Branch-and-Cut for Mixed-Integer Linear Decision-Dependent Robust Optimization

Henri Lefebvre, Martin Schmidt, Simon Stevens, Johannes Thürauf

Abstract. Decision-dependent robust optimization (DDRO) problems are usually tackled by reformulating them using a strong-duality theorem for the uncertainty set model. If the uncertainty set is, however, described by a mixed-integer linear model, dualization techniques cannot be applied and the literature on solution methods is very scarce. In this paper, we exploit the equivalence of DDRO and optimistic bilevel optimization to derive the first tailored branch-and-cut method for mixed-integer linear DDRO problems. To this end, we consider both general-purpose intersection cuts for the case of decision-dependent uncertainty sets given by general mixed-integer linear models as well as more tailored interdiction cuts for the case of downward monotone uncertainty set models. Using the special structure of robust optimization problems, we also derive a novel version of intersection cuts that are only stated in the space of the upper-level variables of the respective bilevel reformulation. We present an extensive numerical study that demonstrates the applicability of our method.

1. Introduction

Bilevel and robust optimization have many structural similarities since both include a nested optimization problem. In bilevel optimization, this nested problem corresponds to the follower's problem whereas in robust optimization, the nested problem is used to determine the worst-case realization of the uncertainty within a prescribed set. While the commonalities between these two fields of optimization have been mentioned already in Stein (2013) and Leyffer et al. (2020), the first systematic study of the relations between bilevel and robust optimization has been published only recently by Goerigk et al. (2025). In particular, the latter paper includes a result showing that optimistic bilevel optimization and decision-dependent robust optimization (DDRO) are actually equivalent.

Originally, decision-dependent uncertainties stem from the field of stochastic optimization problems, where they are often called endogenous uncertainties; see, e.g., Jonsbråten et al. (1998), Zhan et al. (2016), Apap and Grossmann (2017), Hellemo et al. (2018), and Motamed Nasab and Li (2021). A bit later, decision-dependent uncertainties have been studied in the field of robust optimization as well. For instance, Nohadani and Sharma (2018) consider shortest-path problems with decision-dependent uncertainties for the arc lengths. The latter paper also includes a hardness result showing that even linear DDRO problems with polyhedral uncertainty sets and affine decision-dependence are NP-complete in general. Poss (2013) and Poss (2014) examines combinatorial optimization problems such as the knapsack problem under budgeted and knapsack uncertainties. DDRO has also been used in applications; see Spacey et al. (2012) for software partitioning problems, Lappas and Gounaris (2016) and Vujanic et al. (2016) for scheduling applications,

---

Date: March 5, 2026.

2020 *Mathematics Subject Classification.* 90C70, 90C17, 91A65, 90C11.

Key words and phrases. Robust optimization, Bilevel optimization, Decision-dependent uncertainty sets, Branch-and-cut, Intersection cuts, Interdiction cuts.

Aigner et al. (2022) and Zhang et al. (2025) for DDRO problems in the field of energy networks and renewable energy carriers in energy systems, or Zhu et al. (2022) for health care applications. Last but not least, rather recently, DDRO aspects can also be found in other fields of robust optimization such as two-stage and multistage robust optimization (Zhang and Feng 2020; Avraamidou and Pistikopoulos 2020; Zeng and Wang 2022) or distributionally robust optimization (Luo and Mehrotra 2020; Feng et al. 2021; Basciftci et al. 2021; Yu and Shen 2022; Doan 2022; Ryu and Jiang 2025).

The equivalence of DDRO and optimistic bilevel optimization shown by Goerigk et al. (2025) paved the way for the follow-up paper by Lefebvre et al. (2025), in which the authors present the first extensive numerical study demonstrating the potential of solving DDRO problems as bilevel optimization problems. The bottom line is that classic techniques from robust optimization outperform the bilevel reformulation of the problem if the worst-case uncertainty realization can be obtained by dualization techniques. However, dualization cannot be applied if the uncertainty set of the decision-dependent robust problem is given by a mixed-integer linear programming (MILP) model. Contrarily, the bilevel reformulation yields a mixed-integer bilevel problem and remains valid, as it does not rely on dualization techniques.

For DDRO problems with a mixed-integer linear uncertainty set, the literature on solution methods is rather scarce. To the best of our knowledge, the only publicly available and maintained solver that can be used is the Yasol solver that is originally designed for solving quantified mixed-integer problems (Ederer et al. 2017; Hartisch and Lorenz 2022). As a special case, Yasol can also handle DDRO problems with mixed-integer linear uncertainty sets; see also Lefebvre et al. (2025), where this has been done. More recently, new approaches have also been introduced by Lozano and Borrero (2025). In particular, the authors present a column-and-constraint generation algorithm and a method based on decision diagrams.

On the other hand, the literature on solution techniques for mixed-integer linear bilevel problems has grown a lot over the last one to two decades. For an overview, we refer to the survey by Kleinert et al. (2021) or the paper by Thürauf et al. (2024) about the instance collection BOBILib (`https://bobilib.org`), which also contains the largest up-to-date computational study in this field. Consequently, and as a result of the observations made by Lefebvre et al. (2025), it is promising to tailor the most successful techniques from mixed-integer linear bilevel optimization to DDRO problems with mixed-integer linear uncertainty set models. In this paper, we exactly pursue this approach. We reformulate the DDRO problem as a bilevel MILP and design a branch-and-cut method to solve them. While branch-and-bound for bilevel MILPs goes back to Moore and Bard (1990), the first branch-and-cut method has been published by DeNegre and Ralphs (2009). Afterward, many research has been carried out in the field of cutting planes for bilevel MILPs; see, e.g., Tahernejad et al. (2020), Tahernejad and Ralphs (2025), Fischetti et al. (2017), Fischetti et al. (2018), and Fischetti et al. (2019). In particular, the latter three papers consider the application of intersection cuts (Balas 1971; Cornúejols 2008; Conforti et al. 2014) to bilevel optimization problems. For general mixed-integer linear lower-level problems, these cuts can safely be considered the state-of-the-art in branch-and-cut for bilevel MILPs. On the contrary, if the lower-level problem has a more specific structure, this is exploited to obtain better performing methods. One of the most prominent of these structures is the downward monotonicity property having interdiction-like constraints in the lower-level problem; see Fischetti et al. (2019). The resulting interdiction cuts can only be used for these specific settings but are also much more effective than the general-purpose intersection cuts.

Our contribution is the following. We design the first general branch-and-cut method for mixed-integer linear DDRO problems. If the uncertainty set is given by a general MILP model, we use intersection cuts to tackle this situation. In case of downward monotone uncertainty sets, e.g., if the constraints are interdiction-like, we exploit more specific and stronger interdiction cuts. Note that general bilevel problems are formulated in both the upper- and lower-level variables and that the upper-level problem, in general, explicitly depends on the lower-level variables. Consequently, general-purpose cuts such as intersection cuts are defined in the entire variable space as well. The situation is different if the DDRO problem is reformulated as a bilevel problem. In this case, the upper-level problem does not depend on the specific uncertainty (or lower-level variables), but only on the respective objective function value of the lower-level problem. While this specific property also occurs in bilevel interdiction problems, leading to interdiction cuts that are originally formulated only in the upper-level variable space, this specific setup also enables us to develop intersection cuts that are defined only in the upper-level variable space as well.

The remainder of this paper is structured as follows. In Section 2 we briefly present the problem statement, the basic notation, the most central and standing assumptions, and a formal proof for Σ_2^p-completeness of the considered problem class. Afterward, the branch-and-cut framework is discussed in Section 3, where we also derive interdiction cuts (Section 3.1) as well as intersection cuts (Section 3.2) for DDRO. Our computational study is given in Section 4, before we then close the paper with a summary and some pointers to future research questions in Section 5.

2. Problem Statement

In this paper, we consider DDRO problems of the form

$$\min_x \quad c^\top x \tag{1a}$$

$$\text{s.t.} \quad x \in X, \tag{1b}$$

$$a^\top x + b^\top (u \circ x_J) \leq \beta, \quad \text{for all } u \in U(x_I). \tag{1c}$$

The index set $I \subseteq \{1, \dots, n\}$ with $|I| = p$ describes the set of variables that parameterizes the uncertainty set. The index set $J \subseteq \{1, \dots, n\}$ with $|J| = q$ denotes the set of variables that is affected by the uncertainty. We define the element-wise (or Hadamard) product of two vectors $u, x_J \in \mathbb{R}^q$ by $(u \circ x)_i = x_i u_i,\ i \in \{1, \dots, q\}$. The set X contains any integrality constraints, bounds on the variables x, and any other constraints that are not affected by the uncertainty. The continuous relaxation of the set X is denoted by $\mathring{X}$. Note that it is without loss of generality to only consider a single uncertain constraint. All of the following developments can be generalized to multiple uncertain constraints.

Moreover, we consider general polyhedral uncertainty sets of the form

$$U(x_I) := \left\{ u \in \bar{U} \colon Cx_I + Du \leq \alpha \right\} \subseteq \mathbb{R}^q, \tag{2}$$

where $\bar{U} \subseteq \mathbb{R}^q_{\geq 0}$ is non-empty, contains integrality constraints and finite bounds on the uncertainty parameters u. To ensure the existence of solutions for the upcoming models, we need the following assumption.

Assumption 1. *The uncertainty set $U(x_I)$ is non-empty and compact for all $x_I \in X$.*

As typically done in robust optimization, we can re-write Constraint (1c) as

$$a^\top x + \max_{u \in U(x_I)} \left\{ b^\top (u \circ x_J) \right\} \leq \beta. \tag{3}$$

Before we start to derive a solution method for the problem class at hand, we now formally prove that DDRO problems of the form in (1) are Σ_2^p-complete in general. We note that Σ_2^p-hardness is mentioned in the recent work by Lozano and Borrero (2025), but no explicit reduction is provided. We now explicitly present such a reduction in the following.

Proposition 1. *The decision version of Problem (1) with rational input data as well as X and $U(x_I)$ given by mixed-integer linear formulations is Σ_2^p-complete in general.*

Proof. Membership in Σ_2^p follows from the results of Woeginger (2021) because the decision version of Problem (1) admits a formulation with two quantifiers.

The Σ_2^p-hardness result easily follows because the Σ_2^p-hard problem DNeg in Caprara et al. (2014) can be written in the format of Problem (1). DNeg is a variant of a bilevel knapsack interdiction problem and is given by (already written in a notation towards Problem (1))

$$\begin{aligned} \min_{x \in \{0,1\}^n} \quad & b^\top y \\ \text{s.t.} \quad & d^\top x \leq e, \\ & y \in S(x), \end{aligned}$$

where $d \in \mathbb{R}^n$, $e \in \mathbb{R}$, and $S(x)$ is the set of globally optimal solutions to the x-parameterized lower-level problem

$$\begin{aligned} \max_{y \in \{0,1\}^n} \quad & b^\top y \\ \text{s.t.} \quad & b^\top y \leq B, \\ & y_i \leq 1 - x_i, \quad \text{for all } i \in \{1, \dots, n\}. \end{aligned}$$

Using the notation $X := \left\{(x, \eta) \in \{0,1\}^n \times \mathbb{R} \colon d^\top x \leq e\right\}$ and with c being the zero vector except from the last entry, which is 1, this problem is equivalent to

$$\begin{aligned} \min_{x,\eta} \quad & c^\top (x, \eta) \\ \text{s.t.} \quad & (x, \eta) \in X, \\ & \eta \geq b^\top y, \quad \text{for all } y \in U(x) \end{aligned}$$

with

$$U(x) := \left\{y \in \{0,1\}^n : b^\top y \leq B, \, y_i \leq 1 - x_i, \, i \in \{1, \dots, n\}\right\}.$$

It is easy to see that this can be cast as Problem (1), which proves the claim. □

3. A Branch-and-Cut Approach

We now present a branch-and-cut (B&C) algorithm for solving DDRO problems of the form in (1). To this end, we use the epigraph reformulation of the inner maximization problem to obtain the problem

$$\min_{x,\eta} \quad c^\top x \tag{4a}$$

$$\text{s.t.} \quad x \in X, \; \eta \in \mathbb{R}, \tag{4b}$$

$$a^\top x + \eta \leq \beta, \tag{4c}$$

$$\max_{u \in U(x_I)} \left\{b^\top (u \circ x_J)\right\} \leq \eta, \tag{4d}$$

which is equivalent to Problem (1).

For the B&C algorithm, we initially relax the robust constraint (4d) as well as the integrality constraints included in the set X. The uncertain constraints are then dynamically added to the model as cuts. Thus, in every node t of the B&C tree, we solve the problem

$$\begin{aligned} \min_{x,\eta} \quad & c^\top x && \text{(5a)} \\ \text{s.t.} \quad & x \in X^t, \ \eta \in \mathbb{R}, \ (x,\eta) \in \Omega^t, && \text{(5b)} \\ & a^\top x + \eta \leq \beta, && \text{(5c)} \end{aligned}$$

where $X^t = \mathring{X} \cap M^t$ and M^t contains the branching decisions and integrality cuts that have been added previously. Moreover, Ω^t contains the cuts that have been added so far.

When an integer-feasible solution (x^t, η^t) is found at node t, i.e., $x^t \in X$, we check of the uncertain constraints (4c) and (4d) are satisfied. To this end, we solve the subproblem

$$\hat{\eta}^t := \Phi(x^t) = \max_{u \in U(x_I^t)} \left\{ b^\top (u \circ x_J^t) \right\}, \tag{6}$$

arising from the inner maximization problem. If the node solution is not feasible for the uncertain constraints, i.e., if $a^\top x^t + \hat{\eta}^t > \beta$, we generate a cut that is valid for the uncertain constraints and add it to Ω^t. Overall, every node t is tackled by the method given in Algorithm 1.

Algorithm 1 Node processing

Input: The problem at node t of the B&C tree, the current incumbent Z^*, and the sets M^t and Ω^t.
1: Solve Problem (5).
2: **if** Problem (5) is infeasible **then**
3: Fathom the current node and **return** to the main B&C loop.
4: Let (x^t, η^t) denote the solution of Problem (5).
5: **if** $c^\top x^t \geq Z^*$ **then**
6: Fathom the current node and **return** to the main B&C loop.
7: **if** $x^t \notin X$ **then**
8: Either generate an integrality cut, augment M^t, and **go to** Step 1 or branch and **return** to the main B&C loop.
9: Solve the Subproblem (6) for x^t to obtain $\hat{\eta}^t = \Phi(x^t)$.
10: **if** (4c) is satisfied for $(x^t, \hat{\eta}^t)$ **then**
11: Set the new incumbent $Z^* \leftarrow c^\top x^t$ and fathom the current node.
12: **else**
13: Either generate a cut for $(x^t, \hat{\eta}^t)$, augment Ω^t, and **go to** Step 1 or branch on the integer-feasible point (x^t, η^t) and **return** to the main B&C loop.

In the following sections, we present two different types of cuts that can be used in Step 13 of the node processing algorithm.

3.1. Interdiction Cuts.

In this section, we derive an interdiction cut to solve decision-dependent robust optimization problems, in which the uncertainty set is described by downward monotone constraints. For the remainder of this section, we make the following assumption.

Assumption 2. *The index sets I and J are disjoint and it holds $|I| = |J| = p$.*

Therefore, we can write $x = (x_I, x_J)$ with $I = \{1, \dots, p\}$ and $J = \{p+1, \dots, n\}$. While this is w.l.o.g., it significantly enhances the readability of the required notation. It also implies that all variables x_J can be interdicted. Furthermore, we require the downward monotonicity property of an x-parameterized problem.

Definition 1 (Fischetti et al. 2019)**.** *Let $x \in X$ be given and fixed. Moreover, let $\hat{u} \in U(x_I)$ be a feasible point for (6). If $u' \in \bar{U}$ is a feasible point for all $0 \le u' \le \hat{u}$, we say that the problem satisfies the* downward monotonicity property.

For the remainder of this section, we consider the DDRO problem in its epigraph formulation (4), where $x_I \in \{0,1\}^p$, $x_J \in \mathbb{R}^p_{\ge 0}$, and $b \ge 0$. The uncertainty set is defined as

$$U(x_I) := \left\{u \in \bar{U} \colon Du \le \alpha,\ u_i \le 1 - x_i,\ i \in I\right\}, \tag{7}$$

which is a special case of the uncertainty set in (2). Here, $D \in \mathbb{R}^{p\times m}_{\ge 0}$ represents the interdiction weights and $\alpha \in \mathbb{R}^p_{\ge 0}$ is the interdiction budget. Note that the interdiction constraints can be generalized to $u_i \le w_i(1 - x_i)$ for some $w_i \in \mathbb{R}_{\ge 0}$.

With this definition, the uncertainty set satisfies the downward monotonicity property from Definition 1 because D has only non-negative entries. The following penalization technique is known from the bilevel literature and is used, e.g., by Wood (2011) and Fischetti et al. (2019). The uncertain constraint (4d) can be reformulated by removing the interdiction constraints from the uncertainty set (7) and penalizing them to obtain

$$\max_{u \in \hat{U}} \left\{ b^\top (u \circ x_J) - \sum_{i \in I} M_i x_i u_i \right\} \le \eta, \tag{8}$$

where $\hat{U} = \{u \in \bar{U} \colon Du \le \alpha,\ 0 \le u_i \le 1,\ i \in I\}$ holds. This is again equivalent to

$$b^\top (u \circ x_J) - \sum_{i \in I} M_i x_i u_i \le \eta, \quad \text{for all } u \in \hat{U}. \tag{9}$$

For sufficiently large values of M_i, the penalty term guarantees that any optimal solution of the subproblem fulfills $x_i u_i = 0$ for all $i \in I$, i.e., the interdiction constraint $u_i \le 1 - x_i$ for binary x_i and u_i is satisfied. Hence, it holds

$$\Phi(x_I) = \max_{u \in U(x_I)} b^\top (u \circ x_J) = \max_{u \in \hat{U}} \left\{ b^\top (u \circ x_J) - \sum_{i \in I} M_i x_i u_i \right\}. \tag{10}$$

Thus, Problem (4) can be reformulated as

$$\begin{aligned} \min_{x,\eta} \quad & c^\top x && \text{(11a)}\\ \text{s.t.} \quad & x \in X,\ \eta \in \mathbb{R}, && \text{(11b)}\\ & a^\top x + \eta \le \beta, && \text{(11c)}\\ & \Phi(x_I) \le \eta. && \text{(11d)} \end{aligned}$$

The following theorems are based on the results of Fischetti et al. (2019).

Theorem 1. *Let $u \in \hat{U}$ be given arbitrarily. Then, the interdiction cut*

$$\eta \ge b^\top (u \circ x_J) - \sum_{k=1}^{p} b_k (x_I \circ x_J)_k u_k \tag{12}$$

is valid for Problem (11).

Proof. Take any feasible point (x, η) for Problem (11) and $\hat{u} \in \hat{U}$. Let $u'_i = \hat{u}_i(1-x_i)$ for all $i \in I$. Since $\hat{u}_i \ge 0$ and $x_i \in \{0,1\}$, it holds $0 \le u'_i \le \hat{u}_i$. Moreover, because the inner maximization problem (10) is downward monotone due to $D \in \mathbb{R}^{p\times m}_{\ge 0}$, it holds $Du' \le D\hat{u} \le \alpha$. The interdiction constraints $u'_i \le 1 - x_i$ hold due to the definition of u'. Thus, u' is feasible for (10) for the given x. Therefore,

$$\begin{aligned}
\eta &\geq \Phi(x_I) \\
&= \max_{u \in U(x_I)} \; b^\top (u \circ x_J) \\
&\geq b^\top (u' \circ x_J) \\
&= \sum_{k=1}^{p} b_k (\hat{u} \circ x_J)_k - \sum_{k=1}^{p} b_k (x_I \circ x_J)_k \hat{u}_k \\
&= b^\top (\hat{u} \circ x_J) - \sum_{k=1}^{p} b_k (x_I \circ x_J)_k \hat{u}_k
\end{aligned}$$

holds as claimed. □

Theorem 2. *The decision-dependent robust optimization problem* (4) *with uncertainty set* (7) *can be equivalently reformulated by replacing Constraint* (4d) *with the family of interdiction cuts* (12)*, leading to the problem*

$$\min_{x,\eta} \quad c^\top x \tag{13a}$$
$$\text{s.t.} \quad x \in X, \;\; \eta \in \mathbb{R}, \tag{13b}$$
$$a^\top x + \eta \leq \beta, \tag{13c}$$
$$\eta \geq b^\top (u \circ x_J) - \sum_{k=1}^{p} b_k (x_I \circ x_J)_k u_k, \quad \textit{for all } u \in \hat{U}. \tag{13d}$$

Proof. We have to show that, for any feasible point (x, η) to Problem (13), the Constraints (13d) hold if and only if Constraint (11d) holds. Let (x, η) be feasible for Problem (13). Then, Constraint (13d) is equivalent to

$$\eta \geq \max_{u \in \hat{U}} \left\{ b^\top (u \circ x_J) - \sum_{k=1}^{p} b_k (x_I \circ x_J)_k u_k \right\} \tag{14a}$$
$$= \max_{u \in U(x_I)} \left\{ b^\top (u \circ x_J) - \sum_{k=1}^{p} b_k (x_I \circ x_J)_k u_k \right\} \tag{14b}$$
$$= \max_{u \in U(x_I)} \quad b^\top (u \circ x_J), \tag{14c}$$

where the equality in (14b) holds due to Theorem 1 as well as $U(x_I) \subseteq \hat{U}$ and the equality in (14c) follows from the interdiction constraints $u_i \leq 1 - x_i$ in the uncertainty set $U(x_I)$, which ensure that $u_i x_i = 0$ holds for all $i \in I$. □

From the last two results, we obtain that $(b \circ x_J)_i$ is an appropriate value for M_i, $i \in I$. To linearize the interdiction cuts, we can use the McCormick inequalities (McCormick 1976) or use any overestimator of $(b \circ x_J)_i$, since any larger penalty parameter is also valid.

Definition 2. *Let $x \in X$ be given. A solution $u^* \in \hat{U}$ to Problem* (10) *is called maximal, if there exists no extreme point $\hat{u} \in \hat{U}$ with $\hat{u} \neq u^*$ and $u^* \leq \hat{u}$.*

In the next theorem, we show that it is sufficient to consider only the maximal solution $u^* \in \hat{U}$ for the inner maximization problem (10) instead of all feasible points $\hat{u} \in \hat{U}$ for Problem (13) to be an equivalent reformulation of Problem (4).

Theorem 3. *Let $\hat{u} \in \hat{U}$ be non-maximal for Problem* (10) *and let $u^* \in \hat{U} \setminus \{\hat{u}\}$ be such that $u^* \geq \hat{u}$. Then, the interdiction cut* (12) *for $\hat{u}$ is dominated by the one for u^*.*

Proof. Let $x_i \in \{0,1\}$ for all $i \in I$. Since $u^* \geq \hat{u}$ holds, it follows $u_i^*(1-x_i) \geq \hat{u}_i(1-x_i)$ for all $i \in I$. Thus, it holds

$$b^\top(\hat{u} \circ x_J) - \sum_{k=1}^{p} b_k(x_I \circ x_J)_k \hat{u}_k \leq b^\top(u^* \circ x_J) - \sum_{k=1}^{p} b_k(x_I \circ x_J)_k u_k^* \leq \eta. \qquad \square$$

Note that, computationally, it is not clear whether generating cuts for non-maximal solutions may be beneficial. It could be that generating weaker cuts is easier and leads to smaller computation times.

The cuts in (13d) can thus be replaced by the respective cuts for the maximal solution $u^* \in \hat{U}$ of the subproblem (10). Consequently, the problem solved at node t is given by

$$\min_{x,\eta} \quad c^\top x \tag{15a}$$
$$\text{s.t.} \quad x \in X^t, \ \eta \in \mathbb{R}, \tag{15b}$$
$$a^\top x + \eta \leq \beta, \tag{15c}$$
$$\eta \geq b^\top(u \circ x_J) - \sum_{k=1}^{p} b_k(x_I \circ x_J)_k u_k, \quad \text{for all } u \in U^t, \tag{15d}$$

in which U^t contains all subproblem solutions u^t used for imposing interdiction cuts of the form (12). Thus, in the root node, we start with the sets $X^0 = \mathring{X}$ and $U^0 = \emptyset$. Then, for every node t in the B&C-tree, we execute Algorithm 1 with cuts in the form of (12) in Line 13.

3.2. **Intersection Cuts.** To tackle DDRO problems with more general uncertainty sets, we derive intersection cuts for the uncertain problem (1) with uncertainty sets of the form (2). Intersection cuts are well-known in the context of mixed-integer bilevel optimization. Due to the equivalence of DDRO and bilevel optimization shown by Goerigk et al. (2025), they can be used to solve DDRO problems as well.

Before deriving intersection cuts tailored to the DDRO setting, let us briefly recall the bilevel branch-and-cut method presented by Fischetti et al. (2018) and how to apply it for DDRO problems. To that end, let us consider the Problem (4) with a single reformulated uncertain constraint in its epigraph reformulation. The inner maximization problem reads

$$\Phi(x_I) = \max_{u \in U(x_I)} b^\top(u \circ x_J), \tag{16}$$

where the non-empty- and compact-valued uncertainty set is given by

$$U(x_I) := \left\{u \in \bar{U} \colon Cx_I + Du \leq \alpha\right\}. \tag{17}$$

Following Goerigk et al. (2025), the corresponding DDRO problem can be solved as a bilevel problem and, in particular, via the bilevel branch-and-cut method designed by Fischetti et al. (2017). The bilevel reformulation reads

$$\begin{aligned} \min_{x,\eta,u} \quad & c^\top x \\ \text{s.t.} \quad & x \in X, \ \eta \in \mathbb{R}, \\ & a^\top x + \eta \leq \beta, \\ & \eta \geq b^\top(u \circ x_J), \\ & u \in \mathcal{R}(x), \end{aligned} \tag{18}$$

where $\mathcal{R}(x)$ is the set of optimal solutions to the lower-level problem

$$\begin{aligned} \max_{u \in \bar{U}} \quad & b^\top (u \circ x_J) \\ \text{s.t.} \quad & Cx_I + Du \le \alpha. \end{aligned} \tag{19}$$

The bilevel branch-and-cut method uses the continuous relaxation of the single-level relaxation of the above bilevel problem as root-node problem, leading to node problems in the (x, η, u)-space. Consequently, intersection cuts within this method are defined in the entire variable space as well. For more details on the cut generation in the classic bilevel branch-and-cut method, we refer to Fischetti et al. (2017) and Fischetti et al. (2018). In the DDRO setting, however, the u-variables do not appear in the definition of Problem (1). We therefore derive intersection cuts directly in the (x, η)-space. This allows us to solve the node problems in this *projected* space at every branch-and-bound node.

Coming back to the general DDRO problem (4), dropping the uncertain constraint (4d), and the integrality constraints leads to the root-node problem of our proposed DDRO branch-and-cut method. The intersection cuts are generated for vertices (x^*, η^*) of the node problem. Typically, two sets are used for the generation of intersection cuts:

(i) A polyhedral cone pointed at the vertex (x^*, η^*) containing all robust-feasible points of the node problem. One can use the so-called corner polyhedron here, which is defined by a basic solution of the model's LP relaxation.
(ii) A convex set S whose interior contains the vertex (x^*, η^*) but no robust-feasible point for the original problem. This set is called robust-free set in our context.

The intersection cut is then generated by finding the hyperplane that is given by the intersection points of these two sets. Since the polyhedral cone is given by the basic solution of the node problem and thus exploits locally-valid information, the intersection cuts are locally valid. This is in contrast to the globally valid interdiction cuts derived in Section 3.1.

In the following, we derive a robust-free set for DDRO problems.

Theorem 4. *For any $\hat{u} \in \bar{U}$, the set*

$$S(\hat{u}) := \left\{(x, \eta) \colon \eta < b^\top (\hat{u} \circ x_J),\ Cx_I + D\hat{u} \le \alpha\right\} \tag{20}$$

does not contain any robust feasible point for Problem (4).

Proof. For any robust feasible point (x, η) with $Cx_I + D\hat{u} \le \alpha$, we have

$$\eta \ge \Phi(x_I) = \max_{u \in U(x_I)} b^\top (u \circ x_J) \ge b^\top (\hat{u} \circ x_J). \tag{21}$$

Hence, a point $(x, \eta) \in S(\hat{u})$ cannot be robust feasible. □

To ensure that any robust infeasible solution to the node problem (x^*, η^*) to be cut belongs to the interior of the robust-free polyhedron, we need to consider an extended version of the above set, whose validity requires the following assumption.

Assumption 3. *$Cx_I + Du - \alpha$ is integer for all $x \in X^t$ and $u \in U(x_I)$.*

The very common assumption that all variables x and u appearing in the inner maximization problem as well as the data (C, D, α) of the problem are integer, ensures that the above assumption holds true. Then, we obtain the following extended robust-free set.

Theorem 5. *Suppose that Assumption 3 holds. For any $\hat{u} \in \bar{U}$, the set*

$$S^+(\hat{u}) := \left\{(x, \eta) \colon \eta \le b^\top (\hat{u} \circ x_J),\ Cx_I + D\hat{u} \le \alpha + \mathbf{1}\right\} \tag{22}$$

does not contain any robust feasible point to Problem (4) in its interior.

Proof. To be in the interior of $S^+(\hat{u})$, a robust feasible point (x, η) needs to satisfy $\eta < b^\top(\hat{u} \circ x_J)$ and $Cx_I + D\hat{u} < \alpha + \mathbf{1}$. By Assumption 3, the latter condition can be replaced by $Cx_I + D\hat{u} \leq \alpha$, hence the claim follows from Theorem 4. □

Consequently, the problem solved in node t reads

$$\min_{x,\eta} \quad c^\top x \tag{23a}$$

$$\text{s.t.} \quad x \in X^t, \ (x, \eta) \in \Omega^t, \tag{23b}$$

$$a^\top x + \eta \leq \beta, \tag{23c}$$

where $X^t = \mathring{X} \cap M^t$ and M^t contains the branching decisions and integrality cuts on x that have been added previously. Moreover, $\Omega^t \subseteq \mathbb{R}^{n+1}$ contains the intersection cuts w.r.t. the extended robust free set $S^+(\hat{u})$ that are valid for the uncertain constraints.

4. Computational Study

In this computational study, we analyze the performance of the proposed branch-and-cut method for solving mixed-integer DDRO problems. We start with the robust knapsack problem with a general knapsack uncertainty set in Section 4.3.1 and then consider interdiction-like uncertainty sets in Section 4.3.2. In both cases, we use the branch-and-cut method for DDRO problems presented in the previous section with intersection cuts from Section 3.2, interdiction cuts from Section 3.1 when applicable, as well as the pure branch-and-bound method. We benchmark those against methods that solve the bilevel reformulation of the DDRO problem, namely our implementation of the bilevel branch-and-cut method developed by Fischetti et al. (2017) and its pure branch-and-bound counterpart. Additionally, we compare to the MibS solver (Tahernejad and Ralphs 2020) for mixed-integer linear bilevel problems (MIBLPs), as well as to the QIP solver Yasol (Ederer et al. 2017). For more details on the MIBLP and QIP reformulations, see Lefebvre et al. (2025), while for QIPs in general, we refer to Goerigk and Hartisch (2021) and the references therein.

Since, to the best of our knowledge, there are no benchmark instances for DDRO that go beyond budgeted or knapsack uncertainty sets, we also compare the DDRO and bilevel versions of the branch-and-cut method on a set of instances taken from the BOBILib (Thürauf et al. 2024); see Section 4.4. We use the bilevel problems from the library and reformulate them as DDRO problems according to Goerigk et al. (2025). We refer to Section 4.2 for details on the instance generation.

Our open-source implementation and all instances used in this computational study are publicly available at `https://github.com/simstevens/bnc-for-ddro`.

4.1. Software and Hardware Setup.

Software and Hardware Setup. All numerical experiments were conducted on a single-core Intel Xeon Gold 6126 at 2.6 GHz with 64 GB RAM and a time limit of 2 h. We implemented the branch-and-cut and branch-and-bound approaches in Python 3.12.2 and used CPLEX 22.1.1 as the underlying MIP solver. To access CPLEX callbacks, we relied on the C++ API and transferred the required functionality to Python.

Internal CPLEX heuristics and preprocessing are deactivated in all experiments because we need to retrieve the LP basis at the nodes to derive intersection cuts. CPLEX's internal cuts remain active in their default setting. We separate the respective cuts only for integer-feasible node solutions, since preliminary experiments showed that separating them at fractional node solutions does not improve the overall performance, i.e., we use "user cuts" in CPLEX's terminology. When applicable, interdiction cuts are separated for all solutions u^t of the subproblem. The number of cuts separated at each node is capped at 20 to limit time-consuming cut generation. A well-known issue of intersection cuts is that they can become very shallow when applied iteratively to the same LP; see Fischetti et al. (2017). To avoid such shallow cuts of the form $\alpha^\top x \leq \gamma$, a cut is discarded if its relative violation in the node solution x^t is too small, i.e., if

$$\alpha^\top x^t - \gamma < 10^{-6}(|\gamma| + 1)$$

holds. Note that the right-hand side is positive since x^t violates the cut. Moreover, we discard cuts for which the ratio between the largest and smallest nonzero coefficient exceeds 10^6 to reduce numerical issues.

For comparison, we use the MIBLP solver MibS 1.2.2 (Tahernejad and Ralphs 2020) to solve the MIBLP reformulation of the DDRO problems and the QIP solver Yasol 4.0.1.5 (Ederer et al. 2017) to solve the QIP reformulation. Both solvers are used with their default settings and CPLEX 22.1.1 as the underlying MIP solver.

4.2. **Instance Generation.** In this section, we describe how the instances for the computational study are generated. Section 4.2.1 covers the decision-dependent knapsack instances and Section 4.2.2 explains how we use the BOBILib instances in the DDRO context.

4.2.1. *Knapsack Instances.* In the following, we analyze two classes of decision-dependent robust knapsack instances. To this end, consider a set of n_y items, each with value c_i and nominal weight $\bar{a}_i$. The actual weight of item i is given by $\bar{a}_i + u_i\hat{a}_i$, where $\hat{a}_i$ is the maximum weight deviation and $u_i \in \{0, 1\}$ indicates whether the weight of item i is at its nominal value or higher. The decision maker selects a subset of items $y \in \{0, 1\}^{n_y}$ to maximize the total value while ensuring that the total weight does not exceed the knapsack capacity d for all realizations of the uncertainty u within the uncertainty set. The first class of knapsack instances is thus given by

$$\max_{y \in \{0,1\}^{n_y}} \quad c^\top y \tag{24a}$$

$$\text{s.t.} \quad \sum_{i \in [n_y]} \bar{a}_i y_i + \sum_{i \in [n_y]} u_i \hat{a}_i y_i \leq d \quad \text{for all } u \in U^k(y), \tag{24b}$$

where the uncertainty set consists of m_u many knapsack constraints and is given as

$$U^k(y) := \left\{u \in \{0, 1\}^{n_x} : f_j^\top u \leq b_j + w_j^\top y,\ j \in [m_u]\right\}. \tag{25}$$

The decision maker therefore influences the uncertainty budget $b_j(y) = b_j + w_j^\top y$ through the number of selected items. This type of uncertainty set was introduced by Poss (2013), who highlights that, for given binary variables y, dense vectors are inherently more affected by the uncertainty than sparse ones w.r.t. classic (decision-independent) uncertainty sets.

The second class of decision-dependent robust knapsack instances contains interdiction-like instances with hedging variables x that incur costs h. Selecting x_i hedges against the uncertainty on item i by restricting it in the uncertainty set. This yields the problem

$$\max_{y,x \in \{0,1\}^{n_y}} \quad c^\top y - h^\top x \tag{26a}$$

$$\text{s.t.} \quad \sum_{i \in [n_y]} \bar{a}_i y_i + \sum_{i \in [n_y]} u_i \hat{a}_i y_i \leq d \quad \text{for all } u \in U^i(x), \tag{26b}$$

with

$$U^i(x) := \left\{u \in \{0, 1\}^{n_y} : f_j^\top u \leq b_j,\ j \in [m_u],\ u_i \leq 1 - x_i,\ i \in [n_y]\right\}. \tag{27}$$

Both instance classes coincide with those in Lefebvre et al. (2025) if $m_u = 1$.

The data for the knapsack instances are generated using the gen2 generator by Martello et al. (1999). It creates knapsack instances with n_y items and parameters within the range from 1 to 1000. The capacity of the knapsack is set to $0.35\,W_{\text{sum}}$, where W_{sum} is the sum of the nominal weights. For a discussion of this capacity choice, we refer to Lefebvre et al. (2025). The nominal weights $\bar{a}$ are strongly correlated to the values c and the maximum weight deviation $\hat{a}$ is set to 10 % of the nominal weight. The hedging costs h_i are taken uniformly at random from $[c_i/10, c_i/5]$. The number of items in the knapsack ranges from $n_y = 20$ to $n_y = 100$ in increments of 20, and the number of constraints in the uncertainty set m_u is taken from $\{1, 10, 100, 1000\}$. For each combination of n_y and m_u, we generate 5 random instances, resulting in a total of 100 instances for each instance class. The data for the uncertainty set is also generated by the gen2 generator with the same parameters. The constant part of the uncertainty budget b equals the knapsack capacity, the coefficients f are the knapsack weights, and w is drawn uniformly from $[0, b/n_y]$ to ensure that $b(y) \geq 0$ holds for all y.

4.2.2. *BOBILib Instances.* To obtain instances that go beyond a budgeted or knapsack uncertainty set, we use the BOBILib containing bilevel optimization instances to generate DDRO instances. Given a bilevel problem of the form

$$\min_{x \in X, y} \quad c_x^\top x + c_y^\top y \tag{28a}$$

$$\text{s.t.} \quad Ax + By \geq a, \tag{28b}$$

$$y \in \arg\min_{\bar{y} \in Y} \{f^\top \bar{y} : Cx + D\bar{y} \geq b\}, \tag{28c}$$

where X and Y contain integrality constraints on the upper- and lower-level variables, respectively, we can equivalently reformulate it as the decision-dependent robust optimization problem

$$\min_{x \in X, y \in Y} \quad c_x^\top x + c_y^\top y \tag{29a}$$

$$\text{s.t.} \quad Ax + By \geq a, \tag{29b}$$

$$Cx + Dy \geq b, \tag{29c}$$

$$f^\top y \leq f^\top u \quad \text{for all } u \in U(x), \tag{29d}$$

where the uncertainty set is given by

$$U(x) := \{u \in Y : Cx + Du \geq b\}. \tag{30}$$

From the BOBILib library, we therefore select all instances from the pure-integer class that have a known optimal solution, resulting in 229 instances. We solve the resulting DDRO problems (29) with the intersection-cut approach from Section 3.2 and with a pure branch-and-bound strategy. For comparison, we solve the bilevel reformulation of the DDRO problem (29) with our own implementation of the bilevel branch-and-cut method by Fischetti et al. (2017) using intersection cuts and pure branch-and-bound. Note that the bilevel reformulation of the DDRO problem is not exactly given by the original bilevel problem (28), but by

$$\begin{aligned} \min_{x \in X, y \in Y, u} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & Cx + Dy \geq b, \\ & f^\top y \leq f^\top u, \\ & u \in \arg\min_{\bar{u} \in Y} \left\{ f^\top \bar{u} \colon Cx + D\bar{u} \geq b \right\}, \end{aligned} \tag{31}$$

where x and y are now upper-level variables and u is a lower-level variable.

TABLE 1. Number of solved instances (out of 25 each) within the time limit of 2 hours for the decision-dependent robust knapsack problem with general knapsack uncertainty set depending on the number of constraints in the uncertainty set m_u.

m_u	1	10	100	1000
MibS	6	12	16	11
Yasol	5	10	13	13
Intersection (DDRO)	6	14	20	18
Intersection (Bilevel)	4	8	6	8
B&B (DDRO)	8	16	20	18

4.3. **Knapsack Instances.** In this section, we analyze the computational performance of the proposed branch-and-cut method for DDRO problems on knapsack problems with decision-dependent uncertainty as described in Section 4.2.1. We first focus on instances with the general knapsack uncertainty set in Section 4.3.1 and then on instances with the interdiction-like uncertainty set in Section 4.3.2. For both problem classes, we compare the DDRO and bilevel versions of the branch-and-cut algorithm against the MIBLP solver MibS and the QIP solver Yasol. For the interdiction-like uncertainty set, we also include a version of the DDRO branch-and-cut method that uses interdiction cuts.

4.3.1. *General Knapsack Uncertainty.* We consider knapsack problems with general knapsack uncertainty set given by Problem (24). Figure 1 shows the empirical cumulative distribution functions (ECDFs) of computation time (top plot) and the number of explored branch-and-bound nodes (bottom plot) for a time limit of 2 hours. The DDRO branch-and-cut and branch-and-bound methods solve more instances within the time limit than the bilevel versions as well as MibS and Yasol. The pure bilevel branch-and-bound approach does not solve a single instance within the time limit, while the pure DDRO branch-and-bound approach solves the most instances overall. The DDRO intersection-cut approach solves more than twice as many instances as the bilevel version. MibS and Yasol solve a similar number of instances, but fewer than the novel DDRO methods. The bottom plot indicates that faster computation times generally coincide with smaller search trees.

Notably, all approaches are able to solve more instances as the number of constraints in the uncertainty set increases, i.e., for larger values of m_u; see Table 1. Although the overall model size increases with a larger m_u, the uncertainty set becomes more restrictive, which seems to have a stronger effect on the computational performance than the increase in model size.

Concluding, the novel DDRO branch-and-cut and branch-and-bound methods perform better than the bilevel versions and the benchmark solvers. Using intersection cuts in the DDRO approach, however, yields no improvements in terms of computational performance for the considered instances compared to the pure branch-and-bound method; see also Remark 1 in this context as well.

4.3.2. *Interdiction-Like Uncertainty.* We now consider decision-dependent robust knapsack problems with interdiction-like uncertainty sets given by Problem (26). In contrast to the general uncertainty sets, we can now also use the interdiction cuts derived in Section 3.1. Figure 2 shows the ECDFs of computation time (top plot) and number of explored branch-and-bound nodes (bottom plot) for a time limit of 2 hours. The interdiction cuts clearly outperform the other approaches, solving 97 out of 100 instances within the time limit; the second-best approach

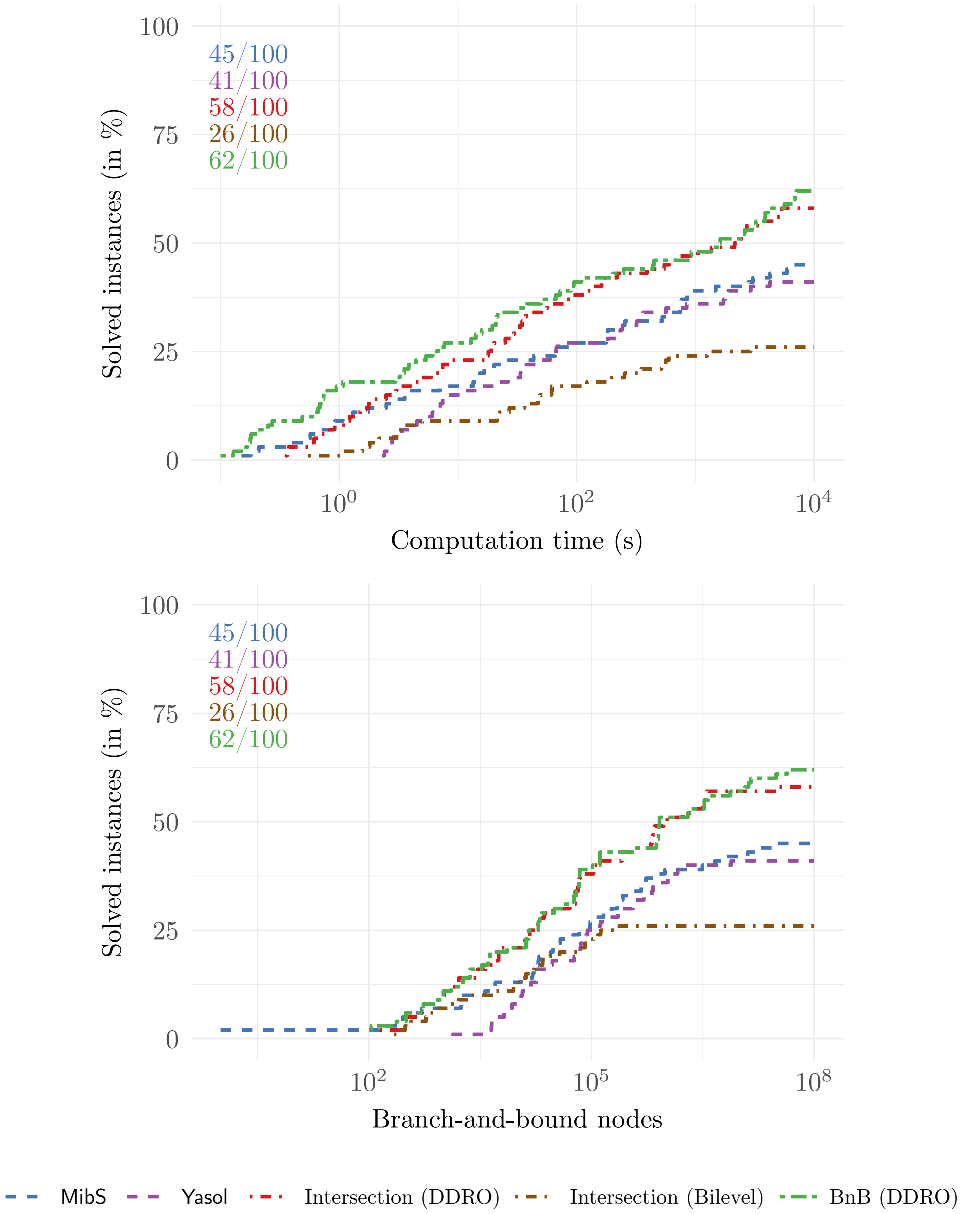

FIGURE 1. ECDF of computation time (top) and number of nodes explored (bottom) for the general knapsack instances. The numbers in the legend indicate the number of instances solved within the time limit of 2 hours.

(DDRO intersection cuts) only solves 52 instances. The interdiction cuts also explore far fewer nodes, indicating that they are highly effective in reducing the search space. For the remaining approaches, the DDRO versions are generally faster than the bilevel ones, though the differences are less pronounced than for the general knapsack uncertainty set. The DDRO branch-and-cut using intersection cuts solves 10 more instances than the pure DDRO branch-and-bound approach and yield smaller search trees, suggesting that the cuts are not as shallow as for the general

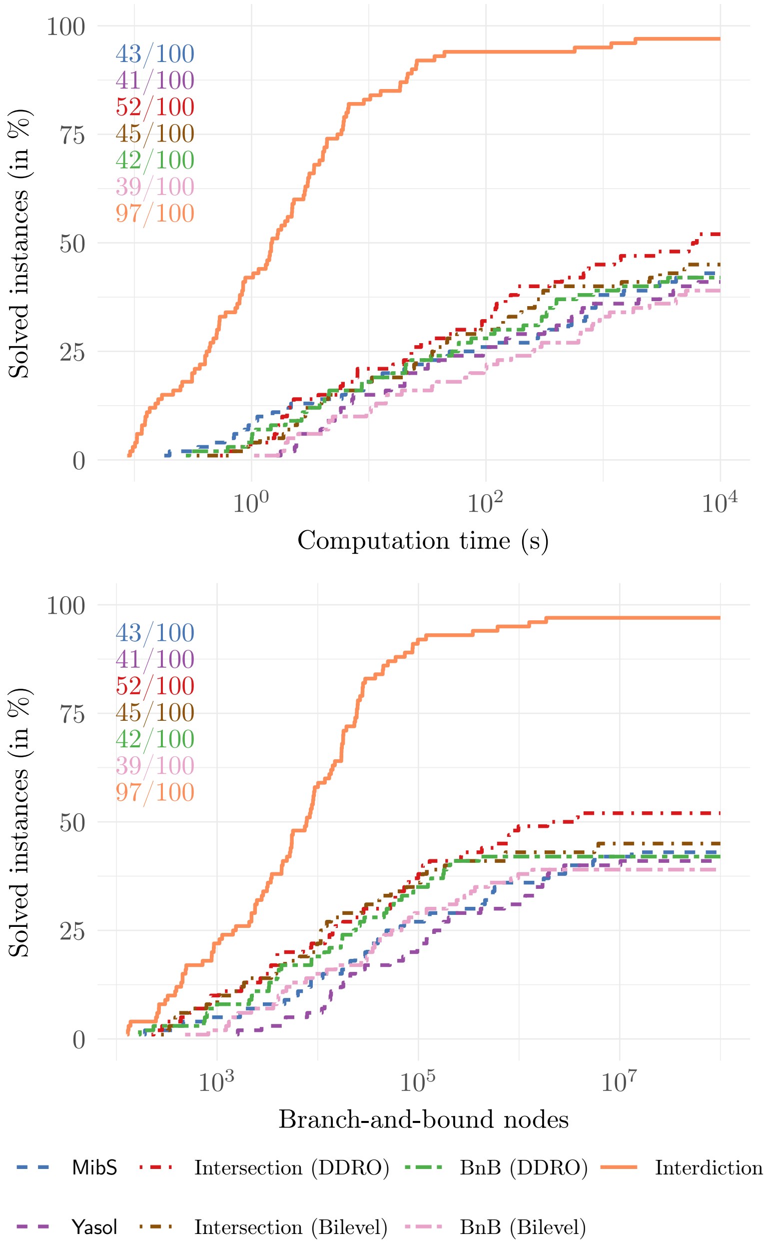

FIGURE 2. ECDF of computation time (top) and number of nodes explored (bottom) for the knapsack interdiction instances. The numbers in the legend indicate the number of instances solved within the time limit of 2 hours.

knapsack instances. The benchmark solvers MibS and Yasol again solve a similar number of instances but are also outperformed by the interdiction cut approach. Note that MibS is not able to detect the interdiction structure of the considered instances and thus cannot exploit its specialized interdiction techniques that would most likely improve the performance. Moreover, the computation times exhibit the same trend with respect to the number of constraints in the uncertainty set m_u as in the general knapsack cases; see Table 2.

TABLE 2. Number of solved instances (out of 25 each) within the time limit of 2 hours for the decision-dependent robust knapsack problem with interdiction-like uncertainty set depending on the number of constraints in the uncertainty set m_u.

m_u	1	10	100	1000
MibS	5	11	16	11
Yasol	5	10	13	13
Intersection (DDRO)	5	12	18	17
Intersection (Bilevel)	5	12	15	13
B&B (DDRO)	5	11	13	13
B&B (Bilevel)	5	11	13	10
Interdiction	22	25	25	25

Overall, the interdiction cuts are highly effective for the knapsack interdiction instances and clearly outperform all other approaches. For the considered instances, the difference between the DDRO and bilevel versions of the branch-and-cut and branch-and-bound methods is less pronounced, but the impact of the intersection cuts is slightly more significant than for the general knapsack instances in Section 4.3.1.

4.4. **BOBILib Instances.** As noted earlier, the literature on decision-dependent robust optimization mainly focuses on budgeted uncertainty sets like those considered for the knapsack problems in Section 4.3. To evaluate the performance of our proposed branch-and-cut method for DDRO compared to the classic bilevel branch-and-cut method by Fischetti et al. (2017), we use instances from the BOBILib, a library of bilevel optimization problems, and reformulate them as decision-dependent robust optimization problems following the derivation in Section 4.2.2. Because we synthetically enlarge the bilevel models for the DDRO reformulation, we refrain from comparing against the MIBLP and QIP solvers and focus on the difference between the DDRO and bilevel branch-and-cut variants. During the solution process, numerical issues occurred for 6 instances, which we exclude from the results. This leaves us with a test set of 223 instances.

Figure 3 reports the ECDFs of computation time (top plot) and number of explored nodes (bottom plot). The DDRO versions of the branch-and-cut and branch-and-bound methods outperform the bilevel versions in both runtime and search-tree size. The pure bilevel branch-and-bound approach performs worst among the four variants. Intersection cuts improve the bilevel branch-and-cut more than they improve the DDRO variant. In fact, pure DDRO branch-and-bound already solves more instances than bilevel branch-and-cut with intersection cuts. Adding intersection cuts to the DDRO version yields only a small additional benefit. The size of the search trees again coincides with the computation times; see the bottom plot in Figure 3.

Overall, intersection cuts do not add much to the computational performance of the DDRO version. However, using the DDRO branch-and-cut method instead of the bilevel version itself is advantageous for the BOBILib instances as well, which may have more complex uncertainty sets than the knapsack instances.

Remark 1. *We have so far only used improving-solution intersection cuts with robust-free sets of the form* (22) *for both the DDRO and bilevel branch-and-cut methods. Other intersection-cut families in the bilevel literature—such as improving-direction and hypercube intersection cuts (Tahernejad and Ralphs 2020)—could be adapted to the DDRO branch-and-cut method to improve the performance. This*

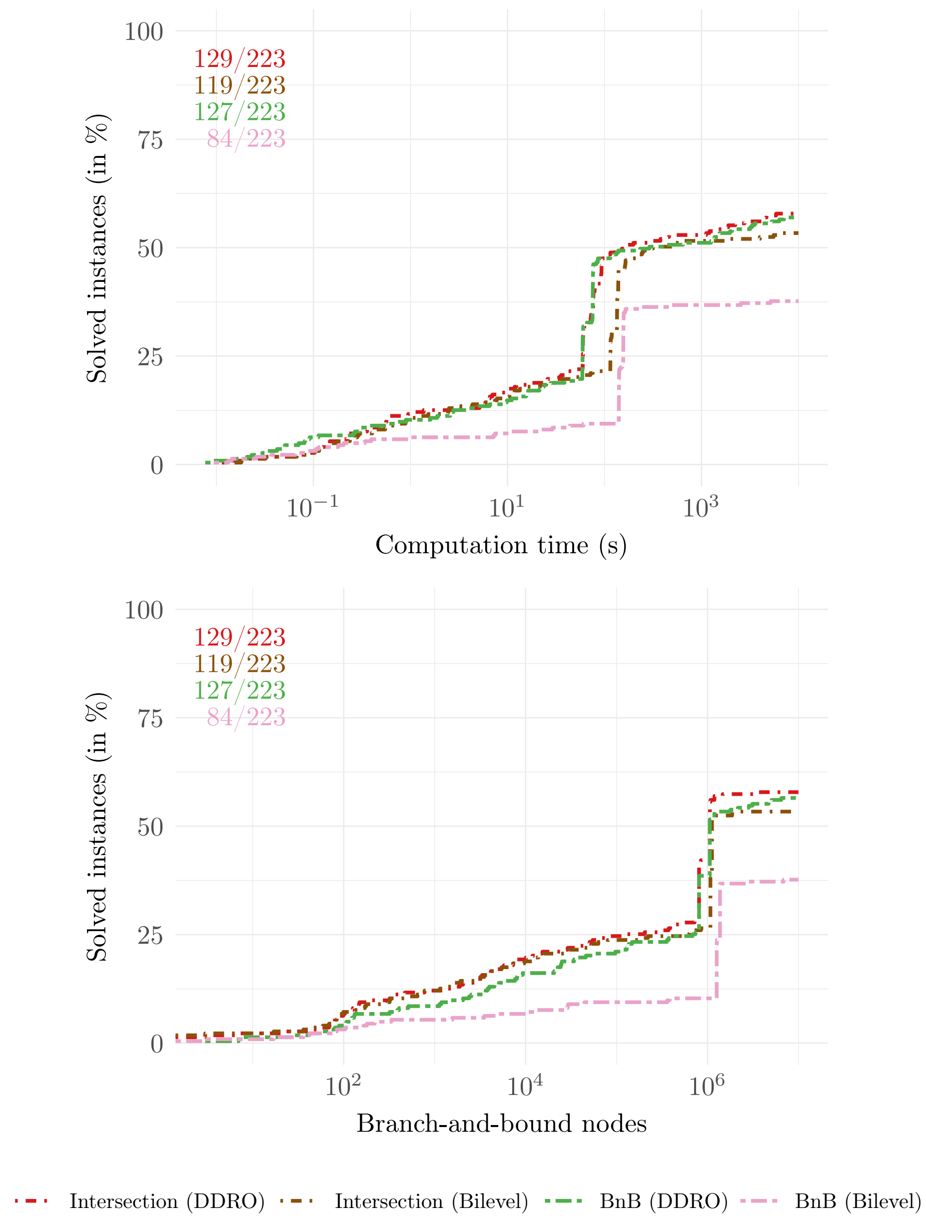

FIGURE 3. ECDF of computation time (left) and number of nodes explored (right) for the BOBILib instances. The numbers in the legend indicate the number of instances solved within the time limit of 2 hours.

seems especially promising for interdiction instances from the BOBILib, which already have the min-max structure of DDRO problems and can therefore be tackled directly without the reformulation in Section 4.2.2. Preliminary experiments with MibS on these instances suggest that improving-direction intersection cuts are more effective than improving-solution cuts. We will analyze these aspects in our future research.

5. Conclusion

In this paper, we exploited the equivalence of DDRO problems with decision-dependent uncertainty sets represented by mixed-integer linear models and mixed-integer linear bilevel optimization problems. This enabled us to derive the first tailored branch-and-cut method for mixed-integer linear DDRO problems using both general-purpose intersection cuts as well as more tailored interdiction cuts that can be applied to uncertainty sets satisfying the so-called downward monotonicity property. Our numerical study clearly demonstrates the applicability of our approach. Since the first branch-and-cut framework is now available for mixed-integer DDRO problems, we hope that this paves the way to more algorithmic research in this direction.

For instance, many different techniques from the entire field of mixed-integer programming including presolve, other cutting planes, primal heuristics, or symmetry detection, to just name a few, can now be carried over to mixed-integer linear DDRO. Moreover, our future research will also contain the computational analysis of other variants of intersection cuts such as improving-direction or hypercube intersection cuts.

Acknowledgements

We acknowledge the support by the German Bundesministerium für Bildung und Forschung within the project "RODES" (Förderkennzeichen 05M22UTB). Moreover, the authors thank the DFG for their support within the research training group 2126 "Algorithmic Optimization". The computations were executed on the high performance cluster "Elwetritsch" at the TU Kaiserslautern, which is part of the "Alliance of High Performance Computing Rheinland-Pfalz" (AHRP). We kindly acknowledge the support of RHRK. The authors acknowledge the use of DeepL and OpenAI's ChatGPT for partly editing and polishing the text and figures for spelling, grammar, and stylistic improvements. Additionally, ChatGPT was utilized for support in basic coding tasks.

References

Aigner, K.-M., J.-P. Clarner, F. Liers, and A. Martin (2022). "Robust approximation of chance constrained DC optimal power flow under decision-dependent uncertainty." In: *European Journal of Operational Research* 301.1, pp. 318–333. DOI: 10.1016/j.ejor.2021.10.051.

Apap, R. M. and I. E. Grossmann (2017). "Models and computational strategies for multistage stochastic programming under endogenous and exogenous uncertainties." In: *Computers & Chemical Engineering* 103, pp. 233–274. DOI: 10.1016/j.compchemeng.2016.11.011.

Avraamidou, S. and E. N. Pistikopoulos (2020). "Adjustable robust optimization through multi-parametric programming." In: *Optimization Letters* 14.4, pp. 873–887. DOI: 10.1007/s11590-019-01438-5.

Balas, E. (1971). "Intersection Cuts–A New Type of Cutting Planes for Integer Programming." In: *Operations Research* 19.1, pp. 19–39. DOI: 10.1287/opre.19.1.19.

Basciftci, B., S. Ahmed, and S. Shen (2021). "Distributionally robust facility location problem under decision-dependent stochastic demand." In: *European Journal of Operational Research* 292.2, pp. 548–561. DOI: 10.1016/j.ejor.2020.11.002.

Caprara, A., M. Carvalho, A. Lodi, and G. J. Woeginger (2014). "A Study on the Computational Complexity of the Bilevel Knapsack Problem." In: *SIAM Journal on Optimization* 24.2, pp. 823–838. DOI: 10.1137/130906593.

Conforti, M., G. Cornúejols, and G. Zambelli (2014). *Integer Programming*. Graduate Texts in Mathematics. Springer Cham. DOI: 10.1007/978-3-319-11008-0.

Cornúejols, G. (2008). "Valid inequalities for mixed integer linear programs." In: *Mathematical Programming* 112.1, pp. 3–44. DOI: 10.1007/s10107-006-0086-0.

DeNegre, S. T. and T. K. Ralphs (2009). "A Branch-and-cut Algorithm for Integer Bilevel Linear Programs." In: *Operations Research and Cyber-Infrastructur*. Springer, Boston, MA, pp. 65–78. DOI: 10.1007/978-0-387-88843-9_4.

Doan, X. V. (2022). "Distributionally robust optimization under endogenous uncertainty with an application in retrofitting planning." In: *European Journal of Operational Research* 300.1, pp. 73–84. DOI: 10.1016/j.ejor.2021.07.013.

Ederer, T., M. Hartisch, U. Lorenz, T. Opfer, and J. Wolf (2017). "Yasol: an open source solver for quantified mixed integer programs." In: *Advances in Computer Games: 15th International Conferences, ACG 2017, Leiden, The Netherlands, July 3–5, 2017, Revised Selected Papers 15*. Springer, pp. 224–233. DOI: 10.1007/978-3-319-71649-7_19.

Feng, W., Y. Feng, and Q. Zhang (2021). "Multistage distributionally robust optimization for integrated production and maintenance scheduling." In: *AIChE Journal* 67.9, e17329. DOI: 10.1002/aic.17329.

Fischetti, M., I. Ljubić, M. Monaci, and M. Sinnl (2017). "A new general-purpose algorithm for mixed-integer bilevel linear programs." In: *Operations Research* 65.6, pp. 1615–1637. DOI: 10.1287/opre.2017.1650.

– (2018). "On the use of intersection cuts for bilevel optimization." In: *Mathematical Programming* 172.1, pp. 77–103. DOI: 10.1007/s10107-017-1189-5.

– (2019). "Interdiction games and monotonicity, with application to knapsack problems." In: *INFORMS Journal on Computing* 31.2, pp. 390–410. DOI: 10.1287/ijoc.2018.0831.

Goerigk, M. and M. Hartisch (2021). "Multistage robust discrete optimization via quantified integer programming." In: *Computers & Operations Research* 135, p. 105434. DOI: 10.1016/j.cor.2021.105434.

Goerigk, M., J. Kurtz, M. Schmidt, and J. Thürauf (2025). "Connections between Robust and Bilevel Optimization." In: *Open Journal of Mathematical Optimization* 6, 2, pp. 1–17. DOI: 10.5802/ojmo.38.

Hartisch, M. and U. Lorenz (2022). *A general model-and-run solver for multistage robust discrete linear optimization*. URL: https://arxiv.org/abs/2210.11132.

Hellemo, L., P. I. Barton, and A. Tomasgard (2018). "Decision-dependent probabilities in stochastic programs with recourse." In: *Computational Management Science* 15.3, pp. 369–395. DOI: 10.1007/s10287-018-0330-0.

Jonsbråten, T. W., R. J. Wets, and D. L. Woodruff (1998). "A class of stochastic programs with decision dependent random elements." In: *Annals of Operations Research* 82, pp. 83–106. DOI: 10.1023/A:1018943626786.

Kleinert, T., M. Labbé, I. Ljubić, and M. Schmidt (2021). "A survey on mixed-integer programming techniques in bilevel optimization." In: *EURO Journal on Computational Optimization* 9, p. 100007. DOI: 10.1016/j.ejco.2021.100007.

Lappas, N. H. and C. E. Gounaris (2016). "Multi-stage adjustable robust optimization for process scheduling under uncertainty." In: *AIChE Journal* 62.5, pp. 1646–1667. DOI: 10.1002/aic.15183.

Lefebvre, H., M. Schmidt, S. Stevens, and J. Thürauf (2025). *A Computational Study for Solving Decision-Dependent Robust Problems as Bilevel Optimization Problems*. Tech. rep. URL: https://optimization-online.org/?p=29420.

Leyffer, S., M. Menickelly, T. Munson, C. Vanaret, and S. M. Wild (2020). "A survey of nonlinear robust optimization." In: *INFOR: Information Systems and Operational Research* 58.2, pp. 342–373. DOI: 10.1080/03155986.2020.1730676.

Lozano, L. and J. S. Borrero (2025). "On solving integer robust optimization problems with integer decision-dependent uncertainty sets." In: *Mathematical Programming*, pp. 1–33. DOI: 10.1007/s10107-025-02249-6.

Luo, F. and S. Mehrotra (2020). "Distributionally robust optimization with decision dependent ambiguity sets." In: *Optimization Letters* 14.8, pp. 2565–2594. DOI: 10.1007/s11590-020-01574-3.

Martello, S., D. Pisinger, and P. Toth (1999). "Dynamic Programming and Strong Bounds for the 0-1 Knapsack Problem." In: *Management Science* 45.3, pp. 414–424. DOI: 10.1287/mnsc.45.3.414.

McCormick, G. P. (1976). "Computability of global solutions to factorable non-convex programs: Part I–Convex underestimating problems." In: *Mathematical Programming* 10.1, pp. 147–175. DOI: 10.1007/BF01580665.

Moore, J. T. and J. F. Bard (1990). "The mixed integer linear bilevel programming problem." In: *Operations Research* 38.5, pp. 911–921. DOI: 10.1287/opre.38.5.911.

Motamed Nasab, F. and Z. Li (2021). "Multistage adaptive stochastic mixed integer optimization under endogenous and exogenous uncertainty." In: *AIChE Journal* 67.10, e17333. DOI: 10.1002/aic.17333.

Nohadani, O. and K. Sharma (2018). "Optimization under Decision-Dependent Uncertainty." In: *SIAM Journal on Optimization* 28.2, pp. 1773–1795. DOI: 10.1137/17m1110560.

Poss, M. (2013). "Robust combinatorial optimization with variable budgeted uncertainty." In: *4OR* 11, pp. 75–92. DOI: 10.1007/s10288-012-0217-9.

– (2014). "Robust combinatorial optimization with variable cost uncertainty." In: *European Journal of Operational Research* 237.3, pp. 836–845. DOI: 10.1016/j.ejor.2014.02.060.

Ryu, M. and R. Jiang (2025). "Nurse staffing under absenteeism: A distributionally robust optimization approach." In: *Manufacturing & Service Operations Management*. DOI: 10.1287/msom.2023.0398.

Spacey, S. A., W. Wiesemann, D. Kuhn, and W. Luk (2012). "Robust software partitioning with multiple instantiation." In: *INFORMS Journal on Computing* 24.3, pp. 500–515. DOI: 10.1287/ijoc.1110.0467.

Stein, O. (2013). *Bi-Level Strategies in Semi-Infinite Programming*. Springer New York, NY. DOI: 10.1007/978-1-4419-9164-5.

Tahernejad, S. and T. K. Ralphs (2025). *Valid Inequalities for Mixed Integer Bilevel Linear Optimization Problems*. URL: https://optimization-online.org/?p=16789.

Tahernejad, S., T. K. Ralphs, and S. T. DeNegre (2020). "A Branch-and-Cut Algorithm for Mixed Integer Bilevel Linear Optimization Problems and Its Implementation." In: *Mathematical Programming Computation* 12, pp. 529–568. DOI: 10.1007/s12532-020-00183-6.

Tahernejad, S. and T. K. Ralphs (2020). *Valid Inequalities for Mixed Integer Bilevel Linear Optimization Problems*. URL: https://coral.ise.lehigh.edu/~ted/files/papers/MibSCuts20.

Thürauf, J., T. Kleinert, I. Ljubić, T. Ralphs, and M. Schmidt (2024). "BOBILib: Bilevel Optimization (Benchmark) Instance Library." In: *Optimization Online Eprint*. URL: https://optimization-online.org/?p=27063.

Vujanic, R., P. Goulart, and M. Morari (2016). "Robust optimization of schedules affected by uncertain events." In: *Journal of Optimization Theory and Applications* 171, pp. 1033–1054. DOI: 10.1007/s10957-016-0920-3.

Woeginger, G. J. (2021). "The trouble with the second quantifier." In: *4OR. A Quarterly Journal of Operations Research* 19.2, pp. 157–181. DOI: 10.1007/s10288-021-00477-y.

Wood, R (2011). "Bilevel Network Interdiction Models: Formulations and Solutions." In: *Wiley Encyclopedia of Operations Research and Management Science* 174. DOI: 10.1002/9780470400531.eorms0932.

Yu, X. and S. Shen (2022). "Multistage distributionally robust mixed-integer programming with decision-dependent moment-based ambiguity sets." In: *Mathematical Programming* 196.1, pp. 1025–1064. DOI: 10.1007/s10107-020-01580-4.

Zeng, B. and W. Wang (2022). *Two-stage robust optimization with decision dependent uncertainty.* DOI: 10.48550/arXiv.2203.16484.

Zhan, Y., Q. P. Zheng, J. Wang, and P. Pinson (2016). "Generation expansion planning with large amounts of wind power via decision-dependent stochastic programming." In: *IEEE Transactions on Power Systems* 32.4, pp. 3015–3026. DOI: 10.1109/TPWRS.2016.2626958.

Zhang, Q. and W. Feng (2020). "A unified framework for adjustable robust optimization with endogenous uncertainty." In: *AIChe journal* 66.12, e17047. DOI: 10.1002/aic.17047.

Zhang, Y., Y. Su, and F. Liu (2025). "On Decision-Dependent Uncertainties in Power Systems with High-Share Renewables." In: *Engineering*. DOI: 10.1016/j.eng.2025.07.013.

Zhu, T., J. Xie, and M. Sim (2022). "Joint estimation and robustness optimization." In: *Management Science* 68.3, pp. 1659–1677. DOI: 10.1287/mnsc.2020.3898.

(M. Schmidt, S. Stevens) Trier University, Department of Mathematics, Universitätsring 15, 54296 Trier, Germany
Email address: martin.schmidt@uni-trier.de
Email address: stevens@uni-trier.de

(H. Lefebvre) LIRMM, University of Montpellier, CNRS, Montpellier, France
Email address: henri.lefebvre@cnrs.fr

(J. Thürauf) University of Technology Nuremberg (UTN), Department Liberal Arts and Social Sciences, Discrete Optimization Lab, Dr.-Luise-Herzberg-Str. 4, 90461 Nuremberg, Germany
Email address: johannes.thuerauf@utn.de