitutes a lower bound on the true solution.*

The set of worst-case realizations Ξ_{w} for Problem (1), can be computed through the solution of the following problem:

$$
\begin{aligned}
\Xi_{\text{w}} := \arg\max_{\xi \in \Xi} \min_{x \in \mathbb{R}^n,\, y \in \{0,1\}^q} \; & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \\
\text{subject to} \quad & A\boldsymbol{x} + B\boldsymbol{y} \le C\boldsymbol{\xi}.
\end{aligned} \tag{31}
$$

By construction, any subset of Ξ_{w} is guaranteed to provide a tight lower bound, i.e., $S(\Xi_{\text{w}}) = S(\Xi)$. Of course, solving Problem (31) is at least as hard as solving Problem (1). As an alternative, Hadjiyiannis *et al.* [27] propose to use the critically binding scenarios $\widehat{\Xi}_{\mathrm{LDR}}$ resulting from linear decision rules, as a proxy to set Ξ_{w}. This choice is motivated by the hypothesis that there may exist at least one worst-case uncertainty realization for (1) that is also a worst-case realization for linear decision rule approximation. In any case, even if $\widehat{\Xi}_{\mathrm{LDR}} \cap \Xi_{\mathrm{w}} = \emptyset$, $\widehat{\Xi}_{\mathrm{LDR}}$ is still a finite subset of Ξ, and thus $S(\widehat{\Xi}_{\mathrm{LDR}})$ will be a valid lower bound on the optimal value of (1).

The flexibility of the proposed decision rules can be used as a tool for identifying a collection of critically binding scenarios that outperform the critically binding scenarios $\widehat{\Xi}_{\mathrm{LDR}}$ resulting from linear decision rules. From Theorems 1 and 2 we know that the functional spaces $\mathcal{PC}$ and $\mathcal{PB}$ coincides with the space of all piecewise linear continuous and binary functions, respectively. Therefore, solving Problem (29) parameterized with the binding scenarios from problem (10), can lead to tight overestimates for the optimal value of (1). A nice property of algorithms discussed in Section 2.3, is that at termination they provide as a byproduct the set of scenarios $\widehat{\Xi}^*$ for which the optimal decision rules are binding. These scenarios can directly be used in the scenario counterpart of (1), with the following relations being valid at optimality:

$$S(\widehat{\Xi}^*) \;\leq\; S(\Xi) = Z(\Xi) \;\leq\; Z(\widehat{\Xi}^*).$$

We emphasize that even if the decision rules are optimal for Problem (1), there is no guarantee that $S(\widehat{\Xi}^*) = S(\Xi)$. This is the case as there is no mechanism for selecting binding scenarios in $\widehat{\Xi}^*$ that are also elements in Ξ_{w}, see [27] for more details. Nevertheless, in Section 5, we show that this approach produces good results in conjunction with the proposed functional approximations $\mathcal{PC}(P)$ and $\mathcal{PB}(P)$.

4 Multistage Adaptive Optimization

In this section, we extend the methodology presented in Sections 2 and 3 to cover multistage adaptive optimization problems. This adaptive decision process can be described as follows: A decision maker first observes an uncertain parameter $\boldsymbol{\xi}_1 \in \mathbb{R}^{k_1}$ and then takes the pair of real-valued decision $\boldsymbol{x}_1(\boldsymbol{\xi}_1) \in \mathbb{R}^{n_1}$ and a binary decision $\boldsymbol{y}_1(\boldsymbol{\xi}_1) \in \{0,1\}^{q_1}$. Subsequently, a second uncertain parameter $\boldsymbol{\xi}_2 \in \mathbb{R}^{k_2}$ is revealed, in response to which the decision maker takes the a second pair of decision $\boldsymbol{x}_2(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) \in \mathbb{R}^{n_2}$ and $\boldsymbol{y}_2(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) \in \{0,1\}^{q_2}$. This sequence of alternating observations and decisions extends over T time stages. To enforce the non-anticipative structure of the decisions, a decision taken at stage t can only depend on the observed parameters up to and including stage t, i.e., $\boldsymbol{x}_t(\boldsymbol{\xi}^t)$ and $\boldsymbol{y}_t(\boldsymbol{\xi}^t)$ where $\boldsymbol{\xi}^t = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t)^\top \in \mathbb{R}^{k^t}$, with $k^t = \sum_{s=1}^{t} k_s$. For consistency with the previous sections, and with slight abuse of notation, we assume that $k_1 = 1$ and $\boldsymbol{\xi}_1 = 1$. As before, setting $\boldsymbol{\xi}_1 = 1$ is a non-restrictive assumption allows to represent affine functions of the non-degenerate outcomes $(\boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_t)^\top$ in a compact manner as linear functions of $(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t)^\top$. We denote by $\boldsymbol{\xi} = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T)^\top \in \mathbb{R}^k$ the vector of all uncertain parameters, where $k = k^T$. Finally, we denote by $\mathcal{R}_t$ the space of all real-valued functions from $\mathbb{R}^{k^t}$ to $\mathbb{R}$ and $\mathcal{B}_t$ the space of binary functions from $\mathbb{R}^{k^t}$ to $\{0, 1\}$.

Given matrices $A_{ts} \in \mathbb{R}^{m_t \times n_s}$, $B_{ts} \in \mathbb{R}^{m_t \times q_s}$ and $C_t \in \mathbb{R}^{m_t \times k^t}$, vectors $\boldsymbol{c}_t \in \mathbb{R}^{n_t}$, $\boldsymbol{d}_t \in \mathbb{R}^{q_t}$ and uncertainty set Ξ given by (2), we are interested to choose function $\boldsymbol{x}_t(\cdot) = (x_{1,t}(\cdot), \ldots, x_{n_t,t}(\cdot))^\top$, $x_{it} \in \mathcal{R}_t$ and $\boldsymbol{y}_t(\cdot) = (y_{1,t}(\cdot), \ldots, y_{q_t,t}(\cdot))^\top$, $y_{it} \in \mathcal{B}_t$ for all $t = 1, \ldots, T$ in order to solve:

$$\begin{array}{lll} \text{minimize} & \displaystyle \max_{\xi \in \Xi} \left(\sum_{t=1}^{T} \boldsymbol{c}_t^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t) + \boldsymbol{d}^\top \boldsymbol{y}_t(\boldsymbol{\xi}^t) \right) & \\ \text{subject to} & \left.\begin{array}{l} \displaystyle \sum_{s=1}^{t} A_{ts} \boldsymbol{x}_s(\boldsymbol{\xi}^s) + B_{ts} \boldsymbol{y}_s(\boldsymbol{\xi}^s) \le C_t \boldsymbol{\xi}^t \quad \forall \boldsymbol{\xi} \in \Xi, \\ x_{i,t}(\cdot) \in \mathcal{R}_t,\ i = 1, \ldots, n_t,\ y_{i,t}(\cdot) \in \mathcal{B}_t,\ i = 1, \ldots, q_t \end{array}\right\} & \forall t = 1, \ldots, T. \end{array} \tag{32}$$

Problem (32) is computationally challenging. To make the problem amenable to numerical solutions, we make similar restrictions to the function space of real-valued and binary recourse decisions as in Section 2. For real-valued decision rules chosen at stage t, let $\mathcal{PC}_t(P)$ be the space of piecewise linear continuous functions from $\mathbb{R}^{k^t}$ to $\mathbb{R}$, that have $P \in \mathbb{N}_+$ linear pieces, and admit the following structure,

$$x(\boldsymbol{\xi}^t) = \max\{\overline{\mathbf{x}}_1^\top \boldsymbol{\xi}^t, \ldots, \overline{\mathbf{x}}_P^\top \boldsymbol{\xi}^t\} - \max\{\underline{\mathbf{x}}_1^\top \boldsymbol{\xi}^t, \ldots, \underline{\mathbf{x}}_P^\top \boldsymbol{\xi}^t\},$$

for some $\overline{\mathbf{x}}_p, \underline{\mathbf{x}}_p \in \mathbb{R}^{k^t}$, $p \in \mathcal{P} := \{1, \ldots, P\}$. The information set of the decision rule ensures the non-anticipativity nature of the decision by allowing dependance only on observed parameters up to stage t and not on future outcomes $\{\boldsymbol{\xi}_{t+1}, \ldots, \boldsymbol{\xi}_T\}$. Similarly, for binary decision rules chosen at stage t, fix $\epsilon > 0$ and let $\mathcal{PB}_t(P)$ be the space of piecewise linear binary functions from $\mathbb{R}^{k^t}$ to $\{0, 1\}$, that have $P \in \mathbb{N}_+$ linear pieces, and admit the following structure,

$$y_t(\boldsymbol{\xi}^t) = \begin{cases} 1, & \max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}^t, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}^t\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}^t, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}^t\} \le 0, \\ 0, & \max\{\overline{\mathbf{y}}_1^\top \boldsymbol{\xi}^t, \ldots, \overline{\mathbf{y}}_P^\top \boldsymbol{\xi}^t\} - \max\{\underline{\mathbf{y}}_1^\top \boldsymbol{\xi}^t, \ldots, \underline{\mathbf{y}}_P^\top \boldsymbol{\xi}^t\} \ge \epsilon, \end{cases}$$

for some $\overline{\mathbf{y}}_p, \underline{\mathbf{y}}_p \in \mathbb{R}^{k^t}$, $p \in \mathcal{P} = \{1, \ldots, P\}$.

Restricting the space of admissible solutions from $\mathcal{R}_t$ to $\mathcal{PC}_t(P)$ and from $\mathcal{B}_t$ to $\mathcal{PB}_t(P)$, yields the following semi-infinite problem which for fixed number of linear pieces P, involves only a finite number of decision variables, and an infinite number of constraints.

$$
\begin{array}{ll}
\underset{\tau\in\mathbb{R}}{\text{minimize}} & \tau \\
\text{subject to} & \left.\begin{array}{l}
\displaystyle\sum_{t=1}^{T} \boldsymbol{c}_t^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t) + \boldsymbol{d}^\top \boldsymbol{y}_t(\boldsymbol{\xi}^t) \le \tau, \;\; \forall \boldsymbol{\xi} \in \Xi, \\
\displaystyle\sum_{s=1}^{t} A_{ts}\boldsymbol{x}_s(\boldsymbol{\xi}^s) + B_{ts}\boldsymbol{y}_s(\boldsymbol{\xi}^s) \le C_t\boldsymbol{\xi}^t, \;\; \forall \boldsymbol{\xi} \in \Xi, \\
x_{i,t}(\cdot) \in \mathcal{PC}_t(P),\; i = 1,\dots,n_t, \\
y_{i,t}(\cdot) \in \mathcal{PB}_t(P),\; i = 1,\dots,q_t,
\end{array}\right\} \forall t = 1,\dots,T.
\end{array}
\tag{33}
$$

Problem (33) has a similar structure as (10). Therefore, one can directly use the solution method presented in Sections 2.3 and 2.4, for finding the optimal solution to Problem (33).

Similar to Section 3, a lower bound for the optimal solution of Problem (32) can be achieved by solving the following scenario tree problem for any $\widehat{\Xi} \subset \Xi$.

$$
\begin{array}{ll}
\text{minimize} & \displaystyle\max_{\xi\in\widehat{\Xi}} \left(\sum_{t=1}^{T} \boldsymbol{c}_t^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t) + \boldsymbol{d}^\top \boldsymbol{y}_t(\boldsymbol{\xi}^t) \right) \\
\text{subject to} & \left.\begin{array}{l}
\displaystyle\sum_{s=1}^{t} A_{ts}\boldsymbol{x}_s(\boldsymbol{\xi}^s) + B_{ts}\boldsymbol{y}_s(\boldsymbol{\xi}^s) \le C_t\boldsymbol{\xi}^t, \\
x_{i,t}(\boldsymbol{\xi}^t) \in \mathbb{R},\; i = 1,\dots,n_t,\; y_{i,t}(\boldsymbol{\xi}^t) \in \{0,1\},\; i = 1,\dots,q_t
\end{array}\right\} \forall \boldsymbol{\xi} \in \widehat{\Xi},\; \forall t = 1,\dots,T, \\
& \left.\begin{array}{l}
\text{if } \boldsymbol{\xi}_j = \boldsymbol{\xi}_k \text{ then } x_{i,t}(\boldsymbol{\xi}_j^t) = x_{i,t}(\boldsymbol{\xi}_k^t),\; i = 1,\dots,n,\; t = 1,\dots,T \\
\text{if } \boldsymbol{\xi}_j = \boldsymbol{\xi}_k \text{ then } y_{i,t}(\boldsymbol{\xi}_j^t) = y_{i,t}(\boldsymbol{\xi}_k^t),\; i = 1,\dots,q,\; t = 1,\dots,T
\end{array}\right\} \forall \boldsymbol{\xi}_j, \boldsymbol{\xi}_k \in \widehat{\Xi}, j \neq k,
\end{array}
\tag{34}
$$

Here, scenarios $\boldsymbol{\xi}_j, \boldsymbol{\xi}_k$ are distinct elements of the finite set $\widehat{\Xi}$. The last two constraints in Problem (34) model the non-anticipative nature of the adaptive decisions $\boldsymbol{x}_t(\cdot)$ and $\boldsymbol{y}_t(\cdot)$. As before, we propose to use the critically binding scenarios $\widehat{\Xi}^*$ associated with Problem (33), to produce lower bounds on the optimal solution of (32). For an introduction to scenario tree problems, we refer to Birge and Louveaux [16].

We end this section by pooling together the central insights of Section 4. If Problem (32) has a non-empty and bounded feasible region, and a polyhedral uncertainty set, then the cutting plane Algorithms 1, 2 or 3 can be applied to Problem (33). At each iteration of the algorithms, the constituent problems can be reformulated into mixed-integer linear optimization problems using (19), (20), (21) for the continuous decision rules and (26), (27), (28) for the binary decision rules. In particular, if one uses reformulations (19) and (26), then the multistage variant of Problem (11) has a total of $1+\sum_{t=1}^{T}[(3|\widehat{\Xi}|+2Pk^t)n_t + (3|\widehat{\Xi}| + 2Pk^t)q_t]$ continuous variables, $\sum_{t=1}^{T}[|\widehat{\Xi}|2Pn_t + |\widehat{\Xi}|(1 + 2P)q_t]$ binary variables and $1 + \sum_{t=1}^{T}[m_t + |\widehat{\Xi}|(3 + 4P)n_t + |\widehat{\Xi}|(5 + 4P)q_t]$ constraints, where $|\widehat{\Xi}|$ indicates the cardinality of $\widehat{\Xi}$. Similarly, each of the multistage variants of Problems (12) has a total of $k + \sum_{t=1}^{T}[3n_t + 3q_t]$ continuous variables, $\sum_{t=1}^{T}[2Pn_t + (1+2P)q_t]$ binary variables and $l + \sum_{t=1}^{T}(3+4P)n_t + (5+3P)q_t$ constraints. The number of decision variables and constraints is of similar order if reformulations (20), (21), (27), (28) are used.

5 Computational Results

In this section, we apply the proposed decision rules to two variants of inventory control problems, and benchmark them against decision rules that appear in the literature. We solve the proposed decision rule problems using Algorithm 3, see Section 2.3. In particular, we use constants $\delta = 0.005$, $\beta = 10^{-4}$, $\widehat{\epsilon} = 0.005$ and $\widehat{\beta} = 10^{-4}$. Moreover, we use the special ordered sets formulation (20) for the reformulation of continuous decisions and the indicator constraint formulation (28) for the reformulation of binary decisions. All of our numerical results are carried out using the IBM ILOG CPLEX 12.5 optimization package on a Intel Core i7-2600 3.40GHz machine with 8GB RAM [30].

5.1 Multistage Inventory Control

In this case study, we consider a single item inventory control model that involves only continuous decisions. The problem can be described as follows. At the beginning of each time period $t \in \mathcal{T} := \{2, \dots, T\}$, the decision maker observes the product demand ξ_t. This demand can be served by either placing an order $x_t(\cdot)$, with unit cost c_x, which is delivered at the beginning of the next period, or by placing an order $z_t(\cdot)$, having a more expensive unit cost c_z, $c_x < c_z$, that is delivered immediately. If the ordered quantity is greater than the demand, the excess units are stored in a warehouse, incurring a unit holding cost c_h, and can be used to serve future demand. If there is a shortfall in the available quantity, then the orders are backlogged incurring a unit cost c_b. The level of available inventory at each period is given by $I_t(\cdot)$. In addition, the cumulative volume of advance orders $x_t(\cdot)$ must not exceed the ordering budget $\overline{x}_{\text{tot},t}$ at any time point t. The decision maker wishes to determine the ordering levels $x_t(\cdot)$ and $z_t(\cdot)$ that minimize the total ordering, backlogging and holding costs, associated with the worst- case demand realization over the planning horizon T. The problem can be formulated as the following multistage adaptive optimization problem.

$$
\begin{array}{ll}
\text{minimize} & \max_{\xi \in \Xi} \sum_{t \in \mathcal{T}} c_x\, x_{t-1}(\boldsymbol{\xi}^{t-1}) + c_z\, z_t(\boldsymbol{\xi}^t) + \max\left\{c_b\, I_t(\boldsymbol{\xi}^t), c_h\, I_t(\boldsymbol{\xi}^t)\right\} \\
\text{subject to} & x_{t-1}(\cdot) \in \mathcal{R}_{t-1},\; z_t(\cdot) \in \mathcal{R}_t,\; I_t(\cdot) \in \mathcal{R}_t \;\; \forall t \in \mathcal{T} \\
& \left.\begin{array}{l}
I_t(\boldsymbol{\xi}^t) = I_{t-1}(\boldsymbol{\xi}^{t-1}) + x_{t-1}(\boldsymbol{\xi}^{t-1}) + z_t(\boldsymbol{\xi}^t) - \xi_t \\
0 \le x_{t-1}(\boldsymbol{\xi}^{t-1}),\; 0 \le z_t(\boldsymbol{\xi}^t) \\
\sum_{s=1}^{t-1} x_s(\boldsymbol{\xi}^s) \le \overline{x}_{\text{tot},t}
\end{array}\right\} \forall t \in \mathcal{T},\, \forall \boldsymbol{\xi} \in \Xi.
\end{array}
\tag{35}
$$

The uncertainty set Ξ is given by,

$$
\Xi := \left\{ \boldsymbol{\xi} \in \mathbb{R}^k :\; \xi_1 = 1,\; l_i \le \xi_i \le u_i,\; i = 2, \ldots, k,\; \left\| \boldsymbol{\xi}_{-1} - \frac{\overline{\xi}_{\max}}{2} \boldsymbol{e} \right\|_1 \le \frac{\overline{\xi}_{\max}}{2} \right\}, \tag{36}
$$

where constant $\overline{\xi}_{\max}$ denote the maximum demand that can occur in each period, and $\boldsymbol{u}, \boldsymbol{l} \in \mathbb{R}^{k-1}$ denotes denotes the upper and lower bounds on the random quantities $\boldsymbol{\xi}_{-1}$, respectively. Here, $\boldsymbol{\xi}_{-1}$ denotes the subvector of the $k-1$ last components of $\boldsymbol{\xi} \in \mathbb{R}^k$.

To bring Problem (35) in the standard form (32), we provide a reformulation for the maximum appearing in the objective. Since all the maxima appear with a positive sign, they can be replaced by introducing auxiliary decision variables $o_t(\cdot) \in \mathcal{R}_t$ for all $t \in \mathcal{T}$, serving as over-estimators for the max functions. Therefore, the objective function can be replaced with

$$
\text{minimize} \;\; \max_{\xi \in \Xi} \sum_{t \in \mathcal{T}} c_x\, x_{t-1}(\boldsymbol{\xi}^{t-1}) + c_z\, z_t(\boldsymbol{\xi}^t) + o_t(\boldsymbol{\xi}^t),
$$

and the following set of linear constraints are added to Problem (35).

$$
\left.\begin{array}{l}
c_b\, I_t(\boldsymbol{\xi}^t) \le o_t(\boldsymbol{\xi}^t) \\
c_h\, I_t(\boldsymbol{\xi}^t) \le o_t(\boldsymbol{\xi}^t)
\end{array}\right\} \quad \forall t \in \mathcal{T},\, \forall \boldsymbol{\xi} \in \Xi.
$$

We emphasize that $o_t(\cdot)$ are adaptive decisions, and the quality of the reformulation will depend on the decision rules used. Exact reformulations for the summation of maxima that is independent of the decision rule approximation used, are discussed in [25].

For our computational experiments we randomly generated 25 instances of Problem (35). The parameters are randomly chosen using a uniform distribution from the following sets: Advanced and instant ordering costs are chosen from $c_x \in [0, 5]$ and $c_z \in [0, 10]$, respectively, such that $c_x < c_z$. Backlogging and holding costs are elements of $c_b \in [-10, 0]$ and $c_h \in [0, 5]$, respectively. The maximum

	Global optimality			1% optimality			5% optimality		
T	$\mathcal{PC}_t(2)$	LDR	Time	$\mathcal{PC}_t(2)$	LDR	Time	$\mathcal{PC}_t(2)$	LDR	Time
5	2.9%	28.1%	2.7sec	4.8%	27.0%	0.6sec	4.6%	28.3%	0.5sec
10	2.5%	34.1%	3.9sec	3.8%	35.2%	3.3sec	4.7%	34.4%	3.0sec
15	1.2%	23.6%	46.8sec	1.3%	23.5%	95.7sec	3.1%	23.4%	9.6sec
20	1.3%	28.1%	2681.1sec	2.1%	33.4%	2083.7sec	4.7%	26.2%	607.9sec
25	0.4%	38.7%	3489.1sec	0.5%	45.3%	695.1sec	2.3%	39.7%	3763.1sec
30	1.6%	52.5%	36703.8sec	3.1%	57.1%	7883.7sec	4.2%	43.9%	4974.6sec

Table 1: *Average optimality gaps from 25 randomly chosen instances for the linear decision rules (LDR), piecewise linear decision rules $x(\xi^t) = x^\top L(\xi^t)$ discussed in [5, 23] and the proposed piecewise linear decision rules $\mathcal{PC}_t(2)$ with two linear pieces. The piecewise linear decisions rule $x(\xi^t) = x^\top L(\xi^t)$, perform the same as the linear decision rules due to the structure of $L(\cdot)$ and uncertainty set (36), see [23, Section 4.1] for further discussion. The time presented corresponds to the average time taken to solve the optimization problems for the proposed approximation. All problems with linear decision rules and piecewise linear decisions rule $x(\xi^t) = x^\top L(\xi^t)$ were solved within 10 seconds.*

demand parameter is set to $\overline{\xi}_{\max} = 100$, and the lower and upper bounds for each random parameter are chosen from $l_i \in [0, 25]$ and $u_i \in [75, 100]$ for $i = 2, \dots, k$. The cumulative ordering budget equals to $\overline{x}_{\text{tot},t} = \sum_{s=1}^{t} \overline{x}_s$, for $t = 2, \dots, T$, with $\overline{x}_t \in [0, 100]$. We also assume that the initial inventory level equals to zero, i.e., $I_1 = 0$.

In the first test series we compare the performance of different decision rules, on randomly generated instances of Problem (35) with planning horizons up to 30 time periods. Note that for a horizon of T, Problem (35) has a total of $4T$ real-valued adaptive decisions, $6T$ semi-infinite constraints and $T-1$ uncertain parameters. For all adaptive decisions in the problem, we consider the following decision rules:

1. Linear decision rules;
2. Piecewise linear decisions expressed in the form $x(\boldsymbol{\xi}^t) = \boldsymbol{x}^\top L(\boldsymbol{\xi}^t)$, where the non-linear operator L creates 4 piecewise linear components for each $\xi_i, i = 2, \dots, k$, see [23, Section 4.1] for more details;
3. The proposed piecewise linear decision rules with 2 linear pieces $\mathcal{PC}_t(2)$.

The approximations are benchmarked against the lower bounding, scenario tree problem (34) constructed using the binding scenarios from the proposed decision rules. Table 1 presents the results from this test series in terms of the average relative gap between the optimal value of the decision rule problem and the optimal value of the lower bounding problem. The relative gap is defined as quantity $\frac{(\text{ub}-\text{lb})}{2(\text{ub}+\text{lb})}$, where ub and lb are the corresponding objective values for the upper and lower bounding problems. Moreover, we present the performance of the approximation when the termination criteria for the mixed-integer linear optimization problem in **Step 2** of the algorithms is set to global optimality, 1% and 5% optimality gap. We emphasize that for linear and piecewise linear decisions expressed in the form $x(\boldsymbol{\xi}^t) = \boldsymbol{x}^\top L(\boldsymbol{\xi}^t)$, the corresponding decision rule problems are solved to global optimality.

An interesting first observation is that the linear and piecewise linear decision rules $x(\boldsymbol{\xi}^t) = \boldsymbol{x}^\top L(\boldsymbol{\xi}^t)$ perform exactly the same. This is an artifact of the additional approximation needed in order to reformulate the problem as a linear optimization problem, see Georghiou *et al.* [23, Section 4] for an extensive discussion. Nevertheless, the optimal solution of these decision rules is no worse than that of the linear decision rules. Indeed, for an arbitrary non-linear operator $L(\cdot)$, and uncertainty set Ξ, producing the exact reformulation for the decision rule problem into a linear optimization problem is NP-hard [23, Lemma 4.4]. We emphasize that the proposed piecewise linear decision rules $\mathcal{PC}_t(P)$ do not require this additional approximation, thus achieving the *best piecewise linear decision rule* within the functional space $\mathcal{PC}_t(P)$.

In general, the linear decision rules produce suboptimal solutions. The average relative gap is more than 20% and can reach 60%. On the other hand, the proposed approach always produces near optimal solutions with the average gap being below the 5% threshold. It is interesting to notice that the solution quality of the proposed approximation does not deteriorate excessively as the termination criteria for the mixed-integer linear optimization problem in **Step 2** is relaxed. Nevertheless, the computational time for these problems is significantly improved.

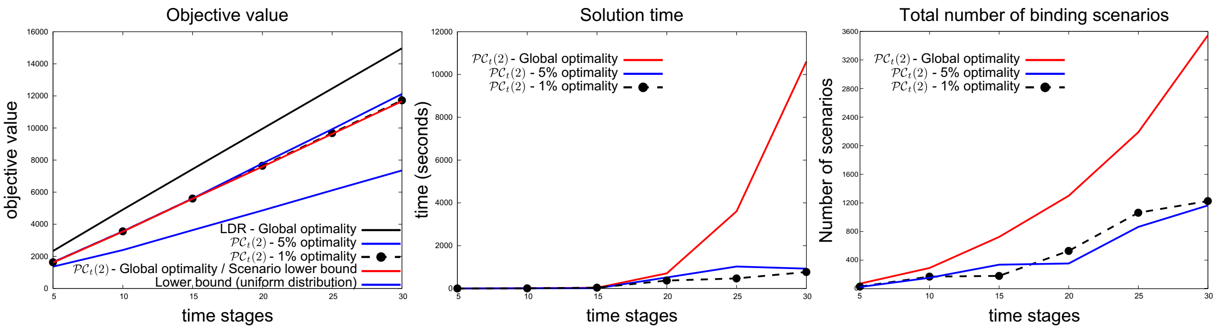

Figure 1: *Performance of decision rule approximations in the problem instance with* $c_x = 5$*,* $c_z = 10$*,* $c_b = -10$*,* $c_h = 5$*,* $l_i = \mathbf{0}$*,* $u_i = 100$*,* $i = 2, \ldots, k$*, and* $\overline{x}_t = 50$*. The left diagram illustrates the performance of the different decision rule approximations considered for Problem* (35)*. The diagram also includes the lower bound proposed in [23, 31] using uniform distribution. The relative performance of the approximations in run times is reported in the center diagram, while the diagram in the right reports the total number of binding scenarios required by the cutting plane algorithm to converge.*

To better understand the behaviour of the proposed approach, Figure 1 presents the results of a single problem instance. For this example we also benchmark the proposed approximation against the lower bound proposed in [23, 31] using uniform distribution. In this particular example, the objective value of the proposed decision rule problem exactly matches the objective value of the lower bounding, scenario tree problem, verifying that it achieves the global optimal solution. Relaxing the termination criteria for the mixed-integer linear optimization problems also produces near optimal solutions, with the relative gap being always below 4%. It is interesting to see that the total number of samples needed for the cutting plane algorithm to converge decreases as the termination criterion is relaxed, Figure 1 (right). This has also a great impact on the solution time as the total number of binary variables needed for the reformulation of Problem (11) grow linearly with respect to the number of binding scenarios.

It is also interesting to see the behavior of the lower bound proposed in [31] which vastly overestimates the true optimal value of the problem. This is a result of the unsuitable choice of the distribution used, a drawback that is not shared by the lower bounding, scenario tree problem (34), see [27] for more details.

Our key conclusions from this study are:

1. The proposed piecewise linear continuous decision rules are practically scalable and provide provable near optimal solutions;

2. Relaxing the termination criteria optimality tolerance for the mixed-integer optimization problems, can boost the scalability of the proposed method;

3. Utilizing the scenario based lower bounding technique in conjunction with the proposed approach can provide tight lower bounds to the global optimal solution of the problem in hand, producing good estimates for the loss of optimality.

5.2 Multistage Lot Sizing

In the second case study, we consider a variant of the inventory control problem presented in the previous section that includes both continuous and binary decisions. This problem features two main differences: (i) The continuous ordering quantity z_t is replaced with N binary ordering decisions y_{nt}, $n = 1\ldots, N$ that deliver immediately fixed quantities q_n, $n = 1\ldots, N$ and incur unit cost c_{y_n}, $n = 1\ldots, N$, respectively; (ii) Unlike the previous problem where unsatisfied demand could be backlogged, in this problem the decision maker needs to robustly satisfy all uncertainty demand ξ_t that arise at every period. As before, the decision maker aims to determine ordering levels that minimize the total ordering and holding costs associated with the worst- case demand realization over the planning horizon T. The problem can be formulated as the following multistage adaptive optimization problem.

$$\begin{array}{ll}
\text{minimize} & \displaystyle\max_{\xi\in\Xi} \sum_{t\in\mathcal{T}} c_x\, x_{t-1}(\boldsymbol{\xi}^{t-1}) + c_b\, I_t(\boldsymbol{\xi}^t) + \sum_{n=1}^{N} c_{y_n} q_n\, y_{nt}(\boldsymbol{\xi}^t) \\
\text{subject to} & x_{t-1}(\cdot) \in \mathcal{R}_{t-1},\, I_t(\cdot) \in \mathcal{R}_t,\, y_{nt}(\cdot), \in \mathcal{B}_t \;\; \forall n = 1,\ldots,N,\, \forall t \in \mathcal{T} \\
& \left.\begin{array}{l}
I_t(\boldsymbol{\xi}^t) = I_{t-1}(\boldsymbol{\xi}^{t-1}) + x_{t-1}(\boldsymbol{\xi}^{t-1}) + \displaystyle\sum_{n=1}^{N} q_n\, y_{nt}(\boldsymbol{\xi}^t) - \xi_t \\
0 \le x_{t-1}(\boldsymbol{\xi}^{t-1}),\, 0 \le I_t(\boldsymbol{\xi}^t) \\
\displaystyle\sum_{s=1}^{t-1} x_s(\boldsymbol{\xi}^s) \le \overline{x}_{\mathrm{tot},t}
\end{array}\right\} \forall t \in \mathcal{T},\, \forall \boldsymbol{\xi} \in \Xi.
\end{array} \tag{37}$$

Binary ordering decisions per stage: $N = 2$

T	1% optimality $\mathcal{PB}_t(1)$	Non-adaptive	Time	5% optimality $\mathcal{PB}_t(1)$	Non-adaptive	Time
2	0.0%	17.6%	0.1sec	0.6%	17.6%	0.4sec
4	24.2%	68.6%	50.6sec	27.3%	68.6%	45.5sec
6	37.4%	62.0%	4833.8sec	38.9%	62.1%	956.8sec
8	37.9%	84.4%	27531.1sec	38.0%	84.4%	19573.1sec
10	39.7%	89.9%	35716.6sec	42.0%	89.9%	31464.1sec

Binary ordering decisions per stage: $N = 3$

T	1% optimality $\mathcal{PB}_t(1)$	Non-adaptive	Time	5% optimality $\mathcal{PB}_t(1)$	Non-adaptive	Time
2	0.0%	27.6%	0.1sec	1.2%	27.6%	0.1sec
4	17.2%	73.3%	3381.8sec	23.9%	73.3%	781.6sec
6	34.5%	66.2%	9181.0sec	38.4%	66.1%	3298.1sec
8	37.6%	83.4%	28742.7sec	38.1%	83.7%	21885.5sec
10	-	89.7%	-	41.1%	90.7%	39141.5sec

Table 2: *Average optimality gaps from 25 randomly chosen instances. The lower bounding problems are formulated using the binding scenarios from the corresponding adaptive $\mathcal{PB}(1)$ and non-adaptive problems. The solution time presented correspond to the average time taken to solve the optimization problems for the adaptive binary decision rules $\mathcal{PB}(1)$. All problems involving non-adaptive decisions were solved within 15 seconds.*

The uncertainty set Ξ is given by,

$$\Xi := \left\{ \boldsymbol{\xi} \in \mathbb{R}^k : \ \xi_1 = 1, \, l_i \le \xi_i \le u_i, \, i = 2, \dots, k \right\}.$$

For our computational experiments we randomly generate 25 instances of Problem (37). The parameters are randomly chosen using a uniform distribution from the following sets: Advanced and instant ordering costs are chose from $c_x \in [0,5]$ and $c_{y_n} \in [0,10]$ for all $n = 1, \dots, N$, respectively, such that $c_x < c_{y_n}$. Holding costs are elements of $c_h \in [0,10]$ with the fixed ordering quantities set to $q_n = 100/N$ for all $n = 1, \dots, N$. As before the cumulative ordering budget equals to $\overline{x}_{\mathrm{tot},t} = \sum_{s=1}^{t} \overline{x}_s$, for $t = 2, \dots, T$, with $\overline{x}_t \in [0,100]$ and the lower and upper bounds for each random parameter are chosen from $l_i \in [0,25]$ and $u_i \in [75,100]$, $i = 2, \dots, k$. We also assume that the initial inventory level equals to zero.

In the first test series, we compare the performance of adaptive versus non-adaptive binary decisions on instances of Problem (37) with planning horizons up to 10 periods. Note that for a horizon of T, Problem (37) has a total of $2T$ real-valued adaptive decisions, NT binary adaptive decisions, $4T$ semi-infinite constraints and $T-1$ uncertain parameters. To this end, we apply linear decision rules to continuous ordering decisions, i.e., $x_t \in \mathcal{PC}_t(1)$ for all $t \in \mathcal{T}$, and compare the performance of the piecewise constant binary decision rules with one piece, $y_{nt} \in \mathcal{PB}_t(1)$, for all $n = 1, \dots, N$, versus the non-adaptive binary decision, $y_{nt} \in \{0,1\}$, for all $n = 1, \dots N$. In addition to the functional approximation $y_{nt} \in \mathcal{PB}_t(1)$, we restrict the information available to the binary decision rules y_{nt} to be only on the random parameter at stage t, i.e., $\boldsymbol{\xi}^t = \xi_t$ and $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_t)$. As we will demonstrate later, this secondary approximation boosts the computational tractability of the approximation. Table 2 presents the results from this test series in terms of the average relative gaps. We emphasize that the lower bounding problems (34) are formulated using the binding scenarios from the corresponding adaptive $\mathcal{PB}(1)$ and non-adaptive problems. From the results, it is evident that the adaptive $\mathcal{PB}(1)$ binary decision rule problem achieves a much better performance than the non-adaptive problem both in the $N = 2$ and $N = 3$ case. The poor relative gaps achieved in the non-adaptive problem are partly attributed to the poor performance of the lower bounding problem, see Figure 2 (left). Since the binding scenarios of the non-adaptive problem do not correspond to the worst-case realization of the adaptive problem (37), the lower bounding problem (34) significantly overestimates the true optimal value of Problem (37).

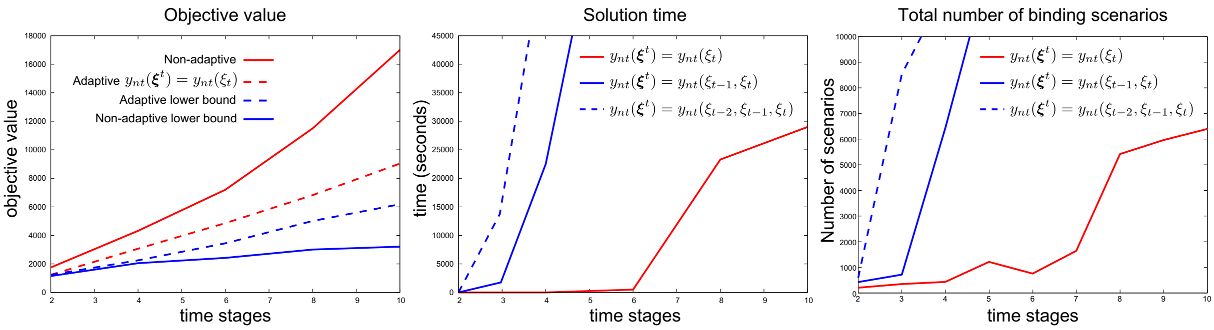

Figure 2: *The left diagram illustrates the performance of the adaptive $\mathcal{PB}(1)$ versus non-adaptive binary decisions together with their corresponding lower bounding problems. The center diagram illustrate the performance of the binary decisions in terms of runtime when we restrict the information available to the decisions, while the diagram in the right reports the total number of binding scenarios required by the cutting plane algorithm to converge. Both tests are run on an instance of* (37) *with* $N = 2$, $c_x = 5$, $c_{y_n} = 10$ *for* $n = 1, 2$, $c_h = 10$, $l_i = 0$, $u_i = 100$, $i = 2, \ldots, k$ *and* $\overline{x}_t = 50$.

In the second test series, we compare the performance of the piecewise linear binary decision rules in terms of run times, when the information available to the decision rule is restricted. The three restrictions are $(i)\, \boldsymbol{\xi}^t = \xi_t$, i.e., $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_t)$, $(ii)\, \boldsymbol{\xi}^t = (\xi_{t-1}, \xi_t)^\top$, i.e., $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_{t-1}, \xi_t)$ and $(iii)\, \boldsymbol{\xi}^t = (\xi_{t-2}, \xi_{t-1}, \xi_t)^\top$, i.e., $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_{t-2}, \xi_{t-1}, \xi_t)$. The results are presented in Figure 2 (center). Our results show that all three restrictions produce the same solution for Problem (37). Nevertheless, restriction $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_t)$ achieves the fastest solution runtime. On the other hand, it is only possible to solve problems with $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_{t-1}, \xi_t)$ and $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_{t-2}, \xi_{t-1}, \xi_t)$ within a reasonable amount of time for problem instances of $T = 4$ and $T = 3$, respectively. This is directly related to the total number of binding scenarios needed for the cutting plane algorithm to converge. From Figure 2 (right), we see that these numbers can be quite large, rendering Problem (11) computationally intractable. We note that for the given input parameters $\delta = 0.005$, $\beta = 10^{-4}$, the worst- case number of scenarios needed to solve the problem with $y_{nt}(\boldsymbol{\xi}^t) = y_{nt}(\xi_{t-2}, \xi_{t-1}, \xi_t)$ and $T = 4$, calculated using the multistage version of bound (14), is close to 9 million.

Our key conclusions from this study are:

1. Although the results are not as strong as in Section 5.1, the proposed binary decision rules out-putperform the simple, non-adaptive binary decisions, and produce good quality results.

2. By controlling the information available to the decision rule, one is able to solve large instances of the problem in hand.

3. We demonstrate that the quality of the scenario-based lower bounding problem (34) benefits from the proposed approach, giving an efficient indication for the loss of optimality.

6 Conclusions

In this paper we present a framework to address the design of decision rules involving both continuous and binary variables as piecewise linear and piecewise constant functions, respectively. The methodology presented addresses the shortcoming of non-linear decision rules appearing in the literature which requires the structure of the decision rules being decided a priori. Instead, we first derive the structure for the optimal decision rules involving both continuous and binary variables as piecewise linear and piecewise constant functions, respectively, and present a solution method that designs the structure of the decision rules endogenously through a sequence of mixed-integer optimization problems. The effectiveness of the proposed methods is demonstrated in the context of two multistage inventory control problems, where we show that our approach is (i) practically tractable and (ii) provides high quality solutions that outperform alternative methods.

References

[1] B.D.O. Anderson and J.B. Moore. *Optimal Control: Linear Quadratic Methods.* Prentice Hall, 1990.

[2] D. Anthony and N. Biggs. *Computational learning theory.* Cambridge University Press, 1992.

[3] D. Bampou and D. Kuhn. Scenario-free stochastic programming with polynomial decision rules. *IEEE Conference on Decision and Control and European Control Conference, Orlando, USA*, December 2011.

[4] 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.

[5] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization.* Princeton University Press, 2009.

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

[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):1–14, 1999.

[9] D. Bertsimas and H. Bidkhori. On the performance of affine policies for two-stage adaptive optimization: a geometric perspective. *Submitted for publication*, 2013.

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

[11] D. Bertsimas and V. Goyal. On the power and limitations of affine policies in two-stage adaptive optimization. *Mathematical Programming*, 134(2):491–531, 2012.

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

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

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

[15] D. Bertsimas and R. Weismantel. *Optimization over integers*. Dynamic Ideas Belmont, 2005.

[16] J.R. Birge and F. Louveaux. *Introduction to Stochastic Programming.* Springer-Verlag, New York, 1997.

[17] J.W. Blankenship and J.E. Falk. Infinitely constrained optimization problems. *Journal of Optimization Theory and Applications*, 19(2):261–281, 1976.

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

[19] G. Calafiore and F. Dabbene. A probabilistic framework for problems with real structured uncertainty in systems and control. *Automatica*, 38(8):1265–1276, 2002.

[20] C. Caramanis. *Adaptable optimization: theory and algorithms*. PhD thesis, Massachusetts Institute of Technology, 2006.

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

[22] D. de Farias and B. Van-Roy. On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research*, 29(3):462–478, 2004.

[23] A. Georghiou, W. Wiesemann, and D. Kuhn. Generalized decision rule approximations for stochastic programming via liftings. *Submitted for publication*, 2010.

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

[25] 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.

[26] C. E. Gounaris, W. Wiesemann, and C. A. Floudas. The robust capacitated vehicle routing problem under demand uncertainty. *Operations Research*, 2013. To appear.

[27] M.J. Hadjiyiannis, P.J. Goulart, and D. Kuhn. A scenario approach for estimating the suboptimality of linear decision rules in two-stage robust optimization. *IEEE Conference on Decision and Control and European Control Conference, Orlando, USA*, December 2011.

[28] A. Hempel, P.J. Goulart, and J. Lygeros. Every Continuous Piecewise Affine Function Can Be Obtained by Solving a Parametric Linear Program. In *European Control Conference*, Zurich, Switzerland, July 2013.

[29] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[30] IBM ILOG CPLEX. http://www.ibm.com/software/integration/optimization/ cplex-optimizer.

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

[32] SCIP. http://scip.zib.de/scip.shtml.

[33] A. Shapiro and A. Nemirovski. On complexity of stochastic programming problems. *Applied Optimization*, 99(1):111–146, 2005.

[34] M Vidyasagar. Randomized algorithms for robust controller synthesis using statistical learning theory. *Automatica*, 37(10):1515–1528, 2001.