Robust Bilevel Optimization with a Wait-and-See Follower: A Column-and-Constraint Generation Approach

Henri Lefebvre, Johannes Thürauf

Abstract. We study optimistic robust bilevel problems with uncertainty in the follower's problem, where the follower adopts a so-called wait-and-see approach. In this setting, the leader decides without knowledge of the specific realization of the uncertainty. Then, the uncertainty realizes in a worst-case manner, and afterward the follower makes her own decisions. For this challenging problem class, we present a general solution approach based on column-and-constraint generation (CCG), which iteratively solves a bilevel master problem involving a single leader and multiple followers, along with (pessimistic) bilevel subproblems. We analyze structural properties of this problem class, including solution attainability, and show that worst-case realizations of the uncertainty need not occur at extreme points of the uncertainty set. Building on these insights, we establish finite termination and exactness for discrete uncertainty or finitely many upper-level decisions. Moreover, for polyhedral uncertainty sets, we prove that a solution of arbitrary precision can be computed in finitely many iterations. Finally, we demonstrate the applicability of the presented approach through an extensive computational study on two application domains, including both continuous and integer lower-level decisions.

1. Introduction

Bilevel optimization is an established and rapidly evolving field in mathematical optimization, originating from the seminal works of Stackelberg (1934, 1952) on leader-follower games. A bilevel problem typically involves two decision makers, referred to as the leader and the follower. The first decision maker (the leader) optimizes her objective function while anticipating the reaction of the second decision maker (the follower), who rationally determines her actions by solving an own optimization problem. On the one hand, this modeling framework enables the representation of hierarchical decision processes involving decision makers with potentially competing objectives. On the other hand, bilevel optimization problems are computationally challenging. Even linear bilevel optimization problems are known to be strongly NP-hard (Hansen et al. 1992; Jeroslow 1985) and if integer variables are present in the follower's problem, then it is Σ_2^P-hard (Caprara et al. 2014). However, numerous algorithmic advances over the past decades have made it possible to solve linear and mixed-integer linear bilevel problems of increasing size. We note that when multiple optimal follower's solutions exist, two standard notions of bilevel optimization are typically distinguished. In optimistic bilevel optimization, the leader assumes that the follower selects an optimal solution that is most favorable with respect to the leader's objective. In contrast, pessimistic bilevel optimization assumes that the follower chooses an optimal solution that is worst for the leader. For a comprehensive overview of these bilevel techniques, we refer to the survey by Kleinert et al. (2021) and the book by Dempe and Zemkoho (2020).

While significant progress has been made in solving bilevel optimization problems, research on bilevel optimization under uncertainty is still in its early stages. However,

Date: April 27, 2026.

2020 *Mathematics Subject Classification.* 90-XX, 90C11, 90C17, 90C57.

Key words and phrases. bilevel optimization, robust optimization, robust bilevel optimization, optimization under uncertainty, column-and-constraint generation, discrete optimization, continuous optimization.

it is well known that many decision-making processes are subject to data uncertainty. Compared to single level optimization, bilevel problems may involve multiple sources of uncertainty, as outlined in the recent survey Beck et al. (2023a). In addition to classic data uncertainty, there may also be so-called decision uncertainty, which arises from the interaction of two decision makers. For example, the leader or follower may be uncertain about the decisions of the other player; see e.g., Beck and Schmidt (2021), Besançon et al. (2024), and Zare et al. (2020).

In this paper, we focus on data uncertainty in optimistic bilevel problems, which is typically addressed using one of two main approaches: stochastic optimization or robust optimization. In bilevel optimization under uncertainty, existing contributions predominantly adopt stochastic approaches to address data uncertainties, typically assuming that the uncertain parameters follow a known probability distribution. In this setting, the leader accounts for uncertainty in a probabilistic manner by optimizing an expected value or a specific risk-measure. For a comprehensive overview on stochastic bilevel optimization, we refer to Section 3.1 of Beck et al. (2023a). However, in this work, we adopt the robust optimization approach to handle data uncertainties in optimistic bilevel problems. Consequently, we aim to compute a solution that is protected against all worst-case uncertainties within an a priori defined uncertainty set.

In the robust bilevel setting, it is important to differentiate the information available to the follower when making her decision, namely whether the realization of the uncertainty is observed before the decision or after. Accordingly, uncertain bilevel problems are categorized as having a "here-and-now" follower or a "wait-and-see" follower. In a here-and-now setup, uncertainty realizes after both the leader's and the follower's decisions, so that neither decision maker observes its realization before taking their decisions; see e.g., Beck et al. (2023a,b, 2025) and Chuong and Jeyakumar (2017). In contrast, in the considered wait-and-see setting, the follower observes the realization of uncertainty after the leader's decision but before making her own decision, i.e., the follower has full knowledge about the realization of the uncertainty. Hence, we consider the timing

$$\text{Leader's decision} \quad \curvearrowright \quad \text{Uncertainty} \quad \curvearrowright \quad \text{Follower's decision.}$$

We note that this wait-and-see setting is closely related to two-stage stochastic or two-stage robust optimization. In particular, Goerigk et al. (2025b) show that two-stage robust problems are contained in the class of robust bilevel problems with a wait-and-see follower. However, unlike in two-stage stochastic or robust optimization, bilevel optimization typically involves a leader and a follower with distinct objectives.

In this paper, we study robust bilevel problems with a wait-and-see follower, where the leader's feasible region is given by a a mixed-integer linear formulation and the follower solves a (possibly mixed-integer) linear problem. We further assume that the uncertainty set is compact and either polyhedral or discrete, as common in robust optimization. Moreover, we focus on uncertainty affecting the right-hand side of the follower's problem and explicitly assume that the leader's decision also affect the feasible region of the follower.

Literature Review. In general, the literature on robust bilevel optimization with a wait-and-see follower is rather scarce. This is particularly striking given the broad range of applications, including energy systems, interdiction problems, defender–attacker games, and management science; see Beck et al. (2025). Kis et al. (2025) present an ε-approximation approach for pessimistic robust problems with a wait-and-see follower. Specifically, they study uncertain bilevel problems in which only the follower's objective is subject to polyedral uncertainty and the follower's feasible region does not depend on the leader's decision. Moreover, they show that this specific class of uncertain bilevel problems is Σ_2^p-hard. However, these results do not directly extend to the class of robust bilevel problems considered here, in which the follower's constraints are affected by both the leader's decision and the uncertainty. Zeng et al. (2020) study a vehicle charging problem and develop an application-specific column-and-constraint generation (CCG) algorithm. In a very recent contribution, Hu et al. (2026) develops an application-specific CCG algorithm for two-stage robust bilevel facility location problems, based on the PhD thesis by Xu (2023). The latter was recently made publicly available, during the preparation of this article, and presents a general CCG framework for two-stage robust bilevel optimization, closely related to the approach studied here. However, the theoretical and computational analysis of the algorithm substantially differs from our paper. More specifically, Xu (2023) shows that the generated sequence of iterates contains a subsequence converging to an ε-optimal solution, provided that the lower-level optimal value function is continuous in the leader's decision and in the uncertain parameters. In contrast, our analysis shows that the algorithm finitely terminates with an ε-optimal solution for any given precision $\varepsilon > 0$ for polyhedral uncertainty sets, using completely different proof techniques. Moreover, we consider the important case of mixed-integer linear lower-level problems, for which the continuity assumption on the lower-level optimal-value function does not hold. For this setting, we show that the algorithm computes a globally optimal solution within a finite number of iterations under common assumptions. Consequently, we give significantly stronger algorithmic guarantees in the considered setting. Finally, to the best of our knowledge, we are the first to consider both settings (continuous and integer lower-level problems) from a computational perspective.

From the perspective of computational complexity, different variants of robust bilevel problems with wait-and-see followers have been studied. For objective uncertainty, Buchheim et al. (2021) show that robust bilevel problems can be Σ_2^p-hard under interval uncertainty and the independence assumption, even when the deterministic problem is NP-easy. But, the same robust problem remains NP-easy under discrete uncertainty. These results are complemented by Henke (2025), who investigate the robust bilevel selection problem with an uncertain follower objective and propose a 2-approximation algorithm under specific conditions. Moreover, Buchheim and Henke (2022) study the pessimistic bilevel continuous knapsack problem with an uncertain follower objective.

When the leader and the follower have the same objective function, robust bilevel optimization problems reduce to two-stage robust problems; see Goerigk et al. (2025b). For this class of problems, the CCG algorithm is one of most prominent methods used in the literature. It was introduced by Zeng and Zhao (2013) for problems with continuous second-stage decisions and has since been extended to various settings such as integer second-stage decisions (Zeng and Zhao 2012), or discrete uncertainty sets (Lefebvre and Subramanyam 2025; Subramanyam 2022). In Goerigk et al. (2025a), the authors study several techniques to speed up the solution process. Specifically, they focus on reducing the computation time needed to generate new scenarios. On the other hand, Lefebvre et al. (2026) addresses the burden of solving the master problem when the number of scenarios becomes large by using column generation. Finally, an "inexact" CCG algorithm is presented by Tsang et al. (2023) in which the solutions to the master problem are allowed to be inexact.

Contributions. In the light of this scarce literature on robust bilevel optimization with a wait-and-see follower, we summarize our main contributions for the presented CCG-approach and considered problem class as follows:

(i) We investigate structural properties, including the attainability of solutions, and show that worst-case realizations of the uncertainty do not necessarily occur at extreme points of the uncertainty set (Example 3), in contrast to classical two-stage robust optimization.

(ii) For both polyhedral or discrete uncertainty sets, as well as the setting of finitely many upper-level decisions, we establish finite termination of the presented algorithm. More precisely, in the case of discrete uncertainty or finitely many upper-level decisions, we show that an optimal solution is computed in finitely many steps. For polyhedral uncertainty, we prove that a solution of arbitrary precision is obtained within finitely many iterations. The latter result is particularly noteworthy, as standard arguments for finite convergence in two-stage robust optimization cannot be directly extended, due to the fact that worst-case realizations of the uncertainty need not occur at extreme points of the uncertainty set.

(iii) We demonstrate the practical applicability of different variants of the CCG approach through an extensive computational study across multiple application domains, including an in-depth analysis of the individual components of the CCG-scheme. To the best of our knowledge, we are the first to consider settings with both continuous and integer lower-level decisions.

(iv) Finally, we provide an open-source implementation of the proposed CCG approach within the idol library (Lefebvre 2025). To the best of our knowledge, this is the first publicly available implementation for solving robust bilevel problems with a wait-and-see follower.

The remainder of this paper is organized as follows. In Section 2, we formally introduce the considered class of robust bilevel problems with a wait-and-see follower and analyze the attainability of solutions. In Section 3, we present the proposed CCG algorithm and prove its correctness in Section 4. Moreover, we present efficient reformulations for solving the arising adversarial subproblems in Section 5. This is followed by an extensive computational study in Section 5. We conclude the paper, we final remarks and possible directions for future research in Section 6.

2. Problem Statement and Basic Properties

Formally, the mixed-integer linear robust bilevel problem with a wait-and-see follower reads as

$$\inf_{x\in X} \left\{ c_x^\top x + \sup_{u\in U} \inf_{y\in S(x,u)} c_y^\top y \right\} \tag{1}$$

with $S(x,u)$ being the set of solutions to the (x,u)-parameterized lower-level problem

$$\inf_{y} \quad d^\top y \tag{2a}$$

$$\text{s.t.} \quad C(u)x + Dy \geq b(u), \tag{2b}$$

$$y \in \mathbb{R}^{n_y - p_y} \times \mathbb{Z}^{p_y}. \tag{2c}$$

Here, $X := \{x \in \mathbb{R}^{n_x - p_x} \times \mathbb{Z}^{p_x} : Ax \geq a\}$ denotes the upper-level feasible region. In line with the literature about robust bilevel optimization with a wait-and-see-follower, we do not consider coupling constraints in (1); see, e.g., Beck et al. (2023a). Further, $U \subset \mathbb{R}^{n_u}$ represents the uncertainty set. For given $(x,u) \in X \times U$, we denote the set of feasible points to the follower's problem by $Y(x,u) := \{y \in \mathbb{R}^{n_y - p_y} \times \mathbb{Z}^{p_y} : (2b)\}$.

Note that we consider uncertainties in the the right-hand side b as well as in the matrix C of the lower-level problem. This is expressed by the dependency of b and C on u, which are mappings into their corresponding spaces. We further note that both the upper- and lower-level problems may contain integer decisions.

Throughout this paper, we make the following standing assumption.

Standing Assumption 1. *The sets X, U, and $\{(x,y,u) : x \in X, u \in U, y \in Y(x,u)\}$ are compact sets and the mappings $C(\cdot)$ and $b(\cdot)$ are continuous.*

In particular, this assumption ensures that Problem (1) attains its infimum in at least two prominent cases, which we discuss in the next theorem. To this end, we introduce the notations

$$\begin{aligned} Q(x) &:= \sup_{u \in U} \ \xi(x,u), \\ \xi(x,u) &:= c_x^\top x + \inf_{y \in S(x,u)} \ c_y^\top y, \ \text{and}, \\ \varphi(x,u) &:= \inf_{y \in Y(x,u)} \ d^\top y. \end{aligned}$$

Hence, Q is the optimal-value function of the supremum in Problem (1), ξ is the optimal-value function of the inner-infimum in Problem (1), and φ is the optimal-value function of the lower-level problem. With a small abuse of notation, we refer to Problem $Q(x)$ (resp., $\xi(x,u)$) as the optimization problem whose optimal-value function is Q (resp., ξ). From Assumption 1, it directly follows that, for given (x,u), if the lower-level problem is feasible, then $\varphi(x,u)$ is finite and the optimal value is attained. We now show sufficient conditions under which attainability is also guaranteed for the robust bilevel problem and $Q(x)$.

Theorem 1 (Attainability)**.** *Let at least one of the following conditions hold.*

(A1) There is no integer lower-level decision, i.e., $p_y = 0$, or,
(A2) the upper-level feasible region X and the uncertainty set U are discrete.

Then, if the respective problem is feasible, the following statements hold:

(i) for all $(x,u) \in X \times U$, Problem $\xi(x,u)$ attains its infimum;
(ii) for all $x \in X$, Problem $Q(x)$ attains its supremum;
(iii) Problem (1) attains its infimum.

Proof. First, note that $S(x,u)$ is a compact set for all $(x,u) \in X \times U$ since it is the set of solutions to an MILP whose feasible region is compact. Hence, by the theorem of Weierstrass, Problem $\xi(x,u)$ attains its infimum, if it is feasible. We now have two cases.

Assume that there are no integer lower-level decisions. First, by Berge's maximum theorem (Aliprantis and Border 2006, Section 17.31) φ is a continuous function since it is the optimal-value function of an LP parameterized by a continuous function of u. Therefore, it follows that

$$\xi(x,u) = c_x^\top x + \inf_y \ c_y^\top y \quad \text{s.t.} \quad y \in Y(x,u), \ d^\top y \le \varphi(x,u)$$

is also continuous, by the same arguments. Thus, Problem $Q(x)$ attains its supremum, if it is feasible, by the theorem of Weierstrass. Finally, Q is a continuous function since it is the maximum of continuous functions over a compact set. Hence, by the theorem of Weierstrass, Problem (1) attains its infimum, if it is feasible.

Otherwise, assume that both X and U are discrete. Then, Problem $Q(x)$ attains its supremum, if feasible, since the finite maximum can be computed by enumeration. Similarly, Problem (1) attains its infimum, if it is feasible, since it is a finite minimum. □

However, there are also cases in which Problem (1) does not attain its infimum. In the following remark, we discuss this according to corresponding results in bilevel optimization. Moreover, we highlight similarities and differences compared to two-stage robust optimization, which is a special case of the robust bilevel problems considered here. A complete overview on attainability results is then reported in Table 1.

Remark 1 (Unattainability)**.** *Köppe et al. (2010) show that deterministic bilevel optimization problems with continuous upper-level decisions and integer lower-level decisions may not attain their infimum. Since Problem (1) is a generalization of such problems by choosing, e.g., U as a singleton, it follows that it may not attain its infimum in case the upper-level feasible region X is a polytope and some lower-level decisions are integer.*

We note that Problem (1) reduces to a two-stage robust optimization problem by setting $d = c_y$. In this setting, Subramanyam et al. (2019) show that Problem $Q(x)$

TABLE 1. Attainability of the infimum for Problem (1).

	Continuous Uncertainty		Discrete Uncertainty	
	Continuous X	Discrete X	Continuous X	Discrete X
Continuous Y	✓	✓	✓	✓
Discrete Y	✗	✓*	✗	✓

*The overall problem attains its infimum, but Problem $Q(x)$ does not attain its supremum.

may not attain its supremum if the uncertainty set is a polytope and the second stage problem contains integer conditions. Nevertheless, the authors show that the overall two-stage robust problem attains its infimum. This stands in contrast to the considered robust bilevel setting. Attainability of Problem (1) *cannot be guaranteed when the upper-level feasible region contains continuous variables since it reduces to the first point of the remark regarding the results of Köppe et al. (2010).*

However, attainability for the robust bilevel problem (1) *is ensured, if X is discrete, since the problem reduces to a finite minimum.*

We note that under a continuity assumption on the lower-level optimal-value function, Xu (2023) (Theorem 4.5) reports sufficient conditions for solution attainability of robust bilevel problems, which additionally require the attainability of solutions to auxiliary optimization problems. In contrast, our complete characterization of attainability solely depends on the problem data.

The ongoing discussion on attainability is the main motivation for the second standing assumption that ensures that every sub-problem considered in this paper attains its infimum or supremum. Accordingly, we now directly write min (resp., max) instead of inf (resp., sup).

Standing Assumption 2. *There are no integer lower-level decisions (i.e., $p_y = 0$), or both the upper-level feasible region X and the uncertainty set U are discrete.*

Finally, we highlight that we do not assume that the lower-level problem is feasible for all $x \in X$ and all $u \in U$. This assumption is known as the "complete recourse" assumption in two-stage robust optimization; see, e.g., Zeng and Zhao (2013).

3. A Column-And-Constraint Generation Algorithm

In this section, we present a CCG algorithm to solve the robust bilevel problem (1). The overall idea of the algorithm follows a typical CCG-scheme. Initially, a finite subset of uncertain parameter realizations is considered in a so-called master problem. The master problem provides a candidate upper-level decision together with an auxiliary variable that estimates the worst-case value of the lower-level problem over the current subset of scenarios. Since this subset is only a restriction of the full uncertainty set, the master problem is a relaxation of the original robust bilevel problem and yields a lower bound on the optimal value of Problem (1).

Given the candidate upper-level decision obtained from the master problem, a subproblem is then solved to identify the worst-case realization of the uncertain parameters within the full uncertainty set. This realization is used to update the best known upper bound on Problem (1). If the optimality gap is closed, the algorithm terminates and returns an optimal solution to Problem (1). Otherwise, the newly identified realization of the uncertainty is added to the master problem, and the procedure is repeated.

In the following sections, we provide a more detailed description of the master problem, the subproblems, and the complete algorithm.

3.1. **The Master Problem.** Given a finite subset of scenarios $U^t \subseteq U$, the master problem is given by

$$
\begin{aligned}
\min_{x,\eta,y_u} \quad & \eta && \text{(3a)}\\
\text{s.t.} \quad & x \in X, && \text{(3b)}\\
& \eta \geq c_x^\top x + c_y^\top y_u, \quad \text{for all } u \in U^t, && \text{(3c)}\\
& y_u \in S(x,u), \quad \text{for all } u \in U^t. && \text{(3d)}
\end{aligned}
$$

We emphasize that, in the considered bilevel context, one has to ensure that y_u is a solution to the (x,u)-parameterized lower-level problem (2), as imposed by Constraints (3d). Consequently, Problem (3) is a bilevel problem with a single leader and multiple independent followers: for each uncertainty $u \in U^t$, an optimal follower's response y_u has to be determined. Solving the master problem (3) leads to an upper-level decision x^t that is feasible for the subset of uncertainties $U^t \subseteq U$. However, this upper-level decision x^t may not be optimal or even feasible for the original uncertainty set U. The following proposition is straightforward.

Proposition 1. *If the master problem (3) is infeasible, then so is the robust bilevel problem (1). Otherwise, its optimal value is a lower bound on that of Problem (1).*

3.2. **The Subproblems.** Let us assume that the master problem (3) is feasible and let (x^t, η^t) denote a (projected) optimal point. As anticipated, x^t is not necessarily feasible nor optimal for the overall problem (1), in which the full uncertainty set is considered. To verify that this is the case, we solve two subproblems, which we describe next.

3.2.1. *Verifying Feasibility.* First, x^t may be feasible for the master problem but infeasible for the overall problem (1). To verify this, we solve the subproblem

$$
\max_{u\in U} \min_{y,s} \sum_{i\in I} s_i \quad \text{s.t.} \quad s + C(u)x^t + Dy \geq b(u), \; s \geq 0, \quad y \in \mathbb{R}^{n_y - p_y} \times \mathbb{Z}^{p_y}. \tag{4}
$$

This max-min bilevel problem aims to identify a scenario for which some lower-level constraints cannot be satisfied. Thus, if the optimal objective value of Problem (4) is positive, then there exists a scenario $u^t \in U$ so that for the given upper-level decision x^t, there is no feasible lower-level decision y. Consequently, x^t is not a feasible point of the robust bilevel problem (1). Otherwise, it holds that x^t is feasible for Problem (1). This is formalized in the following statement.

Proposition 2. *Let $x^t \in X$ be given. Then, x^t is feasible for Problem (1) if and only if the optimal objective function value of Problem (4) is zero.*

Proof. Assume that $x^t \in X$ is feasible for Problem (1). Then, for every uncertainty $u \in U$, there exists a solution $y(u)$ to the lower-level problem (2). It follows that $(y(u), 0)$ is feasible for the lower-level problem in (4). Since the corresponding objective value is zero and $s \geq 0$, this solution is optimal. Hence, the optimal objective value of Problem (4) is zero.

Conversely, assume that the optimal objective value of Problem (4) is zero. Then, for every uncertainty realization $u \in U$, there exists a feasible lower-level solution y. Under our standing assumptions, in particular the compactness of the lower-level feasible region, this implies that an optimal follower's response exists for every $u \in U$. Consequently, x^t is feasible for Problem (1). □

Remark 2. *We note that the subproblem (4) coincides with the one used in the CCG algorithm for two-stage robust optimization, see, e.g., Flambard et al. (2025). Indeed, due to the compactness assumption on the lower-level feasible region, the lower-level problem has a solution if and only if it is feasible.*

3.2.2. *Verifying Optimality.* Assume that the current master solution (x^t, η^t) is such that x^t is feasible for the robust bilevel problem (1), i.e., that for all $u \in U$, there exists $y \in S(x, u)$. Then, it remains to check if x^t is optimal for Problem (1). To this end, we solve the subproblem

$$\max_{u \in U} \min_{y} \ c_x^\top x^t + c_y^\top y \quad \text{s.t.} \quad y \in S(x^t, u). \tag{5}$$

Here, $S(x, u)$ is again the set of solutions to the (x, u)-parameterized lower-level problem (2). Note that this problem corresponds to evaluating the objective function of Problem (1) at the point x^t. Thus, the following proposition directly follows.

Proposition 3. *Let $x^t \in X$ be a feasible point for Problem (1). Then, the optimal objective function value of Problem (5) is an upper bound on that of the robust bilevel problem (1).*

Remark 3. *We emphasize an important observation, which is also key for solving the bilevel problem (5); see Section 5. Problem (5) can be interpreted as a pessimistic bilevel problem in which the upper-level decisions are the uncertainty vector u and the lower-level decisions are the vector y. The intuition for formulating the adversarial problem as a pessimistic bilevel problem is based on the following observation. Since we consider an optimistic robust bilevel problem, the follower is generally cooperating with the leader. In particular, if there are multiple solutions to the lower-level problem ($|S(x, u)| > 1$), then the follower selects the one that is most favorable for the leader, i.e., the one that minimizes $c_y^\top y$. In contrast, the uncertainty always considers the worst-case regarding the leader's objective. Hence, the follower and the uncertainty pursue opposing interests: while the follower acts in favor of the leader, the uncertainty acts against it. Combining these aspects yields the pessimistic bilevel problem (5), in which the follower chooses an optimal response beneficial to the leader and thereby simultaneously counteracts the worst-case behavior of the uncertainty.*

3.3. **The Algorithm.** Combining the master problem (3), the feasibility subproblem as well as the optimality subproblem (5) leads to the CCG algorithm reported in Algorithm 1.

Algorithm 1 CCG Algorithm for the Robust Bilevel Problem (1)

1: **Input**: An initial non-empty finite set $U^0 \subset U$ and a tolerance $\varepsilon \geq 0$.
2: Set $t \leftarrow 1$, $\text{LB}^0 \leftarrow -\infty$, and $\text{UB}^0 \leftarrow \infty$.
3: **while** $\text{UB}^{t-1} - \text{LB}^{t-1} > \varepsilon$ **do**
4:     Solve the master problem (3).
5:     **if** it is infeasible **then**
6:         **return** "Robust bilevel problem (1) is infeasible.".
7:     **end if**
8:     Let $(x^t, \eta^t, y^1, \dots, y^{|U^t|})$ denote a solution to the master problem (3).
9:     Set $\text{LB}^t \leftarrow \eta^t$.
10:     Solve the feasibility subproblem (4) and let $(\tilde{u}^t, \tilde{y}^t, \tilde{s}^t)$ denote a solution.
11:     **if** $e^\top \tilde{s}^t > 0$ **then**
12:         Set $U^{t+1} \leftarrow U^t \cup \{\tilde{u}^t\}$ and $t \leftarrow t + 1$.
13:         **go to** Line 3.
14:     **end if**
15:     Solve the optimality subproblem (5) and let $(\hat{u}^t, \hat{y}^t)$ denote a solution.
16:     Set $\text{UB}^t \leftarrow \min\{\text{UB}^{t-1}, c_x^\top x^t + c_y^\top \hat{y}^t\}$.
17:     Set $U^{t+1} \leftarrow U^t \cup \{\hat{u}^t\}$ and $t \leftarrow t + 1$.
18: **end while**
19: **return** x^{t-1}, which is a solution to the robust bilevel problem (1).

Algorithm 1 requires a non-empty finite set of scenarios $U^0 \subseteq U$ and a tolerance $\varepsilon \geq 0$. The former is required so that the master problem (3), if feasible, attains an optimal solution. The tolerance is used to terminate the algorithm when a sufficiently small optimality gap is reached. Initially, the lower and upper bounds are initialized to $-\infty$ and $+\infty$, respectively.

Then, in each iteration t, we first solve the master problem (3) to obtain a lower bound as well as a candidate upper-level decision x^t. If the master problem is infeasible, then this also applies to the original robust bilevel problem; see Proposition 1. Otherwise, we update the lower bound and check whether x^t is feasible for Problem (1) by solving the feasibility subproblem (4). If this is not the case, we augment the scenario set U^t with the obtained worst-case uncertainty realization $\tilde{u}^t$, which cuts off the infeasible upper-level decision x^t in the master problem (3), and start a new iteration.

Otherwise, x^t is feasible for Problem (1), and we compute an upper bound by solving the adversarial subproblem (5). The scenario obtained from this subproblem is then added to the uncertainty set U^t before starting the next iteration. The stopping criterion ensures that the algorithm terminates when the optimality gap satisfies $\text{UB}^t - \text{LB}^t \leq \varepsilon$.

Remark 4. *We note that Algorithm 1 is also presented in the PhD thesis by Xu (2023), which was very recently made publicly available. The author shows that the generated sequence of iterates contains a subsequence that converges to an ε-optimal solution, provided that the lower-level optimal-value function is continuous in both the upper-level decisions and the uncertainty. Complementing this result, we prove that the algorithm terminates within finitely many iterations for polyhedral uncertainty sets using fundamentally different techniques. Moreover, we consider the case of mixed-integer linear lower-level problems for which the continuity assumption is violated. In this setting, we show finite termination and exactness of the algorithm.*

4. Correctness and Finite Termination

In this section, we study the correctness and finite termination of the CCG algorithm presented in Algorithm 1. We start with the case of discrete uncertainty and show that the algorithm computes a solution to the robust bilevel problem (1) in a finite number of iterations. For polyhedral uncertainty sets, we show that the algorithm computes an ε-optimal solution in finitely many iterations, for any given tolerance $\varepsilon > 0$. Finally, we further show that if the upper-level decisions are discrete, then Algorithm 1 finitely terminates even with $\varepsilon = 0$.

4.1. Discrete Uncertainty.

If the uncertainty set is discrete, Algorithm 1 terminates after a finite number of iterations. The proof is straightforward since a new scenario is generated at each iteration and $|U| < \infty$. This is formally stated in the next theorem.

Theorem 2. *Let the uncertainty set U be a discrete set. Then, Algorithm 1 with tolerance $\varepsilon = 0$ terminates after a finite number of iterations and either returns a solution to the robust bilevel problem (1) or correctly certifies its infeasibility.*

Proof. If Algorithm 1 returns infeasible, it directly follows that the robust bilevel problem itself is infeasible, since the master problem (3) is solved w.r.t. $U^t \subseteq U$.

We now contrarily assume that Algorithm 1 does not terminate. Then, the master problem (3) has to be feasible for all $t \geq 1$. Note that the condition in Line 11 can only be satisfied finitely many times. Whenever it holds in some iteration t, the constraint "$y_u \in S(x, \tilde{u}^t)$" is added to the master problem. Consequently, for any x^j with $j > t$, there exists $y' \in S(x^j, \tilde{u}^t)$. This implies that $(\tilde{u}^t, y')$ has an objective function value of zero in the x^j-parameterized feasibility subproblem (4). Consequently, the condition in Line 11 can be satisfied at most $|U| < \infty$ many times. Thus, Line 17 has to be visited an infinite number of times. Since the uncertainty set is discrete and bounded, there has to exist iterations t and j with $j > t$ such that $\hat{u}^t = \hat{u}^j$. Due to $\hat{u}^t \in U^j$, the inequality $\text{LB}^j \geq \xi(x^j, \hat{u}^t) = \xi(x^j, \hat{u}^j)$ with $\xi(x, u) = c_x^\top x + \min_{y \in S(x,u)} c_y^\top y$ has to hold. By the construction of Algorithm 1, we further obtain $\text{UB}^j \leq \xi(x^j, \hat{u}^j)$.

Consequently, $\mathrm{LB}^j = \mathrm{UB}^j$ holds and the algorithm stops after iteration j, which contradicts the assumption that Algorithm 1 never stops. The correctness is an immediate consequence of the construction of Algorithm 1. ☐

4.2. **Polyhedral Uncertainty.** We now turn to the case of polyhedral uncertainty sets. To this end, we make the following assumption.

Assumption 1. *The uncertainty set U is polyhedral and $C(\cdot)$ and $b(\cdot)$ are affine.*

We note that this assumptions implies that we consider a linear lower-level problem in this section due to the standing assumption 2.

To show the correctness and finite termination of Algorithm 1, we proceed as follows. First, we characterize the worst-case uncertainty realizations obtained by solving the feasibility subproblem (4). Second, we show that the function $\xi(\cdot, u)$ is Lipschitz continuous. Finally, we use these results to show that Algorithm 1 computes a solution to Problem (1) up to a prescribed tolerance within a finite number of iterations.

We start by analyzing the solutions to the feasibility subproblem (4). In the next lemma, we show that these can be taken to be extreme points of U without loss of generality.

Lemma 1. *Let Assumptions 1 hold. Then, for any $x \in X$, the feasibility subproblem problem (4) has a solution that is an extreme point of the uncertainty set U.*

Proof. Let $x \in X$ be given. Then, for any $u \in U$, the inner minimization problem in (4) is feasible and bounded from below. Hence, it has the same optimal objective function value as its dual problem. The dual problem reads

$$\max_{u \in U} \max_{\lambda} \lambda^\top (b(u) - C(u)x) \quad \text{s.t.} \quad D^\top \lambda = 0,\ 0 \leq \lambda \leq 1.$$

Let $\bar{\lambda}^1, \ldots, \bar{\lambda}^K$ denote the finite list of extreme points of the dual feasible region, which is independent of x. It follows that problem (4) can be expressed as

$$\max_{u \in U} \max_{k=1,\ldots,K} (\bar{\lambda}^k)^\top (b(u) - C(u)x).$$

Hence, by Assumption 1, the objective function is a finite maximum of affine functions, which is convex. Thus, problem (4) reduces to maximizing a convex function which implies that at least one solution is an extreme point of U. ☐

Note that Lemma 1 is well-known in the two-stage robust optimization literature; see, e.g., Takeda et al. (2007). We show in the next example, however, that this property does not carry over to the optimality separation problem (5) in the context of robust bilevel optimization.

Example 3 (Interior Solutions to Problem $Q(x)$)**.** *Consider the robust bilevel problem*

$$\min_{x \in [-1,1]} \max_{u \in [-2,2]} \min_{y \in S(x,u)} -y,$$

where the set $S(x,u)$ is given by

$$S(x,u) := \arg\min\{y : y \geq -u - x, y \geq u + x\}.$$

It is clear that $S(x,u) = \{|u + x|\}$ and, thus, $\xi(x,u) = -|u + x|$ holds. Thus, ξ is not a convex function in (x,u). Moreover, it follows that

$$Q(x) = \max_{u \in [-2,2]} -|x + u|$$

has a unique solution given by $u^* = -x$. Since $X := [-1,1] \subset [-2,2] =: U$, it follows that u^* cannot be an extreme point of U, for any x.

This example shows that worst-case uncertainty realizations in Problem $Q(x)$ do not necessarily occur at extreme points of the uncertainty set. Consequently, standard finite termination arguments for CCG algorithms in two-stage robust optimization that rely on extreme points of the uncertainty set (Zeng and Zhao 2013) cannot be employed in the robust bilevel context. Instead, we exploit the Lipschitz continuity property of Q, which we show next, to show finite termination. Before we do so, we first recall a result from the parametric linear optimization literature.

Theorem 4 (Corollary 5.1 and 5.2 in Still (2018))**.** *Let $F(t) := \{x \in \mathbb{R}^n : Ax \geq t\}$ and $v(t) := \min\{c^\top x : x \in F(t)\}$ and assume that $v(t) > -\infty$. Then,*

(i) there exists $L > 0$ such that for any $t, t' \in \operatorname{dom}(F) := \{t : F(t) \neq \emptyset\}$ it holds: to any $x \in F(t)$ one can find a point $x' \in F(t')$ such that

$$\|x - x'\| \leq L\|t - t'\|;$$

(ii) there exists $L' > 0$ such that for any $t, t' \in \operatorname{dom}(F)$ it holds:

$$|v(t) - v(t')| \leq L'\|t - t'\|.$$

Using Theorem 4, we now show that $\xi(\cdot, u)$ is a Lipschitz continuous function for any fixed $u \in U$.

Lemma 2. *There exists $L_\xi > 0$ such that, for all $u \in U$, the optimal-value function $\xi(\cdot, u)$, given by $\xi(x, u) = c_x^\top x + \min_{y \in S(x,u)} \ c_y^\top y$, is Lipschitz continuous with constant L_ξ, i.e.,*

$$|\xi(x, u) - \xi(x', u)| \leq L_\xi\|x - x'\|$$

holds for all $x, x' \in \operatorname{dom}(\xi(\cdot, u))$.

Proof. Let $u \in U$ be fixed and consider any $x, x' \in \operatorname{dom}(Y(\cdot, u))$. Using the optimal-value function of the lower-level φ, we can express $\xi(x, u)$ as

$$\xi(x, u) = c_x^\top x + \min_y \ c_y^\top y \quad \text{s.t.} \quad Dy \geq b(u) - C(u)x, \ d^\top y \leq \varphi(x, u).$$

Note that this is a linear problem for any given (x, u). Moreover, since $x \in \operatorname{dom}(Y(\cdot, u))$, it is feasible. Hence, by Theorem 1, there exists $y \in Y(x, u)$ such that $c_x^\top x + c_y^\top y = \xi(x, u)$. Now, the first part of Theorem 4 implies that there exists $y' \in Y(x', u)$ such that

$$\begin{aligned} \|y - y'\| &\leq L \left\| \begin{bmatrix} C(u)(x - x') \\ \varphi(x, u) - \varphi(x', u) \end{bmatrix} \right\| \\ &\leq L(\|C(u)(x - x')\| + \|\varphi(x, u) - \varphi(x', u)\|) \\ &\leq L(\|C(u)\|\|x - x'\| + \|\varphi(x, u) - \varphi(x', u)\|) \end{aligned} \tag{6}$$

holds. Note that we used $\|(\alpha, \beta)\| \leq \|\alpha\| + \|\beta\|$ and sub-multiplicativity of norms to obtain the two last inequalities.

Next, the second part of Theorem 4 applied to the linear lower-level problem (2) implies that there exists $L' > 0$ that satisfies

$$\begin{aligned} |\varphi(x, u) - \varphi(x', u)| &\leq L'\|b(u) - C(u)x - (b(u) - C(u)x')\| \\ &= L'\|C(u)(x' - x)\| \\ &\leq L'\|C(u)\|\|x - x'\|. \end{aligned}$$

Combining these inequalities with (6), we obtain

$$\|y - y'\| \leq L\|C(u)\|\|x - x'\| + L'\|C(u)\|\|x - x'\| = (L + L')\|C(u)\|\|x - x'\|.$$

By compactness of U and continuity of C, the Weierstrass theorem shows that

$$L'' := \max_u\{(L + L')\|C(u)\| : u \in U\} < \infty.$$

Hence, $\|y - y'\| \leq L''\|x - x'\|$ holds. To conclude, we use $c_x^\top x + c_y^\top y = \xi(x, u)$ and $y' \in Y(x', u)$ to obtain

$$\begin{aligned} \xi(x', u) - \xi(x, u) &\leq c_x^\top(x' - x) + c_y^\top(y' - y) \\ &\leq \|c_x\|\|x - x'\| + \|c_y\|\|y - y'\| \\ &\leq (L'' + 1)\|c\|\|x - x'\|. \end{aligned}$$

Interchanging the role of x and x' leads to the claimed result with $L_\xi = (L'' + 1)\|c\|$. ∎

We are now ready to show finite termination.

Theorem 5. *Let Assumptions 1 hold and let $\varepsilon > 0$ be any given precision. Then, Algorithm 1 terminates after finitely many iterations with a feasible point whose objective function value differs at most by ε to a solution of the robust bilevel problem (1), or correctly identifies it as infeasible.*

Proof. We first show that Algorithm 1 terminates after a finite number of iterations. By contradiction, let's assume that this is not the case. Then, the master problem (3) must be feasible and

$$\mathrm{UB}^t - \mathrm{LB}^t > \varepsilon \tag{7}$$

holds for all $t \geq 1$, since otherwise the algorithm stops. Also note that the condition in Line 11 cannot be satisfied an infinite number of times. Indeed, by Lemma 1, the feasibility subproblem always has a solution which is an extreme point of U. Thus, the number of extreme points of U, which is finite, is an upper bound on the number of times in which this condition is satisfied. Therefore, it cannot repeat infinitely many times.

We now argue that Line 17 cannot be repeated an infinite number of times either. Note that the sequence of lower bounds $(\mathrm{LB}^t)_{t\in\mathbb{N}}$ is non-decreasing and bounded above by, e.g., the optimal value of Problem (1). Hence, it is converging to some limit noted LB^*. Thus, there exists a subsequence of iterations, $\bar{T}$, such that $|\mathrm{LB}^t - \mathrm{LB}^*| < 1/2\varepsilon$ holds for all $t \in \bar{T}$. More specifically, for any $t \in \bar{T}$ and $j \in \bar{T}$ with $j > t$, $|\mathrm{LB}^t - \mathrm{LB}^j| < 1/2\varepsilon$ is satisfied as well.

Now, if the algorithm reaches Line 17 in iteration $t \in \bar{T}$, x^t is a feasible point of the robust bilevel problem (1). Thus, Inequality (7) implies that

$$\max_{u\in U} \ \xi(x^t, u) - \max_{u\in U^t} \ \xi(x^t, u) > \varepsilon. \tag{8}$$

Moreover, since $\hat{u}^t$ is added to the finite subset of scenarios in Line 17, we have that $\hat{u}^t \in U^j$ holds for all $j > t$ with $j \in \bar{T}$, implying that

$$\xi(x^j, \hat{u}^t) \leq \max_{u\in U^j} \ \xi(x^j, u) \tag{9}$$

holds for all $j > t$, by feasibility of the corresponding master problem. Combining (8) and (9), we obtain, for all $j > t$,

$$\max_{u\in U} \ \xi(x^t, u) - \max_{u\in U^t} \ \xi(x^t, u) + \max_{u\in U^j} \ \xi(x^j, u) - \xi(x^j, \hat{u}^t) > \varepsilon.$$

Using the fact that $\xi(x^t, \hat{u}^t) = \max_u \{\xi(x^t, u) \colon u \in U\}$ holds and rearranging the terms, we obtain

$$\xi(x^t, \hat{u}^t) - \xi(x^j, \hat{u}^t) + \max_{u\in U^j} \ \xi(x^j, u) - \max_{u\in U^t} \ \xi(x^t, u) > \varepsilon$$

By definition of LB^j and LB^t as well as Lemma 2, it follows that

$$\xi(x^t, \hat{u}^t) - \xi(x^j, \hat{u}^t) + \mathrm{LB}^j - \mathrm{LB}^t > \varepsilon \implies L_\xi \|x^t - x^j\| + \mathrm{LB}^j - \mathrm{LB}^t > \varepsilon.$$

By construction of the subsequence $(x^t, LB^t)_{t\in\bar{T}}$, it follows that $LB^j - LB^t < 1/2\varepsilon$ holds. Consequently, we obtain

$$L_\xi \|x^t - x^j\| + 0.5\varepsilon > L_\xi \|x^t - x^j\| + \mathrm{LB}^j - \mathrm{LB}^t > \varepsilon \implies \|x^t - x^j\| > 0.5\varepsilon/L_\xi.$$

Hence, there exits a ball around x^t (of radius $0.5\varepsilon/L_\xi$) which is prevented from being reached by any x^j with $j > t$. Yet, this cannot be repeated an infinite number of time since X is compact. Hence, the algorithm must terminate and correctness trivially follows. ☐

4.3. **Discrete Upper-Level.** For discrete upper-level decisions, we show that Algorithm 1 terminates after finitely many iterations, even if $\varepsilon = 0$. This is formally stated in the next theorem.

Theorem 6. *Assume that X is a discrete set. Then, Algorithm 1 with tolerance $\varepsilon = 0$ terminates after a finite number of iterations with a solution to the robust bilevel problem* (1)*, or correctly identifies it as infeasible.*

Proof. We first show that Algorithm 1 terminates after a finite number of iterations. By contradiction, let's assume that it is not the case. Then, the master problem (3) must be feasible for all $t \geq 1$. Because X is bounded and discrete, there exist at least two indices t and t' with $t < t'$ such that $x^t = x^{t'}$, i.e., x^t repeats. Yet, we show that this is not possible. There are two cases. First, assume that x^t is not feasible for Problem (1). Then, the feasibility subproblem (4) at iteration t has a positive optimal objective function value and the corresponding scenario $\tilde{u}^t$ is added to the master problem. Hence, x^t is effectively cut off since the constraint "$y_u \in S(x, \tilde{u}^t)$" is part of the master at iteration $t' > t$ and, by construction, $S(x^t, \tilde{u}^t) = \emptyset$ holds. This can only be repeated finitely many times until infeasibility is detected because $|X| < \infty$ holds.

Second, x^t is a feasible point of Problem (1). Thus, the feasibility subproblem (4) has an optimal objective function value of zero and it holds

$$\mathrm{UB}^t = \min\left\{\mathrm{UB}^{t-1}, Q(x^t)\right\} \leq Q(x^t).$$

Moreover, $t' > t$ implies that $\hat{u}^t \in U^{t'}$ holds. Hence,

$$LB^{t'} = \eta^{t'} \geq \min_{y \in S(x^t, \hat{u}^t)} \ c_x^\top x^t + c_y^\top y = \max_{u \in U} \ \min_{y \in S(x^t, u)} \ c_x^\top x^t + c_y^\top y = Q(x^t),$$

where we used the fact that $\hat{u}^t$ is a solution to the optimality subproblem (5) at iteration t and $x^t = x^{t'}$ holds. Overall, this shows that

$$Q(x^t) \leq \mathrm{LB}^{t'} \leq \mathrm{UB}^{t'} \leq \mathrm{UB}^t = Q(x^t).$$

In turn, this shows that $\mathrm{LB}^{t'} = \mathrm{UB}^{t'}$ which contradicts that the algorithm does not terminate. Thus, Algorithm 1 necessarily terminates after a finite number of iterations and correctness of the algorithm directly follows. □

Remark 5. *We note that the proof of Theorem 6 remains valid if d and D are additionally uncertain, provided that Problem $Q(x)$ attains its supremum.*

5. Solving the Subproblems

In this section, we discuss how the subproblems (4) and (5) can be solved algorithmically.

5.1. **Feasibility Subproblem.** As anticipated in Remark 2, the feasibility separation problem (4) is identical to the feasibility separation problem arising in two-stage robust optimization. Thus, methods developed for two-stage robust optimization can be directly applied to solve Problem (4). For more details, we refer to the recent paper of Flambard et al. (2025), which provides a comprehensive overview of solution approaches for Problem (4) in the case of continuous lower-level decisions. Additionally, when the lower-level decisions are discrete, the problem can be addressed using, e.g., the open-source mixed-integer bilevel solver MibS (Tahernejad et al. 2020).

5.2. **Optimality Subproblem.** We now consider the optimality subproblem (5), which we recall here: given a candidate $x \in X$, solve

$$\max_{u \in U} \ \min_{y \in S(x,u)} \ d^\top y.$$

As already noted in Remark 3, this problem can be interpreted as a pessimistic bilevel problem in which the upper-level decisions are the uncertainty vector u and the lower-level decisions are the vector y. Thus, we can exploit the recent results by Zeng (2020) on pessimistic bilevel optimization, which we detail next.

In Zeng (2020), the author presents a "relaxation-and-correction" scheme to solve pessimistic bilevel problems. The scheme consists of two main steps. First, an optimistic bilevel problem is solved, which provides a tight relaxation of the original pessimistic problem. A solution to this relaxed problem yields both an optimal upper-level decision and the corresponding optimal objective value of the original pessimistic bilevel problem; see Lemma 2 and Proposition 1 of Zeng (2020). However, the resulting lower-level decision may not be a solution to the original lower-level problem. To address this, a "correction step" is applied, which adjusts only the lower-level decisions to ensure feasibility.

In Algorithm 1, we note that only the upper-level decision and the optimal value of the optimality subproblem (5) are used. Hence, it is not necessary to apply the "correction step" and we have the following result.

Theorem 7 (Zeng 2020)**.** *Let x^t be such that, for all $u \in U$, there exists $y \in Y(x^t, u)$. Consider the optimization problem*

$$\max_{\substack{u \in U, \\ \bar{y} \in Y(x^t,u)}} \quad \min_{y} \ c_y^\top y \quad \text{s.t.} \quad y \in Y(x^t, u), \ d^\top y \leq d^\top \bar{y}. \tag{10}$$

Then, Problem (10) is solvable and has the same optimal objective value as the original optimality subproblem (5). Moreover, for any solution $(u^*, \bar{y}^*, y^*)$, the component u^* is an upper-level solution of (5), i.e., there exists y' such that (u^*, y') solves (5).

Thus, Theorem 7 allows to solve the optimality subproblem (5) by solving a classic max-min problem. Compared to the original pessimistic bilevel formulation, Problem (10) has a copy of the lower-level variables and constraints while the lower-level problem has one additional constraint. Intuitively, the upper-level decision $\bar{y}$ restricts the lower-level feasible region as much as possible to increase the value of the inner minimization problem in (10). In doing so, the upper-level effectively minimizes $d^\top \bar{y}$ subject to $\bar{y} \in Y(x^t, u)$.

Exploiting Theorem 7, standard techniques from (optimistic) bilevel optimization can be used to solve the optimality subproblem 5. For continuous lower-level variables, the lower-level problem can be replaced by its KKT conditions, yielding a single-level reformulation that can be further linearized using auxiliary variables; see, e.g., Audet et al. (1997). For discrete lower-level decisions, general-purpose mixed-integer bilevel solvers, such as MibS (Tahernejad et al. 2020), can be employed directly.

6. Computational Study

We consider two popular applications taken from the two-stage robust optimization literature. The first application is the uncapacitated facility location problem (UFLP) with facility disruption studied, for example, by Cheng et al. (2021). The second is the multiple knapsack problem (MKP) with uncertain weights; see, e.g., Lefebvre et al. (2024). We highlight that these two applications are very different in nature. More specifically, while the UFLP has a continuous lower-level problem, the MKP contains integer lower-level decisions.

The computational setup for the following study is as follows. Our implementation is done in C++ and integrated within the open-source idol library (Lefebvre 2025). We solve all single-level reformulations of linear bilevel problems using the commercial solver Gurobi version 11.0. For bilevel problems with an integer lower level, we use the mixed-integer bilevel solver MibS (Tahernejad et al. 2020) version 1.2.0 equipped with CPLEX version 22.1. All experiments were conducted on a single core Intel Xeon Gold 6126 at 2.6 GHz with 64 GB and a global time limit of 3 h.

Algorithm 1 requires an initial set of scenarios as input, for which we use $U^0 := \{u^-, u^+\}$ with

$$u^- \in \arg\min \left\{ e^\top u \colon u \in U \right\}, \quad \text{and} \quad u^+ \in \arg\max \left\{ e^\top u \colon u \in U \right\}.$$

Here, e corresponds to the vector of ones of appropriate size. We note that whenever computation time is reported, the time required to solve these two initial problems is included.

The rest of this section is organized as follows. In Section 6.1, we address the UFLP with facility disruption and compare three different variants of CCG algorithms. We then consider the MKP with uncertain weights in Section 6.2.

6.1. **Uncapacitated Facility Location Problem with Facility Disruption.**

6.1.1. *Problem Description.* We consider an UFLP in which facilities are subject to disruptions. To that end, let V_1 be a set of facility locations and let V_2 be a set of customers. For each facility $i \in V_1$, we let $\bar{f}_i$ denote its opening cost. Each customer $j \in V_2$ is associated to a given demand d_j and a marginal penalty for unmet demand $\bar{p}_j$. Each connection $(i, j) \in V_1 \times V_2$ has a unitary transportation cost noted $\bar{c}_{ij}$. The objective is to decide on a subset of facility locations to be opened so that the total cost is minimized. The total cost is composed by facility opening costs, transportation costs, and penalty costs for unmet demands. The deterministic UFLP can be modeled as

$$\begin{aligned}
\min_{x,y,z} \quad & \sum_{i \in V_1} \bar{f}_i x_i + \sum_{i \in V_1} \sum_{j \in V_2} \bar{c}_{ij} d_j y_{ij} + \sum_{j \in V_2} \bar{p}_j d_j z_j \\
\text{s.t.} \quad & \sum_{i \in V_1} y_{ij} + z_j = 1 \quad \text{for all } j \in V_2, \\
& 0 \le y_{ij} \le x_i \quad \text{for all } i \in V_1, j \in V_2, \\
& z_j \ge 0 \quad \text{for all } j \in V_2, \\
& x_i \in \{0, 1\} \quad \text{for all } i \in V_1.
\end{aligned}$$

Facilities can be disrupted for many reasons such as power outages, weather conditions, or natural disasters. A facility that is disrupted cannot be used to serve customers. It is often hard to foresee such disruptions or to estimate their probability. To account for this fact, Cheng et al. (2021) introduced a two-stage robust version in which facility locations must be selected before any disruption occurs while demand satisfaction is dealt with at a later stage. To model disruptions, the authors use a binary budgeted uncertainty set given by

$$U := \left\{ u \in \{0, 1\}^{|V_1|} \colon \sum_{i \in V_1} u_i \le \Gamma \right\}.$$

Here, a given scenario $u \in U$ is to be understood as follows: for any facility $i \in V_1$, $u_i = 1$ holds if and only if facility i is disrupted.

In a bilevel context, deciding on the facility locations to be opened and assigning opened facilities to customers is done by two different decision makers; see, e.g., Caramia and Mari (2015). Hence, the two decision makers may have conflicting objectives. First, the leader decides on opening facility locations by the leader's decision x. Then, specific facilities are disrupted, which is represented by the uncertainty u. Finally, the follower decides on assigning facilities to customers by solving the (x, u)-parameterized optimization problem

$$\min_{y,z} \quad \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} d_j y_{ij} + \sum_{j \in V_2} p_j d_j z_j \tag{11a}$$

$$\text{s.t.} \quad \sum_{i \in V_1} y_{ij} + z_j = 1, \quad \text{for all } j \in V_2, \tag{11b}$$

$$y_{ij} \le x_i(1 - u_i), \quad \text{for all } i \in V_1, j \in V_2, \tag{11c}$$

$$y_{ij} \ge 0, z_j \ge 0, \quad \text{for all } i \in V_1, j \in V_2. \tag{11d}$$

Here, c_{ij} and p_j are the follower's objective coefficients. The case in which $c_{ij} = \bar{c}_{ij}$ and $p_j = \bar{p}_j$ holds reduces to the classic two-stage robust problem studied by Cheng et al. (2021). Moreover, we note that for any leader's decision x and uncertainty $u \in U$, the follower's problem is feasible.

For a given x and u, let $S(x, u)$ denote the set of solutions to the follower's problem (11). Then, the optimistic robust bilevel UFLP can be stated as

$$\min_{x \in \{0,1\}^{|V_1|}} \left\{ \sum_{i \in V_1} \bar{f}_i x_i + \max_{u \in U} \min_{(y,z) \in S(x,u)} \sum_{i \in V_1} \sum_{j \in V_2} \bar{c}_{ij} d_j y_{ij} + \sum_{j \in V_2} \bar{p}_j d_j z_j \right\}. \tag{12}$$

We note that a similar variant of this optimistic robust bilevel UFLP problem is also studied in the recent work by Hu et al. (2026). The authors developed a UFLP tailored CCG algorithm and compare it to a general CCG scheme. Their computational study primarily focuses on comparing the benefits and differences between modeling the UFLP as a robust bilevel problem and as a two-stage robust problem from an application perspective. In contrast, our focus is on a detailed analysis of the computational impact of the individual components of the presented CCG scheme across different variants.

6.1.2. *Methods.* We consider three different approaches to solve the robust bilevel problem (12).

The first two approaches Alg.1-SOS1 and Alg.1-BigM, are variants of Algorithm 1. They employ the separation approach described in Section 5 and differ only in the type of single-level reformulation in use for solving the subproblem (10). This separation problem is reformulated by replacing the lower-level problem in Problem (10) by its KKT conditions, leading to a nonlinear single-level reformulation of the problem. The nonlinear constraints are then linearized in two ways: either using special ordered constraints of type 1, or by introducing auxiliary binary variables and big-M values. The former is referred to as Alg.1-SOS1 while the latter is Alg.1-BigM. For a detailed derivation of the corresponding valid big-M values, we refer to the Appendix A. Note that, $S(x, u) \neq \emptyset$ holds for all $x \in \{0, 1\}^{|V_1|}$ and $u \in U$. Thus, the feasibility subproblem does not need to be solved at every iteration.

We compare these newly developed approaches with a method referred to as 2RO, in which the lower-level problem of the robust bilevel problem (12) is replaced by its necessary and sufficient optimality conditions. This reformulation yields a two-stage robust optimization problem. We emphasize that this method is only applicable if strong duality holds for the lower-level problem, which is the case in Problem (12).

We now present the approach 2RO in detail. Since the lower-level problem (11) is a (feasible) linear optimization problem, it can be equivalently replaced by its KKT conditions given by

$$(11b)–(11d),$$

$$\alpha_j + \beta_{ij} + \gamma_{ij} = c_{ij} d_j, \quad \text{for all } i \in V_1, j \in V_2, \tag{13a}$$

$$\alpha_j + \delta_j = p_j d_j, \quad \text{for all } j \in V_2, \tag{13b}$$

$$\beta_{ij} \leq 0, \ \gamma_{ij} \geq 0, \ \delta_j \geq 0, \quad \text{for all } i \in V_1, j \in V_2, \tag{13c}$$

$$\beta_{ij}(y_{ij} - x_i(1 - u_i)) = 0, \quad \text{for all } i \in V_1, j \in V_2, \tag{13d}$$

$$\gamma_{ij} y_{ij} = 0, \quad \text{for all } i \in V_1, j \in V_2, \tag{13e}$$

$$z_j \delta_j = 0, \quad \text{for all } j \in V_2. \tag{13f}$$

Here, α_j, $\beta_{ij}, \gamma_{ij}, \delta_j$ denote the Lagrange multipliers of constraints (11b)–(11d), in order. Note that the last three constraints are nonlinear. However, it is well-known that they can be linearized by introducing auxiliary binary variables and bounds on the corresponding dual variables; see Audet et al. (1997). With a sufficiently large value, denoted by M, we let $K(x, u)$ be the set of points $(y, z, \alpha, \beta, \gamma, \delta, w)$ satisfying the linearized KKT conditions

$$
\begin{aligned}
&(11b)\text{–}(11d), \quad (13a)\text{–}(13c),\\
&-Mw_{ij}^1 \le \beta_{ij}, \quad x_i(1-u_i) - y_{ij} \le M(1-w_{ij}^1), \quad \text{for all } i \in V_1, j \in V_2,\\
&\gamma_{ij} \le Mw_{ij}^2, \quad y_{ij} \le M(1-w_{ij}^2), \quad \text{for all } i \in V_1, j \in V_2,\\
&\delta_j \le Mw_j^3, \quad z_j \le M(1-w_j^3), \quad \text{for all } j \in V_2,\\
&w_{ij}^1, w_{ij}^2, w_j^3 \in \{0,1\}, \quad \text{for all } i \in V_1, j \in V_2.
\end{aligned}
$$

Sufficiently large values for M are given in Cheng et al. (2021). Thus, Problem (12) can be cast as a two-stage robust problem with binary uncertainty set and mixed-integer second stage. The corresponding model reads

$$
\min_{x\in\{0,1\}^{|V_1|}} \left\{ \sum_{i\in V_1} \bar{f}_i x_i + \max_{u\in U} \min_{(y,z,\alpha,\beta,\gamma,\delta,w)\in K(x,u)} \sum_{i\in V_1}\sum_{j\in V_2} \bar{c}_{ij} d_j y_{ij} + \sum_{j\in V_2} \bar{p}_j d_j z_j \right\}.
$$

Because U is discrete, this two-stage robust problem can be addressed by the classic CCG algorithm introduced by Zeng and Zhao (2012), which we also employ. The latter requires solving the inner max-min problem to global optimality. In our implementation, we use the mixed-integer bilevel solver MibS to achieve this.

Finally, in all three approaches, the master problem (3) of the respective CCG algorithm is solved by replacing, for each scenario, the corresponding lower-level problem by its linearized KKT system using valid big-M values. In our experiments, we used the big-M values reported by Cheng et al. (2021).

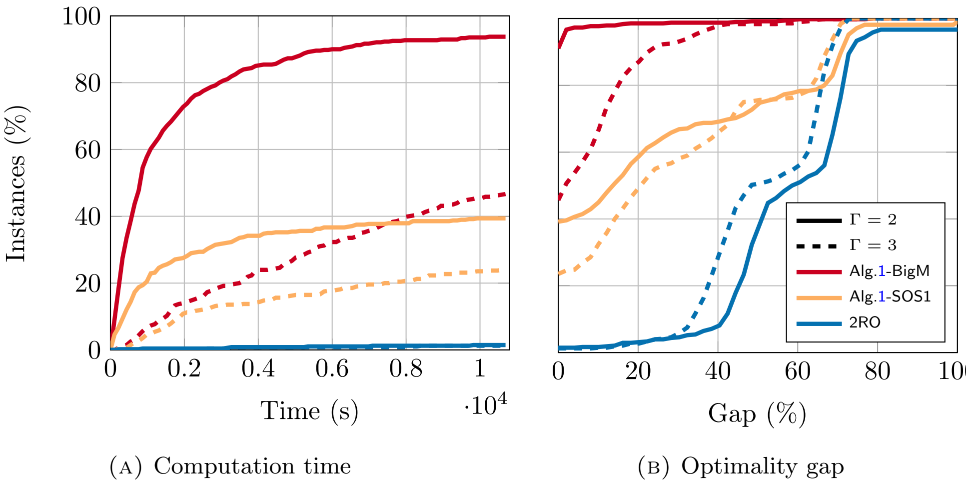
6.1.3. *Instances.* Following Cheng et al. (2021), we consider instances based on the 49-site data set from Daskin (2013), available at `https://daskin.engin.umich.edu/books/network-discrete-location/`. We create new instances by using the first 10, 12, and 14 nodes as facility locations and the subsequent 10, 12, …, 30 nodes as customers. For each customer, the demand is set to $\bar{d}_j = \lfloor P_j/10^5 \rfloor$ where P_j denotes the population at node j. The leader's transportation costs between $i \in V_1$ and $j \in V_2$ are set to $\bar{c}_{ij} = \lfloor E_{ij} \times 20 \rfloor$ where E_{ij} denotes the Euclidean distance between the two nodes. The penalty costs p_j are assumed to be the same for all customers, i.e., $\bar{p}_j = \bar{p}$ for all $j \in V_2$. We consider two possible values for $\bar{p}$. First, transportation costs $\bar{c}_{ij}$ are sorted in non-decreasing order. Then, we set $\bar{p}$ equal to the maximal transportation cost and to the $\lceil 0.8 \times |V_1| \times |V_2| \rceil$-th value in that order. Overall, this represents a total of 30 deterministic instances.

To generate the follower's objective function, we add noise to the leader's objective function in the following way. For every pair $(i,j) \in V_1 \times V_2$, we set $d_{ij} = \lceil |\bar{d}_{ij}\psi_{ij}| \rceil$ and $p_j = \lceil |\bar{p}_j \phi_j| \rceil$ where ψ_{ij} and ϕ_j are random variables following a normal distribution centered around 1 and with standard deviation σ taking values in $\{0.3, 0.5\}$. Empirically, the average angle between the leader's and the follower's objective function is of 17 and 25 degrees, respectively, where the angle is computed based on $\angle(a,b) := \cos^{-1}(a^\top b/(\|a\|\|b\|))$. For each value of σ, 4 instances were generated, leading to 480 nominal bilevel instances.

Finally, we consider an uncertainty budget Γ equal to 2 or 3. Hence, our test bed comprises 960 robust bilevel instances.

6.1.4. *Results.* We compare the two approaches developed in this paper, namely Alg.1-SOS1 and Alg.1-BigM, to the two-stage robust reformulation based on KKT conditions called 2RO.

Figure 1a shows the empirical cumulative distribution function (ECDF) of the computation times. Both Alg.1-SOS1 and Alg.1-BigM clearly outperform 2RO. More specifically, 2RO is able to solve less than 2 % of the test bed with $\Gamma = 2$ within the time limit whereas Alg.1-SOS1 solves roughly 40 %. The best performing approach Alg.1-BigM solves more than 95 % of the instances with $\Gamma = 2$ and more than 80 % of these instances are solved within one hour by Alg.1-BigM. Moreover, Figure 1a



(A) Computation time (B) Optimality gap

FIGURE 1. Empirical cumulative distribution function of computation times and optimality gaps for the UFLP.

also shows that increasing the value of the uncertainty budget Γ highly impacts the performance of Alg.1-SOS1 and Alg.1-BigM. For instance, while more than 95 % of the instances are solved within the time limit with $\Gamma = 2$, this drops to roughly 40 % of the instances for $\Gamma = 3$ when using Alg.1-BigM. However, 2RO is still outperformed and not competitive compared to Algorithm 1.

Figure 1b depicts the ECDF of optimality gap after the time limit. Here again, we see that 2RO is outperformed by Alg.1-SOS1, itself being outperformed by Alg.1-BigM. Nevertheless, we also see that the remaining optimality gaps may be quite large. For instance, 40 % of the instances have an optimality gap which is greater than 20 % for $\Gamma = 2$ and Alg.1-SOS1.

In Table 2 and Table 3, we give a detailed overview of key characteristics such as computation times and number of added scenarios for Alg.1-SOS1 and Alg.1-BigM. More precisely, from left to right and for each value of $|V_1|$, $|V_2|$, and Γ, we report the number of instances solved to optimality (out of 16), the median computation time, the median percentage of the time spent solving the master problem (as opposed to the time spent solving the subproblem), and the median number of scenarios added to the master problem. Note that medians are computed only over those instances which are solved to global optimality within the time limit. Table 2 shows that for Alg.1-SOS1 solving the separation subproblems is the main bottleneck, in particular for larger instances. We typically observe that the more challenging an instance is, the less time is spent solving the master problem. Moreover, harder instances are typically those with larger values of $|V_1|$ and $|V_2|$, or with $\Gamma = 3$ instead of $\Gamma = 2$. In contrast, Table 3 shows that for Alg.1-BigM the main bottleneck consists of solving the master problem. Note that this is to be expected since the master problem is now a single leader multi-follower bilevel problem with an increasing number of followers. Roughly speaking, we see that more than 80 % of the time is spent solving the master problem for instances such that $|V_1| \geq 12$.

The computational results show that the 2RO approach is clearly outperformed by both Alg.1-SOS1 and Alg.1-BigM. If valid big-M values are available, Alg.1-BigM is clearly the best available method. However, Alg.1-SOS1 remains a practical alternative that can solve instances of moderate size.

6.2. Multiple Knapsack Problem with Uncertain Weights.

6.2.1. *Problem Description.* We consider an MKP with uncertain weights. Given a set of n items, each with an associated nominal weight $\bar{w}_j$ and profit p_j for all

TABLE 2. Detailed median computation times and median number of added scenarios w.r.t all solved instances for the UFLP using Alg.1-SOS1. For each value of $(|V_1|, |V_2|, \Gamma)$, the total number of instances is 16.

		$\Gamma = 2$				$\Gamma = 3$			
$\|V_1\|$	$\|V_2\|$	# opt.	time (s)	master (%)	# scen.	# opt.	time (s)	master (%)	# scen.
10	10	16	63.2	60.0	18.0	16	678.5	91.2	44.5
	12	16	153.7	46.8	20.0	16	1043.8	89.0	46.0
	14	16	854.2	20.1	18.5	15	1673.2	91.8	43.0
	16	16	559.8	15.6	17.0	15	2215.4	69.6	40.0
	18	7	1792.8	12.2	15.0	11	3901.6	61.6	43.0
	20	6	3627.9	9.3	17.5	3	6675.8	38.0	40.0
	22	2	5641.7	4.9	14.0	2	9220.7	23.3	30.5
	24	–	–	–	–	–	–	–	–
	26	–	–	–	–	–	–	–	–
	28	–	–	–	–	–	–	–	–
	30	–	–	–	–	–	–	–	–
12	12	16	467.8	67.3	21.0	7	5673.4	97.3	63.0
	14	16	772.2	47.2	18.5	4	2236.0	80.2	38.0
	16	14	1854.8	24.6	20.5	2	6066.1	88.0	50.5
	18	8	1858.9	24.3	18.5	1	4381.5	85.5	38.0
	20	4	4633.7	26.7	22.0	–	–	–	–
	22	–	–	–	–	–	–	–	–
	24	–	–	–	–	–	–	–	–
	26	–	–	–	–	–	–	–	–
	28	–	–	–	–	–	–	–	–
	30	–	–	–	–	–	–	–	–
14	14	15	597.9	56.4	20.0	7	5808.9	94.3	48.0
	16	13	888.6	48.3	19.0	9	8161.7	87.1	49.0
	18	9	1077.9	38.8	19.0	3	4738.4	78.3	42.0
	20	8	2743.9	21.1	19.5	4	5317.6	58.6	37.0
	22	6	5340.6	14.0	21.5	–	–	–	–
	24	1	5486.6	15.4	18.0	–	–	–	–
	26	–	–	–	–	–	–	–	–
	28	–	–	–	–	–	–	–	–
	30	–	–	–	–	–	–	–	–

$j \in \{1, \dots, n\}$, the goal is to select a subset of items and assign them to one of K knapsacks, each with fixed capacity W. Following Lefebvre et al. (2024), we consider a setting in which items must be pre-assigned to knapsacks before their exact weights are revealed. Moreover, we assume that a second decision maker is responsible for the actual packing of the items into the knapsacks after the realization of the uncertainty.

To model weight uncertainty, we assume that the actual weight of item $j \in \{1, \dots, n\}$ may exceed its nominal value $\bar{w}_j$ by at most $\hat{w}_j$. Furthermore, we assume that the weights of at most Γ items can deviate from their nominal values simultaneously. Thus, we consider the uncertainty set

$$U := \left\{ u \in \{0,1\}^n : \sum_{j=1}^{n} u_j \leq \Gamma \right\},$$

where $u_j = 1$ holds if and only if item j has its weight increased by $\hat{w}_j$.

The upper-level decisions consist in assigning each item to exactly one knapsack, before knowing their exact weights. Hence, the upper-level feasible region can be

TABLE 3. Detailed median computation times and median number of added scenarios w.r.t all solved instances for the UFLP using Alg.1-BigM. For each value of $(|V_1|, |V_2|, \Gamma)$, the total number of instances is 16.

		$\Gamma = 2$				$\Gamma = 3$			
$\lvert V_1\rvert$	$\lvert V_2\rvert$	# opt.	time (s)	master (%)	# scen.	# opt.	time (s)	master (%)	# scen.
10	10	16	42.9	87.4	18.0	16	644.8	97.8	44.5
	12	16	92.5	89.8	20.0	16	861.1	98.3	46.0
	14	16	162.2	86.8	18.5	16	1569.9	98.2	43.0
	16	16	156.5	81.8	17.0	16	1747.7	96.7	40.0
	18	16	297.4	82.5	19.0	16	2247.5	96.2	43.0
	20	15	592.6	80.2	17.0	15	4677.5	94.6	42.0
	22	16	335.8	75.6	15.5	16	4212.7	93.3	38.0
	24	15	648.1	73.5	17.0	14	5110.4	92.1	39.0
	26	16	548.0	79.3	14.5	12	4960.1	86.3	34.5
	28	13	1326.4	71.7	16.0	7	7344.1	89.1	34.0
	30	16	1177.5	76.6	16.0	9	6685.3	91.4	34.0
12	12	16	371.3	93.4	21.0	7	5842.4	99.5	63.0
	14	16	342.1	92.2	18.5	5	1652.1	98.5	39.0
	16	15	479.0	92.3	20.0	3	3701.1	99.1	47.0
	18	16	728.0	89.2	19.0	3	6744.3	92.8	41.0
	20	15	1054.4	90.0	19.0	4	6487.1	93.9	39.5
	22	14	2076.7	91.2	23.5	2	8901.6	90.1	40.5
	24	12	1160.4	88.4	16.5	2	4677.7	79.5	30.0
	26	16	2589.7	84.4	19.0	1	6275.5	93.2	23.0
	28	14	4666.5	85.8	21.0	1	7749.8	65.7	25.0
	30	12	3860.0	81.3	20.5	–	–	–	–
14	14	14	321.4	94.8	20.0	7	5397.6	99.3	48.0
	16	14	523.7	90.7	20.0	10	7090.8	98.6	49.0
	18	15	740.4	91.1	20.0	8	7063.6	97.0	44.5
	20	16	846.2	91.5	20.0	10	5930.4	96.2	41.0
	22	15	1517.4	86.3	21.0	4	9454.5	98.0	42.5
	24	16	1431.0	89.2	20.0	4	5926.4	95.1	39.5
	26	15	1749.0	88.6	20.0	–	–	–	–
	28	14	2952.3	91.5	21.0	–	–	–	–
	30	14	4963.2	78.1	21.5	–	–	–	–

modeled as

$$X := \left\{ x \in \{0,1\}^{n \times K} : \sum_{k=1}^{K} x_{jk} = 1, \quad \text{for all } j = 1, \dots, n \right\}.$$

Given a scenario $u \in U$ and an upper-level decision $x \in X$, the lower-level problem reads

$$\min_{y} \quad -\sum_{k=1}^{K} \sum_{j=1}^{n} p_j y_{jk} \tag{14a}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} (\bar{w}_j + u_j \hat{w}_j) y_{jk} \leq W, \quad \text{for all } k = 1, \dots, K, \tag{14b}$$

$$y_{jk} \leq x_{jk}, \quad \text{for all } j = 1, \dots, n, \; k = 1, \dots, K, \tag{14c}$$

$$y_{jk} \in \{0,1\}, \quad \text{for all } j = 1, \dots, n, \; k = 1, \dots, K. \tag{14d}$$

In words, the lower-level problems maximizes the profit by packing as many items as possible while not exceeding the capacity constraints (14b) and respecting the pre-assignments by the upper level.

Remark 6. *We note that Constraint (14b) has an uncertain constraint matrix, in contrast to Problem (2). However, following Lefebvre et al. (2024), the product between*

u_j and y_{jk} can be linearized exactly using McCormick inequalities, which we adopt in our implementation. The resulting problem has uncertain right-hand side only, as it is the case for Problem (2). For more details, please refer Section 3.1 in Lefebvre et al. (2024).

For given $u \in U$ and $x \in X$, let $S(x, u)$ denote the set of solutions to the lower-level problem (14). Then, the robust bilevel MKP can be formulated as

$$\min_{x \in X} \max_{u \in U} \min_{y \in S(x,u)} \sum_{k=1}^{K} \sum_{j=1}^{n} \bar{p}_{ij} y_{ij},$$

where $\bar{p}_{ij}$ denotes the upper-level objective function coefficients.

6.2.2. *Methods.* We consider solving the robust bilevel MKP with Algorithm 1. Note that this problem has integer lower-level decisions. Consequently, a KKT-based reformulation of the lower-level problem, as used for the UFLP, is no longer applicable. Instead the master and subproblems within Algorithm 1 are mixed-integer linear bilevel problems that we solve using the mixed-integer bilevel solver MibS (Tahernejad et al. 2020). Since the lower-level problem is feasible for any upper-level decision, no feasibility subproblem needs to be solved. Moreover, we solve the optimality subproblem (5) by its reformulation (10). This approach is denoted by Alg.1-MibS in the following. We emphasize that to the best of our knowledge, no general-purpose method other than Algorithm 1 is currently available for solving the robust bilevel MKP.

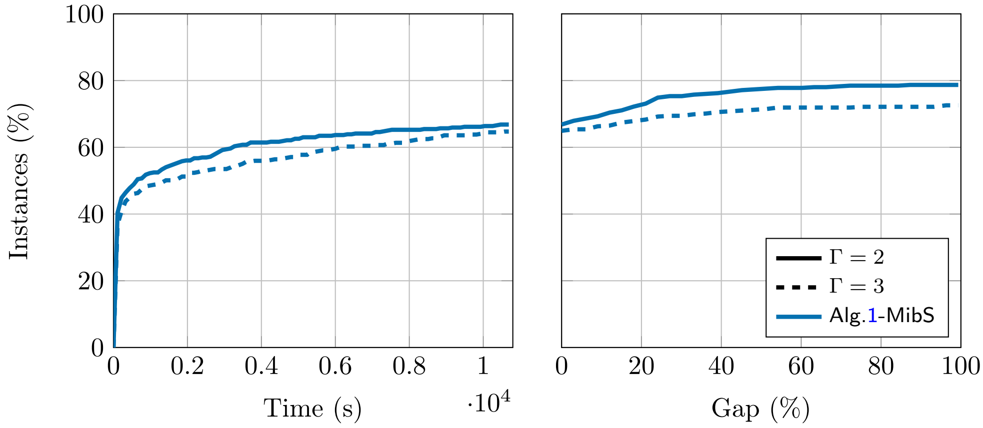
6.2.3. *Instances.* We randomly generate instances in the same way as Lefebvre et al. (2024) by choosing $n \in \{6, 8, 10, 12, 14\}$ and $K \in \{2, 3, 4\}$. For each item j, profits p_j, $\bar{p}_j$ and nominal weights $\bar{w}_j$ are generated according to a discrete and uniform distribution in $[1, 1000]$. The capacity W of each knapsack is set to $\alpha \sum_{j=1}^{n} \bar{w}_j / K$ with α taking values in $\{0.25, 0.5, 0.75\}$. The increase in weight $\hat{w}_j$ is set to the closest integer to $\bar{w}_j(1 + \delta_j)$, where δ_j is drawn uniformly at random from $[0, H]$ with $H \in \{0.2, 0.5\}$.

For each combination of n, K, α and H, we generated 5 instances, making an overall test set of 450 bilevel instances. The uncertainty budget Γ was set equal to 2 and 3, producing a set of 900 robust bilevel instances.

6.2.4. *Results.* Figure 2a depicts the ECDF of computation times of Algorithm 1 equipped with MibS for solving both the master problem and the optimality subproblems. We see that the method is able to solve around 65 % of the instances with $\Gamma = 2$. As observed for the UFLP, increasing the value of Γ again leads to harder instances. However, this effect is significantly less pronounced the MKP than for the UFLP. We note that the instance sizes that can be solved are comparatively smaller than those that can be solved for the UFLP. This is to be expected since, for the MKP, the lower-level problem contains integer variables, which is notoriously difficult to tackle already in a (deterministic) bilevel setting as well as in the context of two-stage robust optimization.

Figure 2b shows the ECDF of the optimality gaps after the time limit. It can be observed that the remaining gaps for those instances that could not be solved to optimality can be rather large. For instance, roughly 30 % of the instances have a remaining gap larger than 20 %.

Analogously to the UFLP, Table 4 reports the median computation times, the median number of added scenarios, and the median time spent solving the master problem. Clearly, the harder instances correspond to those with a larger number of items. Moreover, we observe that the time spent in the master problem increases depending on the numbers of knapsacks and the number of scenarios needed in the master problem to prove optimality is rather low compared to what was needed for the UFLP. It is difficult to identify the general bottleneck of the algorithm in terms of time spent solving the master problem versus the subproblems. For the case of two



(A) Computation time (B) Optimality gap

FIGURE 2. Empirical cumulative distribution function of computation times and optimality gaps for the MKP.

TABLE 4. Detailed median computation times and median number of added scenarios w.r.t all solved instances for the MKP. For each value of N and K, the total number of instances is 30.

		$\Gamma = 2$				$\Gamma = 3$			
N	K	# opt.	time (s)	master (%)	# scen.	# opt.	time (s)	master (%)	# scen.
6	2	30	1.8	17.0	2.5	30	2.7	2.1	2.0
	3	30	1.1	38.3	2.0	30	1.5	15.5	2.0
	4	30	1.4	78.9	2.0	30	1.7	73.5	2.0
8	2	30	26.0	4.6	3.0	30	82.9	0.9	3.0
	3	30	28.5	39.1	3.0	30	46.8	17.8	3.0
	4	30	26.5	79.8	2.0	30	36.8	60.3	2.0
10	2	27	920.2	1.1	3.0	25	3193.7	0.6	4.0
	3	24	1012.6	55.8	3.0	23	1352.4	27.3	3.0
	4	25	869.7	86.1	3.0	24	1099.1	66.0	3.0
12	2	8	525.0	0.4	2.5	10	1904.9	0.5	3.0
	3	12	1613.5	72.7	3.5	8	934.5	42.5	3.5
	4	9	2669.2	97.8	3.0	8	2678.8	86.4	2.5
14	2	5	4682.5	0.1	3.0	3	9993.5	0.6	3.0
	3	6	3273.4	54.7	3.0	6	6685.5	32.8	3.0
	4	2	3426.3	78.1	2.0	2	5935.1	58.7	2.0

knapsacks ($K = 2$), most of the computation time is spent solving the subproblems while for four knapsacks the computation time of the master problem dominates. In general, both the master and subproblems constitute very challenging mixed-integer bilevel problems.

7. CONCLUSION

In this paper, we present a general solution approach based on column-and-constraint generation (CCG) for optimistic robust bilevel problems with a wait-and-see follower. We considered a mixed-integer linear upper-level problem and a (mixed-integer) linear follower's problem that is affected by both right-hand side uncertainty and the leader's decision. The presented CCG approach iteratively solves a bilevel master problem with a single leader and multiple followers together with two different types of subproblems. The first one is an optimistic bilevel problem that verifies feasibility of a given leader's decision, while the second one is a pessimistic bilevel problem that verifies optimality. The latter is solved using recent advances in pessimistic bilevel optimization. We prove that the presented CCG approach computes a solution of arbitrary precision in finitely many iterations for polyhedral uncertainty sets. Moreover, we prove that the algorithm is exact within finitely many iterations if the set of upper-level decisions is finite or the uncertainty is discrete. Finally, we demonstrate the applicability of the approach in a detailed computational study with two different applications including both continuous and integer follower's decisions.

In general, there are numerous opportunities for future research on algorithmic approaches for robust bilevel problems with a wait-and-see follower. Developing methods to compute strong lower and upper bounds could significantly improve the solution times and enable the solution of larger instances. Moreover, techniques for scenario reduction in the master problem as well as tailored methods to solve the single-leader multi follower master problem, pose interesting directions for future research.

Acknowledgements

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 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

Aliprantis, C. D. and K. C. Border (2006). *Infinite dimensional analysis: a hitchhiker's guide*. Springer. DOI: 10.1007/3-540-29587-9.

Audet, C., P. Hansen, B. Jaumard, and G. Savard (May 1997). "Links Between Linear Bilevel and Mixed 0–1 Programming Problems." In: *Journal of Optimization Theory and Applications* 93.2, pp. 273–300. DOI: 10.1023/a:1022645805569.

Beck, Y., I. Ljubić, and M. Schmidt (2023a). "A survey on bilevel optimization under uncertainty." In: *European Journal of Operational Research* 311.2, pp. 401–426. DOI: 10.1016/j.ejor.2023.01.008.

– (2023b). "Exact methods for discrete Γ-robust interdiction problems with an application to the bilevel knapsack problem." In: *Mathematical Programming Computation* 15.4, pp. 733–782. DOI: 10.1007/s12532-023-00244-6.

Beck, Y., I. Ljubić, and M. Schmidt (2025). "Heuristic Methods for Γ-Robust Mixed-Integer Linear Bilevel Problems." In: *INFORMS Journal on Computing*. Online first. DOI: 10.1287/ijoc.2023.0239.

Beck, Y. and M. Schmidt (2021). "A robust approach for modeling limited observability in bilevel optimization." In: *Operations Research Letters* 49.5, pp. 752–758. DOI: 10.1016/j.orl.2021.07.010.

Besançon, M., M. F. Anjos, and L. Brotcorne (2024). "Robust bilevel optimization for near-optimal lower-level solutions." In: *Journal of Global Optimization* 90.4, pp. 813–842. DOI: 10.1007/s10898-024-01422-z.

Buchheim, C. and D. Henke (2022). "The Robust Bilevel Continuous Knapsack Problem with Uncertain Coefficients in the Follower's Objective." In: *Journal of Global Optimization* 83.4, pp. 803–824. DOI: 10.1007/s10898-021-01117-9.

Buchheim, C., D. Henke, and F. Hommelsheim (2021). "On the complexity of robust bilevel optimization with uncertain follower's objective." In: *Operations Research Letters* 49.5, pp. 703–707. DOI: 10.1016/j.orl.2021.07.009.

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.

Caramia, M. and R. Mari (July 2015). "A decomposition approach to solve a bilevel capacitated facility location problem with equity constraints." In: *Optimization Letters* 10.5, pp. 997–1019. DOI: 10.1007/s11590-015-0918-z.

Cheng, C., Y. Adulyasak, and L.-M. Rousseau (2021). "Robust facility location under disruptions." In: *INFORMS Journal on Optimization* 3.3, pp. 298–314. DOI: 10.1287/ijoo.2021.0054.

Chuong, T. D. and V. Jeyakumar (2017). "Finding robust global optimal values of bilevel polynomial programs with uncertain linear constraints." In: *Journal of Optimization Theory and Applications* 173.2, pp. 683–703. DOI: 10.1007/s10957-017-1069-4.

Daskin, M. S. (June 2013). *Network and Discrete Location: Models, Algorithms, and Applications, Second Edition*. Wiley. DOI: 10.1002/9781118537015.

Dempe, S. and A. B. Zemkoho, eds. (2020). *Bilevel Optimization: Advances and Next Challenges*. Vol. 161. Springer Optimization and Its Applications. Cham: Springer. DOI: 10.1007/978-3-030-52119-6.

Flambard, P., A. N. Arslan, and B. Detienne (2025). *A semi-infinite constraint generation algorithm for two-stage robust optimization problems*. working paper or preprint. URL: https://hal.science/hal-05184998.

Goerigk, M., D. Henke, J. Kager, F. Schäfer, and C. Thielen (2025a). *A fast approximate column-and-constraint generation method for two-stage robust mixed-integer programs*. arXiv: 2501.05388 [math.OC]. URL: https://arxiv.org/abs/2501.05388.

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

Hansen, P., B. Jaumard, and G. Savard (1992). "New Branch-and-Bound Rules for Linear Bilevel Programming." In: *SIAM Journal on Scientific and Statistical Computing* 13.5, pp. 1194–1217. DOI: 10.1137/0913069.

Henke, D. (2025). "The Robust Bilevel Selection Problem." In: *Open Journal of Mathematical Optimization* 6, 4, pp. 1–35. DOI: 10.5802/ojmo.40.

Hu, H., J. Li, J. Tang, and B. Zeng (2026). *Two-stage robust bilevel optimization model for facility location considering operational service level under disruption risk*. arXiv: 2603.21225 [math.OC]. URL: https://arxiv.org/abs/2603.21225.

Jeroslow, R. G. (1985). "The polynomial hierarchy and a simple model for competitive analysis." In: *Mathematical Programming* 32.2, pp. 146–164. DOI: 10.1007/bf01586088.

Kis, T., A. Kovács, and C. Mészáros (2025). "Robust bilevel optimization: algorithms, complexity and application." In: *Optimization* 74.12, pp. 2951–2990. DOI: 10.1080/02331934.2025.2468405.

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, Paper No. 100007, 21. DOI: 10.1016/j.ejco.2021.100007.

Köppe, M., M. Queyranne, and C. T. Ryan (2010). "Parametric integer programming algorithm for bilevel mixed integer programs." In: *Journal of Optimization Theory and Applications* 146.1, pp. 137–150. DOI: 10.1007/s10957-010-9668-3.

Lefebvre, H. (2025). *idol: A C++ Framework for Optimization*. publicly available online. URL: https://hlefebvr.github.io/idol/ (visited on 09/01/2025).

Lefebvre, H., E. Malaguti, and M. Monaci (Jan. 2024). "Adjustable Robust Optimization with Discrete Uncertainty." In: *INFORMS Journal on Computing* 36.1, pp. 78–96. DOI: 10.1287/ijoc.2022.0086.

Lefebvre, H., M. Schmidt, and J. Thürauf (2026). "Column generation in column-and-constraint generation for adjustable robust optimization with Interdiction-type linking Constraints." In: *Mathematical Programming Computation*. DOI: 10.1007/s12532-025-00300-3.

Lefebvre, H. and A. Subramanyam (2025). "Remarks on: A Lagrangian dual method for two-stage robust optimization with binary uncertainties." In: *Optimization and Engineering* 27.1, pp. 531–553. DOI: 10.1007/s11081-025-10015-y.

Stackelberg, H. von (1934). *Marktform und Gleichgewicht*. Springer.

– (1952). *The Theory of the Market Economy*. William Hodge.

Still, G. (2018). "Lectures on parametric optimization: An introduction." In: *Optimization Online*. URL: https://optimization-online.org/?p=15154.

Subramanyam, A. (2022). "A Lagrangian dual method for two-stage robust optimization with binary uncertainties." In: *Optimization and Engineering* 23.4, pp. 1831–1871. DOI: 10.1007/s11081-022-09710-x.

Subramanyam, A., C. E. Gounaris, and W. Wiesemann (2019). "K-adaptability in two-stage mixed-integer robust optimization." In: *Mathematical Programming Computation* 12.2, pp. 193–224. DOI: 10.1007/s12532-019-00174-2.

Tahernejad, S., T. K. Ralphs, and S. T. DeNegre (June 2020). "A branch-and-cut algorithm for mixed integer bilevel linear optimization problems and its implementation." In: *Mathematical Programming Computation* 12.4, pp. 529–568. DOI: 10.1007/s12532-020-00183-6.

Takeda, A., S. Taguchi, and R. H. Tütüncü (2007). "Adjustable Robust Optimization Models for a Nonlinear Two-Period System." In: *Journal of Optimization Theory and Applications* 136.2, pp. 275–295. DOI: 10.1007/s10957-007-9288-8.

Tsang, M. Y., K. S. Shehadeh, and F. E. Curtis (2023). "An inexact column-and-constraint generation method to solve two-stage robust optimization problems." In: *Operations Research Letters* 51.1, pp. 92–98. DOI: 10.1016/j.orl.2022.12.002.

Xu, L. (2023). "A Study on Multilevel Optimization Subject to Uncertainty." Doctoral dissertation. University of Pittsburgh. URL: https://d-scholarship.pitt.edu/concern/etds/52182e1e-5550-4916-8066-d856f315251f.

Zare, M. H., O. A. Prokopyev, and D. Sauré (2020). "On bilevel optimization with inexact follower." In: *Decision Analysis* 17.1, pp. 74–95. DOI: 10.1287/deca.2019.0392.

Zeng, B. (2020). "A practical scheme to compute the pessimistic bilevel optimization problem." In: *INFORMS Journal on Computing* 32.4, pp. 1128–1142. DOI: 10.1287/ijoc.2019.0927.

Zeng, B., H. Dong, R. Sioshansi, F. Xu, and M. Zeng (2020). "Bilevel Robust Optimization of Electric Vehicle Charging Stations With Distributed Energy Resources." In: *IEEE Transactions on Industry Applications* 56.5, pp. 5836–5847. DOI: 10.1109/TIA.2020.2984741.

Zeng, B. and L. Zhao (Jan. 2012). *An Exact Algorithm for Two-stage Robust Optimization with Mixed Integer Recourse Problems*. Tech. rep. Applications - OR and Management Sciences, Robust Optimization, Scheduling. Optimization Online. URL: https://optimization-online.org/?p=11876.

– (2013). "Solving two-stage robust optimization problems using a column-and-constraint generation method." In: *Operations Research Letters* 41.5, pp. 457–461. DOI: 10.1016/j.orl.2013.05.003.

Appendix A. Bounds on the Dual Variables of the Lower-Level Problem in the Optimality Subproblem of the UFLP

We follow the approach from Section 5 to address the optimality subproblem for the UFLP considered in Section 6.1. Thus, we consider the optimistic reformulation of the optimality subproblem (5) given in (10). More specifically, let $x \in \{0,1\}^{|V_1|}$ be given and let $(u, \bar{y}, \bar{z})$ satisfy $u \in U$ as well as constraints (11b)–(11d). The lower-level problem in (10) reads

$$\begin{aligned}
\min_{y,z} \quad & \sum_{i \in V_1} \sum_{j \in V_2} \bar{c}_{ij} d_j y_{ij} + \sum_{j \in V_2} \bar{p}_j d_j z_j && (15a)\\
\text{s.t.} \quad & \sum_{i \in V_1} y_{ij} + z_j = 1, \quad \text{for all } j \in V_2, && (15b)\\
& y_{ij} \le x_i(1-u_i), \quad \text{for all } i \in V_1, \text{for all } j \in V_2, && (15c)\\
& y_{ij} \ge 0, z_j \ge 0, \quad \text{for all } i \in V_1, j \in V_2, && (15d)\\
& \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} d_j y_{ij} + \sum_{j \in V_2} p_j d_j z_j \le \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} d_j \bar{y}_{ij} + \sum_{j \in V_2} p_j d_j \bar{z}_j, && (15e)
\end{aligned}$$

The KKT conditions of Problem (15) are

$$\text{Primal Feasibility} = \begin{cases} \sum_{i \in V_1} y_{ij} + z_j = 1, \quad \text{for all } j \in V_2, \\ y_{ij} \le x_i(1-u_i), \quad \text{for all } i \in V_1, \text{for all } j \in V_2, \\ y_{ij} \ge 0, z_j \ge 0, \quad \text{for all } i \in V_1, j \in V_2, \\ \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} d_j (y_{ij} - \bar{y}_{ij}) + \sum_{j \in V_2} p_j d_j (z_j - \bar{z}_j) \le 0, \end{cases}$$

$$\text{Stationarity} = \begin{cases} \alpha_j + \beta_{ij} + \gamma_{ij} + \eta c_{i,j} d_j = \bar{c}_{ij} d_j, \quad \text{for all } i \in V_1, j \in V_2, \\ \alpha_j + \delta_j + \eta p_j d_j = \bar{p}_j d_j, \quad \text{for all } j \in V_2, \end{cases}$$

$$\text{Dual Feasibility} = \begin{cases} \beta_{ij} \le 0, \gamma_{ij} \ge 0, \quad \text{for all } i \in V_1, j \in V_2, \\ \alpha_j \in \mathbb{R}, \delta_j \ge 0, \quad \text{for all } j \in V_2, \\ \eta \le 0, \end{cases}$$

$$\text{Complementarity} = \begin{cases} \beta_{ij}(y_{ij} - x_i(1-u_i)) = 0, \quad \text{for all } i \in V_1, j \in V_2, \\ \gamma_{ij} y_{ij} = 0, \quad \text{for all } i \in V_1, j \in V_2, \\ \delta_j z_j = 0, \quad \text{for all } i \in V_1, j \in V_2, \\ \eta \left(\sum_{i \in V_1} \sum_{j \in V_2} c_{ij} d_j y_{ij} + \sum_{j \in V_2} p_j d_j z_j \right. \\ \quad \left. - \sum_{i \in V_1} \sum_{j \in V_2} c_{ij} d_j \bar{y}_{ij} + \sum_{j \in V_2} p_j d_j \bar{z}_j \right) = 0. \end{cases} \tag{16}$$

Before deriving bounds on the dual variables of Problem (16), we need an additional assumption.

Assumption 2. *The following conditions are satisfied.*

i. It holds $c, \bar{c}, d, p, \bar{p} > 0$.
ii. For all $i \in V_1$ and $j \in V_2$, $p_j - c_{ij} \neq 0$ holds.
iii. For all $i, \tilde{i} \in V_1$ with $i \neq \tilde{i}$ as well as for all $j \in V_2$, $c_{\tilde{i}j} - c_{ij} \neq 0$ holds.

We start with two auxiliary lemma to bound the dual variable η.

Lemma 3. *Let $(y, z, \alpha, \beta, \gamma, \delta, \eta)$ satisfiy the KKT conditions* (16) *as well as*

$$\eta < \eta^* := -\max\left\{\max_{\tilde{i}, i \in V_1, j \in V_2 : i \neq \tilde{i}} \frac{|\bar{c}_{\tilde{i}j} d_j - \bar{c}_{ij} d_j|}{|c_{\tilde{i}j} d_j - c_{ij} d_j|}, \max_{i \in V_1, j \in V_2} \frac{|\bar{p}_j d_j - \bar{c}_{ij} d_j|}{|p_j d_j - c_{ij} d_j|}\right\}.$$

Then, for each index $j \in V_2$, the following two statements are true.

i. If $\delta_j = 0$, then for each $i \in V_1$ the inequalities $\beta_{ij} < 0$ or $\gamma_{ij} > 0$ hold.
ii. If $\delta_j > 0$, then at most one tuple (i, j) with $i \in V_1$ with $\beta_{ij} = \gamma_{ij} = 0$ exists.

Proof. First, we consider the case $\delta_j = 0$. By contradiction, let's assume that there is $i \in V_1$ with $\beta_{ij} = \gamma_{ij} = 0$. Then, from the stationarity constraints we obtain

$$\alpha_j + \eta c_{ij} d_j = \bar{c}_{ij} d_j, \quad \alpha_j + \eta p_j d_j = \bar{p}_j d_j.$$

Subtracting the first equality of the second and solving the equation for η leads to

$$\eta = (\bar{p}_j d_j - \bar{c}_{ij} d_j)/(p_j d_j - c_{ij} d_j),$$

which is a contradiction to the choice of η.

Second, consider $\delta_j > 0$. By contradiction, let's assume that there exists two different tuples (i, j) and $(\tilde{i}, j)$ in $V_1 \times V_2$ with $\beta_{ij} = \gamma_{ij} = \beta_{\tilde{i}j} = \gamma_{\tilde{i}j} = 0$. Then, from the stationarity constraints, we obtain

$$\alpha_j + \eta c_{ij} d_j = \bar{c}_{ij} d_j, \quad \alpha_j + \eta c_{\tilde{i}j} d_j = \bar{c}_{\tilde{i}j} d_j.$$

Subtracting again the first equality of the second and solving the equation for η leads to

$$\eta = (\bar{c}_{\tilde{i}j} d_j - \bar{c}_{ij} d_j)/(c_{\tilde{i}j} d_j - c_{ij} d_j),$$

which is a contradiction to the choice of η. □

Lemma 4. *There exists a point $(y, z, \alpha, \beta, \gamma, \delta, \eta)$ that satisfies the KKT system* (16) *and such that $\eta \geq \eta^*$ holds.*

Proof. We prove the claim by contradiction. Thus, we assume that every feasible KKT point of (16) are such that $\eta < \eta^*$. Let now $(y, z, \alpha, \beta, \gamma, \delta, \eta)$ be a feasible KKT point, in which η is as large as possible w.r.t. y and z, i.e., there is no other feasible KKT point $(y, z, \alpha', \beta', \gamma', \delta', \eta')$ with the same y and z such that $\eta' > \eta$. We note that this point exists because for fixed x, y, and z maximizing η over the variables α, γ, β, and δ corresponds to solving a feasible and bounded linear problem, which always attains its solution.

For fixed $j \in V_2$, increasing η leads to an increase of the left-hand sides of all stationarity constraints indexed by j. However, for a sufficiently small increase of η, we can adjust $\alpha_j, \delta_j, \beta_{ij}$, and γ_{ij} with $i \in V_1$ such that the stationarity constraints are satisfied after increasing η and the signs of the corresponding variables do not change, e.g., if before the increase of η the inequality $\beta_{ij} < 0$ holds, then this holds as well after the adjustment of the variables. This adjustment is possible due to the Lemma 3, which we explicitly outline in the following. Now, $\varepsilon > 0$ denotes the increase of η. Moreover, we note that increasing, respectively decreasing, b_{ij}, γ_{ij} or δ_j has no effect on the dual feasibility and complementarity constraints if we do not change the sign of these variables.

If $\delta_j = 0$ holds, then we decrease α_j by $\varepsilon p_j d_j$ so that the stationarity constraint $\alpha_j - \varepsilon p_j d_j + (\eta + \varepsilon) p_j d_j = \bar{p} d_j$ is satisfied. After this adjustment the remaining stationarity constraints indexed by j have the following left-hand-sides $\alpha_j - \varepsilon p_j d_j + \beta_{ij} + \gamma_{ij} + (\eta + \varepsilon) c_{ij} d_j$. If this term is unequal to $\bar{c}_{ij} d_j$, we can either decrease/increase β_{ij} or γ_{ij} so that their signs do not change. This is possible due to Lemma 3 if ε is sufficiently small.

If $\delta_j > 0$ holds and there exists one tuple $(i, j) \in V_1 \times V_2$ such that $b_{ij} = \gamma_{ij} = 0$ holds. Then, we decrease α_j by $\varepsilon c_{ij} d_j$ so that the corresponding stationarity constraint is satisfied. After this adjustment we can again adjust the remaining stationarity constraints by either increasing or decreasing $\beta_{\tilde{i}j}, \gamma_{\tilde{i}j}$ or δ_j without changing their signs due to Lemma 3.

Finally, we can handle the case $\delta_j > 0$ and $\beta_{ij} < 0$ or $\gamma_{ij} > 0$ by only adjusting these values for a sufficiently small chosen ε.

Because all of the above arguments holds for an arbitrary index j, we can apply these arguments to every index in V_2 by choosing a sufficiently small ε. Consequently, for the considered KKT point $(y, z, \alpha', \beta', \gamma', \delta', \eta')$, we can increase η' by only adjusting $\alpha', \gamma', \beta', \delta'$ while leaving y and z unchanged and still obtain a feasible KKT point, which is a contradiction to the maximal choice of η'. □

We are now ready to state the main result.

Theorem 8. *There exists a point $(y, z, \alpha, \beta, \gamma, \delta, \eta)$ that satisfies the KKT system (16) and such that*

$$0 \geq \eta \geq \eta^*, \tag{17a}$$

$$0 \leq \alpha_j \leq \bar{p}_j d_j + |\eta^*| p_j d_j, \quad \text{for all } j \in V_2, \tag{17b}$$

$$0 \leq \delta_j \leq \bar{p}_j d_j + |\eta^*| p_j d_j, \quad \text{for all } j \in V_2, \tag{17c}$$

$$0 \leq \gamma_{ij} \leq \bar{c}_{ij} d_j + |\eta^*| c_{ij} d_j, \quad \text{for all } i \in V_1, j \in V_2, \tag{17d}$$

$$0 \geq \beta_{ij} \geq -(\bar{p}_j d_j + |\eta^*| p_j d_j), \quad \text{for all } i \in V_1, j \in V_2. \tag{17e}$$

Proof. From Lemma 4, it follows that there exists a feasible KKT point $(y, z, \alpha, \beta, \gamma, \delta, \eta)$ satisfying the dual bound for η, i.e., Conditions (17a).

We now consider this point and prove the bounds for the remaining dual variables. To this end, we consider an arbitrary $j \in V_2$. For α_j, the upper bound directly follows from the stationarity constraint $\alpha_j + \delta_j + \eta p_j d_j = \bar{p}_j d_j$ and $\delta_j \geq 0$. For the lower bound we consider the inequalities

$$\alpha_j = \bar{c}_{ij} d_j - \beta_{ij} - \gamma_{ij} - \eta c_{ij} d_j \geq -\gamma_{ij}, \quad \text{for all } i \in V_1,$$
$$\alpha_j = \bar{p}_j d_j - \delta_j - \eta p_j d_j \geq -\delta_j$$

derived from the stationarity constraints. If $\alpha_j < 0$ holds, then $\delta_j > 0$ and for each $i \in V_1$, $\gamma_{ij} > 0$ hold. Consequently, we can increase α_j and decrease δ_j as well as $\gamma_{ij} > 0$ for all $i \in V_1$ until $\delta_j = 0$ or at least one $\gamma_{ij} = 0$ is satisfied. This modified solution is still valid for the KKT system but satisfies the stated lower bound for α.

The upper bound for δ_j follows from the proven bounds for α and η and the corresponding stationarity constraint.

If $\beta_{ij} < 0$ and $\gamma_{ij} > 0$ hold, then we can equally increase β_{ij} and decrease γ_{ij} until one of these variables is zero. This does not affect the feasibility regarding the KKT system. Consequently, we can assume that at least one of these variables is zero in the considered KKT point. Using this, the stationarity constraint $\alpha_j + \beta_{ij} + \gamma_{ij} + \eta c_{ij} d_j = \bar{c}_{ij} d_j$, and the previously derived lower and upper bounds for α and η, we obtain the stated bounds for β and γ. □

(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, Dr.-Luise-Herzberg-Str. 4, 90461 Nuremberg, Germany
Email address: johannes.thuerauf@utn.de