Open Journal of Mathematical Optimization

Dorothee Henke

The Robust Bilevel Selection Problem

Volume 6 (2025), article no. 4 (35 pages)
https://doi.org/10.5802/ojmo.40

Article submitted on January 7, 2024, revised on August 13, 2024,
accepted on April 10, 2025.

Open Journal of Mathematical Optimization is a member of the
Centre Mersenne for Open Scientific Publishing
http://www.centre-mersenne.org/
e-ISSN: 2777-5860

The Robust Bilevel Selection Problem

Dorothee Henke
University of Passau
School of Business, Economics and Information Systems
Chair of Business Decisions and Data Science
94030 Passau
Germany
dorothee.henke@uni-passau.de

Abstract

In bilevel optimization problems, two players, the leader and the follower, make their decisions in a hierarchy, and both decisions may influence each other. Usually one assumes that both players have full knowledge also of the other player's data. In a more realistic model, uncertainty can be quantified, e.g., using the robust optimization approach: We assume that the leader does not know the follower's objective function precisely, but only knows an uncertainty set of potential follower's objectives, and her aim is to optimize the worst case of the corresponding scenarios. Now the question arises how the computational complexity of bilevel optimization problems changes under the additional complications of this type of uncertainty.

We make a further step towards answering this question by examining an easy bilevel problem. In the Bilevel Selection Problem, the leader and the follower each select some items from their own item set, while a common number of items to select in total is given, and each of the two players minimizes the total costs of the selected items, according to different sets of item costs. We show that this problem can be solved in polynomial time without uncertainty and then investigate the complexity of its robust version. If the leader's item set and the follower's item set are disjoint, it can still be solved in polynomial time in case of discrete uncertainty, interval uncertainty, and discrete uncorrelated uncertainty, using ideas from [6]. Otherwise, we show that the Robust Bilevel Selection Problem becomes NP-hard, even for discrete uncertainty. We present a 2-approximation algorithm and exact exponential-time approaches for this setting, including an algorithm that runs in polynomial time if the number of scenarios is a constant.

Furthermore, we investigate variants of the Bilevel Selection Problem where one or both of the two decision makers take a continuous decision. One variant leads to an example of a bilevel optimization problem whose optimal value may not be attained. For the Robust Continuous Bilevel Selection Problem, where all variables are continuous, we generalize results from [6] and also develop a new approach for the setting of discrete uncorrelated uncertainty, which gives a polynomial-time algorithm for the Robust Continuous Bilevel Selection Problem and a pseudopolynomial-time algorithm for the Robust Bilevel Continuous Knapsack Problem, answering an open question from [6].

Digital Object Identifier 10.5802/ojmo.40

Keywords Bilevel Optimization, Robust Optimization, Combinatorial Optimization, Selection Problem, Computational Complexity.

Acknowledgments The author thanks Kristina Hartmann for her work on several results in this paper in the context of her Master's thesis [23]. In particular, the results in Section 5.2.2 include original ideas of hers and were jointly developed by her and the author of this article, who supervised the thesis. Thanks also to Christoph Buchheim for helpful discussions and for providing feedback on preliminary versions of this article. The author has partially been supported by Deutsche Forschungsgemeinschaft (DFG) under grant no. BU 2313/6.

1 Introduction

In classical optimization problems, there is one decision maker whose aim it is to select a feasible solution that is optimal with respect to some objective function. *Bilevel optimization* considers an important extension of this concept in which there are two decision makers, called the *leader* and the *follower*.[1] Each of them has their own optimization problem (i.e., decision variables, constraints, and objective function), and they select their solutions one after the other, in a hierarchical order: first the leader, then the follower. The key feature is that both

[1] We will always refer to the leader using "she/her" and to the follower using "he/him/his".

decisions may also have an impact on the other decision maker's optimization problem. This interdependence makes bilevel optimization problems very difficult in general.

Bilevel optimization has received a lot of interest lately. A recent overview of more than 1000 references on the topic can be found in [15]. For general introductions to bilevel optimization, we refer to, e.g., [2, 12, 16].

Our focus in this article lies on the perspective of combinatorial optimization and computational complexity. The first complexity results for bilevel optimization problems were obtained by Jeroslow in [25]. His results imply that bilevel linear programs with continuous variables are NP-hard and that bilevel integer linear programs are even Σ_2^{p}-hard. See also [31] for an overview of hardness results for bilevel linear programs, and [36] for illustrations and intuitions regarding the class Σ_2^{p}.

In this article, we will deal with the computational complexity of a specific bilevel optimization problem: The Bilevel Selection Problem is solvable in polynomial time, which, in light of the known complexity results, is unusual for bilevel optimization problems. Since the single-level Selection Problem is solvable in polynomial time, it is not surprising that the Bilevel Selection Problem is not on the second level of the polynomial hierarchy. However, it is not obvious whether it is polynomial-time solvable or NP-hard in the general setting. We show in Section 2.3 that it is indeed polynomial-time solvable.

Moreover, precisely because the Bilevel Selection Problem is relatively easy, we consider it to be a useful starting point for studying the question that motivates this work: We aim for a better understanding of the additional complexity induced by uncertainty in bilevel optimization problems.

An important concept to model uncertainties concerning some parameters of an optimization problem is *robust optimization*. It works with *uncertainty sets* containing all scenarios that are considered to be possible or reasonably likely, and aims at optimizing the worst-case outcome. For an introduction to the field of robust optimization and an overview of important literature, we refer to [10] and the references therein, as well as to the books [3, 29].

Similarly to bilevel optimization, also a robust optimization problem can be understood as the interplay of two decision makers: First, the main decision maker selects a solution, and second, the *adversary* selects a scenario that results in the worst-case objective value. In fact, bilevel optimization and robust optimization are closely related; see [22].

The properties and the complexity of a robust optimization problem highly depend on the structure of the uncertainty set. The following three common types of uncertainty sets will be studied in this article.

- A *discrete uncertainty set* consists of a finite number of scenarios that are explicitly given as part of the input of the robust optimization problem. Typically, the robust counterpart subject to discrete uncertainty is significantly harder than the underlying certain optimization problem. For many combinatorial problems, the robust counterpart is strongly NP-hard in general and weakly NP-hard if the number of scenarios is fixed; see [10]. This is in contrast to the Robust Bilevel Selection Problem and also other robust bilevel optimization problems, where discrete uncertainty seems to introduce less additional complexity than other types of uncertainty, compared to the underlying bilevel problem; see [6, 7].
- In an *interval uncertainty set*, every parameter may independently vary within an interval. Usually, interval uncertainty is not very interesting to study because it can easily be eliminated and therefore does not introduce any additional complexity compared to the underlying certain problem. However, in the robust bilevel setting, it turns out to be more involved; see also [6, 7].
- Another discrete type of uncertainty sets, which we call *discrete uncorrelated uncertainty*, shares the idea of interval uncertainty that each parameter can vary independently in some set, which now is a finite set instead of an interval. Due to the form in which the set is given, the scenarios cannot be enumerated efficiently here, in contrast to discrete uncertainty. However, like interval uncertainty, discrete uncorrelated uncertainty can often be eliminated in a trivial way. Moreover, an uncertainty set can usually be replaced by its convex hull, which, in particular, makes interval uncertainty and discrete uncorrelated uncertainty equivalent. In the robust bilevel setting however, this is not the case in general; see [6, 7]. But, in fact, for the Robust Bilevel Selection Problem, discrete uncorrelated uncertainty turns out to be equivalent to interval uncertainty, for problem-specific reasons (Theorem 9).

The combination of bilevel optimization and uncertainty is a topic that receives increasing attention currently. In a practical setting in which leader and follower are actually two distinct decision makers, it seems to be very natural to adopt some concept of optimization under uncertainty and assume that, e.g., the leader is uncertain about some parameters of the follower's problem. A very recent and comprehensive overview of the area of bilevel optimization under uncertainty is given by Beck, Ljubić, and Schmidt in the survey [1]. It covers both stochastic and robust approaches to model uncertainty, and it classifies various ways of where and how uncertainty can come into play in a bilevel optimization problem. In this article, we focus on uncertainty in the parameters of the follower's objective function from the leader's perspective. The situation can be illustrated by considering three decision makers: The *leader* first chooses a solution, then an *adversary* (of the leader) chooses an objective function for the follower, and the *follower* finally computes an optimal solution according to this objective function and depending on the leader's choice. The leader's aim is to optimize her own objective value, which depends on the follower's choice, while the adversary has the opposite objective. Note that *defender-attacker-defender games* present a similar three-level structure as these robust bilevel optimization problems; see, e.g., [5]. However, there is only one objective function in these problems, as the same decision maker acts on the first and on the third level. Moreover, the attack usually does not affect the objective function, but a combinatorial structure concerning the constraints.

A main goal of this article is to improve the understanding of robust bilevel optimization problems and, in particular, of the additional complexity that this way of respecting uncertainty introduces in bilevel optimization problems. For a more general class of robust bilevel optimization problems, this question has been studied in [7]: It has been shown that interval uncertainty can turn an NP-equivalent bilevel optimization problem into a Σ_2^{p}-hard robust problem, while a robust bilevel optimization problem with discrete uncertainty is at most one level higher in the polynomial hierarchy compared to the follower's problem. In this article, our underlying bilevel optimization problem is solvable in polynomial time; therefore, we can expect to obtain robust problems that are either polynomial-time solvable or NP-equivalent.

As a specific robust bilevel optimization problem to investigate, we develop a natural problem formulation that is based on the single-level SELECTION PROBLEM, which can be seen as one of the most simple combinatorial optimization problems and which is often studied in the field of robust optimization in the context of complexity questions. For robust versions of the SELECTION PROBLEM, see, e.g., [26] and [29, Chapter 3.3].

A related robust bilevel optimization problem, namely the ROBUST BILEVEL CONTINUOUS KNAPSACK PROBLEM, together with its complexity for different types of uncertainty sets, has been studied in [6]. A part of the current work generalizes and extends the results of [6], with a stronger emphasis on combinatorial underlying problems. The (CONTINUOUS) BILEVEL SELECTION PROBLEM with disjoint item sets is more general than the BILEVEL CONTINUOUS KNAPSACK PROBLEM in one aspect, but structurally simpler than it in another; see Section 2.2 for more details. For most of the robust settings we consider, we obtain similar results as in [6], but we will also see some differences. In particular, solving the ROBUST CONTINUOUS BILEVEL SELECTION PROBLEM under discrete uncorrelated uncertainty requires some new ideas, which also lead to a pseudopolynomial-time algorithm for the ROBUST BILEVEL CONTINUOUS KNAPSACK PROBLEM; see Section 6.2.3 and Appendix A. The latter result answers an open question stated in [6].

In addition to the results mentioned above, which concern the BILEVEL SELECTION PROBLEM with disjoint item sets that is closely related to [6], we also study a more general version of the BILEVEL SELECTION PROBLEM where some items might be controlled by both decision makers. We show that it can still be solved in polynomial time (Theorem 4), but its robust version becomes NP-hard in case of discrete uncertainty (Theorem 15). However, we give a 2-approximation algorithm for this case (see Section 5.2.2) and also show how the problem can be solved in exponential time (see Section 5.2.3). One of our approaches implies that, if the number of scenarios in a discrete uncertainty set is a constant, the ROBUST BILEVEL SELECTION PROBLEM can be solved in polynomial time (Theorem 24).

Besides the BILEVEL CONTINUOUS KNAPSACK PROBLEM, which has been studied under robust uncertainty in [6] and under stochastic uncertainty in [9], also other bilevel combinatorial optimization problems have been investigated in the literature, although not in the uncertain setting we consider here. The complexity of several bilevel variants of the KNAPSACK PROBLEM has been studied in [11]. One of them, which has been introduced in [30], can be seen as a generalization of our BILEVEL SELECTION PROBLEM in case of disjoint item sets; see also Section 2.2.1. The results in [11, 30] imply that it is Σ_2^{p}-hard and solvable in pseudopolynomial time. Other bilevel combinatorial optimization problems whose complexity has been studied are, e.g., the bilevel minimum spanning tree problem [8, 33] and the bilevel assignment problem [19, 21]. Note that the bilevel minimum spanning tree problem and the BILEVEL SELECTION PROBLEM can both be seen as special cases of a bilevel minimum matroid basis problem.

This article is organized as follows. In Section 2, we introduce the BILEVEL SELECTION PROBLEM and discuss several variants of the problem, some of which are distinguished later on. We show how the BILEVEL SELECTION PROBLEM can be solved in polynomial time in Section 2.3. The robust problem version is introduced in Section 3. We then present algorithms for the adversary's problem, for different types of uncertainty sets, in Section 4, before turning to the robust leader's problem in Section 5. Here we give polynomial-time algorithms for the simpler case of disjoint item sets in Section 5.1 and show our hardness and algorithmic results for the general case in Section 5.2. Moreover, we give some insights regarding problem variants with continuous variables in Section 6. This includes an example of a bilevel optimization problem whose optimal value may not be attained (Example 25) and some structural properties in Section 6.1 and in the beginning of Section 6.2. We then study the Robust Continuous Bilevel Selection Problem for different types of uncertainty sets. Section 7 concludes. In Appendix A, a new approach developed in Section 6.2 is finally extended to a pseudopolynomial-time algorithm for the Robust Bilevel Continuous Knapsack Problem with discrete uncorrelated uncertainty.

Many of the results of Sections 2 to 5 have already appeared in the unpublished Master's thesis [23].

2 The Bilevel Selection Problem Without Uncertainty

In this section, we introduce the Bilevel Selection Problem, which is an example of a combinatorial bilevel optimization problem that can be solved in polynomial time. It will serve as a basis for the robust problem versions that we will investigate in the remainder of this article. We define the Bilevel Selection Problem in Section 2.1 and discuss several variants of it, together with their relation to bilevel variants of the Knapsack Problem, in Section 2.2, before presenting a polynomial-time algorithm in Section 2.3.

2.1 Problem Definition

We first define the (single-level) Selection Problem as follows: Given a finite set $\mathcal{E}$ of items, a number $b \in \{0, \dots, |\mathcal{E}|\}$, and item costs $c\colon \mathcal{E} \to \mathbb{Q}$, find a subset $X \subseteq \mathcal{E}$ such that $|X| = b$ and $c(X) := \sum_{e\in X} c(e)$ is minimal. This problem is easy to solve in polynomial time because it is clearly optimal to select b items that have the smallest values of c, for example, by sorting the items in $\mathcal{E}$ by increasing values of c and then selecting the first b items of this order. This algorithm has running time $O(n \log n)$, where $n = |\mathcal{E}|$. However, explicitly sorting the items is not even necessary: It suffices to find the b-th best item $e \in \mathcal{E}$ with respect to c, which amounts to computing a weighted median, and then select the set of all items with costs at most $c(e)$. This can be done in linear time $O(n)$ [4]; see also [28, Chapter 17.1].

In our bilevel version of the Selection Problem, we now consider two players, the leader and the follower, who select items from different item sets and according to different item costs, but subject to a common capacity constraint. More specifically, we are given finite leader's and follower's item sets $\mathcal{E}_l$ and $\mathcal{E}_f$, respectively, a number $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$, and leader's and follower's item costs $c\colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$ and $d\colon \mathcal{E}_f \to \mathbb{Q}$, respectively. The task is to find a subset $X \subseteq \mathcal{E}_l$ for the leader such that $c(X \cup Y)$ is minimal, where $Y \subseteq \mathcal{E}_f \setminus X$ is a follower's subset that satisfies $|X \cup Y| = b$ and minimizes $d(Y)$.

Note that we do not require the sets $\mathcal{E}_l$ and $\mathcal{E}_f$ to be disjoint in general. However, we will see that assuming the sets to be disjoint leads to an important special case, actually simplifying the problem in some sense. We denote the cardinalities of the item sets by $n_l = |\mathcal{E}_l|$, $n_f = |\mathcal{E}_f|$, and $n = |\mathcal{E}_l \cup \mathcal{E}_f|$ throughout this article.

In the typical form of a bilevel optimization problem, the Bilevel Selection Problem can be written as follows:

$$
\begin{aligned}
\min_{X}\ & c(X \cup Y) \\
\text{s.t. } & X \subseteq \mathcal{E}_l \\
& Y \in \operatorname*{argmin}_{Y'}\ d(Y') \\
& \qquad \text{s.t. } Y' \subseteq \mathcal{E}_f \setminus X \\
& \qquad\qquad |X \cup Y'| = b
\end{aligned}
\tag{BSP}
$$

The leader can be imagined to make her choice $X \subseteq \mathcal{E}_l$ before the follower and, at the same time, anticipate the follower's optimal solution $Y \subseteq \mathcal{E}_f \setminus X$, while optimizing her objective function c on all items selected by the leader or the follower. Leader's subsets $X \subseteq \mathcal{E}_l$ that do not allow for a feasible follower's response, i.e., for some $Y' \subseteq \mathcal{E}_f \setminus X$ such that $|X \cup Y'| = b$, are considered infeasible for the leader's problem. However, the requirement $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$ ensures that a feasible (leader's and corresponding follower's) solution always exists.

From the follower's perspective, a feasible leader's solution $X \subseteq \mathcal{E}_l$ is already fixed and the task is to select a subset Y' of $b - |X|$ items from the set $\mathcal{E}_f \setminus X$, while minimizing the total costs $d(Y')$ of the selected items. This amounts to a standard single-level Selection Problem and can thus be solved in a greedy way by sorting the follower's items by their costs d, or by computing a weighted median; see above.

Note that the follower's objective function considers only the items selected by the follower, while the leader's costs of all items selected by either player appear in the leader's objective. This is also due to the fact that the leader's solution X is fixed from the follower's perspective. If there were additional follower's item costs for the items in X, they would only amount to a constant term in his objective and would therefore not change the problem.

▶ Remark 1. The above definition of the Bilevel Selection Problem and also the formulation in (BSP) are imprecise if the optimal follower's solution is not unique. As usual in bilevel optimization, the optimistic and the pessimistic setting can be distinguished, i.e., the follower can be assumed to decide either in favor or to the disadvantage of the leader among his optimal solutions. Here, this corresponds to the follower minimizing or maximizing $c(Y')$ as a secondary objective, respectively. For the follower's greedy algorithm this means that c is used as a secondary criterion when non-uniqueness arises in sorting the follower's items with respect to d. More precisely, the required order can be defined such that an item $e_1 \in \mathcal{E}_f$ precedes an item $e_2 \in \mathcal{E}_f$ if either $d(e_1) < d(e_2)$ or $d(e_1) = d(e_2)$ and $c(e_1) < c(e_2)$ $[c(e_1) > c(e_2)]$ in the optimistic [pessimistic] setting.

This already indicates that, for the Bilevel Selection Problem, there is not a significant difference between the two settings from an algorithmic perspective. In fact, as soon as there is any fixed order according to which the follower selects his items greedily, the problem is well-defined and the follower's optimal solution can always be assumed to be unique. Therefore, we will often implicitly assume that a fixed follower's greedy order is given, such that it will usually not be required to discuss the optimistic and the pessimistic setting in detail.

In fact, from the leader's perspective, only this order of the follower's items is relevant, but not the precise follower's item costs. Accordingly, when constructing an instance, we will sometimes only define a follower's greedy order instead of a function d; see, e.g., Theorem 15 and Examples 19 and 20. Observe that, given a follower's greedy order, an appropriate function d (with polynomial-size values) can easily be constructed in polynomial time, e.g., by setting its values to subsequent natural numbers in this order.

2.2 Problem Variants and Relation to Knapsack Problems

In this section, we introduce and discuss several assumptions and modifications that can be made for the Bilevel Selection Problem as we defined it in Section 2.1. This provides context on related problems in the literature and motivates which problem variants will be distinguished in the rest of this article.

2.2.1 Item Sets

As mentioned in Section 2.1, we will mainly distinguish the special case of the Bilevel Selection Problem where $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$ from the general one (see Section 5). This special case of the Bilevel Selection Problem can be considered to be a special case of the Bilevel Knapsack Problem that has been studied in [11, 30]. In fact, the polynomial-time algorithm presented for the Bilevel Selection Problem in Section 2.3, in case of disjoint item sets, can be seen as a special case of the pseudopolynomial-time approaches in [11, 30]. However, our algorithm also works for the general Bilevel Selection Problem, where the situation is more involved.

Moreover, one could think of a simpler bilevel variant of the Selection Problem in which only the follower selects items in the sense of a single-level Selection Problem, while the leader just decides on the number b of items to select and gets some value (which might be positive or negative) from each selected item. This problem variant can be seen as a special case of the Bilevel Knapsack Problem in which the leader provides the capacity for the follower's knapsack problem. This problem has been studied in [11, 17], and the continuous version of it is the underlying Bilevel Continuous Knapsack Problem in [6, 9]. In fact, this bilevel selection problem is equivalent to a special case of the Bilevel Selection Problem as defined above, with disjoint item sets: Consider $\mathcal{E}_l$ and $\mathcal{E}_f$ to be disjoint and the leader's costs of all leader's items to be equal to zero. Then, by selecting some subset $X \subseteq \mathcal{E}_l$ of her items, the leader just determines the capacity $b - |X|$ for the follower and pays her item costs of all items selected by the follower. Therefore, the algorithmic results we obtain in Sections 2.3, 5.1, and 6 can also be applied to this bilevel selection problem, and in fact are similar to the ones in [6]. In Appendix A, we extend our new algorithms for the Robust Continuous Bilevel Selection Problem with discrete uncorrelated uncertainty to the Robust Bilevel Continuous Knapsack Problem.

It is also worth mentioning another special case regarding the two item sets, namely the one where $\mathcal{E}_l \subseteq \mathcal{E}_f$. This assumption can be seen as reasonable and convenient because it ensures that there is always a feasible follower's solution, no matter which (at most b) items the leader has selected. In the general setting, the leader might have to be careful about selecting enough items from $\mathcal{E}_l \setminus \mathcal{E}_f$ in order to obtain a feasible solution. However, assuming $\mathcal{E}_l \subseteq \mathcal{E}_f$ is not a restriction because an instance of the general Bilevel Selection Problem can be transformed into an equivalent instance satisfying this assumption. For this, add all items in $\mathcal{E}_l \setminus \mathcal{E}_f$ to $\mathcal{E}_f$ and set their follower's item costs to some constant larger than all other follower's item costs. Then the follower will not select any of these items unless the leader chooses a solution that is infeasible in the original instance.

2.2.2 Continuous Variables

Another detail that can be altered in the definition of the Bilevel Selection Problem, in order to obtain other possibly interesting variants of the problem, is the integrality of the leader's and the follower's decisions. More precisely, one could allow the two players (or only one of them) to select fractions of items. This is related to the situation of the Bilevel Continuous Knapsack Problem in [6]. We will discuss this setting in more detail in Section 6.

2.2.3 Maximization

The Selection Problem and also the Bilevel Selection Problem are defined as minimization problems according to Section 2.1, but could analogously be defined as maximization problems, i.e., the decision makers could maximize the total value of their selection of items instead of minimizing the total costs. Then the problems can be more directly seen as special cases of the classical Knapsack Problem and a bilevel variant of it; see also Section 2.2.1.

Although we will work with the minimization problem version in the following, almost all of our arguments and approaches are also valid for the maximization version of the problem. Only the approximation procedure in Section 5.2.2 appears to be specific to the minimization problem because it is restricted to nonnegative item costs; see also Section 2.2.5.

2.2.4 Capacity Constraint

In relation to the classical Knapsack Problem, it might be more natural to replace "$= b$" by "$\leq b$" in the capacity constraint (or by "$\geq b$" in the minimization version). In the maximization version, this means that the follower is not forced to use the provided capacity completely if this would mean to select items having a negative value for him. Accordingly, in the minimization version, the follower is allowed to select all items of negative cost, even if there are more than b of them. From the algorithmic perspective, this is only a minor change: In an optimal follower's solution, no [all] items of negative value [cost] are selected in the minimization [maximization] version, and the greedy approach on all other items is still the best strategy. This insight enables to model this problem variant in terms of our original Bilevel Selection Problem, e.g., by removing items for which the decision is clear or adding dummy items, and possibly adjusting the capacity.

We focus on the problem variant with an equality constraint in this article because our arguments often become easier when the number of items in a solution is known in advance.

2.2.5 Item Costs

On a related note, one could also investigate how the Bilevel Selection Problem changes when only allowing for nonnegative leader's and/or follower's item costs (or nonnegative item values in the maximization version), in contrast to arbitrary rational values as introduced so far. In fact, the setting of arbitrary item costs can be easily reduced to any of the restricted settings by adding appropriate constants $C \in \mathbb{Q}_{>0}$ and $D \in \mathbb{Q}_{>0}$ to all leader's and follower's item costs, respectively. The resulting problem is equivalent because, by the capacity constraint $|X \cup Y'| = b$, this changes both the leader's and the follower's objective function value of any given solution (in the latter case assuming that some fixed leader's solution X is given) only by a constant. Note that the distinction between equality and inequality constraints from Section 2.2.4 is not necessary if all costs are nonnegative because we may assume that exactly b items are selected anyway.

In Section 5.2.2, we will deal with approximation algorithms and, for this purpose, assume all leader's item costs to be nonnegative. In terms of approximation results, this is a restriction because the above construction does not preserve any approximation guarantees.

2.3 A Polynomial-Time Algorithm

In order to develop a polynomial-time algorithm for the Bilevel Selection Problem, we first focus on the follower's problem. As already noted in Section 2.1, the follower solves a single-level Selection Problem, which can be done in linear time by selecting the best follower's items according to his (and possibly the leader's, in case of non-uniqueness) item costs. From the follower's perspective, there is no difference among the problem variants concerning the item sets (see Section 2.2.1) because the leader's solution X is already fixed and the item set the follower chooses from is given by $\mathcal{E}_f \setminus X$, regardless of the relation between $\mathcal{E}_l$ and $\mathcal{E}_f$.

We now turn to the leader's perspective and first look at the special case with disjoint item sets. In this case, it is easy to see that the only property of the leader's solution that is relevant for the follower is the number $|X|$ of items she selects. Therefore, for any fixed partition of the total capacity b into a leader's capacity b_l and a follower's capacity b_f, each of the two players will select their number of items from their set using a greedy strategy. The optimal partition of the capacity can be determined by enumerating all possible partitions. This leads to Algorithm 1.

Algorithm 1: Algorithm for the Bilevel Selection Problem with $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$ or $\mathcal{E}_l \subseteq \mathcal{E}_f$

Input : finite sets $\mathcal{E}_l$ and $\mathcal{E}_f$ with $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$ or $\mathcal{E}_l \subseteq \mathcal{E}_f$, $n_l = |\mathcal{E}_l|$, $n_f = |\mathcal{E}_f|$, and $n = |\mathcal{E}_l \cup \mathcal{E}_f|$, $b \in \{0, \dots, n\}$, $c \colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$, $d \colon \mathcal{E}_f \to \mathbb{Q}$

Output : an optimal solution (X, Y) of the Bilevel Selection Problem

1 compute a bijection $p_l \colon \{1, \dots, n_l\} \to \mathcal{E}_l$ such that $c(p_l(1)) \leq \dots \leq c(p_l(n_l))$

2 compute a bijection $p_f \colon \{1, \dots, n_f\} \to \mathcal{E}_f$ such that $d(p_f(1)) \leq \dots \leq d(p_f(n_f))$ and $c(p_f(i)) \leq c(p_f(i+1))$ [or $c(p_f(i)) \geq c(p_f(i+1))$] for all $i \in \{1, \dots, n_f - 1\}$ with $d(p_f(i)) = d(p_f(i+1))$ in the optimistic [pessimistic] setting

3 $b_l^- := \max\{0, b - n_f\}$

4 $b_l^+ := \min\{b, n_l\}$

5 **for** $b_l = b_l^-, \dots, b_l^+$ **do**

6 $\quad X_{b_l} := \{p_l(1), \dots, p_l(b_l)\}$

7 $\quad b_f := b - b_l$

8 $\quad Y_{b_l} := \{p_f(1), \dots, p_f(m)\} \setminus X_{b_l}$, where $m \in \{0, \dots, n_f\}$ is chosen such that $|Y_{b_l}| = b_f$

9 **return** $\operatorname{argmax}\{c(X \cup Y) \mid (X, Y) \in \{(X_{b_l^-}, Y_{b_l^-}), \dots, (X_{b_l^+}, Y_{b_l^+})\}\}$

▶ **Theorem 2.** *Algorithm 1 solves the Bilevel Selection Problem in case of $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$ in time $O(n \log n)$.*

Proof. From the follower's perspective, only the number $|X|$ of items selected by the leader is relevant, not the specific leader's solution X, because the item set in the follower's Selection Problem is always $\mathcal{E}_f$ in case of disjoint sets $\mathcal{E}_l$ and $\mathcal{E}_f$, and only the number $b - |X|$ of items to select from $\mathcal{E}_f$ is influenced by the leader. Hence, for any fixed cardinality of the leader's solution, selecting the corresponding number of items greedily from her item set $\mathcal{E}_l$ is optimal for the leader. The algorithm now enumerates all feasible numbers b_l of leader's items to select, together with the corresponding greedy leader's and follower's solutions, respectively. Note that the order of the follower's items determined by p_f correctly represents the optimistic and the pessimistic setting; see also Remark 1. Moreover, b_l^- and b_l^+ are defined such that exactly the numbers b_l are enumerated that lead to feasible solutions, given that exactly b items have to be selected in total.

The running time of Algorithm 1 is dominated by sorting the leader's and the follower's items in Lines 1 and 2, which can each be done in time $O(n \log n)$. Indeed, the loop in Lines 5 to 8 can be implemented to run in linear time by updating X_{b_l} and Y_{b_l} in every iteration. Note that $\mathcal{E}_l$ and $\mathcal{E}_f$, and hence also X_{b_l} and $\mathcal{E}_f$, are disjoint in the current setting such that Y_{b_l} is simply chosen as $\{p_f(1), \dots, p_f(b_f)\}$ in Line 8. ◀

▶ Remark 3. As mentioned in Section 2.1, a single-level Selection Problem can also be solved in linear time, without the need to sort all items. Hence, a single iteration of the loop in Algorithm 1 could be implemented to run in linear time, even without Lines 1 and 2. However, the overall algorithm becomes faster in general if we precompute the greedy orders, as this results in a running time of $O(n \log n)$ instead of $O(n^2)$. The reason is that we can use the same orders for all iterations, which correspond to Selection Problems on the same item sets, but with different capacities.

Next, we will show that, also in the general case, the Bilevel Selection Problem can be solved in polynomial time. In fact, we can still apply Algorithm 1, although this is less obvious than in the case of disjoint item sets. We prove this for the case of $\mathcal{E}_l \subseteq \mathcal{E}_f$, which is equivalent to the general setting; see Section 2.2.1.

▶ **Theorem 4.** *Algorithm 1 solves the Bilevel Selection Problem in case of $\mathcal{E}_l \subseteq \mathcal{E}_f$ in time $O(n \log n)$.*

Proof. We first argue that the analysis of the running time from the proof of Theorem 2 is also valid here. Indeed, it is still true that the loop in Lines 5 to 8 can be implemented to run in linear time by updating X_{b_l} and Y_{b_l} in every iteration. Here, the new follower's solution Y_{b_l} is obtained from Y_{b_l-1} by either removing the item $p_l(b_l)$ that was just added to the leader's solution if $p_l(b_l) \in Y_{b_l-1}$, or otherwise removing the item $p_f(m)$ from the previous iteration, i.e., decreasing m by one.

It remains to show that Algorithm 1 is still correct for the setting of $\mathcal{E}_l \subseteq \mathcal{E}_f$. Note that, if $\mathcal{E}_l \subseteq \mathcal{E}_f$, all sets $X \subseteq \mathcal{E}_l$ with $|X| \leq b$ are feasible leader's solutions because the follower is always able to select the remaining number of items, assuming $b \leq n = n_f$. In particular, we always have $b_l^- = 0$.

Let p_l be the bijection computed in Line 1 of Algorithm 1. Given some set $X \subseteq \mathcal{E}_l$, we define

$$i_1(X) := \begin{cases} 0 & \text{if } X = \emptyset \\ \max\{i \in \{1, \dots, n_l\} : p_l(i) \in X\} & \text{otherwise} \end{cases}$$

$$i_2(X) := \begin{cases} n_l + 1 & \text{if } X = \mathcal{E}_l \\ \min\{i \in \{1, \dots, n_l\} : p_l(i) \notin X\} & \text{otherwise} \end{cases}$$

and look at the difference $D(X) := i_1(X) - i_2(X)$. We first observe that, for all leader's solutions X_{b_l} in Algorithm 1, we have $D(X_{b_l}) = b_l - (b_l + 1) = -1$, and for all feasible leader's solutions $X \subseteq \mathcal{E}_l$ that do not have this structure, $D(X) > 0$ holds.

Let X be an optimal leader's solution with minimal value $D(X)$. If $D(X) = -1$, Algorithm 1 inspects the solution X and thus returns an optimal solution. Otherwise, we will construct another optimal solution $\overline{X}$ with $D(\overline{X}) < D(X)$, contradicting the assumption that $D(X)$ is minimal. Hence, we will assume $D(X) > 0$ from now on.

Let $e_1 := p_l(i_1(X)) \in X$ and $e_2 := p_l(i_2(X)) \in \mathcal{E}_l \setminus X$, which is well-defined because $\emptyset \subset X \subset \mathcal{E}_l$ in the case where X is feasible but not considered by the algorithm. As we assume $D(X) > 0$, we have $i_1(X) > i_2(X)$ and hence $c(e_1) \geq c(e_2)$. We consider three operations that can be applied to X:

- removing e_1,
- adding e_2, and
- both removing e_1 and adding e_2.

Note that all three operations reduce the value $D(X)$: Removing e_1 decreases $i_1(X)$ and does not change $i_2(X)$ if $i_1(X) > i_2(X)$ was true before. Analogously, adding e_2 increases $i_2(X)$ and does not change $i_1(X)$. When performing both actions one after the other, $D(X)$ is decreased twice if $D(X) > 0$ still holds after the first one. Otherwise, $D(X) = -1$ already holds after the first action and still after the second one.

We now distinguish four cases and show that, in each of them, one of the three operations leads to an optimal solution again. Let Y denote the optimal follower's choice corresponding to X.

Case 1: $e_2 \in Y$. In this case, $|X| < b$ holds and hence, it is still feasible for the leader to choose $X \cup \{e_2\}$. This leads to the follower choosing $Y \setminus \{e_2\}$ since he can select one item less, while one of the items he selected before is not available anymore, so the selection of the other items is still optimal. Thus, the overall solution $X \cup Y$ does not change and $X \cup \{e_2\}$ is still optimal for the leader.

From now on, we assume $e_2 \notin Y$. For the following cases, consider the optimal follower's solution $\overline{Y}$ corresponding to the leader's choice $\overline{X} = X \setminus \{e_1\}$. The follower chooses the same items as before and one additional one, which we call $\bar{e}$, i.e., $\overline{Y} = Y \cup \{\bar{e}\}$.

Case 2: $\bar{e} = e_1$. The overall solution $X \cup Y$ does not change when the leader removes e_1 from her solution X because the follower will take it instead. Hence, $\overline{X}$ is optimal for the leader.

Case 3: $\bar{e} = e_2$. This means that, in the overall solution $X \cup Y$, the item e_1 is replaced by e_2 when the leader chooses $X \setminus \{e_1\}$ instead of X. As $c(e_1) \geq c(e_2)$, this does not make the solution worse for the leader, so it is still optimal.

Case 4: $\bar{e} \notin \{e_1, e_2\}$. Hence, when the leader chooses $\overline{X}$, neither e_1 nor e_2 occur in the overall solution. If the leader adds e_2 to her solution $\overline{X}$ now, the follower will leave out $\bar{e}$ again, but not change his solution apart from that, i.e., he will choose Y again. Therefore, the leader choosing $X \setminus \{e_1\} \cup \{e_2\}$ leads to an overall solution $X \cup Y \setminus \{e_1\} \cup \{e_2\}$, which is still optimal for the leader due to $c(e_1) \geq c(e_2)$. ◀

As argued in Section 2.2.1, the assumption $\mathcal{E}_l \subseteq \mathcal{E}_f$ is not a relevant restriction. Accordingly, with a small modification, Algorithm 1 also solves the Bilevel Selection Problem in the general case:

▶ **Corollary 5.** *The Bilevel Selection Problem can be solved in time $O(n \log n)$.*

3 Definition of the Robust Bilevel Selection Problem

In this section, we start to investigate the Bilevel Selection Problem under uncertainty. More precisely, we assume that the follower's objective function is uncertain from the leader's perspective, and apply the concept of robust optimization, i.e., we assume that an uncertainty set $\mathcal{U} \subseteq \mathbb{R}^{\mathcal{E}_f}$ is given, from which an adversary (of the leader) draws a function $d \colon \mathcal{E}_f \to \mathbb{R}$ of follower's item costs such that the result is as bad as possible for the leader. As an extension of the formulation in (BSP), the Robust Bilevel Selection Problem can be written as follows:

$$\begin{aligned} \min_{X \subseteq \mathcal{E}_l} \ \max_{d \in \mathcal{U}} \ & c(X \cup Y) \\ \text{s.t. } & Y \in \operatorname*{argmin}_{Y'} \ d(Y') \\ & \qquad \text{s.t. } Y' \subseteq \mathcal{E}_f \setminus X \\ & \qquad \qquad |X \cup Y'| = b \end{aligned} \tag{RBSP}$$

The situation of the three decision makers can now be imagined as follows: From the follower's perspective, nothing changes compared to the version without uncertainty, as the leader's and the adversary's decisions are already fixed at this point. Hence, the follower still solves a single-level Selection Problem. The adversary chooses an objective function d for the follower from a given uncertainty set $\mathcal{U}$. We observe that, by doing so, he influences the order of the follower's items that the follower will then use for his greedy solution. The adversary's objective is to choose d such that the resulting follower's optimal solution Y has the worst possible costs $c(Y)$ for the leader. Finally, the leader's task is to anticipate all of this when choosing her own solution X in the beginning such that the resulting overall set $X \cup Y$ of selected items has the minimal possible costs $c(X \cup Y)$ for her.

The adversary's impact on the leader heavily depends on the type of the uncertainty set $\mathcal{U}$. In fact, uncertainty sets of a specific structure can lead to specific structures of the possible follower's greedy orders and therefore of the worst-case follower's reactions that the leader has to take into account. We will discuss the adversary's problem in Section 4, for discrete uncertainty, interval uncertainty, and discrete uncorrelated uncertainty. In Section 5, we then turn to the leader's problem. Here, the general case turns out to be significantly harder than the special case where $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$; see Sections 5.1 and 5.2. Note that the distinction between these two cases is only relevant from the leader's perspective because, after a leader's solution X is fixed, the adversary and the follower just operate on the remaining follower's items $\mathcal{E}_f \setminus X$. Therefore, it is reasonable to first discuss the adversary's problem in Section 4, and distinguish the two cases regarding the item sets only in Section 5.

4 The Adversary's Problem

In this section, we always assume a fixed feasible leader's solution $X \subseteq \mathcal{E}_l$ to be given, and consider the adversary's problem of choosing a scenario from the uncertainty set $\mathcal{U}$ that is a worst possible one for the leader. This problem highly depends on the structure of the uncertainty set and we investigate it for three common types of uncertainty sets in the following. The easiest case from the adversary's perspective is the one of discrete uncertainty in Section 4.1. More work is required to handle the case of interval uncertainty. However, we still achieve a polynomial-time algorithm for this setting in Section 4.2. The strategy is very similar to the one for handling interval uncertainty in the Robust Bilevel Continuous Knapsack Problem in [6]. Finally, the case of discrete uncorrelated uncertainty is investigated in Section 4.3. It turns out to be equivalent to interval uncertainty, now in contrast to [6] where the adversary's problem is NP-hard in case of discrete uncorrelated uncertainty.

4.1 Discrete Uncertainty

First, we focus on the case of discrete uncertainty, where the uncertainty set $\mathcal{U}$ is a finite set that is explicitly given in the input. In this case, it is easy to see that the adversary's problem can be solved in polynomial time by enumerating all scenarios:

▶ **Theorem 6.** *For any fixed feasible leader's solution of the* Robust Bilevel Selection Problem *with a discrete uncertainty set* $\mathcal{U}$*, the adversary's problem can be solved in time* $O(|\mathcal{U}|n)$*.*

Proof. Let $b_f = b - b_l$ be the capacity that is to be filled by the follower, where $b_l = |X|$ is the capacity the given leader's solution $X \subseteq \mathcal{E}_l$ uses. The adversary's task is to choose follower's item costs $d \in \mathcal{U}$ such that the resulting follower's solution is a worst possible one for the leader. We know that the follower's problem, for fixed item costs, is a single-level Selection Problem on $\mathcal{E}_f \setminus X$ with capacity b_f. Such a problem can be solved in linear time; see Section 2.1. When the adversary enumerates all scenarios in $\mathcal{U}$ and computes the resulting follower's solution, together with its leader's costs, for each of them, he thus achieves a running time of $O(|\mathcal{U}|n)$. ◀

This result is in line with the observation in [7] that, also in more general robust bilevel optimization problems, the adversary's problem is on the same level of the polynomial hierarchy as the follower's problem in case of discrete uncertainty, since the adversary can always enumerate all scenarios and solve the corresponding follower's problems.

4.2 Interval Uncertainty

In the case of interval uncertainty, i.e., when the uncertainty set has the form $\mathcal{U} = \prod_{e \in \mathcal{E}_f} [d^-(e), d^+(e)]$ with given functions $d^-, d^+ \in \mathbb{Q}^{\mathcal{E}_f}$ that satisfy $d^-(e) \leq d^+(e)$ for all $e \in \mathcal{E}_f$, we again present a polynomial-time algorithm for the adversary's problem. However, in contrast to the typical setting in robust optimization, it is not true in general for our type of robust bilevel optimization problems that, e.g., the worst case is obtained when all values attain their upper bounds; see also [7].

As already noted in Section 3, the scenarios arising from an uncertainty set in the Robust Bilevel Selection Problem affect the greedy order of the follower's items the follower then uses to determine his solution. For the adversary and the leader, only this order resulting from a scenario is relevant, but not the actual follower's item costs. For this reason, we study which orders can be induced by the scenarios in an interval uncertainty set. We observe that the possible orders correspond to an interval order; see Lemma 7. For more details on interval orders, we refer to, e.g., [32] and the references therein.

The connection between follower's solutions and (prefixes of) interval orders, and also the strategy we will then use for the adversary's algorithm, are very similar to how the case of interval uncertainty is handled in Section 4 of [6] for the Robust Bilevel Continuous Knapsack Problem, and in particular for the adversary's problem there. Therefore, we only briefly repeat the arguments and omit the details here. In fact, the current situation is simpler than the one in [6] because, in contrast to the setting of [6], we are not allowed to select fractions of items here and therefore do not have to care about the special role of the fractional item in a follower's solution.

For the sake of a simpler exposition, we assume that there are no one-point intersections between intervals involved in $\mathcal{U}$. More precisely, we assume that, for all $e_1, e_2 \in \mathcal{E}_f$, we have $d^-(e_1) \neq d^+(e_2)$. Handling the case with one-point intersections requires to deal with some technical details related to the optimistic and pessimistic setting more carefully; see Remark 2 in [6].

Intuitively, items whose follower's costs lie in disjoint intervals in $\mathcal{U}$, always have the same order in the follower's greedy order, independently of the adversary's decision. On the other hand, if the intervals corresponding to two follower's items intersect, then the adversary can decide on the order in which the items will be considered by the follower. This property gives some insight into the structure of the follower's optimal solutions, which can formally be expressed as follows:

▶ **Lemma 7.** *Given an interval uncertainty set* $\mathcal{U} = \prod_{e \in \mathcal{E}_f} [d^-(e), d^+(e)]$*, define a binary relation* $\prec_{\mathcal{U}}$ *on* $\mathcal{E}_f$ *such that, for* $e_1, e_2 \in \mathcal{E}_f$*, we have* $e_1 \prec_{\mathcal{U}} e_2$ *if and only if* $d^+(e_1) < d^-(e_2)$*. Then the relation* $\prec_{\mathcal{U}}$ *is an interval order. In the* Robust Bilevel Selection Problem *with the interval uncertainty set* $\mathcal{U}$*, the set of greedy orders of the follower's items that arise from the scenarios in* $\mathcal{U}$ *is exactly the set of linear extensions of the interval order* $\prec_{\mathcal{U}}$ *on* $\mathcal{E}_f$*. Moreover, the set of follower's optimal solutions, given a fixed feasible leader's solution* $X \subseteq \mathcal{E}_l$ *and any scenario in* $\mathcal{U}$*, is exactly the set of prefixes* $\mathcal{E}_f^0$ *of the restriction of* $\prec_{\mathcal{U}}$ *to* $\mathcal{E}_f \setminus X$ *that satisfy* $|\mathcal{E}_f^0| = b - |X|$*.*

Using Lemma 7, we could now compute the interval order and its linear extensions explicitly in order to solve the adversary's problem. However, this would not give a polynomial-time algorithm in general because the number of linear extensions can be exponential.

Nevertheless, we can solve the adversary's problem in polynomial time. For this, we use that it can be seen as a special case of the *precedence constraint knapsack problem* and that this problem can be solved in pseudopolynomial time by an algorithm described in [35] if the precedence constraints correspond to an interval order; see [6] for more details. The pseudopolynomial-time algorithm runs in polynomial time here because we consider a selection problem instead of a knapsack problem, i.e., the item sizes are all 1 in our case. This results in Algorithm 2 for solving the adversary's problem.

In Line 11 of Algorithm 2, an appropriate subroutine is called for solving a Selection Problem in linear time. The input of the subroutine consists of a finite item set, a number of items to select and a function of item costs; see also the definition of the Selection Problem in Section 2.1.

Algorithm 2: Algorithm for the adversary's problem of the Robust Bilevel Selection Problem with interval uncertainty

---

Input : finite sets $\mathcal{E}_l$ and $\mathcal{E}_f$, $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$, $c\colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$, an interval uncertainty set $\mathcal{U}$ given by $d^-, d^+\colon \mathcal{E}_f \to \mathbb{Q}$ with $d^-(e) \leq d^+(e)$ for all $e \in \mathcal{E}_f$, a feasible leader's solution $X \subseteq \mathcal{E}_l$

Output : an optimal adversary's solution $d \in \mathcal{U}$ (i.e., $d\colon \mathcal{E}_f \to \mathbb{Q}$ with $d^-(e) \leq d(e) \leq d^+(e)$ for all $e \in \mathcal{E}_f$) of the Robust Bilevel Selection Problem

1 $\mathcal{E} := \mathcal{E}_f \setminus X$
2 **if** $\mathcal{E} = \emptyset$ **then**
3 $\quad Y_{\bar{e}} := \emptyset$
4 **else**
5 $\quad b_f := b - |X|$
6 $\quad \bar{\mathcal{E}} := \emptyset$
7 $\quad$ **for** $\bar{e} \in \mathcal{E}$ **do**
8 $\quad\quad \mathcal{E}_{\bar{e}}^- := \{e \in \mathcal{E} \mid d^+(e) < d^-(\bar{e})\}$
9 $\quad\quad \mathcal{E}_{\bar{e}}^0 := \{e \in \mathcal{E} \mid d^-(e) \leq d^-(\bar{e}) \leq d^+(e)\}$
10 $\quad\quad$ **if** $|\mathcal{E}_{\bar{e}}^-| \leq b_f$ *and* $|\mathcal{E}_{\bar{e}}^0| \geq b_f - |\mathcal{E}_{\bar{e}}^-|$ **then**
11 $\quad\quad\quad Y_{\bar{e}}^0 :=$ Selection Problem$(\mathcal{E}_{\bar{e}}^0, b_f - |\mathcal{E}_{\bar{e}}^-|, -c \restriction \mathcal{E}_{\bar{e}}^0)$
12 $\quad\quad\quad Y_{\bar{e}} := \mathcal{E}_{\bar{e}}^- \cup Y_{\bar{e}}^0$
13 $\quad\quad\quad \bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{\bar{e}\}$
14 $\quad$ select $\bar{e} \in \operatorname{argmax}\{c(Y_e) \mid e \in \bar{\mathcal{E}}\}$ arbitrarily
15 **return** $d\colon \mathcal{E}_f \to \mathbb{Q}$ *with* $d(e) := d^-(e)$ *for all* $e \in Y_{\bar{e}}$ *and* $d(e) := d^+(e)$ *for all* $e \in \mathcal{E}_f \setminus Y_{\bar{e}}$

Very similarly to Lemma 1 in [6], one can now prove:

▶ **Theorem 8.** *For any fixed feasible leader's solution of the* Robust Bilevel Selection Problem *with an interval uncertainty set, Algorithm 2 solves the adversary's problem in time* $O(n^2)$*.*

4.3 Discrete Uncorrelated Uncertainty

We now turn to the setting of discrete uncorrelated uncertainty, in which there is a finite number of possible follower's item costs for each follower's item, independently of each other, i.e., $\mathcal{U} = \prod_{e \in \mathcal{E}_f} \mathcal{U}_e$ with finite sets $\mathcal{U}_e \subseteq \mathbb{Q}$ for all $e \in \mathcal{E}_f$.

For every such uncertainty set, its convex hull is an interval uncertainty set and can be written as $\operatorname{conv}(\mathcal{U}) = \prod_{e \in \mathcal{E}_f} [d^-(e), d^+(e)]$, where $d^-(e)$ and $d^+(e)$ are the minimal and maximal value in $\mathcal{U}_e$, respectively, for all $e \in \mathcal{E}_f$.

Recall that, in our setting of robust bilevel optimization problems, one cannot replace an uncertainty set by its convex hull in general without changing the problem; see, e.g., [7]. However, for the Robust Bilevel Selection Problem, we will see in the following that a discrete uncorrelated uncertainty set can still be replaced by its convex hull. We emphasize that this is very specific to the Bilevel Selection Problem and that even closely related problems do not have this property, in particular the continuous variant of the Robust Bilevel Selection Problem and the Robust Bilevel Continuous Knapsack Problem; see Section 6.2.3 and [6].

First observe that the adversary's problem with $\text{conv}(\mathcal{U})$ is a relaxation of the one with $\mathcal{U}$. Moreover, the optimal adversary's solutions that are computed by Algorithm 2 in the case of interval uncertainty always attain an endpoint of each of the intervals. Hence, when replacing a discrete uncorrelated uncertainty set $\mathcal{U}$ by its convex hull as described above and solve the resulting problem with interval uncertainty using Algorithm 2, then only the endpoints of the intervals $[d^-(e), d^+(e)]$ are relevant for the optimal adversary's solutions. As these points are contained in the original sets $\mathcal{U}_e$, this implies that the computed adversary's solutions are also valid, and therefore also optimal, for the original problem with discrete uncorrelated uncertainty. Hence, the two types of uncertainty sets are equivalent for the Robust Bilevel Selection Problem, and we have shown:

▶ **Theorem 9.** *The Robust Bilevel Selection Problem with a discrete uncorrelated uncertainty set can be linearly reduced to the corresponding problem with an interval uncertainty set. The same holds for the adversary's problems, for any fixed feasible leader's solution.*

This implies that Theorem 8 also holds in the setting of discrete uncorrelated uncertainty.

▶ Remark 10. The set of greedy orders of the follower's items that the adversary can enforce cannot be described in terms of partial orders here, as it was the case for interval uncertainty; see Lemma 7. As an example, let $\mathcal{E}_f = \{e_1, e_2, e_3\}$ with $\mathcal{U}_{e_1} = \{1, 4\}$, $\mathcal{U}_{e_2} = \{2\}$, and $\mathcal{U}_{e_3} = \{3\}$. Then the only relation that is true for every possible follower's greedy order is that e_2 precedes e_3, but the linear extension $e_2 \prec e_1 \prec e_3$ of this partial order cannot be enforced. However, every prefix of $e_2 \prec e_1 \prec e_3$ is also the prefix of one of the two linear orders that the adversary can produce. In fact, the sets of optimal follower's solutions that the adversary can enforce, for any fixed feasible leader's solution, are always the same for $\mathcal{U}$ and $\text{conv}(\mathcal{U})$, which is an intuitive reason why the two types of uncertainty sets are equivalent.

5 The Robust Leader's Problem

After having seen polynomial-time algorithms for the adversary's problem in the previous section, we now turn to the leader's perspective in the Robust Bilevel Selection Problem. The case of disjoint item sets turns out to be significantly easier here than the general one. In Section 5.1, we will show how to derive polynomial-time algorithms for the leader's problem from the ones for the adversary's problem, for discrete uncertainty, interval uncertainty, and discrete uncorrelated uncertainty. In Section 5.2, we then prove that the Robust Bilevel Selection Problem is NP-hard in general, for discrete uncertainty, and show how to approximate it and how to solve it in exponential time.

5.1 The Special Case of Disjoint Item Sets

The setting of disjoint leader's and follower's item sets was already easier to understand than the general one in the Bilevel Selection Problem without uncertainty, although the same algorithm was able to solve both variants of the problem; see Section 2.3. In particular, it was clear there that the capacity b has to be split between leader and follower somehow and that each of the players solves a single-level Selection Problem on their own set of items then. Hence, the algorithm mainly needed to determine (by enumeration) how the leader can split the capacity optimally. In the robust setting, this basic idea remains valid. Therefore, for a fixed choice of b_l, the leader's decision can again be seen as a single-level Selection Problem that she can solve in a greedy way, and the best choice of b_l can again be determined by enumeration.

On the follower's side of the problem, now also the adversary comes into play. We do not only solve one single-level Selection Problem for the follower, but solve the adversary's problem of determining a follower's objective function that results in the worst possible follower's solution for the leader. Here, the polynomial-time algorithms that we have presented for the adversary's problem in Section 4 will be useful. The leader's enumeration approach then leads to a polynomial-time algorithm whenever we can solve the adversary's problem in polynomial time:

▶ **Theorem 11.** *Consider the* Robust Bilevel Selection Problem *with* $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$*, and with any type of uncertainty set. Suppose that the corresponding adversary's problem can be solved in time at most* $A(I)$*, given an instance* I *of the* Robust Bilevel Selection Problem[2] *together with any leader's solution. Then the* Robust Bilevel Selection Problem*, given an instance* I *with* n *items, can be solved in time* $O(nA(I))$*.*

Proof. The leader enumerates all feasible values of b_l that determine the capacity used by herself. These are clearly the integer values in the same range as in Algorithm 1, and their number is $O(n)$. For each of them, the leader solves a single-level Selection Problem on $\mathcal{E}_l$ with capacity b_l in order to determine her solution, and, in addition, solves the adversary's problem for this fixed leader's choice. The former can be done in time $O(n)$ and the latter in time at most $A(I)$. The best solution computed in this way clearly gives an optimal leader's solution. Since the running time $A(I)$ for solving the adversary's problem cannot be expected to be faster than $O(n)$, which is the time required to solve the follower's problem for a given scenario, the resulting running time can be written as $O(nA(I))$. ◀

▶ Remark 12. For computing all potential leader's solutions, the algorithm described in the proof of Theorem 11 requires a running time of $O(n)$ in each of the $O(n)$ iterations, i.e., a total running time of $O(n^2)$. This part of the algorithm's total running time can be reduced to $O(n \log n)$ by presorting the leader's items once in the beginning of the algorithm, as in Algorithm 1; see also Remark 3. However, this does not improve the stated total running time of $O(nA(I))$. Moreover, when applying Theorem 11 for specific types of uncertainty sets below, we do not only use it as a black box and improve on this running time anyway, by doing similar preprocessings that depend on the type of uncertainty set and enable to solve the adversary's problems faster.

We can now combine Theorem 11 with the algorithms for the adversary's problem from Section 4, for the different types of uncertainty sets:

▶ **Corollary 13.** *The* Robust Bilevel Selection Problem *with* $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$*, and with a discrete uncertainty set* $\mathcal{U}$*, can be solved in time* $O(|\mathcal{U}| n \log n)$*.*

Proof. The existence of a polynomial-time algorithm follows from Theorems 6 and 11. The desired running time of this algorithm can be achieved by implementing it as a variant of Algorithm 1 (see also Remark 12), as follows: Precompute the bijection p_l and bijections p_f^d for all scenarios $d \in \mathcal{U}$. This takes time $O(|\mathcal{U}| n \log n)$. The enumeration of values b_l and the computation of the corresponding leader's solutions X_{b_l} is done as in Algorithm 1. To determine the worst-case follower's solution Y_{b_l} in every iteration, we compute $Y_{b_l}^d$ from p_f^d for each scenario $d \in \mathcal{U}$ and choose one with the maximal costs $c(Y_{b_l}^d)$. Each of the $O(n)$ iterations of the loop can thus be implemented to run in time $O(|\mathcal{U}|)$. ◀

▶ **Corollary 14.** *The* Robust Bilevel Selection Problem *with* $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$*, and with interval uncertainty or discrete uncorrelated uncertainty, can be solved in time* $O(n^2 \log n)$*.*

Proof. The existence of a polynomial-time algorithm directly follows from Theorems 8 and 11. These results imply a running time of $O(n^3)$. However, as indicated in Remark 12, this can be improved to $O(n^2 \log n)$ by presorting not only the leader's, but also the follower's items. More precisely, we compute the sets $\mathcal{E}_{\bar{e}}^-$ and $\mathcal{E}_{\bar{e}}^0$, together with an order of the items in $\mathcal{E}_{\bar{e}}^0$ by their leader's item costs c, for all $\bar{e} \in \mathcal{E}_f$, in the beginning of the algorithm, before enumerating the splittings of the capacity b into leader's and follower's capacities b_l and b_f, respectively. This requires a running time of $O(n^2 \log n)$ and enables to implement every iteration of the loop in Algorithm 2 to run in constant time. Thus, the overall running time is $O(n^2 \log n)$.

Theorem 9 implies that the statement also holds in the setting of discrete uncorrelated uncertainty. ◀

5.2 The General Case

In the general case, the Robust Bilevel Selection Problem becomes much more involved than in the special case where $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$. In fact, we will prove in Section 5.2.1 that the leader's problem is strongly NP-hard in the general case. We will, however, suggest some algorithms to deal with the general Robust Bilevel Selection Problem: The polynomial-time algorithm presented in Section 5.2.2 leads to a leader's objective function value that is at most by a factor of 2 worse than the optimum. Besides approximation algorithms, another way to deal with NP-hard problems is to solve them in superpolynomial time. We will analyze two such algorithms for the Robust Bilevel Selection Problem in Section 5.2.3. One of them shows that the problem can be solved in polynomial time given a constant number of scenarios (Theorem 24).

---

[2] Note that parts of the structure of the instance I as well as the function A may depend on the type of uncertainty set.

5.2.1 NP-Hardness

When the leader's and the follower's item sets are not disjoint, it is not clear anymore that a greedy decision on the leader's items is optimal. This can make the leader's problem significantly harder than the adversary's problem, in contrast to Theorem 11 for the disjoint setting, which showed that the leader's problem can be polynomially reduced to the adversary's problem in this case. Theorem 15 will show that the general Robust Bilevel Selection Problem with discrete uncertainty is NP-hard, while the corresponding adversary's problem is solvable in polynomial time by Theorem 6. Note that the latter implies that the leader's problem is NP-easy, as it cannot be more than one level harder than the adversary's problem, i.e., the evaluation of the leader's objective function; see also [7]. The Robust Bilevel Selection Problem with discrete uncertainty is therefore NP-equivalent.

For the following proof, we will use a reduction from the well-known strongly NP-hard Vertex Cover Problem [20]. In this problem, we are given an undirected graph $G = (V, E)$, and the task is to find a vertex cover, i.e., a vertex set $X \subseteq V$ such that, for every edge $e \in E$, at least one endpoint of e is contained in X, of minimal cardinality.

▶ **Theorem 15.** *The Robust Bilevel Selection Problem with discrete uncertainty is strongly NP-hard.*

Proof. Consider an instance of the Vertex Cover Problem, consisting of an undirected graph $G = (V, E)$ with $n \in \mathbb{N}$ vertices $V = \{v_1, \dots, v_n\}$ and $m \in \mathbb{N}$ edges $E = \{e_1, \dots, e_m\}$. Without loss of generality, we may assume that the minimal cardinality of a vertex cover in this instance is at least 2 and at most $n - 1$. We build an instance of the Robust Bilevel Selection Problem as follows: $\mathcal{E}_l = V$, $\mathcal{E}_f = V \cup \{h_1, \dots, h_{n+3}\}$, $b = n + 1$, and

$$c(h) = \begin{cases} 1 & \text{for } h \in V \cup \{h_{n+1}, h_{n+2}\} \\ n & \text{for } h = h_{n+3} \\ 0 & \text{for } h \in \{h_1, \dots, h_n\}. \end{cases}$$

The uncertainty set $\mathcal{U} = \{d_1, \dots, d_{m+1}\}$ consists of a scenario d_i for every edge $e_i \in E$ of G and one additional scenario d_{m+1}. Instead of defining the scenarios, i.e., follower's objective functions, d_i explicitly, it suffices to give orders of the items that are used in the follower's greedy algorithm in the respective scenarios. It is always possible to define a function d_i for which this order is the unique optimal one for the follower and, in particular, to compute such a function d_i with polynomial-size values in polynomial time; see also Remark 1. Note that the follower never selects more than $b = n + 1$ items, so that we can neglect the items after the first $n + 1$ ones in the follower's greedy order. For any $i \in \{1, \dots, m\}$, the scenario d_i, corresponding to the edge $e_i = \{v_j, v_k\} \in E$ with $j < k$, is defined by the order $h_{n+1}, h_{n+2}, v_1, \dots, v_{j-1}, v_{j+1}, \dots, v_{k-1}, v_{k+1}, \dots, v_n, h_{n+3}$. The scenario d_{m+1} is defined by the order $h_{n+3}, h_1, \dots, h_n$.

We will prove that every optimal leader's solution $X \subseteq \mathcal{E}_l = V$ is a minimum vertex cover in G. First, let $X \subseteq \mathcal{E}_l$ not be a vertex cover, i.e., there is an edge $e_i = \{v_j, v_k\} \in E$ with $j < k$ such that $v_j \notin X$ and $v_k \notin X$. By definition of the corresponding scenario d_i, all items selected by the leader in X are among the first $n+1$ items in the follower's greedy order. Hence, when the follower completes the leader's solution X greedily to $b = n+1$ items in total, this leads to a follower's solution Y with $X \cup Y$ consisting of exactly the first $n+1$ items in the follower's greedy order, i.e., $X \cup Y = \{h_{n+1}, h_{n+2}, v_1, \dots, v_{j-1}, v_{j+1}, \dots, v_{k-1}, v_{k+1}, \dots, v_n, h_{n+3}\}$, which has a leader's objective function value of $c(X \cup Y) = 2n$. Thus, in the robust setting, every leader's solution X that is not a vertex cover leads to an objective function value of $2n$, which is the worst possible value from the leader's perspective.

Now we turn to leader's solutions $X \subseteq \mathcal{E}_l$ that are vertex covers. By definition, for every edge $e_i \in E$, at least one of its endpoints is in X. Accordingly, for every scenario d_i, at least one item that is not among the first $n+1$ items of the follower's greedy order in this scenario is selected by the leader. When the follower now greedily completes the leader's solution X to $n+1$ items in total, he will never select the item h_{n+3}, which is the $(n+1)$-th item in his greedy order. In fact, we get $X \cup Y \subseteq V \cup \{h_{n+1}, h_{n+2}\}$. Thus, in each of the scenarios $d_1, \dots, d_m$, the leader's objective value is $n + 1$. In scenario d_{m+1}, the follower always selects item h_{n+3} and possibly some items with leader's cost 0. For the leader's solution X and the corresponding follower's solution Y, the leader's objective value is thus $c(X \cup Y) = |X| + n$. This implies that the adversary can be assumed to always select scenario d_{m+1} when the leader has selected a vertex cover X, and that, as a leader's solution, any vertex cover is better than any set that is not a vertex cover. Finally, optimizing the leader's objective value among the solutions that are vertex covers is equivalent to finding a vertex cover $X \subseteq V$ of minimal size $|X|$.

Note that we have indeed shown strong NP-hardness, since the Vertex Cover Problem does not have any numerical parameters and all values of the constructed functions c and $d_i \in \mathcal{U}$ have polynomial size. ◀

For other types of uncertainty sets, it remains an open question whether the Robust Bilevel Selection Problem is NP-hard. We conjecture that this is the case, in particular for interval uncertainty. In view of the algorithmic results in Sections 4 and 5.1, interval uncertainty does not seem to be easier than discrete uncertainty for the Robust Bilevel Selection Problem. Also the more general complexity results in [7] indicate that interval uncertainty is in some sense harder than discrete uncertainty in our robust bilevel setting.

Because of Theorem 15, we cannot hope for a polynomial-time exact algorithm (unless P = NP) for the general Robust Bilevel Selection Problem with discrete uncertainty. Therefore, we develop approximation algorithms and exponential-time exact algorithms for the Robust Bilevel Selection Problem in Sections 5.2.2 and 5.2.3, respectively.

5.2.2 Approximation Algorithms

In the setting of disjoint item sets, we have seen that the leader's problem can be solved in polynomial time for all considered types of uncertainty sets, by reducing it to the adversary's problem; see Theorem 11. In this algorithm, the leader enumerates how many items she selects herself and selects them greedily for every fixed number. We will now see that this algorithm does not yield an optimal solution anymore in the general setting. This is in contrast to the problem without uncertainty, where the same idea could be used for the disjoint and for the general case; see Section 2.3. However, the difference is not surprising here in view of the result of Theorem 15 that the general Robust Bilevel Selection Problem with discrete uncertainty is strongly NP-hard. But the algorithm that is exact in the disjoint case now turns out to be a 2-approximation algorithm, as we will show in Theorem 16. In fact, we again prove this for any type of uncertainty set for which the adversary's problem can be solved in polynomial time, like in Theorem 11. The results in this section are originally due to Hartmann [23], while the proofs have been revised and simplified by the author of this article.

We need to emphasize that, in order to make reasonable statements about approximating objective function values, we have to assume that they are nonnegative. More precisely, we assume, for the following statements, that the function $c\colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}_{\geq 0}$ of leader's item costs only attains nonnegative values. Recall from Section 2.2.5 that all leader's item costs can be shifted arbitrarily by a constant, without changing the optimal solutions. However, this cannot be done without loss of generality here, as the resulting constant offset in the objective values has an impact on approximation factors.

▶ **Theorem 16.** *Consider the Robust Bilevel Selection Problem, with any type of uncertainty set. Suppose that the corresponding adversary's problem can be solved in time at most $A(I)$, given an instance I of the Robust Bilevel Selection Problem together with any leader's solution. Then the problem of 2-approximating the Robust Bilevel Selection Problem, given an instance I with n items, can be solved in time $O(nA(I))$.*

Proof. We assume for this proof that $\mathcal{E}_l \subseteq \mathcal{E}_f$, which is not a restriction compared to the general Bilevel Selection Problem; see Section 2.2.1.

The leader enumerates all values $b_l \in \{0, \dots, \min\{b, n_l\}\}$ and solves a single-level Selection Problem on $\mathcal{E}_l$ with capacity b_l in each iteration. Moreover, she determines an optimal solution of the adversary's problem corresponding to this leader's solution. The best solution achieved throughout these iterations is returned. This is the same algorithm as in Theorem 11, and also its running time is the same.

We will now prove that the algorithm always returns a solution with leader's costs that are at most twice the costs of an optimal solution. For this, denote the leader's solution in iteration b_l of the algorithm by X_{b_l}, the corresponding follower's response in any scenario $d \in \mathcal{U}$ by $Y_{b_l}^d$, and the follower's response in the worst-case scenario by Y_{b_l}, i.e., choose $Y_{b_l} \in \{Y_{b_l}^d \mid d \in \mathcal{U}\}$ with $c(Y_{b_l}) = \max\{c(Y_{b_l}^d) \mid d \in \mathcal{U}\}$. Moreover, denote an optimal leader's solution by X^*, the corresponding follower's response in any scenario $d \in \mathcal{U}$ by Y^d, and the follower's response in the worst-case scenario by Y^*.

For the rest of the proof, choose $b_l \in \{|X^*|, \dots, \min\{b, n_l\}\}$ to be minimal such that $|X_{b_l} \cap Y^d| \leq b_l - |X^*|$ for all $d \in \mathcal{U}$. This is always possible because $|X^*| \leq \min\{b, n_l\}$ and the inequality is always satisfied for $b_l = \min\{b, n_l\}$. Indeed, since X^* and Y^d are disjoint for all $d \in \mathcal{U}$ as corresponding leader's and follower's solutions, we then have either $|X_b \cap Y^d| \leq |Y^d| = b - |X^*|$ or $|X_{n_l} \cap Y^d| = |\mathcal{E}_l \cap Y^d| \leq |\mathcal{E}_l \setminus X^*| = n_l - |X^*|$ for all $d \in \mathcal{U}$.

If $b_l = 0$, we must have $X^* = X_{b_l} = \emptyset$, which means that the algorithm finds an optimal solution. Therefore, we assume $b_l \geq 1$ in the following.

Due to the minimality of b_l, we have $b_l = |X^*|$ or there is some $d \in \mathcal{U}$ with $|X_{b_l-1} \cap Y^d| > b_l - 1 - |X^*|$, which is equivalent to

$$|X_{b_l-1} \cap Y^d| \geq b_l - |X^*|. \qquad (*)$$

In fact, $b_l = |X^*|$ also implies $(*)$, even for all $d \in \mathcal{U}$. Hence, there is always some $d \in \mathcal{U}$ for which $(*)$ is satisfied. Fix such a scenario d in the following.

In order to prove the required approximation guarantee, we will show that the leader's costs of both the leader's and the follower's solution in iteration b_l can be bounded from above by the costs of an optimal solution, i.e., $c(X_{b_l}) \leq c(X^*) + c(Y^*)$ and $c(Y_{b_l}) \leq c(X^*) + c(Y^*)$. Together, this proves that the total costs of the solution that the algorithm returns is at most $c(X_{b_l}) + c(Y_{b_l}) \leq 2(c(X^*) + c(Y^*))$.

We first focus on bounding $c(X_{b_l})$. For this, we partition the set X_{b_l} into the two disjoint sets $B := X_{b_l-1} \cap Y^d$ and $A := X_{b_l} \setminus B$. Note that $B \subseteq X_{b_l}$ because X_{b_l} consists of X_{b_l-1} and one additional item by the construction of the leader's solutions in the algorithm. Clearly, $c(B) \leq c(Y^d) \leq c(Y^*)$ holds, so it remains to show $c(A) \leq c(X^*)$.

Since X_{b_l} is selected greedily from $\mathcal{E}_l$ according to the costs c, we have $c(e_1) \leq c(e_2)$ for all $e_1 \in X_{b_l}$ and all $e_2 \in \mathcal{E}_l \setminus X_{b_l}$, and in particular for all $e_1 \in A \setminus X^* \subseteq X_{b_l}$ and all $e_2 \in X^* \setminus A \subseteq \mathcal{E}_l \setminus X_{b_l}$. The last inclusion holds because $X^* \cap Y^d = \emptyset$ for the follower's response Y^d to the leader's solution X^*, which implies $X^* \cap B = \emptyset$ and hence $X^* \cap X_{b_l} \subseteq A$.

Moreover, we can use $(*)$ to derive $|A| = |X_{b_l}| - |B| = b_l - |X_{b_l-1} \cap Y^d| \leq |X^*|$ and hence $|A \setminus X^*| \leq |X^* \setminus A|$. Together with the previous paragraph, this yields $c(A) = c(A \setminus X^*) + c(A \cap X^*) \leq c(X^* \setminus A) + c(A \cap X^*) = c(X^*)$. This concludes the first part of the proof with $c(X_{b_l}) = c(A) + c(B) \leq c(X^*) + c(Y^*)$.

For the second part, we partition also the set Y_{b_l} into two disjoint sets $A' := Y_{b_l} \cap X^*$ and $B' := Y_{b_l} \setminus X^*$. We directly observe $c(A') \leq c(X^*)$, so it remains to prove $c(B') \leq c(Y^*)$.

Let $d' \in \mathcal{U}$ be the worst-case scenario in the considered solution of the algorithm, i.e., such that $Y_{b_l}^{d'} = Y_{b_l}$. We derive $|B'| = |Y_{b_l} \setminus X^*| \leq |Y_{b_l}| = b - b_l$ and, by the definition of b_l, $|Y^{d'} \setminus X_{b_l}| = |Y^{d'}| - |Y^{d'} \cap X_{b_l}| \geq |Y^{d'}| - (b_l - |X^*|) = b - b_l$. Thus, $|B'| \leq |Y^{d'} \setminus X_{b_l}|$.

We know that $Y_{b_l}^{d'}$ is selected greedily from $\mathcal{E} \setminus X_{b_l}$ by the follower, according to the costs d' (and possibly the leader's costs c as a secondary criterion in the optimistic or pessimistic setting). This means that $Y_{b_l}^{d'}$ consists of the first $|Y_{b_l}^{d'}| = b - b_l$ items of the corresponding order of the set $\mathcal{E} \setminus X_{b_l}$. By removing X^* from the sorted set, we obtain that $B' = Y_{b_l}^{d'} \setminus X^*$ consists of the first $|B'|$ items of the same order of the set $\mathcal{E} \setminus (X_{b_l} \cup X^*)$.

Analogously, $Y^{d'}$ is selected greedily from $\mathcal{E} \setminus X^*$ by the follower, i.e., again according to d' and possibly c as a secondary criterion. We may assume that the same order as above is used. We now remove X_{b_l} and derive that $Y^{d'} \setminus X_{b_l}$ consists of the first $|Y^{d'} \setminus X_{b_l}|$ items of the same order of the set $\mathcal{E} \setminus (X_{b_l} \cup X^*)$.

Together with $|B'| \leq |Y^{d'} \setminus X_{b_l}|$, this implies that $B' \subseteq Y^{d'} \setminus X_{b_l}$ and thus $c(B') \leq c(Y^{d'}) \leq c(Y^*)$, which concludes the second part of the proof. ◀

Combining Theorem 16 with the algorithms for the adversary's problem obtained in Section 4 leads to the following results for our specific types of uncertainty sets.

▶ **Corollary 17.** *The* ROBUST BILEVEL SELECTION PROBLEM *with discrete uncertainty can be 2-approximated in time $O(|\mathcal{U}|n \log n)$.*

Proof. A polynomial-time 2-approximation algorithm directly follows from Theorems 6 and 16. Analogously to Corollary 13, its running time can be improved from $O(|\mathcal{U}|n^2)$ to $O(|\mathcal{U}|n \log n)$ by precomputing the bijections for all scenarios. Note that, in every iteration, we might have to remove the item that the leader adds to her solution from the items the follower chooses from, for each scenario, like in the general version of Algorithm 1; see also the proof of Theorem 4. ◀

▶ **Corollary 18.** *The* ROBUST BILEVEL SELECTION PROBLEM *with interval uncertainty or discrete uncorrelated uncertainty can be 2-approximated in time $O(n^2 \log n)$.*

Proof. The existence of a polynomial-time 2-approximation algorithm for the case of interval uncertainty directly follows from Theorems 8 and 16. Analogously to Corollary 14, it can be implemented in a faster running time of $O(n^2 \log n)$ by making use of a preprocessing. Similarly to the proof of Corollary 17, the precomputed sets the follower's solutions are derived from might be affected by the different leader's solutions here and therefore need to be updated in each iteration.

Theorem 9 implies that the statement also holds in the setting of discrete uncorrelated uncertainty. ◀

The following examples show that the approximation factor of 2 is actually tight for the given algorithm in case of discrete uncertainty and interval uncertainty.

▶ **Example 19.** Consider the following family of instances of the Robust Bilevel Selection Problem with discrete uncertainty: Let $n \in \mathbb{N}$ with $n \geq 4$ and $\varepsilon \in \mathbb{Q}_{>0}$, and set $b = n - 2$. Define the item sets as $\mathcal{E}_l = \{e_1, e_2\}$ and $\mathcal{E}_f = \{e_1, e_2, e_3, \ldots, e_n\}$, with leader's item costs $c(e_1) = 1 - \varepsilon$, $c(e_2) = 1$, $c(e_3) = 3$, and $c(e) = 0$ for all other items $e \in \{e_4, \ldots, e_n\}$, and follower's item costs according to the two scenarios $\mathcal{U} = \{d_1, d_2\}$ such that the follower's greedy order in scenario d_1 is $e_4, \ldots, e_n, e_3, e_1, e_2$, and in scenario d_2 it is $e_2, e_4, \ldots, e_n, e_1, e_3$. Recall that it suffices to define the scenarios via the follower's greedy orders because the actual costs in the follower's objective are not important for the leader; see Remark 1.

The algorithm from Corollary 17 considers the three leader's selections $X_0 = \emptyset$, $X_1 = \{e_1\}$, and $X_2 = \{e_1, e_2\}$. They lead to worst-case follower's responses $Y_0 = \{e_4, \ldots, e_n, e_3\}$ (in scenario d_1), $Y_1 = \{e_2, e_4, \ldots, e_{n-1}\}$ (in scenario d_2), and $Y_2 = \{e_4, \ldots, e_{n-1}\}$ (in both scenarios), respectively. The resulting leader's objective function values are 3, $2 - \varepsilon$, and $2 - \varepsilon$. Hence, the leader will choose either X_1 or X_2 and achieve a cost value of $2 - \varepsilon$. The optimal leader's solution $X^* = \{e_2\}$, however, leads to the worst-case follower's response $Y^* = \{e_4, \ldots, e_{n-1}\}$ (in both scenarios) and therefore to a leader's objective value of only 1. This shows that, for any number of items, the solution returned by the algorithm might achieve an objective value of almost twice the optimal value. Depending on the choice the algorithm makes in case the greedy order of the leader's items is not unique, the same behavior might even occur for $\varepsilon = 0$, which would deliver a factor of exactly 2.

▶ **Example 20.** Consider the family of instances from Example 19 again, but instead of the discrete uncertainty set $\mathcal{U} = \{d_1, d_2\}$, we now work with an interval uncertainty set $\mathcal{U}$ according to which the following three follower's greedy orders are possible: $e_4, \ldots, e_{n-1}, e_2, e_n, e_3, e_1$, or $e_4, \ldots, e_{n-1}, e_n, e_2, e_3, e_1$, or $e_4, \ldots, e_{n-1}, e_n, e_3, e_2, e_1$. This can be achieved by fixing the follower's item costs of all items except for e_2 (i.e., using intervals of length 0) and defining an appropriate interval for the cost of e_2. Analogously to Example 19, it can be easily checked that the algorithm returns either $X_1 = \{e_1\}$ or $X_2 = \{e_1, e_2\}$ as a leader's solution, each achieving a leader's objective function value of $2 - \varepsilon$, while the optimal leader's solution $X^* = \{e_2\}$ leads to a cost value of only 1.

5.2.3 Exact Algorithms

We now turn towards exact algorithms for the general Robust Bilevel Selection Problem that have exponential running time and can be interesting from the perspective of parameterized complexity (see, e.g., [13]).

Clearly, an exact algorithm can be achieved by an enumeration approach:

▶ **Theorem 21.** *Consider the Robust Bilevel Selection Problem, with any type of uncertainty set. Suppose that the corresponding adversary's problem can be solved in time at most $A(I)$, given an instance I of the Robust Bilevel Selection Problem together with any leader's solution. Then the Robust Bilevel Selection Problem, given an instance I with n_l leader's items and capacity b, can be solved in time $O(\min\{2^{n_l}, n_l^b\}A(I))$.*

Proof. The Robust Bilevel Selection Problem can be solved by enumerating all feasible leader's solutions $X \subseteq \mathcal{E}_l$ and solving the adversary's problem for each of them. The latter is done by an algorithm specific for the given type of uncertainty set. This algorithm for the adversary's problem can be assumed to return a corresponding follower's solution $Y \subseteq \mathcal{E}_f$ as well. (If not, we can solve the follower's problem in time $O(n)$ for the given leader's solution and adversary's choice.) By comparing the leader's costs of all resulting solutions $X \cup Y$, an optimal leader's solution can be determined.

For the running time, note that the number of feasible leader's solutions $X \subseteq \mathcal{E}_l$ is bounded by 2^{n_l} and, if $b \leq n_l$, by $\sum_{b_l=0}^{b} \binom{n_l}{b_l} \in O(n_l^b)$ because the leader can select at most b items in a feasible solution. ◀

This implies that the Robust Bilevel Selection Problem is fixed-parameter tractable (FPT) in the parameter n_l as well as slice-wise polynomial (XP) with respect to the parameter b, whenever the adversary's problem can be solved fast enough (e.g., in polynomial time). For the uncertainty sets that we have investigated in Section 4 (see Theorems 6, 8, and 9), we obtain:

▶ **Corollary 22.** *The Robust Bilevel Selection Problem with a discrete uncertainty set $\mathcal{U}$, can be solved in time $O(\min\{2^{n_l}, n_l^b\}|\mathcal{U}|n)$.*

▶ **Corollary 23.** *The Robust Bilevel Selection Problem with interval uncertainty or discrete uncorrelated uncertainty can be solved in time $O(\min\{2^{n_l}, n_l^b\}n^2)$.*

For discrete uncertainty, we also derive a more complex algorithm which proves that the problem is slice-wise polynomial (XP) in the parameter $|\mathcal{U}|$, i.e., for every constant number of scenarios, the problem is solvable in polynomial time; see Theorem 24. We develop the idea of the algorithm in the following. A detailed description can also be found in [23]. We assume here that $\mathcal{E}_l \subseteq \mathcal{E}_f$, which is not a restriction; see Section 2.2.1.

To illustrate the idea, we first make some observations about the Bilevel Selection Problem without uncertainty, and with the assumption $\mathcal{E}_l \subseteq \mathcal{E}_f$. Recall that, from the follower's perspective, a greedy order of his items $\mathcal{E}_f$ can be fixed, independently of the leader's choice, and we may assume that the follower always selects a prefix of this order, possibly after removing the items that have already been selected by the leader. We utilized this also in Algorithm 1. Hence, every overall solution $X \cup Y$ can be assumed to consist of a prefix of the follower's greedy order (some part of which might have been selected by the leader and the rest by the follower) and possibly some single leader's items that appear later in the follower's greedy order. Therefore, we could also find a feasible overall solution $X \cup Y$ that is best possible for the leader as follows: Enumerate all prefixes $\overline{Y} \subseteq \mathcal{E}_f$ of the follower's greedy order and, for each of them, check whether the leader can select some solution $X \subseteq \mathcal{E}_l$ such that, together with the follower's response Y to X, the overall solution $X \cup Y$ consists of the prefix $\overline{Y}$ and $b - |\overline{Y}|$ additional leader's items. If this is possible, find the best such leader's choice X according to the leader's objective c. The best solution obtained in this enumeration process must be an optimal leader's solution.

The subproblem for a fixed prefix $\overline{Y} \subseteq \mathcal{E}_f$ is easy to solve: Clearly, the leader must select $b - |\overline{Y}|$ items from $\mathcal{E}_l \setminus \overline{Y}$. If this set does not contain sufficiently many items, then there is no feasible solution of the desired structure corresponding to the prefix $\overline{Y}$. In addition, the leader may select any number of items from $\mathcal{E}_l \cap \overline{Y}$, but this has no influence on the overall solution $X \cup Y$ because the follower's optimal response is always $Y = \overline{Y} \setminus X$. The selection of $b - |\overline{Y}|$ items from $\mathcal{E}_l \setminus \overline{Y}$ can be made in a greedy way according to the leader's objective c because there is no difference in the follower's reaction among different such selections. Therefore, each iteration in the above algorithm boils down to a single-level Selection Problem.

In fact, this algorithm is very similar to Algorithm 1 because it also enumerates greedy choices of different sizes, for the leader and for the follower. Due to the different perspective of starting with the follower's instead of the leader's prefixes, only the items that could be selected by either player are handled differently: Observe that the leader's greedy solutions that are enumerated in Algorithm 1 correspond to the solutions of the single-level Selection Problems here, together with a specific additional choice from $\mathcal{E}_l \cap \overline{Y}$, which does not influence the overall solution.

We now turn to the robust setting again. The different scenarios now correspond to different follower's greedy orders, and the adversary may choose between them. However, in any given scenario, the above structural observations still apply, and a generalization of the enumeration procedure described above now solves the Robust Bilevel Selection Problem. Instead of one prefix $\overline{Y} \subseteq \mathcal{E}_f$ of a single follower's greedy order, we now consider in each iteration a combination of prefixes of the greedy orders in all scenarios, i.e., some collection $(\overline{Y_d})_{d\in\mathcal{U}}$ such that $\overline{Y_d} \subseteq \mathcal{E}_f$ is a prefix of the follower's greedy order in scenario d, for all $d \in \mathcal{U}$. The problem that has to be solved in a given iteration is then to find the best leader's selection $X \subseteq \mathcal{E}_l$ (or decide that there is no such X) such that, for every scenario $d \in \mathcal{U}$, the overall solution $X \cup Y_d$ in this scenario (i.e., Y_d is the follower's optimal solution in scenario d) consists of the prefix $\overline{Y_d}$ of the follower's greedy order corresponding to this scenario and $b - |\overline{Y_d}|$ additional leader's items.

This subproblem can now be solved by another enumeration and several single-level Selection Problems on appropriate subsets of $\mathcal{E}_l$. We illustrate this for the setting of two scenarios $\mathcal{U} = \{d_1, d_2\}$: Given prefixes $\overline{Y_{d_1}}, \overline{Y_{d_2}} \subseteq \mathcal{E}_f$, in analogy to the situation without uncertainty above, the leader has to select $b - |\overline{Y_{d_1}}|$ items from $\mathcal{E}_l \setminus \overline{Y_{d_1}}$ and $b - |\overline{Y_{d_2}}|$ items from $\mathcal{E}_l \setminus \overline{Y_{d_2}}$. Since these two selections are not independent in general, we reformulate them as follows. If the leader selects b_1 items from $\mathcal{E}_l \setminus (\overline{Y_{d_1}} \cup \overline{Y_{d_2}})$, b_2 items from $(\mathcal{E}_l \cap \overline{Y_{d_1}}) \setminus \overline{Y_{d_2}}$, b_3 items from $(\mathcal{E}_l \cap \overline{Y_{d_2}}) \setminus \overline{Y_{d_1}}$, and b_4 items from $\mathcal{E}_l \cap \overline{Y_{d_1}} \cap \overline{Y_{d_1}}$, then this leads to the desired situation if and only if $b_1 + b_3 = b - |\overline{Y_{d_1}}|$ and $b_1 + b_2 = b - |\overline{Y_{d_2}}|$. Note that it does not matter how many items the leader selects that are contained in all considered prefixes, i.e., what the number b_4 is. The subproblem can now be solved by enumerating all combinations of values for b_1, b_2, and b_3 that satisfy these conditions and solving the three corresponding single-level Selection Problems.

It remains to analyze the running time of the algorithm. In the beginning, the follower's greedy orders in all scenarios are computed in time $O(|\mathcal{U}|n \log n)$. In the outer enumeration, we consider $O(b)$ possible prefixes for each scenario, which gives $O(b^{|\mathcal{U}|})$ combinations $(\overline{Y_d})_{d\in\mathcal{U}}$ of prefixes in total. In the subproblem, the enumeration comprises $O(2^{|\mathcal{U}|})$ values b_i, one for each subset $\overline{\mathcal{U}} \subseteq \mathcal{U}$ of the scenarios, describing the number of items the leader selects from the set $(\mathcal{E}_l \cap \bigcap_{d\in\mathcal{U}\setminus\overline{\mathcal{U}}} \overline{Y_d}) \setminus \bigcup_{d\in\overline{\mathcal{U}}} \overline{Y_d}$. Each value b_i can attain $O(b)$ many different values, resulting in $O(b^{2^{|\mathcal{U}|}})$ cases that have to be checked. For fixed values b_i, we check $O(|\mathcal{U}|)$ conditions on sums of these values and, if they are satisfied, we solve $O(2^{|\mathcal{U}|})$ many single-level Selection Problems, each of them in time $O(n)$. In order to evaluate the leader's objective function value of such a solution, we can compute its leader's costs in all scenarios in $O(|\mathcal{U}|n)$.

In summary, the described algorithm gives the following result:

▶ **Theorem 24.** *The Robust Bilevel Selection Problem with a discrete uncertainty set $\mathcal{U}$ of constant size $|\mathcal{U}|$ can be solved in time polynomial in n.*

6 Problem Variants with Continuous Variables

As already mentioned in Section 2.2.2, we now study variants of the Bilevel Selection Problem with continuous decisions. As a combinatorial optimization problem, the Bilevel Selection Problem can be formulated in terms of binary variables in a straightforward way, and it could be interesting to consider also the corresponding variant with continuous variables, i.e., the setting in which the leader and/or the follower are allowed to select fractions of items. Corresponding to the formulation in (BSP), we can write the (basic) continuous variant of the Bilevel Selection Problem as follows:

$$
\begin{aligned}
\min_{x} \quad & \sum_{e \in \mathcal{E}_l} c(e) x_e + \sum_{e \in \mathcal{E}_f} c(e) y_e \\
\text{s.t.} \quad & x \in [0,1]^{\mathcal{E}_l} \\
& y \in \operatorname*{argmin}_{y'} \sum_{e \in \mathcal{E}_f} d(e) y'_e \\
& \qquad \text{s.t.} \quad x_e + y'_e \leq 1 \qquad \forall\, e \in \mathcal{E}_l \cap \mathcal{E}_f \\
& \qquad \qquad \sum_{e \in \mathcal{E}_l} x_e + \sum_{e \in \mathcal{E}_f} y'_e = b \\
& \qquad \qquad y' \in [0,1]^{\mathcal{E}_f}
\end{aligned}
\tag{CBSP}
$$

Observe that the version of (CBSP) with binary instead of continuous variables is equivalent to (BSP). Note that the constraint that the follower's solution must be disjoint from the leader's one ($Y' \subseteq \mathcal{E}_f \setminus X$ in (BSP)) is modeled by the inequalities $x_e + y'_e \leq 1$ for all $e \in \mathcal{E}_l \cap \mathcal{E}_f$ here.

As explained in Section 2.2.1, the bilevel problem where the leader sets the capacity for the follower's Selection Problem can be modeled as a special case of the Bilevel Selection Problem, and this is also true in the continuous variant. Here, the resulting problem can be seen as a special case of the Bilevel Continuous Knapsack Problem that is considered in [6]. Accordingly, most of the insights presented in this section are generalizations and variations of results in [6]. However, we also emphasize an important difference in complexity between the Robust Continuous Bilevel Selection Problem and the Robust Bilevel Continuous Knapsack Problem for the setting of discrete uncorrelated uncertainty; see Section 6.2.3.

Observe that the follower's problem in (CBSP), for a fixed feasible leader's solution $x \in [0,1]^{\mathcal{E}_l}$, corresponds to a Continuous Selection Problem (in case of disjoint item sets $\mathcal{E}_l$ and $\mathcal{E}_f$) or possibly to a Continuous Knapsack Problem (in the general case) because the leader might have selected fractions of some follower's items, hence leaving items of different reduced sizes for the follower to choose from. However, in both cases, the follower can still be assumed to solve his problem greedily and select items according to a fixed order, i.e., he selects the largest prefix of this order that fits entirely into the given capacity and possibly a fraction of the next item such that the capacity is filled completely. Hence, such a greedy solution can be assumed to select at most one item fractionally. In case of a Continuous Knapsack Problem, the greedy order is determined by the ratios of item costs and item sizes [14]. For more details on the situation of a Continuous Knapsack Problem in the follower's problem, we also refer to [6].

In addition to the original Bilevel Selection Problem and the continuous version (CBSP), two further problem variants can be obtained by relaxing the integrality only in the leader's or only in the follower's decision. In the following, we will compare all four versions of the Bilevel Selection Problem.

First note that the setting in which the leader takes a binary decision and the follower is allowed to select fractions of items is in fact equivalent to the original Bilevel Selection Problem: From the follower's perspective, an integer number of items is left to select and none of his items has been selected fractionally by the leader. Following the greedy approach, the follower will therefore select a binary solution.

We now turn to the problem version in which the leader can select fractions of items, but the follower has binary variables. First focus on the special case of disjoint item sets again. As in the original Bilevel Selection Problem (see Section 2.3), the only relevant interaction between leader and follower then is the number $b_l = \sum_{e \in \mathcal{E}_l} x_e$ of items the leader selects. Although b_l is not automatically an integer here, the leader has to choose her solution such that b_l is an integer in order to enable the follower to complete the selection to exactly b items in total. As before, it is optimal for the leader and for the follower to select items greedily, given a fixed partition of the total capacity, in case of disjoint item sets. Analogously to above, this means that the leader can be assumed to take a binary decision. Thus, if the leader's and the follower's item sets are disjoint, allowing only the leader to select fractions of items does not change the problem compared to the original variant.

If the item sets are not disjoint, the situation is different. In fact, we might obtain a bilevel optimization problem whose optimal value is not attained; see Example 25. This is an interesting property that (mixed-integer) bilevel optimization problems can have, even if the corresponding single-level problems do not indicate such a behavior. The reason here is that the binary follower's problem might introduce an unnatural discontinuity into the leader's objective. We refer to, e.g., [34] for more examples of such a behavior in bilevel optimization problems with continuous leader's variables and integer follower's variables.

▶ **Example 25.** Consider the problem (CBSP) with continuous leader's, but binary follower's variables. Let $\mathcal{E}_l = \{e_1, e_2\}$, $\mathcal{E}_f = \{e_1, e_2, e_3\}$, $b = 2$, $c(e_1) = 1$, $c(e_2) = c(e_3) = 0$, $d(e_1) = 0$, $d(e_2) = d(e_3) = 1$. For the leader, the overall solution consisting of e_2 and e_3 would be best possible, with total costs of 0. However, a solution without e_1 is not possible because the follower prefers e_1 over e_2 and e_3 and the leader cannot select e_3 herself. But, as the leader is allowed to make continuous decisions and the follower is not, it suffices that the leader selects a small fraction of e_1 in order to "block" it for the follower. However, the leader has to select an integer number of items in total in order to ensure feasibility. This can be achieved by selecting a corresponding fraction of e_2. More precisely, for every $\varepsilon \in (0, 1)$, the leader's solution $x = (\varepsilon, 1 - \varepsilon, 0)$ induces the follower's response $y = (0, 0, 1)$ and hence a leader's objective value of ε. Thus, it can get arbitrarily close to 0, but cannot attain 0. This implies that the problem does not have an optimal leader's solution.

Because of the above observations, the setting in which both decision makers take a continuous decision is the most interesting to consider further. Therefore, we focus on the setting of (CBSP) again for the remainder of this section.

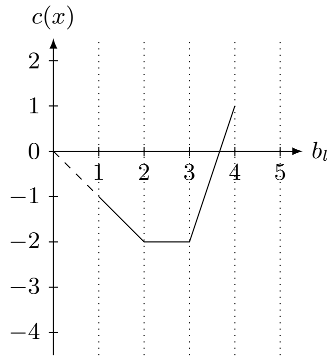
In fact, for the setting without uncertainty, we argue that the continuous problem (CBSP) is equivalent to the original Bilevel Selection Problem, while this will not be the case anymore when we add robustness; see Example 26. To show the equivalence of the two problem versions without uncertainty, first consider the special case of disjoint item sets again. As before, the leader only influences the follower through the (not necessarily integer) number $b_l = \sum_{e \in \mathcal{E}_l} x_e$ of items she selects and, for fixed b_l, it is optimal for each player to solve the corresponding single-level Continuous Selection Problem on their own item set. Hence, every possible such selection of leader's and follower's items can be seen as a convex combination of two integer solutions. This will also be illustrated in Section 6.1. Thus, there is always an optimal solution that is integer. If the item sets are not disjoint, a similar argument can be applied because, in an optimal solution, we may assume that the sets of items the two players select (possibly fractionally) are disjoint.

For the remainder of this section, we focus on the presumably easier special case of $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$ again. From now on, we will also denote the problem (CBSP) with this assumption as the Continuous Bilevel Selection Problem.

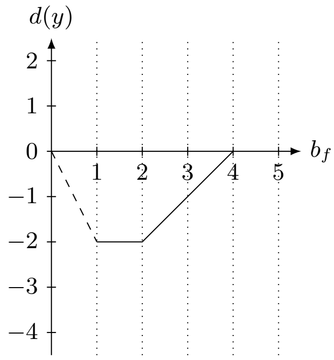
We will give a better understanding of the structure of the Continuous Bilevel Selection Problem in Section 6.1, using piecewise linear functions. Based on this, we investigate the robust problem in Section 6.2. Here, we show that the problem is not equivalent to the binary problem version in the sense we described it above for the setting without uncertainty. However, we also show that the robust problem with discrete uncertainty, interval uncertainty, and discrete uncorrelated uncertainty can be solved in polynomial time.

6.1 Structure of the Leader's Objective Function

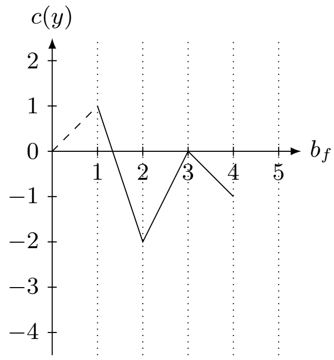
We have already argued above that the Continuous Bilevel Selection Problem (without uncertainty) is equivalent to the Bilevel Selection Problem. In the case of disjoint item sets, this can also be understood by investigating the structure of the leader's objective function in more detail. Moreover, the following insights will be useful in view of the robust problem versions we will study in Section 6.2.



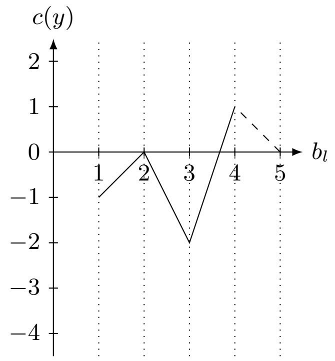
(a) Leader's objective of her own items, $\mathrm{PLF}(\mathcal{E}_l, c \restriction \mathcal{E}_l, b_l^-, b_l^+)$



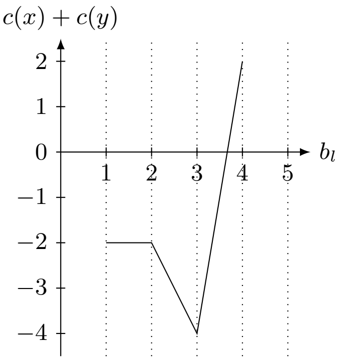
(b) Follower's objective of his own items, $\mathrm{PLF}(\mathcal{E}_f, (d, -c \restriction \mathcal{E}_f), b_f^-, b_f^+)$



(c) Leader's objective of the follower's items, depending on b_f



(d) Leader's objective of the follower's items, depending on b_l, $\overline{\mathrm{PLF}}(\mathcal{E}_f, (d, -c \restriction \mathcal{E}_f), c \restriction \mathcal{E}_f, b_l^-, b_l^+, b)$



(e) Leader's total objective (sum of the functions in Figures 1a and 1d)

Figure 1 Example of leader's and follower's objective functions of the Continuous Bilevel Selection Problem, depending on the splitting of the total capacity b into leader's and follower's capacities b_l and b_f, respectively. Let $\mathcal{E}_l = \{e_1, e_2, e_3, e_4\}$, $\mathcal{E}_f = \{e_5, e_6, e_7, e_8\}$, $b = 5$, $c(e_1) = -1$, $c(e_2) = -1$, $c(e_3) = 0$, $c(e_4) = 3$, $c(e_5) = 1$, $c(e_6) = -3$, $c(e_7) = 2$, $c(e_8) = -1$, $d(e_5) = -2$, $d(e_6) = 0$, $d(e_7) = 1$, and $d(e_8) = 1$. We adopt the pessimistic setting, which is reflected in the order of the items e_7 and e_8.
The terms $c(x)$, $c(y)$, and $d(y)$ are used as shortcuts for the objective function values $\sum_{e \in \mathcal{E}_l} c(e)x_e$, $\sum_{e \in \mathcal{E}_f} c(e)y_e$, and $\sum_{e \in \mathcal{E}_f} d(e)y_e$, where $x \in [0,1]^{\mathcal{E}_l}$ and $y \in [0,1]^{\mathcal{E}_f}$ are optimal leader's and follower's solutions for given values of b_l and b_f, respectively.
The dashed linear pieces are parts of the functions that correspond to infeasible overall solutions because both leader and follower have to select at least one item in order to reach the total desired capacity of $b = 5$. More formally, we have $b_l^- = b_f^- = 1$ and $b_l^+ = b_f^+ = 4$ here.

As before, in case of disjoint item sets, it is clear that, for any fixed splitting of the capacity b into capacities b_l and b_f (which are not necessarily integers now) for the leader and the follower, respectively, it is optimal for each of the players to choose their solution greedily. We will now look at the leader's objective function value resulting from any feasible choice of b_l.

First focus on the leader only solving her own Continuous Selection Problem on $\mathcal{E}_l$ with a varying capacity $b_l \in [b_l^-, b_l^+]$, where $b_l^- = \max\{0, b - n_f\}$ and $b_l^+ = \min\{b, n_l\}$ are defined as in Lines 3 and 4 of Algorithm 1. Note that, as in the original Bilevel Selection Problem, feasible solutions for the bilevel problem arise from exactly these choices of b_l. The order of the items in $\mathcal{E}_l$ can be assumed to be fixed, independent of b_l, and given by sorting the item costs $c(e)$ non-decreasingly. For any b_l, the leader selects a *fractional prefix* of this order, i.e., she selects the first $\lfloor b_l \rfloor$ items of the order completely and a fraction $b_l - \lfloor b_l \rfloor$ of the next one. For the total costs of the selected items, this means that it changes linearly between any two integer values of b_l, where the slope of this linear piece is given by the current fractional item's cost. At an integer b_l, the function proceeds continuously, but the slope changes to the next item's cost. This results in a continuous piecewise linear function with vertices at integer values of b_l and linear pieces corresponding to the items. Because of the greedy order of the items, the slopes of the linear pieces are sorted non-decreasingly. Hence, the function is convex. An example of such a function is displayed in Figure 1a. We denote the function that is constructed in this way by $\mathrm{PLF}(\mathcal{E}_l, c \restriction \mathcal{E}_l, b_l^-, b_l^+)$. The data that are required to define it are the set $\mathcal{E}_l$ of items, the function $c\colon \mathcal{E}_l \to \mathbb{Q}$ of item costs that we sort the items by, and the range $[b_l^-, b_l^+]$ of the piecewise linear function, corresponding to the range of capacities that we allow. When writing $\mathrm{PLF}(\mathcal{E}, c', b^-, b^+)$ for a finite set $\mathcal{E}$, a function $c'\colon \mathcal{E} \to \mathbb{Q}$, and numbers $b^-, b^+ \in \mathbb{N}_0$, we always assume that $0 \leq b^- \leq b^+ \leq |\mathcal{E}|$, i.e., that the function is well-defined on the whole range. The piecewise linear function $\mathrm{PLF}(\mathcal{E}, c', b^-, b^+)$ can be computed in time $O(n' \log n')$, where $n' = |\mathcal{E}|$, because we need to sort the items $e \in \mathcal{E}$ by their costs $c'(e)$, and it can be stored as a list of its vertices, which are given by the function's values in the endpoints and the integers in the range $[b_l^-, b_l^+]$.

For the follower, the setting is analogous: He solves a Continuous Selection Problem on $\mathcal{E}_f$ with varying capacity $b_f \in [b_f^-, b_f^+]$, where $b_f^- = b - b_l^+ = \max\{0, b - n_l\}$ and $b_f^+ = b - b_l^- = \min\{b, n_f\}$. We can assume that the items in $\mathcal{E}_f$ are sorted by their follower's item costs $d(e)$ in non-decreasing order (and, in case of non-uniqueness, additionally respecting the leader's item costs according to the optimistic or pessimistic setting; see Remark 1). Therefore, his objective function, viewed as a function of b_f, has the same structural properties as the part of the leader's objective function that results from the items selected by the leader, as described above. For an example, see Figure 1b. Following the notation introduced above, this function can be denoted by $\mathrm{PLF}(\mathcal{E}_f, (d, \pm c \restriction \mathcal{E}_f), b_f^-, b_f^+)$. By $(d, \pm c \restriction \mathcal{E}_f)$ we mean that the items $e \in \mathcal{E}_f$ are sorted by their costs $d(e)$ and, as a secondary criterion, by the leader's costs $c(e)$, in a non-decreasing or a non-increasing order, in the optimistic or the pessimistic setting, respectively.

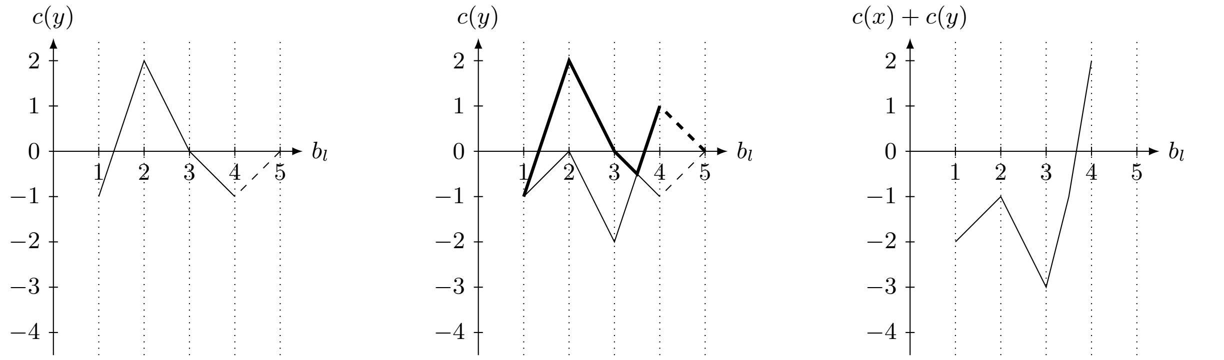
However, we are mainly interested in the contribution the follower's solution makes to the leader's objective function. This means that we need to take the leader's instead of the follower's item costs of the follower's items into account for the leader's objective function, while their order still depends on the follower's item costs. This does not change that the function is continuous and piecewise linear with vertices at integer values of b_f, but the slopes of the linear pieces now correspond to the leader's item costs. Therefore, the function is not convex anymore in general. See Figure 1c for the resulting function in the example.

Moreover, in order to combine the two parts of the leader's objective function, we need to use $b_l = b - b_f$ as a parameter instead of b_f, also for the function described in the previous paragraph. This corresponds to mirroring it such that left and right are swapped. In terms of the follower's greedy algorithm, we can imagine that he starts at $b_l = b_l^-$ by selecting his $b_f^+ = b - b_l^-$ best items and then removing them greedily in the reverse order. The transformed function in the example is displayed in Figure 1d. We will denote this function by $\overline{\mathrm{PLF}}(\mathcal{E}_f, (d, \pm c \restriction \mathcal{E}_f), c \restriction \mathcal{E}_f, b_l^-, b_l^+, b)$. When writing $\overline{\mathrm{PLF}}(\mathcal{E}, (d', \pm c'), c'', b^-, b^+, b')$, the required data now consists of the finite set $\mathcal{E}$ of items and their costs we sort by, in the form $(d', \pm c')$ for functions $d', c'\colon \mathcal{E} \to \mathbb{Q}$ as above, the function $c''\colon \mathcal{E} \to \mathbb{Q}$ of item costs that we use to determine the slopes of the linear pieces, the range $[b^-, b^+]$ of the piecewise linear function, with $b^-, b^+ \in \mathbb{N}_0$, and the total capacity $b' \in \mathbb{N}_0$ that is required to perform the swapping correctly. In order to get a well-defined function, we assume $0 \leq b' - b^+ \leq b' - b^- \leq |\mathcal{E}|$ here. As above, this piecewise linear function can be computed in time $O(n' \log n')$, where $n' = |\mathcal{E}|$.

Finally, the leader's objective, as a function of b_l, is given by the sum of the two functions that describe her objective function values resulting from the greedy choices of herself and the follower, respectively. As a sum of two such functions, it is also a continuous piecewise linear function with vertices at integer values of b_l. See Figure 1e for the total leader's objective function in the example. Given the two functions as described above, their sum can be computed in time $O(n)$.

Optimal leader's solutions now correspond to minima of this function. Because of its continuous piecewise linear structure, the function always has a minimum at an integer value of b_l. Together with integer optimal solutions of the resulting leader's and follower's Continuous Selection Problems, this yields an optimal solution of the Continuous Bilevel Selection Problem that is also valid for the binary problem version.

Note that, also in [6], the leader's objective function of the Bilevel Continuous Knapsack Problem is described as a piecewise linear function. Since the leader directly controls the follower's capacity in this problem, only one function as in Figure 1c is required, while we here need to combine it with another piecewise linear function corresponding to the leader's items. Recall also the relation of the two problems described in Section 2.2.1. Another difference is related to the Continuous Selection Problem being the special case of the Continuous Knapsack Problem where all item sizes are one: The linear pieces all have a width of 1 here (i.e., there might be a vertex at every integer value of b_l or b_f), while the widths correspond to the item sizes in the Bilevel Continuous Knapsack Problem. Accordingly, the slopes are ratios of item values and item sizes in the Bilevel Continuous Knapsack Problem, but only the item costs here.



(a) Leader's objective of the follower's items, in scenario d_2, $\overline{\mathrm{PLF}}(\mathcal{E}_f, (d_2, -c \upharpoonright \mathcal{E}_f), c \upharpoonright \mathcal{E}_f, b_l^-, b_l^+, b)$

(b) Leader's objective of the follower's items, in the worst case (pointwise maximum of the functions in Figures 1d and 2a)

(c) Leader's total objective (sum of the functions in Figures 1a and 2b)

Figure 2 Example of the leader's objective function of the Robust Continuous Bilevel Selection Problem with discrete uncertainty, depending on the leader's capacity b_l. Let $\mathcal{E}_l = \{e_1, e_2, e_3, e_4\}$, $\mathcal{E}_f = \{e_5, e_6, e_7, e_8\}$, $b = 5$, $c(e_1) = -1$, $c(e_2) = -1$, $c(e_3) = 0$, $c(e_4) = 3$, $c(e_5) = 1$, $c(e_6) = -3$, $c(e_7) = 2$, $c(e_8) = -1$, and $\mathcal{U} = \{d_1, d_2\}$ with $d_1(e_5) = -2$, $d_1(e_6) = 0$, $d_1(e_7) = 1$, $d_1(e_8) = 1$, $d_2(e_5) = -1$, $d_2(e_6) = 3$, $d_2(e_7) = 0$, and $d_2(e_8) = -3$. This is the same instance as in Figure 1 without uncertainty, but now with an additional second scenario d_2. We again adopt the pessimistic setting, which is reflected in the order of the items e_7 and e_8 in scenario d_1.

6.2 The Robust Continuous Bilevel Selection Problem

We now extend the general structural observations of Section 6.1 to the corresponding robust problem. We will first describe some general insights and show in Example 26 that the Robust Continuous Bilevel Selection Problem is not equivalent to the binary version, in contrast to the problem without uncertainty. Afterwards, we will investigate the Robust Continuous Bilevel Selection Problem for specific types of uncertainty sets: In Sections 6.2.1, 6.2.2, and 6.2.3, we give polynomial-time algorithms for the Robust Continuous Bilevel Selection Problem with discrete uncertainty, interval uncertainty, and discrete uncorrelated uncertainty, respectively. While the results for discrete and interval uncertainty make use of similar ideas as before and in [6], a modified approach is necessary in case of discrete uncorrelated uncertainty. The latter also provides a pseudopolynomial-time algorithm for the Robust Bilevel Continuous Knapsack Problem with discrete uncorrelated uncertainty, which answers an open question stated in [6]; see Appendix A.

Regarding the representation of the leader's objective function from Section 6.1, observe that the function describing the leader's costs of the items selected by herself is not affected when introducing uncertainty, whereas the function describing the follower's selection can look different in every scenario. Recall that the linear pieces of this function correspond to the selected items, where their slopes are given by the leader's item costs. Hence, when the adversary's decision affects the greedy order of the follower's items, this means that the order of the linear pieces in the function changes (while the function still remains continuous). Extending the example from Figure 1, we refer to Figure 2a for an example of such a function in a different scenario, i.e., only the order of the linear pieces is different compared to the function in Figure 1d.

For understanding the overall leader's objective function in the robust setting, remember the order in which the three decisions are made: First, the leader determines b_l and her own selection of items, which corresponds to the first function (Figure 1a). Second, the adversary chooses a scenario and, with that, one of the other piecewise linear functions (Figures 1d and 2a in our example). The follower's decision is already implicit when the adversary's decision is interpreted as choosing one of the piecewise linear functions. Since the adversary decides after b_l is fixed, the adversary's and the follower's contribution to the leader's objective function is the pointwise maximum of the piecewise linear functions corresponding to the scenarios; see Figure 2b. The overall leader's objective function can thus be written as the sum of a piecewise linear function and a pointwise maximum of the other piecewise linear functions; see Figure 2c. Again, optimal leader's solutions correspond to minima of this function.

One can already see in Figures 2b and 2c that the pointwise maximum coming from the robustness can result in vertices of the piecewise linear functions being not necessarily at integers. We will now show that, for this

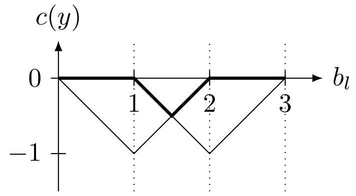


Figure 3 The leader's objective function of the Robust Bilevel Selection Problem instance described in Example 26

reason, there might be no optimal solution that is binary, in contrast to the setting without uncertainty. Hence, the binary and the continuous problem variants are not equivalent in the robust setting.

▶ **Example 26.** Consider the Robust Continuous Bilevel Selection Problem with discrete uncertainty. Let $\mathcal{E}_l = \{e_1, e_2, e_3\}$, $\mathcal{E}_f = \{e_4, e_5, e_6\}$, $b = 3$, $c(e) = 0$ for all $e \in \mathcal{E}_l$, $c(e_4) = -1$, $c(e_5) = 1$, $c(e_6) = 0$, and $\mathcal{U} = \{d_1, d_2\}$ with $d_1(e_4) = 0$, $d_1(e_5) = 1$, $d_1(e_6) = 2$, $d_2(e_4) = 1$, $d_2(e_5) = 2$, and $d_2(e_6) = 0$. Effectively, the leader can give any capacity $b_f = b - b_l \in [0, 3]$ to the follower by choosing any fraction $b_l \in [0, 3]$ of her items, which have objective function value zero; see also Section 2.2.1. Depending on the follower's objective chosen by the adversary, the order of the follower's items in his greedy algorithm is either e_4, e_5, e_6 or e_6, e_4, e_5. As explained above, each of these orders corresponds to a piecewise linear function from the leader's perspective, and the leader's objective function is now the pointwise maximum of these; see Figure 3.

One can easily see that the minimum of the leader's objective function is at $b_l = \frac{3}{2}$, where the two piecewise linear functions corresponding to the two scenarios intersect, i.e., give the same leader's objective value of $-\frac{1}{2}$. In contrast, for every binary leader's decision, corresponding to $b_l \in \{0, 1, 2, 3\}$, the adversary can choose $d \in \mathcal{U}$ (namely d_2 for $b_l \in \{0, 1\}$ and d_1 for $b_l \in \{2, 3\}$) such that the leader's objective value is always 0. This shows that the optimal leader's solution in the continuous problem variant is not necessarily binary in case of discrete uncertainty. Hence, the Robust Continuous Bilevel Selection Problem is indeed not equivalent to the binary Robust Bilevel Selection Problem.

Constructing the leader's objective function as described above, using the pointwise maximum of piecewise linear functions corresponding to the scenarios, directly leads to a polynomial-time solution algorithm for the Robust Continuous Bilevel Selection Problem under discrete uncertainty; see Section 6.2.1. In fact, one could solve the problem using the same approach for every type of uncertainty set, even if $\mathcal{U}$ is not finite. The reason for this is that we do not need to enumerate all elements of $\mathcal{U}$, but only all possible follower's greedy orders arising from the scenarios, corresponding to piecewise linear functions that contribute to the pointwise maximum, and the number of orders of $\mathcal{E}_f$ is finite. Hence, for any uncertainty set, the leader's objective function can be constructed from finitely many piecewise linear functions. However, their number is not polynomial in general, such that the resulting algorithm is usually not efficient. For the cases of interval uncertainty and discrete uncorrelated uncertainty, we will develop efficient algorithms that avoid enumerating all possible follower's greedy orders explicitly, using similar ideas as in Section 4.2; see Sections 6.2.2 and 6.2.3.

6.2.1 Discrete Uncertainty

In case of discrete uncertainty, we have seen in Theorem 6 that the adversary's problem can be solved in polynomial time for the binary setting, by enumerating all scenarios. We now show that the same is true for the continuous setting with disjoint item sets:

▶ **Theorem 27.** *For any fixed feasible leader's solution of the Robust Continuous Bilevel Selection Problem with a discrete uncertainty set $\mathcal{U}$, the adversary's problem can be solved in time $O(|\mathcal{U}|n)$.*

Proof. Let $b_l = \sum_{e \in \mathcal{E}_l} x_e$ be the capacity that the given leader's solution $x \in [0, 1]^{\mathcal{E}_l}$ uses. Then $b_f = b - b_l$ is the capacity to be filled by the follower. Note that, in contrast to Theorem 6, b_l and b_f are not necessarily integers now. However, given disjoint item sets $\mathcal{E}_l$ and $\mathcal{E}_f$, it is still true that the only interaction between leader and follower is via the capacities they use. Hence, the follower's task is to solve a single-level Continuous Selection Problem on $\mathcal{E}_f$ with capacity b_f, with respect to the item costs chosen by the adversary. As in Theorem 6, the adversary can simply enumerate all scenarios in $\mathcal{U}$ and solve the follower's problem for each of them. This can again be done in a running time of $O(|\mathcal{U}|n)$. ◀

As a black box, like in Theorem 11 and Corollary 13, Theorem 27 does not suffice here for constructing an algorithm for the leader's problem because there are infinitely many feasible choices of b_l and Example 26 shows that it is indeed not enough to consider only integers b_l. However, the observations about the leader's objective function in Section 6.1 and in the beginning of Section 6.2 directly imply a polynomial-time algorithm in case of a discrete uncertainty set, using piecewise linear functions:

▶ **Theorem 28.** *The* Robust Continuous Bilevel Selection Problem *with a discrete uncertainty set* $\mathcal{U}$*, can be solved in time* $O(|\mathcal{U}|n\log(|\mathcal{U}|n))$*.*

Proof. As described in the beginning of Section 6.2, the leader's objective function can be written as a sum of two continuous piecewise linear functions one of which is a pointwise maximum of $|\mathcal{U}|$ piecewise linear functions. In order to solve the leader's problem, we can compute all these functions explicitly and minimize the resulting continuous piecewise linear function by evaluating it in all its vertices. This leads to an optimal capacity b_l the leader should choose, together with the corresponding leader's greedy solution.

Regarding the running time, first observe that the function corresponding to the items selected by the leader and all functions corresponding to the items selected by the follower in the different scenarios can be computed in total time $O(|\mathcal{U}|n\log n)$ by sorting the items according to their costs. The pointwise maximum of the $|\mathcal{U}|$ piecewise linear functions with $O(n)$ vertices each can be computed in time $O(|\mathcal{U}|n\log(|\mathcal{U}|))$: Since the vertices of the original functions are all at integers, each of the $O(n)$ sections between two subsequent integer values of b_l can be considered separately. For each of these sections, we need to compute the pointwise maximum of $|\mathcal{U}|$ many linear pieces. By [24], this can be done in $O(|\mathcal{U}|\log(|\mathcal{U}|))$. Together, this results in a running time of $O(|\mathcal{U}|n\log(|\mathcal{U}|n))$. ◀

Note that Theorem 28 can be seen as a generalization of Theorem 1 in [6]; see also Section 2.2.1.

6.2.2 Interval Uncertainty

In the setting of interval uncertainty, almost the same ideas as in Section 4.2 also apply to the continuous case. In particular, the reasoning of Lemma 7 is still true, and also the idea of the precedence constraint knapsack problem from [35] is still applicable. However, as we are allowed to select fractions of items now, we have to use an algorithm for the Continuous Selection Problem instead of the Selection Problem as a subroutine. Moreover, a technical complication arises from the fact that, if a fractional solution is selected, it becomes important which item can be the last one of the *initial set* because this will be the item that is selected fractionally by the follower afterwards. This is the same issue that also appears in case of the Robust Bilevel Continuous Knapsack Problem in [6] and is described using the term of a *fractional prefix* there.

Since the results obtained in this section are closely related to the ones in Section 4 of [6], we omit detailed explanations and proofs here. Recall from Section 2.2.1 that the Robust Continuous Bilevel Selection Problem generalizes the Robust Bilevel Continuous Knapsack Problem from [6] in one aspect, namely the leader controlling her own items instead of just setting the capacity for the follower's problem, while the problem is more special in another aspect, namely being based on the Continuous Selection Problem instead of the Continuous Knapsack Problem. While the latter difference only leads to a minor simplification regarding the approach developed in [6], the former one now makes it necessary to deal with an additional piecewise linear function corresponding to the leader's items (see Section 6.1 and the beginning of Section 6.2) when solving the leader's problem.

As in Section 4.2, for simplicity, we assume throughout this section that there are no one-point intersections between intervals involved in $\mathcal{U}$, i.e., that, for all $e_1, e_2 \in \mathcal{E}_f$, we have $d^-(e_1) \neq d^+(e_2)$; see also Remark 2 in [6].

Translating the idea of Algorithm 2 to the continuous setting results in Algorithm 3 for solving the adversary's problem. Note that we may assume that the subroutine uses the standard greedy algorithm for the Continuous Selection Problem and, in particular, that it always returns a solution with at most one nonbinary value. This assumption ensures that there is indeed at most one $e \in \mathcal{E}_f$ selected fractionally, i.e., with $y_e^{\bar{e}} \in (0,1)$, in Line 12 of the algorithm.

In analogy to Lemma 1 in [6], one can show:

▶ **Theorem 29.** *For any fixed feasible leader's solution of the* Robust Continuous Bilevel Selection Problem *with interval uncertainty, Algorithm 3 solves the adversary's problem in time* $O(n^2)$*.*

Algorithm 3: Algorithm for the adversary's problem of the Robust Continuous Bilevel Selection Problem with interval uncertainty

Input : finite sets $\mathcal{E}_l$ and $\mathcal{E}_f$ with $\mathcal{E}_f \neq \emptyset$ and $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$, $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$, $c \colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$, an interval uncertainty set $\mathcal{U}$ given by $d^-, d^+ \colon \mathcal{E}_f \to \mathbb{Q}$ with $d^-(e) \leq d^+(e)$ for all $e \in \mathcal{E}_f$, a feasible leader's solution $x \in [0,1]^{\mathcal{E}_l}$

Output : an optimal adversary's solution $d \in \mathcal{U}$ (i.e., $d \colon \mathcal{E}_f \to \mathbb{Q}$ with $d^-(e) \leq d(e) \leq d^+(e)$ for all $e \in \mathcal{E}_f$) of the Robust Continuous Bilevel Selection Problem

$b_f := b - \sum_{e \in \mathcal{E}_l} x_e$
$\bar{\mathcal{E}} := \emptyset$
for $\bar{e} \in \mathcal{E}_f$ **do**
  $\mathcal{E}^-_{\bar{e}} := \{e \in \mathcal{E}_f \mid d^+(e) < d^-(\bar{e})\}$
  $\mathcal{E}^0_{\bar{e}} := \{e \in \mathcal{E}_f \mid d^-(e) \leq d^-(\bar{e}) \leq d^+(e)\}$
  if $|\mathcal{E}^-_{\bar{e}}| \leq b_f$ *and* $|\mathcal{E}^0_{\bar{e}}| \geq b_f - |\mathcal{E}^-_{\bar{e}}|$ **then**
    $y^{\bar{e},0} :=$ Continuous Selection Problem$(\mathcal{E}^0_{\bar{e}}, b_f - |\mathcal{E}^-_{\bar{e}}|, -c \upharpoonright \mathcal{E}^0_{\bar{e}})$
    define $y^{\bar{e}} \in [0,1]^{\mathcal{E}_f}$ by $y^{\bar{e}}_e := 1$ for all $e \in \mathcal{E}^-_{\bar{e}}$, $y^{\bar{e}}_e := y^{\bar{e},0}_e$ for all $e \in \mathcal{E}^0_{\bar{e}}$, and $y^{\bar{e}}_e := 0$ for all $e \in \mathcal{E}_f \setminus (\mathcal{E}^-_{\bar{e}} \cup \mathcal{E}^0_{\bar{e}})$
    $\bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{\bar{e}\}$
select $\bar{e} \in \operatorname{argmax}\{\sum_{e' \in \mathcal{E}_f} c(e') y^e_{e'} \mid e \in \bar{\mathcal{E}}\}$ arbitrarily
$\varepsilon := \frac{1}{2} \min_{e \in \mathcal{E}^0_{\bar{e}}} (d^+(\bar{e}) - d^-(e))$
return $d \colon \mathcal{E}_f \to \mathbb{Q}$ *with* $d(e) := d^-(e)$ *for all* $e \in \{e' \in \mathcal{E}_f \mid y^{\bar{e}}_{e'} = 1\}$*,* $d(e) := d^+(e)$ *for all* $e \in \{e' \in \mathcal{E}_f \mid y^{\bar{e}}_{e'} = 0\}$*, and* $d(e) := d^-(\bar{e}) + \varepsilon$ *for the unique (if it exists)* $e \in \mathcal{E}_f$ *with* $y^{\bar{e}}_e \in (0,1)$

Observe that, if b_f is an integer, which is, in particular, the case if the leader has taken a binary decision, then Algorithm 3 coincides with Algorithm 2 for the binary setting. Indeed, the subroutine solving the Continuous Selection Problem is only called for integer capacities then, and hence solves a binary Selection Problem.

In contrast to Corollary 14, we cannot directly derive an algorithm for the leader's problem from Algorithm 3 here because it does not suffice to enumerate integer values of b_l in the continuous setting. However, the adversary's algorithm can be generalized to a polynomial-time algorithm for the leader by combining it with the idea of representing the leader's objective function by piecewise linear functions; see Section 6.1 and the beginning of Section 6.2. The resulting Algorithm 4 computes a small number of linear pieces that might contribute to the pointwise maximum, using the strategy of Algorithm 3. Recall the definitions of PLF and $\overline{\text{PLF}}$ from Section 6.1 (with Figure 1).

Similarly to Theorem 2 in [6] and using the observations from Section 6.1 and the beginning of Section 6.2, we derive:

▶ **Theorem 30.** *Algorithm 4 solves the Robust Continuous Bilevel Selection Problem with interval uncertainty in time* $O(n^2 \log n)$*.*

6.2.3 Discrete Uncorrelated Uncertainty

In the binary setting, we have shown that discrete uncorrelated uncertainty is equivalent to interval uncertainty; see Theorem 9. In particular, the optimal adversary's solutions computed by Algorithm 2 for interval uncertainty were always attained at an endpoint of each interval. The latter is not true anymore in Algorithm 3, the variant of this algorithm that solves the adversary's problem in the continuous case. In fact, this algorithm sets all costs $d(e)$ to endpoints of the respective intervals, except for the one that corresponds to the unique item the follower is supposed to select only a fraction of. The special adversary's choice of this value is necessary because this item has to be guaranteed to be the last one the follower selects, i.e., the last item in the corresponding fractional prefix. This is also demonstrated in Examples 1 and 2 in [6]. Hence, the solution returned by Algorithm 3 cannot be assumed to be feasible for the adversary's problem in case of discrete uncorrelated uncertainty in general. Therefore, we cannot solve the Robust Continuous Bilevel Selection Problem with discrete uncorrelated uncertainty by simply replacing the uncertainty set by the interval uncertainty set that is its convex hull, as in Theorem 9.

Algorithm 4: Algorithm for the leader's problem of the Robust Continuous Bilevel Selection Problem with interval uncertainty

Input : finite sets $\mathcal{E}_l$ and $\mathcal{E}_f$ with $\mathcal{E}_f \neq \emptyset$ and $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$, $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$, $c\colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$, an interval uncertainty set $\mathcal{U}$ given by $d^-, d^+\colon \mathcal{E}_f \to \mathbb{Q}$ with $d^-(e) \leq d^+(e)$ for all $e \in \mathcal{E}_f$

Output : an optimal leader's solution $x \in [0,1]^{\mathcal{E}_l}$ of the Robust Continuous Bilevel Selection Problem

$b_l^- := \max\{0, b - |\mathcal{E}_f|\}$, $b_f^+ := b - b_l^-$
$b_l^+ := \min\{b, |\mathcal{E}_l|\}$, $b_f^- := b - b_l^+$
compute $g_l := \mathrm{PLF}(\mathcal{E}_l, c \restriction \mathcal{E}_l, b_l^-, b_l^+)$
$\bar{\mathcal{E}} := \emptyset$
for $\bar{e} \in \mathcal{E}_f$ **do**
    $\mathcal{E}_{\bar{e}}^- := \{e \in \mathcal{E}_f \mid d^+(e) < d^-(\bar{e})\}$
    $\mathcal{E}_{\bar{e}}^0 := \{e \in \mathcal{E}_f \mid d^-(e) \leq d^-(\bar{e}) \leq d^+(e)\}$
    if $|\mathcal{E}_{\bar{e}}^-| \leq b_f^+$ *and* $|\mathcal{E}_{\bar{e}}^0| \geq b_f^- - |\mathcal{E}_{\bar{e}}^-|$ **then**
        $\overline{b_f^-} := \max\{0, b_f^- - |\mathcal{E}_{\bar{e}}^-|\}$
        $\overline{b_f^+} := \min\{b_f^+ - |\mathcal{E}_{\bar{e}}^-|, |\mathcal{E}_{\bar{e}}^0|\}$
        compute $g_f^{\bar{e}} := \overline{\mathrm{PLF}}(\mathcal{E}_{\bar{e}}^0, -c \restriction \mathcal{E}_{\bar{e}}^0, c \restriction \mathcal{E}_{\bar{e}}^0, b - \overline{b_f^+}, b - \overline{b_f^-}, b)$
        compute $g_f^{\bar{e}} := g_f^{\bar{e}} + (-|\mathcal{E}_{\bar{e}}^-|, \sum_{e \in \mathcal{E}_{\bar{e}}^-} c(e))$ (i.e., shift the function $g_f^{\bar{e}}$ by $-|\mathcal{E}_{\bar{e}}^-|$ in x direction and by $\sum_{e \in \mathcal{E}_{\bar{e}}^-} c(e)$ in y direction)
        $\bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{\bar{e}\}$
compute $g_f := \max\{g_f^{\bar{e}} \mid \bar{e} \in \bar{\mathcal{E}}\}$ (i.e., the pointwise maximum)
compute $g := g_l + g_f$
compute some $b_l^* \in \operatorname{argmin}\{g(b_l) \mid b_l \in [b_l^-, b_l^+]\}$
return *Continuous Selection Problem*$(\mathcal{E}_l, b_l^*, c \restriction \mathcal{E}_l)$

Nevertheless, one can solve the Robust Continuous Bilevel Selection Problem and its adversary's problem in case of discrete uncorrelated uncertainty in polynomial time using similar ideas as for interval uncertainty, as we show in this section. Moreover, in Appendix A we present a generalization of the ideas, resulting in pseudopolynomial-time algorithms for the adversary's problem and the leader's problem of the Robust Bilevel Continuous Knapsack Problem with discrete uncorrelated uncertainty, adding to the complexity results in [6].

We first study the adversary's problem of the Robust Continuous Bilevel Selection Problem with discrete uncorrelated uncertainty. Assume that a feasible leader's solution $x \in [0,1]^{\mathcal{E}_l}$ is given. As before, this defines the capacity $b_f = b - \sum_{e \in \mathcal{E}_l} x_e$ the follower has to fill. First note that, in case b_f is an integer, the follower can be assumed to always make a binary decision because he solves a Continuous Selection Problem with an integer capacity. Therefore, we can handle the adversary's problem like in the binary case, i.e., apply Algorithm 2, because of Theorem 9. For the general case where b_f is not an integer, we develop a modified version of this algorithm and consider the follower's solution as being composed of a binary selection of $\lfloor b_f \rfloor$ items and an additional item of which the follower selects a fraction of $b_f - \lfloor b_f \rfloor$.

Recall that, in Algorithms 2, 3, and 4, we iterate over all possible *heads* $\bar{e}$ of the (fractional) prefix corresponding to the follower's solution and use the left endpoint $d^-(\bar{e})$ of its interval as a reference point for defining the item sets $\mathcal{E}_{\bar{e}}^-$ and $\mathcal{E}_{\bar{e}}^0$ from which the adversary builds a worst-case follower's solution for the current iteration. For more details, we refer to [6]. Algorithm 3 assumes that any item in $\mathcal{E}_{\bar{e}}^0$ can be made the fractional item by selecting the cost $d^-(\bar{e}) + \varepsilon$ for it. On a related note, for a number of intervals that have a proper intersection, the adversary can enforce any order of the corresponding items. When we turn to discrete uncorrelated uncertainty, the adversary's choice is more limited and fewer follower's greedy orders are possible; see also Remark 10. Regarding the possible fractional items, this is a significant limitation. Therefore, we have to take a closer look at the specific cost values the adversary can select for the fractional item.

Instead of the head, which is not necessarily the fractional item in Algorithm 3 and in [6], we now guess the item $e^* \in \mathcal{E}_f$ that the follower selects only a fraction of, together with the cost value $\delta^* \in \mathcal{U}_{e^*}$ the adversary assigns to it. Note that there are $\sum_{e \in \mathcal{E}_f} |\mathcal{U}_e|$ such guesses to consider in total. The remaining adversary's problem is then similar to the problem solved in an iteration of Algorithm 2, i.e., it has to be decided which $\lfloor b_f \rfloor$ items the follower selects before the fractional item. This can again be solved as a binary Selection Problem, while we now use δ^* as a reference point in the definition of the item sets $\mathcal{E}^-_{e^*,\delta^*}$ and $\mathcal{E}^0_{e^*,\delta^*}$. Among all possible choices of (e^*, δ^*), the adversary selects the one that yields the worst leader's costs of the resulting follower's solution. This strategy leads to Algorithm 5.

Algorithm 5: Algorithm for the adversary's problem of the Robust Continuous Bilevel Selection Problem with discrete uncorrelated uncertainty

---

Input : finite sets $\mathcal{E}_l$ and $\mathcal{E}_f$ with $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$, $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$, $c\colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$, a discrete uncorrelated uncertainty set $\mathcal{U}$ given by a finite set $\mathcal{U}_e$ for every $e \in \mathcal{E}_f$, a feasible leader's solution $x \in [0,1]^{\mathcal{E}_l}$

Output: an optimal adversary's solution $d \in \mathcal{U}$ (i.e., a value $d(e) \in \mathcal{U}_e$ for every $e \in \mathcal{E}_f$) of the Robust Continuous Bilevel Selection Problem

$b_f := b - \sum_{e \in \mathcal{E}_l} x_e$
if $b_f = 0$ *or* $b_f = |\mathcal{E}_f|$ **then**
  return *an arbitrary* $d \in \mathcal{U}$
for $e \in \mathcal{E}_f$ **do**
  $d^-(e) := \min \mathcal{U}_e$, $d^+(e) := \max \mathcal{U}_e$
$\bar{\mathcal{E}} := \emptyset$
for $e^* \in \mathcal{E}_f$ **do**
  for $\delta^* \in \mathcal{U}_{e^*}$ **do**
    $\mathcal{E}^-_{e^*,\delta^*} := \{e \in \mathcal{E}_f \mid d^+(e) < \delta^*\}$
    $\mathcal{E}^0_{e^*,\delta^*} := \{e \in \mathcal{E}_f \mid d^-(e) \le \delta^* \le d^+(e)\} \setminus \{e^*\}$
    if $|\mathcal{E}^-_{e^*,\delta^*}| \le \lfloor b_f \rfloor$ *and* $|\mathcal{E}^0_{e^*,\delta^*}| \ge \lfloor b_f \rfloor - |\mathcal{E}^-_{e^*,\delta^*}|$ **then**
      $Y^0_{e^*,\delta^*} :=$ Selection Problem$(\mathcal{E}^0_{e^*,\delta^*}, \lfloor b_f \rfloor - |\mathcal{E}^-_{e^*,\delta^*}|, -c \restriction \mathcal{E}^0_{e^*,\delta^*})$
      $Y_{e^*,\delta^*} := \mathcal{E}^-_{e^*,\delta^*} \cup Y^0_{e^*,\delta^*}$
      $\bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{(e^*, \delta^*)\}$
select $(e^*, \delta^*) \in \operatorname{argmax}\{c(Y_{e,\delta}) + c(e) \cdot (b_f - \lfloor b_f \rfloor) \mid (e, \delta) \in \bar{\mathcal{E}}\}$ arbitrarily
return $d\colon \mathcal{E}_f \to \mathbb{Q}$ *with* $d(e^*) := \delta^*$, $d(e) = d^-(e)$ *for all* $e \in Y_{e^*,\delta^*}$ *and* $d(e) := d^+(e)$ *for all* $e \in \mathcal{E}_f \setminus (Y_{e^*,\delta^*} \cup \{e^*\})$

Again, similarly to the proof of Lemma 1 in [6], one can show:

▶ **Theorem 31.** *For any fixed feasible leader's solution of the Robust Continuous Bilevel Selection Problem with a discrete uncorrelated uncertainty set $\mathcal{U} = \prod_{e \in \mathcal{E}_f} \mathcal{U}_e$, Algorithm 5 solves the adversary's problem in time $O(nu)$, where $u = \sum_{e \in \mathcal{E}_f} |\mathcal{U}_e|$.*

Observe that, if b_f is an integer (and we are not in one of the trivial cases $b_f = 0$ and $b_f = |\mathcal{E}_f|$), then Algorithm 5 selects a fraction of $b_f - \lfloor b_f \rfloor = 0$ of the item e^*. Hence, the algorithm examines binary prefixes, very similarly to Algorithm 2; only more possible choices of the reference point δ^* are considered here, which does not change the result in this case.

In order to devise an algorithm for the leader's problem, in case of interval uncertainty, we combined the approach used for solving the adversary's problem with the idea to represent the leader's objective function by piecewise linear functions; see Algorithm 4. Something similar can be done here in order to generalize Algorithm 5 to an algorithm that solves the leader's problem. This results in Algorithm 6.

The computation of the function g_l (Line 6), representing the leader's costs of the items she selects herself, and the final steps to determine an optimal solution (Lines 22 to 24) are as in Algorithm 4; they are independent of the type of uncertainty set. The function g_f, representing the leader's costs of the items selected by the follower, is now computed by a generalization of Algorithm 5. Indeed, for every fixed follower's capacity $b_f \in [b_f^-, b_f^+]$, the value $g_f^{b_f^*}(b_f)$, for $b_f^* = \lfloor b_f \rfloor$ (or $b_f^* = b_f - 1$ in case $b_f = b_f^+$), and therefore the value $g_f(b_f)$, corresponds to the worst leader's costs of a follower's selection that the adversary can achieve. The approach to determine this value is like in Algorithm 5: All possible fractional items $e^* \in \mathcal{E}_f$ and all possible adversary's choices $\delta^* \in \mathcal{U}_{e^*}$ are enumerated and a Selection Problem is solved in each iteration. This is independent of the precise

Algorithm 6: Algorithm for the leader's problem of the Robust Continuous Bilevel Selection Problem with discrete uncorrelated uncertainty

Input : finite sets $\mathcal{E}_l$ and $\mathcal{E}_f$ with $\mathcal{E}_l \cap \mathcal{E}_f = \emptyset$, $b \in \{0, \dots, |\mathcal{E}_l \cup \mathcal{E}_f|\}$, $c\colon \mathcal{E}_l \cup \mathcal{E}_f \to \mathbb{Q}$, a discrete uncorrelated uncertainty set $\mathcal{U}$ given by a finite set $\mathcal{U}_e$ for every $e \in \mathcal{E}_f$

Output : an optimal leader's solution $x \in [0,1]^{\mathcal{E}_l}$ of the Robust Continuous Bilevel Selection Problem

1 $b_l^- := \max\{0, b - |\mathcal{E}_f|\}$, $b_f^+ := b - b_l^-$
2 $b_l^+ := \min\{b, |\mathcal{E}_l|\}$, $b_f^- := b - b_l^+$
3 **if** $b_l^- = b_l^+$ **then**
4    $b_l^* := b_l^-$
5 **else**
6   compute $g_l := \mathrm{PLF}(\mathcal{E}_l, c \upharpoonright \mathcal{E}_l, b_l^-, b_l^+)$
7   **for** $e \in \mathcal{E}_f$ **do**
8      $d^-(e) := \min \mathcal{U}_e$, $d^+(e) := \max \mathcal{U}_e$
9   **for** $b_f^* \in \{b_f^-, \dots, b_f^+ - 1\}$ **do**
10      $\bar{\mathcal{E}} := \emptyset$
11   **for** $e^* \in \mathcal{E}_f$ **do**
12   **for** $\delta^* \in \mathcal{U}_{e^*}$ **do**
13          $\mathcal{E}^-_{e^*,\delta^*} := \{e \in \mathcal{E}_f \mid d^+(e) < \delta^*\}$
14          $\mathcal{E}^0_{e^*,\delta^*} := \{e \in \mathcal{E}_f \mid d^-(e) \le \delta^* \le d^+(e)\} \setminus \{e^*\}$
15   **if** $|\mathcal{E}^-_{e^*,\delta^*}| \le b_f^*$ *and* $|\mathcal{E}^0_{e^*,\delta^*}| \ge b_f^* - |\mathcal{E}^-_{e^*,\delta^*}|$ **then**
16            $Y^0_{e^*,\delta^*} :=$ Selection Problem$(\mathcal{E}^0_{e^*,\delta^*}, b_f^* - |\mathcal{E}^-_{e^*,\delta^*}|, -c \upharpoonright \mathcal{E}^0_{e^*,\delta^*})$
17            $Y_{e^*,\delta^*} := \mathcal{E}^-_{e^*,\delta^*} \cup Y^0_{e^*,\delta^*}$
18            $g_f^{b_f^*,e^*,\delta^*} :=$ linear piece from $(b - b_f^* - 1, c(Y_{e^*,\delta^*}) + c(e^*))$ to $(b - b_f^*, c(Y_{e^*,\delta^*}))$
19            $\bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{(e^*, \delta^*)\}$
20   compute $g_f^{b_f^*} := \max\{g_f^{b_f^*,e^*,\delta^*} \mid (e^*,\delta^*) \in \bar{\mathcal{E}}\}$ (i.e., the pointwise maximum)
21   join all $g_f^{b_f^*}$ (with range $[b - b_f^* - 1, b - b_f^*]$) to a function g_f with range $[b_l^-, b_l^+]$
22   compute $g := g_l + g_f$
23   compute some $b_l^* \in \operatorname{argmin}\{g(b_l) \mid b_l \in [b_l^-, b_l^+]\}$
24 **return** *Continuous Selection Problem*$(\mathcal{E}_l, b_l^*, c \upharpoonright \mathcal{E}_l)$

fraction $b_f - \lfloor b_f \rfloor \in [0,1)$ according to which the follower selects the fractional item. Therefore, we consider a linear piece corresponding to all possible fractions in Line 18. The worst case among all possible choices of (e^*, δ^*), which can be chosen individually for each b_f by the adversary, now corresponds to the pointwise maximum of these linear pieces (Line 20). Finally, we handle each integer value b_f^* separately, similarly to the binary problem version. On a related note, observe that we solve a binary Selection Problem as a subroutine in Algorithms 5 and 6, and not a Continuous Selection Problem as in Algorithms 3 and 4; see also Remark 33 below.

Using the ideas explained above, one can prove, again similarly to Theorem 2 in [6]:

▶ **Theorem 32.** *Algorithm 6 solves the Robust Continuous Bilevel Selection Problem with a discrete uncorrelated uncertainty set* $\mathcal{U} = \prod_{e \in \mathcal{E}_f} \mathcal{U}_e$ *in time* $O(nu \log(nu))$, *where* $u = \sum_{e \in \mathcal{E}_f} |\mathcal{U}_e|$.

We emphasize that the term u and therefore the running times stated in Theorems 31 and 32 are polynomial in the input size, while the total size $|\mathcal{U}|$ of the discrete uncorrelated uncertainty set is exponential in the input size. Moreover, if the uncertainty set is given by sets $\mathcal{U}_e$ of constant size, in particular if each $\mathcal{U}_e$ consists of only two values $d^-(e)$ and $d^+(e)$, then the running time of Algorithm 5 is $O(n^2)$, and the running time of Algorithm 6 is $O(n^2 \log n)$, like in case of interval uncertainty; see Theorems 8 and 29, Corollary 14 and Theorem 30.

Observe that $g_f^{b_f^*}$ is defined for all $b_f^* \in \{b_f^-, \dots, b_f^+ - 1\}$ and that the function g_f is continuous, although both is not directly obvious from Algorithm 6: The alternative way of describing the function g_f, as a pointwise maximum of continuous piecewise linear functions for all (exponentially many) scenarios in $\mathcal{U}$ (see the beginning of Section 6.2), demonstrates that this is true. In fact, similarly to Algorithm 4, Algorithm 6 provides a way to determine g_f by computing only a polynomial number of linear pieces that might contribute to the pointwise maximum.

▶ Remark 33. The approach we developed for discrete uncorrelated uncertainty is similar to the one for interval uncertainty, but there are some differences we need to point out. As mentioned above, the heads $\bar{e}$ enumerated in Algorithms 3 and 4 are not necessarily the fractional items in the resulting follower's solutions; they only provide useful reference points in order to find all relevant fractional prefixes. Accordingly, we solve a Continuous Selection Problem in a subroutine there and determine the fractional item during that. In contrast, we fix the fractional item e^* in Algorithms 5 and 6, and solve a binary Selection Problem in a subroutine, in order to determine the remaining items of the fractional prefix. In fact, we could use the latter approach also for solving the Robust Continuous Bilevel Selection Problem with interval uncertainty in polynomial time, by enumerating the possible fractional items together with a reasonable finite set of possible cost values the adversary may assign to them (depending on the endpoints of the other intervals). However, this would not give any better results than Theorems 29 and 30.

In the corresponding versions of the Robust Bilevel Continuous Knapsack Problem, the difference between the two approaches is more significant: The Continuous Knapsack Problem can be solved in polynomial time, while the binary Knapsack Problem cannot. Therefore, the Robust Bilevel Continuous Knapsack Problem with interval uncertainty can be solved in polynomial time only by applying the strategy that solves a Continuous Knapsack Problem in each iteration; see [6]. In case of discrete uncorrelated uncertainty, this polynomial-time approach does not work, analogously to the discussion in the beginning of Section 6.2.3. In fact, the Robust Bilevel Continuous Knapsack Problem has been shown to be weakly NP-hard in [6]. However, the new approach we developed in this section can be generalized to a pseudopolynomial-time algorithm for the Robust Bilevel Continuous Knapsack Problem; see Appendix A.

▶ Remark 34. In Sections 4.2 and 6.2.2, for interval uncertainty, we assumed that there are no one-point intersections between the intervals in the uncertainty set, in order to avoid discussing technical details related to the optimistic and the pessimistic case. For discrete uncorrelated uncertainty, one could assume that the sets $\mathcal{U}_e$ are pairwise disjoint. However, analogously to interval uncertainty, Algorithms 5 and 6, as they are stated above, are also correct for the pessimistic setting without this assumption, while an adjusted definition of the item sets $\mathcal{E}^-_{e^*,\delta^*}$ and $\mathcal{E}^0_{e^*,\delta^*}$ is required for the optimistic setting, similarly to Remark 2 in [6].

7 Conclusion

We have investigated the complexity of robust bilevel optimization at the example of the polynomial-time solvable Bilevel Selection Problem, for several problem variants. With this, we add to the still relatively limited amount of literature on bilevel optimization under uncertainty, in particular regarding its complexity.

Firstly, our results generalize and extend the structural and algorithmic results of [6] to a related setting. Here, we place a stronger focus on combinatorial underlying problems, using the fundamental Selection Problem as a basis, which is often used to investigate the complexity of robust optimization concepts. In Section 6.2, we generalized the approach of viewing the leader's objective function as a piecewise linear function, which has also been used in [6].

In relation to [6], we emphasize the following insights regarding the complexity of the Robust Bilevel Continuous Knapsack Problem and the Robust Continuous Bilevel Selection Problem, which is simpler in terms of the items not having different sizes. In case of discrete uncorrelated uncertainty, the leader's as well as the adversary's problem have been shown to be NP-hard in [6], while the corresponding Robust Continuous Bilevel Selection Problem is still solvable in polynomial time. In Appendix A, we generalize our new approach to a pseudopolynomial-time algorithm for the Robust Bilevel Continuous Knapsack Problem and thus showed that the problem with discrete uncorrelated uncertainty is in fact only weakly NP-hard, but not strongly NP-hard, answering an open question stated in [6].

Furthermore, we complemented the algorithmic results that are related to [6] with a complexity study of the more general (Robust) Bilevel Selection Problem in which the leader's and the follower's item sets are not necessarily disjoint. We have shown that the Robust Bilevel Selection Problem is strongly NP-hard in general, but can be 2-approximated in polynomial time. Moreover, we investigated exponential-time solution approaches for this problem, including an algorithm that runs in polynomial time if the number of scenarios in the uncertainty set is a constant.

Another interesting observation is that the variant of the Bilevel Selection Problem with continuous leader's variables and binary follower's variables yields a bilevel optimization problem whose optimal value may not be attained (see Example 25). This demonstrates that, in general, bilevel optimization problems containing integer and continuous variables should be handled with care in this respect. It could be interesting to study this problem variant further, e.g., in view of computing the optimal value or computing feasible solutions that approach the optimal value.

It remains a task for future work to investigate whether the general Robust Bilevel Selection Problem is also NP-hard for other types of uncertainty, as it is for discrete uncertainty. Also in case of disjoint item sets, it could be interesting to study other types of uncertainty sets, e.g., budgeted, polytopal, or ellipsoidal uncertainty, like in [6]. If the Robust Bilevel Selection Problem could be solved in polynomial time here, then it would be again easier than the Robust Bilevel Continuous Knapsack Problem, like for discrete uncorrelated uncertainty, and polynomial-time algorithms could again be helpful to develop pseudopolynomial-time algorithms for the Robust Bilevel Continuous Knapsack Problem in the corresponding cases.

We have defined the Selection Problem and the Bilevel Selection Problem as minimization problems in this article. In relation to the knapsack problem of [6], one could also formulate them as maximization problems. For most of our results, this does not make a difference, but the proof of the approximation result in Theorem 16 is only valid for the minimization version. This suggests the open question whether a similar result can also be obtained for the corresponding maximization problem. Concerning Theorem 16, one could also investigate if a better approximation factor can be achieved by another algorithm or if any lower bound for a possible approximation guarantee can be obtained. Note that the reduction in the proof of Theorem 15 does not imply any statement about the approximability of the problem.

In Section 6, we mostly focused on the problem variant with disjoint item sets. One could also study the general Robust Continuous Bilevel Selection Problem with arbitraty item sets. However, the structural insights regarding piecewise linear functions do not seem to be easily applicable to this setting, and we conjecture that it becomes significantly harder, similarly to the binary case.

As a generalization of the combinatorial (Robust) Bilevel Selection Problem, also the corresponding Bilevel Knapsack Problem could be interesting to consider further in view of its complexity. While the Bilevel Knapsack Problem with disjoint item sets has been proved to be weakly Σ_2^{p}-hard in [11], it is open whether it is hard in the weak or in the strong sense in the more general case of arbitrary item sets. In fact, this problem could be related to the bilevel knapsack problem of [18] that has been shown to be strongly NP-hard in [11]. The Bilevel Knapsack Problem could also be a good starting point to investigate the additional complexity that robustness introduces in a harder underlying bilevel optimization problem because its structure is still relatively easy to understand.

Regarding the Robust Bilevel Continuous Knapsack Problem, the NP-hardness results obtained in [6] for some types of uncertainty sets, in particular for discrete uncorrelated uncertainty, all concern both the leader's and the adversary's problem. In view of multilevel problems and the polynomial hierarchy (see, e.g., [25]), it is possible that the robust leader's problems are actually one level harder than the adversary's problems and thus Σ_2^{p}-hard. Exploring whether this is the case could yield a better understanding of problems on the second level of the polynomial hierarchy and of robust bilevel optimization.

A A Pseudopolynomial-Time Algorithm for the Robust Bilevel Continuous Knapsack Problem

We now adapt the algorithms for the Robust Continuous Bilevel Selection Problem with discrete uncorrelated uncertainty that we presented in Section 6.2.3, in order to apply them to the related Robust Bilevel Continuous Knapsack Problem from [6]. As indicated in Remark 33, this will lead to the first pseudopolynomial-time algorithm for this problem, while its weak NP-hardness has been proved in Theorem 3 of [6]. Also the adversary's problem is weakly NP-hard (see Theorem 4 of [6]), and we give a pseudopolynomial-time algorithm for it as well.

Recall that, in the Robust Bilevel Continuous Knapsack Problem, the leader does not control any items, but directly sets the capacity for the follower's Continuous Knapsack Problem. Moreover, the items do not only have leader's and follower's costs (or values), but also different sizes, as usual in the context of knapsack problems. For more details and other (minor) differences between the Robust Continuous Bilevel

Selection Problem and the Robust Bilevel Continuous Knapsack Problem from [6], we refer to Section 2.2.

In the following, we state the Robust Bilevel Continuous Knapsack Problem again in the variant that has been studied in [6], while adjusting the notation to the one used in this article. We are given a finite set $\mathcal{E}$ of items, item sizes $a\colon \mathcal{E} \to \mathbb{Q}_{>0}$, leader's item values $c\colon \mathcal{E} \to \mathbb{Q}$, and capacity bounds $b^-, b^+ \in \mathbb{Q}$ with $0 \leq b^- \leq b^+ \leq a(\mathcal{E})$. The follower's item values $d\colon \mathcal{E} \to \mathbb{Q}_{>0}$ are subject to uncertainty, which here means that we are given a discrete uncorrelated uncertainty set $\mathcal{U} = \prod_{e\in\mathcal{E}} \mathcal{U}_e$ with finite sets $\mathcal{U}_e \subseteq \mathbb{Q}_{>0}$. The Robust Bilevel Continuous Knapsack Problem can now be formulated as follows:

$$\begin{aligned} \max_{b\in[b^-,b^+]}\ \min_{d\in\mathcal{U}}\ & \sum_{e\in\mathcal{E}} c(e)y_e \\ \text{s.t. } y \in \operatorname*{argmax}_{y'}\ & \sum_{e\in\mathcal{E}} d(e)y'_e \\ \text{s.t. } & \sum_{e\in\mathcal{E}} a(e)y'_e \leq b \\ & y' \in [0,1]^{\mathcal{E}} \end{aligned} \tag{RBCKP}$$

Observe that the follower can always be assumed to fill the capacity b completely here because the follower's item values are positive and $b \leq a(\mathcal{E})$. Therefore, the constraint $\sum_{e\in\mathcal{E}} a(e)y'_e \leq b$ could be equivalently replaced by $\sum_{e\in\mathcal{E}} a(e)y'_e = b$.

In view of the pseudopolynomial-time algorithms that we will discuss, we assume from now on that the item sizes $a\colon \mathcal{E} \to \mathbb{N}$ are all integers (which can be achieved by scaling the instance if necessary). The total size $a(\mathcal{E}) = \sum_{e\in\mathcal{E}} a(e)$ of all items will then appear in the running times of the algorithms.

The item sizes are the main complication of the Robust Bilevel Continuous Knapsack Problem compared to the Robust Continuous Bilevel Selection Problem, and we need to take care of them in several ways:

Firstly, the follower's greedy algorithm depends on the order of the item profits $d(e)/a(e)$ instead of only the item values $d(e)$. This is reflected in the definitions of the sets $\mathcal{E}^-_{e^*,\delta^*}$ and $\mathcal{E}^0_{e^*,\delta^*}$ in Algorithms 7 and 8, analogously to Algorithms 1 and 2 in [6] for the adversary's problem and the leader's problem of the Robust Bilevel Continuous Knapsack Problem with interval uncertainty.

Secondly, when fixing the fractional item of the follower's Continuous Knapsack Problem, the fraction with which it is selected is not uniquely determined by the capacity b, in contrast to the case where the follower solves a Continuous Selection Problem. If b is not an integer, in the latter case, the fraction is always given by $b - \lfloor b \rfloor$, and we know that b items are selected before the fractional item. We made use of this property in Algorithm 5. In the more general case of a Continuous Knapsack Problem, the items selected before the fractional item e^* are known to have an integer total size as well (as we assume all item sizes to be integers), but this can be any integer number b^* from $\lfloor b - a(e^*) \rfloor + 1$ to $\lfloor b \rfloor$ now, corresponding to fractions $y_{e^*} = \frac{1}{a(e^*)}(i + (b - \lfloor b \rfloor))$ for $i \in \{0, \ldots, a(e^*) - 1\}$ (in reverse order). Therefore, we enumerate all these possibilities for b^* in addition to the possible fractional items e^* and their follower's item values δ^* in Algorithm 7. Note that Line 10 of Algorithm 7 combines this with the capacity condition that is used also in previous versions of the algorithm; see, e.g., Line 11 of Algorithm 5. When solving the leader's problem, we have to enumerate all reasonable integer capacities b^* anyway (see Algorithm 6) and do this in a similar way also in Algorithm 8.

For fixed e^*, δ^*, and b^*, the remaining task is now to solve a binary Knapsack Problem instead of a binary Selection Problem as in Algorithms 5 and 6. More precisely, the subroutine corresponds to a binary Knapsack Problem with an equality constraint because the items in the fractional prefix before the fractional item need to have a total size of exactly b^* according to the above considerations. If this problem is infeasible, then we skip the current value of b^*.

Formally, the problem that the subroutine solves can be stated as follows: Given a finite set $\mathcal{E}'$ of items, item sizes $a'\colon \mathcal{E}' \to \mathbb{N}$, a capacity $b' \in \{0, \ldots, a'(\mathcal{E}')\}$, and item values $c'\colon \mathcal{E}' \to \mathbb{Q}$, select a subset $Y \subseteq \mathcal{E}'$ such that $a'(Y) = b'$ and $c'(Y)$ is maximized, or decide that there is no such Y. We write Knapsack Problem$(\mathcal{E}', a', b', c')$ when we call this subroutine and assume that it returns an optimal solution Y if there is one, or the string "infeasible" otherwise. The subroutine can be implemented to run in time $O(|\mathcal{E}'|b')$ using dynamic programming; see, e.g., [27, 30].

We first adapt Algorithm 5 to the adversary's problem of the Robust Bilevel Continuous Knapsack Problem. Respecting all aspects explained above, we obtain Algorithm 7.

Algorithm 7: Algorithm for the adversary's problem of the Robust Bilevel Continuous Knapsack Problem with discrete uncorrelated uncertainty

---

Input : a finite set $\mathcal{E}$, $a\colon \mathcal{E} \to \mathbb{N}$, $c\colon \mathcal{E} \to \mathbb{Q}$, $b^-, b^+ \in \mathbb{Q}$ with $0 \le b^- \le b^+ \le a(\mathcal{E})$, a discrete uncorrelated uncertainty set $\mathcal{U}$ given by a finite set $\mathcal{U}_e \subseteq \mathbb{Q}_{>0}$ for every $e \in \mathcal{E}$, a feasible leader's solution $b \in [b^-, b^+]$

Output : an optimal adversary's solution $d \in \mathcal{U}$ (i.e., a value $d(e) \in \mathcal{U}_e$ for every $e \in \mathcal{E}$) of the Robust Bilevel Continuous Knapsack Problem

1 **if** $b = 0$ *or* $b = a(\mathcal{E})$ **then**
2   **return** *an arbitrary* $d \in \mathcal{U}$
3 **for** $e \in \mathcal{E}$ **do**
4     $d^-(e) := \min \mathcal{U}_e$, $d^+(e) := \max \mathcal{U}_e$
5 $\bar{\mathcal{E}} := \emptyset$
6 **for** $e^* \in \mathcal{E}$ **do**
7   **for** $\delta^* \in \mathcal{U}_{e^*}$ **do**
8         $\mathcal{E}^-_{e^*,\delta^*} := \{e \in \mathcal{E} \mid d^-(e)/a(e) > \delta^*/a(e^*)\}$
9         $\mathcal{E}^0_{e^*,\delta^*} := \{e \in \mathcal{E} \mid d^-(e)/a(e) \le \delta^*/a(e^*) \le d^+(e)/a(e)\} \setminus \{e^*\}$
10   **for** $b^* \in \{\max\{a(\mathcal{E}^-_{e^*,\delta^*}), \lfloor b - a(e^*) \rfloor + 1\}, \dots, \min\{a(\mathcal{E}^-_{e^*,\delta^*} \cup \mathcal{E}^0_{e^*,\delta^*}), \lfloor b \rfloor\}\}$ **do**
11             $Y^0_{e^*,\delta^*,b^*} :=$ Knapsack Problem$(\mathcal{E}^0_{e^*,\delta^*}, a \restriction \mathcal{E}^0_{e^*,\delta^*}, b^* - a(\mathcal{E}^-_{e^*,\delta^*}), -c \restriction \mathcal{E}^0_{e^*,\delta^*})$
12   **if** $Y^0_{e^*,\delta^*,b^*} \neq$ *"infeasible"* **then**
13                 $Y_{e^*,\delta^*,b^*} := \mathcal{E}^-_{e^*,\delta^*} \cup Y^0_{e^*,\delta^*,b^*}$
14                 $\bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{(e^*, \delta^*, b^*)\}$
15 select $(e^*, \delta^*, b^*) \in \operatorname{argmin}\{c(Y_{e,\delta,\bar{b}}) + c(e)/a(e) \cdot (b - \bar{b}) \mid (e, \delta, \bar{b}) \in \bar{\mathcal{E}}\}$ arbitrarily
16 **return** $d\colon \mathcal{E} \to \mathbb{Q}$ *with* $d(e^*) := \delta^*$, $d(e) = d^+(e)$ *for all* $e \in Y_{e^*,\delta^*,b^*}$ *and* $d(e) := d^-(e)$ *for all* $e \in \mathcal{E} \setminus (Y_{e^*,\delta^*,b^*} \cup \{e^*\})$

Theorem 31, together with the insights explained above, implies the correctness of Algorithm 7. For the running time, note that Lines 10 to 14 can be implemented using a single dynamic program because a Knapsack Problem on the same item set is solved for several capacities. This shows:

▶ **Theorem 35.** *For any fixed feasible leader's solution of the Robust Bilevel Continuous Knapsack Problem with a discrete uncorrelated uncertainty set* $\mathcal{U} = \prod_{e \in \mathcal{E}} \mathcal{U}_e$*, Algorithm 7 solves the adversary's problem in time* $O(nuA)$*, where* $n = |\mathcal{E}|$*,* $u = \sum_{e \in \mathcal{E}} |\mathcal{U}_e|$*, and* $A = a(\mathcal{E})$*.*

We now turn to the leader's algorithm and adapt Algorithms 6 and 7 to the leader's problem of the Robust Bilevel Continuous Knapsack Problem. This results in Algorithm 8.

In Algorithm 8, analogously to Algorithm 7, we enumerate all possible fractional items $e^* \in \mathcal{E}$, the values $\delta^* \in \mathcal{U}_{e^*}$ assigned to them by the adversary, and all possible total sizes b^* of the items the follower selects before the fractional item e^*. The feasible values for b^* can be derived from the total sizes of the sets $\mathcal{E}^-_{e^*,\delta^*}$ and $\mathcal{E}^0_{e^*,\delta^*}$, like in Algorithm 7, but without any fixed capacity b, as the leader's decision is not fixed yet. If the subroutine returns a feasible solution, we construct a linear piece corresponding to the leader's values of the follower's solutions consisting of the binary prefix and any fraction $y_{e^*} \in [0, 1]$ of item e^*, similarly to Algorithm 6. Note that the width of the linear piece is now given by the size $a(e^*)$, in contrast to the setting of Algorithm 6 where all items have size 1. Accordingly, we cannot compute the pointwise minimum for every b^* separately like in Algorithm 6, but we collect all linear pieces and compute the pointwise minimum globally in Line 14 of Algorithm 8.

Using the same arguments as in [6] and Section 6.2.3, one can see that the function g computed by Algorithm 8 is exactly the leader's objective function for the Robust Bilevel Continuous Knapsack Problem. The algorithm's running time can be estimated similarly to Theorems 32 and 35. Note that the term $a(\mathcal{E})$ enters the running time due to the subroutine solving a binary Knapsack Problem as well as due to the enumeration of b^*. Together, we get:

Algorithm 8: Algorithm for the leader's problem of the ROBUST BILEVEL CONTINUOUS KNAPSACK PROBLEM with discrete uncorrelated uncertainty

Input : a finite set $\mathcal{E}$, $a\colon \mathcal{E} \to \mathbb{N}$, $c\colon \mathcal{E} \to \mathbb{Q}$, $b^-, b^+ \in \mathbb{Q}$ with $0 \le b^- \le b^+ \le a(\mathcal{E})$, a discrete uncorrelated uncertainty set $\mathcal{U}$ given by a finite set $\mathcal{U}_e \subseteq \mathbb{Q}_{>0}$ for every $e \in \mathcal{E}$

Output : an optimal leader's solution $b \in [b^-, b^+]$ of the ROBUST BILEVEL CONTINUOUS KNAPSACK PROBLEM

for $e \in \mathcal{E}$ **do**
  $d^-(e) := \min \mathcal{U}_e$, $d^+(e) := \max \mathcal{U}_e$
$\bar{\mathcal{E}} := \emptyset$
for $e^* \in \mathcal{E}$ **do**
  for $\delta^* \in \mathcal{U}_{e^*}$ **do**
    $\mathcal{E}^-_{e^*,\delta^*} := \{e \in \mathcal{E} \mid d^-(e)/a(e) > \delta^*/a(e^*)\}$
    $\mathcal{E}^0_{e^*,\delta^*} := \{e \in \mathcal{E} \mid d^-(e)/a(e) \le \delta^*/a(e^*) \le d^+(e)/a(e)\} \setminus \{e^*\}$
    for $b^* \in \{a(\mathcal{E}^-_{e^*,\delta^*}), \dots, a(\mathcal{E}^-_{e^*,\delta^*} \cup \mathcal{E}^0_{e^*,\delta^*})\}$ **do**
      $Y^0_{e^*,\delta^*,b^*} :=$ KNAPSACK PROBLEM$(\mathcal{E}^0_{e^*,\delta^*}, a \restriction \mathcal{E}^0_{e^*,\delta^*}, b^* - a(\mathcal{E}^-_{e^*,\delta^*}), -c \restriction \mathcal{E}^0_{e^*,\delta^*})$
      if $Y^0_{e^*,\delta^*,b^*} \ne$ *"infeasible"* **then**
        $Y_{e^*,\delta^*,b^*} := \mathcal{E}^-_{e^*,\delta^*} \cup Y^0_{e^*,\delta^*,b^*}$
        $g_{e^*,\delta^*,b^*} :=$ linear piece from $(b^*, c(Y_{e^*,\delta^*,b^*}))$ to $(b^* + a(e^*), c(Y_{e^*,\delta^*,b^*}) + c(e^*))$
        $\bar{\mathcal{E}} := \bar{\mathcal{E}} \cup \{(e^*, \delta^*, b^*)\}$
compute $g := \min\{g_{e^*,\delta^*,b^*} \mid (e^*, \delta^*, b^*) \in \bar{\mathcal{E}}\}$ (i.e., the pointwise minimum)
compute some $b \in \operatorname{argmax}\{g(b') \mid b' \in [b^-, b^+]\}$
return b

▶ **Theorem 36.** *Algorithm 8 solves the ROBUST BILEVEL CONTINUOUS KNAPSACK PROBLEM with a discrete uncorrelated uncertainty set* $\mathcal{U} = \prod_{e \in \mathcal{E}} \mathcal{U}_e$ *in time* $O(uA(n + \log(uA)))$*, where* $n = |\mathcal{E}|$*,* $u = \sum_{e \in \mathcal{E}} |\mathcal{U}_e|$*, and* $A = a(\mathcal{E})$*.*

In summary, we complemented the weak NP-hardness results from [6] by pseudopolynomial-time algorithms for both the adversary's problem and the leader's problem of the ROBUST BILEVEL CONTINUOUS KNAPSACK PROBLEM with discrete uncorrelated uncertainty.

References

1 Yasmine Beck, Ivana Ljubić, and Martin Schmidt. A survey on bilevel optimization under uncertainty. *Eur. J. Oper. Res.*, 311(2):401–426, 2023.
2 Yasmine Beck and Martin Schmidt. A Gentle and Incomplete Introduction to Bilevel Optimization. `https://optimization-online.org/?p=17182`, 2021.
3 Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust Optimization*. Number 28 in Princeton Series in Applied Mathematics. Princeton University Press, 2009.
4 Manuel Blum, Robert W. Floyd, Vaughan Pratt, Ronald L. Rivest, and Robert E. Tarjan. Time bounds for selection. *J. Comput. Syst. Sci.*, 7(4):448–461, 1973.
5 Gerald Brown, Matthew Carlyle, Javier Salmerón, and Kevin Wood. Defending critical infrastructure. *Interfaces*, 36(6):530–544, 2006.
6 Christoph Buchheim and Dorothee Henke. The robust bilevel continuous knapsack problem with uncertain coefficients in the follower's objective. *J. Glob. Optim.*, 83(4):803–824, 2022.
7 Christoph Buchheim, Dorothee Henke, and Felix Hommelsheim. On the complexity of robust bilevel optimization with uncertain follower's objective. *Oper. Res. Lett.*, 49(5):703–707, 2021.
8 Christoph Buchheim, Dorothee Henke, and Felix Hommelsheim. On the complexity of the bilevel minimum spanning tree problem. *Networks*, 80(3):338–355, 2022.
9 Christoph Buchheim, Dorothee Henke, and Jannik Irmai. The stochastic bilevel continuous knapsack problem with uncertain follower's objective. *J. Optim. Theory Appl.*, 194(2):521–542, 2022.
10 Christoph Buchheim and Jannis Kurtz. Robust combinatorial optimization under convex and discrete cost uncertainty. *EURO J. Comput. Optim.*, 6(3):211–238, 2018.

11 Alberto Caprara, Margarida Carvalho, Andrea Lodi, and Gerhard J. Woeginger. A study on the computational complexity of the bilevel knapsack problem. *SIAM J. Optim.*, 24(2):823–838, 2014.

12 Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Ann. Oper. Res.*, 153(1):235–256, 2007.

13 Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

14 George B. Dantzig. Discrete-variable extremum problems. *Oper. Res.*, 5(2):266–277, 1957.

15 Stephan Dempe. Bilevel Optimization: Theory, Algorithms, Applications and a Bibliography. In Stephan Dempe and Alain Zemkoho, editors, *Bilevel Optimization*, number 161 in Springer Optimization and Its Applications, pages 581–672. Springer, 2020.

16 Stephan Dempe, Vyacheslav Kalashnikov, Gerardo A. Pérez-Valdés, and Nataliya Kalashnykova. *Bilevel Programming Problems*. Springer, 2015.

17 Stephan Dempe and Katrin Richter. Bilevel programming with knapsack constraints. *Cent. Eur. J. Oper. Res.*, 8(2):93–107, 2000.

18 Scott DeNegre. *Interdiction and discrete bilevel linear programming*. PhD thesis, Lehigh University, Bethlehem, Pennsylvania, 2011.

19 Dennis Fischer, Komal Muluk, and Gerhard J. Woeginger. A note on the complexity of the bilevel bottleneck assignment problem. *4OR*, 20(4):713–718, 2022.

20 Michael R. Garey and David S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness*. A Series of Books in the Mathematical Sciences. W. H. Freeman and Company, 1979.

21 Elisabeth Gassner and Bettina Klinz. The computational complexity of bilevel assignment problems. *4OR*, 7(4):379–394, 2009.

22 Marc Goerigk, Jannis Kurtz, Martin Schmidt, and Johannes Thürauf. Connections between robust and bilevel optimization. *Open J. Math. Optim.*, 6: article no. 2 (17 pages), 2025.

23 Kristina Hartmann. The robust bilevel selection problem. Master's thesis, TU Dortmund University, 2021. unpublished.

24 John Hershberger. Finding the upper envelope of n line segments in $\mathcal{O}(n \log n)$ time. *Inf. Process. Lett.*, 33(4):169–174, 1989.

25 Robert G. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Math. Program.*, 32(2):146–164, 1985.

26 Adam Kasperski and Paweł Zieliński. A randomized algorithm for the min–max selecting items problem with uncertain weights. *Ann. Oper. Res.*, 172(1):221–230, 2009.

27 Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack Problems*. Springer, 2004.

28 Bernhard Korte and Jens Vygen. *Combinatorial Optimization*. Number 21 in Algorithms and Combinatorics. Springer, 6 edition, 2018.

29 Panos Kouvelis and Gang Yu. *Robust Discrete Optimization and Its Applications*. Number 14 in Nonconvex Optimization and Its Applications. Kluwer Academic Publishers, 1997.

30 Raïd Mansi, Cláudio Alves, José Manuel Valério de Carvalho, and Saïd Hanafi. An exact algorithm for bilevel 0-1 knapsack problems. *Math. Probl. Eng.*, 2012: article no. 504713 (23 pages), 2012.

31 Patrice Marcotte and Gilles Savard. Bilevel programming: A combinatorial perspective. In David Avis, Alain Hertz, and Odile Marcotte, editors, *Graph Theory and Combinatorial Optimization*, pages 191–217. Springer, 2005.

32 Rolf H. Möhring. Computationally tractable classes of ordered sets. In Ivan Rival, editor, *Algorithms and Order*, number 255 in NATO ASI Series. Series C. Mathematical and Physical Sciences, pages 105–193. Kluwer Academic Publishers, 1989.

33 Xueyu Shi, Bo Zeng, and Oleg A. Prokopyev. On bilevel minimum and bottleneck spanning tree problems. *Networks*, 74(3):251–273, 2019.

34 Luis Vicente, Gilles Savard, and Joaquim Júdice. Discrete linear bilevel programming problem. *J. Optim. Theory Appl.*, 89(3):597–614, 1996.

35 Gerhard J. Woeginger. On the approximability of average completion time scheduling under precedence constraints. *Discrete Appl. Math.*, 131(1):237–252, 2003.

36 Gerhard J. Woeginger. The trouble with the second quantifier. *4OR*, 19(2):157–181, 2021.