Bilevel Programming and the Separation Problem

Andrea Lodi

DEIS, University of Bologna, Italy

Ted K. Ralphs

Department of Industrial and Systems Engineering, Lehigh University, USA

Gerhard J. Woeginger

Department of Mathematics and Computer Science, TU Eindhoven, The Netherlands

COR@L Technical Report 12T-015-R2

Bilevel Programming and the Separation Problem

ANDREA LODI*[1], TED K. RALPHS†[2], AND GERHARD J. WOEGINGER‡[3]

[1]DEIS, University of Bologna, Italy
[2]Department of Industrial and Systems Engineering, Lehigh University, USA
[3]Department of Mathematics and Computer Science, TU Eindhoven, The Netherlands

Original Publication: March 26, 2012
Last Revised: May 29, 2013

Abstract

In recent years, branch-and-cut algorithms have become firmly established as the most effective method for solving generic mixed integer linear programs (MILPs). Methods for automatically generating inequalities valid for the convex hull of solutions to such MILPs are a critical element of branch-and-cut. This paper examines the nature of the so-called separation problem, which is that of generating a valid inequality violated by a given real vector, usually arising as the solution to a relaxation of the original problem.

We show that the problem of generating a maximally violated valid inequality often has a natural interpretation as a bilevel program. In some cases, this bilevel program can be easily reformulated as a simple single-level mathematical program, yielding a standard mathematical programming formulation for the separation problem. In other cases, no such polynomial-size single-level reformulation exists unless the polynomial hierarchy collapses to its first level (an event considered extremely unlikely in computational complexity theory). We illustrate our insights by considering the separation problem for two well-known classes of valid inequalities.

1 Introduction

We consider a *mixed integer linear program* (MILP) of the form

$$\min\{c^\top x \mid Ax \geq b,\ x \in \mathbb{Z}_+^{|I|} \times \mathbb{R}_+^{|C|}\}, \tag{1}$$

where $A \in \mathbb{Q}^{m\times n}$, $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$, $I \subseteq N = \{1, \ldots, n\}$ is the set of indices of components that must take integer values in any feasible solution and $C = N \setminus I$ consists of the indices of the remaining components. We assume that other bound constraints on the variables (if any) are included among the problem constraints.

*E-mail: `andrea.lodi@unibo.it`, supported by MIUR, Italy under PRIN2009 project
†E-mail: `ted@lehigh.edu`, supported by NSF Grant CMMI-0728011
‡E-mail: `gwoegi@win.tue.nl`

The *continuous* or *linear programming* (LP) relaxation of the above MILP is the mathematical program obtained by dropping the integrality requirement on the variables in I, namely

$$\min_{x \in \mathcal{P}} c^\top x, \tag{2}$$

where $\mathcal{P} = \{x \in \mathbb{R}^n_+ \mid Ax \geq b\}$ is the polyhedron described by the linear constraints of the MILP (1). It is not difficult to show that the convex hull of the set of feasible solutions to (1) is also a polyhedron, which we denote by $\mathcal{P}_I = \text{conv}(\mathcal{P} \cap (\mathbb{Z}^I_+ \times \mathbb{R}^C_+))$. This means that in principle, the MILP (1) is equivalent to a linear program over this implicitly defined polyhedron. In fact, Grötschel et al. [1981] showed that, under mild assumptions, the (linear) optimization problem over $\mathcal{P}_I$ is polynomially equivalent to the *separation problem*, defined formally in Section 2 below, which is the problem of determining whether a given vector is in $\mathcal{P}_I$ and if not, producing a hyperplane separating the vector from $\mathcal{P}_I$.

In this paper, we examine the nature of this separation problem for structured classes of valid inequalities. We show that the separation problem has a natural interpretation as a *bilevel programming problem*, which is equivalent to the optimization problem over the so-called *closure* for the class. In many cases, this bilevel programming problem has a re-formulation as a single-level mathematical program, which implies that there is a short certificate of validity for the associated class of valid inequalities. However, this is not always the case. Our main result is to show formally that for the strongest version of the *generalized subtour elimination constraints* (GSECs) for the well-known *Capacitated Vehicle Routing Problem* (CVRP), the separation problem cannot be reformulated unless the *polynomial hierarchy* (described below) collapses to its first level. This collapse is considered to be extremely unlikely by computational complexity theorists. Before getting to the main result, we introduce the necessary concepts and definitions from both complexity theory and bilevel programming.

1.1 Bilevel Programming

A *bilevel mixed integer linear program* (MIBLP) is a generalization of a standard MILP that models hierarchical decision processes. In a MIBLP, the variables are split into a set of *upper-level variables*, denoted by x below, and a set of *lower-level variables*, denoted by y below. Conceptually, the values of the upper-level variables are fixed first, subject to the restrictions of a set of *upper-level constraints*, after which the second-stage variables are fixed by solving an MILP parameterized on the fixed values of the upper-level variables. The canonical MIBLP is given by

$$\begin{aligned}\min\big\{c^1 x + d^1 y \mid\ & x \in \mathcal{P}_U \cap (\mathbb{Z}^{I_1} \times \mathbb{R}^{C_1}), \\ & y \in \text{argmin}\{d^2 y \mid y \in \mathcal{P}_L(x) \cap (\mathbb{Z}^{I_2} \times \mathbb{R}^{C_2})\}\big\},\end{aligned}$$

where

$$\mathcal{P}_U = \big\{x \in \mathbb{R}^{n_1} \mid A^1 x \geq b^1, x \geq 0\big\}$$

is the polyhedron defining the *upper-level feasible region*;

$$\mathcal{P}_L(x) = \big\{y \in \mathbb{R}^{n_2} \mid G^2 y \geq b^2 - A^2 x, y \geq 0\big\}$$

is the polyhedron defining the *lower-level feasible region* with respect to a given $x \in \mathbb{R}^{n_1}$; $A^1 \in \mathbb{Q}^{m_1 \times n_1}$; $b^1 \in \mathbb{Q}^{m_1}$; $A^2 \in \mathbb{Q}^{m_2 \times n_1}$, $G^2 \in \mathbb{Q}^{m_2 \times n_2}$; and $b^2 \in \mathbb{Q}^{m_2}$. The index sets $I_1 \subseteq N_1 =$

$\{1, \ldots, n_1\}$, $I_2 \subseteq N_1 = \{1, \ldots, n_1\}$, $C_1 = N_1 \setminus I_1$, and $C_2 = N_2 \setminus I_2$ are the bilevel counterparts of the sets I and C defined previously. For more detailed information, Colson et al. [2005] provide an introduction to and comprehensive survey of the bilevel programming literature, while Moore and Bard [1990] introduce the discrete case. Dempe [2003] provides a detailed bibliography.

1.2 The Polynomial Hierarchy

Informally, the polynomial hierarchy is a scheme for classifying multi-level decision problems that extends the well-known complexity classes P and NP to problems with multiple decision-makers (and multiple objectives, in the case of optimization models), such as those arising in multi-round games. Level zero of the hierarchy is denoted Σ_0^P and contains the problems that can be solved in polynomial time; in other words, $\Sigma_0^P = P$. The first level of the hierarchy is formed by the problems in $NP = \Sigma_1^P$ and by their negated versions in $co\text{-}NP = \Pi_1^P$. Level k of the hierarchy consists of the problems in class Σ_k^P together with the negated versions of these problems in class Π_k^P. Roughly speaking, a problem is in class Σ_k^P if it can be solved in nondeterministic polynomial time, given an oracle for solving problems in the class Σ_{k-1}^P.

Equivalently (and still roughly speaking), the class Σ_k^P contains the problems that can be expressed by a logical formula that consists of a sequence of k existentially or universally quantified discrete variables, followed by a Boolean predicate that depends on the variables and on the given instance and that can be evaluated in polynomial time. The first quantifier in the logical formula is an existential quantifier. Hence, the class $NP = \Sigma_1^P$ contains the problems of the form $?\exists x \in X : S(x)$, where $S(x)$ is the aforementioned Boolean predicate. The class Σ_2^P contains the problems of the form $?\exists x \in X$ s.t. $S(x, y)\ \forall y \in Y$, and so on. Note that

$$\Sigma_0^P \subseteq \Sigma_1^P \subseteq \Sigma_2^P \subseteq \Sigma_3^P \subseteq \ldots \subseteq \Sigma_k^P \subseteq \Sigma_{k+1}^P \subseteq \ldots$$

It is not known whether any of these inclusions is strict, but in the computational complexity community, it is strongly conjectured that *all* of them are strict. It has been shown that if $\Sigma_k^P = \Sigma_{k+1}^P$ for some k, then this would imply $\Sigma_k^P = \Sigma_j^P$ for all $j \geq k+1$; in this case, one would say that the polynomial hierarchy collapses to its kth level. A collapse to level zero would mean that $P = NP$. A collapse to any other fixed level would have weaker consequences, but is still considered to be extremely unlikely.

Stockmeyer [1977]'s foundational work introduced the polynomial hierarchy. It also exhibited for every $k \in \mathbb{N}$ certain quantified versions of the well-known satisfiability problem that are complete for the class Σ_k^P and that thus constitute the most difficult problems in the class. A simple way to envision these quantified versions is to interpret them as a multi-round game with k players who are together determining the values of the variables in a first-order Boolean formula. Each player in turn picks the values of a designated subset of the variables, with each "odd" player attempting to force the expression to eventually (once all variables are fixed) evaluate true by tying the hands of the even player who follows her, while said "even" player attempts to find a way to make the expression false. In fact, it is only necessary for there to be two players, an "even" player and an "odd" player, for this game to work as described. It is the number of rounds, not the number of players, that determines the complexity. The relationship between Stockmeyer [1977]'s decision games and optimization was first noted by Jeroslow [1985], who showed that the rather contrived games of Stockmeyer [1977] can be reduced to a similar set of k-level binary optimization problems and thus showed that k-level discrete optimization problems are Σ_k^P-hard, even when the variables are binary and all constraints are linear.

1.3 Membership and Separation Problems

A *valid inequality* for a set $\mathcal{S} \subseteq \mathbb{R}^n$ is a pair (α, β), where $\alpha \in \mathbb{R}^n$ is the *coefficient vector* and $\beta \in \mathbb{R}$ is a *right-hand side*, such that $\alpha^\top x \geq \beta$ for all $x \in \mathcal{S}$. Associated with any valid inequality (α, β) is the half-space $\{x \in \mathbb{R}^n \mid \alpha^\top x \geq \beta\}$, which must contain $\mathcal{S}$. It is easy to see that any inequality valid for $\mathcal{S}$ is also valid for the convex hull of $\mathcal{S}$.

For a polyhedron $\mathcal{Q} \subseteq \mathbb{R}^n$, the so-called *separation problem* is to generate a valid inequality violated by a given vector. Formally, we define the problem as follows.

Definition 1 *The* separation problem *for a polyhedron $\mathcal{Q}$ is to determine for a given $\hat{x} \in \mathbb{R}^n$ whether or not $\hat{x} \in \mathcal{Q}$ and if not, to produce an inequality $(\bar{\alpha}, \bar{\beta}) \in \mathbb{R}^{n+1}$ valid for $\mathcal{Q}$ and for which $\bar{\alpha}^\top \hat{x} < \bar{\beta}$.*

The separation problem is not a traditional decision problem, as stated, since it requires additional output in case the answer is in the negative. The *membership problem* and its complement, the *non-membership problem*, are decision problems closely related to the separation problem. The membership problem associated with a polyhedron $\mathcal{Q}$ and a given point $\hat{x} \in \mathbb{R}^n$ is the question of whether $\hat{x}$ is a member of $\mathcal{Q}$. If $\mathcal{Q}$ is described either explicitly by a set of linear inequalities or as the convex hull of solutions to an explicitly described integer program, there is a short certificate for the membership problem. This is because we can demonstrate that $\hat{x} \in \mathcal{Q}$ by exhibiting a polynomial number of points in $\mathcal{Q}$ whose convex combination yields $\hat{x}$. Carathéodory's Theorem assures us that such a convex combination exists when $\hat{x} \in \mathcal{Q}$.

Assuming $\mathcal{Q}$ is bounded, one way of constructing a certificate of membership is to solve a system of equations in which the columns of the coefficient matrix are the extreme points of $\mathcal{Q}$, the right hand-side is $\hat{x}$, and the variable values to be determined are the weights. By the result of Grötschel et al. [1981], solving this system as a (feasibility) linear program is polynomially equivalent to solving the optimization problem over $\mathcal{Q}$, which is to determine $\min_{x \in \mathcal{Q}} d^\top x$ for a given $d \in \mathbb{R}^n$ (more about this below). Note that a certificate of membership is also a certificate for an upper bound on the optimal solution value of the optimization problem over $\mathcal{Q}$ and the membership problem is thus closely related to the decision version of the optimization problem over $\mathcal{Q}$.

The non-membership problem may not have a short certificate if optimization over $\mathcal{Q}$ is an *NP*-hard problem. Although we can obtain a valid inequality violated by $\hat{x}$ from the Farkas proof of infeasibility, certifying the validity of this inequality requires certifying a *lower bound* on the optimal solution value of an optimization problem over $\mathcal{Q}$. This is a universally quantified decision problem for which a short certificate can only exist if *NP* = *co-NP* whenever optimization over $\mathcal{Q}$ is an *NP*-hard problem.

An optimization problem closely associated with the separation problem is the *maximally violated valid inequality problem* (MVVIP) with respect to $\hat{x} \in \mathbb{R}^n$ and a (bounded) polyhedron $\mathcal{Q}$. This problem can in principle be stated as the linear program

$$\min \quad \alpha^\top \hat{x} - \beta \tag{3}$$

$$\beta \leq \alpha^\top x, \quad \forall x \in \mathcal{E} \tag{4}$$

where $\mathcal{E}$ is the set of extreme points of $\mathcal{Q}$. Problem (3)-(4) takes value zero if there is no violated inequality (the trivial solution $(\alpha, \beta) = (\mathbf{0}, 0)$ is feasible), while it is otherwise unbounded. The unboundedness arises because scaling can increase the degree of violation of any given violated valid inequality by an arbitrary amount. In practice, the solution to this linear program must be normalized in some appropriate way, requiring the addition of one or more normalization constraints to (3)-(4). There are a number of alternative ways in which this could be accomplished, e.g., by adding the constraint $\|\alpha\|_1 = 1$. The difficulty of solving the resulting mathematical program, as well as which valid inequalities may be produced by solving the MVVIP, depends on the selected normalization. This is largely an empirical issue which is beyond the scope of the present work. We refer the reader, e.g., to Fischetti et al. [2011] for details. Note that the linear program (3)-(4) is precisely the dual of the linear programming form of the membership problem mentioned earlier. Solving it is polynomially equivalent to both optimization over $\mathcal{Q}$ and the associated separation problem, again by the result of Grötschel et al. [1981].

1.4 Overview

The remainder of the paper is organized as follows. In Section 2, we explain the bilevel nature of the separation problem. In Section 3, we describe in detail some classes of valid inequalities and analyze the separation problem for each of them in the context of the framework we have laid out in Section 2. In Section 3.1, we consider the well-known class of *disjunctive valid inequalities* for general MILPs and show that in this particular case it is quite straightforward to convert the associated bilevel linear program into a single-level mathematical program. In Section 3.2, we show that this reformulation is not possible for some classes of valid inequalities unless $\Sigma_2^P = NP$ (in other words: unless the polynomial hierarchy collapses to its first level). For the strongest version of the GSECs for the CVRP, we show formally that the separation problem is complete for the class Σ_2^P. Two more examples of separation problems that admit a natural bilevel formulation are discussed in Section 3.3. Finally, in Section 5, we draw some conclusions.

2 The Separation Problem and Bilevel Programming

2.1 Classes of Valid Inequalities

To improve tractability, valid inequalities are often generated by solving (either exactly or approximately) the MVVIP for one or more relaxations of the original problem. These relaxations may arise in considering valid inequalities from a specific *family* or *class*, i.e., inequalities that share a special structure. Applegate et al. [2006] called this paradigm for generation of valid inequalities the *template paradigm*. Generally speaking, a *class* of valid inequalities for a given set $\mathcal{S}$ is simply a subset of all valid inequalities for $\mathcal{S}$. Such subsets can be defined in a number of ways and may be either finite or infinite. Associated with any given class $\mathcal{C} \subseteq \mathbb{R}^{n+1}$ is its *closure* $\mathcal{P}_\mathcal{C}$, defined to be the intersection of all half-spaces associated with inequalities in the class. If the class is finite, then the closure is a polyhedron. Otherwise, it may or may not be a polyhedron.

The MVVIP with respect to a given closure is to produce a valid inequality with maximum violation from a particular class. To formulate such a problem, we must have a method of verifying membership in the class. Membership may be certified either by associating members of the class with a certain well-defined combinatorial structure from which the inequality can be both constructed and verified or by explicitly describing the closure and certifying validity through an optimization oracle. The structure, when provided, may itself constitute a certificate of validity for a given valid inequality and hence also a certificate of non-membership for any solution that violates the valid inequality. Next, we describe the role of proofs of validity in more detail.

2.2 Proofs of Validity

In general, any procedure to produce a valid inequality must implicitly produce a proof of validity. The complexity of producing this proof of validity is closely related to the complexity of the MVVIP. The *validity verification problem* (VVP) is that of verifying the validity of a given inequality with respect to a given polyhedron. The VVP is a decision problem that is usually easier (in a complexity sense) than that of actually producing the inequality.

To illustrate, let us consider the problem of proving the validity of a given inequality with respect to an explicitly described polyhedron. This problem is easy, since we have simply to produce weights with which the inequalities describing the polyhedron can be combined to generate the given inequality. This can be done by solving a (feasibility) linear program. Consider the polyhedron $\mathcal{P}$ from (2), for example. We have that (α, β) is a valid inequality if and only if $\exists u \in \mathbb{R}^m_+$ such that

$$\alpha \quad \geq uA, \tag{5}$$

$$\beta \quad \leq ub. \tag{6}$$

Thus, the set of all valid inequalities for a polyhedron is itself a polyhedron. There are short certificates for both the membership and non-membership problems, and the MVVIP itself is polynomially solvable.

This basic principle can be extended to proofs of validity for unions of polyhedra. This technique is at the core of Balas [1979]'s disjunctive procedure, which is to generate inequalities valid for the convex hull of the union of polyhedra associated with the individual terms of a fixed linear disjunction. It is easy to show that in this case too, there are short certificates for both the membership and non-membership problems, and the MVVIP itself is polynomially solvable. We describe more details of this case in Section 3.1 below.

In general, proving validity can be understood as equivalent to the problem of certifying a lower bound on the optimal value of an optimization problem over $\mathcal{P}_\mathcal{C}$. Given (α, β), we need to know that

$$\beta \leq \min_{x \in \mathcal{P}_\mathcal{C}} \alpha^\top x. \tag{7}$$

Note the distinction between this and the problem of proving validity for $\mathcal{P}_I$ directly. Although this problem is easier in general (since optimization over the closure is relaxation of the original problem), we will see later that it may actually be more difficult in some cases.

For some classes of valid inequalities, verifying validity is easy. For Chvátal-Gomory (C-G) cuts for pure integer programs, for example, proving validity consists of finding a combination of the inequalities from the original formulation (as in (5)–(6)) that produces an integral left-hand side vector and then showing that rounding β up to the nearest integer value produces the inequality in question. There is thus a certificate for the non-membership problem of the associated closure (called the elementary closure) in this case, but this non-membership problem is nevertheless *NP*-complete. For classes for which the inequalities arise from combinatorial structure (such as comb inequalities for TSP, odd holes for stable set, or blossoms for matching), the proof of validity for members of the class arises from the fact that any inequality conforming to a certain template structure is provably valid, though many of these inequalities can be proven valid essentially by applying the C-G procedure with specifically structured weights.

It is important to distinguish between the problem of determining whether a given inequality is valid for the closure of the class from that of determining whether a given inequality is a *member* of the class. In some cases, the former is difficult, while the latter is not. Another point of subtle distinction that can affect the complexity of verifying validity is the form in which the inequality is given. If the inequality is given along with the certificate (consisting of the weights or the combinatorial structure that produced it), it may be easier to verify than if it is presented in the standard form without any additional structural information.

The fact that verifying validity is easy for many of the most commonly known classes of valid inequalities is actually no coincidence. In some sense, this happens by design. There is a strong connection between the existence of such a short certificate and the complexity of the separation problem that can be uncovered by considering the bilevel structure of the separation problem. We explore this next.

2.3 Bilevel Formulation

For now, let us adopt the point of view that verifying validity is equivalent to verifying (7) and consider a given class of valid inequalities $\mathcal{C}$. The bilevel nature of the MVVIP for a class $\mathcal{C}$ then arises as follows. The constraints of the upper level problem describe the structure of the class, usually in the form of explicit constraints on the allowable set of coefficient vectors α (though variables other than α may be required to define the structure). The lower-level problem is to generate the proof of validity, often by calculating the right-hand side β required to ensure (α, β) is valid. The complexity of the separation problem depends strongly on the complexity of this lower-level problem, which we refer to as the right-hand side generation problem (RHSGP), when it can be interpreted as such.

We can now formulate the separation problem, in principle, as the MIBLP

$$\min \quad \alpha^\top \hat{x} - \beta \tag{8}$$

$$\alpha \in \mathcal{C}_\alpha \tag{9}$$

$$\beta = \min_{x \in \mathcal{P}_\mathcal{C}} \alpha^\top x, \tag{10}$$

where $\mathcal{P}_\mathcal{C}$ is the closure and $\mathcal{C}_\alpha$ is the set of admissible coefficient vectors (the projection of $\mathcal{C}$ into the space of coefficient vectors). Note that we expressly include the constraints (9), though their function is only to ensure that the inequality produced is a member of the given class. They are technically unnecessary if the goal is simply to produce an inequality valid for the closure (but not necessarily a member of the class).

It should be emphasized that the formulation (8)–(10) is conceptual in nature. In practice, it presents several challenges. Most importantly, to actually write an instance of this problem down explicitly seems to require that we have compact (linear) descriptions of both the closure $\mathcal{P}_\mathcal{C}$ and the set of admissible coefficient vectors C_α. At the same time, the objective function (10) is also apparently bilinear, which raises other practical issues. Note that only the optimal value of the lower-level problem, not the solution itself, is required for computation of the upper-level objective value. This is similar to the case of a bilevel program that models direct conflict between the leader and the follower in a Stackelberg game.

The observations above highlight the crucial points in understanding the nature of the separation problem. If we do have compact descriptions of $\mathcal{P}_\mathcal{C}$ and C_α (either explicitly or as the convex hull of integer points inside an explicitly described polyhedron), then we can reformulate the separation problem as

$$\min \quad \alpha^\top \hat{x} - \beta \tag{11}$$

$$\alpha \in \mathcal{C}_\alpha \tag{12}$$

$$\beta \leq \alpha^\top x \quad \forall x \in \mathcal{P}_\mathcal{C}. \tag{13}$$

Solution of this reformulation technically only requires an oracle for the optimization problem over the closure $\mathcal{P}_\mathcal{C}$. However, there is a bit of circular logic involved in this statement, since the optimization problem over the closure *is* exactly the separation problem we are trying to solve. As above, the inequalities (12) are technically unnecessary if our goal is only to generate an inequality valid for the closure of the class. In practice, the inequalities (13) can be replaced by the finite set corresponding to just the extreme points of $\mathcal{P}_\mathcal{C}$ when this set is a polytope and this separation problem can be solved using a cutting plane method in which the extreme points of $\mathcal{P}_\mathcal{C}$ are generated dynamically.

In general, it should be clear that the above separation problem may be very difficult to solve. In fact, the complexity depends strongly on the complexity of the RHSGP (when the lower level can be interpreted in this way). A case in which there exists a simple reformulation of the separation problem as a single-level optimization problem is that in which the RHSGP can be solved in closed form or as a feasibility problem, given the vector α (and possibly some auxiliary information, such as the combinatorial structure that certifies validity). When generating a valid inequality from a fixed disjunction, we can solve the RHSGP in closed form, which leads to the well-known single-level reformulation for this class. Not surprisingly, when one examines the (exact) separation algorithms appearing in the literature, it quickly becomes clear that the classes are carefully chosen so as to ensure that the bilevel program (8)–(10) collapses into a polynomial-size single-level program, generally linear or mixed integer linear. However, this is not always possible, as we show below.

3 Examples

In this section, we illustrate our ideas by reviewing a number of examples. The first example (Section 3.1) allows a reformulation as a single-level mathematical program of polynomial size, whereas the second example (Section 3.2) does not allow this unless the polynomial hierarchy collapses. Two further examples are discussed in Section 3.3.

3.1 Disjunctive Valid Inequalities for general MILPs

Given an MILP in the form (1), Balas [1979] showed how to derive a valid inequality by exploiting any disjunction described by linear inequalities. In particular, the procedure applies to disjunctions of the form

$$\pi^\top x \leq \pi_0 \quad \text{OR} \quad \pi^\top x \geq \pi_0 + 1 \; \forall x \in \mathbb{R}^n, \tag{14}$$

where $\pi \in \mathbb{Z}^{|I|} \times \mathbf{0}^{|C|}$ and $\pi_0 \in \mathbb{Z}$, which are always valid for (1). The family of inequalities valid for the union of the two polyhedra, denoted by $\mathcal{P}_1$ and $\mathcal{P}_2$, obtained from $\mathcal{P}$ by adding inequalities $(-\pi, -\pi_0)$ and $(\pi, \pi_0 + 1)$, respectively, are called *split cuts*. Here, we describe the separation problem for split cuts in light of the bilevel formulation given in the previous section.

For a given disjunction of the form (14), the separation problem for the associated family of disjunctive inequalities with respect to a given vector $\hat{x} \in \mathcal{P}$ can be written as the bilevel LP

$$
\begin{aligned}
\min \quad & \alpha^\top \hat{x} - \beta && (15)\\
& \alpha_j \geq u^\top A_j - u_0 \pi_j \quad j \in I \cup C && (16)\\
& \alpha_j \geq v^\top A_j + v_0 \pi_j \quad j \in I \cup C && (17)\\
& u, v, u_0, v_0 \geq 0 && (18)\\
& u_0 + v_0 = 1 && (19)\\
& \beta = \min_{x \in \mathcal{P}_1 \cup \mathcal{P}_2} \alpha^\top x. && (20)
\end{aligned}
$$

Constraints (16) and (17) together with the non-negativity requirements on the dual multipliers (18) ensure that the inequality (α, β) is valid for each of the polyhedra obtained by adding a term of the disjunction (14) to the original formulation. This means that (α, β) satisfies the requirements for being a valid disjunctive inequality corresponding to the disjunction (14) (see Section 2.2 for more details). Constraint (19) is one of the possible normalizations to make the mathematical program above bounded. Once the coefficient vector and the corresponding dual multipliers are known, the RHSGP is easy to solve. To obtain a valid inequality, one has only to set β to $\min\{u^\top b - u_0\pi_0, v^\top b + v_0(\pi_0 + 1)\}$, which is the smallest of the right-hand sides obtained by the sets of multipliers (u, u_0) and (v, v_0) corresponding to the constraints of $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively.

It is easy to convert the bilevel LP above into a single-level linear program.

Proposition 1 *The bilevel LP* (15)–(20) *can be reformulated as the following single-level LP of polynomial size:*

$$
\begin{aligned}
\min \quad & \alpha^\top \hat{x} - \beta && (21)\\
& \alpha_j \geq u^\top A_j - u_0 \pi_j \quad j \in I \cup C \\
& \alpha_j \geq v^\top A_j + v_0 \pi_j \quad j \in I \cup C \\
& \beta \leq u^\top b - u_0 \pi_0 && (22)\\
& \beta \leq v^\top b + v_0(\pi_0 + 1) && (23)\\
& u_0 + v_0 = 1 \\
& u, v, u_0, v_0 \geq 0.
\end{aligned}
$$

Proof. Note that for given values of the remaining variables, any value of β satisfying the two inequalities (22) and (23) above yields a valid disjunctive constraint. Furthermore, these two inequalities ensure that $\beta \leq \min\{u^\top b - u_0\pi_0, v^\top b + v_0(\pi_0 + 1)\}$, while the objective function (21) ensures that the largest possible value of β is indeed selected, i.e., $\beta = \min\{u^\top b - u_0\pi_0, v^\top b + v_0(\pi_0 + 1)\}$. In other words, the objective function (21) yields the best value of the right-hand side for free, thus returning a maximally violated valid inequality. ■

If we wish to *select* the disjunction *and* generate a corresponding valid inequality, then we observe that the problem rises one level higher up in the polynomial hierarchy. For example, the *split closure* is the closure with respect to all inequalities arising from general disjunctions of the form (14). When solving the MVVIP for this closure, the vectors π and π_0 become (integer) variables and the problem is ostensibly a mixed integer non-linear optimization problem. It was shown by Cook et al. [1990] that the split closure is a polyhedron. The complexity of the MVVIP was shown to be *NP*-hard and the non-membership problem shown to be *NP*-complete by Caprara and Letchford [2003]. On the other hand, Balas and Saxena [2006] and Dash et al. [2007] both derive approximate methods of optimizing over the split closure that require solving a sequence of MILPs, indicating that the membership problem may also be difficult (though the complexity is still not known)

Disjunctive inequalities represent an example in which the bilevel nature of the separation problem is only useful to express the problem. In the next section, we will discuss instead a case in which no reformulation of polynomial size exists (modulo certain conjectures from complexity theory).

3.2 GSECs for the CVRP

Here, we consider the classical *Capacitated Vehicle Routing Problem*, as introduced by Dantzig and Ramser [1959], in which a quantity d_i of a single commodity is to be delivered to each customer $i \in N = \{1, \dots, n\}$ from a central depot $\{0\}$ using a homogeneous fleet of k vehicles, each with capacity K. The objective is to minimize total cost, with $c_{ij} \geq 0$ denoting the fixed cost of transportation from location i to location j, for $0 \leq i, j \leq n$. The costs are assumed to be *symmetric*, i.e., $c_{ij} = c_{ji}$ and $c_{ii} = 0$.

This problem is naturally associated with the complete undirected graph consisting of nodes $N \cup \{0\}$, edge set $E = N \cup \{0\} \times N \cup \{0\}$, and edge costs c_{ij} for $\{i, j\} \in E$. In this graph, a solution is the union of k cycles whose nodes sets share only the depot node in common and whose union includes all customers. By associating an integer variable with each edge in the graph, we obtain the following integer programming formulation:

$$\min \sum_{e \in E} c_e x_e$$

$$\sum_{e=\{0,j\} \in E} x_e = 2k \tag{24}$$

$$\sum_{e=\{i,j\} \in E} x_e = 2 \quad \forall i \in N \tag{25}$$

$$\sum_{\substack{e=\{i,j\} \in E \\ i \in S, j \notin S}} x_e \geq 2b(S) \quad \forall S \subset N,\ |S| > 1 \tag{26}$$

$$0 \leq x_e \leq 1 \quad \forall e = \{i,j\} \in E,\ i,j \neq 0 \tag{27}$$

$$0 \leq x_e \leq 2 \quad \forall e = \{0,j\} \in E \tag{28}$$

$$x_e \in \mathbb{Z} \quad \forall e \in E. \tag{29}$$

Constraints (24) and (25) are the *degree constraints*. In constraints (26), referred to as the *capacity constraints*, $b(S)$ is any of several lower bounds on the number of vehicles required to service the customers in set S. These constraints can be viewed as a generalization of the subtour elimination constraints from the *Traveling Salesman Problem* and serve both to enforce the connectivity of the solution and to ensure that no route has total demand exceeding the capacity K. The easily calculated lower bound $\sum_{i \in S} d_i / K$ on the number of trucks is enough to ensure the formulation (24)–(29)

is correct, but increasing this bound through the solution of a more sophisticated RHSGP will yield a stronger version of the constraints.

The MVVIP for capacity constraints with a generic lower bound $b(S)$ can be formulated as a MIBLP of the form (8)–(10) as follows. Because we are looking for a set $\bar{S} \subset N$ for which an inequality (26) is maximally violated, we define the binary variables

$$y_i = \begin{cases} 1 & \text{if customer } i \text{ belong to } \bar{S} \\ 0 & \text{otherwise} \end{cases} \qquad i \in N, \tag{30}$$

and

$$z_e = \begin{cases} 1 & \text{if edge } e \text{ belong to } \delta(\bar{S}) \\ 0 & \text{otherwise} \end{cases} \qquad e \in E, \tag{31}$$

where $\delta(\bar{S})$ denotes the set of edges in E with one endpoint in $\bar{S}$, to model selection of the members of the set $\bar{S}$ and the coefficients of the corresponding inequality. Thus, the formulation is

$$\min \sum_{e \in E} \hat{x}_e z_e - 2b(\bar{S}) \tag{32}$$

$$z_e \geq y_i - y_j \qquad \forall e = \{i,j\} \tag{33}$$

$$z_e \geq y_j - y_i \qquad \forall e = \{i,j\} \tag{34}$$

$$\max b(\bar{S}) \tag{35}$$

$$b(\bar{S}) \text{ is a valid lower bound.} \tag{36}$$

For improved tractability, the RHSGP (35)–(36) can be replaced by the calculation of a specific bound. One of the strongest possible lower bounds is obtained by solving to optimality the (strongly *NP*-hard) *Bin Packing Problem* (BPP) with the customer demands in set $\bar{S}$ being packed into the minimum number of bins of size K (Cornuéjols and Harche [1993] describe a further strengthening of the right-hand side, but we do not consider this bound here). The RHSGP based on the BPP can be modeled by using the binary variables

$$w_i^\ell = \begin{cases} 1 & \text{if customer } i \text{ is served by vehicle } \ell \\ 0 & \text{otherwise} \end{cases} \qquad i \in N, \ell = 1, \dots, k, \tag{37}$$

and

$$h_\ell = \begin{cases} 1 & \text{if vehicle } \ell \text{ is used} \\ 0 & \text{otherwise} \end{cases} \qquad \ell = 1, \dots, k. \tag{38}$$

Then, the full separation problem reads as follows:

$$\min \sum_{e \in E} \hat{x}_e z_e - 2b \tag{39}$$

$$z_e \geq y_i - y_j \qquad \forall e = \{i,j\} \tag{40}$$

$$z_e \geq y_j - y_i \qquad \forall e = \{i,j\} \tag{41}$$

$$b = \min \sum_{\ell=1}^{n} h_\ell \tag{42}$$

$$\sum_{\ell=1}^{n} w_i^\ell = y_i \qquad \forall i \in N \tag{43}$$

$$\sum_{i \in N} d_i w_i^\ell \leq K h_\ell \qquad \ell = 1, \dots, n, \tag{44}$$

where of course all variables y, z, w and h are binary according to definitions (30), (31), (37), and (38), respectively. We refer to this class of inequalities as *strengthened GSECs*.

Intuitively, it is clear that the MIBLP (39)–(44) cannot be straightforwardly reduced to a single-level program because the sense of the optimization of the RHSGP is opposed to that of the MVVIP, i.e., absence of the lower-level objective would result in a BPP solution using the *largest* number of bins instead of the *smallest* number. More formally, certifying validity requires certifying a lower bound on the optimal value to the BPP, which is a universally quantified decision problem that, when embedded inside an existentially quantified optimization problem, yields a problem that is hard for the second level of the polynomial hierarchy. This intuition is supported by the following result.

Theorem 1 *The MVVIP for the strengthened GSECs is* Σ_2^P*-hard.*

We delay presentation of the (somewhat lengthy and very technical) proof of this theorem until the following section in order to continue with the development of the consequences of this theorem, which are quite serious.

Corollary 1 *There is no polynomial-size single-level MILP reformulation of problem* (39)–(44) *unless the polynomial hierarchy collapses to its first level.*

Proof. Suppose that the MIBLP can be reformulated as a single-level MILP of polynomial size. As single-level MILPs of polynomial size can only express problems in *NP*, the MIBLP would also have to be in *NP*. However, as we show in the proof of Theorem 1, solution of the MIBLP (39)–(44) is a Σ_2^P-hard problem, which would then imply that $\Sigma_2^P = NP$. Hence the polynomial hierarchy would collapse to its first level. ■

A rather counterintuitive aspect of this result is that the separation problem for the closure of this class is a problem one level higher in the polynomial hierarchy than the separation problem for the convex hull of solutions to the CVRP. In fact, the VVP for this class is as difficult as the VVP for the CVRP itself. This gives a strong theoretical basis for the idea that this class probably should not be used in practice.

To make the problem a bit more tractable, we can simplify the RHSGP by relaxing the integrality requirement on w and h to obtain

$$b = \min \sum_{\ell=1}^{n} h_\ell \tag{45}$$

$$\sum_{\ell=1}^{n} w_i^\ell = y_i \qquad \forall i \in N \tag{46}$$

$$\sum_{i \in N} d_i w_i^\ell \leq K h_\ell \qquad \ell = 1, \ldots, n \tag{47}$$

$$w_i^\ell \in [0,1], \;\; h_\ell \in [0,1] \qquad i \in N, \ell = 1, \ldots, n, \tag{48}$$

which is also a lower bound for the BPP. In this case, the RHSGP can be solved in closed form, with an optimal solution being

$$b = \frac{\sum_{i \in N} d_i y_i}{K}. \tag{49}$$

Hence, the MVVIP reduces to a single-level MILP that can in turn be solved in polynomial time by transforming it into a network flow problem as proven by McCormick et al. [2003].

An intermediate valid lower bound is obtained by rounding the bound (49), i.e., using $b = \left\lceil \frac{\sum_{i\in N} d_i y_i}{K} \right\rceil$. Although such rounding can be done after the fact, relaxing the integrality on w, but not h, i.e., replacing conditions (48) by

$$w_i^\ell \in [0,1],\ h_\ell \in \{0,1\} \quad i \in N, \ell = 1, \dots, n,$$

results in reduction of the MVVIP to the single-level MILP

$$\begin{aligned}
\min \sum_{e\in E} \hat{x}_e z_e - 2b & \\
z_e \geq y_i - y_j & \qquad \forall e = \{i,j\} \\
z_e \geq y_j - y_i & \qquad \forall e = \{i,j\} \\
\frac{\sum_{i\in N} d_i y_i}{K} + 1 - \varepsilon \geq b & \\
b \in \mathbb{Z} & \\
y_i \in \{0,1\},\ z_e \in \{0,1\} & \qquad \forall i \in N, \forall e \in E,
\end{aligned}$$

which was shown by Cornuéjols and Harche [1993] to be *NP*-hard.

3.3 Further Examples

In this section we briefly present two more examples of separation problems that immediately call for a bilevel interpretation.

The first example is on the *positive side*, i.e., it is a case in which it is rather easy to reformulate the separation problem as a single-level MIP. Precisely, we are again considering the Capacitated Vehicle Routing Problem, but this time we consider the exponential-size formulation

$$\min \quad \sum_{C\in\mathcal{C}} \gamma_C x_C \tag{50}$$

$$\sum_{C\in\mathcal{C}} x_C = k \tag{51}$$

$$\sum_{C\in\mathcal{C}_i} x_C = 1 \quad \forall i \in N \tag{52}$$

$$x_C \in \{0,1\} \quad \forall C \in \mathcal{C}. \tag{53}$$

We denote by $\mathcal{C}$ the collection of all *feasible* cycles for a vehicle, by $\mathcal{C}_i \subseteq \mathcal{C}$ the set of cycles visiting customer i, and by γ_C the cost of a cycle $C \in \mathcal{C}$, i.e., the sum of the costs of the arcs in the cycle. Then, for each of the cycles C, we introduce a binary variable x_C that takes value 1 if and only if the cycle belongs to the solution, and 0 otherwise. Consequently, constraint (51) simply states that any solution must be comprised of k cycles, while constraints (52) guarantee that each customer i is visited exactly once. (Note that the requirements of any cycle visiting the depot and on the capacity of the vehicle are implicitly included in the requirement of any C being feasible.)

As anticipated, this formulation has an exponential number of variables, one for each cycle $C \in \mathcal{C}$. Thus, the continuous relaxation of the problem is solved by *column generation* techniques (see, e.g., Desaulniers et al. [2005]). Roughly speaking, this boils down to starting with a subset of the columns, solving the restricted LP, and generating (and adding) additional columns on the fly when needed, solving the so-called *pricing* problem. Those columns correspond to constraints of the dual of (50)–(52), namely

$$z + \sum_{i \in N_C} y_i \leq \gamma_C \quad \forall C \in \mathcal{C}, \tag{54}$$

where z is the dual variable associated with constraint (51), y_i is that associated with the i^{th} constraint (52), and $N_C \subseteq N$ denotes the set of customers visited by cycle C. Of course, the set of constraints (54) is exponential in size as well, so solving the pricing problem amounts to separation of constraints (54) in the dual. This latter problem has a natural bilevel formulation. Given a fractional (dual) solution $(\hat{z}, \hat{y})$, one is looking for a feasible cycle $\bar{C} \in \mathcal{C}$ *maximizing*

$$\hat{z} + \sum_{i \in N_{\bar{C}}} \hat{y}_i - \gamma_{\bar{C}}. \tag{55}$$

In other words, we are seeking the most violated constraint (54), if any exists. After associating the variables required to model nodes and arcs in the associated graph, quantity (55) becomes the objective function of the upper level model, while $\gamma_{\bar{C}}$ represents the lower level objective function, i.e., the right-hand side of (54). Clearly, given a set of customers T there are many ways of visiting them, making this problem apparently difficult to solve. Fortunately, the strongest inequality (54) associated with any given set T is the one with the smallest value for γ_T, i.e., among all possible cycles visiting the customers in T and the depot, one is looking for the shortest one. In turn, this choice also maximizes the upper level objective function (55), where γ appears with negative sign, thus the two objective functions "agree" and a single-level formulation can be developed straightforwardly.

The second example is instead on the *negative* side. Specifically, following our separation framework, Mattia [2010] has modeled the separation of a special class of valid inequalities for the *Network Loading Problem* as a bilevel programming problem. Given an undirected network, the problem consists in installing integer capacities on the edges of the network so as to be able to simultaneously route a set of point-to-point traffic demands at a minimum cost. The case of the network loading is similar to the one of the capacitated vehicle routing problem described in Section 3.2. The family of constraints considered by Mattia [2010] is the class of the co-called *metric* inequalities of which several variants have been proposed. In particular, computing the right-hand side of those inequalities amounts to solving an *NP*-hard optimization problem, so weaker versions of the inequalities have been used corresponding to the continuous relaxation of such a problem or its rounded version. The separation of the strongest version, called *tight* metric inequalities, has been addressed by bilevel programming in Mattia [2010] and solved in practice through a special purpose (heuristic) algorithm, with promising computational results. Although there is no formal proof of the fact that the separation of tight metric inequalities cannot be reformulated as a polynomial-size single-level MIP, Mattia [2010] considers that very unlikely, especially in light of the results given here and the similarity to the GSEC case.

4 Proof of Theorem 1

This entire section is centered around the following auxiliary subset selection problem. As earlier, we will denote $N = \{1, \ldots, n\}$.

Problem: GOOD-SUBSET

Instance: The complete, undirected graph on the vertex set $N \cup \{0\}$. Every vertex $i \in N$ has a non-negative integer size $\kappa(i)$, and every edge $ij \in N \times N$ has a non-negative integer weight $\lambda(ij)$. A bin size κ^* and a weight bound λ^*.

Question: Does there exist a subset $N' \subseteq N$ such that (i) the weight of the edge cut induced by N' is at most λ^*, and such that (ii) the vertices with sizes $\kappa(i)$ for $i \in N'$ cannot be packed into two bins of size κ^*?

Our objective is to prove that problem GOOD-SUBSET is Σ_2^P-hard. The reduction will be done from an appropriate quantified satisfiability variant, which is easily seen to be Σ_2^P-complete by combining the results of Stockmeyer [1977] with a reduction of Schaefer [1978].

Problem: 2-QUANTIFIED 1-IN-3-SAT

Instance: Two sets $X = \{x_1, \ldots, x_t\}$ and $Y = \{y_1, \ldots, y_t\}$ of Boolean variables. A set $C = \{c_1, \ldots, c_t\}$ of (disjunctive) clauses over $X \cup Y$ where every clause consists of exactly three distinct literals. A truth setting of $X \cup Y$ satisfies a clause c, if exactly one of the three literals in c is true.

Question: Does there exist $x_1, \ldots, x_t$ such that $\forall y_1, \ldots, y_t$, we have $\neg\phi(X, Y)$ is true?

Here, ϕ is a Boolean function that is true if and only if all of the clauses in C evaluate to true. In other words, the goal in 2-QUANTIFIED 1-IN-3-SAT is to fix the truth values for X in such a way that every possible truth setting for Y violates at least one of the clauses in C. A clause is violated if exactly 0 or 2 or 3 of its literals are set to true.

Hence let us consider an arbitrary instance of 2-QUANTIFIED 1-IN-3-SAT, and let us construct a corresponding instance of GOOD-SUBSET from it. For every literal $\ell = x_i$ or $\ell = \neg x_i$, we define an integer value

$$f(\ell) \;=\; \sum \{10^j : \text{ literal } \ell \text{ occurs in clause } c_j\}. \tag{56}$$

Note that all literals satisfy $f(\ell) < 10^{t+1}$. Now let us specify the $6t+2$ vertices in the GOOD-SUBSET instance together with their sizes κ.

- For every literal $\ell \in \{x_i, \neg x_i\}$ there are two corresponding vertices $A_1(\ell)$ and $A_2(\ell)$. The size of vertex $A_1(\ell)$ is $10^{3t+i} + f(\ell)$ and the size of vertex $A_2(\ell)$ is $10^{2t+i} + f(\ell)$. Note that the total size of $A_1(\ell)$ and $A_2(\neg\ell)$ equals the total size of $A_1(\neg\ell)$ and $A_2(\ell)$.
- For every literal $\ell \in \{y_i, \neg y_i\}$ there is a corresponding vertex $B(\ell)$ of size $10^{t+i} + f(\ell)$.
- There is a dummy vertex D_1 of size $10^{5t} + \sum_{i=1}^t 10^{2t+i} + \sum_{i=1}^t 10^i$, and there is another dummy vertex D_2 of size $10^{5t} + \sum_{i=1}^t 10^{3t+i}$.

Most of the edge weights λ are zero, and the only non-zero weights are defined as follows. For every literal $\ell \in \{x_i, \neg x_i\}$, the weight of the edge from vertex 0 to vertex $A_1(\ell)$ is 2^i, and the weight of the edge between vertices $A_1(\ell)$ and $A_2(\neg\ell)$ is $+\infty$ (or some forbiddingly large number). Finally the bin size is defined as

$$\kappa^* \ = \ 10^{5t} + \sum_{i=t+1}^{4t} 10^i + 2\sum_{i=1}^{t} 10^i, \tag{57}$$

and the weight bound is defined as

$$\lambda^* \ = \ \sum_{i=1}^{t} 2^i. \tag{58}$$

We will show that the newly constructed instance of Good-Subset has answer YES, if and only if the underlying instance of 2-Quantified 1-in-3-Sat has answer YES.

Lemma 1 *If the constructed instance of* Good-Subset *has answer YES, then also the underlying* 2-Quantified 1-in-3-Sat *instance has answer YES.*

Proof. Consider a vertex set N' that certifies the answer YES for the instance of Good-Subset. In other words, the weight of the corresponding edge cut is at most λ^* and N' cannot be packed into two bins of size κ^*. We prove several statements on the structure of N'.

(a) Without loss of generality N' contains both dummy vertices D_1 and D_2 and all the vertices $B(\ell)$.

Statement (a) is straightforward, since these vertices do not increase the weight of the cut.

(b) For every variable x_i exactly one of $A_1(x_i)$ and $A_1(\neg x_i)$ is in N'.

Statement (b) is proved by induction on i, starting with $i = t$ and going down to $i = 1$. In the inductive step, it is easy to see that N' cannot simultaneously contain both vertices $A_1(x_i)$ and $A_1(\neg x_i)$, since this would bring the weight of the cut (to vertex 0) above the weight bound λ^*. Furthermore, the size of vertex $A_1(x_i)$ alone is bigger than the sizes of all the vertices $A_1(x_j)$ with $j < i$ and all the vertices $A_2(\ell)$ together; an analogous statement holds for vertex $A_1(\neg x_i)$. If N' contains neither of these of these two bulky vertices, then N' could easily be packed into two bins of size κ^*.

(c) For every variable x_i, set N' contains either both vertices $A_1(x_i)$ and $A_2(\neg x_i)$, or both vertices $A_1(\neg x_i)$ and $A_2(x_i)$.

This follows by using statement (b), and by avoiding to have edges of infinite weight in the cut. Statements (a), (b), (c) fully describe the structure of set N'. Based on statement (c) we introduce the following truth setting for X: Whenever N' contains $A_1(x_i)$ and $A_2(\neg x_i)$, we set variable x_i to true and otherwise we set variable x_i to false. We claim that this is the desired truth setting of X that certifies that the underlying 2-Quantified 1-in-3-Sat instance has answer YES.

Suppose for the sake of contradiction that there exists a truth setting of Y such that every clause in $\phi(X, Y)$ contains exactly one true literal. We translate this truth setting into a packing of N': All vertices $A_1(\ell)$ in N' go into the first bin together with D_1, and all vertices $A_2(\ell)$ in N' go into the second bin together with D_2. Whenever a literal $\ell \in \{y_i, \neg y_i\}$ is true under the considered truth setting of Y, vertex $B(\ell)$ is packed into the first bin and vertex $B(\neg\ell)$ is packed into the second bin. Then the $f(\ell)$-parts of the vertex sizes in the first bin sum up to $\sum_{i=1}^{t} 10^i$ (since every clause contains exactly one true literal), and the $f(\ell)$-parts of the vertex sizes in the second bin sum up to $2\sum_{i=1}^{t} 10^i$ (since every clause contains exactly two false literals). Hence we have found a packing of N' into two bins of size κ^*, which yields the desired contradiction. We conclude that the 2-Quantified 1-in-3-Sat instance indeed has answer YES. ■

Lemma 2 *If the* 2-Quantified 1-in-3-Sat *instance has answer YES, then also the constructed instance of* Good-Subset *has answer YES.*

Proof. We start from the truth setting of the variables in X that makes the 2-Quantified 1-in-3-Sat instance true (as guaranteed in the statement of the lemma). Whenever some literal $\ell \in \{x_i, \neg x_i\}$ is set to true, we select the two vertices $A_1(\ell)$ and $A_2(\neg\ell)$ into the set N'. Furthermore we select all vertices $B(\ell)$ with $\ell \in \{y_i, \neg y_i\}$, and both dummy vertices D_1 and D_2 into the set N'. This completes the description of set N'. The weight of the corresponding cut for N' exactly equals the weight bound λ^*, and the total size of all vertices in N' exactly equals $2\kappa^*$.

We claim that the vertices in N' cannot be packed into two bins of size κ^*. Suppose for the sake of contradiction that such a packing exists. Without loss of generality we assume that the (huge) dummy vertex D_2 is packed into the second bin.

(a) the (large) dummy vertex D_2 is packed into the second bin.

Then the dummy vertex D_1 and the vertices $A_1(\ell)$ in N' do not fit anymore into this second bin, and hence they all must go into the first bin.

(b) Vertex D_1 and all vertices $A_1(\ell)$ in N' are packed into the first bin.

The first bin has now been filled up so much that there is not enough room for any of the vertices $A_2(\ell)$ in N'.

(c) All vertices $A_2(\ell)$ in N' are packed into the second bin.

Next we work through the remaining vertices $B(y_i)$ and $B(\neg y_i)$ in decreasing order of index. An easy inductive argument shows that not both of them can be packed together into the same bin; hence one of them goes into the first bin and the other one goes into the second bin.

(d) One of the vertices $B(y_i)$ and $B(\neg y_i)$ is packed into the first bin and the other one into the second bin.

From this we derive a truth setting for the variables in Y: If vertex $B(y_i)$ is in the first bin, then we set variable y_i to true and otherwise we set variable y_i to false.

Let us ignore for the moment the low-order digits in the $f(\ell)$-parts of the vertex sizes. Then the total size of the vertices assigned to the first bin is $\kappa^* - \sum_{i=1}^{t} 10^i$ and the total size of the vertices assigned to the second bin is $\kappa^* - 2\sum_{i=1}^{t} 10^i$. Therefore the $f(\ell)$-parts of the vertices in the first bin must add up to the sum $\sum_{i=1}^{t} 10^i$. Since there are no carry-overs in the addition, (56) yields that every clause contains exactly one true literal. That's the desired contradiction, and hence the vertices in N' cannot be packed into two bins of size κ^*. We conclude that the Good-Subset instance indeed has answer YES. ■

Lemma 1 and Lemma 2 together yield the following theorem.

Theorem 2 *Problem* Good-Subset *is hard for the complexity class* Σ_2^P*.*

Now let us take the final steps, and let us prove Theorem 1. This is done by embedding problem Good-Subset into the MIBLP (39)–(44). Hence we consider the instance of Good-Subset as constructed above, and we interpret it as the following instance of CVRP:

- vertex 0 is the depot in CVRP;
- the vertices in $N = \{1, \ldots, n\}$ form the customers in CVRP;
- the demand d_i of vertex i in CVRP coincides with the size $\kappa(i)$;
- the cost of edge ij in CVRP is defined as $c_{ij} = 10^{-t}\lambda(ij)$.

For a subset $N' \subseteq N$ of the vertices/customers, let $c(N')$ denote the weight of the edge cut induced by N', and let $b(N')$ denote the smallest number of bins of size κ^* into which N' can be packed.

Lemma 3 *A subset* $N' \subseteq N$ *forms a YES-certificate for the instance of* Good-Subset*, if and only if*

$$c(N') - 2\,b(N') \;\leq\; 10^{-t}\lambda^* - 6. \tag{59}$$

Proof. First assume that the subset N' is a YES-certificate for Good-Subset. Then $c(N') \leq 10^{-t}\lambda^*$ holds as the edge costs c_{ij} are the edge weights $\lambda(ij)$ multiplied by the scaling factor 10^{-t}, and $b(N') \geq 3$ holds as N' cannot be packed into two bins. Hence N' satisfies (59).

Next assume that N' satisfies (59). We observe that all vertices in the instance (and hence all vertices in N') can easily be packed into three bins of size κ^* (for instance: put dummy vertex D_1 into the first bin, dummy vertex D_2 into the second bin, and all the remaining vertices into the third bin). This implies $b(N') \leq 3$. Since $c(N') \geq 0$ and since $10^{-t}\lambda^* < 1$, inequality (59) yields

$$2\,(b(N') - 3) \;\geq\; c(N') - 10^{-t}\,\lambda(N') \;>\; -1.$$

This implies $b(N') \geq 3$. We conclude $b(N') = 3$, and N' indeed cannot be packed into two bins. Finally, with $b(N') = 3$ we get $c(N') \leq 10^{-t}\lambda^*$ from (59). Hence $\lambda(N') \leq \lambda^*$ indeed holds, and N' has all the desired properties. ■

The minimization in the MIBLP (39)–(44) searches for the smallest possible objective value $c(N') - 2\,b(N')$. If we know the optimal objective value, then we also know whether there exists a set N' that satisfies (59), and by Lemma 3 this amounts to deciding the Σ_2^P-hard problem Good-Subset. Consequently, the MIBLP (39)–(44) itself is Σ_2^P-hard. Since the MVVIP can easily be reduced to the MIBLP, the proof of Theorem 1 is complete.

5 Conclusions

We have presented a conceptual framework for the formulation of general separation problems as bilevel programs. This framework reflects the inherent bilevel nature of the separation problem arising from the fact that calculation of a valid right-hand side for a given coefficient vector is itself an optimization problem. In cases where this optimization problem is difficult in a complexity sense, it is generally not possible to formulate the separation problem as a traditional mathematical program. We conjecture that the finding the maximally violated valid inequality for most classes of valid inequalities can be thought of as having this hierarchical structure, but that certain of them can nonetheless be reformulated effectively. This is either because the right-hand side generation problem is easy to solve or because it goes “in the right direction” with respect to finding the most violated valid inequality itself. We believe that the paradigm presented here can be useful for the analysis of other intractable classes of valid inequalities, a first example of that being the paper by Mattia [2010]. In a future study, we plan to further formalize the conceptual framework presented here with a further investigation of the complexity issues, additional examples of these phenomena, and an assessment whether these ideas may be useful from a computational perspective.

Acknowledgements

We warmly thank Leo Liberti and Sara Mattia for useful comments on the paper. The first author is also grateful to the students in the Master Class of “Ricerca Operativa M”, University of Bologna, academic year 2009-2010, teaching to whom he found inspiration for the Vehicle Routing example. Finally, we thank the anonymous referees and associate editor, who also provided insights that improved the presentation of the paper.

References

D. Applegate, R. Bixby, V. Chvátal, and W. Cook. *The Traveling Salesman Problem: A Computational Study.* Princeton University Press, Princeton, New Jersey, 2006.

E. Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–51, 1979.

E. Balas and A. Saxena. Optimizing over the split closure. *Mathematical Programming*, 113: 219–240, 2006.

A. Caprara and A. Letchford. On the separation of split cuts and related inequalities. *Mathematical Programming*, 94:279–294, 2003.

B. Colson, P. Marcotte, and G. Savard. Bilevel programming: A survey. *4OR: A Quarterly Journal of Operations Research*, 3:87–107, 2005.

W. Cook, R. Kannan, and A. Schrijver. Chvátal closures for mixed integer programs. *Mathematical Programming*, 47:155–174, 1990.

G. Cornuéjols and F. Harche. Polyhedral Study of the Capacitated Vehicle Routing Problem. *Mathematical Programming*, 60:21–52, 1993.

G. B. Dantzig and R. H. Ramser. The Truck Dispatching Problem. *Management Science*, 6:80–91, 1959.

S. Dash, O. Günlük, and A. Lodi. On the MIR closure of polyhedra. In M. Fischetti and D. P. Williamson, editors, *Integer Programming and Combinatorial Optimization - IPCO 2007*, volume 4513 of *Lecture Notes in Computer Science*, pages 337–351. Springer-Verlag, 2007.

S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52:333–359, 2003.

G. Desaulniers, J. Desrosiers, and M. Solomon, editors. *Column Generation*. Springer, New York, 2005.

M. Fischetti, A. Lodi, and A. Tramontani. On the separation of disjunctive cuts. *Mathematical Programming*, 128:205–230, 2011.

M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981.

R. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical Programming*, 32:146–164, 1985.

S. Mattia. Separating tight metric inequalities by bilevel programming. *Operations Research Letters*, 40:568–572, 2010.

S. T. McCormick, M. R. Rao, and G. Rinaldi. Easy and difficult objective functions for max cut. *Mathematical Programming*, 94:459–466, 2003.

J. Moore and J. Bard. The mixed integer linear bilevel programming problem. *Operations Research*, 38:911–921, 1990.

T. Schaefer. The complexity of satisfiability problems. In *Proceedings of the 10th Annual ACM Symposium on Theory of Computing - STOC 1978*, pages 216–226, 1978.

L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3:1–22, 1977.