Math. Program., Ser. A (2008) 114:207–234
DOI 10.1007/s10107-007-0098-4

FULL LENGTH PAPER

Second-order cover inequalities

Fred Glover · **Hanif D. Sherali**

Received: 20 October 2005 / Accepted: 13 September 2006 / Published online: 14 February 2007

Abstract We introduce a new class of *second-order cover inequalities* whose members are generally stronger than the classical knapsack cover inequalities that are commonly used to enhance the performance of branch-and-cut methods for 0–1 integer programming problems. These inequalities result by focusing attention on a single knapsack constraint in addition to an inequality that bounds the sum of all variables, or in general, that bounds a linear form containing only the coefficients 0, 1, and –1. We provide an algorithm that generates all non-dominated second-order cover inequalities, making use of theorems on dominance relationships to bypass the examination of many dominated alternatives. Furthermore, we derive conditions under which these non-dominated second-order cover inequalities would be facets of the convex hull of feasible solutions to the parent constraints, and demonstrate how they can be lifted otherwise. Numerical examples of applying the algorithm disclose its ability to generate valid inequalities that are sometimes significantly stronger than those derived from traditional knapsack covers. Our results can also be extended to incorporate multiple choice inequalities that limit sums over disjoint subsets of variables to be at most one.

Keywords Integer programming · Knapsack cover inequalities · 0–1 Pre-processing · Nested cuts · Surrogate constraints · Facets

F. Glover (✉)
University of Colorado, Boulder, CO 80309-0419, USA
e-mail: fred.glover@colorado.edu

H. D. Sherali
Grado Department of Industrial and Systems Engineering,
Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA
e-mail: hanifs@vt.edu

Mathematics Subject Classification 90C10 · 90C27

1 Introduction

The class of *knapsack cover inequalities* (or *cover cuts*) introduced by Balas [1], Hammer et al. [8], and Wolsey (1975) have enjoyed a well-deserved reputation for being useful to improve the solution of 0–1 integer programming (IP) problems, both in pre-processing and in tightening relaxations (see, e.g., [13,16]). In this paper, we introduce a class of *second-order cover* (SOC) *inequalities* whose members are generally stronger than the classical knapsack cover inequalities, based on a proposal of Glover [5] for generating inequalities by reference to the joint implications of a surrogate constraint and supplementary constraints involving bounds on nested sums of variables. The particular second-order cover inequalities for the case that we focus on in this paper, which also relate to the strengthened inequalities of Glover et al. [6], arise in the situation where a knapsack constraint is accompanied by a single additional constraint that bounds the sum of all variables.

Consider the following sets, defined respectively by a knapsack constraint and a supplementary constraint in terms of 0–1 variables $x_j, j \in N = \{1, \ldots, n\}$:

$$K = \left\{ x : \sum_{j \in N} a_j x_j \geq a_0 \right\} \tag{1}$$

$$S = \left\{ x : \sum_{j \in N} x_j \leq u \right\}. \tag{2}$$

Let

$$X = \{x \text{ binary} \colon x \in K \cap S\}.$$

The upper bound u is assumed to be a positive integer less than n, and the a_j-coefficients are real numbers, without restriction on their signs. Consequently, our results also apply to the case in which the supplementary bounding inequality might include coefficients of –1 as well. That is, the inequality defining S might originally arise from some constraint in 0–1 variables y_j in the form $\sum_{j \in N_1} y_j - \sum_{j \in N_2} y_j \leq u_0$, which we can cast in the form (2) [and adjust (1) accordingly] by the customary use of complementation, i.e., by setting $x_j = y_j, \forall j \in N_1$, and $x_j = 1 - y_j, \forall j \in N_2$, taking N to be the union of N_1 and N_2, and letting $u = u_0 + |N_2|$. As a special instance, our analysis also applies to the situation where the inequality defining S has the form $\sum_{j \in N} x_j \geq \ell[$, and by extension, includes the case where this is more generally replaced by

$$\ell \leq \sum_{j \in N} x_j \leq u. \tag{3}$$

As shown in Glover [4], the constraint (3) can be usefully employed in conjunction with (1) to force individual variables to receive a value of 0 or 1. The present work may be seen as a generalization that derives bounds on sums of variables, and not just on individual variables, building on the perspectives underlying the work of Glover [5] for exploiting nested inequalities. The new results differ from those on nested inequalities by characterizing the sets of variables over which non-dominated cuts can be generated, while at the same time identifying the strongest form of these cuts for the chosen sets. Based on this characterization and associated theorems on dominance implications, we design an algorithm that generates all non-dominated second-order cover inequalities, and illustrate how this procedure can be used to yield cuts that are stronger than knapsack cover cuts.

From a practical standpoint, the present work is additionally motivated by the finding of Vasquez and Vimont [20] that a strategy of imposing bounds on the sum of variables can improve the efficiency of solving multi-dimensional knapsack problems. Our results also have application in the context of the logic cuts of Hooker [11] and Hooker and Osorio [12], and more generally, in the setting of cutting planes generated and exploited in Osorio et al. [15]. Related areas of application are also identified in Hanafi [10] and in Spielberg and Guignard [18].

The remainder of this paper is organized as follows. In the next section, we introduce some relevant notation, derive the fundamental second-order cover (SOC) inequality, and discuss the basic concept of dominance. Section 3 discusses some preprocessing strategies, and Sect. 4 presents our main dominance theorem and designs routines for generating SOC inequalities and checking for non-dominance. Several additional dominance results are established in Sect. 5, which lays the groundwork for deriving the entire class of non-dominated SOC inequalities. Conditions under which such non-dominated SOC inequalities are facetial with respect to the convex hull of X [denoted conv(X)], and a technique for lifting these inequalities otherwise, are explored in Sect. 6. Finally, Sect. 7 closes with a discussion on connections with surrogate constraints and extensions to higher-order cover inequalities.

2 Second-order cover inequalities and non-dominance

For the sake of convenience in our derivation, let us assume without loss of generality throughout that

$$a_1 \geq a_2 \geq \cdots \geq a_n \tag{4a}$$

and that

$$X \neq \emptyset, \quad \text{that is} \sum_{\substack{j=1 \\ :a_j>0}}^{u} a_j \geq a_0. \tag{4b}$$

Let J be an arbitrary subset of N containing at most u elements, and denote its complement by $NJ \equiv N - J$. Given any J and NJ, define the subsets $J(h) \subseteq J$ and $NJ(h) \subseteq NJ$, depending on an index count h, as follows:

$$J(h) = \{\text{set of h smallest indices in J}\}, \quad \forall 0 \leq h \leq |J| \tag{5a}$$

$$NJ(h) = \{\text{set of } \min\{h, |NJ|\} \text{smallest indices in } NJ\}, \quad \forall 0 \leq h \leq u. \tag{5b}$$

Accordingly, define the corresponding sums of coefficients

$$S_J(h) = \sum_{j \in J(h): a_j > 0} a_j, \ \ \forall 0 \leq h \leq |J|, \quad \text{and} \quad S_{NJ}(h) = \sum_{j \in NJ(h): a_j > 0} a_j, \ \ \forall 0 \leq h \leq u, \tag{6}$$

where these sums are taken to be zeros if the associated sets are empty.

Proposition 1 *Consider any nonempty $J \subseteq N$, and let $p \in [0, min\{u, |J|\}]$ be the smallest integer such that $S_J(p) + S_{NJ}(u - p) \geq a_0$. Then*

$$\sum_{j \in J} x_j \geq p \tag{7}$$

is a valid second-order cover (SOC) inequality implied by X.

Proof Note that from (4), (5), and (6), we have that the solution to $\min\{\sum_{j \in J} x_j : x \in X\}$ will be realized by finding the smallest $p \in [0, min\{u, |J|\}]$ such that the sum of the p largest (positive) coefficients a_j for $j \in J$, plus the sum of the up to $(u - p)$ largest (positive) coefficients a_j for $j \in NJ$ is at least a_0, i.e., $S_J(p) + S_{NJ}(u - p) \geq a_0$. Hence, we have,

$$\min\left\{\sum_{j \in J} x_j : x \in X\right\} = p, \tag{8}$$

which implies the validity of (7). □

Corollary 1 *The second-order cover inequality (7) implies the following under $x \in S$:*

$$\sum_{j \in NJ} x_j \leq u - p. \tag{9}$$

Proof Follows directly form the inequalities in (2) and (7). □

An obvious implication of Corollary 1 is that if (7) is valid with $p = u$ for some J, then (9) directly yields $x_j = 0, \forall j \in NJ$, i.e., these variables can be eliminated from the problem.

Now, consider a pair of valid SOC inequalities of type (7) given by

$$\sum_{j\in J} x_j \geq p, \quad \text{where } \min\left\{\sum_{j\in J} x_j : x \in X\right\} = p \tag{10a}$$

and

$$\sum_{j\in J'} x_j \geq p', \quad \text{where } \min\left\{\sum_{j\in J'} x_j : x \in X\right\} = p'. \tag{10b}$$

We say that (10a) *dominates* (10b) *over the unit hypercube* $H = \{x : 0 \leq x \leq e\}$, where e is a vector of ones, if (10b) is implied by (10a) over H, i.e.,

$$\min\left\{\sum_{j\in J'} x_j : \sum_{j\in J} x_j \geq p,\ 0 \leq x \leq e\right\} \geq p'. \tag{11a}$$

Observe from (10a, b) that whenever (11a) holds true, we have

$$p' = \min\left\{\sum_{j\in J'} x_j : x \in X\right\} \geq \min\left\{\sum_{j\in J'} x_j : \sum_{j\in J} x_j \geq p, 0 \leq x \leq e\right\} \geq p',$$

that is, equality holds true throughout. Hence, equivalently, (10a) dominates (10b) over H if and only if

$$\min\left\{\sum_{j\in J'} x_j : \sum_{j\in J} x_j \geq p, 0 \leq x \leq e\right\} = p'. \tag{11b}$$

Proposition 2 *Consider the pair of SOC inequalities* (10a) *and* (10b)*, and suppose that* $|J - J'| = r$*. Then* (10a) *dominates* (10b) *over* H *if and only if* $p' = \max\{0, p - r\}$*. In particular, if* $p' \geq 1$*, then this happens if and only if* $p = p' + r$*.*

Proof Observe that the problem on the left-hand side of (11b) is solved by setting $x_j = 1, \forall j \in J - J'$, and then setting $x_j = 1$ for some $\max\{0, p - r\}$ indices $j \in J \cap J'$, and $x_j = 0$ otherwise. Hence, the optimal objective function value of this problem equals $\max\{0, p - r\}$. Therefore, from (11b), the SOC inequality (10a) dominates (10b) over H if and only if $p' = \max\{0, p - r\}$. Moreover, if $p' \geq 1$, then this occurs if and only if $p' = p - r$, i.e., $p = p' + r$. □

In other words, an SOC inequality $\sum_{j\in J'} x_j \geq p'$ with $p' \geq 1$ would be *non-dominated* (ND) by the viewpoint of Proposition 2 if we cannot construct a $J \subseteq N$ that has some r additional elements than J' does, and, say, has some r'

elements removed from J', and yet we have that $\sum_{j\in J} x_j \geq p \equiv p' + r$ is a valid SOC inequality, where at least one of $r \geq 1$ and $r' \geq 1$ holds true. In fact, as we show next, there is an equivalent characterization of non-dominance in terms of a simpler, local non-dominance property. Specifically, we will say that (10a) *locally dominates* (10b) over H if either one of the following conditions holds true:

$$\text{(i)} \quad J \subset J' \text{ and } p = p' \geq 1 (\text{nontrivial case of } r = 0 \text{ and } r' \geq 1) \tag{12a}$$

$$\text{(ii)} \quad J = J' \cup \{j\} \text{for some } j \notin J', \text{ and } p = p' + 1 (\text{case of } r = 1 \text{ and } r' = 0). \tag{12b}$$

Moreover, we will say that an SOC inequality $\sum_{j\in J'} x_j \geq p'$ is *locally non-dominated* (**LND**) if $p' \geq 1$ and there does not exist a $J \subseteq N$ that locally dominates it.

Now, consider the following result.

Proposition 3 *Consider an SOC inequality* (10b) *having* $p' \geq 1$. *Then this is LND if and only if it is ND.*

Proof If the given SOC inequality (10b) is ND, then it is obviously LND. Hence, suppose that (10b) is LND and let us show that it is ND. On the contrary, suppose that there exists an SOC inequality (10a) based on a set $J \subseteq N$, with $|J - J'| = r$ and $|J' - J| = r'$, where at least one of $r \geq 1$ and $r' \geq 1$ holds true, and where $p = p' + r$ (see Proposition 2). Hence, we have

$$\mathbf{P1}: \text{Min} \left\{ \sum_{j\in J} x_j : x \in X \right\} = p = p' + r.$$

For convenience, denote $v(\text{P})$ as the optimal objective value for any given problem P (so $v(\text{P1}) = p = p' + r$ above), and let $J_+ \equiv J - J', J'_+ \equiv J' - J$, and $J'' \equiv J \cap J'$. Consider the following two cases.

Case (i) $\mathbf{r'} \geq \mathbf{1}$ Define the problem

$$\mathbf{P2}: \text{Min} \left\{ \sum_{j\in J''} x_j : x \in X \right\},$$

and suppose that x^* solves Problem P2. Note that we must have $v(\text{P2}) \geq p'$, because otherwise, if $v(\text{P2}) < p'$, then since x^* is feasible to P1 and $J = J'' \cup J_+$ with $|J_+| = r, x^*$ would yield an objective value less than $p' + r = p$, contradicting that $v(\text{P1}) = p$. Hence, $\sum_{j\in J''} x_j \geq p'$ is a valid inequality with $J'' \subset J'$, contradicting the LND Condition (12a).

Case (ii) $\mathbf{r'} = \mathbf{0}$ In this case, if $r = 1$, then we have a direct contradiction to the LND Condition (12b); hence, suppose that $r \geq 2$. Select any $k \in J_+$ and consider the problem

$$\textbf{P3}\text{: Min}\left\{\sum_{j\in J'} x_j + x_k : x \in X\right\}.$$

To complete the proof, let us show that $v(\text{P3}) = p' + 1$, which would mean that $\sum_{j\in J'} x_j + x_k \geq p' + 1$ is a valid SOC inequality that locally dominates (10b) via Condition (12b), contradicting that (10b) is LND. Observe from (10b) that if $v(\text{P3}) \neq p' + 1$, then we have that $v(\text{P3}) = p'$ and that there exists an optimum x^* to Problem P3 having $x_k^* = 0$. But again, this solution x^* would be feasible to P1 and yield an objective value lesser than $p' + r = p = v(\text{P1})$, a contradiction. □

Our focus in this paper will be on characterizing and deriving the entire class of non-dominated SOC inequalities via the equivalent criteria (12a, b) underlying the LND second-order cover inequalities. To emphasize our reliance on (12a, b), we shall refer to these SOC inequalities as LND (rather than ND) inequalities.

Henceforth, to ease notation, we will denote the sets $J \cup \{j\}$ for any $j \notin J$, and $J - \{j\}$, for any $j \in J$, simply as $J + j$ and $J - j$, respectively.

Proposition 4 *Consider an SOC inequality* (7) *of the form* $\sum_{j\in J} x_j \geq p$, *and suppose that* $a_{\hat{j}} \leq 0$ *for some* $\hat{j} \in J$. *Then this inequality is dominated by the valid inequality* $\sum_{j\in J-\hat{j}} x_j \geq p$.

Proof By the condition $a_{\hat{j}} \leq 0$ and the validity of (7), we have from (8) that $x_{\hat{j}}^* = 0$ in an optimal solution x^* to the problem $\min\{\sum_{j\in J} x_j : x \in X\}$, where the optimal objective value equals p. But because $a_{\hat{j}} \leq 0$, we also have that $\min\{\sum_{j\in J-\hat{j}} x_j : x \in X\} = p$, or that $\sum_{j\in J-\hat{j}} x_j \geq p$ is valid, which by (12a), dominates (7). □

Proposition 4 asserts that in determining (locally) non-dominated second-order cover inequalities (7), we can simply focus on the positive coefficient indices for composing J. In fact, suppressing all nonpositive coefficient indices from S, we get a set that is implied by X and we can derive valid inequalities (7) for this set, which would then be valid for X as well. The nonpositive coefficient indices could then be accommodated in NJ for each such J determined for (7), in order to compose the complement inequality (9) as necessary. Therefore, noting (4), we will henceforth assume that

$$n > u \geq 1, \quad a_0 \geq a_1 \geq a_2 \geq \cdots \geq a_n > 0, \quad \text{and that} \sum_{j=1}^{u} a_j \geq a_0, \tag{13}$$

where observe that in the inequality $\sum_{j\in N} a_j x_j \geq a_0$, if $a_j > a_0$ for any $j \in N$, we can perform a standard coefficient-reduction and validly tighten this knapsack inequality by making $a_j = a_0$; hence, the assumption $a_0 \geq a_j$, $\forall j \in N$, in (13).

3 Preprocessing routines

In addition to the pre-processing that led to (13), we can further fix some variables at values 1 or 0 as implied by $x \in X$, thereby eliminating these variables, or, in effect, forcing variables to J or NJ, respectively, in composing non-dominated inequalities.

Proposition 5 *Let* $S_N(u+1) \equiv \sum_{j=1}^{u+1} a_j$. *If* $a_{\hat{j}} > S_N(u+1) - a_0$ *for any* $\hat{j} \in \{1,\ldots,u\}$, *then* $x \in X \Rightarrow x_{\hat{j}} = 1$.

Proof If any such $x_{\hat{j}} = 0$, then the sum of the remaining u largest a_j-coefficients equals $S_N(u+1) - a_{\hat{j}} < a_0$, which contradicts feasibility to X. □

Proposition 6 *Let* $S_N(u-1) \equiv \sum_{j=1}^{u-1} a_j$. *If* $a_{\hat{j}} < a_0 - S_N(u-1)$ *for any* $\hat{j} \in \{u+1,\ldots,n\}$, *then* $x \in X \Rightarrow x_{\hat{j}} = 0$.

Proof If any such $x_{\hat{j}} = 1$, then $a_{\hat{j}}$ plus the remaining $(u-1)$ largest a_j-coefficients sum to $a_{\hat{j}} + S_N(u-1) < a_0$, which contradicts feasibility to X. □

Remark 1 Naturally, for any $\hat{j} \in N$ of the type identified by Propositions 5 and 6, we should simply fix the corresponding $x_{\hat{j}}$ to 1 or 0, respectively, and eliminate it from the problem under consideration. On the other hand, if we do not eliminate such indices from the problem, then any non-dominated SOC inequality (7) must include $\hat{j}$ in J for a $\hat{j}$ of the type identified by Proposition 5, and must exclude $\hat{j}$ from J for a $\hat{j}$ of the type identified by Proposition 6. To see this, suppose that $\hat{j}$ satisfies the condition of Proposition 5, but that $\hat{j} \notin J$ for a valid SOC inequality (7). Hence, by (8), we have $\min\{\sum_{j\in J} x_j : x \in X\} = p$, but since $x \in X \Rightarrow x_{\hat{j}} = 1$, we also have that $\min\{\sum_{j\in J+\hat{j}} x_j : x \in X\} = p+1$, or that (7) is dominated by $\sum_{j\in J+\hat{j}} x_j \geq p+1$ according to (12b). Likewise, if $\hat{j}$ satisfies the condition of Proposition 6 but $\hat{j} \in J$ in a valid inequality (7), then we also have that $\sum_{j\in J-\hat{j}} x_j \geq p$ is valid, which dominates (7) by (12a). Therefore, we will henceforth assume that we have fixed and eliminated variables from the problem according to Propositions 4 and 5, and that (13) holds true for the remaining set of variables, appropriately re-indexed. □

Example 1 Consider the following constraints of type (1) and (2):

$$13x_1 + 9x_2 + 6x_3 + 5x_4 + 5x_5 + 4x_6 + 4x_7 + 3x_8 + 3x_9 + 3x_{10} \geq 27 \quad (14a)$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} \leq 3. \quad (14b)$$

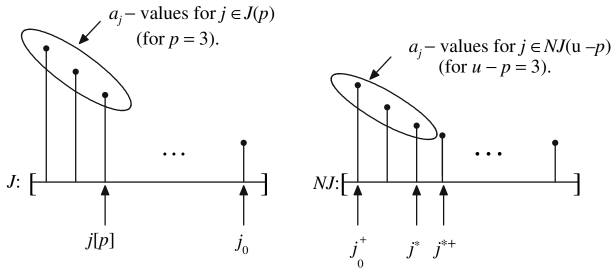


Fig. 1 Illustration of indices defined by CUT(J)

Note that $n = 10$, $u = 3$, and that $S_N(u+1) - a_0 = 33 - 27 = 6$, and $a_0 - S_N(u-1) = 27 - 22 = 5$. Hence, by Proposition 5, we can fix $x_1 = x_2 = 1$, and by Proposition 6, we can fix $x_6 = x_7 = x_8 = x_9 = x_{10} = 0$. This reduces (14a) to $6x_3 + 5x_4 + 5x_5 \geq 5$, which by coefficient-reduction [see (13)], results in $5x_3 + 5x_4 + 5x_5 \geq 5$, or that $x_3 + x_4 + x_5 \geq 1$. Moreover, since (14b) reduces to $x_3 + x_4 + x_5 \leq 1$, the set X in this case collapses to the simple restriction $x_3 + x_4 + x_5 = 1$, in the remaining binary variables (x_3, x_4, x_5).

4 Generating SOC inequalities and checking for non-dominance

Let X be defined by (1) and (2), where (13) holds true, but none of the conditions specified in Propositions 5 and 6 are satisfied. Let us abbreviate this statement as Assumption A, and treat this as a standing assumption throughout the remainder of this paper. Consider any $J \subseteq N$ where $J \neq \emptyset$. The following routine generates an SOC inequality of the type (7) predicated on the set J, and based on (8). Its operation follows the thought-process in the proof of Proposition 1.

Routine CUT(J) given $J \subseteq N$, $J \neq \emptyset$
Initialization: Set $p = 0$, $\Sigma = S_{NJ}(u)$ [see Eq. (6)].
Step 1 If $\Sigma \geq a_0$, go to Output. Else, increment p by 1.
Step 2 Update $\Sigma \equiv S_J(p) + S_{NJ}(u-p)$ and return to Step 1.
Output: CUT(J) produces the value $0 \leq p \leq u$, along with the following indices (see Fig. 1 for a conceptual illustration), where each index below is taken as 0 if undefined:

$$j[p] = p\text{th smallest index in } J \tag{15a}$$

$$j_0 = \text{largest index in } J (\text{note: } j_0 > j[p], \text{ else, we can fix } x_j \equiv 1, \forall j \in J) \tag{15b}$$

$$j_0^+ = \text{ smallest index in } NJ \tag{15c}$$

$$j^* = \text{ largest index in } NJ(u-p) \tag{15d}$$

$$j^{*+} = \text{ smallest index in } NJ \text{ that exceeds } j^*. \tag{15e}$$

Note that (15b) and (15c) are characteristics of the set J itself, but are included in the output of CUT(J) for convenience in discussion. Furthermore, note that while any $J \subseteq N$ produces a unique SOC inequality (7) via the optimal value of Problem (8), the procedure CUT(J) identifies a particular optimal solution (among possible alternative optimal solutions) to Problem (8), and the indices (15) (see also Fig. 1) correspond to this specified solution. Henceforth, we assume that whenever a $J \subseteq N$ produces an SOC inequality (7), the corresponding optimal solution to (8) is identified as the particular solution produced by CUT(J) and recorded in the definition of the indices in (15). The following proposition lays the foundation of our dominance results.

Proposition 7 *Suppose that a nonempty $J \subseteq N$ produces an SOC inequality* (7) *with $p \geq 1$, and let $j[p]$, j_0, j_0^+, j^*, and j^{*+} be as defined in* (15)*. Then* (7) *is LND if and only if the following two conditions hold true*:

(a) *The set $J' \equiv J - j_0$ produces an SOC inequality having $p' = p - 1$.*
(b) *The set $J' \equiv J + j_0^+$ produces an SOC inequality having $p' = p$.*

Moreover, if (7) is LND (with $p \geq 1$), then $a_{j_0} > a_{j^{*+}}$ and $a_{j[p]} > a_{j_0^+}$ (where $a_{\hat{j}} \equiv 0$ if any $\hat{j} = 0$).

Proof Suppose that (7) is LND. If $J' \equiv J - j_0$ yields an SOC inequality with $p' = p$, then (7) would be locally dominated by (12a). Hence, since we cannot have $p' > p$ nor $p' \leq p - 2$ in this case, we must have $p' = p - 1$. Likewise, if $J' \equiv J + j_0^+$ yields an SOC inequality with $p' = p + 1$ (we must have $p+1 \geq p' \geq p$), then (7) would be locally dominated by (12b). Therefore, $p' = p$ in this case. Hence, Conditions (a) and (b) of the proposition hold true.

Conversely, suppose that Conditions (a) and (b) are satisfied. Let us show that neither (12a) nor (12b) can hold true, i.e., we cannot find a $J_0 \subseteq N$ with an accompanying p_0 for the corresponding SOC inequality such that

$$J_0 \subset J \quad \text{and} \quad p_0 = p \tag{16a}$$

or

$$J_0 = J + j \quad \text{for some } j \notin J \quad \text{and} \quad p_0 = p + 1. \tag{16b}$$

Note that for any $J_0 \subset J$, by the definition of j_0 in (15b), we have from (8) that, $p_0 \equiv \min\{\sum_{j \in J_0} x_j : x \in X\} \leq \min\{\sum_{j \in J - j_0} x_j : x \in X\} = (p - 1)$ by Condition (a) of the proposition, and so, (16a) cannot hold true. Similarly, (16b) cannot be satisfied, because otherwise, if there exists such a J_0 and p_0, then noting that $a_{j_0^+} \geq a_j$, $\forall j \in NJ$, we have, $(p + 1) = \min\{\sum_{j \in J+j} x_j : x \in X\} \leq \min\{\sum_{j \in J + j_0^+} x_j : x \in X\} = p$ by Condition (b) of the proposition, which is a contradiction.

Moreover, suppose that (7) is LND (with $p \geq 1$) and that x^* solves (8). Since $j_0 > j[p]$ exists by Assumption A (else we could have fixed all $x_j = 1$ for $j \in J$), in case $j^{*+} > 0$ (the condition $a_{j_0} > a_{j^{*+}} \equiv 0$ is trivial if $j^{*+} = 0$), we must have

$a_{j_0} > a_{j^{*+}}$, because otherwise, if $a_{j_0} \le a_{j^{*+}}$, then x^* would remain as an optimal solution to the problem $\min\{\sum_{J-j_0} x_j : x \in X\}$ with objective value p. Hence, $\sum_{j\in J-j_0} x_j \ge p$ would be valid and locally dominate (7), a contradiction.

Finally, let us establish that if (7) is LND (with $p \ge 1$), then $a_{j[p]} > a_{j_0^+}$. Suppose that $j_0^+ > 0$ (else the result is trivial), and that $a_{j_0^+} \ge a_{j[p]}$. Define $\Sigma \equiv S_J(p) + S_{NJ}(u-p) \ge a_0$, so that, since J yields an SOC inequality having $p \ge 1$, we must have

$$\Sigma + a_{j^{*+}} - a_{j[p]} < a_0, \tag{17}$$

else, the set J would yield an SOC inequality having a smaller p-value. Furthermore, since (7) is LND, then by Condition (b) of the proposition and assuming that $a_{j_0^+} \ge a_{j[p]}$, and noting that Σ then includes $a_{j_0^+}$ (else we can reduce p), we get

$$\Sigma - a_{j[p]} + a_{j^{*+}} \ge a_0, \tag{18}$$

which contradicts (17). □

Corollary 2 *Let $K_{\max} \ge 1$ be the largest index in N for which $a_{K_{\max}} = a_1$. Then $\{1,\ldots,K_{\max}\} \subseteq J$ for all LND SOC inequalities (7) having $p \ge 1$.*

Proof On the contrary, suppose that $\sum_{j\in J} x_j \ge p \ge 1$ is an LND SOC inequality but that there exists a $\hat{j} = \min\{1 \le j \le K_{\max} : j \notin J\}$. By definition then, we have $j_0^+ \equiv \hat{j}$. But this yields $a_{j_0^+} = a_1 \ge a_{j[p]}$, which contradicts the last assertion of Proposition 7. □

In the following section, we will use the characterizations provided by Proposition 7 and Corollary 2 to derive additional dominance results and to help construct the set of LND second-order cover inequalities. We close this section with the statement of a routine LND(J) that checks the non-dominance of (7) produced by CUT(J), returning LND(J) = TRUE if (7) is LND and LND(J) = FALSE, otherwise. This routine is directly based on checking the conditions of Proposition 7.

Local non-dominance routine LND(J) given the output of CUT(J), for $J \subseteq N$, $J \ne \emptyset$

Initialization Given p, j_0, j_0^+, and j^* from the output of CUT(J), let $\Sigma \equiv S_J(p) + S_{NJ}(u-p)$. If $p = 0$, return FALSE.

Step 1 If $\Sigma + a_{j_0} - a_{j[p]} \ge a_0$, proceed to Step 2. Else, return FALSE (Condition (a) of Proposition 7 is violated).

Step 2 If $a_{j[p]} > a_{j_0^+}$, proceed to Step 3. Else, return FALSE (the final condition in Proposition 7 is violated).

Step 3 If (the updated value of Σ given by) $S_{J+j_0^+}(p) + S_{NJ-j_0^+}(u-p) \ge a_0$, then return TRUE. Else, return FALSE [Condition (b) of Proposition 7 is violated].

5 Generating the set of LND second-order cover inequalities

Consider the development of a binary tree to conduct an implicit enumeration of the potential sets $J \subseteq N$, based on the dichotomy that $j \in J$ or $j \in NJ$, and where the branching decisions are made in the order of the indices $1, 2, \ldots, n$. Following the proposal of Glover [4] (also see [3]), we shall explore this tree in a depth-first fashion by maintaining a partial solution list PS that contains the signed index $+j$ if j is restricted to lie in J at the current node of the enumeration tree, $-j$ if j is restricted to lie in NJ, and where these indices are underlined as $\underline{+j}$ or $\underline{-j}$ in case the brother node has already been previously explored. Note that by the branching order considered, if $|PS| = k$, then PS contains the indices $1, \ldots, k$ in this order, with possibly $\pm$ signs and with elements underlined or not. The indices not present in PS are currently unassigned to either of the sets J or NJ. Furthermore, the backtracking process upon *fathoming PS* amounts to identifying the right-most non-underlined element in PS, complementing the sign on this index and underlining it, and deleting all the (underlined) elements to the right of it. By Corollary 2, we shall initialize PS as

$$PS = \{\underline{1}, \ldots, \underline{K_{\max}}\}, \tag{19}$$

and we shall terminate the process whenever $PS = \emptyset$ upon some fathoming process. Note that since $K_{\max} \geq 1$, we always have $J \neq \emptyset$ in any partial solution implied by PS because of (19). Furthermore, given that PS contains indices $\pm j$ for $j = 1, \ldots, k$ (by this notation, we include the underlined signed indices as well), when we *increment PS* by the next index $j_{\text{next}} \equiv k + 1$, we shall do so as $PS \leftarrow PS \cup \{-j_{\text{next}}\}$, i.e., we will first include j_{next} in NJ. Moreoever, by a *completion of PS* that is based on the indices $\{1, \ldots, k\}$, we will mean the assignment of $\pm j$ for all the remaining indices $j = k + 1, \ldots, n$ to PS.

Now, suppose that we have a partial solution list PS based on the indices $\{1, \ldots, k\}$ that induces a set J and NJ^k, defined as $NJ^k \equiv \{1, \ldots, k\} - J$, where

$$k < n, J \neq \emptyset \quad \text{and} \quad \sum_{j=1}^{k} a_j \geq a_0. \tag{20}$$

Define the routine $\widehat{CUT}(PS)$ to be the routine CUT(J) described in Sect. 4 based on the indices $\{1, \ldots, k\}$, i.e., using the sets J and NJ^k. (We analogously define $NJ^k(h)$ and $S_{NJ^k}(h)$ as in (5b) and (6), respectively, with respect to the set NJ^k.) Consider the following result that prescribes a completion to PS for the resulting inequality (7) to be LND.

Proposition 8 *Given a partial solution PS based on the indices* $\{1, \ldots, k\}$ *and with induced sets J and* NJ^k *such that* (20) *holds true, suppose that the routine* $\widehat{CUT}(PS)$ *produces a p and* j^* *such that* j^{*+} *exists* (*i.e.,* $j^{*+} \neq 0$). *Then, in any possible LND SOC inequality arising from a completion of PS, we must have* $NJ = \{1, \ldots, n\} - J$, *and yielding the same value of p.*

Proof Let $J(p)$ and $NJ^k(u-p)$ be as identified by $\widehat{CUT}(PS)$, and define x^* as $x^*_j = 1$, $\forall j \in J(p) \cup NJ^k(u-p), x^*_j = 0$, $\forall j \in N$, otherwise. Then x^* is an optimal solution to the problem

$$\min\left\{\sum_{j\in J} x_j : x \in X, x_j = 0, \forall j > k\right\} \tag{21}$$

with objective value p. Note that the inequality

$$\sum_{j\in J} x_j \geq p \tag{22}$$

(with the same value of p) is a valid SOC inequality that is derived by the same solution x^* to the problem $\min\{\sum_{j\in J} x_j : x \in X\}$ since $a_j \leq a_{j^{*+}}$ for $j = k+1,\ldots,n$, and $x^*_{j^{*+}} = 0$ because $j^{*+} > j^*$ exists. Moreover, if we put any subset of the indices in $\{k+1,\ldots,n\}$ into J to get J', the same solution x^* would evaluate $\min\{\sum_{j\in J'} x_j : x \in X\}$ because $a_j \leq a_{j_0}$, $\forall j > k$. But then, the resultant inequality $\sum_{j\in J'} x_j \geq p$ would be (locally) dominated by (22). Hence, any LND inequality arising from a completion of PS must include all the remaining indices $k+1,\ldots, n$ in NJ. □

Proposition 8 tells us that as we build J and its complement while considering indices in the order 1, 2,..., the moment we discover for a partial solution PS that $\widehat{CUT}(PS)$ yields an index $j^{*+} > j^*$, we can include all the remaining indices in NJ, check for non-dominance, and fathom the given PS. The following proposition refines this result somewhat further and permits an earlier fathoming of PS without a non-dominance check.

Proposition 9 *Given a partial solution PS based on the indices* $\{1,\ldots, k\}$ *and with induced sets J and* NJ^k *such that* (20) *holds true, suppose that the routine* $\widehat{CUT}(PS)$ *produces a* $p \geq 1$ *and* $j^* \geq 0$*, along with* j_0 *and* $j_0^+ \geq 0$*. Let* $j_{next} = k+1$*, and tentatively consider* $PS' = PS \cup \{-j_{next}\}$*. If* $\widehat{CUT}(PS')$ *produces the same value of the index* j^**, and if* $a_{j_0} \leq a_{j_{next}}$*, or* $a_{j[p]} \leq a_{j_0^+}$*, then we can fathom PS in that no completion to it can lead to an LND SOC inequality.*

Proof For the partial solution PS', since $j^{*+} \equiv j_{\text{next}}$ exists by the statement of the proposition, then by Proposition 8, any possible LND cut arising from a completion to PS' must include all the remaining indices within NJ. However, by hypothesis, since either $a_{j_0} \leq a_{j_{\text{next}}} = a_{j^{*+}}$, or $a_{j[p]} \leq a_{j_0^+}$ holds true, the final part of Proposition 7 asserts that the resulting inequality would not be LND. Hence, we can fathom PS' and examine the resulting partial solution $PS'' = PS \cup \{j_{\text{next}}\}$, which adds j_{next} to J instead. If $j_{\text{next}} = n$, then since we know that $x^*_{j_{\text{next}}} = 0$ in an optimal solution x^* to the problem $\min\{\sum_{j\in J-j_{\text{next}}} x_j : x \in X\}$, which has objective value equal to p, the same solution x^* remains optimal for $\min\{\sum_{j\in J} x_j : x \in X\}$ with objective value p, and so, $\sum_{j\in J} x_j \geq p$ would be locally dominated by $\sum_{j\in J-j_{\text{next}}} x_j \geq p$. On the other hand, if $j_{\text{next}} < n$, then the same outcome of the result would be obtained with respect to the revised $j_{\text{next}} = k + 2$, again leading to a fathoming as above. In essence, therefore, we can fathom $PS'' \equiv PS \cup \{\underline{j_{\text{next}}}\}$ as well, which is equivalent to fathoming PS.

□

Propositions 8 and 9 prompt the following strategy. Given a partial solution PS based on the indices $\{1, \ldots, k\}$ and with induced sets J and NJ^k such that (20) holds true, suppose that $\widehat{CUT}(PS)$ produces a $p \geq 1$ and $j^* \geq 0$, along with j_0 and $j_0^+ \geq 0$. Let $j_{\text{next}} \equiv k + 1$. Then define **TEST(*PS*)** to return TRUE if $\widehat{CUT}(PS \cup \{-j_{\text{next}}\})$ produces the same value of the index j^*, and FALSE otherwise. Accordingly, in the case that TEST(PS) returns TRUE, then if either $a_{j_0} \leq a_{j_{\text{next}}}$ or $a_{j[p]} \leq a_{j_0^+}$ holds true, we fathom PS (by Proposition 9), and otherwise, using Proposition 8, we increment $PS \leftarrow PS \cup \{-j_{\text{next}}\}$, check the potential LND status of the cut $\sum_{j\in J} x_j \geq p$ based on J and $NJ \equiv N - J$, and then fathom PS.

A flow-chart for generating all LND second-order cover inequalities is given in Fig. 2 based on Propositions 7 (including Corollary 2), 8, and 9, and under Assumption A based on Propositions 5 and 6, and Remark 1.

Example 2 Consider the following constraints of type (1) and (2):

$$13x_1 + 12x_2 + 9x_3 + 7x_4 + 5x_5 + 4x_6 + 3x_7 + 2x_8 + 2x_9 + 2x_{10} \geq 25 \quad (23a)$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} \leq 3. \quad (23b)$$

Here, $n = 10$, $u = 3$, and it may be readily verified that Assumption A holds true. Note that there are 1,023 possible sets $J \subseteq N$ for this example. The algorithm in Fig. 2 fathomed a majority of these sets, invoking the non-dominance routine LND for 58 sets, and generated the following six LND SOC inequalities based on the corresponding then-current partial solution list PS identified below.

$$PS = \{\underline{1}, -2, \underline{3}, -4\} \quad \text{yielding} \quad x_1 + x_3 \geq 1 \quad (24a)$$

$$PS = \{\underline{1}, \underline{2}, -3, -4, -5\} \quad \text{yielding} \quad x_1 + x_2 \geq 1 \quad (24b)$$

$$PS = \{\underline{1}, \underline{2}, \underline{3}, -4, \underline{5}, -6\} \quad \text{yielding} \quad x_1 + x_2 + x_3 + x_5 \geq 2 \quad (24c)$$

$$PS = \{\underline{1}, \underline{2}, \underline{3}, \underline{4}, -5, -6\} \quad \text{yielding} \quad x_1 + x_2 + x_3 + x_4 \geq \quad (24d)$$

$$PS = \{\underline{1}, -2, -3, \underline{4}, \underline{5}, \underline{6}, -7\} \quad \text{yielding} \quad x_1 + x_4 + x_5 + x_6 \geq 1 \quad (24e)$$

$$PS = \{\underline{1}, \underline{2}, -3, \underline{4}, \underline{5}, \underline{6}, \underline{7}, -8\} \quad \text{yielding} \quad x_1 + x_2 + x_4 + x_5 + x_6 + x_7 \geq 2. \quad (24f)$$

To illustrate the algorithmic procedure, consider a stage when we have just fathomed a partial solution to obtain the revised list $PS = \{\underline{1}, \underline{2}, -3, \underline{4}, \underline{5}, \underline{6}, \underline{7}\}$, so that $j_0 = 7$, $j_0^+ = 3$, $j_{\text{next}} = 8$, and $\Sigma = 53 > a_0$. Applying $\widehat{CUT}(PS)$

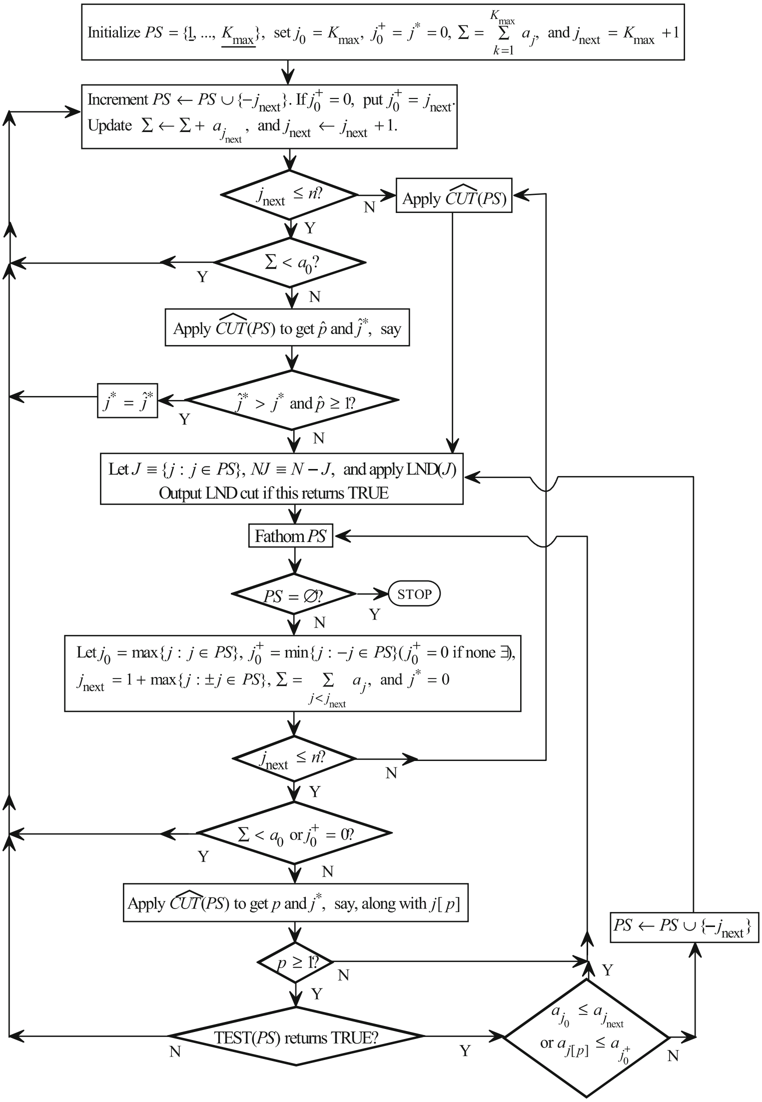


Fig. 2 Flow-chart for generating all LND second-order cover inequalities

with $k = 7$ we get $p = 2$, with $J(p) = \{1, 2\}$, $NJ^k(u - p) = \{3\}$, yielding $j^* = 3$ and $j[p] = 2$. Since $p \geq 1$ (see Fig. 2), we now apply TEST(PS). Since $\widehat{CUT}(PS \cup \{-8\})$ reproduces $j^* = 3$, TEST(PS) returns TRUE. Since $a_{j_0} = 3 > a_{j_{\text{next}}} = 2$ and $a_{j[p]} = 12 > a_{j_0^+} = 9$, we increment $PS \leftarrow PS \cup \{-8\}$ and apply LND(J) with $J = \{1, 2, 4, 5, 6, 7\}$, $NJ - \{3, 8, 9, 10\}$, and $p = 2$. Note that $\Sigma \equiv S_J(p) + S_{NJ}(u - p) = 13 + 12 + 9 = 34$. Since $\Sigma + a_{j_0} - a_{j[p]} = 25 \geq a_0$, and $S_{J+j_0^+}(p) + S_{NJ-j_0^+}(u - p) = 13 + 12 + 2 = 27 \geq a_0$, LND($J$) returns TRUE and produces the LND SOC inequality (24f). We now fathom PS to produce $PS = \{\underline{1}, \underline{2}, -3, \underline{4}, \underline{5}, \underline{6}, \underline{7}, \underline{8}\}$. This time, with $j_{\text{next}} = 9$, TEST(PS) again returns TRUE because $\widehat{CUT}(PS \cup \{-9\})$ reproduces $j^* = 3$, but now, $a_{j_0} \equiv a_8 = 2 = a_9 \equiv a_{j_{\text{next}}}$. Hence, we fathom PS, yielding $PS = \{\underline{1}, \underline{2}, \underline{3}\}$, and we continue the algorithmic process.

We can compare the LND SOC inequalities derived above to analogous knapsack cover inequalities. It is easy to demonstrate that the SOC inequality (7) with $p \geq 1$ will dominate a knapsack cover inequality defined on the same set J if and only if

$$S_J(p-1) + \sum_{j \in NJ} a_j \geq a_0$$

because then, the knapsack cover inequality induced by (1) would be $\sum_{j \in J} x_j \geq p'$ with $p' \leq p - 1$. Checking this condition shows that each of the SOC inequalities of the preceding example dominates the corresponding knapsack cover inequality. If we examine just the minimal cover knapsack inequalities (see [14]) as a basis for comparison, we see for example that the minimal cover $x_1 + x_2 + x_3 + x_4 \geq 1$ is strictly dominated by the SOC inequality (24d) given by $x_1 + x_2 + x_3 + x_4 \geq 2$. Even the non-dominated knapsack cover inequality $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \geq 2$ is strictly dominated by the SOC inequality (24c) given by $x_1 + x_2 + x_3 + x_5 \geq 2$. It is interesting to observe that if the knapsack constraint (1) for this example is expanded to contain additional variables having coefficients of 2, the SOC inequalities (24a–f) will not change, but all the classical knapsack cover inequalities will be weakened. For example, if two additional variables having coefficients of 2 are introduced, the classical knapsack cover inequality given by $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \geq 2$ is weakened to become $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \geq 1$. (The addition of these variables, although not visibly affecting the inequalities (7), actually does evidently strengthen the implied inequalities (9) since these "$\leq$" inequalities will now contain unit coefficients on the left-hand side for a larger number of variables.)

Finally, let us comment on a possible strategy for generating particular SOC inequalities to delete a particular fractional solution. Such separation strategies are well known for minimal cover inequalities based on knapsack constraints, as popularized by Crowder et al. [2] in their seminal paper. In our context, we could commence with a minimal cover inequality, or even a lifted minimal cover inequality, which is generated as in Crowder et al. [2] based on a knapsack constraint of the form (1), then impose a suitable restriction (2), and further lift or strengthen the resultant inequality by commencing the procedure of Fig. 2 with a partial solution list corresponding to the associated set J for the given inequality, and terminating this process with the first resultant non-dominated SOC inequality. Indeed, applying this idea for the above example with the minimal cover inequality $x_1 + x_2 + x_3 + x_4 \geq 1$, as well as with the lifted minimal cover inequality $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \geq 2$, and commencing the procedure of Fig. 2 with the respective partial solution lists PS = $\{\underline{1}, 2, 3, 4\}$ and PS = $\{\underline{1}, 2, 3, 4, 5, 6\}$ yielded the strengthened SOC inequality $x_1 + x_2 + x_3 + x_4 \geq 2$ in both cases. In the first case, this inequality was produced in the initial loop itself, and in the second case, it was generated after invoking Routine LND 17 times, both instances requiring negligible effort. Ideas of this type, with related computational studies, will be explored in future research.

6 Facets and related lifting process

In this section, we identify conditions under which the derived SOC inequalities (7) would be facets of $X_c \equiv \text{conv}\{X\}$, the convex hull of X, and describe a sequential lifting process (see [14]) that could be used otherwise. (Also, see [17] for a polynomial-time lifting of minimal covers for GUB constrained knapsack problems into underlying facets.)

We assume throughout that the following assumption (in addition to Assumption A) holds true, where $u \geq 2$ (the case of $u = 1$ is addressed in [17]).

Assumption A′ $u \geq 2$, Assumption A holds, and also,

$$S_N(u-1) \geq a_0. \tag{25}$$

Note that, as propounded by Proposition 10 below, (25) ensures that (2) does not necessarily hold as an equality, and therefore, that X_c is full-dimensional.

Proposition 10 *The polytope X_c is full-dimensional.*

Proof We establish the result by demonstrating the existence of $n + 1$ affinely independent vectors $v_0, v_1, \ldots, v_n$ belonging to X_c. Letting e_i denote the ith unit vector in R^n, consider the following definitions of these vectors: $v_0 = \sum_{i=1}^{u-1} e_i, v_i = v_0 - e_i + [e_u + e_{u+1}]$ for $i = 1, \ldots, u-1$, and $v_i = v_0 + e_i$ for $i = u, \ldots, n$. Note that $v_0 \in X_c$ by (25), $v_i \in X_c$ for $i = 1, \ldots, u-1$ by Proposition 5, and $v_i \in X_c$ for $i = u, \ldots, n$ by Proposition 6. Moreover, the vectors $v'_i \equiv v_i - v_0$ for $i = 1, \ldots, n$ are linearly independent because $-\sum_{i=1}^{u-1} e_i\lambda_i + [e_u + e_{u+1}]\sum_{i=1}^{u-1} \lambda_i + \sum_{i=n}^{n} e_i\lambda_i = 0$ implies that $\lambda_1, \ldots, \lambda_n = 0$. □

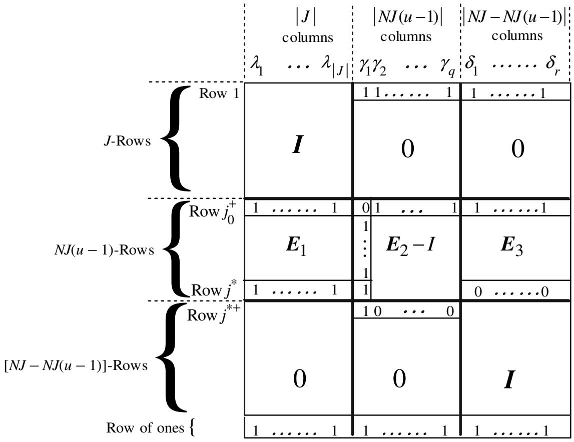
Now, let us first consider the case of $p = 1$ and suppose that we have an LND SOC inequality

$$\sum_{j \in J} x_j \geq p \equiv 1. \tag{26}$$

The following result identifies a sufficient condition under which (26) would be a facet of X_c.

Proposition 11 *Consider the SOC inequality* (26) *with $p = 1$ that is generated based on the set $J \subseteq N$, and suppose that $j^{*+} > 0$ exists. Furthermore, define*

Fig. 3 Matrix B in the proof of Proposition 11



$\Sigma = S_J(1) + S_{NJ}u - 1)$ *and suppose that*

$$\left[\sum\right] - a_{j_0^+} + a_{j^{*+}} \geq a_0 \tag{27a}$$

and that

$$\left[\sum\right] - a_j \geq a_0, \ \forall j \in NJ(u-1)/\{j_0^+\}. \tag{27b}$$

Then (26) *is a facet of* X_c.

Proof It is sufficient to identify n affinely independent points $v_1, \ldots, v_n$ in X_c at which (26) is active. Figure 3 displays a matrix B identifying such a collection $[v_1, \ldots, v_n]$ that is augmented by an additional last row having all elements equal to one. Here, the matrices E_1, E_2, and E_3 are appropriately sized matrices having all elements equal to 1 (see the corresponding rows and columns identified in Fig. 3).

First, examine the three sets of identified columns in Fig. 3, except for the last row, which represent a partition of $\{v_1, \ldots, v_n\}$. Note that all these vectors $v_1, \ldots, v_n$ satisfy (2) as well as satisfy (26) as an equality. Moreover, the first $|J|$ columns satisfy (1) because Condition (a) of Proposition 7 implies by the LND property that $a_{j_0} + S_{NJ}(u-1) \geq a_0$, so that $a_j + S_{NJ}(u-1) \geq a_0$, $\forall j \in J$. The first column in the second set satisfies (1) because of (27a), while the remaining columns in this set, as well as the columns in the third set, satisfy (1) because of (27b). Hence $\{v_1, \ldots, v_n\} \subseteq X_c$ and (26) is active at each of these points. To complete the proof, we need to show that

$$Bw = 0 \Rightarrow w \equiv 0, \qquad \text{where } w \equiv [\lambda_1, \ldots, \lambda_{|J|}, \gamma_1, \ldots, \gamma_q, \delta_1, \ldots, \delta_r]^{\mathrm{T}} \tag{28}$$

and where these components of w are associated with the columns of B as displayed in Fig. 3, with $q \equiv |NJ(u-1)|$ and $r \equiv |NJ - NJ(u-1)|$.

Accordingly, consider the system $Bw = 0$. The rows $2, \ldots, |J|$ in the J-Rows imply that

$$\lambda_2 = \cdots = \lambda_{|J|} = 0. \quad (29a)$$

The Row j_0^+ and the last row imply that

$$\gamma_1 = 0, \quad (29b)$$

which together with the $[NJ - NJ(u-1)]$-Rows yield

$$\delta_1 = \cdots = \delta_r = 0. \quad (29c)$$

Now, Row 1, and the $NJ(u-1)$-Rows excepting Row j_0^+, respectively yield, noting (29a, b, c),

$$\lambda_1 + \sum_{i=2}^{q} \gamma_i = 0$$
$$\lambda_1 + \sum_{\substack{i=2 \\ i \neq k}}^{q} \gamma_i = 0, \quad \forall k = 2, \ldots, q.$$

These two equations imply that

$$\lambda_1 = 0 \quad \text{and} \quad \gamma_2 = \cdots = \gamma_q = 0. \quad (29d)$$

Therefore, from (29a, b, c, d), we get $w \equiv 0$. □

Example 3 Consider X defined by (23a, b) of Example 2. Observe that $S_N(u-1) = 25 \geq a_0$; hence, by Proposition 10, X_c is full-dimensional. Now, let us examine the LND inequalities (24a, b, e) having $p = 1$ in light of Proposition 11 in turn below.

Case of 24(a)($x_1 + x_3 \geq 1$) Here, $\Sigma \equiv S_J(1) + S_{NJ}(u-1) = 32$, $j_0^+ = 2$, $NJ(u-1) = \{2, 4\}$, and $j^{*+} = 5$. Checking (27a, b), we see that $\Sigma - a_{j_0^+} + a_{j^{*+}} = 32 - 12 + 5 = 25 \geq a_0$, and that $\Sigma - a_4 = 32 - 7 = 25 \geq a_0$. Hence, this is a facet of X_c.

Case of 24(e)($x_1 + x_4 + x_5 + x_6$) ≥ 1 Here, $\Sigma \equiv S_J(1) + S_{NJ}(u-1) = 34$, $j_0^+ = 2$, $NJ(u-1) = \{2, 3\}$, and $j^{*+} = 7$. Again, checking (27a, b), we see that $\Sigma - a_2 + a_7 = 34 - 12 + 3 = 25 \geq a_0$, and $\Sigma - a_3 = 34 - 9 = 25 \geq a_0$. Hence, (24e) is also a facet of X_c.

Case of 24(b)($x_1 + x_2 \geq 1$) Here, $\Sigma \equiv S_J(1) + S_{NJ}(u-1) = 29$, $j_0^+ = 3$, $NJ(u-1) = \{3, 4\}$, and $j^{*+} = 5$. However, while (27a) yields $\Sigma - a_3 + a_5 = 29 - 9 + 5 = 25 \geq a_0$, (27b) yields $\Sigma - a_4 = 29 - 7 = 22 < a_0$. Hence, the sufficient condition does not hold true.

In such a case, we can perform a sequential lifting of this SOC inequality by lifting-down from a value of 1 for each $j \in NJ(u-1) = \{3, 4\}$, and lifting-up from a value of 0 for each $j \in NJ - NJ(u-1) = \{5, 6, 7, 8, 9, 10\}$ as follows (see [14] for a general discussion on such sequential liftings). Given a current valid inequality

$$\pi x \geq \pi_0, \tag{30a}$$

for lifting-down from a value of 1 with respect to some presently considered $k \in NJ(u-1)$ in a sequential process, we examine lifting (30a) to

$$\pi x \geq \pi_0 + \theta(1 - x_k), \quad \text{where } \theta = \min\{\pi x - \pi_0 : x \in X, x_k = 0\}. \tag{30b}$$

(Note that the lifted inequality is valid when $x_k = 1$ regardless of θ, given the validity of (30a), and we are interested in a value of $\theta \geq 0$.) Likewise, for lifting-up from a value of 0 with respect to some $k \in NJ - NJ(u-1)$, we lift (the current inequality) (30a) to

$$\pi x \geq \pi_0 + \theta x_k, \quad \text{where } \theta = \min\{\pi x - \pi_0 : x \in X, x_k = 1\}. \tag{30c}$$

In our example, starting with (24b) representing (30a), we get $\theta = 0$ in (30b) for all $k \in NJ(u-1)$, and also $\theta = 0$ in (30c) for $k = 5$, 6, and 7 from the set $NJ - NJ(u-1)$. However, consider x_8, where $8 \in [NJ - NJ(u-1)]$. For this, (30c) yields $\theta = \min\{x_1 + x_2 - 1 : x \in X, x_8 = 1\} = 1$ at the solution $x_1 = x_2 = x_8 = 1$, thereby producing the lifted inequality $x_1 + x_2 - x_8 \geq 1$. Likewise, sequentially, we obtain $\theta = 1$ for each of the liftings with respect to x_9 and x_{10}, producing the following strengthened valid inequality

$$x_1 + x_2 - (x_8 + x_9 + x_{10}) \geq 1. \tag{31}$$

Next, let us address the case of $p = 2$ in a valid LND SOC inequality

$$\sum_{j \in J} x_j \geq p = 2. \tag{32}$$

Similar to Proposition 11, the following result identifies a sufficient condition under which (32) would be a facet of X_c. For this case, in addition to Assumption A$'$, we assume that $u \geq 3$, else, (32) would imply that $x_j = 0$, $\forall j \in NJ$. Note also that we must have $j_0 > j[p]$, else, we could have fixed $x_j = 1$, $\forall j \in J$.

Proposition 12 *Consider the SOC inequality* (32) *with $p = 2$ that is generated based on the set $J \subseteq N$, and suppose that $j^{*+} > 0$ exists. Furthermore, define $\Sigma \equiv S_J(2) + S_{NJ}(u-2)$, denote $j[p+1]$ as the $(p+1)^{\text{st}} \equiv$ third-ordered* (*smallest*)

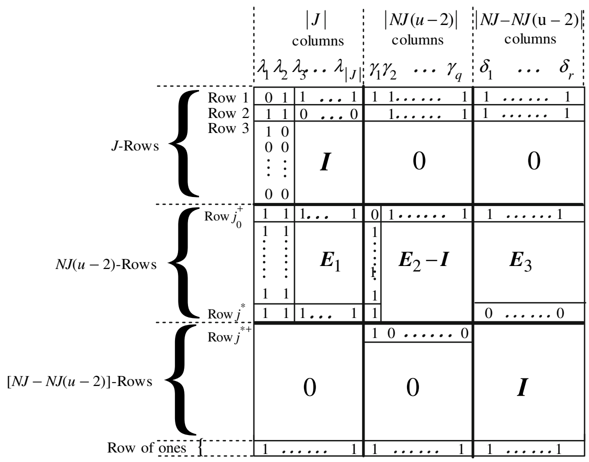


Fig. 4 Matrix B in the proof of Proposition 12

index in J, and suppose that

$$\left[\sum\right] - a_1 + a_{j[p+1]} \geq a_0 \tag{33a}$$

$$\left[\sum\right] - a_{j_0^+} + a_{j^{*+}} \geq a_0 \tag{33b}$$

and

$$\left[\sum\right] - a_j \geq a_0, \quad \forall j \in NJ(u-2)/\{j_0^+\}. \tag{33c}$$

Then (32) *is a facet of* X_c.

Proof Similar to the proof for Proposition 11, consider the matrix B displayed in Fig. 4 having n columns of the type $\begin{bmatrix} v_1, \ldots, v_n \\ 1, \ldots, 1 \end{bmatrix}$ where again, E_1, E_2, and E_3 are appropriately sized matrices having all elements equal to 1. Observe that each of the vectors $v_1, \ldots, v_n$ belongs to X_c by virtue of the following: (33a) applied to the first column; Condition (b) of the LND property of Proposition 7 applied to the columns $2, \ldots, |J|$; (33b) applied to the first column within the second set of columns, and (33c) applied to the remaining columns. Moreover, (32) is active for each of $v_1, \ldots, v_n$. Hence, to complete the proof, we need to verify that (28) holds true for the matrix B of Fig. 4.

By the first row in each set of the J-Rows and the $NJ(u-2)$-Rows, we have that

$$\lambda_1 = 0 \quad \text{and} \quad \gamma_1 = 0. \tag{34a}$$

From the rows $3, \ldots, |J|$ of the J-Rows then, we get

$$\lambda_3 = \cdots = \lambda_{|J|} = 0. \tag{34b}$$

Using $\gamma_1 = 0$ from (34a) in the third set of rows yields

$$\delta_1 = \cdots = \delta_r = 0. \tag{34c}$$

Now, the second row in the set of J-Rows, and the rows in the set of $NJ(u-2)$-Rows except for Row j_0^+, respectively yield, using (34a, b, c),

$$\lambda_2 + \sum_{j=2}^{q} \gamma_j = 0 \quad \text{and} \quad \lambda_2 + \sum_{\substack{j=2 \\ j \neq k}}^{q} \gamma_j = 0, \ \forall k = 2, \ldots, q.$$

These two equations yield $\lambda_2 = 0$ and $\gamma_j = 0$, $\forall j = 2, \ldots, q$, which together with (34a, b, c), gives $w \equiv 0$ in (28). □

Example 4 Continuing Example 3, let us now examine the LND inequalities (24c, d, f). The following table summarizes the computations in applying Proposition 12, and verifies that each of these SOC inequalities are facets of X_c.

Inequality	Σ	Index $j[p+1]$	j_0^+	j^{*+}	Left-hand side of Equation		
					33(a)	33(b)	33(c)
24(c)	32	3	4	6	28	29	N/A
24(d)	29	3	5	6	25	28	N/A
24(f)	34	4	3	8	28	27	N/A

7 Connections with surrogate constraints and higher-order cover inequalities

There is an intimate connection between surrogate constraints and valid inequalities derived from knapsack constraints. For example, it is easy to demonstrate that the classical knapsack cover inequalities can all be formed from elementary types of surrogate constraints obtained as a linear combination of the knapsack inequality (1) with a weight of 1 and subsets of the inequalities $x_j \leq 1$ (in the form $-x_j \geq -1$) with a positive weight of a_j. Then the associated knapsack cover inequality arises simply by applying the rules of Glover [4] (the same paper that introduced surrogate constraints) to identify a lower bound on the sum of the variables having positive coefficients in the surrogate constraint. In fact, as shown in Glover et al. [6], it is possible to generate valid inequalities from surrogate constraints involving linear combinations of (1) and the inequalities $x_j \leq 1$ and $x_j \geq 0$ that dominate the classical knapsack cover inequalities.

Similarly, it is possible to show that the SOC inequalities can be derived by applying the rules of Glover [4] to surrogate constraints formed using linear combinations of (1), (2), and the inequalities $x_j \leq 1$ for $j \in N$. Again, it suffices to give (1) a weight of 1, whereupon the weight of (2) (written in $\geq$ form) equals the value of one of the coefficients a_j, and finally, the weights for various subsets of the inequalities $-x_j \geq -1$, $j \in N$, are equal to the corresponding positive coefficients of the intermediate surrogate constraint obtained by combining (1) and (2). Consequently, we may also equivalently form such intermediate surrogate constraints and generate SOC inequalities by the rules for producing knapsack cover inequalities. Our results show that this produces every inequality of the form $\sum_{j \in J} x_j \geq p$ that is implied by X, and provide special dominance relationships leading to an effective method for generating all non-dominated members of such SOC inequalities. In view of these observations, it may be expected that the results in Glover et al. [6] may be applied to yield additional useful valid inequalities for X.

Finally, the derivations of the preceding sections can be extended to handle more general considerations in which the knapsack constraint (1) and the bounded sum constraint (2) are augmented by additional constraints, to give a system of the form

$$\sum_{j \in N} a_j x_j \geq a_0 \tag{35a}$$

$$\ell \leq \sum_{j \in N} x_j \leq u \tag{35b}$$

$$\ell_i \leq \sum_{j \in N_i} x_j \leq u_i, \quad \forall i \in M \equiv \{1, \ldots, m\}, \tag{35c}$$

where the sets, N_i, $i \in M$, constitute a partition of N. The inclusion of a lower bound (ℓ) in (35b) was not necessary in (2) due to reasons explained in Sect. 1, but provides greater generality when accompanied by the inequalities of (35c).

The relevance of this expanded system for 0–1 programming is illustrated by two special cases of particular interest. One is the situation where $u_i = 1$ for all $i \in M$, capturing the types of constraints found in multiple-choice 0–1 problems, which abound in practical applications. Sherali and Lee [17] characterize facets for such problems. The other case is the situation where (35c) begins as a single constraint ($m = 1$) over a specified proper subset N_1 of N. The condition that the sets N_i constitute a partition of N can be satisfied by introducing the set $N_2 \equiv N - N_1$ and adding the redundant inequality $0 \leq \sum_{j \in N_2} x_j \leq u_2$ with $u_2 \equiv |N_2|$. More pertinently, the constraint over N_1 in (35c) may be one derived as an SOC inequality (7) or (9) by the results of the preceding sections. By embedding this as indicated in (35c), the derived SOC inequality can then be exploited further relative to other knapsack constraints of the type (35a) accompanied by (35b), thereby amplifying the ability to exploit the SOC inequalities of this paper.

Such an approach has particularly useful applications in settings where knapsack constraints arise from surrogate constraints designed to capture different types of problem structure, as by generating weighted combinations of parent constraints having different forms. For example, in multi-demand multidimensional knapsack problems, which contain two classes of constraints, one consisting of $\leq$ inequalities and the other comprised of $\geq$ inequalities, where all constraints have nonnegative coefficients, it is natural to create "opposing" surrogate constraints derived from the members of these two classes.

In a sequel paper, we devise mechanisms for generating all valid inequalities of the form (7) for the system (35a, b, c) and identify dominance relationships leading to an effective method for generating non-dominated members of these cuts. To provide a foretaste of these more general results, we briefly sketch a method that applies to the simpler case where the lower bounds ℓ and ℓ_i, $\forall i \in M$, are omitted. That is, we address the system (35a, b, c) with $\ell = 0$ and $\ell_i = 0$, $\forall i \in M$.

7.1 Notation

We maintain the convention that the a_j-coefficients are indexed in nonincreasing order and, for reasons similar to those noted previously, we restrict attention to positive a_j-coefficients. (This is not an appropriate restriction for the general case where the bounds ℓ and ℓ_i, $i \in M$, are included in (35b, c).) It is also convenient to order the a_j-coefficients for each set N_i in a likewise fashion. For ease in discussion, we also make reference to linked lists that identify

$$\text{First}(i) = \text{Min}\{j \in N_i\}(= \arg\max\{a_j : j \in N_i\}) \quad \text{for each } i \in M. \tag{36}$$

Furthermore, let the linked list $\text{Next}_i(j)$, starting with $j = \text{First}(i)$, identify the indices $j \in N_i$ in the desired order by iteratively setting $j \leftarrow \text{Next}_i(j)$. By convention, the last index j of N_i is flagged by setting $\text{Next}_i(j) = 0$.

To facilitate the description of procedures that follow, we further specialize such a linking by also applying it to the two subsets J and $NJ \equiv N - J$. That is, we define

$$J\text{-First}(i) = \text{Min }\{j \in N_i \cap J\}(= \arg\max\{a_j : j \in N_i \cap J\}) \tag{37a}$$

$$NJ\text{-First}(i) = \text{Min }\{j \in N_i \cap NJ\}(= \arg\max\{a_j : j \in N_i \cap NJ\}). \tag{37b}$$

Once again, we adopt linked lists $\text{Next}_J(\cdot)$ and $\text{Next}_{NJ}(\cdot)$ to identify the successive elements of each set $N_i \cap J$ and each set $N_i \cap NJ$, respectively, in their appropriate order. If $N_i \cap J = \emptyset$ or $N_i \cap NJ = \emptyset$, we set $J\text{-First}(i) = 0$ or $NJ\text{-First}(i) = 0$, respectively.

Our goal is to identify a valid *higher-order cover* (*HOC*) inequality of the type

$$\sum_{j \in J} x_j \geq p \tag{38}$$

for any specified subset J, along with an associated value of p. Analogous to (8), the value p is essentially given by the optimal objective value of the following problem.

$$\text{Minimize}\left\{\sum_{j\in J}x_j : \sum_{j\in N}a_jx_j \geq a_0, \sum_{j\in N}x_j \leq u, \sum_{j\in N_i}x_j \leq u_i, \quad \forall i \in M, x \text{ binary}\right\}. \tag{39}$$

7.2 Algorithm for generating higher-order cover inequalities (38)

For the more general setting considered here, we are no longer able, as in Proposition 1, to specify a closed-form expression for the conditions that produce (38), but instead, require an algorithm to generate such an inequality. In essence, in lieu of solving the problem (39) directly, we examine the following feasibility problem $F(p)$, for successive values of $p = 0, 1, \ldots$

$$\textbf{F(p):}\ \text{Maximize}\left\{\sum_{j\in N}a_jx_j : \sum_{j\in N}x_j \leq u, \sum_{j\in N_i}x_j \leq u_i, \forall i \in M, \sum_{j\in J}x_j \leq p, x \text{ binary}\right\}. \tag{40}$$

To begin with, we set $p = 0$ and devise a simple scheme to solve (40). If the optimal objective value is at least a_0, then (38) is a trivial inconsequential inequality having $p \equiv 0$. Otherwise, we continue incrementing p by one successively until the objective value in (40) becomes greater than or equal to a_0, whence we will have solved (39) and thereby generated (38). Note that in this process, for any value of p, having obtained sets $J^* \subseteq J$ and $NJ^* \subseteq NJ$ that correspond to indices in J and NJ, respectively, for which $x_j = 1$ at optimality in (40), in case the optimal value in (40) is less than a_0, then the corresponding sets J^*_{new} and NJ^*_{new} for $F(p+1)$ can be obtained by updating J^* and NJ^*, noting that $J^*_{\text{new}} = J^* \cup \{j\}$, for some $j \in J - J^*$. This follows from the observation that in the process for solving (40) (from scratch), we can adopt a greedy sequential scheme in which we commence with $x = 0$, and then at each step, we set $x_j = 1$ corresponding to the largest a_j-coefficient from among all admissible x-variables that can be switched from 0 to 1 subject to the constraints in (40).

This algorithmic scheme is formalized below, where we adopt the following additional notation. For $j \in N$, we let $IN(j)$ identify the index i such that $j \in N_i$. Furthermore, corresponding to the current solution x, we let $s_i \equiv \sum_{j\in N_i} x_j$, and $\Sigma \equiv \sum_{j\in N} a_jx_j$. Part A of the algorithm solves Problem (40) for $p = 0$ and generates the corresponding set NJ^* (with $J^* = \emptyset$) by sequentially selecting the smallest possible indices $j \in NJ$ (i.e., having the largest possible a_j-values) while ensuring that no more than u total indices are selected from NJ, and no more than u_i indices are selected from each $N_i \cap NJ$, $i \in M$. Part B then modifies J^* and NJ^* while sequentially incrementing p in (40) by one in each loop. It does so by identifying (if they exist), the best (smallest) next index $j(i)$ to possibly select from each $J \cap N_i$, $i \in M$, to include into J^*; the worst currently selected index $j[i]$ (smallest a_j-value) from $N_i \cap NJ^*$, $\forall i \in M$, and the worst currently selected index j^* from NJ^* if u total indices are selected (else $j^* \equiv 0$). It also identifies two sets I_1 and I_2, where I_1 contains $i \in M$ for which u_i indices are already selected, but both $j(i)$ and $j[i]$ exist, and I_2 contains $i \in M$ for which fewer than u_i indices are currently selected and $j(i)$ exists. For each $i \in I_1$, it next computes the best advantage $\alpha(i) \equiv a_{j(i)} - a_{j[i]}$ of swapping by selecting $j(i)$ in place of $j[i]$. Similarly, for each $i \in I_2$, the procedure computes the best advantage $\alpha(i) = a_{j(i)} - a_{j^*}$ of selecting $j(i)$ and dropping j^* (if $j^* \neq 0$). The actual swap made is the one that yields the highest advantage $\alpha(i)$ from $i \in I_1 \cup I_2$, and the corresponding selected sets J^* and NJ^* are updated, along with the counters s_i, $i \in M$, and the objective value Σ of (40). Given feasibility of (39) (which the procedure automatically detects in this process), the algorithm loops until $\Sigma \geq a_0$ is obtained.

Higher-order cover inequality algorithm

Begin with $\Sigma = 0$ and $s_i = 0$, $\forall i \in M$. Also, set $J^* = NJ^* = \emptyset$, and $p = 0$.
Part A: Generate *NJ**
A0. Let $j(i) = NJ\text{-First}(i)$, $\forall i \in M$, and let $M(NJ) = \{i \in M : j(i) \neq 0\}$.
A1. If $M(NJ) = \emptyset$, or if $|NJ^*| = u$, proceed to Part B. Otherwise, let $j^* = \min\{j(i) : i \in M(NJ)\}$. Then set $NJ^* \leftarrow NJ^* + \{j^*\}$ and $\Sigma \leftarrow \Sigma + a_{j^*}$.
A2. If $\Sigma \geq a_0$, the cut (38) is degenerate with $p = 0$ and the algorithm stops.
Otherwise, let $i = IN(j^*)$, and set $s_i \leftarrow s_i + 1$ and $j(i) = \text{Next}_{NJ}(j^*)$. If either $s_i = u_i$ or $j(i) = 0$, set $M(NJ) \leftarrow M(NJ) - \{i\}$. Return to Step A1.
Part B: Introduce J^* and modify *NJ**
B0. Redefine $j(i)$ to refer to the set J instead of NJ by setting $j(i) = J\text{-First}(i)$, $\forall i \in M$, and let $M(J) = \{i \in M : j(i) \neq 0\}$.
B1. (a) For each $i \in M(J)$ such that $s_i = u_i$, let $j[i] = \arg\min\{a_j : j \in N_i \cap NJ^*\}$. If $N_i \cap NJ^* = \emptyset$ set $j[i] = 0$.
(b) Let $j^* = \arg\min\{a_j : j \in NJ^*\}$. If $NJ^* = \emptyset$ or if $|NJ^* + J^*| < u$, set $j^* = 0$.
B2. Define $I_1 = \{i \in M : s_i = u_i, j(i) \neq 0, \text{ and } j[i] \neq 0\}$ and $I_2 = \{i \in M : s_i < u_i, j(i) \neq 0\}$
If $I_1 \cup I_2 = \emptyset$, stop; Problem (39) is infeasible.
B3. For $i \in I_1$, let $\alpha(i) = a_{j(i)} - a_{j[i]}$. For $i \in I_2$, let $\alpha(i) = a_{j(i)} - a_{j^*}$ if $j^* > 0$ and $\alpha(i) = a_{j(i)}$, otherwise. Then let

$$i^* = \arg\max\{\alpha(i) : i \in I_1 \cup I_2\}.$$

If $\alpha(i^*) \leq 0$ stop; Problem (39) is infeasible.
B4. Let $J^* \leftarrow J^* + \{j(i^*)\}$. If $i^* \in I_1$, set $NJ^* \leftarrow NJ^* - j[i^*]$. If $i^* \in I_2$, set $s_{i^*} \leftarrow s_{i^*} + 1$ and if $j^* > 0$, set $NJ^* \leftarrow NJ^* - \{j^*\}$ and $s_h \leftarrow s_h - 1$ for $h = IN(j^*)$ (possibly, $h = i^*$).
B5. Set $\Sigma \leftarrow \Sigma + \alpha(i^*)$ and $p \leftarrow p + 1$. If $\Sigma \geq a_0$, then the cut (38) is obtained and the method stops. Otherwise, if $p = u$, stop; Problem (39) is infeasible.

Finally, if the foregoing conditions do not hold, set $j(i^*) \leftarrow \text{Next}_J[j(i^*)]$, and if $j(i^*) = 0$, set $M(J) \leftarrow M(J) - \{i^*\}$. Return to Step B1.

The inclusion of the lower bounds ℓ and $\ell_i, i \in M$, to give the more general system (35a, b, c) requires a somewhat more complex process to generate the appropriate valid inequalities. The theorems applicable to this system, as well as to its special case sketched above, yield a set of dominance relationships that are appreciably different and invite different methods of exploitation than those for the SOC inequalities. We therefore relegate the consideration of such generalized HOC inequalities and their generation and dominance results, as well as associated issues of solving suitable separation problems to generate SOC or HOC inequalities (see Example 2 for some relevant comments), along with computational experimental studies, for follow-on research.

Acknowledgements This research has been supported in part by the *National Science Foundation* under Grant Number DMI-0552676. The authors also thank two anonymous referees for the comments that led to an improvement of this paper.

References

1. Balas, E.: Facets of the Knapsack Polytope. Math. Program. **8**, 146–164 (1975)
2. Crowder, H.P., Johnson, E.L., Padberg, M.W.: Solving large-scale zero-one linear programming problems. Oper. Res. **31**, 803–834 (1983)
3. Geoffrion, A.M.: An improved implicit enumeration approach for integer programming. Oper. Res. **17**, 437–454 (1969)
4. Glover, F.: A multiphase-dual algorithm for the zero-one integer programming problem. Oper. Res. **13**, 879–919 (1965)
5. Glover, F.: Flows in arborescences. Manage. Sci. **17**, 568–586 (1971)
6. Glover, F., Sherali, H.D., Lee, Y.: Generating cuts from surrogate constraint analysis for zero-one and multiple choice programming. Comput. Optim. Appl. **8**(2), 152–172 (1997)
7. Gu, Z., Nemhauser, G.L., Savelsbergh, M.W.P.: Cover inequalities for 0–1 linear programs: computation. INFORMS J. Comput. **10**, 427–437 (1998)
8. Hammer, P.L., Johnson, E.L., Peled, U.N.: Facets of regular 0–1 polytopes. Math. Program. **8**, 179–206 (1975)
9. Hanafi, S.: Contribution à la résolution de problèmes duaux de grande taille en optimisation combinatoire. Ph.D. Thesis, University of Valenciennes, France (1993)
10. Hanafi, S., Glover, F.: Exploiting nested inequalities and surrogate constraints. Research Report, University of Valenciennes, France, and University of Colorado, Boulder, CO, USA (2005)
11. Hooker, J.N.: Logic-based methods for optimization. In: Borning, A. (ed.) Principles and Practice of Constraint Programming. Lecture Notes in Computer Science, Vol. 874, pp. 336–349 (1994)
12. Hooker, J.N., Osorio, M.A.: Mixed logical/linear programming. Discrete Appl. Math. **96–97**, 395–442 (1999)
13. Nemhauser, G.L., Savelsbergh, M.W.P., Sigismondi, G.S.: MINTO, a mixed INTeger optimizer. Oper. Res. Lett. **15**, 47–58 (1994)
14. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. 2nd edn. Wiley, New York (1999)
15. Osorio, M.A., Glover, F., Hammer, P.: Cutting and surrogate constraint analysis for improved multidimensional knapsack solutions. Ann. Oper. Res. **117**, 71–93 (2002)
16. Savelsbergh, M.W.P.: Preprocessing and probing for mixed integer programming problems. ORSA J. Comput. **6**, 445–454 (1994)

17. Sherali, H.D., Lee, Y.: Sequential and simultaneous liftings of minimal cover inequalities for generalized upper bound constrained knapsack polytopes. SIAM J. Discrete Math. **8**(1), 133–153 (1995)
18. Spielberg, K., Guignard, M.: A sequential (Quasi) hot start method for BB (0, 1) mixed integer programming. Mathematical Programming Symposium, Atlanta, GA (2000)
20. Vasquez, M., Vimont, Y.: Improved results on the 0–1 multidimensional knapsack problem. Eur. J. Oper. Res. **165**, 70–81 (2005)
20. Wolsey, L.A.: Faces for a linear inequality in 0–1variables. Math. Program. **8**, 165–178 (1975)