COVER INEQUALITIES

KONSTANTINOS KAPARIS
School of Mathematics, University of Southampton, Southampton, UK

ADAM N. LETCHFORD
Department of Management Science, Lancaster University, Lancaster, UK

Many problems arising in OR/MS can be formulated as *mixed-integer linear programs* (MILPs): see the article titled ***Formulating Good MILP Models*** in this encyclopedia. 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 ***Branch and Cut***).

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 the section titled "Theory," we recall the theory underlying the inequalities. In the section titled "Algorithmic Aspects," we discuss some algorithmic questions that must be addressed if one wishes to actually use cover inequalities in a cutting-plane algorithm. In the section titled "Applications," 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.

THEORY

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:

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

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, that is, the set

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

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 ***Basic Polyhedral Theory***).

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 $\sum_{j\in C} x_j \le |C| - 1$ is valid for $KP(w,c)$. This is the *cover inequality* (CI).

The CIs were introduced independently by Balas [1], Hammer *et al*. [2], and Wolsey [3]. 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 [1]. Compute $w^* := \max_{j\in C} w_j$ and define the *extension* of C as $E(C) := C \cup \{j \in N \setminus C : w_j \ge w^*\}$. The *extended* cover inequality (ECI) $\sum_{j\in E(C)} x_j \le |C| - 1$ is then easily seen to be valid for $KP(w,c)$.

Wiley Encyclopedia of Operations Research and Management Science, edited by James J. Cochran

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 $x_1 + x_2 + x_6 + x_7 + x_8 \leq 3$.

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

Lifted Cover Inequalities

Balas [1] and Wolsey [3] showed that, given any minimal cover C, there exists at least one facet-defining *lifted cover inequality* (LCI) of the form

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

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:

$$x_1 + x_2 + x_3 + x_6 + x_7 + 2x_8 \leq 3$$

$$x_1 + x_2 + x_4 + x_6 + x_7 + 2x_8 \leq 3$$

$$x_1 + x_2 + x_5 + x_6 + x_7 + 2x_8 \leq 3$$

$$x_1 + x_2 + \tfrac{1}{2}x_3 + \tfrac{1}{2}x_4 + \tfrac{1}{2}x_5 + x_6 + x_7 + 2x_8 \leq 3.$$

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

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

Van Roy and Wolsey [4] noted that one can derive more general LCIs of the form:

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

where C is a cover and $D \subset C$. We will call the inequalities (2) "general" LCIs, and we will follow Gu *et al*. [5] 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 $D = \{8\}$.

We remark that the general LCIs with $|D| = 1$ dominate the so-called "$(1, k)$-configuration" inequalities of Padberg [6].

ALGORITHMIC ASPECTS

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 ***Lifting Techniques for Mixed Integer Programming*** for more details on lifting.)

We will follow Gu *et al*. [5] 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 [1] and Wolsey [3] both pointed out that different lifting sequences can yield different simple LCIs. (This is how we generate the three simple LCIs with integral coefficients in the example in the section titled "Lifted Cover Inequalities.") 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 and Zemel [7] 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*. [8]. Crowder *et al*. [9] 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 [10] 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 the section titled “Lifted Cover Inequalities” 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 and Zemel [11] showed that even *recognizing* 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 [8,12].

As for general LCIs, Gu *et al*. [5] 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 conditions on the lifting sequence, is described in our article [13]. A simpler and faster alternative is to solve the continuous relaxation of the auxiliary 0–1 LPs, just as Crowder *et al*. [9] did for simple LCIs.

Separation Algorithms

In order to use a class of valid inequalities as cutting planes, one needs a *separation algorithm*, that is, a routine for generating the inequalities when they are violated (see ***Branch and Cut***). Unfortunately, the separation problems for CIs, simple LCIs, and general LCIs have all been shown to be $\mathcal{NP}$-hard [14–16], 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*. [9] 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 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*. [9] suggested solving it heuristically instead, by inserting items into C in nondecreasing 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 and Minoux [17] showed that the separation problem can be reduced to a sequence of knapsack-type problems. In our article [13], we presented a simplified version of their algorithm, which runs in $\mathcal{O}(n^2 c)$ 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*. [9] 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 articles address the separation of general LCIs [4,5,13,18]. Van Roy and Wolsey [4] proposed to include only one item into D—namely, the item in C with the largest x^*-value—and leave down-lifting to the end. Hoffman and Padberg [18] 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*. [5] used the same scheme, but used a different heuristic to construct C: they simply inserted items into C in nonincreasing order of x^*-value, until a cover was obtained. In our article [13], 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 Ref. 13.

APPLICATIONS

Application to Combinatorial Optimization Problems

Many important combinatorial optimization 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 KP* [14],
- the *multidimensional KP* [13,17,19,20],
- the *generalized assignment problem* [21,22],
- the *capacitated facility location problem* [23,24],
- the *fixed-charge transportation problem* [25],
- the *capacitated network design problem* [26],
- the *prize-collecting traveling salesman problem* [27,28],
- the *orienteering problem* [29],
- the *resource-constrained project scheduling problem* [30].

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 nonnegative 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.

To our knowledge, Crowder *et al*. [9] 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 and Padberg [18] 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*. [5].

Table 1. Percentage Gap Closed by Inequalities on MIPLIB Instances

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
Average	50.00	57.80	57.80	63.13	74.49

CIs, ECIs, and LCIs turn out to be remarkably effective when applied to large, sparse 0−1 LPs. Table 1, adapted from Ref [13], shows the percentage of the integrality gap that is closed by various inequalities, when applied to twelve 0−1 LPs taken from the MIPLIB [31]. The columns labeled "sLCI" and "gLCI" correspond to simple and general LCIs, respectively. The column labeled "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 [13]. 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 [15,32].

LCIs are not the only facet-defining inequalities for 0–1 knapsack polytopes. For other such inequalities, see Refs 33–35.

Separation algorithms have been devised for the 0–1 knapsack polytope itself, rather than for specific classes of inequalities [13,36–38].

Cover inequalities have been derived for more complex KPs, with general integer variables, continuous variables, and so on [20,39–46].

An important related class of inequalities are the *lifted flow cover* inequalities [47–49], which are valid for fixed-charge problems.

REFERENCES

1. Balas E. Facets of the knapsack polytope. Math Program 1975;8:146–164.
2. Hammer PL, Johnson EL, Peled UN. Facets of regular 0-1 polytopes. Math Program 1975; 8:179–206.
3. Wolsey LA. Faces for linear inequalities in 0-1 variables. Math Program 1975;8:165–178.
4. Van Roy TJ, Wolsey LA. Solving mixed integer programming problems using automatic reformulation. Oper Res 1987;35:45–57.
5. Gu Z, Nemhauser GL, Savelsbergh MWP. Lifted cover inequalities for 0-1 linear programs: computation. INFORMS J Comput 1998;10:427–437.
6. Padberg MW. $(1, k)$-configurations and facets for packing problems. Math Program 1980;18:94–99.
7. Balas E, Zemel E. Facets of the knapsack polytope from minimal covers. SIAM J Appl Math 1978;34:119–148.
8. Gu Z, Nemhauser GL, Savelsbergh MWP. Sequence independent lifting in mixed integer programming. J Comb Opt 2000;4: 109–129.
9. Crowder H, Johnson E, Padberg MW. Solving large-scale 0-1 linear programming programs. Oper Res 1983;31:803–834.
10. Zemel E. Easily computable facets of the knapsack polytope. Math Oper Res 1989;14: 760–765.
11. Hartvigsen D, Zemel E. The complexity of lifted inequalities for the knapsack problem. Discr Appl Math 1992;39:113–123.
12. Easton T, Hooker K. Simultaneously lifting sets of binary variables into cover inequalities for knapsack polytopes. Discr Opt 2008;5:254–261.
13. Kaparis K, Letchford AN. Separation algorithms for 0-1 knapsack polytopes. Math Program 2009. DOI: 10.1007/s10107-010-0359-5
14. Ferreira CE, Martin A, Weismantel R. Solving multiple knapsack problems by cutting planes. SIAM J Opt 1996;6:858–877.
15. Gu Z, Nemhauser GL, Savelsbergh MWP. Lifted cover inequalities for 0-1 integer programs: complexity. INFORMS J Comput 1999;11:117–123.
16. Klabjan D, Nemhauser GL, Tovey C. The complexity of cover inequality separation. Oper Res Lett 1998;23:35–40.
17. Gabrel V, Minoux M. A scheme for exact separation of extended cover inequalities and application to multidimensional knapsack problems. Oper Res Lett 2002;30:252–264.
18. Hoffman KL, Padberg MW. Improving LP-representations of zero-one linear programs for branch-and-cut. ORSA J Comput 1991;3:121–134.
19. Bektas T, Oguz O. On separating cover inequalities for the multidimensional knapsack problem. Comput Oper Res 2007;34: 1771–1776.
20. Kaparis K, Letchford AN. Local and global lifted cover inequalities for the multidimensional knapsack problem. Eur J Oper Res 2008;186:91–103.
21. Cattrysse D, Degraeve Z, Tistaert J. Solving the generalised assignment problem using polyhedral results. Eur J Oper Res 1996;108:618–628.
22. Gottlieb ES, Rao MR. The generalized assignment problem: valid inequalities and facets. Math Program 1990;46:31–52.
23. Aardal K. Capacitated facility location: separation algorithms and computational experience. Math Program 1998;81:149–175.

24. Aardal K, Pochet Y, Wolsey LA. Capacitated facility location: valid inequalities and facets. Math Oper Res 1995;20:562–582.
25. Nemhauser GL, Wolsey LA Integer and combinatorial optimization. New York: Wiley; 1988.
26. Chouman M, Crainic TG, Gendron B. A cutting plane algorithm for multicommodity capacitated fixed-charge network design. Technical Report. Montreal: CIRRELT; 2009.
27. Balas E. The prize collecting traveling salesman problem. Networks 1989;19:621–636.
28. Bérubé J-F, Gendreau M, Potvin J-Y. A branch-and-cut algorithm for the undirected prize collecting traveling salesman problem. Networks 2009;54:56–67.
29. Fischetti M, Salazar-Gonzalez JJ. Solving the orienteering problem through branch-and -cut. INFORMS J Comput 1998;10:133–148.
30. Hardin JR, Nemhauser GL, Savelsbergh MWP. Strong valid inequalities for the resource-constrained scheduling problem with uniform resource constraints. Discr Opt 2008;5:19–35.
31. Achterberg T, Koch T, Martin A. MIPLIB 2003. Oper Res Lett 2006;34:361–372.
32. Hunsaker B, Tovey C. Simple lifted cover inequalities and hard knapsack problems. Discr Opt 2005;2:219–228.
33. Atamtürk A. Cover and pack inequalities for (mixed) integer programming. Ann Oper Res 2005;139:21–38.
34. Weismantel R. Hilbert bases and the facets of special knapsack polytopes. Math Oper Res 1996;21:886–904.
35. Weismantel R. On the 0-1 knapsack polytope. Math Program 1997;77:49–68.
36. Avella P, Boccia M, Vasilyev I. A computational study of exact knapsack separation for the generalized assignment problem. Comput Optim Appl 2010;45:543–555.
37. Boyd EA. A pseudo-polynomial network flow formulation for exact knapsack separation. Networks 1992;22:503–514.
38. Boyd EA. Generating Fenchel cutting planes for knapsack polyhedra. SIAM J Opt 1993;3:734–750.
39. Boyd EA. Polyhedral results for the precedence-constrained knapsack problem. Discr Appl Math 1993;41:185–201.
40. Ceria S, Cordier C, Marchand H, *et al.* Cutting planes for integer programs with general integer variables. Math Program 1998; 81:201–214.
41. De Farias IR, Johnson EL, Nemhauser GL. Facets of the complementarity knapsack polytope. Math Oper Res 2002;27:210–226.
42. Fernández E, Jørnsten K. Partial cover and complete cover inequalities. Oper Res Lett 1994;15:19–33.
43. Glover F, Sherali HD. Second-order cover inequalities. Math Program 2008;114: 207–234.
44. Louveaux Q, Weismantel R. Polyhedral properties for the intersection of two knapsacks. Math Program 2008;113:15–38.
45. Marchand H, Wolsey LA. The 0-1 knapsack problem with a single continuous variable. Math Program 1999;85:15–33.
46. Nemhauser GL, Vance PH. Lifted cover facets of the 0-1 knapsack polytope with GUB constraints. Oper Res Lett 1994;16: 255–263.
47. Gu Z, Nemhauser GL, Savelsbergh MWP. Lifted flow cover inequalities for mixed 0-1 integer programs. Math Program 1999;85:439–467.
48. Padberg MW, Van Roy TJ, Wolsey LA. Valid linear inequalities for fixed charge problems. Oper Res 1985;33:842–861.
49. Van Roy TJ, Wolsey LA. Valid inequalities for mixed 0-1 programs. Discr Appl Math 1986;14:199–213.