

Cover Inequalities

Konstantinos Kaparis Adam N. Letchford

Many problems arising in OR/MS can be formulated as *Mixed-Integer Linear Programs* (MILPs); see entry #1.4.1.1. If one wishes to solve a class of MILPs to proven optimality, or near optimality, it is often useful to have a class of *cutting planes* available. Cutting planes are linear inequalities that are satisfied by all feasible solutions to the MILP, but may be violated by solutions to its continuous relaxation (see entry #1.4.1.5).

This article is concerned with one particular class of cutting planes — the *cover inequalities* — along with some variants of them. Cover inequalities were originally defined in the context of the *0-1 Knapsack Problem* (0-1 KP), but can be applied to any *0-1 Linear Program* (0-1 LP). Moreover, the idea underlying the cover inequalities has been extended to yield cutting planes for other knapsack-type problems and other kinds of MILPs.

The article is structured as follows. In Section 1, we recall the theory underlying the inequalities. In Section 2, we discuss some algorithmic questions that must be addressed if one wishes to actually use cover inequalities in a cutting-plane algorithm. In Section 3, we list some of the problems to which cover inequalities have been applied. Finally, at the end of the article, we give some pointers for further reading.

1 Theory

1.1 Knapsack polytopes

As mentioned above, the cover inequalities were first presented in the context of the 0-1 KP. The 0-1 KP takes the following form:

$$\max \{p^T x : w^T x \leq c, x \in \{0, 1\}^n\}.$$

Here, $p \in \mathbb{Z}_+^n$ is the vector of *profits*, $w \in \mathbb{Z}_+^n$ is the vector of *weights* and $c \in \mathbb{Z}_+$ is the *knapsack capacity*.

The convex hull of feasible solutions to a 0-1 KP, i.e., the set

$$\text{conv} \{x \in \{0, 1\}^n : w^T x \leq c\},$$

is called a *knapsack polytope*. We will let $KP(w, c)$ denote this polytope. In theory, the strongest possible cutting planes, for a given w and c , are those that define *facets* of $KP(w, c)$ (see entry #1.4.3.1).

1.2 Cover and extended cover inequalities

Now, let N denote $\{1, \dots, n\}$. A set $C \subset N$ is called a *cover* if it satisfies $\sum_{j \in C} w_j > c$. Since it is impossible to set all of the variables in C to 1 simultaneously while maintaining feasibility, the linear inequality $\sum_{j \in C} x_j \leq |C| - 1$ is valid for $KP(w, c)$. This is the *cover inequality* (CI).

The CIs were introduced independently by Balas [6], Hammer *et al.* [30] and Wolsey [48]. These authors observed that the strongest CIs are obtained when the cover C is *minimal*, in the sense that no proper subset of C is also a cover.

A simple way to strengthen the CIs was given by Balas [6]. Compute $w^* := \max_{j \in C} w_j$ and define the *extension* of C as $E(C) := C \cup \{j \in N \setminus C : w_j \geq w^*\}$. The *extended cover inequality* (ECI) $\sum_{j \in E(C)} x_j \leq |C| - 1$ is then easily seen to be valid for $KP(w, c)$.

Example: Suppose that $n = 8$ and that the knapsack constraint takes the form $4x_1 + 6x_2 + 6x_3 + 6x_4 + 6x_5 + 7x_6 + 8x_7 + 15x_8 \leq 22$. The set $\{1, 2, 6, 7\}$ forms a cover, since $4 + 6 + 7 + 8 = 25 > 22$. Moreover, it is a minimal cover. The corresponding CI takes the form $x_1 + x_2 + x_6 + x_7 \leq 3$. Moreover, we have $w^* = 8$ and $E(C) = \{1, 2, 6, 7, 8\}$, and therefore the corresponding ECI takes the form $x_1 + x_2 + x_6 + x_7 + x_8 \leq 3$. \square

Although the ECIs dominate the CIs, they are still not guaranteed to define facets of $KP(w, c)$, as explained in the next subsection.

1.3 Lifted cover inequalities

Balas [6] and Wolsey [48] showed that, given any minimal cover C , there exists at least one facet-defining *lifted cover inequality* (LCI) of the form

$$\sum_{j \in C} x_j + \sum_{j \in N \setminus C} \alpha_j x_j \leq |C| - 1, \quad (1)$$

where $\alpha_j \geq 0$ for all $j \in N \setminus C$. Moreover, each such LCI dominates the ECI.

Example (continued): From the given cover $\{1, 2, 6, 7\}$, one can derive the following four LCIs:

$$\begin{aligned} x_1 + x_2 + x_3 + x_6 + x_7 + 2x_8 &\leq 3 \\ x_1 + x_2 + x_4 + x_6 + x_7 + 2x_8 &\leq 3 \\ x_1 + x_2 + x_5 + x_6 + x_7 + 2x_8 &\leq 3 \\ x_1 + x_2 + \frac{1}{2}x_3 + \frac{1}{2}x_4 + \frac{1}{2}x_5 + x_6 + x_7 + 2x_8 &\leq 3. \end{aligned}$$

Each of these LCIs clearly dominates the ECI, and all of them can be shown to define facets of $KP(w, c)$. \square

Note that a facet-defining LCI can have fractional coefficients.

Van Roy & Wolsey [45] noted that one can derive more general LCIs of the form:

$$\sum_{j \in C \setminus D} x_j + \sum_{j \in N \setminus C} \alpha_j x_j + \sum_{j \in D} \beta_j x_j \leq |C \setminus D| + \sum_{j \in D} \beta_j - 1, \quad (2)$$

where C is a cover and $D \subset C$. We will call the inequalities (2) ‘general’ LCIs, and we will follow Gu *et al.* [26] in referring to the inequalities (1) as ‘simple’ LCIs.

Example (continued): Assume as before that the knapsack constraint takes the form $4x_1 + 6x_2 + 6x_3 + 6x_4 + 6x_5 + 7x_6 + 8x_7 + 15x_8 \leq 22$. The inequality $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + 2x_7 + 3x_8 \leq 4$ can be shown to define a facet of $KP(w, c)$. It can be derived as a general LCI by setting $C = \{1, 2, 8\}$ and $D = \{8\}$. \square

We remark that the general LCIs with $|D| = 1$ dominate the so-called ‘ $(1, k)$ -configuration’ inequalities of Padberg [42].

2 Algorithmic Aspects

2.1 Computing lifting coefficients

A natural question is how to actually compute the coefficients α_j and β_j . The process of computing these coefficients is called *lifting*. One normally distinguishes between *sequential* lifting, in which the coefficients are computed one at a time, and *simultaneous* lifting, in which they are computed ‘all at once’, so to speak. (See entry #1.4.3.3 for more details on lifting.)

We will follow Gu *et al.* [26] in referring to the computation of the α_j and β_j as *up-lifting* and *down-lifting*, respectively.

The case that has received most attention is *sequential up-lifting*, which leads to simple LCIs with integral coefficients. Balas [6] and Wolsey [48] both pointed out that different lifting sequences can yield different simple LCIs. (This is how we generated the three simple LCIs with integral coefficients in the example in Subsection 1.3.) They also showed that, for a given lifting sequence, the up-lifting coefficients can be computed by solving a series of auxiliary 0-1 KPs.

Since solving 0-1 KPs is rather time-consuming, researchers looked for faster and simpler methods. Balas & Zemel [8] showed how to compute, in $O(n + |C| \log |C|)$ time, tight lower and upper bounds on the up-lifting coefficients, which are independent of the lifting sequence. The upper bounds were later improved by Gu *et al.* [29]. Crowder *et al.* [17] proposed to compute *approximate* up-lifting coefficients, by solving the continuous relaxation of the auxiliary 0-1 KPs, and rounding the answers down to integers. Finally, Zemel [49] pointed out that one can compute exact up-lifting coefficients for all variables in $O(n|C|)$ time, by dynamic programming.

The example given in Subsection 1.3 shows that there exist simple LCIs with fractional coefficients, which cannot be obtained with sequential up-lifting. To derive such simple LCIs, one must perform simultaneous up-lifting. Unfortunately, this is not easy. Indeed, Hartvigsen & Zemel [32] showed that even *recognising* simultaneously-lifted simple LCIs is \mathcal{NP} -hard. On the other hand, there exist a couple of simple and fast procedures for *approximate* simultaneous up-lifting [18, 29].

As for general LCIs, Gu *et al.* [26] claim that sequential lifting can be performed in $\mathcal{O}(n^3|C|)$ time, provided that the lifting sequence satisfies a certain condition. An $\mathcal{O}(nc)$ lifting algorithm, which imposes no condition on the lifting sequence, is described in our paper [36]. A simpler and faster alternative is to solve the continuous relaxation of the auxiliary 0-1 LPs, just as Crowder *et al.* [17] did for simple LCIs.

2.2 Separation algorithms

In order to use a class of valid inequalities as cutting planes, one needs a *separation algorithm*, i.e., a routine for generating the inequalities when they are violated (see entry #1.4.1.5). Unfortunately, the separation problems for CIs, simple LCIs and general LCIs have all been shown to be \mathcal{NP} -hard [21, 27, 37], and the same seems likely to be true for ECIs. Nevertheless, some useful exact and heuristic separation algorithms are known.

We begin with CIs. Crowder *et al.* [17] noted that CI separation is equivalent to the following knapsack-type problem:

$$\min \left\{ \sum_{j \in N} (1 - x_j^*) y_j : w^T y > c, y \in \{0, 1\}^n \right\}.$$

Here, y_j is a 0-1 variable, taking the value 1 if and only if j is to be inserted into the cover C . It is not hard to show that this problem can be solved by dynamic programming in $\mathcal{O}(nc)$ time, which is acceptable if c is not too large. Crowder *et al.* [17] suggested solving it heuristically instead, by inserting items into C in non-decreasing order of $(1 - x_j^*)/w_j$ until a cover is obtained. This heuristic takes $\mathcal{O}(n \log n)$ time.

As for ECIs, Gabrel & Minoux [23] showed that the separation problem can be reduced to a sequence of knapsack-type problems. In our paper [36], we presented a simplified version of their algorithm, which runs in $\mathcal{O}(n^2c)$ time, along with a dynamic programming algorithm that runs in only $\mathcal{O}(nc)$ time. We also presented an effective heuristic, that runs in $\mathcal{O}(n^2)$ time. All of our algorithms can produce more than one violated ECI per call.

As for simple LCIs, Crowder *et al.* [17] suggested using their heuristic to obtain a CI, and then performing up-lifting to strengthen it. They pointed out that it pays to up-lift the variables with positive x^* -value first, before up-lifting the variables with zero x^* -value.

Several papers address the separation of general LCIs [26, 33, 36, 45]. Van Roy & Wolsey [45] proposed to put only one item into D — namely, the item in C with the largest x^* -value — and leave down-lifting to the end. Hoffman & Padberg [33] proposed to set $D = \{j \in C : x_j^* = 1\}$, and then lift in the following order: up-lift the variables in $N \setminus C$ with positive x^* -value, down-lift the variables in D , then up-lift the variables in $N \setminus C$ with zero x^* -value. Gu *et al.* [26] used the same scheme, but used a different heuristic to construct C : they simply inserted items into C in non-increasing order of x^* -value, until a cover was obtained. In our paper [36], we used a similar scheme, but generated the initial covers by running our ECI heuristic.

A comparison of several different separation algorithms for CIs, ECIs, simple LCIs and general LCIs can be found in [36].

3 Applications

3.1 Application to combinatorial optimisation problems

Many important combinatorial optimisation problems can be relaxed, in a natural way, so that they decompose into one or more 0-1 KPs. If one solves such a problem with an LP-based solution algorithm, such as branch-and-cut, then one can use CIs, ECIs or LCIs as cutting planes.

For the sake of brevity, we mention just a few problems to which this approach has been applied:

- the *Multiple Knapsack Problem* [21],
- the *Multidimensional Knapsack Problem* [9, 23, 35, 36],
- the *Generalised Assignment Problem* [14, 25],
- the *Capacitated Facility Location Problem* [1, 2],
- the *Fixed-Charge Transportation Problem* [41],
- the *Capacitated Network Design Problem* [16],
- the *Prize-Collecting Traveling Salesman Problem* [7, 10],
- the *Orienteering Problem* [22],
- the *Resource-Constrained Project Scheduling Problem* [31].

3.2 Application to general 0-1 linear programs

As mentioned in the introduction, valid inequalities for 0-1 knapsack polytopes can in fact be used as cutting planes for general 0-1 LPs. The idea is to consider each individual constraint of the 0-1 LP as a knapsack constraint (complementing variables if necessary to obtain non-negative coefficients), and derive inequalities that are valid for the associated knapsack polytope. Provided the constraint matrix of the 0-1 LP is sparse, the inequalities generated in this way can be expected to be useful.

name	CI	ECI	sLCI	gLCI	All
P0033	63.55	71.93	71.93	80.62	87.42
P0040	76.82	100.00	100.00	100.00	100.00
P0201	33.78	33.78	33.78	33.78	33.78
P0282	94.27	94.35	94.35	96.21	98.59
P0291	95.23	95.23	95.23	96.40	99.43
P0548	67.67	67.68	67.68	67.71	84.34
P2756	86.26	86.26	86.26	86.26	86.36
bm23	5.56	15.82	15.82	16.11	20.08
lseu	39.87	61.36	61.36	66.20	76.09
mod008	3.75	17.59	17.59	18.24	89.23
pipex	16.68	27.96	27.96	73.34	86.55
sentoy	16.58	21.57	21.57	22.72	30.97
Aver.	50.00	57.80	57.80	63.13	74.49

Table 1: Percentage gap closed by inequalities on MIPLIB instances

To our knowledge, Crowder *et al.* [17] were the first to test this idea computationally. They designed what is now called a ‘cut-and-branch’ algorithm, in which cutting planes are added only at the root node of the branch-and-bound tree. The cutting planes used were simple LCIs. Hoffman & Padberg [33] extended this approach in two ways, by using full branch-and-cut (in which cutting planes can be added at any node), and using general LCIs in place of simple LCIs. Further experiments with this scheme were performed by Gu *et al.* [26].

CI, ECIs and LCIs turn out to be remarkably effective when applied to large, sparse 0-1 LPs. Table 1, adapted from [36], shows the percentage of the integrality gap that is closed by various inequalities, when applied to twelve 0-1 LPs taken from the MIPLIB [3]. The columns labelled ‘sLCI’ and ‘gLCI’ correspond to simple and general LCIs, respectively. The column labelled ‘All’ was obtained by calling an exact (and time-consuming) separation algorithm for general facets of knapsack polytopes. The CIs and ECIs were separated exactly, whereas the LCIs were separated heuristically. In all cases, apart from the ‘All’ column, the running times were negligible compared to the time taken to solve the instances to proven optimality.

Observe that even the (theoretically weak) CIs close half of the integrality gap on average, and the other inequalities close even more. This good performance is typical, though only when the constraint matrix is sparse [36]. Observe also that the simple LCIs gave exactly the same results as the ECIs. Indeed, we observed that up-lifting coefficients greater than 1 occurred very infrequently. This suggests that, if one does not wish to implement down-lifting, then one might as well use ECIs.

For Further Reading

There exist hard instances of the 0-1 KP that require an exponential number of nodes when solved by branch-and-bound, even if all simple LCIs are included in the LP relaxation [27, 34].

LCIs are not the only facet-defining inequalities for 0-1 knapsack polytopes. For other such inequalities, see [4, 46, 47].

Separation algorithms have been devised for the 0-1 knapsack polytope itself, rather than for specific classes of inequalities [5, 11, 12, 36].

Cover inequalities have been derived for more complex knapsack problems, with general integer variables, continuous variables, and so on [13, 15, 19, 20, 24, 35, 38, 39, 40].

An important related class of inequalities are the *lifted flow cover* inequalities [28, 43, 44], which are valid for fixed-charge problems.

References

- [1] K. Aardal (1998) Capacitated facility location: separation algorithms and computational experience. *Math. Program.*, 81, 149–175.
- [2] K. Aardal, Y. Pochet & L.A. Wolsey (1995) Capacitated facility location: valid inequalities and facets. *Math. Oper. Res.*, 20, 562–582.
- [3] T. Achterberg, T. Koch & A. Martin (2006) MIPLIB 2003. *Oper. Res. Lett.*, 34, 361–372.
- [4] A. Atamtürk (2005) Cover and pack inequalities for (mixed) integer programming. *Ann. Oper. Res.*, 139, 21–38.
- [5] P. Avella, M. Boccia & I. Vasilyev (2008) A computational study of exact knapsack separation for the generalized assignment problem. *Comput. Optim. & Appl.*, to appear.
- [6] E. Balas (1975) Facets of the knapsack polytope. *Math. Program.*, 8, 146–164.
- [7] E. Balas (1989) The prize collecting traveling salesman problem. *Networks*, 19, 621–636.
- [8] E. Balas & E. Zemel (1978) Facets of the knapsack polytope from minimal covers. *SIAM J. Appl. Math.*, 34, 119–148.
- [9] T. Bektas & O. Oguz (2007) On separating cover inequalities for the multidimensional knapsack problem. *Comput. & Oper. Res.*, 34, 1771–1776.

- [10] J.-F. Bérubé, M. Gendreau & J.-Y. Potvin (2009) A branch-and-cut algorithm for the undirected prize collecting traveling salesman problem. *Networks*, 54, 56–67.
- [11] E.A. Boyd (1992) A pseudo-polynomial network flow formulation for exact knapsack separation. *Networks*, 22, 503–514.
- [12] E.A. Boyd (1993) Generating Fenchel cutting planes for knapsack polyhedra. *SIAM J. Opt.*, 3, 734–750.
- [13] E.A. Boyd (1993) Polyhedral results for the precedence-constrained knapsack problem. *Discr. Appl. Math.*, 41, 185–201.
- [14] D. Cattrysse, Z. Degraeve & J. Tistaert (1996) Solving the generalised assignment problem using polyhedral results. *Eur. J. Oper. Res.*, 108, 618–628.
- [15] S. Ceria, C. Cordier, H. Marchand & L.A. Wolsey (1998) Cutting planes for integer programs with general integer variables. *Math. Program.*, 81, 201–214.
- [16] M. Chouman, T.G. Crainic & B. Gendron (2009) A cutting plane algorithm for multicommodity capacitated fixed-charge network design. *Technical report*, CIRRELT, Montreal.
- [17] H. Crowder, E. Johnson & M.W. Padberg (1983) Solving large-scale 0-1 linear programming programs. *Oper. Res.*, 31, 803–834.
- [18] T. Easton & K. Hooker (2008) Simultaneously lifting sets of binary variables into cover inequalities for knapsack polytopes. *Discr. Opt.*, 5, 254–261.
- [19] I.R. De Farias, E.L. Johnson & G.L. Nemhauser (2002) Facets of the complementarity knapsack polytope. *Math. Oper. Res.*, 27, 210–226.
- [20] E. Fernández & K. Jørnsten (1994) Partial cover and complete cover inequalities. *Oper. Res. Lett.*, 15, 19–33.
- [21] C.E. Ferreira, A. Martin & R. Weismantel (1996) Solving multiple knapsack problems by cutting planes. *SIAM J. Opt.*, 6, 858–877.
- [22] M. Fischetti & J.J. Salazar-Gonzalez (1998) Solving the orienteering problem through branch-and-cut. *INFORMS J. Comput.*, 10, 133–148.
- [23] V. Gabrel & M. Minoux (2002) A scheme for exact separation of extended cover inequalities and application to multidimensional knapsack problems. *Oper. Res. Lett.*, 30, 252–264.

- [24] F. Glover & H.D. Sherali (2008) Second-order cover inequalities. *Math. Program.*, 114, 207–234.
- [25] E.S. Gottlieb & M.R. Rao (1990) The generalized assignment problem: valid inequalities and facets. *Math. Program.*, 46, 31–52.
- [26] Z. Gu, G.L. Nemhauser & M.W.P. Savelsbergh (1998) Lifted cover inequalities for 0-1 linear programs: computation. *INFORMS J. Comput.*, 10, 427–437.
- [27] Z. Gu, G.L. Nemhauser & M.W.P. Savelsbergh (1999) Lifted cover inequalities for 0-1 integer programs: complexity. *INFORMS J. Comput.*, 11, 117–123.
- [28] Z. Gu, G.L. Nemhauser & M.W.P. Savelsbergh (1999) Lifted flow cover inequalities for mixed 0-1 integer programs. *Math. Program.*, 85, 439–467.
- [29] Z. Gu, G.L. Nemhauser & M.W.P. Savelsbergh (2000) Sequence independent lifting in mixed integer programming. *J. Comb. Opt.*, 4, 109–129.
- [30] P.L. Hammer, E.L. Johnson & U.N. Peled (1975) Facets of regular 0-1 polytopes. *Math. Program.*, 8, 179–206.
- [31] J.R. Hardin, G.L. Nemhauser & M.W.P. Savelsbergh (2008) Strong valid inequalities for the resource-constrained scheduling problem with uniform resource constraints. *Discr. Opt.*, 5, 19–35.
- [32] D. Hartvigsen & E. Zemel (1992) The complexity of lifted inequalities for the knapsack problem. *Discr. Appl. Math.*, 39, 113–123.
- [33] K.L. Hoffman & M.W. Padberg (1991) Improving LP-representations of zero-one linear programs for branch-and-cut. *ORSA J. Comput.*, 3, 121–134.
- [34] B. Hunsaker & C. Tovey (2005) Simple lifted cover inequalities and hard knapsack problems. *Discr. Opt.*, 2, 219–228.
- [35] K. Kaparis & A.N. Letchford (2008) Local and global lifted cover inequalities for the multidimensional knapsack problem. *Eur. J. Oper. Res.*, 186, 91–103.
- [36] K. Kaparis & A.N. Letchford (2009) Separation algorithms for 0-1 knapsack polytopes. *Math. Program.*, to appear.
- [37] D. Klabjan, G.L. Nemhauser & C. Tovey (1998) The complexity of cover inequality separation. *Oper. Res. Lett.*, 23, 35–40.

- [38] Q. Louveaux & R. Weismantel (2008) Polyhedral properties for the intersection of two knapsacks. *Math. Program.*, 113, 15–38.
- [39] H. Marchand & L.A. Wolsey (1999) The 0-1 knapsack problem with a single continuous variable. *Math. Program.*, 85, 15–33.
- [40] G.L. Nemhauser & P.H. Vance (1994) Lifted cover facets of the 0-1 knapsack polytope with GUB constraints. *Oper. Res. Lett.*, 16, 255–263.
- [41] G.L. Nemhauser & L.A. Wolsey (1988) *Integer and Combinatorial Optimization*. New York: Wiley.
- [42] M.W. Padberg (1980) $(1, k)$ -configurations and facets for packing problems. *Math. Program.*, 18, 94–99.
- [43] M.W. Padberg, T.J. Van Roy & L.A. Wolsey (1985) Valid linear inequalities for fixed charge problems. *Oper. Res.*, 33, 842–861.
- [44] T.J. Van Roy & L.A. Wolsey (1986) Valid inequalities for mixed 0-1 programs. *Discr. Appl. Math.*, 14, 199–213.
- [45] T.J. Van Roy & L.A. Wolsey (1987) Solving mixed integer programming problems using automatic reformulation. *Oper. Res.*, 35, 45–57.
- [46] R. Weismantel (1996) Hilbert bases and the facets of special knapsack polytopes. *Math. Oper. Res.*, 21, 886–904.
- [47] R. Weismantel (1997) On the 0-1 knapsack polytope. *Math. Program.*, 77, 49–68.
- [48] L.A. Wolsey (1975) Faces for linear inequalities in 0-1 variables. *Math. Program.*, 8, 165–178.
- [49] E. Zemel (1989) Easily computable facets of the knapsack polytope. *Math. Oper. Res.*, 14, 760–765.

Author Addresses:

Konstantinos Kaparis
School of Mathematics
University of Southampton
Southampton SO17 1BJ
United Kingdom.
E-mail: k.kaparis@soton.ac.uk

Adam N. Letchford
Department of Management Science
Lancaster University
Lancaster LA1 4YX
United Kingdom.
E-mail: a.n.letchford@lancaster.ac.uk

Abstract: Cover inequalities are a special kind of linear inequalities that are satisfied by the feasible solutions to knapsack problems. Although initially defined in the context of knapsack problems, cover inequalities and their variants can be used as cutting planes for a variety of integer programming and combinatorial optimisation problems. We review the theory of these inequalities, discuss certain algorithmic aspects of them, and survey some of the applications.

Key Words: integer programming — knapsack problems — cutting planes.