Distributionally Robust Optimization with Decision-Dependent Information Discovery

Qing Jin[1,2*], Angelos Georghiou[3], Phebe Vayanos[1,2,4], Grani A. Hanasusanto[5]

[1]Department of Industrial and Systems Engineering, University of Southern California.
[2]Center for Artificial Intelligence in Society, University of Southern California.
[3]Department of Business and Public Administration, University of Cyprus.
[4]Department of Computer Science, University of Southern California.
[5]Department of Industrial & Enterprise Systems Engineering, University of Illinois Urbana-Champaign.

*Corresponding author(s). E-mail(s): qingjin@usc.edu;
Contributing authors: georghiou.angelos@ucy.ac.cy;
phebe.vayanos@usc.edu; gah@illinois.edu;

Abstract

We study two-stage distributionally robust optimization (DRO) problems with decision-dependent information discovery (DDID) wherein (a portion of) the uncertain parameters are revealed only if an (often costly) investment is made at the first stage. This class of problems finds many important applications in selection problems (e.g., in hiring, project portfolio optimization, or optimal sensor location). Despite the wide applicability of the problem, it has not been previously studied. We propose a framework for modeling and approximately solving DRO problems with DDID. We formulate the problem as a min-max-min-max problem and adopt the popular K-adaptability approximation scheme, which chooses K candidate recourse actions here-and-now and implements the best of those actions after the uncertain parameters that were chosen to be observed are revealed. We then present a decomposition algorithm that solves the K-adaptable formulation exactly. In particular, we devise a cutting plane algorithm that iteratively solves a relaxed version of the problem, evaluates the true objective value of the corresponding solution, generates valid cuts, and imposes them on the relaxed problem. For the evaluation problem, we develop a branch-and-cut algorithm that provably converges to an optimal solution. We showcase the effectiveness of our framework on the R&D project portfolio optimization problem and the best box problem.

1

1 Introduction

1.1 Background & Motivation

Distributionally robust optimization, which optimizes for a safe action that performs best under the most adverse distribution in an *ambiguity set* of distributions consistent with known information, is a popular modeling and solution paradigm for decision-making under uncertainty. It can be used to tackle both single-stage (see, e.g., Delage and Ye (2010), Ben-Tal et al (2013), Wiesemann et al (2014), Jiang and Guan (2016), Gao and Kleywegt (2023), Mohajerin Esfahani and Kuhn (2018)) and multi-stage (see e.g., Goh and Sim (2010), Hanasusanto and Kuhn (2018), Bertsimas et al (2019), Chen et al (2020), Yu and Shen (2022)) problems. In multi-stage problems, uncertain parameters are revealed sequentially over time, and decisions are allowed to *adapt* to the history of observations. Mathematically, decisions must be modeled as functions of the uncertain parameters that have been revealed by the time the decision is made.

To the best of our knowledge, all models and solution approaches assume that the sequence in which uncertain parameters are revealed is *exogenous*, i.e., *independent* of the decision-maker's actions. However, this assumption fails to hold in many real-world applications where uncertain parameters (*information*) only become observable following an often costly investment and *measurement decisions* control the *time of information discovery*. We now describe some important examples of such problems that involve *decision-dependent information discovery*.

R&D Project Portfolio Optimization. Companies often maintain long pipelines of projects that are candidates to be undertaken (Solak et al, 2010). The return of each project is uncertain and will only be revealed if the project is completed, which

often necessitates substantial resources to be expanded. Thus, the costly decisions to undertake and complete a project are measurement decisions that control the time of information discovery in this problem.

Optimal Sensor Placement for Emergency Warning. Emergency warning systems (that issue warnings about, e.g., landslides (Chu et al, 2021)) rely on information collected from sensors (about, e.g., soil moisture) to make predictions. Sensors are often costly or difficult to install (due to e.g., difficulty in accessing remote locations). The information available to help inform the warning system depends on the location of the sensors. Thus, the costly sensor placement decisions are measurement decisions that control the time of information discovery in this problem.

Selection Problems. Selection problems, such as those arising in company hiring, loan approval, or bail decisions, involve selecting, over time, a subset of available candidates to move to the next stage until a final selection decision is made in the last stage. The personal characteristics (e.g., qualifications, gender, race) of individuals are uncertain and only become observed progressively over time as they get selected into subsequent stages. Accordingly, the outcomes (e.g., job performance, loan repayment, recidivism) are also uncertain and only become observed for individuals who make it into the final stage. Thus, the candidate selection decisions are measurement decisions that control the time of information discovery in this problem.

In most real-world situations, the precise distribution of the uncertain parameters in such problems is unknown (e.g., the joint distribution of project returns in R&D, the joint distribution of soil characteristics at different remote locations, the joint distribution of job candidates' characteristics, and corresponding job performance). Yet, basic characteristics of this joint distribution, such as first-order moments, may be available (e.g., through past project performance, current weather conditions, and past worker performance data). In our work, we propose to leverage such information in *distributionally robust optimization problems with decision-dependent information*

discovery where such moment information is added explicitly as constraints in the ambiguity set.

1.2 Literature Review

Our work most closely relates to the literature on optimization under decision-dependent information discovery, to distributionally robust optimization with decision-dependent uncertainty sets, and to the works on the K-adaptability approximation to (distributionally) robust optimization. We review all of these in turn.

Decision-making problems under decision-dependent information discovery have mostly been studied in the stochastic programming literature. The majority of works assume the distribution of the uncertain parameters is discrete or require the distribution to be discretized before they can be applied, see Colvin and Maravelias (2010); Goel and Grossmann (2004, 2006); Gupta and Grossmann (2011, 2014); Apap and Grossmann (2017). Vayanos et al (2011) investigate a problem with continuous uncertain parameters and propose a conservative solution approach based on piecewise constant and piecewise linear decision rule approximations. In the robust optimization literature, Zhang and Feng (2020) derive a decision rule approximation for multi-stage robust optimization with endogenous uncertainty. Paradiso et al (2022) develop an exact solution scheme based on a nested decomposition algorithm for two-stage robust optimization problems with DDID and objective uncertainty. Omer et al (2024) provide a mixed-integer linear programming formulation for robust combinatorial problems with DDID under polyhedral uncertainty through column generation and constraint generation algorithms.

Distributionally robust optimization with a decision-dependent ambiguity set has been studied in both static and adaptive settings. Such a framework can model the case where the distribution of the underlying random variables depends on decisions. One class of decision-dependent ambiguity sets allows moments of the distributions to

depend on the decisions, see Zhang et al (2016); Luo and Mehrotra (2020); Ryu and Jiang (2019); Basciftci et al (2021), and Yu and Shen (2022). Another class allows decisions to influence the nominal or marginal distributions; see Luo and Mehrotra (2020); Noyan et al (2022), and Doan (2021). None of the models considers the case where decisions affect the time of information discovery.

The K-adaptability approximation scheme, originally introduced by Bertsimas and Caramanis (2010) for two-stage robust optimization, involves selecting K candidate policies here-and-now and implementing the best of these policies after the uncertain parameters are revealed. This method naturally accommodates discrete adaptive variables. Bertsimas and Caramanis (2010) study the complexity of the problem and provide an exact finite-dimensional bilinear formulation for the case where $K = 2$. Subramanyam et al (2020) propose a branch-and-bound algorithm to speed-up computation. For the class of problems involving only binary adaptive variables, Hanasusanto et al (2015) characterize the problem's complexity in terms of the number of second-stage policies K needed to obtain an optimal solution to the original, fully adaptive problem and derive practical explicit mixed-integer linear optimization (MILO) reformulations. Buchheim and Kurtz (2017, 2018) study the complexity of two-stage robust combinatorial optimization problems under objective uncertainty and propose efficient algorithms for the cases of polyhedral and discrete uncertainty sets, respectively. Chassein et al (2019) propose a faster algorithm for problems with a budgeted uncertainty set. Kurtz (2024) derives bounds on the number of policies K which guarantee optimality for non-linear two-stage robust optimization problems with integer decisions, and extends the results to the case with DDID. Hanasusanto et al (2016) adapt the K-adaptability approximation for binary decision variables to the case of two-stage distributionally robust optimization problems, derive explicit mixed-integer linear programming reformulations, and provide efficient methods for bounding the selection

5

probabilities of each of the K second-stage policies. Han et al (2023) provide a single-stage robust optimization reformulation for computing optimal K-adaptable policies with optimal partitions for a two-stage distributionally robust optimization problem with constraint uncertainty, show the reformulated problem is $\mathcal{NP}$-complete, and propose a partitioning framework that results in a mixed-integer optimization problem of moderate size. Our work most closely relates to the paper of Vayanos et al (2020), wherein the authors study a robust optimization problem with DDID and propose a conservative approximation based on K-adaptability. We study the more general class of *distributionally* robust problems with DDID.

1.3 Proposed Approach & Contributions

We now summarize our proposed approach and main contributions in this paper.

1. We study two-stage distributionally robust optimization problems with decision-dependent information discovery. We formulate this class of problems as adaptive distributionally robust optimization problems that optimize over decision rules on which we impose decision-dependent non-anticipativity constraints. We establish that this model admits an equivalent two-and-a-half-stage (min-max-min-max) robust optimization reformulation and adapt the popular K-adaptability approach to obtain a conservative approximation. The parameter K, which denotes the number of candidate policies, enables us to conveniently trade off computational complexity with solution quality.

2. We show that applying the K-adaptability approximation to the reformulated robust optimization problems leads to a bilinear optimization problem involving products of *real-valued* decision variables (see Theorem 3). For moderately sized problems, this can be solved with off-the-shelf solvers. For larger instances, we propose a decomposition algorithm that solves the problem exactly. The scheme iteratively solves the main problem with only the measurement variables and

adds optimality, feasibility, and Benders-like cuts generated by an evaluation problem. We develop a provably convergent branch-and-cut algorithm to evaluate the objective with fixed measurement variables.

3. We conduct numerical experiments on stylized instances of the R&D project portfolio optimization and best box problems that showcase the effectiveness of our approach. We show that the solutions to the K-adaptability approximation can be obtained by either solving the bilinear optimization problem directly or by employing our decomposition algorithm. In the R&D project portfolio optimization problem, the decomposition algorithm outperforms the state-of-the-art solver, finding better solutions in up to 100% more cases. We assess the value of incorporating distributional information by comparing our method to that returned by a purely robust solution. Our results show that the distributionally robust optimization solution improves the robust solution by up to 51% on average in terms of the relative difference in the objective value under the worst-case distribution and by 49% in out-of-sample performance.

1.4 Organization of the Paper and Notations

The remainder of the paper is organized as follows. Section 2 introduces the two-stage distributionally robust optimization problem with the decision-dependent information discovery and derives an equivalent min-max-min-max robust optimization counterpart. Section 3 presents the K-adaptability approximation scheme, and Section 4 describes the decomposition algorithm. Section 5 conducts numerical experiments to test the performance of our approach. All proofs are relegated to the appendix.

Notation. Throughout the paper, vectors (matrices) are denoted by boldface lowercase (uppercase) letters. Uncertainty is modeled by the probability space $\left(\mathbb{R}^k, \mathcal{B}\left(\mathbb{R}^k\right), \mathbb{P}\right)$, which consists of the sample space $\mathbb{R}^k$, the Borel σ-algebra $\mathcal{B}\left(\mathbb{R}^k\right)$ and the probability measure $\mathbb{P}$, whose support we denote by Ξ. We let $\mathcal{L}_{N_\xi}^{N_y}$ be the

7

space of all measurable functions from $\mathbb{R}^{N_\xi}$ to $\mathbb{R}^{N_y}$ that are bounded on compact sets. Given two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ of the same dimension, $\boldsymbol{x} \circ \boldsymbol{y}$ denotes their Hadamard product. Given two sets $\mathcal{X}$ and $\mathcal{Y}$, $\mathcal{X} \xrightarrow{L} \mathcal{Y}$ denotes a linear mapping from $\mathcal{X}$ to $\mathcal{Y}$. We use $\boldsymbol{A}_i$ and $[\boldsymbol{A}]_i$ to denote the ith column and the ith row of matrix $\boldsymbol{A}$, respectively. With a slight abuse of notations, we use the maximum and minimum operators even when the optimum may not be achieved. In such instances, these operators should be interpreted as suprema and infima, respectively.

2 DRO with Decision-Dependent Information Discovery

In this section, we describe the two-stage distributionally robust optimization problems with decision-dependent information discovery. We propose a decision rule-based formulation for this problem and demonstrate that it is equivalent to a two-and-a-half-stage min-max-min-max robust problem.

2.1 Decision Rule Formulation

The DRO-DDID problem minimizes the worst-case expected objective value over all distributions $\mathbb{P}$ of the uncertain parameters $\boldsymbol{\xi} \in \mathbb{R}^{N_\xi}$, supported on Ξ, and belonging to the ambiguity set $\mathcal{P}$. The ambiguity set contains plausible distributions that share certain properties that are known about the true distribution. The first-stage decisions $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^{N_x}$, and $\boldsymbol{w} \in \mathcal{W} \subseteq \{0,1\}^{N_\xi}$ are selected here-and-now, before any of the uncertain parameters $\boldsymbol{\xi} \in \Xi \subseteq \mathbb{R}^{N_\xi}$ are realized. The wait-and-see decisions $\boldsymbol{y}(\boldsymbol{\xi}) \in \mathcal{Y} \subseteq \mathbb{R}^{N_y}$ are dependent on the observed portion of $\boldsymbol{\xi}$. The sets $\mathcal{X}$ and $\mathcal{Y}$ may involve integrality constraints, implying that both the first and second stage decisions may involve a mix of real- and discrete-valued decisions. The binary vector $\boldsymbol{w}$ collects the measurement decisions determining which components of the uncertain

vector $\boldsymbol{\xi}$ to observe. Specifically, ξ_i is revealed between the first and second stages if and only if $w_i = 1$.

The two-stage distributionally robust problem with decision-dependent information discovery can be written mathematically as

$$
\begin{aligned}
\min \quad & \sup_{\mathbb{P} \in \mathcal{P}} \quad \mathbb{E}_{\mathbb{P}} \Big(\boldsymbol{\xi}^\top \boldsymbol{C} \, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \boldsymbol{y}(\boldsymbol{\xi}) \Big) \\
\text{s.\,t.} \quad & \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W}, \ \boldsymbol{y} \in \mathcal{L}_{N_\xi}^{N_y} \\
& \left. \begin{array}{l} \boldsymbol{y}(\boldsymbol{\xi}) \in \mathcal{Y} \\[4pt] \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}(\boldsymbol{\xi}) \leq \boldsymbol{H}\boldsymbol{\xi} \end{array} \right\} \quad \forall \boldsymbol{\xi} \in \Xi \\
& \boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}(\boldsymbol{\xi}') \quad \forall \boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi \ : \ \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}',
\end{aligned} \tag{1}
$$

where $\boldsymbol{C} \in \mathbb{R}^{N_\xi \times N_x}$, $\boldsymbol{D} \in \mathbb{R}^{N_\xi \times N_\xi}$, $\boldsymbol{Q} \in \mathbb{R}^{N_\xi \times N_y}$, and $\boldsymbol{H} \in \mathbb{R}^{L \times N_\xi}$. This problem optimizes over the here-and-now decisions $\boldsymbol{x} \in \mathcal{X}$, $\boldsymbol{w} \in \mathcal{W}$, and over the generic *functional* decision rules $\boldsymbol{y} \in \mathcal{L}_{N_\xi}^{N_y}$, which include any (measurable) function form, such as linear, piecewise linear, etc. Modeling $\boldsymbol{y}$ as (measurable) functions of $\boldsymbol{\xi}$ ensures that these recourse decisions enjoy a lot of modeling flexibility, being able to adapt *fully* to the (portion of) uncertain parameters that have been revealed between the first and second decision-stages. We assume that the left-hand-side of the constraints is linear in $\boldsymbol{\xi}$, specifically, $\boldsymbol{T}(\boldsymbol{\xi}) : \Xi \xrightarrow{L} \mathbb{R}^{L \times N_x}$, $\boldsymbol{V}(\boldsymbol{\xi}) : \Xi \xrightarrow{L} \mathbb{R}^{L \times N_\xi}$, $\boldsymbol{W}(\boldsymbol{\xi}) : \Xi \xrightarrow{L} \mathbb{R}^{L \times N_y}$. We can account for affine dependencies on $\boldsymbol{\xi}$ by introducing an auxiliary uncertain parameter $\xi_{N_\xi+1}$ restricted to equal one.

The last set of constraints, which is a decision-dependent non-anticipativity constraint, enforces that $\boldsymbol{y}(\cdot)$ can only depend on the observed uncertain parameters. Given two realizations $\boldsymbol{\xi}$ and $\boldsymbol{\xi}'$, the equality $\boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}'$ holds if and only if the observed portions of the two realizations are the same. In this case, the corresponding wait-and-see decisions $\boldsymbol{y}(\boldsymbol{\xi})$ and $\boldsymbol{y}(\boldsymbol{\xi}')$ should also be consistent with one

9

another. The uncertainty set Ξ in (1) is a nonempty bounded polyhedron expressible as $\Xi = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{N_\xi} : \boldsymbol{A}\boldsymbol{\xi} \leq \boldsymbol{b} \right\}$ for some matrix $\boldsymbol{A} \in \mathbb{R}^{R \times N_\xi}$ and vector $\boldsymbol{b} \in \mathbb{R}^R$.

The ambiguity set $\mathcal{P}$ takes the form

$$\mathcal{P} = \left\{ \mathbb{P} \in \mathcal{M}_+(\mathbb{R}^{N_\xi}) \ : \ \mathbb{P}(\boldsymbol{\xi} \in \Xi) = 1, \ \ \mathbb{E}_\mathbb{P}(\boldsymbol{g}(\boldsymbol{\xi})) \leq \boldsymbol{c} \right\}, \tag{2}$$

where $\mathcal{M}_+\left(\mathbb{R}^{N_\xi}\right)$ denotes the cone of nonnegative Borel measures supported on $\mathbb{R}^{N_\xi}$. The uncertainty set Ξ is also the support set, being defined as the smallest set that is known to satisfy $\boldsymbol{\xi} \in \Xi$ with probability 1 for some distribution $\mathbb{P} \in \mathcal{P}$. We assume that $\boldsymbol{c} \in \mathbb{R}^{N_g}$ and that $\boldsymbol{g} : \mathbb{R}^{N_\xi} \to \mathbb{R}^{N_g}$ is a convex piecewise linear multifunction defined as

$$g_s(\boldsymbol{\xi}) = \max_{t \in \mathcal{T}} \boldsymbol{g}_{st}^\top \boldsymbol{\xi} \quad \forall s \in \mathcal{S} = \{1, \dots, N_g\}.$$

The vector $\boldsymbol{g}_{st} \in \mathbb{R}^{N_\xi}$ collects the coefficients of the t-th piece of the function $g_s(\boldsymbol{\xi})$. Without loss of generality, the index t of the linear pieces ranges over the same index set $\mathcal{T} = \{1, \dots, T\}$ for each $g_s(\boldsymbol{\xi})$, where $s \in \mathcal{S}$. Finally, we assume that the ambiguity set $\mathcal{P}$ contains a Slater point in the sense that there is a distribution $\mathbb{P} \in \mathcal{P}$ such that $\mathbb{E}_\mathbb{P}\left[g_s(\boldsymbol{\xi})\right] < c_s$ for all $s \in \mathcal{S}$ for which $g_s(\boldsymbol{\xi})$ is nonlinear.

In the DDID problem, since the information is discovered after implementing the measurement decision, we do not have empirical data to build the data-driven ambiguity set. For example, in the landslide prediction and warning problem, we may have information from other sites that could be used to obtain statistics on moments, but if we never installed sensors on the slope, no data was collected. As such, we consider using the moment-based ambiguity set (2), which can be built judiciously using domain knowledge and expert assessments. The ambiguity set (2) has the flexibility to characterize many parameters of the unknown true distribution $\mathbb{P}_0$, e.g., mean,

mean absolute deviation. It can also approximate the nonlinear parameters, e.g., variance and standard deviation. All of these parameters can be calibrated using domain expertise or historical data (e.g., data from related projects in R&D project portfolio optimization can be used to calibrate the mean return and cost of each project as well as the mean deviation). Another benefit of (2) is tractability, as it usually leads to mixed integer linear programming or mixed integer nonlinear programming formulations.

Remark 1. When $\boldsymbol{w} = \boldsymbol{0}$, no uncertain parameters are revealed between the first and second decision stages. In this case, problem (1) reduces to a single-stage DRO problem, where the decision-maker chooses $\boldsymbol{x}$, $\boldsymbol{w}$, $\boldsymbol{y}$ here-and-now to optimize in view of the worst-case distribution. When $\boldsymbol{w} = \mathbf{e}$, the problem (1) reduces to a two-stage adaptive DRO problem. Finally, when the ambiguity set only contains the support information, we recover the robust optimization problem studied in Vayanos et al (2020).

Solving the decision rule-based formulation (1) is computationally challenging since it optimizes over a function space $\mathcal{L}_{N_\xi}^{N_y}$. To the best of our knowledge, no approximation schemes for the adaptive binary variables can be directly applied. The prepartioning approach proposed in Vayanos et al (2011) is tailored to stochastic and pure robust optimization problems and does not generalize to cases where the distribution is unknown. The approaches proposed in Han et al (2023) and Hanasusanto et al (2016) can only solve the special instance of (1) where $\boldsymbol{w} = \mathbf{e}$. The lack of approximation schemes applicable to formulation (1) motivates us to derive an equivalent nested reformulation that gives us access to a broader set of approximation and solution schemes.

2.2 An Equivalent min-max-min-max Reformulation

In this section, we derive an equivalent min-max-min-max reformulation of problem (1) that yields the same set of optimal first-stage decisions $(\boldsymbol{x}, \boldsymbol{w})$ and corresponding optimal objective values. We formalize the equivalence in the following theorem.

Theorem 1. *The objective value and set of optimal solutions $(\boldsymbol{x}, \boldsymbol{w})$ to problem (1) are equal to those of the min-max-min-max problem*

$$
\min_{\overline{\boldsymbol{\xi}} \in \Xi} \max \left\{ \begin{array}{l} \displaystyle\min_{\boldsymbol{y} \in \mathcal{Y}} \ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})} \ \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y} - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \text{s.t.} \quad \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y} \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}}) \end{array} \right\} \tag{4}
$$
$$
\text{s.t.} \quad \boldsymbol{\psi} \in \mathbb{R}_+^{N_g},\ \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{w} \in \mathcal{W},
$$

where $\Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}}) := \{\boldsymbol{\xi} \in \Xi : \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}}\}$.

Problem (4) can be interpreted as a sequential game against "nature". In the first stage, the decision-maker first decides on $\boldsymbol{\psi}$, $\boldsymbol{x}$, and $\boldsymbol{w}$. Then, nature chooses a realization $\overline{\boldsymbol{\xi}}$ of the uncertain parameters, of which the decision-maker can only observe those elements ξ_i such that $w_i = 1$. In the second stage, the decision-maker selects $\boldsymbol{y}$ in a way that is robust to the remaining elements of ξ_i (that have not been observed). Finally, nature selects a realization of $\boldsymbol{\xi}$ that coincides with $\overline{\boldsymbol{\xi}}$ for those elements that have been observed between the first and second decision stages, i.e., such that $\xi_i = \overline{\xi}_i$ for $i \in \mathcal{I}$ such that $w_i = 1$.

The sketch of the proof of Theorem 1 is as follows. For fixed $\boldsymbol{w}$, $\boldsymbol{x}$, $\boldsymbol{y}(\cdot)$ we identify the inner maximization of problem (1) as a moment problem and dualize it. Strong duality guarantees that the optimal solution $(\boldsymbol{x},\ \boldsymbol{w},\ \boldsymbol{y}(\cdot))$ in the dualized problem is also optimal in the original problem (1) and that both problems have the same objective value. The dual problem then reduces to a robust optimization problem with decision-dependent information discovery. Applying Vayanos et al (2020) Theorem 1 yields the equivalent nested formulation (4). The full proof is deferred to Appendix A.

3 K-Adaptability Counterpart

The min-max-min-max reformulation (4) enables us to propose a new approximate and potentially exact solution approach to solve the DRO problem with DDID. The new solution approach is based on the K-adaptability approximation, which allows us to control the trade-off between complexity and solution quality. In the K-adaptability approximation, K different policies are chosen here-and-now before any uncertain parameters are revealed. After a portion of uncertain parameters $\{\xi_i : w_i = 1\}$ are observed, the best candidate policy is implemented among those that are robustly feasible considering the uncertain parameters that remain unknown. If there exists some $\boldsymbol{\xi} \in \Xi$ such that no policies are feasible, the K-adaptability problem evaluates to $+\infty$. While in problem (4), the wait-and-see decisions $\boldsymbol{y}$ can take on any value that satisfies the constraint in $\mathcal{Y}$, the K-adaptability approximation only allows us to choose one of the K candidate values identified in the first stage. It thus results in a conservative approximation to problem (4).

The K-adaptability counterpart of problem (4) can be written as

$$
\begin{aligned}
\min_{\overline{\boldsymbol{\xi}} \in \Xi} \ \max_{\overline{\boldsymbol{\xi}} \in \Xi} & \left\{ \begin{array}{l} \min_{k \in \mathcal{K}} \ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})} \ \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \text{s.\,t.} \quad \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}}) \end{array} \right\} \\
\text{s.\,t.} \quad & \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W}, \ \boldsymbol{y}^k \in \mathcal{Y}, \ \forall k \in \mathcal{K},
\end{aligned}
\tag{5}
$$

where $\mathcal{K} = \{1, \ldots, K\}$.

Remark 2. The K-adaptability counterpart (5) has the following equivalent decision rule formulation

$$
\begin{aligned}
\min \quad & \sup_{\mathbb{P} \in \mathcal{P}} \ \mathbb{E}_{\mathbb{P}}\Big(\boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}(\boldsymbol{\xi})\Big) \\
\mathrm{s.\,t.} \quad & \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{w} \in \mathcal{W},\ \boldsymbol{y} \in \mathcal{L}_{N_\xi}^{N_y}, \boldsymbol{y}^k \in \mathcal{Y}, \forall k \in \mathcal{K} \\
& \left.\begin{array}{l} \boldsymbol{y}(\boldsymbol{\xi}) \in \{\boldsymbol{y}^1,\ \ldots,\boldsymbol{y}^K\} \\[4pt] \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}(\boldsymbol{\xi}) \le \boldsymbol{H}\boldsymbol{\xi} \end{array}\right\} \quad \forall \boldsymbol{\xi} \in \Xi \\
& \boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}(\boldsymbol{\xi}') \quad \forall \boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi \ :\ \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}'.
\end{aligned} \tag{6}
$$

We provide reformulations of the K-adaptability problem (5) that can be fed into off-the-shelf solvers, beginning with the case involving only objective uncertainty. In the absence of uncertainty in the constraints, the problem is expressible as

$$
\begin{aligned}
\min \quad & \max_{\bar{\boldsymbol{\xi}} \in \Xi} \ \min_{k \in \mathcal{K}} \left\{ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}})} \ \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \right\} \\
\mathrm{s.\,t.} \quad & \boldsymbol{\psi} \in \mathbb{R}_+^{N_g},\ \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{w} \in \mathcal{W},\ \boldsymbol{y}^k \in \mathcal{Y},\ \forall k \in \mathcal{K} \\
& \boldsymbol{T}\boldsymbol{x} + \boldsymbol{V}\boldsymbol{w} + \boldsymbol{W}\boldsymbol{y}^k \le \boldsymbol{h},\ \forall k \in \mathcal{K},
\end{aligned} \tag{7}
$$

where $\boldsymbol{h} \in \mathbb{R}^L$. We show the equivalent mixed binary bilinear reformulation of problem (7) in the following theorem.

Theorem 2. *The K-adaptability problem with objective uncertainty (7) is equivalent to the following mixed binary bilinear problem*

$$
\begin{aligned}
\min \quad & \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{b}^\top \boldsymbol{\beta} + \sum_{k \in \mathcal{K}} \boldsymbol{b}^\top \boldsymbol{\beta}^k \\
\text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W}, \ \boldsymbol{y}^k \in \mathcal{Y}, \ \forall k \in \mathcal{K} \\
& \boldsymbol{\alpha} \in \mathbb{R}_+^K, \ \boldsymbol{\beta} \in \mathbb{R}_+^R, \ \boldsymbol{\beta}^k \in \mathbb{R}_+^R, \ \forall k \in \mathcal{K}, \ \boldsymbol{\gamma}^k \in \mathbb{R}^{N_\xi}, \ \forall k \in \mathcal{K} \\
& \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \delta_{s,t}^k \in \mathbb{R}_+, \ \forall s \in \mathcal{S}, \ \forall t \in \mathcal{T}, \ \forall k \in \mathcal{K} \\
& \boldsymbol{A}^\top \boldsymbol{\beta}^k + \boldsymbol{w} \circ \boldsymbol{\gamma}^k + \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}} \delta_{s,t}^k \boldsymbol{g}_{s,t} = \alpha^k \left(\boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{w} + \boldsymbol{Q}\boldsymbol{y}^k \right) \quad \forall k \in \mathcal{K} \\
& \sum_{t \in \mathcal{T}} \boldsymbol{\delta}_t^k = \alpha^k \boldsymbol{\psi} \quad \forall k \in \mathcal{K} \\
& \mathbf{e}^\top \boldsymbol{\alpha} = 1 \\
& \boldsymbol{A}^\top \boldsymbol{\beta} = \sum_{k \in \mathcal{K}} \boldsymbol{w} \circ \boldsymbol{\gamma}^k \\
& \boldsymbol{T}\boldsymbol{x} + \boldsymbol{V}\boldsymbol{w} + \boldsymbol{W}\boldsymbol{y}^k \le \boldsymbol{h} \quad \forall k \in \mathcal{K}.
\end{aligned}
\tag{8}
$$

For the case of constraint uncertainty, to keep the paper concise, we only present the main theorem and defer the full mixed integer nonlinear optimization (MINLO) reformulation to Appendix A. The dual variables $\boldsymbol{\alpha}$ are associated with the set of epigraph constraints in problem (A.7). The variables $\boldsymbol{\beta}$ and $\{\boldsymbol{\beta}^k\}_{k \in \mathcal{K}}$ are associated with the constraints $\boldsymbol{A}\bar{\boldsymbol{\xi}} \le \boldsymbol{b}$ and $\boldsymbol{A}\boldsymbol{\xi}^k \le \boldsymbol{b}$, $\forall k \in \mathcal{K}$, respectively. The dual variables $\{\boldsymbol{\gamma}^k\}_{k \in \mathcal{K}}$ are associated with the constraint $\boldsymbol{w} \circ \boldsymbol{\xi}^k = \boldsymbol{w} \circ \bar{\boldsymbol{\xi}}$, $\forall k \in \mathcal{K}$. Finally, $\{\delta_{s,t}^k\}_{s \in \mathcal{S}, t \in \mathcal{T}, k \in \mathcal{K}}$ arises from dualizing the set of constraints $\zeta_s^k \le -\boldsymbol{g}_{s,t}^\top \boldsymbol{\xi}^k$ with respect to the moment information.

Theorem 3. *The K-adaptability problem (5) with constraint uncertainty has an equivalent mixed binary bilinear reformulation.*

In the case with only objective uncertainty, formulation (8) presents two sets of bilinear terms. The first set of constraints contain $\alpha^k \boldsymbol{x}$, $\alpha^k \boldsymbol{w}$, and $\alpha^k \boldsymbol{y}^k$, which can be linearized if $\boldsymbol{x}$ and $\boldsymbol{y}^k$ are discrete variables. The second set of constraints involves

the term $\alpha^k \boldsymbol{\psi}$ which is a product between continuous variables (recall that $\boldsymbol{\psi}$ is the dual variable of the constraint $\mathbb{E}_{\mathbb{P}}(\boldsymbol{g}(\boldsymbol{\xi})) \leq \boldsymbol{c}$ in the ambiguity set (2)). The number of nonlinear terms scales as $O(K \cdot N_g)$. In cases where problem (5) involves constraint uncertainty, its MINLO reformulation (A.18) also includes bilinear terms between continuous variables, scaling as $O((1+L)^K \cdot K \cdot N_g)$. These terms do not appear if the ambiguity set (2) only has support information.

Solving a mixed-integer bilinear problem to global optimality is difficult in general. Since version 9.0, Gurobi has added support for solving nonconvex MINLO problems using a spatial branching algorithm. The generic algorithm in Gurobi converges slowly when the number of bilinear terms is large (see Table 3). Instead of solving the monolithic MINLO problems (8) and (A.18) in the case of objective uncertainty and constraint uncertainty, respectively, in the following section, we propose a decomposition algorithm that solves problem (5) using a nested decomposition algorithm.

4 Decomposition Algorithm

In this section, we leverage the nested structure of problem (5) to design a two-layer decomposition algorithm that solves it to global optimality. A decomposition algorithm has been used to solve a two-stage robust optimization problem with DDID and objective uncertainty in Paradiso et al (2022). We extend it to the DRO setting with both objective and constraint uncertainty. Figure 1 provides an overview of our overall proposed solution procedure. We present an L-shaped algorithm designed to optimize the measurement variables $\boldsymbol{w}$, as introduced in Section 4.1. A key subroutine within the L-shaped algorithm involves the evaluation of the true objective value of (5) for a given $\boldsymbol{w}$. We propose to address this evaluation problem through a branch-and-cut algorithm, the details of which are discussed in Section 4.2. In addition, we

develop three cut families to strengthen the main problem in the L-shaped algorithm. We discuss them in Section 4.3.
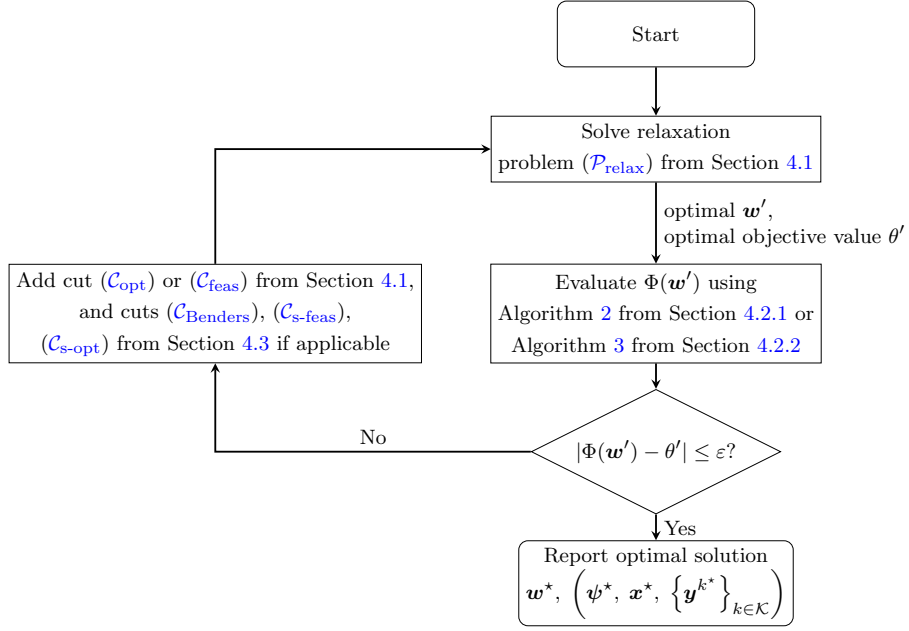
Fig. 1 Overview of our proposed decomposition Algorithm 1

4.1 L-shaped Algorithm to Optimize w

The objective function of (5) can be written as a function of $\boldsymbol{w}$. We present in this section an algorithm to solve the decomposed problem

$$
\begin{aligned}
\min \quad & \Phi(\boldsymbol{w}) \\
\text{s.t.} \quad & \boldsymbol{w} \in \mathcal{W},
\end{aligned}
\tag{$\mathcal{P}$}
$$

17

where

$$\Phi(\boldsymbol{w}) = \min \ \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\overline{\boldsymbol{\xi}} \in \Xi} \left\{ \begin{array}{l} \min_{k \in \mathcal{K}} \ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})} \ \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \text{s.\,t.} \ \ \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \ \ \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}}) \end{array} \right\}$$

$$\text{s.\,t.} \quad \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{y}^k \in \mathcal{Y}, \ \forall k \in \mathcal{K}.$$

$$(\mathcal{IP}_{\boldsymbol{w}})$$

The function $\Phi(\cdot)$ is, in general, nonconvex on $[0,1]^{N_\xi}$. However, since $\boldsymbol{w}$ is binary, Laporte and Louveaux (1993, Proposition 1) show that $\Phi(\cdot)$ can be replaced by its convex lower envelope, a convex piecewise linear function on $[0,1]^{N_\xi}$, that coincides with $\Phi(\cdot)$ at all binary points in $\{0,1\}^{N_\xi}$. We construct the convex lower envelope using two sets of constraints.

The first set of constraints contains the integer optimality inequalities first proposed by Laporte and Louveaux (1993) for a stochastic problem with integer first-stage variables, defined as

$$\Phi(\boldsymbol{w}) \geq (\Phi(\boldsymbol{w}^r) - L) \left(\sum_{i \in \mathcal{I}_r} w_i - \sum_{i \notin \mathcal{I}_r} w_i \right) - (\Phi(\boldsymbol{w}^r) - L)(|\mathcal{I}_r| - 1) + L \quad \forall \boldsymbol{w}^r \in \mathcal{W},$$

$$(\mathcal{C}_{\text{opt}})$$

where L is any lower bound on $\Phi(\boldsymbol{w})$. $\mathcal{I}_r$ collects the indices of the components of vector $\boldsymbol{w}^r$ that take a value of 1. We can verify that the integer optimality cuts are both valid and tight at each binary point $\boldsymbol{x}$. When $\boldsymbol{w} = \boldsymbol{w}^r$, the term $\sum_{i \in \mathcal{I}_r} w_i - \sum_{i \notin \mathcal{I}_r} w_i$ is equal to $|\mathcal{I}_r|$, and the right-hand side of the inequality ($\mathcal{C}_{\text{opt}}$) is equal to $\Phi(\boldsymbol{w}^r)$. In the case where $\boldsymbol{w} \neq \boldsymbol{w}^r$, the value of the term $\sum_{i \in \mathcal{I}_r} w_i - \sum_{i \notin \mathcal{I}_r} w_i$ is less than $|\mathcal{I}_r|$, thereby ensuring that the right-hand side of the inequality ($\mathcal{C}_{\text{opt}}$) does not exceed L. The lower bound L can be determined by the problem structure or obtained heuristically, for example, by solving a deterministic problem which replaces the uncertainty sets in both the inner and outer maximization problems with a singleton $\{\overline{\boldsymbol{\xi}}\}$, where $\overline{\boldsymbol{\xi}} \in \Xi$.

For points $\boldsymbol{w}^r$ such that $\Phi(\boldsymbol{w}^r) = +\infty$, we can either set $\Phi(\boldsymbol{w}^r)$ as a sufficiently large number in inequalities $(\mathcal{C}_{\mathrm{opt}})$, or exclude them from the feasible region $\mathcal{W}$ by enforcing the following set of feasibility cuts (see Balas and Jeroslow (1972) and Nannicini and Belotti (2012))

$$\sum_{i \in \mathcal{I}_r} (1 - w_i) + \sum_{i \notin \mathcal{I}_r} w_i \geq 1, \quad \forall \boldsymbol{w}^r \in \mathcal{W}_{\mathrm{inf}}, \qquad (\mathcal{C}_{\mathrm{feas}})$$

where $\mathcal{W}_{\mathrm{inf}} = \{\boldsymbol{w} : \boldsymbol{w} \in \mathcal{W}, \ \Phi(\boldsymbol{w}) = +\infty\}$. The cuts state that any feasible points $\boldsymbol{w}$ should be different on at least one component from all infeasible points $\boldsymbol{w}^r \in \mathcal{W}_{\mathrm{inf}}$. Expressing the feasibility cut separately in $(\mathcal{C}_{\mathrm{feas}})$, we can then exclude them from $(\mathcal{C}_{\mathrm{opt}})$ and define the set of optimality inequalities on the set $\mathcal{W}_{\mathrm{feas}} = \mathcal{W} \setminus \mathcal{W}_{\mathrm{inf}}$.

For any $\widehat{\mathcal{W}}_{\mathrm{feas}} \subseteq \mathcal{W}_{\mathrm{feas}}$ and $\widehat{\mathcal{W}}_{\mathrm{inf}} \subseteq \mathcal{W}_{\mathrm{inf}}$, the following problem $(\mathcal{P}_{\mathrm{relax}})$ is a relaxation of problem $(\mathcal{P})$ since its constraints enforce a subset of the feasibility cuts $(\mathcal{C}_{\mathrm{feas}})$, and the objective takes an epigraph form on a subset of the optimality cuts $(\mathcal{C}_{\mathrm{opt}})$.

$$\begin{aligned}
\min \quad & \theta \\
\mathrm{s.\,t.} \quad & \theta \in \mathbb{R}, \ \boldsymbol{w} \in \mathcal{W} \\
& \theta \geq (\Phi(\boldsymbol{w}^r) - L) \left(\sum_{i \in \mathcal{I}_r} w_i - \sum_{i \notin \mathcal{I}_r} w_i \right) - (\Phi(\boldsymbol{w}^r) - L)(|\mathcal{I}_r| - 1) + L \qquad (\mathcal{P}_{\mathrm{relax}}) \\
& \hspace{8cm} \forall \boldsymbol{w}^r \in \widehat{\mathcal{W}}_{\mathrm{feas}} \\
& \sum_{i \in \mathcal{I}_r} (1 - w_i) + \sum_{i \notin \mathcal{I}_r} w_i \geq 1 \quad \forall \boldsymbol{w}^r \in \widehat{\mathcal{W}}_{\mathrm{inf}}.
\end{aligned}$$

We use the relaxed problem $(\mathcal{P}_{\mathrm{relax}})$ as the main problem of the L-shaped algorithm. Algorithm 1 describes the L-shaped procedure. It iteratively solves problem $(\mathcal{P}_{\mathrm{relax}})$, evaluates the value $\Phi(\boldsymbol{w}')$ at the current optimal solution $\boldsymbol{w}'$, adds new cuts, and updates the bounds. The proof of finite convergence in Laporte and Louveaux (1993, Proposition 1) for a two-stage integer stochastic problem can be easily adapted. To keep the paper concise, we omit it here.

Algorithm 1: L-shaped algorithm for solving problem $(\mathcal{P})$

Inputs: Any lower bound L of θ, optimality gap tolerance ε.

Output: Optimal solution $\boldsymbol{w}^\star$, and optimal objective value $\theta^\star$.

Initialization: Set $\theta^\star \leftarrow +\infty$, $\boldsymbol{w}^* \leftarrow \emptyset$, $\widehat{\mathcal{W}}_{\text{feas}} \leftarrow \emptyset$, $\widehat{\mathcal{W}}_{\text{inf}} \leftarrow \emptyset$.

1. Solve problem $(\mathcal{P}_{\text{relax}})$, let $(\boldsymbol{w}', \theta')$ be an optimal solution. If $|\theta^\star - \theta'| \leq \varepsilon$, stop.

2. Evaluate $\Phi(\boldsymbol{w}')$ by solving problem $(\mathcal{IP}_{\boldsymbol{w}})$.

3. If $\Phi(\boldsymbol{w}') = +\infty$, set $\widehat{\mathcal{W}}_{\text{inf}} \leftarrow \widehat{\mathcal{W}}_{\text{inf}} \bigcup \{\boldsymbol{w}'\}$, go to step 1; else, go to step 4.

4. If $\Phi(\boldsymbol{w}') < \theta^\star$, set $\boldsymbol{w}^\star \leftarrow \boldsymbol{w}'$ and $\theta^\star \leftarrow \Phi(\boldsymbol{w}')$. Set $\widehat{\mathcal{W}}_{\text{feas}} \leftarrow \widehat{\mathcal{W}}_{\text{feas}} \bigcup \{\boldsymbol{w}'\}$, go to step 1.

Result: Return $\boldsymbol{w}^\star$ and $\theta^\star$.

4.2 Branch-and-Cut Algorithm to Evaluate $\Phi(\boldsymbol{w})$

We present in this section a branch-and-cut algorithm for evaluating the objective function $\Phi(\boldsymbol{w}')$ at the current solution $\boldsymbol{w}'$, which is a crucial component of Algorithm 1. We begin the section by discussing a basic branch-and-bound algorithm adapted from Subramanyam et al (2020) to find an optimal solution for a min-max-min problem with K-adaptability. Then, we design a cutting plane algorithm to solve the inner min-max problem and merge it into the branch-and-bound tree to form the final branch-and-cut algorithm. Finally, we prove the convergence of the branch-and-cut algorithm.

Before proceeding with the algorithm, we introduce an equivalent formulation of problem $(\mathcal{IP}_{\boldsymbol{w}})$ as given in the following proposition.

Proposition 4. *The following problem is equivalent to problem* ($\mathcal{IP}_{\boldsymbol{w}}$).

$$
\begin{aligned}
\min \quad & \theta \\
\text{s.t.} \quad & \theta \in \mathbb{R},\ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g},\ \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{y}^k \in \mathcal{Y}\ \ \forall k \in \mathcal{K},\ \Xi^k \subseteq \mathbb{R}^{N_\xi}\ \ \forall k \in \mathcal{K} \\
& \bigcup_{k \in \mathcal{K}} \Xi^k = \Xi \\
& \left. \begin{aligned} \theta \geq \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}',\bar{\boldsymbol{\xi}})} \quad & \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w}' + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}^k \\ & \qquad\qquad\qquad - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w}' + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad & \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}',\bar{\boldsymbol{\xi}}) \end{aligned} \right\} \begin{aligned} & \forall \bar{\boldsymbol{\xi}} \in \Xi^k, \\ & \forall k \in \mathcal{K}, \end{aligned} \qquad (\mathcal{IP}_{\boldsymbol{w}}(\Xi))
\end{aligned}
$$

where, for each $k \in \mathcal{K}$, Ξ^k *denotes the set of scenarios for which the corresponding optimal policy is* $\boldsymbol{y}^k$.

In problem ($\mathcal{IP}_{\boldsymbol{w}}$), the best candidate policy $\boldsymbol{y}^k$ is selected for each realization $\bar{\boldsymbol{\xi}}$. Problem ($\mathcal{IP}_{\boldsymbol{w}}(\Xi)$) achieves the same goal by optimizing the decision variables Ξ^k, $k \in \mathcal{K}$. This is equivalent to finding an optimal partition of the uncertainty set Ξ. Problem ($\mathcal{IP}_{\boldsymbol{w}}(\Xi)$) is a robust optimization problem with a decision-dependent uncertainty set, since the uncertainty sets $\{\Xi^k\}_{k \in \mathcal{K}}$ in the constraints are decision variables themselves.

4.2.1 Constraint Generation Algorithm

To solve problem ($\mathcal{IP}_{\boldsymbol{w}}(\Xi)$), we use a constraint generation algorithm. The main problem $\mathcal{IP}_{\boldsymbol{w}}(\widehat{\Xi})$ is a relaxation of ($\mathcal{IP}_{\boldsymbol{w}}(\Xi)$) where the set Ξ in the partition constraint is substituted with a finite set $\widehat{\Xi} \subseteq \Xi$. In each iteration, a scenario is generated and added to $\widehat{\Xi}$.

We now describe our proposed procedure for solving the main problem. To this end, we present the following fixed partition problem, where $\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}$ are parameter

inputs rather than decision variables.

$$\min \ \theta$$

$$\text{s.t. } \theta \in \mathbb{R}, \ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{y}^k \in \mathcal{Y} \quad \forall k \in \mathcal{K}$$

$$\left. \begin{array}{l} \theta \geq \boldsymbol{c}^\top \boldsymbol{\psi} + \displaystyle\max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}', \bar{\boldsymbol{\xi}})} \quad \boldsymbol{\xi}^\top \boldsymbol{C} \, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \, \boldsymbol{w}' + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \boldsymbol{y}^k \\ \hspace{8cm} - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w}' + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}', \bar{\boldsymbol{\xi}}) \end{array} \right\} \begin{array}{l} \bar{\boldsymbol{\xi}} \in \widehat{\Xi}^k, \\ \ \\ \forall k \in \mathcal{K}. \end{array}$$

$$(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}))$$

Any problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}))$ defined on partition $\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}$, where $\bigcup_{k \in \mathcal{K}} \widehat{\Xi}^k = \widehat{\Xi}$ serves as a conservative approximation to $\mathcal{IP}_{\boldsymbol{w}}(\widehat{\Xi})$. Among these fixed partition problems, the one that achieves an optimal partition of $\widehat{\Xi}$ will have the lowest objective value, which coincides with the value of $\mathcal{IP}_{\boldsymbol{w}}(\widehat{\Xi})$.

Once we have obtained an optimal solution $(\bar{\theta}, \ \bar{\boldsymbol{\psi}}, \ \bar{\boldsymbol{x}}, \ \{\bar{\boldsymbol{y}}^k\}_{k \in \mathcal{K}})$ to $\mathcal{IP}_{\boldsymbol{w}}(\widehat{\Xi})$, we solve the following separation problem to check whether the current solution is feasible and optimal across the entire uncertainty set Ξ

$$z = \max_{\bar{\boldsymbol{\xi}} \in \Xi} \ \min_{k \in \mathcal{K}} \ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}', \bar{\boldsymbol{\xi}})} \ \max \left\{ \boldsymbol{c}^\top \bar{\boldsymbol{\psi}} + \boldsymbol{\xi}^\top \boldsymbol{C} \, \bar{\boldsymbol{x}} + \boldsymbol{\xi}^\top \boldsymbol{D} \, \boldsymbol{w}' + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \bar{\boldsymbol{y}}^k - \bar{\boldsymbol{\psi}}^\top \boldsymbol{g}(\boldsymbol{\xi}) - \bar{\theta} \, , \right.$$

$$\left. \max_{l \in \{1,\dots,L\}} \left\{ [\boldsymbol{V}(\boldsymbol{\xi})]_l \bar{\boldsymbol{x}} + [\boldsymbol{T}(\boldsymbol{\xi})]_l \boldsymbol{w}' + [\boldsymbol{W}(\boldsymbol{\xi})]_l \bar{\boldsymbol{y}}^k - [\boldsymbol{H}]_l \boldsymbol{\xi} \right\} \right\} ,$$

$$(\mathcal{IP}^{\text{out}}_{\boldsymbol{w}})$$

where $l \in \{1, \dots, L\}$ indices the uncertain constraints. If $z > 0$, this indicates that either the solution is infeasible due to the maximum constraint violation taking a positive value or that there is an optimal $\bar{\boldsymbol{\xi}}$ causing an objective value larger than θ'. Theorem A.1 shows that problem $(\mathcal{IP}^{\text{out}}_{\boldsymbol{w}})$ is amenable to an MIO reformulation.

To obtain an optimal solution of $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$, we extend the branch-and-bound algorithm proposed in Subramanyam et al (2020, Section 3.1) to a two-and-a-half-stage robust optimization problem. The algorithm starts by solving problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}))$ with $\widehat{\Xi} = \{\bar{\boldsymbol{\xi}}_0\}$, where $\bar{\boldsymbol{\xi}}_0$ is an arbitrary scenario from Ξ. After getting an optimal solution, it solves the separation problem $(\mathcal{IP}^{\text{out}}_{\boldsymbol{w}})$ to check for any

violations. Note that instead of optimizing over $\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}$, problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}))$ fixes the partition of $\widehat{\Xi}$. So to achieve an optimal solution to $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$, it separately adds the violated scenario into $\widehat{\Xi}^1, \ldots, \widehat{\Xi}^k$ to create K new scenario-based problems. The procedure stops when there is no violation. We summarize the algorithm in Algorithm 2.

Algorithm 2: branch-and-bound algorithm for solving problem $(\mathcal{IP}_{\boldsymbol{w}})$

Inputs: Initial scenario $\overline{\boldsymbol{\xi}}_0$ and number of policies K.

Output: Optimal solution $(\boldsymbol{\psi}^\star, \boldsymbol{x}^\star, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}})$, and optimal objective
  value θ^* of problem $(\mathcal{IP}_{\boldsymbol{w}})$.

Initialize: Let $\{\widehat{\Xi}^1, \ldots, \widehat{\Xi}^K\} \leftarrow \{\{\overline{\boldsymbol{\xi}}_0\}, \{\}, \ldots, \{\}\}$, set problem
  list $\mathcal{Q} \leftarrow \{\mathcal{P}'(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}})\}$, and $\theta^\star \leftarrow +\infty$.

while $\mathcal{Q} \neq \emptyset$ **do**

  1. Select, solve and remove a problem $\mathcal{P}'(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}})$ from $\mathcal{Q}$.
     Let $(\overline{\theta}, \overline{\boldsymbol{\psi}}, \overline{\boldsymbol{x}}, \{\overline{\boldsymbol{y}}^k\}_{k \in \mathcal{K}})$ be an optimal solution; if $\theta' > \theta^\star$, continue.

  2. Solve the separation problem $(\mathcal{IP}_{\boldsymbol{w}}^{\text{out}})$ let z be the optimal objective value
     and $\overline{\boldsymbol{\xi}}$ be an optimal solution. If $z > 0$, create K problems $\mathcal{P}'(\{\widehat{\Xi}^1, \ldots, \widehat{\Xi}^k \cup \{\overline{\boldsymbol{\xi}}\}, \ldots, \widehat{\Xi}^K\})$, for each $k \in \mathcal{K}$, and add it to the problem list $\mathcal{Q}$; else,
     set $(\theta^\star, \boldsymbol{\psi}^\star, \boldsymbol{x}^\star, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}}) \leftarrow (\overline{\theta}, \overline{\boldsymbol{\psi}}, \overline{\boldsymbol{x}}, \{\overline{\boldsymbol{y}}^k\}_{k \in \mathcal{K}})$.

end

Result: Report $(\theta^\star, \boldsymbol{\psi}^\star, \boldsymbol{x}^\star, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}})$ as an optimal solution if $\theta^\star < +\infty$,
  and problem is infeasible otherwise.

4.2.2 A Branch-and-Cut Algorithm.

In step 1 of Algorithm 2, problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}))$ can be reformulated as a single-stage MILO by dualization. However, Theorem A.2 shows that the size of the problem grows with $|\widehat{\Xi}|$ with the number of dual variables being $O(|\widehat{\Xi}|)$ and the number of constraints being $O(|\widehat{\Xi}|)$, which is computationally challenging. Therefore, we propose to solve the robust optimization problem using a constraint generation algorithm, which will be embedded within the branch-and-cut algorithm that we now present.

23

We start with the cut generation algorithm for solving problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$. Given collections of scenarios $\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}$, the following relaxation of $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ is solved

$$
\begin{aligned}
\min \quad & \theta \\
\text{s.t.} \quad & \theta \in \mathbb{R}, \ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{y}^k \in \mathcal{Y} \quad \forall k \in \mathcal{K}. \\
& \left.
\begin{aligned}
\theta \geq \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\boldsymbol{\xi} \in \widehat{\Xi}_{\text{in}}^k} \ & \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w}' + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k \\
& \qquad\qquad\qquad\qquad -\boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
\boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w}' + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \widehat{\Xi}_{\text{in}}^k
\end{aligned}
\right\} \ \forall k \in \mathcal{K},
\end{aligned}
$$

$$(\mathcal{IP}''_{\boldsymbol{w}}(\{\widehat{\Xi}_{\text{in}}^k\}_{k\in\mathcal{K}}))$$

where $\widehat{\Xi}_{\text{in}}^k$ is a finite subset of $\bigcup_{\overline{\boldsymbol{\xi}}\in\widehat{\Xi}^k}\Xi(\boldsymbol{w}',\overline{\boldsymbol{\xi}})$. After getting a solution $(\overline{\theta}, \overline{\boldsymbol{\psi}}, \overline{\boldsymbol{x}}, \{\overline{\boldsymbol{y}}^k\}_{k\in\mathcal{K}})$ of $(\mathcal{IP}''_{\boldsymbol{w}}(\{\widehat{\Xi}_{\text{in}}^k\}_{k\in\mathcal{K}}))$, the following subproblems are solved to evaluate the violation of the constraints in $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$

$$
\begin{aligned}
v^k = \max_{\overline{\boldsymbol{\xi}}\in\widehat{\Xi}^k} \max_{\boldsymbol{\xi}\in\Xi(\boldsymbol{w}',\overline{\boldsymbol{\xi}})} \ \max\Big\{ & \boldsymbol{c}^\top\overline{\boldsymbol{\psi}} + \boldsymbol{\xi}^\top\boldsymbol{C}\,\overline{\boldsymbol{x}} + \boldsymbol{\xi}^\top\boldsymbol{D}\,\boldsymbol{w}' + \boldsymbol{\xi}^\top\boldsymbol{Q}\,\overline{\boldsymbol{y}}^k - \overline{\boldsymbol{\psi}}^\top\boldsymbol{g}(\boldsymbol{\xi}) - \overline{\theta}, \\
& \max_{l\in\{1,\dots,L\}} \big\{ \boldsymbol{\xi}^\top\boldsymbol{T}_l\overline{\boldsymbol{x}} + \boldsymbol{\xi}^\top\boldsymbol{V}_l\boldsymbol{w}' + \boldsymbol{\xi}^\top\boldsymbol{W}_l\overline{\boldsymbol{y}}^k - [\boldsymbol{H}]_l\boldsymbol{\xi} \big\} \Big\} \ \forall k \in \mathcal{K}.
\end{aligned}
$$

$$(\mathcal{IP}_{\boldsymbol{w}}^{\text{in}})$$

We have $v^k \leq 0$ for all $k \in \mathcal{K}$, if and only if a robust optimal solution of $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ has been found.

The branch-and-cut algorithm for evaluating $\Phi(\boldsymbol{w}')$ starts by solving problem $(\mathcal{IP}''_{\boldsymbol{w}}(\{\widehat{\Xi}_{\text{in}}^k\}_{k\in\mathcal{K}}))$. At incumbent nodes, it checks the inner robust feasibility by solving problem $(\mathcal{IP}_{\boldsymbol{w}}^{\text{in}})$ and adds a cut until a feasible solution to $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ is found. The feasible solution is then used to create branches or update the bounds as stated in Algorithm 2. Algorithm 3 summarizes the procedure, which we also illustrate in Figure 2.

Algorithm 3: branch-and-cut algorithm for solving problem ($\mathcal{IP}_{\boldsymbol{w}}$).

Inputs: Initial scenario $\overline{\boldsymbol{\xi}}_0$, vector $\boldsymbol{w}'$, and number of policies K.

Output: Optimal solution ($\boldsymbol{\psi}^\star$, $\boldsymbol{x}^\star$, $\{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}$) and optimal objective

value $\theta^\star$ of problem ($\mathcal{IP}_{\boldsymbol{w}}$).

Initialize: Set $\{\widehat{\Xi}^1,\ldots,\widehat{\Xi}^K\} \leftarrow \{\{\overline{\boldsymbol{\xi}}_0\},\{\},\ldots,\{\}\}$, $\widehat{\Xi}_{\text{in}}^k \leftarrow \widehat{\Xi}^k$, $\forall k\in\mathcal{K}$,

$\theta^\star \leftarrow +\infty$. Start the branch-and-bound tree from problem $\mathcal{P}''(\{\widehat{\Xi}_{in}^k\}_{k\in\mathcal{K}})$.

1. When an incumbent node is reached, get solution ($\overline{\theta}$, $\overline{\boldsymbol{\psi}}$, $\overline{\boldsymbol{x}}$, $\{\overline{\boldsymbol{y}}^k\}_{k\in\mathcal{K}}$), and

solve problem ($\mathcal{IP}_{\boldsymbol{w}}^{\text{in}}$). If $\max_{k\in\mathcal{K}} v^k > 0$, go to step 2; else, go to step 3.

2. For all $k\in\mathcal{K}$, let $\boldsymbol{\xi}^k$ be an optimal solution to the kth problem; if $v^k > 0$,

set $\widehat{\Xi}_{\text{in}}^k \leftarrow \widehat{\Xi}_{\text{in}}^k \bigcup\{\boldsymbol{\xi}^k\}$. Go to step 5.

3. Solve problem ($\mathcal{IP}_{\boldsymbol{w}}^{\text{out}}$). If $z > 0$, let $\overline{\boldsymbol{\xi}}$ be an optimal solution, go to step 4;

else, if $\overline{\theta} \leq \theta^\star$, set $\theta^\star \leftarrow \overline{\theta}$. Go to step 1.

4. Create K branches, each for problem $\mathcal{P}'(\{\widehat{\Xi}^1,\ldots,\widehat{\Xi}^k\cup\{\overline{\boldsymbol{\xi}}\},\ldots,\widehat{\Xi}^K\})$, $k\in\mathcal{K}$.

For each branch, start with solving the corresponding relaxation

problem $\mathcal{P}''(\{\widehat{\Xi}_{in}^1,\ldots,\widehat{\Xi}_{in}^k\cup\{\overline{\boldsymbol{\xi}}\},\ldots,\widehat{\Xi}_{in}^K\})$. Go to step 5.

5. If the global lower bound $lb_{\text{global}} \geq \theta^\star$ or no feasible node exist, terminate

with status `Opt` or `Infeas`, and report the current solution; else if the node

lower bound $lb_{\text{node}} > \theta^\star$, fathom the node. Go to step 1.

Result: The branch-and-bound tree will end with either giving an optimal

solution ($\theta^\star$, $\boldsymbol{\psi}^\star$, $\boldsymbol{x}^\star$, $\{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}$) or identifying that the problem as infeasible.

4.2.3 Convergence Analysis

We discuss the correctness as well as the asymptotic convergence of Algorithm 3 in

the following theorem.

Theorem 5. *If the branch-and-cut Algorithm 3 terminates, then it either returns*

an optimal solution to problem ($\mathcal{IP}_{\boldsymbol{w}}$) or correctly identifies it as infeasible. If there

is no finite convergence, then in an infinite branch, every accumulation point of the

sequence of the solutions obtained in step 1 of Algorithm 3 corresponds an optimal

solution ($\boldsymbol{\psi}^\star$, $\boldsymbol{x}^\star$, $\{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}$) of problem ($\mathcal{IP}_{\boldsymbol{w}}$) with objective value $\theta^\star$.
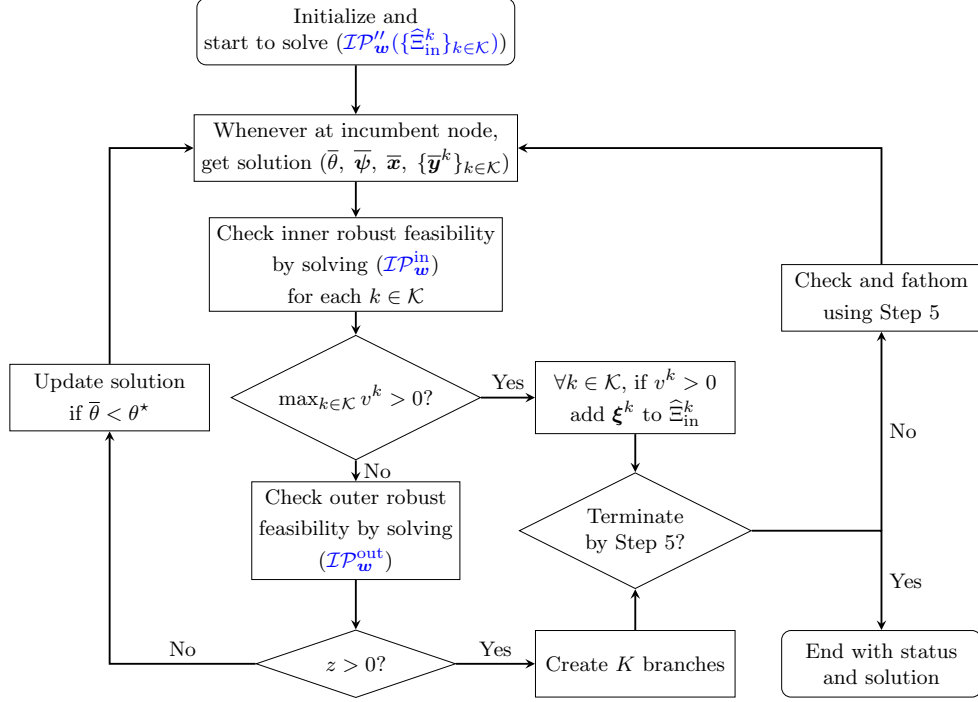
Fig. 2 Diagram of Algorithm 3 for solving problem ($\mathcal{IP}_{\boldsymbol{w}}$)

4.3 Improved *L*-shaped Algorithm

In this section, we introduce three types of cuts to be added to Algorithm 1, which have the potential to improve its performance (see Table 3).

4.3.1 Benders' Cut

A popular family of cuts in the integer programming literature is Benders' cuts, see Benders (1962). In each iteration of Algorithm (1), for a fixed $\boldsymbol{w}^r$, we relax the integrality of the decisions and solve an LP relaxation of problem ($\mathcal{IP}_{\boldsymbol{w}}$) and get the optimal dual solution to calculate the Benders' cut. Specifically, we consider the

following LP problem

$$
\begin{aligned}
\min \quad & \theta \\
\text{s.t.} \quad & \theta \in \mathbb{R},\ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g},\ \boldsymbol{x} \in \mathcal{X}^{\mathrm{LP}},\ \boldsymbol{y}(\boldsymbol{\xi}) \in \mathcal{Y}^{\mathrm{LP}} \\
& \boldsymbol{w} = \boldsymbol{w}^r \\
& \left. \begin{array}{l} \theta \geq \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\, \boldsymbol{y}(\boldsymbol{\xi}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}(\boldsymbol{\xi}) \leq \boldsymbol{H}\boldsymbol{\xi} \end{array} \right\} \quad \forall \boldsymbol{\xi} \in \widehat{\Xi},
\end{aligned}
\tag{9}
$$

where $\mathcal{X}^{\mathrm{LP}}$ and $\mathcal{Y}^{\mathrm{LP}}$ are linear relaxations of $\mathcal{X}$ and $\mathcal{Y}$ by allowing all variables to be continuous, and $\widehat{\Xi}$ can be any finite subset of Ξ.

Problem (9) is a relaxation of $(\mathcal{IP}_{\boldsymbol{w}})$ in the sense that (i) it solves a fully adaptable problem on a finite set $\widehat{\Xi}$, (ii) it ignores the fact that $\boldsymbol{y}(\boldsymbol{\xi})$ needs to be robust to the unobserved portion of $\boldsymbol{\xi}$, and (iii) the mixed-integer sets $\mathcal{X}$ and $\mathcal{Y}$ are replaced by their linear relaxation. The Benders' cut $(\mathcal{C}_{\mathrm{Benders}})$ added in step 3 of Algorithm 1 is generated by obtaining the optimal objective value $\overline{\theta}$ of problem (9), and the values $\boldsymbol{\pi}$ of the dual multipliers associated with the constraints $\boldsymbol{w} = \boldsymbol{w}^r$

$$
\theta \geq \overline{\theta} + \boldsymbol{\pi}^\top (\boldsymbol{w} - \boldsymbol{w}^r). \tag{$\mathcal{C}_{\mathrm{Benders}}$}
$$

In the implementation, we can either randomly generate or collect the scenario generated in step 3 of Algorithm 3 to construct set $\widehat{\Xi}$.

4.3.2 Strengthened Feasibility Cut

If problem (1) contains a subset of constraints in the form

$$
\boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} \leq \boldsymbol{H}\boldsymbol{\xi}, \ \ \forall \boldsymbol{\xi} \in \Xi, \tag{10}
$$

and the matrix $\boldsymbol{V}(\boldsymbol{\xi})$ is component-wise non-negative for all $\boldsymbol{\xi} \in \Xi$, then for any infeasible solution $\boldsymbol{w}^r$, a solution $\boldsymbol{w}$ with $\boldsymbol{w} \geq \boldsymbol{w}^r$ is also infeasible for problem ($\mathcal{P}$). In this scenario, we incorporate the following strengthened feasibility cut in step 2 of Algorithm 1

$$\sum_{i \in \mathcal{I}_r} 1 - w_i \geq 1. \qquad (\mathcal{C}_{\text{s-feas}})$$

The cut ($\mathcal{C}_{\text{s-feas}}$) is stronger than ($\mathcal{C}_{\text{feas}}$) in the sense that the former excludes all solutions $\boldsymbol{w} \geq \boldsymbol{w}^r$ from the feasible region, while the latter excludes only $\boldsymbol{w}^r$ itself.

Theorem 6. *If problem* (1) *contains a subset of constraints in the form of* (10), *and the matrix $\boldsymbol{V}(\boldsymbol{\xi})$ is component-wise non-negative for all $\boldsymbol{\xi} \in \Xi$, then for any solution $\boldsymbol{w}^r$ that is infeasible to problem* (1), *any solutions that violate constraint* ($\mathcal{C}_{\text{s-feas}}$) *are also infeasible.*

4.3.3 Strengthened Optimality Cut

When the variable $\boldsymbol{w}$ only appears in the objective function, and the cost vector of it is deterministic, problem ($\mathcal{IP}_{\boldsymbol{w}}$) can be written as

$$\begin{aligned} \min \quad & \boldsymbol{d}^\top \boldsymbol{w} + \Phi(\boldsymbol{w}) \\ \text{s.t.} \quad & \boldsymbol{w} \in \mathcal{W}, \end{aligned} \qquad (11)$$

where

$$\Phi(\boldsymbol{w}) = \min \quad \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\bar{\boldsymbol{\xi}} \in \Xi} \left\{ \begin{array}{l} \min\limits_{k \in \mathcal{K}} \max\limits_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}})} \boldsymbol{\xi}^\top \boldsymbol{C} \, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \text{s.t.} \quad \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}}) \end{array} \right\} \quad (12)$$
$$\text{s.t.} \quad \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \; \boldsymbol{x} \in \mathcal{X}, \; \boldsymbol{y}^k \in \mathcal{Y}, \; \forall k \in \mathcal{K}.$$

For the function $\Phi(\boldsymbol{w})$ defined in (12), consider two vectors $\boldsymbol{w}$ and $\boldsymbol{w}^r$. If $\boldsymbol{w}$ is component-wise smaller than $\boldsymbol{w}^r$, i.e., $\boldsymbol{w} \leq \boldsymbol{w}^r$, then $\Phi(\boldsymbol{w}) \geq \Phi(\boldsymbol{w}^r)$. The intuition behind this is that if more entries of $\boldsymbol{w}$ take the value 1, the decision-maker has more

information when making the recourse decision. This leads to the function $\Phi(\boldsymbol{w})$ having a non-decreasing value, as discussed in Paradiso et al (2022, Proposition 2). When applicable, we use the following strengthened optimality cut in step 3 of Algorithm 1

$$\theta \geq \Phi(\boldsymbol{w}') - (\Phi(\boldsymbol{w}') - L) \sum_{i \notin \mathcal{I}_r} w_i. \qquad (\mathcal{C}_{\text{s-opt}})$$

4.3.4 Upper Cutoff

To avoid spending time evaluating the function value $\Phi(\boldsymbol{w}')$ with suboptimal $\boldsymbol{w}'$, in the initialization step of Algorithm 3, we set the cutoff value as the current upper bound $\theta^\star$ found by Algorithm 1. The termination condition for Algorithm 3 is met when the current lower bound exceeds the cutoff value. In such cases, $\boldsymbol{w}'$ is proven to be suboptimal. In the case of early termination, the integer cut $(\mathcal{C}_{\text{opt}})$ is generated using the cutoff value.

The alternative to step 2 of Algorithm 1 is as follows:

$2'$: Evaluate the value $\Phi(\boldsymbol{w}')$ with cutoff value $\theta^\star$.

The additional initialization and add-on to step 5 of Algorithm 3 are given as follows:

Initialize: Set cutoff value $\theta_{\text{ub}} \leftarrow \theta^\star$.

5^+: If the global lower bound $\text{lb}_{\text{global}} > \theta_{\text{ub}}$, terminate and return the current incumbent solution.

5 Numerical Results

In this section, we investigate the performance of our approach on the best box and R&D project portfolio optimization problems. We show the trade-off between optimality and computational effort as the number of policies K is varied and the suboptimality of the robust optimization approach proposed in Vayanos et al (2020) under worst-case distribution, thereby demonstrating the value of our method relative

to existing literature. In both problems, we report the results of solving the MINLO problem by Gurobi 9.0.1. In the R&D project portfolio optimization problem, we showcase the attractive scalability properties of our proposed decomposition algorithm. To solve the robust optimization counterpart, where the ambiguity set (2) only includes the support, we employ the ROC++ package Vayanos et al (2022) for coding and solving the problem. Furthermore, in Appendix B.3, we present the numerical results of an extension to the R&D problem. The results show that the proposed solution method works equally effectively as in the R&D problem.

All of our experiments are performed on the High-Performance Computing Cluster of our university. Each job is allocated 2GB of RAM, 4 cores, and a 2.1 GHz Xeon processor. For each instance, an optimality gap of 0.1% and a time limit of 7,200 seconds are allowed to solve the problem.

5.1 Distributionally Robust Best Box problem

5.1.1 Problem Description

We perform experiments on several instances of the two-stage best box problem that subsumes selection problems as special cases. It is a distributionally robust variant of the robust problem in Vayanos et al (2020, 2022). In this problem, we have N boxes indexed in the set $\mathcal{N} = \{1, \ldots, N\}$ to choose from. Each box n contains an unknown prize with value ξ_n^{r}, and opening it incurs an unknown cost ξ_n^{c}. The return and cost are uncertain and will only be revealed if the box is opened. In the first stage, the decision-maker decides whether to open each box $n \in \mathcal{N}$, which we indicate with the decision variable $w_n^r \in \{0, 1\}$. Thus, ξ_n^r and ξ_n^c are revealed between the first and second time stage if only $w_n^r = 1$. The total budget available to open boxes is B. In the second stage, the decision-maker keeps one of the opened boxes and earns its value, indicated with the decision variable $y_n \in \{0, 1\}$. The goal of the decision-maker is to select the boxes to open and keep to maximize the worst-case expected value of the box

kept. We assume that the values and costs are expressible as $\xi_n^{\mathrm{r}} = (1 + \boldsymbol{\Phi}_n^\top \boldsymbol{\zeta}/2) r_n$ and $\xi_n^{\mathrm{c}} = (1 + \boldsymbol{\Psi}_n^\top \boldsymbol{\zeta}/2) c_n$, where r_n and c_n correspond to the nominal value and cost for box n, respectively, $\boldsymbol{\zeta} \in [-1, 1]^M$ are M risk factors, and the vectors $\boldsymbol{\Phi}_n, \boldsymbol{\Psi}_n \in \mathbb{R}^M$ collect the loading factors for box n. The support of the distribution of the uncertain parameters is described as follows

$$\Xi = \left\{ (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}}) \in \mathbb{R}^{2N} : \exists \boldsymbol{\zeta} \in [-1, 1]^M : \xi_n^{\mathrm{r}} = \left(1 + \boldsymbol{\Phi}_n^\top \boldsymbol{\zeta}/2 \right) r_n, \ \xi_n^{\mathrm{c}} = \left(1 + \boldsymbol{\Psi}_n^\top \boldsymbol{\zeta}/2 \right) c_n, \right.$$
$$\left. n \in \mathcal{N} \right\}.$$

We define $\boldsymbol{\xi} = (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}})$, $\boldsymbol{\mu} = (\boldsymbol{r}, \boldsymbol{c})$, and use the following ambiguity set for the uncertain values and costs

$$\mathcal{P}_{\mathrm{BB}} = \left\{ \mathbb{P} \in \mathcal{M}_+(\mathbb{R}^{N_\xi}) \ : \ \mathbb{P}(\boldsymbol{\xi} \in \Xi) = 1, \ \mathbb{E}_{\mathbb{P}} \left[\left\{ |\mathbf{e}^\top (\boldsymbol{\xi} - \boldsymbol{\mu})| \right\} \right] \leq 0.25 N^{-1/2} \mathbf{e}^\top \boldsymbol{\mu} \right\}. \tag{13}$$

The last constraint in the ambiguity set follows the intuition of the central limit theorem, and it imposes an upper bound on the cumulative deviation of the boxes' returns and costs from their expected values.

The distributionally robust best box problem can be written as

$$\begin{aligned}
\min \quad & \sup_{\mathbb{P} \in \mathcal{P}} \ \mathbb{E}_{\mathbb{P}} \left[-\boldsymbol{\xi}^{\mathrm{r}\top} \boldsymbol{y}(\boldsymbol{\xi}) \right] \\
\mathrm{s.\,t.} \quad & \boldsymbol{w} = (\boldsymbol{w}^r, \boldsymbol{w}^r), \ \boldsymbol{w}^r \in \{0, 1\}^N \\
& \left. \begin{aligned}
& \boldsymbol{y}(\boldsymbol{\xi}) \in \{0, 1\}^N \\
& \boldsymbol{\xi}^{\mathrm{c}\top} \boldsymbol{w}^r \leq B \\
& \boldsymbol{y}(\boldsymbol{\xi}) \leq \boldsymbol{w}^r \\
& \mathbf{e}^\top \boldsymbol{y}(\boldsymbol{\xi}) = 1
\end{aligned} \right\} \quad \forall \boldsymbol{\xi} \in \Xi, \\
& \boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}(\boldsymbol{\xi}') \quad \forall \boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi \ : \ \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}'.
\end{aligned} \tag{14}$$

Besides the support, the ambiguity set (13) has only one constraint, leading to a scalar dual variable in the MINLO reformulation of (14). Therefore, the reformulation has only one bilinear term with continuous variables, and the reformulated MINLO problem can be handled by Gurobi efficiently.

5.1.2 Computational Results

We evaluate the performance of our approach on 80 randomly generated instances of problem (14): 20 instances for each $N \in \{10, 20, 30, 40\}$ with risk factors $M \in \{5, 10, 15, 20\}$. In each instance, c is drawn uniformly at random from the hypercube $[0, 10]^N$, and $r = c/5$, $B = e^\top c/2$. The risk loading factors Φ_n and Ψ_n are sampled uniformly at random from the hypercube $[-1, 1]^M$. We solve the K-adaptability counterpart of (14) for $K \in \{1, 2, 3, 4\}$. We summarize our results in the following.

In the best box problem, as the number of bilinear terms between continuous variables is small, solving the monolithic MINLO is more efficient than using the decomposition algorithm in all cases. Table 1 summarizes the computational results across the generated instances. The improvement over the static solution increases rapidly with the number of policies. The average improvements over the static solution for all sizes are 87.1%, 111.3%, 121.3% for $K = 2$, 3, 4, respectively. The improvement in solution quality comes at a computational cost.

To study the quality of the solutions obtained by the K-adaptability approximation, we solve a small-sized problem with $N = 10$ to optimality for K varying from 1 to 11. We report the average improvements over the static solution as K increases in Section B.1.2. The results show that a small number of policies (3 out of 11) is sufficient for the K-adaptability approximation (5) to be optimal in the original problem (4).

We now assess the suboptimality of an optimal solution to the RO problem in the DRO problem. To compute such suboptimality, we first solve the robust optimization

32

K	$N = 10$	$N = 20$	$N = 30$	$N = 40$
1	$0.0\%/0.1s/0.0\%$	$0.0\%/0.2s/0.0\%$	$0.0\%/0.4s/0.0\%$	$0.0\%/0.5s/0.0\%$
2	$42.6\%/0.2s/0.0\%$	$86.5\%/6.0s/0.0\%$	$118.2\%/1090.5s/0.0\%$	$101.1\%/2530.6s/0.0\%$
3	$43.9\%/0.4s/0.0\%$	$117.2\%/8.8s/0.0\%$	$148.9\%/1168.9s/0.0\%$	$135.2\%/2297.8s/0.0\%$
4	$43.9\%/0.8s/0.0\%$	$125.4\%/16.8s/0.0\%$	$162.2\%/3031.6s/0.0\%$	$153.5\%/4819.0s/0.0\%$

Table 1 Summary of computational results for the best box problem. Each entry corresponds to: the average objective value improvement of the best K-adaptability solution found in the time limit over the best static solution found in the time limit; the improvement is calculated as the ratio of the difference between the objective values of the K-adaptability solution and the static solution to the objective value of the static solution / the average solution time of the solved instances/ the average optimality gap across unsolved cases.

counterpart of problem (14), where the ambiguity set (13) includes only support information. Let $(\boldsymbol{x}_{\mathrm{RO}},\ \boldsymbol{w}_{\mathrm{RO}},\ \{\boldsymbol{y}_{\mathrm{RO}}^{k}\}_{k\in\mathcal{K}})$ denote an optimal solution to the RO problem, we fix these values in (14) and solve the inner maximization problem to obtain the objective value of the RO solution under the worst case distribution, denoted by $z_{\mathrm{RO}}^{\star}$. Accordingly, we let $z_{\mathrm{DRO}}^{\star}$ denote the optimal objective value obtained by optimizing problem (14). Table 2 summarizes the relative difference between $z_{\mathrm{RO}}^{\star}$ and $z_{\mathrm{DRO}}^{\star}$. From the table, we see that the average suboptimality of the RO solution is as high as 40.59%.

K	$N = 10$	$N = 20$	$N = 30$	$N = 40$
1	40.59%	13.19%	3.94%	6.54%
2	25.40%	1.60%	1.18%	1.43%
3	20.09%	2.10%	1.52%	0.77%
4	19.65%	1.98%	1.05%	1.75%

Table 2 Suboptimality of the RO solutions to the best box problem. Each cell represents an average of 20 instances. In each instance, the relative differences are calculated by $\dfrac{(z_{\mathrm{DRO}}^{\star} - z_{\mathrm{RO}}^{\star})}{(z_{\mathrm{DRO}}^{\star} + z_{\mathrm{RO}}^{\star})/2}$.

To further assess the performance of the DRO and RO solutions, in Appendix B.1.3, we assess the out-of-sample performance of these solutions. The results show that the

DRO solutions, in general, outperform the RO solutions also under non-worst-case distributions.

5.2 Distributionally Robust R&D Project Portfolio Optimization

5.2.1 Problem Description

We next perform experiments on several instances of the distributionally robust R&D project portfolio optimization problem – this is a DRO variant of the robust version proposed in Vayanos et al (2020, 2022). In this problem, there are N projects indexed in the set $\mathcal{N} = \{1, \ldots, N\}$, and the decision-maker decides whether and when to invest in a project. Each project n has an uncertain return ξ_n^{r}, and undertaking it incurs an uncertain cost ξ_n^{c}. The return and cost of the project n can only be observed if we undertake this project. The decision to invest in project n in the first (resp. second) stage is denoted by w_n^r (resp. y_n), $n \in \mathcal{N}$. Thus, ξ_n^r and ξ_n^c are revealed between the first and second time stage if only $w_n^r = 1$. The total budget available across two years is B. A project can be invested in at most once, and only a fraction $\theta \in [0, 1)$ of the return is obtained for investments made in the second stage. The goal of the decision-maker is to maximize the worst-case expected return. We assume the returns and costs are expressible as $\xi_n^{\mathrm{r}} = (1 + \mathbf{\Phi}_n^\top \boldsymbol{\zeta}/2) r_n$ and $\xi_n^{\mathrm{c}} = (1 + \mathbf{\Psi}_n^\top \boldsymbol{\zeta}/2) c_n$, where r_n, c_n correspond to the nominal return and cost of project n, respectively, $\boldsymbol{\zeta} \in [-1, 1]^M$ are M risk factors, and the vectors $\mathbf{\Phi}_n$ and $\mathbf{\Psi}_n \in \mathbb{R}^M$ collect the loading factors for project n. The support of the distribution of the uncertain parameters is described as follows

$$\Xi = \big\{ (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}}) \in \mathbb{R}^{2N} : \exists \boldsymbol{\zeta} \in [-1, 1]^M : \xi_n^{\mathrm{r}} = \big(1 + \mathbf{\Phi}_n^\top \boldsymbol{\zeta}/2 \big) r_n, \ \xi_n^{\mathrm{c}} = \big(1 + \mathbf{\Psi}_n^\top \boldsymbol{\zeta}/2 \big) c_n, \\ n \in \mathcal{N} \big\} .$$

We define $\boldsymbol{\xi} = (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}})$, $\boldsymbol{\mu} = (\boldsymbol{r}, \boldsymbol{c})$, and use the following ambiguity set for the uncertain returns and costs

$$
\mathcal{P}_{\mathrm{RD}} = \Big\{ \mathbb{P} \in \mathcal{M}_+(\mathbb{R}^{N_\xi}) \; : \; \mathbb{P}(\boldsymbol{\xi} \in \Xi) = 1, \quad \mathbb{E}_{\mathbb{P}}\left[|\boldsymbol{\xi} - \boldsymbol{\mu}|\right] \leq 0.15\boldsymbol{\mu},
$$
$$
\mathbb{E}_{\mathbb{P}}\left[\left|\mathbf{e}^\top (\boldsymbol{\xi} - \boldsymbol{\mu})\right|\right] \leq 0.15 N^{-1/2} \mathbf{e}^\top \boldsymbol{\mu})\Big\}.
$$
$$(15)$$

The second set of constraints in the ambiguity set enforces the mean absolute deviation of the individual returns and costs to be bounded. The last constraint imposes an upper bound on the cumulative deviation of the projects' returns and costs from their expected value.

The distributionally robust R&D project portfolio optimization problem can be written as

$$
\begin{aligned}
\min \quad & \sup_{\mathbb{P} \in \mathcal{P}} \; \mathbb{E}_{\mathbb{P}}\left[-\boldsymbol{\xi}^{r\top}(\boldsymbol{w} + \theta\boldsymbol{y}(\boldsymbol{\xi}))\right] \\
\mathrm{s.\,t.} \quad & \boldsymbol{w} = (\boldsymbol{w}^r, \, \boldsymbol{w}^r), \; \boldsymbol{w}^r \in \{0, \, 1\}^N \\
& \left.\begin{aligned}
& \boldsymbol{y}(\boldsymbol{\xi}) \in \{0, \, 1\}^N \\
& \boldsymbol{\xi}^{c\top}(\boldsymbol{w}^r + \boldsymbol{y}(\boldsymbol{\xi})) \leq B \\
& \boldsymbol{w}^r + \boldsymbol{y}(\boldsymbol{\xi}) \leq \mathbf{e}
\end{aligned} \right\} \quad \forall \boldsymbol{\xi} \in \Xi \\
& \boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}(\boldsymbol{\xi}') \quad \forall \boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi \; : \; \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}'.
\end{aligned}
$$
$$(16)$$

5.2.2 Computational Results

We run experiments on 80 randomly generated instances of the K-adaptability counterpart of problem (16): 20 instances for each $N \in \{5, 10, 15, 20\}$ with risk factors $M \in \{4, 5, 8, 10\}$, respectively. We draw $\boldsymbol{c}$ uniformly at random from the box $[0, 10]^N$ and let $\boldsymbol{r} = \boldsymbol{c}/5$, $B = \mathbf{e}^\top \boldsymbol{c}/2$. The risk load factor $\boldsymbol{\Phi}_n$ and $\boldsymbol{\Psi}_n$ are sampled uniformly at random from the standard simplex.

We report the results of solving problem (16) by utilizing the monolithic MINLO reformulation and applying the decomposition Algorithm 1 with and without the cuts introduced in Section 4.3. In the decomposition Algorithm 1, we set the time

limit of evaluating $\Phi(\boldsymbol{w})$ to $[300s, 600s, 1200s, 2400s]$ for problems with size $N \in \{5,\ 10,\ 15,\ 20\}$. The lower bound value L in Algorithm 1 is found by solving a deterministic problem where the uncertain parameters are fixed to $\boldsymbol{\xi} = (\boldsymbol{r}, \boldsymbol{c})$. If the subproblem evaluating $\Phi(\boldsymbol{w})$ is not solved to optimality within the time limit, we use the current lower bound of the subproblem to generate a valid integer cut of the form ($\mathcal{C}_{\mathrm{opt}}$) and use the upper bound to update the objective value $\theta^\star$ in step 4. If the main problem is not solved to optimality within the time limit, we report $\theta^\star$ as the objective value and the current optimal objective value of the main problem ($\mathcal{P}_{\mathrm{relax}}$) as the lower bound.

N	K	MINLO					Decomposition					Decomposition with Added Cuts				
		Opt(#)	Time(s)	Gap	Better(#)	Improvement	Opt(#)	Time(s)	Gap	Better(#)	Improvement	Opt(#)	Time(s)	Gap	Better(#)	Improvement
	2	**20**	**1.6**	**0.0%**	**0**	**17.5%**	**20**	54.3	**0.0%**	**0**	**17.5%**	**20**	16.0	**0.0%**	**0**	**17.5%**
5	3	15	912.3	0.03%	**0**	**21.3%**	**20**	435.4	**0.0%**	**0**	**21.3%**	**20**	**239.0**	**0.0%**	**0**	**21.3%**
	4	0	7200	0.2%	**1**	**22.3%**	**20**	854.8	**0.0%**	0	**22.3%**	**20**	**475.0**	**0.0%**	1	**22.3%**
	2	19	**730.5**	-	**0**	16.9%	0	**7200**	27.2%	**0**	20.3%	**20**	1826.8	**0.0%**	**0**	**23.1%**
10	3	**0**	**7200**	-	2	11.8%	**0**	**7200**	30.2%	1	20.7%	**0**	**7200**	16.7%	16	**26.7%**
	4	**0**	**7200**	-	1	23.8%	**0**	**7200**	29.4%	2	21.6%	**0**	**7200**	16.5%	16	**27.4%**
	2	**0**	**7200**	-	**18**	16.1%	**0**	**7200**	32.9%	0	8.4%	**0**	**7200**	23.9%	2	**20.1%**
15	3	**0**	**7200**	-	7	21.5%	**0**	**7200**	30.5%	0	12.4%	**0**	**7200**	22.2%	13	**23.0%**
	4	**0**	**7200**	-	7	20.9%	**0**	**7200**	28.8%	0	15.1%	**0**	**7200**	21.9%	13	**23.2%**
	2	**0**	**7200**	-	**20**	**21.9%**	**0**	**7200**	36.9%	0	0.9%	**0**	**7200**	24.5%	0	18.7%
20	3	**0**	**7200**	-	7	20.7%	**0**	**7200**	34.8%	0	4.3%	**0**	**7200**	22.5%	13	**21.6%**
	4	**0**	**7200**	-	1	18.9%	**0**	**7200**	33.9%	0	5.7%	**0**	**7200**	21.5%	19	**23.6%**

Table 3 Summary of computational results of the R&D problem. **Decomposition** and **Decomposition with Added Cuts** refer to Algorithm 1 without and with the cuts introduced in Section 4.3, respectively. **Opt**(#) corresponds to the number of instances solved to optimality, **Time**(s) to the average computational time (in seconds) for instances solved to optimality, and **Gap** to the average optimality gap for the instances not solved within the time limit. We write '-' when no valid lower bound was found. **Better**(#) denotes the number of instances in each method that achieved a better objective value than the other method. **Improvement** denotes the average improvement in the objective value of the K-adaptability solution found in the time limit over the static solution found in the time limit. The improvement is calculated as the ratio of the difference between the objective values of the K-adaptability solution and the static solution to the objective value of the static solution.

Table 3 summarizes the computational results across those instances. From the table, we observe that the improved decomposition algorithm achieves optimality or finds better solutions for more instances than the basic decomposition algorithm and solves the monolithic MINLO problem when K is large. Across all instances, the proposed algorithm has a significantly smaller optimality gap,, which decreases with K. We notice that solving the MINLO problem often results in the inability to obtain a satisfactory or valid lower bound, especially when $N \geq 10$. As the number of policies K increases, the number of instances for which the improved decomposition algorithm achieves a better objective value increases. For example, when $N = 20$, the number of better solutions found by our approach is $\{0, 13, 19\}$ with $K \in \{2, 3, 4\}$, respectively. The improvements in the objective value over the static solutions of the improved decomposition algorithm are $\{18.7\%, 21.6\%, 23.6\%\}$, whereas the improvements obtained by solving the MINLO problem and using the basic decomposition algorithm are $\{21.9\%, 20.7\%, 18.9\%\}$ and $\{0.9\%, 4.3\%, 5.7\%\}$, respectively. Note that in some settings, MINLO solutions achieve a better objective value in more instances, but the average improvement in the objective value is worse. This discrepancy arises when the MINLO solutions are sometimes significantly suboptimal, thereby reducing the overall average improvement. Finally, in all cases, the improvement in the objective value over the static solutions of the decomposition algorithm increases with the number of policies K. This aligns with the rationale of the K-adaptability approach, where a larger K offers more flexibility for the recourse decisions, leading to a better objective value. However, the trend is not clear in MINLO solutions when $N \geq 10$ due to computational difficulties.

To study the quality of the solutions obtained by the K-adaptability approximation, in Section B.2.2, we solve a small-sized problem with $N = 4$ and $M = 2$ to optimality for all K varying from 1 to 16. We report the average improvements over the static solution as K increases. The results show that a small number of policies (5

out of 16) is sufficient for the K-adaptability approximation (5) to be optimal in the original problem (4).

Next, we examine the suboptimality of the RO solution. We define $z_{\text{RO}}^\star$ and $z_{\text{DRO}}^\star$ as in Section 5.1.2. The relative differences between $z_{\text{RO}}^\star$ and $z_{\text{DRO}}^\star$ are summarized in Table 4. From the table, we see that the average suboptimality of the RO solution

K	$N = 5$	$N = 10$	$N = 15$	$N = 20$
1	0.37%	1.60%	3.24%	2.08%
2	17.33%	23.03%	20.37%	19.61%
3	14.62%	19.15%	17.65%	18.33%
4	15.05%	11.25%	7.81%	8.73%

Table 4 Suboptimality of RO solution to the R&D problem. Each cell represents an average of 20 instances. In each instance, the relative differences are calculated by $\dfrac{(z_{\text{DRO}}^* - z_{\text{RO}}^*)}{(z_{\text{DRO}}^* + z_{\text{RO}}^*)/2}$.

among all instances is up to 23.03%, which shows the importance of incorporating distributional information in the uncertainty set.

To further assess the performance of the DRO and RO solutions, in Appendix B.2.3, we assess the out-of-sample performance of these solutions. The results show that the DRO approach, in general, performs better than RO on non-worst-case distributions.

6 Conclusion

In this paper, we studied two-stage distributionally robust optimization problems with decision-dependent information discovery. We proposed a framework for modeling these problems, a K-adaptability scheme to approximately solve them, and a decomposition algorithm to exactly solve the resulting K-adaptable problem. We demonstrated the effectiveness of our approaches in terms of computational efficiency, the benefits of adaptability, and out-of-sample performance, compared to the robust optimization framework, which assumes only support information is known.

There are several promising directions for future research. First, in this paper, we adopted a moment-based ambiguity set, motivated by the fact that data are not available during the design phase and only become available after implementing the first- and second-stage decisions. A natural extension would be to explore data-driven ambiguity sets in settings where data are available. This requires the development of alternative ambiguity sets, possibly in the spirit of Chen et al (2020) and Zhang et al (2024), which in turn would lead to different reformulations and corresponding solution methods. Second, it would be valuable to examine how solution quality varies with the parameter K. While there are some theoretical results in the literature regarding robust optimization, see Kurtz (2024), it is of practical importance to assess the performance of the K-adaptability method in the DRO problem with DDID. In our paper, we take a step in this direction through numerical experiments that evaluate the empirical improvement over the static solution as the number of policies increases. It would be interesting to further derive theoretical results, particularly regarding how the optimality gap varies with K and how many policies are required to achieve optimality. Finally, it would be interesting to extend the proposed algorithm to problems with multiple time stages, as this would broaden its applicability and enable it to address a wider range of practical problems.

Acknowledgements

References

Apap RM, Grossmann IE (2017) Models and computational strategies for multistage stochastic programming under endogenous and exogenous uncertainties. Computers & Chemical Engineering 103:233–274. https://doi.org/10.1016/j.compchemeng.2016.11.011

Balas E, Jeroslow R (1972) Canonical cuts on the unit hypercube. SIAM Journal on Applied Mathematics 23(1):61–69. https://doi.org/10.1137/0123007

Basciftci B, Ahmed S, Shen S (2021) Distributionally robust facility location problem under decision-dependent stochastic demand. European Journal of Operational Research 292(2):548–561. https://doi.org/10.1016/j.ejor.2020.11.002

Ben-Tal A, Den Hertog D, De Waegenaere A, Melenberg B, Rennen G (2013) Robust solutions of optimization problems affected by uncertain probabilities. Management Science 59(2):341–357. https://doi.org/10.1287/mnsc.1120.1641

Benders JF (1962) Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik 4(1):238–252. https://doi.org/10.1007/BF01386316

Bertsimas D, Caramanis C (2010) Finite adaptability in multistage linear optimization. IEEE Transactions on Automatic Control 55(12):2751–2766. https://doi.org/10.1109/TAC.2010.2049764

Bertsimas D, Sim M, Zhang M (2019) Adaptive distributionally robust optimization. Management Science 65(2):604–618. https://doi.org/10.1287/mnsc.2017.2952

Blankenship JW, Falk JE (1976) Infinitely constrained optimization problems. Journal of Optimization Theory and Applications 19(2):261–281.

https://doi.org/10.1007/BF00934096

Buchheim C, Kurtz J (2017) Min–max–min robust combinatorial optimization. Mathematical Programming 163:1–23. https://doi.org/10.1007/s10107-016-1053-z

Buchheim C, Kurtz J (2018) Complexity of min–max–min robustness for combinatorial optimization under discrete uncertainty. Discrete Optimization 28:1–15. https://doi.org/10.1016/j.disopt.2017.08.006

Chassein A, Goerigk M, Kurtz J, Poss M (2019) Faster algorithms for min-max-min robustness for combinatorial problems with budgeted uncertainty. European Journal of Operational Research 279:308–319. https://doi.org/10.1016/j.ejor.2019.05.045

Chen Z, Sim M, Xiong P (2020) Robust stochastic optimization made easy with RSOME. Management Science 66(8):3329–3339. https://doi.org/10.1287/mnsc.2020.3603

Chu M, Patton A, Roering J, Siebert C, Selker J, Walter C, Udell C (2021) SitkaNet: A low-cost, distributed sensor network for landslide monitoring and study. HardwareX 9:e00191. https://doi.org/10.1016/j.ohx.2021.e00191

Colvin M, Maravelias CT (2010) Modeling methods and a branch and cut algorithm for pharmaceutical clinical trial planning using stochastic programming. European Journal of Operational Research 203(1):205–215. https://doi.org/10.1016/j.ejor.2009.07.022

Delage E, Ye Y (2010) Distributionally robust optimization under moment uncertainty with application to data-driven problems. Operations Research 58(3):595–612. https://doi.org/10.1287/opre.1090.0741

Doan XV (2021) Distributionally robust optimization under endogenous uncertainty with an application in retrofitting planning. European Journal of Operational Research 300(1):73–84. https://doi.org/10.1016/j.ejor.2021.07.013

Gao R, Kleywegt A (2023) Distributionally robust stochastic optimization with wasserstein distance. Mathematics of Operations Research 48(2):603–655. https://doi.org/10.1287/moor.2022.1275

Goel V, Grossmann IE (2004) A stochastic programming approach to planning of offshore gas field developments under uncertainty in reserves. Computers & Chemical Engineering 28(8):1409–1429. https://doi.org/https://doi.org/10.1016/j.compchemeng.2003.10.005

Goel V, Grossmann IE (2006) A class of stochastic programs with decision dependent uncertainty. Mathematical programming 108(2):355–394. https://doi.org/10.1007/s10107-006-0715-7

Goh J, Sim M (2010) Distributionally robust optimization and its tractable approximations. Operations Research 58(4 PART 1):902–917. https://doi.org/10.1287/opre.1090.0795

Gupta V, Grossmann IE (2011) Solution strategies for multistage stochastic programming with endogenous uncertainties. Computers & Chemical Engineering 35(11):2235–2247. https://doi.org/10.1016/j.compchemeng.2010.11.013

Gupta V, Grossmann IE (2014) A new decomposition algorithm for multistage stochastic programs with endogenous uncertainties. Computers & Chemical Engineering 62:62–79. https://doi.org/10.1016/j.compchemeng.2013.11.011

Han E, Bandi C, Nohadani O (2023) On finite adaptability in two-stage distributionally robust optimization. Operations Research 71(6):2307–2327.

43

https://doi.org/10.1287/opre.2022.2273

Hanasusanto GA, Kuhn D (2018) Conic programming reformulations of two-stage distributionally robust linear programs over Wasserstein balls. Operations Research 66(3):849–869. https://doi.org/10.1287/opre.2017.1698

Hanasusanto GA, Kuhn D, Wiesemann W (2015) K-adaptability in two-stage robust binary programming. Operations Research 63(4):877–891. https://doi.org/10.1287/opre.2015.1392

Hanasusanto GA, Kuhn D, Wiesemann W (2016) K-adaptability in two-stage distributionally robust binary programming. Operations Research Letters 44(1):6–11. https://doi.org/10.1016/j.orl.2015.10.006

Jiang R, Guan Y (2016) Data-driven chance constrained stochastic program. Mathematical Programming 158(1-2):291–327. https://doi.org/10.1007/s10107-015-0929-7

Kurtz J (2024) How many policies do we need in k-adaptability for two-stage robust integer optimization? arXiv preprint arXiv:240912630

Laporte G, Louveaux FV (1993) The integer L-shaped method for stochastic integer programs with complete recourse. Operations Research Letters 13(3):133–142. https://doi.org/10.1016/0167-6377(93)90002-X

Luo F, Mehrotra S (2020) Distributionally robust optimization with decision dependent ambiguity sets. Optimization Letters 14(8):2565–2594. https://doi.org/10.1007/s11590-020-01574-3

Mohajerin Esfahani P, Kuhn D (2018) Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and

tractable reformulations. Mathematical Programming 171(1-2):115–166. https://doi.org/10.1007/s10107-017-1172-1

Nannicini G, Belotti P (2012) Rounding-based heuristics for nonconvex MINLPs. Mathematical Programming Computation 4(1):1–31. https://doi.org/10.1007/s12532-011-0032-x

Noyan N, Rudolf G, Lejeune M (2022) Distributionally robust optimization under a decision-dependent ambiguity set with applications to machine scheduling and humanitarian logistics. INFORMS Journal on Computing 34(2):729–751. https://doi.org/10.1287/ijoc.2021.1096

Omer J, Poss M, Rougier M (2024) Combinatorial Robust Optimization with Decision-Dependent Information Discovery and Polyhedral Uncertainty. Open Journal of Mathematical Optimization 5:5. https://doi.org/10.5802/ojmo.33, URL https://ojmo.centre-mersenne.org/articles/10.5802/ojmo.33/

Paradiso R, Georghiou A, Dabia S, Tönissen D (2022) Exact and approximate schemes for robust optimization problems with decision dependent information discovery. arXiv preprint arXiv:220804115

Ryu M, Jiang R (2019) Nurse staffing under absenteeism: A distributionally robust optimization approach. arXiv preprint arXiv:190909875

Shapiro A (2001) On duality theory of conic linear problems. Nonconvex Optimization and its Applications 57:135–155. https://doi.org/10.1007/978-1-4757-3403-4_7

Solak S, Clarke JPB, Johnson EL, Barnes ER (2010) Optimization of R&D project portfolios under endogenous uncertainty. European Journal of Operational Research 207(1):420–433. https://doi.org/10.1016/j.ejor.2010.04.032

Subramanyam A, Gounaris CE, Wiesemann W (2020) K-adaptability in two-stage mixed-integer robust optimization. Mathematical Programming Computation 12(2):193–224. https://doi.org/10.1007/s12532-019-00174-2

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

Vayanos P, Georghiou A, Yu H (2020) Robust optimization with decision-dependent information discovery. arXiv preprint arXiv:200408490

Vayanos P, Jin Q, Elissaios G (2022) ROC++: Robust optimization in C++. INFORMS Journal on Computing 34(6):2873–2888. https://doi.org/10.1287/ijoc.2022.1209

Wiesemann W, Kuhn D, Sim M (2014) Distributionally robust convex optimization. Operations Research 62(6):1358–1376. https://doi.org/10.1287/opre.2014.1314

Yu X, Shen S (2022) Multistage distributionally robust mixed-integer programming with decision-dependent moment-based ambiguity sets. Mathematical Programming 196(1):1025–1064. https://doi.org/10.1007/s10107-020-01580-4

Zhang J, Xu H, Zhang L (2016) Quantitative stability analysis for distributionally robust optimization with moment constraints. SIAM Journal on Optimization 26(3):1855–1882. https://doi.org/10.1137/15M1038529

Zhang L, Yang J, Gao R (2024) Optimal robust policy for feature-based newsvendor. Management Science 70(4):2315–2329

Zhang Q, Feng W (2020) A unified framework for adjustable robust optimization with endogenous uncertainty. AIChE Journal 66(12):e17047. https://doi.org/10.1002/aic.17047

A Omitted proofs

The following lemma is used to prove the strong duality result in Theorem 1.

Lemma A.1. *The point* $(1, \boldsymbol{c})$ *resides in the interior of the convex cone*

$$
\mathcal{V} = \left\{ (a, \boldsymbol{b}) \in \mathbb{R} \times \mathbb{R}^{N_g} : \exists \mu \in \mathcal{M}_+(\Xi) \text{ such that } \begin{array}{l} \int \mu(\mathrm{d}\boldsymbol{\xi}) = a, \\ \int \boldsymbol{g}(\boldsymbol{\xi})\mu(\mathrm{d}\boldsymbol{\xi}) \leq \boldsymbol{b} \end{array} \right\}. \tag{A.1}
$$

Proof. Let $\mathbb{B}_r(\boldsymbol{o})$ be the closed Euclidean ball of radius $r \geq 0$ centered at $\boldsymbol{o}$. Choose any point $(s,\ \boldsymbol{s}) \in \mathbb{B}_\kappa(1) \times \mathbb{B}_\kappa(\boldsymbol{c})$, where κ is a sufficiently small constant. Consider a point $\tilde{\boldsymbol{\xi}} \in \mathrm{int}\, \Xi$ such that $\boldsymbol{g}(\tilde{\boldsymbol{\xi}}) < \boldsymbol{c}$. The existence of such a point is guaranteed by the Slater condition. Because the function $\boldsymbol{g}$ is continuous, we can always find an interior point $\tilde{\boldsymbol{\xi}}$ of Ξ. Next, consider a scaled Dirac measure $s \cdot \delta_{\tilde{\boldsymbol{\xi}}/s}$ that places mass s at $\tilde{\boldsymbol{\xi}}/s$. By construction, this measure satisfies $\int s \cdot \delta_{\tilde{\boldsymbol{\xi}}/s}(\mathrm{d}\boldsymbol{\xi}) = s$. Moreover, for sufficiently small κ the measure is supported on Ξ (since $\tilde{\boldsymbol{\xi}} \in \mathrm{int}\, \Xi$) and satisfies $\int \boldsymbol{g}(\boldsymbol{\xi}) s \cdot \delta_{\tilde{\boldsymbol{\xi}}/s}(\mathrm{d}\boldsymbol{\xi}) \leq \boldsymbol{s}$ (since $\boldsymbol{g}(\tilde{\boldsymbol{\xi}}) < \boldsymbol{c}$ and the function $\boldsymbol{g}$ is continuous). $\qquad \square$

Proof of Theorem 1. For fixed $\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}(\cdot)$, we can express the objective function of (1) as the optimal value of the moment problem

$$
\begin{aligned}
\max \quad & \int_\Xi \boldsymbol{\xi}^\top \boldsymbol{C}\, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\, \boldsymbol{y}(\boldsymbol{\xi})\ \mu(\mathrm{d}\boldsymbol{\xi}) \\
\mathrm{s.\,t.} \quad & \mu \in \mathcal{M}_+(\mathbb{R}^{N_\xi}) \\
& \int_\Xi \mu(\mathrm{d}\boldsymbol{\xi}) = 1 \\
& \int_\Xi \boldsymbol{g}(\boldsymbol{\xi})\mu(\mathrm{d}\boldsymbol{\xi}) \leq \boldsymbol{c}.
\end{aligned} \tag{A.2}
$$

The dual problem is given by

$$
\begin{aligned}
\min \ & \phi + \boldsymbol{c}^\top \boldsymbol{\psi} \\
\text{s.t.} \ & \phi \in \mathbb{R}, \ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g} \\
& \phi + \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \geq \boldsymbol{\xi}^\top \boldsymbol{C} \, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \boldsymbol{y}(\boldsymbol{\xi}) \qquad \forall \boldsymbol{\xi} \in \Xi.
\end{aligned}
\tag{A.3}
$$

Strong duality holds by Lemma A.1 and Shapiro (2001, Proposition 3.4). Eliminating the variable ϕ and combining with the outer minimization for $(\boldsymbol{x}, \ \boldsymbol{w}, \ \boldsymbol{y}(\cdot))$, we get

$$
\begin{aligned}
\min \quad & \max_{\boldsymbol{\xi} \in \Xi} \ \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \boldsymbol{y}(\boldsymbol{\xi}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
\text{s.t.} \quad & \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W} \\
& \left. \begin{aligned} & \boldsymbol{y}(\boldsymbol{\xi}) \in \mathcal{Y} \\ & \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}(\boldsymbol{\xi}) \leq \boldsymbol{H}\boldsymbol{\xi} \end{aligned} \right\} \ \forall \boldsymbol{\xi} \in \Xi \\
& \boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}(\boldsymbol{\xi}') \quad \forall \boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi \ : \ \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}'.
\end{aligned}
\tag{A.4}
$$

The problem can be interpreted as a two-stage robust optimization with DDID, where the dual multipliers $\boldsymbol{\psi}$ constitute first-stage decisions.

We now show the equivalence between problem (4) and (A.4). We first show that problem (4) lower bounds (A.4). Let $(\boldsymbol{x}, \ \boldsymbol{w}, \ \boldsymbol{\psi}, \ \boldsymbol{y}(\cdot))$ be an optimal solution of problem (A.4). By construction, $(\boldsymbol{x}, \ \boldsymbol{w}, \ \boldsymbol{\psi})$ is also feasible in (4). For any fixed $\overline{\boldsymbol{\xi}}$, we set

$$
\boldsymbol{y}'(\overline{\boldsymbol{\xi}}) \in \operatorname*{argmin}_{\boldsymbol{y} \in \mathcal{Y}} \left\{ \begin{aligned} \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})} \ & \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \, \boldsymbol{y} - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\ \text{s.t.} \quad & \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y} \leq \boldsymbol{H}\boldsymbol{\xi} \ \ \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}}) \end{aligned} \right\}.
$$

49

It holds that

$$
\begin{aligned}
&\max_{\overline{\boldsymbol{\xi}} \in \Xi} \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})} \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}'(\overline{\boldsymbol{\xi}}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
&= \max_{\boldsymbol{\xi} \in \Xi} \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}'(\boldsymbol{\xi}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \quad\quad\quad (\text{A}.5) \\
&\leq \max_{\boldsymbol{\xi} \in \Xi} \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}(\boldsymbol{\xi}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi})
\end{aligned}
$$

Thus, we have shown that the optimal solution of (A.4) is feasible in (4) with an objective value no greater than the one attained in (A.4).

Next, we show conversely that problem (A.4) lower bounds (4). To this end, let $(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\psi})$ be an optimal solution in (4) and define $\boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}'(\boldsymbol{w} \circ \boldsymbol{\xi})$. By construction, the quadruple $(\boldsymbol{x},\ \boldsymbol{w},\ \boldsymbol{\psi},\ \boldsymbol{y}(\boldsymbol{\xi}))$ is feasible in (A.4) and attains the objective value

$$
\begin{aligned}
&\max_{\boldsymbol{\xi} \in \Xi} \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}(\boldsymbol{\xi}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
&= \max_{\overline{\boldsymbol{\xi}} \in \Xi} \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})} \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C} \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D} \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q} \boldsymbol{y}'(\overline{\boldsymbol{\xi}}) - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi})
\end{aligned}
\quad\quad (\text{A}.6)
$$

which is the same value attained in (4).

Thus, we have shown that the optimal objective values attained by the two problems coincide. And we can construct an optimal solution in problem (A.4) from an optimal solution in (4). This completes the proof. $\qquad\square$

Proof of Theorem 2. We linearize the vector $\boldsymbol{g}(\boldsymbol{\xi}^k)$, $\forall k \in \mathcal{K}$ of piecewise linear functions by introducing an epigraph uncertain parameter ζ_s^k for $g_s(\boldsymbol{\xi}^k)$, $\forall s \in \mathcal{S}, \forall k \in \mathcal{K}$,

and write the inner maximization problem in the following epigraph form

$$
\begin{aligned}
\max \quad & \tau \\
\text{s.t.} \quad & \tau \in \mathbb{R},\ \overline{\boldsymbol{\xi}} \in \mathbb{R}^{N_\xi},\ \boldsymbol{\xi}^k \in \mathbb{R}^{N_\xi}, k \in \mathcal{K} \\
& \tau \leq (\boldsymbol{\xi}^k)^\top \boldsymbol{C}\,\boldsymbol{x} + (\boldsymbol{\xi}^k)^\top \boldsymbol{D}\,\boldsymbol{w} + (\boldsymbol{\xi}^k)^\top \boldsymbol{Q}\,\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{\zeta}^k \quad \forall k \in \mathcal{K} \\
& \boldsymbol{A}\overline{\boldsymbol{\xi}} \leq \boldsymbol{b} \\
& \boldsymbol{A}\boldsymbol{\xi}^k \leq \boldsymbol{b} \quad \forall k \in \mathcal{K} \\
& \boldsymbol{w} \circ \boldsymbol{\xi}^k = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}} \quad \forall k \in \mathcal{K} \\
& \zeta_s^k \leq -\boldsymbol{g}_{s,t}^\top \boldsymbol{\xi}^k \qquad \forall s \in \mathcal{S},\ \forall t \in \mathcal{T},\ \forall k \in \mathcal{K}.
\end{aligned}
\tag{A.7}
$$

Problem (A.7) is a feasible LP problem, so strong duality implies that it has the same optimal value as its dual problem. Dualizing and grouping with the outer minimization yields problem (8) in Theorem 2. □

Proof of Theorem 3. Applying the uncertainty set lifting procedure in Vayanos et al (2020, Lemma 1) to problem (5), we can lift the space of the uncertainty set in the inner maximization, exchange it with the inner minimization, and finally get the equivalent min-max-min problem

$$
\begin{aligned}
\min \quad & \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\{\boldsymbol{\xi}^k\}_{k \in \mathcal{K}} \in \Xi^K(\boldsymbol{w})}
\left\{
\begin{aligned}
& \min_{k \in \mathcal{K}}\ (\boldsymbol{\xi}^k)^\top \boldsymbol{C}\boldsymbol{x} + (\boldsymbol{\xi}^k)^\top \boldsymbol{D}\boldsymbol{w} + (\boldsymbol{\xi}^k)^\top \boldsymbol{Q}\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}^k) \\
& \text{s.t.}\ \ \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi}
\end{aligned}
\right\} \\
\text{s.t.} \quad & \boldsymbol{\psi} \in \mathbb{R}_+^{N_g},\ \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{w} \in \mathcal{W},\ \boldsymbol{y}^k \in \mathcal{Y} \quad \forall k \in \mathcal{K},
\end{aligned}
\tag{A.8}
$$

where

$$
\Xi^K(\boldsymbol{w}) := \left\{ \{\boldsymbol{\xi}^k\}_{k \in \mathcal{K}} \in \Xi : \exists\,\overline{\boldsymbol{\xi}} \in \Xi \ \text{such that}\ \boldsymbol{\xi}^k \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})\ \forall k \in \mathcal{K} \right\}.
\tag{A.9}
$$

We then shift the second stage constraints into the uncertainty set and formulate the K-adaptability problem with constraint uncertainty (5) equivalently as

$$\min \quad \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\boldsymbol{\ell} \in \mathcal{L}} \max_{\{\boldsymbol{\xi}^k\}_{k \in \mathcal{K}} \in \Xi^K(\boldsymbol{w},\boldsymbol{\ell})} \min_{\substack{k \in K: \\ \boldsymbol{\ell}_k = 0}} \left\{ (\boldsymbol{\xi}^k)^\top \boldsymbol{C}\,\boldsymbol{x} + (\boldsymbol{\xi}^k)^\top \boldsymbol{D}\,\boldsymbol{w} + (\boldsymbol{\xi}^k)^\top \boldsymbol{Q}\,\boldsymbol{y} \right.$$
$$\left. - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}^k) \right\}$$
$$\text{s.t.} \quad \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W}, \ \boldsymbol{y}^k \in \mathcal{Y} \quad \forall k \in \mathcal{K},$$
$$\tag{A.10}$$

where $\mathcal{L} := \{1, \ldots, L\}^K$, L is the number of second-stage constraints with uncertainty in problem (5). The uncertainty sets $\Xi^K(\boldsymbol{w},\boldsymbol{\ell})$, $\boldsymbol{\ell} \in \mathcal{L}$, are defined as

$$\Xi^K(\boldsymbol{w},\boldsymbol{\ell}) := \left\{ \left\{\boldsymbol{\xi}^k\right\}_{k \in \mathcal{K}} \in \Xi^K : \begin{array}{c} \boldsymbol{w} \circ \boldsymbol{\xi}^k = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}}, \ \ \forall k \in \mathcal{K} \text{ for some } \overline{\boldsymbol{\xi}} \in \Xi \\[4pt] \boldsymbol{T}(\boldsymbol{\xi}^k)\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi}^k)\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^k)\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi}^k \\[4pt] \forall k \in \mathcal{K} : \boldsymbol{\ell}_k = 0 \\[4pt] \left[\boldsymbol{T}(\boldsymbol{\xi}^k)\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi}^k)\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^k)\boldsymbol{y}^k\right]_{\boldsymbol{\ell}_k} > \left[\boldsymbol{H}\boldsymbol{\xi}^k\right]_{\boldsymbol{\ell}_k} \\[4pt] \forall k \in \mathcal{K} : \boldsymbol{\ell}_k \neq 0 \end{array} \right\}.$$
$$\tag{A.11}$$

The components of vector $\boldsymbol{\ell} \in \mathcal{L}$ encode which policies are robust feasible for the parameter realizations $\left\{\boldsymbol{\xi}^k\right\}_{k \in \mathcal{K}}$. Policy $\boldsymbol{y}^k$ is robust feasible in problem (5) if $\ell_k = 0$. If $\ell_k \neq 0$, then the ℓ_k-th constraint in problem (5) is violated for some $\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})$.

Problem (A.10) involves many open uncertainty sets. Thus, we employ inner approximations $\Xi_\epsilon^K(\boldsymbol{w},\boldsymbol{\ell})$ of the uncertainty sets (A.11) that are parameterized by scalar ϵ. The approximate problem can be written as

$$\min \quad \boldsymbol{c}^\top \boldsymbol{\psi} + \max_{\boldsymbol{\ell} \in \mathcal{L}} \max_{\{\boldsymbol{\xi}^k\}_{k \in \mathcal{K}} \in \Xi_\epsilon^K(\boldsymbol{w},\boldsymbol{\ell})} \min_{\substack{k \in K: \\ \boldsymbol{\ell}_k = 0}} \left\{ (\boldsymbol{\xi}^k)^\top \boldsymbol{C}\,\boldsymbol{x} + (\boldsymbol{\xi}^k)^\top \boldsymbol{D}\,\boldsymbol{w} + (\boldsymbol{\xi}^k)^\top \boldsymbol{Q}\,\boldsymbol{y} \right.$$
$$\left. - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}^k) \right\}$$
$$\text{s.t.} \quad \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W}, \ \boldsymbol{y}^k \in \mathcal{Y} \quad \forall k \in \mathcal{K},$$
$$\tag{A.12}$$

where

$$
\Xi_{\epsilon}^{K}(\boldsymbol{w}, \boldsymbol{\ell}) := \left\{ \left\{\boldsymbol{\xi}^{k}\right\}_{k \in \mathcal{K}} \in \Xi^{K} : \begin{array}{r} \boldsymbol{w} \circ \boldsymbol{\xi}^{k} = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}}, \ \ \forall k \in \mathcal{K} : \exists \overline{\boldsymbol{\xi}} \in \Xi \\[2mm] \boldsymbol{T}(\boldsymbol{\xi}^{k})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi}^{k})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^{k})\boldsymbol{y}^{k} \leq \boldsymbol{H}\boldsymbol{\xi}^{k} \\[2mm] \forall k \in \mathcal{K} : \boldsymbol{\ell}_{k} = 0 \\[2mm] \left[\boldsymbol{T}(\boldsymbol{\xi}^{k})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi}^{k})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^{k})\boldsymbol{y}^{k}\right]_{\boldsymbol{\ell}_{k}} \\[2mm] \geq \left[\boldsymbol{H}\boldsymbol{\xi}^{k}\right]_{\boldsymbol{\ell}_{k}} + \epsilon \\[2mm] \forall k \in \mathcal{K} : \boldsymbol{\ell}_{k} \neq 0 \end{array} \right\}
$$

We next reformulate the approximate problem (A.12) as a mixed binary bilinear program. The outer maximization of the approximation problem (A.12) is identical to

$$
\max_{\boldsymbol{\ell} \in \mathcal{L}} \max_{\{\boldsymbol{\xi}^{k}\}_{k \in \mathcal{K}} \in \Xi_{\epsilon}^{K}(\boldsymbol{w}, \boldsymbol{\ell})} \min_{\boldsymbol{\delta} \in \Delta_{K}(\boldsymbol{\ell})} \left\{ \sum_{k \in \mathcal{K}} \boldsymbol{\delta}_{k} \left[\left(\boldsymbol{\xi}^{k}\right)^{\top} \boldsymbol{C}\boldsymbol{x} + \left(\boldsymbol{\xi}^{k}\right)^{\top} \boldsymbol{D}\boldsymbol{w} + \left(\boldsymbol{\xi}^{k}\right)^{\top} \boldsymbol{Q}\boldsymbol{y}^{k} \right. \right.
$$
$$
\left. \left. - \boldsymbol{\psi}^{\top} \boldsymbol{g}(\boldsymbol{\xi}) \right] \right\},
$$
(A.13)

where $\Delta_{K}(\boldsymbol{\ell}) := \left\{\boldsymbol{\delta} \in \mathbb{R}_{+}^{K} : \mathbf{e}^{\top}\boldsymbol{\delta} = 1, \boldsymbol{\delta}_{k} = 0 \quad \forall k \in \mathcal{K} : \ell_{k} \neq 0\right\}$. If $\Xi_{\epsilon}^{K}(\boldsymbol{w}, \boldsymbol{\ell}) = \emptyset$ for all $\boldsymbol{\ell} \in \mathcal{L}_{+}$, apply the min-max theorem, the problem is equivalent to

$$
\min_{\boldsymbol{\delta}(\boldsymbol{\ell}) \in \Delta_{K}(\boldsymbol{\ell}), \ \boldsymbol{\ell} \in \partial\mathcal{L}} \max_{\{\xi^{k}\}_{k \in \mathcal{K}} \in \Xi_{\epsilon}^{K}(w, \boldsymbol{\ell})} \left\{ \sum_{k \in \mathcal{K}} \boldsymbol{\delta}_{k} \left[\left(\boldsymbol{\xi}^{k}\right)^{\top} \boldsymbol{C}\boldsymbol{x} + \left(\boldsymbol{\xi}^{k}\right)^{\top} \boldsymbol{D}\boldsymbol{w} + \left(\boldsymbol{\xi}^{k}\right)^{\top} \boldsymbol{Q}\boldsymbol{y}^{k} \right. \right.
$$
$$
\left. \left. - \boldsymbol{\psi}^{\top} \boldsymbol{g}(\boldsymbol{\xi}) \right] \right\}.
$$
(A.14)

Use an epigraph formulation, we conclude that (A.12) is equivalent to the following formulation

$$
\begin{aligned}
\max \quad & \tau \\
\text{s.t.} \quad & x \in \mathcal{X}, \ \boldsymbol{w} \in \mathcal{W}, \ \boldsymbol{y}^k \in \mathcal{Y}, \ k \in \mathcal{K}, \ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g} \\
& \tau \in \mathbb{R}, \ \boldsymbol{\delta}(\boldsymbol{\ell}) \in \boldsymbol{\Delta}_K(\boldsymbol{\ell}), \ \boldsymbol{\ell} \in \partial \mathcal{L} \\
& \tau \geq \textstyle\sum_{k \in \mathcal{K}} \boldsymbol{\delta}_k(\boldsymbol{\ell}) \left[\left(\boldsymbol{\xi}^k\right)^\top \boldsymbol{C} x \ + \left(\boldsymbol{\xi}^k\right)^\top \boldsymbol{D} \boldsymbol{w} + \left(\boldsymbol{\xi}^k\right)^\top \boldsymbol{Q} \boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \right] \\
& \hspace{4cm} \forall \boldsymbol{\ell} \in \partial \mathcal{L}, \left\{ \boldsymbol{\xi}^k \right\}_{k \in \mathcal{K}} \in \Xi_\epsilon^K(\boldsymbol{w}, \boldsymbol{\ell}) \\
& \Xi_\epsilon^K(\boldsymbol{w}, \boldsymbol{\ell}) = \emptyset \quad \forall \boldsymbol{\ell} \in \mathcal{L}_+.
\end{aligned}
\tag{A.15}
$$

The semi-infinite constraint associated with $\boldsymbol{\ell} \in \partial \mathcal{L}$ is satisfied if and only if the optimal value of

$$
\begin{aligned}
\max \ & \textstyle\sum_{k \in \mathcal{K}} \boldsymbol{\delta}_k(\boldsymbol{\ell}) \left[\left(\boldsymbol{\xi}^k\right)^\top \boldsymbol{C} x + \left(\boldsymbol{\xi}^k\right)^\top \boldsymbol{D} \boldsymbol{w} + \left(\boldsymbol{\xi}^k\right)^\top \boldsymbol{Q} \boldsymbol{y}^k + \boldsymbol{\psi}^\top \boldsymbol{\zeta}^k \right] \\
\text{s.t.} \ & \overline{\boldsymbol{\xi}} \in \mathbb{R}^{N_\xi}, \boldsymbol{\xi}^k \in \mathbb{R}^{N_\xi}, k \in \mathcal{K} \\
& \boldsymbol{A} \overline{\boldsymbol{\xi}} \leq \boldsymbol{b} \\
& \boldsymbol{A} \boldsymbol{\xi}^k \leq \boldsymbol{b} \quad \forall k \in \mathcal{K} \\
& \boldsymbol{T}(\boldsymbol{\xi}^k) x + \boldsymbol{V}(\boldsymbol{\xi}^k) \boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^k) \boldsymbol{y}^k \leq \boldsymbol{H} \boldsymbol{\xi}^k \quad \forall k \in \mathcal{K} : \ell_k = 0 \\
& \left[\boldsymbol{T}(\boldsymbol{\xi}^k) x + \boldsymbol{V}(\boldsymbol{\xi}^k) \boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^k) \boldsymbol{y}^k \right]_{\ell_k} \geq \left[\boldsymbol{H} \boldsymbol{\xi}^k \right]_{\ell_k} + \epsilon \quad \forall k \in \mathcal{K} : \ell_k \neq 0 \\
& \boldsymbol{w} \circ \boldsymbol{\xi}^k = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}} \quad \forall k \in \mathcal{K} \\
& \boldsymbol{\zeta}_s^k \leq -\boldsymbol{g}_{s,t}^\top \boldsymbol{\xi}^k \quad \forall s \in \mathcal{S}, \forall t \in \mathcal{T}, \ \forall k \in \mathcal{K}
\end{aligned}
\tag{A.16}
$$

does not exceed τ.

The last constraint in (A.15) is satisfied for $\boldsymbol{\ell} \in \mathcal{L}$ whenever the linear program problem

$$
\begin{aligned}
\max \quad & 0 \\
\text{s.t.} \quad & \overline{\boldsymbol{\xi}} \in \mathbb{R}^{N_\xi}, \ \boldsymbol{\xi}^k \in \mathbb{R}^{N_\xi}, k \in \mathcal{K} \\
& \boldsymbol{A}\overline{\boldsymbol{\xi}} \leq \boldsymbol{b} \\
& \boldsymbol{A}\boldsymbol{\xi}^k \leq \boldsymbol{b} && \forall k \in \mathcal{K} \\
& \boldsymbol{w} \circ \boldsymbol{\xi}^k = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}} && \forall k \in \mathcal{K} \\
& \left[\boldsymbol{T}(\boldsymbol{\xi}^k)\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi}^k)\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi}^k)\boldsymbol{y}^k\right]_{\ell_k} \geq \left[\boldsymbol{H}\boldsymbol{\xi}^k\right]_{\ell_k} + \epsilon \ \forall k \in \mathcal{K}
\end{aligned}
\tag{A.17}
$$

is infeasible. Strong duality holds for both (A.16) and (A.17). After taking dualization of (A.16), (A.17),, and combining the dual problem with the outer minimization

problem, we can get the following problem

$$\min \; \boldsymbol{c}^\top \boldsymbol{\psi} + \tau$$

$$\text{s.t.} \; x \in \mathcal{X}, \; \boldsymbol{w} \in \mathcal{W}, \; \boldsymbol{y}^k \in \mathcal{Y}, \; k \in \mathcal{K}, \; \tau \in \mathbb{R}, \; \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}$$

$$\boldsymbol{\alpha}(\boldsymbol{\ell}) \in \mathbb{R}_+^R, \; \boldsymbol{\alpha}^k(\boldsymbol{\ell}) \in \mathbb{R}_+^R, \; k \in \mathcal{K}, \boldsymbol{\eta}^k(\boldsymbol{\ell}) \in \mathbb{R}^{N_\xi}, \; k \in \mathcal{K}$$

$$\boldsymbol{\Lambda}^k(l) \in \mathbb{R}_+^{N_g \times T} \; k \in \mathcal{K}, \; \boldsymbol{\beta}^k(\boldsymbol{\ell}) \in \mathbb{R}_+^L, \; k \in \mathcal{K}$$

$$\boldsymbol{\delta}(\boldsymbol{\ell}) \in \Delta_K(\boldsymbol{\ell}), \; \boldsymbol{\gamma}(\boldsymbol{\ell}) \in \mathbb{R}_+^K$$

$$\boldsymbol{A}^\top \boldsymbol{\alpha}(\boldsymbol{\ell}) = \sum_{k \in \mathcal{K}} \boldsymbol{w} \circ \boldsymbol{\eta}^k(\boldsymbol{\ell})$$

$$\boldsymbol{\delta}_k(\boldsymbol{\ell}) \boldsymbol{\psi} = \boldsymbol{\Lambda}^k(\boldsymbol{\ell}) \boldsymbol{e}$$

$$\boldsymbol{\delta}_k(\boldsymbol{\ell}) \left[\boldsymbol{Cx} + \boldsymbol{Dw} + \boldsymbol{Qy}^k \right] - \sum_{l=1}^{L} \left(\boldsymbol{T}_l x + \boldsymbol{V}_l \boldsymbol{w} + \boldsymbol{W}_l \boldsymbol{y}^k \right) \boldsymbol{\beta}_l^k(\boldsymbol{\ell})$$
$$= \boldsymbol{A}^\top \boldsymbol{\alpha}^k(\boldsymbol{\ell}) - \boldsymbol{H}^\top \boldsymbol{\beta}^k(\boldsymbol{\ell}) + \boldsymbol{w} \circ \boldsymbol{\eta}^k(\boldsymbol{\ell}) + \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}} \boldsymbol{\Lambda}_{st}(\boldsymbol{\ell}) \boldsymbol{g}_{st}$$
$$\forall k \in \mathcal{K} : \boldsymbol{\ell}_k = 0$$

$$\boldsymbol{\delta}_k(\boldsymbol{\ell}) \left[\boldsymbol{Cx} + \boldsymbol{Dw} + \boldsymbol{Qy}^k \right] + \left(\boldsymbol{T}_{\boldsymbol{\ell}_k} x + \boldsymbol{V}_{\boldsymbol{\ell}_k} \boldsymbol{w} + \boldsymbol{W}_{\boldsymbol{\ell}_k} \boldsymbol{y}^k \right) \boldsymbol{\gamma}_k(\boldsymbol{\ell})$$
$$= \boldsymbol{A}^\top \boldsymbol{\alpha}^k(\boldsymbol{\ell}) + [\boldsymbol{H}]_{\boldsymbol{\ell}_k} \boldsymbol{\gamma}_k(\boldsymbol{\ell}) + \boldsymbol{w} \circ \boldsymbol{\eta}^k(\boldsymbol{\ell}) + \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}} \boldsymbol{\Lambda}_{st}(\boldsymbol{\ell}) \boldsymbol{g}_{st}$$
$$\forall k \in \mathcal{K} : \boldsymbol{\ell}_k \neq 0$$

$$\tau \geq \boldsymbol{b}^\top \left(\boldsymbol{\alpha}(\boldsymbol{\ell}) + \sum_{k \in \mathcal{K}} \boldsymbol{\alpha}^k(\boldsymbol{\ell}) \right) - \epsilon \sum_{\substack{k \in \mathcal{K}: \\ \boldsymbol{\ell}_k \neq 0}} \boldsymbol{\gamma}_k(\boldsymbol{\ell})$$

$$\left. \right\} \forall \boldsymbol{\ell} \in \partial\mathcal{L}$$

$$\boldsymbol{\phi}(\boldsymbol{\ell}) \in \mathbb{R}_+^R, \; \boldsymbol{\phi}^k(\boldsymbol{\ell}) \in \mathbb{R}_+^R, \; k \in \mathcal{K}, \; \boldsymbol{\rho}(\boldsymbol{\ell}) \in \mathbb{R}_+^k, \; \boldsymbol{\chi}^k(\boldsymbol{\ell}) \in \mathbb{R}^{N_\xi}, \; k \in \mathcal{K}$$

$$\boldsymbol{A}^\top \boldsymbol{\phi}(\boldsymbol{\ell}) = \sum_{k \in \mathcal{K}} \boldsymbol{w} \circ \boldsymbol{\chi}^k(\boldsymbol{\ell})$$

$$\boldsymbol{A}^\top \boldsymbol{\phi}^k(\boldsymbol{\ell}) + [\boldsymbol{H}]_{\boldsymbol{\ell}_k} \boldsymbol{\rho}_k(\boldsymbol{\ell}) + \boldsymbol{w} \circ \boldsymbol{\chi}^k(\boldsymbol{\ell})$$
$$= \left(\boldsymbol{T}_{\boldsymbol{\ell}_k} x + \boldsymbol{V}_{\boldsymbol{\ell}_k} \boldsymbol{w} + \boldsymbol{W}_{\boldsymbol{\ell}_k} \boldsymbol{y}^k \right) \boldsymbol{\rho}_k(\boldsymbol{\ell})$$
$$\forall k \in \mathcal{K}$$

$$\boldsymbol{b}^\top \left(\boldsymbol{\phi}(\boldsymbol{\ell}) + \sum_{k \in \mathcal{K}} \boldsymbol{\phi}^k(\boldsymbol{\ell}) \right) - \epsilon \sum_{k \in \mathcal{K}} \boldsymbol{\rho}_k(\boldsymbol{\ell}) \leq -1$$

$$\left. \right\} \forall \boldsymbol{\ell} \in \mathcal{L}_+,$$

$$\text{(A.18)}$$

where $\mathcal{L} := \{1, \ldots, L\}^K$, $\Delta_K(\boldsymbol{\ell}) := \left\{ \boldsymbol{\delta} \in \mathbb{R}_+^K : \boldsymbol{e}^\top \boldsymbol{\delta} = 1, \boldsymbol{\delta}_k = 0 \quad \forall k \in \mathcal{K} : \boldsymbol{\ell}_k \neq 0 \right\}$, $\partial\mathcal{L} := \{ \boldsymbol{\ell} \in \mathcal{L} : \boldsymbol{\ell} \not> \boldsymbol{0} \}$ and $\mathcal{L}^+ := \{ \boldsymbol{\ell} \in \mathcal{L} : \boldsymbol{\ell} > \boldsymbol{0} \}$. Matrices $\boldsymbol{T}_l$, $\boldsymbol{V}_l$, $\boldsymbol{W}_l$ are the

coefficient matrices in the lth constraint, in other words, $[\boldsymbol{T}(\boldsymbol{\xi})]_l = \boldsymbol{\xi}^\top \boldsymbol{T}_l$, $[\boldsymbol{V}(\boldsymbol{\xi})]_l = \boldsymbol{\xi}^\top \boldsymbol{V}_l$, $[\boldsymbol{W}(\boldsymbol{\xi})]_l = \boldsymbol{\xi}^\top \boldsymbol{W}_l$. □

Proof of Proposition 4. For any feasible solution $(\boldsymbol{\phi}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}})$ in $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$, we show that it is also feasible in $(\mathcal{IP}_{\boldsymbol{w}})$ with the same or lower objective value. Let $\{\Xi^k\}_{k\in\mathcal{K}}$ be the corresponding optimal partition. Since $\{\Xi^k\}_{k\in\mathcal{K}}$ exhausts all points in Ξ, we see that $(\boldsymbol{\phi}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}})$ is indeed feasible in $(\mathcal{IP}_{\boldsymbol{w}})$ because for any $\overline{\boldsymbol{\xi}} \in \Xi$, there must exist $k \in \mathcal{K}$ such that $\overline{\boldsymbol{\xi}} \in \Xi^k$ and, therefore, $\boldsymbol{y}^k$ is feasible in the inner minimization problem. Next, we show that the optimal objective value of $(\mathcal{IP}_{\boldsymbol{w}})$ is at most θ by bounding it from above, written as

$$
\begin{aligned}
&\max_{\overline{\boldsymbol{\xi}}\in\Xi}
\left\{
\begin{aligned}
&\min_{k\in\mathcal{K}} \; \max_{\boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})} \; \boldsymbol{c}^\top\boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
&\text{s.t.} \quad \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \le \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})
\end{aligned}
\right\} \\
&= \max_{j\in\mathcal{K}} \max_{\overline{\boldsymbol{\xi}}\in\Xi^j}
\left\{
\begin{aligned}
&\min_{k\in\mathcal{K}} \; \max_{\boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})} \; \boldsymbol{c}^\top\boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
&\text{s.t.} \quad \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \le \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})
\end{aligned}
\right\} \\
&\le \max_{j\in\mathcal{K}} \max_{\overline{\boldsymbol{\xi}}\in\Xi^j} \max_{\boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})} \; \boldsymbol{c}^\top\boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^j - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
&\le \theta.
\end{aligned}
\tag{A.19}
$$

Here, the first inequality holds because we do not optimize for the best index k but instead simply set it to the index j from the outer maximization.

Conversely, we show that any solution $(\boldsymbol{\phi}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}})$ feasible in $(\mathcal{IP}_{\boldsymbol{w}})$ is also feasible in $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$ with the same or lower objective value. To this end, we construct a partition feasible for $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$, written as

$$
\Xi^j = \left\{
\boldsymbol{\xi} \in \Xi :
\begin{aligned}
&j \in \operatorname*{argmin}_{k\in\mathcal{K}} \; \max_{\boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})} \; \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
&\text{s.t.} \quad \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \le \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi}\in\Xi(\boldsymbol{w},\overline{\boldsymbol{\xi}})
\end{aligned}
\right\} \\
\forall j \in \mathcal{K}.
$$

The set Ξ^j contains all points $\boldsymbol{\xi} \in \Xi$ for which the policy $\boldsymbol{y}^j$ is optimal in the second-stage subproblem. Under this partition, one can verify that the solution $(\boldsymbol{\phi}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k \in \mathcal{K}})$ is feasible in $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$ with the same objective value. Since the partition can still be optimized, we conclude that the optimal value of problem $(\mathcal{IP}_{\boldsymbol{w}}(\Xi))$ is at most equal to that of problem $(\mathcal{IP}_{\boldsymbol{w}})$, which completes the proof. $\qquad\square$

Theorem A.1. *Problem* $(\mathcal{IP}_{\boldsymbol{w}}^{\mathrm{out}})$ *is equivalent to the following MIO problem*

$$
\begin{aligned}
\max \quad & z \\
\mathrm{s.\,t.} \quad & z \in \mathbb{R},\ \overline{\boldsymbol{\xi}} \in \mathbb{R}^{N_\xi},\ \boldsymbol{\xi}^k \in \mathbb{R}^{N_\xi}, \forall k \in \mathcal{K},\ z_{kl} \in \{0,1\}, (k,l) \in \mathcal{K} \times \{0,1,\ldots,L\} \\
& A\overline{\boldsymbol{\xi}} \leq \boldsymbol{b} \\
& A\boldsymbol{\xi}^k \leq \boldsymbol{b} \\
& \boldsymbol{w}' \circ \boldsymbol{\xi}^k = \boldsymbol{w}' \circ \overline{\boldsymbol{\xi}} \\
& z_{k0} = 1 \ \Rightarrow\ z \leq \boldsymbol{c}^\top \boldsymbol{\psi}' + {\boldsymbol{\xi}^k}^\top \boldsymbol{C}\, \boldsymbol{x}' + {\boldsymbol{\xi}^k}^\top \boldsymbol{D}\, \boldsymbol{w}' + {\boldsymbol{\xi}^k}^\top \boldsymbol{Q}\, \boldsymbol{y}^{k'} \\
& \hspace{5cm} -{\boldsymbol{\psi}'}^\top \boldsymbol{g}(\boldsymbol{\xi}^k) - \theta \\
& z_{kl} = 1 \ \Rightarrow\ z \leq {\boldsymbol{\xi}^k}^\top \boldsymbol{T}_l \boldsymbol{x}' + {\boldsymbol{\xi}^k}^\top \boldsymbol{V}_l \boldsymbol{w}' + {\boldsymbol{\xi}^k}^\top \boldsymbol{W}_l \boldsymbol{y}^{k'} - \boldsymbol{H}_l \boldsymbol{\xi} \\
& \hspace{6cm} \forall l \in \{1,\ldots,L\} \\
& \textstyle\sum_{l=0}^{L} z_{kl} = 1
\end{aligned} \qquad \left. \vphantom{\begin{aligned}&\\&\\&\\&\\&\\&\\&\\&\\&\\&\end{aligned}} \right\} \quad \forall k \in \mathcal{K}.
$$

$$\tag{A.20}$$

Proof. Following the same uncertainty lifting procedure that we used in the proof of Theorem 3, we can write $(\mathcal{IP}_{\boldsymbol{w}}^{\mathrm{out}})$ equivalently as

$$
\begin{aligned}
\max_{\{\boldsymbol{\xi}^k\}_{k \in \mathcal{K}} \in \Xi^K(\boldsymbol{w}')} \ \min_{k \in \mathcal{K}} \ \max \Big\{ & \boldsymbol{c}^\top \boldsymbol{\psi}' + (\boldsymbol{\xi}^k)^\top \boldsymbol{C}\, \boldsymbol{x}' + (\boldsymbol{\xi}^k)^\top \boldsymbol{D}\, \boldsymbol{w}' + (\boldsymbol{\xi}^k)^\top \boldsymbol{Q}\, \boldsymbol{y}^{k'} \\
& \hspace{3cm} -{\boldsymbol{\psi}'}^\top \boldsymbol{g}(\boldsymbol{\xi}^k) - \theta', \quad \text{(A.21)} \\
& \max_{l \in \{1,\ldots,L\}} \Big\{ \boldsymbol{\xi}^\top \boldsymbol{T}_l \boldsymbol{x}' + \boldsymbol{\xi}^\top \boldsymbol{V}_l \boldsymbol{w}' + \boldsymbol{\xi}^\top \boldsymbol{W}_l \boldsymbol{y}^{k'} - \boldsymbol{H}_l \boldsymbol{\xi} \Big\} \Big\},
\end{aligned}
$$

where $\Xi^K(\boldsymbol{w}')$ is defined as in (A.9).

Then, using an epigraph to represent the minimization problem over k and assigning indicator variables for each objective of the inner maximization, we can write problem (A.21) as (A.20). □

Theorem A.2. *Problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ admits an equivalent deterministic counterpart whose size grows linearly with the cardinality of $\widehat{\Xi}$.*

Proof. For each $k \in \mathcal{K}$, consider any scenario $\overline{\boldsymbol{\xi}} \in \widehat{\Xi}^k$. The maximization problem on the right-hand side of the first constraint of problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ can be written as

$$
\begin{aligned}
\max \quad & \boldsymbol{\xi}^\top \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\,\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\,\boldsymbol{y}^k + \boldsymbol{\psi}^\top \boldsymbol{\zeta} \\
\text{s.t.} \quad & \boldsymbol{\xi} \in \mathbb{R}^{N_\xi} \\
& \boldsymbol{A}\boldsymbol{\xi} \leq \boldsymbol{b} \\
& \boldsymbol{w} \circ \boldsymbol{\xi}^k = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}} \\
& \zeta_s \leq -\boldsymbol{g}_{s,t}^\top \boldsymbol{\xi}^k \qquad \forall s \in \mathcal{S}, \forall t \in \mathcal{T}.
\end{aligned}
\tag{A.22}
$$

This linear program admits a strong dual given by the minimization problem

$$
\begin{aligned}
\min \quad & \boldsymbol{b}^\top \boldsymbol{\beta}(\overline{\boldsymbol{\xi}}) + \left(\boldsymbol{w} \circ \overline{\boldsymbol{\xi}}\right)^\top \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}) \\
\text{s.t.} \quad & \boldsymbol{\beta}(\overline{\boldsymbol{\xi}}) \in \mathbb{R}^R_+,\ \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}) \in \mathbb{R}^{N_\xi},\ \delta(\overline{\boldsymbol{\xi}})_{s,t} \in \mathbb{R}_+,\ \forall s \in \mathcal{S},\ \forall t \in \mathcal{T} \\
& \boldsymbol{A}^\top \boldsymbol{\beta}(\overline{\boldsymbol{\xi}}) + \sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}} \delta(\overline{\boldsymbol{\xi}})_{s,t}\boldsymbol{g}_{s,t} + \boldsymbol{w} \circ \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}) = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{w} + \boldsymbol{Q}\boldsymbol{y}^k \\
& \sum_{t\in\mathcal{T}} \boldsymbol{\delta}(\overline{\boldsymbol{\xi}})_t = \boldsymbol{\psi}.
\end{aligned}
\tag{A.23}
$$

Similarly, in the second set of constraints, for any scenario $\overline{\boldsymbol{\xi}} \in \Xi^k$, the maximization problem on the left-hand side of the lth constraint can be written as

$$
\begin{aligned}
\max \quad & \boldsymbol{\xi}^\top \boldsymbol{T}_l\boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{V}_l\boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{W}_l\boldsymbol{y}^k - \boldsymbol{H}_l\boldsymbol{\xi} \\
\text{s.t.} \quad & \boldsymbol{\xi} \in \mathbb{R}^{N_\xi} \\
& \boldsymbol{A}\boldsymbol{\xi} \leq \boldsymbol{b} \\
& \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \overline{\boldsymbol{\xi}}.
\end{aligned}
\tag{A.24}
$$

59

By taking dual of problem (A.24), we obtain a minimization problem whose optimal value coincides with that of the primal being expressible as

$$
\begin{aligned}
\min \quad & \boldsymbol{b}^\top \boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l + \left(\boldsymbol{w} \circ \overline{\boldsymbol{\xi}}\right)^\top \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l \\
\text{s.t.} \quad & \boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l \in \mathbb{R}_+^R, \ \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l \in \mathbb{R}^{N_\xi} \\
& \boldsymbol{A}^\top \boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l + \boldsymbol{w} \circ \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l = \boldsymbol{T}_l \boldsymbol{x} + \boldsymbol{V}_l \boldsymbol{w} + \boldsymbol{W}_l \boldsymbol{y}^k - \boldsymbol{H}_l^\top.
\end{aligned} \tag{A.25}
$$

Therefore, replacing the constraints in $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ with the respective dual reformulations (A.23) and (A.25), we can write $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k\in\mathcal{K}}))$ as the deterministic problem

$$
\begin{aligned}
\min \quad & \theta \\
\text{s.t.} \quad & \theta \in \mathbb{R}, \ \boldsymbol{\psi} \in \mathbb{R}_+^{N_g}, \ \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{y}^k \in \mathcal{Y} \quad \forall k \in \mathcal{K} \\
& \left. \begin{aligned}
& \boldsymbol{\beta}(\overline{\boldsymbol{\xi}}) \in \mathbb{R}_+^R, \ \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}) \in \mathbb{R}^{N_\xi}, \ \delta(\overline{\boldsymbol{\xi}})_{s,t} \in \mathbb{R}_+, \ \forall s \in \mathcal{S}, \ \forall t \in \mathcal{T} \\
& \boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l \in \mathbb{R}_+^R, \ \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l \in \mathbb{R}^{N_\xi}, \ \forall l \in \mathcal{L} \\
& \theta \geq \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{b}^\top \boldsymbol{\beta}(\overline{\boldsymbol{\xi}}) + \left(\boldsymbol{w} \circ \overline{\boldsymbol{\xi}}\right)^\top \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}) \\
& \boldsymbol{A}^\top \boldsymbol{\beta}(\overline{\boldsymbol{\xi}}) + \sum_{s\in\mathcal{S}} \sum_{t\in\mathcal{T}} \delta(\overline{\boldsymbol{\xi}})_{s,t} \boldsymbol{g}_{s,t} + \boldsymbol{w} \circ \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}) \\
& \qquad\qquad\qquad = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{w} + \boldsymbol{Q}\boldsymbol{y}^k \\
& \sum_{t\in\mathcal{T}} \boldsymbol{\delta}(\overline{\boldsymbol{\xi}})_t = \boldsymbol{\psi} \\
& \left. \begin{aligned} & \boldsymbol{b}^\top \boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l + \left(\boldsymbol{w} \circ \overline{\boldsymbol{\xi}}\right)^\top \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l \leq 0 \\ & \boldsymbol{A}^\top \boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l + \boldsymbol{w} \circ \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l = \boldsymbol{T}_l \boldsymbol{x} + \boldsymbol{V}_l \boldsymbol{w} + \boldsymbol{W}_l \boldsymbol{y}^k - \boldsymbol{H}_l^\top \end{aligned} \right\} \forall l \in \mathcal{L}
\end{aligned} \right\} \begin{aligned} & \forall \overline{\boldsymbol{\xi}} \in \widehat{\Xi}^k, \\ & \forall k \in \mathcal{K}, \end{aligned}
\end{aligned} \tag{A.26}
$$

where $\bigcup_{k\in\mathcal{K}} \widehat{\Xi}^k = \widehat{\Xi}$. Note that whenever a new scenario $\overline{\boldsymbol{\xi}}$ is added to the set $\widehat{\Xi}$, we need to add a set of dual variables $(\boldsymbol{\beta}(\overline{\boldsymbol{\xi}}), \boldsymbol{\gamma}(\overline{\boldsymbol{\xi}}), \boldsymbol{\delta}(\overline{\boldsymbol{\xi}}), \{\boldsymbol{\alpha}(\overline{\boldsymbol{\xi}})_l, \boldsymbol{\eta}(\overline{\boldsymbol{\xi}})_l\}_{l\in\mathcal{L}})$ into the deterministic problem (A.26). $\qquad\square$

Lemma A.2. *If problem* $(\mathcal{IP}_{\boldsymbol{w}})$ *is feasible, then* $\boldsymbol{\psi}$ *admits a finite optimal solution.*

Proof. Consider the equivalent reformulation (A.4) of problem (1) in the proof of Theorem 1. By Lemma A.1 and Shapiro (2001, Proposition 3.4), for any feasible $\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}(\boldsymbol{\xi})$, an optimal solution $\boldsymbol{\psi}$ is attained by the dual problem. This claim also holds when we restrict $\boldsymbol{y}(\boldsymbol{\xi})$ in (A.4) to feasible K-adaptable policies. The claim then follows since problem $(\mathcal{IP}_{\boldsymbol{w}})$ constitutes an equivalent K-adaptable reformulation of (A.4). $\qquad\square$

Lemma A.3. *The cutting-plane algorithm for solving the scenario-based problem $(\mathcal{IP}'_{\boldsymbol{w}}(\{\widehat{\Xi}^k\}_{k \in \mathcal{K}}))$ converges in finitely many iterations.*

Proof. Fix a given quadruplet $(\overline{\theta}, \overline{\boldsymbol{\psi}}, \overline{\boldsymbol{x}}, \{\overline{\boldsymbol{y}}^k\}_{k \in \mathcal{K}})$ in $(\mathcal{IP}^{\text{in}}_{\boldsymbol{w}})$, for any $\overline{\boldsymbol{\xi}} \in \widehat{\Xi}^k$, $\forall k \in \mathcal{K}$, the inner maximization problem of $\boldsymbol{\xi}$ is a maximization problem of a convex function of $\boldsymbol{\xi}$ over a polyhedron $\Xi(\boldsymbol{w}, \overline{\boldsymbol{\xi}})$. Thus, Blankenship and Falk (1976, Corollary 2.1) guarantees the finite convergence of the cutting-plane algorithm. $\qquad\square$

Proof of Theorem 5. We first establish the correctness of the algorithm after termination. If problem $(\mathcal{IP}_{\boldsymbol{w}})$ is infeasible, then in the branch-and-cut algorithm, there is no solution that can give a negative objective value in problem $(\mathcal{IP}^{\text{out}}_{\boldsymbol{w}})$ and be identified as feasible. Assume now that problem $(\mathcal{IP}_{\boldsymbol{w}})$ is feasible and let $(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}})$ be an optimal solution with objective value $\theta^\star$. The branch-and-cut tree must have a leaf node since it terminates. Consider an arbitrary leaf node in the branch-and-cut tree for which the optimal solution $(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}})$ is feasible. Such a leaf node exists because an optimal solution is feasible in the root node, and our branching mechanism ensures that the feasibility is maintained in at least one child node every time the algorithm branches. By definition, the leaf node was not branched. The node could have been fathomed in step 1, where the current incumbent solution must also be optimal since it weakly dominates $(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}})$. Otherwise, the node could have been fathomed in step 3, where the incumbent solution was updated to $(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k \in \mathcal{K}})$.

For the asymptotic result, within any infinite branch, the finite convergence result derived in Lemma A.3 ensures the existence of an infinite sequence of solutions in step 1 of Algorithm 3. For each solution in this sequence, the value $v_{k^\star}$ obtained by solving problem $(\mathcal{IP}_{\boldsymbol{w}}^{\text{in}})$ is less than or equal to 0. We denote the sequence as $\{(\theta^n, \boldsymbol{\psi}^n, \boldsymbol{x}^n, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}}^n)\}_1^{+\infty}$ and the associated solutions of the separation problem as $\{\boldsymbol{\xi}^n\}_1^{+\infty}$, where n is the order of node appear in the branch. Since $\mathcal{X}$ and $\mathcal{Y}$ are compact sets, and an optimal solution $\boldsymbol{\psi}$ is attained by Lemma A.2, the Bolzano–Weierstrass theorem implies that the sequence $\{(\theta^n, \boldsymbol{\psi}^n, \boldsymbol{x}^n, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}}^n), \boldsymbol{\xi}^n\}_1^{+\infty}$ has at least one accumulation point, denoted as $\{(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}), \boldsymbol{\xi}^\star\}$.

We first prove the feasibility of the accumulation point. Denote the value of problem $(\mathcal{IP}_{\boldsymbol{w}}^{\text{out}})$ under the solution $(\theta, \boldsymbol{\psi}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}})$ and scenario $\boldsymbol{\xi}$ as $z((\theta, \boldsymbol{\psi}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}}), \boldsymbol{\xi})$. Suppose that, given the accumulation solution $(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}})$, there exists a scenario $\boldsymbol{\xi}'$ such that $z((\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}), \boldsymbol{\xi}') > 0$. Since problem $(\mathcal{IP}_{\boldsymbol{w}}^{\text{out}})$ finds the worst case scenario for the solution $(\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}})$, the largest violation $z((\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}), \boldsymbol{\xi}^\star) \geq z((\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}), \boldsymbol{\xi}') > 0$. The violation of the sequence $\{(\theta^{n+1}, \boldsymbol{\psi}^{n+1}, \boldsymbol{x}^{n+1}, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}}^{n+1}), \boldsymbol{\xi}^n)\}$ is always less than zero, and the sequence also converges to the same accumulation point. Thus, we have $z((\theta^\star, \boldsymbol{\psi}^*, \boldsymbol{x}^*, \{\boldsymbol{y}^{k^\star}\}_{k\in\mathcal{K}}), \boldsymbol{\xi}^\star) \leq 0$ and raise the desired contradiction.

We now show the optimality of the accumulation point of the solution sequence corresponding to an infinite branch. If there is a solution of the problem $(\mathcal{IP}_{\boldsymbol{w}})$, denoted as $(\overline{\theta}, \overline{\boldsymbol{\psi}}, \overline{\boldsymbol{x}}, \{\overline{\boldsymbol{y}}^k\}_{k\in\mathcal{K}})$, where $\theta' \leq \theta^\star$. Then, using the same logic as the correctness proof, it is either a node solution of the branch-and-cut tree or an accumulation point of a sequence. In each case, the current objective value will be updated to $\theta' + \delta$, where $\delta \geq 0$ and is an arbitrarily small number. Consequently, the branch corresponding to the sequence $\{(\theta^n, \boldsymbol{\psi}^n, \boldsymbol{x}^n, \{\boldsymbol{y}^k\}_{k\in\mathcal{K}}^n)\}_1^{+\infty}$ must be finite since it will be cut by

a better solution. Therefore, we reach a contradiction with the infinite branch and conclude our proof. □

Proof of Theorem 6. For any given $\boldsymbol{w}^r$, if a solution $\boldsymbol{w}$ violates the cut $(\mathcal{C}_{\text{s-feas}})$, then we have

$$\sum_{i \in \mathcal{I}_r} 1 - w_i \leq 0, \tag{A.27}$$

and thus $\boldsymbol{w} \geq \boldsymbol{w}^r$. If instead the solution $\boldsymbol{w}^r$ violates the constraint (10) under the scenario $\bar{\boldsymbol{\xi}}$, then for any solution $\boldsymbol{w}$ such that $\boldsymbol{w} \geq \boldsymbol{w}^r$, we have

$$\boldsymbol{V}(\bar{\boldsymbol{\xi}})\boldsymbol{w} \geq \boldsymbol{V}(\bar{\boldsymbol{\xi}})\boldsymbol{w}^r \geq \boldsymbol{H}\bar{\boldsymbol{\xi}}, \tag{A.28}$$

where the first inequality holds because the matrix $\boldsymbol{V}(\boldsymbol{\xi})$ is component-wise non-negative, and the second one holds because of the infeasibility of $\bar{\boldsymbol{\xi}}$. Thus $\boldsymbol{w}$ is infeasible in problem (1). This concludes the proof. □

B Supplementary Numerical Results

B.1 Supplementary Numerical Results to Section 5.1.2

B.1.1 Supplementary Runtime Results

In Section 5.1.2, we state that since the number of bilinear terms between continuous variables is small, solving the monolithic MINLO is more efficient than using the decomposition algorithm in all cases. From Table 1, we know that Gurobi solves all instances to optimality within the given time limit. In this section, for computational completeness, we compare and report the performance of the decomposition Algorithm 1, which uses the Branch-and-Cut Algorithm 3 to evaluate $\Phi(\boldsymbol{w})$, both with and without the cuts introduced in Section 4.3. We also evaluate its performance when the Branch-and-Bound Algorithm 2 is used with the same cuts in Section 4.3.

In Algorithm 1, the time limit for evaluating $\Phi(\boldsymbol{w})$ is set to 300 seconds, and the lower bound L is obtained by solving a deterministic problem with $\boldsymbol{\xi} = (\boldsymbol{r}, \boldsymbol{c})$.

Table B.1 summarizes the computational results for the same randomly generated instances. From the table, we observe that the improved decomposition algorithm, which uses Algorithm 2 to evaluate $\Phi(\boldsymbol{w})$, outperforms the basic decomposition algorithm. Across all instances, the proposed algorithm solves more instances to optimality and achieves a smaller optimality gap, which decreases as K increases. As the number of policies K increases, the improved decomposition algorithm outperforms the other two approaches in an increasing number of instances. For example, when $N = 40$, the number of times a better solution was found by the improved decomposition approach is $\{12, 19, 20\}$ for $K \in \{2, 3, 4\}$, respectively. Additionally, the improvements in objective value over the static solutions achieved by the improved decomposition algorithm are greater than those of the other two methods across all cases. Notably, the decomposition algorithm using Algorithm 2 to evaluate $\Phi(\boldsymbol{w})$ rarely finds high-quality solutions when the problem size is large (e.g., $N > 10$), and its solution time is longer than the other two methods for small-sized problems. This result highlights the efficiency of the proposed Branch-and-Cut Algorithm 3 and its advantages.

		Decomposition					Decomposition with Added Cuts					Decomposition with Algorithm 2 and Cuts				
N	K	Opt(#)	Time(s)	Gap	Better(#)	Improvement	Opt(#)	Time(s)	Gap	Better(#)	Improvement	Opt(#)	Time(s)	Gap	Better(#)	Improvement
	2	**20**	20.7	**0.0%**	**0**	**42.6%**	20	**4.2**	**0.0%**	**0**	**42.6%**	20	14.2	**0.0%**	**0**	**42.6%**
10	3	**20**	31.1	**0.0%**	**0**	**43.9%**	20	**7.2**	**0.0%**	**0**	**43.9%**	20	50.9	**0.0%**	**0**	**43.9%**
	4	**20**	52.8	**0.0%**	**0**	**43.9%**	20	**26.6**	**0.0%**	**0**	**43.9%**	20	101.8	**0.0%**	**0**	**43.9%**
	2	0	7200.0	40.7%	1	84.6%	**13**	**2769.5**	**23.6%**	1	**85.9%**	0	7200.0	100.0%	0	−100.0%
20	3	0	7200.0	32.4%	4	110.6%	**5**	**1071.4**	**19.2%**	8	**108.1%**	0	7200.0	100.0%	0	−100.0%
	4	0	7200.0	30.5%	3	116.6%	**8**	**3494.4**	**15.7%**	13	**120.3%**	0	7200.0	100.0%	0	−100.0%
	2	0	7200.0	41.4%	0	111.9%	**2**	**1799.4**	**34.1%**	**2**	**118.2%**	0	7200.0	100.0%	0	−100.0%
30	3	**0**	**7200.0**	36.2%	2	131.2%	**0**	**7200.0**	**28.9%**	13	**147.7%**	**0**	**7200.0**	100.0%	**0**	−100.0%
	4	0	7200.0	37.0%	1	127.9%	1	**1182.8**	**25.5%**	19	**158.3%**	0	7200.0	100.0%	0	−100.0%
	2	0	7200.0	48.3%	2	86.4%	1	**3356.4**	**44.9%**	12	**98.0%**	0	7200.0	100.0%	0	−100.0%
40	3	**0**	**7200.0**	44.2%	0	101.0%	**0**	**7200.0**	**35.8%**	19	**129.6%**	**0**	**7200.0**	100.0%	0	−100.0%
	4	**0**	**7200.0**	42.6%	0	106.8%	**0**	**7200.0**	**30.1%**	20	**149.3%**	**0**	**7200.0**	100.0%	0	−100.0%

Table B.1 Supplementary computational results of the best box problem. **Decomposition** and **Decomposition with Added Cuts** refer to Algorithm 1 using the Branch-and-Cut Algorithm 3 to evaluate $\Phi(\boldsymbol{w})$, without and with the cuts introduced in Section 4.3, respectively. **Decomposition with Algorithm 2 and Cuts** refers to Algorithm 1 using the Branch-and-Bound Algorithm 2 with the cuts introduced in Section 4.3. **Opt**(#) corresponds to the number of instances solved to optimality, **Time**(s) to the average computational time (in seconds) for instances solved to optimality, and **Gap** to the average optimality gap for the instances not solved within the time limit. **Better**(#) denotes the number of instances in each method that achieved a better objective value than the other method. **Improvement** denotes the average improvement in the objective value of the K-adaptability solution found in the time limit over the static solution found in the time limit. The improvement is calculated as the ratio of the difference between the objective values of the K-adaptability solution and the static solution to the objective value of the static solution.

B.1.2 Improvement over K

In this section, to study the quality of the solutions obtained by the K-adaptability approximation, we randomly generated 20 instances of a small-sized best box problem with $N = 10$, which we can solve to optimality. Based on the structure of the problem, the original problem (4) with full adaptability needs at most 11 candidate policies (one for selecting each box at the second stage, and another one if no box is opened). We now summarize the average improvements over the static solution as K increases in Figure B.1.
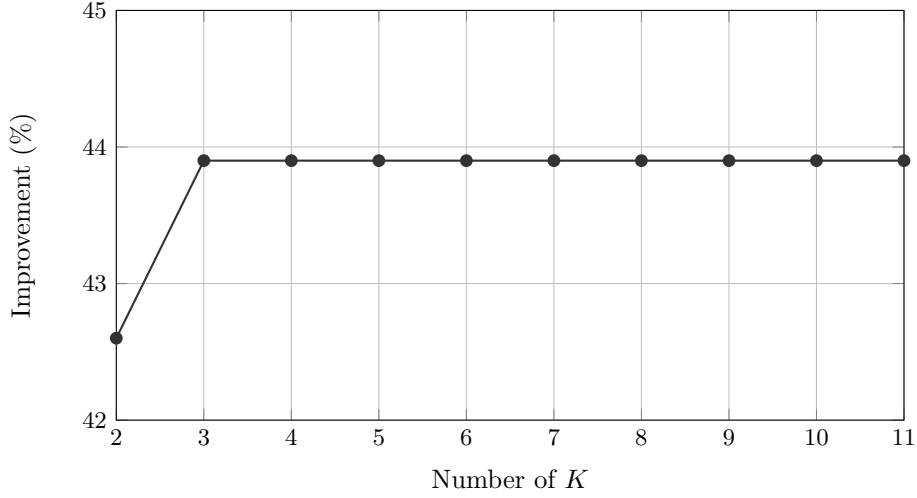
Fig. B.1 Average improvement over the static solution in the best box problem as K increases

The average improvements over the static solution for $K = 2$ and $K \geq 3$ are 42.6% and 43.9%, respectively. This result shows that, in the small-sized best box problem, 3 (out of a total of 11) candidate policies are sufficient for the K-adaptability approximation (5) to be optimal in the original problem (4).

B.1.3 Out of Sample Performance

In this section, we compare the out-of-sample performance of the RO solution and the DRO solution in the best box problem. For all the problems, we sample 200

distributions and calculate the expected costs under both the RO and DRO solutions. For each distribution, we draw 1000 samples of the random parameter $\bar{\boldsymbol{\zeta}}$ from the uniform distribution $U[-1,1]$, and generate the scenarios $\bar{\boldsymbol{\xi}}$ using the relationship between them defined in the support set:

$$\bar{\boldsymbol{\xi}} = (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}}), \text{ where } \bar{\xi}_n^{\mathrm{r}} = \left(1 + \boldsymbol{\Phi}_n^\top \bar{\boldsymbol{\zeta}}/2\right) r_n, \ \bar{\xi}_n^{\mathrm{c}} = \left(1 + \boldsymbol{\Psi}_n^\top \bar{\boldsymbol{\zeta}}/2\right) c_n, \ n \in \mathcal{N}.$$

If the constraints in the ambiguity set are not satisfied, we decrease the interval length of the uniform distribution by $2/3$ and resample the scenarios of $\boldsymbol{\zeta}$ until the constraints are satisfied. For each sample $\bar{\boldsymbol{\xi}}$ and DRO solution, denoted as $\{\boldsymbol{w}, \boldsymbol{x}, \boldsymbol{\psi}, \{\boldsymbol{y}^k\}_{k \in \mathcal{K}}\}$, we solve the following inner maximization problem to find the best candidate policy $\boldsymbol{y}^{k^\star}$:

$$
\begin{aligned}
\min_{k \in \mathcal{K}} \ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}})} \ & \boldsymbol{c}^\top \boldsymbol{\psi} + \boldsymbol{\xi}^\top \boldsymbol{C}\, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\, \boldsymbol{y}^k - \boldsymbol{\psi}^\top \boldsymbol{g}(\boldsymbol{\xi}) \\
\text{s.t.} \quad & \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}}).
\end{aligned}
\tag{B.29}
$$

The objective value associated with $\bar{\boldsymbol{\xi}}$ is calculated as $\bar{\boldsymbol{\xi}}^\top \boldsymbol{C} \boldsymbol{x} + \bar{\boldsymbol{\xi}}^\top \boldsymbol{D} \boldsymbol{w} + \bar{\boldsymbol{\xi}}^\top \boldsymbol{Q} \boldsymbol{y}^{k^\star}$. We then use the objective value of each sample to calculate the expectation of the discrete distribution.

For the RO solution $\{\boldsymbol{w}, \boldsymbol{x}, \{\boldsymbol{y}^k\}_{k \in \mathcal{K}}\}$, we solve the following inner maximization problem to find the best candidate policy:

$$
\begin{aligned}
\min_{k \in \mathcal{K}} \ \max_{\boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}})} \ & \boldsymbol{\xi}^\top \boldsymbol{C}\, \boldsymbol{x} + \boldsymbol{\xi}^\top \boldsymbol{D}\, \boldsymbol{w} + \boldsymbol{\xi}^\top \boldsymbol{Q}\, \boldsymbol{y}^k \\
\text{s.t.} \quad & \boldsymbol{T}(\boldsymbol{\xi})\boldsymbol{x} + \boldsymbol{V}(\boldsymbol{\xi})\boldsymbol{w} + \boldsymbol{W}(\boldsymbol{\xi})\boldsymbol{y}^k \leq \boldsymbol{H}\boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \Xi(\boldsymbol{w}, \bar{\boldsymbol{\xi}}),
\end{aligned}
\tag{B.30}
$$

and calculate the expected value in the same way described above.

Table B.2 reports the relative difference of the median, the 95% and 99% quantiles between the two methods. From the simulation results, we can see that the DRO

approach, in general, produces less conservative solutions that perform better than those of the RO approach even on non-worst-case distributions.

N	K	median	95% quantile	99% quantile
10	1	7.10%	7.20%	7.32%
	2	2.44%	2.34%	2.31%
	3	4.04%	3.91%	3.90%
	4	4.03%	3.90%	3.89%
20	1	6.84%	6.80%	6.83%
	2	0.23%	0.22%	0.23%
	3	0.96%	0.96%	0.97%
	4	1.06%	1.05%	1.07%
30	1	2.77%	2.75%	2.73%
	2	1.69%	1.69%	1.68%
	3	1.64%	1.65%	1.66%
	4	1.87%	1.87%	1.88%
40	1	3.68%	3.65%	3.67%
	2	0.58%	0.57%	0.57%
	3	−0.33%	−0.31%	−0.31%
	4	0.43%	0.44%	0.44%

Table B.2 The relative difference between the median, the 95% and 99% quantiles of the expected objective values in the 200 simulated distributions given by the RO and DRO solutions to the Best Box problem. Each cell represents an average of 20 instances.

B.2 Supplementary Numerical Results for Section 5.2.2

B.2.1 Supplementary Runtime Results

In this section, for computational completeness, we present the performance of the decomposition Algorithm 1 using the Branch-and-Bound Algorithm 2 with the cuts introduced in Section 4.3. In Algorithm 1, the time limits for evaluating $\Phi(\boldsymbol{w})$ are set to $[300s, 600s, 1200s, 2400s]$ for problem sizes $N \in \{5, \ 10, \ 15, \ 20\}$, consistent with

Section 5.2.2. The lower bound L is again obtained by solving a deterministic problem with $\boldsymbol{\xi} = (\boldsymbol{r}, \boldsymbol{c})$.

Table B.3 summarizes the computational results for the same randomly generated instances and should be read in tandem with Table 3. From the table, we observe that, across all cases, using Algorithm 3 to evaluate $\Phi(\boldsymbol{w})$ solves more instances to optimality and, if the time limit is reached, it returns solutions with better objective values. Algorithm 3 also requires shorter solving times, has smaller gaps, and shows more improvement. Particularly, when the problem size is large (e.g., $N > 10$), the decomposition algorithm using Algorithm 3 finds solutions having considerable improvement over the static solutions, while Algorithm 2 has difficulty finding high-quality solutions. This again demonstrates the efficiency and advantages of the proposed Branch-and-Cut Algorithm 3.

B.2.2 Improvement over K

In this section, to study the quality of the solutions obtained by the K-adaptability approximation, we consider a small-sized problem with $N = 4$ and $M = 2$, where we are able to solve the problem with the set of candidate policies containing all possible second-stage decisions. We randomly generate 20 instances, and summarize the average improvements over the static solution as K increases in Figure B.2.
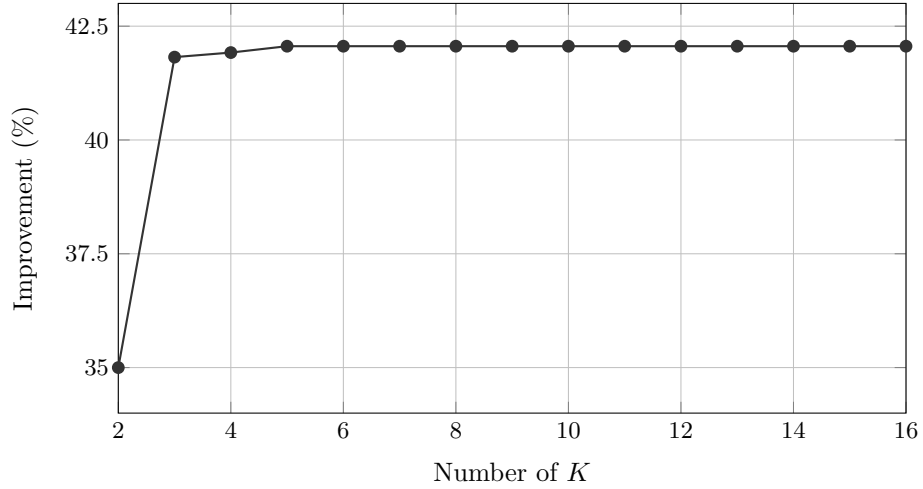
The average improvement over the static solutions for $K = 2, 3, \ldots, 6$ are 35.0%, 41.8%, 41.9%, 42.1%, 42.1%, respectively. This result shows that, in the small-sized R&D problem, 5 (out of a total of 16) candidate policies are sufficient for the K-adaptability approximation (5) to be optimal compared to the original problem (4).

B.2.3 Out of Sample Performance

In this section, we compare the out-of-sample performance of the RO solution and the DRO solution in the R&D problem. We sample 200 distributions that satisfy

69

N	K	Opt(#)	Time(s)	Gap	Better(#)	Improvement
	2	6	253.0	17.9%	0	10.1%
5	3	0	7200.0	17.8%	0	12.4%
	4	0	7200.0	18.4%	0	12.4%
	2	0	7200.0	34.1%	0	4.4%
10	3	0	7200.0	28.1%	1	14.2%
	4	0	7200.0	31.9%	0	5.6%
	2	0	7200.0	88.9%	0	−73.0%
15	3	0	7200.0	91.1%	0	−73.0%
	4	0	7200.0	88.8%	0	−73.0%
	2	0	7200.0	95.9%	0	−79.8%
20	3	0	7200.0	94.4%	0	−79.8%
	4	0	7200.0	94.9%	0	−79.8%

Table B.3 Computational results of Algorithm 1 using the Branch-and-Bound Algorithm 2 with the cuts introduced in Section 4.3 for R&D problem. **Opt**(#) corresponds to the number of instances solved to optimality, **Time**(s) to the average computational time (in seconds) for instances solved to optimality, and **Gap** to the average optimality gap for the instances not solved within the time limit. **Better**(#) denotes the number of instances in each method that achieved a better objective value than the other method. **Improvement** denotes the average improvement in the objective value of the K-adaptability solution found in the time limit over the static solution found in the time limit. The improvement is calculated as the ratio of the difference between the objective values of the K-adaptability solution and the static solution to the objective value of the static solution.

Fig. B.2 Average improvement over static solution in the R&D project portfolio optimization problem as K increases

the moment constraints in $\mathcal{P}_{\mathrm{RD}}$. For each instance and distribution, we evaluate the expected return $\mathbb{E}_{\mathbb{P}}[-{\boldsymbol{\xi}^r}^{\top}(\boldsymbol{w}+\theta\boldsymbol{y}(\boldsymbol{\xi}))]$ using $\boldsymbol{y}(\boldsymbol{\xi})$ as the solution computed by RO and DRO approaches. The details of the sampling procedure are similar to Section B.1.3. Table B.4 reports the relative difference of the median, the 95% and 99% quantiles between the two methods. Similar to the best box problem, Table B.4 shows that, in the R&D problem, the DRO solutions, in general, also outperform the RO solutions on non-worst-case distributions.

N	K	median	95% quantile	99% quantile
5	1	−0.03%	−0.03%	−0.03%
	2	9.55%	9.78%	9.87%
	3	4.53%	4.68%	4.71%
	4	2.90%	3.03%	3.06%
10	1	−0.27%	−0.26%	−0.25%
	2	22.92%	22.95%	22.96%
	3	4.50%	4.80%	4.92%
	4	4.46%	4.54%	4.56%
15	1	−0.60%	−0.59%	−0.60%
	2	21.66%	21.67%	21.66%
	3	3.07%	3.51%	3.60%
	4	3.93%	3.95%	4.03%
20	1	−0.26%	−0.25%	−0.25%
	2	21.44%	21.44%	21.43%
	3	4.85%	5.32%	5.47%
	4	4.20%	4.21%	4.25%

Table B.4 The relative difference between the median, the 95% and 99% quantiles of the expected objective values in the 200 simulated distributions given by the RO and DRO solutions to the R&D problem. Each cell represents an average of 20 instances.

B.3 Distributionally Robust R&D Project Portfolio Optimization with Loans

B.3.1 Problem Description

We perform experiments on several instances of a variant of the distributionally robust R&D project portfolio optimization problem studied in Section 5.2. In this problem, in the first stage, the decision-maker is allowed to borrow an amount x of money which she must pay back, with interest α in the final stage. We use the same ambiguity set as the one described in 5.2. The support of the distribution of the uncertain parameters is given by:

$$\Xi = \left\{ (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}}) \in \mathbb{R}^{2N} : \exists \boldsymbol{\zeta} \in [-1, 1]^M : \xi_n^{\mathrm{r}} = \left(1 + \boldsymbol{\Phi}_n^\top \boldsymbol{\zeta}/2\right) r_n, \ \xi_n^{\mathrm{c}} = \left(1 + \boldsymbol{\Psi}_n^\top \boldsymbol{\zeta}/2\right) c_n, \right.$$
$$\left. n \in \mathcal{N} \right\}.$$

We define $\boldsymbol{\xi} = (\boldsymbol{\xi}^{\mathrm{r}}, \boldsymbol{\xi}^{\mathrm{c}})$, $\boldsymbol{\mu} = (\boldsymbol{r}, \boldsymbol{c})$, and use the following ambiguity set for the uncertain returns and costs:

$$\mathcal{P}_{\mathrm{RD}} = \left\{ \mathbb{P} \in \mathcal{M}_+(\mathbb{R}^{N_\xi}) \ : \ \mathbb{P}(\boldsymbol{\xi} \in \Xi) = 1, \quad \mathbb{E}_{\mathbb{P}}\left[|\boldsymbol{\xi} - \boldsymbol{\mu}|\right] \leq 0.15\boldsymbol{\mu}, \right.$$
$$\left. \mathbb{E}_{\mathbb{P}}\left[\left|\mathbf{e}^\top (\boldsymbol{\xi} - \boldsymbol{\mu})\right|\right] \leq 0.15 N^{-1/2}\mathbf{e}^\top \boldsymbol{\mu}) \right\}.$$
$$(\text{B.31})$$

The second set of constraints in the ambiguity set enforces the mean absolute deviation of the individual returns and costs to be bounded. The last constraint imposes an upper bound on the cumulative deviation of the projects' returns and costs from their expected value.

The distributionally robust R&D project portfolio optimization problem with loans can be written as

$$
\begin{aligned}
\min \quad & \sup_{\mathbb{P} \in \mathcal{P}} \; \mathbb{E}_{\mathbb{P}} \left[-\boldsymbol{\xi}^{r\top} \left(\boldsymbol{w} + \theta \boldsymbol{y}(\boldsymbol{\xi}) \right) + (1 + \alpha) x \right] \\
\text{s.t.} \quad & \boldsymbol{w} = (\boldsymbol{w}^r, \, \boldsymbol{w}^r), \; \boldsymbol{w}^r \in \{0, \, 1\}^N, \; x \in \mathbb{R}_+ \\
& \left. \begin{aligned}
& \boldsymbol{y}(\boldsymbol{\xi}) \in \{0, \, 1\}^N \\
& \boldsymbol{\xi}^{c\top} \left(\boldsymbol{w}^r + \boldsymbol{y}(\boldsymbol{\xi}) \right) \leq B + x \\
& \boldsymbol{w}^r + \boldsymbol{y}(\boldsymbol{\xi}) \leq \mathbf{e}
\end{aligned} \right\} \quad \forall \boldsymbol{\xi} \in \Xi \\
& \boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{y}(\boldsymbol{\xi}') \quad \forall \boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi \, : \, \boldsymbol{w} \circ \boldsymbol{\xi} = \boldsymbol{w} \circ \boldsymbol{\xi}'.
\end{aligned}
\tag{B.32}
$$

B.3.2 Computational Results

We run experiments on 80 randomly generated instances of the K-adaptability counterpart of problem (16): 20 instances for each $N \in \{5, 10, 15, 20\}$ with risk factors $M \in \{4, 5, 8, 10\}$, respectively. We draw $\boldsymbol{c}$ uniformly at random from the box $[0, 10]^N$ and let $\boldsymbol{r} = 5\boldsymbol{c}$, $B = \mathbf{e}^\top \boldsymbol{c}/2$ and $\alpha = 0.2$. The risk load factors $\boldsymbol{\Phi}_n$ and $\boldsymbol{\Psi}_n$ are sampled uniformly at random from the standard simplex.

We report the results of solving problem (16) by utilizing the monolithic MINLO reformulation and applying the decomposition Algorithm 1 with and without the cuts introduced in Section 4.3. In the decomposition Algorithm 1, we set the time limit of evaluating $\Phi(\boldsymbol{w})$ to $[300s, 600s, 1200s, 2400s]$ for problems with size $N \in \{5, 10, 15, 20\}$. The lower bound value L in Algorithm 1 is found by solving a deterministic problem obtained by setting $\boldsymbol{\xi} = (\boldsymbol{r}, \boldsymbol{c})$. If the subproblem evaluating $\Phi(\boldsymbol{w})$ is not solved to optimality within the time limit, we use the current lower bound of the subproblem to generate a valid integer cut of the form ($\mathcal{C}_{\text{opt}}$) and use the upper bound to update the objective value $\theta^\star$ in step 4. If the main problem is not solved to optimality within the time limit, we report $\theta^\star$ as the objective value and the current optimal objective value of the main problem ($\mathcal{P}_{\text{relax}}$) as the lower bound.

N	K	MINLO					Decomposition					Decomposition with Added Cuts				
		Opt(#)	Time(s)	Gap	Better(#)	Improvement	Opt(#)	Time(s)	Gap	Better(#)	Improvement	Opt(#)	Time(s)	Gap	Better(#)	Improvement
	2	0	7200.0	–	2	−21.6%	**6**	1218.2	0.2%	**5**	**30.4%**	6	**426.0**	**0.1%**	4	**30.4%**
5	3	0	7200.0	–	1	14.7%	0	7200.0	5.6%	**7**	36.5%	2	**663.0**	**1.1%**	5	**36.9%**
	4	0	7200.0	–	2	13.4%	0	7200.0	9.9%	1	37.2%	2	**936.0**	**6.1%**	13	**37.6%**
	2	**1**	**4404.5**	–	2	3.6%	0	7200.0	29.5%	3	43.3%	0	7200.0	**20.2%**	11	**43.8%**
10	3	**0**	**7200.0**	–	3	38.8%	**0**	**7200.0**	26.4%	7	51.0%	**0**	**7200.0**	**20.4%**	10	**51.3%**
	4	**0**	**7200.0**	–	0	9.6%	**0**	**7200.0**	25.1%	8	53.9%	**0**	**7200.0**	**20.0%**	12	**54.2%**
	2	**0**	**7200.0**	–	1	−3.5%	**0**	**7200.0**	29.8%	7	36.9%	**0**	**7200.0**	**28.1%**	11	**37.1%**
15	3	**0**	**7200.0**	–	1	−20.9%	**0**	**7200.0**	25.9%	3	44.5%	**0**	**7200.0**	**24.0%**	16	**45.6%**
	4	**0**	**7200.0**	–	0	−41.3%	**0**	**7200.0**	24.4%	6	47.4%	**0**	**7200.0**	**22.3%**	14	**48.7%**
	2	**0**	**7200.0**	–	1	−26.9%	**0**	**7200.0**	28.5%	6	40.1%	**0**	**7200.0**	**27.6%**	13	**40.6%**
20	3	**0**	**7200.0**	–	1	−19.2%	**0**	**7200.0**	25.6%	1	45.8%	**0**	**7200.0**	**23.2%**	18	**49.3%**
	4	**0**	**7200.0**	–	0	−64.5%	**0**	**7200.0**	25.4%	1	46.3%	**0**	**7200.0**	**21.6%**	19	**52.4%**

Table B.5 Summary of computational results of the R&D problem with loans. **Decomposition** and **Decomposition with Added Cuts** refer to Algorithm 1 without and with the cuts introduced in Section 4.3, respectively. **Opt**(#) corresponds to the number of instances solved to optimality, **Time**(s) to the average computational time (in seconds) for instances solved to optimality, and **Gap** to the average optimality gap for the instances not solved within the time limit. We write '-' when no valid lower bound was found. **Better**(#) denotes the number of instances in each method that achieved a better objective value than the other method. **Improvement** denotes the average improvement in the objective value of the K-adaptability solution found in the time limit over the static solution found in the time limit. The improvement is calculated as the ratio of the difference between the objective values of the K-adaptability solution and the static solution to the objective value of the static solution.

Table B.5 summarizes the computational results across those instances. From the table, we observe that the improved decomposition algorithm achieves optimality or finds better solutions for more instances than the basic decomposition algorithm and solves the monolithic MINLO problem when K is large. Across all instances, the proposed algorithm has a significantly smaller optimality gap,, which decreases with K. We notice that the MINLO problem is often unable to obtain a valid lower bound across all problem sizes. As the number of policies K increases, the number of instances for which the improved decomposition algorithm achieves a better objective value increases. For example, when $N = 20$, the number of better solutions found by our approach is $\{13, 18, 19\}$ with $K \in \{2, 3, 4\}$, respectively. The improvements in the objective value given by using the improved decomposition algorithm over the static solutions are $\{40.6\%, 49.3\%, 52.4\%\}$, whereas the improvements obtained by solving the MINLO problem and using the basic decomposition algorithm are $\{-26.9\%, -19.2\%, -64.5\%\}$ and $\{40.1\%, 45.8\%, 46.3\%\}$, respectively. In all cases, the improvement in the objective value produced by the decomposition algorithm over the static solutions increases with the number of policies K. This is expected since larger values K offer more flexibility in the recourse decisions, leading to a better objective value. However, the trend is not clear in MINLO solutions due to computational difficulties, and the improvement can even be negative.

We also present the performance of the decomposition Algorithm 1 using the Branch-and-Bound Algorithm 2 with the cuts introduced in Section 4.3. Table B.6 summarizes the computational results for the same randomly generated instances and should be read in companion with Table 3. From the table, we observe that, across all cases, using Algorithm 2 to evaluate $\Phi(\boldsymbol{w})$ solves fewer instances to optimality and with better objective values. It also requires longer solving times, has larger gaps, and shows less improvement. Particularly, when the problem size is large (e.g., $N > 10$), the decomposition algorithm using Algorithm 2 has difficulty finding high-quality solutions

with positive improvement compared to the static solution. This again demonstrates the efficiency and advantages of the proposed Branch-and-Cut Algorithm 3.

N	K	Opt(#)	Time(s)	Gap	Better(#)	Improvement
	2	0	7200.0	27.1%	0	7.6%
5	3	0	7200.0	33.1%	0	4.6%
	4	0	7200.0	34.0%	0	2.8%
	2	0	7200.0	48.6%	0	−1.9%
10	3	0	7200.0	48.8%	0	−2.3%
	4	0	7200.0	48.3%	0	−2.0%
	2	0	7200.0	53.6%	0	−10.0%
15	3	0	7200.0	56.2%	0	−13.2%
	4	0	7200.0	56.1%	0	−11.2%
	2	0	7200.0	61.8%	0	−23.4%
20	3	0	7200.0	65.9%	0	−27.9%
	4	0	7200.0	72.3%	0	−39.3%

Table B.6 Computational results of Algorithm 1 using the Branch-and-Bound Algorithm 2 with the cuts introduced in Section 4.3 for R&D problem with loans. **Opt**(#) corresponds to the number of instances solved to optimality, **Time**(s) to the average computational time (in seconds) for instances solved to optimality, and **Gap** to the average optimality gap for the instances not solved within the time limit. **Better**(#) denotes the number of instances in each method that achieved a better objective value than the other method. **Improvement** denotes the average improvement in the objective value of the K-adaptability solution found in the time limit over the static solution found in the time limit. The improvement is calculated as the ratio of the difference between the objective values of the K-adaptability solution and the static solution to the objective value of the static solution.

Next, we examine the suboptimality of the RO solution. We define $z^\star_{\text{RO}}$ and $z^\star_{\text{DRO}}$ as in Section 5.1.2. The relative differences between $z^\star_{\text{RO}}$ and $z^\star_{\text{DRO}}$ are summarized in Table B.7. From the table, we see that the average suboptimality of the RO solution among all instances is up to 51.25%, which shows the importance of incorporating distributional information in the uncertainty set.

We also compare the out-of-sample performance of the RO solution and the DRO solution. We sample 200 distributions that satisfy the moment constraints in $\mathcal{P}_{\text{RD}}$.

K	$N=5$	$N=10$	$N=15$	$N=20$
1	15.74%	13.58%	20.38%	19.47%
2	42.54%	49.12%	48.81%	50.5%
3	43.83%	44.93%	49.71%	51.25%
4	43.71%	43.83%	46.92%	45.71%

Table B.7 Suboptimality of RO solution to the R&D problem with loans. Each cell represents an average of 20 instances. In each instance, the relative differences are calculated by $\dfrac{(z_{\text{DRO}}^* - z_{\text{RO}}^*)}{(z_{\text{DRO}}^* + z_{\text{RO}}^*)/2}$.

For each instance and distribution, we evaluate the expected net return $\mathbb{E}_{\mathbb{P}}[-\boldsymbol{\xi}^{r\top}(\boldsymbol{w} + \theta\boldsymbol{y}(\boldsymbol{\xi})) + (1+\alpha)x]$ using $\boldsymbol{y}(\boldsymbol{\xi})$ as the solution computed by RO and DRO approaches. The details of the sampling procedure can be found in Section B.1.3. Table B.8 reports the relative difference of the median, the 95% and 99% quantiles between the two methods. From the table, we can see that, on the simulated non-worst-case distributions, the DRO solutions outperform the RO solutions by up to 49%.

N	K	median	95% quantile	99% quantile
5	1	18.96%	18.66%	18.55%
	2	39.90%	40.01%	40.07%
	3	35.93%	35.97%	35.99%
	4	33.84%	33.92%	33.94%
10	1	17.68%	17.35%	17.27%
	2	47.68%	47.62%	47.62%
	3	31.81%	31.82%	31.88%
	4	38.31%	38.26%	38.28%
15	1	18.26%	18.05%	17.81%
	2	49.08%	48.98%	48.85%
	3	34.21%	34.33%	34.24%
	4	36.88%	36.87%	36.80%
20	1	18.13%	17.90%	17.75%
	2	49.43%	49.31%	49.25%
	3	30.77%	30.89%	30.90%
	4	36.33%	36.30%	36.27%

Table B.8 The relative difference between the median, the 95% and 99% quantiles of the expected objective values in the 200 simulated distributions given by the RO and DRO solutions to the R&D problem with loans. Each cell represents an average of 20 instances.