Chapter 18
A Unified Framework for Multistage Mixed Integer Linear Optimization

Suresh Bolusani, Stefano Coniglio, Ted K. Ralphs, and Sahar Tahernejad

Abstract We introduce a unified framework for the study of multilevel mixed integer linear optimization problems and multistage stochastic mixed integer linear optimization problems with recourse. The framework highlights the common mathematical structure of the two problems and allows for the development of a common algorithmic framework. Focusing on the two-stage case, we investigate, in particular, the nature of the value function of the second-stage problem, highlighting its connection to dual functions and the theory of duality for mixed integer linear optimization problems, and summarize different reformulations. We then present two main solution techniques, one based on a Benders-like decomposition to approximate either the risk function or the value function, and the other one based on cutting plane generation.

Keywords Multilevel optimization · Multistage stochastic optimization · Discrete optimization · Primal and dual functions · Decomposition methods · Convexification-based methods

18.1 Introduction

This article introduces a unified framework for the study of *multilevel mixed integer linear optimization problems* and *multistage stochastic mixed integer linear optimization problems with recourse*. This unified framework provides insights into

S. Bolusani · T. K. Ralphs
Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA
e-mail: bsuresh@lehigh.edu; ted@lehigh.edu

S. Coniglio (✉)
Department of Mathematical Sciences, University of Southampton, Southampton, UK
e-mail: s.coniglio@soton.ac.uk

S. Tahernejad
Lindo Systems, Inc., Chicago, IL, USA
e-mail: sahar@lindo.com

S. Dempe, A. Zemkoho (eds.), *Bilevel Optimization*, Springer Optimization and Its Applications 161, https://doi.org/10.1007/978-3-030-52119-6_18

the nature of these two well-known classes of optimization problems, highlights their common mathematical structure, and allows results from the wider literature devoted to both classes to be exploited for the development of a common algorithmic framework.

18.1.1 Motivation

Historically, research in mathematical optimization, which is arguably the most widely applied theoretical and methodological framework for solving optimization problems, has been primarily focused on "idealized" models aimed at informing the decision process of a single *decision maker* (DM) facing the problem of making a single set of decisions at a single point in time under perfect information. Techniques for this idealized case are now well developed, with efficient implementations widely available in off-the-shelf software.

In contrast, most real-world applications involve multiple DMs, and decisions must be made at multiple points in time under uncertainty. To allow for this additional complexity, a number of more sophisticated modeling frameworks have been developed, including multistage and multilevel optimization. In line with the recent optimization literature, we use the term *multistage optimization* to denote the decision process of a single DM over multiple time periods with an objective that factors in the (expected) impact at future stages of the decisions taken at the current stage. With the term *multilevel optimization*, on the other hand, we refer to game-theoretic decision processes in which multiple DMs with selfish objectives make decisions in turn, competing to optimize their own individual outcomes in the context of settings such as, e.g., economic markets.

Because the distinction between multistage and multilevel optimization problems appears substantial from a modeling perspective, their development has been undertaken independently by different research communities. Indeed, multistage problems have arisen out of the necessity to account for stochasticity, which is done by explicitly including multiple decision stages in between each of which the value of a random variable is realized. Knowledge of the precise values of the quantities that were unknown at earlier stages allows for so-called *recourse decisions* to be made in later stages in order to correct possible mis-steps taken due to the lack of precise information. On the other hand, multilevel optimization has been developed primarily to model multi-round games (technically known as *finite, extensive form games with perfect information* in the general case and *Stackelberg games* in the case of two rounds) in which the decision (or strategy) of a given player at a given round must take into account the reactions of other players in future rounds.

Despite these distinctions, these classes of problems share an important common structure from a mathematical and methodological perspective that makes considering them in a single, unifying framework attractive. It is not difficult to understand the source of this common structure—from the standpoint of an individual DM, the complexity of the decision process comes from uncertainty about the future.

From an analytical perspective, the methods we use for dealing with the uncertainty arising from a lack of knowledge of the precise values of input parameters to later-stage decision problems can also be used to address the uncertainty arising from a lack of knowledge of the future actions of another self-interested player. In fact, one way of viewing the outcome of a random variable is as a "decision" made by a DM about whose objective function nothing is known. Both cases require consideration of a set of outcomes arising from either the different ways in which the uncertain future could unfold or from the different possible actions the other players could take. Algorithms for solving these difficult optimization problems must, either explicitly or implicitly, rely on efficient methods for exploring this outcome space. This commonality turns out to be more than a philosophical abstraction. The mathematical connections between multistage and multilevel optimization problems run deep and existing algorithms for the two cases already exhibit common features, as we illustrate in what follows.

18.1.2 Focus

In the rest of this article, we address the broad class of optimization problems that we refer to from here on by the collective name *multistage mixed integer linear optimization problems*. Such problems allow for multiple decision stages, with decisions at each stage made by a different DM and with each stage followed by the revelation of the value of one or more random variables affecting the available actions at the subsequent stages. Each DM is assumed to have their own objective function for the evaluation of the decisions made at all stages following the stage at which they make their first decision, including stages whose decision they do not control. Importantly, the objective functions of different DMs may (but do not necessarily) conflict. Algorithmically, the focus of such problems is usually on determining an optimal decision at the first stage. At the point in time when later-stage decisions must be made, the optimization problem faced by those DMs has the same form as the one faced by early-stage DMs but, in it, the decisions made at the earlier stages act as deterministic inputs. Note that, while we have assumed different DMs at each stage, it is entirely possible to model scenarios in which a single DM makes multiple decisions over time. From a mathematical standpoint, this latter situation is equivalent to the case in which different DMs share the same objective function and we thus do not differentiate these situations in what follows.

Although the general framework we introduce applies more broadly, we focus here on problems with two decision stages and two DMs, as well as stochastic parameters whose values are realized between the two stages. We further restrict our consideration to the case in which we have both continuous and integer variables but the constraints and objective functions are linear. We refer to these problems as *two-stage mixed integer linear optimization problems* (2SMILPs). Despite this restricted setting, the framework can be extended to multiple stages and more general forms of constraints and objective functions in a conceptually straightforward way (see, e.g., Sect. 18.3.3).

18.2 Related and Previous Work

In this section, we give a brief overview of related works. The literature on these topics is vast and the below overview is not intended to be exhaustive by any means, but only to give a general sense of work that has been done to date. The interested reader should consult other articles in this volume for additional background and relevant citations.

18.2.1 Applications

Multilevel and multistage structures, whose two level/stage versions are known as *bilevel optimization problems* and *two-stage stochastic optimization problems with recourse* (2SPRs), arise naturally in a vast array of applications of which we touch on only a small sample here.

In the modeling of large organizations with hierarchical decision-making processes, such as corporations and governments, Bard [8] discusses a bilevel corporate structure in which top management is the leader and subordinate divisions, which may have their own conflicting objectives, are the followers. Similarly, government policy-making can be viewed from a bilevel optimization perspective: [10] models a government encouraging biofuel production through subsidies to the petro-chemical industry, while [5] models a central authority that sets prices and taxes for hazardous waste disposal while polluting firms respond by making location-allocation and recycling decisions.

A large body of work exists on *interdiction problems*, which model competitive games where two players have diametrically opposed goals and the first player has the ability to prevent one or more of the second player's possible activities (variables) from being engaged in at a non-zero level. Most of the existing literature on these problems has focused on variations of the well-studied *network interdiction problem* [47, 65, 76, 80, 81, 105, 139, 141], in which the lower-level DM is an entity operating a network of some sort and the upper-level DM (or interdictor) attempts to *interdict* the network as much as possible via the removal (complete or otherwise) of portions (subsets of arcs or nodes) of the network. A more general case which does not involve networks (the so-called *linear system interdiction problem*) was studied in [79] and later in [52]. A related set of problems involves an attacker disrupting warehouses or other facilities to maximize the resulting transportation costs faced by the firm (the follower) [35, 117, 144]. A *trilevel* version of this problem involves the firm first fortifying the facilities, then the attacker interdicting them, and finally the firm re-allocating customers [34]. More abstract graph-theoretical interdiction problems in which the vertices of a graph are removed in order to reduce the graphs' stability/clique number are studied in [38, 59, 113].

Multilevel problems arise in a wide range of industries. For instance, in the context of the electricity industry, Hobbs and Nelson [78] applies bilevel optimization to demand-side management while [68] formulates a trilevel problem with power network expansion investments in the first level, market clearing in the second, and redispatch in the third. Coniglio et al. [43] and Côté et al. [48] address the capacity allocation and pricing problem for the airline industry. Dempe et al. [51] presents a model for the natural-gas shipping industry. A large amount of work has been carried out in the context of traffic planning problems, including constructing road networks to maximize users' benefits [16], toll revenue maximization [93], and hazardous material transportation [84]. For a general review on these problems and for one specialized to price-setting, the reader is referred to [92, 106]. More applications arise in chemical engineering and bioengineering in the design and control of optimized systems. For example, Clark and Westerberg [36] optimizes a chemical process by controlling temperature and pressure (first-stage actions) where the system (second stage) reaches an equilibrium as it naturally minimizes the Gibbs free energy. Burgard et al. [25] develops gene-deletion strategies (first stage) to allow overproduction of a desired chemical by a cell (second stage). In the area of telecommunication networks, bilevel optimization has been used for modeling the behavior of a networking communication protocol (second-level problem) which the network operator, acting as first-level DM, can influence but not directly control. The case of routing over a TCP/IP network is studied in [1–4, 39].

The literature on game theory features many works on bilevel optimization problems naturally arising from the computation of *Stackelberg equilibria* in different settings. Two main variants of the Stackelberg paradigm are typically considered: one in which the followers can observe the action that the leader draws from its commitment and, therefore, the commitment is in pure strategies [133], and one in which the followers cannot do that directly and, hence, the leader's commitment can be in mixed strategies [45, 134]. While most of the works focus on the case with a single leader and a single follower (which leads to a proper bilevel optimization problem), some work has been done on the case with more than two players: see [12–14, 31, 41, 42, 44, 104] for the *single*-leader *multi*-follower case, [61, 95, 98, 99, 124] for the *multi*-leader *single*-follower case, or [30, 91, 96, 108] for the *multi*-leader *multi*-follower case. Practical applications are often found in *security games*, which correspond to competitive situations where a defender (leader) has to allocate scarce resources to protect valuable targets from an attacker (follower) [6, 87, 109, 128]. Other practical applications are found in, among others, inspection games [7] and mechanism design [116]. The works on the computation of a *correlated equilibrium* [32] as well those on Bayesian persuasion [33], where a leader affects the behavior of the follower(s) by a *signal*, also fall in this category.

Finally, there are deep connections between bilevel optimization and the algorithmic decision framework that drives branch and bound itself, and it is likely that the study of bilevel optimization problems may lead to improved methods for solving single-level optimization problems. For example, the problem of determining the disjunction whose imposition results in the largest bound improvement within a branch-and-bound framework and the problem of determining the maximum bound-improving inequality are themselves bilevel optimization problems [40, 102, 103]. The same applies in n-ary branching when one looks for a branching decision leading to the smallest possible number of child nodes [97, 115].

Multistage problems and, in particular, two-stage stochastic optimization problems with recourse, arise in an equally wide array of application areas, including scheduling, forestry, pollution control, telecommunication and finance. Grass and Fischer [67] surveys literature, applications, and methods for solving disaster-management problems arising in the humanitarian context. Gupta et al. [69] addresses network-design problems where, in the second stage, after one of a finite set of scenarios is realized, additional edges of the network can be bought, and provides constant-factor approximation algorithms. A number of works address the two-stage stochastic optimization with recourse version of classical combinatorial optimization problems: among others, [54] considers the spanning-tree problem, [86] the matching problem, and [64, 66] the vehicle routing problem. For references to other areas of applicability, see the books [18, 83] and, in particular, [135].

18.2.2 Algorithms

The first recognizable formulations for bilevel optimization problems were introduced in the 1970s in [24] and this is when the term was also first coined. Beginning in the early 1980s, these problems attracted increased interest. Vicente and Calamai [131] provides a large bibliography of the early developments.

There is now a burgeoning literature on continuous bilevel linear optimization, but it is only in the past decade that work on the discrete case has been undertaken in earnest by multiple research groups. Moore and Bard [107] was the first to introduce a framework for general integer bilevel linear optimization and to suggest a simple branch-and-bound algorithm. The same authors also proposed a more specialized algorithm for binary bilevel optimization problems in [9]. Following these early works, the focus shifted primarily to various special cases, especially those in which the lower-level problem has the integrality property. Dempe [49] considers a special case characterized by continuous upper-level variables and integer lower-level variables and uses a cutting plane approach to approximate the lower-level feasible region (a somewhat similar approach is adopted in [50] for solving a bilinear mixed integer bilevel problem with integer second-level variables). Wen and Yang [137] considers the opposite case, where the lower-level problem is a linear optimization problem and the upper-level problem is an integer optimization problem, using linear optimization duality to derive exact and heuristic solutions.

The publication of a general algorithm for pure integer problems in [53] (based on the groundwork laid in a later-published dissertation [52]) spurred renewed interest in developing general-purpose algorithms. The evolution of work is summarized in Table 18.1, which indicates the types of variables supported in both the first and

Table 18.1 Evolution of algorithms for bilevel optimization

Citation	Stage 1 variable types	Stage 2 variable types
[137]	B	C
[9]	B	B
[56]	B, C	B, C
[114]	B, C	B, C
[62]	B	C
[52, 53]	G	G
[90]	G or C	G
[11]	B, C	C
[142]	G	G, C
[143]	G, C	G, C
[28]	G	G
[27]	B	B
[77]	B, C	B, C
[127]	G, C	G, C
[136]	G	G
[101]	G	G, C
[57, 58]	G, C	G, C

second stages (C indicates continuous, B indicates binary, and G indicates general integer). The aforementioned network interdiction problem is a special case that continues to receive significant attention, since tractable duality conditions exist for the lower-level problem [47, 65, 76, 80, 81, 105, 139, 141].

As for the case of multistage stochastic optimization, the two-stage linear stochastic optimization problem with recourse in which both the first- and second-stage problems contain only continuous variables has been well studied both theoretically and methodologically. Birge and Louveaux [18] and Kall and Mayer [83] survey the related literature. The integer version of the problem was first considered in the early 1990s by Louveaux and van der Vlerk [100] for the case of two-stage problems with *simple integer recourse*. Combining the methods developed for the linear version with the branch-and-bound procedure, Laporte and Louveaux [94] proposed an algorithm known as the *integer L-shaped method* where the first-stage problem contains only binary variables. Due to the appealing structural properties of a (mixed) binary integer optimization problem, a substantial amount of literature since then has been considering the case of a two-stage stochastic problem with (mixed) binary variables in one or both stages [60, 120, 123]. The case of two-stage problems with a pure integer recourse has also been frequently visited, see [89, 119]. It must be noted that methods such as these, which are typically developed for special cases, often rely on the special structure of the second-stage problem, thus being often not applicable to the two-stage problem with mixed integer restrictions. Algorithms for stochastic optimization problems with integer recourse were proposed by Carøe and Tind [29] and Sherali and Fraticelli [122].

18.3 Setup and Preliminaries

The defining feature of a multistage optimization problem is that the values of the *first-stage* variables (sometimes called *upper-level* or *leader* variables in the bilevel optimization literature) must be (conceptually) fixed without explicit knowledge of future events, the course of which can be influenced by the first-stage decision itself. Due to this influence, the perceived "value" of the first-stage decision must take into account the effect of this decision on the likelihood of occurrence of these future events.

More concretely, the first-stage DM's overall objective is to minimize the sum of two terms, the first term representing the immediate cost of implementation of the first-stage solution and the second term representing the desirability of the first-stage decision in terms of its impact on the decisions taken at later stages. The general form of a *two-stage mixed integer linear optimization problem* is then

$$\min_{x \in \mathcal{P}_1 \cap X} \{cx + \Xi(x)\}, \tag{2SMILP}$$

where

$$\mathcal{P}_1 = \left\{x \in \mathbb{R}^{n_1} \mid A^1 x \geq b^1\right\}$$

is the *first-stage feasible region*, with $A^1 \in \mathbb{Q}^{m_1 \times n_1}$ and $b^1 \in \mathbb{Q}^{m_1}$ defining the associated linear constraints, and $X = \mathbb{Z}_+^{r_1} \times \mathbb{Q}_+^{n_1 - r_1}$ representing integrality, rationality, and non-negativity requirements on the first-stage variables, denoted by x. Note that we require the continuous variables to take on rational values in order to ensure that the second-stage problem defined in (SS) below has an optimal value that is attainable when that value is finite. In practice, solvers always return such solutions, so this is a purely technical detail. The linear function cx with $c \in \mathbb{Q}^{n_1}$ is the aforementioned term that reflects the immediate cost of implementing the first-stage solution. The function $\Xi : \mathbb{Q}^{n_1} \to \mathbb{Q} \cup \{\pm\infty\}$ is the *risk function*, which takes only rational input for technical reasons discussed later. Ξ is the aforementioned term representing the first-stage DM's evaluation of the impact of a given choice for the value of the first-stage variables on future decisions. Similar concepts of risk functions have been employed in many different application domains and will be briefly discussed in Sect. 18.3.3. To enable the development of a practical methodology for the solution of these problems, however, we now define the specific class of functions we consider.

18.3.1 Canonical Risk Function

Our canonical risk function is a generalization of the risk function traditionally used in defining 2SPRs. As usual, let us now introduce a random variable U over an outcome space Ω representing the set of possible future scenarios that could be realized between the making of the first- and second-stage decisions. The values of this random variable will be input parameters to the so-called *second-stage problem* to be defined below.

As is common in the literature on 2SPRs, we assume that U is discrete, i.e., that the outcome space Ω is finite, so that $\omega \in \Omega$ represents which of a finite number of explicitly enumerated scenarios is actually realized. In practice, this assumption is not very restrictive, as one can exploit any algorithm for the case in which Ω is assumed finite to solve cases where Ω is not (necessarily) finite by utilizing a technique for discretization, such as *sample average approximation* (SAA) [121]. As U is discrete in this work, we can associate with it a probability distribution defined by $p \in \mathbb{R}^{|\Omega|}$ such that $0 \leq p_\omega \leq 1$ and $\sum_{\omega \in \Omega} p_\omega = 1$.

With this setup, the canonical risk function for $x \in \mathbb{Q}^{n_1}$ is

$$\Xi(x) = \mathbb{E}\left[\Xi_\omega(x)\right] = \sum_{\omega \in \Omega} p_\omega \Xi_\omega(x), \tag{RF}$$

where $\Xi_\omega(x)$ is the *scenario risk function*, defined as

$$\Xi_\omega(x) = \min \left\{ d^1 y^\omega \;\middle|\; y^\omega \in \operatorname{argmin}\{d^2 y \mid y \in \mathcal{P}_2(b^2_\omega - A^2_\omega x) \cap Y\} \right\}; \tag{2SRF}$$

the set

$$\mathcal{P}_2(\beta) = \left\{ y \in \mathbb{R}^{n_2} \mid G^2 y \geq \beta \right\}$$

is one member of a family of polyhedra that is parametric w.r.t. the right-hand side vector $\beta \in \mathbb{R}^{m_2}$ and represents the second-stage feasibility conditions; and $Y = \mathbb{Z}^{r_2}_+ \times \mathbb{Q}^{n_2 - r_2}_+$ represents the second-stage integrality and non-negativity requirements. The deterministic input data defining Ξ_ω are $d^1, d^2 \in \mathbb{Q}^{n_2}$ and $G^2 \in \mathbb{Q}^{m_2 \times n_2}$. $A^2_\omega \in \mathbb{Q}^{m_2 \times n_1}$ and $b^2_\omega \in \mathbb{Q}^{m_2}$ represent the realized values of the random input parameters in scenario $\omega \in \Omega$, i.e., $U(\omega) = (A^2_\omega, b^2_\omega)$.

As indicated in (RF) and (2SRF), the inner optimization problem faced by the second-stage DM is parametric only in its right-hand side, which is determined jointly by the value ω of the random variable U and by the chosen first-stage solution. It will be useful in what follows to define a family of *second-stage* optimization problems

$$\inf \left\{ d^2 y \;\middle|\; y \in \mathcal{P}_2(\beta) \cap Y \right\} \tag{SS}$$

that are parametric in the right-hand side $\beta \in \mathbb{R}^{m_2}$ (we use "inf" instead of "min" here because, for $\beta \notin \mathbb{Q}^{m_2}$, the minimum may not exist). By further defining

$$\beta^{\omega}(x) = b^2_{\omega} - A^2_{\omega}x$$

to be the parametric right-hand side that arises when the chosen first-stage solution is $x \in X$ and the realized scenario is $\omega \in \Omega$, we can identify the member of the parametric family defined in (SS) in scenario $\omega \in \Omega$ when the chosen first-stage solution is $x \in X$ as that with feasible region $\mathcal{P}_2(\beta^{\omega}(x)) \cap Y$.

Associated with each $x \in \mathbb{Q}^{n_1}$ and $\omega \in \Omega$ is the set of all alternative optimal solutions to the second-stage problem (SS) (we allow for $x \notin X$ here because such solutions arise when solving certain relaxations), called the *rational reaction set* and denoted by

$$\mathcal{R}^{\omega}(x) = \operatorname{argmin}\{d^2 y \mid y \in \mathcal{P}_2(b^2_{\omega} - A^2_{\omega}x) \cap Y\}.$$

For a given $x \in \mathbb{Q}^{n_1}$, $\mathcal{R}^{\omega}(x)$ may be empty if $\mathcal{P}_2(b^2_{\omega} - A^2_{\omega}x) \cap Y$ is itself empty or if the second-stage problem is unbounded (we assume in Sect. 18.3.2 that this cannot happen, however).

When $|\mathcal{R}^{\omega}(x)| > 1$, the second-stage DM can, in principle, choose which alternative optimal solution to implement. We must therefore specify in the definition of the risk function a rule by which to choose one of the alternatives. According to our canonical risk function (RF) and the corresponding scenario risk function (2SRF), the rule is to choose, for each scenario $\omega \in \Omega$, the alternative optimal solution that minimizes $d^1 y^{\omega}$, which corresponds to choosing the collection $\{y^{\omega}\}_{\omega \in \Omega}$ of solutions to the individual scenario subproblems that minimizes

$$d^1 \left(\sum_{\omega \in \Omega} p_{\omega} y^{\omega} \right).$$

This is known as the *optimistic* or *semi-cooperative* case in the bilevel optimization literature, since it corresponds to choosing the alternative that is most beneficial to the first-stage DM. Throughout the article, we consider this case unless otherwise specified. In Sect. 18.3.3, we discuss other forms of risk function.

Because of the subtleties introduced above, there are a number of ways one could define the "feasible region" of (2SMILP). We define the *feasible region* for scenario ω (with respect to both first- and second-stage variables) as

$$\mathcal{F}^{\omega} = \{(x, y^{\omega}) \in X \times Y \mid x \in \mathcal{P}_1, y^{\omega} \in \mathcal{R}^{\omega}(x)\} \tag{FR}$$

and members of $\mathcal{F}^{\omega}$ as *feasible solutions* for scenario ω. Note that this definition of feasibility does not prevent having $(x, y^{\omega}) \in \mathcal{F}^{\omega}$ but $d^1 y^{\omega} > \Xi_{\omega}(x)$. This will not cause any serious difficulties, but is something to keep in mind.

We can similarly define the feasible region with respect to just the first-stage variables as

$$\mathcal{F}^1 = \bigcap_{\omega \in \Omega} \operatorname{proj}_x(\mathcal{F}^\omega). \tag{FS-FR}$$

Since $\Xi(x) = \infty$ for $x \in \mathbb{Q}^{n_1}$ if the feasible region $\mathcal{P}_2(\beta^\omega(x)) \cap Y$ of the second-stage problem (SS) is empty for some $\omega \in \Omega$, we have that, for $x \in \mathcal{P}_1 \cap X$, the following are equivalent:

$$x \in \mathcal{F}^1 \Leftrightarrow x \in \bigcap_{\omega \in \Omega} \operatorname{proj}_x(\mathcal{F}^\omega) \Leftrightarrow \mathcal{R}^\omega(x) \neq \emptyset \ \forall \omega \in \Omega \Leftrightarrow \Xi(x) < \infty.$$

Finally, it will be convenient to define $\mathcal{P}^\omega$ to be the feasible region of the relaxation of the deterministic two-stage problem under scenario $\omega \in \Omega$ that is obtained by dropping the optimality requirement for the second-stage variables y^ω, as well as any integrality restrictions. Formally, we have:

$$\mathcal{P}^\omega = \left\{(x, y^\omega) \in \mathbb{R}_+^{n_1+n_2} \mid x \in \mathcal{P}_1, y^\omega \in \mathcal{P}_2(b^2_\omega - A^2_\omega x)\right\}.$$

Later in Sect. 18.7, we will use these sets to define a relaxation for the entire problem that will be used as the basis for the development of a branch-and-cut algorithm.

18.3.2 Technical Assumptions

We now note the following assumptions made in the remainder of the article.

Assumption 18.3.1 *$\mathcal{P}^\omega$ is bounded for all $\omega \in \Omega$.* △

This assumption, which is made for ease of presentation and can be relaxed, results in the boundedness of (2SMILP).

Assumption 18.3.2 *All first-stage variables with at least one non-zero coefficient in the second-stage problem (the so-called* linking variables*) are integer, i.e.,*

$$L = \left\{i \in \{1, \ldots, n_1\} \;\middle|\; a_i^\omega \neq 0 \textit{ for some } \omega \in \Omega\right\} \subseteq \{1, \ldots, r_1\},$$

where a_i^ω represents the i^{th} column of matrix A^2_ω. △

These two assumptions together guarantee that an optimal solution exists whenever (2SMILP) is feasible [132]. It also guarantees that the convex hull of $\mathcal{F}^\omega$ is a polyhedron, which is important for the algorithms we discuss later. Note that, due to the assumption of optimism, we can assume w.l.o.g. that all first-stage variables are linking variables by simply interpreting the non-linking variables as belonging to the second stage. While this may seem conceptually inconsistent with the intent of the original model, it is not difficult to see that the resulting model is *mathematically* equivalent, since these variables do not affect the second-stage problem and thus, the optimistic selection of values for those variables will be the same in either case.

Before closing this section, we remark that, in this article, we do not allow second-stage variables in the first-stage constraints. While this case can be handled with techniques similar to those we describe in the article from an algorithmic perspective, it does require a more complicated notation which, for the sake of clarity, we prefer not to adopt. Detailed descriptions of algorithms for this more general case in the bilevel setting are provided in [23, 127].

18.3.3 Alternative Models

18.3.3.1 Alternative Form of (2SMILP)

For completeness, we present here an alternative form of (2SMILP) that is closer to the traditional form in which bilevel optimization problems are usually specified in the literature. Adopting the traditional notation, (2SMILP) can be alternatively written as

$$
\begin{aligned}
\min_{x,\{y^\omega\}_{\omega\in\Omega}} \quad & cx + \sum_{\omega\in\Omega} p_\omega d^1 y^\omega \\
\text{s.t.} \quad & A^1 x \geq b^1 \\
& x \in X \\
& \left.\begin{aligned} y^\omega \in \operatorname{argmin}_y \ & d^2 y \\ \text{s.t.} \quad & A^2_\omega x + G^2 y \geq b^2_\omega \\ & y \in Y \end{aligned}\right\} \forall \omega \in \Omega.
\end{aligned}
\tag{2SMILP-Alt}
$$

Note that, in the first stage, the minimization is carried out with respect to both x and y. This again specifies the optimistic case discussed earlier, since the above formulation requires that, for a given $x \in X$, we select $\{y^\omega\}_{\omega\in\Omega}$ such that

$$
d^1\left(\sum_{\omega\in\Omega} p_\omega y^\omega\right)
$$

is minimized.

18.3.3.2 Pessimistic Risk Function

As already pointed out, the canonical risk function defined in (RF) assumes the *optimistic case*, since it encodes selection of the alternative optimal solution to the second-stage problem that is most beneficial to the first-stage DM. This is the case we focus on in the remainder of the article. The pessimistic case, on the other hand, is easily modeled by defining the scenario risk function to be

$$\Xi_\omega(x) = \max \left\{ d^1 y^\omega \;\middle|\; y^\omega \in \operatorname{argmin}\{d^2 y \mid y \in \mathcal{P}_2(\beta^\omega(x)) \cap Y\} \right\}.$$

We remark that, while the optimistic and pessimistic cases may coincide in some cases (e.g., when (SS) admits a single optimal solution for every x), this coincidence is rarely observed in practice and would be hard to detect in any case. In general, the pessimistic case is more difficult to solve, though the algorithms discussed in Sect. 18.7 can be modified to handle it.

18.3.3.3 Recursive Risk Functions

Although we limit ourselves to problems with two stages in this article, we briefly mention that more general risk functions can be defined by recursively defining risk functions at earlier stages in terms of later-stage risk functions. This is akin to the recursive definition of the cost-to-go functions that arise in stochastic dynamic programming (see [17]). With such recursive definitions, it is possible to generalize much of the methodology described here in a relatively straightforward way, though the algorithm complexity grows exponentially with the addition of each stage. It is doubtful exact algorithms can be made practical in such cases.

18.3.3.4 Other Risk Functions

Other forms of risk function have been used in the literature, especially in finance. In robust optimization, for example, one might consider a risk function of the form

$$\Xi(x) = \max_{\omega \in \Omega} \{\Xi_\omega(x)\},$$

which models the impact on the first-stage DM of the worst-case second-stage realization of the random variables. A popular alternative in finance applications that is slightly less conservative is the *conditional value at risk*, the expected value taken over the worst α-percentile of outcomes [112, 129]. While it is possible to incorporate such risk functions into the general algorithmic framework we present here, for the purposes of limiting the scope of the discussion, we focus herein only on risk functions in the canonical form (RF).

18.3.4 Related Classes

With Ξ defined as in (RF), the problem (2SMILP) generalizes several well-known classes of optimization problems.

18.3.4.1 Single-Stage Problems

When $d^1 = d^2$ and $|\Omega| = 1$, the two stages of (2SMILP) can be collapsed into a single stage and the problem reduces to a traditional mixed integer linear optimization problem (MILP). It is natural that algorithms for (2SMILP) rely heavily on solving sequences of related single-stage MILPs and we discuss parametric versions of this class in later sections. For continuity, we utilize the notation for the second-stage variables and input data throughout. The case of $r_2 = 0$ (in which there are no integer variables) further reduces to a standard linear optimization problem (LP).

18.3.4.2 Bilevel Problems

When $|\Omega| = 1$ and assuming that we may have $d^1 \neq d^2$, (2SMILP) takes the form of a *mixed integer bilevel linear optimization problem* (MIBLP). Dropping the scenario super/subscript for simplicity, this problem is more traditionally written as

$$\min_{x \in \mathcal{P}_1 \cap X, y \in Y} \left\{ cx + d^1 y \;\middle|\; y \in \underbrace{\arg\min\{d^2 y \mid y \in \mathcal{P}_2(b^2 - A^2 x) \cap Y\}}_{\mathcal{R}(x)} \right\}. \quad \text{(MIBLP)}$$

Note that this formulation implicitly specifies the optimistic case since, if $\mathcal{R}(x)$ is not a singleton, it requires that, among the alternative optima, the solution minimizing $d^1 y$ be chosen. In this setting, the *bilevel risk function* can be written as

$$\Xi(x) = \min \left\{ d^1 y \mid y \in \mathcal{R}(x) \right\}.$$

18.3.4.3 Two-Stage Stochastic Optimization Problems with Recourse

When $d^1 = d^2$, either the inner or the outer minimization in (2SRF) is redundant and (2SMILP) takes the form of a *two-stage stochastic mixed integer linear optimization problem with recourse*. In this case, for each scenario $\omega \in \Omega$ we can write the scenario risk function more simply as

$$\Xi_\omega(x) = \min \left\{ d^1 y^\omega \mid y^\omega \in \mathcal{P}_2(b^2_\omega - A^2_\omega x) \cap Y \right\}.$$

The second-stage solution y^{ω} corresponding to scenario $\omega \in \Omega$ is usually called the *recourse decision*. These problems involve a single DM optimizing a single objective function, but capable of controlling two sets of variables: the first-stage *here-and-now* variables x and the second-stage *wait-and-see* variables y^{ω}, whose value is set after observing the realization of the random event ω.

18.3.4.4 Zero-Sum and Interdiction Problems

For $d^1 = -d^2$ (and typically, $|\Omega| = 1$), (2SMILP) subsumes the case of *zero-sum problems*, which model competitive games in which two players have exactly opposing goals. An even more specially-structured subclass of zero-sum problems are *interdiction problems*, in which the first-stage variables are in one-to-one correspondence with those of the second stage and represent the ability of the first-stage DM to "interdict" (i.e., forcing to take value zero) individual variables of the second-stage DM. Formally, the effect of interdiction can be modeled using a variable upper-bound constraint

$$y \leq u(e - x)$$

in the second-stage problem, where $u \in \mathbb{R}^n$ is a vector of natural upper bounds on the vector of variables y and e is an n-dimensional column vector of ones (here, $n = n_1 = n_2$). Formally, the *mixed integer interdiction problem* is

$$\max_{x \in \mathcal{P}_1 \cap X} \min_{y \in \mathcal{P}_2(x) \cap Y} d^2 y,$$

where (abusing notation slightly), we have

$$\mathcal{P}_2(x) = \left\{ y \in \mathbb{R}^{n_2} \mid G^2 y \geq b^2, y \leq u(e - x) \right\}.$$

18.4 Computational Complexity

Within the discrete optimization community, the framework typically used for assessing problem complexity is based primarily on the well-known theory of NP-completeness, which has evolved from the foundational work of [46, 63, 85]. This has led to the ubiquitous practice of classifying optimization problems as being either in the class P or the class NP-hard, the latter being an all-encompassing and amorphous class that includes essentially all optimization problems not known to be polynomially solvable. This categorization lacks the refinement necessary for consideration of classes such as those described in this article. It is indeed easy to show that multistage optimization problems are NP-hard in general [15, 26, 73, 82], but this merely tells us that these problems are not in P (assuming $\mathsf{P} \neq \mathsf{NP}$), which is not surprising. What we would really like to know is for which complexity class (the decision versions of) these problems are *complete*.

In the presence of a hierarchical structure with k levels (and when Ω is a singleton), the natural complexity class to consider is Σ_k^{P}, i.e., the k^{th} level of the *polynomial hierarchy*. From an optimization perspective, this hierarchy (originally introduced in [125]) is a scheme for classifying multilevel decision problems beyond the usual classes P and NP. The class P (which contains all decision problems that can be solved in polynomial time) occupies the 0^{th} level, also known as Σ_0^{P}. The first level, Σ_1^{P}, is the class also known as NP, which consists of all problems for which there exists a certificate verifiable in polynomial time or, equivalently, all problems that can be solved in non-deterministic polynomial time. The k^{th} level, Σ_k^{P}, contains all problems with certificates that can be verified in polynomial time (equivalently, all problems solvable in non-deterministic polynomial time), assuming the existence of an oracle for solving problems in the class $\Sigma_{k-1}^{\mathsf{P}}$. While it is clear that $\Sigma_k^{\mathsf{P}} \subseteq \Sigma_\ell^{\mathsf{P}}$ for any $k, \ell \in \mathbb{N} \cup \{0\}$ with $k \leq \ell$, $\Sigma_k^{\mathsf{P}} \subset \Sigma_\ell^{\mathsf{P}}$ is conjectured to hold for all $k, \ell \in \mathbb{N} \cup \{0\}$ with $k < \ell$ (the well-known $\mathsf{P} \neq \mathsf{NP}$ conjecture is a special case). It is also known that $\Sigma_k^{\mathsf{P}} = \Sigma_{k+1}^{\mathsf{P}}$ would imply $\Sigma_k^{\mathsf{P}} = \Sigma_\ell^{\mathsf{P}}$ for all $\ell \geq k+1$, which would cause the polynomial hierarchy to collapse to level k (for $k = 0$, we would have P = NP). The notions of completeness and hardness commonly used for NP translate directly to Σ_k^{P}. A proof that k-level optimization problems with binary variables, linear constraints, and linear objective functions are hard for Σ_k^{P} is contained in [82]. Such result suffices to show that the multistage problems with k stages treated in this article are (in their optimization version) Σ_k^{P}-hard (and those with $k = 2$ stages are Σ_2^{P}-hard). A compendium of Σ_2^{P}-complete/hard problems, somewhat similar in spirit to [63], can be found in [118], with more recent updates available online.

For the case of two-stage stochastic optimization problems with recourse with linear constraints, linear objective functions, and mixed integer variables, the assumption of a finite outcome space Ω of either fixed or polynomially bounded size suffices to guarantee that the decision version of such a problem is NP-complete. Indeed, when $|\Omega|$ is considered a constant or is bounded by a polynomial in the total number of variables and constraints, one can directly introduce a block-structured reformulation of the problem with one block per scenario $\omega \in \Omega$ that contains the coefficients of the constraints that y^ω should satisfy (we discuss such a reformulation in Sect. 18.6). As such reformulation is of polynomial size, solutions to the corresponding optimization problem can clearly be certified in polynomial time by checking that they satisfy all the polynomially-many constraints featured in the formulation, which, in turn, implies that the problem belongs to NP. When the outcome space Ω is continuous, the problem becomes #P-hard in general [55, 72]. While a single sample average approximation problem with a finite or polynomially-bounded number of samples can be used to approximate a continuous problem by solving a single discrete optimization problem of polynomial size, Hanasusanto et al. [72] shows that even finding an approximate solution using the SAA method is #P-hard. New results on the complexity of 2SPRs featuring a double-exponential algorithm can be found in [88].

18.5 Duality and the Value Function

Virtually all algorithms for the exact solution of optimization problems produce a proof of optimality that depends on the construction of a solution to a *strong dual*. Although the duality theory for MILPs is not widely known, the most effective algorithms for solving MILPs (which are variants of the well-known branch-and-bound algorithm) *do* produce a solution to a certain dual problem. A natural approach to solving (2SMILP) is therefore to embed the production of the "dual proof" of optimality of the second-stage problem (SS) into the formulation of the first-stage problem, reducing the original two-stage problem to a traditional single-stage optimization problem.

The reformulations and algorithmic approaches that we present in Sects. 18.6 and 18.7 all use some variant of this strategy. In particular, the algorithms we describe are based on iteratively strengthening an initial relaxation in a fashion reminiscent of many iterative optimization algorithms. The strengthening operation essentially consists of the dynamic construction of both a proof of optimality of the second-stage problem and of corresponding first- and second-stage solutions.

In the remainder of the section, we introduce the central concepts of a duality theory for mixed integer linear optimization problems (and more general discrete optimization problems), emphasizing its connection to solution methods for (2SMILP). This introduction is necessarily brief and we refer the reader to [71, 74, 75] for more details specific to the treatment here and to [138, 140] for earlier foundational work on IP duality. Although the "dual problem" is usually a fixed (non-parametric) optimization problem associated with a fixed (non-parametric) "primal problem," the typical concepts of duality employed in constructing dual proofs of optimality and in designing solution algorithms inherently involve parametric families of optimization problems. This makes the tools offered by this theory particularly suitable for employment in this setting. To preserve the connection with the material already introduced, we consider the family of MILPs parameterized on the right-hand side $\beta \in \mathbb{R}^{m_2}$ that was introduced earlier as (SS) and use the same notation. We reproduce it here for convenience:

$$\inf\left\{d^2y \;\middle|\; y \in \mathcal{P}_2(\beta) \cap Y\right\}, \tag{SS}$$

where

$$\mathcal{P}_2(\beta) = \left\{y \in \mathbb{R}^{n_2} \mid G^2y \geq \beta\right\}$$

and $\beta \in \mathbb{R}^{m_2}$ is the input parameter. When we want to refer to a (fixed) generic instance in this parametric family, the notation b will be used to indicate a fixed (but arbitrary) right-hand side. We also refer to specific right-hand sides arising in the solution of (2SMILP) using the notation defined earlier.

18.5.1 Value Functions

Among possible notions of duality, the one most relevant to the development of optimization algorithms is one that also has an intuitive interpretation in terms of familiar economic concepts. This theory rests fundamentally on an understanding of the so-called *value function*, which we introduce below. The value function of an MILP has been studied by a number of authors and a great deal is known about its structure and properties. Early work on the value function includes [19–22], while the material here is based on the work in [70, 71, 74, 75].

As a starting point, consider an instance of (SS) with fixed right-hand side b and let us interpret the values of the variables as specifying a numerical "level of engagement" in certain activities in an economic market. Further, let us interpret the constraints as corresponding to limitations imposed on these activities due to available levels of certain scarce resources (it is most natural to think of "$\leq$" constraints in this interpretation). In each row j of the constraint matrix, the coefficient G^2_{ij} associated with activity (variable) i can then be thought of as representing the rate at which resource j is consumed by engagement in activity i. In this interpretation, the optimal primal solution then specifies the level of each activity in which one should engage in order to maximize profits (it is most natural here to think in terms of maximization), given the fixed level of resources b.

Assuming that additional resources were available, how much should one be willing to pay? The intuitive answer is that one should be willing to pay at most the marginal amount by which profits would increase if more of a particular resource were made available. Mathematically, this information can be extracted from the *value function* $\phi : \mathbb{R}^{m_2} \rightarrow \mathbb{R} \cup \{\pm\infty\}$ associated with (SS), defined by

$$\phi(\beta) = \inf_{y \in \mathcal{P}_2(\beta) \cap Y} d^2 y, \tag{2SVF}$$

for $\beta \in \mathbb{R}^{m_2}$. Since this function returns the optimal profit for any given basket of resources, its gradient at b (assuming ϕ is differentiable at b) tells us what the marginal change in profit would be if the level of resources available changed in some particular direction. Thus, the gradient specifies a "price" on that basket of additional resources.

The reader familiar with the theory of duality for linear optimization problems should recognize that the solution to the usual dual problem associated with an LP of the form (SS) (i.e., assuming $r_2 = 0$) provides exactly this same information. In fact, we describe below that the set of optimal solutions to the LP dual are precisely the subgradients of its associated value function. This dual solution can hence be interpreted as a linear price on the resources and is sometimes referred to as a vector of "dual prices." The optimal dual prices allow us to easily determine whether it will be profitable to enter into a particular activity i by comparing the profit d^2_i obtained by entering into that activity to the cost uG^2_i of the required resources, where u is a given vector of dual prices and G^2_i is the i-th column of G^2. The difference $d^2_i - uG^2_i$

between the profit and the cost is the *reduced profit/cost* in linear optimization. It is easily proven that the reduced profit of each activity entered into at a non-zero level (i.e., reduced profits of the variables with non-zero value) must be non-negative (again, in the case of maximization) and duality provides an intuitive economic interpretation of this result.

Although the construction of the full value function is challenging even in the simplest case of a linear optimization problem, approximations to the value function in the local area around b can still be used for sensitivity analysis and in optimality conditions. The general dual problem we describe next formalizes this idea by formulating the problem of constructing a function that bounds the value function from below everywhere but yields a strong approximation near a fixed right-hand side b. Such a so-called "dual function" can yield approximate "prices" and its iterative construction can also be used in a more technical way to guide the evolution of an algorithm by providing gradient information helpful in finding the optimal solution, as well as providing a proof of its optimality.

18.5.2 Dual Functions

The above discussion leads to the natural concept of a *dual (price) function* from which we can derive a general notion of a *dual problem*.

Definition 18.5.1 A *dual function* $F : \mathbb{R}^{m_2} \to \mathbb{R}$ is one that satisfies $F(\beta) \leq \phi(\beta)$ for all $\beta \in \mathbb{R}^{m_2}$. We call such a function *strong* at $b \in \mathbb{R}^{m_2}$ if $F(b) = \phi(b)$. △

Dual functions are naturally associated with *relaxations* of the original problem, as the value function of any relaxation yields a feasible dual function. In particular, the value function of the well-known LP relaxation is the best convex under-estimator of the value function.

Also of interest are functions that bound the value function from above, which we refer to as *primal functions*.

Definition 18.5.2 A *primal function* $H : \mathbb{R}^{m_2} \to \mathbb{R}$ is one that satisfies $H(\beta) \geq \phi(\beta)$ for all $\beta \in \mathbb{R}^{m_2}$. We call such a function *strong* at b if $H(b) = \phi(b)$. △

In contrast to dual functions, primal functions are naturally associated with *restrictions* of the original problem and the value function of any such restriction yields a valid primal function.

It is immediately evident that a pair of primal and dual functions yields optimality conditions. If we have a primal function H^* and a dual function F^* such that $F^*(b) = \gamma = H^*(b)$ for some $b \in \mathbb{R}^{m_2}$, then we must also have $\phi(b) = \gamma$. Proofs of optimality of this nature are produced by many optimization algorithms.

18.5.3 Dual Problems

The concepts we have discussed so far further lead us to the definition of a generalized dual problem, originally introduced in [71], for an instance of (SS) with right-hand side $b \in \mathbb{R}^{m_2}$. This problem simply calls for the construction of a dual function that is strong for a particular fixed right-hand side $b \in \mathbb{R}^{m_2}$ by determining

$$\max_{F \in \Upsilon^{m_2}} \{F(b) : F(\beta) \le \phi(\beta),\ \forall \beta \in \mathbb{R}^{m_2}\}, \tag{MILPD}$$

where $\Upsilon^{m_2} \subseteq \{f \mid f : \mathbb{R}^{m_2} \to \mathbb{R}\}$. Here, Υ^{m_2} can be taken to be a specific class of functions, such as linear or subadditive, to obtain specialized dual problems for particular classes of optimization problems. It is clear that (MILPD) always has a solution F^* that is strong, provided that the value function is real-valued everywhere (and hence belongs to Υ^{m_2}, however it is defined), since ϕ itself is a solution whenever it is finite everywhere.[1]

Although it may not be obvious, this notion of a dual problem naturally generalizes existing notions for particular problem classes. For example, consider again a parametric family of LPs defined as in (SS) (i.e., assuming $r_2 = 0$). We show informally that the usual LP dual problem with respect to a fixed instance with right-hand side b can be derived by taking Υ^{m_2} to be the set of all non-decreasing linear functions in (MILPD) and simplifying the resulting formulation. First, let a non-decreasing linear function $F : \mathbb{R}^{m_2} \to \mathbb{R}$ be given. Then, $\exists u \in \mathbb{R}^{m_2}_{+}$ such that $F(\beta) = u\beta$ for all $\beta \in \mathbb{R}^{m_2}$. It follows that

$$F(\beta) = u\beta \le uG^2 y = \sum_{j=1}^{n_2} uG^2_j y_j \ \forall \beta \in \mathbb{R}^{m_2}.$$

From the above, it then follows that, for any $\beta \in \mathbb{R}^{m_2}$, we have

$$\begin{aligned} uG^2_j \le d^2_j \ \forall j \in \{1, \ldots, n_2\} &\Rightarrow u\beta \le uG^2 y \le d^2 y \ \forall y \in \mathcal{P}_2(\beta) \cap Y \\ &\Rightarrow u\beta \le \min_{y \in \mathcal{P}_2(\beta) \cap Y} d^2 y \\ &\Rightarrow u\beta \le \phi(\beta) \\ &\Rightarrow F(\beta) \le \phi(\beta). \end{aligned}$$

The conditions on the left-hand side above are exactly the feasibility conditions for the usual LP dual and the final condition on the right is the feasibility condition

[1] When the value function is not real-valued everywhere, we have to show that there is a real-valued function that coincides with the value function when it is real-valued and is itself real-valued everywhere else, but is still a feasible dual function (see [140]).

for (MILPD). Hence, the usual dual feasibility conditions ensure that u defines a linear function that bounds the value function from below and is a dual function in the sense we have defined. The fact that the epigraph of ϕ is a convex polyhedral cone in this case (it is the max of linear functions associated with extreme points of the feasible region of the dual problem) is enough to show that the dual (MILPD) is strong in the LP case, even when we take Υ^{m_2} to be the set of (non-decreasing) linear functions. Furthermore, it is easy to show that any subgradient of ϕ at b is an optimal solution (and in fact, the set of all dual feasible solutions is precisely the subdifferential of the value function at the origin).

The concepts just discussed can be easily seen in the following small example (note that this example is equality-constrained, in which case most of the above derivation carries through unchanged, but the dual function no longer needs to be non-decreasing).

Dual Function of an LP

$$\begin{aligned} \min\ & 6y_1 + 7y_2 + 5y_3 \\ \text{s.t. } & 2y_1 - 7y_2 + y_3 = b \\ & y_1, y_2, y_3 \in \mathbb{R}_+. \end{aligned}$$

The solution to the dual of this LP is unique whenever b is non-zero and can be easily obtained by considering the ratios c_j/a_j of objective coefficient to constraint coefficient for $j = 1, 2, 3$, which determine which single primal variable will take a non-zero value in the optimal basic feasible solution. Depending on the sign of b, we obtain one of two possible dual solutions:

$$u^* = \begin{cases} 6/2 = 3 & \text{if } b > 0 \\ 7/(-7) = -1 & \text{if } b < 0. \end{cases}$$

Thus, the value function associated with this linear optimization problem is as shown in Fig. 18.1. Note that, when $b = 0$, the dual solution is not unique and can take any value between -1 and 3. This set of solutions corresponds to the set of subgradients at the single point of non-differentiability of the value function. This function has one of two gradients for all points that are differentiable, and these gradients are equal to one of the two dual solutions derived above.

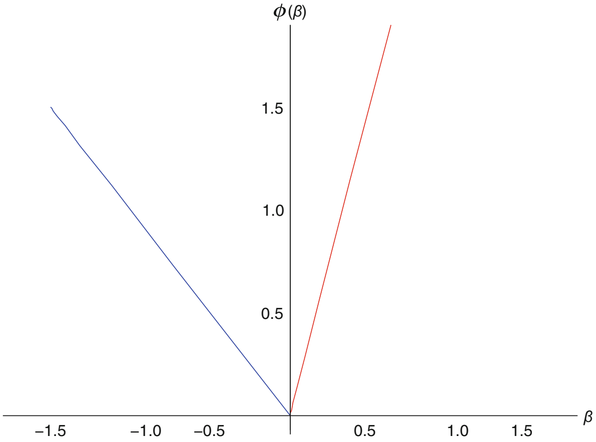

Fig. 18.1 Value function of the LP in the example

18.5.4 Structure of the Value Function

As we mentioned previously, solution methods for (2SMILP) inherently involve the explicit or implicit approximation of several functions, including the value function ϕ in (2SVF) and the risk function Ξ in (RF), which ultimately derives its structure from ϕ. Here, we summarize the results described in the series of papers [70, 71, 75]. Most importantly, the function is piecewise polyhedral, lower semi-continuous, subadditive, and has a discrete structure that is derivative of the structure of two related value functions which we now introduce.

Let y_C, d^2_C, and G^2_C be the parts of each of the vectors/matrices describing the second-stage problem (SS) that are associated with the continuous variables and let y_I, d^2_I, and G^2_I be likewise for the integer variables. The *continuous restriction* (CR) is the LP obtained by dropping the integer variables in the second-stage problem (or equivalently, setting them to zero). This problem has its own value function, defined as

$$\phi_C(\beta) = \min_{y_C \in \mathbb{R}^{n_2 - r_2}_+} \{d^2_C y_C \mid G^2_C y_C \geq \beta\}. \tag{CRVF}$$

On the other hand, if we instead drop the continuous variables from the problem, we can then consider the *integer restriction* (IR), which has value function

$$\phi_I(\beta) = \min_{y_I \in \mathbb{Z}^{r_2}_+} \{d^2_I y_I \mid G^2_I y_I \geq \beta\}. \tag{IRVF}$$

To illustrate how these two functions combine to yield the structure of ϕ and to briefly summarize some of the important results from the study of this function carried out in the aforementioned papers, consider the following simple example of a two-stage stochastic mixed integer linear optimization problem with a single constraint. Note that, in this example, $d^1 = d^2$ and we are again considering the equality-constrained case in order to make the example a bit more interesting.

Value Function of a 2SMILP
We consider the 2SMILP

$$\begin{aligned} \min \quad & \Psi(x_1, x_2) = -3x_1 - 4x_2 + \mathbb{E}[\phi(b^2_\omega - 2x_1 - 0.5x_2)] \\ \text{s.t.} \quad & x_1 \leq 5, x_2 \leq 5 \\ & x_1, x_2 \in \mathbb{R}_+, \end{aligned}$$

where

$$\begin{aligned} \phi(\beta) = \min \quad & 6y_1 + 4y_2 + 3y_3 + 4y_4 + 5y_5 + 7y_6 \\ \text{s.t.} \quad & 2y_1 + 5y_2 - 2y_3 - 2y_4 + 5y_5 + 5y_6 = \beta \\ & y_1, y_2, y_3 \in \mathbb{Z}_+, y_4, y_5, y_6 \in \mathbb{R}_+, \end{aligned}$$

with $\Omega = \{1, 2\}$, $b^2_1 = 6$, and $b^2_2 = 12$. Figure 18.2 shows the objective function Ψ and the second-stage value function ϕ for this example.

Examining Fig. 18.2, it appears that ϕ is comprised of a collection of translations of ϕ_C, each of which has a structure that is similar to the value function of the LP in the Example on page 533. At points where ϕ is differentiable, the gradient always corresponds to one of the two solutions to the dual of the continuous restriction (precisely as in the Example on page 533, dual solutions are the ratios of objective function coefficients to constraint coefficients for the continuous variables), which are in turn also the gradients of ϕ_C. The so-called points of *strict local convexity* are the points of non-differentiability that are the extreme points of the epigraphs of the translations of ϕ_C and are determined by the solutions to the integer restriction. In particular, they correspond to points at which ϕ_I and ϕ are coincident. For instance, observe that, in the example, $\phi_I(5) = \phi(5) = 4$.

Finally, we can observe in Fig. 18.2 how the structure of ϕ translates into that of Ψ.

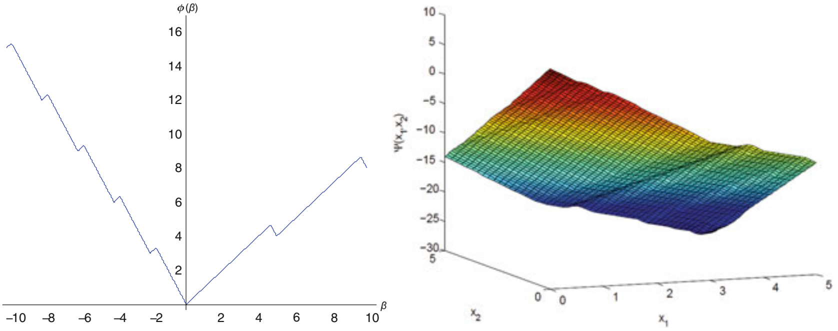

Fig. 18.2 Illustration of the functions from the example, with the second-stage value function $\phi(\beta)$ (left) and the objective function $\Psi(x_1, x_2)$ (right)

Although the case illustrated in the example is of a single constraint, these properties can be made rigorous and do generalize to higher dimension with roughly the same intuition. The general principle is that the value function of an MILP is the minimum of translations of the value function of the continuous restriction ϕ_C, where the points of translation (the aforementioned points of *strict local convexity*) are determined by the value function of the integer restriction ϕ_I.

Theorem 18.5.3 ([75]) *Under the assumption that* $\{\beta \in \mathbb{R}^{m_2} \mid \phi_I(\beta) < \infty\}$ *is compact and* $\mathrm{epi}(\phi_C)$ *is pointed, there exists a finite set* $\mathcal{S} \subseteq Y$ *such that*

$$\phi(\beta) = \min_{y_I \in \mathcal{S}} \{d_I^2 y_I + \phi_C(\beta - G_I^2 y_I)\}. \qquad \triangle$$

Under the assumptions of the theorem, this result provides a finite description of ϕ.

18.5.5 Approximating the Value Function

Constructing functions that bound the value function of an MILP is an important part of solution methods for both traditional single-stage optimization problems and their multistage counterparts. Functions bounding the value function from below are exactly the *dual functions* defined earlier and arise from *relaxations* of the original problem (such as the LP relaxation, for instance). They are naturally obtained as by-products of common solution algorithms, such as branch and bound, which itself works by iteratively strengthening a given relaxation and produces a dual proof of optimality.

Functions that bound the value function from above are the *primal functions* defined earlier and can be obtained by considering the value function of a *restriction* of the original problem. While it is a little less obvious how to obtain such functions in general (solution algorithms generally do not produce practically useful primal functions), they can be obtained by taking the minimum over the value functions of restrictions obtained by fixing the values of integer variables, as we describe below.

Both primal and dual functions can be iteratively improved by producing new such functions and combining them with existing ones by taking the maximum over several bounding functions in the dual case or the minimum over several bounding functions in the primal case. When such functions are iteratively constructed to be strong at different right-hand side values, such as when solving a sequence of instances with different right-hand sides, such a technique can yield an approximation with good fidelity across a larger part of the domain than any singly constructed function could—this is, in fact, the principle implicitly behind the algorithms we describe in Sect. 18.7.1.

18.5.5.1 Dual Functions from Branch and Bound

Dual functions can be obtained from most practical solution algorithms for solving the MILP associated with (2SVF) with input $\beta = b^2_\omega - A^2_\omega x$, i.e., for computing $\phi(b^2_\omega - A^2_\omega x)$. This is because their existence is (at least implicitly) the very source of the proof of optimality produced by such algorithms. To illustrate, we show how a dual function can be obtained as the by-product of the branch-and-bound algorithm.

Branch and bound is an algorithm that searches the feasible region by partitioning it and then recursively solving the resulting subproblems. Implemented naively, this results in an inefficient complete enumeration, but this potential inefficiency is avoided by utilizing lower and upper bounds computed for each subproblem to intelligently “prune” the search. The recursive partitioning process can be envisioned as a process of searching a rooted tree, each of whose nodes corresponds to a subproblem. Although it is not usually described this way, the branch-and-bound algorithm can be interpreted as constructing a function feasible to (MILPD), thus producing not only a solution but also a dual proof of its optimality.

To understand this interpretation, suppose we evaluate $\phi(b)$ for $b \in \mathbb{R}^{m_2}$ by solving the associated MILP using a standard branch-and-bound algorithm with branching done on elementary (a.k.a. variable) disjunctions of type $y_j \leq \pi_0 \vee y_j \geq \pi_0 + 1$ for some $j \in \{1, \ldots, r_2\}$ and $\pi_0 \in \mathbb{Z}$. Because the individual subproblems in the branch-and-bound tree differ only by the bounds on the variables, it will be convenient to assume that all integer variables have finite initial lower and upper bounds denoted by the vectors $l, u \in \mathbb{Z}^{r_2}$ (such bounds exist by Assumption 1, even if they are not part of the formulation). In this case, we have that

$$\mathcal{P}_2(\beta) = \{y \in \mathbb{R}^{n_2} \mid G^2 y \geq \beta, y_I \geq l, -y_I \geq -u\}.$$

The solution of (SS) by branch and bound for right-hand side b yields a branch-and-bound tree whose leaves, contained in the set $\mathcal{T}$, are each associated with the subproblem

$$\min\{d^2 y \mid y \in \mathcal{P}^t_2(b) \cap Y\},$$

where

$$\mathcal{P}_2^t(\beta) = \{y \in \mathbb{R}^{n_2} \mid G^2 y \geq \beta, y_I \geq l^t, -y_I \geq -u^t\}$$

is the parametric form of the polytope containing the feasible points associated with subproblem $t \in \mathcal{T}$, with l^t, $u^t \in \mathbb{Z}_+^{r_2}$ being the modified bounds on the integer variables imposed by branching.

Assuming that no pruning took place, the validity of the overall method comes from the fact that valid methods of branching ensure that

$$\bigcup_{t \in \mathcal{T}} \mathcal{P}_2^t(b) \cap Y = \mathcal{P}_2(b) \cap Y,$$

so that the feasible regions of the leaf nodes constitute a partition of the original feasible region. This partition can be thought of as a single logical disjunction that serves to strengthen the original LP relaxation. The proof of optimality that branch and bound produces derives from global lower and upper bounds derived from local bounds associated with each node $t \in \mathcal{T}$. We denote by L^t and U^t, respectively, the lower and upper bounds on the optimal solution value of the subproblem, where

$$L^t = \phi_{\mathrm{LP}}^t(b), \quad U^t = d^2 y^t,$$

in which ϕ_{LP}^t is as defined in (NVF) below and $y^t \in Y$ is the best solution known for subproblem $t \in \mathcal{T}$ ($U^t = \infty$ if no solution is known). Assuming (SS) is solved to optimality and $y^* \in Y$ is an optimal solution, we must have

$$L := \min_{t \in \mathcal{T}} L^t = d^2 y^* = \min_{t \in \mathcal{T}} U^t =: U,$$

where L and U are the global lower and upper bounds.

From the information encoded in the branch-and-bound tree, the overall dual function can be constructed by deriving a parametric form of the lower bound, combining dual functions for the individual subproblems in set $\mathcal{T}$. For this purpose, we define the value function

$$\phi_{\mathrm{LP}}(\beta, \lambda, \nu) = \min\{d^2 y \mid y \in \mathcal{P}_2(\beta), \lambda \leq y_I \leq \nu, y_C \geq 0\} \qquad \text{(PNVF)}$$

of a generic LP relaxation, which captures the bounds on the integer variables as also being parametric. Using (PNVF), the value function of the LP relaxation associated with a particular node $t \in \mathcal{T}$ (only parametric in the original right-hand side) can be obtained as

$$\phi_{\mathrm{LP}}^t(\beta) = \phi_{\mathrm{LP}}(\beta, l^t, u^t). \qquad \text{(NVF)}$$

For all $t \in \mathcal{T}$ such that $\phi^t_{\text{LP}}(b) < \infty$, let $(v^t, \underline{v}^t, \bar{v}^t)$ be an optimal solution to the dual of the LP relaxation at node $t \in \mathcal{T}$, where v^t, $\underline{v}^t$, and $\bar{v}^t$ are, respectively, the dual variables associated with the original inequality constraints, the lower bounds on integer variables, and the upper bounds on integer variables. Then, by LP duality we have that

$$\phi^t_{\text{LP}}(b) = v^t b + \underline{v}^t l^t - \bar{v}^t u^t.$$

For each node $t \in \mathcal{T}$ for which $\phi^t_{\text{LP}}(b) = \infty$ (the associated subproblem is infeasible), we instead let $(v^t, \underline{v}^t, \bar{v}^t)$ be a dual feasible solution that provides a finite bound exceeding U (such can be found by, e.g., adding some multiple of the dual ray that proves infeasibility to the feasible dual solution found in the final iteration of the simplex algorithm).

Finally, from the above we have that

$$\underline{\phi}^t_{\text{LP}}(\beta) = v^t \beta + \underline{v}^t l^t - \bar{v}^t u^t \tag{NDF}$$

is a dual function w.r.t. the LP relaxation at node t that is strong at b. Finally, we can combine these individual dual functions together to obtain

$$\underline{\phi}^{\mathcal{T}}(\beta) = \min_{t \in \mathcal{T}} \underline{\phi}^t_{\text{LP}}(\beta) = \min_{t \in \mathcal{T}} \{v^t \beta + \underline{v}^t l^t - \bar{v}^t u^t\}, \tag{BB-DF}$$

a dual function for the second-stage problem yielded by the tree $\mathcal{T}$ and which is also *strong* at the right-hand side b, i.e., $\underline{\phi}^{\mathcal{T}}(b) = \phi(b)$.

In principle, a stronger dual function can be obtained by replacing the single linear dual function (which is strong at b for the LP relaxation) associated with each subproblem above by its full value function ϕ^t_{LP} to obtain

$$\underline{\phi}^{\mathcal{T}}_*(\beta) = \min_{t \in \mathcal{T}} \phi^t_{\text{LP}}(\beta). \tag{BB-DF-BIS}$$

In practice, constructing a complete description of $\underline{\phi}^{\mathcal{T}}_*$ is not practical (even evaluating it for a given β requires the solution of $|\mathcal{T}|$ LPs). We can instead construct a function that bounds *it* from below (and hence is also a dual function for the original problem) by exploiting the entire collection of dual solutions arising during the solution process. For example, let

$$\underline{\phi}_{\text{LP}}(\beta, \lambda, \nu) = \max_{t \in \mathcal{T}} \{v^t \beta + \underline{v}^t \lambda - \bar{v}^t \nu\},$$

which consists of an approximation of the full value function ϕ_{LP} using the optimal dual solutions at each leaf node. Replacing $\underline{\phi}^t_{\text{LP}}(\beta)$ with $\underline{\phi}_{\text{LP}}(\beta, l^t, u^t)$ in (BB-DF) results in a potentially stronger but still practical dual function. Of course, it is also possible to add dual solutions found at non-leaf nodes, as well as other suboptimal dual solutions arising during the solution process, but there is an obvious trade-off between strength and tractability. More details on this methodology are contained in [71, 74].

18.5.5.2 Iterative Refinement

In iterative algorithms such as those we introduce in Sect. 18.7, the single dual function (BB-DF) we get by evaluating the value function for one right-hand side can be iteratively augmented by taking the maximum over a sequence of similarly derived dual functions. Taking this basic idea a step further, [70, 110, 111] developed methods of warm starting the solution process of an MILP. Such methods may serve to enhance tractability, though this is still an active area of research. When evaluating ϕ repeatedly for different values in its domain, we do not need to solve each instance from scratch—it is possible to use the tree resulting from a previous solve as a starting point and simply further refine it to obtain a dual function that remains strong for the previous right-hand side of interest and is made to be strong for a new right-hand side.

Hassanzadeh and Ralphs [74] shows how to use this iterative-refinement approach to construct a lower approximation of the value function of an MILP in the context of a Benders-like algorithm for two-stage stochastic optimization within a single branch-and-bound tree. In fact, with enough sampling this method can be used to construct a single tree whose associated dual function is strong at every right-hand side (provided the set of feasible right-hand sides is finite). The following example illustrates the concept of using this iterative refinement approach in approximating the value function of the example on page 535.

Approximating the Value Function

Consider the value function of the example on page 535, reported in Fig. 18.2. The sequence of evaluations of the value function in this example are the ones arising from first-stage solutions generated by solving the master problem in a generalized Benders algorithm, such as the one described in Sect. 18.7.1. Here, we only illustrate the way in which the dual function is iteratively refined in each step.

We first evaluate $\phi(3.5)$ by branch and bound. Figure 18.3 shows both the tree obtained (far left) and the value function itself (in blue). The dual function arises from the solution to the dual of the LP relaxation in each of the nodes in the branch-and-bound tree. We exhibit the values of the dual solution for each node in the tree in Table 18.2. Explicit upper and lower bounds were added with upper bounds initially taking on a large value M, representing infinity. Note that the dual values associated with the bound constraints are actually nothing more than the reduced costs associated with the variables.

The dual function associated with this first branch-and-bound tree is the minimum of the two linear functions shown in Fig. 18.3 in green and labeled as “Node 1” and “Node 2.” Formally, this dual function is

$$\underline{\phi}^{\mathcal{T}_1} = \min\{\underline{\phi}^1_{\mathrm{LP}}, \underline{\phi}^2_{\mathrm{LP}}\},$$

(continued)

where the nodal dual functions for the three nodes are

$$\underline{\phi}^0_{\mathrm{LP}}(\beta) = 0.8\beta$$
$$\underline{\phi}^1_{\mathrm{LP}}(\beta) = \beta$$
$$\underline{\phi}^2_{\mathrm{LP}}(\beta) = -1.5\beta + 11.5.$$

In other words, we have $v_0^1 = 1$ (the value of the dual variable associated with the single equality constraint in Node 1), while $\underline{v}^1 l^1 - \bar{v}^1 u^1 = 0$ (this is the contribution from the dual value corresponding to the bound constraints, which we take to be a constant here, as in (NDF)). Similarly, $v_0^2 = -1.5$ and $\underline{v}^1 l^1 - \bar{v}^1 u^1 = 11.5$. The dual function $\underline{\phi}^{\mathcal{T}_1}_{\mathrm{LP}}$ is strong in the interval [0, 5], but yields a weaker lower bound outside this interval. If we subsequently evaluate the right-hand side 9.5, we see that

$$\underline{\phi}^{\mathcal{T}_1}_{\mathrm{LP}}(9.5) = \min\{9.5, -2.75\} = -2.75 \neq \phi(9.5) = 8.5.$$

To obtain a strong dual function for the new right-hand side, we identify that node 2 is the node whose bound needs to be improved by further refining the tree by branching (this is the linear function yielding the bound in this part of the domain). By further partitioning the subproblem associated with node 2, we obtain the tree pictured to the right of the first tree in Fig. 18.3. We obtain the dual function

$$\underline{\phi}^{\mathcal{T}_2}_{\mathrm{LP}} = \min\{\underline{\phi}^1_{\mathrm{LP}}, \underline{\phi}^3_{\mathrm{LP}}, \underline{\phi}^4_{\mathrm{LP}}\},$$

which is strong at the right-hand side 9.5.

Note that this new function is no longer strong at the initial right-hand side of 3.5. To ensure that this single dual function remains strong for all previously evaluated right-hand sides, we must take the max over the collection of dual functions found at each iteration. This function is still obtained from the single tree, but we are effectively strengthening the leaf node dual functions by taking the max over all dual solutions arising on the path from the root subproblem (this is still a bound on the optimal solution value to the LP relaxation). In this case, we get the strengthened function

$$\min\left\{\max\left\{\underline{\phi}^1_{\mathrm{LP}}, \underline{\phi}^0_{\mathrm{LP}}\right\}, \max\left\{\underline{\phi}^3_{\mathrm{LP}}, \underline{\phi}^2_{\mathrm{LP}}, \underline{\phi}^0_{\mathrm{LP}}\right\}, \max\left\{\underline{\phi}^4_{\mathrm{LP}}, \underline{\phi}^2_{\mathrm{LP}}, \underline{\phi}^0_{\mathrm{LP}}\right\}\right\}.$$

This can be seen as an approximation of $\phi^{\mathcal{T}}_*$ by replacing the full value function at each node with an approximation made of just the dual solutions arising on the path to the root node.

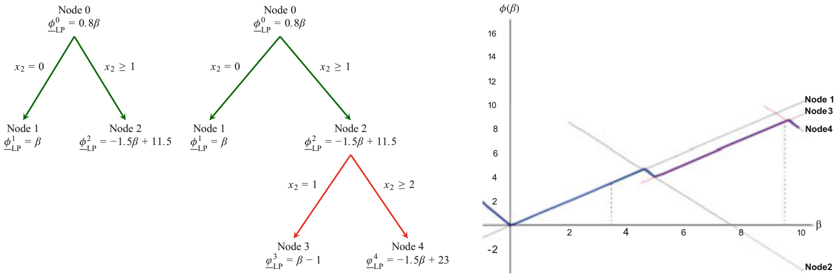
Node 0
$\underline{\phi}^0_{LP} = 0.8\beta$
$x_2 = 0$
$x_2 \geq 1$
Node 1
$\underline{\phi}^1_{LP} = \beta$
Node 2
$\underline{\phi}^2_{LP} = -1.5\beta + 11.5$
Node 0
$\underline{\phi}^0_{LP} = 0.8\beta$
$x_2 = 0$
$x_2 \geq 1$
Node 1
$\underline{\phi}^1_{LP} = \beta$
Node 2
$\underline{\phi}^2_{LP} = -1.5\beta + 11.5$
$x_2 = 1$
$x_2 \geq 2$
Node 3
$\underline{\varphi}^3_{LP} = \beta - 1$
Node 4
$\underline{\varphi}^4_{LP} = -1.5\beta + 23$
$\phi(\beta)$
16
14
12
10
8
6
4
2
-2
2
4
6
8
10
β
Node 1
Node3
Node4
Node2

Fig. 18.3 Approximation of the value function of the 2SMILP instance in the example on page 535

Table 18.2 Dual solutions for each node in the branch-and-bound tree

t	v^t	$\underline{v}^t$						$\bar{v}^t$					
0	0.8	4.4	0.0	4.6	5.6	1.0	3.0	0.0	0.0	0.0	0.0	0.0	0.0
1	1.0	4.0	0.0	5.0	6.0	0.0	2.0	0.0	−1.0	0.0	0.0	0.0	0.0
2	−1.5	9.0	11.5	0.0	1.0	12.5	14.5	0.0	0.0	0.0	0.0	0.0	0.0

18.5.5.3 Primal Functions

Upper approximations of ϕ can be obtained by considering the value functions of the second-stage problem (SS) obtained by fixing variables. For example, consider the *integer-fixing value function*

$$\bar{\phi}_{\hat{y}}(\beta) = d^2_I \hat{y}_I + \phi_C(\beta - A^2_{\omega I}\hat{y}_I) \qquad \text{(IFVF)}$$

obtained by fixing the integer part $\hat{y}_I \in \mathbb{Z}^{r_2}$ to be equal to that from some previously found second-stage solution $\hat{y} \in Y$, where ϕ_C is as defined in (CRVF). Then, we have $\bar{\phi}_{\hat{y}_I}(\beta) \geq \phi(\beta)$ for all $\beta \in \mathbb{R}^{m_2}$. If $\hat{y}$ is an optimal solution to the second-stage problem with respect to a given first-stage solution $x \in \mathcal{P}_1 \cap X$ and a given realized value of ω, then we have $\hat{y} \in \text{argmin}_{y \in \mathcal{P}_2(b^2_\omega - A^2_\omega x) \cap Y}\, d^2 y$ and $\bar{\phi}_{\hat{y}}$ is strong at $\beta = b^2_\omega - A^2_\omega x$.

In a fashion similar to a cutting plane method, we can iteratively improve the global upper bounding function by taking the minimum of all bounding functions of the form (IFVF) found so far, i.e.,

$$\bar{\phi}(\beta) = \min_{y \in \mathcal{R}} \bar{\phi}_y(\beta),$$

where $\mathcal{R}$ is the set of all second-stage solutions that have been found when evaluating $\phi(\beta)$ for different values of β.

A pictorial example of this type of upper bounding function is shown in Fig. 18.4, where each of the labeled cones shown is the value function of a restriction of the original MILP. The upper bounding function is the minimum over all of these cones.

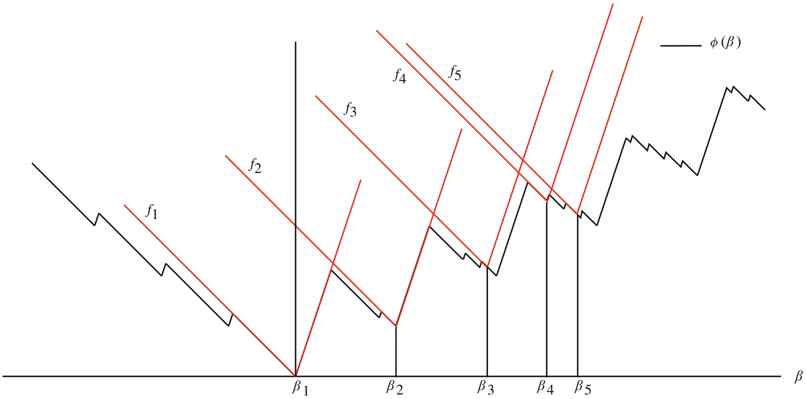

Fig. 18.4 Upper bounding functions obtained at right-hand sides β_i, $i = 1, \ldots, 5$

18.5.6 Reaction and Risk Functions

Because it is particularly relevant in the present context, we also now introduce a function known as the *second-stage (optimistic) reaction function*. This function is closely related to the risk function but its input is a second stage right-hand side, rather than a first-stage solution. Although this function, like the second-stage value function ϕ, takes a right-hand side β as its input, it can nevertheless be used to evaluate a first-stage solution $x \in X$ in scenario $\omega \in \Omega$ by evaluating it with respect to $\beta_\omega(x)$. The function is defined as

$$\rho(\beta) = \inf\left\{d^1 y \mid y \in \operatorname{argmin}\{d^2 y \mid y \in \mathcal{P}_2(\beta) \cap Y\}\right\}, \tag{ReF}$$

for $\beta \in \mathbb{R}^{m_2}$. Note that, although the evaluation of this function appears to require solving a bilevel optimization problem, its evaluation is actually equivalent to solving a lexicographic optimization problem, a somewhat easier computational task.

The importance of the function ρ is to enable us to see that, for (2SMILP), the scenario risk functions Ξ_ω defined in (2SRF) are not in fact completely independent functions but, rather, are connected, since

$$\Xi_\omega(x) = \rho(\beta_\omega(x)).$$

Thus, these functions only differ from each other in the affine map $\beta_\omega(x) = b^2_\omega - A^2_\omega x$ that is applied to x.

The structure of the functions ρ and Ξ derives from that of ϕ and can be understood through a somewhat more involved application of the same principles used to derive the function ϕ. Their structure, though combinatorially much more complex, is nevertheless also piecewise polyhedral. Approximations of Ξ can be derived easily from approximation of ρ in a straightforward way, since $\Xi = \mathbb{E}[\Xi_\omega]$ and the scenario risk functions are themselves defined in terms of ρ, as discussed earlier. The approximation of ρ is quite involved, but it can be obtained by methods that are natural generalizations of those used to approximate ϕ. The main challenge is that the evaluation of $\rho(\beta)$ for a particular value of β itself reduces to the solution of a lexicographic optimization problem, which in turn requires knowledge of ϕ. We may approximate ρ by repeatedly evaluating it, extracting primal and dual information from the solution algorithm as we do with ϕ, but this requires repeatedly evaluating ϕ, which is itself expensive.

In the algorithms we discuss in Sect. 18.7, we approach this difficulty by constructing a single approximation of ϕ in tandem with the approximation of ρ. We need only ensure that the approximation of ϕ is guaranteed to be strong (i.e., equal to the value function) exactly in the region needed to properly evaluate ρ. The result is an approximation of ρ that is strong in the region of a given right-hand side and this is exactly what is needed for a Benders-type algorithm to converge. More details regarding the Benders-type algorithm are contained next in Sect. 18.7. Further details on the structure of and methods for approximating ρ and Ξ can be found in [23], which describes a Benders-type algorithm for solving (2SMILP).

18.5.7 Optimality Conditions

To solidify the connection between the notion of duality described in this section and captured in the dual problem (MILPD), we end this section by formally stating both the weak and strong duality properties arising from this theory. These properties are a proper generalization of the well-known ones from linear optimization and can be used to derive optimality conditions that generalize those from LP duality. These are, in turn, the conditions that can be used to derive the reformulations presented in Sect. 18.6.

Theorem 18.5.4 (*Weak Duality*) *If $F \in \Upsilon^{m_2}$ is feasible for* (MILPD) *and $y \in \mathcal{P}_2(b) \cap Y$, then $F(b) \leq d^2 y$.* △

Theorem 18.5.5 (*Strong Duality*) *Let $b \in \mathbb{R}^{m_2}$ be such that $\phi(b) < \infty$. Then, there exists both $F \in \Upsilon^{m_2}$ that is feasible for* (MILPD) *and $y^* \in \mathcal{P}_2(b) \cap Y$ such that $F(b) = \phi(b) = d^2 y^*$.* △

The form of the dual (MILPD) makes these properties rather self-evident, but Theorem 18.5.5 nevertheless yields optimality conditions that are useful in practice. In particular, the dual functions arising from branch-and-bound algorithms that were described earlier in Sect. 18.5.5.1 are the strong dual functions that provide the optimality certificates for the solutions produced by such algorithms and are the basis on which the algorithms described in Sect. 18.7.1 are developed.

18.6 Reformulations

A crucial step in deriving solution algorithms is to consider some conceptual reformulations that underlie the problems under study, each of which suggests a particular algorithmic strategy. These formulations are heavily based on the duality theory and methodology in the previous section, as we better clarify in the following. In all cases, the goal is to derive, from some initial problem description, a formulation that is, at least in principle, a single-stage mathematical optimization problem that can be tackled with (possibly generalized versions of) the standard algorithmic approaches used for solving mathematical optimization problems. In this case, as in the solution of single-stage MILPs, the main tools are cutting plane methods (branch and cut) and decomposition methods (Benders' algorithm and the exploitation of the block structure using a Dantzig-Wolfe decomposition).

18.6.1 Value-Function (Optimality-Based) Reformulation

The first reformulation we describe is a variation on (2SMILP-Alt), the standard formulation used in most of the bilevel optimization literature. This formulation introduces the second-stage variables explicitly and formulates the problem in the form of a classical mathematical optimization problem using a technique that is standard—replacing the requirement that the solution to the second-stage problem be optimal with explicit optimality conditions. To achieve this, we introduce a constraint involving the value function ϕ, as well as the second-stage feasibility conditions. This is roughly equivalent to imposing primal and dual feasibility along with equality of the primal and dual objectives in the linear optimization case (the constraint involving the value function must be satisfied at equality, though it is stated as an inequality). The formulation is as follows.

$$
\begin{aligned}
\min \quad & cx + \sum_{\omega\in\Omega} p_\omega d^1 y^\omega && && \\
\text{s.t.} \quad & A^1 x \geq b^1 && && \text{(VFRa)}\\
& G^2 y^\omega \geq b^2_\omega - A^2_\omega x && \forall \omega \in \Omega && \text{(VFRb)}\\
& d^2 y^\omega \leq \phi(b^2_\omega - A^2_\omega x) && \forall \omega \in \Omega && \text{(VFRc)}\\
& x \in X && && \text{(VFRd)}\\
& y^\omega \in Y && \forall \omega \in \Omega. && \text{(VFRe)}
\end{aligned}
$$

It is clear that this formulation cannot be constructed explicitly, but rather, must be solved by the iterative approximation of the constraints involving the value function (which we refer to as the second-stage optimality constraints). This reformulation suggests a family of methods described in Sect. 18.7 in which we replace ϕ with a primal function $\underline{\phi}$, as defined in Definition 18.5.2.

Notice that, when $r_2 = 0$, so that the second-stage problem (SS) is a linear optimization problem, we can exploit the fact that the optimality conditions for this problem involve linear functions. This allows for, in essence, substituting for ϕ the objective function of the classical LP dual of (SS), after introducing the corresponding variables and constraints. This, overall, leads to a tractable *primal-dual* reformulation—the technique is applied, for instance, in [40]. The alternative idea of, rather than the dual of (SS), introducing its KKT conditions, is arguably more popular and has been often exploited in a number of "classical" works on mixed integer bilevel optimization problems, including, among others [93]. Note, however, that while there is an analog of this reformulation that applies in the setting of (2SMILP) (see [52]), it has so far proved not to be practical and, therefore, we will not present any algorithms for its solution in Sect. 18.7.

18.6.2 Risk-Function (Projection-Based) Reformulation

The next reformulation we consider exploits the finiteness of Ω and avoids introducing the second-stage variables explicitly. It reads as follows.

$$\begin{aligned} \min \quad & c^1 x + \sum_{\omega \in \Omega} p_\omega z^\omega \\ \text{s.t.} \quad & z^\omega \geq \Xi_\omega(x) && \omega \in \Omega \\ & x \in \mathcal{P}_1 \cap X \\ & z^\omega \in \mathbb{R} && \omega \in \Omega. \end{aligned} \tag{RFR}$$

This reformulation mirrors the original formulation implicitly adopted when we first defined (2SMILP), in which the second-stage variables are not (explicitly) present. However, we can also interpret it as a projection onto the X-space of the value-function reformulation described in the previous section. In fact, it is not hard to see that the set $\{x \in \mathcal{P}_1 \mid \Xi(x) < \infty\}$ is exactly $\mathcal{F}^1$ (the projection of the feasible region onto the space of the first-stage variables) as defined in (FS-FR). As such, this formulation is a natural basis for a Benders-type algorithm that we describe in Sect. 18.7.1, in which we replace Ξ with an under-estimator to obtain a *master problem* which is then iteratively improved until convergence.

18.6.3 Polyhedral (Convexification-Based) Reformulation

An apparently unrelated reformulation generalizes the notion of *convexification* used heavily in the polyhedral theory that underlies the solution methodology of standard MILPs. Convexification considers the following conceptual reformulation:

$$\begin{aligned} \min \quad & cx + \sum_{\omega\in\Omega} p_\omega d^1 y^\omega \\ \text{s.t.} \quad & (x, y^\omega) \in \operatorname{conv}(\mathcal{F}^\omega) \quad \forall\omega \in \Omega, \end{aligned} \tag{POLY-R}$$

where $\mathcal{F}^\omega$ is the feasible region under scenario ω, defined as in (FR). Under our earlier assumptions, the convex hull of $\mathcal{F}^\omega$ is a polyhedron whose extreme points are members of $\mathcal{F}^\omega$. Thus, due to the linearity of the objective function, we can w.l.o.g. replace the requirement that $(x, y^\omega) \in \mathcal{F}^\omega$ with the requirement that $(x, y^\omega) \in \operatorname{conv}(\mathcal{F}^\omega)$, thereby convexifying the feasible region.

With this reformulation, we have transformed the requirement that the second-stage solution be optimal for the parameterized second-stage problem (SS) into a requirement that the combination of first- and second-stage solutions lie in a polyhedral feasible region. This reformulation suggests a different class of algorithms based on the dynamic generation of valid inequalities, such as those so successfully employed in the case of MILPs. We describe an algorithm of such class in Sect. 18.7.2.

18.6.4 Deterministic Equivalent (Decomposition-Based) Reformulation

Finally, we remark that the finiteness of Ω allows for solving the problem via a block-angular reformulation based on the formulation (2SMILP-Alt) presented earlier, which is in the spirit of the so-called deterministic equivalent reformulation used in two-stage stochastic optimization. This renders the stochastic problem as a deterministic MIBLP, which can then be solved via standard methods for that case with the requisite further reformulations (of course, exploiting the block structure of the resulting matrices). For details, see [126].

18.7 Algorithmic Approaches

We now summarize a range of methodologies that arise naturally from the reformulations of the previous section. Any practical method of solving (2SMILP) must have as a fundamental step the evaluation of $\phi(\beta)$ for certain fixed values of $\beta \in \mathbb{R}^{m_2}$, an operation which can be challenging in itself, since the corresponding problem is NP-hard. From the evaluation of ϕ, both primal and dual information is obtained, which can be used to build approximations. While some methods explicitly build such approximations, other methods do it only implicitly. In all cases, information about the value function that is built up through repeated evaluations can be exploited.

Similarly, in the dual methods that we describe below, the function Ξ is also evaluated for various values of $x \in X$ (or rather the function ρ) and, similarly, approximations of this function can be built from primal and dual information obtained during its evaluation. In order to develop computationally tractable methods, a key aspect is to limit the number of such function evaluations as much as possible and to exploit to the maximum extent possible the information generated as a by-product of these single function evaluations.

18.7.1 Decomposition Methods

Decomposition methods are based on generalizations of Benders' original method of decomposing a given mathematical optimization problem by splitting the variables into two subsets and forming a master problem by projecting out one subset. More concretely, we are simply solving a reformulation of the form (RFR).

18.7.1.1 Continuous Problems

For illustration, we consider the simplest case in which we have only continuous variables in both stages ($r_1 = r_2 = 0$) and $d^1 = d^2$. Since the first- and second-stage objectives are the same in this case, the full problem is nothing more than a standard linear optimization problem, but Benders' approach nevertheless applies when either fixing the first stage variables results in a more tractable second-stage LP (such as a min-cost flow problem). In the Benders approach, we (conceptually) rewrite the LP as

$$\min \left\{ cx + \sum_{\omega \in \Omega} p_\omega \phi(\beta^\omega(x)) \;\middle|\; x \in \mathcal{P}_1 \right\},$$

where ϕ is the value function (2SVF). Note that, because $\Xi(x) = \sum_{\omega \in \Omega} p_\omega \phi$ $(\beta^\omega(x))$, this is just a simplification of the original formulation implicitly adopted when we first defined (2SMILP). As we observed earlier, the value function in the LP case is the maximum of linear functions associated with the dual solutions. Recalling that we can restrict the description to only the extreme points of the dual feasible region, we can further rewrite the LP as

$$\min \left\{ cx + \sum_{\omega \in \Omega} p_\omega z^\omega \;\middle|\; \begin{array}{ll} x \in \mathcal{P}_1 & \\ z^\omega \geq u\beta^\omega(x), & u \in \mathcal{E}, \omega \in \Omega \\ z^\omega \in \mathbb{R}, & \omega \in \Omega \end{array} \right\}, \tag{LP}$$

where $\mathcal{E}$ is the set of such extreme points of the dual of the second-stage LP (which we assumed to be bounded and nonempty). Thus, the linear constraints involving the variable z^ω (along with the fact that z^ω is minimized) are precisely equivalent to requiring $z^\omega = \phi(b^2_\omega - A^2_\omega x)$, so this reformulation is exactly the formulation (RFR) specialized to this case.

A straightforward solution approach is then to solve (LP) by a cutting plane algorithm, which results in the classical L-shaped method for solving (continuous) stochastic linear optimization problems [130]. From the point of view we have taken in this article, this method can be interpreted as one that approximates the value function from below as the maximum of the strong dual functions generated in each iteration. The strong dual functions arise from the solutions to the dual of the second-stage problem and yield what are typically called *Benders cuts* (inequalities of the form imposed in (LP)). The Benders approach is then to iteratively improve this approximation until convergence.

The case $d^1 \neq d^2$ is more complex. The epigraph of the value function of the second-stage problem is no longer necessarily a polyhedral cone, and the function itself is no longer necessarily convex. Formulating the equivalent of (LP) thus requires integer variables. Alternative formulations using the related complementary slackness optimality conditions are also commonly used (see [37]).

18.7.1.2 Discrete Problems

For the case in which there are integer variables, the approach just described can be applied by simply replacing the linear strong dual functions (Benders' cuts) with strong under-estimators of the risk function constructed from dual functions arising from solution algorithms for the second-stage problem, such as those based on branch and bound described in Sect. 18.5.5.1. In this approach, we work directly with the reformulation (RFR), employing the generalized Benders-type algorithm summarized in Fig. 18.5 and a master problem defined as follows.

$$
\begin{aligned}
\min \quad & c^1 x + \sum_{\omega\in\Omega} p_\omega z^\omega \\
\text{s.t.} \quad & z^\omega \geq \underline{\Xi}_\omega(x) && \omega \in \Omega \\
& x \in \mathcal{P}_1 \\
& z^\omega \in \mathbb{R} && \omega \in \Omega .
\end{aligned}
\tag{MASTER}
$$

When $d^1 = d^2$, the approximation of the scenario risk function and of the risk function itself reduces to the direct approximation of the second-stage value function, and the algorithm can be described rather succinctly. A basic version was originally proposed as the integer L-shaped algorithm for two-stage stochastic optimization problems with integer recourse by Laporte and Louveaux [94] and Carøe and Tind [29]. The version based on dual functions from branch and bound that we describe here is described in [74].

```
Step 0. Initialize $k \leftarrow 1$, $\underline{\Xi}^0_\omega(x) = -\infty$ for all $x \in \mathbb{Q}^{n_1}$, $\omega \in \Omega$.

Step 1. Solve the Master Problem
  a) Set $\underline{\Xi}_\omega = \max_{i=0,\ldots,k-1} \underline{\Xi}^i_\omega$ for $\omega \in \Omega$.
  b) Solve (MASTER) to obtain an optimal solution $(x^k, \{z^\omega_k\}_{\omega \in \Omega})$.

Step 2. Solve the Subproblem
  a) Evaluate $\Xi_\omega(x^k)$ to obtain an optimal solution $y^{\omega,k}$ for $\omega \in \Omega$ and the strong under-
     estimator $\underline{\Xi}^k_\omega$.
  b) Termination check: Is $z^\omega_k = d^1 y^{\omega,k}$ for $\omega \in \Omega$?
     1. If yes, STOP. $x^k$ is an optimal solution to (RFR).
     2. If no, set $k \leftarrow k + 1$ and go to Step 1.
```

Fig. 18.5 Generalized Benders algorithm for solving 2SMILPs

To briefly summarize, as in the LP case, we rewrite (2SMILP) as

$$\min \left\{ cx + \sum_{\omega \in \Omega} p_\omega z^\omega \;\middle|\; \begin{array}{ll} x \in \mathcal{P}_1 \cap X & \\ z^\omega \geq \phi(\beta^\omega(x)) & \omega \in \Omega \\ z^\omega \in \mathbb{R} & \omega \in \Omega \end{array} \right\}.$$

By replacing ϕ with the maximum of a set $\mathcal{G}^\omega$ of dual functions associated with scenario $\omega \in \Omega$ (alternatively, we can employ one universal set of dual functions, as indicated in (LP) above), we obtain a convergent Benders-like algorithm based on iteratively solving a master problem of the form

$$\min \left\{ cx + \sum_{\omega \in \Omega} p_\omega z^\omega \;\middle|\; \begin{array}{ll} x \in \mathcal{P}_1 \cap X & \\ z^\omega \geq F(\beta^\omega(x)) & F \in \mathcal{G}^\omega, \omega \in \Omega \\ z^\omega \in \mathbb{R} & \omega \in \Omega \end{array} \right\},$$

which generalizes (LP). The key to making this approach work in practice is that the dual functions we need be easily available as a by-product of evaluating the second-stage value function ϕ for a fixed value of β.

The most general case in which $d^1 \neq d^2$ is conceptually no more complex than that described above, but the details of the methodology for approximating Ξ and in linearizing the master problem are quite involved. The reader is referred to [23] for the details.

18.7.2 Convexification-Based Methods

Primal algorithms are based on the implicit solution of (POLY-R) and generalize the well-known framework of branch and cut that has been so successful in the MILP case. This class of algorithms is based on the iterative approximation of conv($\mathcal{F}^\omega$) beginning with the approximation $\mathcal{P}^\omega$, the feasible region in scenario ω of the following relaxation obtained by dropping both the value-function constraint (VFRc) and the integrality requirements (VFRd) and (VFRe) from (VFR).

$$\begin{aligned}
\min \quad & cx + \sum_{\omega \in \Omega} p_\omega d^1 y^\omega && && \\
\text{s.t.} \quad & A^1 x \geq b^1 && && \text{(LPRa)} \\
& G^2 y^\omega \geq b^2_\omega - A^2_\omega x && \forall \omega \in \Omega && \text{(LPRb)} \\
& x \in \mathbb{R}^{n_1}_+ && && \text{(LPRc)} \\
& y^\omega \in \mathbb{R}^{n_2}_+ && \forall \omega \in \Omega. && \text{(LPRd)}
\end{aligned}$$

Being an LP, this relaxation is easily solved, but it is not difficult to see, however, that it is rather weak (see, e.g., the example on page 552). A straightforward way to strengthen it is simply by including the integrality constraints (VFRd) and (VFRe) from (VFR). This leads to an MILP relaxation with feasible set

$$\mathcal{S}^\omega = \mathcal{P}^\omega \cap (X \times Y)$$

in scenario $\omega \in \Omega$, which, while stronger, is clearly more difficult to solve and also still potentially weak—whether adopting it is a computationally good idea is a purely empirical question. When the number of scenarios is large, dropping constraints (LPRb) from the relaxation may also be advantageous, since this may reduce the size of the relaxation.

As in cutting plane methods for MILPs, the idea is to improve this initial formulation with the addition of inequalities valid for $\mathcal{S}^\omega$, $\mathcal{F}^\omega$, $\bigcup_{\omega \in \Omega} \mathcal{F}^\omega$, or even $\mathcal{F}^1$. In some cases, inequalities may first be derived with respect to $\mathcal{S}^\omega$ or $\mathcal{F}^\omega$ for some particular scenario $\omega \in \Omega$ and then lifted to become valid for a larger set. Inequalities valid for $\mathcal{S}^\omega$ (which can be referred to as *feasibility cuts*) can be generated using any number of well-known procedures associated with cutting plane algorithms for mixed integer linear programming. Inequalities valid for $\mathcal{F}^\omega$ (which can be referred to as *optimality cuts*) are the more interesting case because they can incorporate information about the value function in order to eliminate members of $\mathcal{P}^\omega$ that are not two-stage feasible.

In early work on these methods, the authors of [53] developed inequalities valid for $\mathcal{F}^\omega$ in the case for which Ω is a singleton and the variables must all be integer ($r_1 = n_1$ and $r_2 = n_2$), which illustrate the basic principles. When the input data are integer, a very simple argument can be used to generate an inequality valid for $\mathcal{F}^\omega$ but violated by $(\hat{x}, \hat{y})$, an extreme point of $\mathcal{P}^\omega$ not in $\mathcal{F}^\omega$, by taking advantage of the discrete nature of the feasible set. Assuming the solution of the LP relaxation is an extreme point of $\mathcal{P}^\omega$, there is thus a hyperplane supporting $\mathcal{P}^\omega$ and inducing a face of dimension 0. As such, there exist $f \in \mathbb{R}^{n_1}$, $g \in \mathbb{R}^{n_2}$, and $\gamma \in \mathbb{R}$ such that the hyperplane $\{(x, y) \in \mathbb{R}^{n_1+n_2} \mid fx + gy = \gamma\}$ intersects $\mathcal{P}^\omega$ in the unique point $(\hat{x}, \hat{y})$. Hence, we have that $fx + gy \leq \gamma$ for all $(x, y) \in \mathcal{P}^\omega$. Finally, since the face of $\mathcal{P}^\omega$ induced by this inequality does not contain any other members of $\mathcal{S}^\omega$, we can "push" the hyperplane until it meets the next integer point without separating any additional members of $\mathcal{F}^\omega$. Hence,

$$fx + gy \leq \gamma - 1$$

is valid for $\mathcal{F}^\omega$. This procedure is similar in spirit to the Gomory procedure for standard MILPs. It is used, for instance, in [50]. We next describe the method with a brief example.

Example of Valid Inequalities
Consider the instance

$$\max_x \min_y \{y \mid -x + y \leq 2, -2x - y \leq -2, 3x - y \leq 3, y \leq 3, x, y \in \mathbb{Z}_+\},$$

with $|\Omega| = 1$, whose feasible region is the set $\mathcal{F} = \{(0, 2), (1, 0), (2, 3)\}$ shown in Fig. 18.6. Solving the LP relaxation yields the point $(1, 3)$, which is not feasible. This point is eliminated by the addition of the inequality $x - 2y \geq -4$, which is valid for the feasible region $\mathcal{F}$ and is obtained as a strengthening of the inequality $x - 2y \geq -5$, which is valid for the LP relaxation itself.

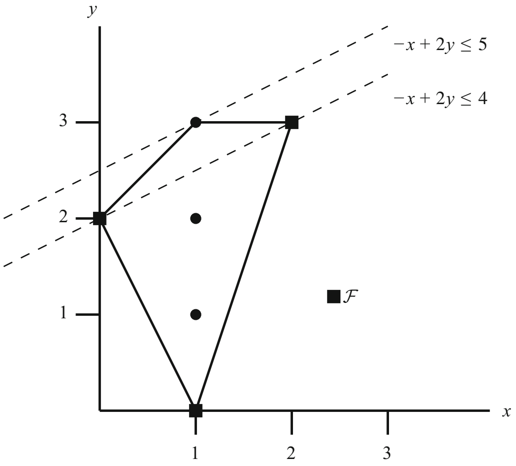

Fig. 18.6 Example of a valid inequality

This cut generation procedure is enough to yield a converging algorithm in this case, but it amounts to removing infeasible points one by one and is not scalable in general. An important observation is that this cut only exploits integrality of the solutions and does not take into account any information about the second-stage value function.

A generalized version of this basic methodology has since been described in [127] and enhanced with additional classes of inequalities, including those valid for the general mixed integer case described in [57, 58]. Inequalities valid for more general discrete probability spaces are derived in [60] for the case $d^1 = d^2$.

Stronger cuts can be obtained by using disjunctive arguments based on knowledge of the value function. In particular, an option is to add constraints of the form

$$d^2 y^{\omega} \leq \bar{\phi}(b^2_{\omega} - A^2_{\omega} x),$$

where $\bar{\phi}$ is a primal function, as defined in Definition 18.5.2. Such primal functions can take many forms and imposing such constraints may be expensive. In general, the form of such functions will be either affine or piecewise polyhedral ("standard" disjunctive programming techniques can be used to obtain a reformulation which only involves linear functions).

18.8 Conclusions

We have introduced a unified framework for multistage mixed integer linear optimization problems which encompasses both multilevel mixed integer linear optimization problems and multistage mixed integer linear optimization problems with recourse. Focusing on the two-stage case, we have investigated the nature of the value function of the second-stage problem and highlighted its connection to dual functions and the theory of duality for mixed integer linear optimization problems. We have summarized different reformulations for this broad class of problems, which rely on either the risk function, the value function, or on their polyhedral nature. We have then presented the main solution techniques for problems of this class, considering both dual- and primal-type methods, the former based on a Benders-like decomposition to approximate either the risk function or the value function, and the latter based on a cutting plane technique that relies on the polyhedral nature of these problems. While much work is still to be done for solving multistage mixed integer linear optimization problems with techniques that are (*mutatis mutandis*, given their intrinsic harder nature) of comparable efficiency to those for solving single-level problems, we believe that the theoretical understanding of multistage mixed integer linear problems is now sufficiently mature to make this an achievable objective.

References

1. E. Amaldi, A. Capone, S. Coniglio, L.G. Gianoli, Energy-aware traffic engineering with elastic demands and MMF bandwidth allocation, in *Proc of 18th IEEE Int. Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD 2013)* (IEEE, Piscataway, 2013), pp. 169–174
2. E. Amaldi, A. Capone, S. Coniglio, L.G. Gianoli, Network optimization problems subject to max-min fair flow allocation. IEEE Commun. Lett. **17**(7), 1463–1466 (2013)
3. E. Amaldi, S. Coniglio, L.G. Gianoli, C.U. Ileri, On single-path network routing subject to max-min fair flow allocation. Electron Notes Discrete Math. **41**, 543–550 (2013)
4. E. Amaldi, S. Coniglio, L. Taccari, Maximum throughput network routing subject to fair flow allocation, in *Proc. of Int. Symp. on Combinatorial Optimization (ISCO 2014)* (Springer, Berlin, 2014), pp. 1–12
5. M.A. Amouzegar, K. Moshirvaziri, Determining optimal pollution control policies: an application of bilevel programming. Eur. J. Oper. Res. **119**(1), 100–120 (1999)
6. B. An, J. Pita, E. Shieh, M. Tambe, C. Kiekintveld, J. Marecki, Guards and protect: next generation applications of security games. ACM SIGecom Exch. **10**(1), 31–34 (2011)
7. R. Avenhaus, A. Okada, S. Zamir, Inspector leadership with incomplete information, in *Game Equilibrium Models IV* (Springer, Berlin, 1991), pp. 319–361
8. J. Bard, An algorithm for solving the general bilevel programming problem. Math. Oper. Res. **8**(2), 260–272 (1983)
9. J. Bard, J.T. Moore, An algorithm for the discrete bilevel programming problem. Nav. Res. Logist. **39**(3), 419–435 (1992)
10. J.F. Bard, J. Plummer, J.C. Sourie, A bilevel programming approach to determining tax credits for biofuel production. Eur. J. Oper. Res. **120**, 30–46 (2000)
11. L. Baringo, A.J. Conejo, Transmission and wind power investment. IEEE Trans. Power Syst. **27**(2), 885–893 (2012)
12. N. Basilico, S. Coniglio, N. Gatti, Methods for finding leader-follower equilibria with multiple followers (extended abstract), in *Proc. of 2016 Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2016)* (2016), pp. 1363–1364
13. N. Basilico, S. Coniglio, N. Gatti, A. Marchesi, Bilevel programming approaches to the computation of optimistic and pessimistic single-leader-multi-follower equilibria, in *Proc. of 16th Int. Symp. on Experimental Algorithms (SEA 2017)* (Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, Wadern, 2017)
14. N. Basilico, S. Coniglio, N. Gatti, A. Marchesi, Bilevel programming methods for computing single-leader-multi-follower equilibria in normal-form and polymatrix games. EURO J. Comput. Optim. **8**, 3–31 (2020)
15. O. Ben-Ayed, C. Blair, Computational difficulties of bilevel linear programming. Oper. Res. **38**, 556–560 (1990)
16. O. Ben-Ayed, C. Blair, D. Boyce, L. LeBlanc, Construction of a real-world bilevel linear programming model of the highway network design problem. Ann. Oper. Res. **34**(1), 219–254 (1992)
17. D.P. Bertsekas, *Dynamic Programming and Optimal Control* (Athena Scientific, Belmont, 2017)
18. J.R. Birge, F. Louveaux, *Introduction to Stochastic Programming* (Springer Science & Business Media, New York, 2011)
19. C.E. Blair, A closed-form representation of mixed-integer program value functions. Math. Program. **71**(2), 127–136 (1995)
20. C.E. Blair, R.G. Jeroslow, The value function of a mixed integer program: I. Discret. Math. **19**(2), 121–138 (1977)
21. C.E. Blair, R.G. Jeroslow, The value function of a mixed integer program: II. Discret. Math. **25**(1), 7–19 (1979)

22. C.E. Blair, R.G. Jeroslow, Constructive characterizations of the value-function of a mixed-integer program I. Discret. Appl. Math. **9**(3), 217–233 (1984)
23. S. Bolusani, T.K. Ralphs, A framework for generalized Benders' decomposition and its application to multilevel optimization. Technical report 20T-004, COR@L Laboratory, Lehigh University, 2020. http://coral.ie.lehigh.edu/~ted/files/papers/MultilevelBenders20.pdf
24. J. Bracken, J.T. McGill, Mathematical programs with optimization problems in the constraints. Oper. Res. **21**(1), 37–44 (1973)
25. A.P. Burgard, P. Pharkya, C.D. Maranas, OptKnock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. Biotechnol. Bioeng. **84**, 647–657 (2003)
26. P. Calamai, L. Vicente, Generating quadratic bilevel programming problems. ACM Trans. Math. Softw. **20**, 103–119 (1994)
27. A. Caprara, M. Carvalho, A. Lodi, G. Woeginger, Bilevel knapsack with interdiction constraints. INFORMS J. Comput. **28**(2), 319–333 (2016)
28. M. Caramia, R. Mari, Enhanced exact algorithms for discrete bilevel linear problems. Optim. Lett. **9**(7), 1447–1468 (2015)
29. C.C. Carøe, J. Tind, L-shaped decomposition of two-stage stochastic programs with integer recourse. Math. Program. **83**(1), 451–464 (1998)
30. M. Castiglioni, A. Marchesi, N. Gatti, Be a leader or become a follower: the strategy to commit to with multiple leaders, in *Proc. of 28th Int. Joint Conf. on Artificial Intelligence (IJCAI 2019)* (2019)
31. M. Castiglioni, A. Marchesi, N. Gatti, S. Coniglio, Leadership in singleton congestion games: what is hard and what is easy. Artif. Intell. **277**, 103–177 (2019)
32. A. Celli, S. Coniglio, N. Gatti, Computing optimal ex ante correlated equilibria in two-player sequential games, in *Proc. of 18th Int. Conf. on Autonomous Agents and MultiAgent Systems (AAMAS 2019)* (International Foundation for Autonomous Agents and Multiagent Systems, Richland, 2019), pp. 909–917
33. A. Celli, S. Coniglio, N. Gatti, Private Bayesian persuasion with sequential games, in *Proc. of 34th AAAI Conf. on Artificial Intelligence (AAAI 2020)* (AAAI Press, New York, 2020), pp. 1–8
34. R.L. Church, M.P. Scaparra, Protecting critical assets: the r-interdiction median problem with fortification. Geogr. Anal. **39**(2), 129–146 (2006)
35. R.L. Church, M.P. Scaparra, R.S. Middleton, Identifying critical infrastructure: the median and covering facility interdiction problems. Ann. Assoc. Am. Geogr. **94**(3), 491–502 (2004)
36. P.A. Clark, A.W. Westerberg, Bilevel programming for steady-state chemical process design I. Fundamentals and algorithms. Comput. Chem. Eng. **14**(1), 87–97 (1990)
37. B. Colson, P. Marcotte, G. Savard, Bilevel programming: a survey. 4OR **3**(2), 87–107 (2005)
38. S. Coniglio, S. Gualandi, On the separation of topology-free rank inequalities for the max stable set problem, in *Proc. of 16th Int. Symp. on Experimental Algorithms (SEA 2017)* (Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, Wadern, 2017)
39. S. Coniglio, L.G. Gianoli, E. Amaldi, A. Capone, Elastic Traffic Engineering Subject to a Fair Bandwidth Allocation via Bilevel Programming, in *IEEE/ACM Transactions on Networking*, https://doi.org/10.1109/TNET.2020.3007572
40. S. Coniglio, M. Tieves, On the generation of cutting planes which maximize the bound improvement, in *Proc. of 14th Int. Symp. on Experimental Algorithms (SEA 2015)* (Springer, Berlin, 2015), pp. 97–109
41. S. Coniglio, N. Gatti, A. Marchesi, Pessimistic leader-follower equilibria with multiple followers, in *Proc. of 26th Int. Joint Conf. on Artificial Intelligence (IJCAI 2017)* (AAAI Press, New York, 2017), pp. 171–177
42. S. Coniglio, N. Gatti, A. Marchesi, Computing a pessimistic Stackelberg equilibrium with multiple followers: the mixed-pure case. Algorithmica **82**, 1189–1238 (2020)
43. S. Coniglio, M. Sirvent, M. Weibelzahl, Airport capacity extension, fleet investment, and optimal aircraft scheduling in a multi-level market model: quantifying the costs of imperfect markets (2020, under review). http://www.optimization-online.org/DB_HTML/2017/05/5989.html

44. V. Conitzer, D. Korzhyk, Commitment to correlated strategies, in *Proc. of 25th AAAI Conf. on Artificial Intelligence (AAAI 2011)* (2011), pp. 632–637
45. V. Conitzer, T. Sandholm, Computing the optimal strategy to commit to, in *Proc. of 7th ACM Conf. on Electronic Commerce (EC 2006)* (2006), pp. 82–90
46. S.A. Cook, The complexity of theorem-proving procedures, in *Proc. of 3rd Annual ACM Symposium on Theory of Computing* (ACM, New York, 1971), pp. 151–158
47. K.J. Cormican, D.P. Morton, R.K. Wood, Stochastic network interdiction. Oper. Res. **46**(2), 184–197 (1998)
48. J.-P. Côté, P Marcotte, G. Savard, A bilevel modelling approach to pricing and fare optimisation in the airline industry. J. Revenue Pricing Manag. **2**(1), 23–36 (2003)
49. S. Dempe, Discrete bilevel optimization problems. Technical Report D-04109, Institut fur Wirtschaftsinformatik, Universitat Leipzig, Leipzig, 2001
50. S. Dempe, F. Mefo Kue, Solving discrete linear bilevel optimization problems using the optimal value reformulation. J. Glob. Optim. **68**(2), 255–277 (2017)
51. S. Dempe, V. Kalashnikov, R.Z. Rios-Mercado, Discrete bilevel programming: application to a natural gas cash-out problem. Eur. J. Oper. Res. **166**(2), 469–488 (2005)
52. S. DeNegre, Interdiction and discrete bilevel linear programming. PhD, Lehigh University, 2011 http://coral.ie.lehigh.edu/~ted/files/papers/ScottDeNegreDissertation11.pdf
53. S. DeNegre, T.K. Ralphs, A branch-and-cut algorithm for bilevel integer programming, in *Proc. of 11th INFORMS Computing Society Meeting* (2009), pp. 65–78. http://coral.ie.lehigh.edu/~ted/files/papers/BILEVEL08.pdf
54. K. Dhamdhere, R. Ravi, M. Singh, On two-stage stochastic minimum spanning trees, in *International Conference on Integer Programming and Combinatorial Optimization* (Springer, Berlin, 2005), pp. 321–334
55. M. Dyer, L. Stougie, Computational complexity of stochastic programming problems. Math. Program. **106**(3), 423–432 (2006)
56. N.P. Faísca, V. Dua, B. Rustem, P.M. Saraiva, E.N. Pistikopoulos, Parametric global optimisation for bilevel programming. J. Glob. Optim. **38**, 609–623 (2007)
57. M. Fischetti, I. Ljubić, M. Monaci, M. Sinnl, A new general-purpose algorithm for mixed-integer bilevel linear programs. Oper. Res. **65**(6), 1615–1637 (2017)
58. M. Fischetti, I. Ljubić, M. Monaci, M. Sinnl, On the use of intersection cuts for bilevel optimization. Math. Program. **172**, 77–103 (2018)
59. F. Furini, I. Ljubic, S. Martin, P. San Segundo, The maximum clique interdiction game. Eur. J. Oper. Res. **277**(1), 112–127 (2019)
60. D. Gade, S. Küçükyavuz, S. Sen, Decomposition algorithms with parametric gomory cuts for two-stage stochastic integer programs. Math. Program. **144**, 1–26 (2012)
61. J. Gan, E. Elkind, M. Wooldridge, Stackelberg security games with multiple uncoordinated defenders, in *Proc. of 17th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)* (2018)
62. L.P. Garcés, A.J. Conejo, R. García-Bertrand, R. Romero, A bilevel approach to transmission expansion planning within a market environment. IEEE Trans. Power Syst. **24**(3), 1513–1522 (2009)
63. M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W.H. Freeman and Company, New York, 1979)
64. M. Gendreau, G. Laporte, R. Séguin, Stochastic vehicle routing. Eur. J. Oper. Res. **88**(1), 3–12 (1996)
65. P.M. Ghare, D.C. Montgomery, W.C. Turner, Optimal interdiction policy for a flow network. Nav. Res. Logist. Q. **18**, 27–45 (1971)
66. I.L. Gørtz, V. Nagarajan, R. Saket, Stochastic vehicle routing with recourse, in *International Colloquium on Automata, Languages, and Programming* (Springer, Berlin, 2012), pp. 411–423
67. E. Grass, K. Fischer, Two-stage stochastic programming in disaster management: a literature survey. Surv. Oper. Res. Manag. Sci. **21**(2), 85–100 (2016)

68. V. Grimm, A. Martin, M. Martin, M. Weibelzahl, G. Zöttl, Transmission and generation investment in electricity markets: the effects of market splitting and network fee regimes. Eur. J. Oper. Res. **254**(2), 493–509 (2016)
69. A. Gupta, R. Ravi, A. Sinha, Lp rounding approximation algorithms for stochastic network design. Math. Oper. Res. **32**(2), 345–364 (2007)
70. M. Güzelsoy, Dual methods in mixed integer linear programming. PhD, Lehigh University, 2009. http://coral.ie.lehigh.edu/~ted/files/papers/MenalGuzelsoyDissertation09.pdf
71. M. Güzelsoy, T.K. Ralphs, Duality for mixed-integer linear programs. Int. J. Oper. Res. **4**, 118–137 (2007). http://coral.ie.lehigh.edu/~ted/files/papers/MILPD06.pdf
72. G.A. Hanasusanto, D. Kuhn, W. Wiesemann, A comment on “computational complexity of stochastic programming problems”. Math. Program. **159**(1), 557–569 (2016)
73. P. Hansen, B. Jaumard, G. Savard, New branch-and-bound rules for linear bilevel programming. SIAM J. Sci. Stat. Comput. **13**(5), 1194–1217 (1992)
74. A. Hassanzadeh, T.K. Ralphs, A generalized Benders' algorithm for two-stage stochastic program with mixed integer recourse. Technical Report COR@L Laboratory Report 14T-005, Lehigh University, 2014. http://coral.ie.lehigh.edu/~ted/files/papers/SMILPGenBenders14.pdf
75. A. Hassanzadeh, T.K. Ralphs, On the value function of a mixed integer linear optimization problem and an algorithm for its construction. Technical report, COR@L Laboratory Report 14T-004, Lehigh University, 2014. http://coral.ie.lehigh.edu/~ted/files/papers/MILPValueFunction14.pdf
76. H. Held, D.L. Woodruff, Heuristics for multi-stage interdiction of stochastic networks. J. Heuristics **11**(5–6), 483–500 (2005)
77. M. Hemmati, J.C. Smith, A mixed integer bilevel programming approach for a competitive set covering problem. Technical report, Clemson University, 2016
78. B.F. Hobbs, S.K. Nelson, A nonlinear bilevel model for analysis of electric utility demand-side planning issues. Ann. Oper. Res. **34**(1), 255–274 (1992)
79. E. Israeli, System interdiction and defense. PhD thesis, Naval Postgraduate School, 1999
80. E. Israeli, R.K. Wood, Shortest path network interdiction. Networks **40**(2), 97–111 (2002)
81. U. Janjarassuk, J. Linderoth, Reformulation and sampling to solve a stochastic network interdiction problem. Networks **52**, 120–132 (2008)
82. R.G. Jeroslow, The polynomial hierarchy and a simple model for competitive analysis. Math. Program. **32**(2), 146–164 (1985)
83. P. Kall, J. Mayer, *Stochastic Linear Programming: Models, Theory, and Computation* (Springer, Berlin, 2010)
84. B. Kara, V. Verter, Designing a road network for hazardous materials transportation. Transp. Sci. **38**(2), 188–196 (2004)
85. R.M. Karp, On the computational complexity of combinatorial problems. Networks **5**, 45–68 (1975)
86. I. Katriel, C. Kenyon-Mathieu, E. Upfal, Commitment under uncertainty: two-stage stochastic matching problems, in *International Colloquium on Automata, Languages, and Programming* (Springer, Berlin, 2007), pp. 171–182
87. C. Kiekintveld, M. Jain, J. Tsai, J. Pita, F. Ordóñez, M. Tambe, Computing optimal randomized resource allocations for massive security games, in *Proceedings of AAMAS* (2009), pp. 689–696
88. K.-M. Klein, About the complexity of two-stage stochastic IPs (2019)
89. N. Kong, A.J. Schaefer, B. Hunsaker, Two-stage integer programs with stochastic right-hand sides: a superadditive dual approach. Math. Program. **108**(2), 275–296 (2006)
90. M. Köppe, M. Queyranne, C.T. Ryan, Parametric integer programming algorithm for bilevel mixed integer programs. J. Optim. Theory Appl. **146**(1), 137–150 (2010)
91. A.A. Kulkarni, U.V. Shanbhag, A shared-constraint approach to multi-leader multi-follower games. Set-Valued Var. Anal. **22**(4), 691–720 (2014)
92. M. Labbé, A. Violin, Bilevel programming and price setting problems. 4OR **11**(1), 1–30 (2013)

93. M. Labbé, P. Marcotte, G. Savard, A bilevel model of taxation and its application to optimal highway pricing. Manag. Sci. **44**, 1608–1622 (1998)
94. G. Laporte, F.V. Louveaux, The integer l-shaped method for stochastic integer programs with complete recourse. Oper. Res. Lett. **13**(3), 133–142 (1993)
95. A. Laszka, J. Lou, Y. Vorobeychik, Multi-defender strategic filtering against spear-phishing attacks, in *Proc. of 30th AAAI Conf. on Artificial Intelligence (AAAI 2016)* (2016)
96. S. Leyffer, T. Munson, Solving multi-leader-common-follower games. Optim. Methods Softw. **25**(4), 601–623 (2010)
97. A. Lodi, T.K. Ralphs, F. Rossi, S. Smriglio, Interdiction branching. Technical Report OR/09/10, DEIS-Università di Bologna, 2009
98. J. Lou, Y. Vorobeychik, Equilibrium analysis of multi-defender security games, in *Proc. of 24th Int. Joint Conf. on Artificial Intelligence (IJCAI 2019)* (2015)
99. J. Lou, A.M. Smith, Y. Vorobeychik, Multidefender security games. IEEE Intell. Syst. **32**(1), 50–60 (2017)
100. F.V. Louveaux, M.H. van der Vlerk, Stochastic programming with simple integer recourse. Math. Program. **61**(1), 301–325 (1993)
101. L. Lozano, J.C. Smith, A value-function-based exact approach for the bilevel mixed-integer programming problem. Oper. Res. **65**(3), 768–786 (2017)
102. A. Mahajan, On selecting disjunctions in mixed integer linear programming. PhD, Lehigh University, 2009. http://coral.ie.lehigh.edu/~ted/files/papers/AshutoshMahajanDissertation09.pdf
103. A. Mahajan, T.K. Ralphs, On the complexity of selecting disjunctions in integer programming. SIAM J. Optim. **20**(5), 2181–2198 (2010). http://coral.ie.lehigh.edu/~ted/files/papers/Branching08.pdf
104. A. Marchesi, S. Coniglio, N. Gatti, Leadership in singleton congestion games, in *Proc. of 27th Int. Joint Conf. on Artificial Intelligence (IJCAI 2018)* (AAAI Press, New York, 2018), pp. 447–453
105. A.W. McMasters, T.M. Mustin, Optimal interdiction of a supply network. Nav. Res. Logist. Q. **17**, 261–268 (1970)
106. A. Migdalas, Bilevel programming in traffic planning: models, methods and challenge. J. Glob. Optim. **7**, 381–405 (1995)
107. J.T. Moore, J.F. Bard, The mixed integer linear bilevel programming problem. Oper. Res. **38**(5), 911–921 (1990)
108. J-S. Pang, M. Fukushima, Quasi-variational inequalities, generalized Nash equilibria, and multi-leader-follower games. Comput. Manag. Sci. **2**(1), 21–56 (2005)
109. P. Paruchuri, J.P. Pearce, J. Marecki, M. Tambe, F. Ordonez, S. Kraus, Playing games for security: an efficient exact algorithm for solving Bayesian Stackelberg games, in *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)* (2008), pp. 895–902
110. T.K. Ralphs, M. Güzelsoy, The SYMPHONY callable library for mixed integer programming, in *Proceedings of the Ninth INFORMS Computing Society Conference* (2005), pp. 61–76. http://coral.ie.lehigh.edu/~ted/files/papers/SYMPHONY04.pdf
111. T.K. Ralphs, M. Güzelsoy, Duality and warm starting in integer programming, in *The Proceedings of the 2006 NSF Design, Service, and Manufacturing Grantees and Research Conference* (2006). http://coral.ie.lehigh.edu/~ted/files/papers/DMII06.pdf
112. R.T. Rockafellar, S. Uryasev, Optimization of conditional value-at-risk. J. Risk **2**, 21–42 (2000)
113. V. Rutenburg, Propositional truth maintenance systems: classification and complexity analysis. Ann. Math. Artif. Intell. **10**(3), 207–231 (1994)
114. G.K. Saharidis, M.G. Ierapetritou, Resolution method for mixed integer bi-level linear problems based on decomposition technique. J. Glob. Optim. **44**(1), 29–51 (2008)
115. P. San Segundo, S. Coniglio, F. Furini, I. Ljubić, A new branch-and-bound algorithm for the maximum edge-weighted clique problem. Eur. J. Oper. Res. **278**(1), 76–90 (2019)
116. W.H. Sandholm, Evolutionary implementation and congestion pricing. Rev. Econ. Stud. **69**(3), 667–689 (2002)

117. M.P. Scaparra, R.L. Church, A bilevel mixed-integer program for critical infrastructure protection planning. Comput. Oper. Res. **35**(6), 1905–1923 (2008)
118. M. Schaefer, C. Umans, Completeness in the polynomial-time hierarchy: a compendium. SIGACT News **33**(3), 32–49 (2002)
119. R. Schultz, L. Stougie, M.H. Van Der Vlerk, Solving stochastic programs with integer recourse by enumeration: a framework using Gröbner basis. Math. Program. **83**(1), 229–252 (1998)
120. S. Sen, J.L. Higle, The C^3 theorem and a D^2 algorithm for large scale stochastic mixed-integer programming: set convexification. Math. Program. **104**(1), 1–20 (2005)
121. A. Shapiro, Monte Carlo sampling methods, in *Handbooks in Operations Research and Management Science*, vol. 10 (Elsevier, Amsterdam, 2003), pp. 353–425
122. H.D. Sherali, B.M.P. Fraticelli, A modification of Benders' decomposition algorithm for discrete subproblems: an approach for stochastic programs with integer recourse. J. Glob. Optim. **22**(1), 319–342 (2002)
123. H.D. Sherali, X. Zhu, On solving discrete two-stage stochastic programs having mixed-integer first-and second-stage variables. Math. Program. **108**(2), 597–616 (2006)
124. A. Smith, Y. Vorobeychik, J. Letchford, Multidefender security games on networks. ACM SIGMETRICS Perform. Eval. Rev. **41**(4), 4–7 (2014)
125. L.J. Stockmeyer, The polynomial-time hierarchy. Theor. Comput. Sci. **3**, 1–22 (1976)
126. S. Tahernejad, Two-stage mixed integer stochastic bilevel linear optimization. PhD, Lehigh University, 2019
127. S. Tahernejad, T.K. Ralphs, S.T. DeNegre, A branch-and-cut algorithm for mixed integer bilevel linear optimization problems and its implementation. Math. Program. Comput. (2020). http://coral.ie.lehigh.edu/~ted/files/papers/MIBLP16.pdf
128. M. Tambe, *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned* (Cambridge University Press, Cambridge, 2011)
129. S. Uryasev, Conditional value-at-risk: optimization algorithms and applications, in *Proc. of the IEEE/IAFE/INFORMS 2000 Conference on Computational Intelligence for Financial Engineering (CIFEr)* (IEEE, Piscataway, 2000), pp. 49–57
130. R.M. Van Slyke, R. Wets, L-shaped linear programs with applications to optimal control and stochastic programming. SIAM J. Appl. Math. **17**, 638–663 (1969)
131. L.N. Vicente, P.H. Calamai, Bilevel and multilevel programming: a bibliography review. J. Glob. Optim. **5**(3), 291–306 (1994)
132. L. Vicente, G. Savard, J. Júdice, Discrete linear bilevel programming problem. J. Optim. Theory Appl. **89**(3), 597–614 (1996)
133. H. Von Stackelberg, *Market Structure and Equilibrium* (Springer Science & Business Media, Berlin, 2010)
134. B. von Stengel, S. Zamir, Leadership games with convex strategy sets. Games Econom. Behav. **69**(2), 446–457 (2010)
135. S.W. Wallace, W.T. Ziemba, *Applications of Stochastic Programming* (SIAM, Philadelphia, 2005)
136. L. Wang, P. Xu, The watermelon algorithm for the bilevel integer linear programming problem. SIAM J. Optim. **27**(3), 1403–1430 (2017)
137. U.P. Wen, Y.H. Yang, Algorithms for solving the mixed integer two-level linear programming problem. Comput. Oper. Res. **17**(2), 133–142 (1990)
138. H.P. Williams, Duality in mathematics and linear and integer programming. J. Optim. Theory Appl. **90**(2), 257–278 (1996)
139. R. Wollmer, Removing arcs from a network. Oper. Res. **12**(6), 934–940 (1964)
140. L.A. Wolsey, Integer programming duality: price functions and sensitivity analysis. Math. Program. **20**(1), 173–195 (1981)
141. R.K. Wood, Deterministic network interdiction. Math. Comput. Model. **17**(2), 1–18 (1993)
142. P. Xu, L. Wang, An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. Comput. Oper. Res. **41**, 309–318 (2014)

143. B. Zeng, Y. An, Solving bilevel mixed integer program by reformulations and decomposition. Technical report, University of South Florida, 2014

144. Y. Zhang, L.V. Snyder, T.K. Ralphs, Z. Xue, The competitive facility location problem under disruption risks. Transport. Res. Part E Log. Transport. Rev. **93** (2016). http://coral.ie.lehigh.edu/~ted/files/papers/CFLPD16.pdf