

Chapter 1

BILEVEL PROGRAMMING: A COMBINATORIAL PERSPECTIVE

Patrice Marcotte

DIRO and CRT, Université de Montréal

marcotte@iro.umontreal.ca

Gilles Savard

MAGI and GERAD, École Polytechnique de Montréal

gilles.savard@polymtl.ca

Abstract Bilevel programming is a branch of optimization where a subset of variables is constrained to lie in the optimal set of an auxiliary mathematical program. This chapter presents an overview of two specific classes of bilevel programs, and in particular their relationship to well-known combinatorial problems.

1. Introduction

In optimization and game theory, it is frequent to encounter situations where conflicting agents are taking actions according to a predefined sequence of play. For instance, in the Stackelberg version of duopolistic equilibrium [Sta52], a *leader* firm incorporates within its decision process the reaction of the *follower* firm to its course of action. By extending this concept to a pair of arbitrary mathematical programs, one obtains the class of *bilevel programs*, which allow the modeling of many decision processes. The term “bilevel programming” appeared for the first time in a paper by Candler and Norton [CN77], who considered a multi-level formulation in the context of agricultural economics. Since that time, hundreds of papers have been dedicated to this topic. The reader interested in the theory and applications of bilevel programming is referred

to the recent books by Shimizu, Ishizuka and Bard [SIB97], Luo, Pang and Ralph [LPR96], Bard [Bar98] and Dempe [Dem02].

Generically, a bilevel program assumes the form

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

where $\mathcal{S}(x)$ denotes the solution set of a mathematical program parameterized in the vector x , i.e.,

$$\begin{aligned} \mathcal{S}(x) = \quad & \arg \min_y \quad g(x,y) \\ \text{s.t.} \quad & (x,y) \in Y. \end{aligned}$$

In this formulation, the leader is free, whenever the set $\mathcal{S}(x)$ does not shrink to a singleton, to select an element of $\mathcal{S}(x)$ that suits her best. This corresponds to the *optimistic* formulation. Alternatively, the *pessimistic* formulation refers to the case where the leader protects herself against the worst possible situation, and is formulated as

$$\begin{aligned} \min_x \max_y \quad & f(x,y) \\ \text{s.t.} \quad & (x,y) \in X \\ & y \in \mathcal{S}(x). \end{aligned}$$

The scope of this chapter is limited to the optimistic formulation. The reader interested in the pessimistic formulation is referred to the paper [LM96] by Loridan and Morgan.

In many applications, the lower level corresponds to an equilibrium problem that is best represented as a (parametric) variational inequality or, equivalently, a generalized equation. We then obtain an MPEC (*Mathematical Program with Equilibrium Constraints*), that is expressed as¹

¹Throughout the paper, we assume that vectors on the left-hand side of an inner product are row vectors. Symmetrically, right-hand side vectors are understood to be column vectors. Thus primal (respectively dual) variables usually make up column (respectively row) vectors. Transpose are only used when absolutely necessary.

$$\begin{aligned}
\text{MPEC :} \quad & \min_{x,y} \quad f(x,y) \\
& \text{s.t.} \quad (x,y) \in X \\
& \quad y \in Y(x) \\
& \quad -G(x,y) \in N_{Y(x)}(y),
\end{aligned}$$

where $Y(x) = \{y : (x,y) \in Y\}$ and $N_C(z)$ denotes the normal cone to the set C at the point z . If the vector function G represents the gradient of a differentiable convex function g and the set Y is convex, then MPEC reduces to a bilevel program. Conversely, an MPEC can be reformulated as a standard bilevel program by noting that a vector y is solution of the lower level variational inequality if and only if it globally minimizes, with respect to the argument y , the strongly convex function $\text{gap}(x,y)$ defined as (see Fukushima [Fuk92]):

$$\text{gap}(x,y) = \max_{y' \in Y(x)} G(x,y)(y - y') - \frac{1}{2}\|y - y'\|^2.$$

Being generically non-convex and non-differentiable, bilevel programs are intrinsically hard to solve. For one, the linear bilevel program which corresponds to the simple situation where all functions involved are linear, is strongly \mathcal{NP} -hard (see Section 1.2.2). Further, determining whether a solution is locally optimal is also strongly \mathcal{NP} -hard [VSJ94]. In view of these results, most research has followed two main avenues, either *continuous* or *combinatorial*. The continuous approach is mainly concerned with the characterization of necessary optimality conditions and the development of algorithms that generate sequences converging toward a local solution. Along that line, let us mention works based on the implicit function approach (Kočvara and Outrata [KO94]), on classical nonlinear programming techniques such as SQP (Sequential Quadratic Programming) applied to a single-level reformulation of the bilevel problem (Scholtes and Stöhr [SS99]) or smoothing approaches (Fukushima and Pang [FP99]) and Marcotte et al. [MSZ01]). Most work done on MPECs follows that line.

The combinatorial approach takes a global optimization point of view and looks for the development of algorithms with a guarantee of global optimality. Due to the intractability of the bilevel program, these algorithms are limited to specific subclasses possessing properties such as linear, bilinear or quadratic objectives, which allow for the development of “efficient” algorithms. We consider two classes that are amenable to a global approach, namely bilevel programs involving linear or bilinear objectives. The first class is important as it encompasses a large number

of combinatorial problems (e.g. 0-1 mixed integer programs) while the second allows for the modeling of a rich class of pricing applications. This chapter focuses on the combinatorial structure of these two classes.

2. Linear bilevel programming

The linear/linear bilevel problem (LLBP) takes the form

$$\begin{aligned} \text{LLBP :} \quad & \max_{x,y} \quad c_1x + d_1y \\ & \text{s.t.} \quad A_1x + B_1y \leq b_1 \\ & \quad x \geq 0 \\ & \quad y \in \arg \max_y \quad d_2y \\ & \quad \text{s.t.} \quad A_2x + B_2y \leq b_2 \\ & \quad y \geq 0, \end{aligned}$$

where $c_1 \in \mathbb{R}^{n_x}$, $d_1, d_2 \in \mathbb{R}^{n_y}$, $A_1 \in \mathbb{R}^{n_u \times n_x}$, $A_2 \in \mathbb{R}^{n_l \times n_x}$, $b_1 \in \mathbb{R}^{n_u}$, $B_1 \in \mathbb{R}^{n_u \times n_y}$, $B_2 \in \mathbb{R}^{n_l \times n_y}$, $b_2 \in \mathbb{R}^{n_l}$. The constraints $A_1x + B_1y \leq b_1$ (respectively $A_2x + B_2y \leq b_2$) are the upper (respectively lower) level constraints. The linear term $c_1x + d_1y$ (respectively d_2y) are the upper (respectively lower) level objective function, while x (respectively y) is the vector of upper (respectively lower) level variables².

In order to characterize the solution of LLBP, the following definitions are required.

DEFINITION 1.1

1. The **feasible set** of LLBP is defined as

$$\Omega = \{(x, y) : x \geq 0, y \geq 0, A_1x + B_1y \leq b_1, A_2x + B_2y \leq b_2\}.$$

2. For every $x \geq 0$, the **feasible set of the lower level problem** is defined as

$$\Omega_y(x) = \{y : y \geq 0, B_2y \leq b - A_2x\}.$$

3. The **trace** of the lower level problem with respect to the upper level variables is

$$\Omega_x^2 = \{x : x \geq 0, \Omega_y(x) \neq \emptyset\}.$$

²We slightly abuse notation and use the letter y to denote both the optimal solution (left-hand-side) and the argument (right-hand-side) of the lower level program.

4. For a given vector $x \in \Omega_x^2$, the set of optimal solutions of the lower problem is

$$S(x) = \{y : y \in \arg \max\{d_2 y : y \in \Omega_y(x)\}.$$

A point (x, y) is said to be **rational** if $x \in \Omega_x^2 \cap S(x)$.

5. The **optimal value function** for $x \in \Omega_x^2$ is

$$v(x) = d_2 y, y \in S(x).$$

6. The **admissible set** (also called **induced region**) is

$$\Upsilon = \{(x, y) : x \geq 0, A_1 x + B_1 y \leq b_1, y \in S(x)\}.$$

A point (x, y) is **admissible** if it is feasible and lies in $S(x)$.

Based on the above notations, we characterize optimal solutions for the LLBP.

DEFINITION 1.2 A point (x^*, y^*) is optimal for LLBP if it is admissible and, for all admissible (x, y) , there holds $c_1 x^* + d_1 y^* \geq c_1 x + d_1 y$.

Note that, whenever the upper level constraints involve no lower level variables, then rational points are also admissible. The converse may fail to hold in the presence of joint upper level constraints.

To illustrate some geometric properties of bilevel programs (see Figure 1.2), let us consider the following two-dimensional example (see also Figure 1.1 (a)):

$$\begin{array}{ll} \max_{x,y} & -x - 4y \\ \text{s.t.} & x \geq 0 \end{array}$$

$$\begin{array}{ll} y \in \arg \max_y & y \\ \text{s.t.} & -2x - y \leq 8 \\ & -3x + 2y \leq 6 \\ & 5x + 6y \leq 60 \\ & 2x + y \leq 16 \\ & 2x - 5y \leq 0 \\ & y \geq 0. \end{array}$$

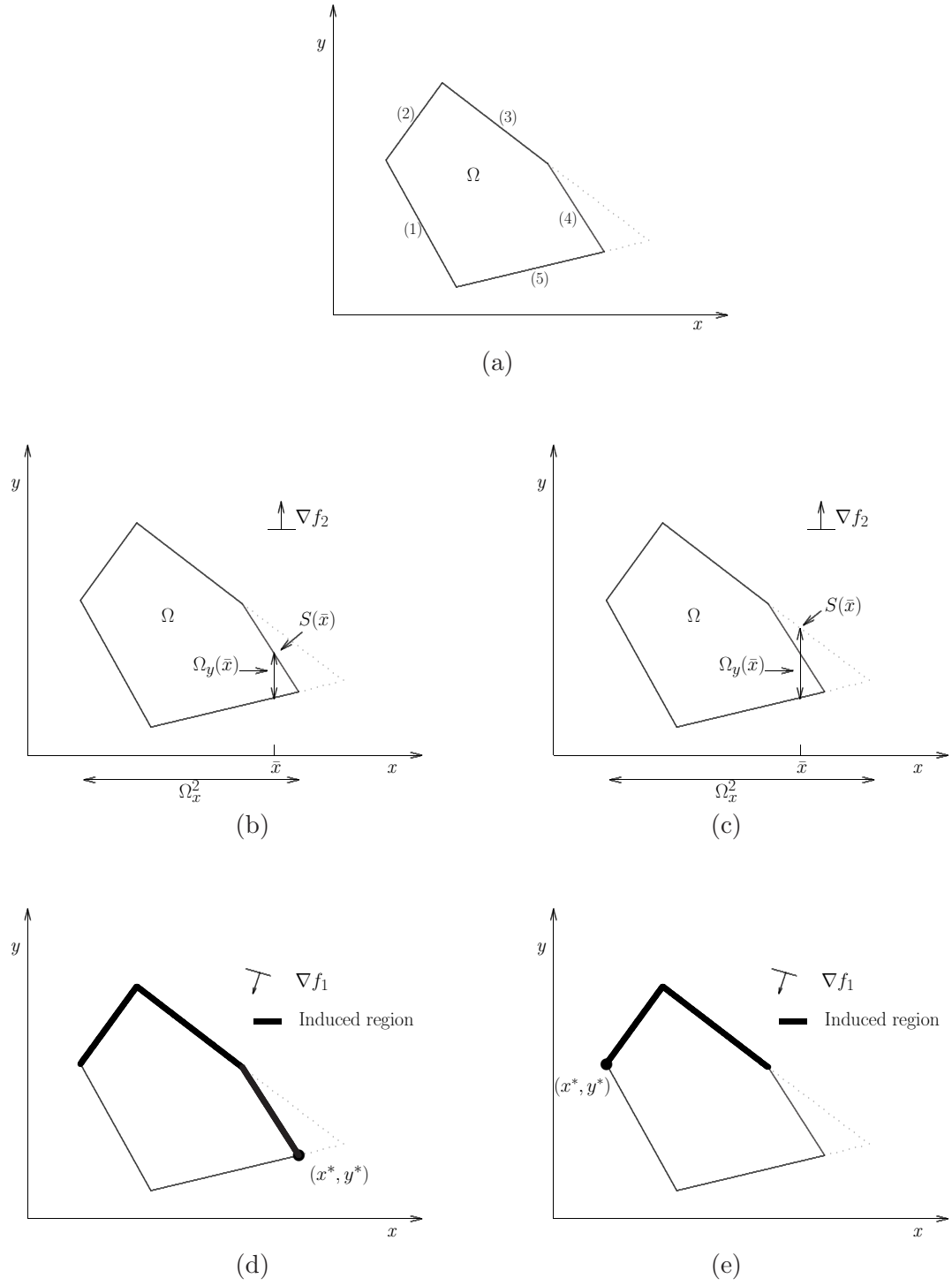


Figure 1.1. Two linear/linear bilevel programs

The left-hand-side graphs (b) and (d) illustrate the example's geometry, while right-hand-side graphs (c) and (e) correspond to the bilevel program obtained after moving the next-to-last constraint from the lower to the upper level, showing the impact of upper level constraints on the admissible set. We observe that the admissible set, represented by thick lines, is not convex. Indeed, its analytic expression is

$$\Upsilon = \{(x, y) : x \geq 0, A_1x + B_1y \leq b_1\} \cap \Gamma_S$$

where

$$\Gamma_S = \{(x, y) : x \in \Omega_x^2, d_2y = v(x)\}$$

represents the union of a finite (possibly empty) set of polyhedra [Sav89]. Based on a result of Hogan [Hog73], one can show that the multi-valued mapping $S(x)$ is closed, whenever the set Ω is compact. In the particular case where $S(x)$ shrinks to a singleton for every $x \in \Omega_x^2$, it follows that the reaction function $y(x) = S(x)$ is continuous.

As seen in Figure 1.2 the presence of joint upper level constraints may considerably modify the structure of the admissible set. It can make this set disconnected, finite or even empty. This is illustrated in Figure 1.2, where a single upper level constraint is slid. In the next section, we will construct a bilevel program with an admissible set corresponding solely of integer points.

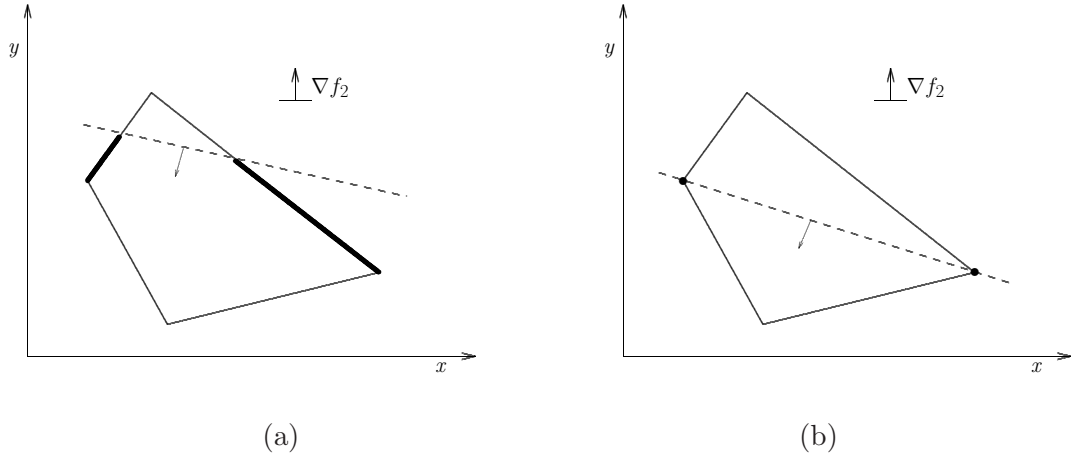


Figure 1.2. A disconnected admission set

The following theorem is a direct consequence of the polyhedral nature of the admissible set. It emphasizes the combinatorial nature of the LLBP.

THEOREM 1.3 *If LLBP has a solution, an optimal solution is attained at an extreme point of Ω .*

The combinatorial nature of bilevel programming can also be observed by studying the single-level reformulation obtained by replacing the lower level problem by its (necessary and sufficient) optimality conditions:

$$\begin{aligned} \text{LLBP}_1 : \quad & \max_{x,y,\lambda} \quad c_1x + d_1y \\ & \text{s.t.} \quad A_1x + B_1y \leq b_1 \\ & \quad \quad A_2x + B_2y \leq b_2 \\ & \quad \quad \lambda B_2 \geq d_2 \\ & \quad \quad \lambda(b_2 - A_2x - B_2y) = 0 \\ & \quad \quad (\lambda B_2 - d_2)y = 0 \\ & \quad \quad x \geq 0, \quad y \geq 0, \quad \lambda \geq 0, \end{aligned}$$

where $\lambda \in \mathbb{R}^{n_l}$. The combinatorial nature is entirely captured by the two orthogonality constraints, which can actually be added to form a single constraint. Their disjunctive nature relates the LLBP to linear mixed integer programming and allows for the development of algorithms based on enumeration and/or cutting plane approaches.

2.1 Equivalence between LLBP and classical problems

In this section, we show that simple polynomial transformations allow to formulate linear mixed 0-1 integer programs (MIP_{0-1}) and bilinear disjoint programs (BDP) as linear bilevel programs, and vice versa. The interest in these reformulations goes beyond the complexity issue. Indeed, Audet [Aud97] and Audet et al. [AHJS97] have uncovered equivalences between algorithms designed to solve (MIP_{0-1}) and LLBP. They have shown that the *HJS* algorithm of [HJS92] designed for solving the LLBP can be mapped onto a standard branch-and-bound method (see for instance [BS65] for addressing an equivalent mixed 0 – 1 program, provided that mutually consistent branching rules are applied. We may therefore claim that the mixed 0 – 1 algorithm is subsumed (the authors use the term *embedded*) by the bilevel algorithm. This result shows that the structure of both problems is virtually indistinguishable, and that any algorithmic improvement on one problem can readily be adapted to the other [AHJS97]: solution techniques developed for solving mixed 0 – 1 programs may be tailored to the LLBP, and vice-versa.

2.1.1 LLBP and MIP_{0-1} . The linear mixed 0–1 programming problem (MIP_{0-1}) is expressed as

$$\begin{aligned} \text{MIP}_{0-1} : \quad & \max_{x,u} \quad cx + eu \\ & \text{s.t.} \quad Ax + Eu \leq b \\ & \quad \quad x \geq 0, \ u \text{ binary valued,} \end{aligned}$$

where $c \in \mathbb{R}^{n_x}, e \in \mathbb{R}^{n_u}, A \in \mathbb{R}^{m \times n_x}, E \in \mathbb{R}^{m \times n_u}, b \in \mathbb{R}^m$.

We first note that the binary condition is equivalent to:

$$\begin{aligned} 0 &\leq u \leq \mathbf{1} \\ 0 &= \min\{u, \mathbf{1} - u\}, \end{aligned}$$

where $\mathbf{1}$ denotes the vector of “all ones”. Next, by introducing an upper level variable y , and defining a second level problem such that the optimal solution corresponds to this minimum, we obtain the equivalent bilevel programming reformulation:

$$\begin{aligned} \text{LLBP}_2 : \quad & \max_{x,y,u} \quad cx + eu \\ & \text{s.t.} \quad Ax + Eu \leq b \\ & \quad \quad 0 \leq u \leq \mathbf{1} \\ & \quad \quad x \geq 0 \\ & \quad \quad y = 0 \\ & \quad \quad y \in \arg \max_w \sum_{i=1}^{n_u} w_i \\ & \quad \quad \text{s.t.} \quad w \leq u \\ & \quad \quad \quad \quad w \leq \mathbf{1} - u. \end{aligned}$$

where $y, w \in \mathbb{R}^{n_u}$. In this formulation, the integrality constraints are no more required, as they are enforced by the upper level constraints $y = 0$, together with the lower level optimality conditions.

In general, upper level constraints make the problem more difficult to solve. Actually, some algorithms only address instances where such constraints are absent. However, as suggested by Vicente, Savard and Júdeice [VSJ96], the constraint $y = 0$ can be enforced by incorporating an *exact penalty* within the leader’s objective, i.e., there exists a threshold value M^* such that, whenever M exceeds M^* , the solution of the following bilevel program satisfies the condition $y = 0$, i.e., the integrality

condition:

$$\begin{aligned}
\text{LLBP}_3 : \quad & \max_{x,y,u} \quad cx + eu - M\mathbf{1}y \\
& \text{s.t.} \quad Ax + Eu \leq b \\
& \quad 0 \leq u \leq \mathbf{1} \\
& \quad x \geq 0 \\
& \quad y \in \arg \max_w \sum_{i=1}^{n_u} w_i \\
& \quad \text{s.t.} \quad w \leq u \\
& \quad \quad w \leq \mathbf{1} - u.
\end{aligned}$$

Conversely, LLBP may be polynomially reduced to a MIP_{0-1} . First, one replaces the lower level problem by its optimality conditions, yielding a single-level program with the complementarity constraints

$$\begin{aligned}
\lambda(b_2 - A_2x - B_2y) &= 0 \\
(\lambda B_2 - d_2)y &= 0.
\end{aligned}$$

The second transformation consists in linearizing the complementarity constraints by introducing two binary vectors u and v and a sufficiently large finite constant $L > 0$, the existence of which is discussed in [VSJ96]):

$$\begin{aligned}
b_2 - A_2x - B_2y &\leq L(\mathbf{1} - u), & \lambda &\leq Lu^\top, \\
y &\leq L(\mathbf{1} - v) & \lambda B_2 - d_2 &\leq Lv^\top.
\end{aligned}$$

This leads to the equivalent MIP_{0-1} reformulation of LLBP:

$$\begin{aligned}
\text{MIP}_{\text{LLBP}} : \quad & \max_{x,y,\lambda,u,v} \quad c_1x + d_1y \\
& \text{s.t.} \quad A_1x + B_1y \leq b_1 \\
& \quad x \geq 0 \\
& \quad A_2x + B_2y \leq b_2 & -\lambda B_2 \leq -d_2 \\
& \quad y \geq 0 & \lambda \geq 0 \\
& \quad -A_2x - B_2y + Lu \leq L\mathbf{1} - b_2 & \lambda - Lu^\top \leq 0 \\
& \quad y + Lv \leq L\mathbf{1} & \lambda B_2 - Lv^\top \leq d_2 \\
& \quad u \text{ binary valued} & v \text{ binary valued.}
\end{aligned}$$

2.1.2 LLBP and BILP. The disjoint bilinear programming problem BILP was introduced by Konno [Kon71] to generalize Mills' approach [Mil60] for computing Nash equilibria [Nas51] of bimatrix games. It can be expressed as follows:

$$\begin{aligned}
\text{BILP :} \quad & \max_{x,u} \quad cx - uQx + ud \\
& \text{s.t.} \quad Ax \leq b_1 \\
& \quad \quad uB \leq b_2 \\
& \quad \quad x \geq 0 \\
& \quad \quad u \geq 0,
\end{aligned}$$

where $c \in \mathbb{R}^{n_x}$, $d \in \mathbb{R}^{n_u}$, $Q \in \mathbb{R}^{n_u \times n_x}$, $A \in \mathbb{R}^{n_v \times n_x}$, $B \in \mathbb{R}^{n_u \times n_y}$, $b_1 \in \mathbb{R}^{n_v}$, $b_2 \in \mathbb{R}^{n_y}$, and the matrix Q assumes no specific structure.

By exploiting the connection between LLBP and BILP, Audet et al. [AHJS99] and Alarie et al. [AAJS01] have been able to construct improved branch-and-cut algorithms for the BILP. Their approach relies on the separability, with respect to the vectors x and u , of the feasible set of BILP. Let us introduce the sets $X = \{x \geq 0 : Ax \leq b_1\}$ and $U = \{u \geq 0 : uB \leq b_2\}$. If both sets are nonempty and the optimal solution of BILP is bounded, we can rewrite BILP as

$$\text{BILP}_2 : \quad \max_{x \in X} \quad cx + \max_{u \in U} u(d - Qx).$$

For fixed $x \in X$, one can replace the inner optimization problem by its dual, to obtain

$$\begin{aligned}
\max_{x \in X} \quad & cx + \min_y \quad b_2 y \\
& \text{s.t.} \quad Qx + By \geq d \\
& \quad \quad y \geq 0.
\end{aligned}$$

Under the boundedness assumption, the dual of the inner problem is feasible and bounded for each $x \in X$. In a symmetric way, one can reverse the roles of x and u to obtain the equivalent formulation

$$\begin{aligned}
\max_{u \in U} \quad & ud + \min_v \quad vb_1 \\
& \text{s.t.} \quad uQ + vA \geq c \\
& \quad \quad v \geq 0.
\end{aligned}$$

Thus, the solution of BILP can be obtained by solving either one of the symmetric bilevel programs

LLBP ₄	LLBP ₅
$ \begin{aligned} \max_{x,y} \quad & cx + b_2 y \\ \text{s.t.} \quad & Ax \leq b_1 \\ & x \geq 0 \\ & y \in \arg \min_y \quad b_2 y \\ & \quad \text{s.t.} \quad Qx + By \geq d \\ & \quad \quad y \geq 0 \end{aligned} $	$ \begin{aligned} \max_{u,v} \quad & ud + vb_1 \\ \text{s.t.} \quad & uB \leq b_2 \\ & u \geq 0 \\ & v \in \arg \min_v \quad vb_1 \\ & \quad \text{s.t.} \quad uQ + vA \geq c \\ & \quad \quad v \geq 0, \end{aligned} $

These two problems correspond to “max-min” programs, i.e., bilevel program involving opposite objective functions.

If BILP is unbounded, the above transformations are no longer valid as the inner problem may prove infeasible for some values of $x \in X$ (or $u \in U$). For instance, the existence of a ray (unbounded direction) in u -space implies that there exist $\bar{x} \in X$ and \bar{u} with $\bar{u}B \leq 0$ such that $\bar{u}(d - Q\bar{x}) > 0$. Equivalently there exists a vector \bar{x} such that the inner problem in BILP₂ is unbounded, which implies in turn that its dual is infeasible with respect to \bar{x} .

In order to be equivalent to BILP, LLBP₄ should therefore select an x -value for which the lower level problem is infeasible. However, this is inconsistent with the optimal solution of a bilevel program being admissible. Actually, Audet et al. [AHJS99] have shown that determining whether there exists an x in X such that

$$Y(x) = \{y \geq 0 : By \geq d - Qx\}$$

is empty, is strongly \mathcal{NP} -complete. Equivalently, determining if BILP is bounded is strongly \mathcal{NP} -complete. This was achieved by constructing an auxiliary bilinear program BILP' (always bounded) such that BILP is unbounded whenever the optimal value of BILP' is positive. Based on this technique, the bilevel reformulation can be used to “solve” separable bilinear programs, whether they are bounded or not.

2.2 Complexity of linear bilevel programming

While one may derive complexity results about bilevel programs via the bilinear programming connection, it is instructive to perform directly reductions from standard combinatorial problems. After Jeroslow [Jer85] initially proved that LLBP is NP-hard, Hansen, Jaumard and Savard [HJS92] showed NP-hardness, using a reduction from KERNEL (see Garey and Johnson [GJ79]. Vicente, Savard and Júdice [VSJ94] strengthened these results and proved that checking strict and local optimality are also NP-hard. In this section, we present different proofs, based on a reduction from 3-SAT.

Let x_1, \dots, x_n be n Boolean variables and

$$F = \bigwedge_{i=1}^m (l_{i1} \vee l_{i2} \vee l_{i3})$$

be a 3-CNF formula involving m clauses with literals l_{ij} .³ To each clause $(l_{i1} \vee l_{i2} \vee l_{i3})$ we associate a linear Boolean inequality of the form

$$v_{i1} + v_{i2} + v_{i3} \geq 1$$

where

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

According to this scheme, the inequality

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

corresponds to the clause $(x_1 \vee \bar{x}_4 \vee x_6)$. Using matrix notation, the inequalities take the form

$$A_S x \geq \mathbf{1} + c$$

where A_S is a matrix with entries in $\{0, 1, -1\}$. By definition, S is satisfiable if and only if a feasible binary solution of this linear system exists. We have seen that it is indeed easy to force variables to take binary values through a bilevel program. The reduction makes use of this transformation.

THEOREM 1.4 *LLBP is strongly NP-hard.*

Proof. Consider the following LLBP:

$$\begin{aligned} \min_{x, z} \quad & F(x, z) = \sum_{i=1}^n z_i \\ \text{subject to} \quad & A_S x \geq \mathbf{1} + c \\ & 0 \leq x_i \leq 1, \quad i = 0, \dots, n \\ & z \in \operatorname{argmax} \left\{ \sum_{i=1}^n z_i : \right. \\ & \quad z_i \leq x_i \\ & \quad z_i \leq 1 - x_i \\ & \quad \left. z \geq 0 \right\} \end{aligned}$$

We claim that S is satisfiable if and only if the optimal solution of the LLBP is 0 (note that 0 is a lower bound on the optimal value).

³A literal consists in a variable or its negation.

First assume that S is satisfiable and let $x = (x_1, \dots, x_n)$ be a truth assignment for S . Then the first level constraints are verified and the sole feasible lower level solution corresponds to setting $z_i = 0$ for all i . Since this rational solution (x, z) achieves a value of 0, it is optimal. Assume next that S is not satisfiable. Any feasible x -solution must be fractionary and, since every rational solution satisfies $z_i = \min\{x_i, 1 - x_i\}$, at least one z_i must assume a positive value, and the objective $F(x, z)$ cannot be driven to zero. This completes the proof. \square

COROLLARY 2.1 *There is no fully polynomial approximation scheme for LLBP unless $\mathcal{P} = \mathcal{NP}$.*

To prove the local optimality results, Vicente, Savard and Júdeice adapted techniques developed by Pardalos and Schnitger [PS88] for non-convex quadratic programming, where the problem of checking (strict or not) local optimality was proved to be equivalent to solving a 3-SAT problem. The present proof differs slightly from the one developed in [VSJ94].

The main idea consists in constructing an equivalent but *degenerate* bilevel problem of 3-SAT. For that, we *augment* the Boolean constraints with an additional variable x_0 , change the right hand-side to $3/2$, and bound the x variables. For each instance S of 3-SAT, let us consider the constraint set:

$$\begin{aligned} A_S x + I x_0 &\geq \frac{3}{2} + c \\ \frac{1}{2} - x_0 &\leq x_i \leq \frac{1}{2} + x_0, \quad i = 1, \dots, n \\ x_i &\geq 0, \quad i = 0, \dots, n. \end{aligned}$$

Obviously, the solution $x^* = (0, 1/2, \dots, 1/2)$ satisfies the above linear inequalities, but this does not guarantee that S is satisfiable. Hence, we will consider a bilevel program that will have, at this solution, the same objective value than we would obtain if S is satisfiable.

THEOREM 2.1 *Checking strict local optimality in linear bilevel programming is \mathcal{NP} -Hard.*

Proof. Consider the following instance of a linear bilevel program:

$$\begin{aligned}
& \min_{x, l, m, z} && F(x, l, m, z) = \sum_{i=1}^n z_i \\
& \text{subject to} && A_S x + I x_0 \geq \frac{3}{2} + c \\
& && \frac{1}{2} - x_0 \leq x_i \leq \frac{1}{2} + x_0, \quad i = 1, \dots, n \\
& && x_i \geq 0, \quad i = 0, \dots, n \\
& && l, m, z \in \operatorname{argmax}_{\substack{l, m, z \\ i=1}}^n \left\{ \sum_{i=1}^n z_i : \right. \\
& && \quad x_i - l_i = \frac{1}{2} - x_0 \\
& && \quad x_i + m_i = \frac{1}{2} + x_0 \\
& && \quad z_i \leq l_i, \quad z_i \leq m_i, \quad i = 1, \dots, n \\
& && \quad \left. z \geq 0 \right\}
\end{aligned}$$

Let $x^* = (0, 1/2, \dots, 1/2)$ and $l^* = m^* = z^* = 0$. We claim that S is satisfiable if and only if the point (x^*, l^*, m^*, z^*) is not a strict minimum. Since all variables z_i are forced to be nonnegative then:

$$F(x, l, m, z) \geq 0.$$

First, assume that S is satisfiable. Let x_1, \dots, x_n be a true assignment for S and set, for any $x_0 \in [0, 1/2]$

$$\bar{x} = \begin{cases} 1/2 - x_0 & \text{if } x_i = 0, \\ 1/2 + x_0 & \text{if } x_i = 1, \end{cases}$$

i.e., \bar{x} satisfies the upper level constraints. Furthermore $\bar{l} = 0$, $\bar{m} = 0$ and $\bar{z} = 0$ is the optimal solution of the lower level problem for \bar{x} fixed. Hence $(\bar{x}, \bar{l}, \bar{m}, \bar{z})$ belongs to the induced region associated with the linear bilevel program. Since $F(\bar{x}, \bar{l}, \bar{m}, \bar{z}) = 0$, we claim that $(\bar{x}, \bar{l}, \bar{m}, \bar{z})$ is a global minimum of the linear bilevel program.

Clearly, $F(x, l, m, z) = 0$ if and only if $x_i \in \{\frac{1}{2} - x_0, \frac{1}{2} + x_0\}$, for all $i = 1, \dots, n$. If this last condition holds, then $l_i = 0$ or $m_i = 0$ and $z_i = 0$ for all $i = 1, \dots, n$ and $F(x, l, m, z) = 0$. Since x_0 can be chosen arbitrarily close to 0, x^* cannot be a strict local minimum.

Assume next that (x^*, l^*, m^*, z^*) is not a strict local minimum. There exists a rational point (x^1, l^1, m^1, z^1) such that $F(x^1, l^1, m^1, z^1) = 0$, and this point satisfies $l^1 = m^1 = z^1 = 0$ and $x^1 = \frac{1}{2} - x_0$ or $x^1 = \frac{1}{2} + x_0$ for

all i and some x_0 . Then the assignment

$$\begin{aligned} x_i &= 0 & \text{if } 1/2 - x_0, \\ x_i &= 1 & \text{if } 1/2 + x_0 \end{aligned}$$

is a truth assignment for S . \square

THEOREM 2.2 *Checking local optimality in linear bilevel programming is \mathcal{NP} -hard.*

The proof, which is based on complexity results developed in [PS88] and [VSJ94], will not be presented. Let us however mention that the underlying strategy consists in slightly discriminating against the rational points assuming value 0, through the addition of penalty factor with respect to x_0 , yielding the LLBP

$$\begin{aligned} \min_{x, l, m, z, w} \quad & F(x, l, m, z, w) = \sum_{i=1}^n z_i - \frac{1}{2n} \sum_{i=1}^n w_i \\ \text{subject to} \quad & A_S x \geq \frac{3}{2} + c \\ & \frac{1}{2} - x_0 \leq x_i \leq \frac{1}{2} + x_0, \quad i = 1, \dots, n \\ & x_i \geq 0, \quad i = 0, \dots, n \\ & l, m, z, w \in \operatorname{argmax} \left\{ \sum_{i=1}^n z_i - \sum_{i=1}^n w_i : \right. \\ & \quad x_i - l_i = \frac{1}{2} - x_0 \\ & \quad x_i + m_i = \frac{1}{2} + x_0 \\ & \quad z_i \leq l_i, \quad z_i \leq m_i, \quad i = 1, \dots, n \\ & \quad w_i \geq x_i - \frac{1}{2}, \quad w_i \geq \frac{1}{2} - x_i, \quad i = 1, \dots, n \\ & \quad \left. z, w \geq 0. \right\} \end{aligned}$$

3. Optimal pricing via bilevel programming

Although much attention has been devoted to linear bilevel programs, their mathematical structure does not fit many real life situations, where it is much more likely that interaction between conflicting agents occurs through the model's objectives rather than joint constraints. In this section, we consider such an instance that, despite its simple structure,

forms the paradigm that lies behind large-scale applications in revenue management and pricing, such as considered by Côté et al. [CMS03].

3.1 A simple pricing model

Let us consider a firm that wants to price independently (bundling is not allowed) a set of products aimed at customers having specific requirements and alternative purchasing sources. If the requirements are related in a linear manner to the resources (products), one obtains the bilinear-bilinear bilevel program (BBBP):

$$\begin{aligned} \text{BBBP :} \quad & \max_{t,x,y} && tx \\ & \text{s.t.} && (x,y) \in \arg \min_{x,y} (c+t)x + dy \\ & && Ax + By = b \\ & && x, y \geq 0, \end{aligned}$$

where t denotes the upper level decision vector, (c, d) the “before tax” price vector, (x, y) the consumption vector, (A, B) the “technology matrix” and b the demand vector. In the above, a trade-off must be achieved between high t -values that price the leader’s products away from the customer(s), and low prices that induce a low revenue.

In a certain way, the structure of BBBP is dual to that of LLBP, in that the constraint set is separable and interaction occurs only through the objective functions. The relationship between LLBP and BBBP actually goes further. By replacing the lower level program by its primal-dual characterization, one obtains the equivalent bilinear and single-level program

$$\begin{aligned} \max_{t,x,y} \quad & tx \\ \text{s.t.} \quad & Ax + By = b \\ & x, y \geq 0 \\ & \lambda A \leq c + t \\ & \lambda B \leq d \\ & (c + t - \lambda A)x = 0 \\ & (d - \lambda B)y = 0. \end{aligned}$$

Without loss of generality, one can set $t = \lambda A - c$. Indeed, if $x_i > 0$, $t_i = (\lambda A)_i - c_i$ follows from the next-to-last orthogonality conditions whereas, if $x_i = 0$, the leader’s objective is not affected by the value of t_i . Now, a little algebra yields:

$$tx = \lambda Ax - cx = \lambda(b - By) - cx = \lambda b - (cx + dy)$$

and one is left with a program involving a single nonlinear (actually bilinear and separable) constraint, that can be penalized to yield the bilinear program

$$\begin{array}{ll} \text{PENAL :} & \max_{x,y,\lambda} \quad \lambda b - (cx + dy) - M(d - \lambda B)y \\ & \text{s.t.} \quad Ax + By = b \\ & \quad x, y \geq 0 \\ & \quad \lambda B \leq d. \end{array}$$

Under mild feasibility and compactness assumptions, it has been shown by Labbé et al. [LMS98] that there exists a finite value M^* of the penalty parameter M such that, for every value of M larger than M^* , any optimal solution of the penalized problem satisfies the orthogonality constraint $(d - \lambda B)y = 0$, i.e., the penalty is exact.⁴ Since the penalized problem is bilinear and separable, optimality must be achieved at some extreme point of the feasible polyhedron. Moreover, the program can, using the techniques of Section 1.2.1.2, be reformulated as a linear bilevel program of a special type.

The reverse transformation, from a generic LLBP to BBBP, is not straightforward and could not be achieved by the authors. However, since BBBP is strongly \mathcal{NP} -hard, such polynomial transformation *must* exist.

3.2 Complexity

In this section, we consider a subclass of BBBP initially considered by Labbé et al., where the feasible set $\{(x, y) : Ax + By = b, x, y \geq 0\}$ is that of a multicommodity flow problem, without upper bound constraints on the links of the network. For a given upper level vector t , a solution to the lower level problem corresponds to assigning demand to shortest paths linking origin and destination nodes. This yields:

⁴Be careful though: the stationary points of the penalized and original problems need not be in one-to-one relationship!

$$\text{TOLL : } \begin{array}{ll} \max_{t,x,y} & t \sum_{k \in K} x^k \\ \text{s.t.} & (x^k, y^k) \in \left. \begin{array}{l} \arg \min_{x^k, y^k} tx^k + dy^k \\ Ax^k + By^k = b^k \\ x^k, y^k \geq 0 \end{array} \right\} \quad \forall k \in K, \end{array}$$

where (A, B) denotes the node-arc incidence matrix of the network, and b^k denotes the demand vector associated with the origin-destination pair, or “commodity” $k \in K$.

Note that since a common toll vector t applies to *all* commodities, TOLL does not quite fit the format of BBBP. However, by setting $x = \sum_{k \in K} x^k$ for both objectives⁵ and incorporating the compatibility constraint $x = \sum_{k \in K} x^k$ (at either level), we obtain a *bona fide* BBBP.

THEOREM 1.5 *TOLL is strongly NP-hard, even when $|K| = 1$.*

The proof relies on the reduction on the reformulation of 3-SAT as toll problem involving a single origin-destination pair, and is directly adapted from the paper by Roch et al. [RSM04]. Let x_1, \dots, x_n be n Boolean variables and

$$F = \bigwedge_{i=1}^m (l_{i1} \vee l_{i2} \vee l_{i3}) \quad (1.1)$$

be a 3-CNF formula consisting of m clauses with literals (variables or their negations) l_{ij} . For each clause, we construct a “cell”, i.e., a sub-network comprising one toll arc for each literal. Cells are connected by a pair of parallel arcs, one of which is toll-free, and by arcs linking literals that cannot be simultaneously satisfied (see Figure 1.3).

The idea is the following: if the optimal path goes through toll arc T_{ij} , then the corresponding literal l_{ij} is TRUE. The sub-networks are connected by two parallel arcs, a toll-free arc of cost 2 and a toll arc of cost 0, as shown in Figure 1.3.

If F is satisfiable, we want the optimal path to go through a single toll arc per sub-network (i.e., one TRUE literal per clause) and simultaneously want to make sure that the corresponding assignment of variables is consistent; i.e., paths that include a variable and its negation must be ruled out. For that purpose, we assign to every pair of literals corresponding to a variable and its negation an inter-clause toll-free arc

⁵This is allowed by the fact that the lower level constraints are separable by commodity.

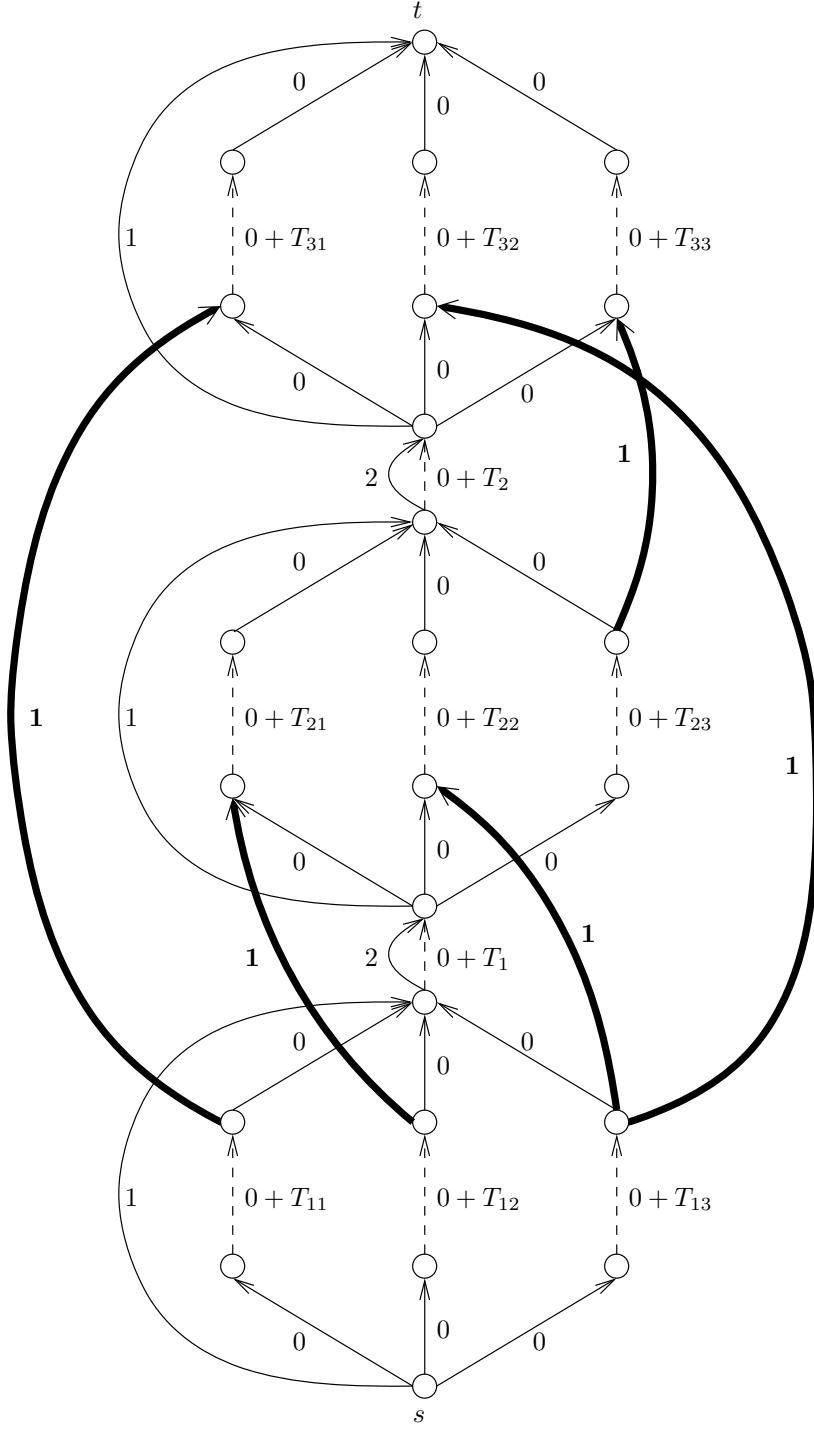


Figure 1.3. Network for the formula $(x_1 \vee x_2 \vee \bar{x}_3) \wedge (\bar{x}_2 \vee x_3 \vee \bar{x}_4) \wedge (\bar{x}_1 \vee x_3 \vee x_4)$. Inter-clause arcs are bold. Path through T_{12} , T_{22} , T_{32} is optimal ($x_2 = x_3 = \text{TRUE}$).

between the corresponding toll arcs (see Figure 1.3). As we will see, this implies that *inconsistent* paths, involving a variable and its negation, are suboptimal.

Since the length of a shortest toll-free path is $m + 2(m - 1) = 3m - 2$ and that of a shortest path with zero tolls is 0, $3m - 2$ is an upper bound on the revenue. We claim that F is satisfiable if and only if the optimal revenue is equal to that bound.

Assume that the optimal revenue is equal to $3m - 2$. Obviously, the length of the optimal path when tolls are set to 0 must be 0, otherwise the upper bound cannot be reached. To achieve this, the optimal path has to go through one toll arc per sub-network (it cannot use inter-clause arcs) and tolls have to be set to 1 on selected literals, $C + 1$ on other literals and 2 on tolls T_k , $\forall k$. We claim that the optimal path does not include a variable and its negation. Indeed, if that were the case, the inter-clause arc joining the corresponding toll arcs would impose a constraint on the tolls between its endpoints. In particular, the toll T_k immediately following the initial vertex of this inter-clause arc would have to be set at most to 1, instead of 2. This yields a contradiction. Therefore, the optimal path must correspond to a consistent assignment, and F is satisfiable (note: if a variable and its negation do not appear on the optimal path, this variable can be set to any value).

Conversely if F is satisfiable, at least one literal per clause is TRUE in a satisfying assignment. Consider the path going through the toll arcs corresponding to these literals. Since the assignment is consistent, the path does not simultaneously include a variable and its negation, and no inter-clause arc limits the revenue. Thus, the upper bound of $3m - 2$ is reached on this path.

Another instance, involving several commodities but restricting each path to use a single toll arc, also proved \mathcal{NP} -hard. Indeed, consider the “river tarification problem”, where users cross a river by either using one of many toll bridges, or by flying directly to their destination on a toll-free arc. The proof of \mathcal{NP} -completeness also makes use of 3-SAT, but there is a twist: each cell now corresponds to a variable rather than a clause, and is thus “dual” to the previous transformation (see Grigoriev et al. [GvHvdK⁺04] for details). Apart of its elegance, the dual reduction has the advantage of being related to the corresponding optimization problem, i.e., one can maximize the number of satisfied clauses by solving the related TOLL problem. This is not true of the primal reduction, where the truth assignment is only valid when the Boolean formula can be satisfied. Indeed, the solution of the TOLL reduction may attain a near-optimal value of $3m - 3$ without any clause being satisfied, thus making the truth assignment of the variables irrelevant. For instance,

consider an instance where a variable and its negation appear as literals in the first and last clauses.⁶ Then, a revenue of $3m - 3$, one less than the optimal revenue, is achieved on the path that goes through the two literals and the toll-free link between them, by setting the tolls on the two toll arcs of that path to 0 and $3m - 3$ respectively.

We conclude this section by mentioning that TOLL is polynomially solvable when the number of toll arcs is bounded by some constant. If the set of toll arcs reduces to a singleton, a simple ordering strategy can be applied (see [LMS98]). In the general case, path enumeration yields a polynomial algorithm that is unfortunately not applicable in practice (see Grigoriev et al. [GvHvdK⁺04]). Other polynomial cases have been investigated by van Hoesel et al. [vHvdKM⁺03].

3.3 The traveling salesman problem

Although the relationship between the traveling salesman problem (TSP in short) and TOLL is not obvious, the first complexity result involved TSP or, to be more precise, the Hamiltonian path problem (HPP). The reduction considered in [LMS98] goes as follows: Given a directed graph with n nodes, among them two distinguished nodes: an origin s and a destination t the destination, we consider the graph obtained by creating a toll-free arc from s to t , with length $d_{st} = n - 1$. Next, we endow the remaining arcs, all toll arcs, with cost -1 and impose a lower bound of 2 on all of them. Then, it is not difficult to see that the maximal toll revenue, equal to $2n - 2$, is obtained by setting $t_a = 2$ on the arcs of any Hamiltonian path, and $t_a = n + 1$ elsewhere.

The weakness of that reduction is that it rests on two assumptions that are not required in the reductions presented in the previous sections, that is, negativity of arc lengths and lower bounds on toll values. Notwithstanding, the relationship between TOLL and TSP has proved fruitful. To see this, let us follow Marcotte et al. [MSS03] and consider a TSP involving a graph G and a length vector c . First, we transform the TSP into an HPP by duplicating the origin node s and replacing all arcs (i, s) by arcs from i to t . It is clear that the solutions to TSP and HPP are in one-to-one correspondence. Second, we incorporate a toll-free arc (s, t) with cost n , we set the fixed cost of the remaining arcs to $-1 + c_a/L$ and the lower bounds on tolls to $2 - c_a/L$, where L is some suitably large constant, $L = n \times \max_a \{c_a\}$ for instance. Then, any

⁶Remark: The clauses involving the two opposite literals can always be made the first and the last, through a straightforward permutation. This shows that the model is sensitive to the rearrangement of clauses.

solution to the toll problem on the modified network yields a shortest Hamiltonian path. This toll problem takes the form

$$\begin{aligned}
\max_{t,x,y} \quad & \sum_a t_a x_a \\
(x,y) \in \quad & \arg \min_{x,y} \sum_a (-1 + c_a/L + t_a) x_a + n y_{st} \\
\text{s.t.} \quad & \text{flow conservation} \\
& x \geq 0.
\end{aligned}$$

Replacing the lower level linear program by its optimality conditions, one obtains a linear program including additional complementarity constraints. The latter, upon the introduction of binary variables, can be linearized to yield a MIP formulation of the TSP that, after some transformations, yields:

$$\begin{aligned}
\min_{x,u} \quad & \sum_a c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_j x_{ij} = 1 \quad \forall i \\
& \sum_i x_{ij} = 1 \quad \forall j \\
& u_i - u_j \leq (n-2) + (1-n)x_{ij} + (3-n)x_{ji} \quad \forall (i,j) \\
& u_j \leq (n-2) + (3-n)x_{1j} + x_{j(n+1)} \quad \forall j \neq 1 \\
& u_j \geq (n-3)x_{j(n+1)} - x_{1j} + 2 \quad \forall j \neq 1 \\
& x \text{ binary valued,}
\end{aligned}$$

where u corresponds to the dual vector associated with the lower level program. It is in a certain way surprising, and certainly of theoretical interests that, through standard manipulations, one achieves the mixed integer program. Note that this program is nothing but the lifted formulation of the Miller-Tucker-Zemlin constraints derived by Desrochers and Laporte [DL91], where the three constraints involving the vector u are facet-defining.

For complete directed graphs, the analysis supports a multicommodity extension, where each commodity is assigned to a subtour between two prespecified vertices. More precisely, let $[v_1, v_2, \dots, v_{|K|}]$ be a sequence of vertices. Then, the flow for commodity $k \in K$ must follow a path from vertex v_k to v_{k+1} ⁷, and the sequence of such paths must form

⁷By convention, $v_{K+1} \equiv v_1$.

a Hamiltonian circuit. If the number of commodity is 3 or less, the ordering of the vertices is irrelevant. If $|K|$ is more than 3, it is yet possible to find a set of vertices that are extreme points of the convex hull of vertices, together with the order in which they must be visited in some optimal tour (see Flood [Flo56]).

When applied to graphs from the TSPLIB library [TSP], the linear relaxation of the three-commodity reformulation provides lower bounds of quality comparable to those obtained by the relaxation proposed by Dantzig, Fulkerson and Johnson [DFJ54]. This is all the more surprising in the view that the latter formulation is exponential, while the former is in $O(n^2)$.

3.4 Final considerations

This chapter has provided a very brief overview of two important classes of bilevel programs, from the perspective of combinatorial optimization. Those classes are not the only ones to possess a combinatorial nature. Indeed, let us consider a bilevel program (or an MPEC) where the induced region is the union of polyhedral faces.⁸ A sufficient condition that an optimal solution be attained at an extreme point of the induced region is then that the upper level objective be concave in both upper and lower level variables. An interesting situation also occurs when the upper level objective is quadratic and *convex*. In this case, the solution of the problem restricted to a polyhedral face occurs at an extreme point of the primal-dual polyhedron, and it follows that the problem is also combinatorial.

Actually, bilevel programs almost always integrate a combinatorial element. For instance, let us consider the general bilevel program:

$$\begin{aligned} \min_{x,y} \quad & f(x, y) \\ \text{s.t.} \quad & y \in \arg \min_y g(x, y) \\ & G(x, y) \leq 0. \end{aligned}$$

Under suitable constraints (differentiability, convexity and regularity of the lower level problem), one can replace the lower level problem by its Kuhn-Tucker conditions and obtain the equivalent program

⁸This situation is realized when the lower level is a linear, a convex quadratic, or a linear complementarity problem, and joint constraints, whenever they exist, are linear.

$$\begin{aligned}
\text{BLKKT :} \quad & \min_{x,y} && f(x,y) \\
& \text{s.t.} && G(x,y) \leq 0 \\
& && \nabla_y g(x,y) + \lambda \nabla_y G(x,y) = 0 \\
& && \lambda G(x,y) = 0.
\end{aligned}$$

If the set of active constraints were known a priori, BLKKT would reduce to a standard nonlinear program. Provided that f , g and each of the G_i 's be convex, the last constraint could yet make it non-convex, albeit “weakly”, in the sense that replacing all functions by their quadratic approximations would make the bilevel problem convex. The main computational pitfall is actually the identification of the active set. This two-sided nature of bilevel programming and MPEC is well captured in the formulation proposed by Scholtes [Sch04], which distinguishes between the continuous and combinatorial natures of MPECs. By rearranging variables and constraints, one can reformulate BLKKT as the generic program

$$\begin{aligned}
& \min_x && f(x) \\
& \text{s.t.} && G(x) \in \mathcal{Z}.
\end{aligned}$$

If \mathcal{Z} is the negative orthant, this is nothing more than a standard nonlinear program. However, special choices of \mathcal{Z} , may force pairs of variables to be complementary. It is then ill-advised to linearize \mathcal{Z} , and the right approach is to develop a calculus that does not sidestep the combinatorial nature of the set \mathcal{Z} . Along that line of reasoning, Scholtes proposes an SQP (Sequential Quadratic Programming) algorithm that leaves \mathcal{Z} untouched and is guaranteed, under mild assumptions, to converge to a strong stationary solution. While this approach is satisfactory from a local analysis point of view, it does not settle the main challenge, that is, aiming for an optimal or near-optimal solution. In our view, progress in this direction will be achieved by addressing problems with specific structures, such as the BBBP.

Acknowledgments

We would like to thank the following collaborators with whom most of the results have been obtained: Charles Audet, Luce Brotcorne, Benoît Colson, Pierre Hansen, Brigitte Jaumard, Joachim Júdice, Martine Labbé, Sébastien Roch, Frédéric Semet and Luís Vicente.

References

- S. Alarie, C. Audet, B. Jaumard, and G. Savard. Concavity cuts for disjoint bilinear programming. *Mathematical Programming*, 90:373–398, 2001.
- C. Audet, P. Hansen, B. Jaumard, and G. Savard. Links between linear bilevel and mixed 0-1 programming problems. *Journal of Optimization Theory and Applications*, 93:273–300, 1997.
- C. Audet, P. Hansen, B. Jaumard, and G. Savard. A symmetrical linear maxmin approach to disjoint bilinear programming. *Mathematical Programming*, 85:573–592, 1999.
- C. Audet. *Optimisation globale structurée : propriétés, équivalences et résolution*. PhD thesis, École Polytechnique de Montréal, 1997.
- J. F. Bard. *Practical bilevel optimization – Algorithms and applications*. Kluwer Academic Publishers, 1998.
- E.M.L. Beale and R.E. Small. Mixed integer programming by a branch and bound technique. In *Proceedings of the 3rd IFIP Congress*, pages 450–451, 1965.
- J.-P. Côté, P. Marcotte, and G. Savard. A bilevel modeling approach to pricing and fare optimization in the airline industry. *Journal of Revenue and Pricing Management*, 2:23–36, 2003.
- W. Candler and R. Norton. Multilevel programming. Technical Report 20, World Bank Development Research Center, Washington D.C., 1977.
- S. Dempe. *Foundations of Bilevel Programming*, volume 61 of *Nonconvex optimization and its applications*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.
- G.B. Dantzig, D.R. Fulkerson, and S.M. Johnson. Solution of a large-scale traveling salesman problem. *Operations Research*, 2:393–410, 1954.
- M. Desrochers and G. Laporte. Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. *Operations Research Letters*, 10:27–36, 1991.
- M.M. Flood. The traveling salesman problem. *Operations Research*, 4:61–75, 1956.

- M. Fukushima and J.S. Pang. Complementarity constraint qualifications and simplified B-stationarity conditions for mathematical programs with equilibrium constraints. *Computational Optimization and Applications*, 13:111–136, 1999.
- M. Fukushima. Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems. *Mathematical Programming*, 53:99–110, 1992.
- M.R. Garey and D.S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness*. W.H. Freeman, New York, 1979.
- A. Grigoriev, S. van Hoesel, A.F. van der Kraaij, M. Uetz, and M. Bouhtou. Pricing network edges to cross a river. Technical Report RM04009, Maastricht Economic Research School on Technology and Organisation, Maastricht, The Netherlands, 2004.
- P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on Scientific and Statistical Computing*, 13:1194–1217, 1992.
- W.W. Hogan. Point-to-set maps in mathematical programming. *SIAM Review*, 15:591–603, 1973.
- R.G. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical Programming*, 32:146–164, 1985.
- M. Kočvara and J.V. Outrata. On optimization of systems governed by implicit complementarity problems. *Numerical Functional Analysis and Optimization*, 15:869–887, 1994.
- H. Konno. Bilinear programming: Part ii. applications of bilinear programming. Technical Report Technical Report No.71-10, Operations Research House, Department of Operations Research, Stanford University, Stanford, 1971.
- P. Loridan and J. Morgan. Weak via strong Stackelberg problem: New results. *Journal of Global Optimization*, 8:263–287, 1996.
- M. Labbé, P. Marcotte, and G. Savard. A bilevel model of taxation and its applications to optimal highway pricing. *Management Science*, 44:1608–1622, 1998.
- Z.Q. Luo, J.S. Pang, and D. Ralph. *Mathematical programs with equilibrium constraints*. Cambridge University Press, 1996.
- H. Mills. Equilibrium points in finite games. *Journal of the Society for Industrial and Applied Mathematics*, 8:397–402, 1960.
- P. Marcotte, G. Savard, and F. Semet. A bilevel programming approach to the travelling salesman problem. *Operations Research Letters*, 32:240–248, 2003.
- P. Marcotte, G. Savard, and D.L. Zhu. A trust region algorithm for nonlinear bilevel programming. *Operations Research Letters*, 29:171–179, 2001.

- J.F. Nash. Noncooperative games. *Annals of Mathematics*, 14:286–295, 1951.
- P. Pardalos and G. Schnitger. Checking local optimality in constrained quadratic programming is np-hard. *Operations Research Letters*, 7:33–35, 1988.
- S. Roch, G. Savard, and P. Marcotte. An approximation algorithm for Stackelberg network pricing, 2004.
- G. Savard. *Contribution à la programmation mathématique à deux niveaux*. PhD thesis, Ecole Polytechnique de Montréal, Université de Montréal, April 1989.
- S. Scholtes. Nonconvex structures in nonlinear programming. *Operations Research*, 52:368–383, 2004.
- K. Shimizu, Y. Ishizuka, and J.F. Bard. *Nondifferentiable and two-level mathematical programming*. Kluwer Academic Publishers, 1997.
- S. Scholtes and M. Stöhr. Exact penalization of mathematical programs with equilibrium constraints. *SIAM Journal on Control and Optimization*, 37(2):617–652, 1999.
- H. Stackelberg. *The theory of market economy*. Oxford University Press, Oxford, 1952.
- TSPLIB. A library of traveling salesman problems, available at <http://www.iwr.uni-heidelberg.de/groups/comopt/software/tsplib95>.
- S. van Hoesel, A.F. van der Kraaij, C. Mannino, G. Oriolo, and M. Bouhtou. Polynomial cases of the tariffication problem. Technical Report RM03053, Maastricht Economic Research School on Technology and Organisation, Maastricht, The Netherlands, 2003.
- L.N. Vicente, G. Savard, and J.J. Júdice. Descent approaches for quadratic bilevel programming. *Journal of Optimization Theory and Applications*, 81:379–399, 1994.
- L.N. Vicente, G. Savard, and J.J. Júdice. The discrete linear bilevel programming problem. *Journal of Optimization Theory and Applications*, 89:597–614, 1996.