Examples of new facets for the precedence constrained knapsack problem

Christopher Fricke

1 Introduction

We consider the polyhedral structure of the *precedence constrained knapsack (PCK) problem*, also known as the *partially ordered knapsack problem*. A set of items $\mathcal{N}$ is given, along with a partial order, or set of precedence relationships, on the items, denoted by $\mathcal{S} \subseteq \mathcal{N} \times \mathcal{N}$. A precedence relationship $(i, j) \in \mathcal{S}$ exists if item i can be placed in the knapsack only if item j is in the knapsack. Each item $i \in \mathcal{N}$ has a value $c_i \in \mathbb{Z}$ and a weight $a_i \in \mathbb{Z}^+$, and the knapsack has a capacity $b \in \mathbb{Z}^+$. The PCK problem seeks a maximum value subset of $\mathcal{N}$ whose total weight does not exceed the knapsack capacity, and that also satisfies the precedence relationships.

The precedence constraints can be represented by the directed acyclic graph $G = (\mathcal{N}, \mathcal{S})$, where the node set is the set of all items $\mathcal{N}$, and each precedence constraint in $\mathcal{S}$ is represented by a directed arc. Note that the precedence constraints are transitive, so without loss of generality we assume that $\mathcal{S}$ does not contain any redundant relationships, that is, $\mathcal{S}$ is the set of all immediate predecessor arcs. If G contains a cycle, all nodes within the cycle must either all be included in, or all be excluded from, the knapsack. Hence the cycle can be contracted into a single node, with cumulative value and weight coefficients, and the resulting directed graph is acyclic.

An integer programming formulation of the PCK problem is as follows. Let

$$x_i = \begin{cases} 1, & \text{if item } i \text{ is included in the knapsack} \\ 0, & \text{otherwise} \end{cases} \qquad \text{for all } i \in \mathcal{N}.$$

Then the PCK problem may be written as:

$$\text{(PCKP)} \qquad \begin{aligned} \max \quad & \sum_{i \in \mathcal{N}} c_i \, x_i & & (1) \\ \text{s.t.} \quad & \sum_{i \in \mathcal{N}} a_i \, x_i \le b & & (2) \\ & x_i \le x_j & \text{for all } (i,j) \in \mathcal{S} \quad & (3) \\ & x_i \in \{0,1\} & \text{for all } i \in \mathcal{N}. \quad & (4) \end{aligned}$$

The PCK problem appears in a wide range of applications. These include investment problems (Ibarra and Kim [3]), tool management problems (Stecke and Kim [8]), strip mining (Johnson and Niemi [4]) and local access telecommunication network design (Shaw et. al. [7]). In these cases the PCK problem has generally been solved using dynamic programming

Christopher Fricke was the winner of the T.M. Cherry Prize for best student talk at ANZIAM 2006, held in Mansfield, February 2006.

algorithms, when the underlying precedence graph has a special structure such as a tree, or heuristics.

Johnson and Niemi [4] showed that the PCK problem is NP-complete. The polyhedral structure of the problem was first investigated by Boyd [2], who extended the concept of a cover inequality for the standard 0-1 knapsack polyhedron to the PCK polyhedron. Further investigation of the PCK polyhedron is presented by both Park and Park [6] and van de Leensel et. al. [9], where lifting orders and general sequential lifting procedures are derived to lift valid knapsack cover-based inequalities from lower dimensional polyhedrons into facets of the PCK polyhedron.

In [1], we determine a new class of facet-defining inequalities for the convex hull $conv(P)$ of the PCK feasible set P defined by (2)-(4). Unlike previous work [2, 6, 9], we do not take knapsack covers as our starting point, but instead investigate clique inequalities derived from a graph representing pairwise conflict relationships between variables. A brief summary of our approach is presented in this article. In addition, in [1] we present a comparison with previous polyhedral approaches to the PCK problem based on knapsack cover-like inequalities, which demonstrates that our clique-based approach can generate facet-defining inequalities that cannot be found through the cover-based approach of previous authors. We also note that a relaxation of the conditions under which previous results were obtained could have allowed additional facet-defining inequalities to be determined, and provide a thorough classification of PCK covers and cliques, showing the relationships between them.

2 Fundamental Terminology and Conflict Graphs

For each $(i, j) \in \mathcal{S}$, item i is an *immediate predecessor* of item j and item j is an *immediate successor* of item i. Let S_i be the set of immediate predecessors of item i, that is let $S_i = \{j \in \mathcal{N} : (i, j) \in \mathcal{S}\}$. It follows that the set of all precedence relationships $\mathcal{A}$ is the transitive closure of $\mathcal{S}$, and $(i, j) \in \mathcal{A}$ if and only if there exists a path from node i to node j in the directed acyclic graph $G = (\mathcal{N}, \mathcal{S})$. Let A_i be the *minimal set of items*, including item i, that must be included in the knapsack for item i to be included, that is $A_i = \{j \in \mathcal{N} : (i, j) \in \mathcal{A}\} \cup \{i\}$. Note that inclusion in the set A_i is also transitive, so if $j \in A_k$ and $k \in A_i$ then $j \in A_i$. Consider also a set of items $B \subseteq \mathcal{N}$. Let $A(B) = \cup_{i \in B} A_i$ be the union of the A_i sets for the items in the set B. Then $A(B)$ gives the minimal set of items that must be included for all items in the set B to be included in the knapsack. An example of a PCK problem is presented in Figure 1. For illustrative purposes note that in this example $A_1 = \{1, 5, 11\}$ and $A(\{1, 3\}) = \{1, 3, 5, 7, 10, 11, 12, 13\}$.

We now combine the precedence sets with the knapsack constraint (2) to determine the minimum capacity required to include each item in the knapsack. Let $H(B) = \sum_{j \in A(B)} a_j$ be the total capacity required to include the items in the set B in the knapsack. It follows that $H(\{i\}) = \sum_{j \in A_i} a_j$ is the capacity required to include item i in the knapsack. For ease of notation let $H_i = H(\{i\})$. In the example presented in Figure 1 we have that $H_1 = \sum_{j \in A_1} a_j = 3 + 5 + 4 = 12$, and $H(\{1, 3\}) = \sum_{j \in A(\{1,3\})} a_j = 25$. Furthermore, we assume that for every individual item, there exists a feasible solution in which it is included in the knapsack. Otherwise, the item can be deleted from the problem instance.

Assumption 1 Each item in the set $\mathcal{N}$ can be included in the knapsack, that is $H_i \leq b$ for all $i \in \mathcal{N}$.

In order to identify potential facet-defining inequalities for $conv(P)$, we require the following definition of a conflict graph for the instance of the PCK problem under consideration.

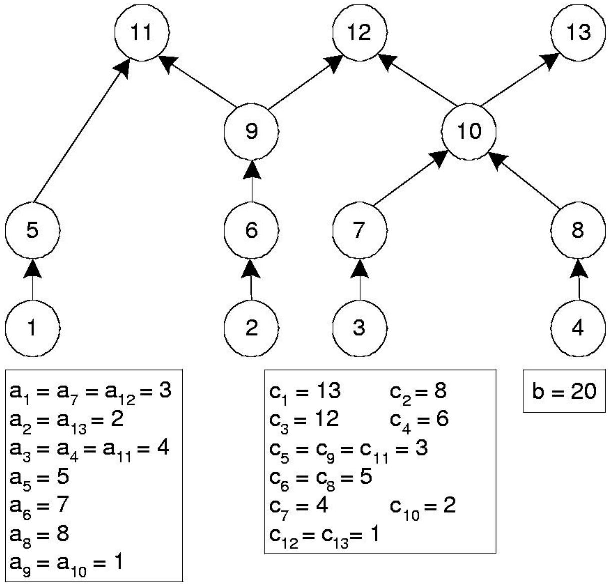

Figure 1. PCK Problem Example.

Definition 1 A **conflict graph** $CG = (\mathcal{N}, E)$ contains the edge $\{i, j\} \in E$ if and only if the pair of items i and j **cannot** be included in the knapsack together, that is if and only if $H(\{i, j\}) > b$.

As noted above, in the example in Figure 1 we have that $H(\{1, 3\}) = 25$, and since $b = 20$ it follows that the edge $\{1, 3\}$ is included in the conflict graph for this example. A **clique** $\mathcal{C} \subseteq \mathcal{N}$ in the conflict graph CG is a set of nodes such that every pair of nodes in $\mathcal{C}$ is joined by an edge. Hence each pair of items in $\mathcal{C}$ cannot be included in the knapsack simultaneously, and it follows that at most one item in $\mathcal{C}$ can be included in the knapsack. A **maximal clique** is a clique that cannot be enlarged by adding any additional node. We also require the concept of an intersection set, defined as follows.

Definition 2 Let $\mathcal{C} \subseteq \mathcal{N}$ be a clique in the conflict graph CG. Let $\mathcal{P}(\mathcal{C})$ be the set of items in the intersection of the entire precedence sets of all the items in the clique $\mathcal{C}$, that is $\mathcal{P}(\mathcal{C}) = \{k : k \in \cap_{j \in \mathcal{C}} A_j\}$.

The intersection set $\mathcal{P}(\mathcal{C})$ may or may not be empty. We consider these two cases separately. For example in Figure 1 we have that $\mathcal{P}(\{1, 3\}) = \emptyset$ and $\mathcal{P}(\{1, 2\}) = \{11\}$. The terminology introduced in this section is required, along with a number of additional properties of the precedence sets and conflict graphs, to determine new facet-defining inequalities for $conv(P)$. Note that it follows directly from Assumption 1 that the PCK polyhedron is full-dimensional, which allows us to apply a simplified version of Theorem 3.6 from Nemhauser and Wolsey [5] to determine necessary and sufficient conditions for our clique-based inequalities to be facet-defining for $conv(P)$.

3 Clique-Based Facets for the PCK Polyhedron

Suppose we have constructed a conflict graph $CG = (\mathcal{N}, E)$ according to Definition 1. The following result is obvious.

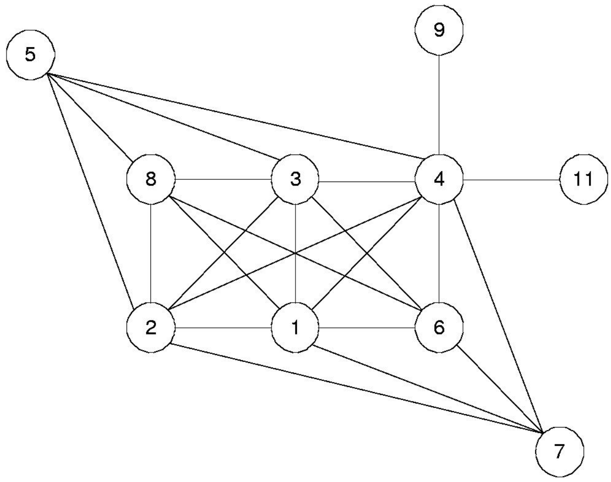

Figure 2. Conflict Graph for PCK Problem Example.

Lemma 1 *Let $\mathcal{C} \subseteq \mathcal{N}$ be a clique in the conflict graph CG. Then the clique inequality*

$$\sum_{j \in \mathcal{C}} x_j \leq 1 \tag{5}$$

is valid for P.

In the case where $\mathcal{P}(\mathcal{C}) = \emptyset$, we have determined necessary and sufficient conditions under which (5) is facet-defining for $conv(P)$, where $\mathcal{C}$ is a maximal clique in the conflict graph CG. We have also developed a straightforward procedure that, given any maximal clique $\mathcal{C} \subseteq \mathcal{N}$ with $\mathcal{P}(\mathcal{C}) = \emptyset$, can generate a maximal clique satisfying these conditions.

In the case where $\mathcal{P}(\mathcal{C}) \neq \emptyset$, we have determined necessary and sufficient conditions under which a strengthened form of (5) is facet-defining for $conv(P)$, where $\mathcal{C}$ is a maximal clique in the conflict graph CG.

Lemma 2 *Let $\mathcal{C} \subseteq \mathcal{N}$ be a clique in the conflict graph CG with $\mathcal{P}(\mathcal{C}) \neq \emptyset$, and let $i \in \mathcal{P}(\mathcal{C})$. Then the inequality*

$$\sum_{j \in \mathcal{C}} x_j \leq x_i \tag{6}$$

is valid for P.

Proof. Let $\mathcal{C} \subseteq \mathcal{N}$ be a clique in the conflict graph CG with $\mathcal{P}(\mathcal{C}) \neq \emptyset$, and let $i \in \mathcal{P}(\mathcal{C})$. Hence $i \in A_j$ for each $j \in \mathcal{C}$. It follows from the transitivity of the precedence constraints (3) that for all $j \in \mathcal{C}$, $x_j \leq x_i$. Hence if $x_i = 0$ it must be that $x_j = 0$ for all $j \in \mathcal{C}$, and (6) holds. Otherwise, $x_i = 1$ and (6) is equivalent to the clique inequality (5), which is valid for P by Lemma 1. So we have that the strengthened clique inequality (6) is valid for P when $\mathcal{P}(\mathcal{C}) \neq \emptyset$. □

Again in this case we have also developed a straightforward procedure that, given any maximal clique $\mathcal{C} \subseteq \mathcal{N}$ with $\mathcal{P}(\mathcal{C}) \neq \emptyset$, can generate a maximal clique satisfying these conditions. Further detail can be found in [1].

4 Application of Clique-Based Inequalities to a PCK Problem Example

We now demonstrate the relative strengthening effect of our clique-based facet-defining inequalities for $conv(P)$ on the linear programming relaxation of the PCK problem. Consider again the PCK problem example given in Figure 1. Applying the approach for deriving facets of $conv(P)$ from clique inequalities to this example, we obtain the conflict graph given in Figure 2. There are ten maximal cliques $\mathcal{C} \subseteq \mathcal{N}$ in this conflict graph, as shown in Table 1, all of which are such that $\mathcal{P}(\mathcal{C}) = \emptyset$, and six of which do not satisfy our necessary and sufficient conditions. However, a maximal clique that does satisfy our necessary and sufficient conditions can be derived from these maximal cliques in all instances, by the application of a simple reduction procedure described in [1]. In this example, this procedure does not generate any new facets beyond the four already found. These four clique-based facet-defining inequalities are shown in Table 1.

Maximal Clique	Corresponding facet-defining clique inequality
$\{1,2,3,4\}$	Not facet-defining
$\{1,2,3,8\}$	Not facet-defining
$\{1,3,4,6\}$	Not facet-defining
$\{1,3,6,8\}$	$x_1 + x_3 + x_6 + x_8 \leq 1$
$\{1,2,4,7\}$	Not facet-defining
$\{1,4,6,7\}$	$x_1 + x_4 + x_6 + x_7 \leq 1$
$\{2,3,4,5\}$	Not facet-defining
$\{2,3,5,8\}$	$x_2 + x_3 + x_5 + x_8 \leq 1$
$\{4,9\}$	Not facet-defining
$\{4,11\}$	$x_4 + x_{11} \leq 1$

Table 1. Maximal Cliques for PCK Problem Example

A comparison of the linear programming (LP) relaxations for this PCK problem example is presented in Table 2. The two cases tested are (i) the standard integer programming formulation (PCKP), and (ii) this formulation with the addition of the facet-defining clique inequalities given in Table 1. It is evident from Table 2 that the addition of the facet-defining clique inequalities to (PCKP) results in a reduction in root node gap, defined by $\left(\frac{\text{LP relaxation - optimal integer solution}}{\text{optimal integer solution}}\right) \times 100\%$. In fact, when the facet-defining clique inequalities are added in this example, the optimal integer solution may be found simply by solving the linear programming relaxation. These results indicate that the addition of facet-defining clique-based inequalities for the PCK problem is beneficial in certain instances.

Formulation	LP relaxation	IP value	Root Node Gap (%)
(PCKP) only	32.31	26.00	24.26
(PCKP) and facet-defining clique inequalities	26.00	26.00	0.00

Table 2. Summary of Results for PCK Problem Example

5 Conclusions and Future Work

We have derived a new approach for determining facets of the PCK polyhedron based on clique inequalities. The necessary and sufficient conditions that have been derived in [1] can be checked to determine whether clique inequalities determined from the conflict graph are facet-defining whenever the problem instance contains pairs of items that cannot be included in the knapsack together. A procedure to generate a facet-defining clique inequality from any maximal clique in the conflict graph has also been developed in [1]. We also show computationally that the addition of facet-defining clique-based inequalities for the PCK problem is beneficial in certain instances.

Our current focus is on the computational application of the results presented in this article. Particular areas of interest include the *a priori* addition of facet-defining clique inequalities to (PCKP), and the solution of the separation problem for finding clique inequalities violated by a fractional solution of the LP relaxation of (PCKP). In both cases the enumeration of maximal cliques in the conflict graph CG is required. This work is in progress.

References

[1] N. Boland, C. Fricke, G. Froyland and R. Sotirov, *Clique-based facets for the precedence constrained knapsack problem*, submitted to Mathematical Programming (2005).

[2] E.A. Boyd, *Polyhedral results for the precedence-constrained knapsack problem*, Discrete Appl. Math. **41** (1993), 185–201.

[3] O.H. Ibarra and C.E. Kim, *Approximation algorithms for certain scheduling problems*, Math. Oper. Res. **4** (1978), 197–204.

[4] D.S. Johnson, and K.A. Niemi, *On knapsacks, partitions, and a new dynamic programming technique for trees*, Math. Oper. Res. **8** (1983), 1–14.

[5] G.L. Nemhauser, and L.A. Wolsey, *Integer and Combinatorial Optimization*, (Wiley-Interscience New York 1988).

[6] K. Park and S. Park, *Lifting cover inequalities for the precedence-constrained knapsack problem*, Discrete Appl. Math. **72** (1997), 219–241.

[7] D.X. Shaw, G. Cho and H. Chang, *A depth-first dynamic programming procedure for the extended tree knapsack problem in local access network design*, Telecommunication Systems **7** (1997), 29–43.

[8] K.E. Stecke and I. Kim, *A study of part type selection approaches for short-term production planning*, Int. J. Flexible Manufacturing Systems **1** (1988), 7–29.

[9] R.L.M.J. van de Leensel, C.P.M. van Hoesel and J.J. van de Klundert, *Lifting valid inequalities for the precedence constrained knapsack problem*, Math. Program. **86** (1999), 161–185.

Department of Mathematics and Statistics, University of Melbourne, VIC 3010
E-mail: C.Fricke@ms.unimelb.edu.au

Received 19 May 2006, accepted for publication 16 June 2006.