Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Improved Decision Rule Approximations for Multistage Robust Optimization via Copositive Programming

Guanglin Xu, Grani A. Hanasusanto

informs
https://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Articles in Advance, pp. 1–20
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Methods

Improved Decision Rule Approximations for Multistage Robust Optimization via Copositive Programming

Guanglin Xu,[a,b] **Grani A. Hanasusanto**[c,*]

[a] Department of Systems Engineering and Engineering Management, University of North Carolina at Charlotte, Charlotte, North Carolina 28270; [b] Institute for Mathematics and Its Applications, University of Minnesota, Minneapolis, Minnesota 55455; [c] Department of Industrial and Enterprise Systems Engineering, University of Illinois Urbana–Champaign, Urbana, Illinois 61801

*Corresponding author

Contact: gxu7@uncc.edu, https://orcid.org/0000-0002-5786-4913 (GX); gah@illinois.edu, https://orcid.org/0000-0003-4900-2958 (GAH)

Received: September 8, 2018
Revised: April 28, 2021
Accepted: June 12, 2023
Published Online in Articles in Advance: September 21, 2023

Area of Review: Optimization

https://doi.org/10.1287/opre.2018.0505

Abstract. We study decision rule approximations for generic multistage robust linear optimization problems. We examine linear decision rules for the case when the objective coefficients, the recourse matrices, and the right-hand sides are uncertain, and we explore quadratic decision rules for the case when only the right-hand sides are uncertain. The resulting optimization problems are NP hard but amenable to copositive programming reformulations that give rise to tight, tractable semidefinite programming solution approaches. We further enhance these approximations through new piecewise decision rule schemes. Finally, we prove that our proposed approximations are tighter than the state-of-the-art schemes and demonstrate their superiority through numerical experiments.

Funding: G. A. Hanasusanto was supported by the National Science Foundation [Grants 1752125 and 2153606].

Supplemental Material: The e-companion is available at https://doi.org/10.1287/opre.2018.0505.

Keywords: multistage robust optimization • decision rules • piecewise decision rules • conservative approximations • copositive programming • semidefinite programming

1. Introduction

Decision making under uncertainty arises in a wide spectrum of applications in operations management, engineering, finance, and process control. A prominent modeling approach for decision making under uncertainty is *robust optimization* (RO), whereby one seeks for a decision that hedges against the worst-case realization of uncertain parameters; see Ben-Tal and Nemirovski (2002), Ben-Tal et al. (2009), and Bertsimas et al. (2011a). The RO paradigm is appealing because it leads to computationally tractable solution schemes for many *static* decision-making problems under uncertainty. However, real-life problems are often *dynamic* in nature, where the uncertain parameters are revealed sequentially and the decisions must be adapted to the current realizations. The adaptive decisions are fundamentally infinite dimensional as they constitute mappings from the space of uncertain parameters to the space of actions. This setting gives rise to the *multistage robust optimization* (MSRO) problems, which in general, are computationally challenging to solve. Only in a few cases and under very stringent conditions are the problems efficiently solvable; see, for instance, Ben-Tal and Nemirovski (1999), Guslitser (2002), and Bertsimas et al. (2015). Consequently, the design of solution schemes for MSRO necessitates reconciling the conflicting objectives of optimality and scalability.

Conservative approximations for MSRO can be derived by using *linear decision rules*, where we restrict the adaptive decisions to be affine functions in the uncertain parameters. Popularized by Ben-Tal et al. (2004), linear decision rules have found successful applications in various areas of decision-making problems under uncertainty (Ben-Tal et al. 2005; Atamtürk and Zhang 2007; Chen et al. 2007, 2008; Calafiore 2008; Rocha and Kuhn 2012; Gounaris et al. 2013) as they are simple yet valuable to implement in practice. Moreover, linear decision rules are optimal for some instances of MSRO (Bertsimas and Goyal 2012, Iancu et al. 2013), linear quadratic optimal control (Anderson and Moore 2007), and robust vehicle routing (Gounaris et al. 2013) problems. The resulting optimization problems, however, are tractable only under the restrictive setting of *fixed recourse* (i.e., when the adaptive decisions are not multiplied with the uncertain parameters in the problem's formulation). Many decision-making problems under uncertainty, such as portfolio optimization (Dantzig and Infanger 1993, Ben-Tal et al. 2000, Rocha and Kuhn 2012), energy systems operation planning (Martins da Silva Rocha 2013), inventory planning (Bertsimas and Georghiou 2018), etc., do not satisfy the fixed recourse assumption. For these problem instances, the linear decision rule approximation is NP hard already in a two-stage setting; see, for example, Guslitser (2002) and Ben-Tal et al. (2004).

The basic linear decision rules have been extended to truncated linear (See and Sim 2010), segregated linear (Chen et al. 2008, Chen and Zhang 2009, Goh and Sim 2010), and piecewise linear (Georghiou et al. 2015, Ben-Tal et al. 2020) functions in the uncertain parameters. If the MSRO problem has fixed recourse, then one can formally prove that the optimal adaptive decisions are piecewise linear (Bemporad et al. 2003), which justifies the use of these enhanced approximations. Unfortunately, optimizing for the best piecewise linear decision rule entails globally solving a nonconvex optimization problem, which is inherently difficult; see Bertsimas and Georghiou (2015) and Ben-Tal et al. (2020). If, in addition, some basic descriptions about the piecewise linear structure are prescribed, then one can derive tractable linear programming approximations for problem instances with fixed recourse by Georghiou et al. (2015). Their piecewise linear decision rule scheme is a generalization of the aforementioned methods, including the truncated linear decision rule by See and Sim (2010) and the segregated linear decision rule by Chen et al. (2008), Chen and Zhang (2009), and Goh and Sim (2010).

If a tighter approximation is desired or the problem has nonfixed recourse, then one can in principle develop a hierarchy of increasingly tight semidefinite approximations using *polynomial decision rules* (Bertsimas et al. 2011b). Although optimizing for the best polynomial decision rule of fixed degree is difficult, tractable conservative approximations can be obtained by employing the Lasserre hierarchy (Parrilo 2000, Lasserre 2009). Such approximations are attractive because they do not require prior structural knowledge about the optimal adaptive decisions. However, the resulting semidefinite programs scale poorly with the degree of the polynomial decision rules. A decent trade-off between suboptimality and scalability is attained in *quadratic decision rules* (QDRs), where one merely optimizes over polynomial functions of degree 2. Their semidefinite approximations, based on the well-known approximate S lemma (Ben-Tal et al. 2009), have been applied successfully to instances of inventory planning (Bertsimas et al. 2011b, Hanasusanto et al. 2015) and electricity capacity expansion (Bampou and Kuhn 2011) problems. A posteriori lower bounds to the MSRO problem can be derived by applying decision rules to the problem's dual formulation; see Bampou and Kuhn (2011), Kuhn et al. (2011), and Lasserre (2009). Alternative schemes that similarly provide aggressive bounds for MSRO are proposed in Hadjiyiannis et al. (2011) and Bertsimas and de Ruiter (2016). All the methods mentioned can be applied to different paradigms in optimization under uncertainty, such as stochastic programming, robust optimization, and distributionally robust optimization. Our paper focuses on the robust optimization setting because it requires minimal assumptions about the uncertainty, which allows us to present the main idea cleanly. If distributional information is available, then the proposed methods can be directly applied to the other settings in a relatively straightforward fashion.

Global optimization approaches have also been designed to derive exact solutions of MSRO problems. In the two-stage robust optimization setting, these methods include Benders' decomposition (Bertsimas et al. 2013, Hashemi Doulabi et al. 2021), column and constraint generation (Zeng and Zhao 2013), extreme point enumeration combined with decision rules (Georghiou et al. 2020), and Fourier–Motzkin elimination (Zhen et al. 2018). The Benders' decomposition scheme has been extended to the multistage setting for MSRO problems where the uncertain parameters exhibit a *stage-wise rectangular* structure (Georghiou et al. 2019). Bertsimas and Dunning (2016) and Postek and den Hertog (2016) develop adaptive uncertainty set partitioning schemes that generate a sequence of increasingly accurate conservative approximations for MSRO. A global optimization scheme has also been conceived through the lens of conic reformulations. Hanasusanto and Kuhn (2018) and Xu and Burer (2018) propose independently equivalent copositive programming reformulations for two-stage robust optimization problems and develop conservative semidefinite approximations for the reformulations.

Using *copositive programming* techniques, this paper takes a first step toward addressing a generic linear MSRO problem where the objective coefficients, the recourse matrix, and the right-hand sides are uncertain. A copositive program is a convex program that optimizes a linear function over the cone of copositive matrices subject to linear constraints; see Dür (2010), Bomze (2012), and Burer (2012). Bomze et al. (2000) are the first to reformulate an NP-hard problem, namely the standard quadratic optimization problem, to an equivalent copositive program. The seminal work of Burer (2009) shows that a generic quadratic program can be reformulated to an equivalent copositive program. In another work, Burer and Dong (2012) establish the equivalence between a nonconvex quadratically constrained quadratic program (QCQP) and a generalized copositive program under certain conditions. We refer the reader to Natarajan et al. (2011), Burer and Dong (2012), Chen and Burer (2012), Kong et al. (2013), and Natarajan and Teo (2017) for more works on using copositive techniques to reformulate nonconvex quadratic programs arising in different applications.

Our *key contribution* is to utilize copositive programming techniques to develop *stronger* decision rule approximations for *generic* MSRO problems. In the generic setting, the direct use of decision rules leads to computationally intractable semiinfinite programs, with finitely many decision variables but infinitely many constraints. The standard dualization procedure in robust optimization does not apply because these constraints involve nonconvex QCQPs. We leverage the copositive reformulation techniques to convexify the QCQPs, which enable the dualization of the constraints to arrive at finite-dimensional convex optimization problems. The copositive techniques further allow us to handle complex uncertainty sets (e.g., integrating complementary constraints), which lead to exact convex reformulations for a class of piecewise decision rule approximations. All these new reformulations enjoy tractable semidefinite approximations that are provably superior to the state-of-the-art schemes. We summarize the contributions of the paper as follows.

1. For the generic MSRO problems, we derive new copositive programming reformulations in view of the popular linear decision rules. For MSRO problems with fixed recourse, we derive new copositive programming reformulations in view of the more powerful quadratic decision rules. The exactness results are general. They hold for MSRO problems *without* relatively complete recourse and under very minimal assumption about the compactness of the uncertainty set *without* requiring it to exhibit stage-wise rectangularity.

2. The emerging copositive programs are amenable to a hierarchy of increasingly tight conservative semidefinite programming approximations. We formulate the simplest of these approximations and prove that it is tighter than the state-of-the-art scheme by Ben-Tal et al. (2004) and also, the polynomial decision rule scheme by Bertsimas et al. (2011b) when the degree of the polynomial is set to the degree of our decision rules (degree 1 for problems with nonfixed recourse and degree 2 for problems with fixed recourse). We demonstrate empirically that our proposed approximation is competitive to polynomial decision rules of higher degrees while displaying more favorable scalability.

3. We propose piecewise linear decision rules for MSRO problems with nonfixed recourse and piecewise quadratic decision rules for MSRO problems with fixed recourse. To our best knowledge, these decision rules are new for their respective problem classes. By leveraging recent techniques in copositive programming, we derive equivalent copositive programs for the piecewise decision rule approximations. For MSRO problems with fixed recourse, we show that the state-of-the-art scheme by Georghiou et al. (2015) can be futile even on trivial two-stage problem instances, whereas our semidefinite approximation produces high-quality solutions. We formally prove that our proposed approximation is indeed tighter than that of Georghiou et al. (2015) and further identify the simplest set of semidefinite constraints that retains the outperformance while maintaining scalability.

The remainder of the paper is organized as follows. We derive the copositive programming reformulations for two-stage robust optimization problems in Section 2. In Section 3, we develop the conservative semidefinite programming approximations. We extend all results to the multistage setting in Section 4 and present the numerical results in Section 5.

1.1. Notation and Terminology

For any $M \in \mathbb{N}$, we define $[M]$ as the set of running indices $\{1, \ldots, M\}$. We let $[M] \backslash \{1\}$ be the set of running indices $\{2, \ldots, M\}$. We denote by $\mathbf{e}$ the vector of all ones and by $\mathbf{e}_i$ the ith standard basis vector. For notational convenience, we use both v_i and $[\boldsymbol{v}]_i$ to denote the ith component of the vector $\boldsymbol{v}$. The p-norm of a vector $\boldsymbol{v} \in \mathbb{R}^N$ is defined as $\|\boldsymbol{v}\|_p$. We will drop the subscript for the Euclidean norm (i.e., $\|\boldsymbol{v}\| := \|\boldsymbol{v}\|_2$). For $\boldsymbol{a} \in \mathbb{R}^N$ and $\boldsymbol{b} \in \mathbb{R}^N$, the Hadamard product of $\boldsymbol{a}$ and $\boldsymbol{b}$ is denoted by $\boldsymbol{a} \circ \boldsymbol{b} := (a_1 b_1, \ldots, a_N b_N)^\top$. The trace of a square matrix $\boldsymbol{X}$ is denoted as trace$(\boldsymbol{X})$. We use $[\boldsymbol{A}]_{ij}$ to denote the entry in the ith row and the jth column of the matrix $\boldsymbol{A}$. We define diag$(\boldsymbol{X})$ as the vector comprising the diagonal entries of $\boldsymbol{X}$ and Diag$(\boldsymbol{v})$ as the diagonal matrix with the vector $\boldsymbol{v}$ along its main diagonal. We use $\boldsymbol{X} \geq \mathbf{0}$ to denote that $\boldsymbol{X}$ is a component-wise nonnegative matrix. For any matrix $\boldsymbol{A} \in \mathbb{R}^{M \times N}$, the inclusion Rows$(\boldsymbol{A}) \in \mathcal{K}$ indicates that the column vectors corresponding to the rows of $\boldsymbol{A}$ are members of $\mathcal{K}$. We denote by $\mathscr{F}_{K+1,N}$ the space of all measurable mappings $\boldsymbol{y}(\cdot)$ from $\mathbb{R}^{K+1}$ to $\mathbb{R}^N$.

For any closed and convex cone $\mathcal{K}$, we denote its dual cone as $\mathcal{K}^*$. We define by $\mathcal{SOC} \subseteq \mathbb{R}^{K+1}$ the standard second-order cone (i.e., $\boldsymbol{v} \in \mathcal{SOC} \iff \|(v_1, \ldots, v_K)^\top\| \leq v_{K+1}$). We denote the space of symmetric matrices in $\mathbb{R}^{N \times N}$ as $\mathcal{S}^N$. For any $\boldsymbol{X} \in \mathcal{S}^N$, we set $\boldsymbol{X} \succeq \mathbf{0}$ to denote that $\boldsymbol{X}$ is positive semidefinite. For convenience, we call the cone of positive semidefinite matrices as the semidefinite cone and the cone of symmetric nonnegative matrices as the nonnegative cone. The *copositive cone* is defined as $\mathcal{COP}(\mathbb{R}^N_+) := \{\boldsymbol{M} \in \mathcal{S}^N : \boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x} \geq 0 \;\; \forall \boldsymbol{x} \in \mathbb{R}^N_+\}$. Its dual cone, the *completely positive cone*, is defined as $\mathcal{CP}(\mathbb{R}^N_+) := \{\boldsymbol{X} \in \mathcal{S}^N : \boldsymbol{X} = \sum_i \boldsymbol{x}^i (\boldsymbol{x}^i)^\top, \boldsymbol{x}^i \in \mathbb{R}^N_+\}$, where the summation over i is finite, but its cardinality is unspecified. For a general closed and convex cone $\mathcal{K} \subseteq \mathbb{R}^N$, we define the *generalized copositive cone* as $\mathcal{COP}(\mathcal{K})$ and the *generalized completely positive cone* as $\mathcal{CP}(\mathcal{K})$, respectively, in analogy with $\mathcal{COP}(\mathbb{R}^N_+)$ and $\mathcal{CP}(\mathbb{R}^N_+)$. Note that $\mathcal{COP}(\mathcal{K})$ and $\mathcal{CP}(\mathcal{K})$ are dual cones to each other. The term *copositive programming* refers to linear optimization over $\mathcal{COP}(\mathcal{K})$ or via duality, linear optimization over $\mathcal{CP}(\mathcal{K})$. To distinguish from the standard case where $\mathcal{K} = \mathbb{R}^N_+$, they are sometimes called *generalized copositive programming* or *set-semidefinite optimization*; see Eichfelder and Jahn (2008) and Burer and Dong (2012). In this paper, we work with generalized copositive programming, although we use the shorter phrase for simplicity.

2. Copositive Reformulations for Two-Stage Decision Rule Problems

In this section, we first state the generic setting of a two-stage robust optimization problem. We then consider various decision rules for the two-stage problem and propose copositive programming reformulations for the decision rule problems.

2.1. Two-Stage Robust Optimization Problem

We study adaptive linear optimization problems of the following general structure. A decision maker first takes a here-and-now decision $\boldsymbol{x} \in \mathcal{X}$, which incurs an immediate linear cost $\boldsymbol{c}^\top \boldsymbol{x}$. Nature then reacts with a worst-case parameter realization $\boldsymbol{u} \in \mathcal{U}$. In response, the decision maker takes a recourse action $\boldsymbol{y}(\boldsymbol{u}) \in \mathbb{R}^N$, which incurs a second-stage linear cost $\boldsymbol{d}(\boldsymbol{u})^\top \boldsymbol{y}(\boldsymbol{u})$. In this game against nature, the decision maker endeavors to optimally select a feasible solution $(\boldsymbol{x}, \boldsymbol{y}(\cdot))$ that minimizes the total cost $\boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u}\in\mathcal{U}} \boldsymbol{d}(\boldsymbol{u})^\top \boldsymbol{y}(\boldsymbol{u})$. We note that the second-stage decision vector constitutes a mapping $\boldsymbol{y} : \mathcal{U} \to \mathbb{R}^N$ and is thus infinite dimensional.

The emerging sequential decision problem can be formulated as a two-stage robust optimization problem given by

$$\begin{aligned} Z = \inf\ & \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u}\in\mathcal{U}} \boldsymbol{d}(\boldsymbol{u})^\top \boldsymbol{y}(\boldsymbol{u}) \\ \text{s.t.}\ & \mathcal{A}(\boldsymbol{u})\boldsymbol{x} + \mathcal{B}(\boldsymbol{u})\boldsymbol{y}(\boldsymbol{u}) \geq \boldsymbol{h}(\boldsymbol{u}) \quad \forall \boldsymbol{u} \in \mathcal{U} \\ & \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{y} \in \mathscr{F}_{K+1, N}. \end{aligned} \tag{1}$$

Here, the feasible set of the first-stage decision $\boldsymbol{x}$ is captured by a generic set $\mathcal{X} \subseteq \mathbb{R}^M$, whereas that of the second-stage decision $\boldsymbol{y}(\boldsymbol{u})$ is defined through a linear constraint system $\mathcal{A}(\boldsymbol{u})\boldsymbol{x} + \mathcal{B}(\boldsymbol{u})\boldsymbol{y}(\boldsymbol{u}) \geq \boldsymbol{h}(\boldsymbol{u})$. The uncertain parameter vector $\boldsymbol{u}$ is assumed to belong to a prescribed *uncertainty set* $\mathcal{U}$, which we model as the intersection of a slice of a closed and convex cone $\mathcal{K} \subseteq \mathbb{R}^K \times \mathbb{R}_+$, and the level sets of I quadratic functions. Specifically, we set

$$\mathcal{U} := \left\{ \boldsymbol{u} \in \mathcal{K} : \begin{array}{l} \mathbf{e}_{K+1}^\top \boldsymbol{u} = 1 \\ \boldsymbol{u}^\top \widehat{\boldsymbol{C}}_i \boldsymbol{u} = 0 \quad \forall i \in [I] \end{array} \right\}, \tag{2}$$

where $\widehat{\boldsymbol{C}}_i \in \mathcal{S}^{K+1}$ for all $i \in [I]$. The problem parameters $\mathcal{A}(\boldsymbol{u}) \in \mathbb{R}^{J\times M}$, $\mathcal{B}(\boldsymbol{u}) \in \mathbb{R}^{J\times N}$, $\boldsymbol{d}(\boldsymbol{u}) \in \mathbb{R}^N$, and $\boldsymbol{h}(\boldsymbol{u}) \in \mathbb{R}^J$ in (1) are assumed to be linear in $\boldsymbol{u}$, given by

$$\mathcal{A}(\boldsymbol{u}) = \sum_{k=1}^{K+1} u_k \widehat{\boldsymbol{A}}_k, \quad \mathcal{B}(\boldsymbol{u}) = \sum_{k=1}^{K+1} u_k \widehat{\boldsymbol{B}}_k,$$

$$\boldsymbol{d}(\boldsymbol{u}) = \widehat{\boldsymbol{D}}\boldsymbol{u}, \quad \boldsymbol{h}(\boldsymbol{u}) = \widehat{\boldsymbol{H}}\boldsymbol{u},$$

where $\widehat{\boldsymbol{A}}_k \in \mathbb{R}^{J\times M}$, $\widehat{\boldsymbol{B}}_k \in \mathbb{R}^{J\times N}$, $\widehat{\boldsymbol{D}} := (\widehat{\boldsymbol{d}}_1, \ldots, \widehat{\boldsymbol{d}}_N)^\top \in \mathbb{R}^{N\times(K+1)}$, and $\widehat{\boldsymbol{H}} := (\widehat{\boldsymbol{h}}_1, \ldots, \widehat{\boldsymbol{h}}_J)^\top \in \mathbb{R}^{J\times(K+1)}$ are deterministic data. The nonrestrictive assumption that $u_{K+1} = 1$ in (2) will simplify notation as it allows us to represent affine functions in the primitive uncertain parameters $(u_1, \ldots, u_K)^\top$ in a compact way as linear functions of $\boldsymbol{u}$ (e.g., the problem parameters $\mathcal{A}(\boldsymbol{u})$, $\mathcal{B}(\boldsymbol{u})$, $\boldsymbol{d}(\boldsymbol{u})$, and $\boldsymbol{h}(\boldsymbol{u})$ and the linear decision rule $\boldsymbol{Y}\boldsymbol{u}$) (Section 2.2) and as it also allows us to represent quadratic functions in the primitive uncertain parameters in a homogenized manner (e.g., the quadratic decision rule $\boldsymbol{u}^\top \boldsymbol{Q}\boldsymbol{u}$) (Section 2.3).

The cone $\mathcal{K}$ in the description of $\mathcal{U}$ has a generic form and can model many common uncertainty sets in the literature. We highlight three pertinent examples as follows.

Example 1 (Polytope). If the uncertainty set of the primitive vector $(u_1, \ldots, u_K)^\top$ is given by a polytope $\{\boldsymbol{\xi} \in \mathbb{R}^K : \boldsymbol{P}\boldsymbol{\xi} \geq \boldsymbol{q}\}$, then the corresponding cone is defined as

$$\mathcal{K} := \{(\boldsymbol{\xi}, \tau) \in \mathbb{R}^K \times \mathbb{R}_+ : \boldsymbol{P}\boldsymbol{\xi} \geq \boldsymbol{q}\tau\}.$$

Example 2 (Polytope and Two-Norm Ball). If the uncertainty set of the primitive vector is given by the intersection of a polytope and a transformed two-norm ball, $\{\boldsymbol{\xi} \in \mathbb{R}^K : \boldsymbol{P}\boldsymbol{\xi} \geq \boldsymbol{q}, \|\boldsymbol{R}\boldsymbol{\xi} - \boldsymbol{s}\|_2 \leq t\}$, then the corresponding cone is defined as

$$\mathcal{K} := \{(\boldsymbol{\xi}, \tau) \in \mathbb{R}^K \times \mathbb{R}_+ : \boldsymbol{P}\boldsymbol{\xi} \geq \boldsymbol{q}\tau, \|\boldsymbol{R}\boldsymbol{\xi} - \boldsymbol{s}\tau\| \leq t\tau\}.$$

Example 3 (Ellipsoids). Consider the setting where the uncertainty set of the primitive vector is described by an intersection of L ellipsoids: $\{\boldsymbol{\xi} \in \mathbb{R}^K : \boldsymbol{\xi}^\top \boldsymbol{F}_\ell \boldsymbol{\xi} + 2\boldsymbol{g}_\ell^\top \boldsymbol{\xi} \leq h_\ell,\ \forall \ell \in [L]\}$. Here, $\boldsymbol{F}_\ell \in \mathcal{S}^K$, $\boldsymbol{F}_\ell \succeq \boldsymbol{0}$, $\boldsymbol{g}_\ell \in \mathbb{R}^K$, and $h_\ell \in \mathbb{R}$ for all $\ell \in [L]$. Because $\boldsymbol{F}_\ell$ is positive semidefinite, we have $\boldsymbol{F}_\ell = \boldsymbol{P}_\ell^\top \boldsymbol{P}_\ell$ for some matrix $\boldsymbol{P}_\ell \in \mathbb{R}^{I_\ell \times K}$ whose rank is I_ℓ. In Alizadeh and Goldfarb (2003), it is shown that

$$\boldsymbol{\xi}^\top \boldsymbol{F}_\ell \boldsymbol{\xi} + 2\boldsymbol{g}_\ell^\top \boldsymbol{\xi} \leq h_\ell \Leftrightarrow \begin{pmatrix} \boldsymbol{P}_\ell \boldsymbol{\xi} \\ \frac{1}{2}(1 - h_\ell) + \boldsymbol{g}_\ell^\top \boldsymbol{\xi} \\ \frac{1}{2}(1 + h_\ell) - \boldsymbol{g}_\ell^\top \boldsymbol{\xi} \end{pmatrix} \in \mathcal{SOC}(I_\ell + 2),$$

where $\mathcal{SOC}(I_\ell + 2)$ denotes the second-order cone of dimension $I_\ell + 2$. In this case, the corresponding cone is given by

$$\mathcal{K} := \left\{ (\boldsymbol{\xi}, \tau) \in \mathbb{R}^K \times \mathbb{R}_+ : \begin{pmatrix} \boldsymbol{P}_\ell \boldsymbol{\xi} \\ \frac{1}{2}(1 - h_\ell)\tau + \boldsymbol{g}_\ell^\top \boldsymbol{\xi} \\ \frac{1}{2}(1 + h_\ell)\tau - \boldsymbol{g}_\ell^\top \boldsymbol{\xi} \end{pmatrix} \in \mathcal{SOC}(I_\ell + 2) \quad \forall \ell \in [L] \right\}.$$

In the following, to simplify our exposition, we define the convex set

$$\mathcal{U}^0 := \{\boldsymbol{u} \in \mathcal{K} : \mathbf{e}_{K+1}^\top \boldsymbol{u} = 1\}, \tag{3}$$

which corresponds to the uncertainty set $\mathcal{U}$ in the absence of the nonconvex constraints $\boldsymbol{u}^\top \widehat{\boldsymbol{C}}_i \boldsymbol{u} = 0$, $i \in [I]$. We further assume that the uncertainty set satisfies the following regularity conditions.

Assumption 1. *The set $\mathcal{U}^0$ defined in (3) is nonempty and compact.*

Assumption 2. *The minimum value of the quadratic function $\boldsymbol{u}^\top \widehat{\boldsymbol{C}}_i \boldsymbol{u}$ over the set $\mathcal{U}^0$ is zero for all $i \in [I]$ (i.e., $0 = \min_{\boldsymbol{u} \in \mathcal{U}^0} \boldsymbol{u}^\top \widehat{\boldsymbol{C}}_i \boldsymbol{u}$, $i \in [I]$).*

The quadratic constraints in the description of $\mathcal{U}$ are motivated by both practical and modeling requirements. Numerous applications in robust optimization, including inventory planning and project crashing problems, involve binary uncertain parameters; see Mittal et al. (2020). In this case, we can incorporate binary variables in $\mathcal{U}$ via quadratic constraints of the form in (2). Specifically, we have that $u_k \in \{0, 1\}$ is equivalent to $u_k^2 = u_k$. If the relation $0 \le u_k \le 1$ is implied by $\mathcal{U}^0$ (note that we can explicitly introduce these constraints into $\mathcal{U}^0$ if necessary), then we have $0 = \min_{\boldsymbol{u} \in \mathcal{U}^0} \{-u_k^2 + u_k\}$, which shows that the quadratic constraint $-u_k^2 + u_k = 0$ satisfies the condition in Assumption 2. Furthermore, these constraints will be crucial for deriving our improved decision rules as they enable us to model complementary constraints (e.g., $u_k u_{k'} = 0$); see Section 2.4 for detail. If $\mathcal{U}^0$ implies that both u_k and $u_{k'}$ are nonnegative and bounded, then we have $0 = \min_{\boldsymbol{u} \in \mathcal{U}^0} \{u_k u_{k'}\}$. Thus, the quadratic constraint $u_k u_{k'} = 0$ satisfies the condition in Assumption 2.

Two-stage robust optimization problems of the form (1) are generically NP hard; see Ben-Tal et al. (2004). A popular conservative approximation scheme is obtained in *linear decision rules*, where we restrict the recourse action $\boldsymbol{y}(\cdot)$ to be a linear function of $\boldsymbol{u}$. If the problem has fixed *recourse* (i.e., $\mathcal{B}(\boldsymbol{u})$ and $\boldsymbol{d}(\boldsymbol{u})$ are constant), then the linear decision rule approximation leads to tractable linear programs. On the other hand, if the problem has *nonfixed recourse* (i.e., $\mathcal{B}(\boldsymbol{u})$ or $\boldsymbol{d}(\boldsymbol{u})$ depends linearly in $\boldsymbol{u}$), then the approximation itself is intractable. In the following, we show that the linear decision rule problems are amenable to exact copositive programming reformulations. Furthermore, in the specific case where the problem has fixed recourse, we develop an improved approximation in *quadratic decision rules* and show that the resulting optimization problems can also be reformulated as equivalent copositive programs.

2.2. Linear Decision Rule for Problems with Nonfixed Recourse

In this section, we derive an exact copositive program by applying linear decision rules to Problem (1). Instead of considering all possible choices of functions $\boldsymbol{y} : \mathcal{U} \to \mathbb{R}^N$ from $\mathscr{F}_{K+1,N}$, we restrict ourselves to linear functions of the form

$$\boldsymbol{y}(\boldsymbol{u}) = \boldsymbol{Y}\boldsymbol{u},$$

for some coefficient matrix $\boldsymbol{Y} \in \mathbb{R}^{N \times (K+1)}$. This setting yields the following conservative approximation of Problem (1):

$$\begin{aligned} Z^{\mathcal{L}} = \inf\ & \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u} \in \mathcal{U}} \boldsymbol{d}(\boldsymbol{u})^\top (\boldsymbol{Y}\boldsymbol{u}) \\ \text{s.t.}\ & \mathcal{A}(\boldsymbol{u})\boldsymbol{x} + \mathcal{B}(\boldsymbol{u})\boldsymbol{Y}\boldsymbol{u} \ge \boldsymbol{h}(\boldsymbol{u}) \quad \forall \boldsymbol{u} \in \mathcal{U} \\ & \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{Y} \in \mathbb{R}^{N \times (K+1)}. \end{aligned} \tag{$\mathcal{L}$}$$

Problem ($\mathcal{L}$) is finite dimensional but remains difficult to solve as there are infinitely many constraints parameterized by $\boldsymbol{u} \in \mathcal{U}$. In particular, it is shown in Ben-Tal et al. (2004) that the problem is NP hard via a reduction from the problem of checking matrix copositivity.

We now show that an equivalent copositive programming reformulation can principally be derived for Problem ($\mathcal{L}$). We first introduce the following technical lemmas, which are fundamental for our derivations. The first technical lemma establishes the equivalence between a nonconvex quadratic program

$$\begin{aligned} \sup\ & \boldsymbol{u}^\top \widehat{\boldsymbol{C}}_0 \boldsymbol{u} \\ \text{s.t.}\ & \mathbf{e}_{K+1}^\top \boldsymbol{u} = 1 \\ & \boldsymbol{u}^\top \widehat{\boldsymbol{C}}_i \boldsymbol{u} = 0 \quad \forall i \in [I] \\ & \boldsymbol{u} \in \mathcal{K} \end{aligned} \tag{4}$$

and its copositive relaxation

$$\begin{aligned} \sup\ & \widehat{\boldsymbol{C}}_0 \bullet \boldsymbol{U} \\ \text{s.t.}\ & \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top \bullet \boldsymbol{U} = 1 \\ & \widehat{\boldsymbol{C}}_i \bullet \boldsymbol{U} = 0 \quad \forall i \in [I] \\ & \boldsymbol{U} \in \mathcal{CP}(\mathcal{K}), \end{aligned} \tag{5}$$

where $\widehat{\boldsymbol{C}}_0 \in \mathcal{S}^{K+1}, \mathcal{K} \subseteq \mathbb{R}^{K+1}$ is a closed and convex cone and $\mathcal{CP}(\mathcal{K})$ is the cone of completely positive matrices with respect to $\mathcal{K}$.

Lemma 1 (Burer 2012, corollary 8.4 and theorem 8.3). *Suppose that Assumptions 1 and 2 hold. Then, Problem (5) is equivalent to (4) (i.e., (i) the optimal value of (5) is equal to that of (4); (ii) if $\boldsymbol{U}^\star$ is an optimal solution for (5), then $\boldsymbol{U}^\star \mathbf{e}_1$ is in the convex hull of optimal solutions for (4)).*

Lemma 2. *Suppose Assumption 1 holds. Then, for any $(\boldsymbol{z}, \tau) \in \mathcal{K}$, we have that $\tau = 0$ implies $\boldsymbol{z} = \mathbf{0}$.*

Proof. See the e-companion. □

The dual of Problem (5) is given by the following linear program over the cone of copositive matrices with respect to $\mathcal{K}$:

$$\begin{aligned} &\inf \lambda \\ &\text{s.t. } \lambda\, \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top + \sum_{i=1}^{I} \alpha_i \widehat{\boldsymbol{C}}_i - \widehat{\boldsymbol{C}}_0 \in \mathcal{COP}(\mathcal{K}) \\ &\quad \lambda \in \mathbb{R},\ \boldsymbol{\alpha} \in \mathbb{R}^I. \end{aligned} \tag{6}$$

Our next technical lemma establishes strong duality for the primal and dual pairs.

Lemma 3. *Suppose Assumption 1 holds. Then, strong duality holds between Problems (5) and (6).*

Proof. See the e-companion. □

In the following, we define the auxiliary matrices

$$\widehat{\boldsymbol{\Theta}}_j := \begin{pmatrix} \mathbf{e}_j^\top \widehat{\boldsymbol{A}}_1 \\ \vdots \\ \mathbf{e}_j^\top \widehat{\boldsymbol{A}}_{K+1} \end{pmatrix} \in \mathbb{R}^{(K+1)\times M},$$

$$\widehat{\boldsymbol{\Lambda}}_j := \begin{pmatrix} \mathbf{e}_j^\top \widehat{\boldsymbol{B}}_1 \\ \vdots \\ \mathbf{e}_j^\top \widehat{\boldsymbol{B}}_{K+1} \end{pmatrix} \in \mathbb{R}^{(K+1)\times N}, \text{ and} \tag{7}$$

$$\begin{aligned} \boldsymbol{\Omega}_j(\boldsymbol{x}, \boldsymbol{Y}) := \frac{1}{2}\Big(&\widehat{\boldsymbol{\Theta}}_j \boldsymbol{x}\mathbf{e}_{K+1}^\top + \mathbf{e}_{K+1}\boldsymbol{x}^\top \widehat{\boldsymbol{\Theta}}_j^\top + \widehat{\boldsymbol{\Lambda}}_j \boldsymbol{Y} \\ &+ \boldsymbol{Y}^\top \widehat{\boldsymbol{\Lambda}}_j^\top - \widehat{\boldsymbol{h}}_j \mathbf{e}_{K+1}^\top - \mathbf{e}_{K+1}\widehat{\boldsymbol{h}}_j^\top\Big) \quad \forall j \in [J], \end{aligned} \tag{8}$$

where $\mathbf{e}_j$ represents the jth standard basis vector in $\mathbb{R}^J$. We are now ready to state our main result.

Theorem 1. *Problem ($\mathcal{L}$) is equivalent to the copositive program*

$$\begin{aligned} Z^{\mathcal{L}} = &\inf \boldsymbol{c}^\top \boldsymbol{x} + \lambda \\ &\text{s.t. } \lambda\, \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top - \frac{1}{2}(\widehat{\mathbf{D}}^\top \boldsymbol{Y} + \boldsymbol{Y}^\top \widehat{\mathbf{D}}) + \sum_{i=1}^{I} \alpha_i \widehat{\mathbf{C}}_i \in \mathcal{COP}(\mathcal{K}) \\ &\quad \boldsymbol{\Omega}_j(\boldsymbol{x}, \boldsymbol{Y}) - \pi_j\, \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top - \sum_{i=1}^{I} [\boldsymbol{\beta}_j]_i \widehat{\boldsymbol{C}}_i \in \mathcal{COP}(\mathcal{K}) \quad \forall j \in [J] \\ &\quad \boldsymbol{x} \in \mathcal{X},\ \lambda \in \mathbb{R},\ \boldsymbol{Y} \in \mathbb{R}^{N\times(K+1)},\ \boldsymbol{\pi} \in \mathbb{R}_+^J,\ \boldsymbol{\alpha} \in \mathbb{R}^I, \\ &\quad \boldsymbol{\beta}_j \in \mathbb{R}^I \quad \forall j \in [J], \end{aligned} \tag{9}$$

where the affine functions $\boldsymbol{\Omega}_j(\boldsymbol{x}, \boldsymbol{Y})$, $j \in [J]$ are defined as in (8).

Proof. See the e-companion. □

2.3. Quadratic Decision Rule for Problems with Fixed Recourse

We now study two-stage robust optimization problems with fixed recourse. In this simpler setting, the second-stage cost coefficients and the recourse matrix are deterministic: that is,

$$\boldsymbol{d}(\boldsymbol{u}) = \widehat{\boldsymbol{d}} \in \mathbb{R}^N \quad \text{and} \quad \mathcal{B}(\boldsymbol{u}) = \widehat{\boldsymbol{B}} \in \mathbb{R}^{J\times N} \qquad \forall \boldsymbol{u} \in \mathbb{R}^{K+1}.$$

Using techniques developed in the previous section, we will derive a copositive programming reformulation by applying decision rules to the recourse action $\boldsymbol{y}: \mathcal{U} \to \mathbb{R}^N$. Because $\boldsymbol{d}(\boldsymbol{u})$ and $\mathcal{B}(\boldsymbol{u})$ are constant, we may utilize the more powerful quadratic decision rules defined as

$$[\boldsymbol{y}(\boldsymbol{u})]_n = \boldsymbol{u}^\top \boldsymbol{Q}_n \boldsymbol{u} \qquad \forall n \in [N],$$

for some coefficient matrices $\boldsymbol{Q}_n \in \mathcal{S}^{K+1}$, $n \in [N]$. This yields the following conservative approximation of Problem (1):

$$\begin{aligned} Z^{\mathcal{Q}} = &\inf \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u}\in\mathcal{U}} \sum_{n=1}^{N} \widehat{d}_n \boldsymbol{u}^\top \boldsymbol{Q}_n \boldsymbol{u} \\ &\text{s.t. } \boldsymbol{u}^\top \widehat{\boldsymbol{\Theta}}_j \boldsymbol{x} + \sum_{n=1}^{N} \widehat{b}_{jn} \boldsymbol{u}^\top \boldsymbol{Q}_n \boldsymbol{u} \geq \widehat{\boldsymbol{h}}_j^\top \boldsymbol{u} \quad \forall \boldsymbol{u} \in \mathcal{U}\ \forall j \in [J] \\ &\quad \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{Q}_n \in \mathcal{S}^{K+1} \quad \forall n \in [N]. \end{aligned} \tag{$\mathcal{Q}$}$$

In view of the restriction $u_{K+1} = 1$ in the description of $\mathcal{U}$, the decision rule $[\boldsymbol{y}(\boldsymbol{u})]_n = \boldsymbol{u}^\top \boldsymbol{Q}_n \boldsymbol{u}$ constitutes a homogenized version of a nonhomogenized quadratic function in the primitive vector $(u_1, \ldots, u_K)^\top$. We remark that optimizing for the best quadratic decision rule is generically NP hard (Ben-Tal et al. 2009, section 14.3.2). This justifies our proposed copositive programming reformulation, which we derive in the following theorem. To that end, we define the affine functions

$$\begin{aligned} &\Gamma_j(\boldsymbol{x}, \boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_N) \\ &:= \frac{1}{2}(\widehat{\boldsymbol{\Theta}}_j \boldsymbol{x}\mathbf{e}_{K+1}^\top + \mathbf{e}_{K+1}\boldsymbol{x}^\top \widehat{\boldsymbol{\Theta}}_j^\top - \mathbf{e}_{K+1}\widehat{\boldsymbol{h}}_j^\top - \widehat{\boldsymbol{h}}_j \mathbf{e}_{K+1}^\top) \\ &\quad + \sum_{n=1}^{N} \widehat{b}_{jn} \boldsymbol{Q}_n \qquad \forall j \in [J]. \end{aligned} \tag{10}$$

Theorem 2. *Problem ($\mathcal{Q}$) is equivalent to the copositive program*

$$\begin{aligned} Z^{\mathcal{Q}} = &\min \boldsymbol{c}^\top \boldsymbol{x} + \lambda \\ &\text{s.t. } \lambda\, \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top - \sum_{n=1}^{N} \widehat{d}_n \boldsymbol{Q}_n + \sum_{i=1}^{I} \alpha_i \widehat{\boldsymbol{C}}_i \in \mathcal{COP}(\mathcal{K}) \\ &\quad \boldsymbol{\Gamma}_j(\boldsymbol{x}, \boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_N) - \pi_j\, \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top - \sum_{i=1}^{I} [\boldsymbol{\beta}_j]_i \widehat{\boldsymbol{C}}_i \in \mathcal{COP}(\mathcal{K}) \quad \forall j \in [J] \\ &\quad \boldsymbol{x} \in \mathcal{X},\ \lambda \in \mathbb{R},\ \boldsymbol{\alpha} \in \mathbb{R}^I,\ \boldsymbol{\pi} \in \mathbb{R}_+^J,\ \boldsymbol{Q}_n \in \mathcal{S}^{K+1} \\ &\quad \forall n \in [N],\ \boldsymbol{\beta}_j \in \mathbb{R}^I \quad \forall j \in [J], \end{aligned} \tag{11}$$

where the affine functions $\Gamma_j(\boldsymbol{x}, \boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_N)$, $j \in [J]$ are defined as in (10).

Proof. See the e-companion. □

2.4. Enhanced Decision Rule

In this section, we tighten the basic decision rule approximations by employing piecewise linear and piecewise quadratic decision rules. Although piecewise quadratic decision rules are a new concept, piecewise linear decision rules have been studied extensively in the literature (Chen and Zhang 2009, Georghiou et al. 2015). Their utilization is supported by a strong theoretical justification. For problems with fixed recourse, the optimal recourse action $y(\cdot)$ can be described by a piecewise linear continuous function (Bemporad et al. 2003). However, optimizing for the best piecewise linear decision rule is NP hard even if the folding directions and their respective break points are prescribed a priori (Georghiou et al. 2015, theorem 4.2). We endeavor to derive equivalent copositive reformulations for the piecewise decisions rule problems that lead to tight semidefinite approximations.

To this end, for a prescribed number of pieces L, we define the mappings

$$F_\ell(\boldsymbol{u}) = \max\{0, \boldsymbol{f}_\ell^\top \boldsymbol{u}\} \qquad \forall \boldsymbol{u} \in \mathbb{R}^{K+1} \quad \forall \ell \in [L]. \tag{12}$$

Here, $\boldsymbol{f}_\ell := (\boldsymbol{g}_\ell, -h_\ell) \in \mathbb{R}^{K+1}$, where $\boldsymbol{g}_\ell \in \mathbb{R}^K$ denotes the folding direction of the ℓth mapping, whereas h_ℓ defines its break point. These mappings constitute the building blocks of our improved decision rules. Specifically, by applying the basic linear and quadratic decision rules on the lifted uncertain parameter vector $\boldsymbol{v} := (F_1(\boldsymbol{u}), \dots, F_L(\boldsymbol{u}), \boldsymbol{u}) \in \mathbb{R}^{L+K+1}$, we arrive at the desired piecewise linear and piecewise quadratic decision rules, respectively.

Example 4 (Integer Programming Feasibility Problem). Consider a norm maximization problem given by $\max_{\boldsymbol{u}\in\mathcal{U}} \|\boldsymbol{u}\|_1$, where $\mathcal{U} = \{\boldsymbol{u} \in \mathbb{R}^K : \boldsymbol{P}\boldsymbol{u} \le \boldsymbol{q}\} \subseteq [-1,1]^K$ is a prescribed polytope. An elementary analysis shows that the optimal value of this problem is equal to K if and only if there exists a binary vector $\boldsymbol{u} \in \{-1,1\}^K$ within the polytope $\mathcal{U}$. Thus, it solves the NP-hard integer programming (IP) feasibility problem (Garey and Johnson 1979). We can reformulate the norm maximization problem as a two-stage robust optimization problem without a first-stage decision $\boldsymbol{x}$ given by

$$\begin{aligned} &\inf \sup_{\boldsymbol{u}\in\mathcal{U}} \mathbf{e}^\top \boldsymbol{y}(\boldsymbol{u}) \\ &\text{s.t. } \boldsymbol{y}(\boldsymbol{u}) \ge \boldsymbol{u},\ \boldsymbol{y}(\boldsymbol{u}) \ge -\boldsymbol{u} \quad \forall \boldsymbol{u} \in \mathcal{U} \\ &\qquad \boldsymbol{y} \in \mathscr{F}_{K,K}. \end{aligned}$$

Indeed, at optimality we have $[\boldsymbol{y}(\boldsymbol{u})]_k = |u_k|$, which implies that $\mathbf{e}^\top \boldsymbol{y}(\boldsymbol{u}) = \|\boldsymbol{u}\|_1$. Consider now the mappings

$$F_\ell(\boldsymbol{u}) = \max\{0,\ u_\ell\} \qquad \forall \boldsymbol{u} \in \mathbb{R}^K \quad \forall \ell \in [K].$$

Our previous argument shows that the piecewise linear decision rule given by

$$[\boldsymbol{y}(\boldsymbol{u})]_\ell = -u_\ell + 2F_\ell(\boldsymbol{u}) = -u_\ell + \max\{0, 2u_\ell\} = |u_\ell| \qquad \forall \ell \in [K]$$

is optimal. This decision rule is linear in the lifted parameter vector $(F_1(\boldsymbol{u}), \dots, F_K(\boldsymbol{u}), \boldsymbol{u})$.

To formalize the idea into our setting, we define the lifted set

$$\mathcal{U}' := \{\boldsymbol{v} := (\boldsymbol{w}, \boldsymbol{u}) \in \mathbb{R}^L \times \mathcal{U} :\ w_\ell = F_\ell(\boldsymbol{u}) \quad \forall \ell \in [L]\} \tag{13}$$

and the lifted parameters

$$\mathcal{A}'(\boldsymbol{v}) = \mathcal{A}(\boldsymbol{u}),\ \mathcal{B}'(\boldsymbol{v}) = \mathcal{B}(\boldsymbol{u}),\ \boldsymbol{d}'(\boldsymbol{v}) = \boldsymbol{d}(\boldsymbol{u}),\ \boldsymbol{h}'(\boldsymbol{v}) = \boldsymbol{h}(\boldsymbol{u}),$$
$$\widehat{\boldsymbol{\Theta}}_j' = (\mathbf{0}^\top, \widehat{\boldsymbol{\Theta}}_j^\top)^\top \in \mathbb{R}^{(L+K+1)\times M} \qquad \forall j \in [J].$$

Then, by replacing the set $\mathcal{U}$ with $\mathcal{U}'$ and employing the lifted parameters in ($\mathcal{L}$) and ($\mathcal{Q}$), we obtain the corresponding piecewise decision rule problems. These are given by

$$\begin{aligned} Z^{\mathcal{PL}} = \inf\ & \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{v}\in\mathcal{U}'} \boldsymbol{d}'(\boldsymbol{v})^\top \boldsymbol{Y}\boldsymbol{v} \\ \text{s.t. }\ & \mathcal{A}'(\boldsymbol{v})\boldsymbol{x} + \mathcal{B}'(\boldsymbol{v})\boldsymbol{Y}\boldsymbol{v} \ge \boldsymbol{h}'(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in \mathcal{U}' \\ & \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{Y} \in \mathbb{R}^{N\times(L+K+1)} \end{aligned} \tag{$\mathcal{PL}$}$$

and

$$\begin{aligned} Z^{\mathcal{PQ}} = \inf\ & \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u}'\in\mathcal{U}'} \sum_{n=1}^N \widehat{d}_n \boldsymbol{v}^\top \boldsymbol{Q}_n \boldsymbol{v} \\ \text{s.t. }\ & \boldsymbol{v}^\top \widehat{\boldsymbol{\Theta}}_j' \boldsymbol{x} + \sum_{n=1}^N \widehat{b}_{jn} \boldsymbol{v}^\top \boldsymbol{Q}_n \boldsymbol{v} \ge [\boldsymbol{h}'(\boldsymbol{v})]_n \\ & \qquad \forall \boldsymbol{v} \in \mathcal{U}' \quad \forall j \in [J] \\ & \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{Q}_n \in \mathcal{S}^{L+K+1} \quad \forall n \in [N], \end{aligned} \tag{$\mathcal{PQ}$}$$

respectively.

We now establish that the piecewise decision rule problems can be equivalently reformulated as polynomial-size copositive programs. The reformulations leverage our capability to incorporate complementary constraints in the uncertainty set $\mathcal{U}$. We remark that Problems ($\mathcal{PL}$) and ($\mathcal{PQ}$) share the same structure as their plain vanilla counterparts ($\mathcal{L}$) and ($\mathcal{Q}$). To establish that equivalent copositive programs can also be derived for these problems, we need to show that the set $\mathcal{U}'$ can be brought into the standard form (2). First, we prove that the nonconvex set $\mathcal{U}'$ is equivalent to a concise set involving $\mathcal{O}(L)$ linear and complementary constraints.

Theorem 3. *The lifted uncertainty set in (13) can be represented as the set*

$$\mathcal{U}' = \left\{ (\boldsymbol{w}, \boldsymbol{u}) \in \mathbb{R}^L \times \mathcal{U} : \begin{array}{ll} \mathbf{0} \le \boldsymbol{w} \le \overline{\boldsymbol{w}} & \\ w_\ell \ge \boldsymbol{f}_\ell^\top \boldsymbol{u} & \forall \ell \in [L] \\ w_\ell (w_\ell - \boldsymbol{f}_\ell^\top \boldsymbol{u}) = 0 & \forall \ell \in [L] \end{array} \right\}, \tag{14}$$

where $\overline{\boldsymbol{w}} \in \mathbb{R}^L$ is a vector whose components are upper bounds on the auxiliary parameters $w_1, \ldots, w_L$. These upper bounds can be computed by solving L convex conic optimization problems given by

$$\overline{w}_\ell := \max_{\boldsymbol{u} \in \mathcal{U}^0} \boldsymbol{f}_\ell^\top \boldsymbol{u} \qquad \forall \ell \in [L],$$

where $\mathcal{U}^0$ is defined as in (3).

Proof. For any fixed $\boldsymbol{u} \in \mathcal{U}$ and $\ell \in [L]$, the complementary constraint $w_\ell(w_\ell - \boldsymbol{f}_\ell^\top \boldsymbol{u}) = 0$ implies that either $w_\ell = 0$ or $w_\ell = \boldsymbol{f}_\ell^\top \boldsymbol{u}$. Thus, the constraints $w_\ell \ge 0$ and $w_\ell \ge \boldsymbol{f}_\ell^\top \boldsymbol{u}$ yield $w_\ell = \max\{0, \boldsymbol{f}_\ell^\top \boldsymbol{u}\}$. This completes the proof. □

Remark 1. The inclusion of the upper bound $\overline{\boldsymbol{w}}$ on the lifted parameters $\boldsymbol{w}$ ensures the boundedness of the uncertainty set $\mathcal{U}'$, which is required by Assumption 1. We also note that $\mathcal{U}'$ is closed because $w_\ell(w_\ell - \boldsymbol{f}_\ell^\top \boldsymbol{u}) = 0 \Leftrightarrow w_\ell = 0$ or $w_\ell - \boldsymbol{f}_\ell^\top \boldsymbol{u} = 0$ for all $\ell \in [L]$. Therefore, $\mathcal{U}'$ is a compact set.

Next, in view of the equivalent set in (14), we define the lifted cone

$$\mathcal{K}' := \left\{ (\boldsymbol{w}, \boldsymbol{u}) \in \mathbb{R}^L \times \mathcal{U} : \begin{array}{ll} \mathbf{0} \le \boldsymbol{w} \le \overline{\boldsymbol{w}} u_{K+1} & \\ w_\ell \ge \boldsymbol{f}_\ell^\top \boldsymbol{u} & \forall \ell \in [L] \end{array} \right\}.$$

Letting the matrices $\widehat{\boldsymbol{C}}_\ell$, $\ell \in [L]$ be defined as

$$\widehat{\boldsymbol{C}}_\ell = (\mathbf{e}_\ell^\top, \mathbf{0}^\top)^\top (\mathbf{e}_\ell^\top, \mathbf{0}^\top) - \frac{1}{2} (\mathbf{e}_\ell^\top, \mathbf{0}^\top)^\top (\mathbf{0}^\top, \boldsymbol{f}_\ell^\top) - \frac{1}{2} (\mathbf{0}^\top, \boldsymbol{f}_\ell^\top)^\top (\mathbf{e}_\ell^\top, \mathbf{0}^\top) \qquad \forall \ell \in [L],$$

we can capture the complementarity constraints in $\mathcal{U}'$ via the quadratic equalities $\boldsymbol{v}^\top \widehat{\boldsymbol{C}}_\ell \boldsymbol{v} = 0$, $\ell \in [L]$. Thus, the lifted set coincides with the set

$$\mathcal{U}' := \{ \boldsymbol{v} := (\boldsymbol{w}, \boldsymbol{u}) \in \mathcal{K}' : u_{K+1} = 1, \boldsymbol{v}^\top \widehat{\boldsymbol{C}}_\ell \boldsymbol{v} = 0 \ \ \forall \ell \in [L] \},$$

which indeed assumes the standard form in (2). In summary, we have established that equivalent copositive programs can be derived for the proposed piecewise linear and piecewise quadratic decision rule problems. As described in Section 3, tractable semidefinite programming approximations can then be obtained by replacing the cone $\mathcal{COP}(\mathcal{K}')$ in the respective copositive programs with the inner approximation $\mathcal{IA}(\mathcal{K}')$.

3. Semidefinite Programming Solution Schemes

Our equivalence results imply that the decision rule problems are amenable to semidefinite programming solution schemes. Specifically, there exists a hierarchy of increasingly tight semidefinite-representable inner approximations that converge to $\mathcal{COP}(\mathcal{K})$; see, for example, Parrilo (2000), Bomze and de Klerk (2002), de Klerk and Pasechnik (2002), and Lasserre (2009). Replacing the cone $\mathcal{COP}(\mathcal{K})$ with these inner approximations yields conservative semidefinite programs that can be solved using standard off-the-shelf solvers. In this section, we develop new tractable approximations and exact semidefinite reformulations for the copositive programs derived in Section 2. To this end, we primarily consider polyhedral- and second-order cone-representable uncertainty sets defined via closed and convex cones of the following generic form:

$$\mathcal{K} := \{ \boldsymbol{u} \in \mathbb{R}^K \times \mathbb{R}_+ : \widehat{\boldsymbol{P}} \boldsymbol{u} \ge \mathbf{0}, \widehat{\boldsymbol{R}} \boldsymbol{u} \in \mathcal{SOC}(K_r) \}, \tag{15}$$

with $\widehat{\boldsymbol{P}} \in \mathbb{R}^{K_p \times (K+1)}$ and $\widehat{\boldsymbol{R}} \in \mathbb{R}^{K_r \times (K+1)}$. As illustrated in the examples of Section 2, the generic structure for the cone $\mathcal{K}$ can encompass many commonly used uncertainty sets in practice.

3.1. Conservative Approximation

We consider a semidefinite-representable inner approximation to the cone $\mathcal{COP}(\mathcal{K})$ given by

$$\mathcal{IA}(\mathcal{K}) := \left\{ \boldsymbol{V} \in \mathcal{S}^{K+1} : \begin{array}{l} \boldsymbol{W} \in \mathcal{S}^{K+1}, \boldsymbol{W} \succeq \mathbf{0}, \Sigma \in \mathcal{S}^{K_p} \\ \boldsymbol{\Psi} \in \mathcal{S}^{K+1}, \boldsymbol{\Phi} \in \mathbb{R}^{K_p \times K_r}, \tau \in \mathbb{R} \\ \boldsymbol{V} = \boldsymbol{W} + \tau \widehat{\boldsymbol{S}} + \widehat{\boldsymbol{P}}^\top \boldsymbol{\Sigma} \widehat{\boldsymbol{P}} + \Psi, \boldsymbol{\Sigma} \ge \mathbf{0}, \tau \ge 0 \\ \Psi = \frac{1}{2} (\widehat{\boldsymbol{P}}^\top \boldsymbol{\Phi} \widehat{\boldsymbol{R}} + \widehat{\boldsymbol{R}}^\top \boldsymbol{\Phi}^\top \widehat{\boldsymbol{P}}), \\ \text{Rows}(\boldsymbol{\Phi}) \in \mathcal{SOC}(K_r) \end{array} \right\}, \tag{16}$$

where the matrix $\widehat{\boldsymbol{S}}$ is defined as

$$\widehat{\boldsymbol{S}} := \widehat{\boldsymbol{R}}^\top \mathbf{e}_{K_r} \mathbf{e}_{K_r}^\top \widehat{\boldsymbol{R}} - \sum_{\ell=1}^{K_r - 1} \widehat{\boldsymbol{R}}^\top \mathbf{e}_\ell \mathbf{e}_\ell^\top \widehat{\boldsymbol{R}}. \tag{17}$$

We now establish that $\mathcal{IA}(\mathcal{K})$ is a subset of $\mathcal{COP}(\mathcal{K})$. To this end, we make the following observation.

Lemma 4. *We have $\boldsymbol{u}^\top \widehat{\boldsymbol{S}} \boldsymbol{u} \ge 0$ for all $\boldsymbol{u} \in \mathcal{K}$.*

Proof. See the e-companion. □

Using Lemma 4, we are now ready to prove the containment result.

Proposition 1. *We have $\mathcal{IA}(\widehat{\mathcal{U}}) \subseteq \mathcal{COP}(\widehat{\mathcal{U}})$.*

Proof. See the e-companion. □

Replacing the cone $\mathcal{COP}(\mathcal{K})$ in (9) and (11) with the inner approximation $\mathcal{IA}(\mathcal{K})$ yields conservative semidefinite programs. We denote their optimal values as $Z_{\text{IA}}^{\mathcal{L}}$ and $Z_{\text{IA}}^{\mathcal{Q}}$, respectively. The following proposition summarizes our current findings.

Proposition 2. *We have* $Z^{\mathcal{L}} \le Z_{\text{IA}}^{\mathcal{L}}$ *and* $Z^{\mathcal{Q}} \le Z_{\text{IA}}^{\mathcal{Q}}$.

An alternative conservative approximation scheme is proposed by Ben-Tal et al. (2009) in view of the *approximate S lemma* (Ben-Tal et al. 2009, theorem B.3.1). In this case, the corresponding inner approximation for the cone $\mathcal{COP}(\mathcal{K})$ is given by

$$\mathcal{AS}(\mathcal{K}) := \left\{ \boldsymbol{V} \in \mathcal{S}^{K+1} : \begin{array}{l} \tau \ge 0,\, \boldsymbol{\theta} \in \mathbb{R}_+^{K_p},\, \boldsymbol{W} \in \mathcal{S}^{K+1},\, \boldsymbol{W} \succeq \boldsymbol{0} \\ \boldsymbol{V} = \boldsymbol{W} + \tau \widehat{\boldsymbol{S}} + \frac{1}{2}(\widehat{\boldsymbol{P}}^\top \boldsymbol{\theta} \mathbf{e}_{K+1}^\top + \mathbf{e}_{K+1} \boldsymbol{\theta}^\top \widehat{\boldsymbol{P}}) \end{array} \right\}, \tag{18}$$

where $\widehat{\boldsymbol{S}}$ is defined as in (17). Replacing the cone $\mathcal{COP}(\mathcal{K})$ in (9) and (11) with $\mathcal{AS}(\mathcal{K})$ yields conservative semidefinite programs whose optimal values are denoted as $Z_{\text{AS}}^{\mathcal{L}}$ and $Z_{\text{AS}}^{\mathcal{Q}}$, respectively. We now show that $\mathcal{AS}(\mathcal{K})$ is inferior to $\mathcal{IA}(\mathcal{K})$ for approximating $\mathcal{COP}(\mathcal{K})$.

Proposition 3. *We have* $\mathcal{AS}(\mathcal{K}) \subseteq \mathcal{IA}(\mathcal{K})$.

Proof. The inclusion follows by simply setting $\boldsymbol{\Sigma} = \frac{1}{2}(\boldsymbol{\theta}\mathbf{e}_{K+1}^\top + \mathbf{e}_{K+1}\boldsymbol{\theta}^\top)$ and $\boldsymbol{\Psi} = \boldsymbol{0}$ in cone $\mathcal{IA}(\mathcal{K})$. □

Lastly, another conservative approximation scheme naturally arises in polynomial decision rules (Bertsimas et al. 2010). Here, one first imposes the restriction that the recourse function $\boldsymbol{y}(\cdot)$ in (1) is a polynomial of fixed degree d. Because optimizing for the best polynomial decision rule is generically NP hard, one resorts to another layer of approximation in semidefinite programming. To this end, consider a degree d polynomial decision rule. For problems with nonfixed recourse, we find that each semiinfinite constraint in (1) reduces to the problem of checking the nonnegativity of a polynomial of degree $\widehat{d} = d + 1$ over the set $\mathcal{U}$, whereas for problems with fixed recourse, it reduces to the problem of checking the nonnegativity of a polynomial of degree $\widehat{d} = d$ over the set $\mathcal{U}$. A sufficient condition would be if the polynomial admits a sum-of-squares (SOS) decomposition relative to $\mathcal{U}$, which is equivalent to checking the feasibility of a semidefinite-representable constraint system whose size grows exponentially in d. We refer the reader to Bertsimas et al. (2010) for a more detailed discussion about the SOS decomposition and its parameterization. When the corresponding polynomial in the semiinfinite constraint is of degree $\widehat{d} = 2$, then one can show that the resulting constraint system coincides with that from the approximate S lemma. To this end, let $Z_{\text{SOS}}^{\mathcal{P}_d}$ be the optimal value of the approximation when polynomial decision rules of degree d are employed. Then, we have $Z_{\text{SOS}}^{\mathcal{P}_1} = Z_{\text{AS}}^{\mathcal{L}}$ and $Z_{\text{SOS}}^{\mathcal{P}_2} = Z_{\text{AS}}^{\mathcal{Q}}$. Increasing the degree of the polynomial decision rules helps improve approximation quality at the expense of significant computational burden and numerical instability, even if we merely raise the degree by one (that is, when we employ quadratic decision rules for problems with nonfixed recourse or cubic decision rules for problems with fixed recourse).

The findings of this section culminate in the following theorem.

Theorem 4. *The following chains of inequalities hold*:

$$Z^{\mathcal{L}} \le Z_{\text{IA}}^{\mathcal{L}} \le Z_{\text{AS}}^{\mathcal{L}} = Z_{\text{SOS}}^{\mathcal{P}_1} \quad \textit{and} \quad Z^{\mathcal{Q}} \le Z_{\text{IA}}^{\mathcal{Q}} \le Z_{\text{AS}}^{\mathcal{Q}} = Z_{\text{SOS}}^{\mathcal{P}_2}.$$

3.2. Exact Reformulation

We identify two cases where the semidefinite-based approximations are equivalent to the respective copositive programs. First, in view of the *exact S* lemma, one can show that the inner approximation $\mathcal{IA}(\mathcal{K})$ coincides with $\mathcal{COP}(\mathcal{K})$ whenever the cone $\mathcal{K}$ in (15) is described by only a second-order cone constraint $\widehat{\boldsymbol{R}}\boldsymbol{u} \in \mathcal{SOC}(K_p)$.

Proposition 4 (*S* Lemma). *If* $\mathcal{K} = \{\boldsymbol{u} \in \mathbb{R}^{K+1} : \widehat{\boldsymbol{R}}\boldsymbol{u} \in \mathcal{SOC}(K_p)\}$, *then*

$$\mathcal{COP}(\mathcal{K}) = \mathcal{IA}(\mathcal{K}) = \mathcal{AS}(\mathcal{K}) := \{\boldsymbol{V} \in \mathcal{S}^{K+1} : \boldsymbol{V} \succeq \tau\widehat{\boldsymbol{S}},\, \tau \ge 0\},$$

where $\widehat{\boldsymbol{S}} \in \mathcal{S}^{K+1}$ *is defined as in* (17).

Another exactness result arises when linear constraints are present in $\mathcal{K}$, and they satisfy the following condition.

Assumption 3. *If* $\boldsymbol{u} \in \mathbb{R}^{K+1}$ *satisfies* $\widehat{\boldsymbol{R}}\boldsymbol{u} \in \mathcal{SOC}(K_p)$ *and* $\widehat{\boldsymbol{p}}_\ell^\top \boldsymbol{u} = 0$ *for some* $\ell \in [K_p]$, *then* $\boldsymbol{u} \in \mathcal{K}$.

The condition stipulates that the cone $\{\boldsymbol{u} \in \mathbb{R}^{K+1} : \widehat{\boldsymbol{R}}\boldsymbol{u} \in \mathcal{SOC}(K_p)\}$ must not contain points in the hyperplane $\widehat{\boldsymbol{p}}_\ell^\top \boldsymbol{u} = 0$ that do not belong to $\mathcal{K}$. Applying the restriction $u_{K+1} = 1$, we find that the implied uncertainty set for the primitive vector $(u_1, \ldots, u_K)^\top$ is given by an intersection of a ball and a polytope whose facets do not intersect within the ball.

Example 5. Consider the set

$$\mathcal{U} := \left\{ \boldsymbol{u} \in \mathbb{R}^2 \times \{1\} : u_1^2 + u_2^2 \le 1,\, u_1 \ge -\frac{1}{2},\, u_1 \le \frac{1}{2} \right\}.$$

The two lines $u_1 = -\frac{1}{2}$ and $u_1 = \frac{1}{2}$ do not intersect as they are parallel. Thus, Assumption 3 holds for this uncertainty set.

We state the second exactness result in the following proposition.

Proposition 5 (Theorem 5 in Burer 2015). *If Assumption 3 holds, then* $\mathcal{COP}(\mathcal{K}) = \mathcal{IA}(\mathcal{K})$.

We remark that this positive result holds only for the proposed inner approximation $\mathcal{IA}(\mathcal{K})$ and not for the cone $\mathcal{AS}(\mathcal{K})$, which is obtained from applying the approximate S lemma. Thus, in general, we may still have $\mathcal{AS}(\mathcal{K}) \subsetneq \mathcal{COP}(\mathcal{K})$.

We conclude the section with the following theorem regarding the exactness of the semidefinite programs.

Theorem 5. *If the cone $\mathcal{K}$ is given by $\{\boldsymbol{u} \in \mathbb{R}^{K+1} : \widehat{\boldsymbol{R}}\boldsymbol{u} \in \mathcal{SOC}(K_p)\}$ or if it satisfies Assumption 3, then $Z^{\mathcal{L}}_{\text{IA}} = Z^{\mathcal{L}}$ and $Z^{\mathcal{Q}}_{\text{IA}} = Z^{\mathcal{Q}}$.*

3.3. Approximation Quality of the Enhanced Decision Rule

We now restrict our study to the case of two-stage robust optimization problems with fixed recourse and with piecewise linear decision rules. In this setting, linear programming approximations have been proposed for the decision rule problems (Chen and Zhang 2009, Georghiou et al. 2015). If, in addition, the uncertainty set $\mathcal{U}$ is given by a hyperrectangle and each folding direction $\boldsymbol{g}_\ell$ is aligned with a coordinate axis, then these linear programs become exact (Georghiou et al. 2015). Unfortunately, for generic uncertainty sets, the resulting approximation can sometimes be of poor quality.

Example 6 (Partition Problem). Consider the following instance of the IP feasibility problem (Example 4), which corresponds to the NP-hard partition problem. Given an input vector $\boldsymbol{c} \in \mathbb{N}^K$, the problem asks if one can partition the components of $\boldsymbol{c}$ into two sets so that both sets have an equal sum. We can reduce this problem to the instance of the IP feasibility problem that seeks for a binary vector $\boldsymbol{u} \in \{-1,1\}^K$ within the polytope $\mathcal{U} = \{\boldsymbol{u} \in [-1,1]^K : \boldsymbol{c}^\top \boldsymbol{u} = 0\}$. If a partition exists, then the components of $\boldsymbol{u}$ will denote the indicator function of the two sets. For example, if $\boldsymbol{c} = (1,2,3)^\top$, then the possible solutions are $\boldsymbol{u} = (1,1,-1)^\top$ or $\boldsymbol{u} = (-1,-1,1)^\top$. On the other hand, if $\boldsymbol{c} = (2,2,3)^\top$, then no such solution exists, and necessarily, the optimal value of the corresponding norm maximization problem is strictly less than $K = 3$. In particular, one can show that the optimal value is 2.5, which is attained by the solution $\boldsymbol{u} = (0.5,1,1)^\top$.

For the input $\boldsymbol{c} = (2,2,3)^\top$, the best piecewise linear decision rule approximation in the literature yields a conservative upper bound of three, which fails to certify the nonexistence of binary solutions. On the other hand, the semidefinite programming approximation of the equivalent copositive program yields a tighter upper bound of 2.54 and thus, provides a correct certificate. As the corresponding two-stage problem has fixed recourse, our scheme allows us to utilize quadratic decision rules. In this case, the resulting semidefinite program yields the best optimal value of 2.5.

The example highlights the surprising fact that, even for seemingly trivial low-dimensional problem instances, one necessarily has to go through the copositive programming route in order to obtain a satisfactory approximation for the piecewise decision rule problem.

We now formally establish that the semidefinite programming approximation obtained from applying piecewise linear decision rules is never inferior to the state-of-the-art scheme by Georghiou et al. (2015). In the following, we briefly discuss their setting and formulate the corresponding lifted uncertainty set $\mathcal{U}'$. For cleaner exposition, we primarily consider the setting of piecewise linear decision rules with axial segmentation where each folding direction is aligned with a coordinate axis. We remark that all results extend to the case with general segmentation, albeit at the expense of more cumbersome notation (see section 4.2 of Georghiou et al. 2015). To this end, let the interval $[\underline{u}_k, \overline{u}_k]$ be the marginal support of the kth uncertain parameter. For each coordinate axis u_k, we generate L piecewise linear mappings in view of the prescribed break points $h_{k,1} = \underline{u}_k < h_{k,2} < \cdots < h_{k,L} < \overline{u}_k$, as follows:

$$\tilde{F}_{k,l}(\boldsymbol{u}) = \max\{0, u_k - h_{k,\ell}\} - \max\{0, u_k - h_{k,\ell+1}\} \quad \forall \ell \in [L]. \tag{19}$$

To simplify the notation, we assume that there are exactly L mappings for each coordinate axis. Such a construction leads to the lifted uncertainty set

$$\mathcal{U}' := \{(\boldsymbol{w}, \boldsymbol{u}) \in \mathbb{R}^{KL} \times \mathcal{U} : w_{k,\ell} = \tilde{F}_{k,\ell}(\boldsymbol{u}) \qquad \forall k \in [K]\ \ell \in [L]\}. \tag{20}$$

Note that each mapping in (19) can be defined through the difference $\tilde{F}_{k,l}(\boldsymbol{u}) = F_{k,l}(\boldsymbol{u}) - F_{k,l+1}(\boldsymbol{u})$, where the functions $F_{k,l}(\boldsymbol{u}) = \max\{0, \boldsymbol{f}_{k,\ell}^\top \boldsymbol{u}\}$, $\ell \in [L]$, assume the standard form described in (12), with $\boldsymbol{f}_{k,\ell} = (\mathbf{e}_k, -h_{k,\ell})$, $\ell \in [L]$. By our construction of $\mathcal{U}'$, we can further impose that $F_{k,1}(\boldsymbol{u}) = u_k - \underline{u}_k$ and $F_{k,L+1}(\boldsymbol{u}) = 0$.

Using Theorem 3, the lifted set in (20) can be reformulated as

$$\mathcal{U}' = \left\{ (\boldsymbol{w}, \boldsymbol{u}) \in \mathbb{R}^{KL} \times \mathcal{U} : \begin{array}{l} \boldsymbol{z} \in \mathbb{R}_+^{K(L+1)} \\ w_{k,\ell} = z_{k,\ell} - z_{k,\ell+1} \\ \qquad \forall k \in [K]\ \ell \in [L] \\ z_{k,1} = u_k - \underline{u}_k,\ z_{k,L+1} = 0 \\ \qquad \forall k \in [K] \\ z_{k,\ell} \geq u_k - h_{k,\ell},\ \overline{u}_k \geq z_{k,\ell} \\ \qquad \forall k \in [K]\ \ell \in [L+1] \\ z_{k,\ell}(z_{k,\ell} - u_k + h_{k,\ell}) = 0 \\ \qquad \forall k \in [K]\ \ell \in [L+1] \end{array} \right\}. \tag{21}$$

In view of our discussion in Section 2.4, an equivalent copositive program can thus be derived for the piecewise linear decision rule Problem ($\mathcal{PL}$). We denote by $Z_{\text{IA}}^{\mathcal{PL}}$ the optimal value of the corresponding semidefinite programming approximation. Alternatively, in Georghiou et al. (2015), a tractable outer approximation of $\mathcal{U}'$ is derived as follows:

$$\mathcal{U}^{**} = \left\{ (\boldsymbol{w},\boldsymbol{u}) \in \mathbb{R}^{KL} \times \mathcal{U} : \begin{array}{ll} u_k - \underline{u}_k = \sum_{\ell\in[L]} w_{k,\ell} & \forall k \in [K] \\ h_{k,2} - \underline{u}_k \geq w_{k,1} & \forall k \in [K] \\ (h_{k,\ell+1} - h_{k,\ell}) w_{k,\ell-1} \geq (h_{k,\ell} - h_{k,\ell-1}) w_{k,\ell} & \\ & \forall k \in [K] \ell \in [L] \setminus \{1\} \end{array} \right\}. \tag{22}$$

By replacing the set $\mathcal{U}'$ with $\mathcal{U}^{**}$ in (2.4), one can obtain a linear decision rule approximation problem with a polyhedral uncertainty set, which can be reformulated to a tractable linear program if the recourse matrix $\mathcal{B}'(v)$ is fixed. We denote by $Z_{\text{GWK}}^{\mathcal{PL}}$ its optimal value. We now examine the relation between $Z_{\text{IA}}^{\mathcal{PL}}$ and $Z_{\text{GWK}}^{\mathcal{PL}}$. To this end, by using the copositive programming techniques, we first propose a looser outer approximation $\mathcal{U}^*$ of the lifted set $\mathcal{U}'$ and establish that the set is still tighter than $\mathcal{U}^{**}$. We define this outer approximation as

$$\mathcal{U}^* = \left\{ \begin{array}{l} (\boldsymbol{w},\boldsymbol{u}) \\ \in \mathbb{R}^{KL} \times \mathcal{U} : \end{array} \begin{array}{ll} \boldsymbol{z} \in \mathbb{R}_+^{K(L+1)} & \\ \boldsymbol{Z}^{k,\ell} \in \mathcal{S}^3, \boldsymbol{Z}^{k,\ell} \succeq \boldsymbol{0} & \forall k \in [K] \ell \in [L] \\ w_{k,\ell} = z_{k,\ell} - z_{k,\ell+1} & \forall k \in [K] \ell \in [L] \\ z_{k,1} = u_k - \underline{u}_k, z_{k,L+1} = 0 & \forall k \in [K] \\ z_{k,\ell} \geq u_k - h_{k,\ell}, \overline{u}_k \geq z_{k,\ell} & \forall k \in [K] \ell \in [L+1] \\ Z_{1,3}^k - Z_{3,3}^k + z_{k,\ell+1}(h_{k,\ell-1} - h_{k,\ell+1}) \geq 0 & \\ & \forall k \in [K] \ell \in [L] \\ Z_{2,3}^k - Z_{3,3}^k + z_{k,\ell+1}(h_{k,\ell} - h_{k,\ell+1}) \geq 0 & \\ & \forall k \in [K] \ell \in [L] \\ Z_{1,3}^k - Z_{1,1}^k + z_{k,\ell-1}(h_{k,\ell+1} - h_{k,\ell-1}) & \\ \quad + Z_{2,2}^k - Z_{2,3}^k + z_{k,\ell}(h_{k,\ell} - h_{k,\ell+1}) \geq 0 & \\ & \forall k \in [K] \ell \in [L] \\ Z_{3,3}^k - Z_{1,3}^k + z_{k,\ell+1}(h_{k,\ell+1} - h_{k,\ell-1}) & \\ \quad + Z_{1,2}^k - Z_{2,2}^k + z_{k,\ell}(h_{k,\ell-1} - h_{k,\ell}) \geq 0 & \\ & \forall k \in [K] \ell \in [L] \\ Z_{2,3}^k - Z_{2,2}^k + z_{k,\ell}(h_{k,\ell+1} - h_{k,\ell}) \geq 0 & \\ & \forall k \in [K] \ell \in [L] \end{array} \right\},$$

which is a concise set involving $\mathcal{O}(KL)$ semidefinite constraints of size 3×3. The following proposition describes the chain relation of $\mathcal{U}'$, $\mathcal{U}^*$, and $\mathcal{U}^{**}$.

Proposition 6. *We have* $\mathcal{U}' \subseteq \mathcal{U}^* \subseteq \mathcal{U}^{**}$.

Proof. See the e-companion. □

Finally, we are ready to state the main result of this section in the following theorem.

Theorem 6. *We have* $Z_{\text{IA}}^{\mathcal{PL}} \leq Z_{\text{GWK}}^{\mathcal{PL}}$.

Proof. See the e-companion. □

The proof of Theorem 6 imparts the favorable insight that a tighter approximation can already be obtained by considering a concise set involving $\mathcal{O}(KL)$ semidefinite constraints of size 3×3.

4. Copositive Reformulation for Multistage Decision Rule Problems

We now extend the proposed copositive programming approach to multistage robust optimization problems of the following generic form:

$$\begin{array}{ll} \inf & \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u}\in\mathcal{U}} \sum_{t=1}^{T} \boldsymbol{d}_t(\boldsymbol{u}^t)^\top \boldsymbol{y}_t(\boldsymbol{u}^t) \\ \text{s.t.} & \mathcal{A}(\boldsymbol{u}^1)\boldsymbol{x} + \sum_{t=1}^{T} \mathcal{B}_t(\boldsymbol{u}^t)\boldsymbol{y}_t(\boldsymbol{u}^t) \geq \boldsymbol{h}(\boldsymbol{u}) \quad \forall \boldsymbol{u} \in \mathcal{U} \\ & \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{y}_t \in \mathscr{F}_{K^t+1, N_t} \ \ \forall t \in [T]. \end{array} \tag{23}$$

The vector $\boldsymbol{u}^t$ in (23) collects the history of observations up to time t, and it is defined as

$$\boldsymbol{u}^t = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t, 1) \in \mathbb{R}^{K^t+1},$$

where $\boldsymbol{u}_t \in \mathbb{R}^{K_t}$ contains uncertain parameters observed at time $t \in [T]$ and $K^t := \sum_{s=1}^{t} K_s$. Here, we have appended the constant scalar 1 at the end of the vector so that affine functions in $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$ can be represented as linear functions in $\boldsymbol{u}^t$, whereas quadratic functions in $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$ can be formulated compactly in a homogenized manner. We set the vector of all uncertain parameters in (23) to $\boldsymbol{u} := \boldsymbol{u}^T \in \mathbb{R}^{K+1}$, with $K = K^T$. As in the two-stage setting, the problem parameters $\mathcal{A}(\boldsymbol{u}^1)$, $\mathcal{B}_t(\boldsymbol{u}^t)$, $\boldsymbol{d}_t(\boldsymbol{u}^t)$, and $\boldsymbol{h}(\boldsymbol{u})$ are described by linear functions in their respective arguments as follows:

$$\mathcal{A}(\boldsymbol{u}^1) := \sum_{k=1}^{K^1+1} [\boldsymbol{u}^1]_k \widehat{\boldsymbol{A}}_k, \ \mathcal{B}_t(\boldsymbol{u}^t) := \sum_{k=1}^{K^t+1} [\boldsymbol{u}^t]_k \widehat{\boldsymbol{B}}_{k,t},$$
$$\boldsymbol{d}_t(\boldsymbol{u}^t) := \widehat{\boldsymbol{D}}_t \boldsymbol{u}^t, \ \boldsymbol{h}(\boldsymbol{u}) := \widehat{\boldsymbol{H}} \boldsymbol{u},$$

where $\widehat{\boldsymbol{A}}_k \in \mathbb{R}^{J\times M}$, $\widehat{\boldsymbol{B}}_{k,t} \in \mathbb{R}^{J\times N_t}$, $\widehat{\boldsymbol{D}}_t := (\widehat{\boldsymbol{d}}_{1,t}, \ldots, \widehat{\boldsymbol{d}}_{N_t,t})^\top \in \mathbb{R}^{N_t\times(K^t+1)}$, and $\widehat{\boldsymbol{H}} := (\widehat{\boldsymbol{h}}_1, \ldots, \widehat{\boldsymbol{h}}_J)^\top \in \mathbb{R}^{J\times(K+1)}$ are deterministic data.

The decision vector $\boldsymbol{y}_t(\boldsymbol{u}^t) \in \mathbb{R}^{N_t}$ in (23) is chosen after the realization of uncertain parameters up to time t but before the revelation of future outcomes $\{\boldsymbol{u}_s\}_{s\in[t+1,T]}$. The objective of Problem (23) is to find a here-and-now decision $\boldsymbol{x} \in \mathcal{X}$ and a sequence of nonanticipative decision rules $\{\boldsymbol{y}_t(\cdot)\}_{t\in[T]}$ that are feasible to the semiinfinite constraint in (23) and minimize the total cost $\boldsymbol{c}^\top\boldsymbol{x} + \sup_{\boldsymbol{u}\in\mathcal{U}} \sum_{t=1}^{T} \boldsymbol{d}_t(\boldsymbol{u}^t)^\top \boldsymbol{y}_t(\boldsymbol{u}^t)$. Problem (23) constitutes an extension of the two-stage Problem (1) to the multistage setting, and as such, it is computationally challenging to solve. To this end, we endeavor to derive copositive programming reformulations in view of linear and quadratic decision rules. Tractable semidefinite programming approximations can then be derived using the techniques discussed in Section 3. One can further enhance these approximations by utilizing the piecewise linear and piecewise quadratic decision rules discussed in Section 2.4.

As in the two-stage setting, we assume that the uncertainty set $\mathcal{U}$ is defined as in (2) and satisfies both Assumptions 1 and 2. In the following, we use the linear truncation operator $\mathbf{\Pi}_t : \mathbb{R}^{K+1} \longmapsto \mathbb{R}^{K^t+1}$ that satisfies

$$\mathbf{\Pi}_t \boldsymbol{u} = \boldsymbol{u}^t \quad \forall \boldsymbol{u} \in \mathbb{R}^{K+1}.$$

We first examine the case when the multistage robust optimization problem has nonfixed recourse. Here, we apply the linear decision rules

$$\boldsymbol{y}_t(\boldsymbol{u}^t) = \boldsymbol{Y}_t \boldsymbol{u}^t = \boldsymbol{Y}_t \mathbf{\Pi}_t \boldsymbol{u},$$

for some coefficient matrix $\boldsymbol{Y}_t \in \mathbb{R}^{N_t \times (K^t+1)}$. This yields the following conservative approximation of Problem (23):

$$\begin{aligned} Z^{\mathcal{ML}} = \ & \inf \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u} \in \mathcal{U}} \sum_{t=1}^{T} \boldsymbol{d}_t(\boldsymbol{u}^t)^\top \boldsymbol{Y}_t \mathbf{\Pi}_t \boldsymbol{u} \\ & \text{s.t. } \mathcal{A}(\boldsymbol{u}^1)\boldsymbol{x} + \sum_{t=1}^{T} \mathcal{B}_t(\boldsymbol{u}^t) \boldsymbol{Y}_t \mathbf{\Pi}_t \boldsymbol{u} \geq \boldsymbol{h}(\boldsymbol{u}) \quad \forall \boldsymbol{u} \in \mathcal{U} \\ & \qquad \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{Y}_t \in \mathbb{R}^{N_t \times (K^t+1)} \qquad \forall t \in [T]. \end{aligned} \qquad (\mathcal{ML})$$

Problem ($\mathcal{ML}$) shares the same structure as its two-stage counterpart ($\mathcal{L}$). Hence, by employing the same reformulation techniques described in Section 2.2, we can derive a polynomial-size copositive program for the problem. For notational convenience, in the following, we define the matrices

$$\widehat{\mathbf{\Theta}}_j := \begin{pmatrix} \mathbf{e}_j^\top \widehat{\boldsymbol{A}}_1 \\ \vdots \\ \mathbf{e}_j^\top \widehat{\boldsymbol{A}}_{K^1+1} \end{pmatrix} \in \mathbb{R}^{(K^1+1)\times M},$$

$$\widehat{\mathbf{\Lambda}}_{j,t} := \begin{pmatrix} \mathbf{e}_j^\top \widehat{\boldsymbol{B}}_{1,t} \\ \vdots \\ \mathbf{e}_j^\top \widehat{\boldsymbol{B}}_{K^t+1,t} \end{pmatrix} \in \mathbb{R}^{(K^t+1)\times N_t} \qquad \forall t \in [T] \ \ \forall j \in [J],$$

and we define the affine functions

$$\begin{aligned} \mathbf{\Omega}_j(\boldsymbol{x}, \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_T) := & \frac{1}{2} \mathbf{\Pi}_1^\top (\widehat{\mathbf{\Theta}}_j \boldsymbol{x} \mathbf{e}_{K^1+1}^\top + \mathbf{e}_{K^1+1} \boldsymbol{x}^\top \widehat{\mathbf{\Theta}}_j^\top) \mathbf{\Pi}_1 \\ & + \frac{1}{2} \sum_{t=1}^{T} \mathbf{\Pi}_t^\top (\widehat{\mathbf{\Lambda}}_{j,t} \boldsymbol{Y}_t + \boldsymbol{Y}_t^\top \widehat{\mathbf{\Lambda}}_{j,t}^\top) \mathbf{\Pi}_t \\ & - \frac{1}{2} (\widehat{\boldsymbol{h}}_j \mathbf{e}_{K+1}^\top + \mathbf{e}_{K+1} \widehat{\boldsymbol{h}}_j^\top) \qquad \forall j \in [J]. \end{aligned}$$

The equivalent reformulation is provided in the following theorem. We omit the proof as it closely follows that of Theorem 1.

Theorem 7. *Problem* ($\mathcal{ML}$) *is equivalent to the following copositive program*:

$$\begin{aligned} Z^{\mathcal{ML}} = \ & \inf \boldsymbol{c}^\top \boldsymbol{x} + \lambda \\ & \text{s.t. } \lambda\, \mathbf{e}_{K+1} \mathbf{e}_{K+1}^\top - \frac{1}{2} \sum_{t=1}^{T} \mathbf{\Pi}_t^\top (\widehat{\mathbf{D}}_t^\top \boldsymbol{Y}_t + \boldsymbol{Y}_t^\top \widehat{\mathbf{D}}_t) \mathbf{\Pi}_t \\ & \qquad + \sum_{i=1}^{I} \alpha_i \widehat{\boldsymbol{C}}_i \in \mathcal{COP}(\mathcal{K}) \\ & \qquad \mathbf{\Omega}_j(\boldsymbol{x}, \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_T) - \pi_j \mathbf{e}_{K+1} \mathbf{e}_{K+1}^\top \\ & \qquad - \sum_{i=1}^{I} \mathbf{e}_i^\top \boldsymbol{\beta}_j \widehat{\boldsymbol{C}}_i \in \mathcal{COP}(\mathcal{K}) \quad \forall j \in [J] \\ & \qquad \lambda \in \mathbb{R},\ \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{\alpha} \in \mathbb{R}^I,\ \boldsymbol{\pi} \in \mathbb{R}_+^J,\ \boldsymbol{\beta}_j \in \mathbb{R}^I \\ & \qquad \qquad \forall j \in [J],\ \boldsymbol{Y}_t \in \mathbb{R}^{N_t \times (K^t+1)} \quad \forall t \in [T]. \end{aligned} \qquad (24)$$

Next, we consider the case when the multistage problem has fixed recourse (i.e.,

$$\boldsymbol{d}_t(\boldsymbol{u}^t) = \widehat{\boldsymbol{d}}_t \ \text{ and } \ \mathcal{B}_t(\boldsymbol{u}^t) = \widehat{\boldsymbol{B}}_t \qquad \forall \boldsymbol{u}^t \in \mathbb{R}^{K^t+1} \quad \forall t \in [T],$$

where $\widehat{\boldsymbol{d}}_t \in \mathbb{R}^{N_t}$ and $\widehat{\boldsymbol{B}} \in \mathbb{R}^{J \times N_t}$ are the deterministic vector and matrix, respectively). Here, we can apply the quadratic decision rules

$$[\boldsymbol{y}(\boldsymbol{u}^t)]_{n_t} = (\boldsymbol{u}^t)^\top \boldsymbol{Q}_{n_t,t} \boldsymbol{u}^t = (\mathbf{\Pi}^t \boldsymbol{u})^\top \boldsymbol{Q}_{n_t,t} \mathbf{\Pi}^t \boldsymbol{u} \qquad \forall n_t \in [N_t],$$

for some coefficient matrices $\boldsymbol{Q}_{n_t,t} \in \mathcal{S}^{K^t+1}$, $n_t \in [N_t]$, $t \in [T]$. This yields the following conservative approximation of Problem (23):

$$\begin{aligned} Z^{\mathcal{MQ}} = & \inf \boldsymbol{c}^\top \boldsymbol{x} + \sup_{\boldsymbol{u} \in \mathcal{U}} \sum_{t=1}^{T} \sum_{n_t=1}^{N_t} \widehat{d}_{n_t,t} (\mathbf{\Pi}^t \boldsymbol{u})^\top \boldsymbol{Q}_{n_t,t} \mathbf{\Pi}_t \boldsymbol{u} \\ & \text{s.t. } (\mathbf{\Pi}_1 \boldsymbol{u})^\top \widehat{\mathbf{\Theta}}_j \boldsymbol{x} + \sum_{t=1}^{T} \sum_{n_t=1}^{N_t} \left(b_{j,n_t} (\mathbf{\Pi}_t \boldsymbol{u})^\top \boldsymbol{Q}_{n_t,t} \mathbf{\Pi}_t \boldsymbol{u} \right) \\ & \qquad \geq \boldsymbol{h}(\boldsymbol{u}) \ \ \forall \boldsymbol{u} \in \mathcal{U} \\ & \qquad \boldsymbol{x} \in \mathcal{X}, \boldsymbol{Q}_{n_t,t} \in \mathcal{S}^{K^t+1} \ \ \forall t \in [T]\ \forall n_t \in [N_t]. \end{aligned} \qquad (\mathcal{MQ})$$

Problem ($\mathcal{ML}$) shares the same structure as its two-stage counterpart ($\mathcal{Q}$), which indicates that it is also amenable to an equivalent copositive programming reformulation. To this end, we define the affine functions

$$\begin{aligned} \Gamma_j(\boldsymbol{x}, \boldsymbol{Q}_{1,1} \ldots, \boldsymbol{Q}_{N_T,T}) := & \frac{1}{2} \mathbf{\Pi}_1^\top (\widehat{\mathbf{\Theta}}_j \boldsymbol{x} \mathbf{e}_{K^1+1}^\top + \mathbf{e}_{K^1+1} \boldsymbol{x}^\top \widehat{\mathbf{\Theta}}_j^\top) \mathbf{\Pi}_1 \\ & - \frac{1}{2} (\mathbf{e}_{K+1} \widehat{\boldsymbol{h}}_j^\top - \widehat{\boldsymbol{h}}_j \mathbf{e}_{K+1}^\top) + \sum_{t=1}^{T} \sum_{n_t=1}^{N_t} \widehat{b}_{j,n_t} \mathbf{\Pi}_t^\top \\ & \boldsymbol{Q}_{n_t,t} \mathbf{\Pi}_t \qquad \forall j \in [J]. \end{aligned}$$

The equivalent reformulation is provided in the following theorem, whose proof is omitted as it closely follows that of Theorem 2.

Theorem 8. *Problem* ($\mathcal{MQ}$) *is equivalent to the following copositive program*:

$$\begin{aligned} Z^{\mathcal{MQ}} = \min\ & \mathbf{c}^\top \mathbf{x} + \lambda \\ \text{s.t.}\ & \lambda\, \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top - \sum_{t=1}^{T}\sum_{n_t=1}^{N_t} [\widehat{d}_{n_t,t}\mathbf{\Pi}_t^\top \mathbf{Q}_{n_t,t}\mathbf{\Pi}_t] \\ & + \sum_{i=1}^{I}\alpha_i \widehat{\mathbf{C}}_i \in \mathcal{COP}(\mathcal{K}) \\ & \mathbf{\Gamma}_j(\mathbf{x}, \mathbf{Q}_{1,1}\ldots, \mathbf{Q}_{N_T,T}) - \pi_j \mathbf{e}_{K+1}\mathbf{e}_{K+1}^\top \\ & - \sum_{i=1}^{I}[\mathbf{e}_i^\top \boldsymbol{\beta}_j]\widehat{\mathbf{C}}_i \in \mathcal{COP}(\mathcal{K}) \quad \forall j \in [J] \\ & \lambda \in \mathbb{R}, \mathbf{x} \in \mathcal{X}, \boldsymbol{\alpha} \in \mathbb{R}^I, \boldsymbol{\pi} \in \mathbb{R}_+^J, \boldsymbol{\beta}_j \in \mathbb{R}^I \quad \forall j \in [J] \\ & \mathbf{Q}_{n_t,t} \in \mathcal{S}^{K^t+1} \qquad \forall t \in [T]\ \forall n_t \in [N_t]. \end{aligned} \tag{25}$$

Remark 2. In some multistage robust optimization problems, we may observe that some of the recourse decision variables are multiplied with uncertain parameters, whereas the remaining recourse decisions are multiplied with deterministic terms. In such situations, we can apply quadratic decision rules to the latter, which yields stronger decision rule approximations. With minimum modification, we can reformulate the decision rule problem into an equivalent copositive program similar to (25). We omit the detailed reformulation here.

5. Numerical Experiments

In this section, we assess the effectiveness of our copositive programming approach over three applications in operations management. The first example is a multiitem newsvendor problem, which can be reformulated to a two-stage robust optimization problem with fixed recourse. The following two examples are inventory control and index tracking problems, which correspond to multistage robust optimization problems with nonfixed recourse. All optimization problems are solved using MOSEK 8.1.0.56 (ApS 2016) via the YALMIP interface (Lofberg 2004) on a 16-core 3.4-GHz Linux PC with 32 GB of RAM. The codes for these three examples are available at https://github.com/guxu-iowa/OR-MSRO.

5.1. Multiitem Newsvendor

We consider the following robust multiitem newsvendor problem studied in Ardestani-Jaafari and Delage (2021):

$$\max_{\mathbf{x}\ge \mathbf{0}} \min_{\boldsymbol{\xi}\in\Xi} \sum_{n=1}^{N} (r_n \min(x_n, \xi_n) - c_n x_n - s_n \max(\xi_n - x_n, 0)). \tag{26}$$

Here, N represents the number of products; $\mathbf{x}$ is the vector of order quantities; $\boldsymbol{\xi}$ is the vector of uncertain demands; and $\mathbf{r}$, $\mathbf{c}$, and $\mathbf{s}$ are the vectors of sales prices, order costs, and shortage costs, respectively. We assume that the products do not have a salvage value, and the salvage value is set to zero. Problem (26) can be reformulated as the two-stage robust optimization problem given by

$$\begin{aligned} \max_{\mathbf{x}, \mathbf{y}(\cdot)}\ \min_{\boldsymbol{\xi}\in\Xi}\ & \sum_{n=1}^{N} y_n(\boldsymbol{\xi}) \\ \text{s.t.}\ & y_n(\boldsymbol{\xi}) \le (r_n - c_n)x_n - r_n(x_n - \xi_n) \quad \boldsymbol{\xi}\in\Xi,\ \forall n \in [N] \\ & y_n(\boldsymbol{\xi}) \le (r_n - c_n)x_n - s_n(\xi_n - x_n) \quad \boldsymbol{\xi}\in\Xi,\ \forall n \in [N] \\ & \mathbf{x} \ge \mathbf{0}. \end{aligned} \tag{27}$$

In this problem, the uncertainty set is specified through a factor model defined as

$$\Xi := \left\{ \boldsymbol{\xi} \in \mathbb{R}^N : \begin{array}{l} \boldsymbol{\xi} = \overline{\boldsymbol{\xi}} + \mathrm{Diag}(\widehat{\boldsymbol{\xi}})\mathbf{F}\boldsymbol{\zeta}, \\ \boldsymbol{\zeta} \in \mathbb{R}^N, \|\boldsymbol{\zeta}\|_\infty \le 1, \|\boldsymbol{\zeta}\|_1 \le \rho \end{array} \right\},$$

where $\boldsymbol{\zeta}$ is a vector comprising all factors, $\mathbf{F} \in \mathbb{R}^{N\times N}$ is the factor loading matrix, and $\rho < N$ is a scalar that controls the level of conservativeness. The associated cone $\mathcal{K}$ related to this uncertainty set is written as

$$\mathcal{K} := \left\{ (\boldsymbol{\xi}, \tau) \in \mathbb{R}^N \times \mathbb{R}_+ : \begin{array}{l} \boldsymbol{\xi} = \overline{\boldsymbol{\xi}}\tau + \mathrm{Diag}(\widehat{\boldsymbol{\xi}})\mathbf{F}\boldsymbol{\zeta}, \\ \boldsymbol{\zeta} \in \mathbb{R}^N, \|\boldsymbol{\zeta}\|_\infty \le \tau, \|\boldsymbol{\zeta}\|_1 \le \rho\tau \end{array} \right\}.$$

As the problem has fixed recourse, we can apply the QDR scheme proposed in Section 2.3 and solve the semidefinite approximation, which results from replacing the copositive cone $\mathcal{COP}(\mathcal{K})$ with the inner approximation $\mathcal{IA}(\mathcal{K})$ defined in (16). We compare our QDR scheme with the one proposed by Ben-Tal et al. (2009) (BGGN), where we replace the cone $\mathcal{COP}(\mathcal{K})$ with the inner approximation $\mathcal{AS}(\mathcal{K})$ defined in (18), with the polynomial decision rule scheme of degree 3 (PDR3), and with the piecewise linear decision rule scheme proposed by Georghiou et al. (2015) (GWK). In addition, we also compare our method with state-of-the-art schemes for two-stage robust optimization problems with fixed recourse: the method COP described in Xu and Burer (2018) and the method AJD described in Ardestani-Jaafari and Delage (2021). We note that these two methods generate the same solutions with comparable computational times.

All experimental results are averaged over 100 random instances. We utilize the mechanism in Ardestani-Jaafari and Delage (2021) to set up the parameters and to generate the random instances. For each instance, we consider $n = 5$ items and set $\mathbf{r} = 80\mathbf{e}$ and $\mathbf{p} = 60\mathbf{e}$. We further sample the vector $\mathbf{c}$ uniformly at random from the hypercube $[40, 60]^5$. For the uncertainty set, we set $\rho = 4$ and $\overline{\boldsymbol{\xi}} = 60\mathbf{e}$, whereas the vector $\widehat{\boldsymbol{\xi}}$ is generated uniformly at random from $[50, 60]^5$. We sample each entry of the matrix $\mathbf{F}$ uniformly from $[-1, 1]$ and normalize each row so that its sum is equal to one. Table 1

Table 1. Relative Gaps (in Percentages) Between the Alternative Approximation Schemes and QDR

	Approximation method				
Statistic	BGGN	GWK	COP	AJD	PDR3
10th percentile	26.5	26.5	2.3	2.3	0
Mean	52.0	52.0	6.0	6.0	0
90th percentile	87.3	87.3	9.7	9.7	0

reports several statistics of relative gaps between the optimal value of QDR and those of the other alternative methods. We find that QDR provides a substantial average improvement of 52% over BGGN and an average improvement of 22.3% over GWK. Rather surprisingly, we also find that QDR outperforms the state-of-the-art COP and AJD schemes by 6%. Table 1 indicates that QDR generates the same performance as the less tractable PDR3. Table 2 reports the average computation times of the four methods. We observe that QDR can be solved as fast as BGGN, GWK, COP, and AJD, whereas it takes 40 times as long to solve PDR3. In summary, we may thus conclude that QDR provides high-quality solutions efficiently.

Remark 3. Because COP corresponds to a semidefinite programming approximation of the exact copositive reformulation of the newsvendor problem, it is indeed surprising that QDR can outperform COP. For the temporal network example described in Xu and Burer (2018) where the uncertainty set is given by a one-norm ball, one can formally prove that QDR performs better than COP. In general, however, we cannot prove that one approximation is tighter than the other or vice versa.

We also assess the quality of the first-stage decisions (order quantities) obtained from the different approximation methods by evaluating their true worst-case profits. Because the profit function in (26) is concave, the worst-case profit of any fixed decision occurs at a demand scenario from an extreme point of the uncertainty set Ξ. Thus, we can enumerate all extreme points of the uncertainty set and find the one that minimizes the profit to determine the worst-case scenario profit of each first-stage decision. In general, it is computationally prohibitive to enumerate the extreme points of a polyhedral set. However, it is manageable for our case because there are only a few variables and constraints involved. Table 3 reports the statistics of relative gaps between the worst-case scenario profit of our method and those of other methods. We find that the proposed QDR scheme provides substantial average improvements of 36.3%,

Table 2. The Average Computation Times (in Seconds) of the Different Approximation Schemes

	BGGN	GWK	COP	AJD	QDR	PDR3
Time	1.68	1.75	1.61	1.59	1.62	62.17

Table 3. Relative Gaps (in Percentages) of the Worst-Case Scenario Profits Between the Alternative Approximation Schemes and QDR

	Approximation method				
Statistic	BGGN	GWK	COP	AJD	PDR3
10th percentile	14.5	14.5	2.1	2.1	0
Mean	36.3	36.3	4.7	4.7	0
90th percentile	76.2	76.2	9.3	9.3	0

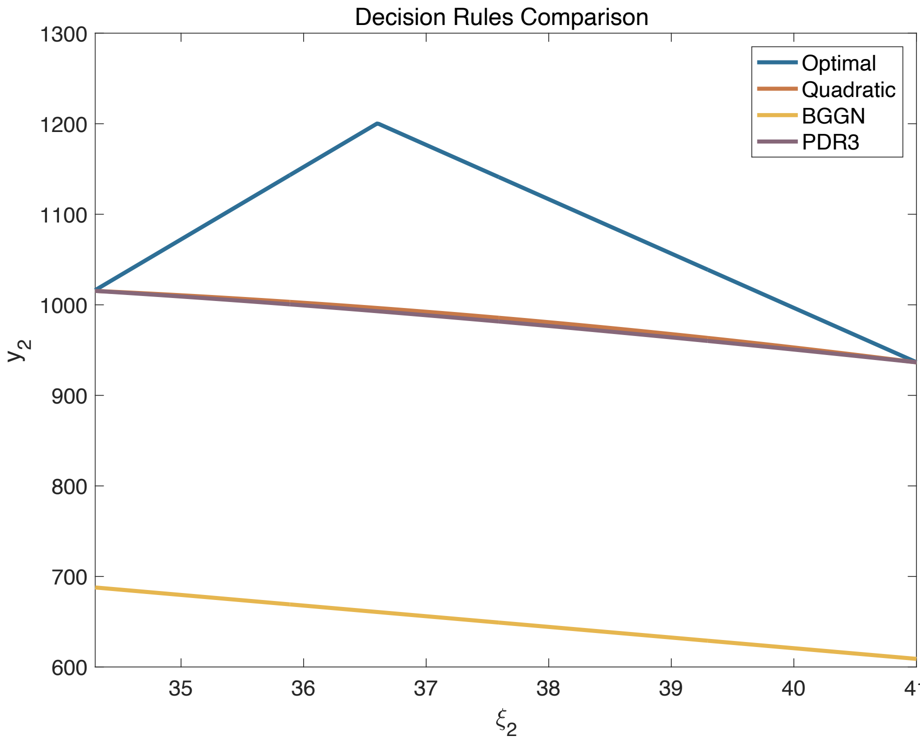
36.3%, 4.7%, and 4.7% over BGGN, GWK, COP, and AJD, respectively.

Finally, we analyze the optimal decision rules from the different approximation methods by considering a two-item instance of the robust newsvendor problems. Figure 1 visualizes the decision rules from the different methods as a function of the demand ξ_2 of the second item. We observe that the quadratic function from QDR and the polynomial function from PDR3 coincide with the optimal decision rule (optimal) at the extreme points of the uncertainty set. This implies that QDR and PDR3 can generate optimal order quantities as their decision rules anticipate the worst-case demand scenarios on par with the optimal one. On the other hand, BGGN generates suboptimal order quantities as its decision rule function does not coincide with the extremes of the optimal decision rule. We note that the COP method does not generate any decision rules. We further remark that GWK and BGGN return the same function, and thus, we only plot the one from BGGN.

5.2. Inventory Control

We next consider a multistage robust inventory control problem with multiple products and backlogging. A stochastic programming version of the problem is described in Georghiou et al. (2015). In this problem, we must determine the sales and order policies that maximize the worst-case profit over a planning horizon of T time stages. At the beginning of each time stage t, we observe a vector of risk factors $\boldsymbol{\xi}_t$ that explains the uncertainty in the current demand $D_{t,p}(\boldsymbol{\xi}_t)$ and the unit sales price $R_{t,p}(\boldsymbol{\xi}_t)$ of each product $p \in [P]$. After $\boldsymbol{\xi}_t$ is revealed at time stage t, we must determine the quantity $s_{t,p}$ of product p to sell at the current price, the amount $o_{t,p}$ of product p to replenish the inventory, and the amount $b_{t,p}$ of product p to backlog to the next time stage at the unit cost C_b. The sales $s_{t,p}$ of product p at time stage t can only be provided by orders placed at time stage $t-1$ or earlier. We denote the inventory level at the beginning of each time stage t by I_t. For simplicity, we assume that one unit of each product occupies the same amount of space and incurs periodically the same inventory holding costs C_h. The inventory level is required to remain nonnegative and is not allowed to exceed the capacity limit $\bar{I}$ throughout the planning

Figure 1. (Color online) Comparison of the Decision Rule Functions from QDR, BGGN, PDR3, and GWK



time horizon. The inventory control problem can be stated as the MSRO problem

$$\begin{aligned}
\max \min_{\boldsymbol{\xi}\in\Xi} & \sum_{t=1}^{T}\sum_{p=1}^{P}[R_{t,p}(\boldsymbol{\xi}_t)s_{t,p}(\boldsymbol{\xi}^t) - C_b b_{t,p}(\boldsymbol{\xi}^t) - C_h I_{t,p}(\boldsymbol{\xi}^t)] \\
\text{s.t.}\ & I_{1,p}(\boldsymbol{\xi}^1) = I_{0,p} - s_{1,p}(\boldsymbol{\xi}^1),\ b_{1,p}(\boldsymbol{\xi}^1) = D_{1,p}(\boldsymbol{\xi}_1) - s_{1,p}(\boldsymbol{\xi}^1) \\
& \qquad \forall \boldsymbol{\xi}\in\Xi,\ \forall p\in[P] \\
& I_{t,p}(\boldsymbol{\xi}^t) = I_{t-1,p}(\boldsymbol{\xi}^{t-1}) + o_{t,p}(\boldsymbol{\xi}^{t-1}) - s_{t,p}(\boldsymbol{\xi}^t) \\
& \qquad \forall \boldsymbol{\xi}\in\Xi,\ \forall p\in[P],\ \forall t\in[T]\backslash\{1\} \\
& b_{t,p}(\boldsymbol{\xi}^t) = b_{t-1,p}(\boldsymbol{\xi}^{t-1}) + D_{t,p}(\boldsymbol{\xi}_t) - s_{t,p}(\boldsymbol{\xi}^t) \\
& \qquad \forall \boldsymbol{\xi}\in\Xi,\ \forall p\in[P],\ \forall t\in[T]\backslash\{1\} \\
& o_{t,p}(\boldsymbol{\xi}^t), s_{t,p}(\boldsymbol{\xi}^t), b_{t,p}(\boldsymbol{\xi}^t), I_{t,p}(\boldsymbol{\xi}^t) \ge 0, I_{t,p}(\boldsymbol{\xi}^t) \le \bar{I} \\
& \qquad \forall \boldsymbol{\xi}\in\Xi,\ \forall p\in[P],\ \forall t\in[T],
\end{aligned} \tag{28}$$

where $I_{0,p}$ are fixed to prespecified quantities for all $p\in[P]$. The product prices are defined as

$$R_{t,p}(\boldsymbol{\xi}_t) = 4 + \alpha_{1,p}\xi_{t,1} + \alpha_{2,p}\xi_{t,2} + \alpha_{3,p}\xi_{t,3} + \alpha_{4,p}\xi_{t,4}$$

with factor loadings $\alpha_{1,p}, \alpha_{2,p}, \alpha_{3,p}, \alpha_{4,p} \in [-1,1]$. Similarly, we set the demands to

$$\begin{aligned}
D_{t,p}(\boldsymbol{\xi}_t) = 2 + \sin\left(\frac{2\pi(t-1)}{12}\right) + \frac{1}{2}[\beta_{1,p}\xi_{t,1} + \beta_{2,p}\xi_{t,2} \\
+ \beta_{3,p}\xi_{t,3} + \beta_{4,p}\xi_{t,4}]
\end{aligned}$$

for $p = 1,\ldots,P/2$ and

$$\begin{aligned}
D_{t,p}(\boldsymbol{\xi}_t) = 2 + \cos\left(\frac{2\pi(t-1)}{12}\right) + \frac{1}{2}[\beta_{1,p}\xi_{t,1} + \beta_{2,p}\xi_{t,2} \\
+ \beta_{3,p}\xi_{t,3} + \beta_{4,p}\xi_{t,4}]
\end{aligned}$$

for $p = 1/P+1,\ldots,P$ with factor loadings $\beta_{1,p}, \beta_{2,p}, \beta_{3,p}, \beta_{4,p} \in [-1,1]$. The sine (cosine) terms in the expression reflect our expectation that the demands of the first (last) $P/2$ products are high in spring (winter) and low in fall (summer). We assume that the vectors of risk factors $\boldsymbol{\xi}_t \in \mathbb{R}^4$ for all $t = 1,\ldots,T$ are serially independent and uniformly distributed on $[-1,1]^4$. Formally, the uncertainty set is defined as

$$\Xi := \{(\boldsymbol{\xi} := \boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_t) : \|\boldsymbol{\xi}_t\|_\infty \le 1\ \ \forall t\in[T]\}.$$

The associated cone $\mathcal{K}$ is written as follows:

$$\mathcal{K} := \{(\boldsymbol{\xi},\tau) \in \mathbb{R}^{4T}\times\mathbb{R}_+ : \|\boldsymbol{\xi}_t\|_\infty \le \tau\ \ \forall t\in[T]\}.$$

In all numerical experiments, we generate 25 random instances of the inventory control problem with $p = 4$ products. We utilize the mechanism in Georghiou et al. (2015) to set up the parameters and to generate the random instances. We set backlogging and inventory holding costs identically to $C_b = C_h = 0.2$. We further set the initial inventory level to $I_{0,p} = 0$ and the inventory capacity to $\bar{I} = 24$. We sample the factor loadings $\alpha_{1,p}$, $\alpha_{2,p}, \alpha_{3,p}, \alpha_{4,p}$ and $\beta_{1,p}, \beta_{2,p}, \beta_{3,p}, \beta_{4,p}$ uniformly from the

Table 4. Relative Gaps (in Percentages) Between the Alternative Approximation Schemes and PLDR

	Number of time stages								
Method and statistic	1	3	6	9	12	15	18	21	24
BGGN									
10th percentile	3.5	9.8	1.7	18.8	8.5	4.6	24.8	23.5	6.6
Mean	17.3	21.0	20.8	42.7	47.9	43.7	99.2	129.3	191.2
90th percentile	39.2	38.3	36.8	70.6	100.5	94.6	154.9	225.4	762.7
PDR3									
10th percentile	0	0	—	—	—	—	—	—	—
Mean	0	−0.1	—	—	—	—	—	—	—
90th percentile	0	−0.2	—	—	—	—	—	—	—

interval $[-1, 1]$. As Problem (28) has nonfixed recourse, we employ linear decision rules and further enhance them by applying the piecewise scheme discussed in Section 2.4, where the folding directions are described by the standard basis vectors $\mathbf{e}_\ell$, $\ell \in [4]$. This gives rise to a semidefinite approximation, which results from replacing the copositive cone $\mathcal{COP}(\mathcal{K})$ in the equivalent copositive program with the inner approximation $\mathcal{IA}(\mathcal{K})$ defined in (16). We compare our piecewise linear decision rule (PLDR) scheme with the one proposed by Ben-Tal et al. (2009) (BGGN), where we replace the cone $\mathcal{COP}(\mathcal{K})$ with the inner approximation $\mathcal{AS}(\mathcal{K})$ defined in (18), and with PDR3.

We test the different schemes on problem instances with planning horizons $T = 1$, 3, 6, 9, 12, 15, 18, 21, and 24. Table 4 reports the relative gaps between the optimal values of PLDR and those of the other two schemes, whereas Table 5 shows the average computation times for the three approximation schemes. Note that PDR3 can only solve instances up to $T = 3$ before it starts experiencing numerical issues. As illustrated in Table 4, the relative gap between PLDR and BGGN increases dramatically with the planning horizon, where the largest average improvement of 191.2% is observed for $T = 24$. Meanwhile, PLDR can generate the same results as PDR3 in the case of $T = 1$ and remain very close to PDR3 for $T = 3$. As illustrated in these tables, our proposed copositive scheme can return solutions that are of very high quality without sacrificing much computational effort.

We also assess the quality of the decision rules obtained from the different approximation methods by evaluating their actual worst-case profits. Because the inventory control problem has nonfixed recourse, the worst-case profit of a fixed decision does not necessarily correspond to an extreme point scenario of the uncertainty set Ξ. Hence, we design a simulation procedure to estimate the worst-case profit as follows. We randomly generate 10,000 samples from the uncertainty set. Each sample point corresponds to a trajectory realization of the demands and prices over the T periods. The approximate worst-case profit is then given by the sample point that generates the smallest profit. Table 6 reports the statistics of relative gaps between the worst-case scenario profit of our method and those of BGGN and PDR3. We find that the proposed PLDR scheme still provides substantial average improvements over the BGGN method.

5.3. Index Tracking

For the last example, we study a dynamic index tracking problem, which aims at matching the performance of a stock index as closely as possible with a portfolio of other financial instruments over a finite discrete planning horizon T. A stochastic programming version of the problem is described in Hanasusanto and Kuhn (2013). To this end, we consider five stock indices where the first four constitute the tracking instruments, whereas the last one corresponds to the target index. Let $\boldsymbol{\xi} \in \mathbb{R}^5_+$ be the vector of total returns (price relatives) of these indices from time stage $t-1$ to time stage t. Here, $\xi_{t,1}$, $\xi_{t,2}$, $\xi_{t,3}$, and $\xi_{t,4}$ are returns of the four tracking instruments, whereas $\xi_{t,5}$ is the return of the target index at time stage t. The robust dynamic index

Table 5. The Average Computation Times (in Seconds) of the Different Approximation Schemes

	Number of time stages								
Method	1	3	6	9	12	15	18	21	24
PLDR	0.02	0.29	2.31	9.66	34.60	99.29	248.40	541.75	1,050.91
BGGN	0.01	0.04	0.33	1.23	4.76	14.48	36.85	94.49	191.30
PDR3	0.13	28.17	—	—	—	—	—	—	—

Table 6. Relative Gaps (in Percentages) of the Simulated Worst-Case Scenario Profits Between the Alternative Approximation Schemes and PLDR

	Number of time stages								
Method and statistic	1	3	6	9	12	15	18	21	24
BGGN									
10th percentile	1.7	6.9	1.5	12.8	6.9	3.7	22.3	19.4	6.1
Mean	15.3	19.6	17.9	35.9	42.8	41.6	79.2	119.9	181.7
90th percentile	34.7	36.1	34.5	61.7	89.1	90.3	132.8	187.8	692.4
PDR3									
10th percentile	0	0	—	—	—	—	—	—	—
Mean	0	−0.1	—	—	—	—	—	—	—
90th percentile	0	−0.1	—	—	—	—	—	—	—

tracking problem is stated as follows:

$$\begin{aligned}\min\ &\max_{\boldsymbol{\xi}\in\Xi}\sum_{t=1}^{T}|\xi_{t,5}-s_t(\boldsymbol{\xi}^t)|\\ \text{s.t. }\ &\boldsymbol{x}_0\ge 0,\ \boldsymbol{e}^\top\boldsymbol{x}_0\le 1,\ s_1(\boldsymbol{\xi}^1)=\boldsymbol{\xi}_1^\top\boldsymbol{x}_0\\ &s_t(\boldsymbol{\xi}^t)=\boldsymbol{\xi}_t^\top\boldsymbol{x}_{t-1}(\boldsymbol{\xi}^{t-1})\quad\forall t\in[T]\backslash\{1\}\\ &\boldsymbol{e}^\top\boldsymbol{x}_t(\boldsymbol{\xi}^t)\le s_t(\boldsymbol{\xi}^t),\ \boldsymbol{x}_t(\boldsymbol{\xi}^t)\ge 0\quad\forall t\in[T].\end{aligned}\tag{29}$$

The decision variable $s_t(\boldsymbol{\xi}^t)\in\mathbb{R}_+$ determines the value of the tracking portfolio at time stage t. Here, we aim to rebalance the portfolio allocation vector $\boldsymbol{x}(\boldsymbol{\xi}^t)\in\mathbb{R}^4$ of the four tracking instruments such that $s_t(\boldsymbol{\xi}^t)$ is as close to $\xi_{t,5}$ as possible throughout the planning time horizon. The uncertainty set Ξ in (29) is specified through a factor model as follows:

$$\Xi:=\left\{\boldsymbol{\xi}:=(\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_T):\begin{array}{ll}\boldsymbol{\xi}_t=\boldsymbol{f}+\boldsymbol{F}\boldsymbol{\zeta}_t,\ \boldsymbol{\zeta}_t\in\mathbb{R}^3 & \forall t\in[T]\\ \|\boldsymbol{\zeta}_t\|_\infty\le 1,\ \|\boldsymbol{\zeta}_t\|_1\le\rho & \forall t\in[T]\end{array}\right\}.$$

The associated cone $\mathcal{K}$ is accordingly written as

$$\mathcal{K}:=\left\{\boldsymbol{\xi},\tau)\in\mathbb{R}^{4T}\times\mathbb{R}_+:\begin{array}{ll}\boldsymbol{\xi}_t=\boldsymbol{f}\tau+\boldsymbol{F}\boldsymbol{\zeta}_t,\ \boldsymbol{\zeta}_t\in\mathbb{R}^3 & \forall t\in[T]\\ \|\boldsymbol{\zeta}_t\|_\infty\le\tau,\ \|\boldsymbol{\zeta}_t\|_1\le\rho\tau & \forall t\in[T]\end{array}\right\}.$$

Because the objective function of (29) is not linear, we introduce auxiliary variables $w_t(\cdot)$ to linearize each absolute term. This yields the multistage robust linear optimization problem

$$\begin{aligned}\min\ &\max_{\boldsymbol{\xi}\in\Xi}\sum_{t=1}^{T}w_t(\boldsymbol{\xi}^t)\\ \text{s.t. }\ &\boldsymbol{x}_0\ge 0,\ \boldsymbol{e}^\top\boldsymbol{x}_0\le 1,\ s_1(\boldsymbol{\xi}^1)=\boldsymbol{\xi}_1^\top\boldsymbol{x}_0\\ &s_t(\boldsymbol{\xi}^t)=\boldsymbol{\xi}_t^\top\boldsymbol{x}_{t-1}(\boldsymbol{\xi}^{t-1})\quad\forall t\in[T]\backslash\{1\}\\ &\boldsymbol{e}^\top\boldsymbol{x}_t(\boldsymbol{\xi}^t)\le s_t(\boldsymbol{\xi}^t),\ \boldsymbol{x}_t(\boldsymbol{\xi}^t)\ge 0\quad\forall t\in[T]\\ &w_t(\boldsymbol{\xi}^t)\ge\xi_{t,5}-s_t(\boldsymbol{\xi}^t),\ w_t(\boldsymbol{\xi}^t)\ge s_t(\boldsymbol{\xi}^t)-\xi_{t,5}\quad\forall t\in[T].\end{aligned}\tag{30}$$

As Problem (30) has nonfixed recourse, we apply linear decision rules to the decision variables $\boldsymbol{x}_t(\cdot)$, $t\in[T]$, which are multiplied with some uncertain parameters. On the other hand, we may utilize quadratic decision rules on $s_t(\cdot)$ and $w_t(\cdot)$, $t\in[T]$ because they are not multiplied with any uncertain parameters. With minimum modification, the copositive approach introduced in Section 4 can be applied, and accordingly, we can solve the semidefinite approximation obtained from replacing the copositive cone $\mathcal{COP}(\mathcal{K})$ with the inner approximation $\mathcal{IA}(\mathcal{K})$ defined in (16). We denote our approach by LQDR. We compare LQDR with the scheme proposed by Ben-Tal et al. (2009) (BGGN) where we replace the cone $\mathcal{COP}(\mathcal{K})$ with the inner approximation $\mathcal{AS}(\mathcal{K})$ defined in (18) and with PDR3.

All experimental results are averaged over 25 randomly generated instances. We utilize the mechanism in Hanasusanto and Kuhn (2013) to set up the parameters and generate the random instances. For each instance, $\boldsymbol{f}$ is set to the vector of all ones, whereas each entry of $\boldsymbol{F}$ is sampled uniformly from the interval $[-1,1]$. We further normalize each row of $\boldsymbol{F}$ such that the sum of the absolute values in each row equals one. We test the different schemes on problem instances with planning horizons $T=1$, 3, 6, 9, 12, 15, and 18. Note that PDR3 can only solve instances up to $T=3$. Table 7 reports the statistics of relative gaps between the optimal values obtained from LQDR and those from the two alternative approximation schemes, whereas Table 8 shows the average computation times for all three approximation schemes. As indicated in Table 7, the relative gap between LQDR and BGGN increases with the planning horizon, where the largest average improvement of 18.2% is observed for $T=18$. On the other hand, LQDR generates similar performance to PLDR3 but with significantly less computational effort.

We also assess the quality of the decision rules obtained from the different approximation methods by evaluating their true worst-case risks. As in the

Table 7. Relative Gaps (in Percentages) Between the Alternative Approximation Schemes and LQDR

	Number of time stages						
Method and statistic	1	3	6	9	12	15	18
BGGN							
10th percentile	0.0	1.0	1.9	2.2	4.7	1.8	2.5
Mean	0.0	7.1	12.5	11.8	14.2	17.0	18.2
90th percentile	0.0	21.7	29.4	29.0	33.8	30.1	34.2
PDR3							
10th percentile	0.0	0.0	—	—	—	—	—
Mean	0.0	−0.1	—	—	—	—	—
90th percentile	0.0	−0.4	—	—	—	—	—

inventory control problem, the index tracking problem has a nonfixed recourse. We adopt the same simulation procedure by using 10,000 sample trajectories from the uncertainty set to estimate the worst-case risks. Table 9 reports the statistics of relative gaps between the worst-case scenario risk of our method and those of BGGN and PDR3. We find that the proposed PLDR scheme provides significant average improvements over the BGGN method.

6. Concluding Remarks

Generic MSRO problems (with nonfixed recourse) have so far resisted strong decision rule approximations. This paper leveraged modern conic programming techniques to derive an exact convex copositive program for the linear decision rule approximation of the generic problems. We further derived an equivalent copositive program for the more powerful quadratic decision rule approximation of instances with fixed recourse. These reformulations enabled us to obtain a new semidefinite approximation that is provably tighter than an existing scheme of similar complexity by Ben-Tal et al. (2009). The copositive approach further inspired us to develop a new piecewise decision rule scheme for the generic problems. For MSRO problems with nonfixed recourse, we proved that the resulting approximation is tighter than the state-of-the-art scheme by Georghiou et al. (2015). Extensive numerical results demonstrate that our scheme can substantially outperform existing schemes in terms of optimality while maintaining scalability when solving large problem instances. We conclude that our proposed copositive approach provides an excellent balance between optimality and scalability.

Table 8. The Average Computation Times (in Seconds) of the Different Approximation Schemes

	Number of time stages						
Method	1	3	6	9	12	15	18
LQDR	0.03	0.40	5.01	32.95	127.34	601.57	1,703.32
BGGN	0.02	0.08	0.69	5.47	24.70	75.48	226.18
PDR3	0.09	8.50	—	—	—	—	—

Table 9. Relative Gaps (in Percentages) of the Simulated Worst-Case Scenario Risks Between the Alternative Approximation Schemes and LQDR

	Number of time stages						
Method and statistic	1	3	6	9	12	15	18
BGGN							
10th percentile	0.0	0.1	1.2	1.6	4.2	1.4	1.9
Mean	0.0	5.2	10.4	9.7	12.1	13.7	14.1
90th percentile	0.0	18.5	24.2	25.2	31.0	27.4	29.2
PDR3							
10th percentile	0.0	0.0	—	—	—	—	—
Mean	0.0	−0.0	—	—	—	—	—
90th percentile	0.0	−0.3	—	—	—	—	—

We mention two promising directions for further research. First, it would be interesting to derive a copositive programming reformulation for the piecewise decision rule scheme that can simultaneously optimize the best folding directions and break points. Second, it is imperative to design a global solution approach for MSRO problems with nonfixed recourse that leverages the proposed decision rule schemes.

Acknowledgments

The authors acknowledge the two anonymous reviewers whose comments led to substantial improvements of the paper.

References

Alizadeh F, Goldfarb D (2003) Second-order cone programming. *Math. Programming* 95(1):3–51.

Anderson BD, Moore JB (2007) *Optimal Control: Linear Quadratic Methods* (Courier Corporation, North Chelmsford, MA).

ApS (2016) The MOSEK optimization toolbox for MATLAB, version 8.0.0.94. Accessed October 31, 2017, http://docs.mosek.com/8.0/toolbox/index.html.

Ardestani-Jaafari A, Delage E (2021) Linearized robust counterparts of two-stage robust optimization problems with applications in operations management. *INFORMS J. Comput.* 33(3):1138–1161.

Atamtürk A, Zhang M (2007) Two-stage robust network flow and design under demand uncertainty. *Oper. Res.* 55(4):662–673.

Bampou D, Kuhn D (2011) Scenario-free stochastic programming with polynomial decision rules. *Proc. 50th IEEE Conf. Decision Control Eur. Control Conf. (CDC-ECC)* (IEEE, Piscataway, NJ), 7806–7812.

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

Ben-Tal A, Nemirovski A (1999) Robust solutions of uncertain linear programs. *Oper. Res. Lett.* 25(1):1–13.

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

Ben-Tal A, El Ghaoui L, Nemirovski A (2009) *Robust Optimization* (Princeton University Press, Princeton, NJ).

Ben-Tal A, El Housni O, Goyal V (2020) A tractable approach for designing piecewise affine policies in two-stage adjustable robust optimization. *Math. Programming* 182(1):57–102.

Ben-Tal A, Margalit T, Nemirovski A (2000) Robust modeling of multi-stage portfolio problems. Frenk H, Roos K, Terlaky T, Zhang S, eds. *High Performance Optimization*, Applied Optimization, vol. 33 (Springer, Boston), 303–328.

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

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

Bertsimas D, de Ruiter FJ (2016) Duality in two-stage adaptive linear optimization: Faster computation and stronger bounds. *INFORMS J. Comput.* 28(3):500–511.

Bertsimas D, Dunning I (2016) Multistage robust mixed-integer optimization with adaptive partitions. *Oper. Res.* 64(4):980–998.

Bertsimas D, Georghiou A (2015) Design of near optimal decision rules in multistage adaptive mixed-integer optimization. *Oper. Res.* 63(3):610–627.

Bertsimas D, Georghiou A (2018) Binary decision rules for multistage adaptive mixed-integer optimization. *Math. Programming* 167(2):395–433.

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

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

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

Bertsimas D, Iancu DA, Parrilo PA (2010) Optimality of affine policies in multistage robust optimization. *Math. Oper. Res.* 35(2): 363–394.

Bertsimas D, Iancu DA, Parrilo PA (2011b) A hierarchy of near-optimal policies for multistage adaptive optimization. *IEEE Trans. Automatic Control* 56(12):2809–2824.

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

Bomze IM (2012) Copositive optimization—Recent developments and applications. *Eur. J. Oper. Res.* 216(3):509–520.

Bomze IM, de Klerk E (2002) Solving standard quadratic optimization problems via linear, semidefinite and copositive programming. *J. Global Optim.* 24(2):163–185.

Bomze IM, Dür M, De Klerk E, Roos C, Quist AJ, Terlaky T (2000) On copositive programming and standard quadratic optimization problems. *J. Global Optim.* 18(4):301–320.

Burer S (2009) On the copositive representation of binary and continuous nonconvex quadratic programs. *Math. Programming* 120(2):479–495.

Burer S (2012) Copositive programming. Anjos MF, Lasserre JB, eds. *Handbook on Semidefinite, Conic and Polynomial Optimization* (Springer, New York), 201–218.

Burer S (2015) A gentle, geometric introduction to copositive optimization. *Math. Programming* 151(1):89–116.

Burer S, Dong H (2012) Representing quadratically constrained quadratic programs as generalized copositive programs. *Oper. Res. Lett.* 40(3):203–206.

Calafiore GC (2008) Multi-period portfolio optimization with linear control policies. *Automatica J. IFAC* 44(10):2463–2473.

Chen J, Burer S (2012) Globally solving nonconvex quadratic programming problems via completely positive programming. *Math. Programming Comput.* 4(1):33–52.

Chen X, Zhang Y (2009) Uncertain linear programs: Extended affinely adjustable robust counterparts. *Oper. Res.* 57(6):1469–1482.

Chen X, Sim M, Sun P (2007) A robust optimization perspective on stochastic programming. *Oper. Res.* 55(6):1058–1071.

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

Dantzig GB, Infanger G (1993) Multi-stage stochastic linear programs for portfolio optimization. *Ann. Oper. Res.* 45(1):59–76.

de Klerk E, Pasechnik DV (2002) Approximation of the stability number of a graph via copositive programming. *SIAM J. Optim.* 12(4):875–892.

Dür M (2010) Copositive programming—A survey. Diehl M, Glineur F, Jarlebring E, Michiels W, eds. *Recent Advances in Optimization and Its Applications in Engineering* (Springer, Berlin), 3–20.

Eichfelder G, Jahn J (2008) Set-semidefinite optimization. *J. Convex Anal.* 15:767–801.

Garey MR, Johnson DS (1979) *Computers and Intractability*, vol. 29 (W. H. Freeman and Company, San Francisco).

Georghiou A, Tsoukalas A, Wiesemann W (2019) Robust dual dynamic programming. *Oper. Res.* 67(3):813–830.

Georghiou A, Tsoukalas A, Wiesemann W (2020) A primal–dual lifting scheme for two-stage robust optimization. *Oper. Res.* 68(2):572–590.

Georghiou A, Wiesemann W, Kuhn D (2015) Generalized decision rule approximations for stochastic programming via liftings. *Math. Programming* 152(1–2):301–338.

Goh J, Sim M (2010) Distributionally robust optimization and its tractable approximations. *Oper. Res.* 58(4 part 1):902–917.

Gounaris CE, Wiesemann W, Floudas CA (2013) The robust capacitated vehicle routing problem under demand uncertainty. *Oper. Res.* 61(3):677–693.

Guslitser E (2002) Uncertainty-immunized solutions in linear programming. PhD thesis, Technion–Israel Institute of Technology, Haifa.

Hadjiyiannis MJ, Goulart PJ, Kuhn D (2011) A scenario approach for estimating the suboptimality of linear decision rules in two-stage robust optimization. *Proc. 50th IEEE Conf. Decision Control Eur. Control Conf. (CDC-ECC)* (IEEE, Piscataway, NJ).

Hanasusanto GA, Kuhn D (2013) Robust data-driven dynamic programming. *Adv. Neural Inform. Processing Systems*, 827–835.

Hanasusanto GA, Kuhn D (2018) Conic programming reformulations of two-stage distributionally robust linear programs over Wasserstein balls. *Oper. Res.* 66(3):849–869.

Hanasusanto GA, Kuhn D, Wallace SW, Zymler S (2015) Distributionally robust multi-item newsvendor problems with multimodal demand distributions. *Math. Programming* 152(1–2):1–32.

Hashemi Doulabi H, Jaillet P, Pesant G, Rousseau LM (2021) Exploiting the structure of two-stage robust optimization models with exponential scenarios. *INFORMS J. Comput.* 33(1):143–162.

Iancu DA, Sharma M, Sviridenko M (2013) Supermodularity and affine policies in dynamic robust optimization. *Oper. Res.* 61(4):941–956.

Kong Q, Lee CY, Teo CP, Zheng Z (2013) Scheduling arrivals to a stochastic service delivery system using copositive cones. *Oper. Res.* 61(3):711–726.

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

Lasserre JB (2009) Convexity in semialgebraic geometry and polynomial optimization. *SIAM J. Optim.* 19(4):1995–2014.

Lofberg J (2004) YALMIP: A toolbox for modeling and optimization in MATLAB. *2004 IEEE Internat. Sympos. Comput. Aided Control Systems Design* (IEEE, Piscataway, NJ), 284–289.

Martins da Silva Rocha PC (2013) Medium-term planning in deregulated energy markets with decision rules. PhD thesis, Imperial College London, London.

Mittal A, Gokalp C, Hanasusanto GA (2020) Robust quadratic programming with mixed-integer uncertainty. *INFORMS J. Comput.* 32(2):201–218.

Natarajan K, Teo CP (2017) On reduced semidefinite programs for second order moment bounds with applications. *Math. Programming* 161(1–2):487–518.

Natarajan K, Teo CP, Zheng Z (2011) Mixed 0-1 linear programs under objective uncertainty: A completely positive representation. *Oper. Res.* 59(3):713–728.

Parrilo PA (2000) Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. PhD thesis, California Institute of Technology, Pasadena, CA.

Postek K, den Hertog D (2016) Multistage adjustable robust mixed-integer optimization via iterative splitting of the uncertainty set. *INFORMS J. Comput.* 28(3):553–574.

Rocha P, Kuhn D (2012) Multistage stochastic portfolio optimisation in deregulated electricity markets using linear decision rules. *Eur. J. Oper. Res.* 216(2):397–408.

See CT, Sim M (2010) Robust approximation to multiperiod inventory management. *Oper. Res.* 58(3):583–594.

Xu G, Burer S (2018) A copositive approach for two-stage adjustable robust optimization with uncertain right-hand sides. *Comput. Optim. Appl.* 70(1):33–59.

Zeng B, Zhao L (2013) Solving two-stage robust optimization problems using a column-and-constraint generation method. *Oper. Res. Lett.* 41(5):457–461.

Zhen J, den Hertog D, Sim M (2018) Adjustable robust optimization via Fourier–Motzkin elimination. *Oper. Res.* 66(4):1086–1100.

Guanglin Xu is an assistant professor of system engineering and engineering management at the University of North Carolina at Charlotte. His research interests are at the intersection of operations research, machine learning, and data science, with their applications in healthcare, energy systems, and supply chain management.

Grani A. Hanasusanto is an associate professor of industrial and enterprise systems engineering at the University of Illinois Urbana–Champaign. His research focuses on the design and analysis of data-driven solution schemes for decision-making problems under uncertainty, with applications in operations management, energy systems, finance, and machine learning.