Linear and Mixed-Integer Bilevel Optimization: Theory and Algorithms

Yasmine Beck, Ivana Ljubić, Martin Schmidt

Author Information

Yasmine Beck
Eindhoven University of Technology
Department of Industrial Engineering and Innovation Sciences
PO Box 513, 5600 MB Eindhoven, the Netherlands
Email address: y.beck@tue.nl

Ivana Ljubić
Vienna University of Economics and Business
Institute for Statistics and Mathematics
Welthandelpatz 1, 1020 Vienna, Austria
Email address: ivana.ljubic@wu.ac.at

Martin Schmidt
Trier University
Department of Mathematics
Universitätsring 15, 54296 Trier, Germany
Email address: martin.schmidt@uni-trier.de

Contents

Preface *page* 8

PART I WHAT IS BILEVEL OPTIMIZATION ABOUT? 11

1 Introduction 13
1.1 What is a Bilevel Problem and Why Should You Care? 13
1.2 A Bit More Formal, Please 16
1.3 A Brief History of Bilevel Optimization 30
1.4 What You Should Know Now! 31

2 Solution Concepts: Optimistic vs. Pessimistic Problems 33
2.1 What You Should Know Now! 39

PART II CONVEX LOWER-LEVEL PROBLEMS 41

3 Single-Level Reformulations 43
3.1 The Optimal-Value-Function Reformulation 44
3.2 The KKT Reformulation for LP-LP Bilevel Problems 45
3.3 The Strong-Duality Based Reformulation 46
3.4 A Mixed-Integer Linear Reformulation of the KKT Reformulation 49
3.5 Excursus: How to Solve a Mixed-Integer Linear Problem? 52
3.6 The KKT Reformulation for Convex Lower-Level Problems 55
3.7 Independence of Irrelevant Constraints 63
3.8 What You Should Know Now! 68

4 Linear Bilevel Problems 70
4.1 Unboundedness of the Shared Constraint Set 72
4.2 Existence of Solutions 75
4.3 Geometric Properties 77
4.4 Complexity Results 80
4.5 What You Should Know Now! 83

5 Algorithms for Linear Bilevel Problems 84
5.1 The *K*th-Best Algorithm 84
5.2 Complementarity-Based Branch-and-Bound 89
5.3 A Penalty Alternating Direction Method 104
5.4 What You Should Know Now! 113

PART III MIXED-INTEGER BILEVEL PROBLEMS 115

6 General Properties 117
6.1 Central Definitions 120
6.2 A Glimpse at the Single-Level Relaxation 121
6.3 Attainability of Optimal Solutions 124
6.4 Complexity Results 128
6.5 Connections Between Bilevel MILPs and Other Problem Classes 129
6.6 What You Should Know Now! 132

7 Excursus: Branch-and-Bound for Single-Level MILPs 134
7.1 What You Should Know Now! 137

8 Branch-and-Bound for Bilevel MILPs 138
8.1 About Relaxations and Restrictions 139
8.2 Why Relaxing Integrality Constraints is not Enough 144
8.3 A Branch-and-Bound Method for Bilevel MILPs 148
8.4 Illustrative Examples 153
8.5 What You Should Know Now! 160

9 Branch-and-Cut for Bilevel MILPs 161
9.1 Valid Inequalities 162
9.2 A Branch-and-Cut Method for Bilevel MILPs 170
9.3 Excursus: Bilevel Solvers You Can Try 181
9.4 What You Should Know Now! 182

10 Interdiction Problems 183
10.1 Different Types of Interdiction Problems 185
10.2 Frequently Studied Interdiction Problems 187

10.3 Solution Approaches for Linear Lower-Level Problems 196
10.4 Solution Approaches for Discrete Lower-Level Problems 201
10.5 Benders-Like Decomposition 203
10.6 The Downward Monotonicity Property 212
10.7 Blocker Problems 219
10.8 Critical Vertex or Edge Detection Problems in Networks 223
10.9 What You Should Know Now! 223

11 Excursus: Intersection Cuts for Single-Level MILPs 225
11.1 The LP Cone Associated with a Basic Solution 226
11.2 Intersection Cuts 228
11.3 Computing Cut Coefficients for the Polyhedral Case 231
11.4 Using Disjunctive Arguments to Derive Intersection Cuts 233
11.5 What You Should Know Now! 236

12 Intersection Cuts for Bilevel MILPs 238
12.1 Intersection Cuts for MILP-MILP Bilevel Problems 239
12.2 Bilevel-Free Sets 241
12.3 How to Enlarge Bilevel-Free Sets 248
12.4 Separation of Intersection Cuts 249
12.5 Illustrative Examples 254
12.6 What You Should Know Now! 260

13 Primal Heuristics 262
13.1 Follower-Priority Heuristic 262
13.2 Penalize-and-Relax-and-Dualize Heuristic 265
13.3 What You Should Know Now! 272

PART IV FURTHER TOPICS 273

14 Bilevel Optimization Under Uncertainty 275
14.1 Motivation 275
14.2 Data Uncertainty 277
14.3 Decision Uncertainty 284

15 Bilevel Problems with Multiple Leaders or Followers 289
15.1 (Generalized) Nash Games and Nash Equilibria 290
15.2 Problem Statements 292
15.3 A Single-Level Reformulation of Linear SLMF Games 295

16 Mind the Gaps 307

PART V SUPPLEMENTARY MATERIAL 311
Appendix A **Optimality Conditions** 313
Appendix B **Polyhedra: Basic Notions and Results** 325
List of Figures 327
List of Algorithms 329
Bibliography 330
Index 349

Preface

Welcome to bilevel optimization—a wonderful sub-field of mathematical optimization with many surprising properties, a lot of relevant and challenging applications, as well as a growing number of algorithms and elegant theoretical aspects.

Although there have been some research monographs and surveys on bilevel optimization, this book is—to the best of our knowledge—the first textbook on the topic that particularly focuses on linear and mixed-integer linear bilevel problems. We hope that it serves the purpose of introducing students, researchers, and practitioners to the field, even if they have not been in touch with bilevel optimization at all before. This book emerged from a lecture on bilevel optimization given by Martin Schmidt and Yasmine Beck at Trier University. The corresponding lecture notes served as the basis for this book. However, these notes have been extended quite significantly so that this book, in its present form, may serve as the basis for a lecture over one semester (mainly considering the field of bilevel problems with linear or nonlinear but convex lower-level problems) or even two semesters (where the second semester can then focus on mixed-integer linear bilevel problems). Our aim is to start from scratch and keep this book as self-contained as possible. Nevertheless, throughout the book, we assume that the reader has a solid knowledge of linear optimization (especially concerning duality theory) and nonlinear optimization (especially concerning first-order optimality conditions).

What is this book about? As we will discuss in more detail later, bilevel optimization is a rather young field that started to develop in the late 1970s and 1980s. However, it has gained significantly increased attention over the last 10 to 15 years. The goal of this book is to introduce both the basics of bilevel optimization and also to discuss the most modern techniques for solving challenging instances. To this end, the book includes the following content. First, the class of bilevel optimization problems is formally introduced and motivated using examples from different fields of application. Afterward, the main focus is on how to solve linear and mixed-integer linear bilevel optimization problems. We start by considering various single-level reformulations of bilevel optimization problems with linear or nonlinear but still convex follower problems, derive theorems of existence of solutions, and then discuss geometric properties of linear bilevel problems. Next, we derive hardness theory for linear bilevel problems and study different algorithms for solving them. Then, we move on to mixed-integer linear bilevel problems and discuss the main obstacles for deriving exact as well as effective solution methods. Afterward, tailored branch-and-bound methods for solving these problems are introduced, which are then extended to branch-and-cut algorithms using today's most powerful cutting planes in bilevel optimization. Here, we particularly focus on the important subclass of interdiction problems. For more general problems, we present the details of so-called intersection cuts. Finally, we have an excursus on very recent topics in bilevel optimization such as bilevel optimization under uncertainty and so-called multi-leader multi-follower problems to also give the motivated reader a pointer to these emerging research topics with many open theoretical questions and algorithmic challenges. We choose these topics of the excursus so that the resulting models stay in the class of linear and mixed-integer linear bilevel problems.

As an additional feature, we close almost every chapter of this book with a section called "What You Should Know Now!", which is mainly thought of as a service to students who take a course on bilevel optimization that is based on this book. These sections list questions about the topics of the different chapters. If you can answer all of these questions correctly, you should not be afraid of the final exam. Moreover, all chapters contain exercises that can be used to deepen the understanding of the content of the respective chapter. Although we strongly encourage you to try to solve them, their content is usually not essential for understanding the following sections or chapters. We do not include solutions to these exercises on purpose. You need to try to solve them on your own. In this way, you gain far more knowledge and understanding about the field than by just studying a given solution.

Before we close this preface with some important acknowledgments, let us also mention that this book does, of course, not provide a comprehensive treatment of the overall field of bilevel optimization. For instance, we do not discuss optimality conditions of bilevel problems and we also do not touch the area of nonlinear and nonconvex lower-level problems. Moreover, we do not discuss any application in detail. Hence, we particularly exclude the very active field of bilevel optimization applied to machine learning problems.

Of course, we have to thank many people who helped us in writing this book.

Thanks a lot Christof Brandstetter, Aloïs Duguet, Julius Hoffmann, Andreas Horländer, Thomas Kleinert, Henri Lefebvre, Anna Laura Pala, Fränk Plein, Simon Stevens, Johannes Thürauf, Alberto Torrejón, and Wenjin Yan for many discussions on the topics of this book and for proof-reading many chapters. Moreover, thanks a lot Ioana Molan, Fabio Furini, and Andreas Horländer for helping us in creating some of the illustrative figures. We also want to thank Stephan Dempe for a private communication about the history of bilevel optimization and for some very useful feedback about an earlier version of this book.

Finally, we acknowledge the use of DeepL and OpenAI's ChatGPT for partly editing and polishing the text and figures for spelling, grammar, and stylistic improvements.

Have fun!

Yasmine Beck, Ivana Ljubić, Martin Schmidt
Eindhoven, Paris, and Trier, 2026

PART ONE

WHAT IS BILEVEL OPTIMIZATION ABOUT?

1
Introduction

1.1 What is a Bilevel Problem and Why Should You Care?

Usual optimization problems are single-level problems, which can be written as

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq 0, \\ & h(x) = 0. \end{aligned}$$

This means that there is only one objective function $f : \mathbb{R}^n \to \mathbb{R}$, one vector of variables $x \in \mathbb{R}^n$, and one feasible set given by the constraint functions $g : \mathbb{R}^n \to \mathbb{R}^m$ and $h : \mathbb{R}^n \to \mathbb{R}^p$. In particular, this models a situation in which a single decision maker takes all decisions, i.e., decides on all variables of the problem. Studying such a problem is often appropriate—for instance, if a single dispatcher controls a gas transport network, if a single investment bank decides on the assets in a portfolio, or if a single logistics company decides on its supply chain.

However, there are many situations in our daily life that are different. As decision makers, we often choose our actions while anticipating the response of another decision maker whose choices depend on our own. At the same time, our own decision-making framework is influenced by how we expect the other party to react. Here and in what follows, we refer to a decision-making framework as an optimization problem. Throughout, we assume that all such responses are rational, meaning they are optimal with respect to each decision maker's own objective. Formalizing this situation leads to hierarchical or bilevel optimization problems. Before we formally define this class of problems, let us first consider some informally stated examples.

Example 1.1 (Pricing) One of the richest classes of applications of bilevel optimization are pricing problems. The first decision maker, which we also call *leader* in the following, decides on a price of a certain good (or maybe on different prices for multiple goods) to maximize her[1] revenue from selling these goods. The second decision maker, also called the *follower*, then decides on purchasing the goods of the leader to generate some utility.

Thus, the leader's decision depends on the optimal reaction of the follower, and the decision of the follower, of course, depends on the pricing decision(s) of the leader. △

Example 1.2 (Toll Setting) Consider a transportation network, e.g., a highway system, via which some drivers want to travel from their origin to their destination. Typically, the objective of these travelers is to arrive at their destination at minimum cost. In this situation, costs can, e.g., be travel times, tolls, or a combination of both.

Usually, there is a toll-setting agency that decides on the tolls imposed on certain parts of the highway system. This toll-setting agency may want to maximize the revenues obtained from imposing tolls, but objectives other than revenue maximization are possible as well. For instance, one can also consider congestion minimization or ecological aspects such as, e.g., avoiding the transport of hazardous materials through natural reserves or reducing the number of large trucks on small roads between villages. In all these settings, the toll-setting agency is the leader and the travelers are the followers. Again, the leader anticipates the optimal reaction, that is the origin-destination paths, of the followers, and the followers' decisions depend on the tolls set by the leader. Whereas we only have one follower in the pricing example (Example 1.1), we now have multiple followers. The former is called a *single-leader single-follower* problem, and the latter is called a *single-leader multi-follower* problem. △

Exercise 1.3 (Toll Setting—Revisited) Consider the transportation network given by the directed graph in Figure 1.1. Here, t_e and γ_e denote the toll spent and the time needed to travel along edge $e \in E = \{a, b, c, d\}$, respectively. A traffic authority now wants to maximize revenues by imposing tolls, while a traveler wants to travel from the origin O to the destination $\mathcal{D}$ at minimum cost. In this example, route costs are the sum of the travel times and the imposed tolls.

The blue solution is worse for the toll-setting agency as well as for the traveler, i.e., for the leader and for the follower: The leader earns less and the follower has larger route costs compared to the "better" orange solution. Does this always need to be the case? Are there settings in which a solution is better for the leader but worse for the follower (or vice versa)?

Example 1.4 (Energy Markets) Another very rich class of applications of

[1] Throughout this book, we use "her" for the leader and "his" for the follower.

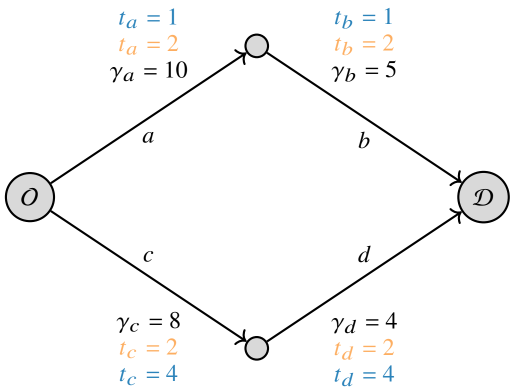


Figure 1.1 The transportation network considered in Exercise 1.3.

bilevel optimization can be found in the energy sector—especially in the sub-field of energy market modeling. In many countries of the world, e.g., in Germany, electricity is traded via auctions at an energy exchange.[2] The rules that determine how this auction is organized are typically imposed by the state government or some regulatory authority. Under these rules, producers and consumers trade electricity at the exchange and thus determine the market outcome and the aim of the mentioned rules is usually to obtain market outcomes that are optimal in terms of social welfare. For some more economic background, we refer, e.g., to Mas-Colell et al. (1995).

As before, the decision of the leader—here, the regulatory authority—depends on the anticipation of the followers' decisions—here, the decisions of the firms trading at the market. Moreover, the firms' decisions depend on the market regime, i.e., on the decision of the leader. △

Example 1.5 (Critical Infrastructure Defense) Bilevel optimization is also of great importance for critical infrastructure defense. Imagine a set of important locations such as airports, central stations, market squares, etc. that may be potential targets of terrorist attacks. This infrastructure needs to be protected, e.g., by police officers. However, there are not enough officers so that every location can be protected. Terrorists then decide to attack one or some of these locations based on their expectations about which locations are protected and which are not. Assuming some utility function for both the police (also called

[2] See, e.g., https://www.eex.com/en/ for the European Energy Exchange in Leipzig, Germany.

defenders in this setting) and the terrorists[3] (also called attackers in this setting), the police (as the leader) assigns officers to certain locations to achieve the worst outcome for the terrorists (acting as followers). △

Example 1.6 (Interdiction Problems) In discrete bilevel optimization, maybe the most heavily studied class of problems are interdiction problems. Here, the leader is a defender that interdicts certain resources of the follower so that they cannot be used by the follower anymore. Many of these problems are defined on graphs. For instance, the follower may want to find a shortest path in a graph from an origin to a destination. The leader, acting as the interdictor, can interdict some of the arcs in the graph so that they cannot be part of a feasible path of the follower anymore. The number of interdicted arcs may further be constrained by an interdiction budget of the leader. △

We now consider a particular example of an interdiction problem: the maximum-flow interdiction problem.

Example 1.7 (Maximum-Flow Interdiction Problem) Assume that the follower solves a maximum-flow problem on a given directed graph and for a given pair of an origin and a destination node. In addition, the leader can interdict a certain number of arcs in the graph so that they cannot be used in the maximum-flow problem of the follower anymore. The interdiction budget in this situation is the number of arcs that can be interdicted. The leader's goal is to interdict arcs so that the follower achieves the smallest possible maximum-flow value.

One classic real-world application for this modeling is the fight against drug smuggling. Here, the graph models the network of drug smuggling routes in which the smugglers want to maximize the flow of drugs from a certain origin to a certain destination. The leader (being the police, for instance) wants to interdict some parts of these routes so that the amount of smuggled drugs is as small as possible. △

The broad range of examples from security, counter-terrorism, drug smuggling, energy markets, revenue management, and transport highlights the importance of formally modeling, analyzing, and solving the respective mathematical models.

1.2 A Bit More Formal, Please

Now that we are convinced that it makes sense to study hierarchical optimization problems because of their practical relevance, let us formally define them.

[3] . . . although this may sound cynical.

Definition 1.8 (Optimistic Bilevel Optimization Problem) An *optimistic bilevel optimization problem* is given by

$$\begin{aligned} \min_{x \in X, y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{1.1}$$

where $\mathcal{S}(x)$ is the set of globally[4] optimal solutions to the x-parameterized problem

$$\begin{aligned} \min_{y} \quad & f(x, y) \\ \text{s.t.} \quad & y \in \mathcal{Y}(x) := \{y \in Y \colon g(x, y) \geq 0\}\,. \end{aligned} \tag{1.2}$$

Problem (1.1) is the so-called *upper-level* (or the *leader's*) *problem* and Problem (1.2) is the so-called *lower-level* (or the *follower's*) *problem*, which is parameterized by the leader's decision x. Moreover, the variables $x \in \mathbb{R}^{n_x}$ are the upper-level variables (or leader's decisions) and $y \in \mathbb{R}^{n_y}$ are the lower-level variables (or follower's decisions). The objective functions are given by $F, f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}$ and the constraint functions are given by $G : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}^m$ as well as $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}^\ell$. Here, we restrict ourselves to inequality constraints just for the ease of presentation. Of course, equality constraints can be added as well. The sets $X \subseteq \mathbb{R}^{n_x}$ and $Y \subseteq \mathbb{R}^{n_y}$ can, e.g., be used to denote integrality constraints. For instance, $Y = \mathbb{Z}^{n_y}$ makes the lower-level problem an integer problem. The set-valued map $\mathcal{Y} : \mathbb{R}^{n_x} \to 2^{\mathbb{R}^{n_y}}$, where $2^{\mathbb{R}^{n_y}}$ is the power set of $\mathbb{R}^{n_y}$, is called the *follower's feasible set mapping*. In what follows, we call upper-level constraints $G_i(x, y) \geq 0$, $i \in \{1, \ldots, m\}$, *coupling constraints* if they explicitly depend on the lower-level variable vector y. Moreover, all upper-level variables that appear in the lower-level problem are called *linking variables* and the respective lower-level constraints in which they appear are called *linking constraints*. The set-valued map $\mathcal{S}$ is the *follower's optimal response mapping*. For a given leader's decision x, the set $\mathcal{S}(x)$ is also called the *rational reaction set* of the follower.

In Problem (1.1), we consider the so-called *optimistic approach* to bilevel optimization, which is expressed by jointly optimizing over the variables x and y in the upper level. This means that, whenever the set $\mathcal{S}(x)$ is not a singleton, i.e., the solution to Problem (1.2) is not unique, the leader can influence the follower to select a decision that favors her the most w.r.t. her own objective function. Nevertheless, other approaches to tackle the situation in which the

[4] In what follows, we omit the term "globally" and always refer to globally optimal solutions when talking about optimal ones, unless explicitly stated otherwise.

lower level does not have a unique solution are possible as well. This is the topic of Chapter 2, in which we further discuss the so-called *pessimistic approach* to bilevel optimization.

We use the nomenclature that the bilevel problem (1.1) is an "UL-LL problem" where UL and LL can be LP, QP, MILP, MIQP, etc. if the upper-/lower-level problem is a linear, quadratic, mixed-integer linear, mixed-integer quadratic, etc. problem in both the variables of the leader and the follower. If the concrete specification of both levels is not required, we also use a shorter nomenclature and say, e.g., that the problem is a bilevel LP, if both levels are LPs.

Before we move on, let us briefly define what we are going to understand as a feasible point and a solution to an optimistic bilevel problem. As usual, we define both locally and globally optimal solutions, which can be defined as for other optimization problems as well; see Appendix A.2 for the definitions for standard single-level optimization problems. For their definition in the bilevel context, we first define some other notions that we use throughout the book.

Definition 1.9 (Graph of the Solution-Set Mapping) The set

$$\operatorname{gph} \mathcal{S} := \{(x, y) \colon y \in \mathcal{S}(x)\}$$

is called the *graph of the solution-set mapping* $\mathcal{S}$.

Definition 1.10 (Shared Constraint Set) The set

$$\Omega := \{(x, y) \in X \times Y \colon G(x, y) \geq 0,\ g(x, y) \geq 0\}$$

is called the *shared constraint set*. Its projection onto the x-space is denoted by

$$\Omega_x := \{x \colon \exists y \text{ with } (x, y) \in \Omega\}\,.$$

With the last two definitions, we can now define feasible points of a bilevel problem.

Definition 1.11 (Bilevel-Feasible Set; Inducible Region) The set

$$\mathcal{F} := \{(x, y) \colon (x, y) \in \Omega,\ y \in \mathcal{S}(x)\} = \Omega \cap \operatorname{gph} \mathcal{S}$$

is called the *bilevel-feasible set* or *inducible region*.

Finally, we can now state the definition of locally and globally optimal solutions to the bilevel problem (1.1).

Definition 1.12 (Locally and Globally Optimal Solution) A feasible point (x^*, y^*) of the bilevel problem (1.1) is a *locally optimal solution* if there exists an $\varepsilon > 0$ such that

$$F(x, y) \geq F(x^*, y^*)$$

holds for all $(x, y) \in \mathcal{F}$ with

$$\|(x, y) - (x^*, y^*)\| < \varepsilon.$$

A locally optimal solution is called a *globally optimal solution* if $\varepsilon > 0$ can be chosen arbitrarily large, i.e., if $F(x, y) \geq F(x^*, y^*)$ holds for all $(x, y) \in \mathcal{F}$.

Here and in what follows, $\|\cdot\|$ denotes the Euclidean norm.

Remark 1.13 As it is stated in Definition 1.8, we consider what is also called the *conventional formulation* of the optimistic bilevel problem. In the literature, there is also the *original optimistic formulation*, which reads

$$\min_{x \in X} \quad \Phi(x) \quad \text{with} \quad \Phi(x) := \min_{y \in Y} \{F(x, y) \colon G(x, y) \geq 0,\ y \in \mathcal{S}(x)\}. \tag{1.3}$$

We point out that the conventional formulation (1.1) and the original formulation (1.3) are equivalent on the level of globally optimal solutions, where "equivalence" is to be understood in the following sense. If (x^*, y^*) is a globally optimal solution to Problem (1.1), x^* is also a globally optimal solution to Problem (1.3). Conversely, if x^* is a globally optimal solution to Problem (1.3), there exists $y^* \in \mathcal{S}(x^*)$ with $G(x^*, y^*) \geq 0$ such that (x^*, y^*) is a globally optimal solution to Problem (1.1). However, as shown in Section 5.5 in Dempe (2002), the two formulations are, in general, not equivalent on the level of locally optimal solutions. As the focus of this book is almost always on solving bilevel problems to global optimality, we adopt the conventional formulation throughout this book. Nevertheless, we refer to Dempe et al. (2012) and Section 5.5 in Dempe (2002) for further discussions on the relation between the conventional and the original optimistic bilevel formulation.

Instead of using the point-to-set mapping $\mathcal{S}$ as in Problem (1.1), one can also use the so-called *optimal-value function*

$$\varphi(x) := \inf_{y \in Y} \{f(x, y) \colon g(x, y) \geq 0\}$$

of the lower-level problem and re-write Problem (1.1) as

$$\begin{aligned} \min_{x \in X, y \in Y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0,\ g(x, y) \geq 0, \\ & f(x, y) \leq \varphi(x), \end{aligned}$$

to which we refer as the optimal-value-function or value-function reformulation. Note that we use "inf" instead of "min" in the above definition of the optimal-value function as we have not made any assumptions to ensure that the infimum is actually attained, i.e., that the lower-lever problem is solvable. Later, when such assumptions are in place, we mainly use "min" when defining φ.

Exercise 1.14 Consider the linear bilevel problem

$$\begin{aligned}\min_{x,y} \quad & x + y \\ \text{s.t.} \quad & -x - 2y \geq -10, \\ & 2x - y \geq 0, \\ & -x + 2y \geq 0, \\ & x \geq 0, \\ & y \in \arg\min_{\bar{y}} \{\bar{y} : x + \bar{y} \geq 3\}.\end{aligned} \tag{1.4}$$

(i) Plot the shared constraint set of Problem (1.4) in a coordinate system.
(ii) Determine the bilevel-feasible set of Problem (1.4).
(iii) Use the optimal-value function to reformulate Problem (1.4) as a single-level problem.
(iv) Determine the optimal solution to the value-function reformulation derived in (iii) in which the lower-level constraint $x + y \geq 3$ is omitted. Is this solution bilevel feasible for the original problem (1.4)?

Definition 1.15 (Single-Level Relaxation) The problem of minimizing the upper-level objective function over the shared constraint set, i.e.,

$$\begin{aligned}\min_{x,y} \quad & F(x, y) \\ \text{s.t.} \quad & (x, y) \in \Omega,\end{aligned}$$

is called *single-level relaxation* of Problem (1.1).

Note that the single-level relaxation is identical to the original bilevel problem (1.1) except for the constraint $y \in \mathcal{S}(x)$, i.e., except for the lower-level optimality condition. Thus, it is indeed a relaxation of (1.1). In most of the literature on bilevel optimization, what we call single-level relaxation here is usually called high-point relaxation. The term "high-point relaxation" is mainly motivated by some examples in the early bilevel optimization literature; see, e.g., Example 1 and Figure 1 in Moore and Bard (1990). In the latter paper, the objective function of the high-point relaxation points "upwards" in the x-y coordinate system of the problem, leading to a solution that is a point being higher (in this coordinate system) than the bilevel-feasible points. Of course, this property depends on the specifically chosen example. In Moore and Bard (1990) and Bard and Moore (1990), the authors introduce this wording by referring to the earlier paper by Bialas and Karwan (1984). However, in the latter paper, the wording "high-point relaxation" never appears. Hence, to the best of our knowledge, there seems to be no paper in which this wording has been formally introduced. Because the name "high-point relaxation" indeed only makes sense for specifically chosen problems, we prefer to use a different wording. In optimization, relaxations are typically called after what you obtain if a certain set of constraints is omitted. This is, e.g., the case for the LP or continuous relaxation in mixed-integer linear optimization. Because omitting the lower-level optimality condition $y \in \mathcal{S}(x)$ leads to a single-level problem, we call the problem in Definition 1.15 the single-level relaxation, which is, e.g., also the wording used by Shi et al. (2023).

Next, we revisit some of the previously discussed examples and formally state the corresponding bilevel problems.

Example 1.16 (Pricing—Revisited) A first bilevel pricing problem with linear constraints and bilinear upper- and lower-level objectives has been proposed by Bialas and Karwan (1984). The following problem considered in Labbé et al. (1998) provides a general framework for such pricing problems:[5]

$$\begin{aligned} \max_{x, y=(y_1,y_2)} \quad & x^\top y_1 \\ \text{s.t.} \quad & Ax \leq a, \\ & y \in \arg\min_{\bar{y}} \left\{ (x + d_1)^\top \bar{y}_1 + d_2^\top \bar{y}_2 \colon D_1 \bar{y}_1 + D_2 \bar{y}_2 \geq b \right\}. \end{aligned}$$

The lower-level variable vector y is partitioned into two sub-vectors y_1 and y_2, called plans, that specify the levels of some activities such as the amount of purchased goods or services. The upper-level player influences the activities from plan y_1 through the price vector x that is imposed onto y_1 in addition to the given disutility d_1. By doing so, the goal of the leader is to maximize her revenue given by $x^\top y_1$. The price vector x is subject to linear constraints that may, among others, impose lower and upper bounds on the prices. The vectors d_1 and d_2 represent linear disutilities faced by the lower-level player when executing the activity plans y_1 and y_2. Note that d_2 includes the price (or disutility) for executing the activities not influenced by the upper-level player. These activities may, e.g., be substitutes offered by competitors of the leader for which prices are known and fixed. The lower-level player determines his activity plans y_1 and y_2 to minimize the sum of the total disutility and the price paid for plan y_1 subject to linear constraints. If the model allows negative prices, it

[5] You see that we are a bit sloppy when it comes to transposition of vectors. Formally, we should write $y = (y_1^\top, y_2^\top)^\top$, which is a bit cumbersome. Hence, we just use $y = (y_1, y_2)$ in what follows to improve readability.

implicitly permits subsidies, which may be appropriate, e.g., in the context of a central agency determining taxes. To avoid the situation in which the leader would maximize her profit by setting prices to infinity for those activities y_1 that are essential, one may assume that the set $\{y_2 \colon D_2 y_2 \geq b\}$ is non-empty. Indeed, in this case, there exists a feasible point for the lower level that does not use any activity influenced by the upper-level player. △

Example 1.17 (Toll Setting—Revisited) We now formalize the toll-setting problem of Example 1.2. To this end, we consider a directed and connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ with node set $\mathcal{V}$ and arc set $\mathcal{A}$. Every arc $a \in \mathcal{A}$ has a travel cost parameter $c_a \geq 0$ that we interpret as the cost induced for a user of the network if the arc is indeed used. The set $\mathcal{I}$ contains the users of the network and every such user $i \in \mathcal{I}$ wants to travel from the respective origin node s_i to the destination node t_i by using the shortest path (in terms of the travel costs) from s_i to t_i. Let $x_{i,a} \in \{0, 1\}$ be the binary variable encoding whether user $i \in \mathcal{I}$ is using arc $a \in \mathcal{A}$ in its shortest path ($x_{i,a} = 1$) or not ($x_{i,a} = 0$). The leader, who is the toll-setting agency, now decides on tolls $t_a \in \mathbb{R}$ for every arc $a \in \mathcal{A}_{\text{toll}}$ of the network, where $\mathcal{A}_{\text{toll}} \subset \mathcal{A}$ is the set of arcs that can be tolled. If the goal of the leader is revenue maximization, the resulting bilevel optimization problem is given by

$$\begin{aligned} \max_{t,x} \quad & \sum_{a \in \mathcal{A}_{\text{toll}}} \sum_{i \in \mathcal{I}} t_a x_{i,a} \\ \text{s.t.} \quad & t = (t_a)_{a \in \mathcal{A}_{\text{toll}}} \in T, \\ & x_i = (x_{i,a})_{a \in \mathcal{A}} \in \mathcal{S}_i(t) \quad \text{for all } i \in \mathcal{I}. \end{aligned}$$

Here, T is a non-empty (and typically bounded) set restricting the possible toll choices of the leader and $\mathcal{S}_i(t)$ is the set containing all shortest paths of user $i \in \mathcal{I}$, which are now parameterized by the tolls imposed by the leader. Hence, $\mathcal{S}_i(t)$ is the set of globally optimal solutions to the ith follower problem

$$\begin{aligned} \min_{x_i} \quad & \sum_{a \in \mathcal{A}} c_a x_{i,a} + \sum_{a \in \mathcal{A}_{\text{toll}}} t_a x_{i,a} \\ \text{s.t.} \quad & \sum_{a \in \delta^{\text{out}}(v)} x_{i,a} - \sum_{a \in \delta^{\text{in}}(v)} x_{i,a} = b_{i,v} \quad \text{for all } v \in \mathcal{V}, \\ & x_i \in \{0, 1\}^{|\mathcal{A}|}, \end{aligned}$$

where

$$\begin{aligned} \delta^{\text{out}}(v) &:= \{a \in \mathcal{A} \colon a = (v, u) \text{ for some } u \in \mathcal{V}\}, \\ \delta^{\text{in}}(v) &:= \{a \in \mathcal{A} \colon a = (u, v) \text{ for some } u \in \mathcal{V}\} \end{aligned}$$

are the sets of out- and ingoing arcs of node $v \in \mathcal{V}$. Moreover, we set

$$b_{i,v} = \begin{cases} 1, & \text{if } v = s_i, \\ -1, & \text{if } v = t_i, \\ 0, & \text{else.} \end{cases}$$

To make this problem well-posed, we have to assume that for every user $i \in \mathcal{I}$, there is always a path from s_i to t_i that only uses arcs in $\mathcal{A} \setminus \mathcal{A}_{\text{toll}}$ because, otherwise, the overall bilevel problem would be unbounded. Note that in this example, all follower problems are independent of each other, but all depend on the toll choices t of the leader. Such problems are usually called *single-leader multi-disjoint-follower problems*, which we discuss in more detail in Chapter 15.

△

Example 1.18 (Bilevel Knapsack Interdiction) We consider the bilevel knapsack interdiction problem that has been introduced by DeNegre (2011). In this setting, the leader and the follower each own a private knapsack, which can be filled with items from a common set of items indexed by $\{1, \dots, n\}$. For each item $i \in \{1, \dots, n\}$, we denote p_i as the corresponding profit. Moreover, v_i and w_i are the associated weights of item i for the leader and the follower, respectively. The leader's aim is to minimize the follower's maximum profit by prohibiting the usage of certain items by the follower. For this purpose, the leader first selects a subset of items respecting her so-called interdiction budget $\bar{b}$. Afterward, the follower can choose from the remaining items, maximizing his profit while respecting his knapsack capacity C. The bilevel knapsack interdiction problem is formally stated as

$$\begin{aligned} \min_{x} \quad & p^\top y \\ \text{s.t.} \quad & v^\top x \leq \bar{b}, \\ & x \in \{0,1\}^n, \\ & y \in \arg\max_{\bar{y}} \left\{ p^\top \bar{y} : \bar{y} \in \mathcal{Y}(x) \right\} \end{aligned} \tag{1.5}$$

with $\bar{b}, C \in \mathbb{Z}_{>0}$, and $p, v, w \in \mathbb{Z}^n_{>0}$. Here,

$$\mathcal{Y}(x) = \left\{ y \in \{0,1\}^n : w^\top y \leq C, \ y_i \leq 1 - x_i, \ i \in \{1, \dots, n\} \right\}$$

denotes the set of feasible decisions of the follower, which is parameterized by the variables x of the leader. In Problem (1.5), both players consider the same objective function that is optimized in opposite directions. We also call such problems *min-max problems*. △

Example 1.19 (Maximum-Flow Interdiction Problem—Revisited) Let us now formalize the drug smuggling interdiction problem from Example 1.7. To this end, we model the drug smuggling routes using a directed and connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$. Moreover, we have a given source node $s \in \mathcal{V}$ and a given target node $t \in \mathcal{V}$ with $s \neq t$. The flow on each arc, i.e., the amount of drugs smuggled on this part of the route, is denoted by the continuous and non-negative variable $f_a \geq 0$, $a \in \mathcal{A}$. The classic flow balance constraint is then given by

$$\sum_{a \in \delta^{\text{out}}(v)} f_a - \sum_{a \in \delta^{\text{in}}(v)} f_a = 0 \quad \text{for all } v \in \mathcal{V} \setminus \{s, t\}\,.$$

Moreover, the objective function to be maximized is given by

$$\sum_{a \in \delta^{\text{out}}(s)} f_a - \sum_{a \in \delta^{\text{in}}(s)} f_a,$$

i.e., the smugglers want maximize the amount of flow leaving the source node s. Capacities, which serve as upper bounds for the flows on each arc, are denoted by $c_a \geq 0$, $a \in \mathcal{A}$. If we further denote $w_a \in \{0, 1\}$, $a \in \mathcal{A}$, as the interdiction decision of the leader for arc a (with $w_a = 1$ modeling the interdiction of arc $a \in \mathcal{A}$), the lower-level problem can be stated as

$$\begin{aligned}
\varphi(w) := \max_{f \in \mathbb{R}^{|\mathcal{A}|}} \quad & \sum_{a \in \delta^{\text{out}}(s)} f_a - \sum_{a \in \delta^{\text{in}}(s)} f_a \\
\text{s.t.} \quad & \sum_{a \in \delta^{\text{out}}(v)} f_a - \sum_{a \in \delta^{\text{in}}(v)} f_a = 0 \quad \text{for all } v \in \mathcal{V} \setminus \{s, t\}\,, \\
& f_a \leq c_a(1 - w_a) \quad \text{for all } a \in \mathcal{A}, \\
& f_a \geq 0 \quad \text{for all } a \in \mathcal{A}.
\end{aligned}$$

This is a maximum-flow problem that is parameterized by the interdiction decisions w. The overall maximum-flow interdiction problem is then given by

$$\begin{aligned}
\min_{w \in \{0,1\}^{|\mathcal{A}|}} \quad & \varphi(w) \\
\text{s.t.} \quad & \sum_{a \in \mathcal{A}} w_a \leq \bar{b},
\end{aligned}$$

where $\bar{b} > 0$ is the interdiction budget of the leader.

Besides this discrete variant of the interdiction problem, one can also study the continuous variant that we obtain by relaxing the binary interdiction decisions to continuous ones, i.e., by considering $w_a \in [0, 1] \subset \mathbb{R}$, $a \in \mathcal{A}$. In this case, arcs are not completely interdicted but their capacities can be gradually reduced. △

Example 1.20 (See Example 1 in Kleinert (2021)) We now consider the bilevel problem

$$\begin{aligned} \min_{x,y} \quad & F(x,y) = x + 6y \\ \text{s.t.} \quad & -x + 5y \leq 12.5, \\ & x \geq 0, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{1.6}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the linear problem

$$\begin{aligned} \min_{y} \quad & f(x,y) = -y \\ \text{s.t.} \quad & 2x - y \geq 0, \\ & -x - y \geq -6, \\ & -x + 6y \geq -3, \\ & x + 3y \geq 3. \end{aligned}$$

Both the upper- and the lower-level problems are linear optimization problems and all variables are continuous. Thus, we consider an LP-LP bilevel problem, which is the "easiest" class of bilevel models. The problem of this example is illustrated in Figure 1.2. The figure reveals several interesting and important obstacles of bilevel optimization:

(i) The graph of the follower's feasible set mapping corresponds to the gray area. Thus, the follower's problem—and therefore the bilevel problem—is infeasible for certain decisions of the leader, e.g., $x = 0$.
(ii) The set $\{(x,y) : x \in \Omega_x,\ y \in \mathcal{S}(x)\}$ denotes the optimal follower solutions lifted to the x-y-space and is given by the dotted green facets in the top figure. Note that this set is nonconvex.
(iii) The single leader constraint illustrated by the dashed black line in the bottom figure renders certain optimal responses of the follower infeasible. Thus, the bilevel-feasible region $\mathcal{F}$ corresponds to the dashed green facets in the bottom figure. Consequently, the feasible set of Problem (1.6) is not only nonconvex but also disconnected.
(iv) The optimal solution to Problem (1.6) is $(3/7, 6/7)$ with an objective function value of $39/7$. Ignoring the follower's objective, i.e., solving the single-level relaxation, yields the optimal solution $(3, 0)$ with a value of 3. Note that the latter point is not bilevel feasible.

This example shows that bilevel optimization problems are inherently difficult to solve because even their easiest instantiations are nonconvex and nonsmooth

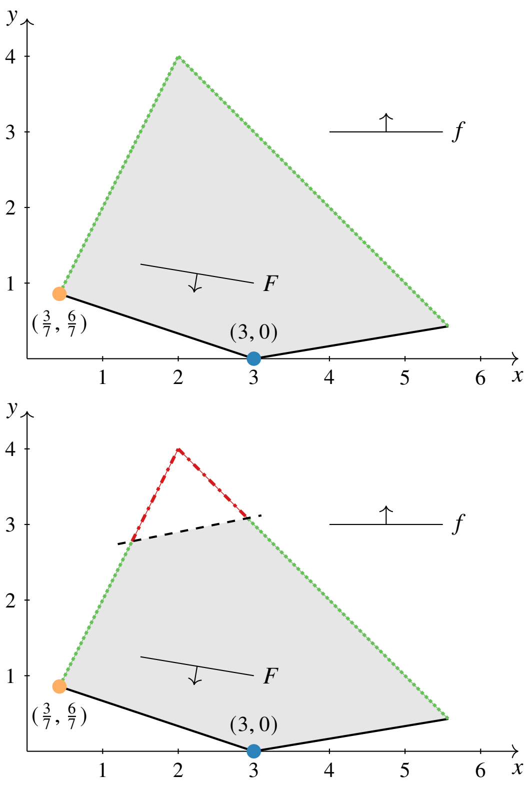


Figure 1.2 The shared constrained set (gray area), the nonconvex bilevel-feasible set (dotted green lines), the optimal solution $(3/7, 6/7)$ to the bilevel problem (orange point), and the optimal solution $(3, 0)$ to the single-level relaxation (blue point) of the bilevel problem (1.6). Top: Without the coupling constraint. Bottom: With the coupling constraint (dashed black line). Note that some lower-level optimal solutions are not bilevel feasible anymore (dash-dotted red lines). Taken and modified from Kleinert (2021).

optimization problems. More examples illustrating the difficulties of (linear) bilevel optimization are discussed in Chapter 4. △

Let us consider one more example, which is taken and adapted from Kleinert et al. (2021a), to highlight another important, but maybe surprising, property of bilevel problems. Even if the leader's and follower's objective functions are perfectly aligned, the bilevel problem and its single-level relaxation may still not be equivalent.

Example 1.21 We consider the linear bilevel problem

$$\max_{x,y} \quad y \quad \text{s.t.} \quad 0 \le x \le 2,\ y \le \frac{3}{2},\ y \in \mathcal{S}(x),$$

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

$$\max_{y} \quad y \quad \text{s.t.} \quad y \ge 0,\ y \le 1 + x,\ y \le 3 - x.$$

Because both the leader and the follower want to maximize y, both objectives are exactly the same. One might think that the bilevel problem could then be equivalent to its single-level relaxation

$$\max_{x,y} \quad y \quad \text{s.t.} \quad 0 \le x \le 2,\ y \le \frac{3}{2},\ y \ge 0,\ y \le 1 + x,\ y \le 3 - x.$$

This is, however, not the case. The set of optimal solutions to the single-level relaxation is given by

$$\left\{(x, y) : x \in \left[\frac{1}{2}, \frac{3}{2}\right],\ y = \frac{3}{2}\right\},$$

whereas the set of solutions to the bilevel problem is given by

$$\left\{\left(\frac{1}{2}, \frac{3}{2}\right), \left(\frac{3}{2}, \frac{3}{2}\right)\right\}.$$

Hence, there are solutions to the single-level relaxation that are not a solution to the bilevel problem and the problems are not equivalent, although the objective functions coincide. △

Before we start to analyze bilevel problems in detail, let us mention some other general literature in which you can find additional information. First, we refer to the survey articles by Colson et al. (2005, 2007) and Kleinert et al. (2021a), as well as to the books by Bard (1998), Dempe (2002), and Dempe et al. (2015). Note that Dempe et al. (2015) also contains a small set of nice examples of bilevel models from the fields of chemical equilibrium, traffic tolls, and electricity market modeling. More recently, the book edited by Dempe and Zemkoho (2020) compiles the latest state-of-the-art. Other very early survey articles include Anandalingam and Friesz (1992), Ben-Ayed (1993), Kolstad (1985), and Vicente and Calamai (1994) as well as Wen and Hsu (1991) regarding the field of linear bilevel optimization. Last but not least, Dempe (2020) contains, to the best of our knowledge, the largest annotated list of references in the field of bilevel optimization.

Remark 1.22 For modeling practically relevant applications, one is, of course, not restricted to use “only” two levels as we do here in bilevel optimization. If the optimization problem has constraints that again contain an optimization problem that has constraints, which again contain an optimization problem . . . and so on, the problem is called a *multilevel optimization problem*. You might imagine that three or four levels do not make the problem easier to analyze and solve—especially because we have already seen that bilevel problems are extremely challenging even in their easiest instantiation. Nevertheless, multilevel optimization is often required to model real-world situations; see, e.g., Ambrosius et al. (2020), Grimm et al. (2019a, 2016, 2019b), Kleinert and Schmidt (2019), and Schewe et al. (2022).

Exercise 1.23 Prof. Jones is a collector of rare artifacts. He recently returned from an adventure in South America from which he also brought some valuable treasures. Alice and Bob, a famous robber couple, plan their next raid on Prof. Jones. Alice is the mastermind of the duo and came across some inside knowledge about the number $n \in \mathbb{N}$, the values $v \in \mathbb{R}^n_{\geq 0}$, and the weights $w \in \mathbb{R}^n_{\geq 0}$ of the artifacts in Prof. Jones’ possession, as well as the security measures installed at his mansion. The property is heavily protected but Bob, the henchman of the duo, can access the building by climbing through a bathroom window. For climbing, Bob needs to have his hands free. Therefore, he will carry the stolen items in a backpack. Alice’s task is to buy a backpack of appropriate size $b \in \mathbb{R}_{\geq 0}$ that Bob will use in the raid. The backpack should be neither too big nor too small, i.e., $b_l \leq b \leq b_u$ with $0 \leq b_l \leq b_u \in \mathbb{R}$. The cost $c \in \mathbb{R}_{\geq 0}$ for the backpack is assumed to be proportional to its size. Bob cannot split any items, he can either take item $i \in \{1, \dots, n\}$ or leave it. Moreover, Bob can only take a subset of items such that the capacity of the backpack is not exceeded, i.e., $w^\top x \leq b$ if x is a binary vector with entries modeling if an item is taken or not. The aim of the robber duo is to maximize their profits, which is the difference between the value $v^\top x$ of the stolen items and the costs cb for buying the backpack.

(i) Formulate a bilevel problem that models an optimal raid strategy.
(ii) Who acts as the leader and who acts as the follower?
(iii) Are there linking variables and/or coupling constraints?
(iv) The raid on Prof. Jones went horribly wrong. Bob was arrested by the police and had to spend time in prison. After his release, Bob still holds a grudge against Alice because she got away with impunity. Therefore, they no longer work together but do solo robberies instead. During his time in prison, fellow criminals helped Bob work on his robbery skills, which now gives him an advantage over Alice. This means that he will always be the first to arrive at a new potential robbery target. Unlike Alice,

Bob is not interested in the value of his stolen items. His sole aim is to leave those items for Alice to steal that yield the worst possible outcome for her. Alice and Bob own a backpack of size $b_{\mathrm{A}} \in \mathbb{R}_{\geq 0}$ and $b_{\mathrm{B}} \in \mathbb{R}_{\geq 0}$, respectively, which they use to store stolen items.

What needs to be adapted in the previous modeling to account for the change of circumstances?

Exercise 1.24 Consider the linear bilevel problem

$$
\begin{aligned}
\min_{x,y} \quad & -x - 2y \\
\text{s.t.} \quad & x + 4y \geq 12, \\
& x \geq 0, \\
& y \in \arg\min_{\bar{y}} \{\bar{y} : x + \bar{y} \geq 5,\ -x - \bar{y} \geq -10,\ -x + 4\bar{y} \geq 0\}.
\end{aligned} \tag{1.7}
$$

(i) Plot the shared constraint set Ω of Problem (1.7) in a coordinate system.
(ii) Determine the projection Ω_x of the shared constraint set Ω onto the x-space.
(iii) Determine the solution to the single-level relaxation of Problem (1.7).
(iv) Suppose that the leader sticks to the solution found in (iii). What would be the optimal response of the follower?
(v) Determine the bilevel-feasible set of Problem (1.7). What are its properties?
(vi) Determine the optimal solution to Problem (1.7).

Exercise 1.25 Consider the linear bilevel problem

$$
\begin{aligned}
\min_{x,y} \quad & x - 2y \\
\text{s.t.} \quad & x + 2y \geq 12, \\
& -x + 2y \geq -2, \\
& y \in \arg\min_{\bar{y}} \{\bar{y} : x - \bar{y} \geq -3,\ -2x - \bar{y} \geq -24,\ \bar{y} \geq 0\}.
\end{aligned} \tag{1.8}
$$

(i) Plot the shared constraint set Ω of Problem (1.8) in a coordinate system.
(ii) Determine the projection Ω_x of the shared constraint set Ω onto the x-space.
(iii) Use the optimal-value function to reformulate Problem (1.8) as a single-level problem.
(iv) Determine the solution to the single-level relaxation of Problem (1.8).
(v) What is the set of optimal follower responses lifted to the x-y-space?

(vi) Determine the bilevel-feasible set. What can you say about the solvability of Problem (1.8)?
(vii) What changes if we move the coupling constraints to the lower level?

1.3 A Brief History of Bilevel Optimization

Bilevel optimization dates back to the seminal publications on leader-follower games by von Stackelberg (1934, 1952). The formulation introduced in the last section was first used in Bracken and McGill (1973) in the context of a military application regarding the cost-minimal mix of weapons. Another very early discussion of multilevel or, in particular, two-level problems can be found in Candler and Norton (1977). Candler and Norton (1977) already recognized in the early days of bilevel optimization that such problems are very challenging to solve. More precisely, the authors noticed that even in the "simplest case" of continuous variables and linear objective functions and constraints, the feasible set of bilevel problems may be nonconvex and disconnected. In fact, formal complexity results, which were derived much later, state that even this "easiest" class of linear bilevel problems is already strongly NP-hard (Hansen et al. 1992). Candler and Norton (1977) also proposed an enumerative algorithm for linear bilevel problems similar to the simplex method, but they had "no doubt others could develop more efficient algorithms". After Bialas and Karwan (1978) proposed the so-called Kth-best algorithm——another enumerative and simplex-inspired method—Fortuny-Amat and McCarl (1981) introduced a game-changing approach for convex-quadratic bilevel problems. They replaced the follower's problem with its necessary and sufficient Karush–Kuhn–Tucker (KKT) conditions to derive an equivalent single-level problem that can be further reformulated and tackled by standard mixed-integer optimization solvers. We discuss this approach in detail in Chapter 3. Bard and Moore (1990), Bard (1988), Edmunds and Bard (1991), and Hansen et al. (1992) picked up the idea later and this approach is still standard for solving bilevel problems with convex follower problems today. Alternative approaches, e.g., penalty methods or descent approaches, have been proposed by Anandalingam and White (1990) as well as by Savard and Gauvin (1994). In the 1990s, the largest instances of linear bilevel problems that have been solved included 250 leader variables, 150 follower variables, and 150 follower constraints; see Hansen et al. (1992). Although cutting planes were derived in the following years, see Audet et al. (2007a) and Audet et al. (2007b), computational linear and convex bilevel optimization did not attract much attention in the 2000s, and not many computational results have been reported.

Moore and Bard (1990) developed a branch-and-bound approach for bilevel problems with mixed-integer follower problems and also reported some first numerical results already in 1990. However, only very little computational progress has been reported until DeNegre and Ralphs (2009) introduced a branch-and-cut approach for purely integer bilevel problems in 2009. This work can be considered a tipping point for computational bilevel optimization and many computationally oriented works for various classes of bilevel problems appeared afterward. For these more recent research works, we refer to the survey by Kleinert et al. (2021a), which focuses on exactly these developments. Part III of this book gives an introduction to algorithmic advances in this context.

Exercise 1.26 Read the early publications by von Stackelberg (1934, 1952), Bracken and McGill (1973), as well as Candler and Norton (1977). If you have the time and find this interesting, read Fortuny-Amat and McCarl (1981) as well. We will come back to their paper anyway.

1.4 What You Should Know Now!

1. In what situations do bilevel problems occur in general?
2. Why are bilevel problems called hierarchical optimization problems?
3. What is a pricing problem?
4. What is the toll-setting problem?
5. In what situations do bilevel problems occur in energy markets?
6. What is the relation between bilevel models and critical infrastructure defense?
7. What are interdiction problems about?
8. Can you explain the maximum-flow interdiction problem?
9. How is a bilevel optimization problem defined formally?
10. What are coupling constraints?
11. What are linking variables?
12. What are linking constraints?
13. What is the follower's feasible set mapping?
14. What is the follower's optimal response mapping?
15. What is the rational reaction set?
16. When do we call a bilevel problem "optimistic"?
17. What is the solution-set mapping?
18. What is the shared constraint set?

19. What is the bilevel-feasible set/the inducible region?
20. What is a local minimizer of a bilevel problem?
21. What is a global minimizer of a bilevel problem?
22. What is the difference between the conventional and the original formulation of the optimistic bilevel problem?
23. What is the optimal-value function of the lower-level problem?
24. How does the optimal-value-function reformulation look like?
25. Why do we have to repeat the lower-level constraints in the optimal-value-function reformulation although they are also part of the right-hand side of the optimal-value function constraint?
26. What is the single-level relaxation? Is it really a relaxation? If yes, a relaxation of what and why?
27. How can a pricing problem be formally modeled as a bilevel optimization problem?
28. Can you formally model the toll-setting problem?
29. How can we formally model the knapsack interdiction problem?
30. Can you formally state the maximum-flow interdiction problem?
31. What nasty properties do we already encounter if we consider the "easiest" case of bilevel problems, i.e., LP-LP bilevel problems?
32. What is the effect of coupling constraints?
33. Is a linear bilevel problem with the same objective function on both levels equivalent to its single-level relaxation?

2

Solution Concepts: Optimistic vs. Pessimistic Problems

In Section 1.2, we define the bilevel problem as

$$\begin{aligned} \min_{x \in X, y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{2.1}$$

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

$$\begin{aligned} \min_{y \in Y} \quad & f(x, y) \\ \text{s.t.} \quad & g(x, y) \geq 0. \end{aligned} \tag{2.2}$$

We call this the optimistic bilevel optimization problem as it models the situation in which, if the lower-level problem has multiple optimal solutions, a solution is chosen that leads to the best upper-level objective function value. In other words, if the lower-level problem's solution is ambiguous, the follower cooperates with the leader. However, this does not necessarily need to be the case. For instance, in the same situation, the follower could also choose a solution that leads to the worst upper-level objective function value. The next example shows that these different notions of a bilevel problem can lead to significantly different solutions.

Example 2.1 (See Example 1.2 in Dempe et al. (2015)) Consider the bilevel problem

$$\text{“}\min_{x}\text{”} \quad F(x, y) = x^2 + y \quad \text{s.t.} \quad y \in \mathcal{S}(x)$$

with

$$\mathcal{S}(x) = \arg\min_{y} \left\{ -xy : 0 \leq y \leq 1 \right\}.$$

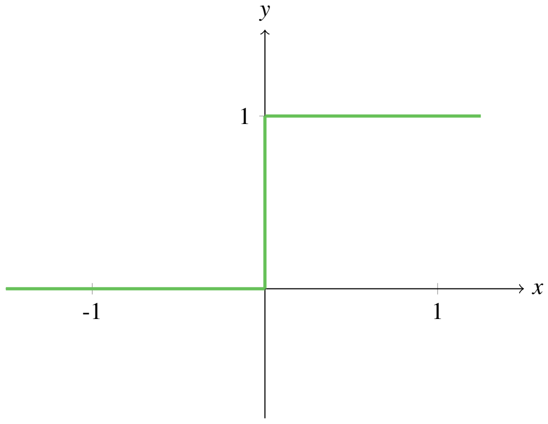


Figure 2.1 The optimal solution y to the lower-level problem in dependence of the upper-level decision x in Example 2.1. Taken and modified from Dempe et al. (2015).

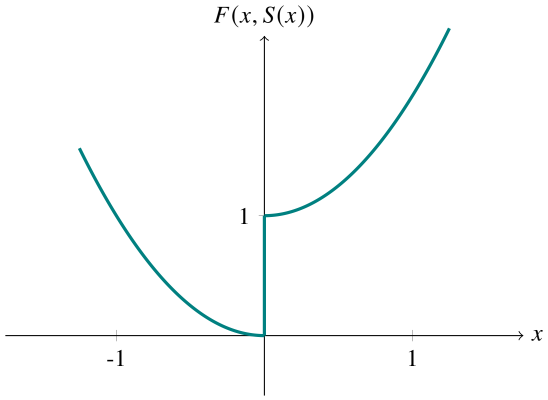


Figure 2.2 The graph of the mapping $x \mapsto F(x, \mathcal{S}(x))$ in Example 2.1. Taken and modified from Dempe et al. (2015).

The best response of the follower is illustrated in Figure 2.1. Formally, it is given by

$$\mathcal{S}(x) = \begin{cases} [0, 1], & x = 0, \\ \{0\}, & x < 0, \\ \{1\}, & x > 0. \end{cases}$$

The mapping $x \mapsto F(x, \mathcal{S}(x))$ is illustrated in Figure 2.2. Note that this is not a function and its minimum is unclear as it depends on the response $y \in \mathcal{S}(x)$ of the follower if the leader chooses $x = 0$. For the follower, all responses $y \in \mathcal{S}(0) = [0, 1]$ are optimal so that the optimal lower-level solution is not unique. Because F is not a function, the overall problem is not well-defined, which is the reason why we put the "min" in quotation marks. If the follower chooses $y = 0$ as an optimal response to $x = 0$, this leads to an upper-level objective function value of 0. However, if the follower chooses $y = 1$, the objective function value of the leader is 1, which is worse than 0 from the leader's point of view.

This means the following. If the follower's problem does not have a unique solution, i.e., the set $\mathcal{S}(x)$ is not a singleton, the follower can be "leader-friendly" (which corresponds to $y = 0$ in our example) or the follower can choose a different solution, e.g., $y = 1$, which is worse for the leader (obtaining an objective function value of 1 in this case). △

Example 2.1 illustrates that the general bilevel problem is ill-posed if one does not decide which solution of the lower-level problem is used in case of multiplicities. To resolve this issue, one usually distinguishes between two different solution concepts in bilevel optimization: the *optimistic solution* and the *pessimistic solution*.

Note that we already formally defined the optimistic bilevel problem in Definition 1.8. In this optimistic case, the leader controls those y that are part of the rational reaction set $\mathcal{S}(x)$ of the follower. This is indicated by also having the y below the "min" of the upper-level objective function in Problem (2.1).

For the pessimistic variant of the problem, the definition further depends on whether the bilevel problem has coupling constraints or not. If no coupling constraints are present, we have the following definition.

Definition 2.2 (Pessimistic Bilevel Problem without Coupling Constraints) Consider the problem

$$\begin{aligned} \min_{x \in X} \quad & \max_{y \in \mathcal{S}(x)} F(x, y) \\ \text{s.t.} \quad & G(x) \geq 0, \\ & \mathcal{S}(x) \neq \emptyset, \end{aligned} \tag{2.3}$$

in which $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized lower-level problem, which is defined as in (2.2). Then, (2.3) is called the *pessimistic bilevel problem without coupling constraints*.

The last definition is a bit more technical compared to what is often stated in the literature. The reason is that without the last constraint $\mathcal{S}(x) \neq \emptyset$, the definition would not define what one usually wants to model in practice, as the leader may have the possibility to choose an x so that $\mathcal{S}(x) = \emptyset$ holds. Then, the inner maximization would be carried out over an empty set, formally leading to the value of $-\infty$, which is optimal for the outer minimization problem, i.e., for the problem of the leader. Note, however, that some papers address this problem by supposing that $\mathcal{S}(x) \neq \emptyset$ holds for all $x \in X$ with $G(x) \geq 0$. With such an assumption, the last constraint in (2.3) is, of course, redundant.

The definition of a pessimistic bilevel problem with coupling constraints reads as follows.

Definition 2.3 (Pessimistic Bilevel Problem with Coupling Constraints) The problem

$$\begin{aligned} \min_{x \in X} \quad & \max_{y \in \mathcal{S}(x)} F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0 \quad \text{for all } y \in \mathcal{S}(x), \\ & \mathcal{S}(x) \neq \emptyset, \end{aligned}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized lower-level problem, which is defined as in (2.2), is called the *pessimistic bilevel problem with coupling constraints*.

As we stated the optimistic bilevel problem in Definition 1.8, it is a problem in the variables x and y. However, for the last definition, this notion does not make sense anymore because the y-variable is connected to a universal quantifier. Hence, there is no single y that can be stated as part of a solution, i.e., the solution space depends on the considered solution concept. There are papers that consider the x-space as the solution space for optimistic bilevel problems and treat the follower's variables y only as a certificate of the feasibility of x. For what follows, we stay with stating (x, y) as a solution of an optimistic bilevel problem. For pessimistic problems without coupling constraints, both variants are valid options, but we decide to only consider the x-space as the solution space to unify the presentation for both types of pessimistic bilevel problems.

Note that the decision on the solution concept (optimistic vs. pessimistic) is very important and can even change whether a solution exists or not. For instance, the optimal solution $(x, y) = (0, 0)$ with optimal objective function value 0 is attained in Example 2.1 if one considers the optimistic bilevel problem. However, for all other choices of $y \in \mathcal{S}(0) = [0, 1]$, the bilevel problem is not solvable because the infimum 0 of the upper-level objective function is not attained anymore. This, in particular, also applies to the pessimistic variant of the bilevel problem in this example. In the pessimistic case, the point $(x, y) = (0, 1)$ is bilevel feasible with an upper-level objective function value of 1, but choosing a sequence $(x^k)_k$ with $x^k \nearrow 0$ leads to follower responses $(y^k)_k$ with $y^k = 0$, which is better from the leader's perspective if k is large enough. However, the limit $(x^*, y^*) = (0, 0)$ is not attained in the pessimistic setting.

Remark 2.4 If the solution to the lower-level problem is unique for all $x \in \Omega_x$, both the optimistic and the pessimistic variant of the bilevel problem coincide.

It has been common sense in bilevel optimization that pessimistic bilevel problems are more complicated than their optimistic counterparts. However, Zeng (2020) showed that this is actually not the case in the sense that, for a given pessimistic bilevel problem, one can derive an optimistic bilevel problem that has the same set of optimal solutions in the x-space. This is what we state and prove in the following theorem for the simplified case in which there are no coupling constraints.

Theorem 2.5 *Consider the pessimistic bilevel problem (2.3) without coupling constraints, i.e.,*

$$\min_{x \in \bar{X}} \max_{y \in \mathcal{S}(x)} F(x, y), \tag{2.4}$$

where we set $\bar{X} := \{x \in X : G(x) \geq 0\}$ *and where the lower-level problem is given by*

$$\min_{y} \quad f(x, y) \quad \text{s.t.} \quad y \in \mathcal{Y}(x) := \{y \in Y : g(x, y) \geq 0\}.$$

Let S denote the set of optimal solutions to Problem (2.4), where we only consider solutions in the x-space. Moreover, consider the optimistic bilevel problem

$$\begin{aligned} \min_{x \in \bar{X}, \bar{y}} \quad & \max_{y \in \bar{\mathcal{Y}}(x, \bar{y})} F(x, y) \\ \text{s.t.} \quad & \bar{y} \in \mathcal{Y}(x) \end{aligned} \tag{2.5}$$

with $\bar{\mathcal{Y}}(x, \bar{y}) := \{y \in \mathcal{Y}(x) : f(x, y) \leq f(x, \bar{y})\}$. *Let $\bar{S}$ be the set of optimal solutions to Problem (2.5), where we only consider solutions in the $(x, \bar{y})$-space. Then, $S = \operatorname{proj}_x(\bar{S})$ holds.*

Proof: First, we re-write both problems in their epigraph reformulation. Hence, (2.4) is equivalent to

$$\begin{aligned} \min_{x \in \bar{X}, t} \quad & t \\ \text{s.t.} \quad & t \geq F(x, y) \quad \text{for all } y \in \mathcal{S}(x) \end{aligned} \tag{2.6}$$

and (2.5) is equivalent to

$$\begin{aligned} \min_{x \in \bar{X}, \bar{y}, t} \quad & t \\ \text{s.t.} \quad & t \geq F(x, y) \quad \text{for all } y \in \bar{\mathcal{Y}}(x, \bar{y}), \\ & \bar{y} \in \mathcal{Y}(x). \end{aligned} \tag{2.7}$$

Second, we show that any feasible point (x^*, t^*) for (2.6) can be extended by a $\bar{y}^*$ so that $(x^*, \bar{y}^*, t^*)$ is feasible for (2.7). To this end, we choose any $\bar{y}^* \in \mathcal{S}(x^*)$. As a solution to the x^*-parameterized lower-level problem, it is also feasible, i.e., $\bar{y}^* \in \mathcal{Y}(x^*)$ holds. With the optimal-value function φ of the lower-level problem, we thus get

$$\begin{aligned} & \bar{\mathcal{Y}}(x^*, \bar{y}^*) \\ &= \{y \in \mathcal{Y}(x^*) \colon f(x^*, y) \leq f(x^*, \bar{y}^*)\} \\ &= \{y \in \mathcal{Y}(x^*) \colon f(x^*, y) \leq \varphi(x^*)\} \\ &= \mathcal{S}(x^*) \end{aligned}$$

Hence, $(x^*, \bar{y}^*, t^*)$ is feasible for (2.7).

Third, we prove that for any point $(x^*, \bar{y}^*, t^*)$ that is feasible for (2.7), the point (x^*, t^*) is feasible for (2.6). It holds

$$t \geq F(x, y) \text{ for all } y \in \mathcal{S}(x) \iff t \geq \max_y \{F(x, y) \colon y \in \mathcal{S}(x)\}$$

and

$$t \geq F(x, y) \text{ for all } y \in \bar{\mathcal{Y}}(x, \bar{y}) \iff t \geq \max_y \left\{F(x, y) \colon y \in \bar{\mathcal{Y}}(x, \bar{y})\right\}.$$

Because of $\bar{y}^* \in \mathcal{Y}(x^*)$, we obtain

$$\begin{aligned} & \max_y \{F(x^*, y) \colon y \in \mathcal{S}(x^*)\} \\ &= \max_y \{F(x^*, y) \colon y \in \mathcal{Y}(x^*),\ f(x^*, y) \leq \varphi(x^*)\} \\ &\leq \max_y \{F(x^*, y) \colon y \in \mathcal{Y}(x^*),\ f(x^*, y) \leq f(x^*, \bar{y}^*)\} \\ &= \max_y \left\{F(x^*, y) \colon y \in \bar{\mathcal{Y}}(x^*, \bar{y}^*)\right\}, \end{aligned}$$

which immediately shows that (x^*, t^*) is feasible for (2.6).

Fourth and finally, we obtain the claim by observing that the objective functions of (2.6) and (2.7) are the same and do not depend on $\bar{y}$. By reversing the epigraph reformulation again, we obtain the statement of the theorem. □

Note that we did not use specific properties of the upper- or lower-level problem in the proof of the last theorem. This means that the same transformation can also be applied to, e.g., mixed-integer linear problems in the lower level.

One can also recover the y-part of a solution to the pessimistic bilevel problem (2.4). The respective details can be found in Section 2 in Zeng (2020). An analogous result to Theorem 2.5 can also be shown for pessimistic bilevel problems with coupling constraints; see Henke et al. (2025b).

Exercise 2.6 Consider the linear bilevel problem

$$\min_{x} \quad x + 2y \quad \text{s.t.} \quad x \geq 0, \; y \in \mathcal{S}(x), \tag{2.8}$$

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

$$\begin{aligned} \min_{y} \quad & x \\ \text{s.t.} \quad & 3x + 2y \leq 20, \\ & x + 2y \leq 12, \\ & x - 2y \leq 4, \\ & 0 \leq y \leq 5. \end{aligned}$$

(i) Plot the shared constraint set of Problem (2.8) in a coordinate system.
(ii) Determine the bilevel-feasible set of Problem (2.8).
(iii) Determine the optimal solution to the optimistic variant of Problem (2.8).
(iv) Suppose that the leader sticks to the solution found in (iii). What would be the optimal response of the follower in the pessimistic variant of the bilevel problem?
(v) Determine the optimal solution to the pessimistic variant of Problem (2.8).

2.1 What You Should Know Now!

1. How do we define an optimistic bilevel problem?
2. How do we define a pessimistic bilevel problem without coupling constraints?
3. How do we define a pessimistic bilevel problem with coupling constraints?
4. Why are solutions to pessimistic bilevel problems with coupling constraints stated only in the x-space?
5. Why do we need to distinguish between optimistic and pessimistic solutions at all?
6. Can you illustrate the differences between these solution concepts using an example?
7. What is nice about uniquely defined lower-level solutions?
8. What is the meaning of the result by Zeng (2020)?

PART TWO

CONVEX LOWER-LEVEL PROBLEMS

3
Single-Level Reformulations

Since the seminal paper by Fortuny-Amat and McCarl (1981), the solution method for bilevel problems most frequently used in practice has been to reformulate the bilevel model as an "ordinary", i.e., a single-level, problem. After some further transformations, the resulting single-level reformulation can then be solved with state-of-the-art general-purpose solvers for the resulting classes of problems.

In this chapter, we consider three different types of single-level reformulations.

(i) The first one is based on the optimal-value function of the lower-level problem and is discussed in Section 3.1.
(ii) The second one uses the Karush–Kuhn–Tucker (KKT) conditions of the lower-level problem, which is the approach originally proposed by Fortuny-Amat and McCarl (1981). For linear problems, we introduce this reformulation in Section 3.2. The resulting problem can even be reformulated again to obtain a single-level mixed-integer linear problem, which we derive in Section 3.4 and for which we show how these models can be implemented in `Python` in Section 3.5.
(iii) The third one is based on a strong-duality theorem for the lower-level problem. The details are discussed in Section 3.3.

Whereas the first single-level reformulation can be applied to any bilevel problem, the two latter ones require that the lower-level problem possesses some compact optimality certificate, which is not only necessary but also sufficient. Here and in what follows, we say that such a certificate is *compact* if it only uses polynomially many additional variables and constraints. Compact optimality certificates are typically only available in the case of convex lower-level problems that satisfy an appropriate constraint qualification (CQ). We discuss a single-level reformulation for the general nonlinear but convex setting in Section 3.6. Finally, in Section 3.7, we discuss another structural property of bilevel problems, called independence of irrelevant constraints (IIC) property, that has an important connection to the single-level relaxation of a given bilevel problem.

3.1 The Optimal-Value-Function Reformulation

Let us start again with the general optimistic bilevel problem

$$\begin{aligned} \min_{x \in X, y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & y \in \mathcal{S}(x), \end{aligned}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized problem

$$\begin{aligned} \min_{y \in Y} \quad & f(x, y) \\ \text{s.t.} \quad & g(x, y) \geq 0. \end{aligned} \tag{3.1}$$

We already considered this problem in Definition 1.8. By using the optimal-value function

$$\varphi(x) := \inf_{y \in Y} \{ f(x, y) \colon g(x, y) \geq 0 \},$$

we can equivalently re-write the problem as

$$\begin{aligned} \min_{x \in X, y \in Y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & g(x, y) \geq 0, \\ & f(x, y) \leq \varphi(x). \end{aligned} \tag{3.2}$$

To the best of our knowledge, this reformulation has been used for the first time by Outrata (1990), where it was the basis of what was called the "quasi-indirect approach". Problem (3.2) looks like a usual single-level problem. However, the use of the optimal-value function $\varphi : \mathbb{R}^{n_x} \to \mathbb{R} \cup \{\pm\infty\}$ in the constraints is problematic. Of course, we can evaluate this function, but this evaluation corresponds to solving the lower-level problem (3.1) for a given x, which is the argument of this function. Thus, the evaluation can be rather expensive. Moreover, in almost all cases, the optimal-value function is not known in algebraic, i.e., in closed, form. Finally, it is usually nonsmooth (even under strong assumptions). This can, e.g., be seen in Example 1.20, where the two dotted green segments in Figure 1.2 represent the function φ restricted to Ω_x.

Although this might suggest that the single-level reformulation based on the optimal-value function is of limited use, it can actually be very useful for problems that exhibit a special structure of the optimal-value function. We come back to this aspect in Part III of this book, in which the optimal-value function is frequently used.

3.2 The KKT Reformulation for LP-LP Bilevel Problems

The most classic approach to obtain a single-level reformulation is to exploit optimality conditions for the lower-level problem. Because these optimality conditions need to be necessary and sufficient, their application is usually only possible for convex lower-level problems that satisfy a reasonable constraint qualification—which is typically Slater's constraint qualification in the convex case; see Definition A.25 in Appendix A.3.

To avoid over-complicating the presentation, we first present the KKT reformulation for LP-LP bilevel problems of the form

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \tag{3.3a}$$

$$\text{s.t.} \quad Ax + By \geq a, \tag{3.3b}$$

$$y \in \arg\min_{\bar{y}} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \tag{3.3c}$$

with $c_x \in \mathbb{R}^{n_x}$, $c_y, d \in \mathbb{R}^{n_y}$, $A \in \mathbb{R}^{m \times n_x}$, $B \in \mathbb{R}^{m \times n_y}$, and $a \in \mathbb{R}^m$ as well as $C \in \mathbb{R}^{\ell \times n_x}$, $D \in \mathbb{R}^{\ell \times n_y}$, and $b \in \mathbb{R}^\ell$. Note that we omit a linear term depending on the upper-level variables x in the lower-level objective function because this term would not have any influence on the optimal solutions to the lower level as it is constant from the follower's point of view.

The lower-level problem in (3.3c) is the x-parameterized linear problem

$$\min_y \quad d^\top y \quad \text{s.t.} \quad Dy \geq b - Cx. \tag{3.4}$$

Its Lagrangian function is given by

$$\mathcal{L}(x, y, \lambda) = d^\top y - \lambda^\top (Cx + Dy - b)$$

and the KKT conditions are given by dual feasibility

$$D^\top \lambda = d, \quad \lambda \geq 0,$$

primal feasibility

$$Cx + Dy \geq b,$$

and the KKT complementarity conditions

$$\lambda_i(C_{i\cdot}x + D_{i\cdot}y - b_i) = 0 \quad \text{for all } i = 1, \dots, \ell.$$

Here and in what follows, $C_{i\cdot}$ and $D_{i\cdot}$ denote the ith rows of C and D, respectively. As the lower-level feasible region is polyhedral, the Abadie constraint qualification (see Definition A.17 in Appendix A.2) holds and the KKT conditions are both necessary and sufficient. By replacing the lower-level problem with its KKT conditions, the LP-LP bilevel problem (3.3) can thus be reformulated as

$$\begin{aligned}
\min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y && (3.5a)\\
\text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, && (3.5b)\\
& D^\top \lambda = d, \; \lambda \geq 0, && (3.5c)\\
& \lambda_i(C_{i\cdot}x + D_{i\cdot}y - b_i) = 0 \quad \text{for all } i = 1, \dots, \ell, && (3.5d)
\end{aligned}$$

which is the KKT reformulation originally proposed by Fortuny-Amat and McCarl (1981). Note that we now optimize over an extended space of variables because we additionally have to include the lower-level dual variables $\lambda \in \mathbb{R}^\ell$. By optimizing over x, y, and λ simultaneously in Problem (3.5), any globally optimal solution to (3.5) corresponds to an optimistic bilevel solution.

Problem (3.5) is linear except for the KKT complementarity conditions in (3.5d) that turn the problem into a nonconvex and nonlinear optimization problem (NLP). More precisely, Problem (3.5) is a mathematical program with complementarity constraints (MPCC); see, e.g., Luo et al. (1996) and Appendix A.4. Thus, and unfortunately, standard NLP algorithms usually cannot be applied for such problems because classic constraint qualifications such as the Mangasarian–Fromowitz or the linear independence constraint qualification are violated at every feasible point; see, e.g., Ye and Zhu (1995) or Appendix A.4 again.

3.3 The Strong-Duality Based Reformulation

Besides the approach based on the KKT conditions of the lower level, one can also use a strong-duality theorem for the lower-level problem to obtain a single-level reformulation, if such a theorem is at hand. In the linear case considered up to now, this is the case. The dual problem of (3.4) is given by

$$\max_{\lambda} \quad (b - Cx)^\top \lambda \quad \text{s.t.} \quad D^\top \lambda = d, \; \lambda \geq 0. \tag{3.6}$$

Note that also the dual problem (3.6) is an x-parameterized linear problem but now the objective function (and not the constraints) depends on x. For a given decision x of the leader, weak duality of linear optimization states that

$$d^\top y \geq (b - Cx)^\top \lambda$$

holds for every primal and dual feasible pair y and λ. Thus, by strong duality, we know that every such feasible pair is a pair of optimal solutions if

$$d^\top y \leq (b - Cx)^\top \lambda$$

holds, which implies that the latter needs to be satisfied with equality. Consequently, we can use strong duality for the follower to reformulate the bilevel problem as

$$\min_{x,y,\lambda} \quad c_x^\top x + c_y^\top y \tag{3.7a}$$

$$\text{s.t.} \quad Ax + By \geq a, \ Cx + Dy \geq b, \tag{3.7b}$$

$$D^\top \lambda = d, \ \lambda \geq 0, \tag{3.7c}$$

$$d^\top y \leq (b - Cx)^\top \lambda. \tag{3.7d}$$

Here, the ℓ many KKT complementarity constraints in (3.5d) are replaced with the single inequality in (3.7d). Note that the general nonconvexity of LP-LP bilevel problems is reflected in this single-level reformulation by the bilinear products of the primal upper-level variables x and the dual lower-level variables λ. Under certain assumptions, these bilinear terms can be equivalently reformulated using so-called *McCormick inequalities* (McCormick 1976), resulting in a model that can be handled by general-purpose solvers for mixed-integer linear optimization problems. We discuss such reformulations in more detail below in Remark 3.2 and later in Chapter 10. Nevertheless, we mention that many modern solvers nowadays also support bilinear terms directly.

Remark 3.1 The KKT reformulation (3.5) and the strong-duality based reformulation (3.7) are equivalent because

$$\begin{aligned}
& \lambda_i(C_{i\cdot}x + D_{i\cdot}y - b_i) = 0 \quad \text{for all } i = 1, \dots, \ell \\
\iff \quad & \lambda^\top(Cx + Dy - b) = 0 \\
\iff \quad & \lambda^\top Dy = \lambda^\top(b - Cx) \\
\iff \quad & d^\top y = \lambda^\top(b - Cx)
\end{aligned}$$

holds, where we use $\lambda \geq 0$ and $Cx + Dy - b \geq 0$ for the first equivalence as well as $D^\top \lambda = d$ for the last one.

Remark 3.2 The only nonlinearity in Problem (3.7) is the term $x^\top C^\top \lambda$ in (3.7d). This means that the main hardness of the problem stems from the product of upper-level primal variables x and lower-level dual variables λ.

Such bilinearities are often tackled by using so-called McCormick inequalities (McCormick 1976). To illustrate this, let us consider two scalar variables $x, \lambda \in \mathbb{R}$ with finite bounds $x \in [\underline{x}, \bar{x}]$ and $\lambda \in [\underline{\lambda}, \bar{\lambda}]$. Moreover, let $w = x\lambda$. Then, the inequalities

$$\begin{aligned} w &\geq \underline{\lambda}x + \underline{x}\lambda - \underline{x}\underline{\lambda}, \\ w &\geq \bar{\lambda}x + \bar{x}\lambda - \bar{x}\bar{\lambda}, \\ w &\leq \underline{\lambda}x + \bar{x}\lambda - \bar{x}\underline{\lambda}, \\ w &\leq \bar{\lambda}x + \underline{x}\lambda - \underline{x}\bar{\lambda} \end{aligned}$$

are valid inequalities, i.e., they are satisfied for all feasible x and λ. It can also be shown that these four linear inequalities are the tightest linear relaxation that one can obtain.

In many mixed-integer linear bilevel problems, one faces upper-level variables x that are binary, i.e., $x \in \mathbb{Z}$ with $\underline{x} = 0$ and $\bar{x} = 1$. Then, the above inequalities read

$$\begin{aligned} w &\geq \underline{\lambda}x, \\ w &\geq \bar{\lambda}x + \lambda - \bar{\lambda}, \\ w &\leq \underline{\lambda}x + \lambda - \underline{\lambda}, \\ w &\leq \bar{\lambda}x. \end{aligned}$$

It is easy to see that $x = 0$ implies $w = 0$ and that $x = 1$ implies $w = \lambda \in [\underline{\lambda}, \bar{\lambda}]$. Hence, if one of the two variables is binary and if the other variable is bounded, the McCormick inequalities do not only lead to a relaxation but to an exact linear reformulation of the originally nonlinear constraints.

Exercise 3.3 Consider the LP-QP bilevel problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \arg\min_{\bar{y}} \left\{ \frac{1}{2}\bar{y}^\top Q\bar{y} + d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \end{aligned} \tag{3.8}$$

with $c_x \in \mathbb{R}^{n_x}$, $c_y, d \in \mathbb{R}^{n_y}$, $A \in \mathbb{R}^{m \times n_x}$, $B \in \mathbb{R}^{m \times n_y}$, $a \in \mathbb{R}^m$, $C \in \mathbb{R}^{\ell \times n_x}$, $D \in \mathbb{R}^{\ell \times n_y}$, $b \in \mathbb{R}^\ell$, and $Q \in \mathbb{R}^{n_y \times n_y}$ being symmetric and positive semi-definite.

(i) Derive the strong-duality based reformulation of Problem (3.8).
(ii) Assume that $\lambda \in \mathbb{R}^\ell$ are the dual variables associated with the lower-level constraints and let $\bar{\lambda}$ be an upper bound on these variables. Moreover,

assume that the leader's variables x are binary. Use the McCormick inequalities to reformulate the problem derived in (i) as a mixed-integer convex quadratic problem. Explain why it is an exact reformulation and not just a relaxation.

3.4 A Mixed-Integer Linear Reformulation of the KKT Reformulation

Taking a closer look at the KKT reformulation (3.5) and the strong-duality reformulation (3.7), we see that both problems are "almost" linear. Only the KKT complementarity conditions in (3.5d) as well as the strong-duality inequality in (3.7d) introduce nonlinearities and nonconvexities. One advantage of the strong-duality reformulation is that the nonlinearity only appears in a single constraint, whereas we have as many KKT complementarity conditions in the KKT reformulation as we have lower-level inequality constraints. However, the KKT complementarity conditions have a special structure that allows for further reformulation. The latter will then allow to solve the resulting problem using standard software for mixed-integer linear optimization.

The reasons for the nonconvexity caused by (3.5d) are the bilinear products of the lower-level dual variables λ_i and the upper-level primal variables x in the term

$$\lambda_i C_{i\cdot} x$$

and the bilinear products of the lower-level dual variables λ_i and the lower-level primal variables y in the term

$$\lambda_i D_{i\cdot} y.$$

We can linearize these terms if we exploit the combinatorial structure of the KKT complementarity conditions in (3.5d). The key idea here is to consider the complementarity conditions $\lambda_i(C_{i\cdot}x + D_{i\cdot}y - b_i) = 0$, $i = 1, \ldots, \ell$, as disjunctions stating that either

$$\lambda_i = 0 \quad \text{or} \quad C_{i\cdot}x + D_{i\cdot}y = b_i$$

needs to hold. These two cases can be modeled using binary variables

$$z_i \in \{0, 1\} \quad \text{for all } i = 1, \ldots, \ell$$

in the following mixed-integer linear way:

$$\lambda_i \le M z_i, \quad C_{i\cdot}x + D_{i\cdot}y - b_i \le M(1 - z_i)$$

with M being a sufficiently large constant. Consequently, $z_i = 1$ models the case in which the primal inequality is active, whereas $z_i = 0$ models the case in which the dual variable is zero. By construction, we thus obtain the following result.

Theorem 3.4 *Suppose that M is a sufficiently large constant. Then, Problem (3.5) is equivalent to the mixed-integer linear optimization problem (MILP)*

$$\begin{aligned} \min_{x,y,\lambda,z} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a,\ Cx + Dy \geq b, \\ & D^\top \lambda = d,\ \lambda \geq 0, \\ & \lambda_i \leq M z_i \quad \text{for all } i = 1, \dots, \ell, \\ & C_{i\cdot} x + D_{i\cdot} y - b_i \leq M(1 - z_i) \quad \text{for all } i = 1, \dots, \ell, \\ & z_i \in \{0, 1\} \quad \text{for all } i = 1, \dots, \ell. \end{aligned} \tag{3.9}$$

Here, "equivalence" means that for every globally optimal solution (x, y, λ) to Problem (3.5), there exists a vector z so that (x, y, λ, z) is a globally optimal solution to Problem (3.9). Conversely, for every globally optimal solution (x, y, λ, z) to Problem (3.9), (x, y, λ) is a globally optimal solution to Problem (3.5).

Due to theorems like the last one we use the wording "single-level reformulation", which should not be confused with the wording "single-level relaxation" from Definition 1.15. The latter refers to a relaxation because the lower-level optimality is omitted, whereas in Problem (3.9) it is actually reformulated.

The MILP reformulation (3.9) can be solved using general-purpose MILP solvers such as

- `Cbc` (`https://github.com/coin-or/Cbc`),
- `CPLEX` (`https://www.ibm.com/de-de/products/ilog-cplex-optimization-studio`),
- `FICO Xpress` (`https://www.fico.com/de/products/fico-xpress-optimization`),
- `GLPK` (`https://www.gnu.org/software/glpk`),
- `Gurobi` (`https://www.gurobi.com`),
- `HiGHS` (`https://highs.dev`),
- `MOSEK` (`https://www.mosek.com`),
- `SCIP` (`https://scipopt.org`),
- `Symphony` (`https://www.coin-or.org/SYMPHONY/index.htm`).

Some of these solvers are commercial software products but often free to use in academic contexts. Some others are open-source software projects. You can find all necessary information on the websites of the solvers.

The so-called big-M reformulation (3.9) has the severe disadvantage that one needs to determine a so-called big-M constant, which is valid for both the primal constraints as well as for the dual variables. Here, validity means that the constant is large enough so that not all primal-dual solutions are cut off that belong to an optimistic bilevel solution. The primal validity is usually ensured by the assumption that the single-level relaxation is bounded, which is often justified in practical applications. However, the dual feasible set is unbounded for bounded primal feasible sets; see Clark (1961) and Williams (1970). Thus, it is rather problematic to bound the dual variables of the follower. On the other hand, the big-M value should also not be unnecessarily large to avoid numerical troubles during the solution process of the resulting MILP. Hence, the value should be chosen as large as necessary but as small as possible. This can often be achieved by not choosing one single value for all primal constraints and all dual variables at the same time but choosing different specific big-M values for every primal constraint and different specific big-M values for every dual variable. Starting from such a reformulation with many different big-Ms, we can then always go back to the formulation in (3.9) by simply using M in the theorem as the maximum over all those values.

In practice, often "standard" values such as magic constants like 10^6 are used without any theoretical justification, or heuristics are applied to compute a big-M value. For instance, in Pineda et al. (2018), big-M values are determined from local solutions to the MPCC (3.5). However, this has to be done with great care. In Pineda and Morales (2019), it is shown using an illustrative counter-example that such heuristics may deliver invalid values, i.e., they may cut off (some of) the primal-dual solutions. Buchheim (2023) showed that computing valid big-M values for LP-LP bilevel problems can be done in polynomial time and that the respective values are also of polynomial size (in terms of the input data of the problem). However, these values are still too large to be of any practical use. Moreover, validating the correctness of a given big-M, which corresponds to computing the tightest big-M, is shown to be NP-hard in general in Kleinert et al. (2020). Let us finally note here that modern solvers for mixed-integer linear optimization problems also provide the possibility of using so-called SOS1 conditions. These can be used to tackle KKT complementarity conditions without choosing big-M values. We discuss this approach in more detail in Section 5.2.

Exercise 3.5 Consider the LP-QP bilevel problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a \\ & y \in \arg\min_{\bar{y}} \left\{ \frac{1}{2} \bar{y}^\top Q \bar{y} + d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \end{aligned}$$

with $c_x \in \mathbb{R}^{n_x}$, $c_y, d \in \mathbb{R}^{n_y}$, $A \in \mathbb{R}^{m \times n_x}$, $B \in \mathbb{R}^{m \times n_y}$, $a \in \mathbb{R}^m$, $C \in \mathbb{R}^{\ell \times n_x}$, $D \in \mathbb{R}^{\ell \times n_y}$, $b \in \mathbb{R}^\ell$, and $Q \in \mathbb{R}^{n_y \times n_y}$ being symmetric and positive semi-definite.

(i) Derive the KKT reformulation of the given LP-QP bilevel problem. What does qualitatively change with respect to the LP-LP case? (*Hint:* What about the dual polyhedron of the lower level?)
(ii) Derive the mixed-integer linear reformulation of the KKT reformulation from (i).

Exercise 3.6 Read the paper "Solving Linear Bilevel Problems Using Big-Ms: Not All That Glitters Is Gold" (Pineda and Morales 2019).

Exercise 3.7 Read the paper "There's No Free Lunch: On the Hardness of Choosing a Correct Big-M in Bilevel Optimization" (Kleinert et al. 2020).

3.5 Excursus: How to Solve a Mixed-Integer Linear Problem?

We have seen that we can re-state the LP-LP bilevel problem as a mixed-integer linear optimization problem if we can find finite but sufficiently large big-M constants. Let us assume that this is possible. How do we then solve the resulting problem?

It is easy! Download a mixed-integer optimization solver, e.g., `Gurobi` at

https://www.gurobi.com.

Get a free academic license and start to code. But how? Here, we discuss how to do this using the programming language `Python`.[1] Let us how this works with an example in which we consider the following single-level mixed-integer

[1] If you do not have it installed on your computer, go to https://www.python.org and download it. It is for free.

linear problem

$$
\begin{aligned}
\max_{x,y,z} \quad & x + y + 2z \\
\text{s.t.} \quad & x + 2y + 3z \le 4, \\
& x + y \ge 1, \\
& x, y, z \in \{0, 1\},
\end{aligned}
$$

which is taken from the Gurobi documentation. The following Python code both models and solves this MILP:

```
##!/usr/bin/python3

# This example is a modified model taken from
# https://tinyurl.com/9xnsx2uz

import gurobipy as gp
from gurobipy import GRB

# Create a new model
model = gp.Model("my-milp")

# Create variables
x = model.addVar(vtype=GRB.BINARY, name="x")
y = model.addVar(vtype=GRB.BINARY, name="y")
z = model.addVar(vtype=GRB.BINARY, name="z")

# Set objective
model.setObjective(x + y + 2 * z, GRB.MAXIMIZE)

# Add constraint: x + 2 y + 3 z <= 4
model.addConstr(x + 2 * y + 3 * z <= 4, "c0")

# Add constraint: x + y >= 1
model.addConstr(x + y >= 1, "c1")

# Optimize model
model.optimize()

for v in model.getVars():
    print("variable " + v.varName + ": " + str(v.x))

print("Objective value: " + str(model.objVal))
```

This is the output that you should get:

```
Academic license - for non-commercial use only
Optimize a model with 2 rows, 3 columns and 5 nonzeros
Variable types: 0 continuous, 3 integer (3 binary)
Coefficient statistics:
  Matrix range     [1e+00, 3e+00]
```

```
  Objective range   [1e+00, 2e+00]
  Bounds range      [1e+00, 1e+00]
  RHS range         [1e+00, 4e+00]
Found heuristic solution: objective 2.0000000
Presolve removed 2 rows and 3 columns
Presolve time: 0.00s
Presolve: All rows and columns removed

Explored 0 nodes (0 simplex iterations) in 0.00
    seconds
Thread count was 1 (of 8 available processors)

Solution count 2: 3 2

Optimal solution found (tolerance 1.00e-04)
Best objective 3.000000000000e+00,
   best bound 3.000000000000e+00, gap 0.0000%
variable x: 1.0
variable y: 0.0
variable z: 1.0
Objective value: 3.0
```

Exercise 3.8 Consider the linear bilevel problem

$$\begin{aligned} \min_{x,y} \quad & x - y \\ \text{s.t.} \quad & x - 2y \geq -8, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{3.10}$$

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

$$\begin{aligned} \min_{y} \quad & y \\ \text{s.t.} \quad & x + y \geq 7, \\ & -x + 5y \geq 2, \\ & -x + 2y \geq -4, \\ & -x - y \geq -13. \end{aligned}$$

(i) Derive the KKT reformulation of Problem (3.10).
(ii) Derive the mixed-integer linear reformulation of the KKT reformulation.
(iii) Determine sufficiently large big-M constants for the reformulation in (ii). (*Hint:* For instance, you can use Fourier–Motzkin elimination to determine valid variable bounds, which can then be used to compute sufficiently large big-M constants.)
(iv) Use a general-purpose MILP solver to solve the mixed-integer linear reformulation of the KKT reformulation of Problem (3.10) using the big-M constants determined in (iii).

(v) Derive the strong-duality based reformulation of Problem (3.10).
(vi) Use a general-purpose solver to solve the strong-duality based reformulation derived in (v).
(vii) Elaborate on possible advantages and/or disadvantages of the considered approaches. Which approach should be preferred and why?

3.6 The KKT Reformulation for Convex Lower-Level Problems

We have stated the KKT reformulation (3.5) of the LP-LP bilevel problem (3.3) but we have not yet discussed the relationship between their solutions if, in a more general setting, a nonlinear but convex lower-level problem is considered. This topic is a bit more delicate as one might think at a first glance. It turns out that the equivalence of these problem depends on whether globally or locally optimal solutions are considered and on the satisfaction of constraint qualifications.

To shed some more light on these aspects, we consider the bilevel problem

$$\begin{aligned} \min_{x,y} \quad & F(x,y) \\ \text{s.t.} \quad & G(x) \geq 0, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{3.11}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized convex problem

$$\begin{aligned} \min_{y} \quad & f(x,y) \\ \text{s.t.} \quad & g(x,y) \geq 0. \end{aligned} \tag{3.12}$$

To this end, we assume that $y \mapsto f(x,y)$ is a smooth and convex function and that $y \mapsto g_i(x,y)$, $i = 1, \dots, \ell$, are smooth and concave functions for all x with $G(x) \geq 0$, i.e., for all feasible leader's decisions. This means that the lower-level problem is indeed an x-parametric convex problem. Please note further that we simplified the upper-level problem a bit so that, for the ease of presentation, no coupling constraints are present anymore.

In what follows, we need Slater's constraint qualification for the lower level. For the ease of presentation, we assume for what follows that all constraints g_i, $i = 1, \dots, \ell$, are nonlinear.

Definition 3.9 (Slater's Constraint Qualification for the Lower Level) For a given upper-level feasible point x of the bilevel problem (3.11), we say that

Slater's constraint qualification holds for the lower-level problem (3.12) if there exists a point $\hat{y} = \hat{y}(x)$ with $g_i(x, \hat{y}(x)) > 0$ for all $i = 1, \ldots, \ell$.

Assumption 3.10 For all x with $G(x) \geq 0$ decided on by the leader, the lower-level problem (3.12) satisfies Slater's constraint qualification.

Under Assumption 3.10, we can reformulate the bilevel problem by replacing the lower level with its necessary and sufficient KKT conditions. Hence, we obtain the single-level reformulation

$$\begin{aligned} \min_{x,y,\lambda} \quad & F(x, y) \\ \text{s.t.} \quad & G(x) \geq 0, \\ & \nabla_y \mathcal{L}(x, y, \lambda) = 0, \\ & g(x, y) \geq 0, \\ & \lambda \geq 0, \\ & \lambda^\top g(x, y) = 0. \end{aligned} \tag{3.13}$$

Here,

$$\nabla_y \mathcal{L}(x, y, \lambda) = \nabla_y f(x, y) - \sum_{i=1}^{\ell} \lambda_i \nabla_y g_i(x, y)$$

is the gradient of the lower level's Lagrangian function w.r.t. y.

Let us now shed some light on the relation between globally optimal solutions to the bilevel problem and globally optimal solutions to its KKT reformulation.

Theorem 3.11 (See Theorem 2.1 in Dempe and Dutta (2012)) *Let (x^*, y^*) be a globally optimal solution to the bilevel problem* (3.11) *and assume that the lower-level problem is a convex optimization problem that satisfies Slater's constraint qualification at x^*. Then, the point (x^*, y^*, λ^*) is a globally optimal solution to the single-level reformulation* (3.13) *for every*

$$\lambda^* \in \Lambda(x^*, y^*) := \left\{\lambda \in \mathbb{R}^\ell : \nabla_y \mathcal{L}(x^*, y^*, \lambda) = 0,\ \lambda \geq 0,\ \lambda^\top g(x^*, y^*) = 0\right\}.$$

Proof: Because the x^*-parameterized lower-level problem is convex and because this parametric convex problem satisfies Slater's constraint qualification for the given x^*, the KKT theorem for convex problems (Theorem A.26) implies that $\lambda^* \in \Lambda(x^*, y^*)$ holds if and only if $(x^*, y^*) \in \operatorname{gph} S$. □

The opposite direction is also true under some stronger assumptions.

Theorem 3.12 (See Theorem 2.3 in Dempe and Dutta (2012)) *Let* (x^*, y^*, λ^*) *be a globally optimal solution to Problem* (3.13) *and let the lower-level problem* (3.12) *be convex. Moreover, suppose that Assumption 3.10 holds. Then,* (x^*, y^*) *is a globally optimal solution to the bilevel problem* (3.11).

Proof: Suppose that (x^*, y^*, λ^*) is a globally optimal solution to Problem (3.13). Thus, $\Lambda(x^*, y^*) \neq \emptyset$ holds due to $\lambda^* \in \Lambda(x^*, y^*)$. Because the objective function F of (3.13) does not depend on $\lambda \in \Lambda(x^*, y^*)$, each point (x^*, y^*, λ) with $\lambda \in \Lambda(x^*, y^*)$ is a globally optimal solution to (3.11) as well.

Assume now that (x^*, y^*) is not a globally optimal solution to the bilevel problem (3.11). Then, there exists a point (x, y) with x satisfying $G(x) \geq 0$ and $y \in \mathcal{S}(x)$ such that

$$F(x, y) < F(x^*, y^*)$$

holds. Because $y \in \mathcal{S}(x)$ and because Slater's constraint qualification holds at x due to Assumption 3.10, the respective KKT theorem (see Theorem A.26) can be used and, thus, there exists a vector $\lambda \in \mathbb{R}^\ell$ of Lagrange multipliers such that

$$\begin{aligned}
\nabla_y f(x, y) - \sum_{i=1}^{\ell} \lambda_i \nabla_y g_i(x, y) &= 0, \\
\lambda^\top g(x, y) &= 0, \\
\lambda &\geq 0, \\
g(x, y) &\geq 0
\end{aligned}$$

holds. Consequently, (x, y, λ) is a feasible point for the KKT reformulation (3.13) that has a better objective function value than (x^*, y^*, λ^*). This is a contradiction to the global optimality of (x^*, y^*, λ^*) and the claim follows. □

The last two theorems state that the original bilevel optimization problem and its single-level KKT reformulation are equivalent under the assumption that Slater's constraint qualification holds for all possible x decided on by the leader (Assumption 3.10) and if globally optimal solutions are considered. The theorems so far do not give any insight on the relationship between the local minimizers of these two problems.

Before we consider these local optima, however, let us study whether the assumptions regarding Slater's constraint qualification in the last two theorems are really necessary.

Example 3.13 (See Example 2.2 in Dempe and Dutta (2012)) Let us consider the x-parameterized convex lower-level problem

$$\min_{y_1, y_2} \quad y_1 \quad \text{s.t.} \quad y_1^2 - y_2 \leq x, \; y_1^2 + y_2 \leq 0; \tag{3.14}$$

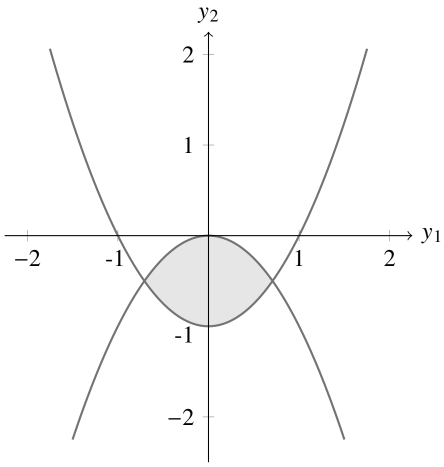


Figure 3.1 The feasible region of the x-parameterized convex lower-level problem (3.14) in Example 3.13 with $x = 1$. Taken and modified from Dempe and Dutta (2012).

see Figure 3.1 for an illustration. If $x = 0$, the only feasible point of this lower-level problem is $y = (y_1, y_2) = (0, 0)$ and, thus, Slater's constraint qualification is violated. If we consider $x \geq 0$ (this will be the upper-level constraint later), the lower-level optimal solutions are given by

$$y(x) = \begin{cases} (0,0), & \text{if } x = 0, \\ \left(-\sqrt{x/2}, -x/2\right), & \text{if } x > 0. \end{cases}$$

Some further calculations reveal that the corresponding Lagrange multipliers are given by

$$\lambda_1(x) = \lambda_2(x) = \frac{1}{4\sqrt{x/2}}$$

for $x > 0$. If $x = 0$, the problem does not satisfy Slater's constraint qualification so that the KKT conditions are not satisfied. Hence, no properly defined Lagrange multipliers exist in this case.

Consider now the bilevel problem

$$\min_{x,y} \quad x \quad \text{s.t.} \quad x \geq 0, \ y \in \mathcal{S}(x),$$

where $\mathcal{S}$ is again the solution-set mapping of the lower-level problem (3.14). The unique globally optimal solution to this bilevel problem is $x = 0$, $y = (0, 0)$

with an objective function value of 0. Moreover, there are no other (e.g., locally) optimal solutions.

Lastly, we consider the corresponding MPCC. The Lagrangian of the lower-level problem reads

$$\mathcal{L}(x, y, \lambda) = y_1 - \lambda_1(x - y_1^2 + y_2) - \lambda_2(-y_1^2 - y_2)$$

and its gradient w.r.t. y is given by

$$\nabla_y \mathcal{L}(x, y, \lambda) = \begin{pmatrix} 1 + 2\lambda_1 y_1 + 2\lambda_2 y_1 \\ -\lambda_1 + \lambda_2 \end{pmatrix}.$$

Hence, the MPCC is given by

$$\begin{aligned} \min_{x, y_1, y_2, \lambda_1, \lambda_2} \quad & x \\ \text{s.t.} \quad & x \geq 0, \\ & y_1^2 - y_2 \leq x, \ y_1^2 + y_2 \leq 0, \\ & \lambda_1 \geq 0, \ \lambda_2 \geq 0, \\ & \lambda_1(x - y_1^2 + y_2) = 0, \ \lambda_2(-y_1^2 - y_2) = 0, \\ & 1 + 2\lambda_1 y_1 + 2\lambda_2 y_1 = 0, \ -\lambda_1 + \lambda_2 = 0. \end{aligned}$$

The point $(x, y(x), \lambda(x))$ is, by construction, feasible for the MPCC for $x > 0$ and the corresponding objective function value converges to 0 for $x \to 0$. However, the problem does not possess an optimal solution because for $x = 0$, the uniquely determined lower-level solution is $y = (0, 0)$ but no feasible multipliers exist in this case. △

The take-home message is the following.

Observation 3.14 A globally optimal solution to the bilevel problem does not need to correspond to a globally optimal solution to its KKT reformulation if the lower-level problem does not satisfy Slater's constraint qualification.

This observation can also be made for the strong-duality instead of the KKT reformulation as it is mathematically equivalent and also uses the additional λ-space. The optimal-value function reformulation does, of course, not suffer from this because it is not stated in a lifted space.

We have seen that the assumption of Slater's CQ in Theorem 3.11 cannot be omitted. Next, we show that the assumptions of Theorem 3.12 are essential as well.

Example 3.15 (See Example 2.4 in Dempe and Dutta (2012)) We consider the bilevel problem

$$\min_{x,y} \quad (x-1)^2 + y^2 \quad \text{s.t.} \quad x \in \mathbb{R},\ y \in \mathcal{S}(x),$$

where $\mathcal{S}$ denotes the solution-set mapping of the x-parameterized convex lower-level problem

$$\min_{y} \quad x^2 y \quad \text{s.t.} \quad y^2 \le 0.$$

The only feasible point of the lower-level problem is $y = 0$ and it is, thus, the uniquely determined globally optimal solution to the lower-level problem (independent of the leader's decision x). In particular, this means that there exists no x for which Slater's constraint qualification holds for the lower-level problem. Because $y = 0$ is always the optimal follower's decision, the uniquely determined globally optimal solution to the bilevel problem is $(x, y) = (1, 0)$.

Let us now consider the corresponding KKT reformulation

$$\begin{aligned} \min_{x,y,\lambda} \quad & (x-1)^2 + y^2 \\ \text{s.t.} \quad & x \in \mathbb{R}, \\ & y^2 \le 0, \\ & \lambda \ge 0, \\ & \lambda y^2 = 0, \\ & x^2 + 2\lambda y = 0. \end{aligned}$$

All feasible points of this MPCC are of the form $(0, 0, \lambda)$ with $\lambda \ge 0$. Because the objective function does not depend on λ, all these points are also globally optimal solutions to the MPCC. However, none of them correspond to the optimal solution $(1, 0)$ to the bilevel problem. △

Now that we have clarified the relationship between globally optimal solutions to the bilevel problem and its KKT reformulation, we also consider the relationship of locally optimal solutions. It turns out that local minimizers of the KKT reformulation do not need to be local optima of the bilevel problem.

Example 3.16 (See Example 3.1 in Dempe and Dutta (2012)) We start by studying the x-parameterized lower-level problem

$$\begin{aligned} \min_{y_1,y_2} \quad & y_1^2 + (y_2 + 1)^2 \\ \text{s.t.} \quad & (y_1 - x_1)^2 + (y_2 - 1 - x_1)^2 \le 1, \\ & (y_1 + x_2)^2 + (y_2 - 1 - x_2)^2 \le 1. \end{aligned}$$

As usual, let $\mathcal{S}(x)$ be the set of all optimal solutions to this problem. The

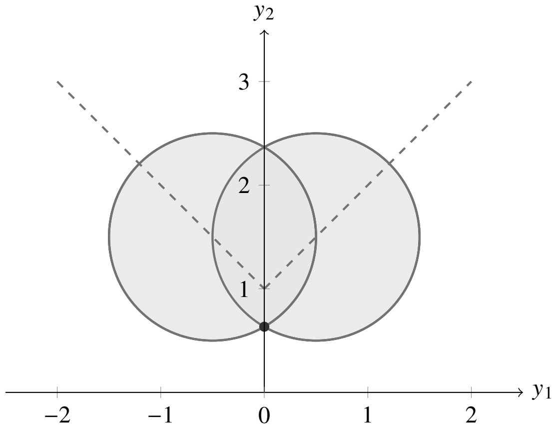


Figure 3.2 An illustration of the lower-level problem in Example 3.16 for fixed $x = (1/2, 1/2)$. The feasible set corresponds to the intersection of the two gray discs and the black dot is the optimal solution to the lower-level problem. Taken and modified from Dempe and Dutta (2012).

problem is illustrated in Figure 3.2 for the exemplary choice of $x = (1/2, 1/2)$. The feasible set is the intersection of the two x-parameterized discs and the solution to the lower-level problem is illustrated by the black dot. We also illustrate the centers of the respective discs for varying choices of x using the gray dotted lines, where the right arc belongs to x_1 and the left arc belongs to x_2. Hence, there are also choices of x_1 and x_2 so that the discs do not intersect and the lower-level problem is infeasible. The upper-level problem is given by

$$\min_{x,y=(y_1,y_2)} \quad -y_2 \quad \text{s.t.} \quad y_1 y_2 = 0,\ x \geq 0,\ y \in \mathcal{S}(x).$$

All points $y \in \mathcal{S}(x)$ with $x \geq 0$ and $y_2 \geq 0$ have a non-positive upper-level objective function value. The points with a strictly negative objective function value are those with $y_2 > 0$, which then have to satisfy $y_1 = 0$ due to upper-level feasibility. We now analyze the relationship between locally optimal solutions to the bilevel problem and its KKT reformulation.

First, let us consider $x^* = (0, 0)$ and $y^* = (0, 0)$. The point (x^*, y^*) is feasible for the bilevel problem and has the upper-level objective function value of 0. However, we now show that it is not a local minimizer of the given bilevel problem. To this end, we construct a sequence of bilevel-feasible points that converge to (x^*, y^*) but for which every member of the sequence has an upper-level objective function value strictly smaller than 0. We first define the lower-level sequence $(y_1^k, y_2^k)_k = (0, y_2^k)_k$ with $y_2^k \searrow 0$. This implies that

$x_1^k = x_2^k$ holds for all k because, otherwise, the bottom intersection point of the boundaries of the two discs would be left or right from the y_2-axis and, thus, $y_1^k \neq 0$ would need to hold. We can choose $(x_1^k, x_2^k)_k$ so that $x_1^k \searrow 0$ and $x_2^k \searrow 0$ holds. Hence, the constructed sequence $(x_1^k, x_2^k, 0, y_2^k)_k$ with $x_1^k = x_2^k$ and $x_1^k, x_2^k, y_2^k \searrow 0$ is bilevel feasible and the upper-level objective function value is strictly negative for all k. This shows that its limit (x^*, y^*) with $x^* = (0, 0)$ and $y^* = (0, 0)$ is not a local minimizer.

Second, we now study the KKT reformulation of the problem, which is given by

$$
\begin{aligned}
\min_{x,y,\lambda} \quad & -y_2 && (3.15a)\\
\text{s.t.} \quad & y_1 y_2 = 0, && (3.15b)\\
& x \geq 0, && (3.15c)\\
& (y_1 - x_1)^2 + (y_2 - 1 - x_1)^2 \leq 1, && (3.15d)\\
& (y_1 + x_2)^2 + (y_2 - 1 - x_2)^2 \leq 1, && (3.15e)\\
& \lambda_1 \geq 0,\ \lambda_2 \geq 0, && (3.15f)\\
& 2y_1 + 2\lambda_1(y_1 - x_1) + 2\lambda_2(y_1 + x_2) = 0, && (3.15g)\\
& 2(y_2 + 1) + 2\lambda_1(y_2 - 1 - x_1) + 2\lambda_2(y_2 - 1 - x_2) = 0, && (3.15h)\\
& \lambda_1(1 - (y_1 - x_1)^2 - (y_2 - 1 - x_1)^2) = 0, && (3.15i)\\
& \lambda_2(1 - (y_1 + x_2)^2 - (y_2 - 1 - x_2)^2) = 0. && (3.15j)
\end{aligned}
$$

Let us again consider points with $y_2 > 0$. Thus, $y_1 = 0$ still needs to hold due to (3.15b) and $x_1 = x_2 > 0$ is feasible, too. The partial derivative of the Lagrangian function w.r.t. y_1 in (3.15g) then leads to $\lambda_1 = \lambda_2$. For the point (x^*, y^*) with $x^* = (0, 0)$ and $y^* = (0, 0)$, we obtain $\lambda_1^* + \lambda_2^* = 1$ from the second partial derivative of the Lagrangian function in (3.15h).

Third, based on the gained insights, we now define a sequence (x^k, y^k, λ^k) that is feasible for the KKT reformulation (3.15) and that converges to $(x^*, y^*, (1/2, 1/2))$ with $x^* = (0, 0)$ and $y^* = (0, 0)$. For $(x^k, y^k)_k$, we take the same sequence as above and extend it by the constant sequence $(\lambda_1^k, \lambda_2^k) = (1/2, 1/2)$ for all k with limit $(\lambda_1^*, \lambda_2^*) = (1/2, 1/2)$. Hence, (x^*, y^*, λ^*) is not a local minimizer of the KKT reformulation (3.15) because the feasible sequence (x^k, y^k, λ^k) converging to it has strictly smaller upper-level objective function values.

Fourth and finally, we see that all other points $(x^*, y^*, \tilde{\lambda})$ with $\tilde{\lambda} \in \Lambda(x^*, y^*)$ are local minimizers of the KKT reformulation. To this end, note that $\Lambda(x^*, y^*) = \Lambda((0, 0), (0, 0)) = \{(\lambda_1, \lambda_2) \geq 0 \colon \lambda_1 + \lambda_2 = 1\}$ and that there cannot be any feasible sequence (x^k, y^k, λ^k) satisfying $y_2^k \searrow 0$ with, e.g., λ^k converging to

$\lambda^* = (1/4, 3/4)$. The reason is that, for sufficiently large k, $\lambda_1^k < \lambda_2^k$ needs to hold. However, just below (3.15), we have shown that for $y_2^k > 0$, we necessarily obtain $\lambda_1^k = \lambda_2^k$; a contradiction.

To sum up, this means that (x^*, y^*) is not a local minimizer of the bilevel problem but that the KKT reformulation has local minimizers that correspond to it. Hence, solving the KKT reformulation for a local minimizer does not necessarily lead to a local minimizer of the bilevel problem. △

Let us summarize the main result of the last example again as an observation.

Observation 3.17 The local minimizers of the KKT reformulation do not need to correspond to local optima of the bilevel problem.

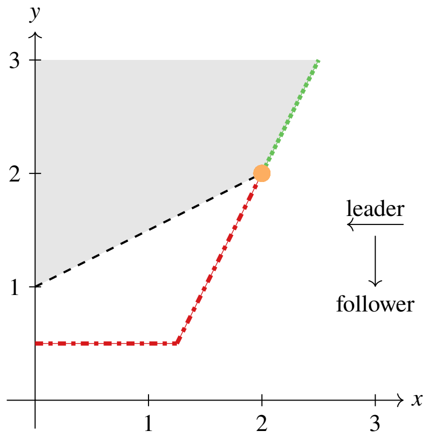
3.7 Independence of Irrelevant Constraints

In single-level optimization, a constraint that is inactive in a solution can be tightened without changing the optimal solution. This is the so-called *independence of irrelevant constraints (IIC) property*. We show in this section that the IIC property does not hold for bilevel optimization problems in general.

Example 3.18 (See Kleinert et al. (2021c)) Let us consider the linear bilevel problem

$$\begin{aligned} \min_{x,y} \quad & x \\ \text{s.t.} \quad & y \geq 0.5x + 1, \ x \geq 0, \\ & y \in \arg\min_{\bar{y}} \{\bar{y} : \bar{y} \geq 2x - 2, \ \bar{y} \geq 0.5\}; \end{aligned} \tag{3.16}$$

see Figure 3.3 (top). The feasible region of the single-level relaxation is the gray area, the bilevel-feasible set is given by the dotted green line, and the bilevel optimal solution $(2, 2)$ is shown in orange. All optimal solutions to the follower's problem that are infeasible for the leader's coupling constraint (the black dashed line) are shown by the dash-dotted red lines. In computational single-level optimization, one is frequently interested in obtaining a formulation of the problem at hand that is as tight as possible. One classic technique for this is *bound strengthening*, which aims at getting variable bounds that are as tight as possible but without changing the set of optimal solutions to the given problem. When strengthening the bound $\bar{y} \geq 0.5$ in the lower-level problem using the constraint $y \geq 0.5x + 1$ of the upper-level problem, one finds that the minimum value of $0.5x + 1$ is 1 due to $x \geq 0$, which increases the bound of $\bar{y}$ to



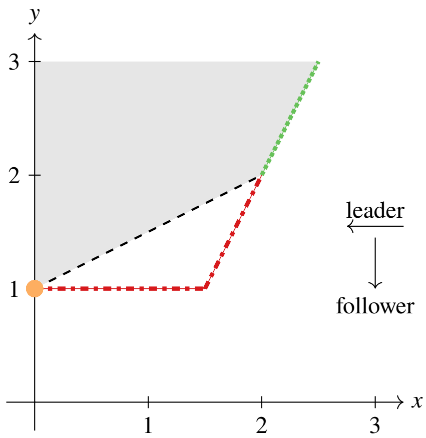


Figure 3.3 The bilevel-feasible sets (dotted green lines) and the optimal solutions $(2, 2)$ and $(0, 1)$, respectively, to the bilevel problem in Example 3.18 without (top) and with (bottom) bound tightening applied (orange dots). Taken and modified from Kleinert et al. (2021c).

$\bar{y} \geq 1$. This yields the modified problem

$$\begin{aligned} \min_{x,y} \quad & x \\ \text{s.t.} \quad & y \geq 0.5x + 1, \ x \geq 0, \\ & y \in \arg\min_{\bar{y}} \{\bar{y} : \bar{y} \geq 2x - 2, \ \bar{y} \geq 1\}, \end{aligned}$$

which has the optimal solution $(0, 1) \neq (2, 2)$; see Figure 3.3 (bottom). See also the thesis by Manns (2020) for further examples.

What have we seen now? We first solved a bilevel problem and then tightened a constraint in the lower-level problem. This constraint was not active in the original solution to the bilevel problem but changes the optimal solution if it is tightened. Note that this cannot happen in single-level optimization. △

In what follows, we formally define the IIC property for bilevel problems, which has been originally done in the literature by Macal and Hurter (1997). Roughly speaking, it means that if there exists an additional constraint that is satisfied in all solutions to the bilevel problem, then adding the constraint to the lower-level problem does not change the set of optimal solutions.

Definition 3.19 (Independence of Irrelevant Constraints (IIC)) Consider the bilevel problem

$$\begin{aligned} \min_{x \in X, y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & y \in \arg\min_{\bar{y} \in Y} \{ f(x, \bar{y}) \colon g(x, \bar{y}) \geq 0 \} \end{aligned} \tag{3.17}$$

and let S be the set of optimal solutions to Problem (3.17).

(i) Consider an additional lower-level constraint $h(x, y) \geq 0$ with $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}$ such that $h(x^*, y^*) \geq 0$ holds for all $(x^*, y^*) \in S$. Let $\tilde{S}$ be the set of optimal solutions to the bilevel problem obtained from Problem (3.17) by adding the constraint $h(x, y) \geq 0$ to the lower-level problem. Suppose now that every solution $(x^*, y^*) \in S$ also satisfies $(x^*, y^*) \in \tilde{S}$. Then, we say that Problem (3.17) is *independent of irrelevant constraint* $h(x, y) \geq 0$.

(ii) We say that Problem (3.17) is *independent of irrelevant constraints* if Part (i) holds for all possible additional lower-level constraints $h(x, y) \geq 0$ that are satisfied at all solutions $(x^*, y^*) \in S$.

Example 3.18 shows that the bilevel problem (3.16) is not independent of the irrelevant constraint $h(x, y) = y - 1 \geq 0$ but that it is, e.g., independent of the irrelevant constraint $\tilde{h}(x, y) = y - 0.75 \geq 0$. Because the IIC is only satisfied if the problem is independent of all irrelevant constraints, the property fails to hold for Problem (3.16).

In Macal and Hurter (1997), it is shown that only those bilevel problems for which every optimal solution to the single-level relaxation is also an optimal solution to the original bilevel problem possess the IIC property. Consequently, most bilevel problems lack the IIC property.

We now present a characterization that allows us to determine whether a bilevel problem possesses the IIC property.

Theorem 3.20 *Suppose that the following assumptions hold.*

(i) The lower-level problem is continuous, i.e., $Y = \mathbb{R}^{n_y}$.

(ii) The lower-level objective function $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}$ *and the lower-level constraint functions* $g_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}$, $i \in \{1, \ldots, \ell\}$, *are continuously differentiable.*

(iii) For every optimal solution $(\hat{x}, \hat{y})$ *to the single-level relaxation of the bilevel problem*

$$\min_{x \in X, y} \quad F(x, y) \tag{3.18a}$$

$$\text{s.t.} \quad G(x, y) \geq 0, \tag{3.18b}$$

$$y \in \operatorname*{arg\,min}_{\bar{y} \in Y} \{ f(x, \bar{y}) : g(x, \bar{y}) \geq 0 \}, \tag{3.18c}$$

the $\hat{x}$*-parameterized lower-level problem (3.18c) is convex and satisfies Slater's constraint qualification, i.e., there exists a point* $\tilde{y} = \tilde{y}(\hat{x}) \in Y$ *with* $g_i(\hat{x}, \tilde{y}(\hat{x})) > 0$ *for all* $i \in \{1, \ldots, \ell\}$.

Further, let $(\hat{x}, \hat{y})$ *be an optimal solution to the single-level relaxation of Problem (3.18), and let* $\mathcal{A}(\hat{x}, \hat{y})$ *denote the set of lower-level inequality constraints that are active in* $(\hat{x}, \hat{y})$. *Then, Problem (3.18) is independent of irrelevant constraints if there exist* $\lambda_i \in \mathbb{R}_{\geq 0}$ *for all* $i \in \mathcal{A}(\hat{x}, \hat{y})$ *such that*

$$\nabla_y f(\hat{x}, \hat{y}) - \sum_{i \in \mathcal{A}(\hat{x}, \hat{y})} \lambda_i \nabla_y g_i(\hat{x}, \hat{y}) = 0 \tag{3.19}$$

holds.

Proof: Let $(\hat{x}, \hat{y})$ be an optimal solution to the single-level relaxation of Problem (3.18) and let $\lambda_i \in \mathbb{R}_{\geq 0}$ for all $i \in \mathcal{A}(\hat{x}, \hat{y})$ be such that (3.19) is satisfied. For all $i \in \{1, \ldots, \ell\}$, we now set

$$\hat{\lambda}_i := \begin{cases} \lambda_i, & i \in \mathcal{A}(\hat{x}, \hat{y}), \\ 0, & \text{otherwise.} \end{cases}$$

By construction, we have

$$(\hat{x}, \hat{y}) \in X \times Y, \quad G(\hat{x}, \hat{y}) \geq 0, \quad g(\hat{x}, \hat{y}) \geq 0, \quad \text{and} \quad \hat{\lambda} \geq 0.$$

Because $g_i(\hat{x}, \hat{y}) = 0$ holds for all $i \in \mathcal{A}(\hat{x}, \hat{y})$ and $\hat{\lambda}_i = 0$ holds for all $i \in \bar{\mathcal{A}}(\hat{x}, \hat{y}) := \{1, \ldots, \ell\} \setminus \mathcal{A}(\hat{x}, \hat{y})$, we obtain

$$\hat{\lambda}^\top g(\hat{x}, \hat{y}) = \sum_{i=1}^{\ell} \hat{\lambda}_i g_i(\hat{x}, \hat{y}) = \sum_{i \in \mathcal{A}(\hat{x}, \hat{y})} \hat{\lambda}_i g_i(\hat{x}, \hat{y}) + \sum_{i \in \bar{\mathcal{A}}(\hat{x}, \hat{y})} \hat{\lambda}_i g_i(\hat{x}, \hat{y}) = 0.$$

Equality (3.19) further yields

$$\begin{aligned}\nabla_y \mathcal{L}(\hat{x}, \hat{y}, \hat{\lambda}) &= \nabla_y f(\hat{x}, \hat{y}) - \sum_{i=1}^{\ell} \hat{\lambda}_i \nabla_y g_i(\hat{x}, \hat{y}) \\ &= \nabla_y f(\hat{x}, \hat{y}) - \sum_{i \in \mathcal{A}(\hat{x}, \hat{y})} \lambda_i \nabla_y g_i(\hat{x}, \hat{y}) \\ &= 0.\end{aligned}$$

Taking all previous considerations into account, we obtain that $(\hat{y}, \hat{\lambda})$ is a KKT point of the $\hat{x}$-parameterized lower-level problem. By assumption, this problem is convex and satisfies Slater's CQ, i.e., the KKT conditions are necessary and sufficient optimality conditions. This implies that $\hat{y}$ is an optimal solution to the $\hat{x}$-parameterized lower-level problem and, thus, $(\hat{x}, \hat{y})$ is bilevel feasible. Because $(\hat{x}, \hat{y})$ is a solution to the single-level relaxation of Problem (3.18), which is a relaxation of the original bilevel problem, this implies that $(\hat{x}, \hat{y})$ is an optimal solution to the bilevel problem (3.18) as well.

Let now $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \to \mathbb{R}$ with $h(\hat{x}, \hat{y}) \geq 0$ be given arbitrarily. The augmented lower-level problem

$$\min_{y \in Y} \quad f(\hat{x}, y) \quad \text{s.t.} \quad g(\hat{x}, y) \geq 0, \; h(\hat{x}, y) \geq 0, \tag{3.20}$$

is a restriction of the original $\hat{x}$-parameterized lower level and $\hat{y}$ is feasible for Problem (3.20). Hence, $\hat{y}$ also solves Problem (3.20). As a result, $(\hat{x}, \hat{y})$ solves the corresponding augmented bilevel problem in which the original lower-level problem (3.18c) is replaced with the problem

$$\min_{y \in Y} \quad f(x, y) \quad \text{s.t.} \quad g(x, y) \geq 0, \; h(x, y) \geq 0.$$

Because h was chosen arbitrarily, this implies that the bilevel problem satisfies the IIC property. □

Remark 3.21 (i) Note that we do not need to impose any assumption such as concavity or differentiability for the function h.

(ii) The condition in (3.19) is a sufficient condition that ensures that the considered solution to the single-level relaxation is a solution to the given bilevel problem as well. This sufficient condition uses the convexity properties of the original problem and can be checked without actually solving the overall bilevel problem.

(iii) The convexity assumptions can even be weakened and one can show that the IIC property holds if any solution to the single-level relaxation is a solution to the given bilevel problem as well.

(iv) The result of Theorem 3.20 also holds if X involves discrete decisions.

Exercise 3.22 Consider the linear bilevel problem

$$\begin{aligned} \min_{x,y} \quad & x + 3y \\ \text{s.t.} \quad & 1 \le x \le 5, \\ & y \in \arg\min_{\bar{y}} \left\{ -\bar{y} \colon x + \bar{y} \ge 3,\ 0 \le \bar{y} \le 3 \right\}. \end{aligned} \tag{3.21}$$

(i) Use a general-purpose LP solver to compute the solution to the single-level relaxation of Problem (3.21).

(ii) Check if Problem (3.21) possesses the IIC property using Theorem 3.20.

(iii) Without explicitly determining an optimal solution to the bilevel problem (3.21), what can you say about the relation between an optimal solution to Problem (3.21) and the solution to its single-level relaxation?

(iv) Consider the augmented bilevel problem, in which the inequality

$$-2x - y \ge -6$$

is added to the lower level of (3.21). Is the augmented bilevel problem independent of irrelevant constraints? What can you say about the relation between an optimal solution to the augmented bilevel problem and the solution to its single-level relaxation?

(v) What can you say about the relation between the set of optimal solutions S to Problem (3.21) and the set of optimal solutions $\tilde{S}$ to the augmented bilevel problem?

3.8 What You Should Know Now!

1. How many single-level reformulations of a bilevel problem do you know?
2. How does the single-level reformulation using the optimal-value function look like?
3. What is the problem (in general) with the single-level reformulation using the optimal-value function?
4. How does a general LP-LP bilevel problem look like?
5. Why did we omit the linear term in x in the lower-level objective function of the LP-LP bilevel problem?
6. Can you derive the KKT reformulation of the LP-LP bilevel problem?
7. What makes this KKT reformulation hard to solve?
8. Is the KKT reformulation of an LP-LP bilevel problem an LP again?
9. What are the nonlinear constraints of the KKT reformulation of an LP-LP bilevel problem? Can we linearize these nonlinear constraints? If yes, how? What is the “price” that we have to pay for it?

10. How does the strong-duality-based single-level reformulation of an LP-LP bilevel problem look like? How is it derived?
11. What makes this strong-duality-based single-level reformulation hard to solve?
12. Why can we not linearize the nonlinearities of the strong-duality-based single-level reformulation similarly to the case of the KKT reformulation?
13. What is the relation between the strong-duality-based single-level reformulation and the KKT reformulation of an LP-LP bilevel problem?
14. What is the challenge of big-Ms?
15. How do we define Slater's CQ for the lower-level problem?
16. What is the relation between globally optimal solutions to the bilevel problem with a convex follower problem and the corresponding KKT reformulation? What assumptions are required to establish this relation?
17. What is the relation between locally optimal solutions to the bilevel problem with a convex follower problem and the corresponding KKT reformulation?
18. What is the IIC property?
19. Does the IIC property hold for bilevel optimization problems?
20. Can you provide an example of a bilevel problem in which the IIC property does not hold?

4

Linear Bilevel Problems

In this chapter, we focus on LP-LP bilevel problems of the general form

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \tag{4.1a}$$

$$\text{s.t.} \quad Ax + By \geq a, \tag{4.1b}$$

$$y \in \arg\min_{\bar{y}} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \tag{4.1c}$$

with $c_x \in \mathbb{R}^{n_x}$, $c_y, d \in \mathbb{R}^{n_y}$, $A \in \mathbb{R}^{m \times n_x}$, $B \in \mathbb{R}^{m \times n_y}$, and $a \in \mathbb{R}^m$ as well as $C \in \mathbb{R}^{\ell \times n_x}$, $D \in \mathbb{R}^{\ell \times n_y}$, and $b \in \mathbb{R}^\ell$.

Before we study the most important properties of linear bilevel optimization problems, let us consider another example.

Example 4.1 (See Kleinert et al. (2021a)) We consider the problem

$$\min_{x,y} \quad y \quad \text{s.t.} \quad 0 \leq x \leq 2, \ y \in \mathcal{S}(x),$$

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

$$\min_{y} \quad -y \quad \text{s.t.} \quad y \geq 0, \ y \leq 1 + x, \ y \leq 3 - x.$$

The set of feasible points of the single-level relaxation is the gray area in Figure 4.1. The blue horizontal segment connecting the origin $(0, 0)$ and the point $(2, 0)$ constitutes the set of optimal solutions to the single-level relaxation, i.e., those points in Ω that minimize the upper-level objective function. Because the corresponding upper-level objective function value is 0 on this segment, this leads to a lower bound of 0 for the entire bilevel LP. The bilevel-feasible region $\mathcal{F}$ is given by the union of the two dotted green segments, i.e., by the "roof of the house". As we have seen before, $\mathcal{F}$ is nonconvex although both levels are linear optimization problems. The problem has the two optimal solutions $(0, 1)$

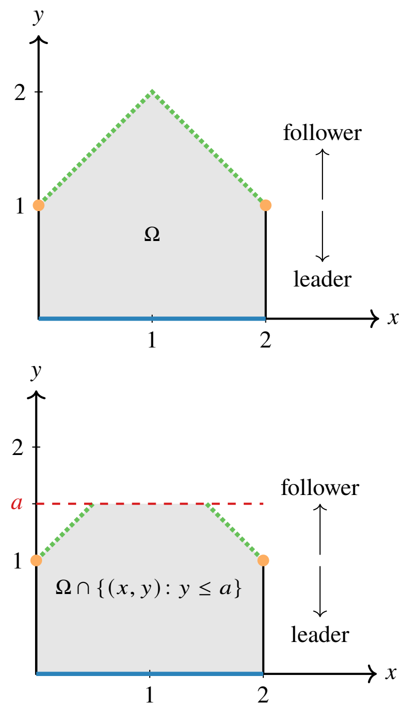


Figure 4.1 Illustration of the LP-LP bilevel problem in Example 4.1. The gray area represents the feasible region of the single-level relaxation and the dotted green lines are the bilevel-feasible set. The optimal solutions to the single-level relaxation are illustrated by the solid blue line and the bilevel optimal points are the two orange dots. The dashed red line in the bottom figure represents the coupling constraint. Taken and modified from Kleinert et al. (2021a).

and $(2, 1)$ with value 1 (orange points). This example also shows that, by moving up the "roof of the house", the solutions to the bilevel problem can be arbitrarily far away from the solutions to the single-level relaxation.

If we now add the coupling constraint $y \leq a$ with $1 < a < 2$ to the upper level, the bilevel-feasible region is reduced to two disconnected dotted green segments as depicted in Figure 4.1 (bottom). Nonetheless, these segments constitute faces of the single-level relaxation. Note that the set of optimal solutions to the bilevel problem remains unchanged. A worse situation occurs if a is set to a value in $(0, 1)$. Then, the bilevel-feasible region is empty, i.e., the bilevel LP has no feasible point, although the single-level relaxation is feasible. This last example is also useful to illustrate the effect of moving coupling constraints, i.e., upper-level constraints involving variables of the lower level, between the two levels. If, e.g., the constraint $y \leq 1/2$ is added to the lower level, then the problem becomes feasible and all points $(x, 1/2)$ with $0 \leq x \leq 2$ are bilevel optimal. The two facts that (i) coupling constraints of a bilevel LP may lead to a disconnected bilevel-feasible region and that (ii) coupling constraints cannot be moved to the lower level, in general, without changing the set of optimal solutions have also been discussed by Benson (1989) as well as Audet et al. (2006) and Mersha and Dempe (2006). The impact of coupling constraints on the set of optimal solutions is discussed in detail in the recent papers by Henke et al. (2025a,b). △

The remainder of this chapter is structured as follows. In Section 4.1, we first study the situation in which the shared constraint set is unbounded. Afterward, in Section 4.2, we study the existence of solutions to linear bilevel problems, before discussing their geometric properties in Section 4.3 and some complexity results in Section 4.4. We present algorithms for solving linear bilevel problems in the next chapter.

4.1 Unboundedness of the Shared Constraint Set

Bilevel problems are particularly difficult to analyze if the shared constraint set is unbounded. To demonstrate this, let us consider the following example.

Example 4.2 (See Example 2 in Xu and Wang (2014)) We consider the linear bilevel problem

$$\begin{aligned} \max_{x,y} \quad & x + y \\ \text{s.t.} \quad & 0 \leq x \leq 2, \\ & y \in \arg\max_{\bar{y}} \, \{d\bar{y} \colon \bar{y} \geq x\} \end{aligned}$$

with $d \in \mathbb{R}$. The shared constraint set of this problem, which is unbounded, is depicted in Figure 4.2. Note that also the single-level relaxation of the bilevel problem, i.e., the linear problem

$$\max_{x,y} \quad x + y \quad \text{s.t.} \quad 0 \leq x \leq 2, \; y \geq x,$$

is unbounded. Nevertheless, let us explore what we can conclude about the solvability of the overall bilevel problem for different values of the parameter $d \in \mathbb{R}$. For $d = 0$, the bilevel problem is unbounded as well because any feasible decision of the follower is optimal for the lower-level problem. For $d > 0$, the lower-level problem is unbounded for all decisions of the leader. Hence,

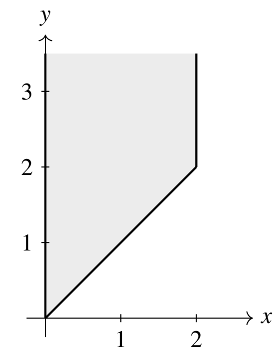


Figure 4.2 The shared constraint set (gray area) of the bilevel problem in Example 4.2.

the follower's optimality condition, i.e., $dy \geq \varphi(x) = +\infty$, renders the bilevel problem infeasible. Finally, for $d < 0$, the bilevel problem admits the unique optimal solution $(2, 2)$. △

Based on Example 4.2, we make the following observation.

Observation 4.3 If the single-level relaxation of a bilevel problem is unbounded, the bilevel problem may be infeasible, unbounded, or admit an optimal solution. In general, it is thus not possible to draw any conclusion from the unboundedness of the single-level relaxation about the solvability, the infeasibility, or the unboundedness of the bilevel problem.

Exercise 4.4 Consider Example 4.2 again but now in its pessimistic variant. Does this change, for the different choices of d, the status (infeasible, unbounded, or solvable) of the overall bilevel problem?

Example 4.2 particularly illustrates what can happen if the lower level of a bilevel problem is unbounded for some decision of the leader. More formally, we have the following result, which was first stated for mixed-integer linear bilevel problems in Lemma 2 of Xu and Wang (2014).

Lemma 4.5 *Suppose there exists a point $x \in \Omega_x$ for which the lower-level problem* (4.1c) *is unbounded. Then, the bilevel problem* (4.1) *is infeasible.*

Proof: Let $x \in \Omega_x$ be given so that the x-parameterized lower-level problem (4.1c) is unbounded. Moreover, let $y \in \mathbb{R}^{n_y}$ be feasible for the lower-level problem and the given x, i.e., $Cx + Dy \geq b$ holds. The unboundedness of the lower level then implies the existence of a direction $\Delta y \in \mathbb{R}^{n_y}$ so that

$$Cx + D(y + k\Delta y) \geq b \quad \text{and} \quad d^\top(y + k\Delta y) < d^\top y$$

holds for any $k > 0$. Hence, the vector Δy satisfies

$$d^\top \Delta y < 0 \quad \text{and} \quad D\Delta y \geq 0.$$

Note that the latter conditions on Δy do not depend on the decision x. Hence, for any $(\bar{x}, \bar{y}) \in \Omega$, the vector Δy is an improving direction for the follower. As a consequence, we have $\varphi(\bar{x}) = -\infty$ and, thus, the value-function constraint $d^\top \bar{y} \leq \varphi(\bar{x})$, which is used to ensure the optimality of the follower's response, cannot be satisfied. □

Luckily, it is possible to determine upfront whether the lower level of a bilevel problem is unbounded. To this end, we only need to solve one auxiliary linear optimization problem, which we formally state in the following theorem.

Theorem 4.6 (See Theorem 1 in Fischetti et al. (2018a)) *Suppose that the shared constraint set Ω of Problem (4.1) is non-empty. Moreover, let v^* be the optimal objective function value of the linear problem*

$$\begin{aligned} \min_{\Delta y} \quad & d^\top \Delta y \\ \text{s.t.} \quad & D\Delta y \geq 0, \\ & -1 \leq \Delta y \leq 1. \end{aligned} \tag{4.2}$$

Then, for any $x \in \Omega_x$, the following is true. If $v^* < 0$ holds, the lower-level problem is unbounded. Otherwise, it has an optimal solution.

Proof: Problem (4.2) is feasible and bounded. Hence, an optimal solution Δy^* with value v^* exists. Moreover, there exists a point $\bar{x} \in \Omega_x$ so that there exists a feasible point $\bar{y}$ for the $\bar{x}$-parameterized lower-level problem (4.1c) because $\Omega \neq \emptyset$ holds by assumption. On the one hand, if $v^* < 0$ holds, we have $d^\top \Delta y^* < 0$ and $D\Delta y^* \geq 0$. Using the insights from the proof of Lemma 4.5, this implies that Δy^* is an improving direction for the lower-level problem at the given point $(\bar{x}, \bar{y}) \in \Omega$. Hence, the lower-level problem is unbounded.

On the other hand, if $v^* \geq 0$ holds, the lower-level problem (4.1c) cannot be unbounded and, thus, it is solvable. □

By Lemma 4.5, if Problem (4.2) has an optimal solution with value $v^* < 0$, we can correctly conclude that the bilevel problem (4.1) is infeasible. To avoid such situations, one typically assumes that the shared constraint set Ω is bounded.

4.2 Existence of Solutions

We first investigate which assumptions are required in order to prove the existence of solutions. To this end, we need one more definition that we then apply to the set $\mathcal{S}(x)$ of optimal lower-level solutions. In what follows, we use the notation 2^X for the power set of a given set X.

Definition 4.7 (Polyhedral Point-to-Set Mapping) A point-to-set mapping $\Gamma : \mathbb{R}^{n_x} \to 2^{\mathbb{R}^{n_y}}$ is called *polyhedral* if its graph

$$\{(x, y) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} : y \in \Gamma(x)\}$$

is the union of a finite number of convex polyhedral sets, where a convex polyhedral set is defined as the intersection of a finite number of halfspaces.

We now consider the LP-LP bilevel problem (4.1) again and first study the rational reaction set

$$\mathcal{S}(x) = \operatorname*{arg\,min}_{y} \left\{d^\top y : Cx + Dy \geq b\right\}$$

of the follower.

Lemma 4.8 *The point-to-set mapping $\mathcal{S}$ is polyhedral.*

Proof: A point (x, y) is in the graph of $\mathcal{S}$ if and only if there exists a dual variable vector $\lambda \in \mathbb{R}^\ell$ so that the KKT conditions

$$Cx + Dy \geq b, \quad \lambda \geq 0, \quad \lambda^\top (Cx + Dy - b) = 0, \quad D^\top \lambda = d \tag{4.3}$$

are satisfied. For all possible sets $I \subseteq \{1, \dots, \ell\}$, we now consider the system

$$\begin{aligned}
(Cx + Dy - b)_i &= 0 \quad \text{for all } i \in I,\\
(Cx + Dy - b)_i &\geq 0 \quad \text{for all } i \in \{1, \dots, \ell\} \setminus I,\\
\lambda_i &\geq 0 \quad \text{for all } i \in I,\\
\lambda_i &= 0 \quad \text{for all } i \in \{1, \dots, \ell\} \setminus I,\\
D^\top \lambda &= d.
\end{aligned}$$

Let us denote the set of solutions to this system with $M(I)$. For every $I \subseteq \{1, \dots, \ell\}$ all points in $M(I)$ satisfy the KKT conditions in (4.3) and $M(I)$ is a polyhedral set in the (x, y, λ)-space. Hence, by Fourier–Motzkin elimination, the projection of $M(I)$ onto the (x, y)-space is polyhedral as well. There are $2^\ell < \infty$ (and thus finitely) many of such polyhedral sets, which completes the proof. □

We now start by proving an existence result for linear bilevel problems including coupling constraints.

Theorem 4.9 *Suppose that Problem (4.1) has a bounded shared constraint set Ω. Then, Problem (4.1) is either infeasible or admits an optimal solution.*

Proof: As in the proof of Lemma 4.8, we consider the optimization problem

$$\begin{aligned}
\min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \\
& (Cx + Dy - b)_i = 0 \quad \text{for all } i \in I, \\
& (Cx + Dy - b)_i \geq 0 \quad \text{for all } i \in \{1, \dots, \ell\} \setminus I, \\
& \lambda_i \geq 0 \quad \text{for all } i \in I, \\
& \lambda_i = 0 \quad \text{for all } i \in \{1, \dots, \ell\} \setminus I, \\
& D^\top \lambda = d,
\end{aligned}$$

for any $I \subseteq \{1, \dots, \ell\}$. For a given I, let $\mathcal{F}_I$ be the feasible set of the above problem. If $\mathcal{F}_I = \emptyset$ holds for all I, the overall bilevel problem is infeasible. Otherwise, there exists at least one $I \subseteq \{1, \dots, \ell\}$ with $\mathcal{F}_I \neq \emptyset$. Because the shared constraint set is bounded, the projection of $\mathcal{F}_I$ onto the (x, y)-space is bounded as well. Moreover, the projection is non-empty and a polyhedron due to Fourier–Motzkin elimination. Hence, the problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \operatorname{proj}_{x,y}(\mathcal{F}_I)$$

has a solution (x^I, y^I) with optimal objective function value $v^I = c_x^\top x^I + c_y^\top y^I$. Taking now

$$J \in \arg\min \left\{ v^I : I \subseteq \{1, \dots, \ell\} \text{ with } \mathcal{F}_I \neq \emptyset \right\}$$

leads to an optimal bilevel solution (x^J, y^J). □

The next result is an existence theorem for linear bilevel problems without coupling constraints.

Theorem 4.10 *Suppose that Problem (4.1) with $B = 0$ has a non-empty and bounded shared constraint set Ω. Then, Problem (4.1) with $B = 0$ admits an optimal solution.*

Proof: Because the shared constraint set of the LP-LP bilevel problem (4.1) is non-empty, there exists a point $x \in \Omega_x$ so that $\mathcal{Y}(x) = \{y \in \mathbb{R}^{n_y} : Cx + Dy \geq b\}$ is non-empty. Moreover, $\mathcal{Y}(x)$ is bounded due to the boundedness of the shared constraint set. This implies that the respective x-parameterized lower-level problem is solvable and, by the strong-duality theorem of linear optimization (Theorem A.5), the corresponding lower-level dual problem is solvable as well. Using the notation of $\mathcal{F}_I$ from the proof of the last theorem (but this time with

$B = 0$), this means that there is at least one I with $\mathcal{F}_I \neq \emptyset$. Using the same arguments as in the end of the proof of the last theorem shows the claim. □

4.3 Geometric Properties

Our goal now is to understand the geometric properties of LP-LP bilevel problems. The main source of the remainder of this section is Chapter 5.2 in the book by Bard (1998).

Theorem 4.11 *Consider the LP-LP bilevel problem* (4.1)*. Suppose that the shared constraint set* Ω *is non-empty and bounded. The bilevel-feasible set of Problem* (4.1) *can then be equivalently written as the intersection of the shared constraint set with the feasible points of a piecewise linear equality constraint. In particular, the bilevel-feasible set is a union of faces of the shared constraint set.*

This claim should not be a surprise at this point as we observed this fact already in Examples 1.20 and 4.1, where the green lines, i.e., the bilevel-feasible sets, are unions of faces of the shared constraint set. Note, however, that the intersection discussed in the theorem can also be empty.

Proof: We start by re-writing the bilevel-feasible set

$$\mathcal{F} := \{(x, y) \colon (x, y) \in \Omega,\ y \in \mathcal{S}(x)\}$$

as

$$\mathcal{F} := \left\{(x, y) \colon (x, y) \in \Omega,\ d^\top y = \min_{\bar{y}} \{d^\top \bar{y} \colon Cx + D\bar{y} \geq b\}\right\}$$

and use the optimal-value function

$$\varphi(x) = \min_y \left\{d^\top y \colon Dy \geq b - Cx\right\}$$

to further obtain

$$\mathcal{F} := \left\{(x, y) \colon (x, y) \in \Omega,\ d^\top y = \varphi(x)\right\}.$$

Because for each $x \in \Omega_x$, the feasible region of the problem

$$\min_y \quad d^\top y \quad \text{s.t.} \quad Dy \geq b - Cx$$

is non-empty and compact, the latter problem always has a solution. By using the strong-duality theorem of linear optimization (Theorem A.5), we can also express the optimal-value function by means of the dual LP as

$$\varphi(x) = \max_\lambda \left\{(b - Cx)^\top \lambda \colon D^\top \lambda = d,\ \lambda \geq 0\right\}.$$

From the classic theory of linear optimization, we know that the optimal solution to the follower's problem is always attained at one of the vertices of the feasible set, which, for the dual LP, does not depend on the leader's decision x anymore. Let $\lambda^1, \dots, \lambda^s$ be the set of all the dual polyhedron's vertices, i.e., the set of vertices of the polyhedron defined by

$$D^\top \lambda = d, \quad \lambda \geq 0. \tag{4.4}$$

Then, we can further equivalently re-write the optimal-value function as

$$\varphi(x) = \max \left\{ (b - Cx)^\top \lambda \colon \lambda \in \{\lambda^1, \dots, \lambda^s\} \right\}. \tag{4.5}$$

This shows that $\varphi(x)$ is a piecewise-linear function and re-writing the bilevel-feasible set as

$$\mathcal{F} = \left\{ (x, y) \colon (x, y) \in \Omega,\ d^\top y - \varphi(x) = 0 \right\} \tag{4.6}$$

shows the claim that the bilevel-feasible set can be written as the intersection of the shared constraint set with a piecewise-linear equality constraint.

Consider now again the definition of the optimal-value function using the vertices of the dual polyhedron of the lower-level problem in (4.5). Suppose that, for a given x, the corresponding optimal solution is the vertex $\lambda(x)$. By using dual feasibility (4.4), we obtain

$$\begin{aligned} 0 &= d^\top y - \varphi(x) \\ &= (D^\top \lambda(x))^\top y - (\lambda(x))^\top (b - Cx) \\ &= (\lambda(x))^\top (Cx + Dy - b). \end{aligned}$$

Thus, for those $\lambda(x)_i$, $i \in \{1, \dots, \ell\}$, with $\lambda(x)_i > 0$ we get

$$(Cx + Dy - b)_i = 0.$$

Hence, the bilevel-feasible set is a union of faces of the shared constraint set. □

In other words, the result of Theorem 4.11 states the following.

Corollary 4.12 *Suppose that the assumptions of Theorem 4.11 hold. Then, the LP-LP bilevel problem* (4.1) *is equivalent to minimizing the upper-level objective function over the intersection of the shared constraint set with a piecewise-linear equality constraint.*

Moreover, we also obtain the following result.

Corollary 4.13 *Suppose that the assumptions of Theorem 4.11 hold and that the bilevel problem is feasible. Then, there always exists an optimal solution to the LP-LP bilevel problem* (4.1) *that is a vertex of the bilevel-feasible set.*

From Theorem 4.11 it follows that the feasible set of a linear bilevel problem is a union of polyhedra; see, e.g., Examples 1.20 and 4.1 again. When we talk about a vertex of the bilevel-feasible set as we do in the last corollary, we are thus referring to an extreme point of one of the polyhedra of the union of polyhedra.

Theorem 4.14 *Suppose that the assumptions of Theorem 4.11 hold and that the bilevel problem is feasible. Then, there always exists an optimal solution* (x^*, y^*) *to the LP-LP bilevel problem* (4.1) *that is a vertex of the shared constraint set* Ω*.*

Proof: Let $(x^1, y^1), \dots, (x^r, y^r)$ be the distinct vertices of the shared constraint set Ω. Because Ω is a convex polytope, any point in Ω can be written as a convex combination of these vertices, i.e.,

$$(x^*, y^*) = \sum_{i=1}^{r} \alpha_i (x^i, y^i)$$

with

$$\sum_{i=1}^{r} \alpha_i = 1 \quad \text{and} \quad \alpha_i \geq 0 \quad \text{for all } i = 1, \dots, r.$$

In the proof of Theorem 4.11, we have seen that the optimal-value function φ is piecewise linear as it is the pointwise maximum of a finite number of affine functions; see (4.5). As such, φ is convex and continuous as well. Because the bilevel solution (x^*, y^*) is also bilevel feasible, the representation in (4.6) leads to

$$\begin{aligned}
0 &= d^\top y^* - \varphi(x^*) \\
&= d^\top \left(\sum_{i=1}^{r} \alpha_i y^i \right) - \varphi \left(\sum_{i=1}^{r} \alpha_i x^i \right) \\
&\geq \sum_{i=1}^{r} \alpha_i d^\top y^i - \sum_{i=1}^{r} \alpha_i \varphi(x^i) \\
&= \sum_{i=1}^{r} \alpha_i \left(d^\top y^i - \varphi(x^i) \right),
\end{aligned} \tag{4.7}$$

where the inequality in (4.7) follows from convexity of φ. By the definition of the optimal-value function, we also have

$$\varphi(x^i) = \min_y \left\{ d^\top y \colon C x^i + D y \geq b \right\} \leq d^\top y^i.$$

This implies $d^\top y^i - \varphi(x^i) \geq 0$. Consequently, for all $i \in \{1, \dots, r\}$ with $\alpha_i > 0$, it holds $d^\top y^i = \varphi(x^i)$. Otherwise, we get a contradiction in (4.7). Hence, for those i with $\alpha_i > 0$, we obtain $(x^i, y^i) \in \mathcal{F}$. From Corollary 4.13, w.l.o.g., we may assume that (x^*, y^*) is a vertex of the bilevel-feasible set. Suppose now that there are two indices i and j with $\alpha_i > 0$ and $\alpha_j > 0$. Thus, $(x^i, y^i) \in \mathcal{F}$ and $(x^j, y^j) \in \mathcal{F}$ holds and we can write (x^*, y^*) as a proper convex combination of two bilevel-feasible points, which is a contradiction to the last corollary. Thus, (x^*, y^*) is a vertex of the shared constraint set. □

By combining Theorem 4.14 with Corollary 4.13, we have seen that the set of vertices of the bilevel-feasible set $\mathcal{F}$ is a subset of the vertices of the shared constraint set Ω. This also shows that $\mathcal{F}$ consists of faces of Ω and that every extreme point of $\mathcal{F}$ is an extreme point of Ω.[1]

4.4 Complexity Results

The first hardness result for LP-LP bilevel problems is due to Jeroslow (1985), where general multilevel models are considered. As a direct consequence of the results by Jeroslow (1985), one obtains the NP-hardness of LP-LP bilevel problems. The problem is also strongly NP-hard, which is shown by Hansen et al. (1992) using a reduction from `KERNEL`; see Problem GT05 in Garey and Johnson (1990), which provides both a good introduction to as well as an excellent encyclopedia of hard problems. Soon after, it has been shown in Vicente et al. (1994) that even checking whether a given point is a local minimizer of a bilevel problem is NP-hard. Further hardness results can also be found in Ben-Ayed and Blair (1990). Surprisingly, the fact that LP-LP bilevel problems belong to NP has only been shown rather recently by Buchheim (2023). As a by-product of this result, it also follows that the big-M values of the respective single-level reformulations (see Section 3.4) admit KKT reformulations of polynomial size.

Before we formally prove the hardness of LP-LP bilevel problems, let us first try to get some intuition about the hardness of these problems. We have already seen that LP-LP bilevel problems are nonconvex optimization problems, which are hard to solve to global optimality in general. Moreover, we know that mixed-integer (or mixed-binary) optimization is hard as well. In the paper by Audet et al. (1997), it is noted that a binary constraint, say $x \in \{0, 1\}$, appearing in a single-level optimization problem can be modeled by an additional lower-level variable y as well as the upper-level coupling constraint $y = 0$ and

$$y = \arg\max_{\bar{y}} \{\bar{y} : \ \bar{y} \leq x, \ \bar{y} \leq 1 - x\} ;$$

[1] A subset E of the convex set $S \subseteq \mathbb{R}^n$ is called an *extremal set* of S if $z \in E$ with $z = \lambda x + (1 - \lambda) y$ and $0 < \lambda < 1$, $x, y \in S$, implies that $x, y \in E$ holds. An *extreme point* of S is an extremal set that is a singleton.

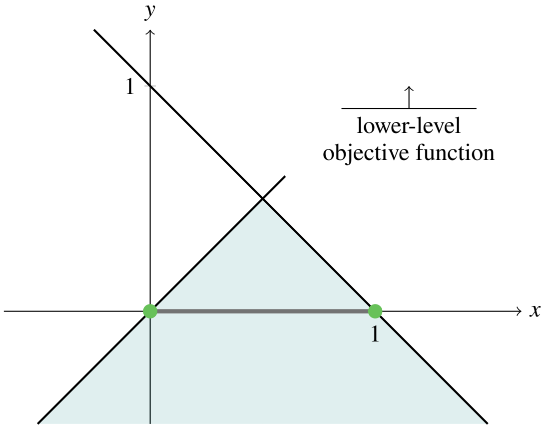


Figure 4.3 Illustration of modeling a binary variable using a linear bilevel feasibility problem. The shaded area corresponds to the graph of the follower's feasible set mapping, the green dots represent the bilevel-feasible points, and the gray line between these dots is the shared constraint set of the bilevel problem.

see Figure 4.3 for an illustration. As a consequence, linear optimization problems with binary variables are a special case of bilevel LPs.

With this intuition, we now prove that linear bilevel problems are strongly NP-hard by a reduction from `3-SAT`, see Problem L02 in Garey and Johnson (1990). The following derivation and the hardness result is taken from Marcotte and Savard (2005).

So what is `3-SAT`? We are given n Boolean variables $x_1, \dots, x_n$, i.e., variables that are either `true` (1) or `false` (0), and the so-called 3-CNF (conjunctive normal form)

$$J = \bigwedge_{i=1}^{m} (l_{i_1} \vee l_{i_2} \vee l_{i_3})$$

having m 3-clauses with the literals l_{i_1}, l_{i_2}, and l_{i_3} for $i \in \{1, \dots, m\}$. A literal is either a Boolean variable or its negation. Given J, `3-SAT` is to determine whether there exists a truth assignment to the Boolean variables that makes the formula true. To each 3-clause $(l_{i_1} \vee l_{i_2} \vee l_{i_3})$, we now associate the linear Boolean inequality

$$v_{i_1} + v_{i_2} + v_{i_3} \geq 1$$

with

$$v_{i_j} = \begin{cases} x_k, & \text{if } l_{i_j} = x_k, \\ 1 - x_k, & \text{if } l_{i_j} = \bar{x}_k, \end{cases}$$

Here and in what follows, $\bar{x}_k$ denotes the negation of the Boolean variable x_k. For instance, the inequality

$$x_1 + (1 - x_4) + x_6 \geq 1$$

corresponds to the 3-clause

$$x_1 \vee \bar{x}_4 \vee x_6.$$

The above inequalities can be written in matrix form as

$$A_J x \geq \mathbb{1} - c,$$

where A_J is a matrix with entries in $\{-1, 0, 1\}$, c is the vector with entries counting the number of negations in the respective 3-clause, and $\mathbb{1}$ is the vector of ones in appropriate dimension. Hence, by definition, J is satisfiable if and only if the latter linear inequality system has a binary solution.

Because we already know that we can use linear bilevel problems to ensure that a continuous variable in $[0, 1]$ only takes binary values, the following theorem is rather straightforward.

Theorem 4.15 (See Theorem 1.4 in Marcotte and Savard (2005)) *The linear bilevel optimization problem* (4.1) *is strongly NP-hard.*

Proof: We consider the linear bilevel problem

$$\begin{aligned}
\min_{x,y} \quad & F(x, y) = \sum_{i=1}^{n} y_i \\
\text{s.t.} \quad & A_J x \geq \mathbb{1} - c,\ x \in [0, 1]^n, \\
& y \in \mathcal{S}(x)
\end{aligned}$$

with $\mathcal{S}(x)$ being the set of optimal solutions to the x-parameterized lower-level problem

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

We now prove that J is satisfiable if and only if the optimal objective function value of the latter bilevel problem is 0, which is the lower bound for this problem.

First, suppose that J is satisfiable and let $x = (x_1, \dots, x_n)$ be the respective binary assignment. Then, by construction, the constraint $A_J x \geq \mathbb{1} - c$ is satisfied and the only feasible point for the x-parameterized lower-level problem is $y_i = 0$ for all $i = 1, \dots, n$. This leads to the lower- and upper-level objective function value of 0, which is optimal because it is the lower bound.

Second, suppose now that J is not satisfiable. By construction of the upper-level linear inequality system, any of its solutions x thus needs to be fractional. Moreover, every solution y of the lower-level problem satisfies $y_i = \min\{x_i, 1 - x_i\}$ for all $i = 1, \dots, n$, which means that at least one y_i has a strictly positive value. This, finally, implies that the objective function $F(x, y)$ cannot be zero, which completes the proof. □

4.5 What You Should Know Now!

1. What are the possible consequences of introducing a coupling constraint to a linear bilevel problem?
2. What can happen if you move coupling constraints from the upper- to the lower-level problem?
3. What do we usually assume for the shared constraint set and why?
4. What do you know about the unboundedness, infeasibility, and solvability of the single-level relaxation of a bilevel problem? Does it allow for any conclusion about the unboundedness, infeasibility, or solvability of the bilevel problem?
5. What can happen if the lower-level problem is unbounded? Can you illustrate this situation using an example?
6. How can we detect the unboundedness of the lower-level problem? Does this property depend on a specific choice of the leader's decision?
7. What is a polyhedral point-to-set mapping?
8. Is the mapping $\mathcal{S}$ in linear bilevel optimization polyhedral or not? Can you prove it?
9. What are the required assumptions to ensure that an LP-LP bilevel problem has at least one solution?
10. What do you know about the geometrical properties of the feasible set of linear bilevel problems?
11. To what points can we restrict our search for optimal solutions of an LP-LP bilevel problem? What role does the single-level relaxation play here?
12. What do you know about the relation of LP-LP bilevel problems and single-level mixed-integer linear problems? What property does your LP-LP bilevel problem need to have to establish this relation?
13. Are linear bilevel problems easy or hard?
14. Can you prove hardness? Which problem is used in the reduction?

5
Algorithms for Linear Bilevel Problems

As we now know about the peculiar properties of linear bilevel problems, their existence theory, and their hardness, the remaining topic to be discussed is how to solve them. This is what we do in this chapter. We present three different algorithms. The first one, the so-called Kth-best algorithm that we discuss in Section 5.1, was among the first methods for solving linear bilevel problems and is directly connected to the geometric properties of linear bilevel problems that we discussed in Section 4.3. The second one is a branch-and-bound method that is specifically tailored to the bilevel setting; see Section 5.2. Finally, it also makes sense to discuss primal heuristics as we have shown that linear bilevel problems are strongly NP-hard. Primal heuristics aim to compute feasible points of good quality, i.e., with a good objective function value, quickly. However, they usually do not give any optimality guarantee. One of the most powerful heuristics in the current literature, which is based on a so-called penalty alternating direction method, is discussed in Section 5.3

5.1 The Kth-Best Algorithm

One of the oldest algorithms to solve LP-LP bilevel problems is the simplex-inspired Kth-best algorithm; see Bialas and Karwan (1984). As before, we consider the general LP-LP bilevel problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \arg\min_{\bar{y}} \left\{ d^\top \bar{y} : Cx + D\bar{y} \geq b \right\} \end{aligned} \tag{5.1}$$

as in Section 4.3. Throughout this chapter, we suppose that the assumptions of Theorem 4.11 hold, i.e., the shared constraint set is non-empty and bounded.

The key idea of the Kth-best algorithm is based on Theorem 4.14, which states that if the bilevel problem is solvable, a bilevel-optimal solution is attained at one of the vertices of the shared constraint set Ω. Thus, similar to the simplex method for linear problems, see, e.g., Chvátal (1983), we can carry out a search over the vertices of Ω to find a solution. To this end, we consider the single-level relaxation

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & Cx + Dy \geq b \end{aligned} \tag{5.2}$$

of Problem (5.1). Let us denote with

$$(x^1, y^1), (x^2, y^2), \dots, (x^r, y^r) \tag{5.3}$$

the ordered set of vertices of Ω, i.e., the ordered set of basic feasible solutions to the single-level relaxation. The ordering is chosen so that

$$c_x^\top x^i + c_y^\top y^i \leq c_x^\top x^{i+1} + c_y^\top y^{i+1}$$

holds for $i = 1, \dots, r-1$.

Hence, the problem of solving the LP-LP bilevel problem can be posed as finding the minimum-index vertex that is feasible for the bilevel problem, i.e., we want to find the index

$$K = \min \left\{ i \in \{1, \dots, r\} \colon (x^i, y^i) \in \mathcal{F} \right\}.$$

This means that we want to find the first vertex in the ordered list in (5.3) whose y-component is an optimal solution to the follower's problem. It is then clear that (x^K, y^K) is a globally optimal solution to the LP-LP bilevel problem (5.1). If, otherwise $(x^i, y^i) \notin \mathcal{F}$ for all $i \in \{1, \dots, r\}$, this proves that the given bilevel problem is infeasible.

The method is formally given in Algorithm 1. Throughout the execution of the method, the set P contains the vertices of the single-level relaxation that have already been processed. It is initialized to be empty in Line 1 and is augmented in Line 7 if the current iterate turned out to be bilevel infeasible. Moreover, the set T always contains those vertices that still need to be tested to determine whether they constitute an optimal solution to the linear bilevel problem. The set is initialized with a basic solution to the single-level relaxation in Line 2 and is augmented by all relevant adjacent extreme points T^i of the current iterate (x^i, y^i) in Line 7. If the set T is getting empty before the algorithm found a bilevel-feasible point, the given LP-LP bilevel problem is infeasible.

Algorithm 1 The Kth-Best Algorithm

Input: An instance of Problem (5.1) satisfying the assumptions of Theorem 4.11.

Output: An optimal solution (x^*, y^*) to Problem (5.1) or the indication that Problem (5.1) is infeasible.

1: Set $i \leftarrow 1$ and $P \leftarrow \emptyset$.

2: Solve Problem (5.2) to obtain a basic optimal solution (x^1, y^1) and set $T \leftarrow \{(x^1, y^1)\}$.

3: Compute an optimal solution $\tilde{y}$ to the follower's x^i-parameterized problem

$$\min_y \quad d^\top y \quad \text{s.t.} \quad Dy \geq b - Cx^i.$$

4: **if** $d^\top y^i \leq d^\top \tilde{y}$ **then**

5:     Set $K \leftarrow i$ and $(x^*, y^*) \leftarrow (x^i, y^i)$.

6:     **return** (x^*, y^*)

7: Let T^i denote the set of adjacent extreme points of (x^i, y^i) such that $(x, y) \in T^i$ implies

$$c_x^\top x + c_y^\top y \geq c_x^\top x^i + c_y^\top y^i.$$

Set $P \leftarrow P \cup \{(x^i, y^i)\}$ and $T \leftarrow (T \cup T^i) \setminus P$.

8: **if** $T = \emptyset$ **then**

9:     **return** the statement "The given LP-LP bilevel problem is infeasible."

10: Set $i \leftarrow i + 1$ and choose (x^i, y^i) with

$$c_x^\top x^i + c_y^\top y^i = \min_{x,y} \left\{ c_x^\top x + c_y^\top y \colon (x, y) \in T \right\}.$$

Go to Step 3.

Remark 5.1 (i) Note that we have to choose an algorithm for linear optimization problems in Line 2 that returns a basic optimal solution. Hence, we can, e.g., use the simplex method for this.

(ii) A crucial and costly part of the algorithm (that we do not discuss here) is the computation of the relevant adjacent extreme points in Step 7. For more details; see Bard (1998).

Example 5.2 We revisit the linear bilevel problem in Example 1.20, i.e., we consider

$$
\begin{aligned}
\min_{x,y} \quad & F(x,y) = x + 6y \\
\text{s.t.} \quad & -x + 5y \le 12.5, \\
& x \ge 0, \\
& y \in \mathcal{S}(x),
\end{aligned}
\tag{5.4}
$$

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

$$
\begin{aligned}
\min_{y} \quad & f(x,y) = -y \\
\text{s.t.} \quad & 2x - y \ge 0, \\
& -x - y \ge -6, \\
& -x + 6y \ge -3, \\
& x + 3y \ge 3.
\end{aligned}
$$

The bilevel problem (5.4) is illustrated again in Figure 5.1. We now apply the Kth-best algorithm (Algorithm 1) to solve it. To this end, we first initialize $i \leftarrow 1$ and $P \leftarrow \emptyset$. Then, we compute a basic optimal solution to the single-level relaxation of Problem (5.4). The optimal solution $(x^1, y^1) = (3, 0)$ to this problem is illustrated by the blue dot in Figure 5.1. We set $T \leftarrow \{(x^1, y^1)\} = \{(3, 0)\}$. Solving the x^1-parameterized lower-level problem then yields $\tilde{y} = 3$. Because $-y^1 > -\tilde{y}$ holds, the point $(x^1, y^1) = (3, 0)$ is not bilevel feasible. Hence, we now need to determine the adjacent extreme points of $(x^1, y^1) = (3, 0)$. From Figure 5.1, it can be seen that the points $(3/7, 6/7)$ and $(39/7, 3/7)$ are adjacent to $(3, 0)$, and both have an upper-level objective function value that is strictly greater than $3 = x^1 + 6y^1$. We thus set

$$
T^1 \leftarrow \left\{ \left(\tfrac{3}{7}, \tfrac{6}{7}\right), \left(\tfrac{39}{7}, \tfrac{3}{7}\right) \right\}.
$$

Next, we update the sets P and T accordingly, i.e., we set

$$
\begin{aligned}
P &\leftarrow P \cup \{(3,0)\} = \{(3,0)\}, \\
T &\leftarrow (T \cup T^i) \setminus P = \left\{ \left(\tfrac{3}{7}, \tfrac{6}{7}\right), \left(\tfrac{39}{7}, \tfrac{3}{7}\right) \right\}.
\end{aligned}
$$

Moreover, we set $i \leftarrow 2$ and choose $(x^2, y^2) = (3/7, 6/7)$ because

$$
\left(\tfrac{3}{7}, \tfrac{6}{7}\right) \in \operatorname*{arg\,min}_{x,y} \{x + 6y \colon (x,y) \in T\}.
$$

We then solve the x^2-parameterized lower-level problem, which has the optimal

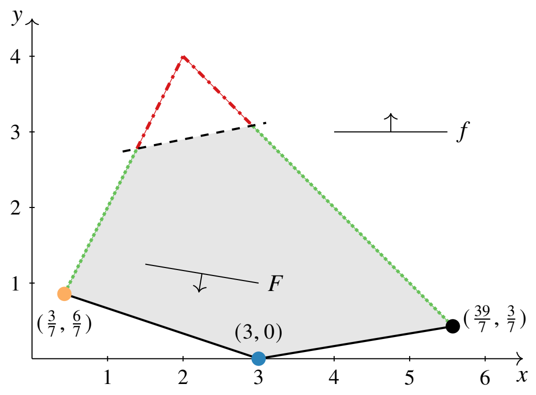


Figure 5.1 The shared constraint set (gray area), the nonconvex bilevel-feasible set (dotted green lines), the optimal solution $(3/7, 6/7)$ to the bilevel problem (orange point), and the optimal solution $(3, 0)$ to the single-level relaxation (blue point) of the bilevel problem (5.4). Taken and modified from Kleinert (2021).

solution $\tilde{y} = 6/7$. Because $\tilde{y} = y^2$ and thus $d^\top \tilde{y} = d^\top y^2$ holds, we set $K \leftarrow 2$ and return the bilevel optimal solution $(x^*, y^*) = (x^2, y^2) = (3/7, 6/7)$. △

Exercise 5.3 Reconsider the linear bilevel problem (3.21) in Exercise 3.22.

(i) Solve Problem (3.21) graphically.
(ii) Determine all vertices of the shared constraint set of Problem (3.21).
(iii) Solve Problem (3.21) using the Kth-best algorithm (Algorithm 1).

Exercise 5.4 Consider the problem

$$\min_{x,y} \quad y \quad \text{s.t.} \quad 0 \le x \le 2,\ y \le \frac{1}{2},\ y \in \mathcal{S}(x),$$

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

$$\min_{y} \quad -y \quad \text{s.t.} \quad y \ge 0,\ y \le 1 + x,\ y \le 3 - x.$$

This is the problem from Example 4.1 with a coupling constraint $y \le a$, where $a = 1/2$ is chosen so that the overall LP-LP bilevel problem is infeasible.

Apply the Kth-best algorithm (Algorithm 1) to this problem.

5.2 Complementarity-Based Branch-and-Bound

In Section 3.2, we have seen that the general LP-LP bilevel problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \arg\min_{\bar{y}} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \end{aligned}$$

can equivalently be re-written via the KKT reformulation as the MPCC

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \ Cx + Dy \geq b, \\ & D^\top \lambda = d, \ \lambda \geq 0, \\ & \lambda_i (C_{i\cdot} x + D_{i\cdot} y - b_i) = 0 \quad \text{for all } i = 1, \dots, \ell. \end{aligned}$$

Moreover, we have seen in Section 3.4 that the latter problem can be re-stated as the mixed-integer linear optimization problem

$$\begin{aligned} \min_{x,y,\lambda,z} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \ Cx + Dy \geq b, \\ & D^\top \lambda = d, \ \lambda \geq 0, \\ & \lambda_i \leq M z_i \quad \text{for all } i = 1, \dots, \ell, \\ & C_{i\cdot} x + D_{i\cdot} y - b_i \leq M(1 - z_i) \quad \text{for all } i = 1, \dots, \ell, \\ & z_i \in \{0, 1\} \quad \text{for all } i = 1, \dots, \ell, \end{aligned} \tag{5.5}$$

for a sufficiently large constant $M > 0$. We can solve this problem without further ado by using a state-of-the-art mixed-integer solver such as those discussed in Section 3.4. These solvers tackle the problem using the classic branch-and-bound method; see Land and Doig (1960) for the original paper as well as the book by Wolsey (2020) and Chapter 7 of this book for an introduction to branch-and-bound. This means that they branch on the auxiliary binary variables that model whether the ith lower-level constraint is binding or whether the ith dual variable vanishes. Alternatively, we can branch on the KKT complementarity constraints

$$\lambda_i (C_{i\cdot} x + D_{i\cdot} y - b_i) = 0 \quad \text{for all } i = 1, \dots, \ell$$

directly, which is, mathematically speaking, the same. However, the complementarity-constraint-based branching does not require choosing sufficiently large big-M values, which is often a drawback of the mixed-integer linear approach (5.5) for solving the KKT reformulation.

The idea of complementarity-constraint-based branch-and-bound for LP-LP bilevel problems is rather simple.

(i) We start by solving the problem

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\ & D^\top \lambda = d, \; \lambda \geq 0. \end{aligned} \tag{5.6}$$

This is the single-level relaxation extended with the dual variables λ and the lower level's dual polyhedron given by

$$D^\top \lambda = d, \quad \lambda \geq 0.$$

Although this is a good starting point, the resulting lower bound is typically very weak because the dual part λ of the problem is completely decoupled from the primal part (x, y).

(ii) Usually, there will be an index $i \in \{1, \dots, \ell\}$ so that the ith KKT complementarity condition is not satisfied, i.e.,

$$\lambda_i (C_{i\cdot} x + D_{i\cdot} y - b_i) > 0$$

holds. We take such an index i and branch by constructing two new sub-problems: one in which the primal constraint

$$C_{i\cdot} x + D_{i\cdot} y = b_i$$

is added and one in which the dual fixing

$$\lambda_i = 0$$

is added.

(iii) Then, we choose one of the unsolved sub-problems and proceed in the same way.

Every node in the branch-and-bound tree is thus defined by the root-node problem (5.6) as well as some index sets $\mathcal{D} \subseteq \{1, \dots, \ell\}$ and $\mathcal{P} \subseteq \{1, \dots, \ell\}$ that contain those indices for which the dual fixing $\lambda_i = 0$ or the primal constraint $C_{i\cdot} x + D_{i\cdot} y = b_i$ is added to Problem (5.6), respectively. Thus, we denote a node by its corresponding index-set pair $(\mathcal{P}, \mathcal{D})$, which corresponds to the linear *node problem*

$$
\begin{aligned}
\min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\
& D^\top \lambda = d, \; \lambda \geq 0, \\
& C_{i\cdot} x + D_{i\cdot} y = b_i \quad \text{for all } i \in \mathcal{P}, \\
& \lambda_i = 0 \quad \text{for all } i \in \mathcal{D}.
\end{aligned} \tag{5.7}
$$

This leads to the branch-and-bound method formally stated in Algorithm 2. In the beginning, we set the so-called *incumbent value* $\mathcal{U}$ i.e., the objective value of the best feasible point found so far, to $+\infty$ and we initialize the set Q of open problems to be solved with the tuple defining the root-node problem (5.6); see Line 1. An "open problem" is a sub-problem corresponding to a node of the branch-and-bound search tree that has been created but not yet processed. As long as there are open problems to be solved, we choose an arbitrary one in Line 3 and solve it. If this problem is infeasible, we do not consider it further and return to the beginning of the while-loop. Otherwise, we check if the solution, which we call *node solution* from now on, leads to a value being not better than the current incumbent value (Line 8) and prune the node if this is the case.[1] If we continue with the current while-loop, this means that we have found a node solution with an objective function value better than the current incumbent value. We then check in Line 10 if the node solution satisfies all complementarity conditions. If this is the case, we found a new *incumbent solution*, i.e., the best feasible point found so far. We then update the *incumbent*, i.e., the incumbent solution and the incumbent value, in Line 11. Otherwise, we branch on a violated complementarity constraint in Line 12 and continue. Here and in what follows, we use the term incumbent to refer to both the incumbent solution and its objective value. When a distinction is needed, we use incumbent solution to refer to the point itself and incumbent value to refer to its associated objective function value.

To sum up, the method is based on (i) a root-node problem that is a relaxation of the problem to be solved, (ii) branching on violated constraints that are omitted in this relaxation, and (iii) the following rules to prune a node.

Rule 1 (Infeasibility) A node is pruned if the node problem is infeasible; see Line 5.

Rule 2 (Bounding) A node is pruned if the node problem has a solution that is not better than the incumbent value; see Line 8.

[1] "Pruning nodes" is also often called "fathoming nodes".

Rule 3 (Feasibility) A node is pruned if the node solution is feasible for the original problem; see Line 11.

Algorithm 2 Branch-and-Bound for LP-LP Bilevel Problems

Input: An instance of Problem (5.1) satisfying that Problem (5.6) is bounded.
Output: An optimal solution (x^*, y^*) to Problem (5.1) together with a dual lower-level solution λ^* or the indication that Problem (5.1) is infeasible.

1: Set $\mathcal{U} \leftarrow +\infty$ and $Q \leftarrow \{(\emptyset, \emptyset)\}$.
2: **while** $Q \neq \emptyset$ **do**
3: Choose any $(\mathcal{P}, \mathcal{D}) \in Q$ and set $Q \leftarrow Q \setminus \{(\mathcal{P}, \mathcal{D})\}$.
4: Solve Problem (5.7) using $\mathcal{P}$ and $\mathcal{D}$.
5: **if** Problem (5.7) for $\mathcal{P}$ and $\mathcal{D}$ is infeasible **then**
6: Go to Step 2.
7: Let $(\bar{x}, \bar{y}, \bar{\lambda})$ denote the solution to Problem (5.7) for $\mathcal{P}$ and $\mathcal{D}$.
8: **if** $c_x^\top \bar{x} + c_y^\top \bar{y} \geq \mathcal{U}$ **then**
9: Go to Step 2.
10: **if** $(\bar{x}, \bar{y}, \bar{\lambda})$ satisfies $\bar{\lambda}_i (C_{i\cdot} \bar{x} + D_{i\cdot} \bar{y} - b_i) = 0$ for all $i \in \{1, \dots, \ell\}$ **then**
11: Set

$$(x^*, y^*, \lambda^*) \leftarrow (\bar{x}, \bar{y}, \bar{\lambda}), \quad \mathcal{U} \leftarrow c_x^\top x^* + c_y^\top y^*$$

and go to Step 2.
12: Choose any $i \in \{1, \dots, \ell\}$ with $\bar{\lambda}_i (C_{i\cdot} \bar{x} + D_{i\cdot} \bar{y} - b_i) > 0$ and set

$$Q \leftarrow Q \cup \{(\mathcal{P} \cup \{i\}, \mathcal{D}), (\mathcal{P}, \mathcal{D} \cup \{i\})\}.$$

13: **if** $\mathcal{U} < +\infty$ **then**
14: **return** (x^*, y^*, λ^*)
15: **else**
16: **return** the statement "The given LP-LP bilevel problem is infeasible."

Remark 5.5 As discussed above, Algorithm 2 maintains an incumbent value $\mathcal{U}$ that always corresponds to the best feasible point found so far, i.e., it is an upper bound of the optimal objective function value. Usual branch-and-bound methods also maintain a global lower bound on the optimal objective function value. To define this formally, let us call a node an *inner node* of the current search tree if the node itself and both child nodes have already been solved (either to optimality or by detecting infeasibility). Then, the global lower bound $\mathcal{L}$ is the minimum over the optimal objective function values of all nodes that have already been solved but that are not inner nodes; see Figure 5.2 for an illustration. The difference $\mathcal{U} - \mathcal{L}$ is called the *(absolute) optimality gap*. Obviously, the

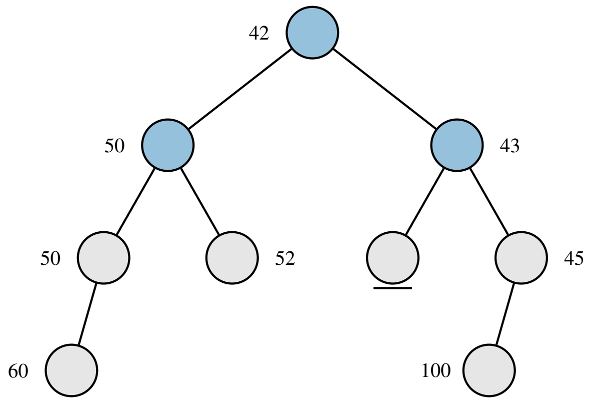


Figure 5.2 An exemplary branch-and-bound tree at an intermediate stage of the overall branch-and-bound method. The values shown next to the nodes are the optimal objective function values of the problems considered at these nodes. Inner nodes are highlighted in blue, whereas gray nodes are not inner nodes. One node has been pruned due to infeasibility, which is indicated by an underline. In this illustrative example, the global lower bound is given by $\mathcal{L} = 45 = \min\{60, 50, 52, 45, 100\}$.

problem is solved and the algorithm can thus terminate if this gap is closed, i.e., if $\mathcal{U} = \mathcal{L}$ holds. Moreover, in many practical situations, it is enough to terminate with a feasible point that is close to being optimal. This can be achieved by terminating the branch-and-bound method whenever the optimality gap falls below a certain user-defined threshold $\varepsilon > 0$, i.e., whenever $\mathcal{U} - \mathcal{L} \leq \varepsilon$ holds.

Note that Algorithm 2 has some degrees of freedom. For instance, we are free to decide which specific node we choose in Line 3. Possible options to traverse the branch-and-bound search tree include a depth-first search (DFS) or breadth-first search strategy (BFS)—or even variants of the two as well as completely different strategies. In the literature of mixed-integer optimization, these strategies are called *node-selection strategies*. For more details, we refer to Section 3.1 in Belotti et al. (2013) and the references therein. Moreover, the specific index i of a violated complementarity constraint to be branched on is also not always uniquely determined. To choose a specific i, there are different *branching rules*; see, e.g., Achterberg et al. (2005).

Let us now analyze the branch-and-bound method in Algorithm 2. To this end, we first formally introduce the notion of a relaxation.

Definition 5.6 (Relaxation) Consider the optimization problem $\min\{f(x) : x \in \mathcal{F}\}$. The optimization problem $\min\{g(x) : x \in \mathcal{F}'\}$ is called a *relaxation* of the other problem if $\mathcal{F} \subseteq \mathcal{F}'$ and if $g(x) \leq f(x)$ holds for all $x \in \mathcal{F}$.

The easiest way to obtain a relaxation is to simply delete constraints from a given set of constraints. This is exactly what we did to derive the single-level relaxation, which means that the wording is reasonable.

Moreover, we see that Problem (5.7) for given sets $\mathcal{P}$ and $\mathcal{D}$, i.e.,

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\ & D^\top \lambda = d, \; \lambda \geq 0, \\ & C_{i\cdot} x + D_{i\cdot} y = b_i \quad \text{for all } i \in \mathcal{P}, \\ & \lambda_i = 0 \quad \text{for all } i \in \mathcal{D}, \end{aligned}$$

is a relaxation of the problem

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\ & D^\top \lambda = d, \; \lambda \geq 0, \\ & \lambda_i (C_{i\cdot} x + D_{i\cdot} y - b_i) = 0 \quad \text{for all } i \in \{1, \dots, \ell\}, \\ & C_{i\cdot} x + D_{i\cdot} y = b_i \quad \text{for all } i \in \mathcal{P}, \\ & \lambda_i = 0 \quad \text{for all } i \in \mathcal{D}. \end{aligned} \tag{5.8}$$

Note that the latter is the KKT reformulation of the LP-LP bilevel problem, which we extended by the equality constraints corresponding to the sets $\mathcal{P}$ and $\mathcal{D}$.

To prove the correctness of the branch-and-bound method in Algorithm 2, we have to show that

(i) the bounding step in Step 8 as well as the pruning of infeasible nodes in Step 5 are correct and that
(ii) the branching step in Step 12 is correct.

This is done in the following two lemmas.

Lemma 5.7 (Bounding Lemma) *Let $\mathcal{P}, \mathcal{D} \subseteq \{1, \dots, \ell\}$ be given. Moreover, denote the optimal objective function value of the relaxation* (5.7) *by z^{rel} and the optimal objective function value of Problem* (5.8) *by z (if they exist; otherwise they are set to ∞). Then, it holds*

$$z^{rel} \leq z.$$

Furthermore, the infeasibility of the relaxation (5.7) implies the infeasibility of Problem (5.8).

Proof: Both statements immediately follow from the definition of a relaxation (Definition 5.6). □

Lemma 5.8 (Branching Lemma) *Let $\mathcal{P}, \mathcal{D} \subseteq \{1, \dots, \ell\}$ be given. Moreover, let the point (x, y, λ) be feasible for Problem (5.8) for the given sets $\mathcal{P}$ and $\mathcal{D}$. Let $i \in \{1, \dots, \ell\}$. Then, the point (x, y, λ) is feasible for Problem (5.8) with the pair of sets $(\mathcal{P} \cup \{i\}, \mathcal{D})$ or for Problem (5.8) with the pair of sets $(\mathcal{P}, \mathcal{D} \cup \{i\})$.*

Proof: Let $\mathcal{P}, \mathcal{D} \subseteq \{1, \dots, \ell\}$ be given and consider a point (x, y, λ) that is feasible for Problem (5.8) for the given sets $\mathcal{P}$ and $\mathcal{D}$. Let any $i \in \{1, \dots, \ell\}$ be given and fixed. Since (x, y, λ) is feasible for (5.8), we have

$$\lambda_i (C_{i\cdot} x + D_{i\cdot} y - b_i) = 0.$$

Hence, either $C_{i\cdot} x + D_{i\cdot} y = b_i$ or $\lambda_i = 0$ holds. In the former case, the point (x, y, λ) is feasible for Problem (5.8) with $(\mathcal{P} \cup \{i\}, \mathcal{D})$, whereas, in the latter case, the point is feasible for (5.8) with $(\mathcal{P}, \mathcal{D} \cup \{i\})$. □

Theorem 5.9 (Correctness Theorem) *Suppose that the root-node relaxation (5.6) is bounded. Then, Algorithm 2 terminates after a finite number of visited nodes with an optimal solution to (5.1) or with the correct indication of infeasibility.*

Proof: The only thing that is left to prove is that the algorithm terminates after a finite number of visited nodes. This, however, follows immediately from the fact that we only have a finite number of KKT complementarity conditions to branch on. □

To sum up, we have seen that we can use a branch-and-bound method to solve LP-LP bilevel problems. In particular, we have seen that it does not require to choose any big-M values.

Remark 5.10 It is rather easy to realize a branch-and-bound method for linear bilevel problems in modern mixed-integer linear solvers such as those mentioned in Section 3.4 by using so-called *special ordered sets of type 1* (SOS1).

A set of non-negative variables $x_1, \dots, x_n \geq 0$ is called a special ordered set of type 1 if there exists exactly one index $i \in \{1, \dots, n\}$ with $x_i > 0$ and $x_j = 0$ for all $j \neq i$. We denote this property of the set of variables $x_1, \dots, x_n$ in the following via

$$\text{SOS1}(x_1, \dots, x_n).$$

This property of a subset of variables of a mixed-integer linear optimization problem can also be communicated to general-purpose solvers.

If we introduce the non-negative auxiliary variables

$$s_i = C_{i\cdot}x + D_{i\cdot}y - b_i \quad \text{for all } i = 1, \dots, \ell,$$

we can equivalently state the complementarity conditions

$$C_{i\cdot}x + D_{i\cdot}y - b_i = 0 \quad \text{or} \quad \lambda_i = 0 \quad \text{for all } i = 1, \dots, \ell,$$

as

$$\mathrm{SOS1}(s_i, \lambda_i) \quad \text{for all } i = 1, \dots, \ell.$$

By doing so, the mixed-integer linear solver takes care of the branching on these SOS1 conditions, which is mathematically equivalent to branching on complementarity conditions as it is done in Algorithm 2.

More information on the usage of SOS1 techniques can be found in Kleinert and Schmidt (2023).

Branch-and-bound is at the core of almost all algorithms that will follow in the remainder of this book. Due to this reason, the next example shows one execution of the branch-and-bound method in great detail. If you are already familiar with branch-and-bound, you might want to skip it. If not, it is most likely a good idea to go through the entire example, although it is a bit lengthy.

Example 5.11 We now consider the linear bilevel problem

$$\begin{aligned} \min_{x, y=(y_1,y_2)} \quad & 7x - 2y_1 + 3y_2 \\ \text{s.t.} \quad & x \geq 0, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{5.9}$$

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

$$\begin{aligned} \min_{y_1, y_2} \quad & 2y_1 - y_2 \\ \text{s.t.} \quad & 4x + y_1 + y_2 \leq 3, \\ & 2x - 2y_1 + 5y_2 \leq 5, \\ & 3x - y_1 - 2y_2 \leq 1, \\ & y_1, y_2 \geq 0. \end{aligned}$$

We now use the branch-and-bound method of Algorithm 2 to solve this problem. We do this both for DFS and BFS as the node-selection rule to implement Line 3.

To this end, we first derive the KKT reformulation of the given problem. The Lagrangian function of the lower-level problem is given by

$$\begin{aligned}\mathcal{L}(x, y, \lambda) = 2y_1 - y_2 & \\ & - \lambda_1(3 - 4x - y_1 - y_2) \\ & - \lambda_2(5 - 2x + 2y_1 - 5y_2) \\ & - \lambda_3(1 - 3x + y_1 + 2y_2) \\ & - \lambda_4 y_1 - \lambda_5 y_2\end{aligned}$$

and the KKT conditions read

$$\begin{aligned}\nabla_{y_1}\mathcal{L}(x, y, \lambda) = 2 + \lambda_1 - 2\lambda_2 - \lambda_3 - \lambda_4 &= 0, \\ \nabla_{y_2}\mathcal{L}(x, y, \lambda) = -1 + \lambda_1 + 5\lambda_2 - 2\lambda_3 - \lambda_5 &= 0, \\ 4x + y_1 + y_2 &\leq 3, \\ 2x - 2y_1 + 5y_2 &\leq 5, \\ 3x - y_1 - 2y_2 &\leq 1, \\ y_1, y_2 &\geq 0, \\ \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 &\geq 0, \\ \lambda_1(3 - 4x - y_1 - y_2) &= 0, \\ \lambda_2(5 - 2x + 2y_1 - 5y_2) &= 0, \\ \lambda_3(1 - 3x + y_1 + 2y_2) &= 0, \\ \lambda_4 y_1 &= 0, \\ \lambda_5 y_2 &= 0.\end{aligned}$$

Thus, the KKT reformulation of the bilevel problem (5.9) can be written as

$$\begin{aligned}
\min_{x,y,\lambda} \quad & 7x - 2y_1 + 3y_2 \\
\text{s.t.} \quad & x \geq 0, \\
& g_1(x, y) = 3 - 4x - y_1 - y_2 \geq 0, \\
& g_2(x, y) = 5 - 2x + 2y_1 - 5y_2 \geq 0, \\
& g_3(x, y) = 1 - 3x + y_1 + 2y_2 \geq 0, \\
& g_4(x, y) = y_1 \geq 0, \\
& g_5(x, y) = y_2 \geq 0, \\
& \lambda_1 - 2\lambda_2 - \lambda_3 - \lambda_4 = -2, \\
& \lambda_1 + 5\lambda_2 - 2\lambda_3 - \lambda_5 = 1, \\
& \lambda_i \geq 0 \quad \text{for all } i \in \{1, \dots, 5\}, \\
& \lambda_i g_i(x, y) = 0 \quad \text{for all } i \in \{1, \dots, 5\}.
\end{aligned}$$

In what follows, we consider the problem

$$\begin{aligned}
\min_{x,y,\lambda} \quad & 7x - 2y_1 + 3y_2 \\
\text{s.t.} \quad & x \geq 0, \\
& g_i(x, y) \geq 0 \quad \text{for all } i \in \{1, \dots, 5\}, \\
& \lambda_1 - 2\lambda_2 - \lambda_3 - \lambda_4 = -2, \\
& \lambda_1 + 5\lambda_2 - 2\lambda_3 - \lambda_5 = 1, \\
& \lambda_i \geq 0 \quad \text{for all } i \in \{1, \dots, 5\}, \\
& g_i(x, y) = 0 \quad \text{for all } i \in \mathcal{P}, \\
& \lambda_i = 0 \quad \text{for all } i \in \mathcal{D},
\end{aligned} \tag{5.10}$$

associated with the index-sets $\mathcal{P}, \mathcal{D} \subseteq \{1, \dots, 5\}$. The first steps of the branch-and-bound method are identical for both search strategies (DFS and BFS):

Processing the Root Node (Node 0)

- We initialize $\mathcal{U} \leftarrow +\infty$ and $\mathcal{Q} \leftarrow \{(\emptyset, \emptyset)\}$.
- We solve Problem (5.10) using the sets $\mathcal{P} = \emptyset$ and $\mathcal{D} = \emptyset$, and we set $\mathcal{Q} \leftarrow \mathcal{Q} \setminus \{(\mathcal{P}, \mathcal{D})\} = \emptyset$.
- We obtain $\bar{x} = 0$, $\bar{y} = (3, 0)$, and $\bar{\lambda} = (1, 0, 0, 3, 0)$ with an objective function value of -6.
- The fourth complementarity constraint is violated, i.e., $\bar{\lambda}_4 g_4(\bar{x}, \bar{y}) > 0$ holds. Thus, we generate two new sub-problems and set

$$\mathcal{Q} \leftarrow \mathcal{Q} \cup \{(\emptyset, \{4\}), (\{4\}, \emptyset)\} = \{(\emptyset, \{4\}), (\{4\}, \emptyset)\}.$$

Processing Node 1

- We solve Problem (5.10) using the sets $\mathcal{P} = \emptyset$ and $\mathcal{D} = \{4\}$, and we set $Q \leftarrow \{(\{4\}, \emptyset)\}$.
- We obtain $\bar{x} = 0$, $\bar{y} = (3, 0)$, and $\bar{\lambda} = (0, 1, 0, 0, 4)$ with an objective function value of -6.

We now need to distinguish between the DFS and the BFS method. We start with DFS.

- The second complementarity constraint is violated, i.e., $\bar{\lambda}_2 g_2(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\emptyset, \{2, 4\}), (\{2\}, \{4\}), (\{4\}, \emptyset)\}.$$

Processing Node 2 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \emptyset$ and $\mathcal{D} = \{2, 4\}$, and we set

$$Q \leftarrow \{(\{2\}, \{4\}), (\{4\}, \emptyset)\}.$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 3 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2\}$ and $\mathcal{D} = \{4\}$, and we set

$$Q \leftarrow \{(\{4\}, \emptyset)\}.$$

- We obtain $\bar{x} = 0$, $\bar{y} = (1.43, 1.57)$, and $\bar{\lambda} = (0, 1, 0, 0, 4)$ with an objective function value of 1.86.
- The fifth complementarity constraint is violated, i.e., $\bar{\lambda}_5 g_5(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{2\}, \{4, 5\}), (\{2, 5\}, \{4\}), (\{4\}, \emptyset)\}.$$

Processing Node 4 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2\}$ and $\mathcal{D} = \{4, 5\}$, and we set

$$Q \leftarrow \{(\{2, 5\}, \{4\}), (\{4\}, \emptyset)\}.$$

- We obtain $\bar{x} = 0$, $\bar{y} = (1.43, 1.57)$, and $\bar{\lambda} = (0, 0.56, 0.89, 0, 0)$ with an objective function value of 1.86.
- The third complementarity constraint is violated, i.e., $\bar{\lambda}_3 g_3(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{2\}, \{3, 4, 5\}), (\{2, 3\}, \{4, 5\}), (\{2, 5\}, \{4\}), (\{4\}, \emptyset)\}.$$

Processing Node 5 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2\}$ and $\mathcal{D} = \{3,4,5\}$, and we set

$$Q \leftarrow \{(\{2,3\},\{4,5\}), (\{2,5\},\{4\}), (\{4\},\emptyset)\}.$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 6 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2,3\}$ and $\mathcal{D} = \{4,5\}$, and we set

$$Q \leftarrow \{(\{2,5\},\{4\}), (\{4\},\emptyset)\}.$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 7 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2,5\}$ and $\mathcal{D} = \{4\}$, and we set

$$Q \leftarrow \{(\{4\},\emptyset)\}.$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 8 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{4\}$ and $\mathcal{D} = \emptyset$, and we set $Q \leftarrow \emptyset$.
- We obtain $\bar{x} = 0$, $\bar{y} = (0,0)$, and $\bar{\lambda} = (1,0,0,3,0)$ with an objective function value of 0.
- The first complementarity constraint is violated, i.e., $\bar{\lambda}_1 g_1(\bar{x},\bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{4\},\{1\}), (\{1,4\},\emptyset)\}.$$

Processing Node 9 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{4\}$ and $\mathcal{D} = \{1\}$, and we set $Q \leftarrow \{(\{1,4\},\emptyset)\}$.
- We obtain $\bar{x} = 0$, $\bar{y} = (0,0)$, and $\bar{\lambda} = (0,0.2,0,1.6,0)$ with an objective function value of 0.
- The second complementarity constraint is violated, i.e., $\bar{\lambda}_2 g_2(\bar{x},\bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{4\},\{1,2\}), (\{2,4\},\{1\}), (\{1,4\},\emptyset)\}.$$

Processing Node 10 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{4\}$ and $\mathcal{D} = \{1,2\}$, and we set

$$Q \leftarrow \{(\{2,4\},\{1\}), (\{1,4\},\emptyset)\}.$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 11 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2, 4\}$ and $\mathcal{D} = \{1\}$, and we set $Q \leftarrow \{(\{1, 4\}, \emptyset)\}$.
- We obtain $\bar{x} = 0$, $\bar{y} = (0, 1)$, and $\bar{\lambda} = (0, 0.2, 0, 1.6, 0)$ with an objective function value of 3.
- The point $(\bar{x}, \bar{y}, \bar{\lambda})$ satisfies all complementarity constraints, i.e., $(\bar{x}, \bar{y}, \bar{\lambda})$ is feasible for the bilevel problem (5.9).
- We set $(x^*, y^*, \lambda^*) \leftarrow (\bar{x}, \bar{y}, \bar{\lambda})$ and update the incumbent value $\mathcal{U} \leftarrow 3$.

Processing Node 12 Using DFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{1, 4\}$ and $\mathcal{D} = \emptyset$, and we set $Q \leftarrow \emptyset$.
- We obtain $\bar{x} = 0.64$, $\bar{y} = (0, 0.46)$, and $\bar{\lambda} = (1, 0, 0, 3, 0)$ with an objective function value of $5.82 > \mathcal{U}$.
- Due to bounding, this node can be pruned.
- Now, $Q = \emptyset$ holds, i.e., there are no more nodes that need to be explored. We have $\mathcal{U} < \infty$ and, thus, the optimal solution to the bilevel problem (5.9) is given by (x^*, y^*, λ^*).

Using BFS, we obtain the following.

- At node 1, the second complementarity constraint is violated, i.e., $\bar{\lambda}_2 g_2(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{4\}, \emptyset), (\emptyset, \{2, 4\}), (\{2\}, \{4\})\} .$$

Processing Node 2 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{4\}$ and $\mathcal{D} = \emptyset$, and we set

$$Q \leftarrow \{(\emptyset, \{2, 4\}), (\{2\}, \{4\})\} .$$

- We obtain $\bar{x} = 0$, $\bar{y} = (0, 0)$, and $\bar{\lambda} = (1, 0, 0, 3, 0)$ with an objective function value of 0.
- The first complementarity constraint is violated, i.e., $\bar{\lambda}_1 g_1(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\emptyset, \{2, 4\}), (\{2\}, \{4\}), (\{4\}, \{1\}), (\{1, 4\}, \emptyset)\} .$$

Processing Node 3 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \emptyset$ and $\mathcal{D} = \{2, 4\}$, and we set

$$Q \leftarrow \{(\{2\}, \{4\}), (\{4\}, \{1\}), (\{1, 4\}, \emptyset)\} .$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 4 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2\}$ and $\mathcal{D} = \{4\}$, and we set

$$Q \leftarrow \{(\{4\}, \{1\}), (\{1, 4\}, \emptyset)\}.$$

- We obtain $\bar{x} = 0$, $\bar{y} = (1.43, 1.57)$, and $\bar{\lambda} = (0, 1, 0, 0, 4)$ with an objective function value of 1.86.
- The fifth complementarity constraint is violated, i.e., $\bar{\lambda}_5 g_5(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{4\}, \{1\}), (\{1, 4\}, \emptyset), (\{2\}, \{4, 5\}), (\{2, 5\}, \{4\})\}.$$

Processing Node 5 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{4\}$ and $\mathcal{D} = \{1\}$, and we set

$$Q \leftarrow \{(\{1, 4\}, \emptyset), (\{2\}, \{4, 5\}), (\{2, 5\}, \{4\})\}.$$

- We obtain $\bar{x} = 0$, $\bar{y} = (0, 0)$, and $\bar{\lambda} = (0, 0.2, 0, 1.6, 0)$ with an objective function value of 0.
- The second complementarity constraint is violated, i.e., $\bar{\lambda}_2 g_2(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$Q \leftarrow \{(\{1, 4\}, \emptyset), (\{2\}, \{4, 5\}), (\{2, 5\}, \{4\}), (\{4\}, \{1, 2\}), (\{2, 4\}, \{1\})\}.$$

Processing Node 6 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{1, 4\}$ and $\mathcal{D} = \emptyset$, and we set

$$Q \leftarrow \{(\{2\}, \{4, 5\}), (\{2, 5\}, \{4\}), (\{4\}, \{1, 2\}), (\{2, 4\}, \{1\})\}.$$

- We obtain $\bar{x} = 0.64$, $\bar{y} = (0, 0.46)$, and $\bar{\lambda} = (1, 0, 0, 3, 0)$ with an objective function value of 5.82.
- The point $(\bar{x}, \bar{y}, \bar{\lambda})$ satisfies all complementarity constraints, i.e., $(\bar{x}, \bar{y}, \bar{\lambda})$ is feasible for the bilevel problem (5.9).
- We set $(x^*, y^*, \lambda^*) \leftarrow (\bar{x}, \bar{y}, \bar{\lambda})$ and update the incumbent value $\mathcal{U} \leftarrow 5.82$.

Processing Node 7 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2\}$ and $\mathcal{D} = \{4, 5\}$, and we set

$$Q \leftarrow \{(\{2, 5\}, \{4\}), (\{4\}, \{1, 2\}), (\{2, 4\}, \{1\})\}.$$

- We obtain $\bar{x} = 0$, $\bar{y} = (1.43, 1.57)$, and $\bar{\lambda} = (0, 0.56, 0.89, 0, 0)$ with an objective function value of $1.86 < \mathcal{U}$.
- The third complementarity constraint is violated, i.e., $\bar{\lambda}_3 g_3(\bar{x}, \bar{y}) > 0$ holds. Thus, we set

$$\begin{aligned} Q \leftarrow &\{(\{2, 5\}, \{4\}), (\{4\}, \{1, 2\}), (\{2, 4\}, \{1\}), \\ &(\{2\}, \{3, 4, 5\}), (\{2, 3\}, \{4, 5\})\}. \end{aligned}$$

Processing Node 8 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2, 5\}$ and $\mathcal{D} = \{4\}$, and we set

$$Q \leftarrow \{(\{4\}, \{1, 2\}), (\{2, 4\}, \{1\}), (\{2\}, \{3, 4, 5\}), (\{2, 3\}, \{4, 5\})\} .$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 9 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{4\}$ and $\mathcal{D} = \{1, 2\}$, and we set

$$Q \leftarrow \{(\{2, 4\}, \{1\}), (\{2\}, \{3, 4, 5\}), (\{2, 3\}, \{4, 5\})\} .$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 10 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2, 4\}$ and $\mathcal{D} = \{1\}$, and we set

$$Q \leftarrow \{(\{2\}, \{3, 4, 5\}), (\{2, 3\}, \{4, 5\})\} .$$

- We obtain $\bar{x} = 0$, $\bar{y} = (0, 1)$, and $\bar{\lambda} = (0, 0.2, 0, 1.6, 0)$ with an objective function value of $3 < \mathcal{U}$.
- The point $(\bar{x}, \bar{y}, \bar{\lambda})$ satisfies all complementarity constraints, i.e., $(\bar{x}, \bar{y}, \bar{\lambda})$ is feasible for the bilevel problem (5.9).
- We set $(x^*, y^*, \lambda^*) \leftarrow (\bar{x}, \bar{y}, \bar{\lambda})$ and update the incumbent value $\mathcal{U} \leftarrow 3$.

Processing Node 11 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2\}$ and $\mathcal{D} = \{3, 4, 5\}$, and we set

$$Q \leftarrow \{(\{2, 3\}, \{4, 5\})\} .$$

- The problem is infeasible, i.e., this node can be pruned.

Processing Node 12 Using BFS

- We solve Problem (5.10) using the sets $\mathcal{P} = \{2, 3\}$ and $\mathcal{D} = \{4, 5\}$, and we set $Q \leftarrow \emptyset$.
- The problem is infeasible, i.e., this node can be pruned.
- Now, $Q = \emptyset$ holds, i.e., there are no more nodes that need to be explored. We have $\mathcal{U} < \infty$ and, thus, the optimal solution to the bilevel problem (5.9) is given by (x^*, y^*, λ^*).

Let us now compare the two different search strategies. Using the depth-first search strategy, we obtain a first feasible solution after the investigation of 11 nodes. To verify the optimality of this solution, one more node problem needs to be solved, i.e., optimality is proven after 12 nodes in total. Using the breadth-first search strategy, we obtain a first feasible solution after the investigation of 6 nodes, which yields an improved upper bound on the objective function value. The algorithm finds the optimal solution after 10 nodes. However,

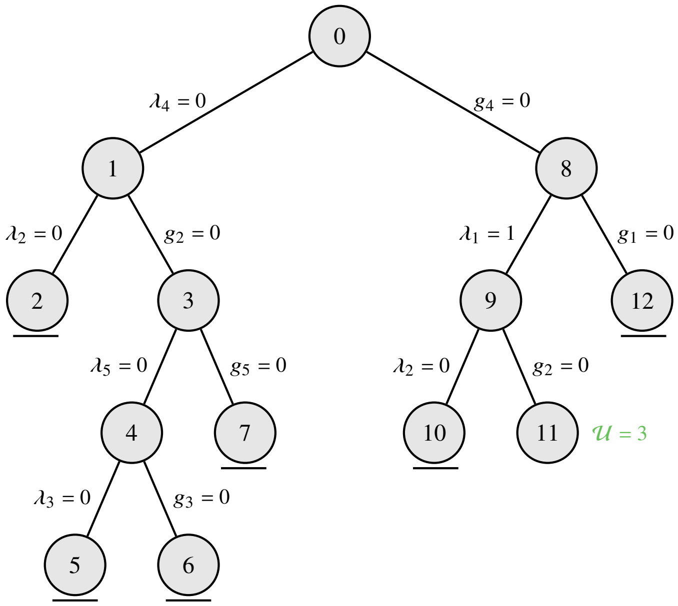


Figure 5.3 The branch-and-bound search tree for the bilevel problem (5.9) using DFS.

to verify optimality, 12 nodes need to be solved in total. For a visualization of both tree search procedures, see Figures 5.3 and 5.4. △

Exercise 5.12 Consider the linear bilevel problem in Example 5.11. Verify the solution found in this example by solving the problem using a general-purpose MILP solver and SOS1-type constraints.

Exercise 5.13 Consider the linear bilevel problem (3.10) in Exercise 3.8 again. You already derived the KKT reformulation of this bilevel problem in Exercise 3.8. Now solve the KKT reformulation using a general-purpose MILP solver and special ordered sets of type 1 (SOS1).

5.3 A Penalty Alternating Direction Method

We have seen in the previous chapters that LP-LP bilevel problems are, in general, hard optimization problems—both in theory and practice. In such a situation, it can be reasonable to also consider heuristics, i.e., methods that do not provably compute a global minimizer in finite time. Instead, such heuristics

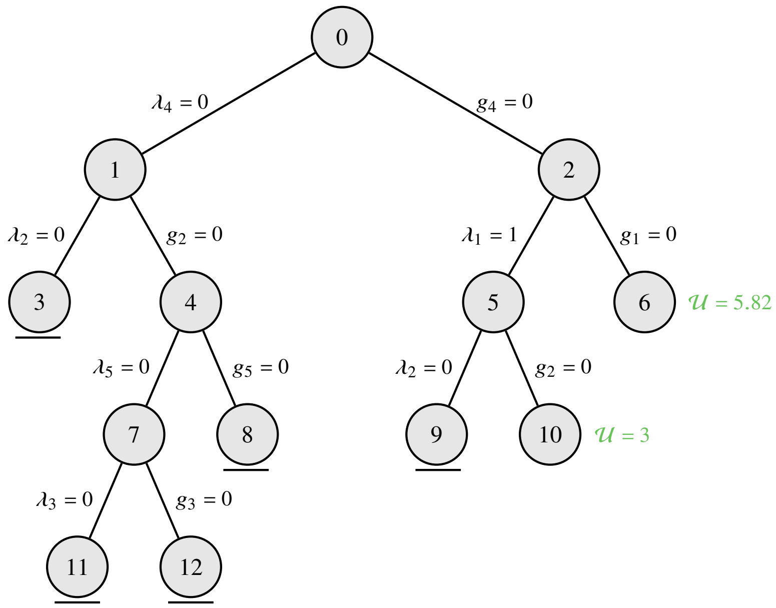


Figure 5.4 The branch-and-bound search tree for the bilevel problem (5.9) using BFS.

usually try to compute feasible points quickly but typically do not provide any optimality guarantee for their output. In this section, we derive such a heuristic for LP-LP bilevel problems.

In contrast to the KKT reformulation that we used to set up the branch-and-bound method in the last section, we now start with the reformulation based on the strong-duality theorem. This means that we consider Problem (3.7) again, i.e.,

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\ & D^\top \lambda = d, \; \lambda \geq 0, \\ & d^\top y \leq (b - Cx)^\top \lambda. \end{aligned}$$

As we have discussed in Section 3.3, the “only” nasty aspect of the latter reformulation is the strong-duality inequality

$$d^\top y \leq (b - Cx)^\top \lambda. \tag{5.11}$$

This constraint includes the bilinear term

$$x^\top C^\top \lambda,$$

which is nonconvex because the upper-level primal variables x and the lower-level dual variables λ are both variables of the strong-duality-based reformulation. The key idea now is to split the reformulated problem (3.7) into two sub-problems that are much easier to solve because they are split along this bilinearity.

Before we do so, let us briefly review the alternating direction method (ADM) and an extension of this method—the penalty ADM (PADM). This is done in the next Section 5.3.1 for single-level optimization problems. Afterward, in Section 5.3.2, we then discuss how these methods can be used to compute a stationary point of the strong-duality-based single-level reformulation of the linear bilevel problem.

5.3.1 A Penalty Alternating Direction Method for Single-Level Problems

We start with discussing general ADMs. To this end, we consider a single-level optimization problem of the form

$$\begin{aligned} \min_{x,y} \quad & f(x, y) \\ \text{s.t.} \quad & g(x, y) = 0, \ h(x, y) \geq 0, \\ & x \in X \subseteq \mathbb{R}^n, \ y \in Y \subseteq \mathbb{R}^m, \end{aligned} \tag{5.12}$$

with variable vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. The feasible set of this problem is abbreviated by

$$\mathcal{F} := \{(x, y) \in X \times Y : g(x, y) = 0, \ h(x, y) \geq 0\}.$$

Here, we use the notation $\mathcal{F}$ for the feasible set as before. Of course, this feasible set should not be confused with the bilevel-feasible set of Definition 1.11. For discussing the theoretical properties of ADMs, we need the following assumption.

Assumption 5.14 The objective function $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and the constraint functions $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^k$ and $h : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^\ell$ are continuous. Moreover, the sets X and Y are non-empty and compact.

A standard ADM works as follows. Given an iterate (x^i, y^i) in iteration i, we first solve Problem (5.12) with y fixed to y^i. We then obtain a new x-iterate x^{i+1}. We now fix x to this new iterate x^{i+1}, solve Problem (5.12), and obtain y^{i+1}. Repeating these two steps yields the method that is given in Algorithm 3.

Algorithm 3 A Standard Alternating Direction Method

Input: An instance of Problem (5.12)

1: Set $i \leftarrow 0$ and choose initial values $(x^i, y^i) \in X \times Y$.
2: **while** (x^i, y^i) is not a partial minimizer of (5.12) **do**
3: Compute
$$x^{i+1} \in \arg\min_x \left\{ f(x, y^i) : g(x, y^i) = 0,\ h(x, y^i) \geq 0,\ x \in X \right\}.$$
4: Compute
$$y^{i+1} \in \arg\min_y \left\{ f(x^{i+1}, y) : g(x^{i+1}, y) = 0,\ h(x^{i+1}, y) \geq 0,\ y \in Y \right\}.$$
5: Increase $i \leftarrow i + 1$.
6: **return** (x^i, y^i)

Note that one usually tries to choose an initial point (x^0, y^0) so that the first problem in the first iteration is feasible. Together with Assumption 5.14, this guarantees that all problems that need to be solved have an optimal solution.

Under certain mild assumptions, one can show that the ADM of Algorithm 3 converges to a so-called partial minimizer.

Definition 5.15 (Partial Minimizer) A feasible point $(x^*, y^*) \in \mathcal{F}$ of Problem (5.12) is called a *partial minimizer* if

$$\begin{aligned} f(x^*, y^*) &\leq f(x, y^*) \quad \text{for all } (x, y^*) \in \mathcal{F}, \\ f(x^*, y^*) &\leq f(x^*, y) \quad \text{for all } (x^*, y) \in \mathcal{F} \end{aligned}$$

holds.

The following theorem shows under which conditions it is guaranteed that the ADM of Algorithm 3 converges to partial minimizers.

Theorem 5.16 (See Gorski et al. (2007)) *Let $(x^i, y^i)_{i=0}^\infty$ be a sequence of iterates generated by Algorithm 3. Suppose that Assumption 5.14 holds and that the solution to one of the two optimization problems solved in each iteration of Algorithm 3 is always unique. Then, every convergent subsequence of $(x^i, y^i)_{i=0}^\infty$ converges to a partial minimizer of Problem (5.12). For two limit points z, z' of such subsequences, it holds that $f(z) = f(z')$.*

For what follows, we also note that stronger convergence results can be obtained if stronger assumptions on f and $\mathcal{F}$ are made. For later reference, we state these results as a corollary; see Geißler et al. (2017, 2018), Gorski et al. (2007), and Wendell and Hurter (1976) for the proofs and more detailed discussions.

Corollary 5.17 *Suppose that the assumptions of Theorem 5.16 are satisfied. Then, the following holds:*

(i) If f is continuously differentiable, then every convergent subsequence of $(x^i, y^i)_{i=0}^\infty$ converges to a stationary point of Problem (5.12).
(ii) If f is continuously differentiable and if f and $\mathcal{F}$ are convex, then every convergent subsequence of $(x^i, y^i)_{i=0}^\infty$ converges to a global minimizer of Problem (5.12).

Let us now comment on the main rationale of the alternating direction method discussed so far. Problem (5.12) can be seen as a *quasi block-separable problem*, where the blocks are given by the variables x and y as well as their respective feasible sets X and Y. We add the notion "quasi" here because there are still the constraints g and h that couple the feasible sets of the two blocks. The main idea of an ADM is to solve alternatingly in the directions of the blocks separately until the method stagnates.

In practice, it can often be observed that an even stronger decoupling of Problem (5.12) is favorable (Boyd et al. 2011; Geißler et al. 2015, 2017, 2018). Thus, we now go one step further and relax the coupling constraints g and h.[2] To this end, we introduce the *weighted ℓ_1-penalty function*

$$\phi_1(x, y; \mu, \rho) := f(x, y) + \sum_{t=1}^{k} \mu_t |g_t(x, y)| + \sum_{t=1}^{\ell} \rho_t [h_t(x, y)]^-.$$

Here, we set $[\alpha]^- := \max\{0, -\alpha\}$ and μ and ρ are non-negative vectors of *penalty parameters* of size k and ℓ, respectively. With this notation at hand, we can now describe the penalty alternating direction method (PADM). The PADM consists of an inner and an outer loop. In the inner loop, we apply the standard ADM of Algorithm 3 to the penalty problem

$$\min_{x,y} \quad \phi_1(x, y; \mu, \rho) \quad \text{s.t.} \quad x \in X,\ y \in Y. \tag{5.13}$$

If this inner loop terminates with a partial minimizer of Problem (5.13), we check whether the coupling constraints $g(x, y) = 0$ and $h(x, y) \geq 0$ are satisfied. If this is the case, we terminate with the obtained feasible point. If not, we increase the penalty parameters and proceed with computing a partial minimizer of the updated penalty problem in the next inner loop. The overall method is formally stated in Algorithm 4. For later reference, we also state the convergence

[2] Note that this notion of "coupling constraints" is different than the one considered for bilevel optimization problems; see Section 1.2.

results for the PADM in Algorithm 4. The proofs of this section can be found in Geißler et al. (2017), where further technical aspects such as penalty parameter initialization and update strategies, additional termination criteria, warmstarts, etc. are discussed as well.

Algorithm 4 The ℓ_1-Penalty Alternating Direction Method

Input: An instance of Problem (5.12)

1: Choose initial values $(x^{0,0}, y^{0,0}) \in X \times Y$ and initial penalty parameters $\mu^0, \rho^0 \geq 0$.
2: **for** $j = 0, 1, \ldots$ **do**
3: Set $i \leftarrow 0$.
4: **while** $(x^{j,i}, y^{j,i})$ is not a partial minimizer of (5.13) with $\mu = \mu^j$ and $\rho = \rho^j$ **do**
5: Compute $x^{j,i+1} \in \arg\min_x \{\phi_1(x, y^{j,i}; \mu^j, \rho^j) : x \in X\}$.
6: Compute $y^{j,i+1} \in \arg\min_y \{\phi_1(x^{j,i+1}, y; \mu^j, \rho^j) : y \in Y\}$.
7: Set $i \leftarrow i + 1$.
8: **if** $g(x^{j,i}, y^{j,i}) = 0$ and $h(x^{j,i}, y^{j,i}) \geq 0$ **then**
9: **return** $(x^{j,i}, y^{j,i})$
10: **else**
11: Choose new penalty parameters $\mu^{j+1} \geq \mu^j$ and $\rho^{j+1} \geq \rho^j$.

Theorem 5.18 *Suppose that Assumption 5.14 holds. Then, Algorithm 4 either terminates after a finite number of iterations in Line 9 or creates an infinite sequence of iterates and penalty parameters $\mu_t^j \nearrow \infty$ for all $t = 1, \ldots, k$ and $\rho_t^j \nearrow \infty$ for all $t = 1, \ldots, \ell$. In the latter case, let $(x^j, y^j)_{j=0}^\infty$ be a sequence of partial minimizers of (5.13), for $\mu = \mu^j$ and $\rho = \rho^j$, generated by Algorithm 4 with $(x^j, y^j) \to (x^*, y^*)$. Then, there exist weights $\bar{\mu}, \bar{\rho} \geq 0$ such that (x^*, y^*) is a partial minimizer of the* weighted ℓ_1-feasibility measure

$$\chi_{\bar{\mu},\bar{\rho}}(x, y) := \sum_{t=1}^{k} \bar{\mu}_t |g_t(x, y)| + \sum_{t=1}^{\ell} \bar{\rho}_t [h_t(x, y)]^-.$$

If, in addition, (x^*, y^*) is feasible for the original problem (5.12), the following holds:

(i) The point (x^*, y^*) is a partial minimizer of Problem (5.12).
(ii) If f is continuously differentiable, then (x^*, y^*) is a stationary point of Problem (5.12).
(iii) If f is continuously differentiable and if f and $\mathcal{F}$ are convex, then (x^*, y^*) is a global minimizer of Problem (5.12).

5.3.2 Applying the PADM to Linear Bilevel Problems

Now, we apply the PADM to the single-level reformulation (3.7) of the original LP-LP bilevel problem. As already discussed, the problematic constraint is the strong-duality inequality (5.11). This is due to two reasons. On the one hand, it is the only nonlinear constraint and the reason for the nonconvexity of the problem. On the other hand, it is the only constraint that couples the variable blocks (x, y) and λ. Thus, we relax this constraint to obtain the penalty problem reformulation

$$\min_{x,y,\lambda} \quad c_x^\top x + c_y^\top y + \rho \left[b^\top \lambda - x^\top C^\top \lambda - d^\top y\right]^- \tag{5.14a}$$

$$\text{s.t.} \quad Ax + By \geq a, \; Cx + Dy \geq b, \tag{5.14b}$$

$$D^\top \lambda = d, \; \lambda \geq 0. \tag{5.14c}$$

Note that we have no penalty parameters μ because we only penalize an inequality constraint. Moreover, we smoothen the penalty term by exploiting weak duality of the lower level that is equivalent to the fact that

$$d^\top y - b^\top \lambda + x^\top C^\top \lambda \geq 0$$

holds for every feasible point of Problem (5.14). Thus,

$$\begin{aligned} &\left[b^\top \lambda - x^\top C^\top \lambda - d^\top y\right]^- \\ = \;& \max\{0, d^\top y - b^\top \lambda + x^\top C^\top \lambda\} \\ = \;& d^\top y - b^\top \lambda + x^\top C^\top \lambda \end{aligned}$$

holds and we obtain the equivalent penalty problem

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y + \rho \left(d^\top y - b^\top \lambda + x^\top C^\top \lambda\right) \\ \text{s.t.} \quad & \text{(5.14b)–(5.14c)}, \end{aligned} \tag{5.15}$$

which is a smooth but still nonconvex optimization problem. To be more specific, Problem (5.15) is a nonconvex quadratic optimization problem. A closer look also reveals that Problem (5.15) is exactly of the form in (5.13) if the first block of variables is (x, y) and if the second block of variables is λ. Thus, the splitting of the feasible set is obtained by identifying[3]

$$\begin{aligned} \text{“}x \in X\text{”} \quad &\longleftrightarrow \quad \text{Constraints (5.14b)}, \\ \text{“}y \in Y\text{”} \quad &\longleftrightarrow \quad \text{Constraints (5.14c)}. \end{aligned} \tag{5.16}$$

[3] Note that in (5.16), x and y denote the variable blocks of the general problem formulation (5.12). All other occurrences of x and y in this section stand for the respective upper- and lower-level variables of the considered bilevel problem.

Note further that this splitting corresponds to a primal-dual splitting of the single-level reformulation (3.7). Given this splitting, the first sub-problem that needs to be solved if we apply Algorithm 4 to Problem (3.7) reads

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y + \rho \left((C^\top \bar{\lambda})^\top x + d^\top y \right) \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \end{aligned} \tag{5.17}$$

where $\bar{\lambda}$ is a given constant vector and where we already omit the constant objective function term $b^\top \bar{\lambda}$. This problem has the same feasible set as the single-level relaxation of the original LP-LP bilevel problem. Thus, under the classic assumption that the shared constraint set Ω is non-empty and bounded, it always has an optimal solution. However, the upper-level objective function coefficients are modified in dependence of the penalty parameter ρ, the current dual estimate $\bar{\lambda}$, and the lower-level objective function coefficients d.

The second sub-problem is equivalent to

$$\begin{aligned} \max_{\lambda} \quad & (b - C\bar{x})^\top \lambda \\ \text{s.t.} \quad & D^\top \lambda = d, \; \lambda \geq 0, \end{aligned} \tag{5.18}$$

which is exactly the dual lower-level problem. Here, three interesting aspects can be observed. First, the second sub-problem only depends on the primal upper-level variables $\bar{x}$ and not on the primal lower-level variables $\bar{y}$. The reason is that we again omit constant terms in the objective function, i.e., $c_x^\top \bar{x} + c_y^\top \bar{y} + \rho d^\top \bar{y}$. Second, Sub-Problem (5.18) does not depend on the penalty parameter ρ anymore because it only scales the remaining objective function. Third, the second sub-problem may be unbounded for a given estimate for $\bar{x}$. This means that the primal lower-level problem is infeasible for the upper-level decision $\bar{x}$. We do not go into the details on how to resolve this here but refer to Kleinert and Schmidt (2021), where more information is given.

When it comes to applying the convergence theory from Section 5.3.1 to the bilevel setup considered here, one has to be careful. The reason is that the dual part in (5.15) does not need to be bounded, which is, however, required in the basic theory; see Assumption 5.14. We again do not go into the details but refer to the papers already cited in this section as well as to the more recent article by Lefebvre and Schmidt (2026), from which the following theoretical results are taken.

Theorem 5.19 *Consider the jth inner loop of Algorithm 4 applied to Problem (3.7) for a fixed penalty parameter $\rho^j > 0$ and let $(x^{j,i}, y^{j,i}, \lambda^{j,i})_{i=0}^{\infty}$ be the generated sequence of iterates. Let each sub-problem be solvable.*

(i) *If the sequence of iterates* $(x^{j,i}, y^{j,i}, \lambda^{j,i})_{i=0}^{\infty}$ *is contained in a compact set, then it has at least one accumulation point* $(x^{j*}, y^{j*}, \lambda^{j*})$.

(ii) *If, in addition, all accumulation points* $(x^{j*}, y^{j*}, \lambda^{j*})$ *are such that one of the two sub-problems has a unique solution, then all accumulation points are partial minimizers of the penalized problem* (5.15).

(iii) *If, in addition, all accumulation points* $(x^{j*}, y^{j*}, \lambda^{j*})$ *are such that both sub-problems have a unique solution, then* $(x^{j,i}, y^{j,i}, \lambda^{j,i}) \to (x^{j*}, y^{j*}, \lambda^{j*})$.

Regarding the points obtained in the inner loop, two situations are possible. First, these points may have a *strong-duality error*

$$\chi^{\mathrm{sd}}(x, y, \lambda) := \left| d^\top y - b^\top \lambda + x^\top C^\top \lambda \right|$$

of zero, which means that the point is bilevel feasible. Second, these points may have a positive strong-duality error $\chi^{\mathrm{sd}}(x, y, \lambda) > 0$ and are thus not bilevel feasible. The latter case then motivates to proceed with a larger penalization of the strong-duality term.

Theorem 5.18 implies the following main convergence result.

Theorem 5.20 *Suppose that* $\rho^j \nearrow \infty$ *holds and let* $(x^j, y^j, \lambda^j)_{j=0}^{\infty}$ *be a sequence of partial minimizers of* (5.15) *for* $\rho = \rho^j$ *generated by the inner loop of Algorithm 4 with* $(x^j, y^j, \lambda^j) \to (x^*, y^*, \lambda^*)$. *Then,* (x^*, y^*, λ^*) *is a partial minimizer of the strong-duality error* χ^{sd}. *If, in addition,* $\chi^{sd}(x^*, y^*, \lambda^*) = 0$ *holds, then* (x^*, y^*, λ^*) *is a stationary point of* (3.7) *and* (x^*, y^*) *is feasible for the original bilevel problem* (5.1).

We close the discussion of applying the PADM to linear bilevel problems with three remarks.

Remark 5.21 Note that Theorem 5.20 "only" makes a statement regarding stationary points of the single-level reformulation (3.7) and not about the original bilevel problem. In general, a stationary point of Problem (3.7) does not need to be a stationary point of the bilevel problem; see Section 3.6 and Dempe and Dutta (2012) for the equivalent setting of a single-level reformulation based on the KKT conditions of the lower level. Although there is thus no theoretical quality guarantee for the bilevel-feasible points obtained by this method, it is shown in Kleinert and Schmidt (2021) that, in practice, the quality of the obtained feasible points is very good.

Remark 5.22 A crucial assumption of Theorem 5.19 (which is also implicitly present in Theorem 5.20) is that one of the two PADM sub-problems—meaning either (5.17) or (5.18)—always needs to have a unique solution. However, this is usually hard to ensure in advance.

Remark 5.23 The approach described in this section can also be applied to bilevel problems for which the upper level is more general. For instance, it may contain integrality constraints and a convex-quadratic objective function so that the resulting bilevel problem is of the form

$$\begin{aligned}
\min_{x,y} \quad & \frac{1}{2}x^\top Hx + c_x^\top x + \frac{1}{2}y^\top Gy + c_y^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \\
& x_i \in \mathbb{Z} \quad \text{for all } i \in I \subseteq \{1, \dots, n\}, \\
& y \in \arg\min_{\bar{y}} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\},
\end{aligned}$$

with symmetric and positive semi-definite matrices H and G in appropriate dimensions. This does not affect the second PADM sub-problem at all. However, the first PADM sub-problem (5.17) is a convex-quadratic problem (QP) for $I = \emptyset$ or a mixed-integer convex-quadratic problem (MIQP) for $I \neq \emptyset$. Solving (MI)QPs to global optimality in every iteration may have a significant impact on the performance of the PADM. For a numerical analysis, we refer to Kleinert and Schmidt (2021).

Exercise 5.24 Implement the penalty alternating direction method of Algorithm 4 for computing feasible points of LP-LP bilevel problems. Use a general-purpose LP solver to solve the linear sub-problems that arise in the algorithm. Validate your implementation by applying it to Problem (1.6) from Example 1.20. If required, use the penalty parameter update rule $\rho^{j+1} = 2\rho^j$.

(i) Use the starting point $(x^{0,0}, y^{0,0}) = (1, 1)$ and $\rho^0 = 1$.
(ii) Use the starting point $(x^{0,0}, y^{0,0}) = (4, 1)$ and $\rho^0 = 1$.
(iii) Use the starting point $(x^{0,0}, y^{0,0}) = (1, 1)$ and $\rho^0 = 0.15$.

In all three cases, you can start the inner loop with solving the dual sub-problem (5.18) first.

What do you observe when changing the starting point $(x^{0,0}, y^{0,0})$ or the initial penalty parameter ρ^0?

5.4 What You Should Know Now!

1. What is the main idea of the Kth-best algorithm?
2. How is the Kth-best algorithm formally defined?

3. What are the crucial steps of the Kth-best algorithm?
4. What is the main idea of the branch-and-bound method for LP-LP bilevel problems? What do we branch on and why?
5. How is the branch-and-bound method for LP-LP bilevel problems stated formally?
6. What is a relaxation in general?
7. Which relaxations are considered for the problems in the branch-and-bound search tree?
8. What is the claim of the bounding lemma?
9. What is the claim of the branching lemma?
10. What theoretical statement do you know about the branch-and-bound method for LP-LP bilevel problems?
11. What are special ordered sets of type 1?
12. What is an alternating direction method (ADM)?
13. What is a partial minimizer?
14. What is the general convergence result for ADMs and what assumptions are required?
15. What can we gain if we make stronger assumptions regarding the properties of the underlying problem?
16. What is the idea behind moving from ADM to the penalty ADM (PADM)?
17. What is the convergence result for the PADM?
18. How do we apply the PADM to LP-LP bilevel problems and why do we do it exactly like this?
19. What theoretical result do we get for the PADM when applied to LP-LP bilevel problems?

PART THREE

MIXED-INTEGER BILEVEL PROBLEMS

6
General Properties

In what follows, we consider mixed-integer linear bilevel problems (bilevel MILPs), which are defined as

$$\min_{x \in X, y} \quad c_x^\top x + c_y^\top y \tag{6.1a}$$

$$\text{s.t.} \quad Ax + By \geq a, \tag{6.1b}$$

$$y \in \operatorname*{arg\,min}_{\bar{y} \in Y} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \tag{6.1c}$$

with $c_x \in \mathbb{R}^{n_x}$, $c_y, d \in \mathbb{R}^{n_y}$, $A \in \mathbb{R}^{m \times n_x}$, $B \in \mathbb{R}^{m \times n_y}$, $a \in \mathbb{R}^m$, $C \in \mathbb{R}^{\ell \times n_x}$, $D \in \mathbb{R}^{\ell \times n_y}$, and $b \in \mathbb{R}^\ell$. Here, the sets X and Y are used to impose integrality constraints on (a subset of) the x- and y-variables, respectively. Moreover, we assume, if not stated otherwise, that any variable bounds are encoded in the systems $Ax + By \geq a$ and $Cx + Dy \geq b$. Before we start with some central definitions and general properties of bilevel MILPs, let us first have a look at two illustrative and famous examples from the literature.

Example 6.1 (See Example 2 in Moore and Bard (1990)) We consider the bilevel problem

$$\max_{x \in \mathbb{Z}, y} \quad -x - 2y$$

$$\text{s.t.} \quad x \geq 0, \ y \in \mathcal{S}(x),$$

where $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized lower-

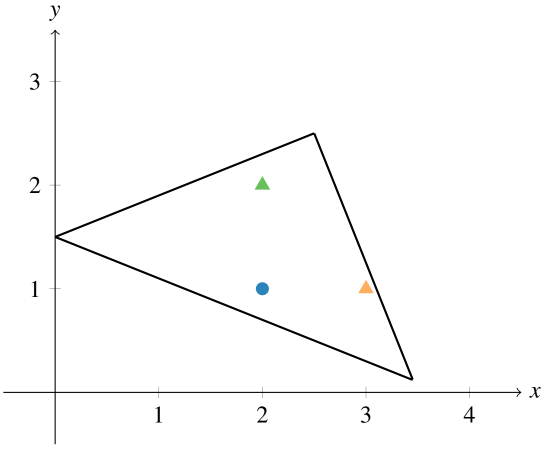


Figure 6.1 The bilevel problem in Example 6.1. The discrete points (dot and triangles) are feasible for the shared constraint set, whereas triangles represent bilevel-feasible points. The point $(2, 1)$, i.e., the blue dot, is the optimal solution to the single-level relaxation of the bilevel problem, whereas the orange triangle, i.e., the point $(3, 1)$, is the optimal solution to the bilevel problem.

level problem

$$\begin{aligned}
\max_{y \in \mathbb{Z}} \quad & y \\
\text{s.t.} \quad & -x + 2.5y \leq 3.75, \\
& x + 2.5y \geq 3.75, \\
& 2.5x + y \leq 8.75, \\
& y \geq 0.
\end{aligned}$$

An illustration of the problem is given in Figure 6.1. The shared constraint set contains three integer points: $(2, 1)$, $(2, 2)$, and $(3, 1)$. Note that the point $(2, 1)$, which is represented with a blue dot in Figure 6.1, is the optimal solution to the single-level relaxation of the bilevel problem. If the leader chooses $x = 2$, the follower responds with $y = 2$, leading to an upper-level objective function value of -6. If the leader decides for $x = 3$, the follower optimally reacts with $y = 1$, leading to an objective function value of -5. Thus, $(x^*, y^*) = (3, 1)$ is the optimal solution to the problem and the optimal objective function value of the leader is -5. △

Example 6.2 (See Example 1 in Moore and Bard (1990)) We consider the bilevel problem

$$\begin{aligned} \min_{x \in \mathbb{Z}, y} \quad & -x - 10y \\ \text{s.t.} \quad & y \in \mathcal{S}(x), \end{aligned}$$

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

$$\begin{aligned} \min_{y \in \mathbb{Z}} \quad & y \\ \text{s.t.} \quad & -25x + 20y \leq 30, \\ & x + 2y \leq 10, \\ & 2x - y \leq 15, \\ & 2x + 10y \geq 15. \end{aligned}$$

The problem is depicted in Figure 6.2. The blue point $(2, 4)$ is the optimal solution to the single-level relaxation of the bilevel problem. The unique optimal solution to the bilevel problem is the point $(x^*, y^*) = (2, 2)$ with an optimal objective function value of -22. Note that this point lies in the interior of the convex hull of the integer points of the single-level relaxation, which is an integer linear problem. This is in contrast to bilevel LPs, whose optimal solution is always attained at a vertex of the shared constraint set; see Theorem 4.14. △

Moore and Bard (1990) initiated the studies of bilevel problems with discrete variables. Their illustrative examples (Examples 6.1 and 6.2) have been widely used in the literature to highlight important differences and challenges arising in (mixed-)integer linear bilevel optimization. We return to these examples in the following chapters and use them to illustrate the key concepts and solution techniques in mixed-integer linear bilevel optimization.

The remainder of this chapter is organized as follows. In Section 6.1, we start with some central definitions that we use throughout the current and the following chapters on bilevel MILPs. Afterward, in Section 6.2, we take a closer look at the single-level relaxation of Problem (6.1) and discuss some properties regarding its relation to the original bilevel problem. In Section 6.3, we elaborate on the existence of optimal solutions to Problem (6.1). The complexity of bilevel MILPs is discussed in Section 6.4. Finally, we conclude with connections between bilevel MILPs and other single- or multilevel problems in Section 6.5.

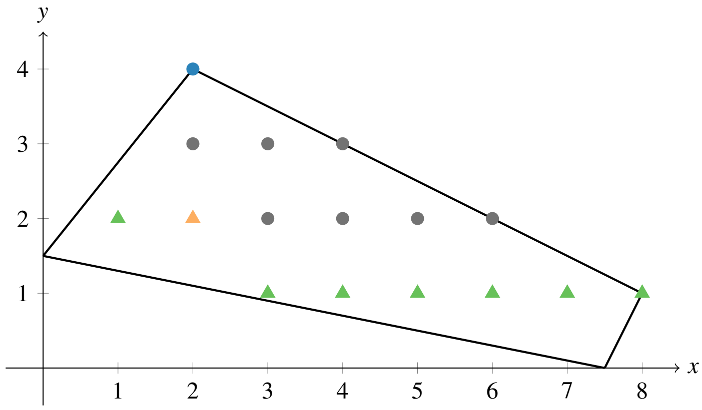


Figure 6.2 The bilevel problem in Example 6.2. The discrete points (dots and triangles) are feasible for the shared constraint set, whereas triangles represent bilevel-feasible points. The point $(2, 4)$, i.e., the blue dot, is the optimal solution to the single-level relaxation of the bilevel problem, whereas the orange triangle, i.e., the point $(2, 2)$, is the optimal solution to the bilevel problem. Taken and modified from Moore and Bard (1990).

6.1 Central Definitions

We repeat some central definitions, which we use throughout the current and the following chapters. As before, the shared constraint set Ω of Problem (6.1) is defined as the set of points $(x, y) \in X \times Y$ satisfying all upper- and lower-level constraints, i.e.,

$$\Omega := \{(x, y) \in X \times Y \colon Ax + By \geq a,\ Cx + Dy \geq b\}.$$

Its projection onto the x-space is given by

$$\Omega_x := \{x \colon \exists y \text{ with } (x, y) \in \Omega\}.$$

The feasible set $\mathcal{F}$ of the bilevel problem (6.1) consists of all points $(x, y) \in \Omega$ for which the vector y is an optimal solution to the x-parameterized lower-level problem (6.1c), i.e.,

$$\mathcal{F} := \left\{(x, y) \colon (x, y) \in \Omega,\ d^\top y \leq \varphi(x)\right\}.$$

Here, $\varphi(x)$ is the follower's optimal-value function, which is defined as

$$\varphi(x) := \inf_{y \in Y} \left\{d^\top y \colon Dy \geq b - Cx\right\}.$$

Note that the optimal-value function $\varphi(x)$ corresponds to a parametric single-level mixed-integer linear problem (MILP). In particular, this means that it is generally nonconvex, not continuous, and very difficult to describe.

Let us also recap our definition of linking variables from Chapter 1. Because the lower-level objective function in Problem (6.1) does not depend on x, all linking variables appear in the constraints of the lower-level problem. The set of indices associated with these variables is given by

$$L := \{i \in \{1, \dots, n_x\} : C_{\cdot i} \neq 0\},$$

where $C_{\cdot i}$ is the ith column of the matrix C.

Definition 6.3 (LP Relaxation of the Single-Level Relaxation) The problem of minimizing the upper-level objective over the shared constraint set in which all integrality constraints are omitted, i.e.,

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega} \end{aligned} \tag{6.2}$$

with

$$\bar{\Omega} := \{(x, y) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} : Ax + By \geq a,\ Cx + Dy \geq b\},$$

is called the *continuous relaxation (or LP relaxation) of the single-level relaxation* of Problem (6.1). Moreover, we call the set $\bar{\Omega}$ the *continuous relaxation (or LP relaxation) of the shared constraint set*.

6.2 A Glimpse at the Single-Level Relaxation

In the following chapters, the (LP relaxation of the) single-level relaxation plays a central role for the development of solution methods for bilevel MILPs. In particular, these methods are based on the following observation.

Observation 6.4 By definition, we have $\mathcal{F} \subseteq \Omega \subseteq \bar{\Omega}$. Thus, the single-level relaxation of Problem (6.1), i.e., the mixed-integer linear problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \Omega,$$

is indeed a relaxation of the original bilevel problem (6.1). Moreover, the LP relaxation of the single-level relaxation, i.e., the linear problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \bar{\Omega},$$

is another (but usually much weaker) relaxation of the bilevel MILP.

To ensure that an optimal solution to the (LP relaxation of the) single-level relaxation exists, the following assumption is frequently made in the literature.

Assumption 6.5 The shared constraint set Ω of Problem (6.1) is non-empty and bounded.

Note that this is the same assumption we make in Chapter 4 for the analysis of LP-LP bilevel problems. Because Ω is described by a finite number of affine and, thus, continuous functions, the set Ω is closed and, as a result, it is compact by Assumption 6.5. We can thus apply the Weierstraß theorem to prove the existence of optimal solutions to the single-level relaxation of Problem (6.1), which is what we do in the following lemma.

Lemma 6.6 *Suppose that Assumption 6.5 holds. Then, the single-level relaxation of Problem (6.1) and its LP relaxation (6.2) have an optimal solution.*

Proof: The claim immediately follows from the Weierstraß theorem. □

Lemma 6.6 states that, under Assumption 6.5, the optimal objective function values of the single-level relaxation and its LP relaxation are finite. Because of Observation 6.4, this means that they provide valid lower bounds for the optimal objective function value of the bilevel MILP (6.1). Nevertheless, let us have a look at what could happen if we were to relax Assumption 6.5.

Observation 6.7 An empty shared constraint set, i.e., $\Omega = \emptyset$, implies the infeasibility of the single-level relaxation of Problem (6.1). This, in turn, implies that the bilevel problem itself is infeasible. However, the bilevel problem can also be infeasible if $\Omega \neq \emptyset$ holds.

Recall from Chapter 4 that an unbounded lower-level problem also leads to infeasibility of the overall bilevel problem. This has been shown in Lemma 4.5 for linear bilevel problems and we also discussed how unboundedness of the lower level can be detected in advance by solving an auxiliary LP; see Theorem 4.6. Let us now state the analogous results for bilevel MILPs.

Lemma 6.8 (See Lemma 2 in Xu and Wang (2014)) *Let $I \subseteq \{1, \dots, n_y\}$ be given and consider Problem (6.1) with $Y = \{y \in \mathbb{R}^{n_y} : y_i \in \mathbb{Z} \text{ for all } i \in I\}$. Suppose there exists a point $x \in \Omega_x$ for which the lower-level problem (6.1c) is unbounded. Then, the bilevel MILP (6.1) is infeasible.*

Exercise 6.9 Prove Lemma 6.8. (*Hint:* Take a look into the proof of Lemma 4.5 again.)

Theorem 6.10 (See Theorem 1 in Fischetti et al. (2018a)) *Suppose that the matrix D is rational, i.e., $D \in \mathbb{Q}^{\ell \times n_y}$, and that the shared constraint set Ω of*

Problem (6.1) is non-empty. Moreover, let v^* be the optimal objective function value of the linear problem

$$\begin{aligned} \min_{\Delta y \in \mathbb{R}^{n_y}} \quad & d^\top \Delta y \\ \text{s.t.} \quad & D\Delta y \geq 0, \\ & -1 \leq \Delta y \leq 1. \end{aligned} \tag{6.3}$$

Then, for any $x \in \Omega_x$, the following is true. If $v^* < 0$ holds, the lower-level problem is unbounded. Otherwise, it has an optimal solution.

The proof of Theorem 6.10 uses the same arguments as the proof of Theorem 4.6, but we now need to be a bit more careful because of the integrality constraints captured in the set Y. Compared to Theorem 4.6, we thus impose an additional assumption on the coefficients of the constraint matrix D, which is necessary to extend the result to the mixed-integer linear case.

Proof of Theorem 6.10: Problem (6.3) is feasible and bounded. Hence, an optimal solution Δy^* with value v^* exists. Because the matrix D is rational by assumption, we can w.l.o.g. suppose that Δy^* is rational as well. Hence, we can multiply Δy^* by a positive scalar to obtain $\Delta\bar{y}$ that satisfies $\Delta\bar{y}_i \in \mathbb{Z}$ for all $i \in I$ and $D\Delta\bar{y} \geq 0$. On the one hand, if $v^* < 0$ holds, we also have $d^\top\Delta\bar{y} < 0$. Moreover, there exists a point $\bar{x} \in \Omega_x$ so that there exists a feasible point $\bar{y}$ for the $\bar{x}$-parameterized lower-level problem (6.1c) because $\Omega \neq \emptyset$ holds. Using the insights from the proof of Lemma 4.5, this implies that $\Delta\bar{y}$ is an improving direction for the lower-level problem at the given point $(\bar{x}, \bar{y}) \in \Omega$. Hence, the lower-level problem is unbounded.

On the other hand, if $v^* \geq 0$ holds, the lower-level problem (6.1c) cannot be unbounded and, thus, it is solvable. □

In addition to the results by Fischetti et al. (2018a) and Xu and Wang (2014), which address the unboundedness of bilevel MILPs more from a computational perspective, the complexity of deciding unboundedness in (mixed-integer) linear bilevel optimization has further been addressed by Rodrigues (2025) and Rodrigues et al. (2025).

Exercise 6.11 Consider the bilevel problem

$$\begin{aligned} \min_{x \in \mathbb{Z}, y} \quad & -x + y \\ \text{s.t.} \quad & 0 \leq x \leq 8, \\ & y \in \arg\min_{\bar{y} \in \mathbb{Z}} \{d\bar{y} \colon x - 3\bar{y} \geq -12,\ -x - \bar{y} \geq -8\} \end{aligned} \tag{6.4}$$

with $d \in \mathbb{R}$.

(i) Plot the shared constraint set of Problem (6.4) in a coordinate system.
(ii) What can you say about the single-level relaxation of Problem (6.4) regarding infeasibility, unboundedness, or solvability?
(iii) Determine for which values of the parameter $d \in \mathbb{R}$ the bilevel problem (6.4) is infeasible, unbounded, or solvable.
(iv) Take the parameters d determined in (iii) for which the bilevel problem (6.4) is either unbounded or solvable. For both cases, write down the associated LP (6.3), determine an optimal solution to this LP, and use Theorem 6.10 to verify your observations from (ii) and (iii).
(v) Now consider the problem in which some (or all) of the integrality constraints are relaxed. Does this affect the infeasibility, unboundedness, or solvability of either the single-level relaxation or the bilevel problem? Explain your observations.

6.3 Attainability of Optimal Solutions

In the last section, we already shed some light on the infeasibility and the unboundedness of mixed-integer linear bilevel problems. Let us now turn to the question of when the bilevel MILP (6.1) attains an optimal solution. To the best of our knowledge, the first systematic step in this direction was taken by Vicente et al. (1996). In their work, the authors elaborate on the attainability of optimal solutions for three cases of bilevel MILPs, which have the following properties:

(i) only upper-level variables can take discrete values,
(ii) all upper- and lower-level variables are discrete, or
(iii) only lower-level variables can take discrete values.

Vicente et al. (1996) discuss that, under some additional assumptions, an optimal solution to Problem (6.1) exists for Cases (i) and (ii). For Case (iii), however, Moore and Bard (1990) and also Vicente et al. (1996) provide examples demonstrating that the bilevel-feasible set may not be closed and, thus, an optimal solution may not exist. We now show this in the following simpler example, which is taken from Köppe et al. (2010).

Example 6.12 (See Example 1.1 in Köppe et al. (2010)) Consider the problem

$$\begin{aligned} \inf_{x \in \mathbb{R},\, y} \quad & x - y \\ \text{s.t.} \quad & 0 \le x \le 1, \\ & y \in \operatorname*{arg\,min}_{\bar{y} \in \mathbb{Z}} \{\bar{y} \colon \bar{y} \ge x,\ 0 \le \bar{y} \le 1\}, \end{aligned}$$

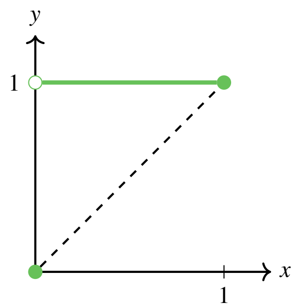


Figure 6.3 The bilevel problem in Example 6.12. The first lower-level constraint is illustrated by the dashed line. The bilevel-feasible region corresponds to the solid green line as well as the points $(0, 0)$ and $(1, 1)$. Taken and modified from Kleinert et al. (2021a).

which is equivalent to

$$\begin{aligned} \inf_{x \in \mathbb{R}} \quad & x - \lceil x \rceil \\ \text{s.t.} \quad & 0 \le x \le 1. \end{aligned}$$

The problem is depicted in Figure 6.3. Note that, in this problem, the infimum of -1 is never attained. Moreover, this example shows that the assumptions that are enough to prove existence of optimal solutions for LP-LP bilevel problems (namely that the problem is feasible and that the shared constraint set is non-empty and bounded) are not enough for the existence of optimal solutions to mixed-integer linear bilevel problems. Finally note that this undesired property even appears for problems that do not have coupling constraints as the example above does not involve any y-dependent upper-level constraints. △

Example 6.12 illustrates that continuous linking variables may lead to situations in which the shared constraint set is non-empty and bounded, the bilevel problem is feasible, no coupling constraints are involved, but the infimum of a bilevel MILP is not attained. As a consequence, it is frequently assumed in the existing literature on mixed-integer bilevel optimization that all linking variables are discrete.

Assumption 6.13 All linking variables are bounded integers, i.e., $x_i \in \mathbb{Z}$, and $\underline{x}_i \le x_i \le \bar{x}_i$ holds for all $i \in L$ with $-\infty < \underline{x}_i \le \bar{x}_i < +\infty$ and $\underline{x}_i, \bar{x}_i \in \mathbb{Z}$.

We can even go one step further and translate Assumption 6.13 into "all upper-level variables are discrete", as it is shown in the next exercise. For further reference, see also Bolusani and Ralphs (2022) and Tahernejad et al. (2020).

Exercise 6.14 Show that, w.l.o.g., all upper-level variables can be assumed to be linking variables. (*Hint:* Use the value-function reformulation of Problem (6.1) and show that all non-linking variables can be moved to the lower-level problem.)

Our goal now is to prove that, if the problem is feasible, an optimal solution to the bilevel MILP (6.1) exists under Assumptions 6.5 and 6.13. To this end, let us first introduce the notion of the *refinement problem*.

Definition 6.15 (Refinement Problem) For a given decision $\hat{x} \in \Omega_x$ of the leader, the $\hat{x}_L$-parameterized mixed-integer linear problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \tag{6.5a}$$
$$\text{s.t.} \quad Ax + By \geq a, \tag{6.5b}$$
$$Cx + Dy \geq b, \tag{6.5c}$$
$$x_i = \hat{x}_i \quad \text{for all } i \in L, \tag{6.5d}$$
$$d^\top y \leq \varphi(\hat{x}), \tag{6.5e}$$
$$x \in X, \ y \in Y, \tag{6.5f}$$

is called the *the $\hat{x}_L$-parameterized refinement problem*.

The refinement problem (6.5) is used to obtain a bilevel-feasible point (x, y), where the leader's linking variables are fixed to the respective values of a given $\hat{x}$ and y is an optimal response of the follower to $\hat{x}$. A solution (x, y) to Problem (6.5) is constructed so that it (i) satisfies all coupling constraints (if present) and (ii) complies with the considered optimistic solution concept; see Chapter 2. Constraints (6.5c) and (6.5e) as well as $y \in Y$ in (6.5f) impose that y has to be an optimal solution to the $\hat{x}$-parameterized lower-level problem (6.1c). Because the leader's linking variables are fixed by the constraints in (6.5d), the constant $\varphi(\hat{x})$ can be computed by solving the $\hat{x}$-parameterized lower level. Constraint (6.5b) ensures that the follower's response satisfies all coupling constraints. Note that, in Problem (6.5), only the non-linking variables of the leader can be adjusted to "fit" the follower's response so that all coupling constraints are satisfied. Among all optimal responses of the follower for which this is the case, the best one w.r.t. the leader's objective function value is chosen to align with the considered optimistic solution approach. Overall, the refinement problem (6.5) gets its name from the fact that it *refines* the follower's decision by selecting only those optimal responses that satisfy all coupling constraints and that are among the optimal ones for the leader.

Lemma 6.16 *Suppose that Assumption 6.5 holds and let $\hat{x} \in \Omega_x$ be given* *arbitrarily. Then, the $\hat{x}_L$-parameterized refinement problem* (6.5) *is either infeasible or solvable.*

Proof: Because $\hat{x} \in \Omega_x$ holds by assumption, there exists a point (x, y) with $x \in X$, $y \in Y$, $Ax + By \geq a$, $Cx + Dy \geq b$, and $x_i = \hat{x}_i$ for all $i \in L$. Hence, the mixed-integer linear problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & Cx + Dy \geq b, \\ & x_i = \hat{x}_i \quad \text{for all } i \in L, \\ & x \in X, \ y \in Y \end{aligned} \tag{6.6}$$

is feasible. Moreover, by Assumption 6.5, Problem (6.6) is bounded. As the refinement problem (6.5) is a restriction of Problem (6.6), it is also bounded. If we now add the constraint $d^\top y \leq \varphi(\hat{x})$ to Problem (6.6), we obtain the refinement problem and two cases can appear. The problem either stays feasible, implying solvability, or it becomes infeasible. This proves the claim. □

Using the last lemma, we can now state and prove existence results for the general bilevel MILP (6.1).

Theorem 6.17 *Suppose that Assumptions 6.5 and 6.13 hold. Then, Problem* (6.1) *is either infeasible or admits an optimal solution.*

Proof: Let us first point out that, w.l.o.g., all upper-level variables can be assumed to be discrete and bounded linking variables; see Assumption 6.13 and Exercise 6.14. Thus, the number of feasible choices for the leader's variables x is finite. This implies that the number of possible x-parameterized lower-level problems to consider is finite as well. By Lemma 6.16, two situations are possible for each choice of the leader's variables x:

(i) There exists an optimal response y of the follower such that (x, y) is feasible for the overall bilevel problem. Although the response of the follower may not be unique, we point out that the upper-level objective function value is uniquely determined by the construction of the refinement problem (6.5).
(ii) The leader's decision x leads to bilevel infeasibility.

If for all finitely many possible choices for x, Case (ii) applies, there is no x that leads to an optimal response y so that (x, y) satisfies all coupling constraints. Hence, the bilevel problem is infeasible. In the other case, Case (i) applies for at least one x. If we then take, among the finite set of (x, y)-pairs of Case (i), the pair with the smallest upper-level objective function value, we obtain the minimum value of the bilevel problem and the corresponding (x, y)-pair is an optimal solution to the bilevel MILP. □

Note that Theorem 6.17 generalizes the results of Vicente et al. (1996) by proving the existence of optimal solutions to general bilevel MILPs in which the lower level may be a mixed-integer linear problem, rather than a purely integer or a purely continuous linear problem. We close this section with the special case in which there are no coupling constraints.

Theorem 6.18 *Suppose that Assumptions 6.5 and 6.13 hold. Then, Problem* (6.1) *with* $B = 0$ *admits an optimal solution.*

Proof: Due to Assumption 6.13, we can again, w.l.o.g., assume that all upper-level linking variables are bounded integers. Hence, there are only finitely many x-parameterized lower-level problems to consider. Because the shared constraint set is non-empty by Assumption 6.5, there exists an $(\hat{x}, \hat{y}) \in \Omega$. This means that for all $\hat{x} \in \Omega_x$, the $\hat{x}$-parameterized lower-level problem is feasible. Moreover, again by Assumption 6.5, these $\hat{x}$-parameterized lower-level problems are also bounded and, thus, solvable. Let us denote an optimal lower-level solution with $\bar{y}$. Because there are no coupling constraints, all pairs $(\hat{x}, \bar{y})$ are bilevel feasible. Taking the best out of these finitely many points leads to a bilevel optimal solution. □

6.4 Complexity Results

Mixed-integer linear bilevel problems are exceptionally challenging to solve. In particular, they are even harder than their continuous counterpart, i.e., linear bilevel problems. Why is this the case? To get a better understanding of the complexity of bilevel MILPs, let us start with some special cases.

Observation 6.19 If X and Y in Problem (6.1) do not impose any integrality restrictions, we obtain a bilevel LP, which is strongly NP-hard in general; see Section 4.4 for further details.

Observation 6.20 If we set $n_y = 0$ in Problem (6.1), the bilevel problem reduces to a single-level mixed-integer linear problem, which is known the be strongly NP-hard in general; see, e.g., Problem MP1 in Garey and Johnson (1990).

From the last two observations, we can already conclude that mixed-integer linear bilevel problems are at least NP-hard in general. Moreover, in contrast to bilevel LPs, it is now even NP-hard to check whether a given point (x, y) is feasible for the bilevel MILP because this requires solving the x-parameterized mixed-integer linear lower-level problem.

Observation 6.21 For a given leader's decision $x \in \Omega_x$, solving the x-parameterized lower-level problem (6.1c) corresponds to solving a mixed-integer linear problem, which is strongly NP-hard in general.

The first complexity result for bilevel MILPs has been established by Jeroslow (1985). More specifically, the author shows that k-level optimization problems for which all variables are binary are Σ_k^p-hard. As a consequence, such discrete bilevel problems are Σ_2^p-hard in general. This means that they can be solved in nondeterministic polynomial time, provided that there exists an oracle that solves problems in NP in constant time. If you are interested in complexity-theoretical aspects, we further recommend the paper by Woeginger (2021) for a more rigorous definition and an in-depth discussion of the complexity class Σ_2^p. In addition, further complexity results in the context of bilevel MILPs can be found in Caprara et al. (2014) as well as in the recent paper by Grüne and Wulf (2025) and the references therein.

6.5 Connections Between Mixed-Integer Linear Bilevel Problems and Other Problem Classes

Although bilevel MILPs are very hard to solve, in some special cases, they can be equivalently reformulated as problems that may be easier to handle in practice. This is the focus of this section. As before, we understand "equivalence" in the sense that every globally optimal solution to the reformulated problem can be transformed in polynomial time into a globally optimal solution to the bilevel problem, and vice versa. We start with the following observation.

Observation 6.22 If the lower level of the mixed-integer linear bilevel problem (6.1) is a continuous linear problem, i.e., $Y = \mathbb{R}^{n_y}$, the same techniques as in Chapter 3 can be applied to obtain a single-level reformulation of the overall bilevel problem.

In Chapter 3, we have seen that a compact optimality certificate for the lower-level problem, which is both necessary and sufficient, is essential to obtain a single-level reformulation that can be tackled by general-purpose solvers. In the case of a bilevel MILP in which the lower level is an LP, the KKT conditions can be used for this purpose.

Proposition 6.23 *Suppose that $Y = \mathbb{R}^{n_y}$, i.e., only the upper-level variables are discrete, and that M is a sufficiently large constant. Then, Problem (6.1) is equivalent to the mixed-integer linear problem*

$$\begin{aligned}
\min_{x,y,\lambda,z} \quad & c_x^\top x + c_y^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\
& D^\top \lambda = d, \; \lambda \geq 0, \\
& \lambda_i \leq M z_i \quad \text{for all } i = 1, \dots, \ell, \\
& C_{i\cdot} x + D_{i\cdot} y - b_i \leq M(1 - z_i) \quad \text{for all } i = 1, \dots, \ell, \\
& x \in X, \; y \in Y, \; \lambda \in \mathbb{R}^\ell, \; z \in \{0,1\}^\ell .
\end{aligned}$$

Proof: The claim immediately follows from the derivations in Sections 3.2 and 3.4. □

Regarding the choice of the big-M value in the latter theorem, we refer back to Section 3.4, where this aspect is discussed in detail.

In Vicente et al. (1996), the authors establish connections between mixed-integer linear bilevel problems and continuous linear multilevel problems. In what follows, we prove these connections by imposing the additional assumption that all discrete variables are bounded and by using the links between binary variables and bilevel LPs; see Section 4.4 and the paper by Audet et al. (1997). However, before we do this, solving the next exercise is a good preparation.

Exercise 6.24 Consider an integer variable $z \in \mathbb{Z}$ with lower bound 0 and finite upper bound $u \in \mathbb{Z}$. Re-write this variable using a finite number of binary variables $x_0, x_1, \dots, x_k$. To this end, specifically determine k as well.

Proposition 6.25 *Suppose that all discrete variables are bounded and let $Y = \mathbb{R}^{n_y}$, i.e., only the upper-level variables are discrete. Then, Problem (6.1) is equivalent to a (continuous) linear bilevel problem.*

Proof: From Exercise 6.24 we know that any bounded integer can be represented using a finite number of binary variables. Hence, we assume, w.l.o.g., that all discrete upper-level variables are binary. Let $I \subseteq \{1, \dots, n_x\}$ denote the set of indices associated with the binary upper-level variables. Then, as it is done in Audet et al. (1997), we can model the binary constraints $x_i \in \{0, 1\}$, $i \in I$, by introducing additional lower-level variables $z \in \mathbb{R}^{|I|}$ as well as additional upper- and lower-level constraints; see Figure 4.3 for an illustration. Overall, we now consider the problem

$$
\begin{aligned}
\min_{x,y,z} \quad & c_x^\top x + c_y^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \ z = 0, \\
& x \in \mathbb{R}^{n_x}, \ (y, z) \in \mathcal{S}(x),
\end{aligned}
\tag{6.7}
$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized problem

$$
\begin{aligned}
\min_{y,z} \quad & d^\top y - \sum_{i \in I} z_i \\
\text{s.t.} \quad & Cx + Dy \geq b, \\
& z_i \leq x_i, \ z_i \leq 1 - x_i \quad \text{for all } i \in I, \\
& y \in \mathbb{R}^{n_y}, \ z \in \mathbb{R}^{|I|}.
\end{aligned}
$$

Problem (6.7) is a linear bilevel problem in which the lower-level variables y and z are only coupled via the follower's objective function. The follower's optimality together with the coupling constraints $z = 0$ enforce that a bilevel-feasible point satisfies $0 = z_i = \min\{x_i, \ 1 - x_i\}$ for all $i \in I$, which is equivalent to $x \in \{0, 1\}^{|I|}$. □

Proposition 6.25 states that we can reduce a bilevel MILP with a mixed-integer upper-level and a continuous lower-level problem to a bilevel LP provided that all integer variables are bounded. However, such a reduction is not possible anymore if the lower level also includes discrete variables. Nevertheless, we can again exploit the ideas of Audet et al. (1997) to re-state a purely discrete bilevel problem as a (continuous) linear trilevel problem. The overall hardness of the considered bilevel problems is thus reflected by introducing a third level.

Proposition 6.26 *Suppose that all discrete variables are bounded and let $X = \mathbb{Z}^{n_x}$ and $Y = \mathbb{Z}^{n_y}$, i.e., all upper- and lower-level variables are bounded integers. Then, Problem (6.1) is equivalent to a (continuous) linear trilevel problem.*

Proof: Because any bounded integer can be represented via a binary expansion (see Exercise 6.24), we can again assume, w.l.o.g., that all upper- and lower-level variables are binary. Hence, exploiting the same ideas as in the proof of Proposition 6.25 and Audet et al. (1997), Problem (6.1) can equivalently be re-stated as the linear trilevel problem

$$
\begin{aligned}
\min_{x,y,z,v} \quad & c_x^\top x + c_y^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \ z = 0, \\
& x \in \mathbb{R}^{n_x}, \ (y, z, v) \in \mathcal{S}(x),
\end{aligned}
$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized problem

$$\begin{aligned}\min_{y,z,v} \quad & d^\top y - \sum_{i \in I} z_i \\ \text{s.t.} \quad & Cx + Dy \geq b, \\ & z_i \leq x_i,\ z_i \leq 1 - x_i \quad \text{for all } i = 1, \dots, n_x, \\ & v = 0, \\ & y \in \mathbb{R}^{n_y},\ z \in \mathbb{R}^{n_x},\ v \in \Psi(y),\end{aligned}$$

and

$$\Psi(y) := \arg\max_{v} \left\{ \sum_{i=1}^{n_y} v_i : v_i \leq y_i,\ v_i \leq 1 - y_i,\ v_i \in \mathbb{R},\ i = 1, \dots, n_y \right\}.$$

This concludes the proof. □

Exercise 6.27 Consider the integer linear bilevel problem in Example 6.2.

(i) Use the illustration in Figure 6.2 to derive bounds for the variables x and y of the bilevel problem.
(ii) Re-write the bilevel problem as a binary linear bilevel problem using the variable bounds obtained in (i) and a binary expansion.
(iii) Exploit Proposition 6.26 to reformulate the binary linear bilevel problem obtained in (ii) as a continuous linear trilevel problem.

6.6 What You Should Know Now!

1. Can you state the general form of a mixed-integer linear bilevel problem?
2. What are linking variables?
3. Can you state the single-level relaxation of a bilevel MILP? Why is it a relaxation?
4. Can you state the continuous relaxation of the single-level relaxation of a bilevel MILP?
5. What do you know about the attainability of optimal solutions for mixed-integer linear bilevel problems?
6. What is the refinement problem?
7. Can you explain why it is typically assumed that all linking variables are bounded integers?
8. What is required to ensure the existence of optimal solutions to bilevel MILPs?
9. What do you know about the hardness of bilevel MILPs in general?

10. What do you know about the hardness of bilevel MILPs if the lower level is a continuous linear problem?
11. How can we solve a bilevel MILP if the lower level is a continuous linear problem? Can the same approach still be applied if there are discrete lower-level variables?
12. What is the key idea to establish a connection between mixed-integer linear bilevel problems and (continuous) linear multilevel problems? Can you illustrate it using an example?
13. What assumptions are needed to establish the connection between mixed-integer linear bilevel problems and (continuous) linear multilevel problems. Why are these assumptions needed?

7

Excursus: Branch-and-Bound for Single-Level MILPs

In this chapter, we discuss the branch-and-bound method for mixed-integer linear single-level problems; see Land and Doig (1960) for the original paper and the book by Wolsey (2020) for a nice introduction to branch-and-bound. The basics of this branch-and-bound method will be used in the following chapters when we discuss branch-and-bound as well as branch-and-cut methods for mixed-integer linear bilevel problems.

We consider problems of the form

$$\begin{aligned} \min_{x} \quad & c^\top x \\ \text{s.t.} \quad & Ax \geq a, \\ & x_i \in \mathbb{Z} \quad \text{for all } i \in I, \end{aligned} \tag{7.1}$$

with $c \in \mathbb{R}^{n_x}$, $A \in \mathbb{R}^{m \times n_x}$, $a \in \mathbb{R}^m$, and $I \subseteq \{1, \ldots, n_x\}$. Throughout the remainder of this chapter, we make the following assumption for the ease of presentation.

Assumption 7.1 The feasible set $\{x \in \mathbb{R}^{n_x} : Ax \geq a\}$ of the continuous relaxation of Problem (7.1) is bounded.

Note that the feasible set of Problem (7.1) is described by a finite number of affine, and thus continuous, functions. Hence, the feasible set is closed and Assumption 7.1 implies that it is compact. The idea of the branch-and-bound method for MILPs is now very similar to the one of the branch-and-bound method for linear bilevel problems; see Algorithm 2.

(i) We start by solving a relaxation of Problem (7.1) that is obtained by omitting all integrality constraints, i.e., we consider the linear problem

$$\min_{x} \quad c^\top x \quad \text{s.t.} \quad Ax \geq a. \tag{7.2}$$

If $\{x \in \mathbb{R}^n : Ax \geq a\} \neq \emptyset$ holds, the Weierstraß theorem ensures that this problem has an optimal solution. Otherwise, the problem is infeasible.

(ii) Typically, a solution $\hat{x}$ to Problem (7.2), if it exists, violates some of the integrality constraints. This means that there is an index $i \in I$ with $\hat{x}_i \notin \mathbb{Z}$. We take such an index i and construct two new sub-problems by imposing a variable disjunction. In the first sub-problem, we add the constraint

$$x_i \geq \lfloor \hat{x}_i \rfloor + 1$$

and in the second one, we add the constraint

$$x_i \leq \lfloor \hat{x}_i \rfloor.$$

Here and in what follows, $\lfloor z \rfloor$ denotes the floor function, which returns the largest integer that is less than or equal to $z \in \mathbb{R}$.

(iii) Next, we choose one of the unsolved sub-problems and proceed in the same way.

Overall, every node in the branch-and-bound search tree is defined by the root-node problem (7.2) as well as additional bounds imposed on (some of) the integer variables. Compared to the branch-and-bound method for bilevel LPs (Algorithm 2), this means that we now branch on integer variables rather than on complementarity constraints. The branch-and-bound method for mixed-integer linear single-level problems is formally stated in Algorithm 5.

In Algorithm 5, we start by initializing the set of open nodes Q and the incumbent value $\mathcal{U}$; see Line 1. As long as there are open nodes, we choose an arbitrary one (Line 3) and solve the optimization problem associated with this node. If this problem is infeasible, we fathom the node, i.e., we do not consider it further and return to the beginning of the while-loop (Line 6). Otherwise, we check if the optimal solution at the current node yields a better value than the current incumbent value. If this is not the case, we fathom the node (Line 9). Otherwise, we proceed and check if the node solution satisfies all integrality constraints. If the current solution is integer feasible, we have found a new incumbent and update the respective data in Line 11. If not, we branch on an integer variable with a fractional value (Line 13) and continue with the next unprocessed node from Q in Line 3. If the incumbent is not updated during the entire execution of the method, no integer feasible point exists, which implies that the problem is infeasible; see Line 17. Otherwise, an optimal solution is returned in Line 15.

To sum up, we use the same pruning rules as in Section 5.2 but now branch on integer variables with fractional values instead of violated complementarity constraints. Moreover, lower bounds and, thus, an optimality gap can be obtained

Algorithm 5 Branch-and-Bound for MILPs

Input: An instance of Problem (7.1) that satisfies Assumption 7.1

Output: An optimal solution x^* to Problem (7.1) or the indication that Problem (7.1) is infeasible.

1: Set $Q \leftarrow \{\mathbb{R}^{n_x}\}$ and $\mathcal{U} \leftarrow +\infty$.
2: **while** $Q \neq \emptyset$ **do**
3:    Choose $X \in Q$ and set $Q \leftarrow Q \setminus \{X\}$.
4:    Solve the linear problem

$$\min_{x \in X} \quad c^\top x \quad \text{s.t.} \quad Ax \geq a. \tag{7.3}$$

5:    **if** Problem (7.3) is infeasible **then**
6:       Go to Line 2.
7:    Let $\hat{x}$ denote the optimal solution to Problem (7.3).
8:    **if** $c^\top \hat{x} \geq \mathcal{U}$ **then**
9:       Go to Line 2.
10:    **if** $\hat{x}_i \in \mathbb{Z}$ for all $i \in I$ **then**
11:       Set $x^* \leftarrow \hat{x}$, $\mathcal{U} \leftarrow c^\top x^*$, and go to Line 2.
12:    Choose an index $i \in I$ with $\hat{x}_i \notin \mathbb{Z}$.
13:    Set

$$\underline{X} \leftarrow X \cap \{x : x_i \geq \lfloor \hat{x}_i \rfloor + 1\}, \quad \bar{X} \leftarrow X \cap \{x : x_i \leq \lfloor \hat{x}_i \rfloor\},$$

   and $Q \leftarrow Q \cup \{\underline{X}, \bar{X}\}$.
14: **if** $\mathcal{U} < +\infty$ **then**
15:    **return** x^*
16: **else**
17:    **return** "The given problem is infeasible."

from the node solutions as well as it is discussed for complementarity-based branch-and-bound; see Remark 5.5.

Remark 7.2 The steps in Lines 6, 9, and 11 of the branch-and-bound method in Algorithm 5, in which the method returns to the beginning of the while-loop, are often called *backtracking* in the literature.

Algorithm 5 allows for some flexibility in choosing which node to process next. As before, possible options include a depth-first search, a breadth-first search, variants of the two, or even completely different strategies; see, e.g., Section 3.1 in Belotti et al. (2013). In addition, the choice of the index i for a variable $\hat{x}_i$ that violates the integrality constraints may not be unique. To select a specific index i, one can apply different branching rules; see, e.g., Achterberg et al. (2005) and the references therein.

Theorem 7.3 *Under Assumption 7.1, Algorithm 5 terminates after a finite number of visited nodes, either with an optimal solution to the mixed-integer linear problem* (7.1) *or with the correct indication that the problem is infeasible.*

The case in which the root-node relaxation is unbounded, i.e., the case in which Assumption 7.1 is violated, is discussed in Section 24.1 in Schrijver (1998).

Exercise 7.4 Prove Theorem 7.3. (*Hint:* Recall the Weierstraß theorem and look into the proofs of the branching and bounding lemmas as well as the proof of the correctness theorem in Section 5.2 again.)

7.1 What You Should Know Now!

1. How does the branch-and-bound method for mixed-integer linear single-level problems work?
2. What is the basic assumption that we make in this chapter and why do we make it?
3. What fathoming rules do we use in the branch-and-bound method for MILPs (Algorithm 5)?
4. What is backtracking?
5. Can you prove the correctness of the branch-and-bound method for mixed-integer linear single-level problems in Algorithm 5?

8
Branch-and-Bound for Bilevel MILPs

Building on the classic branch-and-bound method for mixed-integer linear single-level problems discussed in Chapter 7, the goal of this chapter is to present and discuss a branch-and-bound method for mixed-integer linear bilevel problems of the form

$$\begin{aligned} \min_{x \in X, y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \arg\min_{\bar{y} \in Y} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\} \end{aligned} \tag{8.1}$$

with a, b, c_x, c_y, d, A, B, C, and D defined as in Chapter 6. As before, the sets X and Y impose integrality constraints on (a subset of) the x- and y-variables, respectively. The first branch-and-bound method for solving bilevel MILPs has been published in the paper by Moore and Bard (1990). In their seminal work, the authors consider bilevel problems of the form given in (8.1) but with $B = 0$, i.e., there are no coupling constraints. However, it turns out that the method by Moore and Bard is not very powerful. Effective methods for more general bilevel MILPs have only emerged much later; see, e.g., Fischetti et al. (2018a).

In this chapter, we first discuss how the single-level reformulation using the optimal-value function can be exploited to derive relaxations and restrictions of bilevel problems of the form given in (8.1) in Section 8.1. We then re-visit the ideas of Moore and Bard (1990) to build a better intuition about what works and what does not in mixed-integer linear bilevel optimization. In particular, we illustrate what goes wrong if one tries to build a branch-and-bound scheme on a continuous counterpart of the mixed-integer linear bilevel problem in which one simply relaxes all integrality constraints, which is the key idea of the branch-and-bound method for mixed-integer linear single-level optimization discussed in Chapter 7. The issue is that, in fact, the continuous counterpart is not a relaxation of the mixed-integer linear bilevel problem. We discuss this in detail in Sections 8.1 and 8.2. We then show how to set up a branch-and-bound method (based on a proper relaxation of the bilevel problem) that has been proposed by Fischetti et al. (2018a) for general bilevel MILPs in Section 8.3. Finally, we apply this method to the illustrative examples from Chapter 6 in Section 8.4.

8.1 About Relaxations and Restrictions

Recall that, using the optimal-value function, Problem (8.1) can be reformulated as

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \tag{8.2a}$$
$$\text{s.t.} \quad (x, y) \in \bar{\Omega}, \tag{8.2b}$$
$$d^\top y \le \varphi(x), \tag{8.2c}$$
$$x \in X, \; y \in Y, \tag{8.2d}$$

where

$$\bar{\Omega} := \{(x, y) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} : Ax + By \ge a, \; Cx + Dy \ge b\}$$

is the continuous relaxation of the shared constraint set and

$$\varphi(x) := \min_{y \in Y} \left\{ d^\top y : Cx + Dy \ge b \right\}$$

is the optimal-value function of the lower-level problem.

Let us now consider two sets $\underline{Y} \subset Y$ and $\bar{Y} \supset Y$, i.e., $\underline{Y}$ is a restriction and $\bar{Y}$ is a relaxation of the set Y. The optimal-value functions associated with the sets $\underline{Y}$ and $\bar{Y}$ are now defined as

$$\bar{\varphi}(x) := \min_{y \in \underline{Y}} \left\{ d^\top y : Cx + Dy \ge b \right\}$$

and

$$\underline{\varphi}(x) := \min_{y \in \bar{Y}} \left\{ d^\top y : Cx + Dy \ge b \right\},$$

respectively. Hence, for all $x \in X$, we have

$$\bar{\varphi}(x) \ge \varphi(x) \ge \underline{\varphi}(x).$$

In this section, we discuss what can happen if we replace the set Y with its relaxation or its restriction in the optimal-value function constraint (8.2c), in the domain constraint (8.2d), or in both of them. The following two propositions specify under which conditions we can obtain a relaxation, respectively, a restriction of Problem (8.1).

Proposition 8.1 *The problem*

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega}, \\ & d^\top y \leq \bar{\varphi}(x), \\ & x \in X, \ y \in Y, \end{aligned} \tag{8.3}$$

and the problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega}, \\ & d^\top y \leq \bar{\varphi}(x), \\ & x \in X, \ y \in \bar{Y}, \end{aligned} \tag{8.4}$$

are both relaxations of Problem (8.1) *according to Definition 5.6.*

Proof: Let us consider the optimal-value function reformulation (8.2) of Problem (8.1). If we restrict the follower's variables, we obtain a restriction of the lower-level problem. For a given leader's decision x, we have $\varphi(x) \leq \bar{\varphi}(x)$. Hence, replacing $\varphi(x)$ with $\bar{\varphi}(x)$ in Constraint (8.2c) leads to a relaxation. If we keep the set Y untouched as in Problem (8.3) or if we allow for variables $y \in \bar{Y}$ as in Problem (8.4), we obtain a relaxation of Problem (8.1). □

For the next proposition, we formally define a restriction first.

Definition 8.2 (Restriction) Consider the given optimization problem $\min\{f(x) : x \in \mathcal{F}\}$. The optimization problem $\min\{g(x) : x \in \mathcal{F}'\}$ is called a *restriction* of the other problem if $\mathcal{F} \supseteq \mathcal{F}'$ and if $g(x) \geq f(x)$ holds for all $x \in \mathcal{F}$.

Proposition 8.3 *The problem*

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega}, \\ & d^\top y \leq \underline{\varphi}(x), \\ & x \in X, \ y \in Y, \end{aligned}$$

and the problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega}, \\ & d^\top y \leq \underline{\varphi}(x), \\ & x \in X, \ y \in \underline{Y}, \end{aligned}$$

are both restrictions of Problem (8.1) *according to Definition* 8.2.

Proof: The claim immediately follows from the definitions of the set $\underline{Y}$ and the function $\underline{\varphi}(x)$. □

Next, we also consider the "mixed" cases.

Proposition 8.4 *The problem*

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega}, \\ & d^\top y \leq \underline{\varphi}(x), \\ & x \in X, \ y \in \bar{Y}, \end{aligned} \tag{8.5}$$

and the problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & (x, y) \in \bar{\Omega}, \\ & d^\top y \leq \bar{\varphi}(x), \\ & x \in X, \ y \in \underline{Y}, \end{aligned} \tag{8.6}$$

are neither restrictions nor relaxations of Problem (8.1) *in general.*

Proof: We prove the first statement by using three examples. Example 8.5 illustrates a situation in which the optimal value of Problem (8.5) is better than the optimal value of Problem (8.1). Example 8.6 illustrates a case in which the optimal solution to Problem (8.5) is the same as the optimal solution to Problem (8.1). Finally, Example 8.7 shows an instance for which the optimal value of Problem (8.5) is worse than the optimal value of Problem (8.1).

The proof concerning Problem (8.6) is left as an exercise for the reader; see Exercise 8.8. □

Example 8.5 We consider the integer linear bilevel problem

$$\max_{x \in \mathbb{Z}, y} \quad x \quad \text{s.t.} \quad x \geq 0, \ y \in \mathcal{S}(x),$$

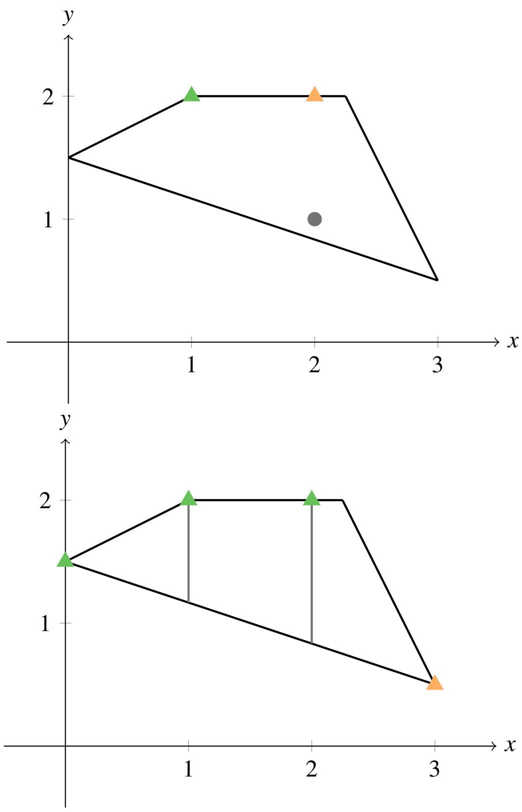


Figure 8.1 The bilevel problem in Example 8.5. The original problem with a discrete lower level is shown in the top figure. Discrete points (gray dot as well as green and orange triangles) are feasible for the shared constraint set, whereas green and orange triangles represent bilevel-feasible points. The orange triangle $(2, 2)$ is the optimal solution to the original bilevel problem. The figure at the bottom shows the bilevel problem in which the lower-level integrality constraints are omitted. Gray vertical lines parallel to the y-axis as well as green and orange triangles correspond to the shared constraint set, whereas the green and orange triangles represent bilevel-feasible points. The orange triangle $(3, 0.5)$ is the optimal solution to the bilevel problem without integrality constraints at the lower level.

where $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized integer linear problem

$$\max_{y \in \mathbb{Z}} \quad y \quad \text{s.t.} \quad y \leq 2, \; -x + 2y \leq 3, \; -2x - 6y \leq -9, \; 4x + 2y \leq 13.$$

An illustration of the problem is given in Figure 8.1 (top). The shared constraint set contains three integer points: $(1, 2)$, $(2, 1)$, and $(2, 2)$. If the leader chooses $x = 1$, the follower responds with $y = 2$. This leads to an upper-level objective function value of 1. If the leader decides for $x = 2$, the follower again optimally responds with $y = 2$. In this case, the objective function value of the leader is 2. Hence, the feasible set of this bilevel problem is given by $\{(1, 2), (2, 2)\}$ and $(x^*, y^*) = (2, 2)$ is the optimal solution to the bilevel problem.

Let us now consider the problem in which we relax the integrality on the follower's variable y, i.e., we consider the linear lower-level problem

$$\max_{y \in \mathbb{R}} \quad y \quad \text{s.t.} \quad y \le 2, \; -x + 2y \le 3, \; -2x - 6y \le -9, \; 4x + 2y \le 13.$$

The bilevel problem in which the lower-level integrality constraint is relaxed is an instance of Problem (8.5) and it is illustrated in Figure 8.1 (bottom). The feasible set of this bilevel problem is given by $\{(0, 1.5), (1, 2), (2, 2), (3, 0.5)\}$, i.e., we have enlarged the bilevel-feasible set by relaxing the lower-level integrality constraint. Because the upper-level objective function does not change, we indeed obtain a relaxation according to Definition 5.6. The optimal solution to this problem is given by $(x, y) = (3, 0.5)$, which is not feasible for the original bilevel problem of the form given in (8.1) with a discrete lower level. The optimal objective function value of the bilevel problem with $y \in \mathbb{R}$ is 3, which is larger than the optimal objective function value of the original bilevel problem with $y \in \mathbb{Z}$. To sum up, omitting the integrality constraint on y indeed leads to a relaxation of the overall bilevel problem in this example. △

Example 8.6 We consider the integer linear bilevel problem

$$\min_{x \in \mathbb{Z}, y} \quad x + 2y \quad \text{s.t.} \quad 0 \le x \le 1, \; y \in \mathcal{S}(x),$$

where $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized integer linear problem

$$\max_{y \in \mathbb{Z}} \quad y \quad \text{s.t.} \quad 0 \le y \le 1 - x.$$

An illustration of the problem is given in Figure 8.2. If the leader chooses $x = 1$, the only feasible response of the follower is $y = 0$, which leads to an upper-level objective function value of 1. If the leader decides for $x = 0$, the follower optimally responds with $y = 1$. This leads to an objective function value of 2 for the leader. Hence, $(x^*, y^*) = (1, 0)$ is the optimal solution to the bilevel problem.

If we now relax the integrality constraint on the follower's variable y and allow for $y \in \mathbb{R}$, the set of bilevel-feasible points does not change. The follower still chooses $y = 0$ if $x = 1$ and $y = 1$ if $x = 0$. Hence, in this example, omitting

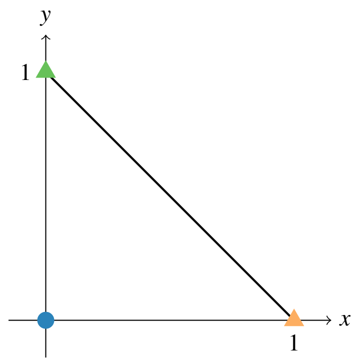


Figure 8.2 The bilevel problem in Example 8.6. Discrete points (blue dot as well as green and orange triangles) are feasible for the shared constraint set, whereas triangles represent bilevel-feasible points. The orange triangle $(1, 0)$ is the optimal solution to the problem.

the integrality constraint on y does not affect the solution to the overall bilevel problem at all. △

Example 8.7 (Example 6.2—Revisited) We have seen that the solution to the bilevel problem in Example 6.2 is given by the point $(x^*, y^*) = (2, 2)$, which leads to an optimal objective function value of -22. The optimal solution to the bilevel problem in which we relax the integrality on the follower's variable y is the point $(x, y) = (8, 1)$. This point is both integer and bilevel feasible. The corresponding objective function value, however, is -18, which is worse than the optimal objective function value. △

Exercise 8.8 Prove the second half of Proposition 8.4.

8.2 Why Relaxing Integrality Constraints is not Enough

In the last chapter, we have seen that the key idea of the branch-and-bound method for mixed-integer linear single-level problems is to consider a root-node problem in which all integrality conditions are relaxed. In this section, we will understand what is happening if we do the same for mixed-integer linear bilevel problems. To see this, let us formally introduce the continuous counterpart of the mixed-integer linear bilevel problem.

Definition 8.9 (Continuous Counterpart) The linear bilevel problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \ x \in \mathbb{R}^{n_x}, \\ & y \in \arg\min_{\bar{y}} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b, \ \bar{y} \in \mathbb{R}^{n_y} \right\}, \end{aligned}$$

is called the *continuous counterpart* of the mixed-integer linear bilevel problem (8.1). Hence, the continuous counterpart of Problem (8.1) corresponds to the bilevel problem in which all integrality constraints are omitted.

Observation 8.10 The continuous counterpart of a mixed-integer linear bilevel problem of the form given in (8.1) can be reformulated as an instance of Problem (8.5) in which the constraint $x \in X$ is replaced with $x \in \mathbb{R}^{n_x}$.

Proposition 8.4 together with Observation 8.10 leads to the following observation.

Observation 8.11 The solution to the continuous counterpart of a mixed-integer linear bilevel problem does, in general, not provide a valid lower bound for the optimal objective function value of the original problem. Hence, the continuous counterpart of a mixed-integer linear bilevel problem is, in general, not a relaxation of the original problem.

Because of Observation 8.11, the second fathoming rule (bounding) does not carry over to the mixed-integer linear bilevel setting if one only relaxes the integrality constraints. To better understand why this happens and to build some more intuition, let us now take a closer look by distinguishing between omitting the integrality constraints for the leader and the follower. If we only omit the integrality constraints for the leader's variables, we enlarge the feasible set of the leader, whereas any point that is feasible for the original bilevel problem remains feasible. Hence, omitting the integrality constraints only for the leader's variables indeed leads to a relaxation of the overall bilevel MILP. In contrast, as we have seen in Section 8.1, omitting the integrality constraints for the follower's variables leads to a relaxation of the lower-level problem. This means that, for a given leader's decision x, we have $\varphi(x) \geq \underline{\varphi}(x)$, where $\varphi(x)$ and $\underline{\varphi}(x)$ denote the optimal objective function values of the original and the relaxed lower-level problem, respectively. As usual, we can use the inequality $d^\top y \leq \varphi(x)$ to ensure that the follower responds optimally to a given leader's decision x. However, replacing $\varphi(x)$ with $\underline{\varphi}(x)$ in this constraint leads to a restriction of the inequality. Hence, omitting the integrality constraints for the follower's variables neither leads to a relaxation nor a restriction of the original bilevel MILP in general; see

Proposition 8.4. In particular, we have seen in Examples 8.5–8.7 that omitting the lower-level integrality constraints can lead to a relaxation or a restriction of the overall bilevel problem, or it may not affect the problem at all.

Although the second fathoming rule (bounding) does not carry over to the mixed-integer linear bilevel setting if one only relaxes the integrality constraints, we can still use it to gain insight into the limitations of the classic fathoming rules. To this end, let us consider applying the classic branch-and-bound method to a bilevel MILP with the fathoming rules left unchanged and with the continuous counterpart as the root-node problem.

Example 8.12 (Example 6.1—Revisited) We revisit Example 6.1, i.e., we consider the integer linear bilevel problem

$$\begin{aligned} \max_{x \in \mathbb{Z}, y} \quad & F(x, y) = -x - 2y \\ \text{s.t.} \quad & x \geq 0, \ y \in \mathcal{S}(x), \end{aligned} \tag{8.7}$$

where $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized integer linear problem

$$\begin{aligned} \max_{y \in \mathbb{Z}} \quad & f(x, y) = y \\ \text{s.t.} \quad & -x + 2.5y \leq 3.75, \\ & x + 2.5y \geq 3.75, \\ & 2.5x + y \leq 8.75, \\ & y \geq 0; \end{aligned}$$

see Figure 6.1 for an illustration. We now illustrate a depth-first search branch-and-bound method for the setting in which we omit all integrality restrictions and branch on fractional values of integer variables as usual. This means that, at some node k of the branch-and-bound search tree, we solve the continuous linear bilevel problem

$$\begin{aligned} \max_{x \in \mathbb{R}, y} \quad & -x - 2y \\ \text{s.t.} \quad & \underline{x}^k \leq x \leq \bar{x}^k, \ y \in \mathcal{S}_k(x), \end{aligned} \tag{8.8}$$

where $\mathcal{S}_k(x)$ denotes the set of optimal solutions to the x-parameterized continuous linear lower-level problem

$$\begin{aligned}
\min_{y \in \mathbb{R}} \quad & -y \\
\text{s.t.} \quad & -x + 2.5y \leq 3.75, \\
& x + 2.5y \geq 3.75, \\
& 2.5x + y \leq 8.75, \\
& \underline{y}^k \leq y \leq \bar{y}^k
\end{aligned}$$

with $0 \leq \underline{x}^k \leq \bar{x}^k$ and $0 \leq \underline{y}^k \leq \bar{y}^k$. Problem (8.8) corresponds to the continuous counterpart of Problem (8.7) in which additional bounds are imposed on the variables x and y because of branching. At the root node (node 0), we consider the problem with $\underline{x}^0 = 0 = \underline{y}^0$ as well as $\bar{x}^0 = \infty = \bar{y}^0$. The branch-and-bound search tree is depicted in Figure 8.3.

The root-node problem has the solution $(x, y) = (0, 1.5)$ with an objective function value of $F = -3$ for the leader. Because $y = 1.5$ is fractional, we now branch on the follower's variable. This means that we generate two new sub-problems in which we add either the constraint $y \geq 2$ or $y \leq 1$. At node 1 of the branch-and-bound search tree, we consider Problem (8.8) with $\underline{y}^1 = 2$ and $\bar{y}^1 = \infty$. The optimal solution to this problem is $(x, y) = (1.25, 2)$ with $F = -5.25$. Further branching, fathoming due to infeasibility, and backtracking then leads us to node 9. The solution to the problem considered at node 9 is given by $(x, y) = (2, 1)$ with $F = -4$. This point is integer but not bilevel feasible because the follower's optimal reaction to $x = 2$ is $y = 2 \neq 1$. Thus, the objective function value at node 9, i.e., $F = -4$, is not a valid lower bound for the original bilevel problem. If fathoming rule 3 were applied here, node 9 would be fathomed because the point $(x, y) = (2, 1)$ satisfies all integrality constraints. As a consequence, however, we would not find the optimal solution $(x, y) = (3, 1)$, which would require to restrict the variable x even further; see the dashed line in Figure 8.3. Similarly, one can also show that selecting y as the branching variable in node 7 does not help for finding the optimal solution. △

Based on Example 8.12, we make one more observation.

Observation 8.13 An integer-feasible solution found at a node cannot, in general, be used to fathom this node if the continuous counterpart is used as the root-node problem.

To sum up, only fathoming rule 1 (infeasibility) can be used in its original form within a branch-and-bound method for mixed-integer linear bilevel problems if it is based on the continuous counterpart of the problem.

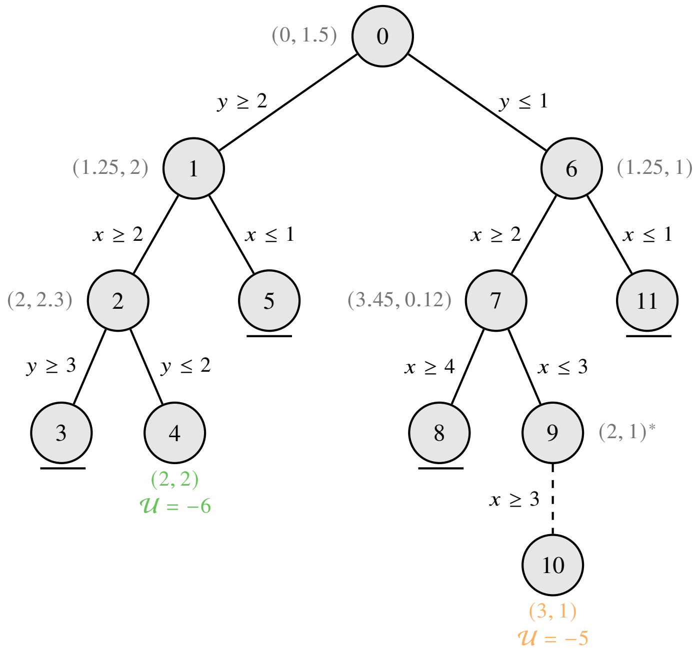


Figure 8.3 The branch-and-bound search tree in Example 8.12. Tuples of the form (x, y) at the branch-and-bound nodes denote the solutions to the problems considered at these nodes. The solution marked with an asterisk at node 9 is integer but not bilevel feasible.

8.3 A Branch-and-Bound Method for Mixed-Integer Linear Bilevel Problems

We now discuss the branch-and-bound method that has been proposed by Fischetti et al. (2018a) for general bilevel MILPs of the form given in (8.1). Recall from the previous chapter and the last section that, in a branch-and-bound method, the node problems in the search tree need to be set up based on some relaxation of the original problem and additionally included bounds due to branching decisions. In the context of bilevel MILPs, we now explain which relaxation is considered at the root node so that, in particular, fathoming rule 2 is valid again. Recall that the continuous counterpart of a bilevel MILP is not a relaxation of the original problem. Thus, it does not provide a valid lower bound; see Observation 8.11. Nevertheless, a valid lower bound can be obtained from the continuous relaxation of the single-level relaxation of the bilevel MILP;

see Observation 6.4. Instead of solving the continuous counterpart, we thus start the branch-and-bound method by solving the continuous relaxation of the single-level relaxation. To ensure that an optimal solution to this problem exists, i.e., Lemma 6.6 holds, we impose Assumption 6.5 for the remainder of this chapter. However, let us point out that a more general approach in which we do not require the shared constraint set to be bounded can also be obtained by exploiting the considerations of Section 6.2; see also Section 3.1 in Fischetti et al. (2018a).

Let us now make this a bit more formal. At the root node of the branch-and-bound search tree, we consider the linear problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \bar{\Omega},$$

where $\bar{\Omega}$ denotes the continuous relaxation of the shared constraint set Ω. A solution to this problem is, most likely, not feasible for the original bilevel problem. To deal with fractional values of integer variables, we can still branch as usual. However, as we have seen in Example 8.12, integer feasibility alone is not sufficient to fathom a node. Hence, we have to include a bilevel-specific incumbent update to handle such situations correctly and to overcome the issues with fathoming rule 3. To make this more specific, let us take a closer look at how to process node k in the branch-and-bound search tree. At this node, we solve the linear problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \bar{\Omega}_k \subseteq \mathbb{R}^{n_x} \times \mathbb{R}^{n_y}. \tag{8.9}$$

Here, $\bar{\Omega}_k$ is obtained from $\bar{\Omega}$ by incorporating all bounds due to branching decisions that have been made at nodes along the path from the root node to node k. Under Assumption 6.5, Problem (8.9) is either infeasible or solvable. We now formally state the procedure for processing node k in Algorithm 6.

The first steps to process node k are similar to how nodes are processed in the branch-and-bound method for mixed-integer linear single-level problems; see Algorithm 5. To this end, Algorithm 6 starts with solving Problem (8.9) in Line 1. If this problem is infeasible, we do not consider it further and fathom the current node; see Line 3. Otherwise, Problem (8.9) has an optimal solution (x^k, y^k). If its objective function value is not better than the current incumbent value $\mathcal{U}$, we can fathom the node because no improvements can be achieved by further branching; see Line 6. If the point (x^k, y^k) yields a better value than the current incumbent solution, we proceed by checking for integer and bilevel feasibility. Integer feasibility is easy to verify so we do this first. If (x^k, y^k) violates (some of) the integrality constraints, we branch; see Line 8. To this end, we choose an integer variable, say z_k, whose current value is fractional. As before, we then

Algorithm 6 Processing Node k of the Branch-and-Bound Search Tree

Input: An instance of Problem (8.9) and an upper bound $\mathcal{U} \in \mathbb{R} \cup \{\infty\}$ for the optimal objective function value of Problem (8.1)

Solve Problem (8.9).
if Problem (8.9) is infeasible **then**
  Fathom the current node, i.e., **stop** this node processing procedure and go back to the main method.
Let (x^k, y^k) denote the optimal solution to Problem (8.9).
if $c_x^\top x^k + c_y^\top y^k \geq \mathcal{U}$ **then**
  Fathom the current node, i.e., **stop** this node processing procedure and go back to the main method.
if $(x^k, y^k) \notin X \times Y$ **then**
  Branch on an integer variable with a fractional value to create two new sub-problems, **stop** this node processing procedure, and go back to the main method.
Solve the x^k-parameterized lower-level problem, i.e., compute

$$\varphi(x^k) = \min_{y \in Y} \left\{ d^\top y \colon Dy \geq b - Cx^k \right\}.$$

if $d^\top y^k \leq \varphi(x^k)$ **then**
  Update the incumbent, i.e., update the incumbent solution with (x^k, y^k) and set the incumbent value to $\mathcal{U} \leftarrow c_x^\top x^k + c_y^\top y^k$.
  Fathom the current node, i.e., **stop** this node processing procedure and go back to the main method.
if not all linking variables x_i^k, $i \in L$, are fixed by branching **then**
  Branch on any x_i^k, $i \in L$, that is not fixed by branching yet to create two new sub-problems, **stop** this node processing procedure, and go back to the main method.
Solve the x_L^k-parameterized mixed-integer linear refinement problem

$$\begin{aligned} \min_{x \in X,\, y \in Y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\ & x_i = x_i^k \quad \text{for all } i \in L, \\ & d^\top y \leq \varphi(x^k). \end{aligned} \tag{8.10}$$

if Problem (8.10) is infeasible **then**
  Fathom the current node, i.e., **stop** this node processing procedure and go back to the main method.
Let $(\tilde{x}, \tilde{y})$ denote an optimal solution to Problem (8.10).
if $c_x^\top \tilde{x} + c_y^\top \tilde{y} < \mathcal{U}$ **then**
  Update the incumbent, i.e., update the incumbent solution with $(\tilde{x}, \tilde{y})$ and set the incumbent value to $\mathcal{U} \leftarrow c_x^\top \tilde{x} + c_y^\top \tilde{y}$.
Fathom the current node, i.e., **stop** this node processing procedure and go back to the main method.

generate two new sub-problems: one in which we add the constraint

$$z \geq \lfloor z_k \rfloor + 1$$

and one in which we add the constraint

$$z \leq \lfloor z_k \rfloor.$$

If (x^k, y^k) satisfies all integrality constraints, we perform a bilevel-feasibility check—important step that distinguishes the method from the classic branch-and-bound approach for MILPs (Algorithm 5) and extends it to the bilevel setting. To check for bilevel feasibility, we solve the x^k-parameterized lower-level problem; see Line 9. Note that, when performing this bilevel-feasibility check, we ignore any bounds that may have been imposed on the follower's variables due to branching. We show in the following that this resolves the issues encountered in Example 8.12. If $d^\top y^k \leq \varphi(x^k)$ holds, the point (x^k, y^k) is bilevel feasible and has a better value than the current incumbent solution. Hence, we update the incumbent and fathom the current node; see Lines 10–12. If (x^k, y^k) is not bilevel feasible, we proceed with a bilevel-specific branching step. We first check whether all linking variables have already been fixed by branching; see Line 13. A variable is considered fixed if its lower and upper bound, imposed by branching, coincide. If there are still linking variables whose values have not been fixed by branching yet, we choose one of these variables to branch on; see Line 14. Note that branching on a linking variable x_i^k now means that we branch on an integer-valued variable. To this end, we impose one of the following two variable disjunctions. The first disjunction is given by

$$x_i \geq x_i^k + 1 \quad \text{or} \quad x_i \leq x_i^k, \tag{8.11}$$

which is equivalent to the one used for branching on integer variables with fractional values. The second variable disjunction is given by

$$x_i \geq x_i^k \quad \text{or} \quad x_i \leq x_i^k - 1. \tag{8.12}$$

Each of the two disjunctions leads to the generation of two new sub-problems, one associated with each inequality. To avoid adding the same constraints in different parts of the search tree, the choice of which disjunction to apply depends on the branching decisions that have been made along the path from the root node to node k. To make this more specific, we illustrate this in Section 8.4 using two examples. More details about branching on linking variables can also be found in Section 2.3 in Tahernejad et al. (2020).

If all linking variables have been fixed by branching, we proceed with what is referred to in the literature as a refinement step (Lines 15–21). In the refinement step, we solve the refinement problem (8.10), see also Definition 6.15, to ensure that an optimal response of the follower (i) satisfies all coupling constraints (if present) and (ii) complies with the optimistic solution concept. By Lemma 6.16, the refinement problem (8.10) is either infeasible or solvable. If the problem is infeasible, we can fathom the current node because x^k cannot be part of a bilevel-feasible point; see Line 17. Otherwise, the pair $(\tilde{x}, \tilde{y})$ computed in Line 15 of Algorithm 6 is bilevel feasible and can be used to update the incumbent; see Lines 18–20. Once all linking variables have been fixed by branching, the lower-level problem is uniquely determined. Because we have already computed an optimal response of the follower using the refinement problem, no further improvements can then be achieved by additional branching. Hence, we can fathom the current node; see Line 21.

Theorem 8.14 *Suppose that Assumptions 6.5 and 6.13 hold. Then, if we embed Algorithm 6 into the branch-and-bound framework of Algorithm 5, we obtain a method that terminates after a finite number of visited nodes with an optimal solution to Problem* (8.1) *or with the correct indication of infeasibility.*

Proof: We start by proving finite termination. By Assumptions 6.5 and 6.13, all integer variables have finite bounds. Hence, only a finite number of branch-and-bound nodes can be generated. At each node, we may need to solve up to two additional MILPs: the lower-level problem and the refinement problem. Given the finiteness of branch-and-bound methods for MILPs and, again, by Assumption 6.5, each node can be processed using a finite number of operations. The correctness of the method now follows from the fact that, due to Line 15, the best bilevel-feasible point $(\tilde{x}, \tilde{y})$ w.r.t. the upper-level objective function value is computed at a node. Hence, no further improvements can be achieved by additional branching. □

Let us point out that Algorithm 6 allows for branching on both the leader's and the follower's variables; see Line 8. This is possible because any bounds that have been imposed on the follower's variables by branching are ignored when computing $\varphi(x^k)$ in Line 9 of the method. Finally, the flexibility to branch on both the leader's and the follower's variables leads to the fact that the resulting branch-and-bound method for bilevel MILPs has many degrees of freedom. As in the single-level case, we are free to choose which variable to branch on and which node to process next; see Section 7 for a more detailed discussion.

8.4 Illustrative Examples

Let us now explore how the branch-and-bound method for bilevel MILPs (Algorithm 5 and 6) works in practice. To this end, we return to the two illustrative examples by Moore and Bard (1990).

Example 8.15 (Example 6.2—Revisited) We consider the bilevel problem

$$\begin{aligned} \min_{x \in \mathbb{Z}, y} \quad & -x - 10y \\ \text{s.t.} \quad & y \in \mathcal{S}(x), \end{aligned}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the problem

$$\begin{aligned} \min_{y \in \mathbb{Z}} \quad & y \\ \text{s.t.} \quad & -25x + 20y \leq 30, \\ & x + 2y \leq 10, \\ & 2x - y \leq 15, \\ & 2x + 10y \geq 15. \end{aligned}$$

We now apply a depth-first search branch-and-bound method with the node processing procedure in Algorithm 6 to this problem. A possible branch-and-bound search tree is depicted in Figure 8.4.

At the root node (node 0), we consider the LP relaxation of the single-level relaxation. The solution to this problem is $(x, y) = (2, 4)$, which is integer but not bilevel feasible. Because we have not branched on the leader's variable yet, we now generate two new sub-problems by branching on the linking variable x. To this end, let us consider the first variable disjunction (8.11). This means that we create two new sub-problems in which we either add the constraint $x \geq 3$ or $x \leq 2$. This is illustrated in Figure 8.5. At node 1 of the branch-and-bound search tree, we consider the problem in which we add the constraint $x \geq 3$. The solution to this problem is $(x, y) = (3, 3.5)$. Because $y = 3.5$ is fractional, we branch on the follower's variable next. Fathoming due to infeasibility (node 2) and backtracking then leads us to node 3, which has the solution $(x, y) = (4, 3)$. Again, because the leader's variable x has not been fixed by branching yet, we generate two new sub-problems using the variable disjunction (8.11). This means that we either add the constraint $x \geq 5$ or $x \leq 4$. Further branching, fathoming due to infeasibility, and backtracking then leads us to node 9, which has the solution $(x, y) = (8, 1)$ with a value of -18. This point is bilevel feasible and its value is better than the current incumbent value, which has been set to $+\infty$ initially. Thus, we update the incumbent, i.e., we update the incumbent solution, set $\mathcal{U} \leftarrow -18$, and fathom the node. Backtracking then leads us to

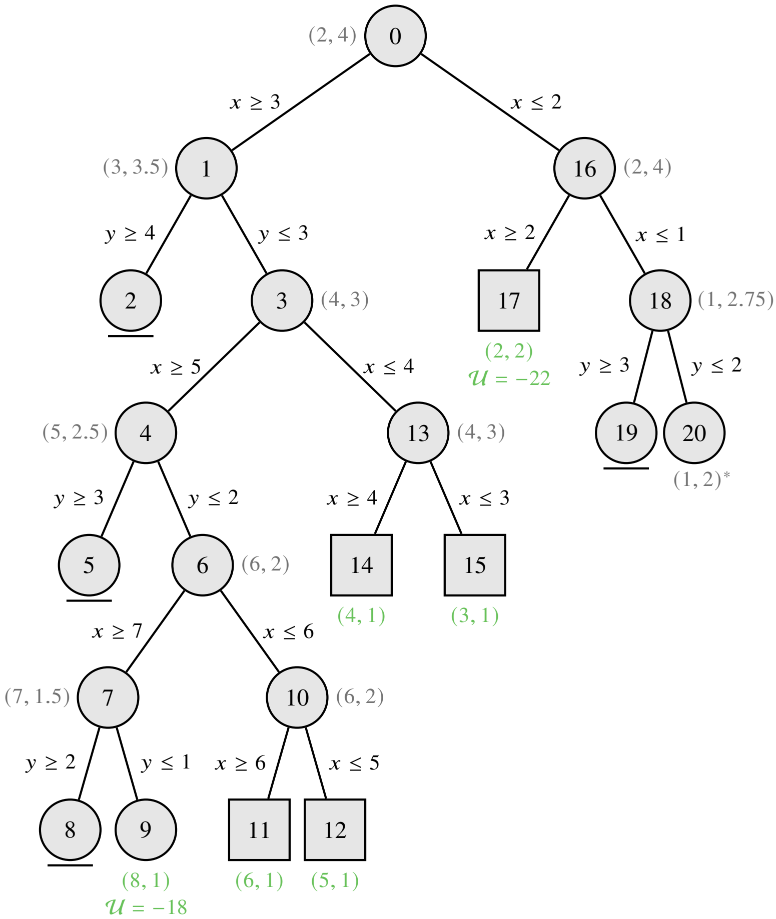


Figure 8.4 The branch-and-bound search tree in Example 8.15. Tuples of the form (x, y) at the branch-and-bound nodes denote the solutions to the problems considered at these nodes. At rectangular nodes, a refinement step is performed. Node 20 is pruned due to bounding, which is illustrated by an asterisk.

node 10. The solution to the problem considered at this node is $(x, y) = (6, 2)$, which is integer but not bilevel feasible. The leader's linking variable x has not been fixed by branching yet. However, we have already applied the first variable disjunction (8.11) when branching on x at node 6. Hence, to avoid duplicating constraints and to ensure that we make progress in the search tree, we now apply the second variable disjunction (8.12) to branch on x. This results in two

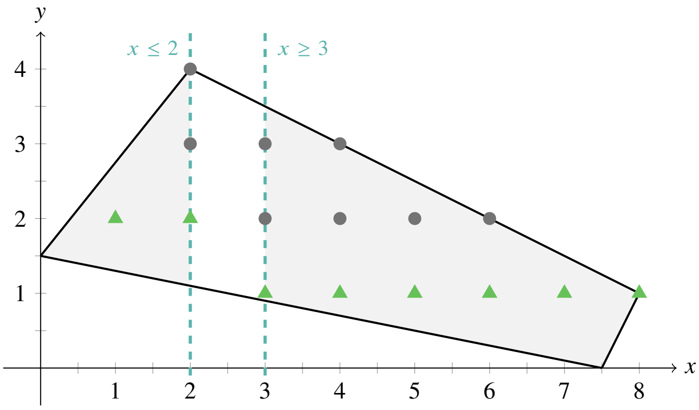


Figure 8.5 An illustration of the two new sub-problems that are generated at node 0 of the branch-and-bound search tree in Example 8.15 by imposing either $x \geq 3$ or $x \leq 2$.

new sub-problems: one in which we add $x \geq 6$ and one in which we add $x \leq 5$. At node 11, we consider the problem in which we add the constraint $x \geq 6$. The solution to the problem considered at this node is $(x, y) = (6, 2)$, which is integer but not bilevel feasible. However, the leader's variable is fixed by branching now. Thus, we perform a refinement step, which yields the bilevel-feasible point $(x, \tilde{y}) = (6, 1)$ with a value of -16. Because $-16 > -18 = \mathcal{U}$ holds, we do not update the incumbent and fathom the node. Backtracking then leads us to node 12, which is processed in the same way. Further backtracking then leads to node 13. The solution to the problem considered at this node is $(x, y) = (4, 3)$, which is again integer but not bilevel feasible. The leader's linking variable x has not been fixed by branching yet, but we have already applied the first variable disjunction (8.11) when branching on x at node 3. Thus, we now apply the second variable disjunction (8.12) to branch on x, i.e., we generate two new sub-problems in which we either add $x \geq 4$ or $x \leq 3$. At node 14, we consider the problem in which we add the constraint $x \geq 4$. The solution to this problem is $(4, 3)$, which is not bilevel feasible. However, the leader's variable is fixed by branching. Therefore, we now perform a refinement step and obtain the bilevel-feasible point $(x, \tilde{y}) = (4, 1)$ with a value of -14. Because $-14 > -18 = \mathcal{U}$ holds, we do not update the incumbent and fathom the node. Backtracking then leads us to node 15, which is processed in the same way. Backtracking and further branching then leads us to node 17. The leader's variable is fixed by branching and the solution to the problem considered at this node is $(x, y) = (2, 4)$, which is integer but not bilevel feasible. Hence, we perform a refinement step to obtain the bilevel-feasible point $(x, \tilde{y}) = (2, 2)$ with a value of $-22 < \mathcal{U}$. We update the incumbent and fathom the node. Finally, backtracking, further branching, and fathoming due to infeasibility (node 19) leads us to node 20, which has the solution $(x, y) = (1, 2)$ with an objective function value of $-21 > -22 = \mathcal{U}$. No further improvements can be achieved by additional branching and, thus, the node can be pruned due to bounding. Because there are no more open branch-and-bound nodes, the method terminates with the optimal solution $(x^*, y^*) = (2, 2)$. △

Example 8.16 (Example 6.1—Revisited) We now apply a depth-first search branch-and-bound method with the node processing scheme in Algorithm 6 to the bilevel problem in Example 6.1. To this end, we first re-state the upper level as a minimization problem:

$$\min_{x \in \mathbb{Z}, y} \quad x + 2y \quad \text{s.t.} \quad x \geq 0, \ y \in \mathcal{S}(x). \tag{8.13}$$

As before, the set $\mathcal{S}(x)$ is the set of optimal solutions to the x-parameterized lower-level problem

$$\begin{aligned} \max_{y \in \mathbb{Z}} \quad & f(x, y) = y \\ \text{s.t.} \quad & -x + 2.5y \leq 3.75, \\ & x + 2.5y \geq 3.75, \\ & 2.5x + y \leq 8.75, \\ & y \geq 0. \end{aligned}$$

A possible branch-and-bound search tree for this example is depicted in Figure 8.6.

At the root node, we consider the LP relaxation of the single-level relaxation, which has the solution $(x, y) = (0, 1.5)$. Because $y = 1.5$ is fractional, we branch on the follower's variable, i.e., we generate two new sub-problems in which we impose either $y \geq 2$ or $y \leq 1$. At node 1 of the branch-and-bound search tree, we consider the problem in which we add the constraint $y \geq 2$. The optimal solution to this problem is $(x, y) = (1.25, 2)$. Because $x = 1.25$ is fractional, we branch on the leader's variable next. At node 2, we consider the problem with $x \geq 2$, which has the optimal solution $(x, y) = (2, 2)$ with an objective function value of 6. This point is bilevel feasible and its value is better than the current incumbent value, which has been set to $+\infty$ initially. Thus, we update the incumbent solution, set $\mathcal{U} \leftarrow 6$, and fathom the node. Backtracking and fathoming due to infeasibility then leads us to node 4, which has the solution $(x, y) = (1.25, 1)$. Further branching on fractional values leads us to node 6. The optimal solution to the problem at this node is $(x, y) = (2, 1)$, which

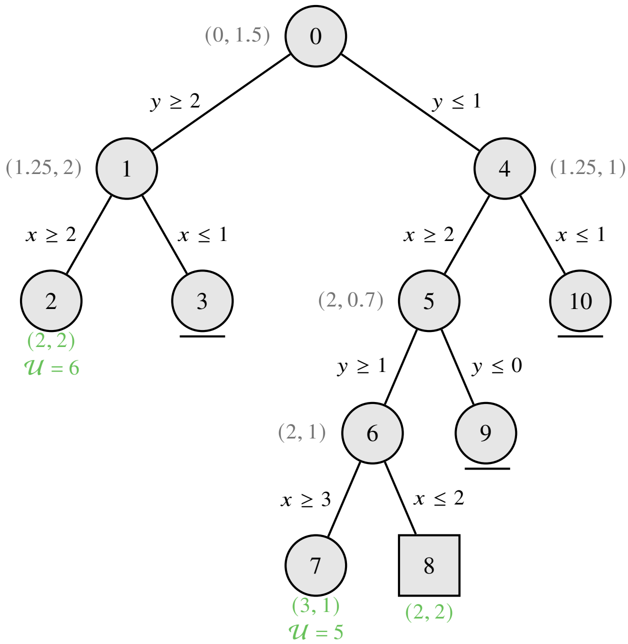


Figure 8.6 The branch-and-bound search tree in Example 8.16. Tuples of the form (x, y) at the branch-and-bound nodes denote the solutions to the problems considered at these nodes. At the rectangular node, a refinement step is performed.

is integer but not bilevel feasible. Because the leader's linking variable x has not been fixed by branching yet, we generate two new sub-problems by applying the first variable disjunction (8.11). At node 7 of the branch-and-bound search tree, we thus consider the problem in which we add the constraint $x \geq 3$. The optimal solution to this problem is the bilevel-feasible point $(x, y) = (3, 1)$ with a value of $5 < \mathcal{U} = 6$. Hence, we update the incumbent and fathom the current node. Backtracking then leads us to node 8. The solution $(x, y) = (2, 1)$ to the problem considered at this node is integer but not bilevel feasible and the leader's variable has been fixed by branching. Hence, we now perform a refinement step. To this end, we solve the refinement problem that is parameterized in $x = 2$, which is given by

$$\min_{y \in \mathbb{Z}} \quad 2 + 2y \quad \text{s.t.} \quad 2.5y \leq 5.75,\ 2.5y \geq 1.75,\ y \leq 3.75,\ y \geq 2,\ y \geq 0.$$

Note that the second-last constraint corresponds to $d^\top y \leq \varphi(x^k)$. The optimal solution to this problem is $\tilde{y} = 2$. Hence, we obtain the bilevel-feasible point $(x, \tilde{y}) = (2, 2)$ with a value of 6. Because $6 > \mathcal{U} = 5$ holds, we do not update the incumbent and fathom the current node. All remaining open nodes are processed by further backtracking and fathoming due to infeasibility. The method terminates with the optimal solution $(x^*, y^*) = (3, 1)$ to Problem (8.13). Finally, multiplying the optimal objective function value by -1 yields the optimal value of -5 for the original bilevel problem, which has a maximization problem in the upper level. △

Exercise 8.17 Consider the integer linear bilevel problem taken from DeNegre and Ralphs (2009), which is given by

$$\begin{aligned} \max_{x,y} \quad & y \\ \text{s.t.} \quad & x \in \mathbb{Z},\ y \in \mathcal{S}(x), \end{aligned} \tag{8.14}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the integer linear problem

$$\begin{aligned} \min_{y} \quad & y \\ \text{s.t.} \quad & -x + y \le 2, \\ & -2x - y \le -2, \\ & 3x - y \le 3, \\ & y \le 3, \\ & y \in \mathbb{Z}. \end{aligned}$$

(i) Plot the shared constraint set of Problem (8.14) in a coordinate system.
(ii) Determine the bilevel-feasible set of Problem (8.14) and highlight it in your plot.
(iii) Use (ii) to determine the optimal solution to the bilevel problem (8.14).
(iv) Apply a depth-first search branch-and-bound method with the node processing scheme in Algorithm 6 to solve Problem (8.14) by hand. Use a general-purpose MILP solver to solve the arising sub-problems and visualize your progress in a search tree. You can use your result from (iii) to verify your solution.

Exercise 8.18 (Bilevel Knapsack Interdiction—Revisited) Consider the bilevel knapsack interdiction problem

$$\begin{aligned} \min_{x,y} \quad & 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 2x_1 + x_2 + x_3 \le 2, \\ & x \in \{0, 1\}^3,\ y \in \mathcal{S}(x), \end{aligned} \tag{8.15}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to the problem

$$\begin{aligned} \max_{y} \quad & 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 4y_1 + 3y_2 + 2y_3 \leq 4, \\ & y_i \leq 1 - x_i, \quad i \in \{1, 2, 3\}, \\ & y \in \{0, 1\}^3 ; \end{aligned}$$

see Example 1 in Caprara et al. (2016).

(i) Determine all feasible decisions of the leader, i.e., determine the set

$$X = \left\{x \in \{0, 1\}^3 : 2x_1 + x_2 + x_3 \leq 2\right\}.$$

(ii) For each $x \in X$, determine the optimal response y^* of the follower and the associated objective function value.

(iii) Use (i) and (ii) to determine the optimal solution to the bilevel knapsack interdiction problem (8.15).

(iv) Apply a depth-first search branch-and-bound method with the node processing scheme in Algorithm 6 to solve Problem (8.15) by hand. Use a general-purpose MILP solver to solve the arising sub-problems and visualize your progress in a search tree. You can use your result from (iii) to verify your solution.

Exercise 8.19 Consider the node processing scheme in Algorithm 6 in which a refinement step (Lines 15–20) is performed whenever the leader's linking variables have been fixed because of branching.

(i) Modify the method so that a refinement step is performed whenever the current node solution is integer but not bilevel feasible, regardless of whether the linking variables have been fixed by branching or not. Write down the pseudo-code for this modified node processing procedure.

(ii) What may be advantages or disadvantages of such a modification?

(iii) Apply a depth-first search branch-and-bound method with the modified node processing procedure to the bilevel problem in Example 8.15. Illustrate a possible search tree. What differences compared to the search tree in Example 8.15 do you observe?

Hint: You may want to read Tahernejad et al. (2020).

8.5 What You Should Know Now!

1. Given a bilevel MILP, what possibilities do you know to construct relaxations or restrictions of this problem?
2. Can you state the continuous counterpart of a bilevel MILP?
3. Is the continuous counterpart of a bilevel MILP always a relaxation of the underlying bilevel problem? Is it always a restriction of the underlying bilevel problem?
4. What are the insights from the examples by Moore and Bard (1990)?
5. What branch-and-bound fathoming rules can we carry over from mixed-integer linear single-level problems to mixed-integer linear bilevel problems if we use the continuous counterpart as the root-node problem?
6. What can go wrong with those fathoming rules that we cannot use if we use the continuous counterpart as the root-node problem?
7. How does the branch-and-bound method for mixed-integer linear bilevel problems by Fischetti et al. (2018a) work?
8. Can you explain why we assume that all linking variables are bounded integers?
9. What is the idea behind the refinement step?
10. Can you explain why, after solving the refinement problem, we can prune a node at which the linking variables have been fixed by branching (Line 21 in Algorithm 6)?
11. Is the assumption of a bounded shared constraint set really necessary? If yes, can you explain why? If not, what can we do if the assumption is relaxed?
12. Can you prove the correctness of the branch-and-bound method for bilevel MILPs?
13. Do you have an idea how we could simplify the refinement step for the case in which there are no coupling constraints?

9

Branch-and-Cut for Bilevel MILPs

In the branch-and-bound method for mixed-integer linear bilevel problems (Algorithm 5 and 6), we branch whenever the solution to the problem considered at a node is integer infeasible or bilevel infeasible. As we have seen in our illustrative examples in Section 8.4, this can have considerable practical implications. Even for moderately sized bilevel problems, we may need to process a significant number of branch-and-bound nodes. For tackling larger bilevel problems, it is thus essential to develop techniques that improve the performance of the branch-and-bound method for bilevel MILPs. One key technique is the integration of so-called *cutting planes*, which then leads to what is called a *branch-and-cut* algorithm for mixed-integer linear bilevel problems. This is what we do in this chapter. To this end, we still consider bilevel problems of the form

$$\begin{aligned} \min_{x \in X, y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \arg\min_{\bar{y} \in Y} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\}. \end{aligned} \tag{9.1}$$

Here and in what follows, we use the definitions and notation formally introduced in Chapter 6. As before, the sets X and Y capture integrality constraints on (a subset of) the x- and y-variables, respectively. Moreover, Assumptions 6.5 and 6.13 are standing assumptions for the remainder of this chapter. For the details regarding their implications, we refer to the respective discussions in Chapter 6.

The first branch-and-cut approach for purely integer linear bilevel problems without coupling constraints, i.e., Problem (9.1) with $B = 0$, $X = \mathbb{Z}^{n_x}$, and $Y = \mathbb{Z}^{n_y}$, was presented in the seminal paper by DeNegre and Ralphs (2009). The authors significantly extend the ideas of the early branch-and-bound method proposed by Moore and Bard (1990) so that the method by DeNegre and Ralphs (2009) may be seen as a turning point in algorithmic mixed-integer bilevel optimization. Over the past 15 years, many other influential works on solution approaches for bilevel MILPs such as, e.g., Fischetti et al. (2017, 2018a), Tahernejad et al. (2020), and Xu and Wang (2014), have followed. Today, branch-and-cut is one of the most commonly used approaches for tackling mixed-integer linear bilevel problems in practice. In what follows, we present a branch-and-cut method that is based on the branch-and-bound framework presented in Section 8.3. To this end, we start by introducing the concept of valid inequalities from single-level mixed-integer optimization and discuss how they are applied in the bilevel context in Section 9.1. In Section 9.2, we then show how these inequalities can be used to develop a branch-and-cut framework for bilevel MILPs and apply the method to illustrative examples. Finally, in Section 9.3, we briefly discuss two general-purpose solvers for bilevel MILPs, which are based on the branch-and-cut framework discussed in Section 9.2.

9.1 Valid Inequalities

When applying the branch-and-bound method for bilevel MILPs (Algorithm 5 and 6) to Problem (9.1), we solve an LP at each node of the branch-and-bound search tree. These linear optimization problems are obtained from the LP relaxation

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \bar{\Omega} \tag{9.2}$$

of the single-level relaxation by including additional variable bounds imposed due to branching. As before, $\bar{\Omega}$ denotes the continuous relaxation of the shared constraint set Ω of Problem (9.1). Our goal now is to obtain stronger formulations for the problems considered at the branch-and-bound nodes to enhance the bounding procedures in the overall branch-and-bound framework. To this end, we iteratively add inequalities to $\bar{\Omega}$ that are valid for the feasible set $\mathcal{F}$ of the bilevel problem (9.1) but that are violated by points in the set $\bar{\Omega} \setminus \mathcal{F}$.

Definition 9.1 ((Globally) Valid Inequality) We call an inequality

$$\alpha^\top x + \beta^\top y \geq \gamma$$

with $\alpha \in \mathbb{R}^{n_x}$, $\beta \in \mathbb{R}^{n_y}$, and $\gamma \in \mathbb{R}$ *(globally) valid* for the feasible set $\mathcal{F}$ of the bilevel problem (9.1) if

$$\mathcal{F} \subseteq \left\{(x, y) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} : \alpha^\top x + \beta^\top y \geq \gamma\right\}$$

holds.

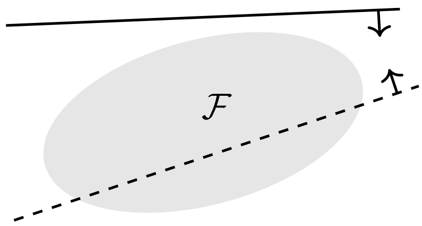


Figure 9.1 The solid line represents a (globally) valid inequality for the set $\mathcal{F}$ (gray area), whereas the dashed line is not globally valid for $\mathcal{F}$.

A globally valid inequality is an inequality that is satisfied for all bilevel-feasible points, independent of where or how it is generated. A simple illustration of a (globally) valid inequality is given in Figure 9.1. Besides globally valid inequalities, there also exist so-called *locally valid inequalities*, which are only valid for certain parts of the bilevel-feasible set $\mathcal{F}$.

Definition 9.2 (Locally Valid Inequality) We call an inequality

$$\alpha^\top x + \beta^\top y \geq \gamma$$

with $\alpha \in \mathbb{R}^{n_x}$, $\beta \in \mathbb{R}^{n_y}$, and $\gamma \in \mathbb{R}$ *locally valid* for the restricted feasible set $\mathcal{F}_k = \mathcal{F} \cap \bar{\Omega}_k$ at node k of a branch-and-cut search tree if

$$\mathcal{F}_k \subseteq \left\{(x, y) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} : \alpha^\top x + \beta^\top y \geq \gamma\right\}$$

holds.

Note that every globally valid inequality is also locally valid. In contrast to globally valid inequalities, however, a locally valid inequality is usually generated from node-specific information such as, e.g., additional variable bounds that are imposed by branching. Hence, it may only be valid for the restricted feasible set considered at that specific node as well as for all nodes in the sub-tree rooted at that node. In particular, this means that a locally valid cut may not be valid for the entire bilevel-feasible set $\mathcal{F}$. Consequently, any locally valid inequality has to be removed once the search leaves the sub-tree at which it has been generated.

Except for the special case in which $\bar{\Omega}$ corresponds to the convex hull of $\mathcal{F}$, there exist inequalities that are valid for $\mathcal{F}$ but that are violated by some $(x, y) \in \bar{\Omega}$. To derive valid inequalities for $\mathcal{F}$, it is thus essential to use information that is not contained in $\bar{\Omega}$. Before we show how this can be done, let us first illustrate the concept of valid inequalities using a small example.

Example 9.3 (Example 6.1—Revisited) We revisit the bilevel problem in

Example 6.1. Recall that the shared constraint set contains three integer points: $(2, 1)$, $(2, 2)$, and $(3, 1)$. The latter two points are also bilevel feasible. We now consider the inequality

$$3x + 2y \geq 9, \tag{9.3}$$

which is depicted in Figure 9.2 (top). We observe that Inequality (9.3) is satisfied by the two bilevel-feasible points $(2, 2)$ and $(3, 1)$ and, thus, it is valid for the feasible set $\mathcal{F}$ of the bilevel problem. In particular, Inequality (9.3) cuts off the integer-feasible but bilevel-infeasible point $(2, 1)$. However, we emphasize that an inequality may still be valid for the bilevel problem even if it does not cut off bilevel-infeasible points. An example, which is obtained by shifting Inequality (9.3) to the left, is depicted in Figure 9.2 (top) as well. Nevertheless, if a valid inequality indeed excludes certain points in the set Ω (or the set $\bar{\Omega}$), it is often referred to as a cutting plane (or simply a *cut*).

Let us now consider the inequality

$$3x - 4y \geq 0 \tag{9.4}$$

shown in Figure 9.2 (bottom) and plug in the bilevel-feasible points $(2, 2)$ and $(3, 1)$. This yields

$$3 \cdot 2 - 4 \cdot 2 = -2 < 0 \quad \text{and} \quad 3 \cdot 3 - 4 \cdot 1 = 5 > 0,$$

i.e., the bilevel-feasible point $(2, 2)$ violates Inequality (9.4). Hence, Inequality (9.4) is not (globally) valid for the feasible set of the bilevel problem. However, it could still be locally valid. For instance, suppose that a branching decision imposes $x \geq 3$. Then, the bilevel-feasible point $(2, 2)$ is no longer feasible for the sub-problem defined by this branching decision. Hence, Inequality (9.4) is valid for the restricted feasible set after introducing the branching decision $x \geq 3$, which is $\{(3, 1)\}$, i.e., it is locally valid; see Figure 9.2 again. △

For the ease of presentation, we now focus on globally valid inequalities in the following and briefly illustrate the use of locally valid inequalities later in Example 9.12. To generate valid inequalities, we consider the so-called *separation problem*, which we formally define in the following.

Definition 9.4 (Separation Problem) The *separation problem* for $\mathcal{F}$ is to determine whether a given point $(\bar{x}, \bar{y}) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y}$ is feasible for the bilevel problem (9.1) and, if not, to generate $\alpha \in \mathbb{R}^{n_x}$, $\beta \in \mathbb{R}^{n_y}$, and $\gamma \in \mathbb{R}$ such that

$$\alpha^\top x + \beta^\top y \geq \gamma \quad \text{for all } (x, y) \in \mathcal{F} \quad \text{and} \quad \alpha^\top \bar{x} + \beta^\top \bar{y} < \gamma.$$

Recall that a point (x, y) is feasible for the bilevel problem (9.1) if and only if it satisfies the following conditions.

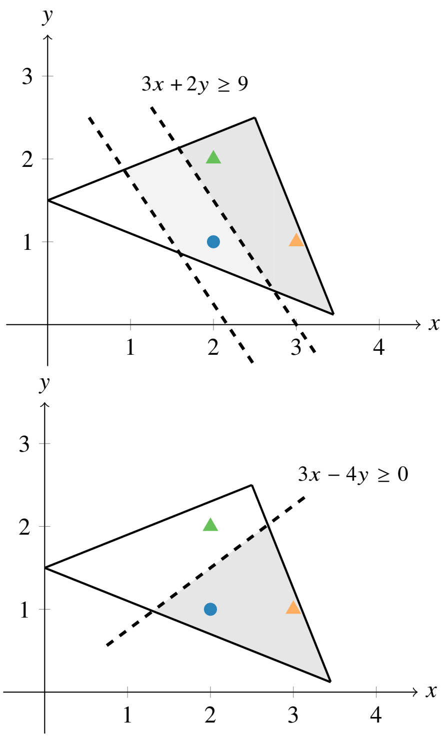


Figure 9.2 Both figures illustrate the bilevel problem in Example 6.1. Discrete points (dot and triangles) are feasible for the shared constraint set. Triangles represent bilevel-feasible points. The inequalities (dashed lines) shown in the top figure are valid for the bilevel-feasible set $\mathcal{F}$, whereas the one shown in the bottom figure is not (globally) valid for $\mathcal{F}$.

Condition 1 $(x, y) \in \bar{\Omega}$.
Condition 2 $(x, y) \in X \times Y$.
Condition 3 $y \in \mathcal{S}(x) := \arg\min_{\bar{y}} \{d^\top \bar{y} : Cx + D\bar{y} \geq b,\ \bar{y} \in Y\}$.

Condition 1 ensures that the point (x, y) satisfies all upper- and lower-level constraints except from the integrality conditions, i.e., it is contained in the continuous relaxation $\bar{\Omega}$ of the shared constraint set. Condition 2 ensures that a bilevel-feasible point satisfies all integrality restrictions, whereas Condition 3 states that the follower has to respond optimally to a given leader's decision. When considered in a branch-and-bound framework, a solution $(\bar{x}, \bar{y})$ to the problem considered at one of the nodes of the search tree already satisfies Condition 1. Hence, to determine whether the point is feasible for the bilevel problem (9.1), one needs to check if Conditions 2 and 3 are satisfied as well. If at least one of these conditions is violated by $(\bar{x}, \bar{y})$, the separation problem is to find a hyperplane that cuts off (or *separates*) the point $(\bar{x}, \bar{y})$ from the bilevel-feasible set $\mathcal{F}$. Generating and adding valid inequalities provides an alternative to branching for reducing the feasible region of an optimization problem. By applying valid inequalities, we aim at eliminating parts of the shared constraint set that do not contain any point that is feasible for the bilevel problem (9.1). For this purpose, we may distinguish between two types of cuts: those that separate points violating Condition 2 and those that separate points violating Condition 3. More formally, we have the following two results.

Lemma 9.5 *If the inequality $\alpha^\top x + \beta^\top y \geq \gamma$ is valid for the shared constraint set Ω, it is also valid for the bilevel-feasible set $\mathcal{F}$ of Problem* (9.1).

Proof: The claim immediately follows from $\mathcal{F} \subseteq \Omega$ and Definition 9.1. □

We have already seen an exemplary inequality that satisfies the requirements of Lemma 9.5 in Example 9.3 (the left inequality shown on the top of Figure 9.2). We exploit Lemma 9.5 to separate points that violate (some of) the integrality restrictions, i.e., points that violate Condition 2. The next lemma is used to separate points that violate Condition 3.

Lemma 9.6 *Let $(\bar{x}, \bar{y}) \in \Omega$ be such that $\bar{y} \notin \mathcal{S}(\bar{x})$ holds, i.e., the point $(\bar{x}, \bar{y})$ violates Condition 3. Then, if the inequality $\alpha^\top x + \beta^\top y \geq \gamma$ is valid for $\Omega \setminus \{(\bar{x}, \bar{y})\}$, it is also valid for $\mathcal{F}$.*

Proof: Due to $\bar{y} \notin \mathcal{S}(\bar{x})$, we have $(\bar{x}, \bar{y}) \notin \mathcal{F}$. Because $\mathcal{F} \subseteq \Omega$ holds, we thus obtain $\mathcal{F} \subseteq \Omega \setminus \{(\bar{x}, \bar{y})\}$. The claim then follows from Definition 9.1. □

There already exist some inequalities that can, in principle, be used to separate integer- or bilevel-infeasible points. In particular, by Lemma 9.5, any of the existing MILP techniques can be applied to separate points that violate Condition 2. An overview of various classes of valid inequalities for this purpose can, e.g., be found in Clautiaux and Ljubić (2025) and Cornúejols (2008). Moreover, we refer to Tahernejad and Ralphs (2025) for an overview of classes of bilevel-specific cuts that can be used to separate points violating Condition 3. Nevertheless, while any existing class of valid inequalities can be used in principle, a suitable choice may depend on the specific structure of the bilevel problem at hand. In the literature, one thus often finds inequalities that are tailored to specific application problems. We explore such tailored cuts for interdiction problems in Chapter 10. Moreover, we discuss the important class of so-called *intersection cuts* in Chapter 11 and 12. For now, however, let us take a look at some generic cuts that are rather easy to generate. In what follows, we use $A_{i\cdot}$ again to denote the ith row of a matrix A. Moreover, let us mention that all inequalities considered in the following are globally valid.

Proposition 9.7 (See Proposition 1 in DeNegre and Ralphs (2009)) *Suppose that the matrices A, B, C, D and the vectors a, b are integer-valued, as well as that $X = \mathbb{Z}^{n_x}$ and $Y = \mathbb{Z}^{n_y}$ holds. Let $(\bar{x}, \bar{y}) \in \Omega \setminus \mathcal{F}$ be a basic feasible solution to the LP relaxation of the single-level relaxation of the bilevel problem (9.1), i.e., of*

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \bar{\Omega}.$$

Further, let $\mathcal{A}_1$ and $\mathcal{A}_2$ be the index sets of the upper- and lower-level constraints, respectively, that are binding in $(\bar{x}, \bar{y})$. Then, the inequality

$$\alpha^\top x + \beta^\top y \geq \gamma + 1$$

with

$$\begin{aligned} \alpha^\top &= \sum_{i\in\mathcal{A}_1} A_{i\cdot} + \sum_{i\in\mathcal{A}_2} C_{i\cdot}, \\ \beta^\top &= \sum_{i\in\mathcal{A}_1} B_{i\cdot} + \sum_{i\in\mathcal{A}_2} D_{i\cdot}, \\ \gamma &= \sum_{i\in\mathcal{A}_1} a_i + \sum_{i\in\mathcal{A}_2} b_i \end{aligned}$$

is valid for the feasible set $\mathcal{F}$ of the bilevel problem (9.1).

Proof: Because $(\bar{x}, \bar{y})$ is basic feasible, there are $n_x + n_y$ linearly independent constraints in the description of $\bar{\Omega}$ that are binding in $(\bar{x}, \bar{y})$. Hence, the point $(\bar{x}, \bar{y})$ is the unique solution to the linear system

$$\begin{aligned} A_{i\cdot}x + B_{i\cdot}y &= a_i \quad \text{for all } i \in \mathcal{A}_1, \\ C_{i\cdot}x + D_{i\cdot}y &= b_i \quad \text{for all } i \in \mathcal{A}_2. \end{aligned}$$

This implies that $(\bar{x}, \bar{y})$ satisfies $\alpha^\top \bar{x} + \beta^\top \bar{y} = \gamma$. Moreover, the inequality $\alpha^\top x + \beta^\top y \geq \gamma$ is valid for Ω because any point $(x, y) \in \Omega$ satisfies

$$\begin{aligned} A_{i\cdot}x + B_{i\cdot}y &\geq a_i \quad \text{for all } i \in \mathcal{A}_1, \\ C_{i\cdot}x + D_{i\cdot}y &\geq b_i \quad \text{for all } i \in \mathcal{A}_2. \end{aligned}$$

In addition, $\alpha^\top x$ and $\beta^\top y$ are integer for any $(x, y) \in \Omega$ because α and β are integer by assumption. Moreover, we have $\gamma \in \mathbb{Z}$. Consequently, there is no point $(x, y) \in \Omega$ for which $\alpha^\top x + \beta^\top y \in (\gamma, \gamma + 1)$ holds. This implies that the inequality

$$\alpha^\top x + \beta^\top y \geq \gamma + 1$$

is valid for $\Omega \setminus \{(\bar{x}, \bar{y})\}$. Because $(\bar{x}, \bar{y}) \notin \mathcal{F}$ holds, we have $\mathcal{F} \subseteq \Omega \setminus \{(\bar{x}, \bar{y})\}$ and, thus, the inequality is also valid for $\mathcal{F}$. □

It is easy to see that the latter proposition can also be used to generate locally valid cuts if the shared constraint set Ω is replaced by the feasible set Ω_k of the problem at some node k in the branch-and-bound search tree.

Example 9.8 (Example 6.2—Revisited) We revisit the bilevel problem in Example 6.2, which is again illustrated in Figure 9.3. The optimal solution $(\bar{x}, \bar{y}) = (2, 4)$ to the single-level relaxation of the problem is not bilevel feasible. The constraints

$$-25x + 20y \leq 30 \quad \text{and} \quad x + 2y \leq 10 \tag{9.5}$$

are binding in $(\bar{x}, \bar{y}) = (2, 4)$. Note that the requirements of Proposition 9.7 are satisfied. Hence, we can derive a valid inequality for the feasible set of the bilevel problem in Example 6.2 by adding both constraints in (9.5) and by decreasing the right-hand side by 1. Note that we subtract 1 on the right-hand side here instead of adding 1 (as in Proposition 9.7) because we have a $\leq$-constraint in this example. This yields the inequality

$$-24x + 22y \leq 39,$$

which corresponds to the red dashed line depicted in Figure 9.3. We observe that the inequality is very shallow and removes only a small part of the set $\bar{\Omega} \setminus \mathcal{F}$.

In this context, let us emphasize that different equivalent formulations of the same problem can lead to different valid inequalities. To see this, let us re-write

$$-25x + 20y \leq 30 \iff -5x + 4y \leq 6.$$

Adding $x + 2y \leq 10$ to the latter yields

$$-4x + 6y \leq 16 \iff -2x + 3y \leq 8.$$

Applying Proposition 9.7 again, we then obtain the valid inequality

$$-2x + 3y \leq 7,$$

which is the red dash-dotted line in Figure 9.3. We observe that reformulating the bilevel problem produces a "stronger" inequality in the sense that the inequality $-2x + 3y \leq 7$ eliminates a larger part of the set $\bar{\Omega} \setminus \mathcal{F}$ than the inequality $-24x + 22y \leq 39$. △

Example 9.8 further illustrates that a valid inequality obtained from Proposition 9.7 only separates the point $(\bar{x}, \bar{y}) \in \Omega \setminus \mathcal{F}$, whereas it does not cut off any other points in Ω. This can lead to the situation in which a branch-and-cut method

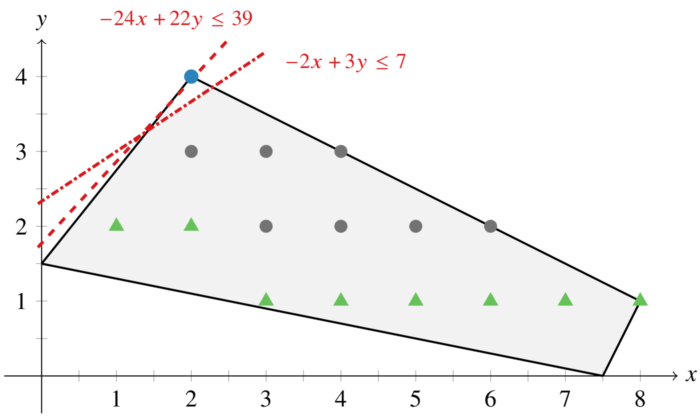


Figure 9.3 The bilevel problem in Example 6.2. Discrete points (dots and triangles) are feasible for the shared constraint set. Triangles represent bilevel-feasible points. The inequalities $-24x + 22y \leq 39$ and $-2x + 3y \leq 7$, which are obtained from Proposition 9.7, are both valid for the bilevel-feasible set. Moreover, they separate the blue point $(2, 4)$, which is the solution to the single-level relaxation of the bilevel problem.

generates a sequence of points $(\bar{x}, y^1)$, $(\bar{x}, y^2), \ldots, (\bar{x}, y^k)$ with $(\bar{x}, y^i) \in \Omega$ but $y^i \notin \mathcal{S}(\bar{x})$ for $i < k$. Nevertheless, it may be possible to obtain stronger cuts that prevent this situation. In the following, we show this for the special case in which the upper level is a binary problem.

Proposition 9.9 *Let $\bar{x} \in X = \{0,1\}^{n_x}$ be given such that*

$$\left\{y \in Y : A\bar{x} + By \geq a,\, C\bar{x} + Dy \geq b,\, d^\top y \leq \varphi(\bar{x})\right\} = \emptyset$$

holds. Then, the inequality

$$\sum_{i \in Z} x_i + \sum_{i \in O} (1 - x_i) \geq 1 \tag{9.6}$$

with

$$Z := \{i \in \{1, \ldots, n_x\} : \bar{x}_i = 0\} \quad \text{and} \quad O := \{i \in \{1, \ldots, n_x\} : \bar{x}_i = 1\}$$

is valid for the feasible set $\mathcal{F}$ of the bilevel problem (9.1).

Proof: By assumption, there is no optimal response of the follower to the given leader's decision $\bar{x}$. Let now $(x, y) \in \mathcal{F}$ be given arbitrarily. Because there is no y so that $(\bar{x}, y)$ is a bilevel-feasible point, the vector x must differ from $\bar{x}$ in at least one component. Because we have $x \in X = \{0,1\}^{n_x}$ by assumption, this means that $\|\bar{x} - x\|_1 \geq 1$ has to hold. Hence, and by the definition of the index sets Z and O, we obtain

$$1 \leq \|\bar{x} - x\|_1 = \sum_{i \in Z} |\bar{x}_i - x_i| + \sum_{i \in O} |\bar{x}_i - x_i| = \sum_{i \in Z} x_i + \sum_{i \in O} (1 - x_i).$$

Because $(x, y) \in \mathcal{F}$ has been chosen arbitrarily, this concludes the proof. □

Inequality (9.6) is used to eliminate points that are considered "no good" for the problem, which is why it is commonly known as a *no-good cut* in the literature; see, e.g., Fischetti et al. (2018a), Tahernejad and Ralphs (2025), and Wolsey (2020). A no-good cut (9.6) obtained from a vector $\bar{x}$ satisfying the requirements of Proposition 9.9 separates all points contained in the set $\{(\bar{x}, y) \colon y \in \mathbb{R}^{n_y}\}$. Compared to the valid inequalities from Proposition 9.7, no-good cuts are also more general as they allow for general MILPs at the lower level. Finally, let us mention that the assumption of binary leader's variables is not restrictive as any bounded integer can be represented using binary expansion; see Exercise 6.24.

9.2 A Branch-and-Cut Method for Mixed-Integer Linear Bilevel Problems

We now introduce the branch-and-cut method for bilevel MILPs of the form given in (9.1). The method extends the branch-and-bound framework (Algorithm 5 and 6) by incorporating valid inequalities into the optimization problems considered at the nodes of the branch-and-bound search tree. To streamline the discussion, we again only discuss the procedure for processing nodes in the following. The method to process node k of the branch-and-cut search tree is formally stated in Algorithm 7. At this node, we consider the problem

$$\min_{x,y} \quad c_x^\top x + c_y^\top y \quad \text{s.t.} \quad (x, y) \in \bar{\Omega}_k \subseteq \mathbb{R}^{n_x} \times \mathbb{R}^{n_y}. \tag{9.7}$$

Here, $\bar{\Omega}_k$ is obtained from $\bar{\Omega}$ by adding all valid inequalities that have been generated at nodes along the path from the root node to node k and by imposing all branching decisions that have been made along that path.

Under Assumption 6.5, Problem (9.7) is either infeasible or solvable. In the case of infeasibility, we do not consider the problem further and fathom the current node; see Line 3 of Algorithm 7. Otherwise, Problem (9.7) has an optimal solution (x^k, y^k). If its value is not better than the current incumbent value, we can fathom the node because no further improvements can be achieved through additional branching or adding cuts; see Line 6. If the point (x^k, y^k) yields a better value than the current incumbent solution, we proceed by checking

Algorithm 7 Processing Node k of the Branch-and-Cut Search Tree

Input: An instance of Problem (9.7) and an upper bound $\mathcal{U} \in \mathbb{R} \cup \{\infty\}$ for the optimal objective function value of Problem (9.1)

1: Solve Problem (9.7).
2: **if** Problem (9.7) is infeasible **then**
3: Fathom the current node, i.e., **stop** the node processing procedure and go back to the main method.
4: Let (x^k, y^k) denote the solution to Problem (9.7).
5: **if** $c_x^\top x^k + c_y^\top y^k \geq \mathcal{U}$ **then**
6: Fathom the current node, i.e., **stop** the node processing procedure and go back to the main method.
7: **if** $(x^k, y^k) \notin X \times Y$ **then**
8: Either generate valid inequalities for $\bar{\Omega}_k \cap (X \times Y)$ that separates (x^k, y^k), augment $\bar{\Omega}_k$, and go to Line 1, or branch on an integer variable with a fractional value to create two new sub-problems, **stop** the node processing procedure, and go back to the main method.
9: Solve the x^k-parameterized lower-level problem, i.e., compute

$$\varphi(x^k) = \min_{y \in Y} \left\{ d^\top y : Dy \geq b - Cx^k \right\}.$$

10: **if** $d^\top y^k \leq \varphi(x^k)$ **then**
11: Update the incumbent, i.e., update the incumbent solution with (x^k, y^k) and set the incumbent value to $\mathcal{U} \leftarrow c_x^\top x^k + c_y^\top y^k$.
12: Fathom the current node, i.e., **stop** the node processing procedure and go back to the main method.
13: Solve the x_L^k-parameterized mixed-integer linear refinement problem

$$\begin{aligned} \min_{x \in X, y \in Y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \; Cx + Dy \geq b, \\ & x_i = x_i^k \quad \text{for all } i \in L, \\ & d^\top y \leq \varphi(x^k). \end{aligned} \tag{9.8}$$

14: **if** Problem (9.8) is feasible **then**
15: Let $(\tilde{x}, \tilde{y})$ denote an optimal solution to Problem (9.8).
16: **if** $c_x^\top \tilde{x} + c_y^\top \tilde{y} < \mathcal{U}$ **then**
17: Update the incumbent, i.e., update the incumbent solution with $(\tilde{x}, \tilde{y})$ and set the incumbent value to $\mathcal{U} \leftarrow c_x^\top \tilde{x} + c_y^\top \tilde{y}$.
18: **if** all linking variables x_i^k, $i \in L$, are fixed **then**
19: Fathom the current node, i.e., **stop** the node processing procedure and go back to the main method.
20: Generate a valid inequality that separates (x^k, y^k) from $\bar{\Omega}_k$, augment $\bar{\Omega}_k$, and go to Line 1.

for integer and bilevel feasibility, i.e., we check whether Conditions 2 and 3 from Section 9.1 are satisfied. If the point (x^k, y^k) violates (some of) the integrality constraints, i.e., $(x^k, y^k) \notin X \times Y$, we can branch on fractional values of integer variables as usual. Alternatively, we can also generate an inequality that separates $(x^k, y^k) \in \bar{\Omega} \setminus \Omega$ from $\mathcal{F} \subseteq \Omega$, augment the set $\bar{\Omega}_k$, and re-solve Problem (9.7); see Line 8. If the point (x^k, y^k) satisfies all integrality constraints, we proceed by checking for bilevel feasibility. To this end, we solve the x^k-parameterized lower-level problem and determine whether y^k is an optimal response to the leader's decision, i.e., $d^\top y^k \leq \varphi(x^k)$. If the point (x^k, y^k) is bilevel feasible, we can update the incumbent and fathom the current node; see Lines 10–12. Otherwise, we solve the refinement problem (Line 13) to obtain a bilevel-feasible point $(\tilde{x}, \tilde{y})$, which may be used to update the incumbent. In contrast to the node processing procedure in Algorithm 6, we perform a refinement step regardless of whether the linking variables have already been fixed by branching or not. This can help improve the upper bound more quickly, potentially leading to more nodes being fathomed due to bounding; see also Exercise 8.19. As before, if all linking variables have been fixed, we can fathom the current node; see Line 19. If this is not the case, we generate a valid inequality that separates the bilevel-infeasible point (x^k, y^k), augment the set $\bar{\Omega}_k$, and re-solve Problem (9.7); see Line 20. For this purpose, we can, e.g., use the valid inequalities obtained from Proposition 9.7 or Proposition 9.9. Note that the separation procedure in Line 20 is the main difference between the branch-and-cut method discussed here and the branch-and-bound method from Section 8.3. Nevertheless, we emphasize that the node processing scheme presented above represents only one possible way to design a branch-and-cut method for mixed-integer linear bilevel problems, which still has several degrees of freedom. Let us mention five of them.

(i) How do we choose which node to process next? Again, possible options include a depth-first search (DFS), a breadth-first search (BFS), variants of the two, or even completely different strategies; see, e.g., Section 3.1 in Belotti et al. (2013).

(ii) When should we branch on fractional values of integer variables and when should we add a cut to separate integer-infeasible points in Line 8 of Algorithm 7? In particular, it is also possible to use only branching or only cuts to separate integer-infeasible points without affecting the correctness of the overall branch-and-cut method. More details on this aspect can be found in Section 2.3 in Tahernejad et al. (2020).

(iii) In contrast to the node processing scheme in Algorithm 6, we do not branch on linking variables anymore when we encounter an integer-feasible but bilevel-infeasible point in Algorithm 7. Instead, we separate such a point by generating and adding valid inequalities. We make this choice mainly for the ease of presentation. Of course, one could still branch on linking variables as it is done in Algorithm 6. Moreover, it is possible to combine both strategies; see, e.g., Tahernejad et al. (2020) again for further details.

(iv) What type of cuts should we use for separating integer- or bilevel-infeasible points? We refer to Clautiaux and Ljubić (2025) and Cornúejols (2008) as well as Tahernejad and Ralphs (2025) for overviews of classes of valid inequalities that can be used to separate integer- or bilevel-infeasible points, respectively.

(v) In Algorithm 7, cuts for bilevel-infeasible points are only generated from integer-feasible points. However, integer feasibility is not required to generate valid inequalities, i.e., cuts can also be derived from fractional points. Further discussions can be found in, e.g., Fischetti et al. (2017) and Tahernejad et al. (2020).

In addition, let us point out that Tahernejad et al. (2020) propose more sophisticated conditions for deciding when or when not to solve the lower-level problem (Line 9) and the refinement problem (Line 13). Overall, these choices can significantly impact the performance of the branch-and-cut method and the most suitable ones often depend on the specific bilevel problem under consideration. To make this more concrete, let us point out one particular case in which the refinement problem does not need to be solved at all.

Remark 9.10 If the underlying bilevel MILP does not contain coupling constraints, we can simplify the node processing procedure in Algorithm 7 by omitting the solution of the refinement problem (9.8). In this case, any optimal follower's response to a given leader's decision x^k, say $\tilde{y}$, computed in Line 9 already yields a bilevel-feasible point $(x^k, \tilde{y})$. Although the follower's response may not be unique and an optimistic response may not be returned in Line 9, the point $(\tilde{x}, \tilde{y}) = (x^k, \tilde{y})$ can still be used to potentially update the incumbent in Line 17.

Theorem 9.11 *Suppose that Assumptions 6.5 and 6.13 hold. Assume further that each valid inequality added in Line 20 of Algorithm 7 excludes at least one assignment for the linking variables. Then, if we embed Algorithm 7 into the branch-and-bound framework of Algorithm 5, we obtain a method that terminates with an optimal solution to Problem (9.1) or with the correct indication of infeasibility after a finite number of visited nodes and after adding an overall finite number of cuts.*

Proof: By Theorem 7.3, it only remains to show that the number of added cuts is finite and that a separated pair (x, y) cannot occur again in later iterations. Finiteness follows from Assumption 6.5, which ensures that there is only a finite number of bounded integer variables to branch on. This also implies that only finitely many valid inequalities that cut off integer-feasible but bilevel-infeasible points can be derived in Line 20 of Algorithm 7. Furthermore, by Definition 9.1, a point that has been separated by a cut cannot re-enter the feasible set of any problem in the sub-tree rooted at the node at which the cut has been introduced. Finally, as before, at the end of the algorithm, $\mathcal{U} = \infty$ indicates that the overall problem is infeasible. □

Note that the cuts obtained from Proposition 9.9 exclude assignments for the leader's variables and, thus, meet the requirement of Theorem 9.11. In contrast, the cuts from Proposition 9.7 only separate a single integer-feasible but bilevel-infeasible point. In particular, they do not cut off any other points in Ω, i.e., they may not exclude an assignment for the leader's linking variables directly. Nevertheless, if $X = \mathbb{Z}^{n_x}$, $Y = \mathbb{Z}^{n_y}$, and Assumption 6.5 hold, successively generating valid inequalities using Proposition 9.7 eventually excludes assignments for the leader's variables as well. Hence, finite termination can still be guaranteed using the cuts from Proposition 9.7 in this case.

Let us also note that a global lower bound can be maintained in the algorithm as it is the case for the branch-and-bound method in Section 5.2; see Remark 5.5 for the details.

Example 9.12 (Example 6.2—Revisited) We revisit the bilevel problem in Example 6.2, i.e., we consider the problem

$$\begin{aligned} \min_{x \in \mathbb{Z}, y} \quad & -x - 10y \\ \text{s.t.} \quad & y \in \mathcal{S}(x), \end{aligned}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to

$$\begin{aligned} \min_{y \in \mathbb{Z}} \quad & y \\ \text{s.t.} \quad & -25x + 20y \le 30, \\ & x + 2y \le 10, \\ & 2x - y \le 15, \\ & 2x + 10y \ge 15. \end{aligned}$$

In Example 8.15, we have already seen a branch-and-bound search tree for this problem; see Figure 8.4. We now apply a depth-first search branch-and-cut method in which we use the node processing scheme in Algorithm 7 and

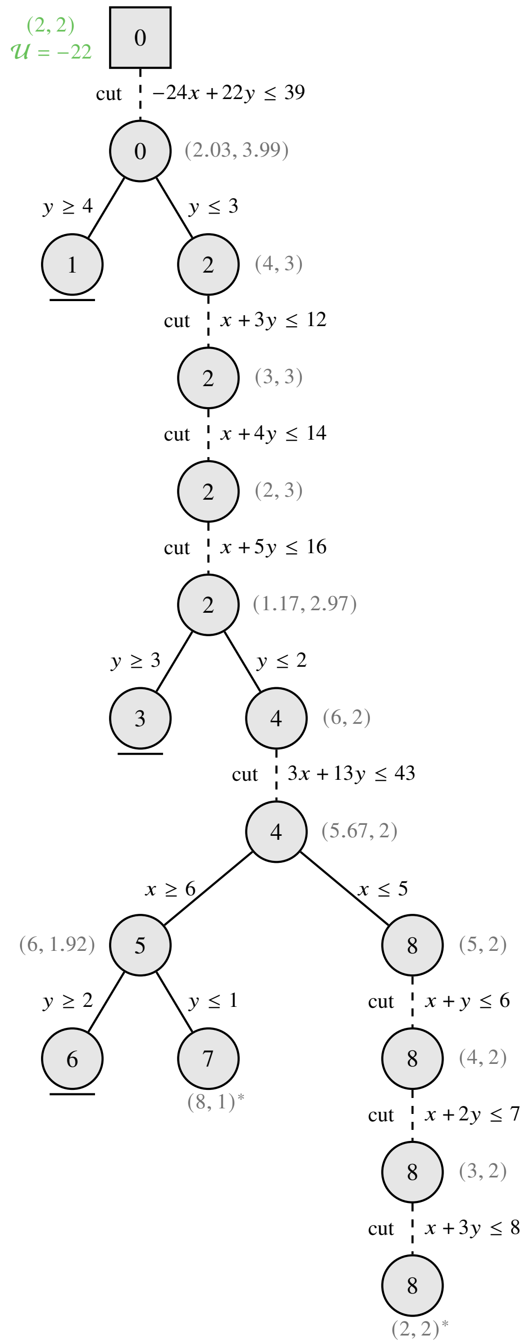


Figure 9.4 The branch-and-cut search tree in Example 9.12. Tuples of the form (x, y) at the nodes denote the solutions to the problems considered at these nodes. Nodes whose solutions are marked with an asterisk are pruned due to bounding. At the rectangular node, the incumbent is updated.

the cuts from Proposition 9.7 to separate bilevel-infeasible points. A possible branch-and-cut search tree is depicted in Figure 9.4.

The solution to the root-node problem is given by $(x, y) = (2, 4)$, which is integer but not bilevel feasible because the optimal objective function value of the x-parameterized lower-level problem is $2 < 4$. Solving the refinement problem (9.8) then yields the bilevel-feasible point $(\tilde{x}, \tilde{y}) = (x, \tilde{y}) = (2, 2)$ with an objective function value of -22. Because this value is better than the current incumbent value, which has been set to $+\infty$ initially, we update it by setting $\mathcal{U} \leftarrow -22$. In Figure 9.4, nodes at which the incumbent is updated are shown as rectangular nodes. Next, we generate a valid inequality to separate the bilevel-infeasible point $(x, y) = (2, 4)$. To this end, we apply Proposition 9.7 and follow the derivations in Example 9.8. This yields the inequality

$$-24x + 22y \leq 39,$$

which we add to the root-node problem. Re-solving this problem then yields the fractional point $(x, y) \approx (2.03, 3.99)$, which we now handle by branching on the follower's variable y. To this end, we generate two new sub-problems in which we impose either $y \geq 4$ or $y \leq 3$. Note, however, that we can also branch on the leader's variable x or add a valid inequality to separate $(x, y) \approx (2.03, 3.99)$. At node 1 of the branch-and-cut search tree, we consider the problem in which we add the constraint $y \geq 4$. This problem is infeasible and, thus, we prune the current node. Backtracking then leads to node 2 at which we consider the problem depicted in Figure 9.5.[1] The optimal solution to this problem is given by $(x, y) = (4, 3)$, which is integer but not bilevel feasible. Solving the refinement problem (9.8) then yields the bilevel-feasible point $(\tilde{x}, \tilde{y}) = (x, \tilde{y}) = (4, 1)$ with an objective function value of $-14 > -22 = \mathcal{U}$, i.e., we do not update the incumbent. Instead, we now generate a valid inequality that separates $(x, y) = (4, 3)$ using Proposition 9.7. For this purpose, we first need to determine the inequalities that are binding in $(4, 3)$. By Figure 9.5, we observe that this is the case for

$$x + 2y \leq 10 \quad \text{and} \quad y \leq 3.$$

Whereas the first inequality is contained in the set $\bar{\Omega}$, the latter one is imposed by branching. Nevertheless, we can still apply Proposition 9.7 if we include the branching restriction $y \leq 3$ in the linear system described by $Ax + By \geq a$ and $Cx + Dy \geq b$. Applying Proposition 9.7 to such an augmented linear

[1] To simplify the presentation, we do not show the valid inequality generated at node 0 (again). Nevertheless, you can verify for yourself that it is redundant for the problem considered at node 2.

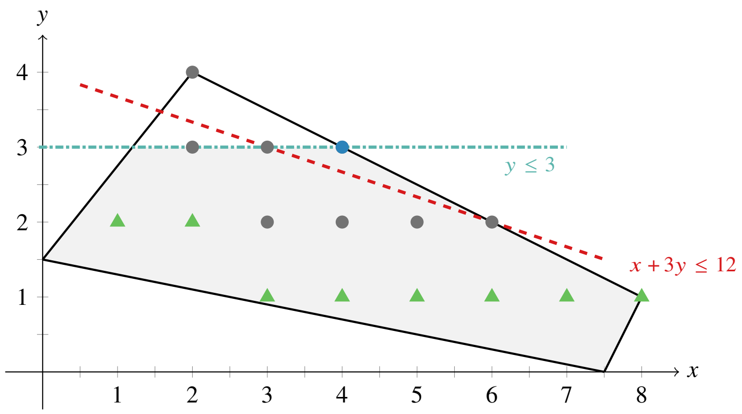


Figure 9.5 The problem at node 2 of the branch-and-cut search tree in Example 9.12. The branching restriction imposed at node 2 is illustrated by the turquoise dash-dotted line. The red dashed line corresponds to the inequality $x + 3y \leq 12$ that is generated using Proposition 9.7.

inequality system then yields the locally valid cut

$$x + 3y \leq 12,$$

which is valid for the set

$$\mathcal{F} \cap \{(x, y) \colon -24x + 22y \leq 39, \ y \leq 3\};$$

see Figure 9.5. Adding the inequality $x + 3y \leq 12$ to the problem considered at node 2 of the branch-and-cut search tree and re-solving the problem then yields the point $(x, y) = (3, 3)$. This point is integer but not bilevel feasible. Solving the refinement problem now yields $(\tilde{x}, \tilde{y}) = (x, \tilde{y}) = (3, 1)$ with an objective function value of -13, which is again worse than the current incumbent value. Hence, we generate a valid inequality that separates the point $(x, y) = (3, 3)$. Note that the inequalities

$$x + 3y \leq 12 \quad \text{and} \quad y \leq 3$$

are binding in $(3, 3)$, where the first inequality corresponds to the cut added previously and the second one is imposed by branching. Applying Proposition 9.7 again then yields the locally valid cut

$$x + 4y \leq 14.$$

Re-solving the problem at node 2 of the branch-and-cut search tree yields the point $(x, y) = (2, 3)$, which is again integer but not bilevel feasible. We now

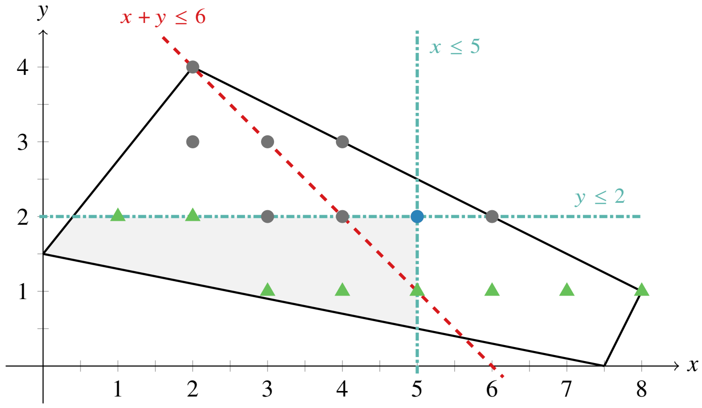


Figure 9.6 The problem at node 8 of the branch-and-cut search tree in Example 9.12. Branching restrictions imposed at node 8 are illustrated by the turquoise dash-dotted lines. The red dashed line corresponds to the locally valid inequality $x + y \leq 6$ that is generated using Proposition 9.7.

solve the refinement problem and obtain $(\tilde{x}, \tilde{y}) = (x, \tilde{y}) = (2, 2)$, which does not produce a better incumbent. Hence, we apply Proposition 9.7 again to separate the point $(x, y) = (2, 3)$. We then re-solve the problem considered at node 2 of the branch-and-cut search tree and obtain the fractional point $(x, y) \approx (1.17, 2.97)$. We now branch on the variable $y \approx 2.97$, i.e., we generate two new subproblems in which we add either the constraint $y \geq 3$ or $y \leq 2$. Fathoming due to infeasibility and backtracking then leads to node 4, which has the solution $(x, y) = (6, 2)$. Because this point is integer but not bilevel feasible, we again solve the refinement problem, which does not lead to an incumbent update, and then generate a valid inequality using Proposition 9.7 to separate this point. Afterward, we obtain $(x, y) \approx (5.67, 2)$. Because $x \approx 5.67$ is fractional, we branch on the leader's variable next. Further branching, fathoming due to infeasibility, and backtracking then leads to node 7, which has the solution $(x, y) = (8, 1)$ with a value of $-18 > \mathcal{U}$. No further improvements can be achieved by branching or adding cuts and, thus, the node can be pruned due to bounding. Backtracking then leads to node 8 at which we consider the problem depicted in Figure 9.6.[2] The solution to this problem is given by $(x, y) = (5, 2)$, which is integer feasible but bilevel infeasible. We solve the refinement problem, which does not lead to an incumbent update, and generate a valid inequality to separate the point $(x, y) = (5, 2)$. To apply Proposition 9.7, let us determine the

[2] Again, to simplify the presentation, we do not plot the valid inequalities that have been generated earlier in the search tree as they are redundant for the problem considered at node 8.

constraints that are binding in $(5, 2)$. By Figure 9.6, we observe that this is the case for

$$x \le 5 \quad \text{and} \quad y \le 2,$$

which are both imposed by branching. Proposition 9.7 thus yields the locally valid cut

$$x + y \le 6,$$

which corresponds to the red dashed line shown in Figure 9.6. Generating and adding further cuts finally leads us to revisit the point $(x, y) = (2, 2)$ with an objective function value of $-22 \ge \mathcal{U}$. No further improvements can be achieved by branching or adding cuts and, thus, the node can be pruned due to bounding. Because there are no more open nodes, the method terminates with the optimal solution $(x^*, y^*) = (2, 2)$ after investigating 9 nodes and adding 8 cuts. Compared to the branch-and-bound search tree in Figure 8.4, the branch-and-cut method investigates 12 nodes less. Although we have to re-solve some of the problems at nodes of the search tree multiple times, this is typically less costly because information from previous solutions can be exploited to speed up the process. △

Remark 9.13 In the branch-and-cut search tree in Example 9.12, we have seen both globally and locally valid cuts:

(i) At the root node (node 0), we generated the inequality $-24x + 22y \le 39$. Because this inequality has been obtained from Proposition 9.7 using only the information contained in the set $\bar{\Omega}$, it is globally valid for the bilevel-feasible set $\mathcal{F}$; see Figure 9.3.
(ii) The inequality $x + 3y \le 12$ generated at node 2 of the branch-and-cut search tree has been obtained by exploiting node-specific information (the branching restriction $y \le 3$). Nevertheless, it can be seen from Figure 9.5 that the inequality is valid for the bilevel-feasible set $\mathcal{F}$. Hence, it is globally valid.
(iii) The inequality $x + y \le 6$ generated at node 8 has been obtained from branching restrictions specific to that node. In particular, this inequality cuts off the bilevel-feasible points $(6, 1)$, $(7, 1)$, and $(8, 1)$. Hence, it is only locally valid (at node 8 and its descendants).

Exercise 9.14 (Example 6.1—Revisited) We consider the bilevel problem

$$\min_{x \in \mathbb{Z}, y} \quad x + 2y \quad \text{s.t.} \quad x \ge 0, \ y \in \mathcal{S}(x),$$

where $\mathcal{S}(x)$ is the set of optimal solutions to

$$\begin{aligned} \max_{y \in \mathbb{Z}} \quad & f(x,y) = y \\ \text{s.t.} \quad & -4x + 10y \le 15, \\ & 4x + 10y \ge 15, \\ & 10x + 4y \le 35, \\ & y \ge 0. \end{aligned}$$

Note that this problem is equivalent to the one in Example 6.1. We simply multiplied the lower-level constraints (except for the nonnegativity constraint) by 4 so that all problem data is integer.

(i) Solve the problem by hand by applying a depth-first search branch-and-cut method with the node processing scheme in Algorithm 7. Use a general-purpose MILP solver to solve the arising sub-problems and visualize your progress in a search tree. Handle fractional points by branching and generate valid inequalities to separate bilevel-infeasible points using Proposition 9.7.

(ii) Compare your search tree from (i) with the one of the branch-and-bound method in Example 8.16; see Figure 8.6. What differences do you observe between the branch-and-bound and the branch-and-cut methods applied to the bilevel problem in Example 6.1?

Exercise 9.15 (Exercise 8.18—Revisited) Consider the bilevel knapsack interdiction problem in Exercise 8.18 again, i.e., consider the problem

$$\begin{aligned} \min_{x,y} \quad & 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 2x_1 + x_2 + x_3 \le 2, \\ & x \in \{0,1\}^3, \ y \in \mathcal{S}(x), \end{aligned}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to

$$\begin{aligned} \max_{y} \quad & 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 4y_1 + 3y_2 + 2y_3 \le 4, \\ & y_i \le 1 - x_i \quad \text{for all } i = 1,2,3, \\ & y \in \{0,1\}^3. \end{aligned}$$

(i) Solve this problem by hand by applying a depth-first search branch-and-cut method with the node processing scheme in Algorithm 7. Use a general-purpose MILP solver to solve the arising sub-problems and visualize your progress in a search tree. Handle fractional points by branching and generate valid inequalities to separate bilevel-infeasible points. To this end, apply the branch-and-cut method twice and do the following.

(a) First, use Proposition 9.7 for generating valid inequalities to separate bilevel-infeasible points.

(b) Second, use Proposition 9.9 for generating valid inequalities to separate bilevel-infeasible points.

You can use your results from Exercise 8.18 to verify your solution.

(ii) Compare the two search trees obtained in (i). What do you observe? Can you explain your observations?

(iii) What differences do you observe between the branch-and-bound (Exercise 8.18) and the branch-and-cut methods applied to the example?

9.3 Excursus: Bilevel Solvers You Can Try

The development of general-purpose solvers for bilevel MILPs is still in its early stages. Nevertheless, if you are interested in solving mixed-integer linear bilevel problems yourself, there are currently two solvers that are available for you to try. Both solvers build on the branch-and-cut framework introduced in this chapter but they also incorporate techniques that we explore in the upcoming chapters 10–12.

The first one is the open-source solver `MibS`, which is freely available; see DeNegre et al. (2026). The method comes with excellent documentation, which makes it particularly user-friendly. For the mathematical details, we refer to DeNegre and Ralphs (2009), Tahernejad and Ralphs (2025), and Tahernejad et al. (2020).

The second solver is the branch-and-cut method by Fischetti et al. (2026), which is available as a pre-compiled binary and which can be used after requesting a license file from the authors. For the mathematical details, we refer to Fischetti et al. (2017, 2018a, 2019).

A large and well-curated set of test instances for (mixed-)integer bilevel problems is the `BOBILib`, which can also be used to test the mentioned two solvers. It is available online at `bobilib.org`. The details of this library are all given in the corresponding paper by Thürauf et al. (2026). The `BOBILib` also defines a standard for encoding mixed-integer linear bilevel problems that is an extension of the classic `LP` or `MPS` formats in single-level mixed-integer linear optimization. All the details can be found in Section 4 of the paper by Thürauf et al. (2026).

9.4 What You Should Know Now!

1. What is a drawback of the branch-and-bound method for bilevel MILPs that motivated us to study branch-and-cut?
2. How is a valid inequality defined in general?
3. What is the difference between a globally valid and a locally valid inequality?
4. Can you illustrate a valid inequality using an example?
5. What is the separation problem?
6. Can you state a valid inequality that can be used to separate a bilevel-infeasible point?
7. What do you know about valid inequalities for the special case of bilevel MILPs with binary upper-level variables?
8. How does the branch-and-cut method for bilevel MILPs work?
9. How do we process nodes in the branch-and-cut search tree?
10. How do we handle integer-infeasible points?
11. How do we handle bilevel-infeasible points?
12. Can you explain why we first check for integer feasibility and, afterward, we check for bilevel feasibility? What would happen if we swap the order?
13. What can you say about finite termination of the branch-and-cut method?
14. Can you prove the correctness of the node processing procedure?
15. What solvers for bilevel MILPs do you know?

10

Interdiction Problems

In this chapter, we address interdiction problems, a special class of bilevel problems whose aims include monitoring or halting an adversary's activity in a given environment or identifying the most vulnerable parts of a system with respect to potential attacks or failures. Typically, interdiction problems model *defender-attacker* settings in which the follower acts as the attacker and the leader attempts to protect or defend a system by anticipating the attacker's malicious activities. Many interdiction problems are applied in network environments in which the leader has limited resources to protect this environment, e.g., by disabling some vertices or edges in a network or by changing their capacity, so as to achieve the worst possible outcome for the follower. In some of the most prominent examples of *network interdiction*, the follower searches for a shortest path or a maximum flow in a given network (Israeli and Wood 2002). Overall, interdiction problems have applications in numerous sectors such as security and defense (see Examples 1.5 and 1.19), public health (where interdiction problems can be used to control the spread of infectious diseases), social networks (where they can be used to control the spread of fake news), or general communication networks. For further details, we refer to the surveys in Beck et al. (2023), Kleinert et al. (2021b), and Smith and Song (2020).

In contrast to general bilevel problems, in which the leader and the follower may cooperate and have similar objectives, interdiction problems model a competitive setting in which the objective functions of the two players are diametrically opposed, i.e., it holds

$$F(x, y) = -f(x, y).$$

Here, the objective functions of the leader and the follower are again given by F and f, respectively. More specifically, we can say that an interdiction problem can be formulated as a special case of the min-max problem

$$\begin{aligned} \min_{x \in X} \quad & \varphi(x) \\ \text{s.t.} \quad & Ax \geq a, \end{aligned}$$

where the follower's problem is given by

$$\varphi(x) = \max_{y \in Y} \{f(x, y) \colon g(x, y) \geq 0\}. \tag{10.1}$$

The constraints $Ax \geq a$ and $x \in X$ determine the search space of the leader and often include budget or resource constraints. In particular, we do not consider coupling constraints in this chapter. Moreover, f and g have a specific structure that we discuss below.

The general idea of an interdiction problem is that the leader prevents certain activities of the follower by reducing the availability of some objects or resources such as vertices or edges in a network. This is reflected by either modifying the feasible space of the follower through the constraints $g(x, y) \geq 0$, which link the actions of the leader to those of the follower, or by worsening the objective function value $f(x, y)$ due to the leader's interdiction decisions. Before we discuss the specific nature of these interactions, we point out that, due to the min-max objective, the follower always selects the worst possible outcome for the leader. As a result, there is no need to distinguish between the optimistic and the pessimistic solution concept (see Chapter 2) as the two cases are equivalent in this setting.

Throughout this chapter, we again impose that Assumption 6.5 is satisfied, i.e., the shared constraint set is non-empty and bounded. Moreover, as usual for interdiction problems, we restrict the upper-level variables to be in $[0, 1]$ and we assume that the available budget of the leader (included in the constraints $Ax \geq a$) is limited so that the follower's problem stays feasible. More formally, we impose the following.

Assumption 10.1 The shared constraint set Ω is non-empty and bounded, the set of upper-level feasible decisions satisfies $\{x \in X \colon Ax \geq a\} \subseteq [0, 1]^{n_x}$, and the lower-level problem is feasible for all x with $x \in X$ and $Ax \geq a$.

Note that, if the leader's budget is sufficiently large to render the lower-level problem infeasible, there exists a feasible decision x for the leader such that $\varphi(x) = -\infty$. Consequently, the overall interdiction problem becomes unbounded. In practice, however, the interdiction budget is typically limited, allowing the follower to always identify a feasible solution regardless of the leader's interdiction decision. In this case, the leader aims to determine an interdiction strategy that results in the worst possible outcome for the follower.

We again use $L \subseteq \{1, \dots, n_x\}$ to denote the index set of the linking variables. So far in this book, linking variables have only appeared in the constraints of the lower-level problem. For interdiction problems, this is not the only option anymore because x variables can now also appear in the objective function of the follower. We make the following assumption.

Assumption 10.2 There is a bijection between the index set of the linking variables L and the index set of the lower-level variables $\{1, \dots, n_y\}$.

Assumption 10.2 particularly implies $n_y = |L|$. We make this assumption for the ease of presentation. However, we show in Section 10.6 how we can handle extended formulations of the lower-level problem, i.e., formulations with additional lower-level variables that are not directly linked to the leader's decision x.

Remark 10.3 (Complexity) In general, interdiction problems are as hard as general bilevel problems, i.e., the result of Jeroslow (1985) holds; see Section 6.4. For example, as for general bilevel MILPs, if the follower solves an NP-hard problem, e.g., the knapsack problem or the maximum-clique problem, very often the corresponding interdiction problem turns out to be Σ_2^p-hard (Caprara et al. 2014; Rutenburg 1994). For more details, see also Grüne and Wulf (2025).

Interdiction problems have a very specific structure. This structure can be exploited to develop tailored solution methods that are usually more effective than the generic ones from the last chapters. This chapter provides an overview of the most commonly used algorithmic techniques for solving interdiction problems. To this end, we begin by discussing various types of interdiction problems in Section 10.1. Definitions and models for several frequently studied cases are presented in Section 10.2 and exact solution methods are discussed in Sections 10.3–10.5. Section 10.6 focuses on interdiction problems characterized by a specific property in the lower-level problem called *downward monotonicity*. Closely related to interdiction problems are so-called *blocker problems*, which we discuss in Section 10.7. Finally, Section 10.8 concludes the chapter with additional examples of critical vertex and edge detection problems.

10.1 Different Types of Interdiction Problems

We first distinguish between discrete and continuous interdiction depending on whether the leader's linking variables are discrete or continuous. Moreover, based on the functions f and g, we make a distinction between interdiction actions that (i) affect the feasible region of the follower or (ii) modify the follower's objective function.

10.1.1 Discrete vs. Continuous Interdiction

In the discrete interdiction setting, the linking variables x_i, $i \in L$, are assumed to be binary and they are set to one if and only if the respective object i is interdicted, meaning, e.g., that it is made unavailable to the follower. In the continuous interdiction setting, the linking variables x_i are continuous, i.e., $0 \leq x_i \leq 1$ for all $i \in L$. As we will see in the following, these linking variables can either model a continuous decrease of profit if the objective function f is affected by x or a reduction of available capacities imposed on the interdicted objects if the constraint g is affected by x.

10.1.2 Interdiction Through Modification of the Follower's Feasible Region

We first provide a model in which the interactions between the leader and the follower are expressed through lower-level constraints. In this case, and in line with the focus of this book, the objective function $f(x, y)$ is linear and not affected by the leader's decisions, i.e., $f(x, y) = d^\top y$. Moreover, due to Assumption 10.1, the lower-level problem is always feasible and bounded, independent of the decision of the leader. Hence, there exist finite upper bounds $u_i \geq 0$, $i \in \{1, \dots, n_y\}$, that are valid for the lower-level variables y_i independent of whether object i is interdicted or not. In this type of interdiction problems, the only constraints linking x and y variables are the so-called *capacity reduction* or *interdiction constraints* (10.2c) given below. The lower-level problem is defined as

$$\varphi(x) := \max_{y \in Y} \quad d^\top y \tag{10.2a}$$

$$\text{s.t.} \quad Dy \leq b, \ y \geq 0, \tag{10.2b}$$

$$y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L. \tag{10.2c}$$

The constraints $Dy \leq b$, $y \geq 0$ impose restrictions on the follower's solution that are independent of the leader's decisions. In this context, the variables y may be continuous, discrete, or a combination of both. If present, integrality constraints are again imposed through the set Y. Likewise, the interdiction variables x can also be continuous or discrete. For example, in maximum-flow interdiction problems, the leader has a limited budget to reduce the capacity of certain arcs, whereas the follower seeks to maximize the flow in the resulting network (Lim and Smith 2007; Wood 1993); see Example 1.19. In this case, both the leader's and the follower's variables are continuous. In contrast, in the knapsack interdiction problem (see Example 1.18), the linking variables x_i, $i \in L$, are binary and for $x_i = 1$, the leader removes item i entirely from the set of items available to the follower. Here, both x and y are binary.

10.1.3 Interdiction Through Cost Modification

In Problem (10.2), the interaction between the leader and the follower is captured by the capacity reduction constraints (10.2c). Alternatively, the actions of the leader may worsen the cost or profit of the interdicted objects for the follower. The latter affects the objective function of the follower so that the nominal "profit" d_i (in case of maximization) is worsened by $\delta_i x_i$ for $i \in L$. Here, $\delta_i > 0$, $i \in L$, represents the maximum decrease of the profit of object i, which can be achieved if $x_i = 1$ holds. Hence, the function $f(x, y)$ is bilinear (in x and y) and the lower-level problem reads

$$\varphi(x) := \max_{y \in Y} \left\{ \sum_{i \in L} (d_i - \delta_i x_i) y_i : Dy \leq b, \ y \geq 0 \right\}. \tag{10.3}$$

Also in this case, there are no restrictions regarding the discrete or continuous nature of the variables x and y beyond Assumption 10.1. For example, if the follower solves the shortest-path problem, a discrete interdiction of an arc indexed by $i \in \{1, \dots, n_y\}$ implies that the length of this arc in the interdicted network can be either d_i if $x_i = 0$ or $d_i + \delta_i$ if $x_i = 1$. In the case of a continuous interdiction, the leader can choose to set the length of arc i to any value within the interval $[d_i, d_i + \delta_i]$.

10.2 Frequently Studied Interdiction Problems

We now give a few examples of interdiction problems that are frequently studied in the literature. These problems are typically named after the underlying lower-level problem. For instance, if we consider a shortest-path problem at the lower level, we obtain a shortest-path interdiction problem, whereas a knapsack problem at the lower level leads to a knapsack interdiction problem.

10.2.1 Shortest-Path Interdiction

We consider a simple and directed graph $G = (\mathcal{V}, \mathcal{A})$ with a set of vertices $\mathcal{V}$ and a set of arcs $\mathcal{A}$. With each arc $a \in \mathcal{A}$, we associate an arc length $d_a \geq 0$. The follower aims to find a shortest path between two given and distinct vertices s and t in G. The leader has a limited budget $\bar{b} > 0$ to prohibit (or interdict) the use of some arcs in $\mathcal{A}$ so that the shortest s-t path in the interdicted graph is as large as possible. As before, we now distinguish between continuous and discrete interdiction activities.

Continuous Shortest-Path Interdiction

In the continuous setting, the leader can increase the length of each arc $a \in \mathcal{A}$ by at most $\delta_a \geq 0$ and the continuous variable $0 \leq x_a \leq 1$ denotes the fraction of the length-increase for arc a. The follower solves the shortest-path problem in the graph with the modified length parameters. In the following model, the lower-level variable can be set to $y_a = 1$ if and only if arc a belongs to a shortest path. The continuous shortest-path interdiction problem thus reads

$$\max_{x} \quad \varphi(x) \tag{10.4a}$$

$$\text{s.t.} \quad \sum_{a \in \mathcal{A}} b_a x_a \leq \bar{b}, \tag{10.4b}$$

$$x \in [0, 1]^{|\mathcal{A}|}, \tag{10.4c}$$

where $\varphi(x)$ is the optimal-value function of the lower-level problem

$$\min_{y \geq 0} \quad \sum_{a \in \mathcal{A}} (d_a + \delta_a x_a)\, y_a \tag{10.5a}$$

$$\text{s.t.} \quad \sum_{a \in \delta^{\text{out}}(i)} y_a - \sum_{a \in \delta^{\text{in}}(i)} y_a = \begin{cases} 1, & i = s, \\ -1, & i = t, \\ 0, & \text{otherwise}, \end{cases} \quad \text{for all } i \in \mathcal{V}. \tag{10.5b}$$

The budget constraint for the leader is given by (10.4b). The objective function of the follower ensures that the length of each arc $a \in \mathcal{A}$ increases by $\delta_a x_a$, whereas the constraints in (10.5b) guarantee that there is one unit of flow sent from the source s to the target t in the graph G. Problem (10.4) is an example of a very rare and special case in which an interdiction problem can be solved efficiently. Indeed, Fulkerson and Harding (1977) show that the problem can equivalently be stated as a minimum-cost flow problem and it can thus be solved in polynomial time; see Exercise 10.10.

Example 10.4 (Continuous Shortest-Path Interdiction; see Smith and Song (2020)) Let us now consider the continuous shortest-path interdiction problem (10.4) for the network $G = (\mathcal{V}, \mathcal{A})$ depicted in Figure 10.1 (top). Here, the follower searches for a shortest path from $s = 1$ to $t = 6$. The interdiction costs are $b_a = 1$ for all arcs $a \in \mathcal{A}$ and the leader has a fixed interdiction

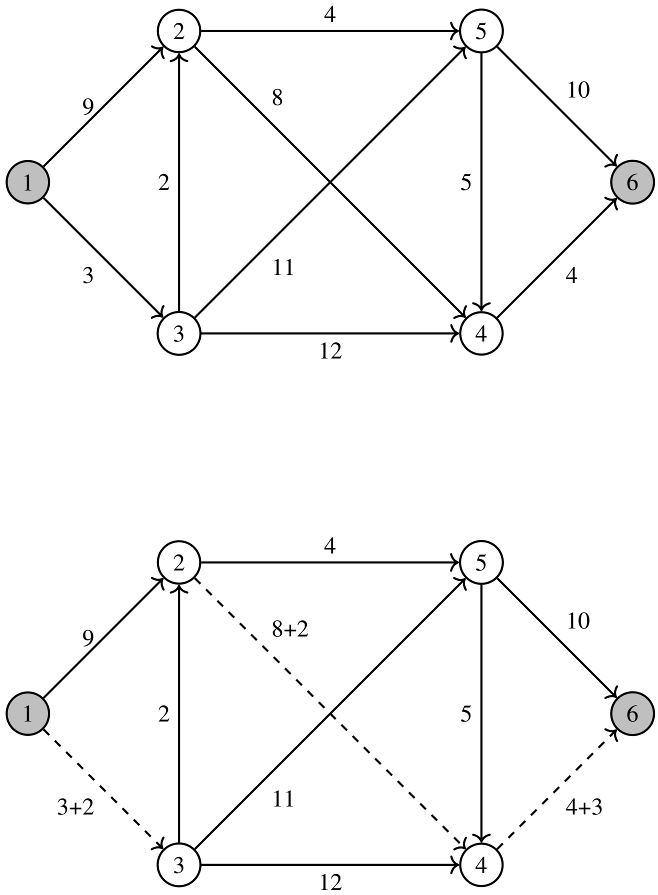


Figure 10.1 The continuous shortest-path interdiction problem in Example 10.4. Top: The graph with arc lengths shown next to each arc. Bottom: A feasible solution with dashed arcs being interdicted. Taken and modified from Smith and Song (2020).

budget of $\bar{b} = 3$. The arc lengths d_a are the numbers shown next to each arc in Figure 10.1. We assume that $\delta_{13} = \delta_{24} = 2$, $\delta_{46} = 3$, and $\delta_a = 0$ holds for all the remaining arcs a in the network. Without any interdiction, the shortest path chosen by the follower is $1 \to 3 \to 2 \to 4 \to 6$, which has a length of 17. A feasible interdiction decision of the leader is shown in Figure 10.1 (bottom). The leader interdicts the arcs in $\mathcal{A}_I = \{(1,3), (2,4), (4,6)\}$, i.e., $x_a = 1$ holds for all $a \in \mathcal{A}_I$, and the follower then chooses the shortest path $1 \to 3 \to 2 \to 5 \to 6$, which has a length of 21. △

Discrete Shortest-Path Interdiction

We now consider the situation in which the decisions of the leader are discrete, i.e., an arc $a \in \mathcal{A}$ is either completely removed from $\mathcal{A}$ if $x_a = 1$ or it can still be part of a shortest path if $x_a = 0$. Hence, the linking variables modify the feasible set of the follower. If $x_a = 0$ for some arc $a \in \mathcal{A}$, its length remains d_a. Hence, the discrete shortest-path interdiction problem is modeled as

$$\begin{aligned} \max_{x} \quad & \varphi(x) \\ \text{s.t.} \quad & \sum_{a \in \mathcal{A}} b_a x_a \leq \bar{b}, \\ & x \in \{0,1\}^{|\mathcal{A}|} \end{aligned} \tag{10.6}$$

with

$$\varphi(x) = \min_{y} \left\{ \sum_{a \in \mathcal{A}} d_a y_a : (10.5b),\ 0 \leq y_a \leq 1 - x_a,\ a \in \mathcal{A} \right\}.$$

This problem is indeed much more difficult to solve than its continuous counterpart from the last paragraph. Even if the budget constraint is a simple cardinality constraint, i.e., $\sum_{a \in \mathcal{A}} x_a \leq \bar{b}$, the problem is already NP-hard (Ball et al. 1989). The latter problem is frequently referred to as the *k-most-vital arcs problem* in the literature, where then $k = \bar{b}$ holds.

Example 10.5 (Discrete Shortest-Path Interdiction) We again consider the graph in Example 10.4 but we now solve the discrete shortest-path interdiction problem (10.6). The budget of the leader is now $\bar{b} = 1$ and all other parameters are chosen as in Example 10.4. Then, a feasible interdiction decision of the leader is to set $x_{13} = 1$, in which case the arc $(1, 3)$ is removed from G. The resulting shortest path for the follower is $1 \to 2 \to 4 \to 6$, which has a length of 21. △

Applications of the shortest-path interdiction problem are broad. When it comes to smuggling prevention, for example, security agencies are interested in interdicting smuggling operations that use shortest paths to cross territories. Solutions to the shortest-path interdiction problem may give insights into (i) which border crossing points should be monitored more heavily to ensure that smugglers must take significantly longer routes or (ii) where the patrols should be placed to cover the most likely smuggling paths. These applications align with the concept of continuous interdiction, based on the assumption that stronger enforcement can prevent more smuggling operations but not eliminate them entirely. Another interesting application can be found in cyber-network defense. In computer networks, data packets and potential cyber attacks often follow shortest paths. In this case, the shortest-path interdiction problem helps (i) to identify which network connections should be monitored to detect the most likely intrusions or (ii) to determine which firewalls would most effectively increase the “distance” (difficulty) for attackers. In the latter case, assuming that the installation of firewalls eliminates all (known) threats corresponds to the discrete shortest-path interdiction model. Finally, in urban transportation planning, where commuters typically take shortest paths through road networks, the shortest-path interdiction problem helps (i) to identify which roads would cause the most commute disruption if closed for construction, (ii) to determine which alternative routes to keep open during major events that close primary transport routes, or (iii) to find the critical intersections that, if congested, would increase the overall travel times the most. In these applications, closing roads corresponds to the discrete interdiction, whereas road congestion corresponds to the continuous interdiction model.

We discuss tailored solution methods to solve shortest-path interdiction problems as well as other interdiction problems with a continuous lower-level problem in Section 10.3.

10.2.2 Knapsack Interdiction

In the shortest-path interdiction problem, the lower-level problem is a linear optimization problem, which allows us to use LP duality theory to derive a single-level reformulation; see Chapter 3. One of the most studied (and structurally easiest) variants of an interdiction problem, in which the lower-level problem is discrete and NP-hard, is the knapsack interdiction problem. In this problem, the leader and the follower share a set of knapsack items. The leader intends to prohibit some items for the follower (respecting a given interdiction budget) so that the maximum profit that the follower can achieve by packing the remaining items is minimized. The problem, which is formally defined in Example 1.18, is Σ_2^p-complete as shown by Caprara et al. (2014). Because LP duality theory can no longer be applied to the lower-level problem, different algorithmic techniques have been proposed to reformulate and solve the knapsack interdiction problem; see, e.g., Caprara et al. (2016), Della Croce and Scatamacchia (2020), Fischetti et al. (2019), and Weninger and Fukasawa (2025). A branch-and-cut approach specifically tailored to solve the knapsack interdiction problem and related problems is discussed in Section 10.6.

10.2.3 Maximum-Clique Interdiction

Given an undirected and simple graph $G = (\mathcal{V}, \mathcal{E})$ with the set of vertices $\mathcal{V}$ and the set of edges $\mathcal{E}$, a *clique* K is a subset of vertices that induces a fully connected subgraph of G. A *maximal clique* K is an inclusion-wise maximal subset of vertices that induces a clique. The maximum-clique problem for a given graph G is to find a clique of maximum cardinality and it is a well-known NP-hard problem. The *clique number* of a graph G is the size of the maximum cliques of the graph and it is denoted by $\omega(G)$.

The maximum-clique interdiction problem is thus formally given as follows. Given an interdiction budget $\bar{b} > 0$ and interdiction cost $b_i \geq 0$ for removing vertex i from $\mathcal{V}$, the *maximum-clique interdiction problem* is to find a subset $S \subset \mathcal{V}$ to remove from G such that (i) the interdiction cost of vertices from S does not exceed $\bar{b}$ and (ii) the size of the maximum clique in the remaining graph $G[\mathcal{V} \setminus S]$, denoted as $\omega(G \setminus S)$, is minimized. Here, $G[\mathcal{V} \setminus S]$ refers to the subgraph of G that is induced by the vertices in $\mathcal{V} \setminus S$.

Example 10.6 (Maximum-Clique Interdiction) Figure 10.2 shows a graph as well as its maximum clique before and after removing a single vertex. The interdicted vertices are colored in black, whereas those belonging to an optimal response of the follower, i.e., vertices of a maximum clique, are shown in red. Before interdiction, $\omega(G) = 5$ holds. After removing a single vertex (v_8 in this case), the clique number of the interdicted graph reduces to 4.

For the same graph, assuming that the cost of removal are one for every vertex, Figure 10.3 shows optimal solutions for $\bar{b} = 2$ and $\bar{b} = 3$, respectively. In this example, increasing the budget by one, i.e., allowing the leader to remove two instead of just a single vertex, additionally reduces the clique number in the interdicted graph by one. Indeed, as we can see from Figure 10.3 (top), the optimal value of the maximum-clique interdiction problem becomes three, when increasing $\bar{b}$ from one to two. By allowing the leader to remove three vertices, i.e., setting $\bar{b} = 3$, the leader can, for example, choose to remove the vertices from $S = \{v_3, v_7, v_8\}$ so that it is no longer possible to find a clique larger than two in the resulting interdicted graph. This interdiction choice for the leader is depicted in Figure 10.3 (bottom). △

The maximum-clique interdiction problem can be modeled as a bilevel problem in which binary variables x_i at the upper level model the decision of the leader to remove vertex i from $\mathcal{V}$ and binary variables y_i at the lower level indicate if vertex i is part of a maximum clique of the follower. The model can

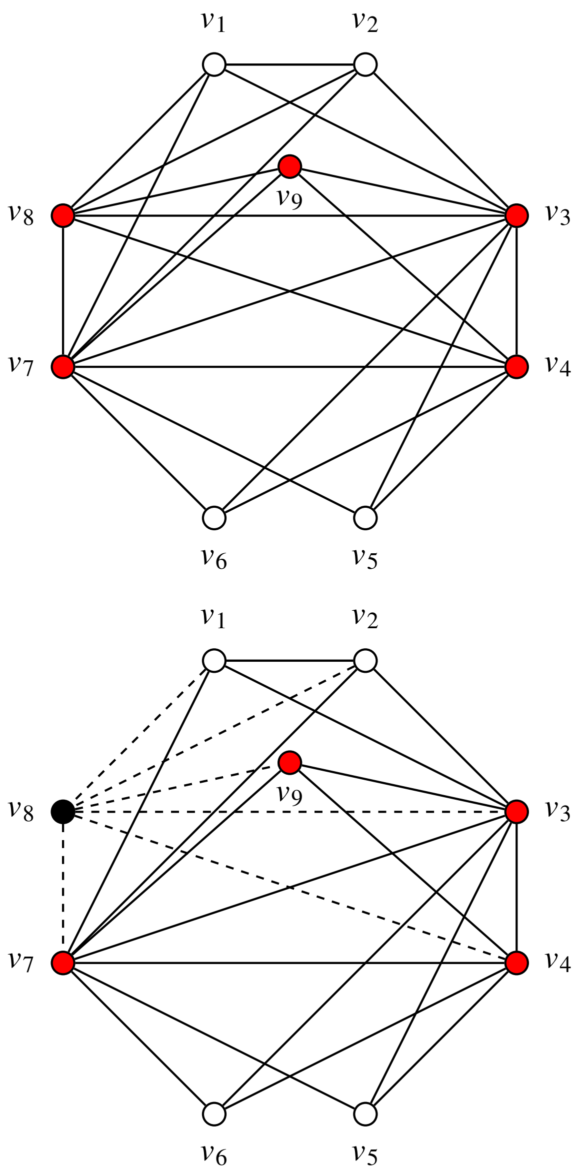


Figure 10.2 The maximum-clique interdiction problem in Example 10.6. Top: The given graph G with $\omega(G) = 5$ and a maximum clique $K_1 = \{v_3, v_4, v_7, v_8, v_9\}$. Bottom: After removing v_8, $\omega(G \setminus \{v_8\}) = 4$ holds and a maximum clique in the remaining graph is given by $K_2 = \{v_3, v_4, v_7, v_9\}$. Vertices of the respective maximum cliques are depicted in red.

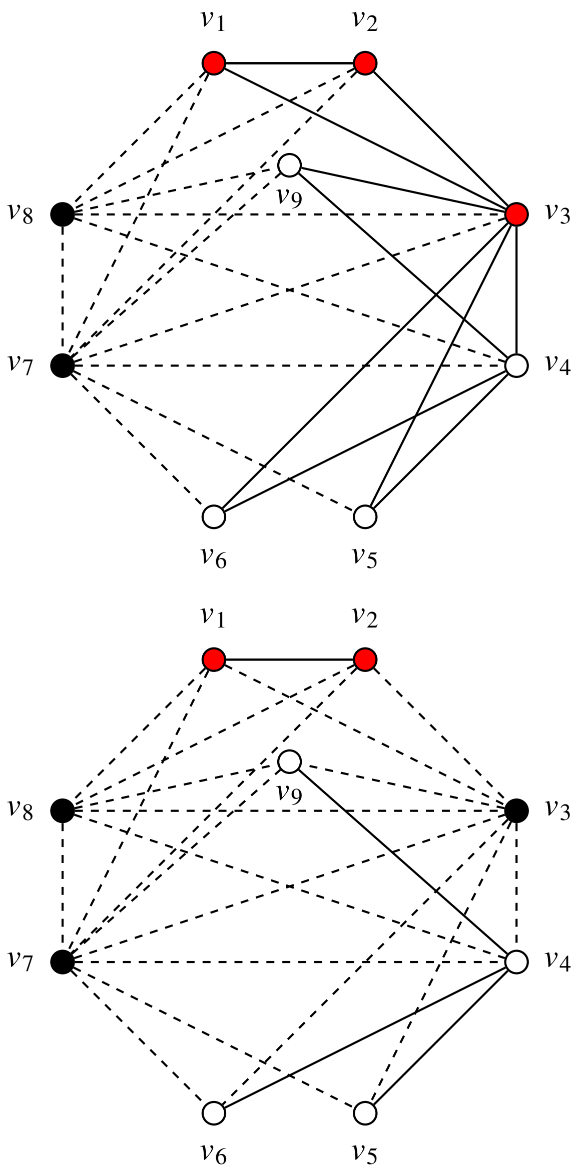


Figure 10.3 The maximum-clique interdiction problem in Example 10.6 revisited. For interdiction costs $b_i = 1$ for all $i \in \mathcal{V}$, the leader removes a set S of vertices from $\mathcal{V}$ such that $|S| = \bar{b}$. The vertices of a maximum clique in $G[\mathcal{V} \setminus S]$ are shown in red. Top: An optimal solution for $\bar{b} = 2$ is $S = \{v_7, v_8\}$, resulting in $\omega(G \setminus S) = 3$. Bottom: An optimal solution for $\bar{b} = 3$ is $S = \{v_3, v_7, v_8\}$, resulting in $\omega(G \setminus S) = 2$.

thus be written as

$$\begin{aligned} \min_{x} \quad & \varphi(x) \\ \text{s.t.} \quad & b^\top x \le \bar{b}, \\ & x \in \{0,1\}^{|\mathcal{V}|} \end{aligned}$$

with

$$\varphi(x) = \max_{y} \quad \sum_{i \in \mathcal{V}} y_i \tag{10.7a}$$

$$\text{s.t.} \quad y_i + y_j \le 1 \quad \text{for all } \{i, j\} \notin \mathcal{E}, \tag{10.7b}$$

$$y_i \le 1 - x_i \quad \text{for all } i \in \mathcal{V}, \tag{10.7c}$$

$$y \in \{0,1\}^{|\mathcal{V}|}. \tag{10.7d}$$

The objective function of the follower maximizes the size of the clique. Constraints (10.7b) ensure that no two distinct vertices i and j can be included in a solution unless there is an edge in $\mathcal{E}$ connecting them. Constraints (10.7c) are classic interdiction constraints, prohibiting the follower to use vertex i if it has been removed by the leader. The maximum-clique interdiction problem, which is Σ_2^p-complete (Grüne and Wulf 2025; Rutenburg 1994), is another example of an interdiction problem whose lower level is NP-hard and for which LP duality techniques cannot be applied. In Section 10.6, we illustrate how to solve this problem using a branch-and-cut procedure.

Remark 10.7 What we have seen so far is that there is not a single definition of an interdiction problem but that there exists an entire family of interdiction problems. The different aspects that we now already know and that can vary between different problems are the following:

(i) The interdiction decision of the leader can either be discrete, i.e., in $\{0,1\}^{n_x}$, or continuous, i.e., in $[0,1]^{n_x}$.
(ii) The linking variables can either modify the follower's feasible set or the objective function of the lower-level problem.
(iii) The follower's problem can be a discrete or a continuous optimization problem, which is dictated by the real-world problem that is modeled in the lower level.

These three aspects can, in general, be freely combined with each other, leading to an overall number of eight different types of interdiction problems.

10.3 Solution Approaches for Linear Lower-Level Problems

The particular interaction between the leader and the follower in interdiction problems can be exploited to develop problem-specific exact approaches that are computationally more efficient than the general methods used for generic (MI)LP bilevel problems discussed in the previous chapters. We now provide an overview of the most common techniques used in the literature and start with those that are applicable if the lower-level problem is linear.

10.3.1 Dualization

If the lower-level problem is a linear optimization problem, we can exploit LP duality theory to derive a single-level reformulation; see Chapter 3. Let us assume first that we have a lower-level problem of the form given in (10.2) in which the leader's decision x affects the feasible space of the follower. Hence, the lower-level problem is given by

$$\begin{aligned} \max_{y} \quad & d^\top y \\ \text{s.t.} \quad & Dy \leq b, \ y \geq 0, \\ & y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L. \end{aligned}$$

Recall that we impose Assumption 10.1 throughout this chapter. Let now λ and μ be the dual variables associated with the lower-level constraints $Dy \leq b$ and $y_i \leq u_i(1 - x_i)$, $i \in L$, respectively. The dual of the lower-level problem then reads

$$\begin{aligned} \min_{\lambda,\mu} \quad & b^\top \lambda + \mu^\top \operatorname{Diag}(u)(\mathbb{1} - x_L) \\ \text{s.t.} \quad & D^\top \lambda + \mu \geq d, \\ & \lambda \geq 0, \ \mu \geq 0. \end{aligned}$$

Here, x_L refers to the linking variables of x, indexed by the set L, and $\operatorname{Diag}(u)$ is the $L \times L$ diagonal matrix with diagonal entries u_i, $i \in L$. Moreover, we use $\mathbb{1}$ to denote a vector of ones of appropriate dimension, i.e., $\mathbb{1} = (1, \ldots, 1)^\top \in \mathbb{R}^{|L|}$. If we replace the primal lower-level problem with its dual, we obtain a min-min problem that does not contain any coupling constraints. In this situation, the two minimization problems can be merged and we finally obtain the equivalent single-level reformulation of the interdiction problem given by

$$\min_{x,\lambda,\mu} \quad b^\top \lambda + \mu^\top \mathrm{Diag}(u)(\mathbb{1} - x_L) \tag{10.8a}$$

$$\text{s.t.} \quad Ax \geq a, \; x \in X, \tag{10.8b}$$

$$D^\top \lambda + \mu \geq d, \tag{10.8c}$$

$$\lambda \geq 0, \; \mu \geq 0, \tag{10.8d}$$

which has bilinear terms in the objective function. The second term in (10.8a) is equal to

$$\sum_{i \in L} \mu_i u_i (1 - x_i),$$

which can be re-written as

$$\mu^\top u - \sum_{i \in L} \mu_i u_i x_i.$$

If the leader's interdiction variables are binary, the bilinear terms $\mu_i x_i$ are usually linearized so that one obtains a single-level mixed-integer linear reformulation. The most classic technique to reformulate these bilinearities is based on McCormick's inequalities (McCormick 1976); see Remark 3.2. To use these inequalities, let $M_i \geq 0$ be a valid upper bound on μ_i for all $i \in L$. Here and in what follows, "valid" means that at least one dual optimal solution for μ is not cut off by the chosen upper bound. One can then turn Problem (10.8) into an MILP by introducing new continuous variables ρ_i and by substituting the products $\mu_i x_i$ with ρ_i for all $i \in L$. This yields

$$\min_{x,\lambda,\mu,\rho} \quad b^\top \lambda + u^\top \mu - u^\top \rho \tag{10.9a}$$

$$\text{s.t.} \quad Ax \geq a, \; x \in X, \tag{10.9b}$$

$$D^\top \lambda + \mu \geq d, \tag{10.9c}$$

$$\rho \leq \mathrm{Diag}(M) x_L, \tag{10.9d}$$

$$\rho \leq \mu, \tag{10.9e}$$

$$(\lambda, \mu, \rho) \geq 0. \tag{10.9f}$$

To enforce the identity $\rho_i = \mu_i x_i$, the McCormick inequalities (10.9d)–(10.9e) are added to the model. Note that, except for these McCormick inequalities, the ρ variables only appear in the objective function. Because the respective objective function coefficients $-u$ are negative (with u_i being the default upper bound on y_i, $i \in \{1, \ldots, n_y\}$) and due to the minimization objective, we always have $\rho_i = \min\{M_i x_i, \mu_i\}$ for all $i \in L$ in a solution.

Remark 10.8 The technique discussed above is, e.g., applied by Wood (1993) for reformulating the maximum-flow interdiction problem. Lim and Smith (2007) apply it to another network interdiction problem in which the leader removes arcs from the graph, whereas the follower maximizes the profit from routing a given set of commodities in the resulting network. Note that finding the smallest possible upper bounds M_i is not an easy task and can be NP-hard in general (Kleinert et al. 2020); see also the detailed discussion about choosing big-M values in Section 3.4.

Remark 10.9 In the continuous interdiction setting, i.e., for $x_i \in [0, 1] \subset \mathbb{R}$, $i \in L$, the McCormick inequalities (10.9d)–(10.9e) only lead to a relaxation but not to a reformulation of the overall bilevel problem. Thus, we need a different approach in this setting. We observe that the bilinear formulation (10.8) represents what is also called a *disjointly constrained bilinear program*, i.e., an optimization problem in which no constraint depends on both x and (λ, μ). The only coupling between these blocks of variables occurs in the objective function. This implies that an optimal solution is attained at a point (x^*, λ^*, μ^*), where x^* is an extreme point of the convex hull of the set $\{x \in X : Ax \geq a\}$ and (λ^*, μ^*) is an extreme point of the polytope determined by the constraints in (10.8c) and (10.8d). This property is typically exploited to derive exact and tailored solution approaches; see, e.g., Lim and Smith (2007).

Let us now turn our attention to lower-level problems of the form given in (10.3) in which the leader controls the coefficients of the follower's objective function but the feasible space of the y variables, namely $\{y \in \mathbb{R}^{n_y} : Dy \leq b,\ y \geq 0\}$, does not depend on x. This means, the overall interdiction problem now reads

$$\begin{aligned} \min_x \quad & \varphi(x) \\ \text{s.t.} \quad & Ax \geq a,\ x \in X \end{aligned}$$

with

$$\varphi(x) := \max_y \left\{ \sum_{i \in L} (d_i - \delta_i x_i) y_i : Dy \leq b,\ y \geq 0 \right\}.$$

Because the lower-level problem is an LP for a given x, we can again dualize it. By doing so, we get rid of the bilinear terms $x_i y_i$, $i \in L$, and obtain the single-level reformulation

$$\begin{aligned} \min_{x,\lambda} \quad & b^\top \lambda \\ \text{s.t.} \quad & Ax \geq a, \; x \in X, \\ & D^\top \lambda \geq d - \operatorname{Diag}(\delta) x_L, \\ & \lambda \geq 0. \end{aligned} \tag{10.10}$$

Here, λ are the dual variables associated with the constraints $Dy \leq b$ and $\operatorname{Diag}(\delta)$ is an $L \times L$ diagonal matrix with entries δ_i for all $i \in L$. If the upper-level variables are continuous, the reformulation (10.10) is a linear optimization problem and, hence, the underlying interdiction problem is solvable in polynomial time; see, e.g., the continuous shortest-path interdiction problem introduced above and the result by Fulkerson and Harding (1977). If the set X restricts all or some of the x variables to be binary, the resulting reformulation is an MILP; see e.g., Israeli and Wood (2002) for a reformulation of the discrete shortest-path interdiction problem.

Exercise 10.10 Use the dualization technique to derive a single-level reformulation of the continuous shortest-path interdiction problem introduced in Section 10.2.1. What do you observe about this model? Can it be solved in polynomial time?

Exercise 10.11 Consider the discrete shortest-path interdiction problem introduced in Section 10.2.1. Derive a single-level reformulation of this problem using the dualization technique. What differences do you observe compared to the model derived in Exercise 10.10?

10.3.2 Penalization

Besides dualization, tailored penalization techniques can be used to tackle interdiction problems as well. The next theorem again deals with interdiction problems in which the x variables enter the lower-level feasible set through constraints of the type $y_i \leq u_i(1 - x_i)$ for $i \in L$. The idea of the penalization approach is to replace these constraints with the x-independent constraints $y \leq u$ and to penalize the violations of the original linking constraints by adding additional terms to the objective function. The following result is one of the most central results in the area of interdiction problems; see also Wood (2011).

Theorem 10.12 *Suppose that Assumption 10.1 holds. Moreover, let M_i be a strict upper bound for an optimal dual solution associated with the* *constraint* $y_i \le u_i(1 - x_i)$*,* $i \in L$*, taken over all* $\{x \in \{0,1\}^{n_x} : Ax \ge a\}$*. Then, the problem*

$$\min_x \quad \varphi^p(x) \quad \text{s.t.} \quad x \in \{0,1\}^{n_x}, \; Ax \ge a, \tag{10.11}$$

with φ^p *being the optimal-value function of*

$$\max_y \quad \left(d^\top - x_L^\top \operatorname{Diag}(M)\right) y \tag{10.12a}$$

$$\text{s.t.} \quad Dy \le b, \; y \ge 0, \tag{10.12b}$$

$$y \le u \tag{10.12c}$$

and the original interdiction problem

$$\min_x \quad \varphi(x) \quad \text{s.t.} \quad x \in \{0,1\}^{n_x}, \; Ax \ge a,$$

with φ *being the optimal-value function of*

$$\begin{aligned} \max_y \quad & d^\top y \\ \text{s.t.} \quad & Dy \le b, \; y \ge 0, \\ & y_i \le u_i(1 - x_i) \quad \text{for all } i \in L \end{aligned}$$

have the same set of optimal solutions and, thus, the same optimal objective function values.

Proof: Using the exactness of the ℓ_1-penalization for linear optimization problems given in Theorem A.10, the result immediately follows by proving that $x_L^\top \operatorname{Diag}(M) y$ is the ℓ_1-penalization (with penalty parameters M) of the constraints $y_i \le u_i(1 - x_i)$, $i \in L$. Hence, it is enough to show that

$$M_i x_i y_i = M_i [u_i(1 - x_i) - y_i]^-$$

holds for all $i \in L$. To this end, let $i \in L$ be chosen arbitrarily. First, for $x_i = 0$ and by using $y \le u$, we obtain

$$M_i x_i y_i = 0 = M_i [u_i - y_i]^- = M_i [u_i(1 - x_i) - y_i]^-.$$

Second, for $x_i = 1$ and by using $y \ge 0$, we get

$$M_i x_i y_i = M_i y_i = M_i [u_i(1 - x_i) - y_i]^-.$$

This completes the proof. □

In Theorem 10.12, we assume that the lower-level problem is an LP. Hence, after the transformation into (10.11), the inner maximization problem remains a linear problem and we can apply LP duality theory again to further reformulate the problem as a single-level problem, following the technique described in

Section 10.3.1. The respective dual variables $\lambda \geq 0$ and $\mu \geq 0$ are associated with the constraints $Dy \leq b$ and $y \leq u$, respectively. We then obtain the model

$$\begin{aligned}\min_{x,\lambda,\mu} \quad & b^\top \lambda + u^\top \mu \\ \text{s.t.} \quad & Ax \geq a, \ x \in \{0,1\}^{n_x}, \\ & D^\top \lambda + \mu + \operatorname{Diag}(M) x_L \geq d, \\ & (\lambda, \mu) \geq 0,\end{aligned}$$

which can be solved using an off-the-shelf general-purpose MILP solver; see Section 3.4 for a list of possible options. As already mentioned, finding tight values for M_i may not be an easy task. For some special cases such as the maximum-flow interdiction problem with integral capacities u_i, however, tight values of M_i can be obtained. Indeed, increasing the capacity of an arc by one unit increases the maximum flow by at most one unit as well. Hence, $M_i = 1$ holds; see, e.g., Cormican et al. (1998).

10.4 Solution Approaches for Discrete Lower-Level Problems

Whereas the dualization technique of the last section works exclusively for convex lower-level problems (LPs, in particular), the penalization technique is more general and can also be applied in situations in which the lower level includes discrete decisions.

The next theorem is the discrete analogue of Theorem 10.12. For the ease of presentation, let us assume in the following that the variable bounds $y \leq u$ are encoded in the system $Dy \leq b$.

Theorem 10.13 (See Theorem 3.1 in Fischetti et al. (2018b)) *Suppose that $x_i \in \{0,1\}$ for all $i \in L$ and that Assumption 10.1 as well as Assumption 10.2 hold. In particular, this implies that, for all $x \in X$ such that $Ax \geq a$ holds, the lower-level feasible set*

$$\{y \in \mathbb{R}^{n_y} : Dy \leq b, \ y \geq 0, \ y_i \leq u_i(1 - x_i), \ y_i \in \mathbb{Z}, \ i \in L\}$$

is non-empty and bounded. Moreover, let

$$M_i > \max\left\{d_i, \ \bar{v} - \underline{v}\right\} \quad \textit{for all } i \in L,$$

with

$$\bar{v} := \sum_{i=1}^{n_y} \max\{0, d_i u_i\} \quad \textit{and} \quad \underline{v} := \sum_{i=1}^{n_y} \min\{0, d_i u_i\}$$

be given as well. Then, any optimal solution to the x-parameterized integer linear problem

$$\begin{aligned} \max_y \quad & d^\top y - \sum_{i \in L} M_i x_i y_i \\ \text{s.t.} \quad & Dy \le b, \ y \ge 0, \\ & y_i \in \mathbb{Z} \quad \text{for all } i \in L, \end{aligned} \tag{10.13}$$

also solves the x-parameterized lower-level problem

$$\max_y \quad d^\top y \tag{10.14a}$$

$$\text{s.t.} \quad Dy \le b, \ y \ge 0, \tag{10.14b}$$

$$y_i \le u_i(1 - x_i) \quad \text{for all } i \in L, \tag{10.14c}$$

$$y_i \in \mathbb{Z} \quad \text{for all } i \in L. \tag{10.14d}$$

Proof: The Weierstraß theorem ensures that Problems (10.13) and (10.14) have an optimal solution. We now prove the claim by contradiction. To this end, we assume that the penalized problem (10.13) admits an optimal solution that violates (some of) the interdiction constraints in (10.14c). We then show that this leads to an objective value of Problem (10.13) that cannot be optimal.

More formally, let $\hat{y}$ be an optimal solution to Problem (10.13). Suppose that there exists an index $j \in L$ for which the inequality $\hat{y}_j \le u_j(1 - x_j)$ is violated. Because the constraints $y \le u$ are included in the system $Dy \le b$ by assumption, this means that $x_j = 1$ and $\hat{y}_j > 0$ has to hold. In particular, because of $\hat{y}_j \in \mathbb{Z}$, this implies $\hat{y}_j \ge 1$. Moreover, we have

$$\begin{aligned} & d^\top \hat{y} - \sum_{i \in L} M_i x_i \hat{y}_i \\ = & \sum_{i \in L \setminus \{j\}} d_i \hat{y}_i + d_j \hat{y}_j - \sum_{i \in L \setminus \{j\}} M_i x_i \hat{y}_i - M_j x_j \hat{y}_j \\ \le & \sum_{i \in L \setminus \{j\}} \max\{0, d_i u_i\} + d_j \hat{y}_j - M_j x_j \hat{y}_j \\ = & \sum_{i \in L \setminus \{j\}} \max\{0, d_i u_i\} + d_j \hat{y}_j - M_j x_j \hat{y}_j \\ & \quad + \max\left\{0, d_j u_j\right\} - \max\left\{0, d_j u_j\right\} \\ = & \bar{v} - \max\left\{0, d_j u_j\right\} + d_j \hat{y}_j - M_j \hat{y}_j \\ \le & \bar{v} - M_j \hat{y}_j \\ \le & \bar{v} - M_j \\ < & \underline{v}. \end{aligned}$$

Here, the first equality follows from considering the terms for the jth index separately. The first inequality follows from $0 \leq \hat{y} \leq u$ and from $M_i \geq 0$ for all $i \in \{1, \dots, n_y\}$. The second equality is obtained by adding and subtracting the term $\max\{0, d_j u_j\}$, whereas the third equality follows from the definition of $\bar{v}$ and from $x_j = 1$. The second inequality again follows from $0 \leq \hat{y} \leq u$, whereas the third inequality is due to $\hat{y}_j \geq 1$. Finally, the strict inequality follows from $M_j > \bar{v} - \underline{v}$.

Let now y^* be an optimal solution to the x-parameterized lower-level problem (10.14). Then, y^* is also feasible for Problem (10.13). Moreover the interdiction constraints in (10.14c) imply $x_i y_i^* = 0$ for all $i \in L$. Let now $\hat{v}$ be the optimal objective function value of Problem (10.13). Then, we have

$$\underline{v} \leq \varphi(x) = d^\top y^* = d^\top y^* - \sum_{i \in L} M_i x_i y_i^* \leq \hat{v}.$$

Here, the first inequality follows from $0 \leq y \leq u$ and the definition of $\underline{v}$. The first equality follows from the optimality of y^* for Problem (10.14), whereas the second equality is due to $x_i y_i^* = 0$ for all $i \in L$. Finally, the second inequality follows from the feasibility of y^* for Problem (10.13).

To sum up, we have shown

$$\hat{v} := d^\top \hat{y} - \sum_{i \in L} M_i x_i \hat{y}_i < \underline{v} \leq d^\top y^* - \sum_{i \in L} M_i x_i y_i^* \leq \hat{v},$$

which is a contradiction and, thus, concludes the proof. □

Once the penalized lower level is obtained by applying the last theorem, one can apply a Benders-like decomposition approach, which we discuss in the next section.

10.5 Benders-Like Decomposition

Since its introduction by Benders (1962), *Benders decomposition* has become a very useful method for a large number of MILP applications such as facility location, network design, or scheduling; see, e.g., the surveys by Clautiaux and Ljubić (2025) and Rahmaniani et al. (2017) for further details.

Roughly speaking, Benders decomposition for single-level MILPs works as follows. The original MILP is decomposed into a (relaxed) master problem and one or more sub-problems. Typically, discrete variables are kept in the master problem so that the remaining part of the problem, which then forms the sub-problem, becomes continuous and thus much easier to solve. This means that the original MILP is reformulated in the space of the discrete decision variables, whereas the variables of the sub-problem are projected out. Constraints linking the variables from the master and the sub-problem(s) are then replaced by a family of so-called *Benders cuts*, which are derived using LP duality theory.

The method has been originally proposed for problems in which an LP is solved once the discrete variables (which remain in the master problem) are fixed. Benders decomposition has been later generalized by Geoffrion (1972) to deal with convex sub-problems. The generalized Benders cuts of Geoffrion are derived using Lagrangian duality theory. More recently, logic-based Benders decomposition has been introduced; see the recent book by Hooker (2024) for further details. The latter allows to tackle more general sub-problems, e.g., discrete and nonlinear problems, and relies on logical inference instead of duality theory for convex optimization. In what follows, we exploit the idea of Benders decomposition to tackle interdiction problems.

For interdiction problems through cost modification, in which the value function is given as in Problem (10.3), the interdiction problem is given by

$$\min_{x \in X} \left\{ \max_{y \in Y} \left\{ \sum_{i \in L} (d_i - \delta_i x_i) y_i : Dy \leq b, \ y \geq 0 \right\} : Ax \geq a \right\}. \tag{10.15}$$

Throughout this section, we impose Assumption 10.1 and consider both continuous and discrete lower-level problems. Let us note that problems obtained after applying the penalization technique can also be stated as an instance of Problem (10.15). In this setting, we have $\delta_i = M_i$ for all $i \in L$ and we again assume that the constraints $y \leq u$ are included in the system $Dy \leq b$; see also Theorem 10.13.

In Problem (10.15), the set $\{y \in Y : Dy \leq b, \ y \geq 0\}$ of lower-level feasible points does not depend on the decisions of the leader. Moreover, there are no coupling constraints in the upper level. In this case, an alternative and natural modeling approach is to derive a formulation in the x-space only, which resembles the Benders decomposition approach in the following sense. The variables x are kept in the (master) problem, whereas the variables y are projected out and become part of the Benders sub-problem. The links between x and y are then replaced by Benders-like cuts. Instead of using LP duality theory, as in the traditional Benders approach, these cuts are now derived from the primal description of the lower-level problem. Specifically, we observe that

Problem (10.15) can be re-stated using its epigraph reformulation

$$\begin{aligned} \min_{x,\theta} \quad & \theta \\ \text{s.t.} \quad & \theta \geq \varphi(x), \\ & Ax \geq a, \\ & x \in X. \end{aligned} \tag{10.16}$$

The latter is the Benders master problem. Whenever $\tilde{x} \in \{x \in X : Ax \geq a\}$ is fixed, we solve the sub-problem

$$\varphi(\tilde{x}) = \max_{y \in Y} \left\{ \sum_{i \in L} y_i (d_i - \delta_i \tilde{x}_i) : Dy \leq b, \; y \geq 0 \right\},$$

which is an (MI)LP in the y-space. Under Assumption 10.1, the lower-level problem is feasible and bounded. Hence, the convex hull of all feasible points of the lower level, i.e.,

$$\operatorname{conv}\left(\{y \in Y : Dy \leq b, \; y \geq 0\}\right),$$

is a polytope. Let now $\hat{Y}$ be the finite set of extreme points of this polytope. The optimal-value function φ can then be re-written as

$$\varphi(x) = \max \left\{ \sum_{i \in L} (d_i - \delta_i x_i) \hat{y}_i : \hat{y} \in \hat{Y} \right\}. \tag{10.17}$$

For any given extreme point $\hat{y} \in \hat{Y}$, the term

$$\sum_{i \in L} (d_i - \delta_i x_i) \hat{y}_i$$

is an affine function in x. Hence, the function φ is the finite maximum of a given set of affine functions. This implies that φ is convex. Altogether, we can now re-state Problem (10.16) as

$$\min_{x,\theta} \quad \theta \tag{10.18a}$$

$$\text{s.t.} \quad \theta \geq \sum_{i \in L} (d_i - \delta_i x_i) \hat{y}_i \quad \text{for all } \hat{y} \in \hat{Y}, \tag{10.18b}$$

$$Ax \geq a, \tag{10.18c}$$

$$x \in X, \tag{10.18d}$$

which is an (MI)LP in the (x, θ)-space. We refer to a constraint in (10.18b) as an *interdiction cut* induced by a point $\hat{y} \in \hat{Y}$. Because these constraints resemble Benders cuts, we refer to Problem (10.18) as the *Benders-like problem reformulation*. One key advantage of reformulating the interdiction problem (10.15) as

Problem (10.18) is that it offers a solution framework applicable to problems with discrete lower levels.

In general, the number of interdiction cuts is exponential. Thus, it is not practical to put all of these constraints into an off-the-shelf MILP solver. A possible way to deal with the interdiction cuts in (10.18b) is to generate them dynamically because, usually, only a subset of these cuts is needed to determine an optimal solution. In what follows, we discuss two possible ways to solve Problem (10.18). The first one is an iterative cutting-plane procedure. The second one is a branch-and-cut approach with a node-processing procedure similar to the one presented in Algorithm 7 in Chapter 9.

Finally note that if we obtain the problem in the form given in (10.15) by using the penalization approach, the values of δ_i are replaced by M_i for all $i \in L$. Again, these big-M values need to be chosen carefully as, otherwise, the continuous relaxation of Problem (10.18) can be rather weak.

10.5.1 Cutting-Plane Procedure: Multi-Tree Implementation

Recall that $\hat{Y}$ is the finite set of extreme points of $\operatorname{conv}(\{y \in Y : Dy \le b,\ y \ge 0\})$. Let now $\hat{Y}_k \subset \hat{Y}$ be a subset of these extreme points. Moreover, let ℓ_θ denote a global lower bound on the value of $\varphi(x)$ for any $x \in X$ that satisfies $Ax \ge a$. For example, if $d \ge 0$, we can set ℓ_θ to zero. We now define the *relaxed master problem* (RMP) of (10.18) associated with the set $\hat{Y}_k$ as

$$
\begin{aligned}
\min_{x,\theta} \quad & \theta && \text{(10.19a)}\\
\text{s.t.} \quad & \theta \ge \sum_{i \in L} (d_i - \delta_i x_i) \hat{y}_i \quad \text{for all } \hat{y} \in \hat{Y}_k, && \text{(10.19b)}\\
& \theta \ge \ell_\theta, && \text{(10.19c)}\\
& Ax \ge a, && \text{(10.19d)}\\
& x \in X. && \text{(10.19e)}
\end{aligned}
$$

Lemma 10.14 *Let $(\tilde{x}^k, \tilde{\theta}^k)$ be an optimal solution to the relaxed master problem (10.19) and let θ^* be the optimal objective function value of Problem (10.18). Then,*

$$\tilde{\theta}^k \le \theta^* \le \varphi(\tilde{x}^k) \qquad (10.20)$$

holds. Moreover, the point $(\tilde{x}^k, \tilde{\theta}^k)$ is an optimal solution to Problem (10.18) if and only if

$$\tilde{\theta}^k = \varphi(\tilde{x}^k)$$

holds.

Proof: Note that (10.19) is a relaxation of (10.18). Therefore, its optimal value $\tilde{\theta}^k$ is a lower bound for the value of θ^*. This shows the first inequality in (10.20). Similarly, $\tilde{x}^k$ is a feasible point for the leader and, thus, the objective function value of the follower evaluated at this point, i.e., $\varphi(\tilde{x}^k)$, is an upper bound for Problem (10.18). This proves (10.20).

To show the necessary and sufficient conditions, assume first that $(\tilde{x}^k, \tilde{\theta}^k)$ is an optimal solution to Problem (10.18). Then, $(\tilde{x}^k, \tilde{\theta}^k)$ satisfies all constraints in (10.18b), i.e.,

$$\tilde{\theta}^k \geq \max_{y \in \hat{Y}} \left\{ \left(d^\top - \tilde{x}_L^{k\top} \operatorname{Diag}(\delta) \right) \hat{y} \right\} = \varphi(\tilde{x}^k) \geq \ell_\theta$$

holds. Because (10.18) is a minimization problem and because no other constraints of the problem involve the variable θ, the minimum is attained at $\tilde{\theta}^k = \varphi(\tilde{x}^k)$.

Assume now that $(\tilde{x}^k, \tilde{\theta}^k)$ is an optimal solution to (10.19) and that $\varphi(\tilde{x}^k) = \tilde{\theta}^k$ holds. Then, Inequality (10.20) yields $\theta^* = \varphi(\tilde{x}^k) = \tilde{\theta}^k$. □

Remark 10.15 Given a family of sets $\hat{Y}_1 \subset \hat{Y}_2 \subset \cdots \subset \hat{Y}_k \subset \hat{Y}$, let $(\tilde{x}^l, \tilde{\theta}^l)$ be an optimal solution to the RMP associated with the set $\hat{Y}_l$ with $l \in \{1, \ldots, k\}$. Then, the sequence of lower bounds obtained by solving the RMP associated with $\hat{Y}_l$, $l \in \{1, \ldots, k\}$, is non-decreasing, i.e., it satisfies

$$\tilde{\theta}^1 \leq \tilde{\theta}^2 \leq \cdots \leq \tilde{\theta}^k \leq \theta^*.$$

This is because the RMP associated with $\hat{Y}_{l-1}$ is a relaxation of the RMP associated with $\hat{Y}_l$. However, the sequence of upper bounds $\varphi(\tilde{x}^l)$ associated with $\hat{Y}_l$ is not necessarily monotone. Hence, the global upper bound is obtained as

$$\mathcal{U} = \min_{l \in \{1, \ldots, k\}} \left\{ \varphi(\tilde{x}^l) \right\}.$$

The last remark serves as a motivation for a cutting-plane approach, which is formally stated in Algorithm 8.

The algorithm maintains global lower and upper bounds, $\mathcal{L}$ and $\mathcal{U}$, respectively, which are initialized in Line 1. In each iteration of the while-loop, we solve an RMP (see Line 4), so that Algorithm 8 overall solves a sequence of RMPs. We start by solving the relaxed master problem associated with $\hat{Y}_1 = \emptyset$, i.e., the RMP without interdiction cuts, and then iteratively enlarge the set $\hat{Y}_k$ and solve the associated RMPs until the optimality gap is closed, i.e., $\mathcal{U} = \mathcal{L}$ holds. Let $(\tilde{x}^k, \tilde{\theta}^k)$ be an optimal solution to the RMP (10.19) solved in iteration k. To verify whether this point is an optimal solution to Problem (10.18), we need to check if $\tilde{\theta}^k = \varphi(\tilde{x}^k)$ holds according to Lemma 10.14. To this end, we solve Problem (10.17) for the given $\tilde{x}^k$ in Line 6. If $\tilde{\theta}^k = \varphi(\tilde{x}^k)$ holds, the algorithm

Algorithm 8 Multi-Tree Implementation of Benders Decomposition

Input: An instance of Problem (10.18) and a lower bound ℓ_θ for $\varphi(x)$ for any $x \in X$ with $Ax \geq a$.

Output: An optimal solution to Problem (10.18)

1: Set $\mathcal{L} \leftarrow -\infty$, $\mathcal{U} \leftarrow \infty$, $k \leftarrow 0$, and $\hat{Y}_1 \leftarrow \emptyset$.
2: **while** $\mathcal{U} > \mathcal{L}$ **do**
3: Set $k \leftarrow k + 1$.
4: Let $(\tilde{x}^k, \tilde{\theta}^k)$ denote an optimal solution to the RMP (10.19).
5: Update the lower bound $\mathcal{L} \leftarrow \tilde{\theta}^k$.
6: Solve Problem (10.17) to obtain $\varphi(\tilde{x}^k)$.
7: Update the upper bound $\mathcal{U} \leftarrow \min\{\mathcal{U}, \varphi(\tilde{x}^k)\}$.
8: **if** $\tilde{\theta}^k < \varphi(\tilde{x}^k)$ **then**
9: Let $\hat{y}$ be an optimal solution to Problem (10.17).
10: Set $\hat{Y}_{k+1} \leftarrow \hat{Y}_k \cup \{\hat{y}\}$.
11: **return** $(\tilde{x}^k, \tilde{\theta}^k)$

terminates because $\mathcal{L} = \tilde{\theta}^k = \mathcal{U}$. Otherwise, by Lemma 10.14, $\tilde{\theta}^k$ is a valid lower and $\varphi(\tilde{x}^k)$ is a valid upper bound with $\tilde{\theta}^k < \varphi(\tilde{x}^k)$. Let $\hat{y} \in \hat{Y}$ be a point that maximizes $\varphi(\tilde{x}^k)$. Then, the point $\hat{y}$ belongs to $\hat{Y} \setminus \hat{Y}_k$ because, otherwise, the constraints in (10.19b) evaluated at $(\tilde{x}^k, \tilde{\theta}^k)$ would imply

$$\tilde{\theta}^k \geq \sum_{i \in L} (d_i - \delta_i \tilde{x}_i^k) \hat{y}_i = \varphi(\tilde{x}^k).$$

Thus, the point $\hat{y} \in \hat{Y} \setminus \hat{Y}_k$ induces an interdiction cut that is violated by the current RMP solution $(\tilde{x}^k, \tilde{\theta}^k)$. Hence, the point $\hat{y}$ is added to the set $\hat{Y}_k$ to obtain $\hat{Y}_{k+1}$. The last steps discussed above correspond to Lines 8–10 of Algorithm 8.

Proposition 10.16 *Algorithm 8 terminates with an optimal solution to Problem (10.18) after adding a finite number of interdiction cuts.*

Proof: At iteration k, the set $\hat{Y}_k$ is augmented with (at least) one element that does not yet belong to it and, in the worst case, all finitely many extreme points of $\hat{Y}$ need to be enumerated according to Lemma 10.14 and Remark 10.15. □

For continuous interdiction problems, Algorithm 8 terminates after solving a finite number of restricted master problems that are LPs. In the case of discrete interdiction problems, each iteration of the above procedure requires solving an MILP associated with the relaxed master problem and the set $\hat{Y}_k$. This approach is thus referred to as a *multi-tree implementation* of Benders decomposition because each iteration involves a branch-and-bound procedure applied to the underlying mixed-integer linear RMP.

Finally, we note that it is also possible to terminate Algorithm 8 with a solution that is close to being optimal. This can be achieved by stopping the method as soon as $\mathcal{U} - \mathcal{L} \leq \varepsilon$ holds for a given tolerance $\varepsilon > 0$; see also Remark 5.5.

10.5.2 Branch-and-Cut Procedure: Single-Tree Implementation

For discrete interdiction, i.e., for interdiction problems with binary variables x, solving the RMP (10.19) in each iteration k as it is done in Algorithm 8 can be time-consuming. Hence, one might prefer to dynamically generate interdiction cuts within a single branch-and-bound tree. To this end, we again start from the RMP associated with $\hat{Y}_1 = \emptyset$ but, instead of solving the RMP as an MILP, we solve its LP relaxation in which the integrality constraints for the x variables are relaxed. We then branch whenever the obtained solution violates (some of) the integrality constraints and we dynamically separate interdiction cuts otherwise. The overall procedure resembles the one presented in Algorithm 7 in Chapter 9. For the sake of completeness, the method for processing node k of the branch-and-cut search tree is formally stated in Algorithm 9. At node k, we consider the LP

$$
\begin{aligned}
\min_{x,\theta} \quad & \theta \\
\text{s.t.} \quad & \theta \geq \ell_\theta, \\
& Ax \geq a, \\
& (x, \theta) \in \bar{\Omega}_k \cap \bar{\Omega}_{\text{global}}.
\end{aligned}
\tag{10.21}
$$

Here, $\bar{\Omega}_k$ is the feasible set of the LP relaxation of the single-level relaxation that is additionally restricted by all locally valid inequalities that have been generated by the solver at nodes along the path from the root node to node k as well as by all branching decisions that have been made along that path. In addition, we maintain a pool of globally valid inequalities of the form $\theta \geq \sum_{i \in L} (d_i - \delta_i x_i) \hat{y}_i$ that have been separated so far at any node of the search tree. The set induced by these inequalities is denoted by $\bar{\Omega}_{\text{global}}$.

Exercise 10.17 Prove that, if we embed Algorithm 9 into a classic branch-and-bound framework (see Algorithm 5), we obtain a method that terminates with an optimal solution to Problem (10.18) after adding an overall finite number of interdiction cuts. (*Hint:* The proof goes along the lines of the proof of Theorem 9.11 and Proposition 10.16.)

Recall that, if the follower solves an NP-hard problem, the resulting interdiction

Algorithm 9 Processing Node k of the Branch-and-Cut Search Tree

Input: An instance of Problem (10.21), an upper bound $\mathcal{U} \in \mathbb{R} \cup \{\infty\}$ for the optimal objective function value of Problem (10.18), a lower bound ℓ_θ for $\varphi(x)$ for any $x \in X$ with $Ax \geq a$, and the set $\bar{\Omega}_{\text{global}}$ induced by the pool of interdiction cuts separated so far

1: Solve Problem (10.21).
2: **if** Problem (10.21) is infeasible **then**
3: Fathom the current node, i.e., **stop** the node processing procedure and go back to the main method.
4: Let $(\tilde{x}^k, \tilde{\theta}^k)$ denote an optimal solution to Problem (10.21).
5: **if** $\tilde{\theta}^k \geq \mathcal{U}$ **then**
6: Fathom the current node, i.e., **stop** the node processing procedure and go back to the main method.
7: Solve Problem (10.17) to obtain $\varphi(\tilde{x}^k)$.
8: Let $\hat{y}$ be an optimal solution to Problem (10.17).
9: **if** $\tilde{x}^k \notin X$ **then**
10: Either generate an interdiction cut $\theta \geq \sum_{i \in L}(d_i - \delta_i x_i)\hat{y}_i$, augment $\bar{\Omega}_{\text{global}}$, and go to Line 1, or branch on a binary variable with a fractional value to create two new sub-problems, **stop** this node processing procedure, and go back to the main method.
11: **if** $\varphi(\tilde{x}^k) < \mathcal{U}$ **then**
12: Update the incumbent, i.e., update the incumbent solution with $(\tilde{x}^k, \varphi(\tilde{x}^k))$ and set the incumbent value to $\mathcal{U} \leftarrow \varphi(\tilde{x}^k)$.
13: **if** $\tilde{\theta}^k < \varphi(\tilde{x}^k)$ **then**
14: Augment $\bar{\Omega}_{\text{global}}$ with the interdiction cut $\theta \geq \sum_{i \in L}(d_i - \delta_i x_i)\hat{y}_i$ and go to Line 1.

problems are typically Σ_2^p-hard. Further recall that there is no way of formulating Σ_2^p-hard problems as single-level integer programs of polynomial size unless the polynomial hierarchy collapses. In particular, this means that separating the interdiction cuts in (10.18b) for any given solution $(\tilde{x}^k, \tilde{\theta}^k)$ of the leader requires solving the NP-hard follower's problem. If effective algorithms exist for solving these problems, they should be embedded in Line 7 of the node-processing procedure in Algorithm 9 to compute $\varphi(\tilde{x}^k)$ and find $\hat{y}$—rather than using MILPs in combination with general-purpose solvers. For example, dynamic programming can be used to compute $\varphi(\tilde{x}^k)$ for knapsack interdiction (Fischetti et al. 2019), whereas tailored branch-and-bound procedures can be used for clique interdiction (Furini et al. 2019).

To sum up, one can choose between a single-tree or a multi-tree implementation. In practice, the single-tree implementation (i.e., the branch-and-cut approach) often outperforms its multi-tree counterpart (i.e., Algorithm 8). However, there is no general rule and the effectiveness of either approach depends on the specific application.

10.5.3 Interdiction Cuts Derived from Heuristic Follower's Solutions

The separation of interdiction cuts consists of finding a point $\hat{y} \in \hat{Y}$ for which the current RMP solution violates the associated constraint in (10.18b). In Algorithms 8 and 9, we propose to solve the lower-level problem to global optimality to obtain the corresponding point $\hat{y}$. Because the lower-level problem can be NP-hard, this procedure may be time-consuming. Hence, we now discuss how to exploit heuristic solutions to the lower-level problem to generate interdiction cuts. To this end, we first need to show that a heuristic solution to the lower-level problem induces an interdiction cut that is valid for Problem (10.18). This is the task of the following exercise.

Exercise 10.18 Observe that the points $\hat{y} \in \hat{Y}$ do not depend on x. Prove that not only the extreme points $\hat{y} \in \hat{Y}$ but also any arbitrary point from the set $\bar{Y} = \{y \in Y : Dy \leq b,\ y \geq 0\}$ induces a valid interdiction cut (10.18b) for Problem (10.18).

There is the following trade-off. On the one hand, an interdiction cut induced by an extreme point in $\hat{Y}$ is usually stronger than one induced by a general feasible point from $\bar{Y}$. On the other hand, the first one is also more expensive to compute in general. Because solving the lower-level problem to global optimality may be too time-consuming, one can interrupt searching for the optimal value $\varphi(\tilde{x}^k)$ as soon as a feasible point $\bar{y} \in \bar{Y}$ is found that leads to a lower bound $\mathcal{L}_{\varphi(\tilde{x}^k)} \leq \varphi(\tilde{x}^k)$ with $\tilde{\theta}^k < \mathcal{L}_{\varphi(\tilde{x}^k)}$. In such a case, a valid interdiction cut (10.18b) induced by the point $\bar{y} \in \bar{Y}$ may be added to the relaxed master problem. If a general-purpose solver is used to search for $\varphi(\tilde{x}^k)$, this means that we can give a cut-off value of $\tilde{\theta}^k$ to the solver so that the execution is stopped as soon as a feasible point whose value is greater than $\tilde{\theta}^k$ is found. Alternatively, we could also apply a heuristic procedure to search for $\varphi(\tilde{x}^k)$ and separate the point $(\tilde{x}^k, \tilde{\theta}^k)$ if the heuristic value obtained is greater than $\tilde{\theta}^k$. The only time an exact solution is needed to separate the point $(\tilde{x}^k, \tilde{\theta}^k)$, or to verify its feasibility, is when $\tilde{x}^k$ is binary and no violated cuts induced by a heuristic solution are found. Because the interdiction cuts induced by points from $\bar{Y}$ are globally valid, once they are added to the relaxed master problem, they can no longer be violated. Therefore, assuming that the lower-level problem is discrete, only a finite number of cuts induced by points from $\bar{Y}$ can be added to the RMP. Hence, both the cutting-plane procedure and the branch-and-cut method terminate after finitely many steps.

10.6 The Downward Monotonicity Property

We now focus again on discrete interdiction problems in which the feasible region of the lower-level problem is modified through the linking constraints $y_i \leq u_i(1 - x_i)$, $i \in L$, and the penalization technique is applied so that the problem can be re-stated as

$$\min_{x \in X} \left\{ \max_{y \in Y} \left\{ \left(d^\top - x_L^\top \mathrm{Diag}(M)\right) y \colon Dy \leq b,\ y \geq 0 \right\} : Ax \geq a \right\}. \qquad (10.22)$$

Note that we keep the set Y here as the results in this section hold both for continuous as well as for discrete lower-level problems.

The choice of the coefficients M_i, $i \in L$, is crucial for the effectiveness of any solution approach based on Problem (10.22). In principle, the coefficients M_i have to be sufficiently large to ensure that there always exists an optimal solution to the modified follower's problem in which $x_i = 1$ implies $y_i = 0$ for all $i \in L$. In general, not much is known regarding the choice of these coefficients and finding the tightest one is NP-hard; see Kleinert et al. (2020). However, tight bounds can be found for some problems with a very special structure, which we show using Theorem 10.25 below. Whereas Theorem 10.12 only considers LPs at the lower level, for problems that satisfy a so-called *downward monotonicity property* (including those with binary or integer lower-level variables), it has been shown that M can be chosen as d. In this section, we present some important details regarding the implementation of the Benders-like decomposition and the associated branch-and-cut method for interdiction problems that satisfy this downward monotonicity property.

Remark 10.19 Recall that the only link between the upper- and lower-level decisions is given by the constraints $y_i \leq u_i(1 - x_i)$ for $i \in L$. The parametric follower's problem can often be modeled in the canonical space of y variables, but sometimes there are formulations in an extended space. Think, for example, of a spanning-tree formulation in the lower level, where the leader interdicts edges of the graph. A canonical formulation can be represented in the space of edge variables y only but with an exponential number of sub-tour elimination constraints (Edmonds 1971). Alternatively, we can have a flow-based compact formulation in the space of edges y and flows z; see, e.g., Section 3.2 in Magnanti and Wolsey (1995).

Similarly, we may have situations in which the leader interdicts only a subset of variables, whereas the other lower-level variables are not directly affected by the values of x. To illustrate this, we now consider a more general lower-level problem including additional z variables that are not in a 1-to-1 correspondence to the upper-level interdiction variables. The generalized model is given by

$$\begin{aligned} \varphi(x) := \max_{y \in Y, z \in Z} \quad & d_y^\top y + d_z^\top z \\ \text{s.t.} \quad & Dy + Ez \leq b, \ y \geq 0, \\ & y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L, \end{aligned} \tag{10.23}$$

where the set Z is a polytope ensuring that the variables $z \in Z$ are bounded, possibly intersected with the integer lattice.

Definition 10.20 (Interdiction Problem with Downward Monotonicity Property) An interdiction problem $\min_x \{\varphi(x) \colon x \in X, \ Ax \geq a\}$ for which the function $\varphi(x)$ is given by Problem (10.23) is said to satisfy the *downward monotonicity property* if and only if, for any $x \in X$ with $Ax \geq a$, the following property holds: If $(\bar{y}, \bar{z})$ is a feasible lower-level solution for a given x, then any $(\hat{y}, \bar{z})$ such that $0 \leq \hat{y} \leq \bar{y}$ and $\hat{y} \in Y$ holds is also a feasible lower-level solution for x.

Theorem 10.21 *An interdiction problem* $\min_x \{\varphi(x) \colon x \in X, \ Ax \geq a\}$ *satisfies the downward monotonicity property if and only if the lower-level problem can be formulated as Problem* (10.23) *with* $D \geq 0$*.*

Proof: Let us first assume that $D \geq 0$ holds. Then, for any feasible point $(\bar{y}, \bar{z})$ of Problem (10.23), it directly follows that $(\hat{y}, \bar{z})$ is also feasible for any $\hat{y}$ with $0 \leq \hat{y} \leq \bar{y}$ and $\hat{y} \in Y$.

For the other direction, we consider a formulation of the lower-level problem given as in (10.23) that satisfies the downward monotonicity property. We show that every negative entry in the matrix D can be replaced with zero, resulting in a new formulation whose optimal solution is not worse than the original one. To this end, suppose that there are m inequalities in the system $Dy + Ez \leq b$ and let $D_{i\cdot} y + E_{i\cdot} z \leq b_i$ denote the ith inequality. Furthermore, suppose that $D_{ih} < 0$ holds for some columns $h \in H \subseteq L$. We now show that the inequality $\hat{D}_{i\cdot} y + E_{i\cdot} z \leq b_i$ is valid for Problem (10.23), where $\hat{D}_{i\cdot}$ is obtained from $D_{i\cdot}$ as follows

$$\hat{D}_{ih} = \begin{cases} 0, & \text{if } h \in H, \\ D_{ih}, & \text{otherwise.} \end{cases}$$

To this end, let $(\bar{y}, \bar{z})$ be any feasible follower solution and let $(\hat{y}, \bar{z})$ be obtained from $(\bar{y}, \bar{z})$ by setting

$$\hat{y}_i = \begin{cases} 0, & \text{if } h \in H, \\ \bar{y}_i, & \text{otherwise.} \end{cases}$$

Due to the downward monotonicity assumption, $(\hat{y}, \bar{z})$ is a feasible follower decision. Hence, $D_{i\cdot}\hat{y} + E_{i\cdot}\bar{z} \leq b_i$ holds. By construction, we obtain

$$\hat{D}_{i\cdot}\bar{y} + E_{i\cdot}\bar{z} = \sum_{j \notin H} \hat{D}_{ij}\bar{y}_j + E_{i\cdot}\bar{z} = \sum_{j \notin H} D_{ij}\hat{y}_j + E_{i\cdot}\bar{z} = D_{i\cdot}\hat{y} + E_{i\cdot}\bar{z} \leq b_i$$

and, thus, $\hat{D}_{i\cdot}y + E_{i\cdot}z \leq b_i$ is a valid inequality because $(\bar{y}, \bar{z})$ has been chosen arbitrarily. By repeating this procedure for all rows i that contain at least one negative entry in the matrix D, we obtain a reformulation of Problem (10.23) with $D \geq 0$. □

Given the above result, we make the following assumption w.l.o.g. for the remainder of this section.

Assumption 10.22 It holds $D \geq 0$ and $d_y > 0$.

The additional assumption $d_y > 0$ is also w.l.o.g. If it would not hold, we could fix all variables y_j with $j \in L$ and $(d_y)_j \leq 0$ to zero and remove them from the model.

Exercise 10.23 Show that the following problems satisfy the downward monotonicity property:

(i) The maximum-clique interdiction problem defined in Section 10.2.3.
(ii) The knapsack interdiction problem defined in Example 1.18.

Exercise 10.24 Show that the following problems satisfy the downward monotonicity property and find the respective MILP-MILP bilevel formulation such that $D \geq 0$ holds.

(i) The maximum-matching interdiction problem (Zenklusen 2010): Here, the leader removes edges of the given graph $G = (\mathcal{V}, \mathcal{E})$ with interdiction cost $b_e > 0$ for each $e \in \mathcal{E}$, while respecting a given budget $\bar{b} > 0$. The follower searches for a maximum weighted matching in the resulting graph in which a weight $w_e \geq 0$ is associated with each $e \in \mathcal{E}$.
(ii) The prize-collecting traveling salesperson interdiction problem (Fischetti et al. 2019): We are given a complete graph $G = (\mathcal{V}, \mathcal{E})$ with vertex prizes $p_v \geq 0$, $v \in \mathcal{V}$, and edge costs $t_e \geq 0$, $e \in \mathcal{E}$. The leader removes vertices of G with interdiction cost $b_v > 0$ for each $v \in V$, while respecting a given budget $\bar{b} > 0$. The follower searches for a tour in the remaining

graph that maximizes the profit, which is defined as the sum of prizes of visited vertices minus the cost of the edges in the resulting tour.

(iii) The prize-collecting Steiner tree interdiction problem (Fischetti et al. 2019): We are given a complete graph $G = (\mathcal{V}, \mathcal{E})$ with vertex prizes $p_v \geq 0$, $v \in \mathcal{V}$, and edge costs $t_e \geq 0$, $e \in \mathcal{E}$. The leader removes vertices of G with interdiction cost $b_v > 0$ for each $v \in \mathcal{V}$, while respecting a given budget $\bar{b} > 0$. The follower searches for a subtree in the remaining graph that maximizes the profit, which is defined as the sum of prizes of connected vertices minus the cost of the edges in the resulting subtree.

We now want to apply the penalization technique to interdiction problems with the downward monotonicity property in which the lower-level problem is stated as Problem (10.23). To this end, let

$$Y_z = \operatorname{conv}(\{(y, z) \in Y \times Z \colon Dy + Ez \leq b,\ y \geq 0,\ y_i \leq u_i,\ i \in L\})$$

be the convex hull of all points of the modified lower-level problem in which the constraints $y_i \leq u_i(1 - x_i)$ are replaced with $y_i \leq u_i$ for all $i \in L$. Observe that, due to Assumption 10.1, Y_z is a polytope. Hence, let $\hat{Y}_z$ be the finite set of extreme points of Y_z. As in the previous section, we again generate interdiction cuts induced by extreme points from $\hat{Y}_z$. Moreover, the following theorem shows that, in such a case, tight big-M coefficients can be obtained for the problem reformulation resulting from penalization.

Theorem 10.25 *Let $X = \{0, 1\}^{n_x}$ and suppose that the interdiction problem*

$$\min_{x \in X} \quad \varphi(x) \quad \text{s.t.} \quad Ax \geq a \tag{10.24}$$

with φ being the optimal-value function of

$$\begin{aligned} \max_{y \in Y, z \in Z} \quad & d_y^\top y + d_z^\top z \\ \text{s.t.} \quad & Dy + Ez \leq b,\ y \geq 0, \\ & y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L, \end{aligned}$$

satisfies the downward monotonicity property. Then, it has the same objective function value as

$$\min_{x \in X} \quad \varphi^p(x) \quad \text{s.t.} \quad Ax \geq a \tag{10.25}$$

with φ^p being the optimal-value function of

$$\max_{(y,z) \in Y_z} \quad \left(d_y^\top - x_L^\top \operatorname{Diag}(d_y)\right) y + d_z^\top z.$$

Moreover, x^* solves (10.24) if and only if it solves (10.25).

Proof: Take any point $x^* \in X$ such that $Ax^* \geq a$ and let (y^*, z^*) be an optimal solution to the x^*-parameterized lower-level problem in (10.24), i.e., $\varphi(x^*) = d_y^\top y^* + d_z^\top z^*$. In particular, this means that $y^* \in Y$, $z^* \in Z$, $y^* \geq 0$, and $Dy^* + Ez^* \leq b$ holds. Hence, we have $(y^*, z^*) \in Y_z$, i.e., (y^*, z^*) is feasible for the problem

$$\max_{(y,z)\in Y_z} \quad \left(d_y^\top - (x_L^*)^\top \mathrm{Diag}(d_y)\right) y + d_z^\top z.$$

In addition, the constraints $y_i^* \leq u_i(1 - x_i^*)$ ensure that $x_i^* = 1$ implies $y_i^* = 0$ for all $i \in L$ and, thus, $(x_L^*)^\top \mathrm{Diag}(d_y) y^* = 0$. Overall, this yields $\varphi(x^*) \leq \varphi^{\mathrm{p}}(x^*)$.

Conversely, let $(\bar{y}, \bar{z})$ be an extreme point of the polytope Y_z that corresponds to an optimal solution to the lower-level problem in (10.25) for the given x^*. We now define a follower solution $(\hat{y}, \bar{z})$ with $\hat{y}_i = \bar{y}_i(1 - x_i^*)$ for all $i \in L$. Due to the downward monotonicity property, $(\hat{y}, \bar{z})$ is feasible for the follower problems in both (10.24) and (10.25) for the given x^*. In particular, $(\hat{y}, \bar{z})$ is an optimal solution to the lower-level problem in (10.25) for the given x^* because of Assumption 10.22 and $x^*, \bar{y}, \hat{y} \geq 0$. Moreover, it holds $(x_L^*)^\top \mathrm{Diag}(d_y)\hat{y} = 0$ by construction. Hence, we obtain

$$\varphi^{\mathrm{p}}(x^*) = (d_y^\top - (x_L^*)^\top \mathrm{Diag}(d_y))\hat{y} + d_z^\top \bar{z} = d_y^\top \hat{y} + d_z^\top \bar{z} \leq \varphi(x^*).$$

To sum up, we have shown $\varphi(x^*) \leq \varphi^{\mathrm{p}}(x^*) \leq \varphi(x^*)$, i.e, the objective function values of Problems (10.24) and (10.25) are the same. Because x^* has been chosen arbitrarily, this particularly holds for the optimal leader's decision. This concludes the proof. □

Corollary 10.26 *Suppose that Problem* (10.24) *satisfies Assumption* 10.22*. Then, we can equivalently reformulate it as*

$$\min_{x\in X, \theta} \quad \theta \tag{10.26a}$$

$$\text{s.t.} \quad \theta \geq \left(d_y^\top - x_L^\top \mathrm{Diag}(d_y)\right) \hat{y} + d_z^\top \hat{z} \quad \text{for all } (\hat{y}, \hat{z}) \in \hat{Y}_z, \tag{10.26b}$$

$$Ax \geq a. \tag{10.26c}$$

The constraints in (10.26b) are the strengthened and generalized version of the interdiction cuts previously introduced in (10.18b). First, they are stronger because the big-M coefficients are replaced with d_y. These tight values are guaranteed to be valid by the downward monotonicity property. Second, they are a generalization of the previously introduced interdiction cuts because they allow to handle extended formulations including variables z that are not directly linked to the leader.

The cuts (10.26b) can also be used within the node processing procedure of the generic branch-and-cut method given in Algorithm 9. They are separated by taking the solution $\tilde{x}^k$ to the relaxed master problem at node k of the branch-and-cut search tree and computing the optimal value $\varphi(\tilde{x}^k)$ of the follower. Let $(\tilde{x}^k, \tilde{\theta}^k)$ be the solution to the LP relaxation at node k. The separation problem then consists of solving

$$\max_{(y,z)\in\hat{Y}_z} \quad \sum_{i\in L} d_i^* y_i + d_z^\top z \tag{10.27}$$

with $d^* = (1-\tilde{x}_L^k)^\top \mathrm{Diag}(d_y)$. Let $(\hat{y}, \hat{z})$ be the computed solution. If $\tilde{\theta}^k < \varphi(\tilde{x}^k)$, then $(\hat{y}, \hat{z})$ produces a maximally-violated interdiction cut

$$\theta \geq \left(d_y^\top - x_L^\top \mathrm{Diag}(d_y)\right) \hat{y} + d_z^\top \hat{z}$$

that we then add to the relaxed master problem; see Steps 10 and 14 of Algorithm 9.

Exercise 10.27 Write a pseudo-code for the cutting-plane procedure that solves Problem (10.24) using a multi-tree implementation.

10.6.1 Interdiction Cuts Derived from Maximal Follower's Solutions

We now show that, among all extreme points $(\hat{y}, \hat{z}) \in \hat{Y}_z$ used for generating interdiction cuts, it is sufficient to consider so-called *maximal* points.

Definition 10.28 A point $(\hat{y}, \hat{z}) \in \hat{Y}_z$ is called *maximal* if there is no point $(\bar{y}, \hat{z}) \in \hat{Y}_z \setminus \{(\hat{y}, \hat{z})\}$ such that $\bar{y} \geq \hat{y}$ holds.

The notion of maximal points can be exploited when deriving separation procedures to avoid the generation of dominated interdiction cuts. Let us now formally define dominance between cuts.

Definition 10.29 Given two valid inequalities $\pi^\top x \leq \pi_0$ and $\tilde{\pi}^\top x \leq \tilde{\pi}_0$ for an optimization problem whose LP relaxation is described by a polyhedron P, we say that $\pi^\top x \leq \pi_0$ *dominates* $\tilde{\pi}^\top x \leq \tilde{\pi}_0$ if and only if

$$\pi^\top \hat{x} \leq \pi_0 \implies \tilde{\pi}^\top \hat{x} \leq \tilde{\pi}_0 \quad \text{for all } \hat{x} \in P.$$

We also say that $\tilde{\pi}^\top x \leq \tilde{\pi}_0$ is *dominated by* $\pi^\top x \leq \pi_0$.

In the context of dynamic cut separation, it is preferable to generate non-dominated inequalities because dominated cuts from earlier iterations become redundant and may unnecessarily overload the relaxed master problem. Note that entries $x_i^* = 1$ lead to zero-coefficients d_i^* in the objective function of the separation problem (10.27). Hence, an optimal solution $(\hat{y}, \hat{z})$ found by the separation procedure may not be maximal. This means that there could exist an alternative solution $(\bar{y}, \hat{z})$ such that $\bar{y} \neq \hat{y}$ and $\bar{y} \geq \hat{y}$, which is also optimal for (10.27). It turns out that maximal solutions can produce stronger cuts. The following theorem shows that interdiction cuts induced by non-maximal points from $\hat{Y}_z$ are dominated by the cuts induced by maximal points.

Theorem 10.30 *Let $(\hat{y}, \hat{z}) \in \hat{Y}_z$ be non-maximal and let $(\bar{y}, \hat{z}) \in \hat{Y}_z \setminus \{(\hat{y}, \hat{z})\}$ be such that $\bar{y} \geq \hat{y}$. Then, under Assumption 10.22, the interdiction inequality (10.26b) induced by $(\hat{y}, \hat{z})$ is dominated by the one induced by $(\bar{y}, \hat{z})$.*

Proof: For all $i \in L$, $x_i \in [0, 1]$ implies $\bar{y}_i(1 - x_i) \geq \hat{y}_i(1 - x_i)$. Therefore, for all $x \in X$ with $Ax \geq a$, we have

$$\theta \geq \left(d_y^\top - x_L^\top \mathrm{Diag}(d_y)\right) \bar{y} + d_z^\top \hat{z} \implies \theta \geq \left(d_y^\top - x_L^\top \mathrm{Diag}(d_y)\right) \hat{y} + d_z^\top \hat{z}. \quad \square$$

To favor maximal solutions to Problem (10.27), we slightly modify the separation procedure and perturb the objective function regarding the entries associated with the y variables. Each $d_i^* = 0$, $i \in L$, is replaced with a very small $\varepsilon > 0$ that ensures that the perturbation does not change the optimal solution of the starting problem. The choice of ε is problem-dependent. For the knapsack interdiction problem, for example, with n items and integer item prices, one can set $\varepsilon = 1/(n + 1)$. Alternatively, one can try to (heuristically) enlarge the optimal solution obtained by solving (10.27) until the maximality property is achieved.

Exercise 10.31 Derive a single-level reformulation using interdiction cuts for the following problems:

(i) The maximum-clique interdiction problem defined in Section 10.2.3.
(ii) The knapsack interdiction problem defined in Example 1.18.
(iii) The maximum-matching interdiction problem defined in Exercise 10.24.
(iv) The prize-collecting traveling salesperson interdiction problem defined in Exercise 10.24.
(v) The prize-collecting Steiner tree interdiction problem defined in Exercise 10.24.

Exercise 10.32 For each of the problems listed in Exercise 10.31:

(i) Explain the respective interdiction-cut separation procedure.
(ii) Discuss possible ways to guarantee that the resulting cut is induced by a maximal follower's solution.

10.7 Blocker Problems

In the interdiction problems studied so far, the leader has a limited interdiction budget and uses it so that the objective of the follower in the remaining structure is the worst possible. Closely related to interdiction problems are the so-called *blocker problems* in which the leader minimizes the cost of blocking (interdicting) the activities of the follower, while ensuring that the optimal follower's response is bounded by a user-defined threshold $r > 0$. Hence, we consider the problem

$$\begin{aligned} \min_{x \in X} \quad & c_x^\top x \\ \text{s.t.} \quad & \varphi(x) \le r, \\ & Ax \ge a, \end{aligned} \tag{10.28}$$

where the lower-level problem is defined as in (10.1). The blocking activities can be discrete or continuous and they can affect the follower's feasible region, the objective function, or both. Similarly, the lower-level problem can be continuous or discrete.

Exercise 10.33 (Minimum-Cost Vertex Blocker Clique Problem; see Pajouh et al. (2014)) Given an undirected and simple graph $G = (\mathcal{V}, \mathcal{E})$ with vertex weights $w_i > 0$, a maximum-weighted clique is a complete subgraph of G, determined by the set of vertices $K \subseteq \mathcal{V}$, that maximizes $\sum_{i \in K} w_i$. Given the blocking cost $c_i > 0$ for each vertex $i \in \mathcal{V}$, the minimum-cost vertex blocker clique problem seeks for a minimum-cost subset of vertices $S \subseteq \mathcal{V}$ to be removed from G so that the value of the maximum-weighted clique in the remaining graph is bounded from above by a given $r > 0$. Provide a bilevel MILP-MILP formulation of the problem.

Many other blocker variants of classic combinatorial optimization problems have been studied in the literature. For example, in the minimum-cost edge blocker clique problem, the leader blocks a minimum-cost subset of edges so that the weight of any clique in the remaining graph is bounded from above by r (Pajouh 2020). The blocking of vertices or edges has been studied with respect to other graph optimization problems such as maximum matchings (Zenklusen et al. 2009), shortest paths (Golden 1978), spanning trees (Bazgan et al. 2013), or dominating sets (Pajouh et al. 2015).

Remark 10.34 (Problem Complexity) Interdiction and blocker problems are two versions of the same decision problem. Therefore, a given blocker problem is in the same complexity class as its interdiction variant.

Exact MILP-based solution approaches for blocker problems share similarities with the methods derived for interdiction problems. For example, all reformulation strategies described above can be used for blocker problems as well. In the remainder of this section, we discuss a *set-covering* type formulation that leads to a tailored solution approach for blocker problems with binary interdiction variables.

10.7.1 A Set-Covering Type Formulation for Binary Blocker Problems

Let us now consider a discrete interdiction setting in which the lower-level problem is discrete as well, i.e., $x_i \in \{0,1\}$ and $y_i \in \{0,1\}$ for all $i \in L$. The leader blocks the resources of the follower, while minimizing the cost of blocking and ensuring that the solution of the follower does not pass a given threshold $r \in \mathbb{Z}$. Hence, we start with the bilevel formulation (10.28) in which x_i, $i \in L$, are binary variables associated with blocking activities of the leader. The lower-level problem is defined as

$$\begin{aligned} \varphi(x) := \max_{y \in Y} \quad & d^\top y \\ \text{s.t.} \quad & Dy \le b, \ y \ge 0, \\ & y_i \le 1 - x_i \quad \text{for all } i \in L, \end{aligned} \tag{10.29}$$

where the set Y contains integrality restrictions on all y variables. Note that, because the y variables are binary, we have upper bounds $u_i = 1$ for all $i \in L$.

Without loss of generality, we assume that the coefficients of d are integer. Let us now consider the set of all feasible points of the non-blocked lower-level problem, whose objective function value is strictly greater than the given threshold r:

$$\hat{\mathcal{Y}}(r) = \left\{ y \in Y : Dy \le b, \ y \ge 0, \ d^\top y \ge r + 1 \right\}.$$

Then, any feasible decision x of the leader must ensure that all points $\hat{y} \in \hat{\mathcal{Y}}(r)$ are blocked. Hence, we obtain the reformulation

$$\min_{x \in X} \quad c_x^\top x \tag{10.30a}$$

$$\text{s.t.} \quad \sum_{\{i \in L : \ \hat{y}_i = 1\}} x_i \ge 1 \quad \text{for all } \hat{y} \in \hat{\mathcal{Y}}(r), \tag{10.30b}$$

$$Ax \ge a \tag{10.30c}$$

of the blocker problem (Wei and Walteros 2022).

In general, Problem (10.30) contains an exponential number of constraints. We refer to an inequality of the type given in (10.30b) as a *set-covering constraint* induced by the point $\hat{y} \in \hat{\mathcal{Y}}(r)$. These constraints are globally valid.

Problem (10.30) can be implemented within a cutting-plane or a branch-and-cut framework, similarly to the ones applied to the case of interdiction cuts in Section 10.5. In fact, because both the interdiction cuts (10.18b) and the set-covering constraints allow to project out lower-level variables, these constraints can be combined in a single framework; see, e.g., Bentoumi et al. (2025). Moreover, Wei and Walteros (2022) provide necessary and sufficient conditions for the set-covering inequalities to be facet-defining.

Exercise 10.35 (Most Vital Vertices for the Shortest-Path Problem; see Magnouche and Martin (2020)) Assume we are given a directed graph $G = (\mathcal{V}, \mathcal{A})$ with a source $s \in \mathcal{V}$ and a destination $t \in \mathcal{V}$, $s \neq t$, so that every vertex $i \in \mathcal{V}$ has a blocking cost $c_i > 0$ and every arc $a \in \mathcal{A}$ has a length $d_a > 0$. The most vital vertices for the shortest-path problem seeks for a minimum-cost subset of vertices to be removed from G so that the length of the shortest path in the remaining graph is bounded from below by a given $r > 0$.

(i) Provide a bilevel MILP-MILP formulation of the problem.
(ii) Provide a single-level reformulation of the problem based on interdiction cuts. (*Hint:* For each arc $(u, v) \in \mathcal{A}$, introduce an auxiliary variable in the upper level that is set to one if and only if u or v is blocked. Use these variables as linking variables in the lower-level problem.) How would you choose the penalty parameters?

Similarly to the previous problem in which the leader blocks the vertices of G, we now consider a blocker variant of the shortest-path problem in which the arcs are blocked instead.

Example 10.36 (Most Vital Arcs for the Shortest-Path Problem) Let us consider the shortest-path blocker problem for the graph given in Figure 10.1 (top), in which the follower searches for a shortest path from vertex 1 to vertex 6. Suppose that the blocking cost for each arc is $c_a = 1$ and that the leader is searching for the minimum-cost subset of arcs to remove from G so that the length of the shortest path in the resulting graph is at least $r = 22$. To derive the set-covering problem reformulation, let us consider the set $\hat{\mathcal{Y}}(r)$, which contains all paths in G whose length is strictly less than 22. These are $\{1, 3, 2, 4, 6\}$ of length 17, $\{1, 3, 2, 5, 6\}$ of length 19, $\{1, 3, 4, 6\}$ of length 19, and $\{1, 2, 4, 6\}$

of length 21. Hence, the respective set-covering formulation is given by

$$\begin{aligned} \min_{x\in\{0,1\}^{|\mathcal{A}|}} \quad & \sum_{a\in\mathcal{A}} x_a \\ \text{s.t.} \quad & x_{13} + x_{32} + x_{24} + x_{46} \geq 1, \\ & x_{13} + x_{32} + x_{25} + x_{56} \geq 1, \\ & x_{13} + x_{34} + x_{46} \geq 1, \\ & x_{12} + x_{24} + x_{46} \geq 1. \end{aligned}$$

The optimal value of the problem is two, i.e., the leader needs to block two edges to ensure that all the remaining shortest paths in the network are of length 22 or larger. There are multiple optimal solutions that lead to this result. For example, the leader can block arcs $(1, 3)$ and $(4, 6)$, or $(3, 2)$ and $(4, 6)$, as well as $(1, 3)$ and $(1, 2)$, etc. △

Exercise 10.37 Provide the set-covering formulation for the most vital vertices for the shortest-path problem defined in Exercise 10.35.

Exercise 10.38 For the minimum-cost vertex blocker clique problem defined in Exercise 10.33:

(i) Provide a single-level reformulation based on interdiction cuts. How would you choose the penalty parameters?
(ii) Provide a single-level reformulation based on set-covering cuts.
(iii) Write down the set-covering formulation for the example given in Figure 10.2, assuming that $w_1 = w_2 = w_5 = w_6 = 1$ and $w_i = 2$ for all the remaining nodes as well as $c_i = 1$ for all $i \in \mathcal{V}$ and $r = 7$. What is the optimal value of the problem? Are there multiple optimal solutions?

Separation of Set-Covering Inequalities (10.30b)

We now discuss the procedure to separate the set-covering inequalities (10.30b). To this end, we focus on the multi-tree implementation (Algorithm 8) in which the solution x^* to the relaxed master problem is binary. To determine whether x^* is a feasible blocker solution or whether there exists another violated inequality of type (10.30b) that has to be inserted into the model, we need to solve the follower's optimization problem (10.29) associated with the blocking strategy x^*. If for the resulting value it holds $\varphi(x^*) \leq r$, then x^* is an optimal solution and the algorithm terminates; see also Algorithm 8. Otherwise, let $\hat{y}$ be the respective optimal solution to the lower-level problem. The cut we need to insert to the relaxed master problem is then given by

$$\sum_{\{i\in L:\ \hat{y}_i=1\}} x_i \geq 1.$$

Enhancements such as deriving cuts from heuristic (rather than optimal) lower-level solutions can be incorporated for this model as well.

Exercise 10.39 For the single-tree implementation of Model (10.30):

(i) Write down a pseudo-code for processing node k of the branch-and-cut search tree. (*Hint:* You may want to have another look at Algorithm 9.)
(ii) Prove that, if we embed the algorithm from (i) into a classic branch-and-bound framework (see Algorithm 5), we obtain a method that terminates with an optimal solution to Problem (10.30) after adding an overall finite number of set-covering constraints. (*Hint:* See also Exercise 10.17.)

Remark 10.40 Interestingly, even for polynomial lower-level problems (such as, e.g., the shortest-path problem), the separation of fractional solutions x^* to the relaxed master problem may become NP-hard (Magnouche and Martin 2020).

10.8 Critical Vertex or Edge Detection Problems in Networks

We conclude this section by observing that many other problems in graph theory and network analysis can be seen through the lens of interdiction or blocker problems. The leader tries to interdict the network by removing vertices or edges from the network, while the follower optimizes some graph-functionality measure. Graph functionality can be measured as the total number of pair-wise connected vertices (Arulselvan et al. 2009; Di Summa et al. 2012), the number of connected components or the size of the largest connected component (Furini et al. 2022; Shen and Smith 2012; Shen et al. 2012), or the size of its k-core (Cerulli et al. 2023). For further reading, we recommend the surveys by Beck et al. (2023), Kleinert et al. (2021a), Lalou et al. (2018), and Smith and Song (2020).

10.9 What You Should Know Now!

1. What are major differences between interdiction problems and more general bilevel problems?
2. What kind of interdiction activities are usually considered in applications?
3. What are typical single-level reformulation techniques for interdiction problems?
4. What are typical solution methods for interdiction problems?

5. How can we derive a Benders-like single-level reformulation for interdiction problems?
6. What are the main differences between the single-tree and the multi-tree implementation of the Benders-like problem reformulation?
7. What are the so-called interdiction cuts and how can we separate them within a branch-and-cut procedure?
8. What is the downward monotonicity property and why is it relevant for interdiction problems?
9. How can we strengthen interdiction cuts if the downward monotonicity property holds?
10. How can we separate interdiction cuts heuristically when the downward monotonicity property holds?
11. What is the difference between interdiction and blocker problems and why do their single-level reformulations and exact solution methods rely on similar ideas?
12. How can we derive a set-covering formulation for blocker problems?

11

Excursus:
Intersection Cuts for Single-Level MILPs

A large body of work in mixed-integer linear optimization is dedicated to the study of valid inequalities—both from a theoretical and a computational perspective. In particular, general-purpose cutting planes, i.e., cuts that can be derived for an arbitrary MILP, have attracted a lot of attention in the MILP community. Gomory's fractional and mixed-integer cuts (Gomory 1958, 1960) have initiated this research and the discovery of many additional families of general-purpose cuts, including intersection cuts (Balas 1971), disjunctive cuts (Balas 2018), split cuts (Cook et al. 1990), mixed-integer-rounding cuts (Nemhauser and Wolsey 1990), lift-and-project cuts (Balas 2018; Balas and Perregaard 2002), or $\{0, 1/2\}$-cuts (Caprara and Fischetti 1996), to mention a few. For a tutorial on general-purpose cutting planes for MILPs, we refer to Cornúejols (2008) or Conforti et al. (2014).

In this chapter, we focus on one particular family of general-purpose cuts, namely intersection cuts. These cuts are very useful for solving MILP-MILP bilevel problems, which we will discuss in detail in Chapter 12. However, as their derivation is somewhat involved, we first discuss them here for single-level MILPs before we turn our attention to bilevel MILPs again in the next chapter.

Let us consider single-level mixed-integer linear problems of the form

$$\begin{aligned} \max_{x,y} \quad & \hat{c}^\top x + \hat{d}^\top y \\ \text{s.t.} \quad & \hat{A}x + \hat{B}y = b, \\ & (x, y) \in \mathbb{Z}^{n_x}_{\geq 0} \times \mathbb{R}^{n_y}_{\geq 0} \end{aligned} \tag{11.1}$$

with $\hat{A} \in \mathbb{Q}^{m \times n_x}$, $\hat{B} \in \mathbb{Q}^{m \times n_y}$, $b \in \mathbb{Q}^m$, $\hat{c} \in \mathbb{Q}^{n_x}$, $\hat{d} \in \mathbb{Q}^{n_y}$, $m < n_x + n_y$, and the matrix $[\hat{A}, \hat{B}]$ having full row rank m. The continuous variables y are assumed to be non-negative and the integer variables x are assumed to be non-negative and bounded. For the ease of presentation, we assume that any variable bounds are encoded in the linear system $\hat{A}x + \hat{B}y = b$. Let

$$P = \left\{(x, y) \in \mathbb{R}^{n_x}_{\geq 0} \times \mathbb{R}^{n_y}_{\geq 0} : \hat{A}x + \hat{B}y = b\right\}$$

be the set of feasible points for the continuous relaxation of Problem (11.1). Note that the set P is a polyhedron as it is obtained as an intersection of finitely many half-spaces. We now introduce some more notation and terminology. Let $P_I = P \cap (\mathbb{Z}^{n_x}_{\geq 0} \times \mathbb{R}^{n_y}_{\geq 0})$ be the set of feasible points of Problem (11.1) and let $n = n_x + n_y$. The convex hull of P_I, denoted as $\operatorname{conv}(P_I)$, is the set of all points that can be expressed as convex combinations of finitely many points from P_I. An inequality $\alpha^\top x + \beta^\top y \leq \gamma$ is called a valid inequality for Problem (11.1) if it is satisfied by every feasible point $(x, y) \in P_I$; see Chapter 9 for further details. Cutting planes are inequalities that are valid for $\operatorname{conv}(P_I)$ but violated by some $(x, y) \in P \setminus \operatorname{conv}(P_I)$.

11.1 The LP Cone Associated with a Basic Solution

We now reformulate Problem (11.1) using variables $\xi = (x, y)$ as

$$\begin{aligned} \min_{\xi} \quad & c^\top \xi \\ \text{s.t.} \quad & A\xi = b, \\ & \xi \in \mathbb{Z}^{n_x}_{\geq 0} \times \mathbb{R}^{n_y}_{\geq 0} \end{aligned} \tag{11.2}$$

with $c = (\hat{c}, \hat{d})$ and $A = [\hat{A}, \hat{B}]$. The polyhedron of the LP relaxation can now be expressed in the space of ξ-variables as

$$P = \left\{\xi \in \mathbb{R}^n_{\geq 0} : A\xi = b\right\}.$$

When solving the LP relaxation of (11.2) with the simplex method, we can associate a basis $\mathcal{B} \subset \{1, \dots, n\}$, i.e., a set of m linearly independent columns of A, with each vertex ξ^* of P so that the components of ξ^* can be represented as

$$\xi^*_i = \begin{cases} \bar{b}_i, & i \in \mathcal{B}, \\ 0, & i \in \mathcal{N}, \end{cases}$$

where $\mathcal{N} = \{1, \dots, n\} \setminus \mathcal{B}$ is the set of indices of non-basic variables, $\bar{b} = A_{\mathcal{B}}^{-1} b$, and $A_{\mathcal{B}}$ is the submatrix of A induced by the columns indexed by $\mathcal{B}$. Similarly, let $A_{\mathcal{N}}$ be the submatrix of A induced by the columns indexed by $\mathcal{N}$ and let $\xi = (\xi_{\mathcal{B}}, \xi_{\mathcal{N}})$. The associated simplex tableau is given by

$$\begin{aligned} \xi_{\mathcal{B}} &= \bar{b} - A_{\mathcal{B}}^{-1} A_{\mathcal{N}} \xi_{\mathcal{N}}, \\ \xi &\geq 0. \end{aligned}$$

We now associate a pointed polyhedral cone, which we denote as $C(\xi^*)$, with every basic solution ξ^* to the LP relaxation of (11.2). The set $C(\xi^*)$ is obtained from the simplex tableau by omitting the condition $\xi_{\mathcal{B}} \geq 0$. The apex of $C(\xi^*)$ is ξ^* and its facets are defined by the m hyperplanes

$$\xi_i = \bar{b}_i - \sum_{j \in \mathcal{N}} \bar{a}_{ij} \xi_j \quad \text{for all } i \in \mathcal{B},$$

which correspond to the LP basis of ξ^*. Here, $\bar{a}_{ij}$ are the entries of $A_{\mathcal{B}}^{-1} A_{\mathcal{N}}$. More precisely, $C(\xi^*)$ is the set of points in $\mathbb{R}^n$ satisfying the system of linear equations and inequalities given by

$$\begin{aligned} \xi_{\mathcal{B}} &= \bar{b} - A_{\mathcal{B}}^{-1} A_{\mathcal{N}} \xi_{\mathcal{N}}, \\ \xi_{\mathcal{N}} &\geq 0. \end{aligned}$$

Example 11.1 Let us consider the mixed-integer linear optimization problem

$$\begin{aligned} \min_{x \in \mathbb{Z}, y \in \mathbb{R}} \quad & -x - y \\ \text{s.t.} \quad & 2x + y \leq 4, \\ & x + 2y \leq 4, \\ & x, y \geq 0. \end{aligned} \tag{11.3}$$

After relaxing the integrality constraint on x and adding slack variables $s = (s_1, s_2) \geq 0$ associated with the first two constraints, we obtain the continuous linear problem

$$\min_{x,y,s} \quad -x - y \tag{11.4a}$$
$$\text{s.t.} \quad 2x + y + s_1 = 4, \tag{11.4b}$$
$$x + 2y + s_2 = 4, \tag{11.4c}$$
$$x, y, s_1, s_2 \geq 0. \tag{11.4d}$$

Hence, we have $\xi = (x, y, s_1, s_2)$ and $n = 4$. The solution $\xi^* = (4/3, 4/3, 0, 0)$ to Problem (11.4) is fractional, we have $\mathcal{B} = \{1, 2\}$, and the associated simplex tableau is given by

$$\xi_1 = \frac{4}{3} - \frac{2}{3}\xi_3 + \frac{1}{3}\xi_4, \tag{11.5a}$$
$$\xi_2 = \frac{4}{3} + \frac{1}{3}\xi_3 - \frac{2}{3}\xi_4, \tag{11.5b}$$
$$\xi \geq 0. \tag{11.5c}$$

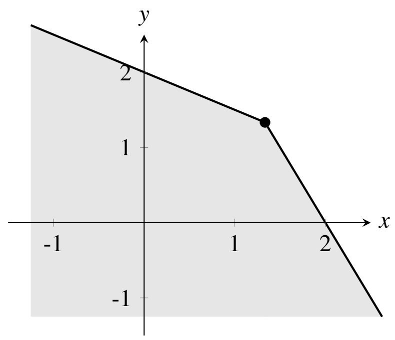


Figure 11.1 An illustration of the polyhedral cone $C(x^*, y^*)$ pointed at $(4/3, 4/3)$ in Example 11.1.

The cone $C(\xi^*)$ is the set of points in $\mathbb{R}^4$ satisfying the following system

$$\begin{aligned}\xi_1 &= \frac{4}{3} - \frac{2}{3}\xi_3 + \frac{1}{3}\xi_4,\\ \xi_2 &= \frac{4}{3} + \frac{1}{3}\xi_3 - \frac{2}{3}\xi_4,\\ \xi_3, \xi_4 &\geq 0.\end{aligned}$$

Projected to the (x, y)-variable space, this cone can be expressed as

$$C(x^*, y^*) = \left\{(x, y) \in \mathbb{R}^2 : 2x + y \leq 4,\ x + 2y \leq 4\right\}.$$

This set, which is illustrated in Figure 11.1, contains two extreme rays pointed at $(x^*, y^*) = (4/3, 4/3)$. The directions of these rays are $(-2/3, 1/3)$ and $(1/3, -2/3)$, respectively. △

11.2 Intersection Cuts

Intersection cuts are typically used to discard the current LP solution ξ^* if it violates some of the integrality conditions. They are derived by intersecting the cone $C(\xi^*)$ with another convex set $\mathcal{T}$, whose interior $\mathrm{int}(\mathcal{T})$ contains ξ^* but no points from P_I. Hence, the region $P \cap \mathrm{int}(\mathcal{T})$ can be cut off without discarding any integer-feasible point. Roughly speaking, we intersect the $n - m$ extreme rays of the cone $C(\xi^*)$ with the boundary of $\mathcal{T}$. The resulting hyperplane that then contains the $n - m$ intersection points is called the intersection cut.

Example 11.2 We revisit Problem (11.3) given in Example 11.1 and consider

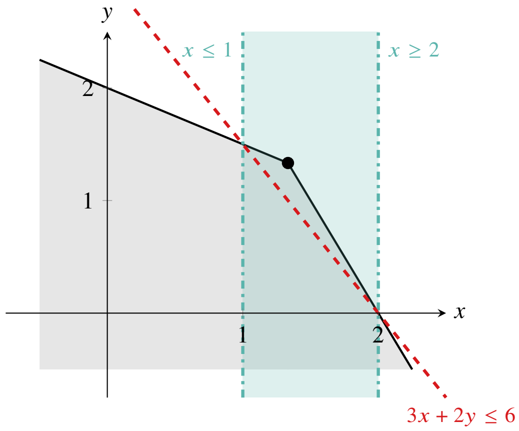


Figure 11.2 An illustration of the intersection cut in Example 11.2. The LP cone pointed at $(4/3, 4/3)$ is the gray area, the split disjunction is represented by the dash-dotted green lines, and the intersection cut $3x + 2y \le 6$ is the dashed red line. The optimal solution $(4/3, 4/3)$ to the LP relaxation of Problem (11.3) is cut off by the intersection cut, which is obtained by intersecting the extreme rays of the cone with the boundary of the set $\mathcal{T} = \{(x, y) \in \mathbb{R}^2 : 1 \le x \le 2\}$. The latter set corresponds to the green shaded area between the dash-dotted lines.

a so-called *split disjunction*

$$x \le 1 \quad \text{or} \quad x \ge 2$$

on x, which is derived from the fact that the variable x must be integer. Moreover, we consider the set $\mathcal{T} = \{(x, y) \in \mathbb{R}^2 : 1 \le x \le 2\}$. The interior of this set can be cut off from the polyhedron of the LP relaxation of Problem (11.3) without discarding any feasible point. Intersecting the extreme rays of the cone $C(x^*, y^*)$ with the boundary of $\mathcal{T}$ results in two points, namely $(2, 0)$ and $(1, 3/2)$, as illustrated in Figure 11.2. The hyperplane $3x + 2y \le 6$ through these two points is the resulting intersection cut that, when added to the LP relaxation of Problem (11.3), removes $(x^*, y^*) = (4/3, 4/3)$ from the feasible set without discarding any feasible point of the problem. △

In what follows, we first prove that intersection cuts are valid inequalities for Problem (11.2) whenever they are derived from an arbitrary closed convex set $\mathcal{T}$ whose interior contains no points from P_I. In Section 11.3, we then discuss how to calculate the coefficients of the intersection cut efficiently in the case in which $\mathcal{T}$ is a polyhedron. Section 11.4 is devoted to obtaining intersection cuts using disjunctive arguments.

Note that extreme rays of $C(\xi^*)$ are one-dimensional faces of $C(\xi^*)$. Hence, their directions can be easily computed by setting all but one inequality from $C(\xi^*)$ to equality; see Proposition B.1. Thus, we have $n - m$ extreme rays, $r^j \in \mathbb{R}^n$, $j \in \mathcal{N}$, that are given by

$$r_i^j = \begin{cases} -\bar{a}_{ij}, & i \in \mathcal{B}, \\ 1, & i = j, \\ 0, & i \in \mathcal{N} \setminus \{j\}, \end{cases} \tag{11.6}$$

for all $i \in \{1, \dots, n\}$.

Let now r^j be such an extreme ray of $C(\xi^*)$ defined by (11.6). To find the intersection point of r^j with the boundary of $\mathcal{T}$, we need to derive

$$\lambda_j^* = \max \left\{\lambda_j \geq 0\colon \xi^* + \lambda_j r^j \in \mathcal{T}\right\}, \tag{11.7}$$

which determines the furthest point on the half-line $\xi^* + \lambda_j r^j$ with $\lambda_j \geq 0$ that still belongs to $\mathcal{T}$. Given that $\xi^* \in \operatorname{int}(\mathcal{T})$, it follows $\lambda_j^* > 0$. If the half-line intersects the boundary of $\mathcal{T}$, the segment $\{\xi^* + \lambda_j r^j : 0 \leq \lambda_j < \lambda_j^*\}$ belongs to the interior of $\mathcal{T}$ and the value of λ_j^* is finite. However, if r^j belongs to the recession cone of $\mathcal{T}$ (see Appendix B), then $\lambda_j^* = \infty$ holds. Let

$$\Lambda_{\mathrm{p}} = \left\{j \in \mathcal{N}\colon \lambda_j^* < \infty\right\} \quad \text{and} \quad \Lambda_{\mathrm{r}} = \left\{j \in \mathcal{N}\colon \lambda_j^* = \infty\right\}$$

be the index sets determining the intersection points and the extreme rays that belong to the recession cone of $\mathcal{T}$, respectively. For what follows, we define $1/\infty := 0$ and use the following result; see also Theorem 1.1 in Balas (2018) or Theorem 6.5 in Conforti et al. (2014).

Proposition 11.3 (Balas (1971)) *For $j \in \mathcal{N}$, let λ_j^* be obtained as in (11.7). Then, the intersection cut*

$$\sum_{j \in \mathcal{N}} \frac{1}{\lambda_j^*} \xi_j \geq 1 \tag{11.8}$$

cuts off ξ^* but no point in P_I.

Proof: Evaluating the left-hand side of (11.8) at the point ξ^* yields zero. Hence, Constraint (11.8) cuts off ξ^*. To see that no point of P_I is cut off, let us consider the polyhedron obtained by intersecting the polyhedral cone $C(\xi^*)$ with the half-space induced by $\sum_{j \in \mathcal{N}} \xi_j / \lambda_j^* \leq 1$, i.e.,

$$\bar{C}(\xi^*) = C(\xi^*) \cap \left\{\xi \in \mathbb{R}^n\colon \sum_{j \in \mathcal{N}} \frac{1}{\lambda_j^*} \xi_j \leq 1\right\},$$

and the set of points

$$\bar{C}'(\xi^*) = C(\xi^*) \cap \left\{ \xi \in \mathbb{R}^n : \sum_{j \in N} \frac{1}{\lambda_j^*} \xi_j < 1 \right\}$$

that are cut off by the intersection cut. We only have to show that $\bar{C}'(\xi^*)$ belongs to the interior of $\mathcal{T}$. The result then follows from the fact that, by definition, the set $\mathcal{T}$ contains no point of P_I in its interior.

Because the set $\bar{C}(\xi^*)$ is a polyhedron, any point belonging to $\bar{C}(\xi^*)$ can be represented as a convex combination of its vertices plus a conic combination of its extreme rays; see Proposition B.3. The vertices of $\bar{C}(\xi^*)$, apart from ξ^*, lie on the hyperplane induced by $\sum_{j \in N} \xi_j / \lambda_j^* = 1$. These vertices are the points $\xi^* + \lambda_j^* r^j$ for $j \in \Lambda_{\mathrm{p}}$ and the extreme rays of $\bar{C}(\xi^*)$ are r^j for $j \in \Lambda_{\mathrm{r}}$. Let us consider an arbitrary point $\xi' \in \bar{C}'(\xi^*)$ with $\xi' \neq \xi^*$. Each such point can be represented as a convex combination of points $\hat{\xi}^j = \xi^* + \hat{\lambda}_j r^j$ for some $0 \leq \hat{\lambda}_j < \lambda_j^*$, $j \in \Lambda_{\mathrm{p}}$, plus a conic combination of extreme rays r^j, $j \in \Lambda_{\mathrm{r}}$. The set $\mathcal{T}$ is a closed convex set and all points $\hat{\xi}^j$ are in the interior of $\mathcal{T}$, whereas the rays r^j, $j \in \Lambda_{\mathrm{r}}$, belong to the recession cone of $\mathcal{T}$. Therefore, the point ξ' belongs to $\operatorname{int}(\mathcal{T})$, which concludes the proof. □

Note that, due to the definition of the λ^* values in (11.7) and because we only have equality and non-negativity constraints in (11.2), the intersection cut (11.8) has only non-negative coefficients.

11.3 Computing Cut Coefficients for the Polyhedral Case

The above procedure for computing the coefficients of an intersection cut and, in particular, the formula in (11.7) are generic and can be applied to any convex set $\mathcal{T}$. However, there is a closed formula that allows to easily derive intersection cuts in the case in which the set $\mathcal{T}$ is a polyhedron. So let us assume now that $\mathcal{T}$ is a polyhedron described by K inequalities

$$\mathcal{T} = \left\{ \xi \in \mathbb{R}^n : g_k^\top \xi \leq g_{0k},\ k \in \{1, \dots, K\} \right\}, \tag{11.9}$$

where the entries of the vectors g_k and the scalars g_{0k} for $k \in \{1, \dots, K\}$ are rational numbers. As before, we assume that $\xi^* \in \operatorname{int}(\mathcal{T})$ and $\operatorname{int}(\mathcal{T}) \cap P_I = \emptyset$.

To determine the intersection points of the extreme rays of the cone $C(\xi^*)$ with each of the hyperplanes of $\mathcal{T}$, we first compute

$$\lambda_{jk}^* = \max \left\{ \lambda : g_k^\top (\xi^* + \lambda r^j) \leq g_{0k},\ \lambda \geq 0 \right\}. \tag{11.10}$$

Hence, we have

$$\lambda^*_{jk} = \max\left\{0, \frac{g_{0k} - g_k^\top \xi^*}{g_k^\top r^j}\right\}, \quad k \in \{1, \dots, K\},\ j \in \mathcal{N}.$$

As discussed in the previous section, if the half-line associated to the ray r^j, $j \in \mathcal{N}$, intersects the kth hyperplane of $\mathcal{T}$, the value of λ^*_{jk} is finite, otherwise $\lambda^*_{jk} = \infty$ holds. Consequently, if there exists a point at which the ray r^j touches the boundary of $\mathcal{T}$, it is the point closest to ξ^* among all the intersection points computed above (see, e.g., Balas (2018), Chapter 11.5):

$$\lambda^*_j = \min\left\{\lambda^*_{jk} \colon k \in \{1, \dots, K\} \text{ with } \lambda^*_{jk} > 0\right\} \quad \text{for all } j \in \mathcal{N}.$$

However, if r^j belongs to the recession cone of $\mathcal{T}$, we have $\lambda^*_{jk} = \infty$ for all $k \in \{1, \dots, K\}$. Therefore, λ^*_j is equal to ∞ and the associated coefficient of the intersection cut, which is equal to $1/\lambda^*_j$, is set to zero as discussed in the previous section.

From a practical perspective, the possibility to use this closed form to compute intersection cuts makes them more attractive compared to the more general disjunctive cuts, which usually require solving a cut-generating linear problem instead.

Example 11.4 The expressions given in the simplex tableau (11.5) define two rays of the feasible region at the LP solution $\xi^* = (4/3, 4/3, 0, 0)$ for $\mathcal{N} = \{3, 4\}$, namely

$$r^3 = \left(-\tfrac{2}{3}, \tfrac{1}{3}, 1, 0\right) \quad \text{from increasing } \xi_3,$$
$$r^4 = \left(\tfrac{1}{3}, -\tfrac{2}{3}, 0, 1\right) \quad \text{from increasing } \xi_4.$$

We now consider the convex set

$$\mathcal{T} = \left\{\xi \in \mathbb{R}^4 \colon 1 \le \xi_1 \le 2\right\},$$

which contains ξ^* in its interior but no points from P_I. Let us express the first and the second inequality of the set $\mathcal{T}$ using $g_1 = (-1, 0, 0, 0)$, $g_{01} = -1$, and $g_2 = (1, 0, 0, 0)$, $g_{02} = 2$, respectively. When calculating λ^* according to (11.10), we obtain

$$\lambda^*_{31} = 1/2 \text{ and } \lambda^*_{32} = 0 \implies \lambda^*_3 = 1/2$$
$$\lambda^*_{41} = 0 \text{ and } \lambda^*_{42} = 2 \implies \lambda^*_4 = 2.$$

Hence, the resulting intersection cut reads

$$2\xi_3 + \frac{1}{2}\xi_4 \ge 1.$$

We can express it in the (x, y)-space by substituting the slack variables according to (11.4b) and (11.4c) so that we obtain

$$2(4 - 2x - y) + \frac{1}{2}(4 - x - 2y) \geq 1 \iff 3x + 2y \leq 6;$$

see Figure 11.2. △

In the following, we discuss an alternative way to derive intersection cuts.

11.4 Using Disjunctive Arguments to Derive Intersection Cuts

As above, let us assume that $\mathcal{T}$ is a polyhedron described by (11.9) such that $\xi^* \in \operatorname{int}(\mathcal{T})$ and $\operatorname{int}(\mathcal{T}) \cap P_I = \emptyset$. We now explain how intersection cuts can be derived using disjunctive arguments.

An intersection cut violated by ξ^* is derived from the feasibility condition "ξ cannot belong to the interior of $\mathcal{T}$". This condition can also be restated using a K-term disjunction:

$$\xi \in P_I \implies \bigvee_{k=1}^{K} \left(g_k^\top \xi \geq g_{0k}\right). \tag{11.11}$$

Note that a feasible ξ can belong to the boundary of $\mathcal{T}$, which is why we may state these disjuncts using "$\geq$" instead of "$>$". Hence, we consider the disjuncts

$$\mathcal{D}_k = P \cap \left\{\xi \in \mathbb{R}^n : g_k^\top \xi \geq g_{0k}\right\}, \quad k \in \{1, \dots, K\},$$

and observe that

$$P_I \subseteq \bigcup_{k=1}^{K} \mathcal{D}_k \subseteq P.$$

For what follows, we need the next technical result.

Proposition 11.5 *Let $\pi_k^\top \xi \geq \pi_{0k}$ be a valid inequality for disjunct $\mathcal{D}_k$ for all $k \in \{1, \dots, K\}$. Then, for any $\xi \in \bigcup_{k=1}^{K} \mathcal{D}_k$, the following inequality holds*

$$\sum_{j \in \{1,\dots,n\}} \max_{k \in \{1,\dots,K\}} \left\{\pi_{kj}\right\} \xi_j \geq \min_{k \in \{1,\dots,K\}} \{\pi_{0k}\}.$$

Proof: Let $\xi \in \bigcup_{k=1}^{K} \mathcal{D}_k$ be given. Then, by definition, $\xi \in \mathcal{D}_{\bar{k}}$ holds for some $\bar{k} \in \{1, \dots, K\}$. Because $\pi_{\bar{k}}^\top \xi \geq \pi_{0\bar{k}}$ is valid for $\mathcal{D}_{\bar{k}}$, we have

$$\sum_{j \in \{1,\dots,n\}} \pi_{\bar{k}j} \xi_j \geq \pi_{0\bar{k}}.$$

Let us define $\pi^*_j = \max_{k\in\{1,\ldots,K\}} \left\{\pi_{kj}\right\}$ for all $j \in \{1, \ldots, n\}$. Because $\xi \geq 0$, we have

$$\pi^*_j \geq \pi_{\bar{k}j} \implies \pi^*_j \xi_j \geq \pi_{\bar{k}j}\xi_j \quad \text{for all } j \in \{1, \ldots, n\}.$$

Summing over all $j \in \{1, \ldots, n\}$, we get

$$\sum_{j\in\{1,\ldots,n\}} \pi^*_j \xi_j \geq \sum_{j\in\{1,\ldots,n\}} \pi_{\bar{k}j}\xi_j \geq \pi_{0\bar{k}} \geq \min_{k\in\{1,\ldots,K\}} \{\pi_{0k}\}$$

and, thus,

$$\sum_{j\in\{1,\ldots,n\}} \max_{k\in\{1,\ldots,K\}} \left\{\pi_{kj}\right\} \xi_j \geq \min_{k\in\{1,\ldots,K\}} \{\pi_{0k}\}$$

holds for all $\xi \in \bigcup_{k=1}^K \mathcal{D}_k$. □

An intersection cut derived from the set $\mathcal{T}$ (as defined in Section 11.3) and a vertex ξ^* of P associated with the basis $\mathcal{B}$ can then be obtained by the following procedure.

(i) For $k \in \{1, \ldots, K\}$, we first re-write $g_k^\top \xi \geq g_{0k}$ by expressing it in terms of non-basic variables. Thus, we obtain the cut in its equivalent "reduced form" with respect to the basis $\mathcal{B}$, i.e.,

$$\bar{g}_k^\top \xi \geq \bar{g}_{0k}$$

with

$$\begin{aligned}\bar{g}_k^\top &= g_k^\top - g_{k_\mathcal{B}}^\top A_\mathcal{B}^{-1} A,\\ \bar{g}_{0k} &= g_{0k} - g_{k_\mathcal{B}}^\top A_\mathcal{B}^{-1} b.\end{aligned}$$

Here and in what follows, $g_{k_\mathcal{B}}$ and $g_{k_\mathcal{N}}$ indicate entries of g_k associated with variables indexed by $\mathcal{B}$ and $\mathcal{N}$, respectively. Note that $\bar{g}_{k_\mathcal{B}} = 0$ holds. Hence, the cut can also be restated as

$$\bar{g}_{k_\mathcal{N}}^\top \xi_\mathcal{N} \geq \bar{g}_{0k}.$$

(ii) Observe that $\bar{g}_k^\top \xi \geq \bar{g}_{0k}$ is violated by ξ^* and $\bar{g}_k^\top \xi^* = 0$ holds by construction. Thus, $\bar{g}_{0k} > 0$ holds. Hence, we can normalize the above inequalities, leading to

$$\sum_{j\in\mathcal{N}} \frac{\bar{g}_{kj}}{\bar{g}_{0k}} \xi_j \geq 1, \quad k \in \{1, \ldots, K\}.$$

(iii) Therefore, the disjunction in (11.11) can be replaced by

$$\bigvee_{k=1}^K \left(\frac{\bar{g}_{k_\mathcal{N}}^\top \xi_\mathcal{N}}{\bar{g}_{0k}} \geq 1 \right).$$

Finally, due to Proposition 11.5, for each variable ξ_j, $j \in \mathcal{N}$, we take the maximum left-hand side coefficient. The resulting cut is then given by

$$\sum_{j \in \mathcal{N}} \max_{k \in \{1,\dots,K\}} \left\{ \frac{\bar{g}_{kj}}{\bar{g}_{0k}} \right\} \xi_j \geq 1. \tag{11.12}$$

As the max-operation does not change the coefficient of each basic variable ξ_j, $j \in \mathcal{B}$, which is zero in all the above constraints, the resulting cut is still violated (by 1) by ξ^*.

(iv) In case there exists some $j \in \mathcal{N}$ such that $\max_{k \in \{1,\dots,K\}}\{\bar{g}_{kj}/\bar{g}_{0k}\} > 1$ and if ξ_j is an integer-constrained variable, we can apply coefficient clipping and replace the corresponding coefficient in (11.12) by 1. The final form of the intersection cut is then given by

$$\sum_{j \in \mathcal{N}} \gamma_j \xi_j \geq 1,$$

where, for each $j \in \mathcal{N}$, we have

$$\gamma_j = \begin{cases} \min\left\{1, \max_{k \in \{1,\dots,K\}} \left\{ \frac{\bar{g}_{kj}}{\bar{g}_{0k}} \right\}\right\}, & \text{if } \xi_j \text{ is constrained to be integer,} \\ \max_{k \in \{1,\dots,K\}} \left\{ \frac{\bar{g}_{kj}}{\bar{g}_{0k}} \right\}, & \text{otherwise.} \end{cases}$$

The validity of the resulting inequality follows from the fact that, if ξ_j is constrained to be integer, either $\xi_j = 0$ or $\xi_j \geq 1$ holds. In the former case, the coefficient γ_j does not play any role. In the latter case, the validity follows from $\gamma \geq 0$ and $\xi \geq 0$.

Example 11.6 Let us consider Problem (11.3) again, its LP relaxation in standard form, and the simplex tableau given by (11.5). The two disjuncts resulting from the set $\mathcal{T} = \{(x, y) \in \mathbb{R}^2 : 1 \leq x \leq 2\}$ are given by

$$\mathcal{D}_1 = P \cap \left\{\xi \in \mathbb{R}^4 : -\xi_1 \geq -1\right\} \quad \text{and} \quad \mathcal{D}_2 = P \cap \left\{\xi \in \mathbb{R}^4 : \xi_1 \geq 2\right\}.$$

After expressing the inequalities $-\xi_1 \geq -1$ and $\xi_1 \geq 2$ in terms of non-basic variables using (11.5a) and normalizing the right-hand side, we obtain

$$2\xi_3 - \xi_4 \geq 1,$$
$$-\xi_3 + \frac{1}{2}\xi_4 \geq 1.$$

Finally, by applying Proposition 11.5, the resulting intersection cut reads

$$2\xi_3 + \frac{1}{2}\xi_4 \geq 1.$$

Because ξ_3 and ξ_4 correspond to slack variables and are thus not integer variables, no coefficient clipping can be applied. Finally, after expressing the above constraint in terms of (x, y)-variables, we again obtain the cut $3x + 2y \leq 6$; see also Example 11.4. △

Exercise 11.7 Find an optimal solution to the LP relaxation of the problem

$$\begin{aligned}\min_{x \in \mathbb{Z}, y \in \mathbb{R}} \quad & -3x - 2y \\ \text{s.t.} \quad & x + y \leq 3, \\ & 2x + y \leq 4, \\ & x, y \geq 0.\end{aligned}$$

If this solution is not integer feasible, derive an intersection cut that cuts off this point by using

(i) the procedure described in Section 11.3;
(ii) the procedure described in Section 11.4.

Provide a graphical illustration of the derived intersection cut in $\mathbb{R}^2$.

Let us close this chapter with a brief remark on implementing intersection cuts. For the ease of presentation, we introduced intersection cuts for problems in standard form, i.e., we only have equality and non-negativity constraints. However, modern mixed-integer programming solvers such as those mentioned in Section 3.4 tackle more general forms of (MI)LPs having equality and inequality constraints as well as general lower and upper bounds on the variables. This has an influence on the representation of the LP basis (see, e.g., Chapter 8 in Chvátal (1983)), which has to be taken into account when implementing intersection cuts in practice.

11.5 What You Should Know Now!

1. How is the polyhedral cone determined by a solution to an LP defined?
2. How is this cone used to cut off a solution to the LP relaxation of an MILP that is infeasible for the MILP?
3. What are the two sets needed to derive an intersection cut and which properties do they need to satisfy?
4. From a geometric perspective, how do we derive an intersection cut?
5. From an algebraic perspective, how do we derive an intersection cut?
6. How do we compute the coefficients of an intersection cut if the set $\mathcal{T}$ is a polyhedron?
7. How can we interpret intersection cuts from a disjunctive perspective?

8. How can we compute coefficients of intersection cuts if we apply this disjunctive point of view?

12

Intersection Cuts for Bilevel MILPs

As outlined in Chapter 9, different types of cuts can be used to discard bilevel-infeasible points within a branch-and-cut framework. One generic approach that guarantees to cut off such points for MILP-MILP bilevel problems (under certain mild conditions) has been proposed by Fischetti et al. (2016, 2017, 2018a). In these works, the authors exploit the theory of intersection cuts. These are general-purpose cuts for bilevel MILPs and belong to the most powerful ones in the current bilevel branch-and-cut literature. However, the use of intersection cuts for bilevel problems is quite novel and, in this chapter, we describe the basic ideas behind this approach.

To this end, we focus on MILP-MILP bilevel problems of the form

$$\begin{aligned}
\min_{x,y} \quad & c_x^\top x + c_y^\top y && \text{(12.1a)}\\
\text{s.t.} \quad & Ax + By \le a, && \text{(12.1b)}\\
& Cx + Dy \le b, && \text{(12.1c)}\\
& l \le y \le u, && \text{(12.1d)}\\
& x_i \in \mathbb{Z} \quad \text{for all } i \in I_x, && \text{(12.1e)}\\
& y_i \in \mathbb{Z} \quad \text{for all } i \in I_y, && \text{(12.1f)}\\
& d^\top y \le \varphi(x), && \text{(12.1g)}
\end{aligned}$$

where the optimal-value function φ of the lower level is computed by solving the follower's parametric MILP

$$\begin{aligned}
\varphi(x) := \min_{y} \quad & d^\top y && \text{(12.2a)}\\
\text{s.t.} \quad & Dy \le b - Cx, && \text{(12.2b)}\\
& l \le y \le u, && \text{(12.2c)}\\
& y_i \in \mathbb{Z} \quad \text{for all } i \in I_y. && \text{(12.2d)}
\end{aligned}$$

Here, we assume that c_x, c_y, A, B, a, C, D, b, l, and u are rational matrices and vectors of appropriate dimension and that $\emptyset \neq I_x \subseteq \{1, \dots, n_x\}$ and $I_y \subseteq \{1, \dots, n_y\}$ identify the sets of indices of the integer-constrained variables in x and y, respectively. We set $n := n_x + n_y$. Note that we explicitly state lower and upper bounds for the follower's variables in Constraint (12.2c) as they will be used for the development of some intersection cuts. Nevertheless, we allow that some entries in l or u are $\pm\infty$. Finally, note that we now changed inequality constraints to $\leq$-constraints to be in line with the most important literature on which this chapter is based on, e.g., Fischetti et al. (2016, 2017, 2018a) and Xu and Wang (2014).

To simplify the presentation, we make the following assumptions. Motivations and more details for these (standard) assumptions are given in Chapter 6.

Assumption 12.1 (i) The shared constraint set Ω is non-empty and bounded.
(ii) Problem (12.1) is feasible.
(iii) Linking variables x_j, $j \in L$, are integer, i.e., $L \subseteq I_x$.

In this chapter, we first explain how to apply the theory of intersection cuts in the context of bilevel optimization. The resulting cuts strongly depend on the choice of the set that does not contain bilevel-feasible solutions in its interior. We refer to such a set as a "bilevel-free set" in the following and discuss possible choices for deriving these sets in Section 12.2. The larger the bilevel-free sets, the stronger the derived intersection cuts. Therefore, we discuss strategies for enlarging such sets in Section 12.3. Separation procedures and the generation of the respective bilevel-free sets are provided in Section 12.4. We close this chapter by providing a series of illustrative examples in Section 12.5.

12.1 Intersection Cuts for MILP-MILP Bilevel Problems

In this section, we discuss how to use intersection cuts within a branch-and-cut framework to cut off bilevel-infeasible points. These cuts will then be integrated in Steps 8 and 20 of the node processing procedure in Algorithm 7, which has been presented in Chapter 9. Before we discuss how this is done, let us recall the MILP setting. As discussed in Chapter 11, we usually use intersection cuts to cut off the current solution, say ξ^*, to the model's LP relaxation if it violates some integrality conditions. To this end, we typically use the following two sets:

(i) A polyhedral cone $C(\xi^*)$ pointed at ξ^* associated with a basic solution to the model's LP relaxation. Such a cone contains all feasible points of the underlying MILP model.

(ii) A convex set $\mathcal{T}$ whose interior contains ξ^* but no feasible point of the underlying MILP model.

Informally speaking, an intersection cut is obtained by first calculating the intersection points of half-lines originating at ξ^*, i.e., the extreme rays of the polyhedral cone $C(\xi^*)$, with the boundary of the set $\mathcal{T}$, and then deriving a hyperplane that passes through these intersection points. Such a hyperplane corresponds to a valid inequality, which is called an intersection cut. Moreover, it separates the point ξ^* from the set of feasible points of the underlying MILP model; see Examples 11.2, 11.4, 11.6, and Figure 11.2 in Chapter 11.

Infeasibility in the MILP context is related to the fact that some of the components of ξ^*, which are supposed to be integer, are fractional. We now generalize this idea and apply it to the MILP-MILP bilevel setting. To this end, we still intersect the two convex sets as above. However, we re-define the convex set $\mathcal{T}$ so that it allows to cut off points that are bilevel infeasible even if they satisfy all integrality conditions.

We embed intersection cuts in the branch-and-cut method presented in Chapter 9. Thus, given a bilevel-infeasible and possibly fractional point (x^*, y^*) that is a solution to the LP relaxation at the current node of the search tree, the goal is to find a cutting plane that cuts off this point without discarding other bilevel-feasible points. To this end, we intersect the following two sets.

(i) A cone pointed at (x^*, y^*) that contains all bilevel-feasible points w.r.t. the current branch-and-bound node. As before, this is the polyhedral cone associated with the current LP-optimal solution (x^*, y^*) and its optimal basis; see Chapter 11.
(ii) A convex set $\mathcal{T}$ that contains (x^*, y^*) but no bilevel-feasible points in its interior. This set is called *bilevel-free set* from now on.

When it comes to the choice of the bilevel-free set $\mathcal{T}$, several possibilities have been exploited in the literature and we discuss the most important results below. Let us emphasize that the strength of the derived intersection cut strongly depends on the choice of this set.

Remark 12.2 As the LP relaxation at a given branch-and-bound node exploits locally-valid information (notably, the reduced variable domains resulting from branching), all intersection cuts are locally (as opposed to globally) valid. This means that they are valid for the subtree originating at the node at which they have been generated but not necessarily at other nodes of the branch-and-bound search tree; see Section 9.1 again where the distinction between globally and locally valid cuts is discussed in more detail.

12.2 Bilevel-Free Sets

Recall that the lower bound obtained by solving the single-level relaxation is typically rather weak. Hence, the more diverse the family of intersection cuts, the larger the portion of the bilevel-infeasible space that can be cut off and, thus, potentially the better the performance of the underlying branch-and-cut algorithm. In this section, we discuss different variants to define bilevel-free sets. Each of them results in a different cutting plane and we show that these cuts do not dominate each other by using illustrative examples.

For the validity of some of the intersection cuts that we introduce below, we need to impose the following.

Assumption 12.3 $Cx + Dy - b$ is integer for all $(x, y) \in \Omega$.

This assumption is needed for two of the following bilevel-free sets in order to guarantee that the resulting intersection cuts off a given bilevel-infeasible point. Note that Assumption 12.3 can be satisfied even if the coefficients in C, D, and b are rational numbers, as they can be properly scaled. The major requirement is thus that the coefficients of the matrix D associated with columns $j \notin I_y$ should be zero. In other words, the linking constraints at the lower level should only involve y variables whose components are restricted to be integer.

12.2.1 Improving Solution Bilevel-Free Sets

The following result is due to Fischetti et al. (2018a) and it has also been used implicitly in Xu and Wang (2014) for deriving branching decisions in a branch-and-bound setting.

Proposition 12.4 (Theorem 3 in Fischetti et al. (2018a)) *Given an arbitrary point $\hat{y} \in \mathbb{R}^{n_y}$ that satisfies (12.2c) and (12.2d), the set*

$$\mathcal{T}(\hat{y}) = \left\{(x, y) \in \mathbb{R}^n : d^\top y > d^\top \hat{y},\ Cx + D\hat{y} \leq b\right\} \tag{12.3}$$

does not contain any bilevel-feasible point.

Proof: For a given $x \in \mathbb{R}^{n_x}$, Constraints (12.2c) and (12.2d) together with $Cx + D\hat{y} \leq b$ imply that $\hat{y}$ is feasible for the x-parameterized lower-level problem. Its objective function value is $d^\top \hat{y}$. Hence, for the optimal follower's response, we have $\varphi(x) \leq d^\top \hat{y}$ and, thus, all points (x, y) with $d^\top y > d^\top \hat{y}$ can be discarded. □

Exercise 12.5 Consider the bilevel problem in Example 6.2. Note that we have $\hat{y} \in \{0, 1, \ldots, 4\}$ due to the lower-level constraints.

(i) For each $\hat{y} \in \{0, 1, \ldots, 4\}$, provide the definition of the corresponding bilevel-free set $\mathcal{T}(\hat{y})$.
(ii) For $\hat{y} \in \{1, \ldots, 3\}$, use the procedure described in Section 11.4 to derive the corresponding intersection cut obtained by intersecting $\mathcal{T}(\hat{y})$ with the LP cone pointed at $(2, 4)$, which is the optimal LP solution to the single-level relaxation.
(iii) What happens if you try to intersect $\mathcal{T}(\hat{y})$ for $\hat{y} = 4$ with the LP cone pointed at $(2, 4)$? Can you cut off the point $(2, 4)$?

Recall that we would like to derive a valid intersection cut to cut off a given vertex (x^*, y^*) of Ω, provided that (x^*, y^*) belongs to the *interior* of $\mathcal{T}(\hat{y})$ for a suitably chosen $\hat{y}$. For the set $\mathcal{T}(\hat{y})$ defined above, this property cannot always be guaranteed as (x^*, y^*) may belong to the boundary of $\mathcal{T}(\hat{y})$. This can, e.g., be observed in Exercise 12.5 for $\hat{y} = 4$ and in Example 12.8 for $\hat{y} = 2$ or $\hat{y} = 1$. If (x^*, y^*) lies at the boundary of $\mathcal{T}(\hat{y})$ determined by one of the hyperplanes of the lower-level problem, the following proposition shows that, using Assumption 12.3, we can extend $\mathcal{T}(\hat{y})$ by "moving apart" its hyperplanes. This results in an *enlarged polyhedron* that guarantees that the point (x^*, y^*) is in its interior.

Proposition 12.6 *Suppose that Assumption 12.3 holds. Then, for any $\hat{y} \in \mathbb{R}^{n_y}$ that satisfies (12.2c) and (12.2d), the* enlarged bilevel-free set

$$\mathcal{T}^+(\hat{y}) = \left\{(x, y) \in \mathbb{R}^n : d^\top y \geq d^\top \hat{y},\ Cx + D\hat{y} \leq b + \mathbb{1}\right\} \tag{12.4}$$

does not contain any bilevel-feasible point in its interior.

Proof: Assume there is a bilevel-feasible point $(\bar{x}, \bar{y})$ such that $(\bar{x}, \bar{y}) \in \operatorname{int}(\mathcal{T}^+(\hat{y}))$, i.e., $d^\top \bar{y} > d^\top \hat{y}$ and $C\bar{x} + D\hat{y} < b + \mathbb{1}$ holds. By Assumption 12.3, $C\bar{x} + D\hat{y} - b + \mathbb{1}$ is integer. Hence, the latter inequality implies $C\bar{x} + D\hat{y} \leq b$. Because $\hat{y}$ satisfies the constraints in (12.2c) and (12.2d) as well as $C\bar{x} + D\hat{y} \leq b$, it is feasible for the $\bar{x}$-parameterized lower-level problem. Therefore, we have $\varphi(\bar{x}) \leq d^\top \hat{y} < d^\top \bar{y}$ for the optimal follower's response, which contradicts the starting assumption that $(\bar{x}, \bar{y})$ is bilevel feasible. □

Remark 12.7 The result from Proposition 12.6 can also be translated into a multi-term disjunction of the form

$$((Cx + D\hat{y})_1 \geq b_1 + 1) \bigvee \cdots \bigvee ((Cx + D\hat{y})_\ell \geq b_\ell + 1) \bigvee \left(d^\top y \leq d^\top \hat{y}\right),$$

where ℓ denotes the number of rows of the matrices C and D. Such multi-disjunctions are used for branching within an enumerative solution scheme proposed by Xu and Wang (2014).

Before we consider an example, let us briefly explain why the bilevel-free sets studied in this section are called "improving solution" bilevel-free sets. The reason is that, for a given node solution (x^*, y^*) that is integer- but not bilevel-feasible, a $\hat{y}$ is chosen to define $\mathcal{T}(\hat{y})$ that is a better, i.e., improving, solution of the follower for the given x^*.

Example 12.8 We slightly modify the bilevel problem in Example 6.2 by adding an additional constraint $y \leq 2$ to the lower level. The optimal LP solution to the single-level relaxation of this problem is attained at the point $(6, 2)$, which is bilevel infeasible. Let us take the LP cone pointed at $(6, 2)$ and intersect it with the bilevel-free set $\mathcal{T}(\hat{y})$ at $\hat{y} = 1$, which is defined as

$$\mathcal{T}(\hat{y}) = \mathcal{T}(1) = \left\{(x, y) \colon \frac{5}{2} \leq x \leq 8,\ y > 1\right\}.$$

The resulting intersection cut $2x + 11y \leq 27$ cuts off $(6, 2)$ and it is illustrated in Figure 12.1. After adding this cut to the LP relaxation of the single-level relaxation, we obtain the new LP solution $(5/2, 2)$. This point lies at the boundary of both sets $\mathcal{T}(2)$ and $\mathcal{T}(1)$, and $\mathcal{T}(0) \cap \{(5/2, 2)\} = \emptyset$ holds. Thus, no additional intersection cuts can be derived to cut off this point. In the context of the node processing procedure presented in Algorithm 7, if we integrate the separation of intersection cuts in Steps 8 and 20 of this algorithm, this means that we would not be able to cut off the fractional point $(5/2, 2)$ using an intersection cut. Thus, we would have to resort to branching.

Let us now derive an alternative intersection cut using the enlarged polyhedron $\mathcal{T}^+(\hat{y})$. We again choose $\hat{y} = 1$ and obtain

$$\mathcal{T}^+(\hat{y}) = \mathcal{T}^+(1) = \left\{(x, y) \colon 2 \leq x \leq \frac{17}{2},\ y \geq 1\right\}.$$

The intersection cut derived from this set is $x + 6y \leq 14$; see Figure 12.2. As we can see, this cut dominates the intersection cut derived from $\mathcal{T}(1)$, which follows from the fact that $\mathcal{T}(1) \subset \mathcal{T}^+(1)$. Moreover, the LP optimal solution obtained after adding $x + 6y \leq 14$ to the single-level relaxation is $(2, 2)$, which is bilevel feasible and therefore also optimal. △

12.2.2 Improving Direction Bilevel-Free Sets

An alternative definition for bilevel-free sets has been proposed by Wang and Xu (2017), where it has been used to determine branching rules in a branch-and-bound setting.

Recall that Lemma 6.8 states that if there is an $x \in X$ such that $\varphi(x)$ is unbounded, the MILP-MILP bilevel problem (12.1) is infeasible. The key idea

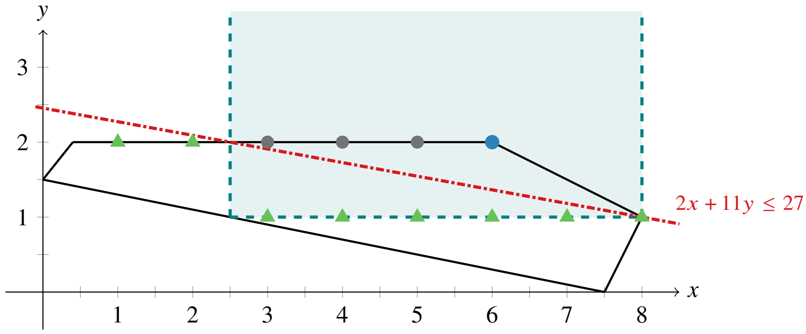


Figure 12.1 An illustration of an "improving solution" intersection cut for the bilevel problem in Example 12.8. Starting with the LP cone of the optimal solution $(6, 2)$ to the single-level relaxation and intersecting it with $\mathcal{T}(1)$, we derive the intersection cut $2x + 11y \leq 27$. The shaded area corresponds to the bilevel-free set $\mathcal{T}(1)$. Note that the boundary $y = 1$ is excluded from $\mathcal{T}(1)$ because the strict inequality $y > 1$ defines an open set. The dash-dotted line represents the intersection cut.

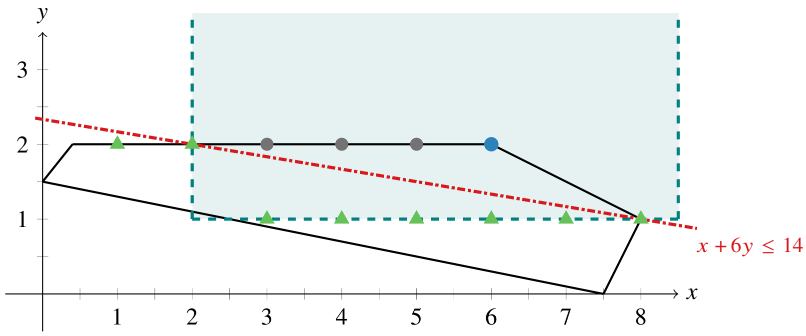


Figure 12.2 An illustration of an "improving solution" intersection cut for the bilevel problem in Example 12.8. Starting with the LP cone of the optimal solution $(6, 2)$ to the single-level relaxation and intersecting it with $\mathcal{T}^{+}(1)$, we derive the intersection cut $x + 6y \leq 14$. The shaded area corresponds to the bilevel-free set $\mathcal{T}^{+}(1)$. The dash-dotted line represents the intersection cut.

for the proof has been the following. If $\varphi(x)$ is unbounded for some $x \in X$, there exists some Δy such that Δy_j is integer for $j \in I_y$, $\Delta y \leq 0$, and $d^\top \Delta y < 0$ holds. Because Δy is independent of x, it exists for all x and, hence, Constraint (12.1g) cannot be satisfied by any $x \in X$. Thus, Problem (12.1) is infeasible.

In the following, let $\mathcal{Y}(x)$ be the set of all feasible responses of the follower for a given $x \in X$, i.e.,

$$\mathcal{Y}(x) = \left\{ y \in Y \colon Dy \leq b - Cx,\ l \leq y \leq u,\ y_i \in \mathbb{Z},\ i \in I_y \right\}.$$

We then obtain the following result.

Lemma 12.9 *If* $(x^*, y^*) \in \Omega$ *with* $y^* \in \mathcal{Y}(x^*) \setminus \mathcal{S}(x^*)$*, then there exists an improving direction* $\Delta y \in \mathbb{R}^{n_y}$ *such that*

$$d^\top \Delta y < 0 \quad \text{and} \quad \Delta y_j \in \mathbb{Z} \quad \text{for all } j \in I_y,$$
$$D(y^* + \Delta y) \le b - Cx^*, \; l \le y^* + \Delta y \le u,$$

holds.

Proof: As $y^* \in \mathcal{Y}(x^*) \setminus \mathcal{S}(x^*)$ holds, there exists a feasible point $y' \in \mathcal{Y}(x^*)$ with a better lower-level objective function value. Thus, we have

$$Dy' \le b - Cx^*, \quad l \le y' \le u, \quad d^\top y' < d^\top y^*$$

and y'_j is integer for all $j \in I_y$. Hence, Δy can be chosen as $\Delta y = y' - y^*$. □

Lemmas 6.8 and 12.9 motivate the definition of additional bilevel-free sets. Suppose there exists an improving direction Δy for the objective of the follower, which is not "in conflict" with the follower's constraints, i.e., the lower-level problem is feasible. Then, we can discard all the points $(x, y) \in \Omega$ for which $y + \Delta y \in \mathcal{Y}(x)$ holds.

Proposition 12.10 *For any* $\Delta\hat{y} \in \mathbb{R}^{n_y}$ *such that* $d^\top \Delta\hat{y} < 0$ *holds and so that* $\Delta\hat{y}_j$ *is integer for all* $j \in I_y$*, the polyhedron*

$$X(\Delta\hat{y}) = \{(x, y) \in \mathbb{R}^n : Cx + D(y + \Delta\hat{y}) \le b, \; l \le y + \Delta\hat{y} \le u\}$$

does not contain any bilevel-feasible point.

Proof: Consider any $(x, y) \in X(\Delta\hat{y})$. If $y \notin \mathcal{Y}(x)$, then (x, y) is bilevel infeasible. Assume now that $y \in \mathcal{Y}(x)$. Let $\Delta\hat{y} \in \mathbb{R}^{n_y}$ be such that $d^\top \Delta\hat{y} < 0$ holds and so that $\Delta\hat{y}_j$ is integer for all $j \in I_y$. Then, the point shifted by $\Delta\hat{y}$ in the y-coordinates, i.e., $y + \Delta\hat{y}$, is also a feasible follower's response. Because $d^\top(y + \Delta\hat{y}) < d^\top y$, it follows that $y \in \mathcal{Y}(x) \setminus \mathcal{S}(x)$, i.e., y cannot be an optimal follower's response for the given x. Hence, (x, y) is not bilevel feasible. □

In Wang and Xu (2017), the sets $X(\Delta\hat{y})$ are called *scoops* of a watermelon, where the watermelon refers to the set Ω. The authors take a bilevel-infeasible point (to which they refer as a *seed* of the watermelon) and then construct a scoop around it, determined by a direction $\Delta\hat{y}$ in which the objective function of the follower can be improved without violating the lower-level constraints. Such a scoop eventually serves to create a multi-branching rule. In the remainder of this chapter, we refer to $X(\Delta\hat{y})$ as the *improving direction bilevel-free set*, and to the intersection cuts derived from it as *improving direction intersection cuts*.

Note that, in contrast to the definition of $\mathcal{T}(\hat{y})$ in (12.3), the inequalities that define $X(\Delta\hat{y})$ also include bound constraints on the y variables. However, also for $X(\Delta\hat{y})$ it may happen that a bilevel-infeasible point (x^*, y^*) representing an LP solution at the current node of the branch-and-bound search tree does not belong to the interior of $X(\Delta\hat{y})$. Hence, we need to make another assumption.

Assumption 12.11 The differences $l - y$ and $y - u$ are integer for all $(x, y) \in \Omega$.

By using the latter assumption together with Assumption 12.3, we can enlarge $X(\Delta\hat{y})$ so that (x^*, y^*) is guaranteed to belong to its interior if $\Delta\hat{y}$ is chosen appropriately.

Proposition 12.12 *Suppose that Assumptions 12.3 and 12.11 hold. Then, for any $\Delta\hat{y} \in \mathbb{R}^{n_y}$ such that $d^\top \Delta\hat{y} < 0$ holds and so that $\Delta\hat{y}_j$ is integer for all $j \in I_y$, the polyhedron*

$$X^+(\Delta\hat{y}) = \{(x, y) \in \mathbb{R}^n : Cx + D(y + \Delta\hat{y}) \le b + \mathbb{1}, \\ l - \mathbb{1} \le y + \Delta\hat{y} \le u + \mathbb{1}\}$$

does not contain any bilevel-feasible point in its interior.

Proof: To be in the interior of $X^+(\Delta\hat{y})$, a bilevel-feasible pair (x, y) needs to satisfy

$$Cx + Dy + D\Delta\hat{y} < b + \mathbb{1} \quad \text{and} \quad l - \mathbb{1} < y + \Delta\hat{y} < u + \mathbb{1}.$$

Because of Assumptions 12.3 and 12.11, the latter conditions can be replaced with $Cx + Dy + D\Delta\hat{y} \le b$ and $l \le y + \Delta\hat{y} \le u$. Hence, the claim follows from Proposition 12.10. □

The size and shape of the set $X^+(\Delta\hat{y})$ depend on the choice of the vector $\Delta\hat{y}$. We discuss some possibilities to choose the improving direction $\Delta\hat{y}$ in Section 12.4.

Exercise 12.13 For the bilevel problem in Example 6.2, show that $\Delta\hat{y} = -1$ defines an improving direction such that $X(\Delta\hat{y}) \ne \emptyset$. How do the sets $X(\Delta\hat{y})$ and $X^+(\Delta\hat{y})$ look like? Take the LP cone pointed at $(2, 4)$ to derive the two respective intersection cuts and provide graphical illustrations of them.

12.2.3 Bilevel-Free Hypercubes

The definitions of the former two bilevel-free polyhedra $\mathcal{T}^+(\hat{y})$ and $X^+(\Delta\hat{y})$ rely on Assumption 12.3 as well as Assumptions 12.3 and 12.11, respectively. Hence, they can mainly be used for solving bilevel MILP-ILP problems. For example, Assumption 12.3 implies that continuous variables in the lower level are allowed only if they do not appear in the linking constraints $Cx + Dy \le b$, i.e., the coefficients of the matrix D associated with the continuous lower-level variables must be zero. Such an assumption may sometimes be too restrictive as it does not allow to handle other situations in which the lower level is purely continuous or in which continuous lower-level variables appear in the linking constraints of the lower-level problem.

We now provide an alternative polyhedron from which valid intersection cuts can be derived even in such situations. When embedded into Steps 8 and 20 of Algorithm 7 presented in Chapter 9, these cuts ensure that the proposed algorithm solves bilevel MILP-MILP problems after adding a finite number of intersection cuts provided that Assumption 12.1 holds; see Exercise 12.16.

To derive these new intersection cuts, we exploit the integrality of the linking variables x_j, $j \in L$, according to Assumption 12.1 and generate a hypercube around a point whose coordinates corresponding to indices in L are fixed to given values x^*_j, $j \in L$. Using this hypercube, we can then generate an intersection cut that discards all solutions whose coordinates are fixed according to x^*_j, $j \in L$. Before discarding these points, it is necessary to compute the best follower response for the given choice of x^*_j, $j \in L$, and to store the respective incumbent solution. We formally state this in the next proposition.

Proposition 12.14 *For a given point $(x^*, y^*) \in \Omega$, let us assume the x^*_L-parameterized refinement problem (6.5) is feasible and let $(\hat{x}, \hat{y})$ be its optimal solution. Then, the hypercube*

$$HC^+(x^*) = \left\{(x, y) \in \mathbb{R}^n : x^*_j - 1 \le x_j \le x^*_j + 1,\ j \in L\right\} \tag{12.5}$$

does not contain any bilevel-feasible solution strictly better than $(\hat{x}, \hat{y})$ in its interior.

Proof: Note that the interior of $\mathrm{HC}^+(x^*)$ only contains bilevel-feasible points (x, y) with $x_j = x^*_j = \hat{x}_j$ for all $j \in L$. By construction, among these points, $(\hat{x}, \hat{y})$ is the best bilevel-feasible one, which already completes the proof. □

Remark 12.15 Given a point $(x^*, y^*) \in \Omega$, if the associated x^*_L-parameterized refinement problem (6.5) is infeasible, the whole search space with $x_L = x^*_L$ can be discarded, i.e., the corresponding node of the branch-and-cut search tree in which variables x_L are fixed to x^*_L can be pruned.

Compared to the other two bilevel-free polyhedra, namely $\mathcal{T}^+(\hat{y})$ and $X^+(\Delta\hat{y})$, the set $\mathrm{HC}^+(x^*)$ is defined only using conditions on x. Intersection cuts derived from $\mathrm{HC}^+(x^*)$ are called *hypercube intersection cuts*. As explained above, they can be applied in the most general MILP-MILP setting under Assumption 12.1, even in the cases in which cuts derived from $\mathcal{T}^+(\hat{y})$ and $X^+(\Delta\hat{y})$ are not guaranteed to cut off a bilevel-infeasible point.

Exercise 12.16 Prove that, under Assumption 12.1, hypercube intersection cuts embedded into Steps 8 and 20 of Algorithm 7 lead to an overall branch-and-cut algorithm that terminates with an optimal solution to the bilevel MILP-MILP problem (12.1) after adding a finite number of hypercube intersection cuts.

12.3 How to Enlarge Bilevel-Free Sets

As illustrated in Example 12.8, the larger a bilevel-free set $\mathcal{T}$, the stronger or deeper the derived intersection cut. Let us assume that the set $\mathcal{T}$ is a polyhedron. Then, the following question arises: Can we find a simple and computationally cheap way to enlarge the bilevel-free sets introduced above? One relatively easy way could be to remove redundant inequalities from the description of $\mathcal{T}$. Intersection cuts are added as locally valid constraints, i.e., they take into account the branching decisions taken at node k of the branch-and-cut search tree. Therefore, even though the original description of $\mathcal{T}$ may not contain redundant inequalities, some of them can become redundant due to certain branching decisions. Recall that $\mathcal{F}_k \subset \mathcal{F}$ denotes the restricted set of bilevel-feasible points at node k of the branch-and-bound search tree, i.e., $\mathcal{F}_k = \mathcal{F} \cap \bar{\Omega}_k$. We have the following result.

Proposition 12.17 *Let $\pi^\top x + \rho^\top y \leq \sigma$ be one of the inequalities included in the description of the bilevel-free set $\mathcal{T}$, which contains no bilevel-feasible points in its interior. Suppose that*

$$\left\{(x, y) \in \mathbb{R}^n : \pi^\top x + \rho^\top y \geq \sigma\right\} \cap \mathcal{F}_k = \emptyset$$

holds. Then, this inequality can be removed from $\mathcal{T}$.

Proof: From the disjunctive (cuts) perspective (see Section 11.4 for the disjunctive interpretation of intersection cuts), we associate the following disjunct

$$\left\{(x, y) \in \mathbb{R}^n : \pi^\top x + \rho^\top y \geq \sigma\right\}$$

with the constraint $\pi^\top x + \rho^\top y \leq \sigma$. Recall that the set of all bilevel-feasible points $\mathcal{F}_k$ is contained in the union of disjuncts derived from the description of $\mathcal{T}$. The assumption $\{(x, y) \in \mathbb{R}^n : \pi^\top x + \rho^\top y \geq \sigma\} \cap \mathcal{F}_k = \emptyset$ implies that there are no bilevel-feasible points in the disjunct derived from $\pi^\top x + \rho^\top y \leq \sigma$. Hence, this disjunct can be removed from the union. Accordingly, the associated inequality $\pi^\top x + \rho^\top y \leq \sigma$ can be removed from the description of $\mathcal{T}$. □

Detecting redundant inequalities that satisfy the condition of Proposition 12.17

can be as difficult as solving the original problem because one needs the description of the set $\mathcal{F}_k$ of all bilevel-feasible points. So, how can we effectively, or even heuristically, detect redundant inequalities in $\mathcal{T}$? For this purpose, we can exploit the local information available at the current node of the branch-and-cut search tree and recall that intersection cuts are locally valid only.

Proposition 12.18 *Let (x^-, y^-) and (x^+, y^+) be lower, respectively, upper bounds on (x, y) at the current node of the branch-and-cut search tree. Moreover, let*

$$\pi^\top x + \rho^\top y \le \sigma \tag{12.6}$$

be one of the inequalities included in the description of a bilevel-free set $\mathcal{T}$, which contains no bilevel-feasible points in its interior. If

$$\sum_{j \in \{1,\dots,n_x\}} \max\left\{\pi_j x_j^+, \, \pi_j x_j^-\right\} + \sum_{j \in \{1,\dots,n_y\}} \max\left\{\rho_j y_j^+, \, \rho_j y_j^-\right\} < \sigma \tag{12.7}$$

holds, Inequality (12.6) can be removed from $\mathcal{T}$.

Proof: At the current node of the branch-and-cut search tree, we have

$$\mathcal{F}_k \subseteq \bar{\Omega}_k \subseteq \left\{(x, y) \in \mathbb{R}^n : x^- \le x \le x^+, \, y^- \le y \le y^+\right\}.$$

Condition (12.7) implies

$$\bar{\Omega}_k \cap \left\{(x, y) \in \mathbb{R}^n : \sum_{j \in \{1,\dots,n_x\}} \max\left\{\pi_j x_j^+, \, \pi_j x_j^-\right\} + \sum_{j \in \{1,\dots,n_y\}} \max\left\{\rho_j y_j^+, \, \rho_j y_j^-\right\} \ge \sigma\right\} = \emptyset.$$

Following the same arguments as in the proof of Proposition 12.17, all bilevel-feasible points that can be found within the current node k of the search tree are in the halfspace defined by $\pi^\top x + \rho^\top y \le \sigma$. Hence, the constraint $\pi^\top x + \rho^\top y \le \sigma$ is redundant. □

Proposition 12.18 can be exploited when separating intersection cuts, which we discuss in the next section.

12.4 Separation of Intersection Cuts

Intersection cuts are separated within the node processing procedure of the generic branch-and-cut approach given in Algorithm 7 in Chapter 9. More precisely, they can be embedded in Steps 8 and 20 of that algorithm.

For the correctness of Algorithm 7, it is not necessary to integrate intersection cuts in Step 8 to separate fractional points as we can always resort to branching or use general-purpose MILP cuts usually integrated in the MILP solvers. Bilevel-specific cuts that are cutting off fractional points are mainly used to potentially enhance the algorithm's performance. Therefore, for the ease of presentation and without loss of generality, we can assume that intersection cuts are only embedded in Step 20 of Algorithm 7. This means that, for the point (x^*, y^*) that we want to cut off, the following conditions hold.

(i) The point (x^*, y^*) is an optimal LP solution at the current node of a problem that contains all constraints from Ω plus all previously separated cuts from the parent nodes, together with additional variable fixings and branching decisions. This set is denoted as $\bar{\Omega}_k$ in Algorithm 7.
(ii) In addition, the point (x^*, y^*) satisfies all integrality conditions, i.e., x_i^*, $i \in I_x$, and y_j^*, $j \in I_y$, are integer.

The node solution (x^*, y^*) is the apex of the LP cone defined by a basic LP solution, and this cone is intersected with one of the three bilevel-free sets discussed above to generate an intersection cut. Once the bilevel-free set is defined, the exact procedure that generates a valid intersection cut is the same one typically used for MILPs and it is explained in Chapter 11. We therefore focus on the derivation of appropriate bilevel-free sets in the following.

Each of the three bilevel-free sets, namely $\mathcal{T}^+(\hat{y})$, $X^+(\Delta\hat{y})$, and $\mathrm{HC}^+(x^*)$ given in Propositions 12.6, 12.12, and 12.14, respectively, can be used to derive different intersection cuts. We point out that the choice of the vectors $\hat{y}$, $\Delta\hat{y}$, and x^* in the definitions of these sets may not be unique. Therefore, different strategies can be employed to determine appropriate vectors $\hat{y}$, $\Delta\hat{y}$, and x^*, respectively, and each of them is likely to result in a different cut, having a different depth and, thus, strength. We now discuss several strategies that have been proven to be useful in implementations of modern branch-and-cut approaches; see, e.g., Fischetti et al. (2017, 2018a) and Gaar et al. (2024).

The following assumption on d can be made w.l.o.g. (after some possible scaling). Moreover, we assume that y is discrete in what follows.

Assumption 12.19 The vector d is an integer vector and $y \in \mathbb{Z}^{n_y}$.

Assumption 12.19 will be a standing assumption for the remainder of this chapter. This assumption is needed in the separation approaches described below so that, instead of imposing $d^\top y < d^\top y^*$ for a given bilevel-infeasible point (x^*, y^*), we can replace it with $d^\top y \le d^\top y^* - 1$.

12.4.1 Separation of Intersection Cuts Derived from $\mathcal{T}^+(\hat{y})$

The separation consists of choosing a point $\hat{y}$, i.e., an improving solution, for which we then compute the bilevel-free set $\mathcal{T}^+(\hat{y})$. In the context of branch-and-cut, when we want to discard a bilevel-infeasible point $(x^*, y^*) \in \bar{\Omega}_k$, a natural choice for $\hat{y}$ is to use an optimal follower's response $\hat{y}$ for the given x^*, i.e., $\hat{y} \in \mathcal{S}(x^*)$. Choosing such a $\hat{y}$ does not even require additional computational efforts because $\varphi(x^*)$ has to be computed anyway to check the bilevel feasibility of (x^*, y^*). The resulting bilevel-free set then maximizes the distance of (x^*, y^*) from the face of $\mathcal{T}^+(\hat{y})$ induced by $d^\top y \geq d^\top \hat{y}$. Such an approach may work well in some cases. However, numerical experiments conducted in Fischetti et al. (2018a) indicate that much more effective intersection cuts can be derived from a point $\hat{y}$ that improves the objective of the follower but, at the same time, allows to discard as many redundant inequalities from $\mathcal{T}^+(\hat{y})$ as possible.

Recall that

$$\mathcal{T}^+(\hat{y}) = \left\{(x, y) \in \mathbb{R}^n : d^\top y \geq d^\top \hat{y},\ C_{i\cdot}x + D_{i\cdot}\hat{y} \leq b_i + 1,\ i \in \{1, \dots, \ell\}\right\}.$$

Following Proposition 12.18, given an inequality $C_{i\cdot}x + D_{i\cdot}\hat{y} \leq b_i + 1$ from the description of $\mathcal{T}^+(\hat{y})$ and a point $\hat{y} \in \mathbb{R}^{n_y}$ together with the lower bounds x^- and upper bounds x^+ on x at the current node of the branch-and-cut search tree, if

$$\sum_{j \in \{1,\dots,n_x\}} \max\left\{C_{ij}x_j^+,\ C_{ij}x_j^-\right\} + D_{i\cdot}^\top \hat{y} \leq b_i \tag{12.8}$$

holds, the inequality is redundant and can be removed from $\mathcal{T}^+(\hat{y})$.

To find such a $\hat{y}$, we solve an MILP with additional binary variables $w \in \{0,1\}^\ell$ that count the number of non-redundant inequalities in the description of $\mathcal{T}^+(\hat{y})$. The goal is to search for a feasible point $\hat{y}$ of the lower level that improves the objective value of the follower by at least one unit, while maximizing the number of inequalities that can be removed from the associated set $\mathcal{T}^+(\hat{y})$. This MILP is given by

$$\min_{y,w} \quad \sum_{i=1}^{\ell} w_i \tag{12.9a}$$

$$\text{s.t.} \quad d^\top y \leq d^\top y^* - 1, \tag{12.9b}$$

$$l \leq y \leq u, \tag{12.9c}$$

$$C_i^{\max} + D_{i\cdot}y - b_i \leq (C_i^{\max} - C_i^*)w_i, \quad i \in \{1, \dots, \ell\}, \tag{12.9d}$$

$$y_j \in \mathbb{Z}, \quad j \in I_y, \tag{12.9e}$$

$$w \in \{0,1\}^\ell, \tag{12.9f}$$

with

$$C_i^* := C_{i\cdot}x^* \le C_i^{\max} := \sum_{j\in\{1,\dots,n_x\}} \max\left\{C_{ij}x_j^-,\, C_{ij}x_j^+\right\}.$$

In Problem (12.9), the binary variable w_i takes the value 0 if and only if the ith inequality satisfies Condition (12.8), i.e., the ith inequality can be removed from $\mathcal{T}^+(\hat{y})$. Hence, the objective function (12.9a) minimizes the number of non-redundant inequalities. Having $w_i = 1$ enforces $C_{i\cdot}x^* + D_{i\cdot}\hat{y} \le b_i$. Together with (12.9b), this guarantees that (x^*, y^*) belongs to the interior of $\mathcal{T}^+(\hat{y})$.

12.4.2 Separation of Intersection Cuts Derived from $X(\Delta\hat{y})$

Given (x^*, y^*), the vector $\Delta\hat{y}$ can be obtained by solving an additional MILP intended to produce a large number of redundant inequalities according to Proposition 12.17. This procedure has been originally proposed by Wang and Xu (2017) to derive multi-disjunctions and it has been exploited for the generation of intersection cuts by Fischetti et al. (2017).

To simplify the notation, let us re-write the lower-level feasible set as

$$\mathcal{Y}(x) = \left\{y \in \mathbb{R}^{n_y} : \tilde{D}y \le \tilde{b} - \tilde{C}x,\ y_j \in \mathbb{Z},\ j \in I_y\right\},$$

where $\tilde{C}x + \tilde{D}y \le \tilde{b}$ contains all lower-level constraints, including the bounds on the y variables. Let $\tilde{\ell}$ denote the number of rows of $\tilde{C}$.

Proposition 12.20 *Let $\Delta\hat{y} \in \mathbb{R}^{n_y}$ be such that $d^\top\Delta\hat{y} < 0$ holds and so that $\Delta\hat{y}_j$ is integer for all $j \in I_y$. Consider the polyhedron*

$$X(\Delta\hat{y}) = \left\{(x, y) \in \mathbb{R}^n : \tilde{C}_{i\cdot}x + \tilde{D}_{i\cdot}(y + \Delta\hat{y}) \le \tilde{b}_i + 1,\ i \in \{1, \dots, \tilde{\ell}\}\right\}$$

that does not contain any bilevel-feasible point in its interior. If $\tilde{D}_{i\cdot}\Delta\hat{y} \le 0$ for some $i \in \{1, \dots, \ell\}$, then the ith inequality in the description of $X(\Delta\hat{y})$ is redundant and can be removed.

Proof: Consider the ith inequality for which $\tilde{D}_{i\cdot}\Delta\hat{y} \le 0$ holds. We have

$$\begin{aligned}\bar{\Omega}_k &\subseteq \left\{(x, y) \in \mathbb{R}^n : \tilde{C}_{i\cdot}x + \tilde{D}_{i\cdot}y \le \tilde{b}_i\right\}\\ &\subseteq \left\{(x, y) \in \mathbb{R}^n : \tilde{C}_{i\cdot}x + \tilde{D}_{i\cdot}(y + \Delta\hat{y}) \le \tilde{b}_i\right\}.\end{aligned}$$

The first relationship holds because the inequality $\tilde{C}_{i\cdot}x + \tilde{D}_{i\cdot}y \le \tilde{b}_i$ is included in the description of the shared constrained set, whereas the second one follows from the assumption of the proposition. Hence,

$$\mathcal{F}_k \cap \left\{(x, y) \in \mathbb{R}^n : \tilde{C}_{i\cdot}x + \tilde{D}_{i\cdot}(y + \Delta\hat{y}) \ge \tilde{b}_i + 1\right\} = \emptyset$$

holds and the result follows using similar arguments as in the proof of Proposition 12.17. □

We now derive an MILP model for computing a vector $\Delta\hat{y}$ such that the "size" of the bilevel-free set $X(\Delta\hat{y})$ created around the point (x^*, y^*) is maximized. The model implicitly maximizes the number of redundant inequalities in the description of $X(\Delta\hat{y})$. To this end, we use continuous variables $t_i \geq 0$ associated with each $i \in \{1, \dots, \tilde{\ell}\}$ to measure the distance between (x^*, y^*) and the boundary of $X(\Delta\hat{y})$. The model reads

$$(\Delta\hat{y}, \hat{t}) \in \arg\min_{\Delta y, t} \quad \sum_{i=1}^{\tilde{\ell}} t_i \tag{12.10a}$$

$$\text{s.t.} \quad d^\top \Delta y \leq -1, \tag{12.10b}$$

$$\tilde{D}\Delta y \leq \tilde{b} - \tilde{C}x^* - \tilde{D}y^*, \tag{12.10c}$$

$$\Delta y_j \in \mathbb{Z}, \quad j \in I_y, \tag{12.10d}$$

$$\tilde{D}\Delta y \leq t, \quad t \geq 0. \tag{12.10e}$$

Constraint (12.10b) ensures that Δy is an improving direction for the lower-level objective. For each $i \in \{1, \dots, \tilde{\ell}\}$, the variable t_i attains the value 0 if $\tilde{D}_{i\cdot}\Delta\hat{y} \leq 0$ holds, meaning that $\tilde{C}_{i\cdot}x + \tilde{D}_{i\cdot}(y + \Delta\hat{y}) \leq \tilde{b}_i + 1$ is a redundant inequality according to Proposition 12.20. On the contrary, if $\tilde{D}_{i\cdot}\Delta\hat{y} > 0$ holds, due to (12.10c), (12.10e) and the minimization objective, the variable t_i measures the slack of the solution (x^*, y^*) with respect to constraint $i \in \{1, \dots, \tilde{\ell}\}$. Thus, the objective function (12.10a) is designed to maximize the size of the bilevel-free set associated with $\Delta\hat{y}$; see Wang and Xu (2017) for further details.

Note that Problem (12.10) cannot be unbounded due to Assumption 12.1. Moreover, we observe that one cannot increase the right-hand side of Constraint (12.10c) by 1 as this would allow the point (x^*, y^*) to be on the boundary of the enlarged polyhedron $X^+(\Delta\hat{y})$.

12.4.3 Separation of Intersection Cuts Derived from $\mathrm{HC}^+(x^*)$

A natural choice for x^* when generating the set $\mathrm{HC}^+(x^*)$ defined in (12.5) is to take the current LP-optimal solution from $\bar{\Omega}_k$ if the x^*_j values are integer for all $j \in L$. Contrary to the previous intersection cuts, for which it is sufficient to add the derived intersection cut that guarantees to cut off the point (x^*, y^*), we also need to make sure that the incumbent solution is properly updated for hypercube intersection cuts. Therefore, the following procedure has to be embedded at the end of Algorithm 7.

(i) Solve the x^*_L-parameterized refinement problem (6.5).
(ii) If the refinement problem is infeasible, fathom the current node, i.e., stop the node processing procedure and go back to the main method.

(iii) Let $(\hat{x}, \hat{y})$ be an optimal solution to the refinement problem.
(iv) Use $(\hat{x}, \hat{y})$ to possibly update the incumbent solution.
(v) Insert the hypercube intersection cut derived from $\mathrm{HC}^+(x^*)$.

In the first step of this procedure, among all follower's responses $y \in \mathcal{S}(x^*)$, we aim for the one which is optimistic for the leader—both with respect to the objective function and the coupling constraints. The remaining x variables, which are not fixed by branching yet, are optimized for the leader as their values are immaterial for the follower.

The correctness of the approach follows from Assumption 12.1 in which the variables indexed by $j \in L$ are assumed to be bounded integers. Indeed, if we would embed the above procedure in an explicit enumeration of all possible configurations of values for x_j, $j \in L$, we would eventually obtain the optimal solution to the underlying MILP-MILP bilevel problem. Even though this requires only a finite number of iterations, such a procedure would be highly inefficient. However, it can be embedded within the branch-and-cut search tree to reduce its size and enhance the pruning procedure.

In the remainder of this chapter, we provide examples of the intersection cuts presented above and illustrate them graphically.

12.5 Illustrative Examples

The bilevel-free sets $\mathcal{T}^+(\hat{y})$, $X^+(\Delta\hat{y})$, and $\mathrm{HC}^+(x^*)$ are not directly comparable, i.e., it is typically not possible to find a $\hat{y}$, a corresponding $\Delta\hat{y}$, and a corresponding x^*, such that one of the sets is a proper subset of another. This is because the inequalities defining $X^+(\Delta\hat{y})$ use the (x, y)-space, but those of $\mathcal{T}^+(\hat{y})$ only use the y-space ($d^\top y \geq d^\top \hat{y}$) or the x-space ($Cx + D\hat{y} \leq b + 1$), whereas those of $\mathrm{HC}^+(x^*)$ only use the x-space with respect to components in L. In addition, $\mathcal{T}^+(\hat{y})$ contains the inequality $d^\top y \geq d^\top \hat{y}$ that directly involves the follower's objective function, whereas the follower's objective function is only implicitly used in the definitions of the sets $X^+(\Delta\hat{y})$ and $\mathrm{HC}^+(x^*)$.

Therefore, all introduced intersection cuts are of interest in the context of developing an effective branch-and-cut procedure for MILP-MILP bilevel problems. We again use Example 6.2 to illustrate the different intersection cuts discussed in this chapter.

Example 12.21 (Example 6.2—Intersection cuts derived from $\mathcal{T}^+(\hat{y})$) We revisit the bilevel problem in Example 6.2. The optimal solution $(x^*, y^*) = (2, 4)$ to the single-level relaxation of the problem is not bilevel feasible and the optimal

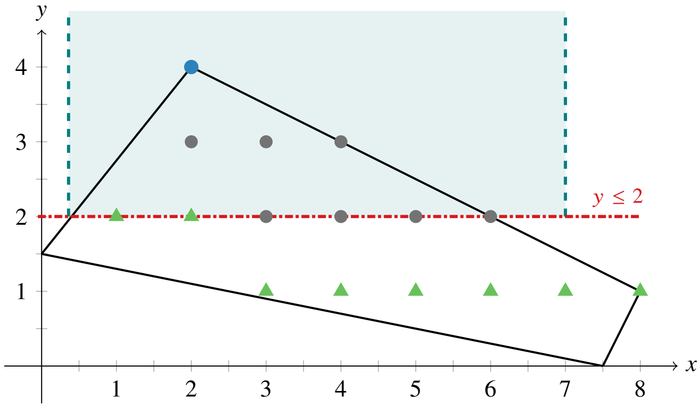


Figure 12.3 An illustration of an "improving solution" intersection cut for the bilevel problem in Example 12.21. Starting with the optimal solution $(2, 4)$ to the single-level relaxation and the optimal follower response $\hat{y} = 2$ for $x^* = 2$, we derive the intersection cut $y \leq 2$ from the improving solution bilevel-free set $\mathcal{T}^+(\hat{y}) = \{(x, y) : \frac{9}{25} \leq x \leq 7,\ y \geq 2\}$. The shaded area corresponds to the bilevel-free set $\mathcal{T}^+(\hat{y})$. The dash-dotted line represents the intersection cut.

follower's response for $x^* = 2$ is $\hat{y} = 2$. We have two choices to derive a bilevel-free set $\mathcal{T}^+(\hat{y})$ from this improving solution. The first choice follows from Proposition 12.6, see (12.4), and reads

$$\mathcal{T}^+(\hat{y}) = \mathcal{T}^+(2) = \left\{(x, y) : \frac{9}{25} \leq x \leq 7,\ y \geq 2\right\}.$$

The intersection cut derived from this set is $y \leq 2$ and it is illustrated in Figure 12.3.

The second choice is obtained by computing an improving solution $\hat{y}$, which is not necessarily an optimal follower's response, but which allows to remove redundant faces from the description of $\mathcal{T}^+(\hat{y})$. Such a solution is obtained by solving Problem (12.9). Specifically, assuming that $x^- = 0$ and $x^+ = 8$, we pre-compute $C^{\max} = [0, 8, 16, 0]$ and $C^* = [-50, 2, 4, -4]$, and we consider

$$\begin{aligned}(\hat{y}, \hat{w}) \in \underset{y \in \mathbb{Z}, w \in \{0,1\}^4}{\arg\min} \quad & \sum_{i=1}^{4} w_i \\ \text{s.t.} \quad & y \leq 3, \\ & 20y - 50w_1 \leq 30, \\ & 2y - 6w_2 \leq 2, \\ & -y - 12w_3 \leq -1, \\ & -10y - 4w_4 \leq -15.\end{aligned}$$

An optimal solution to this problem is not unique. For example, for $\hat{y} = 3$ or $\hat{y} = 2$, we have $\hat{w}_1 = \hat{w}_2 = 1$. This means that the last two constraints in the follower's model, namely

$$2x - \hat{y} \leq 16 \quad \text{and} \quad -2x - 10\hat{y} \leq -14$$

are redundant in the description of $\mathcal{T}^+(\hat{y})$.

The bilevel-free set associated with the improving solution $\hat{y} = 3$ is given by

$$\mathcal{T}^+(\hat{y}) = \mathcal{T}^+(3) = \left\{(x, y) \colon \frac{29}{25} \leq x \leq 5,\ y \geq 3\right\}.$$

The intersection cut derived from this set is $y \leq 3$ and it is illustrated in Figure 12.4. We observe that, in this specific example, the set $\mathcal{T}^+(3)$ is a proper subset of $\mathcal{T}^+(2)$, which means that the intersection cut derived from $\mathcal{T}^+(3)$ is dominated by the intersection cut derived from $\mathcal{T}^+(2)$. △

In the following example, we illustrate intersection cuts derived from the improving direction bilevel-free set $X^+(\Delta\hat{y})$.

Example 12.22 (Example 6.2—Intersection cuts derived from $X^+(\Delta\hat{y})$) We again start with the optimal solution $(x^*, y^*) = (2, 4)$ to the single-level relaxation of the problem in Example 6.2, which is not bilevel feasible. This time, to find an improving direction $\Delta\hat{y}$ around the point $(2, 4)$, we solve

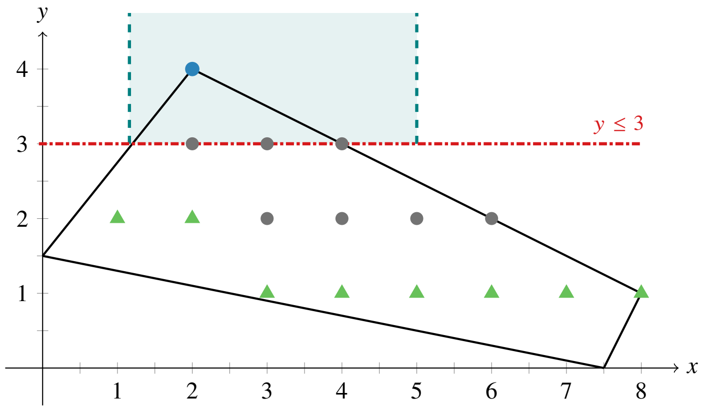


Figure 12.4 An illustration of an "improving solution" intersection cut for the bilevel problem in Example 12.21. Starting with the optimal solution $(2, 4)$ to the single-level relaxation and the improving solution of the follower $\hat{y} = 3$ for $x^* = 2$ found by solving Problem (12.9), we derive the intersection cut $y \leq 3$ from the improving solution bilevel-free set $\mathcal{T}^+(\hat{y}) = \{(x, y) \colon \frac{29}{25} \leq x \leq 5,\ y \geq 3\}$. The shaded area corresponds to $\mathcal{T}^+(\hat{y})$. The dash-dotted line represents the intersection cut.

Problem (12.10), which in our case is given by

$$
\begin{aligned}
(\Delta\hat{y}, \hat{t}) \in \operatorname*{arg\,min}_{\Delta y \in \mathbb{Z},\, t \in \mathbb{R}^4_{\geq 0}} \quad & \sum_{i=1}^{4} t_i \\
\text{s.t.} \quad & \Delta y \leq -1, \\
& 20\Delta y \leq 0, \\
& 2\Delta y \leq 0, \\
& -\Delta y \leq 15, \\
& -10\Delta y \leq 29, \\
& 20\Delta y \leq t_1, \\
& 2\Delta y \leq t_2, \\
& -\Delta y \leq t_3, \\
& -10\Delta y \leq t_4.
\end{aligned}
$$

The optimal value is 11 and it can be obtained by setting, e.g., $\Delta\hat{y} = -1$ with $\hat{t}_1 = \hat{t}_2 = 0$ and $\hat{t}_3 = 1$, $\hat{t}_4 = 10$. Thus, the corresponding set $X^+(\Delta\hat{y})$ can be

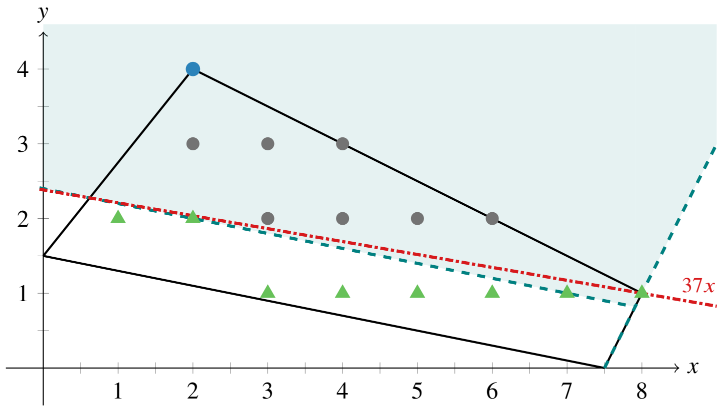


Figure 12.5 An illustration of an "improving direction" intersection cut for the bilevel problem in Example 12.22. Starting with the optimal solution $(2, 4)$ to the single-level relaxation, we derive the intersection cut $37x + 214y \leq 510$ from the improving direction bilevel-free set $X^+(\Delta\hat{y}) = \{(x, y) : 2x - y \leq 15,\ -2x - 10y \leq -24\}$. Dashed lines are faces of the set $X^+(\Delta\hat{y})$. The dash-dotted line represents the intersection cut.

derived from the system of inequalities

$$\begin{aligned} -25x + 20y &\leq 51, \\ x + 2y &\leq 13, \\ 2x - y &\leq 15, \\ -2x - 10y &\leq -24. \end{aligned}$$

A value of $t_i = 0$ in the above model indicates that Constraint i is redundant in the description of $X^+(\Delta\hat{y})$. Hence, we obtain

$$X^+(\Delta\hat{y}) = \{(x, y) : 2x - y \leq 15,\ -2x - 10y \leq -24\}.$$

After intersecting the faces of this polyhedron with the two extreme rays emanating from the corner point $(2, 4)$ associated with constraints $-25x + 20y \leq 30$ and $x + 2y \leq 10$, we obtain the intersection cut $37x + 214y \leq 510$. This cut is illustrated in Figure 12.5. Note that the point $(2, 2)$ is in the strict interior of the resulting single-level relaxation obtained after adding this intersection cut as we have $37 \cdot 2 + 214 \cdot 2 = 502 < 510$, but the bilevel-feasible point $(8, 1)$ is a corner point of the resulting polytope. △

Comparing the two intersection cuts from Figures 12.3 and 12.5, respectively, we observe that none of the two cuts dominates the other. These constraints cut

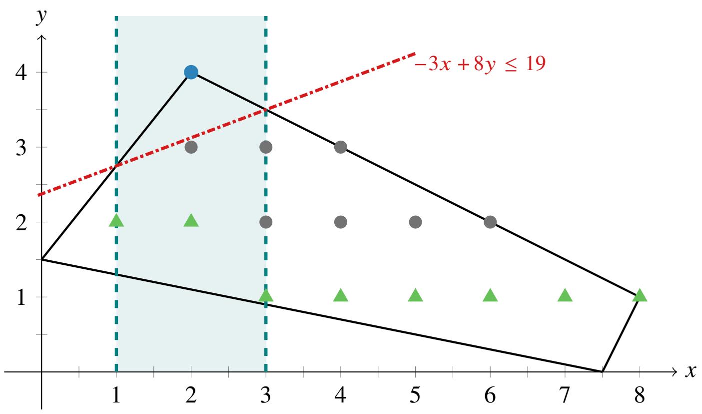


Figure 12.6 An illustration of a hypercube intersection cut for the bilevel problem in Example 12.23. Starting with the optimal solution $(2, 4)$ to the single-level relaxation, we derive the intersection cut $-3x + 8y \le 19$ from the hypercube set $\mathrm{HC}^+(x^*) = \{(x, y) : 1 \le x \le 3\}$. Dashed lines are faces of the set $\mathrm{HC}^+(x^*)$. The dash-dotted line represents the intersection cut.

off different areas of the single-level relaxation, which indicates that both types of constraints should be considered in practical implementations.

Example 12.23 (Example 6.2—Intersection cuts derived from $\mathrm{HC}^+(x^*)$) We again start with the optimal solution $(x^*, y^*) = (2, 4)$ to the single-level relaxation of the problem in Example 6.2, which is not bilevel feasible. This time, we find the optimal follower response $\hat{y} = 2$, store the incumbent solution $(2, 2)$, and then construct a hypercube around the point $x^* = 2$. Note that we do not need to solve the x^*-parameterized refinement problem in this example because there are no coupling constraints in the upper level. We obtain

$$\mathrm{HC}^+(x^*) = \{(x, y) : 1 \le x \le 3\}.$$

After intersecting the faces of this polyhedron with the two extreme rays emanating from the corner point $(2, 4)$ associated with constraints $-25x + 20y \le 30$ and $x + 2y \le 10$, respectively, we obtain the intersection cut $-3x + 8y \le 19$. This cut is illustrated in Figure 12.6. △

We conclude this chapter by noticing that hypercube intersection cuts are usually weaker than the other two alternatives derived using an improving solution or an improving direction. However, this is not always the case. By comparing the cut obtained from $\mathcal{T}^+(\hat{y})$ with $\hat{y} = 3$ (see Figure 12.4) with the cut derived from $\mathrm{HC}^+(x^*)$ with $x^* = 2$ (see Figure 12.6), we observe that these two cuts do not dominate each other.

Exercise 12.24 Insert the intersection cut $y \leq 2$ into the single-level relaxation of the bilevel problem considered in Example 12.21 and do the following.

(i) Find the optimal solution (x^*, y^*) to the resulting problem.
(ii) Is the obtained solution bilevel feasible?
(iii) If not, find two bilevel-free sets $\mathcal{T}(\hat{y})$—one using the optimal follower response $\hat{y}$ for the given x^* and the other by solving problem (12.9).
(iv) Derive intersection cuts from these two bilevel-free sets and find the new optimal solution to the resulting problem to which these cuts are added.

Exercise 12.25 Insert the intersection cut $y \leq 2$ into the single-level relaxation of the bilevel problem considered in Example 12.21, consider the optimal solution (x^*, y^*) to the resulting problem (see Exercise 12.24), and do the following.

(i) Find the improving direction Δy around the point (x^*, y^*) and the corresponding set $X^+(\Delta y)$.
(ii) Derive the intersection cut from this bilevel-free set and find the new optimal solution to the resulting problem to which this cut is added.

Exercise 12.26 (i) Consider the optimal solution (x^*, y^*) to the problem obtained by inserting the intersection cut $y \leq 2$ into the single-level relaxation of the bilevel problem considered in Example 12.21. You already know this optimal solution from the last two exercises. Find the incumbent solution associated with x^* and derive the resulting hypercube intersection cut. Then, find the new optimal solution to the resulting problem to which this cut is added.

(ii) Find the optimal solution (x^*, y^*) to the problem obtained by inserting the intersection cut $y \leq 3$ into the single-level relaxation of the bilevel problem considered in Example 12.21. Find the incumbent solution associated with x^* and derive the resulting hypercube intersection cut. Then, find the new optimal solution to the resulting problem to which this cut is added.

12.6 What You Should Know Now!

1. What is the major difference between intersection cuts for MILPs discussed in Chapter 11 and intersection cuts for MILP-MILP bilevel problems?
2. Which properties does a bilevel-free set have to satisfy to be able to generate a valid intersection cut?
3. How do we derive intersection cuts for MILP-MILP bilevel problems?

4. How can we define bilevel-free sets when exploiting the fact that, for the current bilevel-infeasible point (x^*, y^*), y^* is not an optimal follower response for x^*? How can we separate the associated intersection cuts?
5. How can we define bilevel-free sets when exploiting the fact that there is an improving direction of the objective function? How can we separate the associated intersection cuts?
6. How can we define bilevel-free sets when exploiting the fact that the current point (x^*, y^*), even though it satisfies integrality conditions, is not bilevel feasible? How can we separate the associated intersection cuts?
7. How can we integrate intersection cuts into a branch-and-cut framework for solving bilevel MILP-MILPs?

13
Primal Heuristics

We have seen in Section 6.4 that mixed-integer linear bilevel problems are Σ_2^p-hard in general, which makes them very challenging to solve. Hence, it also makes sense to discuss primal heuristics for these problems. Primal heuristics aim to compute bilevel-feasible points of good quality quickly. However, they usually do not provide any optimality guarantee. Despite significant advances in computational mixed-integer bilevel optimization in recent years, primal heuristics for bilevel MILPs remain scarce and most existing approaches are tailored to special cases; see, e.g., the methods in Beck et al. (2026), DeNegre (2011), Fischetti et al. (2018b), and Tahernejad et al. (2020). In the same spirit, the methods we discuss in this chapter also rely on certain structural assumptions for the underlying bilevel MILPs. In what follows, we present two heuristics: the *follower-priority heuristic* (DeNegre 2011; Tahernejad et al. 2020) and the *penalize-and-relax-and-dualize heuristic* for a special type of interdiction problems (Fischetti et al. 2018b). Both heuristics are designed for problems without coupling constraints.

13.1 Follower-Priority Heuristic

The follower-priority heuristic (also called *second-level priority heuristic*) builds on Section 2.2.1 in DeNegre (2011) and Section 2.5 in Tahernejad et al. (2020). We consider bilevel MILPs of the form

$$
\begin{aligned}
\min_{x \in X, y} \quad & c_x^\top x + c_y^\top y \\
\text{s.t.} \quad & Ax \geq a, \\
& y \in \arg\min_{\bar{y} \in Y} \left\{ d^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\}
\end{aligned}
\tag{13.1}
$$

with vectors a, b, c_x, c_y, d, and matrices A, C, D of appropriate dimension. As before, the sets X and Y are used to impose integrality constraints on (a subset of) the x- and y-variables, respectively. Note that we do not consider coupling constraints in Problem (13.1). To motivate the follower-priority heuristic, let us first have another look at the single-level relaxation of Problem (13.1), i.e., the mixed-integer linear problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax \geq a, \\ & Cx + Dy \geq b, \\ & x \in X, \ y \in Y. \end{aligned}$$

An optimal solution (x^*, y^*) to the single-level relaxation (if it exists) is typically not feasible for the original bilevel problem (13.1). This means that we usually have

$$d^\top y^* > \varphi(x^*) := \min_{y \in Y} \left\{ d^\top y \colon Cx^* + Dy \geq b \right\},$$

which implies that y^* is not an optimal response of the follower. In the branch-and-bound and the branch-and-cut methods discussed in the previous chapters, we have thus incorporated a bilevel-feasibility check to compute an optimal follower's response for the given leader's decision x^*. Afterward, we have discarded the bilevel-infeasible point (x^*, y^*) by branching or by using cutting planes. Still, as we have seen in our illustrative examples, computing a first bilevel-feasible point may require processing several nodes of the respective solution method. Our goal now is to compute a feasible point for the bilevel MILP (13.1) more quickly by prioritizing the optimality of the follower (hence the name "follower-priority heuristic") and thereby addressing the issues of non-optimal responses of the follower. We achieve this by solving the single-level relaxation of Problem (13.1) in which we replace the objective function with the one of the follower, i.e., we consider the mixed-integer linear problem

$$\begin{aligned} \min_{x,y} \quad & d^\top y \\ \text{s.t.} \quad & Ax \geq a, \\ & Cx + Dy \geq b, \\ & x \in X, \ y \in Y. \end{aligned} \tag{13.2}$$

In Problem (13.2), the follower essentially chooses a decision x of the leader that suits him best.

Theorem 13.1 *Suppose that the shared constraint set of Problem* (13.1) *is* *non-empty and bounded. Then, Problem* (13.2) *has an optimal solution* (x^*, y^*)*, which is feasible for the bilevel MILP* (13.1)*.*

Proof: Because the shared constraint set Ω is described by a finite number of continuous functions, it is also closed. Hence, Ω is compact and, by the Weierstraß theorem, Problem (13.2) has an optimal solution (x^*, y^*). In particular, the latter is in the shared constraint set of Problem (13.1). The only thing that is left to prove the claim is to show that y^* is an optimal response of the follower to the leader's decision x^*. Because there are no coupling constraints and the shared constraint set is non-empty and compact, the lower-level feasible set that is parameterized by x^*, i.e., the set

$$\{y \in Y : Cx^* + Dy \geq b\}$$

is non-empty and compact as well. The Weierstraß theorem thus yields that the x^*-parameterized lower-level problem

$$\min_{y \in Y} \quad d^\top y \quad \text{s.t.} \quad Cx^* + Dy \geq b$$

has an optimal solution $\hat{y}$. Hence, the pair $(x^*, \hat{y})$ is feasible for both the mixed-integer linear problem (13.2) and the bilevel MILP (13.1). If $d^\top y^* > d^\top \hat{y}$ holds, i.e., y^* is not an optimal response to the leader's decision x^*, we obtain a contradiction to the optimality of (x^*, y^*) for Problem (13.2). This concludes the proof. □

Problem (13.2) is a single-level mixed-integer linear problem that can be solved using any general-purpose MILP solver; see Section 3.4 for a list of possible options. If one is only interested in obtaining a bilevel-feasible point quickly, Theorem 13.1 states that solving Problem (13.2) is sufficient. This means that only one MILP needs to be solved. However, an optimal solution (x^*, y^*) to Problem (13.2) may be of rather poor quality because the objective of the leader is not taken into account. To find a trade-off between the computational burden and the quality of the obtained solutions, the idea is now to add another constraint to Problem (13.2) that bounds the leader's objective function value from above to ensure a certain quality level of the solution. To this end, we consider the restricted mixed-integer linear problem

$$\begin{aligned} \min_{x,y} \quad & d^\top y && \text{(13.3a)}\\ \text{s.t.} \quad & Ax \ge a, && \text{(13.3b)}\\ & Cx + Dy \ge b, && \text{(13.3c)}\\ & c_x^\top x + c_y^\top y \le \mathcal{U}, && \text{(13.3d)}\\ & x \in X, \ y \in Y, && \text{(13.3e)} \end{aligned}$$

for a given parameter $\mathcal{U} \in \bar{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$.

Lemma 13.2 *Suppose that the shared constraint set of Problem (13.1) is non-empty and bounded. Then, for given $\mathcal{U} \in \bar{\mathbb{R}}$, Problem (13.3) is either infeasible or it admits an optimal solution (x^*, y^*).*

Proof: Depending on the choice of $\mathcal{U}$, the set

$$\{(x, y) \in X \times Y : Ax \ge a, \ Cx + Dy \ge b\} \cap \left\{(x, y) : c_x^\top x + c_y^\top y \le \mathcal{U}\right\}$$

is either empty or non-empty and bounded. In the latter case, applying the Weierstraß theorem yields the claim. □

Depending on the choice of the parameter $\mathcal{U}$, an optimal solution (x^*, y^*) to Problem (13.3), if it exists, may no longer be bilevel feasible as Constraint (13.3d) leads to a restriction of the follower's response. To obtain a pair (x, y) that is feasible for Problem (13.1), we thus need to include a bilevel-feasibility check again. This means that we compute an optimal solution $\tilde{y}$ to the x^*-parameterized lower-level problem, i.e., we solve another MILP. If $d^\top y^* \le d^\top \tilde{y}$ holds, the point (x^*, y^*) is bilevel feasible. Otherwise, we report the bilevel-feasible point $(x^*, \tilde{y})$. The latter, however, may not meet the desired quality level anymore. Nevertheless, we can use this mechanism as an easy-to-implement primal heuristic in which bilevel-feasible points are generated by solving the restricted problem (13.3) for a given $\mathcal{U}$ and by performing a bilevel-feasibility check. Moreover, the method can be embedded into an iterative framework in which the value of the parameter $\mathcal{U}$ is gradually decreased, aiming for bilevel-feasible points of better quality. Overall, only MILPs are solved in such a heuristic framework.

13.2 Penalize-and-Relax-and-Dualize Heuristic

The second primal heuristic for bilevel MILPs that we discuss in this chapter has been proposed in Fischetti et al. (2018b) and is tailored to a special type of interdiction problems. To this end, we consider bilevel problems of the form

$$\begin{aligned} \min_{x \in X, y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax \geq a, \\ & x_i \in \{0,1\} \quad \text{for all } i \in L, \\ & y \in \mathcal{S}(x) \end{aligned} \tag{13.4}$$

with $\mathcal{S}(x)$ being the set of optimal solutions to the x-parameterized lower-level problem

$$\max_{y \in Y} \quad d^\top y \tag{13.5a}$$

$$\text{s.t.} \quad Dy \leq b, \ y \geq 0, \tag{13.5b}$$

$$y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L, \tag{13.5c}$$

$$y_i \in \mathbb{Z} \quad \text{for all } i \in L. \tag{13.5d}$$

Here, a, b, c_x, c_y, d, A, and D are vectors and matrices of appropriate dimension and the sets X and Y are used to impose integrality constraints on (a subset of) the x- and y-variables. Note that we explicitly require some components of x, namely x_i with $i \in L \subseteq \{1, \ldots, n_x\}$, to be binary; see also Assumption 10.2 and the respective discussion. Note further that the considered bilevel problem does not have any coupling constraints. This specific structure is essential for developing the penalize-and-relax-and-dualize heuristic. All remaining integrality restrictions for the leader's variables (if any) are captured in the set X. Moreover, we have a vector of finite upper bounds $u \in \mathbb{R}^{n_y}_{\geq 0}$ for the follower's variables. For the ease of presentation, let us assume in the following that the variable bounds $y \leq u$ are encoded in the system $Dy \leq b$.

Problem (13.4) is called a *generalized interdiction problem*. On the one hand, this is because the leader has the ability to prohibit the use of certain variables by the follower as it is the case for classic interdiction problems; see Chapter 10. This feature is captured by the constraints in (13.5c), where we account for the situation in which a subset $L \subseteq \{1, \ldots, n_y\}$ of the follower's variables may be interdicted by the leader. On the other hand, the setting considered in this chapter generalizes classic interdiction problems in the sense that the leader and the follower may have different objective functions, rather than optimizing the same function in opposing directions. The value-function reformulation of

Problem (13.4) now reads

$$\begin{aligned}
\min_{x,y} \quad & c_x^\top x + c_y^\top y && (13.6a)\\
\text{s.t.} \quad & Ax \geq a, && (13.6b)\\
& Dy \leq b, \; y \geq 0, && (13.6c)\\
& y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L, && (13.6d)\\
& d^\top y \geq \varphi(x), && (13.6e)\\
& x_i \in \{0, 1\}, \; y_i \in \mathbb{Z} \quad \text{for all } i \in L, && (13.6f)\\
& x \in X, \; y \in Y, && (13.6g)
\end{aligned}$$

with the lower-level optimal-value function

$$\varphi(x) = \max_{y \in Y} \left\{ d^\top y \colon Dy \leq b, \; y \geq 0, \; y_i \leq u_i(1 - x_i), \; y_i \in \mathbb{Z}, \; i \in L \right\}.$$

The heuristic for Problem (13.4) presented in Fischetti et al. (2018b), where it is called `ONE-SHOT`, builds on three main steps—penalization, relaxation, and dualization. Hence, we refer to this method as the penalize-and-relax-and-dualize heuristic. In the following sections, we discuss these three main steps in detail.

13.2.1 Penalization

In the lower-level problem (13.5), only the constraints but not the objective function depend on the leader's decision x. Our goal now is to reformulate the problem so that, instead, only the objective function but not the constraints depends on x. To this end, we consider the penalty reformulation of the follower problem

$$\begin{aligned}
\max_{y \in Y} \quad & d^\top y - \sum_{i \in L} M_i x_i y_i \\
\text{s.t.} \quad & Dy \leq b, \; y \geq 0, \\
& y_i \in \mathbb{Z} \quad \text{for all } i \in L,
\end{aligned} \qquad (13.7)$$

with sufficiently large big-M values obtained from Theorem 10.13. Recall that the constraints $y \leq u$ are assumed to be included in $Dy \leq b$. We discuss the advantage of considering the penalty reformulation (13.7) in more detail at the end of this section.

In Problem (13.7), only the objective function but not the feasible set depends on the leader's variables x. To simplify the presentation, let us thus re-state

Problem (13.7) as

$$\begin{aligned}\max_{y \in Y} \quad & d(x)^\top y \\ \text{s.t.} \quad & Dy \le b, \ y \ge 0, \\ & y_i \in \mathbb{Z} \quad \text{for all } i \in L,\end{aligned}$$

with

$$d(x)_i := \begin{cases} d_i(1 - M_i x_i), & i \in L, \\ d_i, & \text{otherwise.}\end{cases}$$

By Theorem 10.13, we then have

$$\varphi(x) = \max_{y \in Y} \left\{ d(x)^\top y \colon Dy \le b, \ y \ge 0, \ y_i \in \mathbb{Z}, \ i \in L \right\},$$

so that Constraint (13.6e) can be expressed in terms of the penalty reformulation of the lower-level problem.

13.2.2 Relaxation of the Lower-Level Integrality Constraints

Let us now relax the integrality constraints for the follower's variables in Problems (13.6) and (13.7). This means that we consider the problem

$$\begin{aligned}\min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax \ge a, \\ & Dy \le b, \ y \ge 0, \\ & y_i \le u_i(1 - x_i) \quad \text{for all } i \in L, \\ & d^\top y \ge \bar{\varphi}(x), \\ & x_i \in \{0, 1\} \quad \text{for all } i \in L, \\ & x \in X, \ y \in \mathbb{R}^{n_y},\end{aligned} \tag{13.8}$$

where $\bar{\varphi}(x)$ is the optimal-value function of the continuous relaxation of the penalized lower-level problem (13.7), i.e.,

$$\bar{\varphi}(x) = \max_{y \in \mathbb{R}^{n_y}} \left\{ d(x)^\top y \colon Dy \le b, \ y \ge 0 \right\}. \tag{13.9}$$

The latter is a relaxation of Problem (13.7) and, thus, $\bar{\varphi}(x) \ge \varphi(x)$ holds for a given leader's decision x. However, this implies that Problem (13.8) is, in general, neither a relaxation nor a restriction of Problem (13.6); see Examples 8.5–8.7. As a result, solving Problem (13.8) neither yields a valid lower nor a valid upper bound for the original bilevel problem (13.4) in general. Nevertheless, we show next how Problem (13.8) can still be used to compute a bilevel-feasible pair (x, y).

13.2.3 Dualization

For a given leader's decision x, Problem (13.9) is a linear optimization problem. This means that a single-level reformulation of Problem (13.8) can be obtained by reformulating the value-function constraint $d^\top y \geq \bar{\varphi}(x)$ using strong duality; see Section 3.3. To this end, let us state the dual of Problem (13.9), which is given by

$$\min_{\lambda \in \mathbb{R}^\ell} \quad b^\top \lambda \quad \text{s.t.} \quad D^\top \lambda \geq d(x),\ \lambda \geq 0.$$

The strong-duality based reformulation of Problem (13.8) then reads

$$\begin{aligned} \min_{x,y,\lambda} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax \geq a, \\ & Dy \leq b,\ y \geq 0, \\ & y_i \leq u_i(1 - x_i) \quad \text{for all } i \in L, \\ & d^\top y \geq b^\top \lambda, \\ & D^\top \lambda \geq d(x), \\ & x_i \in \{0,1\} \quad \text{for all } i \in L, \\ & x \in X,\ y \in \mathbb{R}^{n_y},\ \lambda \in \mathbb{R}^\ell_{\geq 0}. \end{aligned} \tag{13.10}$$

Problem (13.10) is a mixed-integer linear problem and can be solved using any general-purpose MILP solver; see Section 3.4 again for a list of possible options. However, the pair (x^*, y^*) obtained from an optimal solution (x^*, y^*, λ^*) to Problem (13.10) may not be feasible for the original bilevel problem (13.4), which also includes integrality constraints for the follower's variables. It may even be the case that $y^* \notin Y$ holds. Does this mean that our previous derivations have been pointless? Not necessarily. By exploiting structural information of the lower-level problem—even in its relaxed form—it is more likely that an optimal leader's decision x^* obtained from Problem (13.10) is close to a "true" optimal one. To compute a bilevel-feasible point, it then only remains to solve the x^*-parameterized lower-level problem. Summarizing all previous derivations, we now finally state the penalize-and-relax-and-dualize heuristic in Algorithm 10.

In Line 1 of Algorithm 10, we compute an optimal solution $(\hat{x}, \hat{y}, \hat{\lambda})$ to Problem (13.10). By construction, the pair $(\hat{x}, \hat{y}) \in X \times \mathbb{R}^{n_y}$ satisfies

$$\begin{aligned} & A\hat{x} \geq a, \quad D\hat{y} \leq b, \quad \hat{y} \geq 0, \\ & \hat{y}_i \leq u_i(1 - \hat{x}_i), \quad \hat{x}_i \in \{0,1\} \quad \text{for all } i \in L. \end{aligned}$$

However, $(\hat{x}, \hat{y})$ may not be feasible for the bilevel problem (13.4). To ensure bilevel feasibility, we need to check whether $\hat{y}$ satisfies all integrality constraints,

Algorithm 10 Penalize-and-Relax-and-Dualize Heuristic

Input: An instance of Problem (13.4) with a non-empty and bounded shared constraint set

Output: A feasible point (x^*, y^*) for Problem (13.4)

1: Compute an optimal solution $(\hat{x}, \hat{y}, \hat{\lambda})$ to Problem (13.10).

2: Solve the $\hat{x}$-parameterized lower-level problem

$$\varphi(\hat{x}) := \max_{y \in Y} \left\{ d^\top y \colon Dy \leq b,\ y \geq 0,\ y_i \leq u_i(1 - \hat{x}_i),\ i \in L \right\}.$$

3: **if** $\hat{y} \in Y$ and $d^\top \hat{y} \geq \varphi(\hat{x})$ **then**

4: Set $(x^*, y^*) \leftarrow (\hat{x}, \hat{y})$.

5: **else**

6: Compute an optimal solution $(\tilde{x}, \tilde{y})$ to the $\hat{x}_L$-parameterized refinement problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax \geq a,\ Dy \leq b,\ y \geq 0, \\ & x_i = \hat{x}_i \quad \text{for all } i \in L, \\ & y_i \leq u_i(1 - \hat{x}_i) \quad \text{for all } i \in L, \\ & d^\top y \geq \varphi(\hat{x}), \\ & x \in X,\ y \in Y, \end{aligned}$$

and set $(x^*, y^*) \leftarrow (\tilde{x}, \tilde{y})$.

7: **return** (x^*, y^*)

i.e., $\hat{y} \in Y$, and whether $\hat{y}$ is indeed an optimal response of the follower. This is done in the remainder of Algorithm 10. In Line 2, we compute the optimal objective function value $\varphi(\hat{x})$ of the $\hat{x}$-parameterized lower-level problem. If $\hat{y} \in Y$ and $d^\top \hat{y} \geq \varphi(\hat{x})$ holds, $\hat{y}$ is an optimal follower's response that satisfies all integrality restrictions. Hence, the method terminates with the bilevel-feasible point $(\hat{x}, \hat{y})$. Otherwise, we compute an optimistic follower's response $\tilde{y}$ in Line 6 by solving the refinement problem for the fixed leader's decision $\hat{x}$; see Definition 6.15 and the respective discussion. By construction, the pair $(\tilde{x}, \tilde{y})$ obtained in Line 6 is feasible for Problem (13.4). To improve the quality of heuristically obtained solutions, Algorithm 10 can also be embedded into an iterative framework. We do not go into the details here but refer to the `ITERATE` and the `DYN-REF` methods in Fischetti et al. (2018b) for further information.

To conclude this chapter, let us now finally discuss the advantage of using the penalty reformulation (13.7) to derive the heuristic in Algorithm 10. In Problem (13.7), only the objective function but not the feasible set depends on the leader's variables x. As a result, no bilinearities are present in the dual of Problem (13.7) because only the right-hand side vector of the dual feasible region depends on x. This is in contrast to the strong-duality based reformulations discussed in Sections 3.3 and 5.3, which contain nonconvexities due to products of primal leader and dual follower variables. Hence, by applying a penalty reformulation to the lower-level problem, we avoid such complications and obtain formulations that are easier to handle in practice.

Exercise 13.3 (Bilevel Knapsack Interdiction—Revisited) Consider the bilevel knapsack interdiction problem in Exercise 8.18 again, i.e., consider the problem

$$\begin{aligned} \min_{x,y} \quad & 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 2x_1 + x_2 + x_3 \leq 2, \\ & x \in \{0,1\}^3,\ y \in \mathcal{S}(x), \end{aligned}$$

where $\mathcal{S}(x)$ is the set of optimal solutions to

$$\begin{aligned} \max_{y} \quad & 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 4y_1 + 3y_2 + 2y_3 \leq 4, \\ & y_i \leq 1 - x_i \quad \text{for all } i \in \{1,2,3\}, \\ & y \in \{0,1\}^3. \end{aligned}$$

(i) Use Theorem 10.13 to derive sufficiently large big-M values and write down the penalty formulation (13.7) of the lower-level problem.
(ii) Compare the big-M values determined in (i) with the choice $M = (4,3,3)^\top$, which can be obtained because of the downward monotonicity property; see Section 10.6. What can you say about the strength of the two penalty formulations, one using the big-M values from (i) and the other using $M = (4,3,3)^\top$?
(iii) Relax the integrality constraints of the follower's variables in the penalty reformulation derived in (ii) by replacing $y \in \{0,1\}^3$ with $y \in [0,1]^3$.
(iv) Write down the dual of the formulation derived in (iii).
(v) Use (iv) to state the single-level mixed-integer linear formulation (13.10) for the bilevel knapsack interdiction problem.
(vi) Use a general-purpose MILP solver to compute an optimal solution $(\hat{x}, \hat{y}, \hat{\lambda})$ to the MILP derived in (v). Is the pair $(\hat{x}, \hat{y})$ bilevel feasible? If yes, explain why. If not, compute a bilevel-feasible pair $(\tilde{x}, \tilde{y})$.

(vii) Compare the bilevel-feasible pair obtained in (vi) with the global optimal solution to the bilevel knapsack interdiction problem derived in Exercise 8.18. What can you say about the quality of the heuristic solution?

13.3 What You Should Know Now!

1. What is the idea behind the follower-priority heuristic for bilevel MILPs? Why is it called like that?
2. What types of bilevel problems can be addressed using the follower-priority heuristic? What assumptions are made about the structure of the considered bilevel MILPs and why are they needed?
3. What may be a drawback of the restricted mixed-integer linear problem (13.3) when determining bilevel-feasible points?
4. What can you say about the hardness of the optimization problems solved in the follower-priority heuristic?
5. What types of bilevel problems can be addressed using the penalize-and-relax-and-dualize heuristic? What assumptions are made about the structure of the considered bilevel MILPs and at which steps in (the derivation of) the method are these assumptions explicitly used?
6. What can you say about the hardness of the optimization problems solved in Algorithm 10?
7. What are the three main steps involved to arrive at the penalize-and-relax-and-dualize heuristic?
8. Can you explain why we consider a penalty reformulation of the lower-level problem? What is the advantage of this reformulation and what would happen if we skip the penalization step?

PART FOUR

FURTHER TOPICS

14
Bilevel Optimization Under Uncertainty

As we have seen in the previous chapters, bilevel optimization is a powerful tool for modeling hierarchical decision-making processes. However, the resulting problems are challenging to solve—both in theory and practice. Fortunately, there have been significant theoretical and algorithmic advances in the field so that we can solve much larger and also more complicated problems today compared to what was possible to solve two decades ago. Many of these advances have been discussed in the last chapters, including tailored branch-and-bound and branch-and-cut methods using interdiction or intersection cuts. This results in more and more challenging bilevel problems that researchers try to solve today. In this chapter, we give a brief and highly selective primer on one of these more challenging classes of bilevel problems: bilevel problems under uncertainty. To this end, we briefly state different types of uncertain bilevel problems that result from different levels of cooperation of the follower as well as from when (over time) the uncertainty is revealed. Moreover, we illustrate these concepts using academic examples. We also discuss that the sources of uncertainty in bilevel optimization are much richer than in single-level optimization and, to this end, introduce the concept of *decision uncertainty*. This chapter is mainly based on the recent survey article by Beck et al. (2023) and the introductory article on robust bilevel optimization by Beck et al. (2022). Very recently, the general connections between robust and bilevel optimization have been studied as well. The interested reader can find the details in Goerigk et al. (2025).

14.1 Motivation

The field of bilevel optimization under uncertainty is still in its infancy but, due to its relevance in many practical applications, it is developing quickly. In classic, i.e., single-level, optimization, there are two major approaches to address uncertainty in the problem data: stochastic optimization (Birge and Louveaux 2011; Kall and Wallace 1994) and robust optimization (Ben-Tal et al. 2009; Ben-Tal and Nemirovski 1998; Bertsimas et al. 2011; Soyster 1973). The same two paths have been followed as well in bilevel optimization starting from the 1990s. However, the sources of uncertainty are much richer in bilevel optimization compared to single-level optimization. To make this more concrete, let us consider the linear optimization problem

$$\min_x \quad c^\top x \quad \text{s.t.} \quad Ax \geq a.$$

This problem can "only" be subject to uncertainty because of uncertainties in the problem data c, A, and a. Throughout this chapter, we refer to this setting as *data uncertainty*. A bilevel optimization problem may be subject to another source of uncertainty, which is due to its nature to combine different decision makers in one model. Hence, there can be further uncertainty involved either if the leader is not sure about the reaction of the follower or if the follower is not certain about the observed leader's decision. We denote this additional type of uncertainty as *decision uncertainty*. Decision uncertainty does not play any role in single-level optimization because only one decision maker is involved.

In what follows, we give examples for both data and decision uncertainty, as well as for both robust and stochastic techniques to tackle them. For more details and a recent overview of the respective literature, we refer to the survey by Beck et al. (2023). Before turning to uncertain settings, let us start with the deterministic bilevel problem

$$\text{“}\min_{x \in X}\text{”} \quad F(x, y) \tag{14.1a}$$

$$\text{s.t.} \quad G(x) \geq 0, \tag{14.1b}$$

$$y \in \mathcal{S}(x), \tag{14.1c}$$

that we discussed throughout this book, i.e., $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized problem

$$\min_{y \in Y} \quad f(x, y) \quad \text{s.t.} \quad g(x, y) \geq 0.$$

Here, we focus on the setting without coupling constraints, i.e., without upper-level constraints that depend on the lower-level variables y. This is in line with most of the literature in bilevel optimization under uncertainty because coupling constraints seem to significantly increase the difficulty of tackling such problems. Note that, in the case that the lower-level problem does not have a unique solution, the bilevel problem (14.1) can be ill-posed; see Chapter 2. This ambiguity is expressed by the quotation marks in (14.1a). As discussed in

Chapter 2, it is common to pursue either an optimistic or a pessimistic approach to bilevel optimization; see, e.g., Aussel and Svensson (2019a), Dempe (2002), Dempe et al. (2014), and Wiesemann et al. (2013) for more information on the pessimistic case that we have not treated in detail in this book so far. In the optimistic setting, the leader chooses the follower's response among multiple optimal solutions to the lower-level problem such that it favors the leader's objective function value. Hence, the leader also minimizes her objective in the y variables, i.e., we consider the problem

$$\min_{x \in \bar{X}} \min_{y \in \mathcal{S}(x)} F(x, y) \tag{14.2}$$

with $\bar{X} := \{x \in X : G(x) \geq 0\}$ and $G : \mathbb{R}^{n_x} \to \mathbb{R}^m$. In the pessimistic setting, the leader anticipates that, among multiple optimal solutions of the follower, the worst possible response w.r.t. the upper-level objective function is chosen. Thus, one studies the problem

$$\min_{x \in \tilde{X}} \max_{y \in \mathcal{S}(x)} F(x, y)$$

with

$$\tilde{X} = \bar{X} \cap \{x \in \mathbb{R}^{n_x} : \mathcal{S}(x) \neq \emptyset\} .$$

14.2 Data Uncertainty

We now focus on bilevel problems that are affected by data uncertainty. Data uncertainty arises when some of the players only have access to inaccurate or incomplete data. In robust optimization, it is assumed that these uncertainties take values in a given, and usually non-empty and compact, uncertainty set $\mathcal{U}$. The uncertainty sets are typically modeled using boxes, polyhedra, ellipsoids, or cones; see, e.g., Ben-Tal et al. (2009, 2004), Ben-Tal and Nemirovski (1998), Bertsimas et al. (2011), and Soyster (1973). In the context of single-level robust optimization, there are two possibilities to hedge against data uncertainty, which we illustrate in the following.

First, assuming that the coefficients of the objective function are uncertain, one searches for a solution that is optimal for the worst-case realization of the uncertain parameters. The problem can be modeled as

$$\min_{x \in \bar{X}} \max_{u \in \mathcal{U}} F(x, u), \tag{14.3}$$

where the objective function $F : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ is continuous, the set $\mathcal{U} \subseteq \mathbb{R}^{n_u}$ is non-empty and compact, and $\bar{X}$ is defined as above.

Second, in the case that the objective function is certain and that uncertainty only affects the constraints, one is interested in a solution that is feasible for all possible realizations of the uncertainty. This problem can be stated as

$$\min_{x \in X} \quad F(x) \quad \text{s.t.} \quad G(x,u) \geq 0 \quad \text{for all } u \in \mathcal{U}, \tag{14.4}$$

where we consider the objective function $F : \mathbb{R}^{n_x} \to \mathbb{R}$ and the constraint function $G : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^m$. Problem (14.4) can be reformulated as

$$\min_{x \in X} \quad F(x) \quad \text{s.t.} \quad \min \{G(x,u) \colon u \in \mathcal{U}\} \geq 0. \tag{14.5}$$

Moreover, Problem (14.3) can be re-stated as an instance of Problem (14.5) using an epigraph reformulation, i.e.,

$$\min_{x \in \bar{X}, t \in \mathbb{R}} \quad t \quad \text{s.t.} \quad t \geq \max \{F(x,u) \colon u \in \mathcal{U}\}.$$

For more information about epigraph reformulations, we refer to the recent tutorial by Stein (2025).

Note that for the two settings discussed so far, a single decision maker has to take a so-called here-and-now decision, i.e., a decision is taken before the uncertainty is revealed. In bilevel optimization, however, two different timings are possible—one in which the uncertainty realizes after and one in which the uncertainty realizes before the follower makes his decision.

Here-and-Now Follower

In this case, both the leader and the follower have to make their decisions before the uncertainty is revealed, i.e., one considers the timing

$$\text{leader } x \quad \curvearrowright \quad \text{follower } y = y(x) \quad \curvearrowright \quad \text{uncertainty } u. \tag{14.6}$$

This means that the leader anticipates an optimal response of the follower who hedges against data uncertainty. To streamline the discussion, let us assume that the uncertainty does not affect the upper-level problem directly. Hence, the lower-level problem is an x-parameterized problem in which we can embed any of the concepts known for single-level optimization under uncertainty. In particular, either robust or stochastic optimization techniques can be applied and both the objective or the constraints can be uncertain. Out of the many possibilities for this overall setup, let us exemplarily consider the case in which only the lower-level objective function is uncertain and in which the follower is assumed to behave in an optimistic way using robust optimization. Then, we are facing Problem (14.2) with

$$\mathcal{S}(x) := \arg\min_{\bar{y} \in Y} \left\{ \max_{u \in \mathcal{U}} \; f(x,u,\bar{y}) \colon g(x,\bar{y}) \geq 0 \right\}.$$

Wait-and-See Follower

In this setting, the leader first takes a here-and-now decision, i.e., without knowing the realization of uncertainty. Then, the uncertainty is revealed and, finally, the follower decides in a wait-and-see fashion, taking the leader's decision as well as the realization of the uncertainty into account. Hence, one considers the timing

$$\text{leader } x \quad \curvearrowright \quad \text{uncertainty } u \quad \curvearrowright \quad \text{follower } y = y(x, u).$$

This means that the leader does not have full knowledge about the lower-level problem. Thus, she wants to hedge against the worst-case reaction of the follower. Here, "worst-case" may not only imply the robustness of the leader w.r.t. lower-level data uncertainty but also her conservatism regarding the cooperation of the follower. For instance, to protect against the worst-case realization of the uncertainties w.r.t. the leader's objective function, we consider the problem

$$\text{“}\min_{x \in \bar{X}} \max_{u \in \mathcal{U}}\text{”} \quad F(x, y) \quad \text{s.t.} \quad y \in \mathcal{S}(x, u), \tag{14.7}$$

where $\mathcal{S}(x, u)$ is the set of optimal solutions to the (x, u)-parameterized problem

$$\min_{y \in Y} \quad f(x, u, y) \quad \text{s.t.} \quad g(x, u, y) \geq 0.$$

The quotation marks in (14.7) again express the ill-posedness of the bilevel problem in the case in which the set $\mathcal{S}(x, u)$ is not a singleton. Hence, one also needs to distinguish between the optimistic and the pessimistic case in the robust setting. Indeed, both situations can be motivated by practical applications. On the one hand, the pessimistic robust bilevel problem appears when the leader wants to hedge against the worst-case both w.r.t. lower-level data uncertainty as well as w.r.t. the potentially unknown level of cooperation of the follower. On the other hand, there may also be situations in which the follower still hedges against his uncertainties in a robust way but, in the case of ambiguous optimal solutions, acts in an optimistic way. This might be the case in energy markets with sufficiently regulated agents, where a strong level of regulation might lead to an optimistic robust bilevel problem.

14.2.1 Academic Example: Robust Modeling

Let us consider the optimistic linear bilevel problem taken and adapted from Beck et al. (2023) that is given by

$$\begin{aligned} \min_{x \in \mathbb{R}, y} \quad & F(x, y) = x + y \\ \text{s.t.} \quad & 0.8 \leq x \leq 3.8, \\ & y \in \mathcal{S}(x), \end{aligned} \tag{14.8}$$

where $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized lower-level problem

$$\min_{y \in \mathbb{R}} \quad f(x, y) = f(y) = -0.1y \tag{14.9a}$$

$$\text{s.t.} \quad x - y \geq -1, \tag{14.9b}$$

$$3x + y \geq 3, \tag{14.9c}$$

$$-2x + y \geq -7, \tag{14.9d}$$

$$-3x - 2y \geq -14, \tag{14.9e}$$

$$0 \leq y \leq 2.5. \tag{14.9f}$$

The problem is depicted in Figure 14.1 (top). The upper- and lower-level constraints are represented with dashed black as well as solid black and dotted green lines, respectively. The optimal solution $(x^*, y^*) = (0.8, 1.8)$ is illustrated by the thick orange dot. Suppose now that the lower-level objective function is uncertain. To this end, we consider $\tilde{f}(x, u, y) = (-0.1 + u)y$ and assume that u takes values in the uncertainty set $\mathcal{U} = \{u \in \mathbb{R} \colon |u| \leq 0.5\}$. To illustrate the effect of this uncertainty on the solution, let us exemplarily consider a follower taking a here-and-now decision. We thus consider the timing in (14.6) so that the robustified lower-level problem is given by

$$\min_{y \in \mathbb{R}} \max_{u \in \mathcal{U}} \quad \tilde{f}(x, u, y) = (-0.1 + u)y \quad \text{s.t.} \quad (14.9b)\text{–}(14.9f).$$

Using classic techniques from robust optimization, we obtain a modified gradient of the lower-level objective function, which is shown in Figure 14.1 (bottom). The optimal solution $(x^*, y^*) = (1, 0)$ to this problem is again represented by the thick orange dot. Note that the robust solution is attained at a different vertex of the shared constraint set than in the deterministic case. Moreover, the bilevel-feasible sets in the deterministic and the robust setting are disjoint.

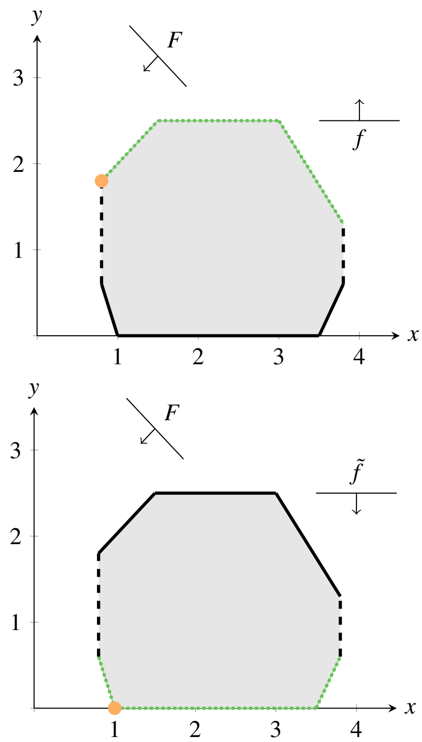


Figure 14.1 Both figures show the upper-level constraints (dashed black lines), the lower-level constraints (solid black and dotted green lines), the shared constraint set (gray area), the bilevel-feasible set (dotted green lines), and the bilevel optimal solution (orange point) for the bilevel problem (14.8). The deterministic variant of the problem is depicted in the top figure and the variant with a here-and-now follower is given in the bottom figure. Taken and modified from Beck et al. (2023).

14.2.2 Academic Example: Stochastic Modeling

We now consider another academic example for an uncertain bilevel problem in which data uncertainty is addressed using techniques from stochastic optimization. To this end, we assume that the uncertainties can be described by a given probability distribution and that we hedge against uncertainties in a probabilistic sense by optimizing, e.g., the expected value. The optimistic stochastic bilevel problem with a wait-and-see follower can then be stated as

$$\min_{x \in \bar{X}} \quad \mathbb{E}\left[\Phi_u(x)\right] \quad \text{with} \quad \Phi_u(x) := \min_{y \in \mathcal{S}(x,u)} F(x, y),$$

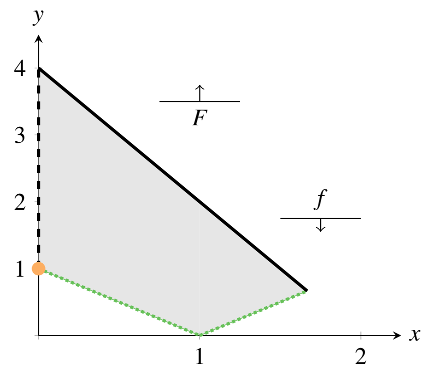


Figure 14.2 The upper-level constraint (dashed black line), the lower-level constraints (solid black and dotted green lines), the shared constraint set (gray area), the bilevel-feasible set (dotted green lines), and the optimal solution (orange point) for the bilevel problem (14.10). Taken and modified from Beck et al. (2023).

where $\mathbb{E}[\cdot]$ denotes the expected value w.r.t. the probability distribution of the uncertain parameter u.

Let us now move on to a simple example. We consider the linear bilevel problem

$$\min_{x,y\in\mathbb{R}} \quad F(x,y) = F(y) = -y \quad \text{s.t.} \quad x \geq 0,\ y \in \mathcal{S}(x), \tag{14.10}$$

where $\mathcal{S}(x)$ denotes the set of optimal solutions to the x-parameterized lower-level problem

$$\min_{y\geq 0} \quad f(x,y) = f(y) = y \tag{14.11a}$$
$$\text{s.t.} \quad x + y \geq 1, \tag{14.11b}$$
$$-x + y \geq -1, \tag{14.11c}$$
$$-2x - y \geq -4. \tag{14.11d}$$

The problem is depicted in Figure 14.2. The upper- and lower-level constraints are represented with dashed black as well as solid black and dotted green lines, respectively. The unique optimal solution $(x, y) = (0, 1)$ to the deterministic bilevel problem (14.10) is illustrated by the thick orange dot. Note that this problem is a min-max problem without coupling constraints, which is why we do not need to distinguish between the optimistic and pessimistic case.

Suppose now that the right-hand side of Constraint (14.11c) is uncertain and assume that the right-hand side $b(\omega) \in \mathbb{R}$ depends on the scenario $\omega \in \Omega = \{\omega^1, \omega^2\}$ with $b(\omega^1) = -1$ and $b(\omega^2) = -1/2$. We further assume that both scenarios have probability $p^1 = p^2 = 1/2$. We start by considering each scenario individually. Because we have a discrete and finite set of scenarios, we can use the so-called deterministic equivalent, which we derive in the following. Note that the realization of ω^1 corresponds to the deterministic setting. Hence, the unique optimal solution for scenario ω^1 is given by $(x, y^1) = (0, 1)$. Here and in what follows, let $\mathcal{S}(x, \omega^i)$ for $i = 1, 2$ be the set of optimal solutions to the lower-level problem, which is now parameterized by x and ω^i. With this at hand, we then write $y^i \in \mathcal{S}(x, \omega^i)$. The realization of scenario ω^2 leads to a parallel shift of the uncertain lower-level constraint. This effect is shown in Figure 14.3 (top). It can also be seen that the solution is not unique anymore if scenario ω^2 is considered. Both $(0, 1)$ and $(3/2, 1)$—which are illustrated by the thick orange dot and the thick orange square, respectively—yield an optimal objective function value of -1. To hedge against lower-level right-hand side uncertainty, we optimize the expected value of the upper-level objective function, i.e., we solve

$$\begin{aligned} \min_{x, y^1, y^2 \in \mathbb{R}} \quad & -p^1 y^1 - p^2 y^2 \\ \text{s.t.} \quad & x \geq 0, \ y^1 \in \mathcal{S}(x, \omega^1), \ y^2 \in \mathcal{S}(x, \omega^2). \end{aligned} \tag{14.12}$$

The unique solution to Problem (14.12) is given by $(x, y^1, y^2) = (0, 1, 1)$. Despite the consideration of data uncertainty, the overall bilevel solution does not change significantly compared to the deterministic setting. However, the following shows that this may not always be the case.

To this end, we assume that the lower-level right-hand side is certain and we now focus on the stochastic modeling of uncertain constraint coefficients in (14.11c). The constraint coefficients $a(\omega) \in \mathbb{R}^2$ are assumed to depend on the scenario $\omega \in \Omega = \{\omega^1, \omega^2\}$ with $a(\omega^1) = (-1, 1)$ and $a(\omega^2) = (-3/2, 1/2)$. We further assume that scenario ω^1 has probability $p^1 = 1/3$, whereas ω^2 has probability $p^2 = 2/3$. Again, the realization of scenario ω^1 corresponds to the deterministic setting. Thus, the unique optimal solution for scenario ω^1 is given by $(x, y^1) = (0, 1)$. The setting in which scenario ω^2 realizes is shown in Figure 14.3 (bottom). The unique optimal solution $(x, y^2) = (6/5, 8/5)$ is illustrated by the thick orange square. Hedging against data uncertainty by optimizing over the expected value yields the unique overall stochastic bilevel solution $(x, y^1, y^2) = (6/5, 1/5, 8/5)$, which can be obtained by solving the corresponding scenario-expanded formulation (14.12). In particular, the solution is attained at a different vertex of the bilevel-feasible set compared to the deterministic case.

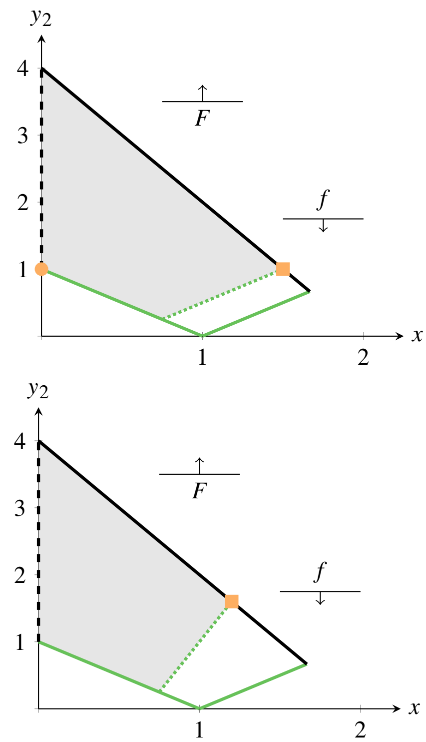


Figure 14.3 The bilevel problem (14.10) with lower-level right-hand side uncertainty (top) and uncertain lower-level constraint coefficients (bottom) for scenario ω^2. Taken and modified from Beck et al. (2023).

14.3 Decision Uncertainty

Up to now, both decision makers of the bilevel problem are assumed to take rational decisions in the sense that they can perfectly anticipate or observe the other's decision and that they can solve their respective problems to global optimality. In decision-making theory, however, it is well known that these assumptions regarding perfect information and rationality are rarely satisfied in a real-world context. As bilevel optimization involves two decision makers, this can lead to additional types of uncertainty. One of such types is decision uncertainty in which, e.g., the leader is not sure about the reaction of the follower (for instance if the follower does not necessarily respond with an optimal solution but with a heuristically chosen feasible point) or in which the follower is not sure about the observed leader's decision. We are not going into the details here but want to give a few pointers to the relevant literature that covers such aspects. If the leader is uncertain about her anticipation of the follower's optimal reaction and, thus, wants to hedge against sub-optimal follower reactions, the resulting setup can be modeled using so-called near-optimal robust bilevel models; see, e.g., Besançon et al. (2024) and Besançon et al. (2021). As an extreme case of the former aspect, it may be the case that the upper-level player knows that the follower will play against her. This is the setting of a pessimistic bilevel optimization problem, which is also rather naturally connected to the field of robust optimization; see, e.g., Aussel and Svensson (2019a), Dempe (2002), Dempe et al. (2014), and Wiesemann et al. (2013). However, if the level of cooperation of the follower is not known, this may lead to intermediate cases in between the optimistic and the pessimistic settings; see, e.g., Aboussoror and Loridan (1995) and Mallozzi and Morgan (1996) and the more recent paper by Salas and Svensson (2023). Moreover, in many situations it is not possible for the follower to perfectly observe the optimal decision of the leader and the follower may want to hedge against all possible leader decisions in some uncertainty set around the observation. Such settings are, e.g., tackled in Bagwell (1995) and van Damme and Hurkens (1997). Finally, even if all data and the rational reaction of the follower is known and even if the leader can, in principle, fully anticipate the (globally) optimal reaction of the follower, it may still be the case that limited intellectual or computational resources render it impossible for the follower to take a globally optimal decision. In such situations, a follower might resort to heuristic approaches and the leader may be uncertain w.r.t. which heuristic is used. For a good primer in this context, we refer to the paper by Zare et al. (2020).

Example: Robust Modeling of Limited Observability To give at least one specific example for decision uncertainty, we briefly present the approach by Beck and Schmidt (2021) to model so-called limited observability, i.e., the situation in which the follower cannot perfectly observe the decision of the leader. The resulting uncertainty in the model of the follower is treated using robust optimization in the following, but a stochastic treatment might be possible as well. Under perfect information, the (deterministic) linear bilevel problem is given by

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \arg\min_{\bar{y}} \left\{ d_x^\top x + d_y^\top \bar{y} \colon Cx + D\bar{y} \geq b \right\}. \end{aligned} \tag{14.13}$$

Note that we consider the optimistic approach to bilevel optimization. As Problem (14.13) is stated, we make the strong assumption that the leader and the follower act perfectly rational. In real-world applications, however, this assumption rarely holds as both players face bounded rationality; see, e.g., Simon (1972) for a general discussion and Pita et al. (2008) for applications using bilevel optimization. As mentioned before, we now assume that the follower cannot perfectly observe the actual leader's decision. Nevertheless, the observed upper-level decision x provides insight into the leader's scope of action. Given this knowledge, the follower's response is based on $\bar{x}$, which is assumed to belong to a given uncertainty set $\mathcal{U}(x)$. This leads to the robust bilevel problem

$$\begin{aligned} \min_{x,y} \quad & c_x^\top x + c_y^\top y \\ \text{s.t.} \quad & Ax + By \geq a, \\ & y \in \mathcal{S}(x), \end{aligned}$$

where $\mathcal{S}(x)$ is now the set of optimal solutions to the x-parameterized problem

$$\begin{aligned} \min_{y,\eta} \quad & d_y^\top y + \eta \\ \text{s.t.} \quad & \eta \geq d_x^\top \bar{x} \quad \text{for all } \bar{x} \in \mathcal{U}(x), \\ & C\bar{x} + Dy \geq b \quad \text{for all } \bar{x} \in \mathcal{U}(x). \end{aligned}$$

i.e., we face a robustified lower-level objective function as well as a robustified feasible set. Moreover the follower is taking a here-and-now decision. Note that because of the robustification of the lower-level objective function, the linear term in x cannot be avoided as it is usually done in linear bilevel optimization. This is due to the fact that, for uncertain x, this term is no longer constant from the follower's point of view. We do not go into the details here w.r.t. the structural properties of this model and potential single-level reformulations but refer to Beck and Schmidt (2021). In particular, let us mention that they also allow for bilinear bilevel problems, which capture the important class of so-called sequential bimatrix games.

The aspects of decision uncertainty listed above and the example of limited observability are, of course, by far not comprehensive. A much more detailed discussion of these and other aspects can be found in the recent survey by Beck et al. (2023). However, it is hopefully clear now how much more diverse the sources of uncertainty can be in bilevel optimization compared to single-level optimization. Hence, we expect a lot of research in the area of decision uncertainty in future years.

Exercise 14.1 (Bilevel Knapsack Problem) Consider an instance of the bilevel knapsack problem introduced in Dempe and Richter (2000) given by

$$\begin{aligned}\max_{x,y} \quad & -2x + 4y_1 + 3y_2 + 3y_3 \\ \text{s.t.} \quad & 3 \leq x \leq 6, \\ & y \in \mathcal{S}(x),\end{aligned}$$

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

$$\begin{aligned}\max_{y} \quad & 2y_1 + y_2 + y_3 \\ \text{s.t.} \quad & 4y_1 + 3y_2 + 2y_3 \leq x, \\ & y \in \{0, 1\}^3 .\end{aligned}$$

In this problem, the leader chooses the capacity of the knapsack and the follower selects items to pack into the knapsack. In what follows, we study different variants of this bilevel knapsack problem under uncertainty.

(i) Robust model under data uncertainty with a here-and-now follower (interval uncertainty): Suppose that the originally given item weights $w = (4, 3, 2)^\top$ are uncertain but known to take values within the intervals

$$w_1 \in [4, 6], \quad w_2 \in [3, 4], \quad w_3 \in [2, 3].$$

   (a) State the robust bilevel knapsack problem with a here-and-now follower under interval uncertainty.
   (b) Determine the worst-case realization of the uncertain weights for the follower.
   (c) Use (b) to state the deterministic equivalent of the robust bilevel knapsack problem under interval uncertainty.

(ii) Robust model under data uncertainty with a here-and-now follower (budgeted uncertainty): We now extend the setting considered in (i) and suppose that at most two uncertain parameters can deviate from their deterministic values $(4, 3, 2)^\top$. This is the so-called Γ-robust model by Bertsimas and Sim (2003).

   (a) Define the budgeted uncertainty set in which at most two uncertain parameters can deviate from their deterministic values.
   (b) State the robust bilevel knapsack problem with a here-and-now follower under budgeted uncertainty.

(c) State the deterministic equivalent of the robust bilevel knapsack problem under budgeted uncertainty. To this end, use your definition of the uncertainty set from (a) and note that the worst-case realization is attained at one of the vertices of this set.

(iii) Robust model under data uncertainty with a wait-and-see follower:

(a) State the robust bilevel knapsack problem with a wait-and-see follower under interval uncertainty as in (i).
(b) State the robust bilevel knapsack problem with a wait-and-see follower under budgeted uncertainty as in (ii).

(iv) Stochastic model under data uncertainty with a wait-and-see follower: Suppose that the uncertain weights can take one of three possible values, i.e., $w \in \{w^1, w^2, w^3\}$ with

$$w^1 = (4, 3, 2)^\top, \quad w^2 = (2, 2, 1)^\top, \quad w^3 = (3, 2, 2)^\top.$$

Each scenario occurs with probability $1/3$.

(a) State the stochastic bilevel knapsack problem with a wait-and-see follower in which the leader hedges against data uncertainty using the expected value.
(b) State the deterministic equivalent of the problem in (a).

(v) Robust model under limited observability: Suppose now that all data of the bilevel knapsack problem is known but that the follower cannot perfectly observe the leader's decision $\bar{x}$. Instead, the follower observes x and hedges against all possible leader decisions $\bar{x} \in [x - 1, x + 1]$.

(a) State the robust model of the bilevel knapsack problem under limited observability.
(b) Determine the worst-case realization of the uncertain leader's decision.
(c) Use (b) to state the deterministic equivalent of the robust bilevel knapsack problem under limited observability.

15

Bilevel Problems with Multiple Leaders or Followers

So far, we studied bilevel optimization problems that model the situation in which two decision makers act in a hierarchical way. In game theory, optimization, or operations research, one also frequently needs to model non-cooperative situations in which two (or more) players act at the same time. The respective games are called simultaneous-move (or Nash) games. We now first define these games and the respective solution concept called Nash equilibrium. Afterward, we combine bilevel optimization with Nash games in the following way. The lower-level or the upper-level problem (or even both), which have been optimization problems so far, can also be replaced by a Nash game. We have briefly touched upon such situations before, e.g., in Example 1.4 about energy market modeling. In this example, we describe a situation in which a regulatory authority decides about the trading rules of an energy market at which producers and consumers trade. In this setting, we have one leader (the regulatory authority) and many followers (the energy producers and consumers). The regulatory authority moves first and its objective function usually depends on the outcome of the lower-level problem. This, however, is not an optimization problem of a single decision maker anymore, but corresponds to an equilibrium (the market outcome) of a game (the market trading process). Of course, the game itself depends on the rules set by the leader. We call the overall problem a single-leader multi-follower game. Other settings are possible as well, leading to multi-leader single-follower or even multi-leader multi-follower games.

In this chapter, we give a primer on these highly challenging models. If you are interested in other applications of these models, we recommend to read Aussel and Svensson (2020), where you also find many other references for papers dealing with industrial eco-parks, liberalized electricity markets, transmission and generation expansion planning for power networks, or demand-side management.

15.1 (Generalized) Nash Games and Nash Equilibria

Games and their equilibria are key concepts in game theory. Game theory dates back to the early works of Cournot and Bertrand in the 19th century and the first formalization of games appeared in von Neumann and Morgenstern (1947), rather short before the seminal publication by Nash (1950), which gives this section its name. Let us start with the very basic mathematical definition of a game.

Definition 15.1 A *(strategic) game* (in normal form) is given by

(i) a (finite) set $[N] := \{1, \dots, N\}$ of *players*,
(ii) a *strategy set* X_ν for every player $\nu \in [N]$,
(iii) and *cost functions* $\theta_\nu : X \to \mathbb{R}$ for every player $\nu \in [N]$. Here, $X := X_1 \times \cdots \times X_N$ is the Cartesian product of all strategy sets.

We abbreviate such a game as $\Gamma = (X_\nu, \theta_\nu)_{\nu \in [N]}$ and call Γ an *N-person game*.

The most common solution concept for such games is a so-called Nash equilibrium.

Definition 15.2 (Nash Equilibrium) Consider the game $\Gamma = (X_\nu, \theta_\nu)_{\nu \in [N]}$. A vector $(x^{*,\nu})_{\nu \in [N]}$ is called a *Nash equilibrium* of the game if $x^{*,\nu} \in X_\nu$ holds for all $\nu \in [N]$ and if

$$\theta_\nu(x^*) \le \theta_\nu(x^{*,1}, \dots, x^{*,\nu-1}, x^\nu, x^{*,\nu+1}, \dots, x^{*,N}) \tag{15.1}$$

holds for all $x^\nu \in X_\nu$ and all $\nu \in [N]$.

The notation used in the right-hand side of (15.1) is a bit lengthy. Thus, we introduce the following alternative and shorter one. Let $x = (x^1, \dots, x^N)^\top$ be a given vector made up of the block vectors x^ν for $\nu \in [N]$. To emphasize the νth component, we write

$$x = (x^\nu, x^{-\nu}),$$

where $x^{-\nu}$ denotes all vectors x^μ with $\mu \neq \nu$. Consequently, $(x^\nu, x^{*,-\nu})$ serves as an abbreviation for

$$\left(x^{*,1}, \dots, x^{*,\nu-1}, x^\nu, x^{*,\nu+1}, \dots, x^{*,N}\right).$$

Using this notation, a vector $x^* = (x^{*,\nu})_{\nu \in [N]}$ is called a Nash equilibrium if $x^{*,\nu} \in X_\nu$ holds for all $\nu \in [N]$ and if

$$\theta_\nu(x^*) \le \theta_\nu(x^\nu, x^{*,-\nu})$$

holds for all $x^\nu \in X_\nu$ and all $\nu \in [N]$. For each $\nu \in [N]$, this means that $x^{*,\nu}$ solves the optimization problem

$$\min_{x^\nu} \quad \theta_\nu(x^\nu, x^{*,-\nu}) \quad \text{s.t.} \quad x^\nu \in X_\nu. \tag{15.2}$$

Here, the objective function of player ν also depends on the strategies $x^{*,-\nu}$ of all other players.

Next, let us write down the definition of a Nash equilibrium in another way. To this end, let $S_\nu(x^{-\nu})$ be the set of optimal solutions to Problem (15.2). With this notation, we can see that $x^* = (x^{*,\nu})_{\nu=1}^N$ is a Nash equilibrium if and only if

$$x^{*,\nu} \in S_\nu(x^{*,-\nu})$$

holds for all $\nu \in [N]$. Thus, we have shown the following result.

Theorem 15.3 *Consider the game $\Gamma = (X_\nu, \theta_\nu)_{\nu \in [N]}$. A vector $x^* = (x^{*,\nu})_{\nu \in [N]}$ is a Nash equilibrium of Γ if and only if $x^{*,\nu} \in S_\nu(x^{*,-\nu})$ holds for all $\nu \in [N]$.*

Definition 15.4 (Best-Reply Function) The mapping $x^{-\nu} \mapsto S_\nu(x^{-\nu})$ is called the *best-reply function* of player ν. The corresponding function $x \mapsto \mathcal{S}(x)$ defined as

$$\mathcal{S}(x) := S_1(x^{-1}) \times \cdots \times S_N(x^{-N})$$

is called the *best-reply function* of the entire game.

With the last definition, we can also state that x^* is a Nash equilibrium if and only if $x^* \in \mathcal{S}(x^*)$, i.e., if and only if x^* is a fixed point of this set-valued map.

Up to now, we have considered games in which every player $\nu \in [N]$ solves an optimization problem of the form in (15.2). In this setting, the objective function of player ν depends on the decision x^ν, of course, but also on the decisions $x^{-\nu}$ of the other players. This is different for the feasible sets. Here, every player has a feasible set X_ν, which does not depend on the decisions $x^{-\nu}$ of the other players. For many real-world problems, however, one needs to model situations in which the feasible set of player ν also depends on the decisions $x^{-\nu}$. Taking such dependencies into account leads to so-called *generalized Nash equilibrium problems*, or GNEPs, for short.

A GNEP is a game in which every player $\nu \in [N]$ solves an optimization problem of the form

$$\min_{x^\nu} \quad \theta_\nu(x^\nu, x^{-\nu}) \quad \text{s.t.} \quad x^\nu \in X_\nu(x^{-\nu})$$

with cost functions θ_ν and feasible strategy sets $X_\nu(x^{-\nu})$. The novel aspect here is the dependence of the feasible set on the decisions of the other players.

This is a usual requirement in situations in which multiple players share, e.g., a common budget or common resources. In general, the feasible sets are of the form

$$X_\nu(x^{-\nu}) = \{x^\nu \in \mathbb{R}^{n_\nu} : h^\nu(x^\nu) \geq 0,\ g^\nu(x^\nu, x^{-\nu}) \geq 0\}.$$

Thus, there may be constraints h^ν that only depend on the own decisions as well as constraints g^ν that may also depend on the decisions of the other players. The definition of a (generalized) Nash equilibrium now follows the same principle as before.

Definition 15.5 (Generalized Nash Equilibria) Consider a given GNEP as introduced above. A vector $x^* = (x^{*,1}, \dots, x^{*,N}) \in X$ is called a *Nash equilibrium* of the GNEP if

$$\theta_\nu(x^{*,\nu}, x^{*,-\nu}) \leq \theta_\nu(x^\nu, x^{*,-\nu})$$

holds for all $x^\nu \in X_\nu(x^{*,-\nu})$.

For a more detailed discussion of GNEPs, we refer to Facchinei and Kanzow (2010).

15.2 Problem Statements

We now present three different types of hierarchical interactions of multiple agents and briefly discuss the main challenges and obstacles for studying and solving these problems. For a more detailed discussion, we refer to the chapter by Aussel and Svensson (2020) in the book by Dempe and Zemkoho (2020). Further theoretical studies can be found in Aussel et al. (2021) and Aussel and Svensson (2018).

15.2.1 Single-Leader Multi-Follower Games

The first class of models are so-called single-leader multi-follower (SLMF) games. In this case, the problem of the (single) leader looks almost as in classic bilevel optimization, namely

$$\begin{aligned} \min_{x \in X, y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & y = (y^\nu)_{\nu \in [N]} \in \mathcal{E}(x). \end{aligned}$$

Here, two things are different compared to classic bilevel optimization. First, we have N followers instead of one and the upper-level problem now depends on the collection of strategies $y = (y^\nu)_{\nu \in [N]}$ of all players that need to be a Nash equilibrium of the lower level. This means that $\mathcal{E}(x)$ is the set of all Nash equilibria of the lower-level game. In addition to Definitions 15.1 and 15.2, we now consider an x-parameterized game, i.e., every player $\nu \in [N]$ solves a problem of the form

$$\min_{y^\nu} \quad f_\nu(x, y^\nu, y^{-\nu}) \quad \text{s.t.} \quad y^\nu \in Y_\nu(x). \tag{15.3}$$

The difference to Problem (15.2) is that both the objective functions and the feasible sets of the lower-level players can now depend on the upper-level decision x as we are used to in bilevel optimization. In particular, if we replace the lower-level players' constraints $y^\nu \in Y_\nu(x)$ with $y^\nu \in Y_\nu(x, y^{-\nu})$, the lower-lever game is a GNEP.

For SLMF games, we face similar challenges as in bilevel optimization. First, we have to be careful regarding the situation in which the lower-level equilibria are not unique. The same concepts as in bilevel optimization, i.e., optimistic or pessimistic settings, can be considered here as well. Second, the set $\mathcal{E}(x)$ is usually hard to describe and not known in closed form. The existing theoretical results for this class of problems mainly boil down to existence results; see, e.g., Section 3.3.1 in Aussel and Svensson (2020) or Aussel and Svensson (2018).

For multiple followers, the special case in which all lower-level players' optimization problems only depend on the upper-level decision x but not on the decisions of the other followers can also occur. In this case, (15.3) turns into

$$\min_{y^\nu} \quad f_\nu(x, y^\nu) \quad \text{s.t.} \quad y^\nu \in Y_\nu(x). \tag{15.4}$$

These games are called single-leader multi-disjoint-follower (SLMDF) games. It is rather easy to see that these games are equivalent to classic bilevel optimization problems. The reason is that the set $\mathcal{E}(x)$ of x-parameterized Nash equilibria, in case of Problem (15.4) for every lower-level player $\nu \in [N]$, is equivalent to the set $\mathcal{S}(x)$ of optimal solutions to the "aggregated" problem

$$\min_{y=(y^\nu)_{\nu\in[N]}} \quad \sum_{\nu\in[N]} f_\nu(x, y^\nu) \quad \text{s.t.} \quad y^\nu \in Y_\nu(x) \quad \text{for all } \nu \in [N].$$

The reason is that the followers' problems do not depend on each other. Hence, all theoretical results and solution techniques of the previous chapters can be applied. For more details on this sub-class of models, see Calvete and Galé (2007).

Example 15.6 (Toll Setting—Revisited) The toll-setting problem in Example 1.17 is a single-leader multi-disjoint-follower game because all followers are independent of each other. However, in traffic models, we often encounter a joint capacity constraint for every arc of the network. In the notation of Example 1.17, for user $i \in \mathcal{I}$ of the network, such a constraint is then given by

$$x_{i,a} \leq u_a - \sum_{j \in \mathcal{I} \setminus \{i\}} x_{j,a} \quad \text{for all } a \in \mathcal{A},$$

where $u_a > 0$ is the capacity of an arc a. If every follower faces such a constraint, the resulting problem is a single-leader multi-follower game in which all follower problems form a GNEP. △

15.2.2 Multi-Leader Single-Follower Games

We now reverse the situation and consider multiple leaders who play a (generalized) Nash game in the upper level that depends on a vector y, which is an optimal solution to a lower-level optimization problem of a single follower. The lower level itself depends on the Nash equilibrium of the upper level.

More formally, we search for a Nash equilibrium among N players, where player $\nu \in [N]$ solves the optimization problem

$$\begin{aligned} \min_{x^\nu} \quad & F_\nu(x^\nu, x^{-\nu}, y) \\ \text{s.t.} \quad & x^\nu \in X_\nu(y), \\ & y \in \mathcal{S}(x) \end{aligned}$$

with $x = (x^\nu)_{\nu \in [N]}$ and $y \in \mathcal{S}(x)$ being a shared constraint, i.e., every leader obtains the same follower's response. Here, both the objective function and the constraints of the νth player can depend on the lower-level optimal solution y. Note that in the given form, the respective game among the leaders is a generalized game because the lower-level solution set is parameterized by the complete strategy vector x. Whether the first constraint $x^\nu \in X_\nu(y)$ also depends on $x^{-\nu}$, i.e., if we consider the constraint $x^\nu \in X_\nu(x^{-\nu}, y)$, depends on the application.

Multi-leader single-follower (MLSF) games have to be treated with great care. The reason is the following. As usual, the lower-level solution set $\mathcal{S}(x)$ does not need to be a singleton for all potential x. However, the notions of optimistic and pessimistic solutions do not make sense anymore as the same best reply y may be optimistic for one leader but pessimistic for another one. Hence, the above defined problem is very likely to be ill-posed unless there is a uniqueness result for the lower-level problem at hand. Nevertheless, this class of models is very important for many applications. For instance, it perfectly fits for describing day-ahead electricity markets; see, e.g., Aussel et al. (2017a,b).

15.2.3 Multi-Leader Multi-Follower Games

Last but not least, we can go to the extreme and consider a hierarchical setting with multiple leaders and multiple followers who each play a (generalized) Nash game. More formally, we search for a Nash equilibrium among N players in which player $\nu \in [N]$ solves the optimization problem

$$\begin{aligned} \min_{x^\nu} \quad & F_\nu(x^\nu, x^{-\nu}, y) \\ \text{s.t.} \quad & x^\nu \in X_\nu(y), \\ & y \in \mathcal{E}(x), \end{aligned}$$

where $x = (x^\nu)_{\nu \in [N]}$ and $y = (y^\mu)_{\mu \in [M]}$ are the overall strategies of all N leaders and all M followers. Hence, $\mathcal{E}(x)$ is the set of Nash equilibria of the followers' game, in which each follower $\mu \in [M]$ solves the problem

$$\min_{y^\mu} \quad f_\mu(x, y^\mu, y^{-\mu}) \quad \text{s.t.} \quad y^\mu \in Y_\mu(x).$$

Again, replacing the feasible sets $Y_\mu(x)$ with $Y_\mu(x, y^{-\mu})$ turns the lower-level game into a generalized Nash game.

As it is the case for MLSF games, also multi-leader multi-follower (MLMF) games are likely to be ill-posed in case of multiplicities in the lower-level equilibrium set.

15.3 A Single-Level Reformulation of Linear SLMF Games

Due to the ill-posedness issues of multi-leader games, we focus on how to solve SLMF games in the remainder of this chapter. In particular, we present and explain a powerful technique to obtain a single-level reformulation of the SLMF game, which is based on the KKT conditions of the followers' problems as we have already studied it in Chapter 3 for bilevel problems. To this end, we need to make the assumption that the players' problems in the lower-level are convex and satisfy a constraint qualification. As discussed in Chapter 5, we again use special ordered sets of type 1 (SOS1), which date back to Beale and Tomlin (1970) and which have been used for the first time in bilevel optimization in Fortuny-Amat and McCarl (1981). Hence, the overall idea is to use the KKT conditions of the lower-level players' problems and reformulate the nonlinear and nonconvex KKT complementarity conditions by using SOS1-type constraints. The advantages of this SOS1-based approach are the following:

(i) It is easy to implement. We present `Python` code that shows how to implement the resulting single-level reformulation of a given SLMF game and then solve it with `Gurobi`.

(ii) The SOS1 functionality is widely available. Not only `Gurobi`, but also other solvers for mixed-integer optimization problems (see Section 3.4) support this modeling technique.

(iii) There is no need for computing big-M values or penalization parameters.

The remainder of this chapter is mainly based on Aussel et al. (2025) and is structured as follows. In Section 15.3.1, we formally present the problem statement and derive a first single-level reformulation based on the KKT conditions of the lower-level players' problems. The SOS1-based reformulation of the single-level KKT reformulation is briefly discussed in Section 15.3.2. This technique is then applied in Section 15.3.3 to two academic examples. Here, we also present `Python` code that shows how easy the implementation is.

15.3.1 SLMF Games with Convex-Linear Follower Problems

Our aim is to derive a reformulation scheme based on the SOS1 approach to solve SLMF games in which the followers have convex objective functions and polyhedral feasible sets. This setup allows to use first-order optimality conditions to ensure global optimality and does not require to discuss any technicalities regarding constraint qualifications. Possible generalizations are discussed in Remark 15.10 below. In their optimistic version, the problems under consideration are thus given by

$$\begin{aligned} \min_{x \in X, y} \quad & F(x, y) \\ \text{s.t.} \quad & G(x, y) \geq 0, \\ & y = (y^\nu)_{\nu \in [N]} \in \mathcal{E}(x), \end{aligned} \tag{15.5}$$

where $x \in X \subseteq \mathbb{R}^{n_0}$ is the vector of the leader's decisions and $y^\nu \in \mathbb{R}^{n_\nu}$ is the variable vector of follower $\nu \in [N]$. The vector $y = (y^\nu)_{\nu \in [N]} \in \mathbb{R}^{n_f}$ collects all decisions of all followers, i.e., $n_f = \sum_{\nu \in [N]} n_\nu$. Hence, we have $F : \mathbb{R}^{n_0} \times \mathbb{R}^{n_f} \to \mathbb{R}$ and $G : \mathbb{R}^{n_0} \times \mathbb{R}^{n_f} \to \mathbb{R}^{m_0}$, where m_0 is the number of upper-level constraints. Furthermore, $\mathcal{E}(x)$ stands for the set of generalized Nash equilibria of the non-cooperative game among the N followers, where the optimization problem of the νth player is given by

$$\min_{y^\nu} \quad f_\nu(x, y^\nu, y^{-\nu}) \tag{15.6a}$$

$$\text{s.t.} \quad D^{\nu,0}x + D^\nu y^\nu + \sum_{\mu \neq \nu} D^{\nu,\mu} y^\mu \geq e^\nu, \tag{15.6b}$$

$$E^0 x + \sum_{\mu \in [N]} E^\mu y^\mu \geq g \tag{15.6c}$$

with $D^\nu \in \mathbb{R}^{m_\nu \times n_\nu}$, $D^{\nu,0} \in \mathbb{R}^{m_\nu \times n_0}$, $D^{\nu,\mu} \in \mathbb{R}^{m_\nu \times n_\mu}$ for all $\mu \neq \nu$, and $e^\nu \in \mathbb{R}^{m_\nu}$. Moreover, we have $E^0 \in \mathbb{R}^{m \times n_0}$, $E^\nu \in \mathbb{R}^{m \times n_\nu}$ for all $\nu \in [N]$, and $g \in \mathbb{R}^m$. Here, (15.6b) is the private constraint of player ν, which may depend on the leader's decision x and the decisions of all other lower-level players $\mu \in [N] \setminus \{\nu\}$, and (15.6c) is the shared constraint of the GNEP at the lower level. This means that the shared constraint is the same in all players' problems, whereas the private ones may also depend on the decisions of the other players but is different for every player. For instance, private constraints may contain technological constraints of different producers, whereas the shared constraint may be a market-clearing constraint stating that the overall power production must meet the demand—a constraint that is the same for every player. For later reference, we denote the feasible set of the νth lower-level player, defined by (15.6b) and (15.6c), as $Y_\nu(x, y^{-\nu})$. In this spirit, we also define $Y_{-\nu} = \prod_{\mu \in [N] \setminus \{\nu\}} Y_\mu(x, y^{-\mu})$. The shared constraint set is then given by

$$\Omega := \{(x, y) \in \mathbb{R}^{n_0} \times \mathbb{R}^{n_f} : G(x, y) \geq 0,\ y^\nu \in Y_\nu(x, y^{-\nu}) \text{ for all } \nu \in [N]\},$$

and its projection onto the space of the leader variables is denoted as

$$\Omega_x := \{x \in \mathbb{R}^{n_0} : \exists y \in \mathbb{R}^{n_f} \text{ with } (x, y) \in \Omega\}.$$

The class of SLMF games we can tackle is quite large because the only restrictions are the following ones.

Assumption 15.7 For the remainder of this chapter, we assume the following.

(i) The functions F and G are continuous on $\mathbb{R}^{n_0} \times \mathbb{R}^{n_f}$.
(ii) For any $x \in \Omega_x$, any $\nu \in [N]$, and any $y^{-\nu} \in Y_{-\nu}$, the function $f(x, \cdot, y^{-\nu})$ is convex and continuously differentiable on $\mathbb{R}^{n_\nu}$.
(iii) The feasible set of each follower is polyhedral, i.e., it is defined by affine functions.

For any follower $\nu \in [N]$, the KKT conditions associated with the player's problem (15.6) are given by

$$\nabla_{y_\nu} f_\nu(x, y^\nu, y^{-\nu}) - (D^\nu)^\top \lambda^\nu - (E^\nu)^\top \delta^\nu = 0, \tag{15.7a}$$

$$D^{\nu,0} x + D^\nu y^\nu + \sum_{\mu \neq \nu} D^{\nu,\mu} y^\mu - e^\nu \geq 0, \tag{15.7b}$$

$$E^0 x + \sum_{\mu \in [N]} E^\mu y^\mu - g \geq 0, \tag{15.7c}$$

$$\lambda^\nu, \delta^\nu \geq 0, \tag{15.7d}$$

$$(\lambda^\nu)^\top \left(D^{\nu,0} x + D^\nu y^\nu + \sum_{\mu \neq \nu} D^{\nu,\mu} y^\mu - e^\nu \right) = 0, \tag{15.7e}$$

$$(\delta^\nu)^\top \left(E^0 x + \sum_{\mu \in [N]} E^\mu y^\mu - g \right) = 0, \tag{15.7f}$$

where λ^ν and δ^ν are the respective Lagrange multipliers of the Constraints (15.6b) and (15.6c).

The KKT conditions of the lower-level problems are both necessary and sufficient for all players. In particular, we do not need any further constraint qualifications for the lower-level problems because the constraint sets are polyhedral and, thus, the Abadie CQ (see Definition A.17) is satisfied. Moreover, we assume w.l.o.g. that all linear problems are stated so that the LICQ (see Definition A.20) holds. Consequently, the Lagrange multipliers of all KKT points are uniquely determined; see Theorem A.21. This is particularly important in applications in which the shared constraint of the GNEP is a market-clearing constraint and in which the KKT complementarity constraint in (15.7f) defines market-clearing prices δ^ν (Samuelson 1952).

Remark 15.8 In many economic models, one aims for a situation in which every player receives the same price signal, i.e., has the same dual variable δ^ν in (15.7f). This leads to so-called variational equilibria (Facchinei and Kanzow 2010; Rosen 1965) of the GNEP in the lower-level problem for which $\delta^\nu = \delta \geq 0$ holds for all $\nu \in [N]$. In this setting, (15.7a) needs to be replaced with

$$\nabla_{y_\nu} f_\nu(x, y^\nu, y^{-\nu}) - (D^\nu)^\top \lambda^\nu - (E^\nu)^\top \delta = 0$$

and (15.7f) needs to be replaced with

$$\delta^\top \left(E^0 x + \sum_{\mu \in [N]} E^\mu y^\mu - g \right) = 0.$$

From now on, we mainly focus on such variational equilibria in this chapter.

Replacing the GNEP in the lower level by the concatenation of all KKT conditions of the lower-level players, we obtain the single-level reformulation

$$\min_{x,y,\lambda,\delta} \quad F(x, y) \tag{15.8a}$$

$$\text{s.t.} \quad G(x, y) \geq 0, \tag{15.8b}$$

$$\nabla_{y_\nu} f_\nu(x, y^\nu, y^{-\nu}) - (D^\nu)^\top \lambda^\nu - (E^\nu)^\top \delta = 0, \quad \nu \in [N], \tag{15.8c}$$

$$D^{\nu,0} x + D^\nu y^\nu + \sum_{\mu \neq \nu} D^{\nu,\mu} y^\mu - e^\nu \geq 0, \quad \nu \in [N], \tag{15.8d}$$

$$E^0 x + \sum_{\mu \in [N]} E^\mu y^\mu - g \geq 0, \tag{15.8e}$$

$$\lambda^\nu, \delta \geq 0, \quad \nu \in [N], \tag{15.8f}$$

$$(\lambda^\nu)^\top \left(D^{\nu,0} x + D^\nu y^\nu + \sum_{\mu \neq \nu} D^{\nu,\mu} y^\mu - e^\nu \right) = 0, \quad \nu \in [N], \tag{15.8g}$$

$$\delta^\top \left(E^0 x - \sum_{\mu \in [N]} E^\mu y^\mu - g \right) = 0. \tag{15.8h}$$

As usual, we use the abbreviations $y = (y^\nu)_{\nu \in [N]}$ and $\lambda = (\lambda^\nu)_{\nu \in [N]}$. Note that this single-level reformulation leads to an optimistic solution to the single-leader multi-follower problem. By construction, we get the following theorem. A formal proof can be deduced from Theorem 3.3.8 in Aussel and Svensson (2020).

Theorem 15.9 *Suppose that Assumption 15.7 holds. Let (x^*, y^*) be a globally optimal solution to the single-leader multi-follower game (15.5). Then, there exist λ^* and δ^* so that $(x^*, y^*, \lambda^*, \delta^*)$ is a globally optimal solution to the single-level reformulation (15.8).*

On the other hand, if $(x^*, y^*, \lambda^*, \delta^*)$ is a globally optimal solution to the single-level reformulation (15.8), then (x^*, y^*) is a globally optimal solution to the single-leader multi-follower game (15.5).

Remark 15.10 We finally discuss some potential generalizations of the above setting.

(i) The setup and the main result can be generalized to convex instead of polyhedral feasible sets if Slater's constraint qualification is satisfied because the corresponding KKT conditions of the followers are still necessary and sufficient. However, in the light of the results by Aussel and Svensson (2019b) and Dempe and Dutta (2012), one needs to be careful with the respective Lagrange multipliers to obtain the correctness of Theorem 15.9.

(ii) The properties of F and G do not influence the reformulation techniques for the lower-level GNEP. Hence, they can also be nonlinear and even be nonconvex. Moreover, the upper-level variables x may be mixed-integer.

(iii) The constraints of the lower-level problems stay linear if one allows for multilinear terms. Hence, the classic KKT reformulation can still be applied. However, the resulting single-level reformulation inherits these multilinear terms. Because one optimizes over all follower variables in the single-level reformulation, this then leads to additional nonconvex nonlinearities in the single-level reformulation compared to those potentially introduced by F or G. The same applies to the lower-level objective functions, where we, however, loose one degree of the multilinear polynomial by taking the respective gradient in the KKT conditions.

15.3.2 SOS1-Based Reformulation

The main burden in Problem (15.8) are the KKT complementarity constraints (15.8g) and (15.8h). These nonlinear and nonconvex constraints can be modeled using SOS1 conditions and the resulting problems can, thus, be solved to global optimality using modern mixed-integer solvers without choosing any big-Ms; see, e.g., Kleinert and Schmidt (2023) for the classic case of bilevel optimization, i.e., for single-leader single-follower games.

To this end, we introduce auxiliary variables $s^{\nu}_{i_\nu} \geq 0$ for all $\nu \in [N]$ and all $i_\nu \in [m_\nu]$ and $t_j \geq 0$ for all $j \in [m]$, as well as the following SOS1 conditions

$$\mathrm{SOS1}(\lambda^{\nu}_{i_\nu}, s^{\nu}_{i_\nu}) \quad \text{with} \quad s^{\nu}_{i_\nu} = \left(D^{\nu,0}x + D^{\nu}y^{\nu} + \sum_{\mu \neq \nu} D^{\nu,\mu}y^{\mu} - e^{\nu}\right)_{i_\nu}$$

for all $\nu \in [N]$ and $i_\nu \in [m_\nu]$ as well as

$$\mathrm{SOS1}(\delta_j, t_j) \quad \text{and} \quad t_j = \left(E^{0}x - \sum_{\mu \in [N]} E^{\nu}y^{\nu} - g\right)_j$$

for all $j \in [m]$. Recall that an SOS1 condition defines a set of variables for which at most one variable in the set may take a value other than zero.

Remark 15.11 If we have provably correct and not too large big-Ms for the dual variables $\lambda^{\nu}_{i_\nu}$ and δ_j as well as for the corresponding primal expressions, we can also use the classic big-M reformulation for these components; see Section 3.4 for a detailed discussion.

15.3.3 Academic Examples

In this section, we use two examples to show how simple the implementation of the SOS1 approach is for computing solutions to single-leader multi-follower games. In the spirit of a tutorial, for the first example, the single-level reformulation of the single-leader multi-follower game is given, then a `Python` code is discussed to compute a solution to the single-level reformulation, and finally the optimization results are presented. For the second example, we also present the single-level reformulation but leave the implementation of it as an exercise.

Example 15.12 Let us start with a very simple and completely linear example of a single-leader multi-follower game. Three agents are considered here so that we have one leader and two followers. The leader's problem is given by

$$\begin{aligned} \min_{x,y} \quad & 2x + y_1 + y_2 - y_3 \\ \text{s.t.} \quad & x \geq 1, \\ & y = (y_1, y_2, y_3)^\top \in \mathcal{E}(x), \end{aligned}$$

where the GNEP in the lower level consists of the following two players. The first follower solves the 2-dimensional optimization problem

$$\begin{aligned} \min_{y_1,y_2} \quad & x - 2y_1 - y_2 \\ \text{s.t.} \quad & y_1 \geq y_2 + y_3, \\ & y_1 \leq x, \end{aligned}$$

whereas the second follower solves the 1-dimensional problem

$$\begin{aligned} \min_{y_3} \quad & x + y_1 + y_2 + y_3 \\ \text{s.t.} \quad & y_3 \geq x. \end{aligned}$$

This SLMF admits only one solution, namely $(\bar{x}, \bar{y}_1, \bar{y}_2, \bar{y}_3) = (1, 1, 0, 1)$. The corresponding KKT conditions of the first follower read

$$\begin{aligned} -2 - \lambda_1^1 + \lambda_2^1 &= 0, \\ -1 + \lambda_1^1 &= 0, \\ y_1 - y_2 - y_3 &\geq 0, \\ x - y_1 &\geq 0, \\ \lambda_1^1(y_1 - y_2 - y_3) &= 0, \\ \lambda_2^1(x - y_1) &= 0, \\ \lambda_1^1, \lambda_2^1 &\geq 0, \end{aligned}$$

and the KKT conditions of the second follower are given by

$$\begin{aligned} 1 - \lambda_1^2 &= 0, \\ y_3 - x &\geq 0, \\ \lambda_1^2 (y_3 - x) &= 0, \\ \lambda_1^2 &\geq 0. \end{aligned}$$

Because of Theorem 15.9, the single-level reformulation is equivalent to the original SLMF game and reads

$$\begin{aligned} \min_{x,y,\lambda} \quad & 2x + y_1 + y_2 - y_3 \\ \text{s.t.} \quad & x \geq 1, \\ & -2 - \lambda_1^1 + \lambda_2^1 = 0, \\ & -1 + \lambda_1^1 = 0, \\ & y_1 - y_2 - y_3 \geq 0, \\ & x - y_1 \geq 0, \\ & \lambda_1^1 (y_1 - y_2 - y_3) = 0, \\ & \lambda_2^1 (x - y_1) = 0, \\ & \lambda_1^1, \lambda_2^1 \geq 0, \\ & 1 - \lambda_1^2 = 0, \\ & y_3 - x \geq 0, \\ & \lambda_1^2 (y_3 - x) = 0, \\ & \lambda_1^2 \geq 0. \end{aligned}$$

Finally, if we use the non-negative slack variables

$$s_1^1 = y_1 - y_2 - y_3, \quad s_2^1 = x - y_1, \quad s_1^2 = y_3 - x,$$

we obtain the SOS1-based single-level reformulation

$$\begin{aligned} \min_{x,y,\lambda,s} \quad & 2x + y_1 + y_2 - y_3 \\ \text{s.t.} \quad & x \geq 1, \\ & -2 - \lambda_1^1 + \lambda_2^1 = 0, \\ & -1 + \lambda_1^1 = 0, \\ & y_1 - y_2 - y_3 \geq 0, \\ & x - y_1 \geq 0, \\ & \text{SOS1}(\lambda_1^1, s_1^1), \end{aligned}$$

$$
\begin{aligned}
&\mathrm{SOS1}(\lambda_2^1, s_2^1),\\
&s_1^1 = y_1 - y_2 - y_3,\ s_1^1 \ge 0,\\
&s_2^1 = x - y_1,\ s_2^1 \ge 0,\\
&\lambda_1^1, \lambda_2^1 \ge 0,\\
&1 - \lambda_1^2 = 0,\\
&y_3 - x \ge 0,\\
&\mathrm{SOS1}(\lambda_1^2, s_1^2),\\
&s_1^2 = y_3 - x,\ s_1^2 \ge 0,\\
&\lambda_1^2 \ge 0
\end{aligned}
$$

with $y = (y_1, y_2, y_3)$, $\lambda = (\lambda_1^1, \lambda_2^1, \lambda_1^2)$, and $s = (s_1^1, s_2^1, s_1^2)$. Using the Python interface of Gurobi, this model can be implemented as follows.

```python
# Import the Gurobi interface
from gurobipy import *

# Create an empty model
model = Model("slmf-example")

# Build all variables
x = model.addVar(name="x", lb=1)
y1 = model.addVar(name="y1", lb=-GRB.INFINITY)
y2 = model.addVar(name="y2", lb=-GRB.INFINITY)
y3 = model.addVar(name="y3", lb=-GRB.INFINITY)
lambda11 = model.addVar(name="lambda11")
lambda12 = model.addVar(name="lambda12")
lambda21 = model.addVar(name="lambda21")
s11 = model.addVar(name="s11")
s12 = model.addVar(name="s12")
s21 = model.addVar(name="s21")

# Build the upper-level objective function
model.setObjective(2*x + y1 + y2 - y3, GRB.MINIMIZE)

# Add lower-level dual feasibility constraints
model.addConstr(-2 - lambda11 + lambda12 == 0)
model.addConstr(-1 + lambda11 == 0)
model.addConstr(1 - lambda21 == 0)

# Add lower-level primal feasibility constraints
model.addConstr(y1 - y2 - y3 >= 0)
model.addConstr(x - y1 >= 0)
model.addConstr(y3 - x >= 0)

# Add slack variable defining constraints
model.addConstr(s11 == y1 - y2 - y3)
model.addConstr(s12 == x - y1)
```

```
model.addConstr(s21 == y3 - x)

# Add SOS1 conditions
model.addSOS(GRB.SOS_TYPE1, [lambda11, s11])
model.addSOS(GRB.SOS_TYPE1, [lambda12, s12])
model.addSOS(GRB.SOS_TYPE1, [lambda21, s21])

# Solve it
model.optimize()
```

After solving the model, we can get the solution values via

```
# Print the solution
print("       x = " + str(x.X))
print("      y1 = " + str(y1.X))
print("      y2 = " + str(y2.X))
print("      y3 = " + str(y3.X))
print("lambda11 = " + str(lambda11.X))
print("lambda12 = " + str(lambda12.X))
print("lambda21 = " + str(lambda21.X))
print("     s11 = " + str(s11.X))
print("     s12 = " + str(s12.X))
print("     s21 = " + str(s21.X))
```

and obtain

```
       x = 1.0
      y1 = 1.0
      y2 = 0.0
      y3 = 1.0
lambda11 = 1.0
lambda12 = 3.0
lambda21 = 1.0
     s11 = 0.0
     s12 = 0.0
     s21 = 0.0
```

△

Example 15.13 The second academic example also involves one leader and two followers but the objective function of one of the followers is now nonlinear in the variables of both followers. Here, the leader's problem is given by

$$\begin{aligned}\min_{x,y} \quad & -2x + y_1 + 3y_2 \\ \text{s.t.} \quad & x \geq -1, \\ & x \leq 1, \\ & y = (y_1, y_2)^\top \in \mathcal{E}(x),\end{aligned}$$

where the GNEP in the lower level consists of two players, each of them solving a 1-dimensional problem. The first follower solves

$$\min_{y_1} \quad y_1 + 2y_2 + 10 \quad \text{s.t.} \quad y_1 + y_2 \geq x,$$

whereas the second follower solves the problem

$$\min_{y_2} \quad y_1 y_2 \quad \text{s.t.} \quad y_2 \geq 0.$$

The best-reply functions of the followers are given by

$$S_1(x, y_2) = \{x - y_2\}$$

and

$$S_2(x, y_1) = S_2(y_1) = \begin{cases} \mathbb{R}_{\geq 0}, & \text{if } y_1 = 0, \\ \{0\}, & \text{if } y_1 > 0, \\ \emptyset, & \text{otherwise.} \end{cases}$$

Note that the optimization problem of the second follower is unbounded for $y_1 < 0$, i.e., the solution set is empty in this case. We have that $(y_1, y_2) \in \mathcal{E}(x)$ is equivalent to $y_1 \in S_1(x, y_2)$ and $y_2 \in S_2(y_1)$. Hence, by only considering optimistic replies of the followers, we obtain

$$\mathcal{E}(x) = \begin{cases} \{(x, 0), (0, x)\}, & \text{if } x \in (0, 1], \\ \{(0, 0)\}, & \text{if } x = 0, \end{cases}$$

where we use that $x \in [-1, 0)$ cannot be optimal for the leader. Thus, the SLMF game in Example 15.13 admits only one (optimistic) solution, which is given by $(x, y_1, y_2) = (1, 1, 0)$.

To describe the application of the SOS1 approach to this example, let us first present the KKT conditions of both followers. The first follower's optimality conditions read

$$\begin{aligned} 1 - \lambda_1^1 &= 0, \\ y_1 + y_2 - x &\geq 0, \\ \lambda_1^1 &\geq 0, \\ \lambda_1^1 (y_1 + y_2 - x) &= 0, \end{aligned}$$

whereas those for the second follower are given by

$$\begin{aligned} y_1 - \lambda_1^2 &= 0, \\ y_2 &\geq 0, \\ \lambda_1^2 &\geq 0, \\ \lambda_1^2 y_2 &= 0. \end{aligned}$$

For this example, we only need one auxiliary slack variable for the first follower, i.e.,

$$s_1^1 = y_1 + y_2 - x, \quad s_1^1 \geq 0.$$

With this, we obtain the single-level reformulation

$$\begin{aligned}
\min_{x,y,\lambda,s} \quad & -2x + y_1 + 3y_2 \\
\text{s.t.} \quad & x \geq -1, \\
& x \leq 1, \\
& 1 - \lambda_1^1 = 0, \\
& y_1 + y_2 - x \geq 0, \\
& \lambda_1^1 \geq 0, \\
& s_1^1 = y_1 + y_2 - x, \ s_1^1 \geq 0, \\
& \mathrm{SOS1}(\lambda_1^1, s_1^1), \\
& y_1 - \lambda_1^2 = 0, \\
& y_2 \geq 0, \\
& \lambda_1^2 \geq 0, \\
& \mathrm{SOS1}(\lambda_1^2, y_2)
\end{aligned}$$

with $y = (y_1, y_2)^\top$, $\lambda = (\lambda_1^1, \lambda_1^2)^\top$, and $s = s_1^1$. △

Exercise 15.14 Implement the last single-level reformulation and solve it.

16
Mind the Gaps

This is the end of our journey through the realm of bilevel optimization. We hope that you are now as fascinated by the field as we are and that you have gained a well-rounded overview—both w.r.t. breadth and depth—of the field.

Of course, it is not possible to cover all aspects of such a wide field as bilevel optimization in a single book. Thus, let us close this book with some brief and exemplary discussions of what other topics in bilevel optimization are out there.

(i) In this book, we mainly focused on linear and mixed-integer linear bilevel problems. However, there is a huge body of research that also focuses on mixed-integer nonlinear and, in particular, nonlinear and purely continuous bilevel problems. For purely continuous nonlinear bilevel problems, the required mathematical techniques are much deeper located in the field of analysis compared to what we use in this book. For some primers on this literature, take a look at the references in (viii) and (ix) below. Regarding mixed-integer nonlinear bilevel problems, we refer to Section 5.3 in Kleinert et al. (2021a) and the references therein.

(ii) We did not discuss any real-world applications in detail but, obviously, there are plenty of them. Examples include energy markets, see the references in (x) below, pricing and revenue management (Bialas and Karwan 1984; Brotcorne et al. 2011, 2001, 2008; Labbé et al. 1998; Labbé and Violin 2013), critical infrastructure defense (Alguacil et al. 2014; Borrero et al. 2019; Caprara et al. 2016; DeNegre 2011; Fioretto et al. 2019; Scaparra and Church 2008; Wood 2011), or applications including sustainability aspects (Caselli et al. 2026).

(iii) Most likely the largest field of applications of bilevel optimization today is machine learning. Over the last years, a vast amount of literature has developed that is mainly devoted to computing optimal hyperparameters of machine learning models such as neural networks. In particular, problem instances in this area are much larger than what can usually be tackled using the techniques presented in this book. However, the bilevel optimization literature for machine learning problems studies methods for computing local optimizers of the bilevel problem, whereas our focus is on global ones. Primers on this literature include Franceschi et al. (2018) and Frecon et al. (2018).

(iv) A traditional field of applications are so-called Stackelberg security games, which are also bilevel optimization problems but with a very specific structure that can be used to both classify the hardness of these problems as well as to derive effective solution methods. You may dive into this literature starting with Conitzer and Sandholm (2006) and Tambe (2011), as well as the references in Section 4.2 of the survey by Kleinert et al. (2021a).

(v) Many researchers approach bilevel optimization by using techniques from convex or variational analysis—in particular when it comes to single-leader multi-follower games; see, e.g., Aussel and Svensson (2018), Salas (2025), and Svensson (2024).

(vi) In this book, we always discussed problems defined in finite-dimensional real vector spaces. Besides this, bilevel problems can also be studied in infinite-dimensional spaces, which are usually function spaces as in Dempe et al. (2022) and Mehlitz and Wachsmuth (2016, 2020).

(vii) As there are multi-objective problems in single-level optimization, there are multi-objective problems in bilevel optimization as well; see, e.g., Eichfelder (2010, 2020) and the references therein for a starting point.

(viii) Although bilevel optimization problems are known to be nonsmooth and to violate standard constraint qualifications, this does not mean that one cannot derive optimality conditions for these problems. This is the topic of the papers by, e.g., Dempe and Zemkoho (2011, 2013).

(ix) Another important topic of bilevel optimization is the study of the lower-level optimal-value function as well as its solution-set mapping. Here, one is often interested in sensitivity and stability aspects, which directly leads to the field of parametric optimization. For a primer, see the lecture notes by Still (2018), the book by Bonnans and Shapiro (2000), or Chapter 4 in the seminal book by Dempe (2002), which also contains many references for further reading.

(x) Last but not least, there are problems with more than two levels. Many examples can be found in the field of energy markets (Ambrosius et al. 2020; Grimm et al. 2019a, 2016, 2019b; Kleinert and Schmidt 2019; Schewe et al. 2022). Moreover, so-called defender-attacker-defender or fortification problems (Leitner et al. 2023; Smith and Lim 2008) naturally lead to trilevel problems.

PART FIVE

SUPPLEMENTARY MATERIAL

Appendix A
Optimality Conditions

In this section, we briefly review the basics of linear and nonlinear optimization that we use throughout the book. We mainly discuss the corresponding duality theorems as well as the classic necessary (and sometimes also sufficient) first-order optimality conditions.

Most theoretical results in this chapter are given without proofs. These proofs can be found in any classic book on the respective topic. For linear optimization, we refer to Chvátal (1983), whereas nonlinear optimization theory is nicely introduced in Nocedal and Wright (2006) and Bertsekas (2016). Convex optimization basics can be found in Boyd and Vandenberghe (2004) and one of the most comprehensive books on mathematical problems with complementarity constraints is the one by Luo et al. (1996).

A.1 Linear Optimization

We first consider linear optimization problems of the form

$$\begin{aligned} \min_{x} \quad & c^\top x \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0, \end{aligned} \tag{A.1}$$

with $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, and $A \in \mathbb{R}^{m \times n}$. Problem (A.1) is the so-called *standard form* of a linear optimization problem (LP). It can be shown that every linear optimization problem can be written in this way by introducing suitable variable splittings or slack variables.

As usual, we call a vector $x \in \mathbb{R}^n$ *feasible* if it satisfies the constraints, i.e., if $Ax = b$ and $x \geq 0$ holds. Moreover, we call the problem *bounded* if there exists a finite constant $C \in \mathbb{R}$ with $c^\top x \geq C$ for all feasible points x. For linear optimization problems, we have the following existence result.

Theorem A.1 *The linear optimization problem* (A.1) *is either infeasible, unbounded, or solvable.*

Exercise A.2 Just as a refresher: Prove Theorem A.1.

The *dual problem* of the linear optimization problem (A.1) is the linear problem

$$\max_{\lambda} \quad b^\top \lambda \quad \text{s.t.} \quad A^\top \lambda \leq c. \tag{A.2}$$

Here and in what follows, we use Latin letters for primal variables and Greek letters for dual variables.

In bilevel optimization, we often use optimality conditions to replace optimization problems with these conditions. For linear optimization problems, these conditions are usually given by the strong-duality theorem. However, let us state the weak-duality theorem first.

Theorem A.3 (Weak-Duality Theorem) *Let $x \in \mathbb{R}^n$ be a feasible point of the primal problem* (A.1) *and let $\lambda \in \mathbb{R}^m$ be a feasible point of the dual problem* (A.2)*. Then,*

$$b^\top \lambda \leq c^\top x$$

holds.

Exercise A.4 Another refresher: Prove Theorem A.3.

Next, we state the strong-duality theorem.

Theorem A.5 (Strong-Duality Theorem) *Consider the pair* (A.1) *and* (A.2) *of primal and dual LPs. Then, the following statements are equivalent:*

(i) *Problems* (A.1) *and* (A.2) *are both feasible.*
(ii) *Problems* (A.1) *and* (A.2) *both have optimal solutions $x^* \in \mathbb{R}^n$ and $\lambda^* \in \mathbb{R}^m$ and*

$$c^\top x^* = b^\top \lambda^*$$

holds.
(iii) *Problems* (A.1) *and* (A.2) *both have a finite optimal objective value.*

Finally, we also state the complementarity slackness theorem.

Theorem A.6 (Complementarity-Slackness Theorem) *Consider the pair* (A.1) *and* (A.2) *of primal and dual LPs. Moreover, let $x \in \mathbb{R}^n$ be feasible for* (A.1) *and let $\lambda \in \mathbb{R}^m$ be feasible for* (A.2)*. Then, the following statements are equivalent:*

(i) The point $x \in \mathbb{R}^n$ *is optimal for* (A.1) *and* $\lambda \in \mathbb{R}^m$ *is optimal for* (A.2).

(ii) It holds

$$(c - A^\top \lambda)^\top x = 0.$$

(iii) For all components x_j *of the primal solution, it holds*

$$x_j > 0 \implies (A^\top \lambda)_j = c_j,$$

i.e., if the primal variable has slack ($x_j > 0$)*, the corresponding* j*th dual inequality is active.*

Next, we state two corollaries that are often used to tackle bilevel optimization problems with linear lower-level problems.

Corollary A.7 *The primal optimization problem* (A.1) *is equivalent to the system*

$$\begin{aligned} Ax &= b, \\ x &\geq 0, \\ A^\top \lambda &\leq c, \\ b^\top \lambda &\geq c^\top x. \end{aligned} \tag{A.3}$$

Here, "equivalent" means the following: Whenever x *is an optimal solution to the LP* (A.1)*, there exists a dual vector* λ *so that the pair* (x, λ) *satisfies* (A.3) *and whenever there exists* (x, λ) *that satisfies* (A.3)*,* x *is an optimal solution to* (A.1).

Proof: The claim follows directly from the strong-duality theorem. □

Corollary A.8 *The primal optimization problem* (A.1) *is equivalent to the system*

$$\begin{aligned} Ax &= b, \\ x &\geq 0, \\ c - A^\top \lambda &\geq 0, \\ x_i (c - A^\top \lambda)_i &= 0 \quad \text{for all } i = 1, \dots, n. \end{aligned} \tag{A.4}$$

Here, "equivalent" again means the following: Whenever x *is an optimal solution to the LP* (A.1)*, there exists a dual vector* λ *so that the pair* (x, λ) *satisfies* (A.4) *and whenever there exists* (x, λ) *that satisfies* (A.4)*,* x *is an optimal solution to* (A.1).

Proof: The claim follows directly from the complementarity slackness theorem. □

Exercise A.9 Consider the linear optimization problem

$$
\begin{aligned}
\min_{x,y} \quad & c^\top x + d^\top y \\
\text{s.t.} \quad & Ax + By \geq a, \\
& Cx + Dy = b, \\
& x \geq 0,
\end{aligned}
\tag{A.5}
$$

with $c \in \mathbb{R}^{n_x}$, $d \in \mathbb{R}^{n_y}$, $A \in \mathbb{R}^{m \times n_x}$, $B \in \mathbb{R}^{m \times n_y}$, $a \in \mathbb{R}^m$, $C \in \mathbb{R}^{\ell \times n_x}$, $D \in \mathbb{R}^{\ell \times n_y}$, and $b \in \mathbb{R}^\ell$. State the dual problem of (A.5).

Finally, we prove that the ℓ_1-penalization is exact for linear problems.

Theorem A.10 *Consider the linear optimization problem*

$$
\begin{aligned}
\min_{x} \quad & c^\top x \\
\textit{s.t.} \quad & Ax \geq b, \\
& Cx \geq d
\end{aligned}
\tag{A.6}
$$

with $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $C \in \mathbb{R}^{\ell \times n}$, and $d \in \mathbb{R}^\ell$. Moreover, let

$$
\begin{aligned}
\min_{x} \quad & c^\top x + \pi \sum_{i=1}^{\ell} [(Cx)_i - d_i]^- \\
\textit{s.t.} \quad & Ax \geq b
\end{aligned}
\tag{A.7}
$$

be its partially penalized counterpart with $[\xi]^- := \max\{0, -\xi\}$ for any $\xi \in \mathbb{R}$. Consider further

$$
\begin{aligned}
\max_{\lambda,\mu} \quad & b^\top \lambda + d^\top \mu \\
\textit{s.t.} \quad & \left[A^\top, C^\top\right] \begin{pmatrix} \lambda \\ \mu \end{pmatrix} = c, \\
& \lambda, \mu \geq 0,
\end{aligned}
\tag{A.8}
$$

which is the dual problem of the LP (A.6). Moreover, suppose that both (A.6) and (A.8) are solvable and that the penalty parameter π in (A.7) is chosen so that

$$
\pi > \|\mu^*\|_\infty \tag{A.9}
$$

holds for μ^* being part of an optimal solution (λ^*, μ^*) to the dual problem (A.8). Then, every optimal solution to (A.6) is an optimal solution to (A.7) and vice versa.

Proof: We start by re-writing (A.7) in its epigraph reformulation

$$\begin{aligned} \min_{x,\eta} \quad & c^\top x + \pi \sum_{i=1}^{\ell} \eta_i \\ \text{s.t.} \quad & Ax \geq b, \\ & \eta_i \geq d_i - C_{i\cdot} x \quad \text{for all } i = 1, \dots, \ell, \\ & \eta_i \geq 0 \quad \text{for all } i = 1, \dots, \ell. \end{aligned}$$

The dual of this problem is given by

$$\begin{aligned} \min_{\alpha,\beta} \quad & b^\top \alpha + d^\top \beta \\ \text{s.t.} \quad & \left[A^\top, C^\top\right] \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = c, \\ & \beta_i \leq \pi \quad \text{for all } i = 1, \dots, \ell, \\ & \alpha, \beta \geq 0. \end{aligned} \tag{A.10}$$

If we now choose π so that (A.9) is satisfied for μ^* being part of an optimal solution (λ^*, μ^*) to (A.8), this implies that the dual problems (A.10) and (A.8) have at least one solution in common if we identify (α, β) with (λ, μ). Consequently, their primal problems have the same solutions, which proves the claim. □

A.2 Nonlinear Optimization

In this section, we consider the situation in which some of the constraints or the objective function can be nonlinear. The general form of such a nonlinear optimization problem (NLP) reads

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \geq 0 \quad \text{for all } i \in I = \{1, \dots, m\}, \\ & h_j(x) = 0 \quad \text{for all } j \in J = \{1, \dots, p\}. \end{aligned} \tag{A.11}$$

We assume that the objective function $f : \mathbb{R}^n \to \mathbb{R}$ as well as the constraint functions $g_i : \mathbb{R}^n \to \mathbb{R}$, $i \in I$, and $h_j : \mathbb{R}^n \to \mathbb{R}$, $j \in J$, are continuously differentiable. The feasible set is denoted by $\mathcal{F}$.

For what follows, we denote by

$$B_\varepsilon(x^*) := \{x \in \mathbb{R}^n : \|x - x^*\| < \varepsilon\}$$

the open ε-ball at x^* and $\|x\| = \sqrt{x^\top x}$ denotes the Euclidean norm in $\mathbb{R}^n$.

Definition A.11 (Local Minimizer) A point $x^* \in \mathbb{R}^n$ is called a *local minimizer* of Problem (A.11) if x^* is feasible and if an $\varepsilon > 0$ exists such that $f(x) \geq f(x^*)$ for all $x \in \mathcal{F} \cap B_\varepsilon(x^*)$.

Definition A.12 (Strict Local Minimizer) A point $x^* \in \mathbb{R}^n$ is called a *strict local minimizer* of Problem (A.11) if x^* is feasible and if an $\varepsilon > 0$ exists such that $f(x) > f(x^*)$ for all $x \in (\mathcal{F} \cap B_\varepsilon(x^*)) \setminus \{x^*\}$.

Besides local minimizers, we also consider global minimizers.

Definition A.13 ((Strict) Global Minimizers) A point $x^* \in \mathbb{R}^n$ is called a *global minimizer* of Problem (A.11) if x^* is feasible and if $f(x) \geq f(x^*)$ holds for all $x \in \mathcal{F}$. The point is called a *strict global minimizer* if $f(x) > f(x^*)$ holds for all $x \in \mathcal{F} \setminus \{x^*\}$.

Naturally, the question arises under which assumptions a (global) minimizer of a nonlinear optimization problem exists. The answer is given by the Weierstraß theorem.

Theorem A.14 (Weierstraß Theorem) *Suppose that the set $\mathcal{F}$ is non-empty and compact and that the function $f : \mathcal{F} \to \mathbb{R}$ is continuous. Then, f has at least one global minimizer and at least one global maximizer on $\mathcal{F}$.*

Our goal now is to state the first-order optimality conditions of Problem (A.11), i.e., the Karush–Kuhn–Tucker (KKT) conditions. To this end, we need some more notation.

Definition A.15 (Lagrangian Function) The function

$$\mathcal{L}(x, \lambda, \mu) := f(x) - \sum_{i=1}^{m} \lambda_i g_i(x) - \sum_{j=1}^{p} \mu_j h_j(x)$$

is called *Lagrangian function* of Problem (A.11).

Using the Lagrangian function, we can now define the KKT conditions.

Definition A.16 (KKT Conditions, KKT Point, Lagrange Multipliers) We consider Problem (A.11) with continuously differentiable functions f, g, and h.[1]

(i) The conditions

$$\begin{aligned} \nabla_x \mathcal{L}(x, \lambda, \mu) &= 0, \\ h(x) &= 0, \\ \lambda \geq 0,\ g(x) \geq 0,\ \lambda^\top g(x) &= 0 \end{aligned} \tag{A.12}$$

[1] A quantity without an index usually stands for the vector; e.g., $h(x) = (h_j(x))_{j=1,\dots,p}$.

are called *Karush–Kuhn–Tucker* (or *KKT*) *conditions* of Problem (A.11). Here and in what follows,

$$\nabla_x \mathcal{L}(x, \lambda, \mu) = \nabla f(x) - \sum_{i=1}^{m} \lambda_i \nabla g_i(x) - \sum_{j=1}^{p} \mu_j \nabla h_j(x)$$

is the gradient of the Lagrangian function with respect to the variables x.

(ii) Every vector $((x^*)^\top, (\lambda^*)^\top, (\mu^*)^\top)^\top \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ that satisfies the KKT conditions is called a *KKT point* of Problem (A.11). The components of λ^* and μ^* are called *Lagrange multipliers*.

For stating the first KKT theorem, we need a so-called constraint qualification (CQ).

Definition A.17 (Abadie Constraint Qualification) We say that a feasible point $x \in \mathcal{F}$ of Problem (A.11) satisfies the *Abadie constraint qualification (ACQ)* if $T_X(x) = T_{\text{lin}}(x)$ holds.

In the last definition, we use the two cones $T_X(x)$, i.e., the tangential cone of X at x, and $T_{\text{lin}}(x)$, i.e., the linearized tangential cone of X at x. We do not discuss these cones in detail here, but they can be found in any textbook on nonlinear optimization; see, e.g., Nocedal and Wright (2006) or Bertsekas (2016).

With these definitions, we can now state the KKT theorem under the ACQ.

Theorem A.18 (KKT Theorem under the ACQ) *Let $x^* \in \mathbb{R}^n$ be a local minimizer of Problem* (A.11)*. Moreover, suppose that the Abadie CQ holds at x^* and that all functions f, g_i, $i \in I$, and h_j, $j \in J$, are continuously differentiable. Then, there exist Lagrange multipliers $\lambda^* \in \mathbb{R}^m$ and $\mu^* \in \mathbb{R}^p$ so that (x^*, λ^*, μ^*) is a KKT point of Problem* (A.11)*.*

The KKT theorem also holds under other constraint qualifications that are stronger than the ACQ, where "stronger" means that the other constraint qualification implies the ACQ. One very prominent example is the LICQ, for which we first need one more notation.

Definition A.19 (Active Inequality Constraints) Let $x \in \mathcal{F}$ be a feasible point of Problem (A.11). Then, the set

$$I(x) := \{i \in I : g_i(x) = 0\}$$

is called the *set of active inequality constraints* at the point x.

Definition A.20 (Linear Independence Constraint Qualification) Let $x \in \mathbb{R}^n$ be feasible for Problem (A.11) and let $I(x)$ be the set of active inequality constraints at x. We say that the *linear independence constraint qualification (LICQ)* is satisfied in x if the gradients

$$\begin{aligned} &\nabla g_i(x) \quad \text{for all } i \in I(x), \\ &\nabla h_j(x) \quad \text{for all } j = 1, \dots, p, \end{aligned}$$

are linearly independent.

Because the LICQ implies the ACQ, the following theorem holds.

Theorem A.21 (KKT Theorem under the LICQ) *Let $x^* \in \mathbb{R}^n$ be a local minimizer of Problem (A.11) that satisfies the LICQ and suppose that all functions f, g_i, $i \in I$, and h_j, $j \in J$, are continuously differentiable. Then, there exist Lagrange multipliers $\lambda^* \in \mathbb{R}^m$ and $\mu^* \in \mathbb{R}^p$ so that (x^*, λ^*, μ^*) is a KKT point of Problem (A.11). Moreover, (λ^*, μ^*) are uniquely defined.*

A.3 Convex Optimization

We now consider convex optimization problems, which are optimization problems of the form given in (A.11) with $f : \mathcal{F} \to \mathbb{R}$ being a convex function and $\mathcal{F}$ being a convex set. Hence, we are minimizing a convex objective function over a convex feasible set. For more details about convex optimization, we refer to the seminal textbook by Boyd and Vandenberghe (2004).

One of the most important properties of convex optimization problems is that local and global optima coincide and that these points are characterized by first-order conditions.

Theorem A.22 *Let $f : \mathcal{F} \to \mathbb{R}$ be a convex function defined on a convex set $\mathcal{F} \subseteq \mathbb{R}^n$. Then, the following statements are true.*

(i) Every local minimizer of f on $\mathcal{F}$ is also a global minimizer of f on $\mathcal{F}$.
(ii) If f is strictly convex, then f has at most one local minimizer on $\mathcal{F}$ and this local minimizer (if it exists) is then also the unique global minimizer of f on $\mathcal{F}$.
(iii) Let $\mathcal{F}$ be open, f be continuously differentiable on $\mathcal{F}$, and suppose that $x^* \in \mathcal{F}$ is a stationary point of f, i.e., a point x for which $\nabla f(x) = 0$ holds. Then, x^* is a global minimizer of f on $\mathcal{F}$.

Exercise A.23 Prove Theorem A.22.

Exercise A.24 Regarding Claim (ii) of Theorem A.22: Construct an optimization problem with f being strictly convex and $\mathcal{F}$ being convex for which no minimizer exists.

In the remainder of this section, we study the meaning of the KKT conditions for convex optimization problems. To this end, we consider the problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \geq 0 \quad \text{for all } i = 1, \dots, m, \\ & b_j^\top x = \beta_j \quad \text{for all } j = 1, \dots, p, \end{aligned} \tag{A.13}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \dots, m$, are continuously differentiable, $b_j \in \mathbb{R}^n$, $j = 1, \dots, p$, are vectors and $\beta_j \in \mathbb{R}$, $j = 1, \dots, p$, are scalars. Moreover, f is convex and the functions g_i, $i = 1, \dots, m$, are concave. Thus, we consider a convex objective function over a convex feasible set.

Definition A.25 (Slater's Constraint Qualification) We say that the convex problem (A.13) satisfies *Slater's constraint qualification* if there exists a vector $\hat{x} \in \mathbb{R}^n$ so that

$$\begin{aligned} g_i(\hat{x}) > 0 \quad & \text{for all } i = 1, \dots, m, \\ b_j^\top \hat{x} = \beta_j \quad & \text{for all } j = 1, \dots, p \end{aligned}$$

holds. This means that $\hat{x}$ is strictly feasible w.r.t. the inequality constraints and feasible w.r.t. the equality constraints.

Note that a convex problem satisfying Slater's CQ possesses a non-empty interior of the feasible set defined by the inequality constraints.

Theorem A.26 (KKT Theorem for Convex Problems under Slater's CQ) *Let $x^* \in \mathbb{R}^n$ be a local (and thus global) minimizer of the convex problem* (A.13). *Moreover, suppose that Slater's CQ is satisfied and that all functions f and g_i, $i \in I$, are continuously differentiable. Then, there exist Lagrange multipliers $\lambda^* \in \mathbb{R}^m$ and $\mu^* \in \mathbb{R}^p$ so that (x^*, λ^*, μ^*) satisfies the KKT conditions*

$$\begin{aligned} \nabla f(x^*) - \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) - \sum_{j=1}^p \mu_j^* b_j = 0, & \\ b_j^\top x^* = \beta_j \quad & \textit{for all } j = 1, \dots, p, \\ g_i(x^*) \geq 0 \quad & \textit{for all } i = 1, \dots, m, \\ \lambda_i^* g_i(x^*) = 0 \quad & \textit{for all } i = 1, \dots, m, \\ \lambda_i^* \geq 0 \quad & \textit{for all } i = 1, \dots, m, \end{aligned}$$

of Problem (A.13).

Up to now, we have seen that the KKT conditions are necessary first-order optimality conditions for convex problems under Slater's CQ. For general nonlinear problems, the KKT conditions are not sufficient. For convex problems, however, the KKT conditions are also sufficient conditions.

Theorem A.27 *Let $(x^*, \lambda^*, \mu^*) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ be a KKT point of the convex problem (A.13). Then, x^* is a local (and thus global) minimizer of Problem (A.13).*

Exercise A.28 Prove Theorem A.27.

Exercise A.29 Consider the quadratic problem

$$\begin{aligned} \min_{x} \quad & \frac{1}{2} x^\top Q x + c^\top x \\ \text{s.t.} \quad & Ax \geq b, \\ & Cx = d, \end{aligned} \tag{A.14}$$

with $c \in \mathbb{R}^n$, $Q \in \mathbb{R}^{n \times n}$ being symmetric and positive semi-definite, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $C \in \mathbb{R}^{\ell \times n}$, and $d \in \mathbb{R}^\ell$.

(i) Is Problem (A.14) a convex optimization problem?
(ii) Derive the KKT conditions of Problem (A.14).

Hint: You may use the following theorem without proof.

Theorem A.30 *Let $f : \mathcal{F} \to \mathbb{R}$ be twice continuously differentiable on the open set $\mathcal{F} \subseteq \mathbb{R}^n$. Then, the function f is convex if and only if the Hessian matrix $\nabla^2 f(x)$ is positive semi-definite for all $x \in \mathcal{F}$.*

A.4 Mathematical Programs with Complementarity Constraints

We now consider problems of the form

$$\min_{x} \quad f(x) \tag{A.15a}$$
$$\text{s.t.} \quad g_i(x) \geq 0 \quad \text{for all } i \in I = \{1, \dots, m\}, \tag{A.15b}$$
$$h_j(x) = 0 \quad \text{for all } j \in J = \{1, \dots, p\}, \tag{A.15c}$$
$$\varphi_\ell(x) \geq 0 \quad \text{for all } \ell \in \{1, \dots, r\}, \tag{A.15d}$$
$$\psi_\ell(x) \geq 0 \quad \text{for all } \ell \in \{1, \dots, r\}, \tag{A.15e}$$
$$\varphi_\ell(x)\psi_\ell(x) = 0 \quad \text{for all } \ell \in \{1, \dots, r\}. \tag{A.15f}$$

Problem (A.15) is called a *mathematical program with complementarity constraints (MPCC)* and it can be seen as the nonlinear optimization problem (A.11) extended by the two functions $\varphi, \psi : \mathbb{R}^n \to \mathbb{R}^r$ and the three last sets of constraints (A.15d)–(A.15f). In addition to the functions f, g, and h, we also assume that the functions φ and ψ are continuously differentiable so that Problem (A.15) "looks like" a usual NLP. However, the three last sets of constraints add significant difficulty to the problem. The reason is that the LICQ does not hold at any feasible point of (A.15).

Theorem A.31 *Let x be feasible for Problem* (A.15)*. Then, the LICQ does not hold at x.*

Exercise A.32 Prove Theorem A.31.

Example A.33 In Corollary A.8, we consider a setting in which the feasible set has the structure of the feasible set of Problem (A.15). The set of inequalities and equations in this corollary reads

$$\begin{aligned} Ax &= b, \\ x &\geq 0, \\ c - A^\top \lambda &\geq 0, \\ x_i (c - A^\top \lambda)_i &= 0 \quad \text{for all } i \in \{1, \dots, n\}. \end{aligned}$$

With $h(x) = Ax - b$, $\varphi(x) = x$, $\psi(x, \lambda) = c - A^\top \lambda$, and g not being present, this exactly matches the setting in Problem (A.15). △

Example A.34 The KKT conditions (A.12) in Definition A.16 can be written in the form of the constraints of the MPCC (A.15). △

The interested reader is referred to the textbook by Luo et al. (1996) for more details on MPCCs.

A.5 What You Should Know Now!

1. What is an LP in standard form?
2. What is the feasible set of an LP?
3. When do we call an LP bounded?
4. What do you know about the solvability of LPs?
5. What is the dual LP?
6. What is the statement of the weak-duality theorem of linear optimization?
7. What is the statement of the strong-duality theorem of linear optimization?

8. What is the statement of the complementarity-slackness theorem of linear optimization?
9. How can we use the strong-duality theorem to re-write an LP as a system of equalities and inequalities?
10. How can we use the complementarity-slackness theorem to re-write an LP as a system of equalities and inequalities?
11. What is the "standard form" of an NLP?
12. How do we define a local minimizer?
13. How do we define a strict local minimizer?
14. How do we define a global minimizer?
15. How do we define a strict global minimizer?
16. What are active inequality constraints?
17. What is the Lagrangian function of an NLP?
18. What is a Lagrange multiplier?
19. What is the relation between Lagrange multipliers and dual variables?
20. What are the KKT conditions of an NLP?
21. What is the ACQ?
22. What is the statement of the KKT theorem?
23. What is the relation between the KKT theorem and the complementarity-slackness theorem?
24. What is the LICQ?
25. What does the KKT theorem under the LICQ say?
26. What is a convex optimization problem?
27. What are the nice aspects about convex optimization problems? Why are they so special?
28. What specific properties do equality and inequality constraints need to have so that the resulting feasible set is convex?
29. How is Slater's CQ defined?
30. What is the geometric meaning of Slater's CQ?
31. Can you state the KKT theorem for convex problems?
32. What do you know about the relationship between KKT points and global minimizers in the case of convex optimization problems?
33. What is an MPCC?
34. Why are MPCCs different than "usual" NLPs?
35. What is the relation between MPCCs and optimality conditions of optimization problems?

Appendix B

Polyhedra: Basic Notions and Results

In this chapter, we briefly recap the basic definitions and results regarding polyhedra that we need throughout the book. For more details, we refer the interested reader to Schrijver (1998).

An intersection of a finite number of halfspaces is a *polyhedron*, which we denote as $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, and a bounded polyhedron is called a *polytope*. Let us now consider a given non-empty polyhedron $P \subseteq \mathbb{R}^n$. A point $x \in P$ is an *extreme point* of P if $x = \lambda x^1 + (1 - \lambda)x^2$ with $0 < \lambda < 1$ and $x^1, x^2 \in P$ implies $x = x^1 = x^2$. Moreover, we call a vector r a *ray* of P if $r \neq 0$ and $x \in P$ implies $x + \mu r \in P$ for all $\mu \geq 0$. A ray r is an *extreme ray* of P if r is a ray of P and $r = \mu_1 r^1 + \mu_2 r^2$ with $(\mu_1, \mu_2) \in \mathbb{R}^2_{\geq 0} \setminus \{0\}$ and r^1, r^2 being rays of P implies $r^1 = \alpha r^2$ for some $\alpha > 0$.

A *polyhedral cone* is a polyhedron for which all hyperplanes contain the origin, i.e., $C = \{x \in \mathbb{R}^n : Ax \leq 0\}$. A *half-line* starting from vertex x_0 in direction d is the set of points

$$\{x \in \mathbb{R}^n : x = x_0 + \lambda d,\ \lambda \geq 0\}.$$

From now on, we assume that $P = \{x \in \mathbb{R}^n : Ax \leq b\} \neq \emptyset$ and $\operatorname{rank}(A) = n$, which is necessary for P to have extreme points. We say that d is a *recession direction* of P if

$$x_0 \in P \implies \{x \in \mathbb{R}^n : x = x_0 + \lambda d,\ \lambda \geq 0\} \subseteq P$$

holds. The set

$$\operatorname{rec}(P) = \{d \in \mathbb{R}^n : Ad \leq 0\}$$

is a polyhedral cone, also known as the *recession cone* of P. Any $d \in \operatorname{rec}(P) \setminus \{0\}$ determines a recession direction of P. Indeed, let us consider a point $x_0 \in P$, an arbitrary $d \in \operatorname{rec}(P) \setminus \{0\}$, and $\lambda \geq 0$. Then, we have $x' = x_0 + \lambda d \in P$ because

$$Ax' = A(x_0 + \lambda d) = Ax_0 + \lambda Ad \leq Ax_0 \leq b.$$

The edges of a pointed polyhedral cone are half-lines starting from the origin and they correspond to the extreme rays of this cone.

An extreme ray of a pointed polyhedral cone C is a face of dimension one of C. The extreme rays of a pointed polyhedron P are the extreme rays of the recession cone $\operatorname{rec}(P)$ of P.

Proposition B.1 *Let $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ be a pointed polyhedron and let r be a ray of P. Then, r is extreme if and only if r satisfies $n - 1$ linearly independent inequalities of $Ax \leq 0$ with equality.*

Proposition B.2 *Let $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ be a pointed polyhedron and let $\bar{x} \in P$. Then, $\bar{x}$ is an extreme point of P if and only if $\bar{x}$ satisfies n linearly independent inequalities of $Ax \leq b$ with equality.*

If a polyhedron P is pointed, it can be written as a Minkowski sum of its extreme points and extreme rays.

Proposition B.3 (Minkowski) *Every polyhedron P can be represented as*

$$P = \left\{ x \in \mathbb{R}^n : x = \sum_{i=1}^{p} \lambda_i v^i + \sum_{j=1}^{q} \mu_j r^j, \ \sum_{i=1}^{p} \lambda_i = 1, \ \lambda \geq 0, \ \mu \geq 0 \right\},$$

where v^i, $i \in \{1, \ldots, p\}$, are the extreme points of P and r^j, $j \in \{1, \ldots, q\}$, are the extreme rays of P.

In other words, P can be represented as a Minkowski sum of a polytope $Q = \operatorname{conv}(v^1, \ldots, v^p)$ and a finitely generated cone $C = \operatorname{cone}(r^1, \ldots, r^q)$:

$$P = Q + C = \operatorname{conv}(v^1, \ldots, v^p) + \operatorname{cone}(r^1, \ldots, r^q).$$

List of Figures

1.1	The transportation network considered in Exercise 1.3.	15
1.2	An illustration of the bilevel problem (1.6).	26
2.1	The optimal solution y to the lower-level problem in Example 2.1.	34
2.2	The graph of the mapping $x \mapsto F(x, \mathcal{S}(x))$ in Example 2.1. Taken and modified from Dempe et al. (2015).	34
3.1	The feasible region of the lower-level problem in Example 3.13.	58
3.2	An illustration of the lower-level problem in Example 3.16.	61
3.3	An illustration of the bilevel problem in Example 3.18.	64
4.1	An illustration of the bilevel problem in Example 4.1.	71
4.2	An illustration of the bilevel problem in Example 4.2.	73
4.3	Modeling a binary variable using a linear bilevel problem.	81
5.1	An illustration of the bilevel problem (5.4).	88
5.2	A branch-and-bound tree illustrating a global lower bound.	93
5.3	The branch-and-bound tree for Problem (5.9) using DFS.	104
5.4	The branch-and-bound tree for Problem (5.9) using BFS.	105
6.1	An illustration of the bilevel problem in Example 6.1.	118
6.2	An illustration of the bilevel problem in Example 6.2.	120
6.3	An illustration of the bilevel problem in Example 6.12.	125
8.1	An illustration of the bilevel problem in Example 8.5.	142
8.2	An illustration of the bilevel problem in Example 8.6.	144
8.3	The branch-and-bound tree in Example 8.12.	148
8.4	The branch-and-bound tree in Example 8.15.	154
8.5	Generating two new sub-problems in Example 8.15.	155
8.6	The branch-and-bound tree in Example 8.16.	157
9.1	An illustration of a generic valid inequality.	163
9.2	An illustration of a valid inequality in Example 9.3.	165
9.3	An illustration of a valid inequality derived from Proposition 9.7.	169
9.4	The branch-and-cut tree in Example 9.12.	175
9.5	A node problem in the branch-and-cut tree in Example 9.12.	177
9.6	A node problem in the branch-and-cut tree in Example 9.12.	178

10.1 An illustration of the interdiction problem in Example 10.4. 189
10.2 An illustration of the interdiction problem in Example 10.6. 193
10.3 An illustration of the interdiction problem in Example 10.6. 194
11.1 The pointed polyhedral cone in Example 11.1. 228
11.2 An illustration of an intersection cut in Example 11.2. 229
12.1 An illustration of an intersection cut in Example 12.8. 244
12.2 An illustration of an intersection cut in Example 12.8. 244
12.3 An illustration of an intersection cut in Example 12.21. 255
12.4 An illustration of an intersection cut in Example 12.21. 257
12.5 An illustration of an intersection cut in Example 12.22. 258
12.6 An illustration of an intersection cut in Example 12.23. 259
14.1 An illustration of the bilevel problem (14.8). 281
14.2 An illustration of the deterministic bilevel problem (14.10). 282
14.3 An illustration of the bilevel problem (14.10) under uncertainty. 284

List of Algorithms

1	The Kth-Best Algorithm	86
2	Branch-and-Bound for LP-LP Bilevel Problems	92
3	A Standard Alternating Direction Method	107
4	The ℓ_1-Penalty Alternating Direction Method	109
5	Branch-and-Bound for MILPs	136
6	Processing Node k of the Branch-and-Bound Search Tree	150
7	Processing Node k of the Branch-and-Cut Search Tree	171
8	Multi-Tree Implementation of Benders Decomposition	208
9	Processing Node k of the Branch-and-Cut Search Tree	210
10	Penalize-and-Relax-and-Dualize Heuristic	270

Bibliography

Aboussoror, A. and P. Loridan (1995). “Strong-weak Stackelberg Problems in Finite Dimensional Spaces.” In: *Serdica. Mathematical Journal* 21.2, pp. 151–170. URL: https://www.math.bas.bg/serdica/1995/1995-151-170.pdf.

Achterberg, T., T. Koch, and A. Martin (2005). “Branching rules revisited.” In: *Operations Research Letters* 33.1, pp. 42–54. DOI: 10.1016/j.orl.2004.04.002.

Alguacil, N., A. Delgadillo, and J. M. Arroyo (2014). “A trilevel programming approach for electric grid defense planning.” In: *Computers & Operations Research* 41, pp. 282–290. DOI: 10.1016/j.cor.2013.06.009.

Ambrosius, M., V. Grimm, T. Kleinert, F. Liers, M. Schmidt, and G. Zöttl (2020). “Endogenous Price Zones and Investment Incentives in Electricity Markets: An Application of Multilevel Optimization with Graph Partitioning.” In: *Energy Economics* 92, p. 104879. DOI: 10.1016/j.eneco.2020.104879.

Anandalingam, G. and T. L. Friesz (1992). “Hierarchical optimization: An introduction.” In: *Annals of Operations Research* 34.1, pp. 1–11. DOI: 10.1007/BF02098169.

Anandalingam, G. and D. J. White (1990). “A solution method for the linear static Stackelberg problem using penalty functions.” In: *IEEE Transactions on Automatic Control* 35.10, pp. 1170–1173. DOI: 10.1109/9.58565.

Arulselvan, A., C. W. Commander, L. Elefteriadou, and P. M. Pardalos (2009). “Detecting critical nodes in sparse graphs.” In: *Computers & Operations Research* 36.7, pp. 2193–2200. DOI: 10.1016/j.cor.2008.08.016.

Audet, C., J. Haddad, and G. Savard (2006). “A note on the definition of a linear bilevel programming solution.” In: *Applied Mathematics and Computation* 181.1, pp. 351–355. DOI: 10.1016/j.amc.2006.01.043.

Audet, C., J. Haddad, and G. Savard (2007a). "Disjunctive Cuts for Continuous Linear Bilevel Programming." In: *Optimization Letters* 1.3, pp. 259–267. DOI: 10.1007/s11590-006-0024-3.

Audet, C., P. Hansen, B. Jaumard, and G. Savard (1997). "Links between linear bilevel and mixed 0–1 programming problems." In: *Journal of Optimization Theory and Applications* 93.2, pp. 273–300. DOI: 10.1023/A:1022645805569.

Audet, C., G. Savard, and W. Zghal (2007b). "New Branch-and-Cut Algorithm for Bilevel Linear Programming." In: *Journal of Optimization Theory and Applications* 134.2, pp. 353–370. DOI: 10.1007/s10957-007-9263-4.

Aussel, D., P. Bendotti, and M. Pištěk (2017a). "Nash equilibrium in a pay-as-bid electricity market: Part 1 — existence and characterization." In: *Optimization* 66.6, pp. 1013–1025. DOI: 10.1080/02331934.2016.1227981.

Aussel, D., P. Bendotti, and M. Pištěk (2017b). "Nash equilibrium in a pay-as-bid electricity market Part 2 — best response of a producer." In: *Optimization* 66.6, pp. 1027–1053. DOI: 10.1080/02331934.2016.1227982.

Aussel, D., G. Bouza, S. Dempe, and S. Lepaul (2021). "Genericity Analysis of Multi-Leader-Disjoint-Followers Game." In: *SIAM Journal on Optimization* 31.3, pp. 2055–2079. DOI: 10.1137/20M1356476.

Aussel, D. and A. Svensson (2018). "Some remarks about existence of equilibria, and the validity of the EPCC reformulation for multi-leader-follower games." In: *Journal of Nonlinear and Convex Analysis* 19.7, pp. 1141–1162. URL: http://www.yokohamapublishers.jp/online2/opjnca/vol19/p1141.html.

Aussel, D. and A. Svensson (2019a). "Is Pessimistic Bilevel Programming a Special Case of a Mathematical Program with Complementarity Constraints?" In: *Journal of Optimization Theory and Applications* 181.2, pp. 504–520. DOI: 10.1007/s10957-018-01467-7.

Aussel, D. and A. Svensson (2019b). "Towards Tractable Constraint Qualifications for Parametric Optimisation Problems and Applications to Generalised Nash Games." In: *Journal of Optimization Theory and Applications* 182, pp. 404–416. DOI: 10.1007/s10957-019-01529-4.

Aussel, D. and A. Svensson (2020). "A Short State of the Art on Multi-Leader-Follower Games." In: *Bilevel Optimization: Advances and Next Challenges*. Ed. by S. Dempe and A. B. Zemkoho. Springer Optimization and Its Applications. Springer Cham, pp. 53–76. DOI: 10.1007/978-3-030-52119-6_3.

Aussel, D., C. Egea, and M. Schmidt (2025). "A tutorial on solving single-leader-multi-follower problems using SOS1 reformulations." In: *International Transactions in Operational Research* 32.3, pp. 1227–1250. DOI: 10.1111/itor.13466.

Bagwell, K. (1995). "Commitment and observability in games." In: *Games and Economic Behavior* 8.2, pp. 271–280. doi: 10.1016/S0899-8256(05)80001-6.

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

Balas, E. (2018). *Disjunctive Programming*. Springer Cham. doi: 10.1007/978-3-030-00148-3.

Balas, E. and M. Perregaard (2002). "Lift-and-Project for Mixed 0–1 Programming: Recent Progress." In: *Discrete Applied Mathematics* 123.1–3, pp. 129–154. doi: 10.1016/S0166-218X(01)00340-7.

Ball, M. O., B. L. Golden, and R. V. Vohra (1989). "Finding the most vital arcs in a network." In: *Operations Research Letters* 8.2, pp. 73–76. doi: 10.1016/0167-6377(89)90003-5.

Bard, J. F. (1988). "Convex two-level optimization." In: *Mathematical Programming* 40.1, pp. 15–27. doi: 10.1007/BF01580720.

Bard, J. F. (1998). *Practical Bilevel Optimization: Algorithms and Applications*. Nonconvex Optimization and Its Applications. Springer New York, NY. doi: 10.1007/978-1-4757-2836-1.

Bard, J. F. and J. T. Moore (1990). "A branch and bound algorithm for the bilevel programming problem." In: *SIAM Journal on Scientific and Statistical Computing* 11.2, pp. 281–292. doi: 10.1137/0911017.

Bazgan, C., S. Toubaline, and D. Vanderpooten (2013). "Critical edges/nodes for the minimum spanning tree problem: complexity and approximation." In: *Journal of Combinatorial Optimization* 26.1, pp. 178–189. doi: 10.1007/s10878-011-9449-4.

Beale, E. M. L. and J. A. Tomlin (1970). "Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables." In: *Proceedings of the Fifth International Conference on Operational Research*. Ed. by J. Lawrence. Tavistock Publications, pp. 447–454.

Beck, Y., I. Ljubić, and M. Schmidt (2022). "A Brief Introduction to Robust Bilevel Optimization." In: *SIAG on Optimization Views and News. A Forum for the SIAM Activity Group on Optimization* 30.2, pp. 1–10. url: https://siagoptimization.github.io/assets/views/ViewsAndNews-30-2.pdf.

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

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

Beck, Y., I. Ljubić, and M. Schmidt (2026). "Heuristic Methods for Mixed-Integer, Linear, and Γ-Robust Bilevel Problems." In: *INFORMS Journal on Computing* 38.1, pp. 165–188. DOI: 10.1287/ijoc.2023.0239.

Belotti, P., C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan (2013). "Mixed-integer nonlinear optimization." In: *Acta Numerica* 22, pp. 1–131. DOI: 10.1017/S0962492913000032.

Ben-Ayed, O. (1993). "Bilevel linear programming." In: *Computers & Operations Research* 20.5, pp. 485–501. DOI: 10.1016/0305-0548(93)90013-9.

Ben-Ayed, O. and C. E. Blair (1990). "Computational Difficulties of Bilevel Linear Programming." In: *Operations Research* 38.3, pp. 556–560. DOI: 10.1287/opre.38.3.556.

Ben-Tal, A., L. El Ghaoui, and A. Nemirovski (2009). *Robust Optimization*. Princeton University Press. URL: https://www2.isye.gatech.edu/~nemirovs/FullBookDec11.pdf.

Ben-Tal, A., A. Goryashko, E. Guslitzer, and A. Nemirovski (2004). "Adjustable robust solutions of uncertain linear programs." In: *Mathematical Programming* 99.2, pp. 351–376. DOI: 10.1007/s10107-003-0454-y.

Ben-Tal, A. and A. Nemirovski (1998). "Robust Convex Optimization." In: *Mathematics of Operations Research* 23.4, pp. 769–805. DOI: 10.1287/moor.23.4.769.

Benders, J. F. (1962). "Partitioning procedures for solving mixed-variables programming problems." In: *Numerische Mathematik* 4.1, pp. 238–252. DOI: 10.1007/BF01386316.

Benson, H. P. (1989). "On the structure and properties of a linear multilevel programming problem." In: *Journal of Optimization Theory and Applications* 60.3, pp. 353–373. DOI: 10.1007/BF00940342.

Bentoumi, I., F. Furini, A. R. Mahjoub, and S. Martin (2025). "Integer linear programming formulations for the maximum flow blocker problem." In: *European Journal of Operational Research* 324.3, pp. 742–758. DOI: 10.1016/j.ejor.2025.02.013.

Bertsekas, D. P. (2016). *Nonlinear Programming*. Athena Scientific Belmont. URL: http://www.athenasc.com/nonlinbook.html.

Bertsimas, D., D. B. Brown, and C. Caramanis (2011). "Theory and Applications of Robust Optimization." In: *SIAM Review* 53.3, pp. 464–501. DOI: 10.1137/080734510.

Bertsimas, D. and M. Sim (2003). “Robust discrete optimization and network flows.” In: *Mathematical Programming* 98, pp. 49–71. DOI: 10.1007/s10107-003-0396-4.

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

Besançon, M., M. F. Anjos, and L. Brotcorne (2021). “Complexity of near-optimal robust versions of multilevel optimization problems.” In: *Optimization Letters* 15.8, pp. 2597–2610. DOI: 10.1007/s11590-021-01754-9.

Bialas, W. F. and M. H. Karwan (1978). *Multilevel linear programming*. Tech. rep. State University of New York at Buffalo.

Bialas, W. F. and M. H. Karwan (1984). “Two-Level Linear Programming.” In: *Management Science* 30.8, pp. 1004–1020. DOI: 10.1287/mnsc.30.8.1004.

Birge, J. R. and F. Louveaux (2011). *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering. Springer New York, NY. DOI: 10.1007/978-1-4614-0237-4.

Bolusani, S. and T. K. Ralphs (2022). “A framework for generalized Benders’ decomposition and its application to multilevel optimization.” In: *Mathematical Programming* 196, pp. 389–426. DOI: 10.1007/s10107-021-01763-7.

Bonnans, J. F. and A. Shapiro (2000). *Perturbation Analysis of Optimization Problems*. Springer New York, NY. DOI: 10.1007/978-1-4612-1394-9.

Borrero, J. S., O. A. Prokopyev, and D. Sauré (2019). “Sequential Interdiction with Incomplete Information and Learning.” In: *Operations Research* 67.1, pp. 72–89. DOI: 10.1287/opre.2018.1773.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011). “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers.” In: *Foundations and Trends in Machine learning* 3.1, pp. 1–122. DOI: 10.1561/2200000016.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press. URL: https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf.

Bracken, J. and J. T. McGill (1973). “Mathematical Programs with Optimization Problems in the Constraints.” In: *Operations Research* 21.1, pp. 37–44. DOI: 10.1287/opre.21.1.37.

Brotcorne, L., F. Cirinei, P. Marcotte, and G. Savard (2011). “An exact algorithm for the network pricing problem.” In: *Discrete Optimization* 8.2, pp. 246–258. DOI: 10.1016/j.disopt.2010.09.003.

Brotcorne, L., M. Labbé, P. Marcotte, and G. Savard (2001). “A Bilevel Model for Toll Optimization on a Multicommodity Transportation Network.” In:

Transportation Science 35.4, pp. 345–358. doi: 10.1287/trsc.35.4.345.10433.

Brotcorne, L., M. Labbé, P. Marcotte, and G. Savard (2008). "Joint Design and Pricing on a Network." In: *Operations Research* 56.5, pp. 1104–1115. doi: 10.1287/opre.1080.0617.

Buchheim, C. (2023). "Bilevel linear optimization belongs to NP and admits polynomial-size KKT-based reformulations." In: *Operations Research Letters* 51.6, pp. 618–622. doi: 10.1016/j.orl.2023.10.006.

Calvete, H. I. and C. Galé (2007). "Linear bilevel multi-follower programming with independent followers." In: *Journal of Global Optimization* 39.3, pp. 409–417. doi: 10.1007/s10898-007-9144-2.

Candler, W. and R. D. Norton (1977). *Multi-level Programming*. Discussion Papers, Development Research Center, International Bank for Reconstruction and Development. World Bank. url: https://books.google.de/books?id=TicmAQAAMAAJ.

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

Caprara, A., M. Carvalho, A. Lodi, and G. J. Woeginger (2016). "Bilevel Knapsack with Interdiction Constraints." In: *INFORMS Journal on Computing* 28.2, pp. 319–333. doi: 10.1287/ijoc.2015.0676.

Caprara, A. and M. Fischetti (1996). "{0, 1/2}-Chvátal-Gomory cuts." In: *Mathematical Programming* 74.3, pp. 221–235. doi: 10.1007/BF02592196.

Caselli, G., M. Iori, and I. Ljubić (2026). "Bilevel optimization with sustainability perspective: A survey on applications." In: *European Journal of Operational Research* 332.2, pp. 357–375. doi: 10.1016/j.ejor.2025.08.051.

Cerulli, M., D. Serra, C. Sorgente, C. Archetti, and I. Ljubić (2023). "Mathematical programming formulations for the Collapsed k-Core Problem." In: *European Journal of Operational Research* 311.1, pp. 56–72. doi: 10.1016/j.ejor.2023.04.038.

Chvátal, V. (1983). *Linear Programming*. A Series of books in the mathematical sciences. Freeman.

Clark, F. E. (1961). "Remark on the Constraint Sets in Linear Programming." In: *The American Mathematical Monthly* 68.4, pp. 351–352. doi: 10.2307/2311583.

Clautiaux, F. and I. Ljubić (2025). "Last fifty years of integer linear programming: A focus on recent practical advances." In: *European Journal of Operational Research* 324.3, pp. 707–731. doi: 10.1016/j.ejor.2024.11.018.

Colson, B., P. Marcotte, and G. Savard (2005). "Bilevel programming: A survey." In: *4OR* 3.2, pp. 87–107. doi: 10.1007/s10288-005-0071-0.

Colson, B., P. Marcotte, and G. Savard (2007). "An overview of bilevel optimization." In: *Annals of Operations Research* 153.1, pp. 235–256. DOI: 10.1007/s10479-007-0176-2.

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

Conitzer, V. and T. Sandholm (2006). "Computing the optimal strategy to commit to." In: *Proceedings of the 7th ACM conference on Electronic commerce*, pp. 82–90. DOI: 10.1145/1134707.1134717.

Cook, W. J., R. Kannan, and A. Schrijver (1990). "Chvátal closures for mixed integer programming problems." In: *Mathematical Programming* 47.1, pp. 155–174. DOI: 10.1007/BF01580858.

Cormican, K. J., D. P. Morton, and R. K. Wood (1998). "Stochastic Network Interdiction." In: *Operations Research* 46.2, pp. 184–197. DOI: 10.1287/opre.46.2.184.

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

Della Croce, F. and R. Scatamacchia (2020). "An exact approach for the bilevel knapsack problem with interdiction constraints and extensions." In: *Mathematical Programming* 183, pp. 249–281. DOI: 10.1007/s10107-020-01482-5.

Dempe, S. (2002). *Foundations of Bilevel Programming*. Nonconvex Optimization and Its Applications. Springer New York, NY. DOI: 10.1007/b101970.

Dempe, S. (2020). "Bilevel Optimization: Theory, Algorithms, Applications and a Bibliography." In: *Bilevel Optimization: Advances and Next Challenges*. Ed. by S. Dempe and A. B. Zemkoho. Springer Optimization and Its Applications. Springer Cham, pp. 581–672. DOI: 10.1007/978-3-030-52119-6_20.

Dempe, S. and J. Dutta (2012). "Is bilevel programming a special case of a mathematical program with complementarity constraints?" In: *Mathematical Programming* 131.1-2, pp. 37–48. DOI: 10.1007/s10107-010-0342-1.

Dempe, S., F. Harder, P. Mehlitz, and G. Wachsmuth (2022). "Analysis and Solution Methods for Bilevel Optimal Control Problems." In: *Non-Smooth and Complementarity-Based Distributed Parameter Systems: Simulation and Hierarchical Optimization*. Ed. by M. Hintermüller, R. Herzog, C. Kanzow, M. Ulbrich, and S. Ulbrich. Vol. 172. International Series of Numerical Mathematics. Springer Cham, pp. 77–99. DOI: 10.1007/978-3-030-79393-7_4.

Dempe, S., V. Kalashnikov, G. A. Pérez-Valdés, and N. Kalashnykova (2015). *Bilevel Programming Problems: Theory, Algorithms and Applications to*

Energy Networks. Energy Systems. Springer. DOI: 10.1007/978-3-662-45827-3.

Dempe, S., B. S. Mordukhovich, and A. B. Zemkoho (2012). "Sensitivity Analysis for Two-Level Value Functions with Applications to Bilevel Programming." In: *SIAM Journal on Optimization* 22.4, pp. 1309–1343. DOI: 10.1137/110845197.

Dempe, S., B. S. Mordukhovich, and A. B. Zemkoho (2014). "Necessary optimality conditions in pessimistic bilevel programming." In: *Optimization* 63.4, pp. 505–533. DOI: 10.1080/02331934.2012.696641.

Dempe, S. and A. B. Zemkoho (2011). "The Generalized Mangasarian-Fromowitz Constraint Qualification and Optimality Conditions for Bilevel Programs." In: *Journal of Optimization Theory and Applications* 148.1, pp. 46–68. DOI: 10.1007/s10957-010-9744-8.

Dempe, S. and A. B. Zemkoho (2013). "The bilevel programming problem: reformulations, constraint qualifications and optimality conditions." In: *Mathematical Programming* 138.1, pp. 447–473. DOI: 10.1007/s10107-011-0508-5.

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

Dempe, S. and K. Richter (2000). "Bilevel programming with knapsack constraints." In: *Central European Journal of Operations Research* 8.2, p. 93.

DeNegre, S. T. (2011). "Interdiction and discrete bilevel linear programming." PhD thesis. Lehigh University. URL: http://coral.ie.lehigh.edu/~ted/files/papers/ScottDeNegreDissertation11.pdf.

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

DeNegre, S. T., T. K. Ralphs, S. Tahernejad, and Y. Xie (2026). Last accessed March 13, 2026. URL: https://github.com/coin-or/MibS.

Di Summa, M., A. Grosso, and M. Locatelli (2012). "Branch and cut algorithms for detecting critical nodes in undirected graphs." In: *Computational Optimization and Applications* 53.3, pp. 649–680. DOI: 10.1007/s10589-012-9458-y.

Edmonds, J. R. (1971). "Matroids and the greedy algorithm." In: *Mathematical Programming* 1, pp. 127–136. DOI: 10.1007/BF01584082.

Edmunds, T. A. and J. F. Bard (1991). "Algorithms for nonlinear bilevel mathematical programs." In: *IEEE Transactions on Systems, Man, and Cybernetics* 21.1, pp. 83–89. DOI: 10.1109/21.101139.

Eichfelder, G. (2010). "Multiobjective bilevel optimization." In: *Mathematical Programming* 123.2, pp. 419–449. DOI: 10.1007/s10107-008-0259-0.

Eichfelder, G. (2020). "Methods for Multiobjective Bilevel Optimization." In: *Bilevel Optimization: Advances and Next Challenges*. Ed. by S. Dempe and A. B. Zemkoho. Springer Optimization and Its Applications. Springer Cham, pp. 423–449. DOI: 10.1007/978-3-030-52119-6_15.

Facchinei, F. and C. Kanzow (2010). "Generalized Nash Equilibrium Problems." In: *Annals of Operations Research* 175.1, pp. 177–211. DOI: 10.1007/s10479-009-0653-x.

Fioretto, F., T. W. K. Mak, and P. Van Hentenryck (2019). "Privacy-Preserving Obfuscation of Critical Infrastructure Networks." In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, pp. 1086–1092. DOI: 10.24963/ijcai.2019/152.

Fischetti, M., I. Ljubić, M. Monaci, and M. Sinnl (2016). "Intersection Cuts for Bilevel Optimization." In: *International Conference on Integer Programming and Combinatorial Optimization*. Springer Cham, pp. 77–88. DOI: 10.1007/978-3-319-33461-5_7.

Fischetti, M., I. Ljubić, M. Monaci, and M. Sinnl (2017). "A New General-Purpose Algorithm for Mixed-Integer Bilevel Linear Programs." In: *Operations Research* 65.6, pp. 1615–1637. DOI: 10.1287/opre.2017.1650.

Fischetti, M., I. Ljubić, M. Monaci, and M. Sinnl (2018a). "On the use of intersection cuts for bilevel optimization." In: *Mathematical Programming* 172.1-2, pp. 77–103. DOI: 10.1007/s10107-017-1189-5.

Fischetti, M., I. Ljubić, M. Monaci, and M. Sinnl (2019). "Interdiction Games and Monotonicity, with Application to Knapsack Problems." In: *INFORMS Journal on Computing* 31.2, pp. 390–410. DOI: 10.1287/ijoc.2018.0831.

Fischetti, M., I. Ljubić, M. Monaci, and M. Sinnl (2026). *Bilevel Integer Programming and Interdiction Problems*. Last accessed March 13, 2026. URL: https://msinnl.github.io/pages/bilevel.html.

Fischetti, M., M. Monaci, and M. Sinnl (2018b). "A dynamic reformulation heuristic for Generalized Interdiction Problems." In: *European Journal of Operational Research* 267.1, pp. 40–51. DOI: 10.1016/j.ejor.2017.11.043.

Fortuny-Amat, J. and B. McCarl (1981). "A Representation and Economic Interpretation of a Two-Level Programming Problem." In: *Journal of the Operational Research Society* 32.9, pp. 783–792. DOI: 10.1057/jors.1981.156.

Franceschi, L., P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil (2018). "Bilevel Programming for Hyperparameter Optimization and Meta-Learning." In:

Proceedings of the 35th International Conference on Machine Learning. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 1568–1577. URL: https://proceedings.mlr.press/v80/franceschi18a.html.

Frecon, J., S. Salzo, and M. Pontil (2018). "Bilevel Learning of the Group Lasso Structure." In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2018/file/56bd37d3a2fda0f2f41925019c81011d-Paper.pdf.

Fulkerson, D. R. and G. C. Harding (1977). "Maximizing the minimum source-sink path subject to a budget constraint." In: *Mathematical Programming* 13.1, pp. 116–118. DOI: 10.1007/BF01584329.

Furini, F., I. Ljubić, E. Malaguti, and P. Paronuzzi (2022). "Casting Light on the Hidden Bilevel Combinatorial Structure of the Capacitated Vertex Separator Problem." In: *Operations Research* 70.4, pp. 2399–2420. DOI: 10.1287/opre.2021.2110.

Furini, F., I. Ljubić, P. San Segundo, and S. Martin (2019). "The maximum clique interdiction problem." In: *European Journal of Operational Research* 277.1, pp. 112–127. DOI: 10.1016/j.ejor.2019.02.028.

Gaar, E., J. Lee, I. Ljubić, M. Sinnl, and K. Tanınmış (2024). "On SOCP-based disjunctive cuts for solving a class of integer bilevel nonlinear programs." In: *Mathematical Programming* 206.1, pp. 91–124. DOI: 10.1007/S10107-023-01965-1.

Garey, M. R. and D. S. Johnson (1990). *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co.

Geißler, B., A. Morsi, L. Schewe, and M. Schmidt (2015). "Solving power-constrained gas transportation problems using an MIP-based alternating direction method." In: *Computers & Chemical Engineering* 82, pp. 303–317. DOI: 10.1016/j.compchemeng.2015.07.005.

Geißler, B., A. Morsi, L. Schewe, and M. Schmidt (2017). "Penalty Alternating Direction Methods for Mixed-Integer Optimization: A New View on Feasibility Pumps." In: *SIAM Journal on Optimization* 27.3, pp. 1611–1636. DOI: 10.1137/16M1069687.

Geißler, B., A. Morsi, L. Schewe, and M. Schmidt (2018). "Solving Highly Detailed Gas Transport MINLPs: Block Separability and Penalty Alternating Direction Methods." In: *INFORMS Journal on Computing* 30.2, pp. 309–323. DOI: 10.1287/ijoc.2017.0780.

Geoffrion, A. (1972). "Generalized Benders Decomposition." In: *Journal of Optimization Theory and Applications* 10, pp. 237–260. DOI: 10.1007/BF00934810.

Goerigk, M., J. Kurtz, M. Schmidt, and J. Thürauf (2025). "Connections between Robust and Bilevel Optimization." In: *Open Journal of Mathematical Optimization* 6.2, pp. 1–17. DOI: 10.5802/ojmo.38.

Golden, B. (1978). "A problem in network interdiction." In: *Naval Research Logistics Quarterly* 25.4, pp. 711–713. DOI: 10.1002/nav.3800250412.

Gomory, R. E. (1958). "Outline of an algorithm for integer solutions to linear programs." In: *Bulletin of the American Mathematical Society* 64, pp. 275–278.

Gomory, R. E. (1960). *An Algorithm for the Mixed Integer Problem*. Tech. rep. Research Memorandum RM-2597. The Rand Corporation.

Gorski, J., F. Pfeuffer, and K. Klamroth (2007). "Biconvex sets and optimization with biconvex functions: a survey and extensions." In: *Mathematical Methods of Operations Research* 66.3, pp. 373–407. DOI: 10.1007/s00186-007-0161-1.

Grimm, V., T. Kleinert, F. Liers, M. Schmidt, and G. Zöttl (2019a). "Optimal price zones of electricity markets: a mixed-integer multilevel model and global solution approaches." In: *Optimization Methods and Software* 34.2, pp. 406–436. DOI: 10.1080/10556788.2017.1401069.

Grimm, V., A. Martin, M. Schmidt, M. Weibelzahl, and G. Zöttl (2016). "Transmission and Generation Investment in Electricity Markets: The Effects of Market Splitting and Network Fee Regimes." In: *European Journal of Operational Research* 254.2, pp. 493–509. DOI: 10.1016/j.ejor.2016.03.044.

Grimm, V., L. Schewe, M. Schmidt, and G. Zöttl (2019b). "A multilevel model of the European entry-exit gas market." In: *Mathematical Methods of Operations Research* 89.2, pp. 223–255. DOI: 10.1007/s00186-018-0647-z.

Grüne, C. and L. Wulf (2025). "Completeness in the Polynomial Hierarchy for Many Natural Problems in Bilevel and Robust Optimization." In: *Integer Programming and Combinatorial Optimization*. Ed. by N. Megow and A. Basu. Springer Cham, pp. 256–269. DOI: 10.1007/978-3-031-93112-3_19.

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

Henke, D., H. Lefebvre, M. Schmidt, and J. Thürauf (2025a). "On coupling constraints in linear bilevel optimization." In: *Optimization Letters* 19, pp. 689–697. DOI: 10.1007/s11590-024-02156-3.

Henke, D., H. Lefebvre, M. Schmidt, and J. Thürauf (2025b). *On Coupling Constraints in Pessimistic Linear Bilevel Optimization*. URL: https://arxiv.org/abs/2503.01563.

Hooker, J. (2024). *Logic-Based Benders Decomposition: Theory and Applications*. Synthesis Lectures on Operations Research and Applications. Springer Cham. DOI: 10.1007/978-3-031-45039-6.

Israeli, E. and R. K. Wood (2002). "Shortest-path network interdiction." In: *Networks* 40.2, pp. 97–111. DOI: 10.1002/net.10039.

Jeroslow, R. G. (1985). "The polynomial hierarchy and a simple model for competitive analysis." In: *Mathematical Programming* 32.2, pp. 146–164. DOI: 10.1007/BF01586088.

Kall, P. and S. W. Wallace (1994). *Stochastic Programming*. Wiley-Interscience Series in Systems and Optimization. Wiley.

Kleinert, T. (2021). "Algorithms for Mixed-Integer Bilevel Problems with Convex Followers." PhD thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg. URL: https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/docId/383.

Kleinert, T., M. Labbé, I. Ljubić, and M. Schmidt (2021a). "A Survey on Mixed-Integer Programming Techniques in Bilevel Optimization." In: *EURO Journal on Computational Optimization* 9, p. 100007. DOI: 10.1016/j.ejco.2021.100007.

Kleinert, T., M. Labbé, F. Plein, and M. Schmidt (2020). "There's No Free Lunch: On the Hardness of Choosing a Correct Big-M in Bilevel Optimization." In: *Operations Research* 68.6, pp. 1716–1721. DOI: 10.1287/opre.2019.1944.

Kleinert, T., M. Labbé, F. Plein, and M. Schmidt (2021b). "Closing the Gap in Linear Bilevel Optimization: A New Valid Primal-Dual Inequality." In: *Optimization Letters* 15, pp. 1027–1040. DOI: 10.1007/s11590-020-01660-6.

Kleinert, T., J. Manns, M. Schmidt, and D. Weninger (2021c). "Presolving linear bilevel optimization problems." In: *EURO Journal on Computational Optimization* 9, p. 100020. DOI: 10.1016/j.ejco.2021.100020.

Kleinert, T. and M. Schmidt (2019). "Global Optimization of Multilevel Electricity Market Models Including Network Design and Graph Partitioning." In: *Discrete Optimization* 33, pp. 43–69. DOI: 10.1016/j.disopt.2019.02.002.

Kleinert, T. and M. Schmidt (2021). "Computing Feasible Points of Bilevel Problems with a Penalty Alternating Direction Method." In: *INFORMS Journal on Computing* 33.1, pp. 198–215. DOI: 10.1287/ijoc.2019.0945.

Kleinert, T. and M. Schmidt (2023). "Why there is no need to use a big-M in linear bilevel optimization: A computational study of two ready-to-use approaches."

In: *Computational Management Science* 20.1, p. 3. DOI: 10.1007/s10287-023-00435-5.

Kolstad, C. D. (1985). *A review of the literature on bi-level mathematical programming*. Tech. rep. Los Alamos National Laboratory Los Alamos, NM. URL: https://fas.org/sgp/othergov/doe/lanl/lib-www/la-pubs/00318870.pdf.

Köppe, M., M. Queyranne, and C. T. Ryan (2010). "Parametric Integer Programming Algorithm for Bilevel Mixed Integer Programs." In: *Journal of Optimization Theory and Applications* 146.1, pp. 137–150. DOI: 10.1007/s10957-010-9668-3.

Labbé, M., P. Marcotte, and G. Savard (1998). "A bilevel model of taxation and its application to optimal highway pricing." In: *Management Science* 44.12-part-1, pp. 1608–1622. DOI: 10.1287/mnsc.44.12.1608.

Labbé, M. and A. Violin (2013). "Bilevel programming and price setting problems." In: *4OR* 11.1, pp. 1–30. DOI: 10.1007/s10288-012-0213-0.

Lalou, M., M. A. Tahraoui, and H. Kheddouci (2018). "The Critical Node Detection Problem in networks: A survey." In: *Computer Science Review* 28, pp. 92–117. DOI: 10.1016/j.cosrev.2018.02.002.

Land, A. H. and A. G. Doig (1960). "An Automatic Method of Solving Discrete Programming Problems." In: *Econometrica* 28.3, pp. 497–520. URL: http://www.jstor.org/stable/1910129.

Lefebvre, H. and M. Schmidt (2026). "Computing Counterfactual Explanations for Linear Optimization: A New Class of Bilevel Models and a Tailored Penalty Alternating Direction Method." In: *European Journal of Operational Research*. DOI: 10.1016/j.ejor.2026.02.007.

Leitner, M., I. Ljubić, M. Monaci, M. Sinnl, and K. Tanınmış (2023). "An exact method for binary fortification games." In: *European Journal of Operational Research* 307.3, pp. 1026–1039. DOI: 10.1016/j.ejor.2022.10.038.

Lim, C. and J. C. Smith (2007). "Algorithms for discrete and continuous multicommodity flow network interdiction problems." In: *IIE Transactions* 39.1, pp. 15–26. DOI: 10.1080/07408170600729192.

Luo, Z.-Q., J.-S. Pang, and D. Ralph (1996). *Mathematical programs with equilibrium constraints*. Cambridge University Press. DOI: 10.1017/CBO9780511983658.

Macal, C. M. and A. P. J. Hurter (1997). "Dependence of bilevel mathematical programs on irrelevant constraints." In: *Computers & Operations Research* 24.12, pp. 1129–1140. DOI: 10.1016/S0305-0548(97)00025-7.

Magnanti, T. L. and L. A. Wolsey (1995). "Optimal trees." In: *Network Models*. Vol. 7. Handbooks in Operations Research and Management Science. Elsevier, pp. 503–615. DOI: 10.1016/S0927-0507(05)80126-4.

Magnouche, Y. and S. Martin (2020). “Most vital vertices for the shortest *s-t* path problem: complexity and Branch-and-Cut algorithm.” In: *Optimization Letters* 14, pp. 2039–2053. DOI: 10.1007/s11590-019-01527-5.

Mallozzi, L. and J. Morgan (1996). “Hierarchical Systems with Weighted Reaction Set.” In: *Nonlinear Optimization and Applications*. Ed. by G. Di Pillo and F. Giannessi. Springer, Boston, MA, pp. 271–282. DOI: 10.1007/978-1-4899-0289-4_19.

Manns, J. (2020). “Presolve of Linear Bilevel Programs.” MA thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg. URL: https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/docId/320.

Marcotte, P. and G. Savard (2005). “Bilevel Programming: A Combinatorial Perspective.” In: *Graph Theory and Combinatorial Optimization*. Ed. by D. Avis, A. Hertz, and O. Marcotte. Springer US, pp. 191–217. DOI: 10.1007/0-387-25592-3_7.

Mas-Colell, A., M. D. Whinston, and J. R. Green (1995). *Microeconomic Theory*. Oxford University Press.

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

Mehlitz, P. and G. Wachsmuth (2016). “Weak and strong stationarity in generalized bilevel programming and bilevel optimal control.” In: *Optimization* 65.5, pp. 907–935. DOI: 10.1080/02331934.2015.1122007.

Mehlitz, P. and G. Wachsmuth (2020). “Bilevel Optimal Control: Existence Results and Stationarity Conditions.” In: *Bilevel Optimization: Advances and Next Challenges*. Ed. by S. Dempe and A. B. Zemkoho. Springer Optimization and Its Applications. Springer Cham, pp. 451–484. DOI: 10.1007/978-3-030-52119-6_16.

Mersha, A. G. and S. Dempe (2006). “Linear bilevel programming with upper level constraints depending on the lower level solution.” In: *Applied Mathematics and Computation* 180.1, pp. 247–254. DOI: 10.1016/j.amc.2005.11.134.

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

Nash, J. F. (1950). “Equilibrium points in n-person games.” In: *Proceedings of the National Academy of Sciences* 36.1, pp. 48–49. DOI: 10.1073/pnas.36.1.48.

Nemhauser, G. L. and L. A. Wolsey (1990). “A Recursive Procedure to Generate All Cuts for 0–1 Mixed Integer Programs.” In: *Mathematical Programming* 46.1, pp. 379–390. DOI: 10.1007/BF01585752.

Nocedal, J. and S. J. Wright (2006). *Numerical Optimization*. Springer New York, NY. DOI: 10.1007/978-0-387-40065-5.

Outrata, J. V. (1990). "On the numerical solution of a class of Stackelberg problems." In: *ZOR - Mathematical Methods of Operations Research* 34.4, pp. 255–277. DOI: 10.1007/BF01416737.

Pajouh, F. M. (2020). "Minimum cost edge blocker clique problem." In: *Annals of Operations Research* 294.1, pp. 345–376. DOI: 10.1007/s10479-019-03315-x.

Pajouh, F. M., V. Boginski, and E. L. Pasiliao (2014). "Minimum vertex blocker clique problem." In: *Networks* 64.1, pp. 48–64. DOI: 10.1002/net.21556.

Pajouh, F. M., J. L. Walteros, V. Boginski, and E. L. Pasiliao (2015). "Minimum edge blocker dominating set problem." In: *European Journal of Operational Research* 247.1, pp. 16–26. DOI: 10.1016/j.ejor.2015.05.037.

Pineda, S., H. C. Bylling, and J. M. Morales (2018). "Efficiently solving linear bilevel programming problems using off-the-shelf optimization software." In: *Optimization and Engineering* 19.1, pp. 187–211. DOI: 10.1007/s11081-017-9369-y.

Pineda, S. and J. M. Morales (2019). "Solving Linear Bilevel Problems Using Big-Ms: Not All That Glitters Is Gold." In: *IEEE Transactions on Power Systems* 34.3, pp. 2469–2471. DOI: 10.1109/TPWRS.2019.2892607.

Pita, J., C. Portway, M. Tambe, P. Paruchuri, and S. Kraus (2008). "Robust Solutions in Stackelberg Games: Addressing Boundedly Rational Human Preference Models." In: *AAAI 2008—4th Multidisciplinary Workshop on Advances in Preference Handling*. URL: https://www.aaai.org/Papers/Workshops/2008/WS-08-09/WS08-09-010.pdf.

Rahmaniani, R., T. G. Crainic, M. Gendreau, and W. Rei (2017). "The Benders decomposition algorithm: A literature review." In: *European Journal of Operational Research* 259.3, pp. 801–817. DOI: 10.1016/j.ejor.2016.12.005.

Rodrigues, B. (2025). "Linear Bilevel Optimization and Applications in the Grid Integration of Energy Storage." PhD thesis. University of Edinburgh. URL: https://era.ed.ac.uk/bitstream/handle/1842/43705/Rodrigues2025.pdf.

Rodrigues, B., M. Carvalho, M. F. Anjos, and N. Sugishita (2025). "Unboundedness in bilevel optimization." In: *Les Cahiers du GERAD* G-2025-22, pp. 1–26. URL: https://www.gerad.ca/fr/papers/G-2025-22.pdf.

Rosen, J. B. (1965). "Existence and Uniqueness of Equilibrium Points for Concave N-Person Games." In: *Econometrica* 33.3, pp. 520–534. URL: https://www.jstor.org/stable/1911749?seq=1.

Rutenburg, V. (1994). “Propositional truth maintenance systems: Classification and complexity analysis.” In: *Annals of Mathematics and Artificial Intelligence* 10.3, pp. 207–231. DOI: 10.1007/BF01530952.

Salas, D. and A. Svensson (2023). “Existence of Solutions for Deterministic Bilevel Games under a General Bayesian Approach.” In: *SIAM Journal on Optimization* 33.3, pp. 2311–2340. DOI: 10.1137/21M1442164.

Salas, D. (2025). *Introduction to Bilevel Optimization: A perspective from Variational Analysis*. URL: https://arxiv.org/abs/2511.05793.

Samuelson, P. A. (1952). “Spatial Price Equilibrium and Linear Programming.” In: *The American Economic Review* 42.3, pp. 283–303. URL: http://www.jstor.org/stable/1810381.

Savard, G. and J. Gauvin (1994). “The steepest descent direction for the nonlinear bilevel programming problem.” In: *Operations Research Letters* 15.5, pp. 265–272. DOI: 10.1016/0167-6377(94)90086-8.

Scaparra, M. P. and R. L. Church (2008). “A bilevel mixed-integer program for critical infrastructure protection planning.” In: *Computers & Operations Research* 35.6, pp. 1905–1923. DOI: 10.1016/j.cor.2006.09.019.

Schewe, L., M. Schmidt, and J. Thürauf (2022). “Global Optimization for the Multilevel European Gas Market System with Nonlinear Flow Models on Trees.” In: *Journal of Global Optimization* 82.3, pp. 627–653. DOI: 10.1007/s10898-021-01099-8.

Schrijver, A. (1998). *Theory of linear and integer programming*. John Wiley & Sons.

Shen, S. and J. C. Smith (2012). “Polynomial-time algorithms for solving a class of critical node problems on trees and series-parallel graphs.” In: *Networks* 60.2, pp. 103–119. DOI: 10.1002/net.20464.

Shen, S., J. C. Smith, and R. Goli (2012). “Exact interdiction models and algorithms for disconnecting networks via node deletions.” In: *Discrete Optimization* 9.3, pp. 172–188. DOI: 10.1016/j.disopt.2012.07.001.

Shi, X., O. A. Prokopyev, and T. K. Ralphs (2023). “Mixed integer bilevel optimization with a k-optimal follower: a hierarchy of bounds.” In: *Mathematical Programming Computation* 15.1, pp. 1–51. DOI: 10.1007/s12532-022-00227-z.

Simon, H. (1972). “Theories of Bounded Rationality.” In: *McGuire, C. B. and Radner, R., Eds., Decision and Organization*. North Holland Publishing Company, pp. 161–176. URL: http://innovbfa.viabloga.com/files/Herbert_Simon___theories_of_bounded_rationality___1972.pdf.

Smith, J. C. and C. Lim (2008). “Algorithms for Network Interdiction and Fortification Games.” In: *Pareto Optimality, Game Theory And Equilibria*.

Ed. by A. Chinchuluun, P. M. Pardalos, A. Migdalas, and L. Pitsoulis. Springer New York, pp. 609–644. DOI: 10.1007/978-0-387-77247-9_24.

Smith, J. C. and Y. Song (2020). "A survey of network interdiction models and algorithms." In: *European Journal of Operational Research* 283.3, pp. 797–811. DOI: 10.1016/j.ejor.2019.06.024.

Soyster, A. L. (1973). "Technical Note—Convex Programming with Set-Inclusive Constraints and Applications to Inexact Linear Programming." In: *Operations Research* 21.5, pp. 1154–1157. DOI: 10.1287/opre.21.5.1154.

Stein, O. (2025). "A tutorial on properties of the epigraph reformulation." In: *EURO Journal on Computational Optimization* 13, p. 100109. DOI: 10.1016/j.ejco.2025.100109.

Still, G. (2018). *Lectures on parametric optimization: An introduction*. URL: https://optimization-online.org/wp-content/uploads/2018/04/6587.pdf.

Svensson, A. (2024). "Lower Semicontinuity of Intersections of Set-Valued Maps and Applications on Bilevel Games." In: *Set-Valued and Variational Analysis* 32.3. Article number 24. DOI: 10.1007/s11228-024-00728-0.

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

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

Tambe, M. (2011). *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press. DOI: 10.1017/CBO9780511973031.

Thürauf, J., T. Kleinert, I. Ljubić, T. Ralphs, and M. Schmidt (2026). "BOBILib: Bilevel Optimization (Benchmark) Instance Library." In: *Mathematical Programming Computation*. DOI: 10.1007/s12532-025-00294-y.

van Damme, E. and S. Hurkens (1997). "Games with Imperfectly Observable Commitment." In: *Games and Economic Behavior* 21.1–2, pp. 282–308. DOI: 10.1006/game.1997.0524.

Vicente, L. and P. H. Calamai (1994). "Bilevel and multilevel programming: A bibliography review." In: *Journal of Global Optimization* 5.3, pp. 291–306. DOI: 10.1007/BF01096458.

Vicente, L., G. Savard, and J. Júdice (1994). "Descent approaches for quadratic bilevel programming." In: *Journal of Optimization Theory and Applications* 81.2, pp. 379–399. DOI: 10.1007/BF02191670.

Vicente, L., G. Savard, and J. Júdice (1996). "Discrete Linear Bilevel Programming Problem." In: *Journal of Optimization Theory and Applications* 89.3, pp. 597–614. DOI: 10.1007/BF02275351.

von Neumann, J. and O. Morgenstern (1947). *Theory of games and economic behavior*. Princeton University Press.

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

von Stackelberg, H. (1952). *Theory of the market economy*. Oxford University Press.

Wang, L. and P. Xu (2017). "The watermelon algorithm for the bilevel integer linear programming problem." In: *SIAM Journal on Optimization* 27.3, pp. 1403–1430. DOI: 10.1137/15M1051592.

Wei, N. and J. L. Walteros (2022). "Integer programming methods for solving binary interdiction games." In: *European Journal of Operational Research* 302.2, pp. 456–469. DOI: 10.1016/j.ejor.2022.01.009.

Wen, U.-P. and S.-T. Hsu (1991). "Linear Bi-Level Programming Problems — A Review." In: *The Journal of the Operational Research Society* 42.2, pp. 125–133. URL: http://www.jstor.org/stable/2583177.

Wendell, R. E. and A. P. J. Hurter (1976). "Minimization of a non-separable objective function subject to disjoint constraints." In: *Operations Research* 24.4, pp. 643–657. DOI: 10.1287/opre.24.4.643.

Weninger, N. and R. Fukasawa (2025). "A fast combinatorial algorithm for the bilevel knapsack problem with interdiction constraints." In: *Mathematical Programming* 210.1, pp. 847–879. DOI: 10.1007/s10107-024-02133-9.

Wiesemann, W., A. Tsoukalas, P.-M. Kleniati, and B. Rustem (2013). "Pessimistic Bilevel Optimization." In: *SIAM Journal on Optimization* 23.1, pp. 353–380. DOI: 10.1137/120864015.

Williams, A. C. (1970). "Boundedness relations for linear constraint sets." In: *Linear Algebra and its Applications* 3.2, pp. 129–141. DOI: 10.1016/0024-3795(70)90009-1.

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

Wolsey, L. A. (2020). *Integer Programming*. John Wiley & Sons, Inc. DOI: 10.1002/9781119606475.

Wood, R. K. (1993). "Deterministic network interdiction." In: *Mathematical and Computer Modelling* 17.2, pp. 1–18. DOI: 10.1016/0895-7177(93)90236-R.

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

Xu, P. and L. Wang (2014). "An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions." In: *Computers & Operations Research* 41, pp. 309–318. DOI: 10.1016/j.cor.2013.07.016.

Ye, J. J. and D. L. Zhu (1995). "Optimality conditions for bilevel programming problems." In: *Optimization* 33.1, pp. 9–27. DOI: 10.1080/02331939508844060.

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

Zeng, B. (2020). "A Practical Scheme to Compute the Pessimistic Bilevel Optimization Problem." In: *INFORMS Journal on Computing* 32.4, pp. 1128–1142. DOI: 10.1287/ijoc.2019.0927.

Zenklusen, R. (2010). "Matching interdiction." In: *Discrete Applied Mathematics* 158.15, pp. 1676–1690. DOI: 10.1016/j.dam.2010.06.006.

Zenklusen, R., B. Ries, C. Picouleau, D. de Werra, M.-C. Costa, and C. Bentz (2009). "Blockers and transversals." In: *Discrete Mathematics* 309.13, pp. 4306–4314. DOI: 10.1016/j.disc.2009.01.006.

Index

3-SAT, 81
3-CNF, 81
3-clause, 81

Abadie constraint qualification, 46, 298, 319
ACQ, *see* Abadie constraint qualification
Active inequality constraints, 319
ADM, *see* Alternating direction method
Alternating direction method, 104, 106
Apex, 227, 250
Attainability, *see* Existence of solutions

Backtracking, 136, 147, 176
Basis, 226
Benders
  -like problem reformulation, 205
  Cuts, 204
  Decomposition, 203
    Multi-tree implementation, 208
    Single-tree implementation, 209
Best-reply function, 291
BFS, *see* Breadth-first search
Big-M reformulation, 50, 51, 130, 300
Bilevel knapsack problem, 287
Bilevel optimization problem
  Bilinear, 21, 187, 286
  Discrete, 16
  Linear, 18
  Mixed-integer linear, 117
  Under uncertainty, 275
Bilevel-feasible
  Point, 18
  Set, 18
Bilevel-free
  Hypercubes, 246
Bilevel-free set, 239, 240
  Enlarged, 242
  Improving direction, 243, 245
  Improving solution, 241
Bilinear bilevel problem, 21, 187, 286
Blocker problem, 185, 219
Bound strengthening, 63
Bounded rationality, 286
Boundedness, 314
Bounding lemma, 94, 137
Branch-and-bound, 31, 84, 89, 92, 134, 138, 148
Branch-and-cut, 31, 134, 161, 209
Branching
  Lemma, 95, 137
  Rules, 93, 137
Breadth-first search, 93, 96, 136, 172
Budgeted uncertainty set, 287

Capacity reduction constraints, 186
Clique, 192
  Maximal, 192
  Number, 192
Complementarity slackness, 314
Complexity, 30, 80, 119, 128, 185, 219
Cone
  Linearized tangential, 319
  LP, 226, 250
  Polyhedral, 239, 325
    Pointed, 227
  Recession, 230, 325
  Tangential, 319
Conjunctive normal form, 81
Constraint qualification, 43, 45, 46, 55, 66, 295, 298, 299, 308, 319–321
Continuous counterpart, 144, 268

Conventional formulation, 19
Convex
  Analysis, 308
  Hull, 119
  Optimization, 320
Coupling constraints, 17, 35, 36, 55, 63, 71, 72, 75, 76, 80, 108, 125, 126, 131, 138, 152, 161, 173, 184, 196, 204, 254, 262–264, 266, 276
Critical
  Edge detection, 223
  Infrastructure defense, 15, 307
  Vertex detection, 223
Cut, 161, 164
  Interdiction, 205
  Intersection, 166, 225, 228, 230
    Hypercube, 247
  No-good, 170
  Valid
    Globally, 162
    Locally, 177
Cutting plane, 226, *see* Cut

Data uncertainty, 276, 277
Decision uncertainty, 275, 276, 284
Defender-attacker problem, 183
Defender-attacker-defender problem, 309
Depth-first search, 93, 96, 136, 146, 153, 156, 172, 174, 180
Deterministic equivalent, 283, 287
DFS, *see* Depth-first search
Discrete bilevel problem, 16
Disjointly constrained bilinear program, 198
Disjunction, 49, 135, 151, 233, 242
  Split, 229
Downward monotonicity, 185, 212, 213, 271
Drug smuggling, 16, 24
Dual problem, 314
Duality theorem, 313, 314
  Strong, 314
  Weak, 314
Dualization, 196, 269

Electricity market modeling, 27
Energy markets, 14, 289, 294, 307
Enlarged bilevel-free set, 242
Epigraph reformulation, 37, 205, 278, 317
Equilibrium, 290, 292
Existence of solutions, 20, 75, 76, 119, 124, 127
Expected value, 282
Extremal set, 80
Extreme
  Point, 80, 86, 205, 325
  Ray, 325

Fathoming rules, *see* Pruning rules
Feasible point, 313
  Strictly, 321
Floor function, 135
Follower, 14
Follower's
  Decision, 17
  Feasible set mapping, 17
  Optimal response mapping, 17
  Problem, 17
Follower-priority heuristic, 262
Fortification problem, 309
Fourier–Motzkin elimination, 54, 75

Game, 290
  N-person, 290
  Leader-follower, 30
  Strategic, 290
  Theory, 289, 290
Gap, 92
Generalized interdiction problem, 266
Generalized Nash equilibrium, 292, 296
  Problem, 291, 293–295
Globally valid cut, 162
GNEP, *see* Generalized Nash equilibrium problems
Graph of the solution-set mapping, 18

Half-line, 230, 325
Hardness, 30, 80, 82, 84, 104, 128, 185
Here-and-now
  Decision, 278, 279, 286
  Follower, 278
Heuristic, 104
  Follower-priority, 262
  Penalize-and-relax-and-dualize, 262, 265, 269
  Primal, 84, 262
Hierarchical optimization problem, 13, 16
High-point relaxation, 20
HPR, *see* High-point relaxation
Hypercube intersection cut, 247
Hyperparameter tuning, 307

IIC, *see* Independence of irrelevant constraints
Improving direction, 74

Bilevel-free set, 245
Intersection cut, 245
Incumbent, 91, 135, 153, 170, 172, 176
Independence of irrelevant constraints, 63, 65, 66, 68
Inducible region, 18
Inner node, 92
Interdiction
Constraints, 186
Cut, 205
Problem, 16, 23, 24, 29, 158, 166, 180, 183, 266, 271
Generalized, 266
Knapsack, 23, 29, 158, 180, 191, 271
Maximum-clique, 192
Maximum-flow, 16, 24
Network, 183
Shortest-path, 187, 188, 190
Continuous, 188
Discrete, 190
Intersection cut, 166, 225, 228, 230
Hypercube, 247
Improving direction, 245

Kth-best algorithm, 30, 84, 86
`KERNEL`, 80
k-most-vital arcs problem, 190
Karush–Kuhn–Tucker
Complementarity conditions, 46, 49
Conditions, 30, 43, 75, 129, 295, 297, 302, 318, 319
Point, 298, 319
Reformulation, 43, 45, 55, 80, 89, 94, 97, 130, 300
Theorem
Under ACQ, 319
Under LICQ, 320
Under Slater, 321
KKT, *see* Karush–Kuhn–Tucker
Knapsack, 287
Interdiction, 23, 29, 158, 180, 191, 271

ℓ_1-penalization, 316
Lagrange multipliers, 319
Lagrangian function, 45, 56, 67, 97, 318
Leader, 14
Decision, 17
Problem, 17
Leader-follower game, 30
LICQ, *see* Linear independence constraint qualification
Limited observability, 285, 287, 288
Linear bilevel problem, 18
Linear independence constraint qualification, 46, 298, 320
Linear optimization, 313
Problem in standard form, 313
Linearized tangential cone, 319
Linking
Constraints, 17, 199
Variables, 17, 121, 125, 126, 151, 172, 173, 185
Literal, 81
Locally valid cut, 177
Lower bound, 122
Lower-level problem, 17

Machine learning, 307
Mangasarian–Fromowitz constraint qualification, 46
Mathematical program with complementarity constraints, 46, 89, 313, 322, 323
Maximal follower's solution, 217
Maximum-clique interdiction problem, 192
Maximum-flow interdiction problem, 16, 24
McCormick inequalities, 47, 48, 197
MFCQ, *see* Mangasarian–Fromowitz constraint qualification
MILP reformulation, 50
Min-max problem, 23, 184, 282
Minimizer
Global, 18, 318
Local, 18, 318
Partial, 107
Strict global, 318
Strict local, 318
Minimum-cost vertex blocker clique problem, 219
Minkowski sum, 326
Mixed-integer linear bilevel problem, 117
MLMF, *see* Multi-leader multi-follower problem
MLSF, *see* Multi-leader single-follower problem
Most vital
Arcs, 221
Vertices, 221
MPCC, *see* Mathematical program with complementarity constraints
Multi-leader multi-follower problem, 289, 295
Multi-leader single-follower problem, 289, 294
Multi-objective optimization, 308

Multilevel optimization problem, 28, 130, 309

Nash
  Equilibrium, 289, 290, 293, 294
    Generalized, 292
  Game, 289, 290
Near-optimality, 285
Network interdiction, 183
No-good cut, 170
Node
  Problem, 91
  Solution, 91
Node-selection strategies, 93, 136, 172
Nonlinear optimization, 317

Optimal-value function, 19, 38, 43, 44, 77, 120, 138, 139
  Reformulation, 19, 20, 30, 43, 44, 266
Optimality
  Conditions, 308, 313
  Gap, 92
Optimistic
  Approach, 17
  Bilevel optimization, 17, 19, 33, 39, 46, 184, 277, 293
  Solution, 18, 35
Original optimistic formulation, 19

PADM, *see* Penalty alternating direction method
Parametric optimization, 308
Penalization, 199, 267
Penalize-and-relax-and-dualize heuristic, 262, 265, 269
Penalty
  Function, 108
  Method, 104
  Parameter, 108
  Reformulation, 201
Penalty alternating direction method, 84, 104, 108, 109
Pessimistic
  Approach, 18
  Bilevel optimization, 33, 35, 39, 184, 277, 293
    With coupling constraints, 36
    Without coupling constraints, 35
  Solution, 35
Player, 21
Polyhedral
  Cone, 239, 325
  Point-to-set mapping, 75
Polyhedron, 325
Polytope, 325
Power set, 75
Pricing, 13, 21, 307
Primal heuristic, 84, 262
Probability distribution, 281
Pruning rules, 94

Quasi block-separable problem, 108

Rational reaction set, 17, 35
Ray, 325
  Extreme, 325
Recession
  Cone, 230, 325
  Direction, 325
Refinement problem, 126, 150, 151, 157, 172, 173, 176, 247, 270
Reformulation
  Benders-like, 205
  Big-M, 50, 51, 130, 300
  Epigraph, 37, 205, 278, 317
  KKT, 43, 45, 55, 80, 89, 94, 97, 130, 300
  MILP, 50
  Optimal-value function, 19, 20, 30, 43, 44, 266
  Single-level, 43, 80, 129, 295, 299, 300
  Strong-duality, 43, 46, 269, 271
Relaxation, 91, 93, 139, 145, 268
  High-point, 20
  Single-level, 20, 29, 30, 118, 119, 140, 149
Relaxed master problem, 206
Restriction, 139, 140
Revenue management, 307
RMP, *see* Relaxed master problem
Robust optimization, 276, 277, 280, 285
Root-node problem, 91

Σ_2^p-hardness, 129, 185, 191, 195, 262
Scoops, 245
Sensitivity analysis, 308
Separation problem, 164
Sequential bimatrix game, 286
Set covering, 220
  Constraints, 220
Set-valued map, 17, 19
Shared constraint set, 18, 72, 120, 139, 264, 297
  Continuous relaxation, 121

Projection onto the x-space, 18
Shortest path, 16
Interdiction, 187, 188, 190
Problem, 22
Most vital arcs, 221
Most vital vertices, 221
Simplex
Method, 85, 86, 226
Tableau, 226
Simultaneous-move game, 289
Single-leader multi-disjoint-follower problem, 23, 293
Single-leader multi-follower problem, 14, 289, 292, 308
Single-leader single-follower problem, 14
Single-level reformulation, 43, 80, 129, 295, 299, 300
Single-level relaxation, 20, 27, 29, 30, 63, 66, 70, 71, 85, 94, 111, 118, 119, 121, 140, 149, 263
Continuous relaxation, 121
Slater's constraint qualification, 45, 55, 66, 299, 321
SLMDF, *see* Single-leader multi-disjoint-follower problem
SLMF, *see* Single-leader multi-follower problem
SLSF, *see* Single-leader single-follower problem
Solution, *see* Minimizer
Solution-set mapping, 291
Graph of the, 18
SOS1, *see* Special ordered set of type 1
Special ordered set of type 1, 95, 295, 300
Conditions, 51
Split disjunction, 229
Stability theory, 308
Stackelberg security game, 308
Stochastic optimization, 276, 281
Strong duality, 47, 269
Error, 112
Inequality, 105, 110
Reformulation, 43, 46, 269, 271
Theorem, 43, 46, 76, 77, 105
Sustainability, 307

Tangential cone, 319
Toll setting, 14, 22, 293
Trilevel problem, 309

Unboundedness, 72, 123
Uncertain bilevel problem, 275
Uncertainty set, 277
Budgeted, 287
Upper-level problem, 17

Valid inequality, 48, 162, 225, 226
Globally valid, 162
Locally valid, 163
Variable disjunction, 135, 151
Variational
Analysis, 308
Equilibria, 298

Wait-and-see follower, 279
Weak duality, 47, 110
Weierstraß theorem, 122, 135, 137, 264, 265, 318
Weighted
ℓ_1-feasibility measure, 109
ℓ_1-penalty function, 108