Semi-Continuous Cuts for Mixed-Integer Programming

I. R. de Farias JR.
State University of New York at Buffalo
defarias@buffalo.edu

December 7, 2003

Abstract

We study the convex hull of the feasible set of the *semi-continuous knapsack problem*, in which the variables belong to the union of two intervals. Besides being important in its own right, the semi-continuous knapsack problem is a relaxation of general mixed-integer programming. We show how strong inequalities valid for the *semi-continuous knapsack polyhedron* can be derived and used in a branch-and-cut scheme for mixed-integer programming and problems with semi-continuous variables. We present computational results that demonstrate the effectiveness of these inequalities, which we call collectively *semi-continuous cuts*. Our computational experience also shows that dealing with semi-continuous constraints directly in the branch-and-cut algorithm through a specialized branching scheme and semi-continuous cuts is considerably more practical than the "textbook" approach of modeling semi-continuous constraints through the introduction of auxiliary binary variables in the model.

Keywords: mixed-integer programming, semi-continuous variables, disjunctive programming, polyhedral combinatorics, branch-and-cut

1 Introduction

Let n be a positive integer, $N = \{1, \ldots, n\}$, N^+, N^+_∞, and N^- three disjoint subsets of N with $N^+ \cup N^+_\infty \cup N^- = N$, and $a_j > 0\ \forall j \in N^+ \cup N^+_\infty$ and $a_j < 0\ \forall j \in N^-$. We study the inequality description of $P = \text{clconv}(S)$, the closure of the convex hull of S, where $S = \{x \in \Re^n : x \text{ satisfies } (1), (2), \text{ and } (3)\}$, and

$$\sum_{j \in N} a_j x_j \leq b, \tag{1}$$

$$x_j \in [0, p_j] \cup [l_j, u_j]\ \forall j \in N^+, \tag{2}$$

and

$$x_j \in [0, p_j] \cup [l_j, \infty)\ \forall j \in N^+_\infty \cup N^-. \tag{3}$$

When $p_j < l_j$, we call x_j a *semi-continuous variable*, and the corresponding constraint (2) or (3) a *semi-continuous constraint*. Note that our definition of semi-continuous variable is more general than the usual one (Beale [3]), in which $p_j = 0$. When $p_j \geq l_j$, we say that x_j is a continuous variable. The set P is the *semi-continuous knapsack polyhedron*. An optimization problem whose constraints are (1), (2), and (3) is a *semi-continuous knapsack problem* (SCKP).

Semi-continuous constraints appear in general mixed-integer programming (MIP), since $x_j \in Z \cap [0, u_j] \Rightarrow x_j \in [0, p_j] \cup [p_j + 1, u_j]$, where p_j is an integer. They also appear more explicitly in other applications, including production scheduling (Beale [3], Biegler et al. [5]), portfolio optimization (Bienstock [6], Perold [28]), and blending problems (Williams [31]).

The "textbook" approach to dealing with the semi-continuous constraints (2) is to introduce a binary variable y_j in the model for each semi-continuous variable x_j, $j \in N^+$, and the constraints

$$x_j \geq l_j y_j \tag{4}$$

and

$$x_j \leq p_j + (u_j - p_j) y_j, \tag{5}$$

see for example Nemhauser and Wolsey [27]. This approach has the potential disadvantage of considerably increasing:

- the size of the problem, since the inclusion of the auxiliary binary variables doubles the number of variables, and the inclusion of (4) and (5) adds $2n$ constraints to the model;
- the amount of branching, since it may happen that in an optimal solution x to the LP relaxation, x satisfies (2) but $y_j \in (0, 1)$ for some $j \in N^+$, in which case branching on y_j may be performed to enforce integrality;

- degeneracy of the LP relaxation, due to (4) and (5);

see de Farias et al. [10]. Note that to use the same device to model (3) it is necessary to assign *big-M* values to u_j in (5). Among other issues, the introduction of big-M constants may cause severe numerical difficulties. For example, in a basic solution to the LP relaxation, many times some of the constraints (5) are satisfied at equality. If $x_j \in (p_j, l_j)$ but it is much smaller than the big-M constant assigned to it, y_j will be very small, and the solver may consider it to be 0. As a result, the solver may consider the LP relaxation solution feasible (i.e. such that $y_j \in \{0, 1\}$ $\forall j \in N$) and fathom the node. The common end result is that the solver gives as optimal solution one that does not satisfy all semi-continuous constraints.

An alternative to the "textbook" approach that dispenses with the introduction of the auxiliary binary variables y_j and the constraints (4) and (5) in the model was suggested by Beale [3]. It consists in branching directly on the semi-continuous variables x_j by adding the bound $x_j \leq p_j$ in one branch and $x_j \geq l_j$ in the other branch, thus enforcing the semi-continuous constraints (2) and (3) algorithmically, directly in the branch-and-bound scheme. Note that, this way, all the issues explained in the previous paragraph are immediately resolved.

We explore Beale's approach further by deriving strong cuts in the x-space valid for P and using them in a *branch-and-cut scheme without auxiliary binary variables* (de Farias et al. [10]) for solving problems that contain semi-continuous constraints. Branch-and-cut without auxiliary binary variables has been successfully used in the study of other combinatorial constraints, including complementarity (de Farias et al. [11], Ibaraki [20], Ibaraki et al. [21], Jeroslow [22]), cardinality (Bienstock [6], de Farias and Nemhauser [12, 13], Laundy [23]), and special ordered sets (Beale and Tomlin [4], de Farias et al. [9]).

The main tool used in this paper to study the inequality representation of semi-continuous knapsack polyhedra in the x-space is lifting, see Louveaux and Wolsey [24] for the basic theory of lifting and recent developments. As part of our lifting approach, we generalize the concepts of cover and cover inequality (Balas [2], Hammer et al. [18], Wolsey [32]), which have proven to be of great use in 0-1 programming, for semi-continuous constraints. We also show how to lift our cover inequalities to derive strong cuts to be used in branch-and-cut. We call our cuts collectively *semi-continuous cuts*.

Even though our theoretical results are general, we test their practical value specifically on problems with integer constraints, semi-continuous constraints, and the constraints

$$x_j \in [0, p_j] \cup \{s_j\} \cup [l_j, u_j], j \in N, \tag{6}$$

which we call *three-term semi-continuous*. They arise, for example, in financial applications, see Perold [28]. We show through our computational results that semi-continuous cuts can be effective for general MIP, and problems with semi-continuous and three-term semi-continuous constraints. Our computational results also demonstrate that branch-and-cut without auxiliary binary variables is superior to the "textbook" MIP approach for problems with semi-continuous and three-term semi-continuous constraints.

Compared to the 0-1 case, polyhedral studies for general MIP are scarce in the literature. For a review on the use of polyhedral methods for MIP, see the recent studies by Louveaux and Wolsey [24], Marchand et al. [26], and Wolsey [33]; see also Aráoz et al. [1] and Gomory et al. [16] for recent developments on the use of the group approach for deriving cutting planes for general MIP. Louveaux and Wolsey [24] reviews polyhedral results for constraints (4) and (5). We are not aware of any polyhedral study for three-term semi-continuous constraints.

Throughout the paper we will denote $(a)^+ = \max\{0, a\}$, where $a \in \Re$. We will adopt the convention that $\sum_{j\in\emptyset} a_j = 0$, $\max\{cx : x \in \emptyset\} = -\infty$, and $\min\{cx : x \in \emptyset\} = \infty$, where $\{a_j\}$ is a sequence of real numbers and $c \in \Re^n$. We will repeatedly refer to Proposition 1, where the following notation is used. First, $u_j = \infty\ \forall j \in N^+_\infty$. If

$$\sum_{j\in N^+\cup N^+_\infty} a_j u_j > b, \tag{7}$$

let $t \in N^+ \cup N^+_\infty$ be the smallest index such that $\sum_{\{j\in N^+\cup N^+_\infty : j\le t\}} a_j u_j > b$, and $\hat{x}$ be given by $\hat{x}_j = u_j\ \forall j \in \{k \in N^+ : k < t\}$,

$$\hat{x}_t = \frac{b - \sum_{\{j\in N^+ : j<t\}} a_j u_j}{a_t},$$

and $\hat{x}_j = 0$ otherwise. If $N^- \neq \emptyset$, let $r = \max\{j : j \in N^-\}$, and $\tilde{x}$ be given by $\tilde{x}_j = u_j$ $\forall j \in \{k \in N^+ : k < r\}$,

$$\tilde{x}_r = \left(\frac{b - \sum_{\{j\in N^+ : j<r\}} a_j u_j}{a_r}\right)^+,$$

and $\tilde{x}_j = 0$ otherwise. Finally, let $\bar{x}$ be given by $\bar{x}_j = u_j\ \forall j \in N^+$, and $\bar{x}_j = 0$ otherwise.

Proposition 1 *Consider the continuous knapsack problem*

$$max\left\{\sum_{j\in N} c_j x_j : \sum_{j\in N} a_j x_j \le b, x_j \le u_j\ \forall j \in N^+, \text{and } x_j \ge 0\ \forall j \in N\right\}, \tag{8}$$

where $c_j \ge 0\ \forall j \in N^+ \cup N^+_\infty$, $c_j \le 0\ \forall j \in N^-$, *and*

$$\frac{c_1}{a_1} \ge \cdots \ge \frac{c_n}{a_n}.$$

Then, (8) has an optimal solution iff it is feasible and $\forall j \in N^+_\infty, k \in N^-$,

$$\frac{c_j}{a_j} \le \frac{c_k}{a_k}.$$

Assume that (8) has an optimal solution x^*. *If*

$$\frac{c_j}{a_j} > \frac{c_k}{a_k}$$

for some $j \in N^+, k \in N - \{j\}$, *then* $x^*_j = u_j$ *whenever* $k \in N^-$ *or* $x^*_k > 0$. *Finally, suppose that:*

1. *$N^+ \cup N^+_\infty = \emptyset$. Then $\tilde{x}$ is an optimal solution to (8);*

2. *$N^- = \emptyset$. If (7) holds, then $\hat{x}$ is an optimal solution to (8), otherwise $\bar{x}$ is an optimal solution to (8);*

3. *both $N^+ \cup N^+_\infty \neq \emptyset$ and $N^- \neq \emptyset$. If (7) does not hold, then $\bar{x}$ is an optimal solution to (8). Assume that (7) holds. If $t < r$ and $t \in N^+$, then $\tilde{x}$ is an optimal solution to (8), otherwise $\hat{x}$ is an optimal solution to (8).* □

In Section 2 we introduce assumptions and we present a few simple results about P. We give the trivial facet-defining inequalities and a necessary and sufficient condition for them to describe P. We then discuss the nontrivial inequalities. In Section 3 we present the lifting technique and a few lifting results that hold for the nontrivial inequalities for P. We show that in some cases it is easy to obtain the exact lifting coefficients of several variables, and we show how the lifting coefficients of all variables can be calculated approximately in time $O(n^2)$. We also give the full inequality description of P when all variables are continuous with the exception of one semi-continuous variable. In Section 4 we extend the concepts of cover and cover inequality of 0-1 programming to our case. We show that when the cover is *simple*, the cover inequality is the only nontrivial facet-defining inequality of the semi-continuous knapsack polyhedron obtained by fixing at 0 all variables not indexed by the cover. We give the value of the lifting coefficient of a continuous variable x_k, $k \in N^-$, that is fixed at 0, when the cover inequality is lifted with respect to it first. We then show that for mixed-integer programming, our cover inequality is sometimes stronger than the Gomory mixed-integer cut and the cover inequality defined in [8]. In Section 5 we present results of our computational experience on the effectiveness of semi-continuous cuts for MIP, and problems with semi-continuous and three-term semi-continuous variables. In addition, we compare the effectiveness of branch-and-cut without auxiliary binary variables and of the traditional MIP approach for problems with semi-continuous and three-term semi-continuous variables. Finally, in Section 6 we present directions for further research.

2 The Semi-Continuous Knapsack Polyhedron

In this section we introduce assumptions and we present a few simple results about P. We establish the *trivial* facet-defining inequalities and when they suffice to describe P. We present a *nontrivial* facet-defining inequality that dominates (1) and under certain conditions is the only nontrivial facet-defining inequality for P. Finally, we present a few relations that hold among the coefficients of the variables in a nontrivial inequality.

We will assume throughout the paper that:

Assumption 1 $n \geq 2$;

Assumption 2 $p_j \geq 0$ and $l_j > 0$ $\forall j \in N$, and $u_j \geq l_j$ $\forall j \in N^+$;

Assumption 3 when $N^- = \emptyset$, $N^+_\infty = \emptyset$ and $a_j u_j \le b \ \forall j \in N^+$;

Assumption 4 when $N^+_\infty = \emptyset$, $\sum_{j \in N^+} a_j u_j > b$.

When Assumption 1 does not hold, the problem is trivial. So, there is no loss of generality in Assumption 1. In addition, with Assumption 1 it is possible to simplify the presentation of several results that would otherwise have to consider separately the case $n = 1$. If $u_j = 0$ for some $j \in N^+$, x_j can be eliminated from the problem. Therefore, there is no loss of generality in Assumption 2. Assumption 2 implies that $p_j > 0$ whenever x_j is continuous. When $N^- = \emptyset$, x_j is bounded $\forall j \in N$. In addition, for $j \in N^+$, it is possible, in this case, to scale x_j so that $a_j u_j \le b$, unless $b = 0$ or $x_j \in \{0\} \cup [l_j, u_j]$ with $a_j l_j > b$. In the first case the problem is trivial, and in the second case x_j can be eliminated from the problem. Thus, there is no loss of generality in Assumption 3. Assumption 3 implies that when $N^- = \emptyset$, $b > 0$. Finally, if $N^+_\infty = \emptyset$, (1) is redundant unless $\sum_{j \in N^+} a_j u_j > b$. This means that there is no loss of generality in Assumption 4 either. Assumption 4 implies that when $N^+ \cup N^+_\infty = \emptyset$, $b < 0$.

As a result of Assumptions 1–4, Propositions 2–4 follow.

Proposition 2 *P is full-dimensional.* □

Proposition 3 *The inequality*

$$x_j \ge 0 \tag{9}$$

is facet-defining $\forall j \in N^+ \cup N^+_\infty$. For $j \in N^-$, (9) is facet-defining iff either

1. *$|N^-| > 1$ or*
2. *$b > 0$ and $\forall k \in N^+ \cup N^+_\infty$ either $p_k > 0$ or $a_k l_k \le b$.*

□

Proposition 4 *When $N^- \ne \emptyset$,*

$$x_j \le u_j \tag{10}$$

is facet-defining $\forall j \in N^+$. When $N^- = \emptyset$, (10) is facet-defining iff $a_j u_j < b$ and $\forall k \in N^+ - \{j\}$ either $p_k > 0$ or $a_j u_j + a_k l_k \le b$. □

Example 1 Let $S = \{x \in \Re^3 : x$ satisfies (11), $x_1 \in [0, 1] \cup [2, 3]$, $x_2 \in \{0\} \cup [3, \infty)$, and $x_3 \ge 0\}$, where

$$2x_1 + 3x_2 \le 4 + x_3. \tag{11}$$

Then, $x_1 \ge 0$, $x_1 \le 3$, and $x_2 \ge 0$ are facet-defining for P. On the other hand, $x_3 \ge x_2$ $\forall x \in S$, and therefore $x_3 \ge 0$ is not facet-defining. □

Unlike inequalities (9) and (10), there do not seem to exist simple necessary and sufficient conditions to determine when (1) is facet-defining. We now present an inequality that, when $N = N^-$, is valid for P, is at least as strong as (1) (and possibly stronger), and under an additional condition gives, together with (9), a full inequality description for P.

Proposition 5 *Suppose that $N = N^-$, and let $N_0^- = \{j \in N^- : p_j = 0\}$. Then,*

$$\sum_{j \in N_0^-} \frac{a_j}{min\{b, a_j l_j\}} x_j + \frac{1}{b} \sum_{j \in N^- - N_0^-} a_j x_j \geq 1 \tag{12}$$

is valid for P. If $N^- = N_0^-$ or $\exists k \in N^- - N_0^-$ such that x_k is continuous, then (12) is facet-defining. If $N^- = N_0^-$, then $P = \{x \in \Re_+^n : x$ satisfies (12)$\}$.

Proof Let $\tilde{x} \in S$. If $\tilde{x}_j > 0$ for some $j \in N_0^-$ with $a_j l_j < b$, then, since $\tilde{x}_j \geq l_j$, $\tilde{x}$ satisfies (12). If $\tilde{x}_j = 0$ for all $j \in N_0^-$ with $a_j l_j < b$, then (1) $\Rightarrow \tilde{x}$ satisfies (12). Clearly, if $N^- = N_0^-$ or $\exists k \in N^- - N_0^-$ such that x_k is continuous, then (12) is facet-defining.

Suppose that $N = N_0^-$. We prove that $P = \{x \in \Re_+^n : x$ satisfies (12)$\}$ by showing that for an arbitrary nonzero vector $(c_1, \ldots, c_n)$ of objective function coefficients, one of the inequalities (9) or (12) is satisfied at equality for every optimal solution to SCKP. We assume WLOG that SCKP is a minimization problem.

If $c_j < 0$ for some $j \in N$, then SCKP is unbounded. So we assume that $c_j \geq 0 \ \forall j \in N$. Let $I = \{j \in N : c_j = 0\}$. If $I \neq \emptyset$, then for every optimal solution to SCKP, $x_j = 0$ $\forall j \in N - I$. So we assume that $c_j > 0 \ \forall j \in N$. Let x^* be an optimal solution to SCKP, $R = \{j \in N : x_j^* > 0\}$, and

$$s = \text{argmax} \left\{ \frac{c_j}{a_j} : j \in R \right\}.$$

Suppose that x^* does not satisfy (12) at equality. Since

$$\sum_{j \in R} \frac{a_j}{\min\{b, a_j l_j\}} x_j^* > 1 \Rightarrow \sum_{j \in R} a_j x_j^* < b,$$

then $x_j^* = l_j \ \forall j \in R$, and we have that

$$\sum_{j \in R} a_j l_j < b. \tag{13}$$

Note that $|R| \geq 2$. This is true because if $R = \{s\}$, then (13) $\Rightarrow \min\{a_s l_s, b\} = a_s l_s$, and (12) is satisfied at equality. Since $|R| \geq 2$, it is clear that $\forall j \in R$,

$$l_j < \frac{b}{a_j}. \tag{14}$$

Now, let z^* be the optimal value of SCKP. Then, (13) implies that

$$c_s \frac{b}{a_s} < c_s \frac{\sum_{j \in R} a_j l_j}{a_s} \leq \sum_{j \in R} c_j l_j = z^*.$$

However, due to (14), $\hat{x}$ given by

$$\hat{x}_j = \begin{cases} \frac{b}{a_s} & \text{if } j = s \\ 0 & \text{otherwise} \end{cases}$$

is a feasible solution to SCKP. Thus, x^* must satisfy (12) at equality. □

Example 2 Let $S = \{x \in \Re^5_+ : x$ satisfies (15), $x_1 \in \{0\} \cup [1, \infty)$, $x_2 \in \{0\} \cup [2, \infty)$, $x_3 \in [0, 1] \cup [2, \infty)$, and $x_5 \in [0, 1] \cup \{2\}\}$, where

$$2x_1 + 3x_2 + 3x_3 + x_4 \geq 4 + x_5. \tag{15}$$

Then, $P \cap \{x \in \Re^5 : x_3 = x_4 = x_5 = 0\} = \{x \in \Re^5_+ : x$ satisfies (16) and $x_3 = x_4 = x_5 = 0\}$, where

$$x_1 + x_2 \geq 2. \tag{16}$$

□

Let $PR = \{x \in \Re^n_+ : x$ satisfies (1) and (10) $\forall j \in N^+\}$, the feasible set of the LP relaxation of SCKP. The following proposition is easy to prove.

Proposition 6 *Let x be a vertex of PR. Then, with the possible exception of one, all components of x must satisfy*

$$x_j = 0, j \in N^- \cup N^+_\infty \tag{17}$$

and

$$x_j \in \{0, u_j\}, j \in N^+. \tag{18}$$

If one of conditions (17) or (18) is not satisfied by a component of x, then x must satisfy (1) at equality. □

We now give a necessary and sufficient condition for $P = PR$.

Proposition 7 *$P = PR$ iff $\forall T \subseteq N^+$, $i \in N^+ \cup N^+_\infty - T$, and $k \in N^-$, the following two conditions are satisfied*

1. $\sum_{j\in T} a_j u_j + a_i p_i \geq b$ *or* $\sum_{j\in T} a_j u_j + a_i l_i \leq b$

2. $\sum_{j\in T} a_j u_j + a_k p_k \leq b$ *or* $\sum_{j\in T} a_j u_j + a_k l_k \geq b$.

Proof The *if* part follows from Proposition 6. If $a_i p_i < b - \sum_{j\in T} a_j u_j < a_i l_i$ for some $T \subseteq N^+$ and $i \in N^+ \cup N^+_\infty - T$, then $\hat{x}$ given by

$$\hat{x}_j = \begin{cases} u_j & \text{if } j \in T \\ \frac{b - \sum_{v\in T} a_v u_v}{a_i} & \text{if } j = i \\ 0 & \text{otherwise} \end{cases}$$

is a vertex of *PR* that does not satisfy (2). If $a_k p_k > b - \sum_{j\in T} a_j u_j > a_k l_k$ for some $T \subseteq N^+$ and $k \in N^-$, then $\tilde{x}$ given by

$$\tilde{x}_j = \begin{cases} u_j & \text{if } j \in T \\ \frac{b - \sum_{v\in T} a_v u_v}{a_k} & \text{if } j = k \\ 0 & \text{otherwise} \end{cases}$$

is a vertex of *PR* that does not satisfy (3). □

Inequalities (1), (9), (10), and any inequality dominated by one of them is called *trivial*. For the remainder of the paper we will study the *nontrivial* inequalities for P. We will denote a generic nontrivial inequality valid for P as

$$\sum_{j\in N} \alpha_j x_j \leq \beta. \tag{19}$$

Note that $\alpha_j \leq 0\ \forall j \in N^-$, $\beta > 0$ whenever $N^- = \emptyset$, and $\beta < 0$ whenever $N^+ \cup N^+_\infty = \emptyset$. In addition, (19) remains valid if we replace α_j with $(\alpha_j)^+\ \forall j \in N^+ \cup N^+_\infty$, and we will then assume that $\alpha_j \geq 0\ \forall j \in N^+ \cup N^+_\infty$.

Proposition 8 *Let* $k \in N^-$. *If*

1. $i \in N^+_\infty$ *and (19) is satisfied at equality for at least one point* $\tilde{x}$ *of* S, *then*

$$\frac{\alpha_i}{a_i} \leq \frac{\alpha_k}{a_k}; \tag{20}$$

2. $i \in N^+$, x_k *is continuous, and (19) is satisfied at equality for at least one point* $\tilde{x}$ *of* S *with* $\tilde{x}_i < u_i$, *then*

$$\frac{\alpha_i}{a_i} \leq \frac{\alpha_k}{a_k}, \tag{21}$$

or in case $\tilde{x}_i > 0$, *then*

$$\frac{\alpha_i}{a_i} \geq \frac{\alpha_k}{a_k}; \tag{22}$$

3. *$i \in N^-$, x_k is continuous, and (19) is satisfied at equality for at least one point $\tilde{x}$ of S with $\tilde{x}_i > 0$, then*

$$\frac{\alpha_i}{a_i} \leq \frac{\alpha_k}{a_k}. \tag{23}$$

Proof We prove the proposition for statement 1, the proofs for statements 2 and 3 being similar. For ϵ big enough, $\tilde{x}'$ given by

$$\tilde{x}'_j = \begin{cases} \tilde{x}_i + \epsilon & \text{if } j = i \\ \tilde{x}_k - \epsilon \frac{a_i}{a_k} & \text{if } j = k \\ \tilde{x}_j & \text{otherwise} \end{cases}$$

also belongs to S. Therefore, it satisfies (19), and

$$\alpha_i \epsilon - \alpha_k \epsilon \frac{a_i}{a_k} \leq 0,$$

which implies (20). □

Proposition 9 *Suppose that (19) is facet-defining. If $i \in N^+_\infty$ and x_i is continuous, then $\alpha_i = 0$.*

Proof Because (19) is nontrivial and facet-defining, there exists a point $x^* \in P$ that satisfies (19) at equality and (1) strictly at inequality. Thus, for $\epsilon > 0$ sufficiently small, x' given by

$$x'_j = \begin{cases} x^*_i + \epsilon & \text{if } j = i \\ x^*_j & \text{otherwise} \end{cases}$$

belongs to P. This means that $\alpha_i \leq 0$, and therefore, $\alpha_i = 0$. □

From statement 3 of Proposition 8 it follows that if (19) is facet-defining, and x_i and x_k, $i, k \in N^-$, are continuous,

$$\frac{\alpha_i}{a_i} = \frac{\alpha_k}{a_k}. \tag{24}$$

Because of (24) and Proposition 9 we will continue our polyhedral study under the following two assumptions:

Assumption 5 There is at most one continuous variable x_i with $i \in N^-$;

Assumption 6 $p_j < l_j \ \forall j \in N^+_\infty$.

3 Lifting

In this section we present the lifting technique and a few lifting results that hold for the nontrivial inequalities for P. We show that in some cases it is easy to obtain the exact lifting coefficients of several variables. We then apply the lifting technique to obtain a nontrivial family of facets of P when all variables are continuous with the exception of one semi-continuous variable. We show that, in this particular case, this family of facets, together with the trivial facets, gives P. Finally, we show how all the lifting coefficients can be calculated approximately in time $O(n^2)$.

Let $\tilde{x} \in P$, $T^+ \subseteq N^+$, $T^+_\infty \subseteq N^+_\infty$, $T^- \subseteq N^-$, and $T = T^+ \cup T^+_\infty \cup T^-$. We will henceforth denote a generic nontrivial valid inequality for $P_T = P \cap \{x \in \Re^n : x_j = \tilde{x}_j \; \forall j \in N - T\}$ as

$$\sum_{j \in T} \alpha_j x_j \leq \beta. \tag{25}$$

We will also denote the resulting knapsack constraint (1) when $x_j = \tilde{x}_j \; \forall j \in N - T$ as

$$\sum_{j \in T} a_j x_j \leq b - \sum_{j \in N-T} a_j \tilde{x}_j. \tag{26}$$

In Lemma 1 we establish the lifting technique, see for example Louveaux and Wolsey [24].

Lemma 1 *Let $i \in N - T$,*

$$\alpha_i^{min} = max \left\{ \frac{\sum_{j \in T} \alpha_j x_j - \beta}{\tilde{x}_i - x_i} : x \in P_{T \cup \{i\}} \text{ and } x_i < \tilde{x}_i \right\} \tag{27}$$

and

$$\alpha_i^{max} = min \left\{ \frac{\beta - \sum_{j \in T} \alpha_j x_j}{x_i - \tilde{x}_i} : x \in P_{T \cup \{i\}} \text{ and } x_i > \tilde{x}_i \right\}. \tag{28}$$

Then

$$\sum_{j \in T} \alpha_j x_j + \alpha_i x_i \leq \beta + \alpha_i \tilde{x}_i \tag{29}$$

is a valid inequality for $P_{T \cup \{i\}}$ if and only if $\alpha_i \in [\alpha_i^{min}, \alpha_i^{max}]$. Moreover, if (25) defines a face of P_T of dimension t, and $\alpha_i = \alpha_i^{min}$ or $\alpha_i = \alpha_i^{max}$, then (29) defines a face of $P_{T \cup \{i\}}$ of dimension at least $t + 1$. □

It is well known that when $P_{T \cup \{i\}} \cap \{x : x_i < \tilde{x}_i\} \cap \{x : x_i > \tilde{x}_i\} \neq \emptyset$, it may happen that $[\alpha_i^{\min}, \alpha_i^{\max}] = \emptyset$, in which case it is not possible to find a lifting coefficient for x_i. Because of this, in most of the lifting theory presented in this paper, we will fix variables at their bounds. We leave as an open question what values variables can be fixed at before lifting.

For the particular case of 0-1 mixed-integer programming this issue was settled in Richard et al. [29, 30].

As we establish in Propositions 10 and 11, in some cases it is easy to obtain the exact lifting coefficients of several variables. In Proposition 10 we establish that when $\tilde{x}_i = 0$, $i \in (N^+ \cup N^+_\infty) - T$, and $p_i > 0$, the lifting coefficient of x_i is equal to 0. We omit the proof of Proposition 10, which is similar to the proof of Proposition 9.

Proposition 10 *Let $i \in (N^+ \cup N^+_\infty) - T$, $\tilde{x}_i = 0$, and suppose that (25) defines a facet of P_T. If $p_i > 0$, the lifting coefficient of x_i is 0.* □

Example 2 (Continued) The lifting coefficient of x_5 in (16) is 0 (regardless of the lifting order), and therefore $P \cap \{x \in \Re^5 : x_3 = x_4 = 0\} = \{x \in \Re^5_+ : x$ satisfies (16) and $x_3 = x_4 = 0\}$. □

We now establish that when (25) contains a continuous variable x_k with $k \in N^-$, it is easy to obtain the lifting coefficient of a variable x_i with $i \in N^-$ and $p_i > 0$ that is fixed at 0, or with $i \in N^+$ and $l_i < u_i$ that is fixed at u_i. The proof of Proposition 11 follows from Proposition 8.

Proposition 11 *Let x_k, $k \in T^-$, be a continuous variable, and suppose that (25) defines a facet of P_T. If*

1. *$i \in N^- - T$, $p_i > 0$, and $\tilde{x}_i = 0$; or*
2. *$i \in N^+ - T$, $l_i < u_i$, and $\tilde{x}_i = u_i$,*

the lifting coefficient of x_i is

$$\alpha_i = \frac{a_i}{a_k}\alpha_k.$$

□

Example 2 (Continued) The lifting coefficient of x_4 in (16) is the optimal value of

$$\max\left\{\frac{2 - x_1 - x_2}{x_4} : x \in S, x_4 > 0, \text{ and } x_3 = 0\right\}. \tag{30}$$

Let x^* be an optimal solution to (30). Because $x_2 > 0 \Rightarrow x_2 \geq 2$, $x^*_2 = 0$. In addition, it is clear that x^* must satisfy (15) at equality, i.e. $2x^*_1 + x^*_4 = 4$. Thus,

$$\max\left\{\frac{2 - x_1 - x_2}{x_4} : x \in S, x_4 > 0, \text{ and } x_3 = 0\right\} = \frac{2 - x^*_1}{4 - 2x^*_1} = \frac{1}{2}.$$

So,

$$2x_1 + 2x_2 + x_4 \geq 4 \tag{31}$$

defines a facet of $P \cap \{x \in \Re^5 : x_3 = 0\}$. From Proposition 11, if we now lift (31) with respect to x_3, the lifting coefficient is 3, and thus

$$2x_1 + 2x_2 + 3x_3 + x_4 \geq 4$$

defines a facet of P. □

We now apply the lifting technique to derive a family of nontrivial facet-defining inequalities for P when all variables are continuous, with the exception of one semi-continuous variable x_i with $i \in N^+$. Note that, due to Assumptions 5 and 6, in this case $|N^-| \leq 1$ and $N^+_\infty = \emptyset$.

Proposition 12 *Let $i \in N^+$ with $p_i < l_i$. Suppose that x_j is continuous $\forall j \in N - \{i\}$. Let $U \subseteq N^+ - \{i\}$ be such that*

$$a_i l_i + \sum_{j \in U} a_j u_j > b \tag{32}$$

and

$$a_i p_i + \sum_{j \in U} a_j u_j < b. \tag{33}$$

Then,

$$\Delta_U x_i + \sum_{j \in U} a_j x_j + \sum_{j \in N^-} a_j x_j \leq p_i \Delta_U + \sum_{j \in U} a_j u_j, \tag{34}$$

with

$$\Delta_U = \frac{\sum_{j \in U} a_j u_j - b + a_i l_i}{l_i - p_i},$$

is valid and facet-defining for P.

Proof Because of (32),

$$\sum_{j \in U} a_j x_j + \sum_{j \in N^-} a_j x_j \leq b - a_i l_i \tag{35}$$

is facet-defining for $P \cap \{x : x_i = l_i \text{ and } x_j = 0 \ \forall j \in N^+ - (U \cap \{i\})\}$. We derive (34) by lifting (35).

Let α_i be the lifting coefficient of x_i. Using the notation of Lemma 1,

$$\alpha_i^{\min} = \max\{\frac{\sum_{j \in U} a_j x_j + \sum_{j \in N^-} a_j x_j - b + a_i l_i}{l_i - x_i} : x \in S \text{ and } x_i \leq p_i\}$$

and

$$\alpha_i^{\max} = \min\{\frac{b - a_i l_i - \sum_{j\in U} a_j x_j - \sum_{j\in N^-} a_j x_j}{x_i - l_i} : x \in S \text{ and } x_i > l_i\}. \tag{36}$$

Because of (33),

$$\max\{\frac{\sum_{j\in U} a_j x_j + \sum_{j\in N^-} a_j x_j - b + a_i l_i}{l_i - x_i} : x \in S \text{ and } x_i \le p_i\} = \frac{\sum_{j\in U} a_j u_j - b + a_i l_i}{l_i - p_i} = \Delta_U.$$

Now, let x^* be an optimal solution to (36). Clearly

$$\sum_{j\in U} a_j x_j^* + \sum_{j\in N^-} a_j x_j^* + a_i x_i^* = b,$$

otherwise, by decreasing x_j, $j \in N^-$, or increasing x_i or $x_j, j \in U$, it is possible to improve over x^*. This means that

$$\alpha_i^{\max} = \frac{b - a_i l_i - \sum_{j\in U} a_j x_j^* - \sum_{j\in N^-} a_j x_j^*}{x_i^* - l_i} = \frac{b - a_i l_i - b + a_i x_i^*}{x_i^* - l_i} = a_i.$$

Since $\Delta_U < a_i$, Lemma 1 implies that it is possible to lift (35) with respect to x_i, and by choosing $\alpha_i = \Delta_U$, the lifted inequality defines a nontrivial facet of $P \cap \{x \in \Re^n : x_j = 0 \ \forall j \in N^+ - (U \cap \{i\})\}$. From Proposition 10, the lifting coefficient of x_j is 0 $\forall j \in N^+ - (U \cap \{i\})$. Thus, (34) defines a facet of P. □

As we show next, under the assumptions of Proposition 12, P is given by (34) and the trivial facet-defining inequalities. For general MIP, a result of this type has been given by Magnanti et al. [25].

Theorem 1 *Let $i \in N^+$ with $l_i < p_i$. Suppose that x_j is continuous $\forall j \in N - \{i\}$. Then $P = \{x \in \Re^n : x$ satisfies (1), (9), (10), and (34)$\}$.*

Proof As we did in Proposition 5, we prove the theorem by showing that for an arbitrary nonzero vector $(c_1, \ldots, c_n)$ of objective function coefficients, the set of all optimal solutions to SCKP is contained in the face defined by an inequality of one of the families (1), (9), (10), or (34). We assume that $N^- = \{k\}$, the proof for when $N^- = \emptyset$ being similar. We assume WLOG that SCKP is a maximization problem.

If $c_j < 0$ for some $j \in N^+$, then in all optimal solutions to SCKP $x_j = 0$, so we assume that $c_j \ge 0 \ \forall j \in N^+$. If $c_k > 0$, then SCKP is unbounded. So we assume that $c_k \le 0$.

If $c_j = 0 \ \forall j \in N^+$, then since $(c_1, \ldots, c_n)$ is a nonzero vector, $c_k < 0$, and in all optimal solutions to SCKP (1) is satisfied at equality in case $b < 0$, and $x_k = 0$ in case $b \ge 0$. So let $I = \{j \in N^+ : c_j > 0\}$ and assume that $I \ne \emptyset$.

If

$$\frac{c_j}{a_j} > \frac{c_k}{a_k}$$

for some $j \in I$, then in all optimal solutions to SCKP, $x_j = u_j$. So we assume that $\forall j \in I$

$$\frac{c_j}{a_j} \leq \frac{c_k}{a_k}.$$

If $b \leq 0$, then in all optimal solutions to SCKP, (1) is satisfied at equality. So we assume that $b > 0$.

If $\sum_{j \in I} a_j u_j \leq b$, then in all optimal solutions to SCKP, $x_r = u_r \ \forall r \in I$. So we assume that

$$\sum_{j \in I} a_j u_j > b. \tag{37}$$

If $i \notin I$, then (37) implies that in all optimal solutions to SCKP (1) is satisfied at equality. So we assume that $i \in I$. If $\sum_{j \in I - \{i\}} a_j u_j + a_i p_i \geq b$ or $\sum_{j \in I - \{i\}} a_j u_j + a_i l_i \leq b$, then again (37) implies that in all optimal solutions to SCKP (1) is satisfied at equality. So we assume that

$$\sum_{j \in I - \{i\}} a_j u_j + a_i p_i < b \tag{38}$$

and

$$\sum_{j \in I - \{i\}} a_j u_j + a_i l_i > b. \tag{39}$$

Let $I' = I - \{i\}$. Inequalities (38) and (39) imply that

$$\Delta_{I'} x_i + \sum_{j \in I'} a_j x_j + a_k x_k \leq p_i \Delta_{I'} + \sum_{j \in I'} a_j u_j \tag{40}$$

is a facet-defining inequality of the family (34).

Suppose now that SCKP has an optimal solution $\hat{x}$ that does not satisfy (1) at equality. Because of (38) and (39), $\hat{x}_i = p_i$, $\hat{x}_j = u_j \ \forall j \in I - \{i\}$, and $\hat{x}_k = 0$, in which case $\hat{x}$ satisfies (40) at equality. Let $\bar{x}$ be an alternative optimal solution that satisfies (1) at equality. Then,

$$\bar{x}_i \geq l_i.$$

We show that $\bar{x}$ satisfies (40) at equality. Note that

$$c_i \bar{x}_i + \sum_{j \in I'} c_j \bar{x}_j + c_k \bar{x}_k = c_i p_i + \sum_{j \in I'} c_j u_j. \tag{41}$$

Suppose that $\bar{x}_i = l_i$. It follows that $a_i l_i + \sum_{j \in I'} a_j \bar{x}_j + a_k \bar{x}_k = b$, or

$$\frac{a_i l_i + \sum_{j \in I'} a_j u_j - b}{l_i - p_i}(l_i - p_i) + \sum_{j \in I'} a_j \bar{x}_j + a_k \bar{x}_k = \sum_{j \in I'} a_j u_j,$$

and so

$$\Delta_{I'} l_i + \sum_{j \in I'} a_j \bar{x}_j + a_k \bar{x}_k = \Delta_{I'} p_i + \sum_{j \in I'} a_j u_j.$$

Finally, suppose $\bar{x}_i > l_i$. We have that

$$\bar{x}_i = \frac{b - \sum_{j \in I'} a_j \bar{x}_j - a_k \bar{x}_k}{a_i}.$$

Because of (41),

$$c_i \frac{b - \sum_{j \in I'} a_j \bar{x}_j - a_k \bar{x}_k}{a_i} + \sum_{j \in I'} c_j \bar{x}_j + c_k \bar{x}_k = c_i p_i + \sum_{j \in I'} c_j u_j,$$

or

$$b - \sum_{j \in I'} a_j \bar{x}_j - a_k \bar{x}_k = a_i p_i + \sum_{j \in I'} \frac{a_i}{c_i} c_j (u_j - \bar{x}_j) - \frac{a_i}{c_i} c_k \bar{x}_k. \tag{42}$$

For $j \in I'$, if

$$\frac{c_j}{a_j} > \frac{c_i}{a_i},$$

then $\bar{x}$ can be an optimal solution only if $\bar{x}_j = u_j$. This means that $\forall j \in I'$,

$$\frac{a_i}{c_i} c_j (u_j - \bar{x}_j) \leq a_j (u_j - \bar{x}_j).$$

In the same way, if

$$\frac{c_k}{a_k} > \frac{c_i}{a_i},$$

then $\bar{x}$ can be an optimal solution only if $\bar{x}_k = 0$. This means that

$$\frac{a_i}{c_i} c_k \bar{x}_k \geq a_k \bar{x}_k.$$

So,

$$a_i p_i + \sum_{j \in I'} \frac{a_i}{c_i} c_j (u_j - \bar{x}_j) - \frac{a_i}{c_i} c_k \bar{x}_k \leq a_i p_i + \sum_{j \in I'} a_j (u_j - \bar{x}_j) - a_k \bar{x}_k. \tag{43}$$

Combining (42) and (43), we obtain $b \leq a_i p_i + \sum_{j \in I'} a_j u_j$, which is inconsistent with (38). Thus, it must be that $\bar{x}_i = l_i$, and in all optimal solutions to SCKP (34) is satisfied at equality. □

Example 3 Let $S = \{x \in \Re^3_+ : x$ satisfies (44), $x_1 \in [0,1] \cup [2,3]$, and $x_2 \leq 2\}$, where

$$2x_1 + 3x_2 \leq 9 + x_3. \tag{44}$$

Then, $P = \{x \in \Re^3_+ : x$ satisfies (44), (45), $x_1 \leq 3$, and $x_2 \leq 2\}$, where

$$x_1 + 3x_2 \leq 8 + x_3. \tag{45}$$

□

In 0-1 programming, the objective function denominator in either (27) or (28) is always equal to 1, and the lifting problem is a linear 0-1 knapsack problem. In practice, it is common, in this case, to solve the lifting problem approximately by solving its LP relaxation and rounding the resulting optimal value down for (27) and up for (28); see Gu et al. [17], where an extensive computational study is presented that shows, among other things, that it is more practical to use the LP relaxation approximation to compute the lifting coefficients than to use dynamic programming to compute them exactly.

In the case of semi-continuous variables, however, the objective function denominator in (27) or (28) may not be a constant. We now show how to solve the LP relaxation of (27) and (28) for this case to obtain approximate values for the lifting coefficients of x_j, $j \in N - T$, in (25). As in 0-1 programming, the procedure gives all lifting coefficients approximately in time $O(n^2)$. We will discuss specifically the case where the next variable to be lifted is x_k, $k \in N^+$, $p_k = 0$, $p_k < l_k < u_k$, and $\tilde{x}_k = 0$. The other cases can be treated in a similar way.

Let $k \in N^+ - T$. Suppose that $p_k = 0$, $p_k < l_k < u_k$, $\tilde{x}_k = 0$, and that we lift (25) next with respect to x_k. The approximate lifting coefficient α_k of x_k is given by the optimal value of the LP relaxation of (28), i.e.

$$\alpha_k = \min \left\{ \frac{\beta - \sum_{j \in T} \alpha_j x_j}{x_k} : x \in \bar{S}_k \right\}, \tag{46}$$

where

$$\bar{S}_k = \{x \in \Re^n_+ : \begin{array}{l} \sum_{j \in T} a_j x_j + a_k x_k \leq b - \sum_{j \in N-T} a_j \tilde{x}_j, \\ x_j \leq u_j \ \forall j \in N^+, x_j = \tilde{x}_j \ \forall j \in N - (T \cup \{k\}), \text{and } x_k \geq l_k\}. \end{array}$$

We assume that $\bar{S}_k \neq \emptyset$, otherwise x_k cannot be the next lifted variable. Rather than solving (46), we solve

$$\max \left\{ \sum_{j \in T} \alpha_j x_j + \tilde{\alpha}_k x_k : x \in \bar{S}_k \right\} = \beta, \tag{47}$$

where the variables are $\tilde{\alpha}_k$ and x_j, $j \in N$. Problem (47) is to find a value $\tilde{\alpha}^*_k$ for the variable $\tilde{\alpha}_k$ such that $\forall x \in \bar{S}_k$

$$\sum_{j \in T} \alpha_j x_j + \tilde{\alpha}^*_k x_k \leq \beta,$$

and values x_j^* for the variables x_j, $j \in N$, such that $x^* \in \bar{S}_k$ and

$$\sum_{j \in T} \alpha_j x_j^* + \tilde{\alpha}_k^* x_k^* = \beta.$$

We now show that problems (46) and (47) are equivalent.

Proposition 13 *Let α_k be the optimal value of (46) and $(\tilde{\alpha}_k^*, x^*)$ be a solution to (47). Then, $\alpha_k = \tilde{\alpha}_k^*$.*

Proof It is clear that $\sum_{j \in T} \alpha_j x_j + \alpha_k x_k \le \beta \ \forall x \in \bar{S}_k$ and $\alpha_k \ge \tilde{\alpha}_k^*$. Now, since $(\tilde{\alpha}_k^*, x^*)$ is a solution to (47),

$$\sum_{j \in T} \alpha_j x_j^* + \alpha_k x_k^* \le \beta = \sum_{j \in T} \alpha_j x_j^* + \tilde{\alpha}_k^* x_k^*.$$

This means that $\alpha_k \le \tilde{\alpha}_k^*$, and therefore $\alpha_k = \tilde{\alpha}_k^*$. □

Suppose WLOG that $T = \{1, \ldots, t\}$ and

$$\frac{\alpha_1}{a_1} \ge \cdots \ge \frac{\alpha_t}{a_t}. \tag{48}$$

We now show how to solve (47) in linear time, once $\frac{\alpha_1}{a_1}, \cdots, \frac{\alpha_t}{a_t}$ have been sorted as in (48). We first note that due to Propositions 1 and 8, the maximization problem in (47) is never unbounded. If we know whether

$$\frac{\alpha_k}{a_k} \ge \frac{\alpha_1}{a_1}, \tag{49}$$

or

$$\frac{\alpha_j}{a_j} \ge \frac{\alpha_k}{a_k} \ge \frac{\alpha_{j+1}}{a_{j+1}} \tag{50}$$

for some $j \in \{1, \ldots, t-1\}$, or else

$$\frac{\alpha_t}{a_t} \ge \frac{\alpha_k}{a_k}, \tag{51}$$

then, using Proposition 1, it is possible to find an optimal solution x^* to the maximization problem in (47), even if we do not know the value of α_k. But then,

$$\alpha_k = \frac{\beta - \sum_{j \in T} \alpha_j x_j^*}{x_k^*}. \tag{52}$$

We determine α_k by calculating the values that result from (52) under the cases (49), (50), and (51). Specifically, let α_k^0 be the value of α_k resulting from case (49), α_k^j, $j \in \{1, \ldots, t-1\}$, the values of α_k resulting from case (50), and α_k^t the value of α_k resulting from case (51). Then,

$$\alpha_k = (\min\{\alpha_k^0, \dots, \alpha_k^t\})^+.$$

By calculating $\alpha_k^t, \dots, \alpha_k^0$ in this order, it is possible to calculate all of them in time $O(n)$. Also, by maintaining a list of the variable indices sorted as in (48) every time a new lifting coefficient is calculated, it will not be necessary to sort the indices, except for the initial inequality, before any variable is lifted. Therefore, all lifting coefficients can be calculated approximately in time $O(n^2)$.

Example 4 Let $S = \{x \in Z^3 : x$ satisfies (53), $x_1 \in [0,1] \cup \{2\}$, $x_2 \in [0,2] \cup [3,4]$, and $x_3 \in \{0\} \cup [1,3]\}$, where

$$4x_1 + 3x_2 + 3x_3 \leq 16. \tag{53}$$

As we will show in Section 4,

$$2x_1 + x_2 \leq 6 \tag{54}$$

defines a facet of conv(S) $\cap\{x \in \Re^3 : x_3 = 0\}$. We now lift x_3 approximately. The approximate lifting problem is

$$\max\{2x_1 + x_2 + \alpha_3 x_3 : 4x_1 + 3x_2 + 3x_3 \leq 16, x_1 \in [0,2], x_2 \in [0,4], x_3 \in [1,3]\} = 6. \tag{55}$$

If $\frac{\alpha_3}{3} \leq \frac{1}{3}$, then

$$x_1^{(2)} = 2, x_2^{(2)} = \frac{5}{3}, x_3^{(2)} = 1$$

is an optimal solution to the maximization problem in (55), and $\alpha_3^2 = \frac{1}{3}$. If $\frac{1}{3} \leq \frac{\alpha_3}{3} \leq \frac{1}{2}$, then

$$x_1^{(1)} = 2, x_2^{(1)} = 0, x_3^{(1)} = \frac{8}{3}$$

is an optimal solution to the maximization problem in (55), and $\alpha_3^1 = \frac{8}{3}$. If $\frac{1}{2} \leq \frac{\alpha_3}{3}$, then

$$x_1^{(0)} = \frac{7}{4}, x_2^{(1)} = 0, x_3^{(1)} = 3$$

is an optimal solution to the maximization problem in (55), and $\alpha_3^0 = \frac{5}{6}$. Thus, the approximate lifting coefficient is $\alpha_3 = \frac{1}{3}$, which coincides with the exact lifting coefficient. □

4 Cover Inequalities

In this section we extend the concepts of cover and cover inequality, commonly used in 0-1 programming, to our case. We show that when the cover is *simple*, the cover inequality is the only nontrivial facet-defining inequality of the semi-continuous knapsack polyhedron obtained by fixing at 0 all variables not indexed by the cover. We then give the value of the lifting coefficient of a continuous variable x_k, $k \in N^-$, that is fixed at 0, when the cover inequality is lifted with respect to it first. Finally, we show that for mixed-integer programming, our cover inequality is sometimes stronger than the Gomory mixed-integer cut and the cover inequality defined in [8].

We now define covers and simple covers for semi-continuous knapsack polyhedra.

Definition 1 *Let $C \subseteq N^+$ with $p_j < l_j \ \forall j \in C$. We say that C is a cover if*

$$\sum_{j \in C} a_j l_j > b. \tag{56}$$

If in addition

$$\sum_{j \in C - \{i\}} a_j u_j + a_i p_i \leq b \ \forall i \in C, \tag{57}$$

we say that the cover is simple. □

Note that our definitions of cover and simple cover coincide with the definitions of cover and minimal cover of 0-1 programming, where $p_j = 0$ and $l_j = u_j = 1 \ \forall j \in N$.

We now introduce cover inequalities and show that they are the only nontrivial facet-defining inequalities for $P^0 = P \cap \{x \in \Re^n : x_j = 0 \ \forall j \in N - C\}$ when the cover is simple.

Proposition 14 *Let C be a cover. Then,*

$$\sum_{j \in C} \frac{u_j - x_j}{u_j - p_j} \geq 1 \tag{58}$$

is valid for P. If C is a simple cover, then (58) is facet-defining for P^0, and $P^0 = \{x \in \Re^n : x$ satisfies (9), (10), and (58)$\}$.

Proof Let C be a cover. The points $x^{(i)}, i \in C$, given by

$$x_j^{(i)} = \begin{cases} p_i & \text{if } j = i \\ u_j & \text{if } j \in C - \{i\} \\ 0 & \text{otherwise} \end{cases}$$

satisfy (58) (some of these points may not belong to P.) Now, let $x \in S$. As C is a cover, $x_j \le p_j$ for some $j \in C$. Since $x_j \le u_j$ $\forall j \in N$, $\exists\, k \in C$ such that $x_j \le x_j^{(k)}$ $\forall j \in C$. Thus, x satisfies (58).

If C is a simple cover, then the points $x^{(j)} \in P^0$ and satisfy (58) at equality $\forall j \in C$. Because the set $\{x^{(j)} : j \in C\}$ is linearly independent, (58) defines a facet for P^0. Now, let $t \in C$ and

$$d = \sum_{j \in C - \{t\}} \frac{u_j}{u_j - p_j} + \frac{p_t}{u_t - p_t}$$

(note that the value of d does not depend on t.) Since the maximal vertices of the polytope

$$\{y \in \Re^{|C|} : \sum_{j=1}^{|C|} x_{k_j}^{(i)} y_j \le 1 \ \forall i \in C \text{ and } y \ge 0\}$$

are $(\frac{1}{u_{k_1}}, 0, \ldots, 0)$, $(0, \frac{1}{u_{k_2}}, \ldots, 0)$, ..., $(0, 0, \ldots, \frac{1}{u_{k_{|C|}}})$, and $\frac{1}{d}(\frac{1}{u_{k_1} - p_{k_1}}, \frac{1}{u_{k_2} - p_{k_2}}, \ldots, \frac{1}{u_{k_{|C|}} - p_{k_{|C|}}})$, it follows from antiblocking theory [14, 15, 27] that the only nontrivial facet-defining inequality for P^0 is

$$\sum_{j \in C} \frac{x_j}{u_j - p_j} \le d,$$

which is the same as (58). Thus, $P^0 = \{x \in \Re^n : x$ satisfies (9), (10), (58), and $x_j = 0$ $\forall j \in N - C\}$. □

When $p_j = 0$ $\forall j \in C$, i.e. for the "usual" semi-continuous variables, (58) becomes

$$\sum_{j \in C} \frac{x_j}{u_j} \le |C| - 1.$$

In particular, for the case of 0-1 variables, (58) becomes $\sum_{j \in C} x_j \le |C| - 1$.

Example 5 Let $N = \{1, \ldots, 5\}$, $x_1 \in \{0\} \cup [2, 3]$, $x_2 \in [0, 1] \cup [3, 4]$, $x_3 \in [0, 2] \cup [4, 5]$, $x_4 \in \{0\} \cup [1, 2]$, and $x_5 \in [0, 1] \cup \{2\}$. Consider the knapsack inequality

$$2x_1 + 3x_2 + 4x_3 + x_4 + x_5 \le 33.$$

Note that $2l_1 + 3l_2 + 4l_3 + l_4 + l_5 = 32 < 33$, and therefore N is not a cover. However, by fixing $x_1 = 3$, $x_2 = 4$, and $x_4 = x_5 = 0$, $\{3\}$ becomes a simple cover, and

$$x_3 \le 2 \tag{59}$$

is valid and facet-defining for $P \cap \{x \in \Re^5 : x_1 = 3, x_2 = 4, x_4 = x_5 = 0\}$. □

We now give the lifting coefficient of a continuous variable x_k, $k \in N^-$, that is fixed at 0, when the cover inequality is lifted with respect to it first.

Proposition 15 *Let C be a simple cover and x_k, $k \in N^-$, a continuous variable that is fixed at 0. Then,*

$$\sum_{j \in C} \frac{u_j - x_j}{u_j - p_j} - \frac{a_k}{\sum_{j \in C} a_j l_j - b} \left(1 - \sum_{j \in C} \frac{u_j - l_j}{u_j - p_j} \right) x_k \geq 1 \tag{60}$$

is facet-defining for $P_{C \cup \{k\}}$.

Proof Let

$$\sum_{j \in C} \frac{u_j - x_j}{u_j - p_j} + \alpha_k x_k \geq 1$$

be the facet-defining inequality that results from lifting (58) with respect to x_k. Then,

$$\alpha_k = \max \left\{ \frac{1}{x_k} \left(1 - \sum_{j \in C} \frac{u_j - x_j}{u_j - p_j} \right) : \sum_{j \in C} a_j x_j + a_k x_k \leq b, x_j \text{ satisfies (2) } \forall j \in C, \text{ and } x_k > 0 \right\}. \tag{61}$$

Note that $\alpha_k > 0$, since $\bar{x}_j = u_j \ \forall j \in C$ and

$$\bar{x}_k = -\frac{\sum_{j \in C} a_j u_j - b}{a_k}$$

is a feasible solution to (61) with objective function value

$$\bar{\alpha}_k = -\frac{a_k}{\sum_{j \in C} a_j u_j - b}.$$

On the other hand, when $x_j < l_j$ for some $j \in C$, the objective function value is at most 0. Therefore, $x_j \geq l_j \ \forall j \in C$ in an optimal solution to (61), and clearly $\sum_{j \in C} a_j x_j + a_k x_k = b$. Problem (61) is then equivalent to

$$\alpha_k = \max \left\{ \frac{-a_k}{\sum_{j \in C} a_j x_j - b} \left(1 - \sum_{j \in C} \frac{u_j - x_j}{u_j - p_j} \right) : x_j \in [l_j, u_j] \ \forall j \in C \right\}. \tag{62}$$

Consider now the solution to (62) $x_j^* = l_j \ \forall j \in C$ with objective function value

$$\alpha_k^* = -\frac{a_k}{\sum_{j \in C} a_j l_j - b} \left(1 - \sum_{j \in C} \frac{u_j - l_j}{u_j - p_j} \right).$$

We prove the proposition by showing that none of the vertices of $(\times [l_j, u_j])_{j \in C}$, has an objective function value greater than α_k^*.

Let i =argmin$\{a_j(u_j - p_j) : j \in C\}$. Let $T \subseteq C$ and suppose that $T \neq \emptyset$ and $i \notin T$. Consider the solution $\tilde{x}_j = l_j \ \forall j \in C - T$ and $\tilde{x}_j = u_j \ \forall j \in T$. The objective function value is

$$\tilde{\alpha} = \frac{-a_k}{\sum_{j \in C-T} a_j l_j + \sum_{j \in T} a_j u_t - b} \left(1 - \sum_{j \in C-T} \frac{u_j - l_j}{u_j - p_j}\right).$$

We now show that $\alpha_k^* \geq \tilde{\alpha}_k$.

Condition (57) implies that

$$b \geq \sum_{j \in C-T} a_j l_j + \sum_{j \in T} a_j u_j + \sum_{j \in C-T \cup \{i\}} a_j (u_j - l_j) - a_i (l_i - p_i). \tag{63}$$

Clearly,

$$\sum_{j \in C-T \cup \{i\}} a_j (u_j - l_j) - a_i (l_i - p_i) < 0$$

and

$$\sum_{j \in T} \frac{a_j (u_j - l_j)}{a_i (u_i - p_i)} \geq \sum_{j \in T} \frac{u_j - l_j}{u_j - p_j}.$$

Therefore, (63) implies that

$$b \geq \sum_{j \in C-T} a_j l_j + \sum_{j \in T} a_j u_j + \frac{1}{\sum_{j \in T} \frac{u_j - l_j}{u_j - p_j}} \sum_{j \in T} \frac{a_j (u_j - l_j)}{a_i (u_i - p_i)} \left(\sum_{j \in C-T \cup \{i\}} \frac{a_i (u_i - p_i)}{a_j (u_j - p_j)} a_j (u_j - l_j) - a_i (l_i - p_i) \right)$$

$$= \sum_{j \in C-T} a_j l_j + \sum_{j \in T} a_j u_j + \frac{1}{\sum_{j \in T} \frac{u_j - l_j}{u_j - p_j}} \sum_{j \in T} a_j (u_j - l_j) \left(\sum_{j \in C-T \cup \{i\}} \frac{u_j - l_j}{u_j - p_j} - \frac{l_i - p_i}{u_i - p_i} \right)$$

$$= \sum_{j \in C-T} a_j l_j + \sum_{j \in T} a_j u_j - \frac{1}{\sum_{j \in T} \frac{u_j - l_j}{u_j - p_j}} \sum_{j \in T} a_j (u_j - l_j) \left(1 - \sum_{j \in C-T} \frac{u_j - l_j}{u_j - p_j}\right).$$

Thus,

$$b \sum_{j \in T} \frac{u_j - l_j}{u_j - p_j} \geq \left(\sum_{j \in C-T} a_j l_j + \sum_{j \in T} a_j u_j \right) \sum_{j \in T} \frac{u_j - l_j}{u_j - p_j} - \sum_{j \in T} a_j (u_j - l_j) \left(1 - \sum_{j \in C-T} \frac{u_j - l_j}{u_j - p_j}\right),$$

or,

$$\sum_{j\in T} a_j u_j \left(1 - \sum_{j\in C} \frac{u_j - l_j}{u_j - p_j}\right) \geq \left(\sum_{j\in C-T} a_j l_j - b\right) \sum_{j\in T} \frac{u_j - l_j}{u_j - p_j} + \sum_{j\in T} a_j l_j \left(1 - \sum_{j\in C-T} \frac{u_j - l_j}{u_j - p_j}\right). \tag{64}$$

By adding

$$\left(\sum_{j\in C-T} a_j l_j - b\right) \left(1 - \sum_{j\in C} \frac{u_j - l_j}{u_j - p_j}\right)$$

to both sides of (64),we obtain

$$\left(\sum_{j\in C-T} a_j l_j + \sum_{j\in T} a_j u_j - b\right) \left(1 - \sum_{j\in C} \frac{u_j - l_j}{u_j - p_j}\right) \geq \left(\sum_{j\in C} a_j l_j - b\right) \left(1 - \sum_{j\in C-T} \frac{u_j - l_j}{u_j - p_j}\right),$$

or

$$\frac{1}{\sum_{j\in C} a_j l_j - b} \left(1 - \sum_{j\in C} \frac{u_j - l_j}{u_j - p_j}\right) \geq \frac{1}{\sum_{j\in C-T} a_j l_j + \sum_{j\in T} a_j u_j - b} \left(1 - \sum_{j\in C-T} \frac{u_j - l_j}{u_j - p_j}\right),$$

and so, $\alpha_k^* \geq \tilde{\alpha}_k$. The case $i \in T$ can be treated similarly. □

We now show in Example 6 that our cover inequality can be stronger than the Gomory mixed-integer cut and the cover inequality defined in [8]. The data for the example is taken from [8], Example 2.2.

Example 6 Let $S_1 = \{(x, s) \in Z_+^2 \times \Re_+ : (x, s)$ satisfies (65), $x_1 \leq 2$, and $x_2 \leq 3\}$ and $S_2 = \{(x, s) \in Z_+^2 \times \Re_+ : (x, s)$ satisfies (65), $x_1 \leq 2$, and $x_2 \leq 4\}$, where

$$4x_1 + 3x_2 \leq 16 + s. \tag{65}$$

The point

$$(x^*, s^*) = \left(2, \frac{8}{3}, 0\right)$$

is a fractional vertex of the LP relaxation for both sets. (It is the only fractional vertex in the case of S_2.)

Consider S_1. The Gomory mixed-integer cut for the basis of (x^*, s^*) is

$$2x_1 + x_2 \leq 6 + s. \tag{66}$$

Inequality (60) for $x_1 \in [0, 1] \cup \{2\}$ and $x_2 \in [0, 2] \cup \{3\}$ is

$$x_1 + x_2 \leq 4 + s, \tag{67}$$

which is facet-defining and stronger than (66), since

$$(2x_1 + x_2 \le 6 + s) = (x_1 + x_2 \le 4 + s) + (x_1 \le 2).$$

Note that in this case (67) coincides with the cover inequality of Ceria et al. [8].

Consider now the case of S_2. In [8] the following cover inequality is proposed

$$x_1 + x_2 \le 5 + \frac{1}{4}s. \tag{68}$$

Inequality (68) does not cut off (x^*, s^*). On the other hand, inequality (60) for $x_1 \in [0,1] \cup \{2\}$ and $x_2 \in [0,2] \cup [3,4]$ is

$$2x_1 + x_2 \le 6 + \frac{1}{2}s, \tag{69}$$

which is facet-defining and cuts off (x^*, s^*). Note that in this case (69) coincides with the Gomory mixed-integer cut for the basis of (x^*, s^*). □

5 Computational Experience

In this section we present our computational results on the effectiveness of semi-continuous cuts for problems with general integer, semi-continuous, and three-term semi-continuous variables. In addition, we compare the effectiveness of branch-and-cut without auxiliary binary variables and the traditional MIP approach for problems with semi-continuous and three-term semi-continuous variables. Our platform consisted of CPLEX 8.0 Callable Library on a Gateway desktop with 1.5 GHz Pentium 4 CPU, 2 Gb RAM, 120 Gb HD, and Windows XP. We kept all CPLEX parameters at their default values, except for preprocessing, which we turned off for all tests on the instances with semi-continuous and three-term semi-continuous variables. The reason that we turned off preprocessing for those tests is that no preprocessing strategy was developed for branch-and-cut without auxiliary binary variables. However, we kept preprocessing on for the tests on the instances with general integer variables. We limited the number of nodes to 1 million in all tests. In any instance there are exclusively integer, semi-continuous, or three-term semi-continuous variables.

The computational results are summarized in Tables 1-4. In the tables we report the number of nodes, computational time, number of CPLEX cuts, number of semi-continuous cuts, and gap improvement. In addition, in Table 1, we report the LP relaxation and the IP optimal values for each instance. The number of CPLEX cuts is the total number of cuts generated automatically by CPLEX throughout the branch-and-bound tree. The CPLEX cuts generated were mainly Gomory cuts, MIR, lifted flow cover, and lifted cover inequalities. On the other hand, the semi-continuous cuts were generated exclusively at the root node and kept in the cut pool for the remainder of the enumeration process. The gap improvement is given by

$$gapimp = 100 \times \frac{z(lp) - z(root)}{z(lp) - z(ip)},$$

where $z(lp)$, $z(root)$, and $z(ip)$ are the optimal values of the LP relaxation, the LP relaxation with the cuts generated at the root node added, and the IP, respectively.

The instances reported in Tables 2-4 were randomly generated. In Tables 2-4 m denotes the number of knapsack constraints and n denotes the number of integer or semi-continuous variables. We generated 3 instances for each pair $m \times n$. The results reported in Tables 2-4 are the rounded down average results for the instances with the same $m \times n$. The densities of the constraint matrices of these instances range between 15% and 50%. The objective function coefficients are integers uniformly chosen between 10 and 25. The nonzero knapsack coefficients are integers uniformly chosen between 5 and 20. The values of the right-hand-sides are $\lambda \times \max\{\lfloor .3 \sum_{j \in N} a_{ij} \rfloor, \max\{a_{ij} : j \in N\} + 1\}$, where $\lambda = 2$ for the IP instances, and $\lambda = 4$ for the semi-continuous and three-term semi-continuous instances. For the integer variables we chose $u_j = 2$. For the semi-continuous variables we chose $p_j = 0$, $l_j = 3$, and $u_j = 4$. Finally, for the three-term semi-continuous variables we chose $r_j = 1$, $s_j = 2$, $t_j = 3$, and $u_j = 4$.

In Tables 1 and 2 we give the results for the instances with general integer variables. The results in Table 1 refer to 12 instances of the MIPLIB [7]. Because we are interested in problems that contain general integer variables, we changed the domain of the variables in p0033, p0201, p0282, p0548, p2576, and set1ch, which is originally $\{0, 1\}$, to $\{0, 1, 2\}$. These instances are denoted with boldface fonts in Table 1. In Tables 1 and 2, the "sc" columns refer to the tests in which semi-continuous cuts were used (in addition to CPLEX cuts), while the "no sc" columns refer to the tests in which only the CPLEX cuts were used.

In Table 3 we give the results for the instances with semi-continuous variables, and in Table 4 for the instances with three-term semi-continuous variables. In Tables 3 and 4, the "mip" columns refer to the tests in which the traditional MIP approach was used, while the "sc" columns refer to the tests in which branch-and-cut without auxiliary binary variables was used. The "B&B" columns refer to pure branch-and-bound (i.e. no cuts added), while the "B&C" columns refer to branch-and-cut. Finally, the column "cut mip" gives the average number of CPLEX cuts generated for the MIP approach, while the column "cuts sc" gives the average number of semi-continuous cuts generated for branch-and-cut without auxiliary binary variables. In this second case, no integer variables are present in the models, and therefore CPLEX did not generate any cuts.

The branching scheme used for the tests with semi-continuous and three-term semi-continuous variables in branch-and-cut without auxiliary binary variables was the one suggested by Beale [3], and explained in Section 1. For the particular case of three-term semi-continuous variables we implemented Beale's branching scheme as follows. If $x_j \in (p_j, s_j)$ we add the bounds $x_j \leq p_j$ in one branch and $x_j \geq s_j$ in the other branch. If $x_j \in (s_j, l_j)$ we add the bounds $x_j \leq s_j$ in one branch and $x_j \geq l_j$ in the other branch.

In all cases, semi-continuous cuts were separated as follows. Let x^* be an optimal solution to the LP relaxation that is not feasible for the integer, semi-continuous, or three-term semi-continuous instance that is being solved. We fix any variable x_j for which $x_j^* \in \{0, u_j\}$ at the value of x_j^*, i.e. we take $\tilde{x}_j = x_j^*$. For integer variables, if $x_j^* \notin Z$, we take $p_j = \lfloor x_j^* \rfloor$ and $l_j = \lceil x_j^* \rceil$. If $x_j^* \in \{0, 1\}$, we take $p_j = 0$ and $l_j = 1$. If $x_j^* = 2$, we take $p_j = 1$ and $l_j = 2$. For three-term semi-continuous variables, if $x_j^* \leq 2$, we take $p_j = 1$ and $l_j = 2$, otherwise we take $p_j = 2$ and $l_j = 3$. To derive the semi-continuous cuts we consider only the knapsack constraints active at x^*. Let $C = \{j \in N : 0 < x_j^* < u_j\}$. If C is a cover, we delete from it as many elements in nondecreasing order of $a_j l_j$ as possible so that the resulting set is still a cover. Note that in the case of IP, C will always be a cover. If the corresponding cover inequality is violated, we lift it approximately, as described in Section 3, and we include it in the formulation, otherwise we discard it.

As shown in Tables 1 and 2, semi-continuous cuts can be of considerable use for integer programming problems. Note that CPLEX failed to complete the enumeration within 1 million nodes for rout and set1ch. The computational time and number of cuts reported in the "no sc" columns for these instances are the values when the enumeration was interrupted. In the particular case of (semi-continuous) lifted cover inequalities, the results show that they can be effective for the type of instances tested, i.e. sparse constraint matrices and variables with small upper bounds. In average, semi-continuous cuts reduced the number of nodes in 64% and the computational time in 56% for the MIPLIB instances. For the randomly generated IP instances, the reduction in the number of nodes was 81% and the computational time 68%.

As shown in Tables 3 and 4, semi-continuous cuts without auxiliary binary variables can be considerably more effective than the traditional MIP approach for problems with semi-continuous, and especially for problems with three-term semi-continuous variables. Note that CPLEX with the MIP approach failed to complete the enumeration within 1 million nodes for the instances with three-term semi-continuous variables for which $m = 70$ and $n = 7,000$. The computational time and number of cuts reported in the "mip" columns for these instances are the values when the enumeration was interrupted. In average, branch-and-cut without auxiliary binary variables reduced the number of nodes in 43% and the computational time in 30% over the MIP approach for the semi-continuous instances. For the three-term semi-continuous instances, the reduction in the number of nodes was 75% and the computational time 80%.

6 Further Research

We are currently studying semi-continuous cuts other than cover inequalities. In particular, we are evaluating the effectiveness of the simple cuts introduced in [10]. We are also studying the semi-continuous cuts that arise in structures other than the knapsack polyhedron, for example the Simplex tableaux and knapsack intersections. We are also testing the effectiveness of semi-continuous cuts for partial-integer and discrete variables.

Very often, semi-continuous constraints arise together with other combinatorial constraints in applications. Perhaps the most frequent one is fixed-charge, which is interesting

Table 1: Number of nodes, time, and number of cuts for IP: MIPLIB instances

problem	nodes no sc	nodes sc	time no sc	time sc	cuts cplex	cuts sc	gapimp no sc	gapimp sc	LP	IP
arki001	23,030	11,020	132.21	64	75	127	2	34	7,579,599.81	7,581,040
blend2	1,541	312	1.15	0.05	39	18	19	94	6.92	7.59
flugpl	68	334	0.03	0.02	5	2	47	47	1,167,185.73	1,201,500
gesa2	290	35	1.7	0.02	105	28	84	99	25,476,489.67	25,780,523.02
lseu	586	230	0.11	0.05	21	4	83	69	472.52	700
p0033	314	69	0.04	0.01	13	6	65	93	2,053.71	2,428
p0201	537	131	0.6	0.2	19	4	34	88	6,875	7,615
p0282	2,080	28	1.07	0.12	77	28	81	99	108,732.52	155,146
p0548	277	121	0.4	0.62	121	21	86	95	314.88	1,330
p2576	53,997	317	116.58	3.2	45	12	14	88	2,508	2,592
rout	10^6+	507,814	3,900	2,128	870	38	0.5	90	981.86	1077.56
set1ch	10^6+	228,350	1,533	320	424	38	79	99	32,007.72	54,788.25

Table 2: Number of nodes, time, and number of cuts for IP: random instances

$m \times n$	nodes no sc	nodes sc	time no sc	time sc	cuts cplex	cuts sc	gapimp no sc	gapimp sc
$20 \times 1,000$	3,024	154	5	2	14	18	54	78
$20 \times 1,500$	3,018	210	8	4	19	29	31	72
$30 \times 2,000$	7,980	312	17	7	19	31	39	82
$30 \times 2,500$	15,230	829	31	12	54	25	41	87
$40 \times 3,000$	32,620	9,730	855	238	98	58	68	72
$40 \times 3,500$	48,750	12,530	2,330	925	134	216	47	89
$50 \times 4,000$	75,520	18,990	4,330	1,022	118	207	52	65
$50 \times 4,500$	112,830	27,620	6,520	1,740	235	198	63	73
$60 \times 5,000$	248,330	31,540	9,278	3,550	334	201	59	87

in its own. We are currently studying a branch-and-cut approach without auxiliary binary variables for problems with fixed-charge constraints.

Acknowledgment This work was partially supported by NSF grants DMI-0100020 and DMI-0121495. It was developed while the author was a visiting faculty at CORE, Université Catholique de Louvain, during the academic year of 2001-2002, and at GSIA, Carnegie Mellon University, during the academic year of 2002-2003. Their support and hospitality is gratefully acknowledged. The intellectual environment at both places was exceptionally exciting and inspirational. This work benefited a great deal from it. Special thanks are due to Laurence Wolsey for several enlightening discussions. In particular, Proposition 15 was originally suggested by him. He also made suggestions that helped to improve the presentation.

Table 3: Number of nodes, time, and cuts for problems with semi-continuous constraints

$m \times n$	nodes			time			cuts		gapimp	
	mip	sc		mip	sc		mip	sc	mip	sc
		B&B	B&C		B&B	B&C				
$30 \times 3,000$	681	920	680	27	11	22	12	10	45	43
$30 \times 3,500$	430	780	540	21	7	16	28	90	55	43
$40 \times 4,000$	2,038	5,327	1,005	31	40	18	79	93	57	88
$40 \times 4,500$	12,738	22,753	1,729	294	702	39	98	60	53	92
$50 \times 5,000$	46,873	93,479	41,157	497	1,059	558	170	92	48	49
$50 \times 5,500$	196,010	403,927	101,092	2,038	3,288	1,570	120	98	65	62
$60 \times 6,000$	200,746	405,161	184,000	2,199	3,507	2,044	212	99	54	56
$60 \times 6,500$	180,529	603,289	112,259	3,099	4,798	2,570	249	99	77	71
$70 \times 7,000$	260,811	344,230	71,358	4,094	3,601	1,758	295	73	59	81

Table 4: Number of nodes, time, and cuts for problems with three-term semi-continuous constraints

$m \times n$	nodes			time			cuts		gapimp	
	mip	sc		mip	sc		mip	sc	mip	sc
		B&B	B&C		B&B	B&C				
$30 \times 3,000$	421	738	312	31	51	23	28	17	52	39
$30 \times 3,500$	1,034	641	205	71	52	12	37	54	47	58
$40 \times 4,000$	12,527	8,310	1,425	640	320	89	96	92	71	81
$40 \times 4,500$	9,581	4,515	1,130	580	207	71	94	84	68	77
$50 \times 5,000$	65,491	3,979	1,157	3,108	319	98	217	86	49	86
$50 \times 5,500$	381,520	203,386	91,010	5,375	1,386	973	415	95	54	70
$60 \times 6,000$	581,940	109,145	83,678	6,812	1,026	1,052	438	99	48	55
$60 \times 6,500$	980,529	603,289	259,612	8,408	4,736	2,000	497	88	75	81
$70 \times 7,000$	10^6+	785,310	317,950	10,804	6,812	2,730	420	97	58	89

References

[1] J. Aráoz, L. Evans, R.E. Gomory, and E.L. Johnson, "Cyclic Group and Knapsack Facets," *Mathematical Programming* 96, 377-408 (2003).

[2] E. Balas, "Facets of the Knapsack Polytope," *Mathematical Programming* 8, 146-164 (1975).

[3] E.M.L. Beale, "Integer Programming," in K. Schittkowski (Ed.), *Computational Mathematical Programming*, NATO ASI Series, Vol. f15, Springer-Verlag, 1985, pp. 1-24.

[4] E.M.L. Beale and J.A. Tomlin, "Special Facilities in a General Mathematical Programming System for Nonconvex Problems Using Ordered Sets of Variables," in J. Lawrence

(Ed.), *Proceedings of the Fifth International Conference on Operations Research*, Tavistock Publications, 1970, pp. 447-454.

[5] L.T. Biegler, I.E. Grossmann, and A.W. Westerberg, *Systematic Methods of Chemical Process Design*, Prentice Hall, 1997.

[6] D. Bienstock, "Computational Study of a Family of Mixed-Integer Quadratic Programming Problems," *Mathematical Programming* 74, 121-140 (1996).

[7] R.E. Bixby, S. Ceria, C.M. McZeal, and M.W.P. Savelsbergh, "An Updated Mixed Integer Programming Library: MIPLIB 3.0," *Technical Report TR98-03*, Department of Computational and Applied Mathematics, Rice University (1998).

[8] S. Ceria, C. Cordier, H. Marchand, and L.A. Wolsey, "Cutting Planes for Integer Programs with General Integer Variables," *Mathematical Programming* 81, 201-214 (1998).

[9] I.R. de Farias JR., E.L. Johnson, and G.L. Nemhauser, "A Generalized Assignment Problem with Special Ordered Sets: A Polyhedral Approach," *Mathematical Programming* 89, 187-203 (2000).

[10] I.R. de Farias JR., E.L. Johnson, and G.L. Nemhauser, "Branch-and-Cut for Combinatorial Optimization Problems without Auxiliary Binary Variables," *Knowledge Engineering Review* 16, 25-39 (2001).

[11] I.R. de Farias JR., E.L. Johnson, and G.L. Nemhauser "Facets of the Complementarity Knapsack Polytope," *Mathematics of Operations Research* 27, 210-226 (2002).

[12] I.R. de Farias JR. and G.L. Nemhauser, "A Polyhedral Study of the Cardinality Constrained Knapsack Problem," in W.J. Cook and A.S. Schulz (Eds.), *Integer Programming and Combinatorial Optimization*, Lecture Notes in Computer Science, Vol. 2337, Springer, 2002, pp. 291-303.

[13] I.R. de Farias JR. and G.L. Nemhauser, "A Polyhedral Study of the Cardinality Constrained Knapsack Problem," *Mathematical Programming* 96, 439-467 (2003).

[14] D.R. Fulkerson, "Blocking and Antiblocking Pairs of Polyhedra," *Mathematical Programming* 1, 168-194 (1971).

[15] D.R. Fulkerson, "Antiblocking Polyhedra," *Journal of Combinatorial Theory B* 12, 56-71 (1972).

[16] R.E. Gomory, E.L. Johnson, and L. Evans, "T-Space and Cutting Planes," *Mathematical Programming* 96, 341-375 (2003).

[17] Z. Gu, G.L. Nemhauser, and M.W.P. Savelsbergh, "Lifted Cover Inequalities for 0-1 Integer Programs: Computation," *INFORMS Journal on Computing* 4, 427-437 (1998).

[18] P.L. Hammer, E.L. Johnson, and U.N. Peled, "Facets of Regular 0-1 Polytopes," *Mathematical Programming* 8, 179-206 (1975).

[19] J.N. Hooker, "Logic, Optimization, and Constraint Programming," *INFORMS Journal on Computing* 14, 295-321 (2002).

[20] T. Ibaraki, "The Use of Cuts in Complementary Programming," *Operations Research* 21, 353-359 (1973).

[21] T. Ibaraki, T. Hasegawa, K. Teranaka, and J. Iwase, "The Multiple-Choice Knapsack Problem," *Journal of the Operations Research Society of Japan* 21, 59-95 (1978).

[22] R.G. Jeroslow, "Cutting Planes for Complementarity Constraints," *SIAM Journal on Control and Optimization* 16, 56-62 (1978).

[23] R.S. Laundy, "Some Logically Constrained Mathematical Programming Problems," *Ph.D. Thesis*, University of Southampton, Southampton, UK (1983).

[24] Q. Louveaux and L.A. Wolsey, "Lifting, Superadditivity, Mixed Integer Rounding, and Single Node Flow Sets Revisited," CORE Discussion Paper 2003/1 (2003).

[25] T.L. Magnanti, P. Mirchandani, and R. Vachani, "The Convex Hull of Two Core Capacitated Network Design Problems," *Mathematical Programming* 60, 233-250 (1993).

[26] H. Marchand, A. Martin, R. Weismantel, and L.A. Wolsey, "Cutting Planes in Integer and Mixed Integer Programming," *Discrete Applied Mathematics* 123, 397-446 (2002).

[27] G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization*, Wiley-Interscience, 1988.

[28] A.F. Perold, "Large-Scale Portfolio Optimization," *Management Science* 30, 1143-1160 (1984).

[29] J.P.P. Richard, I.R. de Farias JR., and G.L. Nemhauser, "Lifted Inequalities for 0–1 Mixed-Integer Programming: Basic Theory and Algorithms," in W.J. Cook and A.S. Schulz (Eds.), *Integer Programming and Combinatorial Optimization*, Lecture Notes in Computer Science, Vol. 2337, Springer, 2002, pp. 161-175.

[30] J.P.P. Richard, I.R. de Farias JR., and G.L. Nemhauser, "Lifted Inequalities for 0–1 Mixed-Integer Programming: Basic Theory and Algorithms," *Mathematical Programming* 98, 89-113 (2003).

[31] H.P. Williams, "The Reformulation of Two Mixed Integer Programming Problems," *Mathematical Programming* 14, 325-331 (1978).

[32] L.A. Wolsey, "Faces for a Linear Inequality in 0-1 Variables," *Mathematical Programming*, 8, 165-178 (1975).

[33] L.A. Wolsey, "Strong Formulations for Mixed-Integer Programs: Valid Inequalities and Extended Formulations," *Mathematical Programming* 97, 423-447 (2003).