Contributions to General Mixed Integer Knapsack Problems

Alexander Martin   Robert Weismantel*

Abstract

This paper deals with a general mixed integer knapsack polyhedron for which we introduce and analyze a new family of inequalities. We discuss the value of this family both from a theoretic and a computational point of view.

Keywords: knapsack problem, mixed integer programming, polyhedral combinatorics.

1 Introduction

Various classical combinatorial optimization problems become today tractable with polyhedral methods. Indeed, the last decade has emerged a large number of results that play the basis of modern cutting plane algorithms for many combinatorial optimization problems. On the other hand, quite little research has been carried out that deals with a polyhedral study of general mixed integer programming models of the form

$$\max c^T x + d^T y : \ Ax + By \leq \alpha, \ x \in \mathbb{Z}^n, y \in \mathbb{R}^q,$$

with matrices $A \in \mathbb{Z}^{m \times n}$, $B \in \mathbb{Z}^{m \times q}$ and vectors $\alpha \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$ and $d \in \mathbb{Z}^q$. For quite a long time, Gomory cutting planes [G60] have been and still are the main ingredient for current cutting plane implementations.

The situation changes when we restrict our attention to general mixed 0/1 programming problems, i.e., problems for which all integer variables are bounded by zero and one. For such problems, the disjunctive programming approach (see, for instance [B75]), the method of lift and project (see, for instance [BCC93]) and flow cover inequalities introduced in [PRW85] support the cutting plane phase of current solvers for mixed integer programming problems.

This paper is an extension of a paper of the second author about the 0/1 knapsack polytope [W97]. We deal with the polyhedron associated with the feasible solutions of a general mixed integer knapsack problem in which all variables may have arbitrary, but finite bounds. We introduce in Section 2 our model and the family of weight inequalities that turn out to be valid for this model. Section 3 deals with an analysis of weight inequalties in special cases. We present a family of knapsack polyhedra for which essentially weight inequalities are sufficient to describe the associated polyhedron. Computational experiments with this family of inequalities are reported in Section 4.

*Supported by a "Gerhard-Hess-Forschungsförderpreis" of the German Science Foundation (DFG).

2 The Family of Weight Inequalities

Let N, Q be mutually disjoint finite subsets of $\mathbb{N} \setminus \{0\}$ of cardinality n and q, respectively. For given vectors $a \in \mathbb{N}^n$, $b \in \mathbb{N}^q$, $u \in \mathbb{N}^{n+q}$ and a number $\alpha \in \mathbb{N} \setminus \{0\}$, we investigate

$$\begin{aligned} X(N,Q) := \quad & \{\, (x,y) \in \mathbb{Z}^n \times \mathbb{R}^q : \\ & \textstyle\sum_{i \in N} a_i x_i + \sum_{i \in Q} b_i y_i \leq \alpha \\ & 0 \leq (x,y) \leq u \,\}, \end{aligned} \tag{2.1}$$

where we use the notation (x, y) to abbreviate $(x^T, y^T)^T$. By definition, the members in $X(N,Q)$ are feasible solutions of the mixed integer knapsack problem with general lower and upper bounds on the variables,

$$\max c^T x + d^T y : \; (x,y) \in X(N,Q).$$

In this section we study the polyhedron $\operatorname{conv}(X(N,Q))$. There is one elementary family of inequalities that is valid for $\operatorname{conv}(X(N,Q))$ that under certain assumptions suffices to describe $\operatorname{conv}(X(N,Q))$. We call this family *weight inequalities.*

Definition 2.1. *For $T \subseteq N$ and $S \subseteq Q$ such that $\sum_{i \in T} a_i u_i + \sum_{i \in S} b_i u_i < \alpha$, we denote by*

$$r(T,S) := \alpha - \sum_{i \in T} a_i u_i - \sum_{i \in S} b_i u_i$$

the residual knapsack capacity *of the feasible solution $(x,y) \in X(N,Q)$ with $x_i := u_i$ for all $i \in T$, $y_i := u_i$ for all $i \in S$, $x_i, y_i := 0$, otherwise. The* weight inequality with respect to T, S *is the inequality*

$$\sum_{i \in T} a_i x_i + \sum_{i \in S} b_i y_i + \sum_{i \in N \setminus T} (a_i - r(T,S))^+ x_i \leq \alpha - r(T,S),$$

where $v^+ := \max\{0, v\}$ for $v \in \mathbb{R}$.

The name *weight inequality* reflects that the coefficients of the variables in $T \cup S$ equal their original weights. The coefficients of the (rational) variables in $Q \setminus S$ are zero. Likewise, the coefficient of an (integer) variable i in $N \setminus T$ is zero, if the weight a_i is smaller than $r(T,S)$. Otherwise, it is the weight a_i reduced by the value $r(T,S) > 0$.

Example 2.1. *Consider the convex hull of tuples of vectors $(x,y) \in \{0,1\}^6 \times [0,1]^2$ that satisfy*

$$x_1 + x_2 + x_3 + x_4 + 3x_5 + 4x_6 + 2y_1 + 3y_2 \leq 4$$

Setting $T := \{2,3,4\}$, $S = \emptyset$, we obtain that $r(T,S) = 1$. Then $a_1 - r(T,S) = 0$, $a_5 - r(T,S) = 2$ and $a_6 - r(T,S) = 3$. The weight inequality associated with T, S reads

$$x_2 + x_3 + x_4 + 2x_5 + 3x_6 \leq 3.$$

For $T := \{1\}$, $S = \{1\}$, the weight inequality associated with T, S reads

$$x_1 + 2y_1 + 2x_5 + 3x_6 \leq 3.$$

Weight inequalities are valid for the mixed integer knapsack polyhedron $\mathrm{conv}(X(N,Q))$.

Proposition 2.1. *For $T \subseteq N$, $S \subseteq Q$ such that $\sum_{i\in T} a_i u_i + \sum_{i\in S} b_i u_i < \alpha$, the weight inequality with respect to T, S is valid for* $\mathrm{conv}(X(N,Q))$.

Proof. Let $(x,y) \in X(N,Q)$. If $\sum_{i\in N\setminus T:\, a_i > r(T,S)} x_i = 0$, then by definition,

$$\sum_{i\in T} a_i x_i + \sum_{i\in S} b_i y_i + \sum_{i\in N\setminus T} (a_i - r(T,S))^+ x_i \le \alpha - r(T,S).$$

Otherwise, $\sum_{i\in N\setminus T:\, a_i > r(T,S)} x_i \ge 1$ holds. We obtain

$$\begin{array}{l} \sum_{i\in T} a_i x_i + \sum_{i\in S} b_i y_i + \sum_{i\in N\setminus T} (a_i - r(T,S))^+ x_i \quad \le \\ \sum_{i\in N} a_i x_i + \sum_{i\in Q} b_i y_i - r(T,S) \sum_{i\in N\setminus T:\, a_i > r(T,S)} x_i \quad \le \\ \alpha - r(T,S). \end{array}$$

This proves the statement. □

The family of weight inequalities subsume various preprocessing operations that software packages have sometimes incorporated for tightining integer programming formulations.

One such example is the following coefficient-reduction operation. For $u, \alpha \in \mathbb{N}$, $u < \alpha$ consider the feasible set

$$\mathcal{F} := \{x \in \mathbb{Z}^2 : \ 0 \le x_1 \le u,\ 0 \le x_2 \le 1,\ x_1 + \alpha x_2 \le \alpha\}.$$

It is easy to see that this feasible set has an equivalent formulation as

$$\mathcal{F} = \{x \in \mathbb{R}^2 : \ 0 \le x_1 \le u,\ 0 \le x_2 \le 1,\ x_1 + u x_2 \le u\}.$$

The inequality $x_1 + u x_2 \le u$ is a weight inequality (w.r.t. $T = \{1\}$, $S = \emptyset$) that is valid for $\mathrm{conv}(\mathcal{F})$. In fact, this operation of tightining coefficients generalizes to the following situation.

Consider the constraint

$$\sum_{i\in R} x_i + \sum_{j\in F} \alpha x_j \le \alpha, \tag{2.2}$$

where F is a subset of 0/1 variables, R is a subset of integer variables, each variable $i \in R$ having an upper bound u_i and $\sum_{i\in R} u_i < \alpha$. The constraint (2.2) may be replaced by

$$\sum_{i\in R} x_i + \sum_{j\in F} (\sum_{i\in R} u_i) x_j \le \sum_{i\in R} u_i. \tag{2.3}$$

The inequality (2.3) is a weight inequality for

$$P := \mathrm{conv}\{x \in \mathbb{Z}^R \times \{0,1\}^F : \ x \text{ satisfies } (2.2)\}.$$

In fact, P may be described as

$$P = \mathrm{conv}\{x \in \mathbb{Z}^R \times \{0,1\}^F : \ x \text{ satisfies } (2.3)\}.$$

Investigating the latter formulation, one recognizes that all the facets of P are induced by the family of weight inequalities,

$$x_i + \sum_{j \in F} u_i x_j \leq u_i \text{ for all } i \in R. \tag{2.4}$$

This system of inequalities together with lower and upper bound constraints on the variables describes P.

A special case of this latter preprocessing operation may also be found under the name *probing*.

3 A Family of Mixed Integer Knapsack Polyhedra

Having introduced weight inequalities for a general mixed integer knapsack problem, we indicate in this and the subsequent section that these inequalities are useful, at least in special cases. This section is devoted to this question from a more theoretical point of view. We demonstrate that there is a family of general mixed integer knapsack problems for which weight inequalities are needed in order to describe the associated polyhedron. This family of problems is defined as follows.

Let N_1, N_2, Q be mutually disjoint finite subsets of $\mathbb{N} \setminus \{0\}$ of cardinality n_1, n_2 and q, respectively. Let $n = n_1 + n_2$. For $\alpha \in \mathbb{N} \setminus \{0\}$ and given vectors $a \in \mathbb{N}^n$ such that $a_i = 1$ for all $i \in N_1$ and $a_i \geq \lfloor \frac{\alpha}{2} \rfloor + 1$ for all $i \in N_2$, $b \in \mathbb{N}^q$ and $u \in \mathbb{N}^{n+q}$, we study the mixed integer knapsack polyhedron $\operatorname{conv}(X(N_1 \cup N_2, Q))$ defined in (2.1).

Theorem 1. *The system of all weight inequalities w.r.t. subsets T, S of N_1 and Q, respectively, the set of all lower and upper bounds, and the two inequalities*

$$\begin{array}{lcl} \sum_{i \in N_2} x_i & \leq & 1 \\ \sum_{i \in N_1} x_i + \sum_{i \in N_2} a_i x_i + \sum_{i \in Q} b_i y_i & \leq & \alpha \end{array}$$

describe $\operatorname{conv}(X(N_1 \cup N_2, Q))$ *if $a_i = 1$ for all $i \in N_1$ and $a_i \geq \lfloor \frac{\alpha}{2} \rfloor + 1$ for all $i \in N_2$.*

Proof. Obviously, the system of inequalities is valid for $\operatorname{conv}(X(N_1 \cup N_2, Q))$, see Proposition 2.1. It remains to show that it suffices to describe $\operatorname{conv}(X(N_1 \cup N_2, Q))$. To see this, first notice that $\operatorname{conv}(X(N_1 \cup N_2, Q))$ is full dimensional. Let the inequality $c^T x + d^T y \leq \gamma$ induce a facet F of $\operatorname{conv}(X(N_1 \cup N_2, Q))$. We assume that $c^T x + d^T y \leq \gamma$ is not a positive multiple of one of the non-negativity constraints, the upper bound constraints and the knapsack inequality. Since $a, b > 0$, we have that $c_i, d_j \geq 0$ for all $i \in N_1 \cup N_2$, $j \in Q$. We define $T := \{i \in N_1 : c_i > 0\}$ and $S := \{i \in Q : d_i > 0\}$. There are three claims that we need in order to show the statement,

1. $\sum_{i \in T} u_i + \sum_{i \in S} b_i u_i < \alpha$;
2. The vector (x^0, y^0) with $x_i^0 := u_i$ for all $i \in T \cup S$, $x_i^0 := 0$, otherwise is contained in F;

3. $c_i = 0$ for all $i \in N_1 \setminus T$, $c_i = 0$ for all $i \in \{j \in N_2 : \; a_j \le r(T,S)\}$ and $d_i = 0$ for all $i \in Q \setminus S$.

Claim 1 follows by noting that if $\sum_{i\in T} u_i + \sum_{i\in S} b_i u_i \ge \alpha$ would hold, then every point in $X(N_1 \cup N_2, Q) \cap F$ would satisfy the equation $\sum_{i\in N_1} x_i + \sum_{i\in N_2} a_i x_i + \sum_{i\in Q} b_i y_i = \alpha$. (Recall that $a_i = 1$ for all $i \in T$.) This contradicts our assumption that F is not induced by the original knapsack inequality.

Note that Claim 1 implies $(x^0, y^0) \in X(N_1 \cup N_2, Q)$ and $r(T,S) > 0$.

We may assume that Claim 2 holds, for if not, then every integral point in F would satisfy the equation $\sum_{i\in N_2} x_i = 1$. Therefore, F must be the facet induced by this inequality.

Denoting by e^i the i-th unit vector we derive from Claims 1 and 2 that $(x^0, y^0) + e^i \in X(N_1 \cup N_2, Q)$ for all $i \in N_1 \setminus T$ and for all $i \in \{j \in N_2 : a_j \le r(T,S)\}$. Moreover, $(x^0, y^0) + e^i \in F$. Accordingly, $(x^0, y^0) + \frac{1}{b_i} e^i \in X(N_1 \cup N_2, Q)$ and $(x^0, y^0) + \frac{1}{b_i} e^i \in F$ for all $i \in Q \setminus S$. This shows Claim 3.

We define a function $f : N_2 \mapsto \mathbb{R}^+$ by setting

$$
\begin{array}{rll}
f(i) := & \min & \sum_{j\in T} c_j z_j + \sum_{j\in S} d_j t_j, \\
 & \text{s.t.} & \sum_{j\in T} z_j + \sum_{j\in S} b_j t_j = a_i - r(T,S), \\
 & & 0 \le z_j \le u_j, z_j \in \mathbb{Z} \text{ for all } j \in T, \\
 & & 0 \le t_j \le u_j, t_j \in \mathbb{Q} \text{ for all } j \in S.
\end{array}
\tag{3.1}
$$

Note that a solution of problem (3.1) attains the minimal value (w.r.t. c, d) by which the solution (x^0, y^0) must be decreased in order to obtain a feasible solution with $x_i = 1$. We show

4. $c_i = f(i)$ for all $i \in N_2$.

To see that $c_i \le f(i)$, let (z,t) be a solution of the program (3.1). The vector (x^i, y^i) defined as $x^i_j = x^0_j - z_j$ for all $j \in T$, $x^i_j = 0$ for all $j \in N_2 \setminus \{i\}$, $x^i_i = 1$, $y^i_j = y^0_j - t_j$ for all $j \in S$ and $y^i_j = 0$ for all $j \in Q \setminus S$ satisfies the condition that $(x^i, y^i) \in X(N_1 \cup N_2, Q)$. This implies that $c_i \le f(i)$.

To see that $c_i \ge f(i)$ note that F is a facet. Therefore, there exists, for every $i \in N_2$, a feasible point $(x,y) \in F$ such that $x_i = 1$. We may assume that $x_j = 0$ for all $j \in N_1 \cup N_2$ with $c_j = 0$. Accordingly, we can assume that $y_j = 0$ for all $j \in Q$ such that $d_j = 0$. Since $(x^0, y^0) \in F$, $c^T(x - x^0) + d^T(y - y^0) = 0$. Setting $z_j := x^0_j - x_j$ for all $j \in T$ and $t_j := y^0_j - y_j$ for all $j \in S$, we obtain that (z,t) is a feasible solution of (3.1). Taking into account that both (x,y) and (x^0, y^0) are feasible solutions contained in F, we obtain $c_i = \sum_{j\in T} c_j z_j + \sum_{j\in S} d_j t_j \ge f(i)$.

It follows that every x in $F \cap X(N_1 \cup N_2, Q)$ also satisfies the weight inequality associated with T,S as an equation. Since F is a facet, the inequality $c^T x + d^T y \le \gamma$ (after appropriate scaling) must coincide with the weight inequality with respect to T,S. □

In the more general case when we neglect the condition that $a_i \ge \lfloor \frac{\alpha}{2} \rfloor + 1$ for all $i \in N_2$, the associated polyhedron $\text{conv}(X(N_1 \cup N_2, Q))$ is not necessarily described only by weight inequalities. However, weight inequalities are still needed in a minimal description of $\text{conv}(X(N_1 \cup N_2, Q))$, because they induce facets. This follows from

Proposition 3.1. *Let* $\text{conv}(X(N,Q))$ *be the mixed integer knapsack polyhedron defined in (2.1). A weight inequality w.r.t. subsets* T, S *of* N *and* Q *defines a facet of* $\text{conv}(X(N,Q))$ *if* $r(T,S) > 0$ *and* $a_i = 1$ *for all* $i \in T$.

4 Experiments with Weight Inequalities

In this section we investigate whether weight inequalities occur in real world models and examine to which extend they help in solving practical problems faster. We use the library of mixed integer programs `MIPLIB`[1] as our test set. To answer both questions we have incorporated weight inequalities into a general mixed integer programming solver, called `SIP`. `SIP` is an LP-based branch-and-bound algorithm and is currently under our development; a description of its features will be published in a forthcoming paper. Input of our computational tests are the `MIPLIB` problems, presolved by `SIP`. We interpret each single inequality of the constraint system as a mixed integer knapsack problem. To meet the requirements in (2.1) all coefficients have to be positive, which can easily be obtained by complementing variables. In addition, all variables with a non-zero coefficient in the particular inequality must have finite lower and upper bounds. Moreover, we require that the support of the inequality is at least three, otherwise a complete description of the mixed integer knapsack polytope associated with this inequality is already obtained after presolve, see the discussions in Section 2. Finally, we do not consider inequalities whose coefficients are solely $0, \pm 1$. For convenience we call inequalities satisfying all these requirements *feasible*. It turns out that 48 out of 68 `MIPLIB` problems contain feasible inequalities. For each of these 48 instances and for each feasible inequality we try to derive weight inequalities that cut off the current optimal LP solution. To do so, we must solve the following separation problem.

Problem 4.1. *Given* $a \in \mathbb{N}^n, b \in \mathbb{N}^q$, $\alpha \in \mathbb{N}$ *and* $(x, y) \in \mathbb{Q}^{n+q}$. *Decide whether* (x, y) *satisfies all weight inequalities of* $\operatorname{conv}(X(N, Q))$. *If not, find one that is violated by* (x, y).

Problem 4.1 turns out to be $\mathcal{NP}$-hard. Therefore, we developed a heuristic which proceeds along the following lines.

Algorithm 4.2.

1. *Sort the components of the vector* $((x^*, y^*) - u) \circ (a, b)$ *in increasing order, where* (x^*, y^*) *is the current optimal LP solution, and for vectors* v, w *the symbol* $v \circ w$ *denotes the vector with components* $v_i w_i$.

2. *Construct sets* $T \subseteq N$ *and* $S \subseteq Q$ *following the order of Step 1 in a greedy fashion as long as* $\sum_{i \in T} u_i a_i + \sum_{i \in S} u_i b_i < \alpha$.

3. *Check whether the weight inequality w.r.t.* S, T *is violated.*

We have experimented with different strategies to determine sets S and T. The ordering of the variables according to their contribution to the slack of the inequality (= left hand side - right hand side) as outlined in Step 1 turned out to perform best. First test runs with this heuristic show that many inequalities are found whose slack is positive. The question arises which of these inequalities should be added to the current LP. Taking all with positive slack is not a good choice, since weight inequalities tend to have high coefficients. We performed several tests on how to accomplish this fact and our final choice fell on the following. We add a violated inequality to the LP if the slack divided by the

Example	SIP with kn			SIP with kn + wi			
	B&B	kn	Time	B&B	kn	wi	Time
bell3b	14301	0	125.6	10948	0	111	98.4
bell5	2456	0	15.1	8542	0	15	39.5
enigma	7224	2625	156.7	3631	1425	1	64.1
gt2	68547	0	623.9	264	0	1	2.5
mitre	1143	2325	2756.1	673	2117	741	1930.2
noswot	99366	0	3600.0	95418	0	540	3600.0

Table 1: A comparison of `SIP` with and without weight inequalities

absolute value of the largest coefficient is at least $\epsilon = 0.05$. Table 4 shows some results.

We made the following comparison. We used `SIP` with default settings of the parameters and turned on the options *strong branching* and *knapsack separation*; *strong branching* is a branching strategy introduced by `CPLEX`[2] that is in particular useful for difficult MIP problems; separation of knapsack inequalities is applied to all inequalities containing only 0/1 variables. (How these inequalities are separated is of no importance for this test.) We compared the strategy to the one where we separate weight inequalities in addition. We tested both alternatives for all 48 `MIPLIB` problems that contain feasible inequalities. For just the six problems that we list in Table 4, a significant difference in time or solution quality occured. Columns 2 through 4 show the number of branch-and-bound nodes, the number of added knapsack inequalities, and the time (measured on a SUN Enterprise 3000) needed to solve the problem using `SIP` without weight inequalities. Columns 5 through 8 present the corresponding numbers when weight inequalities are separated. Column 7 contains information about the number of weight inequalities added to the linear programs. We stoped after 3600 CPU seconds. Within this time limit we could not solve example *noswot* to optimality. For problem *noswot* weight inequalities help to find a significantly better solution, a phenomenon we do not have an explanation for. For the other five models we find the optimal solution with any of the two strategies. With the exception of *bell5* it pays to incorporate weight inequalities. The most impressive example is *gt2* where just one weight inequality speeds up the solution time enormously. The conclusion we draw from our computational experiences is that in general weight inequalities do not produce significantly worse results (in time and quality). On the other hand, there are examples like *gt2* and *mitre* where weight inequalities help to solve general mixed integer programming problems much faster.

References

[B75] E. Balas, *Disjunctive programming: Cutting planes from logical conditions*, in O. L. Mangasarian et al. (eds), Nonlinear Programming 2, Academic Press, London, 279–312 (1975).

[1] available from URL `http://www.caam.rice.edu:80/~bixby/miplib/miplib.html`
[2] CPLEX is a registered trademark of CPLEX Optimization, Inc.

[BCC93] E. Balas, S. Ceria and G. Cornuéjols, *A lift-and-project cutting plane algorithm for mixed* $0-1$ *programs*, Mathematical Programming 58, 295–324 (1993).

[G60] R. Gomory, *Solving linear programming problems in integers*, in R. Bellman and M. Hall (eds), Combinatorial analysis, Proc. of Symposia in Applied Mathematics, vol 10, Providence RI (1960).

[PRW85] M.W. Padberg, T.J. Van Roy, L.A. Wolsey, *Valid inequalities for fixed charge problems*, Operations Research 33, 842–861 (1985).

[W97] R. Weismantel, *On the* $0/1$ *knapsack polytope*, Mathematical Programming 77, 49 –68 (1994).