Optimization Online

A New Dual-Based Cutting Plane Algorithm for Nonlinear Adjustable Robust Optimization

Abbas Khademi · Ahmadreza Marandi · Majid Soleimani-damaneh

Abstract This paper explores a class of nonlinear Adjustable Robust Optimization (ARO) problems, containing here-and-now and wait-and-see variables, with uncertainty in the objective function and constraints. By applying Fenchel's duality on the wait-and-see variables, we obtain an equivalent dual reformulation, which is a nonlinear static robust optimization problem. Using the dual formulation, we provide conditions under which the ARO problem is convex on the here-and-now decision. Furthermore, since the dual formulation contains a non-concave maximization on the uncertain parameter, we use perspective relaxation and an alternating method to handle the non-concavity. By employing the perspective relaxation, we obtain an upper bound, which we show is the same as the static relaxation of the considered problem. Moreover, invoking the alternating method, we design a new dual-based cutting plane algorithm that is able to find a reasonable lower bound for the optimal objective value of the considered nonlinear ARO model. In addition to sketching and establishing the theoretical features of the algorithms, including convergence analysis, by numerical experiments we reveal the abilities of our cutting plane algorithm in producing locally robust solutions with an acceptable optimality gap.

Keywords Adjustable Robust Optimization · Fenchel Duality · Biconvex Programming · Perspective Function · Alternating Method · Cutting Plane Methods

Mathematics Subject Classification (2000) 90C17 · 90C26 · 90C30 · 90C46

1 Introduction

Mathematical optimization has become an important part of many decision-making problems in management, economics, medicine, engineering, etc. In classical optimization models, all parameters are considered to be exactly known, resulting in deterministic problems. In real-life, however, many of the parameters are not known at the moment of decision-making and have uncertainty in their essence. There are various approaches for dealing with uncertainty in the optimization and mathematical modeling literature. The most commonly used ones are stochastic optimization and robust optimization.

In stochastic optimization, probabilistic information (distribution) on the uncertain parameters is required and the decision-maker aims to optimize expected objective values; for a more detailed description

A. Khademi
School of Mathematics, Statistics and Computer Science, College of Science,
University of Tehran,
Tehran, Iran.
E-mail: abbaskhademi@ut.ac.ir

A. Marandi
Department of Industrial Engineering and Innovation Sciences,
Eindhoven University of Technology,
Eindhoven, The Netherlands.
E-mail: a.marandi@tue.nl

M. Soleimani-damaneh
School of Mathematics, Statistics and Computer Science, College of Science,
University of Tehran,
Tehran, Iran.
E-mail: soleimani@khayam.ut.ac.ir

of stochastic optimization, we refer to the textbook [40] and the references therein. Obtaining the precise distribution may be challenging [12]. Furthermore, applying this approach may be computationally difficult in some cases [41]. In contrast, Robust Optimization (RO) does not require any probabilistic information. In RO, the best solution is chosen among those that are safe-guarded against all scenarios in a pre-specified set, called uncertainty set.

The concept of static RO was first proposed by Soyster in the 1970s, who studied a linear optimization problem with a box uncertainty set [42]. Later in the 1990s, static RO was formally introduced [7,8,18] and its computational advantage has resulted in its wide usage in applications, including in portfolio selection [26,47], scheduling [15], operations management [30], etc.

In static RO, all decision variables represent here-and-now decisions, meaning the decisions are made before realization of the uncertain parameters [5]. However, in many practical applications, the value of some decisions can be adjusted after realization of (part of) uncertain parameters. These kinds of decisions are called wait-and-see decisions. Adjustable Robust Optimization (ARO) is an extension of the static RO wherein decision variables are divided into two types: here-and-now and wait-and-see [6]. In recent years, the application of ARO has been widespread in many areas such as network design [53], location-transportation problems [32], facility location problems [17], logistics [25], chemical engineering [24,28], radiotherapy [38], to name a few.

Although ARO improves solution quality (in the sense of being less conservative), its computational complexity is higher than static RO and is computationally more demanding [49]. A way to approximate an ARO problem is by restricting wait-and-see variables to have special form in the uncertain parameter, called *decision rules*. In the literature many decision rules are introduced including constant [31], piecewise constant [36], affine [6,10], quadratic [46], and polynomial [13]. In addition to decision rules, there are several other approximation techniques for solving linear ARO problems in the literature, including Benders decomposition [51], finite adaptability [21], partitioning the uncertainty set [9,34], copositive approach [48], saddle-point approximation approach [52], etc.

Most studies in ARO are focusing on linear and integer-linear [1,11,22]; for more additional details, see the survey paper [49]. There are only a few papers devoted to the nonlinear case due to its theoretical and computational challenges. In [44], the authors considered a nonlinear ARO problem with a polytope uncertainty set and proposed a method to solve such problems under some quasi-convexity conditions. ARO models with second-order cone constraints and ellipsoidal uncertainty sets are considered in [14], where the authors show that applying affine decision rules would result in a semi-definite optimization problem. In [39], the authors considered a nonlinear ARO model with linear uncertainty (the functions are linear in the uncertain parameters and the uncertainty set is a polyhedron), and derived an equivalent ARO problem, which is linear in the wait-and-see decisions.

In this paper, we show how to use duality to reformulate a general nonlinear ARO problem and solve it. More specifically, the main contribution of our work can be summarized as follows:

- First, we consider a general nonlinear adjustable robust optimization problem. Applying Fenchel's duality and dualizing over the wait-and-see decisions, we obtain an equivalent static robust optimization reformulation (dual reformulation). Then, we provide conditions under which the dual reformulation is convex on decision variables.
- Second, we show under some conditions that a convex relaxation of the dual reformulation is equivalent to approximating the ARO problem using constant decision rule.
- Finally, we design an algorithm based on the dual reformulation. The algorithm consists of two main phases: In the first phase, we use an alternating method exploiting the structure of the dual reformulation. We show under which conditions, the alternating method converges to a local worst-case scenario within the uncertainty set. In the second phase, we use finite-scenario approach, given the obtained scenarios in the first phase, to find a solution. Given this solution, we find new local worst-case scenarios and repeat this two-phase procedure until satisfying a stopping criterion. Using this algorithm, we have a lower bound on the original problem and obtain a locally robust solution. We further improve the lower bound by introducing new cuts. Our computational results show that our algorithm can provide a locally robust solution with an acceptable optimality gap.

The rest of the paper is organized as follows. In Section 2, some preliminaries and definitions from the convex analysis and robust optimization are given. We, then, reformulate a general nonlinear adjustable robust problem as a nonlinear static robust counterpart using Fenchel's duality in Section 3. In Section 4, we apply a convex relaxation technique on the dual reformulation to obtain an upper bound and show the relationship between the corresponding static robust counterpart of the ARO problem and this relaxation. Finally, we propose a new algorithm in Section 5 to construct a lower bound and obtain a locally robust solution. Our numerical results are presented in Section 6.

2 Preliminaries

In this work, we use the following definitions and notations. We first recall some standard terminology from convex analysis. The *domain* of a convex function $g : \mathbb{R}^{n_x} \to [-\infty, \infty]$ is defined as $dom(g) = \{x \in \mathbb{R}^{n_x} \mid g(x) < +\infty\}$. The function g is *proper* if $g(x) > -\infty$ for all $x \in \mathbb{R}^{n_x}$ and $g(x) < +\infty$ for at least one $x \in \mathbb{R}^{n_x}$. This function is said to be *closed* if for each $\ell \in \mathbb{R}$, the sub-level set $\{x \in \mathbb{R}^{n_x} \mid g(x) \leq \ell\}$ is closed.

We use $[m]$ to denote the set $\{1, 2, \ldots, m\}$, and $[m_0]$ to denote the set $\{0, 1, \ldots, m\}$. The column vector of all zeros will be denoted by $\mathbf{0}$. The sets of all non-negative and all extended-real numbers are denoted by $\mathbb{R}_+ := [0, \infty)$ and $\bar{\mathbb{R}} := [-\infty, \infty]$, respectively. For a given vector $x \in \mathbb{R}^{n_x}$, its transpose is denoted by $x^\top$.

Definition 1 (Conjugate Function, [35]) The convex conjugate of a function $g : \mathbb{R}^{n_x} \to \bar{\mathbb{R}}$ is the function $g^* : \mathbb{R}^{n_x} \to \bar{\mathbb{R}}$ defined as $g^*(y) := \sup_{x \in \mathbb{R}^{n_x}} \left\{ y^\top x - g(x) \right\}$, where $y \in \mathbb{R}^{n_x}$.

The *indicator function* of a set $S \subseteq \mathbb{R}^{n_x}$, denoted by δ_S, is defined as

$$\delta_S(x) = \begin{cases} 0, & x \in S, \\ \infty, & x \notin S. \end{cases}$$

The *support function* $\delta_S^* : \mathbb{R}^{n_x} \to \bar{\mathbb{R}}$ of a set $S \subseteq \mathbb{R}^{n_x}$ is $\delta_S^*(y) := \sup_{x \in S} \left\{ y^\top x \right\}$, where $y \in \mathbb{R}^{n_x}$. It is worth mentioning that the support function corresponding to S is the conjugate of δ_S.

Definition 2 (Perspective Function, [35, 54]) The convex perspective of a proper, closed, and convex function $g : \mathbb{R}^{n_x} \to \bar{\mathbb{R}}$ is the function $g^{per} : \mathbb{R}^{n_x} \times \mathbb{R}_+ \to \bar{\mathbb{R}}$ defined by

$$g^{per}(x, t) = \begin{cases} tg\left(\frac{x}{t}\right), & t > 0, \\ \delta^*_{dom(g^*)}(x), & t = 0. \end{cases}$$

The convex perspective of a proper, closed, and convex function is also proper, closed, and convex (more precisely, jointly in (x, t)); see [35, page 35] for convexity and properness, and [35, page 67 and Theorem 13.3] for closedness.

Remark 1 A proper, closed, and convex function g conforms to the following relation with its convex conjugate and perspective functions

$$g^{per}(x, t) = \sup_y \left\{ y^\top x - tg^*(y) \mid y \in dom(g^*) \right\}.$$

In the literature of convex analysis, $g^{per}(x, 0)$ is called the *asymptotic function* or *recession function* of g. Moreover, $g^{per}(x, 0) = \liminf_{\substack{x' \to x \\ t' \downarrow 0}} t' g\left(\frac{x'}{t'}\right)$ [2, 23]. So, we have

$$\sup_{t > 0, x \in \mathbb{R}^{n_x}} g^{per}(x, t) = \sup_{t \geq 0, x \in \mathbb{R}^{n_x}} g^{per}(x, t),$$

and also

$$\inf_{t > 0, x \in \mathbb{R}^{n_x}} g^{per}(x, t) = \inf_{t \geq 0, x \in \mathbb{R}^{n_x}} g^{per}(x, t).$$

Details can be found in Appendix 1. □

The definitions are extended to partial conjugate and perspective. The partial conjugate of a function $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \bar{\mathbb{R}}$ with respect to its second argument (likewise for first argument) is the function $g^{*2} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \bar{\mathbb{R}}$ defined as $g^{*2}(x, w) = sup_{y \in \mathbb{R}^{n_y}} \{w^\top y - g(x, y)\}$, and its domain is denoted by $dom(g^{2*})(x, \cdot)$. If $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \bar{\mathbb{R}}$ is a proper, closed, and concave function in its second argument, then its concave partial perspective $h^{per} : \mathbb{R}^{n_x} \times \mathbb{R}_+ \times \mathbb{R}^{n_u} \to \bar{\mathbb{R}}$ is defined as

$$h^{per}(x, t, u) = \begin{cases} th\left(x, \frac{u}{t}\right), & t > 0, \\ -\delta^*_{dom((-h)^{2*}(x, .))}(u), & t = 0. \end{cases}$$

Henceforth, for the ease of notation, we use $0h\left(x, u/0\right)$ instead of $h^{per}(x, 0, u)$.

Definition 3 (Fenchel's Dual Problem of a Convex Programming, [37, 54]) Consider the following primal convex optimization problem:

$$\begin{aligned} \inf_{y} \quad & g_0(y) \\ \text{s.t.} \quad & g_i(y) \le 0, \; i \in [m], \end{aligned} \tag{P}$$

where the functions $g_i : \mathbb{R}^n \to \bar{\mathbb{R}}$, $i \in [m_0]$, are proper, closed, and convex. The Fenchel dual of (P) is defined as

$$\begin{aligned} \sup_{\lambda, \{w^i\}_{i=0}^m} \quad & -g_0^*(w^0) - \sum_{i=1}^m (g_i^*)^{per}(w^i, \lambda_i) \\ \text{s.t.} \quad & \lambda \ge \mathbf{0}, \; \sum_{i=0}^m w^i = \mathbf{0}, \; w^0 \in dom(g_0^*), \\ & (w^i, \lambda_i) \in dom\left((g_i^*)^{per}\right), \; i \in [m]. \end{aligned} \tag{D}$$

For the ease of notation in (D), we use $\lambda_i g_i^* \left(\frac{w^i}{\lambda_i}\right)$ to denote $(g_i^*)^{per}(w^i, \lambda_i)$, even for $\lambda_i = 0$. So, we may write (D) as follows:

$$\begin{aligned} \sup_{\lambda, \{w^i\}_{i=0}^m} \quad & -g_0^*(w^0) - \sum_{i=1}^m \lambda_i g_i^* \left(\frac{w^i}{\lambda_i}\right) \\ \text{s.t.} \quad & \lambda \ge \mathbf{0}, \; \sum_{i=0}^m w^i = \mathbf{0}, \; w^0 \in dom(g_0^*), \\ & \frac{w^i}{\lambda_i} \in dom(g_i^*), \; i \in [m], \end{aligned}$$

where $\frac{w^i}{\lambda_i} \in dom(g_i^*)$ for $\lambda_i = 0$ means $\delta^*_{dom(g_i)}(w^i) < \infty$.

Remark 2 In problem (D), constraints corresponding to the domain are essential and in many cases, they also lead to convex constraints for (D). Moreover, since (D) is a maximization problem, these constraints hold explicitly. These constraints, in many cases, enable us to eliminate the variables w^i. For brevity, in [37, 54] the dual problem has been written as follows:

$$\begin{aligned} \sup_{\lambda, \{w^i\}_{i=0}^m} \quad & -g_0^*(w^0) - \sum_{i=1}^m \lambda_i g_i^* \left(\frac{w^i}{\lambda_i}\right) \\ \text{s.t.} \quad & \lambda \ge \mathbf{0}, \; \sum_{i=0}^m w^i = \mathbf{0}. \end{aligned}$$

□

Definition 4 (Slater Regularity for Optimization Problem) Problem (P) is Slater regular when there exists some feasible solution $y^s \in \cap_{i \in [m_0]} ri\left(dom(g_i)\right)$, and $g_i(y^s) < 0$ for all $i \in [m]$.

It is important to note that if (P) is Slater regular, then the optimal values of (P) and (D) are equal [54].

In the rest of this section, we recall definitions for robust optimization problems. The general form of an *uncertain nonlinear optimization problem* is as follows

$$\inf_{x \in \mathcal{X}} \inf_{y \in \mathcal{Y}(x,u)} f_0\left(x, y, u\right), \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ is a vector containing non-adjustable (here-and-now) decisions, $y \in \mathcal{Y}(x,u) \subseteq \mathbb{R}^{n_y}$ is a vector containing adjustable (wait-and-see) decisions, $\mathcal{Y}(x,u) = \{y \in \mathbb{R}^{n_y} : f_j\left(x, y, u\right) \le 0, \; j \in [m]\}$, $u \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$ is an uncertain parameter, $\mathcal{U}$ is the uncertainty set, and $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ is a set with additional constraints on x.

Assumption 1 *In (1), we assume that $f_j : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R}^{n_u} \to \bar{\mathbb{R}}$, $j \in [m_0]$ are convex in x, proper, closed, and convex in y, and concave in u.*

The static and adjustable robust counterparts corresponding to uncertain problem (1) can be defined as follows.

Definition 5 (Static Robust Optimization, [31]) The static Robust Counterpart (RC) for uncertain problem (1) is defined by

$$\begin{aligned} \inf_{x \in \mathcal{X}, y \in \mathbb{R}^{n_y}} \quad & \sup_{u \in \mathcal{U}} f_0\left(x, y, u\right) \\ \text{s.t.} \quad & \sup_{u \in \mathcal{U}} f_j\left(x, y, u\right) \le 0, \quad j \in [m]. \end{aligned} \tag{RC}$$

Definition 6 (Adjustable Robust Optimization, [31,44]) The Adjustable Robust Counterpart (ARC) for uncertain problem (1) is defined by

$$\begin{aligned} \inf_{x\in\mathcal{X}} \ \sup_{u\in\mathcal{U}} \ \inf_{y\in\mathbb{R}^{n_y}} \ & f_0\left(x,y,u\right) \\ \text{s.t.} \ & f_j\left(x,y,u\right) \leq 0, \quad j\in[m]. \end{aligned} \tag{ARC}$$

Definition 7 (Fixed Recourse Problem, [31]) Problem (1) has fixed-recourse when

$$f_j(x,y,u) = \hat{f}_j(x,u) + \hat{g}_j(x,y), \quad j\in[m_0].$$

In this paper, we consider a special case of the above notion defined as follows.

Definition 8 (Separable Fixed Recourse Problem) We say (1) has separable fixed-recourse when

$$f_j(x,y,u) = \hat{f}_j(x,u) + g_j(y), \quad j\in[m_0].$$

3 Dual Reformulation

In this section, we derive the dual formulation of (ARC). In the next theorem, we show how Fenchel duality is used for this goal.

Theorem 1 *Let Assumption 1 hold. Also, in (ARC) let us assume that*

$$\forall (x\in\mathcal{X}, u\in\mathcal{U}), \exists y: f_j\left(x,y,u\right) < 0, \quad j\in[m]. \tag{2}$$

Then, (ARC) is equivalent to the nonlinear static robust counterpart

$$\begin{aligned} \inf_{x\in\mathcal{X},\tau} \ & \tau \\ \text{s.t.} \ & -f_0^{*2}\left(x,w^0,u\right) - \sum_{j=1}^{m}\lambda_j f_j^{*2}\left(x,\frac{w^j}{\lambda_j},u\right) \leq \tau, \quad \forall \begin{pmatrix}\lambda\\ \{w^j\}_{j=0}^m \\ u\end{pmatrix} \in \mathcal{Z}, \end{aligned} \tag{3}$$

where

$$\mathcal{Z} = \left\{ \begin{pmatrix}\lambda\\ \{w^j\}_{j=0}^m \\ u\end{pmatrix} \in \mathbb{R}^{m+n_y(m+1)+n_u} \;\middle|\; \begin{array}{l} \lambda\geq \mathbf{0}, \ u\in\mathcal{U}, \ \sum_{j=0}^{m} w^j = \mathbf{0}, \\ w^0 \in dom\left(f_0^{*2}(x,\cdot,u)\right), \\ \frac{w^j}{\lambda_j} \in dom\left(f_j^{*2}(x,\cdot,u)\right), \ j\in[m] \end{array} \right\}.$$

Proof In (ARC), we consider the inner minimization problem over y for a given $x\in\mathcal{X}$ and $u\in\mathcal{U}$. Because of (2) and Assumption 1, we know for the given $x\in\mathcal{X}$ and $u\in\mathcal{U}$, the inner minimization is Slater regular. Therefore, we can apply Fenchel's duality (Definition 3), and rewrite (ARC) as follows:

$$\begin{aligned} \inf_{x\in\mathcal{X}} \ \sup_{u\in\mathcal{U}} \ \sup_{\lambda,w^j} \ & -f_0^{*2}(x,w^0,u) - \sum_{j=1}^{m}\lambda_j f_j^{*2}\left(x,\frac{w^j}{\lambda_j},u\right) \\ \text{s.t.} \ & \sum_{j=0}^{m} w^j = \mathbf{0}, \ \lambda\geq\mathbf{0}, \\ & w^0\in dom\left(f_0^{*2}(x,\cdot,u)\right), \ \frac{w^j}{\lambda_j}\in dom\left(f_j^{*2}(x,\cdot,u)\right), \quad j\in[m]. \end{aligned}$$

Therefore, (ARC) can be reformulated as

$$\begin{aligned} \inf_{x\in\mathcal{X}} \ \sup_{u\in\mathcal{U},\lambda,w^j} \ & -f_0^{*2}(x,w^0,u) - \sum_{j=1}^{m}\lambda_j f_j^{*2}\left(x,\frac{w^j}{\lambda_j},u\right) \\ \text{s.t.} \ & \sum_{j=0}^{m} w^j = \mathbf{0}, \ \lambda\geq\mathbf{0}, \\ & w^0\in dom\left(f_0^{*2}(x,\cdot,u)\right), \ \frac{w^j}{\lambda_j}\in dom\left(f_j^{*2}(x,\cdot,u)\right), \quad j\in[m]. \end{aligned}$$

Using the definition of $\mathcal{Z}$ and epigraph reformulation, we may rewrite problem (ARC) as (3), which completes the proof. □

The above theorem shows that a nonlinear adjustable robust optimization can be reformulated as a nonlinear static robust optimization under a Slater condition. In the equivalent dual reformulation (3), the uncertain parameters include the dual multipliers (i.e., $\lambda, \{w^j\}_{j=0}^m$), in addition to the original uncertain parameter u.

The conjugate functions and their domains can be easily computed for a wide range of convex functions; see, e.g., [37, Table E.1].

In the proof of the above theorem, we did not use the convexity of f_j functions on x and their concavity on u. However, we usually take convex functions on decision variables and concave functions on uncertain parameters to get tractable models. The benefit of the dual reformulation obtained in Theorem 1 is that we can get upper and lower bounds for the optimal objective value of the original model (ARC). Later, in Sections 4 and 5, we explain how to achieve these goals.

In the following corollary, we derive the formulation of the dual problem for cases where $f_j(x, y, u)$, $j \in [m_0]$ are separable.

Corollary 1 *Consider the following ARC:*

$$\begin{aligned} \inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U}} \inf_{y} \; & \hat{f}_0(x) + h_0(u) + g_0(y) \\ \text{s.t. } & \hat{f}_j(x) + h_j(u) + g_j(y) \le 0, \quad j \in [m]. \end{aligned} \tag{4}$$

Let g_j be proper, convex, and closed in y. Assume that there exists some y such that

$$g_j(y) < -\sup_{x\in\mathcal{X}} \hat{f}_j(x) - \sup_{u\in\mathcal{U}} h_j(u), \quad j \in [m].$$

Then, the nonlinear ARC (4) is equivalent to the following static RO problem

$$\begin{aligned} \inf_{x\in\mathcal{X},\tau} \; & \tau \\ \text{s.t. } & \sum_{j=0}^m \lambda_j \hat{f}_j(x) + \sum_{j=0}^m \lambda_j h_j(u) - \sum_{j=0}^m \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}) \le \tau, \quad \begin{pmatrix} \lambda \\ \{w^j\}_{j=0}^m \\ u \end{pmatrix} \in \mathcal{P}, \end{aligned}$$

where

$$\mathcal{P} = \left\{ \begin{pmatrix} \lambda \\ \{w^j\}_{j=0}^m \\ u \end{pmatrix} \middle| \begin{array}{l} u \in \mathcal{U}, \; \sum_{j=0}^m w^j = \boldsymbol{0}, \\ \lambda_0 = 1, \; \lambda_j \ge 0, \\ \frac{w^j}{\lambda_j} \in dom(g_j^*), \; j \in [m_0] \end{array} \right\}.$$

Proof By setting $f_j(x, y, u) := \hat{f}_j(x) + h_j(u) + g_j(y)$, for $\lambda_j > 0$ we have

$$\begin{aligned} \lambda_j {f_j}^{*2}(x, \tfrac{w^j}{\lambda_j}, u) &= \lambda_j \sup_y \left\{ \left(\tfrac{w^j}{\lambda_j}\right)^\top y - f_j(x, y, u) \right\} \\ &= \sup_y \left\{ \left(w^j\right)^\top y - \lambda_j \left(\hat{f}_j(x) + h_j(u) + g_j(y)\right) \right\} \\ &= \sup_y \left\{ \left(w^j\right)^\top y - \lambda_j \hat{f}_j(x) - \lambda_j h_j(u) - \lambda_j g_j(y) \right\} \\ &= -\lambda_j \hat{f}_j(x) - \lambda_j h_j(u) + \sup_y \left\{ \left(w^j\right)^\top y - \lambda_j g_j(y) \right\} \\ &= -\lambda_j \hat{f}_j(x) - \lambda_j h_j(u) + \lambda_j \sup_y \left\{ \left(\tfrac{w^j}{\lambda_j}\right)^\top y - g_j(y) \right\} \\ &= -\lambda_j \hat{f}_j(x) - \lambda_j h_j(u) + \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}). \end{aligned}$$

Furthermore, for $\lambda_j = 0$ we have

$$\begin{aligned} \lambda_j {f_j}^{*2}(x, \tfrac{w^j}{\lambda_j}, u) &= \delta^*_{dom(({f_j}^{*2})^{*2}(x,.,u))}(w^j) \\ &= \delta^*_{dom(f_j(x,.,u))}(w^j) \\ &= \delta^*_{dom(g_j)}(w^j) \\ &= \delta^*_{dom({g_j}^*)^*}(w^j), \end{aligned}$$

where the first equality follows from the definition of the partial convex perspective, the second equality holds because of the closedness and convexity of $f_j(x, ., u)$. Therefore, Theorem 1 and the above equivalences concludes the corollary. □

A natural question is whether (ARC) (or its equivalent form (3)) is convex with respect to x. In other words, for a given optimal decision rule and a worst-case scenario, whether optimization on x is convex. The following example shows that the answer to this question is negative in general.

Example 1 Consider an instance of (ARC) with $m = 1$, $\mathcal{X} = [1, +\infty)$, $\mathcal{U} = [1, 2]$, $f_0(x, y, u) = x^2uy$, and $f_1(x, y, u) = -x + \frac{1}{2}y^2 - u$. The partial conjugate of f_0 and f_1 with respect to their second argument are given by

$$f_0^{*2}(x, w^0, u) = \begin{cases} 0, & w^0 = x^2u, \\ \infty, & w^0 \neq x^2u, \end{cases} \qquad f_1^{*2}(x, w^1, u) = x + \frac{(w^1)^2}{2} + u.$$

Thus, the dual reformulation of (ARC) for this example is

$$\begin{aligned} \inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U},\lambda,w^0,w^1} \quad & -f_0^{*2}(x, w^0, u) - \lambda f_1^{*2}\left(x, \frac{w^1}{\lambda}, u\right) \\ \text{s.t.} \quad & w^0 + w^1 = 0,\ \lambda \geq 0, \\ & w^0 \in dom\left(f_0^{*2}(x, \cdot, u)\right),\ \frac{w^1}{\lambda} \in dom\left(f_1^{*2}(x, \cdot, u)\right). \end{aligned} \tag{5}$$

For fixed x and u, if $\lambda = 0$, then

$$\begin{aligned} -\lambda_1 f_1^{*2}(x, \frac{w^1}{\lambda_1}, u) &= -\delta^*_{dom(({f_j}^{*2})^{*2}(x,\cdot,u))}(w^1) \\ &= -\delta^*_{dom(f_j(x,\cdot,u))}(w^1) \\ &= -\delta^*_{\mathbb{R}}(w^1) = -\delta_{\{0\}}(w^1). \end{aligned}$$

Thus, $-\lambda_1 f_1^{*2}(x, \frac{w^1}{\lambda_1}, u) = -\infty$ when $w^1 \neq 0$. Furthermore, $w^1 = 0$ is infeasible, due to $w^1 = -w^0 = -x^2u \neq 0$. So we can ignore $\lambda = 0$.
So, (5) is equivalent to

$$\begin{aligned} \inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U}} \sup_{\lambda,w^0,w^1} \quad & -\frac{(w^1)^2}{2\lambda} - \lambda(x + u) \\ \text{s.t.} \quad & w^0 + w^1 = 0,\ \lambda > 0,\ w^0 = x^2u. \end{aligned} \tag{6}$$

Given x and u, the inner suprimum can be written as

$$\sup_{\lambda>0} -\frac{x^4u^2}{2\lambda} - \lambda(x + u), \tag{7}$$

as we know $x \geq 1$ and $u \geq 1$. Moreover, the objective function is concave in λ. So, the supremum happens at $\lambda = \sqrt{\frac{x^4u^2}{2(x+u)}}$. Hence, (6) is equivalent to

$$\inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U}} -\sqrt{2}x^2u\sqrt{x + u}.$$

Given $x \geq 1$, the inner suprimum is $\sup\left\{-\sqrt{2}x^2u\sqrt{x+u}\,\middle|\, u \in [1, 2]\right\}$, whose objective function is decreasing on the given interval $[1, 2]$. Therefore, $u = 1$ is the worst-case scenario, and so

$$\sup_{u\in\mathcal{U}} -\sqrt{2}x^2u\sqrt{x + u} = -\sqrt{2}x^2\sqrt{x + 1}.$$

Finally, we get $\inf_x \left\{-\sqrt{2}x^2\sqrt{x+1}\,\middle|\, x \in \mathcal{X}\right\}$, which is a non-convex problem. □

Remark 3 Note that

$$ {f_j}^{*2}(x, w^j, u) = \sup_y \left\{ \left(w^j\right)^\top y - f_j(x, y, u) \right\}$$

implies

$$-{f_j}^{*2}(x, w^j, u) = \inf_y \left\{ -\left(w^j\right)^\top y + f_j(x, y, u) \right\}.$$

Indeed, the function $\mathcal{K}_{w^j,y,u}(x) := -\left(w^j\right)^\top y + f_j(x, y, u)$ is convex on x for any w^j, y, u, but the inf operator breaks down the convexity. In other words, the conjugate function $-{f_j}^{*2}(x, w^j, u)$ is not convex in x. Therefore, based on the dual formulation, we see if the conjugate function $-{f_j}^{*2}(x, w^j, u)$ is convex in x, then the problem is a convex optimization problem in x. In the next theorem, we show this for separable fixed-recourse problems. □

Theorem 2 *Under the assumption of Theorem 1, if (ARC) is separable fixed-recourse, then the dual reformulation of (ARC) is convex in x.*

Proof Since $f_j(x,y,u) = \hat{f}_j(x,u) + g_j(y)$ for all $j \in [m_0]$, according to Theorem 1, (ARC) is equivalent to

$$\begin{aligned} \inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U},\lambda_j,w^j} \;& \sum_{j=0}^{m} \lambda_j \hat{f}_j(x,u) - \sum_{j=0}^{m} \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}) \\ \text{s.t.} \;& \sum_{j=0}^{m} w^j = \mathbf{0},\ \lambda_0 = 1,\ \lambda_j \geq 0, \quad j \in [m], \\ & w^0 \in dom(g_0^*),\ \tfrac{w^j}{\lambda_j} \in dom(g_j^*), \quad j \in [m]. \end{aligned} \tag{8}$$

Note that $\hat{f}_j(x,u)$ is convex on x for each $j \in [m_0]$. By denoting

$$F_{u,\lambda,w^j}(x) := \sum_{j=0}^{m} \lambda_j \hat{f}_j(x,u) - \sum_{j=0}^{m} \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}), \quad \text{and} \quad \mathcal{F}(x) := \sup_{u,\lambda,w^j} F_{u,\lambda,w^j}(x),$$

which are convex on x, problem (8) is equivalent to $\inf_{x\in\mathcal{X}} \mathcal{F}(x)$ which is a convex optimization problem. □

Considering Theorem 2, we focus on separable fixed-recourse case in the rest of the paper.

4 On Upper Bound Calculation

In this section, we assume that the non-empty uncertainty set $\mathcal{U}$ has the following structure:

$$\mathcal{U} := \{u \in \mathbb{R}^{n_u} |\ c_i(u) \leq 0,\ i \in [t]\},$$

where the function $c_i : \mathbb{R}^{n_u} \to \bar{\mathbb{R}}$ is proper, closed, and convex for each $i \in [t]$.

In the next theorem, we show how using perspective functions result in an upper bound for (8).

Theorem 3 *For any fixed $x \in \mathcal{X}$, let $\hat{f}_j(x,u)$ be proper and concave in u for each $j \in [m_0]$. Then,*

$$\begin{aligned} \sup_{u,\lambda_j,w^j,\theta^j} \;& \sum_{j=0}^{m} \lambda_j \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) - \sum_{j=0}^{m} \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}) \\ \text{s.t.} \;& \sum_{j=0}^{m} w^j = \mathbf{0},\ \lambda_0 = 1,\ \theta^0 = u,\ \lambda_j \geq 0, \quad j \in [m], \\ & w^0 \in dom(g_0^*),\ \tfrac{w^j}{\lambda_j} \in dom(g_j^*), \quad j \in [m], \\ & \lambda_j c_i(\tfrac{\theta^j}{\lambda_j}) \leq 0, \quad j \in [m_0], i \in [t], \end{aligned} \tag{9}$$

provides an upper bound on the optimal value of

$$\begin{aligned} \sup_{u,\lambda_j,w^j} \;& \sum_{j=0}^{m} \lambda_j \hat{f}_j(x,u) - \sum_{j=0}^{m} \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}) \\ \text{s.t.} \;& \sum_{j=0}^{m} w^j = \mathbf{0},\ \lambda_0 = 1,\ \lambda_j \geq 0, \quad j \in [m], \\ & w^0 \in dom(g_0^*),\ \tfrac{w^j}{\lambda_j} \in dom(g_j^*), \quad j \in [m], \\ & c_i(u) \leq 0, \quad i \in [t]. \end{aligned} \tag{10}$$

Proof We show that any feasible solution to (10) corresponds to a feasible solution to (9) with the same objective value. For this goal, for any solution $\left(u, \lambda, \{w^j\}_{j=0}^m\right)$, define $\theta^j := \lambda_j u$ for each $j \in [m]$. We prove that $\left(u, \lambda, \{w^j\}_{j=0}^m, \{\theta^j\}_{j=1}^m\right)$ is a feasible solution to (9). For each $j \in [m]$ and $i \in [t]$ with $\lambda_j > 0$ obviously we have $\lambda_j c_i(\frac{\theta^j}{\lambda_j}) \leq 0$. If $\lambda_j = 0$ for some $j \in [m]$, we get $\theta^j = \mathbf{0}$ and

$$0c_i(\mathbf{0}/0) = \delta^*_{dom(c_i^*)}(\mathbf{0}) = 0, \quad \forall i \in [t],$$

where the first equality follows from the definition of the convex perspective, and the second equality holds because the conjugate function of a proper convex function is also proper (from [3, Theorem 4.5]), and so $dom(c_i^*) \neq \emptyset$. All other constraints of (9) are clearly satisfied. Now, we show that the objective value at $\left(u, \lambda, \{w^j\}_{j=0}^m\right)$ in (10) equals to that value at $\left(u, \lambda, \{w^j\}_{j=0}^m, \{\theta^j\}_{j=1}^m\right)$ in (9). To this end, it is sufficient to show that $\lambda_j \hat{f}_j(x, \frac{\theta^j}{\lambda_j}) = \lambda_j \hat{f}_j(x, u)$ for all $j \in [m]$. It is trivial for the case $\lambda_j > 0$. If $\lambda_j = 0$ for some $j \in [m]$, then $\theta^j = \mathbf{0}$, and so

$$0\hat{f}_j(x, \mathbf{0}/0) = -\delta^*_{dom\left((-\hat{f}_j)^{*2}(x,.)\right)}(\mathbf{0}) = 0 = 0\hat{f}_j(x, u),$$

where the first equality follows from the definition of the partial concave perspective, and the second equality holds because (for any x) the partial conjugate $(-\hat{f}_j)^{*2}(x, .)$ of proper convex function $-\hat{f}_j(x, .)$ is also proper, leading to $dom\left((-\hat{f}_j)^{*2}(x, .)\right) \neq \emptyset$. □

Problem (10) is not a convex programming in general, while problem (9) is. More specifically, by lifting the problem to a higher dimension and using the perspective functions, we obtain a concave relaxation on λ_j, w^j, θ^j. This approach has been recently used in the literature of nonlinear optimization for other purposes [16,27,45]. In the next example, we show that this relaxation may not be tight.

Example 2 Let $x \in \mathcal{X}$. Consider an instance of problem (10) with $t = m = 1$, $n_u = 1$, $n_y = 2$, $\hat{f}_0(x, u) = -u^2$, $\hat{f}_1(x, u) = \frac{1}{u}$, $c_1(u) = u + 1$, $g_0(y) = y_1$, and $g_1(y) = \frac{1}{2}y^\top y + y_2$. Set $p^0 := (1, 0)^\top$, $p^1 := (0, 1)^\top$. The conjugates of g_0 and g_1 are given by

$$g_0^*(w^0) = \begin{cases} 0, & w^0 = p^0, \\ \infty, & w^0 \neq p^0, \end{cases}$$
$$g_1^*(w^1) = \tfrac{1}{2}\left(w^1 - p^1\right)^\top \left(w^1 - p^1\right).$$

Hence, problem (10) in this example reads as

$$\begin{aligned} \sup_{u,\lambda_1,w^0,w^1} \quad & \hat{f}_0(x, u) + \lambda_1 \hat{f}_1(x, u) - g_0^*(w^0) - \lambda_1 g_1^*(\tfrac{w^1}{\lambda_1}) \\ \text{s.t.} \quad & \lambda_1 \geq 0,\ w^0 + w^1 = \mathbf{0},\ u \leq -1, \\ & w^0 = p^0. \end{aligned}$$

If $\lambda_1 = 0$ in some feasible solution of the above problem, then

$$-\lambda_1 g_1^*(\tfrac{w^1}{\lambda_1}) = -\delta^*_{dom(g_1)}(w^1) = -\delta_{\{\mathbf{0}\}}(w^1).$$

So, $-\lambda_1 g_1^*(\frac{w^1}{\lambda_1}) = -\infty$ when $w^1 \neq \mathbf{0}$. Furthermore, $w^1 = \mathbf{0}$ is infeasible, due to $w^1 = -w^0 = -p^0 = (-1, 0)^\top$. Hence, we can ignore $\lambda_1 = 0$. Now, due to $w^0 = p^0$ and $w^0 + w^1 = \mathbf{0}$, the last problem can be rewritten as

$$\begin{aligned} z_1 := \sup_{u,\lambda_1} \quad & -u^2 + \tfrac{\lambda_1}{u} - \tfrac{1}{2}\left(\lambda_1 + \tfrac{1}{\lambda_1}\right) \\ \text{s.t.} \quad & \lambda_1 > 0,\ u \leq -1. \end{aligned}$$

Let us denote the objective function of the last problem by

$$J(u, \lambda_1) = -u^2 + \tfrac{\lambda_1}{u} - \tfrac{1}{2}\left(\lambda_1 + \tfrac{1}{\lambda_1}\right).$$

This function is bounded above over the feasible set $\mathcal{K} = \{(u, \lambda_1)|\ u \leq -1,\ \lambda_1 > 0\}$. To obtain z_1, first we examine the points for which the gradient of $J(\cdot, \cdot)$ vanishes. We have

$$\nabla J(u, \lambda_1) = (-2u - \tfrac{\lambda_1}{u^2}, \tfrac{1}{u} - \tfrac{1}{2} + \tfrac{1}{2\lambda_1^2})^\top.$$

Thus, $\nabla J(u, \lambda_1) = 0$ implies $\lambda_1 = -2u^3$, and $-4u^6 + 8u^5 + 1 = 0$. On the other hand, $u \leq -1$ (feasibility) leads $-4u^6 + 8u^5 + 1 < 0$. So, the maximizers of $J(u, \lambda_1)$ are not in the interior of the feasible set $\mathcal{K}$. They are on the boundary of $\mathcal{K}$, i.e., $\lambda_1 = 0$ or $u = -1$. As $\lambda_1 > 0$, we continue with $u = -1$, and we have

$$\sup_{\lambda_1 > 0} \quad -1 - \lambda_1 - \tfrac{1}{2}\left(\lambda_1 + \tfrac{1}{\lambda_1}\right).$$

The optimal solution of the above problem occurs at $\lambda_1 = \frac{1}{\sqrt{3}}$, and hence $z_1 = -1 - \sqrt{3}$. Analogously, problem (9) in this example is

$$\begin{aligned} z_2 := \sup_{u,\lambda_1,\theta_1} \quad & -u^2 + \tfrac{(\lambda_1)^2}{\theta_1} - \tfrac{1}{2}\left(\lambda_1 + \tfrac{1}{\lambda_1}\right) \\ \text{s.t.} \quad & \lambda_1 > 0,\ u \leq -1,\ \theta_1 \leq -\lambda_1. \end{aligned}$$

It is not difficult to see that -2 is an upper bound for the objective function of the above problem on its feasible region. Furthermore, the objective value at the feasible sequence $\{u_n = -1,\ (\lambda_1)_n = 1,\ (\theta_1)_n = -n\}_{n\geq 1}$ equals to $-2-\frac{1}{n}$ which goes to -2 as $n \to +\infty$. This implies $z_2 = -2$. Therefore, $z_1 = -1-\sqrt{3} < -2 = z_2$. □

In Theorem 3, $x \in \mathcal{X}$ is fixed and arbitrary. Now, by taking minimum over all $x \in \mathcal{X}$ in (10) and (9), we obtain an upper bound for the separable fixed-recourse version of the dual reformulation of (ARC) as follow:

$$\begin{aligned} \inf_{x\in\mathcal{X}} \sup_{u,\lambda_j,w^j,\theta^j} \ & \sum_{j=0}^{m} \lambda_j \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) - \sum_{j=0}^{m} \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}) \\ \text{s.t.} \ & \sum_{j=0}^{m} w^j = \mathbf{0},\ \lambda_0 = 1,\ \theta^0 = u,\ \lambda_j \geq 0, \quad j \in [m], \\ & w^0 \in dom(g_0^*),\ \tfrac{w^j}{\lambda_j} \in dom(g_j^*), \quad j \in [m], \\ & \lambda_j c_i(\tfrac{\theta^j}{\lambda_j}) \leq 0, \quad j \in [m_0],\ i \in [t]. \end{aligned} \tag{PERS}$$

We call this problem (PERS) as it is obtained by using a *perspectification approach* corresponding to problem (8). Moreover, (PERS) is a convex-concave programming, while it is not the case for (8).

Problem (PERS) can be seen as a relaxation of the dual reformulation of (ARC) when it has separable fixed-recourse. So, it is important to know the interpretation of such a relaxation for the primal problem, i.e., (ARC). The next theorem shows that (PERS) is actually equivalent to the static robust counterpart (RC) in the separable fixed-recourse case when the uncertainty set is compact.

Theorem 4 *Consider* (RC) *with separable fixed-recourse as follows:*

$$\begin{aligned} \inf_{x\in\mathcal{X},y} \ & \sup_{u\in\mathcal{U}} \hat{f}_0(x, u) + g_0(y) \\ \text{s.t.} \ & \sup_{u\in\mathcal{U}} \hat{f}_j(x, u) + g_j(y) \leq 0, \quad j \in [m]. \end{aligned} \tag{11}$$

Suppose that the uncertainty set $\mathcal{U}$ is compact, $\hat{f}_j$ is proper concave in u, and g_j is closed convex and real-valued, for each $j \in [m_0]$. If

$$\forall x \in \mathcal{X}\ \exists y \text{ such that } \sup_{u\in\mathcal{U}} \hat{f}_j(x, u) + g_j(y) < 0, \quad j \in [m],$$

then (PERS) *and static robust counterpart* (11) *are equivalent.*

Proof Without loss of generality, since $\mathcal{U}$ is compact, we assume that there exists some i, for which $c_i(u) = ||u||_2 - \rho$, for some $\rho > 0$. By setting $F_j(x) := \sup_u \left\{ \hat{f}_j(x, u) \middle| \ u \in \mathcal{U} \right\}$ for each $j \in [m_0]$ and $x \in \mathcal{X}$, we can rewrite (11) as

$$\begin{aligned} \inf_{x\in\mathcal{X},y\in\mathbb{R}^{n_y}} \ & F_0(x) + g_0(y) \\ \text{s.t.} \ & F_j(x) + g_j(y) \leq 0, \quad j \in [m]. \end{aligned} \tag{12}$$

By applying the Fenchel's duality over y, (12) is equivalent to

$$\begin{aligned} \inf_{x\in\mathcal{X}} \sup_{\lambda_j,w^j} \ & \sum_{j=0}^{m} \lambda_j F_j(x) - \sum_{j=0}^{m} \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}) \\ \text{s.t.} \ & \sum_{j=0}^{m} w^j = \mathbf{0},\ \lambda_0 = 1,\ \lambda_j \geq 0, \quad j \in [m], \\ & w^0 \in dom(g_0^*),\ \tfrac{w^j}{\lambda_j} \in dom(g_j^*), \quad j \in [m]. \end{aligned} \tag{RC-1}$$

Now, we show that, for a given $x \in \mathcal{X}$, the inner suprimums of (RC-1) and (PERS) have the same optimal value. To prove this claim, let $x \in \mathcal{X}$ be fixed. Let $\left(u, \lambda, \{w^j\}_{j=0}^m, \{\theta^j\}_{j=1}^m\right)$ be a feasible solution for the inner suprimum in (PERS).
If $\lambda_j > 0$, then

$$\begin{aligned} \lambda_j c_i(\tfrac{\theta^j}{\lambda_j}) \leq 0, \quad \forall i \in [t] \Rightarrow c_i(\tfrac{\theta^j}{\lambda_j}) \leq 0, \quad \forall i \in [t] & \\ \Rightarrow \tfrac{\theta^j}{\lambda_j} \in \mathcal{U} \Rightarrow \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) \leq F_j(x) & \\ \Rightarrow \lambda_j \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) \leq \lambda_j F_j(x). & \end{aligned}$$

If $\lambda_j = 0$, then $\theta^j = \mathbf{0}$. To prove this, as $c_i(u) = \|u\|_2 - \rho$ for some $\rho > 0$ and some $i \in [t]$, by taking $\lambda_j = 0$ into account, we have

$$0 \geq \lambda_j c_i(\tfrac{\theta^j}{\lambda_j}) = \delta^*_{dom(c_i^*)}(\theta^j) = \sup_{\|\gamma\|_2 \leq 1} \left\{\gamma^\top \theta^j\right\} = \|\theta^j\|_2 \geq 0.$$

This implies $\theta^j = \mathbf{0}$. Hence, in this case

$$\lambda_j \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) = -\delta^*_{dom((-f_j)^{*2}(x,.))}(\mathbf{0}) = 0 = \lambda_j F_j(x),$$

where first equality comes from the definition of the partial concave perspective, and the second equality holds as $dom\left((-f_k)^{*2}(x,.)\right) \neq \emptyset$. So,

$$\lambda_j \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) \leq \lambda_j F_j(x), \quad j \in [m_0].$$

Summing over j yields $\sum_j \lambda_j \hat{f}_j(x, \frac{\theta^j}{\lambda_j}) \leq \sum_j \lambda_j F_j(x)$. Thus,

$$\sum_{j=0}^{m} \lambda_j \hat{f}_j(x, \tfrac{\theta^j}{\lambda_j}) - \sum_{j=0}^{m} \lambda_j g_j{}^*(\tfrac{w^j}{\lambda_j}) \leq \sum_{j=0}^{m} \lambda_j F_j(x) - \sum_{j=0}^{m} \lambda_j g_j{}^*(\tfrac{w^j}{\lambda_j}).$$

Therefore, the optimal value of the objective function of the inner suprimum in (PERS) is less than or equal to that in (RC-1).

Conversely, let $\left(\bar{\lambda}, \{\bar{w}^j\}_{j=0}^m\right)$ be a feasible solution for inner suprimum of (RC-1). By choosing

$$\bar{u}^j \in \text{argmax}\{\hat{f}_j(x, u) |\ u \in \mathcal{U}\}, \quad j \in [m_0],$$

and setting

$$\bar{\theta}^j = \begin{cases} \bar{\lambda}_j \bar{u}^j, & \lambda_j > 0, \\ 0, & \lambda_j = 0, \end{cases} \qquad j \in [m],$$

and then setting $\bar{\theta}^0 = \bar{u}$, the vector $\left(\bar{u}, \bar{\lambda}, \{\bar{w}^j\}_{j=0}^m, \{\bar{\theta}^j\}_{j=1}^m\right)$ is feasible for (PERS). Furthermore, for $\bar{\lambda}_j > 0$,

$$\bar{\lambda}_j F_j(x) = \bar{\lambda}_j \sup_{u \in \mathcal{U}} \hat{f}_j(x, u) = \bar{\lambda}_j \hat{f}_j(x, \bar{u}^j) = \bar{\lambda}_j \hat{f}_j(x, \tfrac{\bar{\theta}^j}{\bar{\lambda}_j}).$$

This equality is trivial for $\bar{\lambda}_j = 0$. Hence,

$$\sum_{j=0}^{m} \bar{\lambda}_j F_j(x) - \sum_{j=0}^{m} \bar{\lambda}_j g_j{}^*(\tfrac{\bar{w}^j}{\bar{\lambda}_j}) = \sum_{j=0}^{m} \bar{\lambda}_j \hat{f}_j(x, \tfrac{\bar{\theta}^j}{\bar{\lambda}_j}) - \sum_{j=0}^{m} \lambda_j g_j{}^*(\tfrac{\bar{w}^j}{\bar{\lambda}_j}).$$

This implies that the optimal value of the objective function of (RC-1) is less than or equal to that in (PERS). This completes the proof. □

Theorem 4 states that, under some assumptions, the upper bound obtained based on the perspective relaxation of the dual reformulation of (ARC) is the same as the robust counterpart, which is a conservative approximation. In other words, the perspectification approach yields an upper bound for (ARC); nevertheless, there are stronger upper bounds in the literature of adjustable robust optimization, such as K-*adaptability* or *finite adaptability* approaches [34,43], which can straightforwardly be extended to nonlinear problems. One way to obtain a stronger upper bound is by applying a piece-wise constant decision rule to (ARC) using finite adaptability approach. In this approach, the uncertainty set is partitioned into subsets and a constant decision rule is obtained for each of the subsets. In the numerical experiments, we show how much stronger the upper bound obtained by finite adaptability compared to the one obtained from (PERS).

5 Lower Bound Calculation

In Section 3, we showed that the dual reformulation of (ARC) in the fixed-resource case is a convex programming on here-and-now decision variables. We have also shown how to construct an upper bound. In this section, we design methods to approximate (ARC) from below. Let us set

$$v := \begin{pmatrix} \lambda \\ \{w^j\}_{j=0}^m \end{pmatrix} \in \mathbb{R}^m \times \mathbb{R}^{n_y(m+1)},$$

$$G(v) := -g_0^*(w^0) - \sum_{j=1}^m \lambda_j {g_j}^*(\tfrac{w^j}{\lambda_j}),$$

$$\mathcal{V} := \left\{ v = \begin{pmatrix} \lambda \\ \{w^j\}_{j=0}^m \end{pmatrix} \middle| \begin{array}{l} \sum_{j=0}^m w^j = \mathbf{0},\ \lambda \geq \mathbf{0}, \\ w^0 \in dom(g_0^*), \\ \frac{w^j}{\lambda_j} \in dom(g_j^*),\ j \in [m] \end{array} \right\}.$$

Since $\lambda_j {g_j}^*(\frac{w^j}{\lambda_j})$ is jointly convex in (w^j, λ_j), the set $\mathcal{V}$ is convex and G is a concave function (details are provided in Appendix 1). Also, let us set

$$F(x,u) := \left(\hat{f}_0(x,u), \dots, \hat{f}_m(x,u), 0, \dots, 0\right)^\top \in \mathbb{R}^{m+1} \times \mathbb{R}^{n_y(m+1)},$$

$$\mathcal{L}(x,u,v) := (1, v^\top) F(x,u) + G(v).$$

Thus, the dual formulation of (ARC) in the separable fixed-recourse case reads as

$$\inf_{x \in \mathcal{X}} \sup_{\substack{u \in \mathcal{U} \\ v \in \mathcal{V}}} \mathcal{L}(x,u,v). \tag{13}$$

Given $\bar{x} \in \mathcal{X}$, we define $\mathcal{L}_{\bar{x}}(u,v) := (1, v^\top) F(\bar{x}, u) + G(v)$. Clearly, $\mathcal{L}_{\bar{x}}(u,v)$ is concave in u and concave in v. Therefore,

$$\sup \left\{ \mathcal{L}_{\bar{x}}(u,v) \mid \ u \in \mathcal{U},\ v \in \mathcal{V} \right\}, \tag{14}$$

is a disjoint biconcave maximization problem. A common way to find a solution for such problems is by using alternating methods, which obtain a local optimizer. In these methods, a decision variable is divided into several blocks, and optimization can be performed explicitly in each block when the variables of other blocks are fixed (see Chapter 14 of [3] for more details). These methods also appear in the literature as block coordinate methods. The performance of the alternating method is closely related to finding the optimizers for each block.

5.1 Alternating Iterative Algorithm

As was mentioned above, we use a two-block alternating method to solve (14). In this method, we alternatively fix u to find v and fix v to find u until no improvement is achieved or the prescribed computational limit is reached. This method is described follow in detail.

Alternating Method

Input: initial value $\bar{u}^{(0)} \in \mathcal{U}$
Initialization:

Set iteration counter $k \leftarrow 0$, choose $\bar{v}^{(0)} \in \operatorname{argmax}_{v \in \mathcal{V}} \ \mathcal{L}_{\bar{x}}(\bar{u}^{(0)}, v)$.

Repeat

Find optimal u: $\bar{u}^{(k+1)} \in \operatorname{argmax}_{u \in \mathcal{U}} \ \mathcal{L}_{\bar{x}}(u, \bar{v}^{(k)})$,
Find optimal v: $\bar{v}^{(k+1)} \in \operatorname{argmax}_{v \in \mathcal{V}} \ \mathcal{L}_{\bar{x}}(\bar{u}^{(k+1)}, v)$,
Update iteration counter $k \leftarrow k+1$,

Until: time limit is reached, or no improvement is possible.
Return: $(\bar{u}^{(k)}, \bar{v}^{(k)})$.

In Theorem 5 below, we discuss the convergence of the addressed alternating method. It is done assuming some appropriate conditions, under which the alternating method is well-defined and the sequence $\{(u^k, v^k)\}_{k\geq 0}$ admits limit point(s).

Theorem 5 *Let $\mathcal{L}_{\bar{x}}(\cdot,\cdot)$ be continuously differentiable and bounded above on Cartesian product of two closed convex sets $\mathcal{U}$ and $\mathcal{V}$. Suppose that every sub-problem of the alternating method has an optimal solution and $\{z^k \equiv (u^k, v^k)\}_{k\geq 0}$, as the sequence generated by the alternating method, has at least a limit point. Then, every limit point of $\{z^k\}_{k\geq 0}$ is a stationary point of problem (14).*

Proof See Appendix 2. □

Theorem 5 provides conditions under which the limit points of the sequence obtained by the alternating method are helpful in solving problem (14). These conditions can be checked for (1). More specifically, $\mathcal{L}$ is continuously differentiable if $\hat{f}_j$ and g_j^* are so. Furthermore, it is bounded above if static robust counterpart (11) has an optimal solution. Finally, $\mathcal{V}$ is a closed set when $dom(g_j^*)$ is closed for all j.

Remark 4 In Theorem 5, under some assumptions, it is established that all limit points of the sequence generated by the alternating method are stationary. Generally, stationarity is necessary for local optimality [4]. However, stationary points are not necessarily optimal solutions. Such a property requires (generalized) concavity assumption to hold. Under generalized concavity assumptions on problem (14), the alternating method globally converges; For more details, see [19, Proposition 6] and [50]. □

Using this theorem, we can find a lower bound for (ARC) in the following way: starting from initial solution $x^{(0)}$ and initial scenario $u^{(0)}$, we can find the limit points of $\{z^k\}$, denoted by $\bar{z}^{(0)}$. Let us denote by $\bar{\mathcal{W}}$ the set of limit points after each iteration. Limiting ourselves to $\bar{\mathcal{W}}$, we can find a here-and-now solution $x^{(\ell)}$ using the following optimization problem

$$\begin{aligned} \inf_{x,\tau} \quad & \tau \\ \text{s.t.} \quad & \mathcal{L}(x, \bar{u}^{(i)}, \bar{v}^{(i)}) - \tau \leq 0, \quad 1 \leq i \leq |\bar{\mathcal{W}}|, \\ & x \in \mathcal{X}. \end{aligned} \tag{P-1}$$

By fixing this decision, we can find new limit point $\bar{z}^{(\ell)}$ to be added to $\bar{\mathcal{W}}$. Algorithm 1 provides the pseudo-code of this procedure.

Algorithm 1

Input: $\epsilon > 0$, initial value $x^{(0)} \in \mathcal{X}$, $u^{(0)} \in \mathcal{U}$.
Initialization: Set iteration counter $\ell \leftarrow 0$, and set $\bar{\mathcal{W}} = \emptyset$.
Repeat: Execute the following steps:

> **(Step 1)** obtain $(\bar{u}^{(\ell)}, \bar{v}^{(\ell)})$ as a stationary point of $\mathcal{L}_{x^{(\ell)}}(u, v)$ by applying Alternating Method.
> Set $\bar{\mathcal{W}} = \bar{\mathcal{W}} \cup \{(\bar{u}^{(\ell)}, \bar{v}^{(\ell)})\}$.
> **(Step 2)** Find (x^*, τ^*) by solving (P-1).
> Update iteration counter $\ell \leftarrow \ell + 1$, and set
> $x^{(\ell)} \leftarrow x^*$,
> $\tau^{(\ell)} \leftarrow \tau^*$,

Until: $\|\tau^{(\ell)} - \tau^{(\ell-1)}\| \leq \epsilon$.
Return: $x^{(\ell)}$, $\tau^{(\ell)}$.

It can be seen that the optimal value of (P-1) in Algorithm 1 is a lower bound for problem (13). Since, we add one more constraint to (P-1) in each iteration, the sequence of the lower bounds is non-decreasing.

Another way to generate lower bounds is to use finite-scenario approach. In the next section, we show how one can improve the lower bounds obtained by finite-scenario approach.

5.2 Dual-Based Cutting Plane Algorithm

Based on the original form of (ARC), one can find a lower bound by using finite-scenario approach and only considering a finite subset $\{u^1, ..., u^\ell\}$ of $\mathcal{U}$. This idea leads to the following convex programming problem:

$$\begin{aligned} &\inf_{\substack{x\in\mathcal{X},\tau\\ \{y^k\}_k}} \ \tau \\ &\text{s.t.} \ \ f_j(x,y^k,u^k)\le 0, \quad j\in[m],\ k\in[\ell], \\ &\qquad f_0(x,y^k,u^k)\le\tau, \quad k\in[\ell], \end{aligned} \tag{15}$$

which is called finite-scenario approach of the (ARC) problem. A technique to obtain a finite set of scenarios is by (i) approximating (ARC) with a suitable decision rule and (ii) finding the active scenarios in the uncertainty set [20]. Since we are considering a nonlinear problem, we use constant decision rule. So, we first find an optimal solution (x^*,y^*) of (RC) with the optimal value t^*. After that, by fixing the obtained (x^*,y^*), we take an active (binding) scenario on each constraint

$$\begin{aligned} &f_j(x^*,y^*,u)\le 0, \quad j\in[m], \\ &f_0(x^*,y^*,u)\le t^*. \end{aligned} \tag{16}$$

The optimal value of the finite-scenario approach problem is a lower bound for the optimal objective value of the original (ARC) model since feasibility is fulfilled for only a subset of the uncertainty set.

In Theorem 6 we show how to construct a lower bound by means of dual cuts.

Theorem 6 *Let $\{u^1,...,u^\ell\}\subseteq\mathcal{U}$ and $\{v^1,...,v^\ell\}\subseteq\mathcal{V}$. Then optimal value of*

$$\begin{aligned} &\inf_{\substack{x\in\mathcal{X},\tau\\ \{y^k\}_k}} \ \tau \\ &\text{s.t.} \ \ f_j(x,y^k,u^k)\le 0, \quad j\in[m],\ k\in[\ell], \\ &\qquad f_0(x,y^k,u^k)\le\tau, \quad k\in[\ell], \\ &\qquad \mathcal{L}(x,u^k,v^k)\le\tau, \quad k\in[\ell], \end{aligned} \tag{17}$$

provides a lower bound for (ARC).

Proof Let us denote the optimal value of (ARC) by *Opt*. Also, we have the following inequality due to the weak duality

$$\forall x\in\mathcal{X} \quad \sup_{u\in\mathcal{U}}\mathcal{T}(x,u)\ge\sup_{\substack{u\in\mathcal{U}\\ v\in\mathcal{V}}}\mathcal{L}(x,u,v),$$

where $\mathcal{T}(x,u):=\inf_{y\in\mathbb{R}^{n_y}}\{f_0(x,y,u):\ f_j(x,y,u)\le 0,\ j\in[m]\}$. Therefore,

$$\begin{aligned} Opt = &\inf_{\substack{x\in\mathcal{X}\\ \tau\in\mathbb{R}}} \ \tau \\ &\text{s.t.} \ \sup_{u\in\mathcal{U}}\mathcal{T}(x,u)\le\tau, \\ &\qquad \sup_{\substack{u\in\mathcal{U}\\ v\in\mathcal{V}}}\mathcal{L}(x,u,v)\le\tau, \end{aligned}$$

where the second constraint is redundant. Let $\bar{\mathcal{U}}\subseteq\mathcal{U}$ and $\bar{\mathcal{V}}\subseteq\mathcal{V}$. Then

$$\begin{aligned} Opt \ge &\inf_{x\in\mathcal{X},\tau} \ \tau \\ &\text{s.t.} \ \sup_{u\in\bar{\mathcal{U}}}\mathcal{T}(x,u)\le\tau, \\ &\qquad \sup_{\substack{u\in\bar{\mathcal{U}}\\ v\in\bar{\mathcal{V}}}}\mathcal{L}(x,u,v)\le\tau. \end{aligned}$$

So, if $\bar{\mathcal{U}}=\{u^1,...,u^\ell\}$ and $\bar{\mathcal{V}}=\{v^1,...,v^\ell\}$, then we have

$$\begin{aligned} Opt \ge &\inf_{x\in\mathcal{X},\tau} \ \tau \\ &\text{s.t.} \ \inf_{y\in\mathbb{R}^{n_y}}\left\{f_0(x,y,u^k):\ f_j(x,y,u^k)\le 0,\ j\in[m]\right\}\le\tau, \quad k\in[\ell], \\ &\qquad \mathcal{L}(x,u^k,v^k)\le\tau, \quad k\in[\ell], \end{aligned} \tag{18}$$

which is equivalent to (17). □

Using Theorem 6, we develop Algorithm 2, which generates potentially better lower bounds compared to finite-scenario approach.

Algorithm 2

(Step 1) Find (x^*, y^*) as a static solution.
(Step 2) Given (x^*, y^*), find the active scenarios $\{u^0, u^1, ..., u^m\}$.
(Step 3) Use Alternating Method, to find $(\bar{u}^{(k)}, \bar{v}^{(k)})$ as a stationary point of $\mathcal{L}_{x^*}(u, v)$ starting from u^k $(k \in [m_0])$.
(Step 4) Given $\{(\bar{u}^{(k)}, \bar{v}^{(k)})\}_k$, solve

$$\begin{aligned}\inf_{\substack{x\in\mathcal{X},\tau\\ \{y^k\}_k}} \quad & \tau \\ \text{s.t.} \quad & f_0(x, y^k, \bar{u}^{(k)}) \leq \tau,\ k \in [m_0], \\ & f_j(x, \bar{u}^{(k)}, y^k) \leq 0,\ j \in [m],\ k \in [m_0], \\ & \mathcal{L}(x, \bar{v}^{(k)}, \bar{u}^{(k)}) \leq \tau,\ k \in [m_0].\end{aligned}$$

We emphasize that the algorithms to construct a sequence of lower bounds are applicable for any nonlinear ARO problem.

6 Numerical Experiments

In this section, we illustrate the performance of the discussed algorithms. All the numerical results were carried out on a laptop featuring Intel(R) Core(TM) i5-3210M CPU, 2.50 GHz processor, and 8 GB of RAM. We implemented the algorithms in MATLAB (2022a) and used YALMIP toolbox [29] to pass the optimization problems to MOSEK as a solver [33]. All results of this section are presented with four decimals.

6.1 Problem Setting

We consider the following uncertain problem:

$$\inf_{x\in\mathcal{X}} \inf_{y\in\mathcal{Y}(x,u)} \hat{f}_0(x, u) + \left\|A_0 y - b^0\right\|_2 - \left(p^0\right)^\top y + q_0, \tag{19}$$

where

$$\mathcal{Y}(x, u) = \left\{y : \hat{f}_j(x, u) + \left\|A_j y - b^j\right\|_2 - \left(p^j\right)^\top y + q_j \leq 0,\ j \in [m]\right\} \subseteq \mathbb{R}^{n_y},$$

$A_j \in \mathbb{R}^{r\times n_y}$, $b^j \in \mathbb{R}^r$, and $p^j \in \mathbb{R}^{n_y}$. For $j \in [m]$, let us set

$$g_j(y) := \left\|A_j y - b^j\right\|_2 - \left(p^j\right)^\top y + q_j.$$

Thus, the perspective functions corresponding to the conjugate of $g_j(y)$ is given by

$$\begin{aligned}\lambda_j > 0: \quad & \lambda_j g_j^*\left(\frac{w^j}{\lambda_j}\right) = \inf_{z^j}\left\{\lambda_j((b^j)^\top z^j - q_j)\middle|\ \|z^j\|_2 \leq 1,\ (A_j)^\top z^j - p^j = \frac{w^j}{\lambda_j}\right\}, \\ \lambda_j = 0: \quad & \lambda_j g_j^*\left(\frac{w^j}{\lambda_j}\right) = \delta^*_{dom(g_j)}(w^j) = \delta^*_{\mathbb{R}^{n_y}}(w^j) = \delta_{\{0\}}(w^j).\end{aligned}$$

Now we consider (ARC) version of the uncertain problem (19). According to Theorem 1, after dualizing over the wait-and-see variable y, with some algebra (see Section 6.4 of [37]), we get the following equivalent dual reformulation:

$$\begin{aligned}\inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U},\lambda_j,z^j} \quad & \sum_{j=0}^m \lambda_j \hat{f}_j(x, u) + \sum_{j=0}^m \left(\lambda_j q_j - \lambda_j \left(b^j\right)^\top z^j\right) \\ \text{s.t.} \quad & \sum_{j=0}^m \left(\lambda_j (A_j)^\top z^j - \lambda_j p^j\right) = \mathbf{0}, \\ & \left\|z^j\right\|_2 \leq 1,\ \lambda_0 = 1,\ \lambda_j \geq 0, \quad j \in [m_0].\end{aligned} \tag{20}$$

Let us consider the parameters in a matrix form, i.e.,

$$A^\top := \left[(A_0)^\top \dots (A_m)^\top\right] \in \mathbb{R}^{n_y \times r(m+1)},$$
$$P := \left[p^0 \dots p^m\right] \in \mathbb{R}^{n_y \times (m+1)},$$
$$b^\top := [(\,b^0\,)^\top \dots (\,b^m\,)^\top] \in \mathbb{R}^{r(m+1)},$$
$$\bar{z}^j := \lambda_j z^j,\ \bar{z}^\top := [(\,\bar{z}^0\,)^\top \dots (\,\bar{z}^m\,)^\top] \in \mathbb{R}^{r(m+1)}.$$

In addition, by setting

$$v^\top := \left(\lambda^\top, \bar{z}^\top\right),$$
$$\mathcal{V} := \left\{ v \,\middle|\, A^\top \bar{z} - P\begin{pmatrix}1\\ \lambda\end{pmatrix} = \mathbf{0},\ \lambda \geq \mathbf{0},\ \left\|\bar{z}^0\right\|_2 \leq 1,\ \left\|[\bar{z}]^j\right\|_2 \leq \lambda_j,\ j \in [m] \right\},$$
$$\mathcal{L}(x,u,v) := (1,\lambda^\top)\hat{f}(x,u) + (1,\lambda^\top)q - \bar{z}^\top b,$$

where $\hat{f}(x,u)$ is a vector-valued function with components $\hat{f}_j(x,u)$, and

$$[\bar{z}]^j = [\bar{z}]_{jr+1,\dots,(j+1)r},$$

we can write (20) as

$$\inf_{x\in\mathcal{X}} \sup_{u\in\mathcal{U},v\in\mathcal{V}} \mathcal{L}(x,u,v).$$

We use the optimality gap to compare the quality of the lower bounds obtained by applying the finite-scenario approach (15), Algorithm 1, and Algorithm 2:

$$OptGap = \left(\frac{UB - LB}{|LB| + 10^{-4}}\right) \times 100,$$

where LB is the obtained lower bound, and UB is the best obtained upper bound for a given instance. Note that the constant 10^{-4} is added to the denominator to avoid division over zero.

We consider two classes of randomly generated problems, each containing 100 instances.
Class One: In this class, we consider small-sized instances. We consider $n_y = 2$, $n_x = 2$, $m = 2$, and $r = 5$. Furthermore,

$$\mathcal{U} = \mathcal{U}_1 := \{u \in \mathbb{R}^2 :\ \|u\|_2 \leq 1\},$$
$$\mathcal{X} = \mathcal{X}_1 := \left\{x \in \mathbb{R}^2 :\ x_1 + 2x_2 \leq 3,\ 2x_1 + x_2 \leq 3,\ x_1, x_2 \geq 0\right\},$$

and

$$\hat{f}_j(x,u) := {c^j}^\top x + {\alpha^j}^\top u, \quad j \in [m_0].$$

In this class, we obtained upper bounds by solving the static robust counterpart problem, which is equivalent to the perspectification approach. Additionally, we employed the K-adaptability approach to obtain other upper bounds. The K-adaptability approach involves splitting the uncertainty set $\mathcal{U}$ into K partitions ($\mathcal{U} = \cup_{k=1}^K \mathcal{U}_k$) and then solving the following problem

$$\begin{aligned} \inf_{\substack{x\in\mathcal{X},\tau\\ \{y^k\}_k}}\ & \tau \\ \text{s.t.}\ & \hat{f}_j(x,u) + \left\|A_j y^k - b^j\right\|_2 - \left(p^j\right)^\top y^k + q_j \leq 0,\ j \in [m], k \in [K], \forall u \in \mathcal{U}_k, \\ & \hat{f}_0(x,u) + \left\|A_0 y^k - b^0\right\|_2 - \left(p^0\right)^\top y^k + q_0 \leq \tau,\ k \in [K], \forall u \in \mathcal{U}_k. \end{aligned}$$

We set $K = 8$ and partitioned the uncertainty set into eight regions, each being an octant.
Class Two: This class contains large-sized instances. We consider $n_y = 100$, $n_x = 100$, $m = 5$, and $r = 120$. Furthermore,

$$\mathcal{U} = \mathcal{U}_2 := \{u \in \mathbb{R}^{20} :\ \|u\|_2 \leq 1\},$$
$$\mathcal{X} = \mathcal{X}_2 := \left\{x \in \mathbb{R}^{100} :\ \|x\|_2 \leq 1,\ e^\top x \leq 1,\ d^\top x \geq 0\right\},$$

where $e \in \mathbb{R}^{n_x}$ is the vector of all ones, $d \in \mathbb{R}^{n_x}$ is a random vector, and

$$\hat{f}_j(x,u) := {c^j}^\top x + {\alpha^j}^\top u, \quad j \in [m_0].$$

We use static approximation to obtain an upper bound on the optimal value of the instances in this class.

To generate random instances, for each $j \in [m_0]$, we randomly generate A_j, p^j, b^j, α^j, c^j, and d by drawing their (entries) values from a standard normal distribution using a built-in MATLAB function "`randn`".

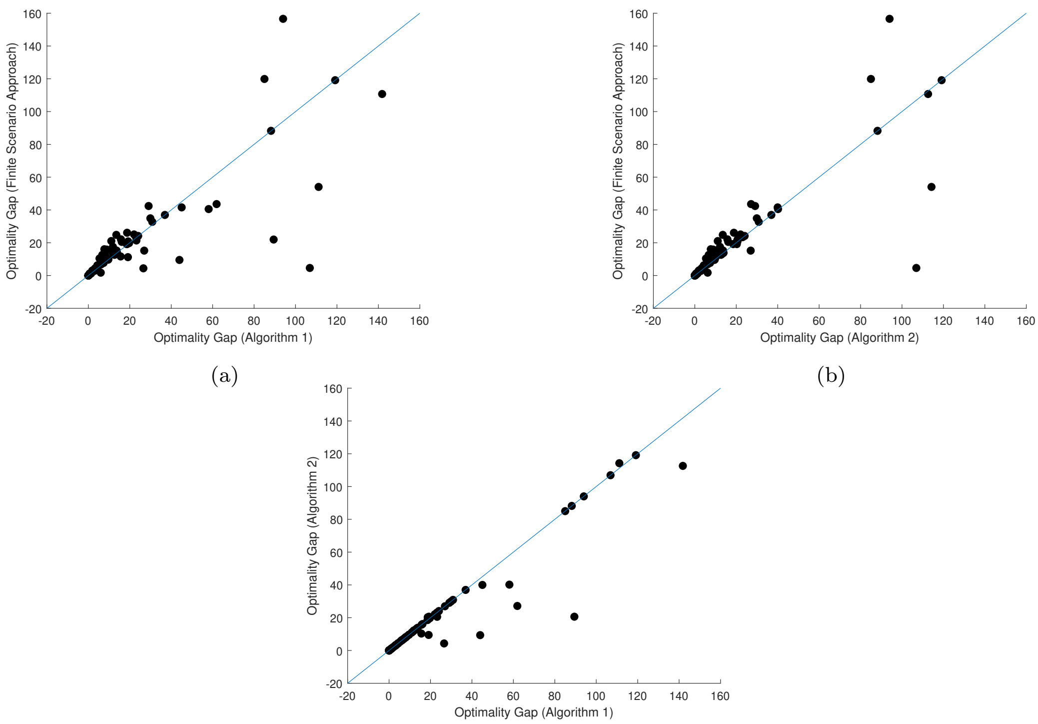
6.2 Numerical results

In this section, we present the results of the numerical experiments.
Class One: We present the statistic on the optimality gaps of the finite-scenario approach (15), Algorithm 1, and Algorithm 2 in Table 1 (details can be found in Table 3 of Appendix 3). Since the upper bound obtained using the K-adaptability approach for the instances of this class is lower than the one derived from the perspectification approach, which is equivalent to the static approximation, we report the gap using the former. As one can see, Algorithm 2 outperforms the other methods on average.

Table 1: Statistic of optimality gaps of instances in Class One.

Method	Mean	Standard deviation
Algorithm 1	26.4925	75.3803
Algorithm 2	**22.2883**	**72.4447**
Finite-scenario approach	49.7797	336.4677

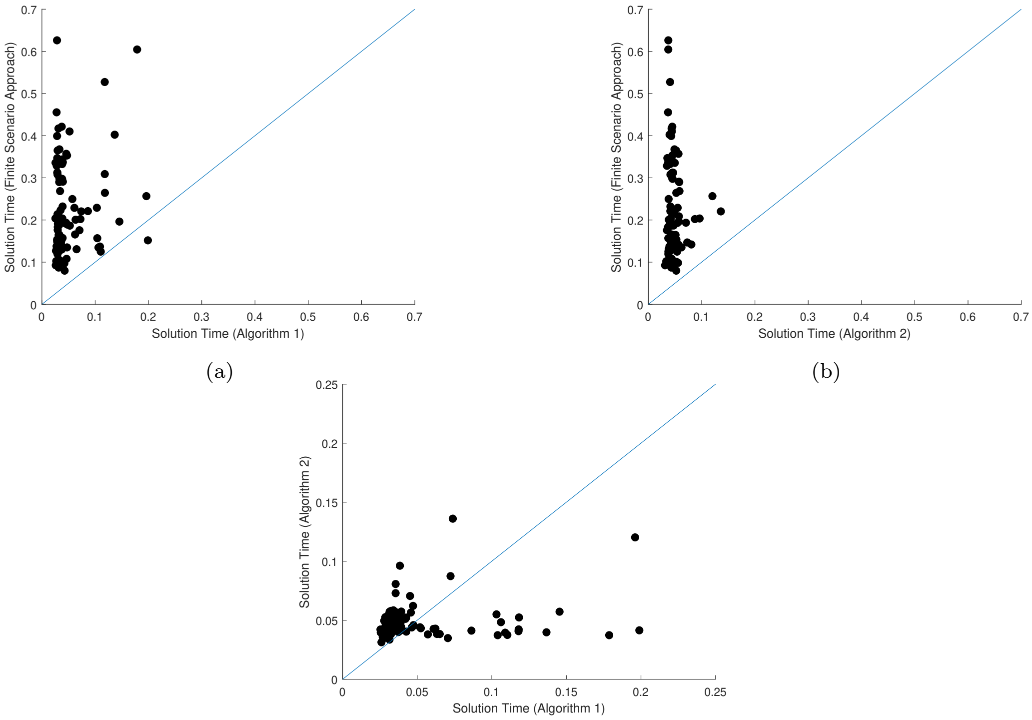
Figure 1 compares the optimality gaps of the solutions obtained by Algorithm 1, Algorithm 2, and the finite-scenario approach, where each point corresponds to an instance. As shown in Figures 1a and 1b, Algorithms 1 and 2 outperform the finite-scenario approach. More specifically, in 84 instances, Algorithm 2 generates better lower bounds, while the finite-scenario approach generates better lower bounds in only 15 instances. We should emphasize that if the scenarios considered in both approaches are the same, Theorem 6 shows that the lower bound obtained by Algorithm 2 should outperform the finite-scenario approach. However, these two methods do not generate the same scenarios. Additionally, Figure 1c shows that the solutions obtained by Algorithm 2 have a similar or better optimality gap to the ones obtained from Algorithm 1. Next to the quality of the approaches, we also report their solution times. The



(a) (b) (c)

Fig. 1: The comparison of the optimality gaps of the solutions obtained by Algorithm 1, Algorithm 2, and finite-scenario approach for the instances in Class One.

average solution times for Algorithm 1, Algorithm 2, and finite-scenario approach are 0.0500, 0.0483, and 0.2253 seconds, respectively. Figure 2 depicts a scatter plot comparing the solution times in each instance. Illustrated in Figures 2a and 2b, Algorithms 1 and 2 reach lower bounds more rapidly compared



(a) (b) (c)

Fig. 2: The comparison of the solution times of Algorithm 1, Algorithm 2, and finite-scenario approach for the instances in Class One.

to the finite-scenario approach. In Figure 2c, we observe that Algorithm 2 reached a solution faster than Algorithm 1.

Hitherto, we have seen that Algorithm 2 performs well in the instances in Class One. In what follows, we analyze the performance of the algorithms for the instances of Class Two.

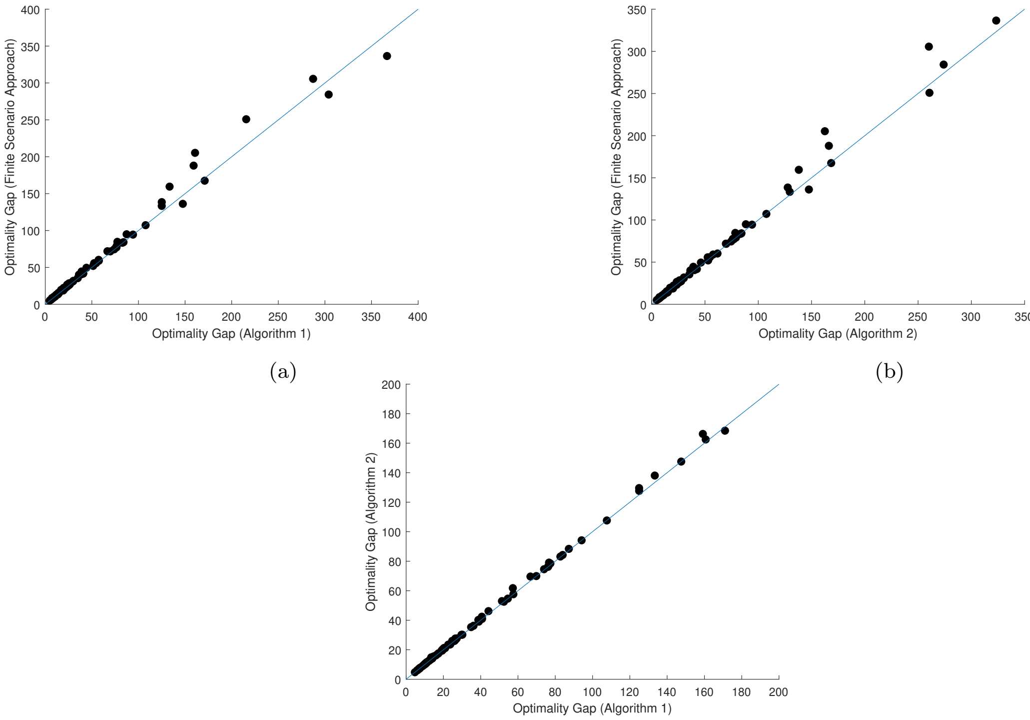
Class Two: We present the statistic on the optimality gaps of the solutions obtained by different algorithms in Table 2 (details can be found in Table 4 of Appendix 3). As one can see, Algorithm 1 outperforms the other methods on average.

Table 2: Statistic of optimality gaps of instances in Class Two.

Method	Mean	Standard deviation
Algorithm 1	**56.5417**	**83.8414**
Algorithm 2	57.7425	88.1209
Finite-scenario approach	64.5980	114.8234

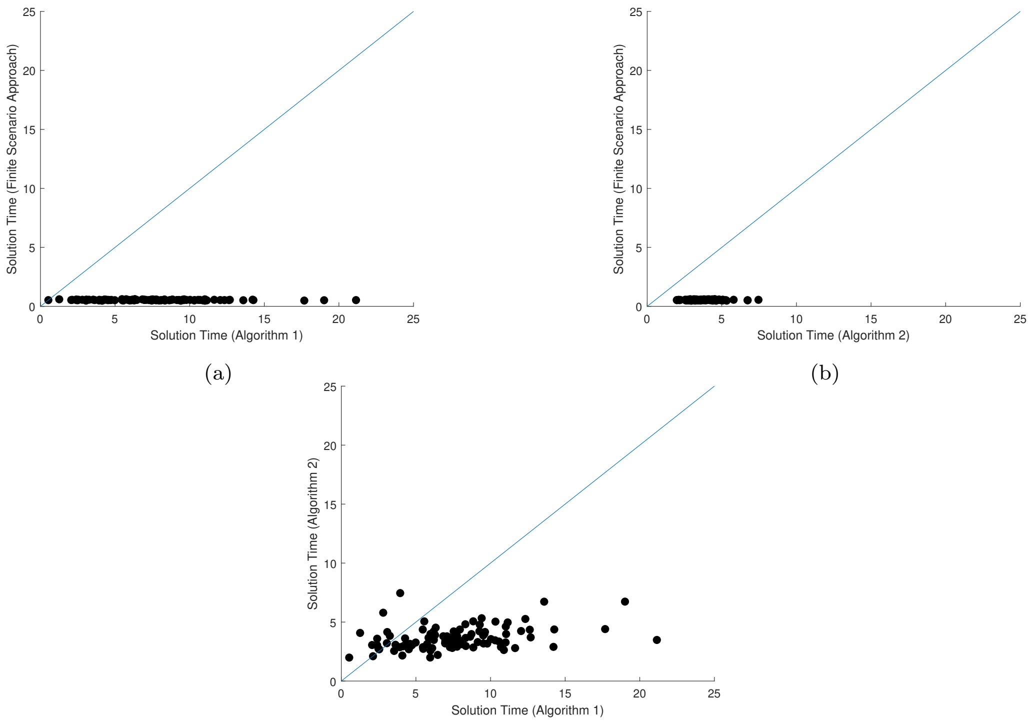
To have a clearer comparison, we illustrate the optimality gaps in Figure 3. Remarkably, both algorithms exhibit better performance in nearly all instances compared to the finite-scenario approach (Figures 3a and 3b). Furthermore, as one can see in Figure 3c, the optimality gap of the solutions obtained by Algorithms 1 and 2 are close, and in all instances (except one of them), Algorithm 1 provides a solution with a slightly lower optimality gap compared to Algorithm 2.

The average solution times for Algorithm 1, Algorithm 2, and finite-scenario approach are 7.5085, 3.6327, and 0.5417 seconds, respectively. Figure 4 presents the scatter plot of the solution times of these approaches on each instance. In this class, the computation times of Algorithms 1 and 2 are higher than the finite-scenario approach because they solve more (sub-)optimization problems than the finite-scenario approach to reach a lower bound (as shown in Figures 4a and 4b). From Figure 4c, across a significant proportion of instances within this classification, it is evident that Algorithm 2 exhibited notable performance in achieving solutions faster than Algorithm 1.



(c)

Fig. 3: The comparison of the optimality gaps of the solutions obtained by Algorithm 1, Algorithm 2, and finite-scenario approach for the instances in Class Two.



(c)

Fig. 4: The comparison of solution times of Algorithm 1, Algorithm 2, and the finite-scenario approach for instances in Class Two.

7 Conclusions

This paper studied a general nonlinear ARO model with objective and constraint uncertainty. We obtained an equivalent dual formulation by applying Fenchel's duality on the wait-and-see variable, a nonlinear static robust optimization. We investigated when the dual formulation is convex in the decision variables. Also, we explored reaching upper and lower bounds for the original problem based on the dual formulation. Thanks to the equivalent dual reformulation, we presented and analyzed two algorithms. These algorithms aimed to find a lower bound on the optimal objective value of the general nonlinear ARO model. We demonstrated by numerical results that our algorithm could produce a locally robust solution with an acceptable optimality gap.

Acknowledgements The first author conducted some parts of this work while he was a visiting researcher at the Eindhoven University of Technology. He wants to express his gratitude for the hospitality of the Department of Industrial Engineering and Innovation Sciences at this institution. The research of the first author was partially supported by INSF (No. 4000183).

Appendices

The appendix of this work is divided into three sections. The first section contains the proof of several points mentioned in the main text. In the second section, we provide a proof for Theorem 5. The final section contains a table related to the numerical experiments.

Appendix 1 Additional Results

In this appendix, first, we provide more details on a point for perspective function mentioned after Remark 1.

Proposition 1 *If g is a proper, closed, and convex function, then*

$$\sup_{t>0,x\in\mathbb{R}^{n_x}} g^{per}(x,t) = \sup_{t\geq 0,x\in\mathbb{R}^{n_x}} g^{per}(x,t),$$

and

$$\inf_{t>0,x\in\mathbb{R}^{n_x}} g^{per}(x,t) = \inf_{t\geq 0,x\in\mathbb{R}^{n_x}} g^{per}(x,t).$$

Proof Let $x^0 \in \mathbb{R}^{n_x}$. We have

$$\begin{aligned} g^{per}(x^0, t_0 = 0) &= \liminf_{(x^i,t_i)\to(x^0,0)} g^{per}(x^i, t_i > 0) \\ &\leq \sup_{(x^i,t_i)\to(x^0,0)} g^{per}(x^i, t_i > 0) \\ &\leq \sup_{t>0,x\in\mathbb{R}^{n_x}} g^{per}(x,t). \end{aligned}$$

So, $\sup_{t>0,x\in\mathbb{R}^{n_x}} g^{per}(x,t) = \sup_{t\geq 0,x\in\mathbb{R}^{n_x}} g^{per}(x,t)$.
As $\inf_{t>0,x} g^{per}(x,t) \geq \inf_{t\geq 0,x} g^{per}(x,t)$, let $\ell \in \{g^{per}(x,t) | t \geq 0, x \in \mathbb{R}^{n_x}\}$. We want to show $\ell \geq \inf_{t>0,x} g^{per}(x,t)$.
1. If $\ell = g^{per}(x^0, t_0)$ for some $x^0 \in \mathbb{R}^{n_x}$ and $t_0 > 0$, then $\ell \geq \inf_{t>0,x} g^{per}(x,t)$.
2. If $\ell = g^{per}(x^0, 0)$ for some $x^0 \in \mathbb{R}^{n_x}$, then

$$\begin{aligned} \ell = g^{per}(x^0, 0) &= \liminf_{(x^i,t_i)\to(x^0,0)} g^{per}(x^i, t_i > 0) \\ &\geq \inf_{(x^i,t_i)\to(x^0,0)} g^{per}(x^i, t_i) \\ &\geq \inf_{t>0,x\in\mathbb{R}^{n_x}} g^{per}(x,t). \end{aligned}$$

The proof is complete. □

As a consequence of the above proposition, we have

$$\sup_{t>0,x} -g^{per}(x,t) = \sup_{t\geq 0,x} -g^{per}(x,t).$$

The next proposition proves the convexity of the set $\mathcal{V}$ and the concavity of the function G introduced in the beginning of Section 5.

Proposition 2 *The set $\mathcal{V}$ is convex, and G is a concave function on $\mathcal{V}$.*

Proof We consider two points $\bar{v} = \begin{pmatrix} \bar{\lambda} \\ \{\bar{w}^j\}_{j=0}^m \end{pmatrix}, \tilde{v} = \begin{pmatrix} \tilde{\lambda} \\ \{\tilde{w}^j\}_{j=0}^m \end{pmatrix} \in \mathcal{V}$ and $\ell \in [0,1]$. Since $\frac{\bar{w}^j}{\bar{\lambda}_j}, \frac{\tilde{w}^j}{\tilde{\lambda}_j} \in dom(g_j^*)$, and $\lambda_j {g_j}^*(\frac{w^j}{\lambda_j})$ for each j is jointly convex in (w^j, λ_j), we have the following possible cases:
Case 1. $\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j > 0$: In this case,

$$\begin{aligned}
&(\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j) g_j^* \left(\frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \right) \leq \ell\bar{\lambda}_j g_j^*(\frac{\bar{w}^j}{\bar{\lambda}_j}) + (1-\ell)\tilde{\lambda}_j g_j^*(\frac{\tilde{w}^j}{\tilde{\lambda}_j}) < \infty \\
&\Rightarrow \frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \in dom(g_j^*)
\end{aligned}$$

Case 2. $\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j = 0$: In this case, if $0 < \ell < 1$, then $\bar{\lambda}_j = 0 = \tilde{\lambda}_j$, and so

$$\begin{aligned}
(\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j) g_j^* \left(\frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \right) &= \delta^*_{dom(g_j)}(\ell\bar{w}^j + (1-\ell)\tilde{w}^j) \\
&\leq \delta^*_{dom(g_j)}(\ell\bar{w}^j) + \delta^*_{dom(g_j)}((1-\ell)\tilde{w}^j) \\
&< \infty.
\end{aligned}$$

If $\ell = 0$, then $\tilde{\lambda}_j = 0$, and hence

$$\begin{aligned}
(\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j) g_j^* \left(\frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \right) &= \delta^*_{dom(g_j)}(\ell\bar{w}^j + (1-\ell)\tilde{w}^j) \\
&= \delta^*_{dom(g_j)}(\tilde{w}^j) < \infty.
\end{aligned}$$

If $\ell = 1$, then $\bar{\lambda}_j = 0$, and thus

$$\begin{aligned}
(\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j) g_j^* \left(\frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \right) &= \delta^*_{dom(g_j)}(\ell\bar{w}^j + (1-\ell)\tilde{w}^j) \\
&= \delta^*_{dom(g_j)}(\bar{w}^j) < \infty
\end{aligned}$$

So, in all above three cases, we get

$$\begin{aligned}
&(\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j) g_j^* \left(\frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \right) = \delta^*_{dom(g_j)}(\ell\bar{w}^j + (1-\ell)\tilde{w}^j) < \infty \\
&\Rightarrow \frac{\ell\bar{w}^j + (1-\ell)\tilde{w}^j}{\ell\bar{\lambda}_j + (1-\ell)\tilde{\lambda}_j} \in dom(g_j^*).
\end{aligned}$$

Convexity in all other constraints of $\mathcal{V}$ obviously holds. So, $\ell\bar{v} + (1-\ell)\tilde{v} \in \mathcal{V}$ which shows that $\mathcal{V}$ is a convex set. The function G on the convex set $\mathcal{V}$ is a concave function due to the concavity of each $-\lambda_j {g_j}^*(\frac{w^j}{\lambda_j})$. □

Appendix 2 Proof of Theorem 5

We first recall optimality condition for a constrained differentiable problem (for more details see e.g., [4]). Consider a (non-convex) problem of the form

$$\sup_{y} \{g(y) |\ y \in \mathcal{S}\}, \tag{21}$$

where g is a real-valued continuously differentiable function, and $\mathcal{S}$ is a nonempty closed convex set. A vector $y^* \in \mathcal{S}$ is called a *stationary point* of problem (21) if

$$\nabla g(y^*)^\top (y - y^*) \leq 0, \quad \forall y \in \mathcal{S},$$

where $\nabla g(y^*)$ is the gradient of g at y^*.

Lemma 1 *Let g be a real-valued continuously differentiable function defined on the Cartesian product of two closed convex sets $C_1 \subseteq \mathbb{R}^{n_1}$, $C_2 \subseteq \mathbb{R}^{n_2}$. Suppose that $\bar{y} = (\bar{y}^1, \bar{y}^2) \in C_1 \times C_2$. Then*

$$\nabla g(\bar{y})^\top (y - \bar{y}) \leq 0, \quad \forall y \in C_1 \times C_2, \tag{22}$$

if and only if the following properties hold:

(i) $\nabla_1 g(\bar{y})^\top (y^1 - \bar{y}^1) \leq 0, \quad \forall y^1 \in C_1,$
(ii) $\nabla_2 g(\bar{y})^\top (y^2 - \bar{y}^2) \leq 0, \quad \forall y^2 \in C_2,$

where the vector y is partitioned into two component vectors $y^1 \in \mathbb{R}^{n_1}$, $y^2 \in \mathbb{R}^{n_2}$, as $y \equiv (y^1, y^2)$, and $\nabla_1 g(\bar{y}) = \left(\frac{\partial g}{\partial y^1}(\bar{y})\right)$, and $\nabla_2 g(\bar{y}) = \left(\frac{\partial g}{\partial y^2}(\bar{y})\right)$ denote the corresponding gradient vectors.

Proof ($\Rightarrow$) Let $y = (y^1, y^2) \in C_1 \times C_2$. By setting $y := (y^1, \bar{y}^2)$ and $y := (\bar{y}^1, y^2)$ in inequality (22), inequalities (i) and (ii) are derived.
($\Leftarrow$) Clearly, (i) and (ii) lead (22). □

Now we are ready to prove Theorem 5. The main line of reasoning can be found in [19] but given here for completeness.

Proof of Theorem 5. Suppose that $z^* = (z^{1*}, z^{2*})$ is a limit point of the sequence $\{z^k\}_{k \geq 0}$. Without loss of generality, we assume that $z^k = (u^k, v^k) \to (z^{1*}, z^{2*})$. Our goal is to show that for any $\zeta = (\zeta^1, \zeta^2) \in \mathcal{U} \times \mathcal{V}$, we have

$$\nabla \mathcal{L}_{\bar{x}}(z^*)^\top (\zeta - z^*) \leq 0.$$

According to Lemma 1, the above inequality is equivalent to

$$\nabla_1 \mathcal{L}_{\bar{x}}(z^*)^\top (\zeta^1 - z^{1*}) \leq 0, \quad \forall \zeta^1 \in \mathcal{U}, \tag{23}$$

$$\nabla_2 \mathcal{L}_{\bar{x}}(z^*)^\top (\zeta^2 - z^{2*}) \leq 0, \quad \forall \zeta^2 \in \mathcal{V}, \tag{24}$$

where $\nabla \mathcal{L}_{\bar{x}}(z^*) = \left(\nabla_1 \mathcal{L}_{\bar{x}}(z^*)^\top, \nabla_2 \mathcal{L}_{\bar{x}}(z^*)^\top\right)^\top$ is the gradient of $\mathcal{L}_{\bar{x}}$ at z^*. By contradiction, suppose that there exists a vector $\tilde{\zeta}^2 \in \mathcal{V}$, such that

$$\nabla_2 \mathcal{L}_{\bar{x}}(z^*)^\top (\tilde{\zeta}^2 - z^{2*}) > 0. \tag{25}$$

Set $r^k := \tilde{\zeta}^2 - v^k$. As the sequence $\{v^k\}_{k \geq 0}$ converges to z^{2*}, the sequence $\{r^k\}_{k \geq 0}$ converges to $\tilde{\zeta}^2 - z^{2*}$. Thus, due to the continuity of the gradient, there exists $N > 0$ such that for all $k > N$ we have

$$\nabla_2 \mathcal{L}_{\bar{x}}(z^k)^\top r^k > 0.$$

So, $d^k := (\mathbf{0}^\top, (r^k)^\top)^\top$ is an ascent direction of $\mathcal{L}_{\bar{x}}$ at z^k. By backtracking line search [4, Lemma 4.3], for given parameter $\alpha \in (0,1)$, there exists a step size $t_k \in (0,1)$ such that

$$\mathcal{L}_{\bar{x}}(z^k + t_k d^k) - \mathcal{L}_{\bar{x}}(z^k) \geq \alpha t_k \nabla \mathcal{L}_{\bar{x}}(z^k)^\top d^k, \quad \forall k > N.$$

Therefore

$$\mathcal{L}_{\bar{x}}(u^k, v^k + t_k r^k) - \mathcal{L}_{\bar{x}}(u^k, v^k) \geq \alpha t_k \nabla_2 \mathcal{L}_{\bar{x}}(z^k)^\top r^k > 0, \quad \forall k > N. \tag{26}$$

Since $\mathcal{V}$ is convex, we have

$$v^k + t_k r^k = (1 - t_k) v^k + t_k \tilde{\zeta}^2 \in \mathcal{V}, \quad \forall k > N.$$

Hence,

$$\mathcal{L}_{\bar{x}}(u^{k+1}, v^{k+1}) \geq \mathcal{L}_{\bar{x}}(u^k, v^{k+1}) \geq \mathcal{L}_{\bar{x}}(u^k, v^k + t_k r^k) > \mathcal{L}_{\bar{x}}(u^k, v^k), \quad \forall k > N.$$

So, the sequence of function values $\left\{\mathcal{L}_{\bar{x}}(u^k, v^k)\right\}$ is non-decreasing and also bounded above. Therefore, it is convergent. The last inequality and the convergence of $\left\{\mathcal{L}_{\bar{x}}(u^k, v^k)\right\}$ implies

$$\lim_{k\to\infty} \mathcal{L}_{\bar{x}}(u^k, v^k + t_k r^k) - \mathcal{L}_{\bar{x}}(u^k, v^k) = 0.$$

The above equation and (26) gives

$$\nabla_2 \mathcal{L}_{\bar{x}}(z^*)^\top (\tilde{\zeta}^2 - z^{2*}) = 0,$$

which contradicts (25). This prove (24). The inequality (23) can be proved similarly. □

Appendix 3 Detailed Results of Numerical Experiments

This appendix contains results from numerical experiments.

Table 3: Detailed numerical results from Class One.

Case	LB_1	LB_2	LB_3	UB_8	UB_1	Case	LB_1	LB_2	LB_3	UB_8	UB_1
1	**259.1977**	**259.1977**	259.1946	260.0564	261.3407	51	**3.3127**	**3.3127**	3.3122	3.3225	3.3274
2	**9.3817**	**9.3817**	9.3658	9.6877	10.0478	52	0.4577	0.5206	**0.5252**	1.1068	3.6220
3	**6.9970**	6.9935	6.7271	7.5588	7.9666	53	**2.9971**	**2.9971**	**2.9971**	2.9971	3.0049
4	**2.7486**	**2.7486**	2.7103	2.9390	3.0800	54	**0.8657**	**0.8657**	0.8653	1.6291	1.9446
5	11.0089	**11.0090**	10.8160	11.4931	11.8715	55	**8.7115**	**8.7115**	8.7114	8.7119	8.7119
6	**15.3520**	**15.3520**	15.2586	16.3023	17.8020	56	0.8995	0.8865	**1.2330**	1.8994	3.1152
7	**-1.9373**	**-1.9373**	**-1.9373**	-1.9357	-1.9357	57	**3.3758**	3.3751	3.3751	3.3934	3.4227
8	**9.3109**	**9.3109**	9.3063	10.2061	11.1458	58	**2.8508**	**2.8508**	**2.8508**	2.8508	3.3457
9	10.6770	10.6762	**11.7674**	13.5560	15.4325	59	**3.4591**	3.4590	3.4419	3.5102	3.5223
10	**5.4643**	**5.4643**	5.3333	6.6699	8.1682	60	**4.7327**	**4.7327**	4.7325	10.3711	19.4164
11	**5.4997**	5.4970	5.4996	5.5553	5.7042	61	**2.1088**	**2.1088**	1.9812	2.2950	2.4424
12	**4.8946**	**4.8946**	4.4919	5.4370	5.6290	62	**2.4899**	**2.4899**	2.4305	2.6931	2.8786
13	-1.7771	**-1.2453**	-1.2531	-0.7450	-0.2057	63	**3.5704**	**3.5704**	3.5606	4.3928	5.2596
14	**15.0970**	**15.0970**	14.8494	15.7447	16.1069	64	**2.4027**	**2.4027**	**2.4027**	2.4027	2.4215
15	**3.1803**	**3.1803**	3.1792	3.1816	3.1816	65	**4.0104**	**4.0104**	3.9697	4.0845	4.1758
16	**32.7964**	**32.7964**	**32.7964**	32.7979	32.8014	66	**3.4362**	**3.4362**	3.3095	4.4646	5.5969
17	**0.1421**	**0.1421**	0.0323	1.1248	2.3101	67	**18.8952**	**18.8952**	17.9105	20.1629	21.3240
18	**16.5824**	**16.5824**	15.0801	18.8211	19.4832	68	**2.8023**	**2.8023**	**2.8023**	2.8025	2.9026
19	**3.6685**	**3.6685**	3.3255	4.7357	5.8387	69	**1.6128**	1.5944	1.5914	1.9228	1.9755
20	7.0912	**8.6052**	8.5969	8.9749	9.3419	70	**3.9882**	**3.9882**	3.9630	4.0116	4.0224
21	**49.5685**	**49.5685**	49.5664	50.9667	52.7088	71	**31.4786**	**31.4786**	31.3828	32.6877	33.7343
22	**7.3621**	**7.3621**	7.3552	8.8104	11.2659	72	**756.1827**	**756.1827**	756.1826	756.2022	756.2175
23	21.5784	**28.4078**	28.3726	31.0675	38.7033	73	**6.4255**	**6.4255**	6.2120	6.9449	7.2463
24	**12.5946**	**12.5946**	12.5854	15.6238	20.4960	74	**2.4011**	2.4010	2.4000	2.4714	2.6531
25	**14.1311**	**14.1311**	14.1301	14.1403	14.1403	75	**2.0766**	**2.0766**	2.0765	2.3492	2.4106
26	**52.2638**	**52.2638**	52.2504	53.0016	53.6820	76	3.0271	3.0215	**3.1531**	3.2083	3.5310
27	**-0.4110**	**-0.4110**	-0.4118	-0.4073	-0.3732	77	**2.2849**	**2.2849**	**2.2849**	2.3563	2.4955
28	**393.0423**	**393.0423**	377.7744	436.0650	452.3473	78	**1.3092**	**1.3092**	**1.3092**	1.3112	1.3126
29	**7.9651**	**7.9651**	7.6756	9.2498	10.8485	79	**7.8062**	**7.8062**	5.9023	15.1441	23.1083
30	**30.2205**	**30.2205**	28.6526	34.9758	38.2203	80	3.3318	**3.4906**	3.4497	3.8514	4.1593
31	**7.2094**	**7.2094**	7.1337	7.3583	7.3583	81	2.6656	**2.8999**	2.8545	3.1747	3.6565
32	**6.6596**	6.6595	6.6530	6.7380	6.9949	82	**7.9201**	**7.9201**	7.9107	7.9570	8.3552
33	**95.2210**	**95.2210**	94.9450	102.2506	110.4956	83	**4.1187**	4.1080	3.9211	4.6071	4.9426
34	**21.5530**	**21.5530**	21.5225	21.7216	21.9525	84	**3.4734**	**3.4734**	3.2243	3.7412	3.9954
35	**4.7314**	**4.7314**	4.6987	4.8901	4.9057	85	3.5410	**3.6188**	3.5939	4.3623	5.8730
36	18.2915	**18.2916**	18.2899	18.5365	19.1278	86	**11.5812**	11.5660	10.9026	13.7476	15.7727
37	5.8375	**16.7701**	16.7202	17.6550	18.3373	87	**10.9663**	**10.9663**	10.9658	10.9887	10.9990
38	5.6850	**7.2390**	6.4109	9.2046	11.3974	88	**14.1618**	**14.1618**	14.1615	14.1639	14.1641
39	**86.3898**	**86.3898**	86.3881	86.5066	86.5195	89	**5.0708**	5.0023	5.0490	6.0166	7.0369
40	**-0.3083**	**-0.3083**	-0.3084	-0.1944	0.0310	90	4.8473	4.8461	**4.8483**	5.5185	5.8588
41	**75.7971**	**75.7971**	75.7851	76.0008	76.6293	91	**9.3160**	**9.3160**	9.2139	9.6205	10.2391
42	**10.9392**	**10.9392**	10.8825	12.9686	18.0775	92	**45.8951**	**45.8951**	45.3398	48.8046	50.3318
43	**32.2960**	**32.2960**	32.2940	32.3780	32.3964	93	1.9520	1.9519	**3.8607**	4.0389	4.4740
44	**-0.6766**	**-0.6766**	**-0.6766**	-0.6766	-0.6185	94	**0.7561**	**0.7561**	0.6361	1.3990	2.2513
45	**1.1652**	**1.1652**	1.1425	1.3013	1.4361	95	-0.7423	**-0.6797**	-0.6980	-0.4079	0.0277
46	**2.5419**	**2.5419**	2.5418	2.8630	3.2250	96	**22.3622**	**22.3622**	22.2758	23.3531	23.9807
47	**1.0582**	**1.0582**	1.0431	1.3847	2.2834	97	3.6372	**5.7130**	5.6500	6.8904	7.9793
48	**3.6334**	**3.6334**	3.5831	4.1290	4.8998	98	**6.6315**	**6.6315**	6.3300	6.9888	7.1655
49	**22.1946**	**22.1946**	22.1167	24.3342	27.4000	99	**18.9985**	**18.9985**	18.9977	19.5478	20.0648
50	**3.3475**	**3.3475**	3.3208	3.4439	3.4843	100	**2.7358**	**2.7358**	2.7216	2.8101	2.8288

Note. The column entitled Case contains the instance number, columns LB_1, LB_2, and LB_3 report the lower bounds obtained by Algorithms 1, 2, and the finite-scenario approach, respectively, and the columns entitled UB_1 and UB_8 report the upper bounds obtained by perspectification approach and 8-adaptability approach, respectively. The accuracy digit is four. In this table, for each instance, the best lower bound is in boldface.

Table 4: Detailed numerical results from Class Two.

Case	LB_1	LB_2	LB_3	UB_1	Case	LB_1	LB_2	LB_3	UB_1
1	**3.9001**	3.8934	3.8854	4.8279	51	**-2.9744**	-2.9971	-3.0021	-1.9376
2	**9.6268**	9.6178	9.6015	10.4601	52	**3.5619**	3.5499	3.5264	4.6196
3	**1.6344**	1.6123	1.6121	2.8872	53	**-1.7460**	-1.7904	-1.8189	-0.3012
4	**10.8829**	10.8773	10.8506	11.6009	54	**3.2074**	3.1680	3.1807	4.5110
5	**12.2019**	12.1472	12.1530	13.5152	55	**-1.0219**	-1.0273	-1.0839	0.0784
6	**2.7891**	2.7762	2.7543	3.8914	56	**-5.9077**	-5.9099	-6.0075	-4.5074
7	**4.2798**	4.2738	4.2123	5.3923	57	**9.0173**	8.9964	9.0082	10.0026
8	**-1.6253**	-1.6291	-1.6727	-0.7387	58	**-5.3944**	-5.4257	-5.4375	-4.3260
9	**3.5420**	3.5370	3.5318	4.2847	59	**5.9016**	**5.9016**	5.8349	6.3315
10	**0.2770**	0.2365	0.1944	1.6372	60	**0.9850**	0.9656	0.8858	2.2989
11	**-1.3185**	-1.3505	-1.3593	-0.3106	61	**7.2530**	7.2352	7.2083	8.2546
12	**-2.0551**	-2.1027	-2.1003	-0.5346	62	0.5393	**0.5800**	0.5150	2.0891
13	**8.1494**	8.1280	8.1285	8.9722	63	**5.2223**	5.1960	5.1005	6.5493
14	**8.3511**	8.3408	8.3322	8.8412	64	**-8.0248**	-8.0617	-8.0568	-6.4983
15	**-1.9698**	-2.0357	-1.9960	-0.9572	65	**7.0972**	7.0572	6.9516	8.5001
16	**7.3853**	7.3711	7.3958	8.3874	66	**-0.4458**	-0.5322	-0.5025	1.1889
17	**16.0063**	15.9723	15.9882	17.1185	67	**-0.5969**	-0.6999	-0.6607	1.2182
18	**2.9578**	2.9267	2.9187	4.1038	68	**-1.1199**	-1.1293	-1.1896	-0.0657
19	**2.4222**	2.4132	2.4054	3.4077	69	**2.2536**	2.2530	2.1678	3.1326
20	**-1.6671**	-1.6730	-1.7801	-0.5029	70	**7.2135**	7.2085	7.1412	8.6105
21	**1.0836**	1.0649	1.0506	1.8063	71	**9.6860**	9.6677	9.6259	10.3959
22	**9.8687**	9.8084	9.8125	11.3247	72	**4.5704**	4.5428	4.5342	5.5110
23	**0.6145**	0.5376	0.5527	1.9393	73	**-3.7698**	-3.7862	-3.7970	-2.8077
24	**-9.4925**	-9.5133	-9.5277	-8.5633	74	**-5.0022**	-5.0044	-5.1230	-3.4930
25	**2.1541**	2.1525	2.0924	2.9331	75	**-1.3936**	-1.4183	-1.4138	-0.2234
26	**9.5053**	9.4669	9.4446	10.8387	76	**5.3090**	5.2698	5.2504	6.7112
27	**-1.2397**	-1.2863	-1.3041	0.8808	77	**-2.0718**	-2.0727	-2.0879	-1.5332
28	**10.8428**	10.8244	10.7536	11.8421	78	**16.0217**	16.0155	15.9792	16.7758
29	**4.6773**	4.6652	4.6285	5.9266	79	**8.1416**	8.1328	8.1075	8.9623
30	**9.9538**	9.9512	9.8376	11.3559	80	**0.7576**	0.7532	0.7278	1.4189
31	**0.6215**	0.6172	0.5307	1.6204	81	**4.5890**	4.5885	4.4751	5.6660
32	**4.5631**	4.5559	4.5147	5.2915	82	**6.3865**	6.3751	6.3194	7.4986
33	**7.0305**	6.9845	6.9944	8.0498	83	**6.1873**	6.1780	6.0546	7.2406
34	**9.9680**	9.8466	9.9100	11.3081	84	**2.9913**	2.9078	2.9352	4.7038
35	**0.8044**	0.7829	0.7239	2.0848	85	**10.6753**	10.6411	10.6368	11.7620
36	**-4.8340**	-4.9150	-4.8925	-3.5569	86	**-7.7191**	-7.7389	-7.7566	-6.2191
37	**4.9906**	4.9789	4.9619	5.9426	87	**11.0850**	11.0523	11.0027	12.4703
38	**-7.6116**	-7.6907	-7.7910	-6.0688	88	10.3312	10.2796	**10.3872**	12.3348
39	**15.8386**	15.8279	15.8049	16.7978	89	**8.9459**	8.9313	8.8954	10.1933
40	**0.3786**	0.3694	0.2224	1.7918	90	**7.6131**	7.6069	7.5995	8.3429
41	**-2.2577**	-2.2625	-2.3401	-0.9587	91	**4.7490**	4.6998	4.6915	5.9225
42	**0.8659**	0.8557	0.8168	1.9483	92	**10.1188**	10.0803	10.0249	11.2766
43	**12.2575**	12.2277	12.2335	13.1937	93	**-1.0054**	-1.0080	-1.0731	-0.2399
44	**4.7063**	4.7037	4.6871	5.1994	94	**-2.3783**	-2.3824	-2.5552	-1.1306
45	**7.1076**	7.0944	7.0682	8.0319	95	**1.3148**	1.2884	1.2675	2.9579
46	**-1.0208**	-1.0213	-1.3427	0.4856	96	**-2.4070**	-2.4142	-2.4486	-1.4270
47	**5.6107**	5.5709	5.5854	6.8776	97	**3.6744**	3.6589	3.6323	4.6726
48	**7.9571**	7.9486	7.9025	9.0604	98	**1.8353**	1.8244	1.7627	3.2556
49	**5.8116**	5.8047	5.7952	6.4533	99	**-5.3745**	-5.4029	-5.4231	-4.3001
50	**2.9529**	2.9124	2.8474	4.2576	100	**7.8266**	7.8138	7.7616	8.7882

Note. The column entitled Case contains the instance number, columns LB_1, LB_2, and LB_3 report the lower bounds obtained by Algorithms 1, 2, and the finite-scenario approach, respectively, and the column UB_1 contains the upper bound obtained by perspectification approach. In this table, for each instance, the best lower bound is in boldface.

References

1. Arslan, A.N., Detienne, B.: Decomposition-based approaches for a class of two-stage robust binary optimization problems. INFORMS Journal on Computing **34**(2), 857–871 (2022)
2. Auslender, A., Teboulle, M.: Asymptotic cones and functions in optimization and variational inequalities. Springer Science & Business Media, New York (2006)
3. Beck, A.: First-order methods in optimization. SIAM, Philadelphia (2017)
4. Beck, A.: Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with Python and MATLAB, Second Edition. SIAM, Philadelphia (2023)
5. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust optimization. Princeton university press, Princeton (2009)
6. Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A.: Adjustable robust solutions of uncertain linear programs. Mathematical programming **99**(2), 351–376 (2004)
7. Ben-Tal, A., Nemirovski, A.: Robust convex optimization. Mathematics of operations research **23**(4), 769–805 (1998)
8. Ben-Tal, A., Nemirovski, A.: Robust solutions of uncertain linear programs. Operations research letters **25**(1), 1–13 (1999)
9. Bertsimas, D., Dunning, I.: Multistage robust mixed-integer optimization with adaptive partitions. Operations Research **64**(4), 980–998 (2016)
10. Bertsimas, D., Goyal, V.: On the power and limitations of affine policies in two-stage adaptive optimization. Mathematical programming **134**(2), 491–531 (2012)
11. Bertsimas, D., Goyal, V., Lu, B.Y.: A tight characterization of the performance of static solutions in two-stage adjustable robust linear optimization. Mathematical Programming **150**(2), 281–319 (2015)
12. Bertsimas, D., den Hertog, D.: Robust and adaptive optimization. Dynamic Ideas LLC, Belmont, MA (2022)
13. Bertsimas, D., Iancu, D.A., Parrilo, P.A.: A hierarchy of near-optimal policies for multistage adaptive optimization. IEEE Transactions on Automatic Control **56**(12), 2809–2824 (2011)
14. Boni, O., Ben-Tal, A.: Adjustable robust counterpart of conic quadratic problems. Mathematical Methods of Operations Research **68**(2), 211–233 (2008)
15. Breuer, D.J., Lahrichi, N., Clark, D.E., Benneyan, J.C.: Robust combined operating room planning and personnel scheduling under uncertainty. Operations research for health care **27**, 100,276 (2020)
16. Combettes, P.L.: Perspective functions: Properties, constructions, and examples. Set-Valued and Variational Analysis **26**(2), 247–264 (2018)
17. Du, B., Zhou, H., Leus, R.: A two-stage robust model for a reliable p-center facility location problem. Applied Mathematical Modelling **77**, 99–114 (2020)
18. El Ghaoui, L., Oustry, F., Lebret, H.: Robust solutions to uncertain semidefinite programs. SIAM Journal on Optimization **9**(1), 33–52 (1998)
19. Grippo, L., Sciandrone, M.: On the convergence of the block nonlinear gauss–seidel method under convex constraints. Operations research letters **26**(3), 127–136 (2000)
20. Hadjiyiannis, M.J., Goulart, P.J., Kuhn, D.: A scenario approach for estimating the suboptimality of linear decision rules in two-stage robust optimization. In: 2011 50th IEEE Conference on Decision and Control and European Control Conference, pp. 7386–7391. IEEE (2011)
21. Hanasusanto, G.A., Kuhn, D., Wiesemann, W.: K-adaptability in two-stage robust binary programming. Operations Research **63**(4), 877–891 (2015)
22. Hashemi Doulabi, H., Jaillet, P., Pesant, G., Rousseau, L.M.: Exploiting the structure of two-stage robust optimization models with exponential scenarios. INFORMS Journal on Computing **33**(1), 143–162 (2021)
23. Hiriart-Urruty, J.B., Lemaréchal, C.: Fundamentals of convex analysis. Springer Science & Business Media, Berlin (2004)
24. Kammammettu, S., Li, Z.: Two-stage robust optimization of water treatment network design and operations under uncertainty. Industrial & Engineering Chemistry Research **59**(3), 1218–1233 (2019)
25. Ke, G.Y.: Managing reliable emergency logistics for hazardous materials: A two-stage robust optimization approach. Computers & Operations Research **138**, 105,557 (2022)
26. Koushki, J., Miettinen, K., Soleimani-damaneh, M.: LR-NIMBUS: an interactive algorithm for uncertain multiobjective optimization with lightly robust efficient solutions. Journal of Global Optimization **83**, 843–863 (2022)
27. Lee, J., Skipper, D., Speakman, E.: Gaining or losing perspective. Journal of Global Optimization **82**(4), 835–862 (2022)
28. Liang, E., Yuan, Z.: Adjustable robust optimal control for industrial 2-mercaptobenzothiazole production processes under uncertainty. Optimization and Engineering pp. 1–38 (2022)
29. Löfberg, J.: YALMIP: A toolbox for modeling and optimization in MATLAB. In: In Proceedings of the CACSD Conference. Taipei, Taiwan (2004)
30. Lu, M., Shen, Z.J.M.: A review of robust operations management under model uncertainty. Production and Operations Management **30**(6), 1927–1943 (2021)
31. Marandi, A., Den Hertog, D.: When are static and adjustable robust optimization problems with constraint-wise uncertainty equivalent? Mathematical programming **170**(2), 555–568 (2018)
32. Marandi, A., van Houtum, G.J.: Robust location-transportation problems with integer-valued demand. Optimization Online (2020)
33. MOSEK ApS: The MOSEK optimization toolbox for MATLAB manual. version 9.3.21 (2022)
34. Postek, K., Hertog, D.d.: Multistage adjustable robust mixed-integer optimization via iterative splitting of the uncertainty set. INFORMS Journal on Computing **28**(3), 553–574 (2016)
35. Rockafellar, R.T.: Convex analysis, vol. 36. Princeton university press, Princeton (1970)
36. Romeijnders, W., Postek, K.: Piecewise constant decision rules via branch-and-bound based scenario detection for integer adjustable robust optimization. INFORMS Journal on Computing **33**(1), 390–400 (2021)
37. Roos, K., Balvert, M., Gorissen, B.L., den Hertog, D.: A universal and structured way to derive dual optimization problem formulations. Informs Journal on Optimization **2**(4), 229–255 (2020)
38. Roy, A., Dabadghao, S., Marandi, A.: Value of intermediate imaging in adaptive robust radiotherapy planning to manage radioresistance. Annals of Operations Research pp. 1–22 (2022)

39. de Ruiter, F.J., Zhen, J., den Hertog, D.: Dual approach for two-stage robust nonlinear optimization. Operations Research (2022)
40. Shapiro, A., Dentcheva, D., Ruszczynski, A.: Lectures on stochastic programming: modeling and theory. SIAM, Philadelphia (2021)
41. Shapiro, A., Nemirovski, A.: On complexity of stochastic programming problems. In: Continuous optimization, pp. 111–146. Springer, New York (2005)
42. Soyster, A.L.: Convex programming with set-inclusive constraints and applications to inexact linear programming. Operations research **21**(5), 1154–1157 (1973)
43. Subramanyam, A., Gounaris, C.E., Wiesemann, W.: K-adaptability in two-stage mixed-integer robust optimization. Mathematical Programming Computation **12**(2), 193–224 (2020)
44. Takeda, A., Taguchi, S., Tütüncü, R.: Adjustable robust optimization models for a nonlinear two-period system. Journal of Optimization Theory and Applications **136**(2), 275–295 (2008)
45. Wei, L., Gómez, A., Küçükyavuz, S.: Ideal formulations for constrained convex optimization problems with indicator variables. Mathematical Programming **192**(1), 57–88 (2022)
46. Woolnough, D., Jeyakumar, V., Li, G.: Exact conic programming reformulations of two-stage adjustable robust linear programs with new quadratic decision rules. Optimization Letters **15**(1), 25–44 (2021)
47. Xidonas, P., Steuer, R., Hassapis, C.: Robust portfolio optimization: A categorized bibliographic review. Annals of Operations Research **292**(1), 533–552 (2020)
48. Xu, G., Burer, S.: A copositive approach for two-stage adjustable robust optimization with uncertain right-hand sides. Computational Optimization and Applications **70**(1), 33–59 (2018)
49. Yanıkoğlu, İ., Gorissen, B.L., den Hertog, D.: A survey of adjustable robust optimization. European Journal of Operational Research **277**(3), 799–813 (2019)
50. Zadeh, N.: Note—a note on the cyclic coordinate ascent method. Management Science **16**(9), 642–644 (1970)
51. Zeng, B., Zhao, L.: Solving two-stage robust optimization problems using a column-and-constraint generation method. Operations Research Letters **41**(5), 457–461 (2013)
52. Zhang, N., Fang, C.: Saddle point approximation approaches for two-stage robust optimization problems. Journal of Global Optimization **78**(4), 651–670 (2020)
53. Zhang, X., Liu, X.: A two-stage robust model for express service network design with surging demand. European Journal of Operational Research **299**(1), 154–167 (2022)
54. Zhen, J., Kuhn, D., Wiesemann, W.: A unified theory of robust and distributionally robust optimization via the primal-worst-equals-dual-best principle. Operations Research (2023)